

# TUNING FAIRNESS BY BALANCING TARGET LABELS

Thomas Kehrenberg<sup>1</sup>, Zexun Chen<sup>1,2</sup>, and Novi Quadrianto<sup>1</sup>

<sup>1</sup>Predictive Analytics Lab (PAL), University of Sussex, Brighton, UK

<sup>2</sup>BioComplex Laboratory, University of Exeter, Exeter, UK

## ABSTRACT

The issue of fairness in machine learning models has recently attracted a lot of attention as ensuring it will ensure continued confidence of the general public in the deployment of machine learning systems. We focus on mitigating the harm incurred by a biased machine learning system that offers better outputs (e.g. loans, job interviews) for certain groups than for others. We show that bias in the output can naturally be controlled in probabilistic models by introducing a latent target output. This formulation has several advantages: first, it is a unified framework for several notions of group fairness such as Demographic Parity and Equality of Opportunity; second, it is expressed as a marginalisation instead of a constrained problem; and third, it allows the encoding of our knowledge of what unbiased outputs should be. Practically, the second allows us to avoid unstable constrained optimisation procedures and to reuse off-the-shelf toolboxes. The latter translates to the ability to control the level of fairness by directly varying fairness target rates. In contrast, existing approaches rely on intermediate, arguably unintuitive, control parameters such as covariance thresholds.

## 1 INTRODUCTION

Algorithmic assessment methods are used for predicting human outcomes in areas such as financial services, recruitment, crime and justice, and local government. This contributes, in theory, to a world with decreasing human biases. To achieve this, however, we need fair machine learning models that take biased datasets, but output non-discriminatory decisions to people with differing protected attributes such as gender and marital status. Datasets can be biased because of, for example, sampling bias, subjective bias of individuals, and institutionalised biases (Olteanu et al., 2019; Tolan, 2019). Uncontrolled bias in the data can translate into bias in machine learning models.

There is no single accepted definition of algorithmic fairness for automated decision-making but several have been proposed. One definition is referred to as *statistical* or *demographic parity*. Given a binary protected attribute (e.g. married/unmarried) and a binary decision (e.g. yes/no to getting a loan), demographic parity requires equal positive rates (PR) across the two sensitive groups (married and *unmarried* individuals should be equally likely to receive a loan). Another fairness criterion, *equalised odds* (Hardt et al., 2016), takes into account the binary decision, and instead of equal PR requires equal true positive rates (TPR) and false positive rates (FPR). This criterion is intended to be more compatible with the goal of building accurate predictors or achieving high utility (Hardt et al., 2016). We discuss the suitability of the different fairness criteria in the discussion section at the end of the paper.

There are many existing models for enforcing demographic parity and equalised odds (Creager et al., 2019; Agarwal et al., 2018; Calders et al., 2009; Kamishima et al., 2012; Zafar et al., 2017a;b). However, these existing approaches to balancing accuracy and fairness rely on intermediate, unintuitive control parameters such as allowable constraint violation  $\epsilon$  (e.g. 0.01) in Agarwal et al. (2018), or a covariance threshold  $c$  (e.g. 0 that is controlled by another parameters  $\tau$  and  $\mu - 0.005$  and  $1.2$  – to trade off this threshold and accuracy) in Zafar et al. (2017a). This is related to the fact that many of these approaches embed fairness criteria as *constraints* in the optimisation procedure (Donini et al., 2018; Quadrianto & Sharmanska, 2017; Zafar et al., 2017a;b).

In contrast, we provide a probabilistic classification framework with bias controlling mechanisms that can be tuned based on positive rates (PR), an intuitive parameter. Thus, giving humans the control to set the rate of positive predictions (e.g. a PR of 0.6). Our framework is based on the concept of a *balanced dataset* and introduces latent target labels, which, instead of the provided labels, are now the training label of our classifier. We prove bounds on how far the target labels diverge from the dataset labels. We instantiate our approach with a parametric logistic regression classifier and a Bayesian non-parametric Gaussian process classifier (GPC). As our formulation is not expressed as a constrained problem, we can draw upon advancements in automated variational inference (Bonilla et al., 2016; Gardner et al., 2018; Krauth et al., 2016) for learning the fair model, and for handling large amounts of data.

The method presented in this paper is closely related to a number of previous works, e.g. Kamiran & Calders (2012); Calders & Verwer (2010). Proper comparison with them requires knowledge of our approach. We will thus explain our approach in the subsequent sections, and defer detailed comparisons to Section 4 (Related Work).

## 2 TARGET LABELS FOR TUNING GROUP FAIRNESS

We will start by describing several notions of group fairness. For each individual, we have a vector of non-sensitive attributes  $x \in \mathcal{X}$ , a class label  $y \in \mathcal{Y}$ , and a sensitive attribute  $s \in \mathcal{S}$  (e.g. racial origin or gender). We focus on the case where  $s$  and  $y$  are binary. We assume that a positive label  $y = 1$  corresponds to a positive outcome for an individual – for example, being accepted for a loan. *Group fairness* balances a certain condition between groups of individuals with different sensitive attribute,  $s$  versus  $s'$ . The term  $\hat{y}$  below is the prediction of a machine learning model that, in most works, uses only non-sensitive attributes  $x$ . Several group fairness criteria have been proposed (e.g. Zafar et al. (2017a); Chouldechova (2017); Hardt et al. (2016)):

equality of positive rate (Demographic Parity):

$$\Pr(\hat{y} = 1|s) = \Pr(\hat{y} = 1|s') \quad (1)$$

equality of accuracy:

$$\Pr(\hat{y} = y|s) = \Pr(\hat{y} = y|s') \quad (2)$$

equality of true positive rate (Equality of Opportunity):

$$\Pr(\hat{y} = 1|s, y = 1) = \Pr(\hat{y} = 1|s', y = 1) . \quad (3)$$

*Equalised odds* criterion corresponds to Equality of Opportunity (3) plus equality of false positive rate.

The Bayes-optimal classifier only satisfies these criteria if the training data itself satisfies them. That is, in order for the Bayes-optimal classifier to satisfy *demographic parity*, the following must hold:  $\mathbb{P}(y = 1|s) = \mathbb{P}(y = 1|s')$ , where  $y$  is the training label. We call a dataset for which  $\mathbb{P}(y, s) = \mathbb{P}(y)\mathbb{P}(s)$  holds, a *balanced dataset*. Given a balanced dataset, a Bayes-optimal classifier learns to satisfy demographic parity and an approximately Bayes-optimal classifier should learn to satisfy it at least approximately. Here, we motivated the importance of balanced datasets via the demographic parity criterion, but it is also important for *equality of opportunity* which we discuss in Section 2.1.

In general, however, our given dataset is likely to be imbalanced. There are two common solutions to this problem: either pre-process or massage the dataset to make it balanced, or constrain the classifier to give fair predictions despite it having been trained on an unbalanced dataset. Our approach takes parts from both solutions.

An imbalanced dataset can be turned into a balanced dataset by either changing the class labels  $y$  or the sensitive attributes  $s$ . In the use cases that we are interested in,  $s$  is considered an integral part of the input, representing trustworthy information and thus should not be changed.  $y$ , conversely, is often not completely trustworthy; it is not an integral part of the sample but merely an observed outcome. In a hiring dataset, for instance,  $y$  might represent the hiring decision, which can be biased, and not the relevant question of whether someone makes a good employee.

Thus, we introduce new *target labels*  $\bar{y}$  such that the dataset is balanced:  $\mathbb{P}(\bar{y}, s) = \mathbb{P}(\bar{y})\mathbb{P}(s)$ . The idea is that these target labels still contain as much information as possible about the task, while also

forming a balanced dataset. This introduces the concept of the accuracy-fairness trade-off: in order to be completely accurate with respect to the original (not completely trustworthy) class labels  $y$ , we would require  $\bar{y} = y$ , but then, the fairness constraints would not be satisfied.

Let  $\eta_s(x) = \mathbb{P}(y = 1|x, s)$  denote the distribution of  $y$  in the data. The target distribution  $\bar{\eta}_s(x) = \mathbb{P}(\bar{y} = 1|x, s)$  is then given by

$$\bar{\eta}_s(x) = (\mathbb{P}(\bar{y} = 1|y = 1, s) + \mathbb{P}(\bar{y} = 0|y = 0, s) - 1)\eta_s(x) + 1 - \mathbb{P}(\bar{y} = 0|y = 0, s) \quad (4)$$

due to the marginalisation rules of probabilities. The conditional probability  $\mathbb{P}(\bar{y}|y, s)$  indicates with which probability we want to keep the class label. This probability could in principle depend on  $x$  which would enable the realisation of individual fairness. The dependence on  $x$  has to be prior knowledge as it cannot be learned from the data. This prior knowledge can encode the semantics that “similar individuals should be treated similarly” (Dwork et al., 2012), or that “less qualified individuals should not be preferentially favoured over more qualified individuals” (Joseph et al., 2016). Existing proposals for guaranteeing individual fairness require strong assumptions, such as the availability of an agreed-upon similarity metric, or knowledge of the underlying data generating process. In contrast, in group fairness, we partition individuals into protected groups based on some sensitive attribute  $s$  and ask that some statistics of a classifier be approximately equalised across those groups (see (1)–(3)). In this case,  $\mathbb{P}(\bar{y}|y, s)$  does not depend on  $x$ .

Returning to equation 4, we can simplify it with

$$m_s := \mathbb{P}(\bar{y} = 1|y = 1, s) + \mathbb{P}(\bar{y} = 0|y = 0, s) - 1 \quad (5)$$

$$b_s := 1 - \mathbb{P}(\bar{y} = 0|y = 0, s), \quad (6)$$

arriving at  $\bar{\eta}_s(x) = m_s \cdot \eta_s(x) + b_s$ .  $m_s$  and  $b_s$  are chosen such that  $\mathbb{P}(\bar{y}, s) = \mathbb{P}(\bar{y})\mathbb{P}(s)$ . This can be interpreted as shifting the decision boundary depending on  $s$  so that the new distribution is balanced.

As there is some freedom in choosing  $m_s$  and  $b_s$ , it is important to consider what the effect of different values is. The following theorem provides this (the proof can be found in the Supplementary Material):

**Theorem 1.** *The probability that  $y$  and  $\bar{y}$  disagree ( $y \neq \bar{y}$ ) for any input  $x$  in the dataset is given by:*

$$\mathbb{P}(y \neq \bar{y}|s) = \mathbb{P}\left(|\eta(x, s) - \frac{1}{2}| < t_s\right) \quad (7)$$

where

$$t_s = \left| \frac{m_s + 2b_s - 1}{2m_s} \right|. \quad (8)$$

Thus, if the threshold  $t_s$  is small, then only if there are inputs very close to the decision boundary ( $\eta_s(x)$  close to  $\frac{1}{2}$ ) would we have  $\bar{y} \neq y$ .  $t_s$  determines the accuracy penalty that we have to accept in order to gain fairness. The value of  $t_s$  can be taken into account when choosing  $m_s$  and  $b_s$  (see Section 3). If  $\eta_s$  satisfies the Tsybakov condition (Tsybakov et al., 2004), then we can give an upper bound for the probability.

**Definition 1.** A distribution  $\eta$  satisfies the Tsybakov condition if there exist  $C > 0$ ,  $\lambda > 0$  and  $t_0 \in (0, \frac{1}{2}]$  such that for all  $t \leq t_0$ ,

$$\mathbb{P}\left(|\eta(x) - \frac{1}{2}| < t\right) \leq Ct^\lambda. \quad (9)$$

This condition bounds the region close to the decision boundary. It is a property of the dataset.

**Corollary 1.1.** *If  $\eta(x, s) = \mathbb{P}(y = 1|x, s)$  satisfies the Tsybakov condition in  $x$ , with constants  $C$  and  $\lambda$ , then the probability that  $y$  and  $\bar{y}$  disagree ( $y \neq \bar{y}$ ) for any input  $x$  in the dataset is bounded by:*

$$\mathbb{P}(y \neq \bar{y}|s) < C \left| \frac{m_s + 2b_s - 1}{2m_s} \right|^\lambda. \quad (10)$$

Section 3 discusses how to choose the parameters for  $\bar{\eta}$  in order to make it balanced.

## 2.1 EQUALITY OF OPPORTUNITY

In contrast to demographic parity, equality of opportunity (just as equality of accuracy) is satisfied by a perfect classifier. Imperfect classifiers, however, do not by default satisfy it: the true positive rate (TPR) is different for different subgroups. The reason for this is that while the classifier is optimised to have a high TPR overall, it is not optimised to have the same TPR in the subgroups.

The overall TPR is a weighted sum of the TPRs in the subgroups:

$$TPR = \mathbb{P}(s = 0|y = 1) \cdot TPR_{s=0} + \mathbb{P}(s = 1|y = 1) \cdot TPR_{s=1}. \quad (11)$$

In datasets where the positive label  $y = 1$  is heavily skewed toward one of the groups (say, group  $s = 1$ ; meaning that  $\mathbb{P}(s = 1|y = 1)$  is high and  $\mathbb{P}(s = 0|y = 1)$  is low), overall TPR might be maximised by setting the decision boundary such that nearly all samples in  $s = 0$  are classified as  $y = 0$ , while for  $s = 1$  a high TPR is achieved. The low TPR for  $s = 0$  is in this case weighted down and only weakly impacts the overall TPR. For  $s = 0$ , the resulting classifier uses  $s$  as a shorthand for  $y$ , mostly ignoring the other features. This problem usually persists even when  $s$  is removed from the input features because  $s$  is implicit in the other features.

A *balanced* dataset helps with this issue because in such datasets,  $s$  is not a useful proxy for the balanced label  $\bar{y}$  (because we have  $\mathbb{P}(\bar{y}, s) = \mathbb{P}(\bar{y})\mathbb{P}(s)$ ) and  $s$  cannot be used as a shorthand. Assuming the dataset is balanced in  $s$  ( $\mathbb{P}(s = 0) = \mathbb{P}(s = 1)$ ), for such datasets  $\mathbb{P}(s = 0|y = 1) = \mathbb{P}(s = 1|y = 1)$  holds and the two terms in equation 11 have equal weight.

Here as well there is an accuracy-fairness trade-off: assuming the unconstrained model is as accurate as its model complexity allows, adding additional constraints like equality of opportunity can only make the accuracy worse.

## 2.2 CONCRETE ALGORITHM

For training, we are only given the unbalanced distribution  $\eta_s(x)$  and not the target distribution  $\bar{\eta}_s(x)$ . However,  $\bar{\eta}_s(x)$  is needed in order to train a fair classifier. One approach is to explicitly change the labels  $y$  in the dataset, in order to construct  $\bar{\eta}_s(x)$ . We discuss this approach and its drawback in the related work section (Section 4).

We present a novel approach which only implicitly constructs the balanced dataset. This framework can be used with any likelihood-based model, such as Logistic Regression and Gaussian Process models. The relation presented in equation 4 allows us to formulate a likelihood that targets  $\bar{\eta}_s(x)$  while only having access to the imbalanced labels  $y$ . As we only have access to  $y$ ,  $\mathbb{P}(y|x, s, \theta)$  is the likelihood to optimise. It represents the probability that  $y$  is the imbalanced label, given the input  $x$ , the sensitive attribute  $s$  that available in the training set and the model parameters  $\theta$  for a model that is targeting  $\bar{y}$ . Thus, we get

$$\mathbb{P}(y = 1|x, s, \theta) = \sum_{\bar{y} \in \{0,1\}} \mathbb{P}(y = 1, \bar{y}|x, s, \theta) = \sum_{\bar{y} \in \{0,1\}} \mathbb{P}(y = 1|\bar{y}, x, s, \theta) \mathbb{P}(\bar{y}|x, s, \theta). \quad (12)$$

As we are only considering group fairness, we have  $\mathbb{P}(y = 1|\bar{y}, x, s, \theta) = \mathbb{P}(y = 1|\bar{y}, s)$ .

Let  $f_\theta(x, y')$  be the likelihood function of a given model, where  $f$  gives the likelihood of the label  $y'$  given the input  $x$  and the model parameters  $\theta$ . As we do not want to make use of  $s$  at test time,  $f$  does not explicitly depend on  $s$ . The likelihood with respect to  $\bar{y}$  is then given by  $f$ :  $\mathbb{P}(\bar{y}|x, s, \theta) = f_\theta(x, \bar{y})$ ; and thus, does not depend on  $s$ . The latter is important in order to avoid *direct discrimination* (Barocas & Selbst, 2016). With these simplifications, the expression for the likelihood becomes

$$\mathbb{P}(y = 1|x, s, \theta) = \sum_{\bar{y} \in \{0,1\}} \mathbb{P}(y = 1|\bar{y}, s) \mathbb{P}(\bar{y}|x, \theta). \quad (13)$$

The conditional probabilities,  $\mathbb{P}(y|\bar{y}, s)$ , are closely related to the conditional probabilities in equation 4 and play a similar role of “transition probabilities”. Section 3 explains how to choose these transition probabilities in order to arrive at a balanced dataset. For a binary sensitive attribute  $s$  (and binary label  $y$ ), there are 4 transition probabilities (see Algorithm 1 where  $d_{\bar{y}=i}^{s=j} := \mathbb{P}(y = 1|\bar{y} = i, s = j)$ ):

$$\mathbb{P}(y = 1|\bar{y} = 0, s = 0), \quad \mathbb{P}(y = 1|\bar{y} = 1, s = 0) \quad (14)$$

$$\mathbb{P}(y = 1|\bar{y} = 0, s = 1), \quad \mathbb{P}(y = 1|\bar{y} = 1, s = 1). \quad (15)$$

---

**Algorithm 1** Fair learning with target labels  $\bar{y}$ 

---

**Input:** Training set  $\mathcal{D} = \{(x_i, y_i, s_i)\}_{i=1}^N$ , transition probabilities  $d_{\bar{y}=0}^{s=0}, d_{\bar{y}=1}^{s=0}, d_{\bar{y}=0}^{s=1}, d_{\bar{y}=1}^{s=1}$

**Output:** fair model parameters  $\theta$

```
1: Initialise  $\theta$  (randomly)
2: for all  $x_i, y_i, s_i$  do
3:    $\mathbb{P}_{\bar{y}=1} \leftarrow \bar{\eta}(x_i, \theta)$  (e.g.  $\text{logistic}(\langle x, \theta \rangle)$ )
4:    $\mathbb{P}_{\bar{y}=0} \leftarrow 1 - \mathbb{P}_{\bar{y}=1}$ 
5:   if  $s_i = 0$  then
6:      $\mathbb{P}_{y=1} \leftarrow d_{\bar{y}=0}^{s=0} \cdot \mathbb{P}_{\bar{y}=0} + d_{\bar{y}=1}^{s=0} \cdot \mathbb{P}_{\bar{y}=1}$ 
7:   else
8:      $\mathbb{P}_{y=1} \leftarrow d_{\bar{y}=0}^{s=1} \cdot \mathbb{P}_{\bar{y}=0} + d_{\bar{y}=1}^{s=1} \cdot \mathbb{P}_{\bar{y}=1}$ 
9:   end if
10:   $\ell \leftarrow y_i \cdot \mathbb{P}_{y=1} + (1 - y_i) \cdot (1 - \mathbb{P}_{y=1})$ 
11:  update  $\theta$  to maximise likelihood  $\ell$ 
12: end for
```

---

A perhaps useful interpretation of equation 13 is that, even though we don't have access to  $\bar{y}$  directly, we can still compute the expectation value over the possible values of  $\bar{y}$ .

The above derivation applies to binary classification but can easily be extended to the multi-class case.

### 3 TRANSITION PROBABILITIES FOR A BALANCED DATASET

This section focuses on how to set values of the transition probabilities in order to arrive at balanced datasets.

#### 3.1 MEANING OF THE PARAMETERS

Before we consider concrete values, we give some intuition for the transition probabilities. Let  $s = 0$  refer to the protected group. For this group, we want to make more positive predictions than the training labels indicate. Variable  $\bar{y}$  is supposed to be our target proxy label. Thus, in order to make more positive predictions, some of the  $y = 0$  labels should be associated with  $\bar{y} = 1$ . However, we do not know which. So, if our model predicts  $\bar{y} = 1$  (high  $\mathbb{P}(\bar{y} = 1|x, \theta)$ ) while the training label is  $y = 0$ , then we allow for the possibility that this is actually correct. That is,  $\mathbb{P}(y = 0|\bar{y} = 1, s = 0)$  is not 0. If we choose, for example,  $\mathbb{P}(y = 0|\bar{y} = 1, s = 0) = 0.3$  then that means that 30% of positive target labels  $\bar{y} = 1$  may correspond to negative training labels  $y = 0$ . This way we can have more  $\bar{y} = 1$  than  $y = 1$ , overall. On the other hand, predicting  $\bar{y} = 0$  when  $y = 1$  holds, will always be deemed incorrect:  $\mathbb{P}(y = 1|\bar{y} = 0, s = 0) = 0$ ; this is because we do not want any additional negative labels.

For the non-protected group  $s = 1$ , we have the exact opposite situation. If anything, we have too many positive labels. So, if our model predicts  $\bar{y} = 0$  (high  $\mathbb{P}(\bar{y} = 0|x, \theta)$ ) while the training label is  $y = 1$ , then we should again allow for the possibility that this is actually correct. That is,  $\mathbb{P}(y = 1|\bar{y} = 0, s = 1)$  should not be 0. On the other hand,  $\mathbb{P}(y = 0|\bar{y} = 1, s = 1)$  should be 0 because we do not want additional positive labels for  $s = 1$ . It could also be that the number of positive labels is exactly as it should be, in which case we can just set  $y = \bar{y}$  for all data points with  $s = 1$ .

#### 3.2 CHOICE OF PARAMETERS

A balanced dataset is characterised by an independence of the label  $\bar{y}$  and the sensitive attribute  $s$ . Given that we have complete control over the *transition probabilities*, we can ensure this independence by requiring  $\mathbb{P}(\bar{y} = 1|s = 0) = \mathbb{P}(\bar{y} = 1|s = 1)$ . Our constraint is then that both of these probabilities are equal to the same value, which we will call the target rate  $PR_t$  ("PR" as *positive rate*):

$$\mathbb{P}(\bar{y} = 1|s = 0) \stackrel{!}{=} PR_t \quad \text{and} \quad \mathbb{P}(\bar{y} = 1|s = 1) \stackrel{!}{=} PR_t. \quad (16)$$

This leads us to the following constraints for  $s' \in \{0, 1\}$ :

$$PR_t = \mathbb{P}(\bar{y} = 1 | s = s') = \sum_y \mathbb{P}(\bar{y} = 1 | y, s = s') \mathbb{P}(y | s = s'). \quad (17)$$

We call  $\mathbb{P}(y = 1 | s = j)$  the base rate  $PR_b^j$  which we estimate from the training set:

$$\mathbb{P}(y = 1 | s = i) = \frac{\text{number of points with } y = 1 \text{ in group } i}{\text{number of points in group } i}.$$

Expanding the sum, we get

$$PR_t = \mathbb{P}(\bar{y} = 1 | y = 0, s = s') \cdot (1 - PR_b^1) + \mathbb{P}(\bar{y} = 1 | y = 1, s = s') \cdot PR_b^1. \quad (18)$$

This is a system of linear equations consisting of two equations (one for each value of  $s'$ ) and four free variables:  $\mathbb{P}(\bar{y} = 1 | y, s)$  with  $y, s \in \{0, 1\}$ . The two unconstrained degrees of freedom determine how strongly the accuracy will be affected by the fairness constraint. If we set  $\mathbb{P}(\bar{y} = 1 | y = 1, s)$  to 0.5, then this expresses the fact that a train label  $y$  of 1 only implies a target label  $\bar{y}$  of 1 in 50% of the cases. In order to minimise the effect on accuracy, we make  $\mathbb{P}(\bar{y} = 1 | y = 1, s)$  as high as possible and  $\mathbb{P}(\bar{y} = 1 | y = 0, s)$ , conversely, as low as possible. However, the lowest and highest possible values are not always 0 and 1 respectively. To see this, we solve for  $\mathbb{P}(\bar{y} = 1 | y = 0, s = j)$  in equation 18:

$$\mathbb{P}(\bar{y} = 1 | y = 0, s = j) = \frac{PR_b^j}{1 - PR_b^j} \left( \frac{PR_t}{PR_b^j} - \mathbb{P}(\bar{y} = 1 | y = 1, s = j) \right). \quad (19)$$

If  $PR_t/PR_b^j$  were greater than 1, then setting  $\mathbb{P}(\bar{y} = 1 | y = 0, s = j)$  to 0 would imply a  $\mathbb{P}(\bar{y} = 1 | y = 1, s = j)$  value greater than 1. A visualisation that shows why this happens can be found in the Supplementary Material. We thus arrive at the following definitions:

$$\mathbb{P}(\bar{y} = 1 | y = 1, s = j) = \begin{cases} 1 & \text{if } PR_t > PR_b^j \\ \frac{PR_t}{PR_b^j} & \text{otherwise.} \end{cases} \quad (20)$$

$$\mathbb{P}(\bar{y} = 1 | y = 0, s = j) = \begin{cases} \frac{PR_t - PR_b^j}{1 - PR_b^j} & \text{if } PR_t > PR_b^j \\ 0 & \text{otherwise.} \end{cases} \quad (21)$$

Algorithm 2 shows pseudocode of the procedure, including the computation of the allowed minimal and maximal value.

Once all these probabilities have been found, the transition probabilities needed for equation 13 are fully determined by applying Bayes' rule:

$$\mathbb{P}(y = 1 | \bar{y}, s) = \frac{\mathbb{P}(\bar{y} | y = 1, s) \mathbb{P}(y = 1 | s)}{\mathbb{P}(\bar{y} | s)}. \quad (22)$$

### 3.2.1 CHOOSING A TARGET RATE.

As shown, there is a remaining degree of freedom when targeting a balanced dataset: the target rate  $PR_t := \mathbb{P}(\bar{y} = 1)$ . This is true for both fairness criteria that we are targeting. The choice of targeting rate affects how much  $\eta$  and  $\bar{\eta}$  differ as implied by Theorem 1 ( $PR_t$  affects  $m_s$  and  $b_s$ ).  $\bar{\eta}$  should remain close to  $\eta$  as  $\bar{\eta}$  only represents an auxiliary distribution that does not have meaning on its own. The threshold  $t_s$  in Theorem 1 (equation 8) gives an indication of how close the distributions are. With the definitions in equation 20 and equation 21, we can express  $t_s$  in terms of the target rate and the base rate:

$$t_s = \begin{cases} \frac{1}{2} \frac{PR_b^s - PR_t}{PR_t} & \text{if } PR_t > PR_b^j \\ \frac{1}{2} \frac{PR_t - PR_b^s}{1 - PR_t} & \text{otherwise.} \end{cases} \quad (23)$$

This shows that  $t_s$  is smallest when  $PR_b^s$  and  $PR_t$  are closest. However, as  $PR_b^s$  has different values for different  $s$ , we cannot set  $PR_b^s = PR_t$  for all  $s$ . In order to keep both  $t_{s=0}$  and  $t_{s=1}$  small, it follows from equation 23 that  $PR_t$  should at least be between  $PR_b^0$  and  $PR_b^1$ . A more precise statement can be made when we explicitly want to minimise the sum  $t_{s=0} + t_{s=1}$ : assuming

$PR_b^0 < PR_t < PR_b^1$  and  $PR_b^1 < \frac{1}{2}$ , the optimal choice for  $PR_t$  is  $PR_b^1$  (see Supplementary Material for details). We call this choice  $PR_t^{max}$ . For  $PR_b^0 > \frac{1}{2}$ , analogous statements can be made, but this is of less interest as this case does not appear in our experiments.

The previous statements about  $t_s$  do not directly translate into observable quantities like accuracy if the Tsybakov condition is not satisfied, and even if it is satisfied, the usefulness depends on the constants  $C$  and  $\lambda$ . Conversely, the following theorem makes *generally* applicable statement about the accuracy that can be achieved. Before we get to the theorem, we introduce some notation. We are given a dataset  $\mathcal{D} = \{(x_i, y_i)\}_i$ , where the  $x_i$  are vectors of features and the  $y_i$  the corresponding labels. We refer to the tuples  $(x, y)$  as the *samples* of the dataset. The number of samples is  $N = |\mathcal{D}|$ .

We assume binary labels ( $y \in \{0, 1\}$ ) and thus can form the (disjoint) subsets  $\mathcal{Y}^0$  and  $\mathcal{Y}^1$  with

$$\mathcal{Y}^j = \{(x, y) \in \mathcal{D} | y = j\} \quad \text{with } j \in \{0, 1\}. \quad (24)$$

Furthermore, we associate each sample with a classification  $\hat{y} \in \{0, 1\}$ . The task of making the classification  $\hat{y} = 0$  or  $\hat{y} = 1$  can be understood as sorting each sample from  $\mathcal{D}$  into one of two sets:  $\mathcal{C}^0$  and  $\mathcal{C}^1$ , such that  $\mathcal{C}^0 \cup \mathcal{C}^1 = \mathcal{D}$  and  $\mathcal{C}^0 \cap \mathcal{C}^1 = \emptyset$ .

We refer to the set  $\mathcal{A} = (\mathcal{C}^0 \cap \mathcal{Y}^0) \cup (\mathcal{C}^1 \cap \mathcal{Y}^1)$  as the set of correct (or accurate) predictions. The *accuracy* is given by  $acc = N^{-1} \cdot |\mathcal{A}|$ .

**Definition 2.**

$$r_a := \frac{|\mathcal{Y}^1|}{|\mathcal{D}|} = \frac{|\mathcal{Y}^1|}{N} \quad (25)$$

is called the *base acceptance rate* of the dataset  $\mathcal{D}$ .

**Definition 3.**

$$\hat{r}_a = \frac{|\mathcal{C}^1|}{|\mathcal{D}|} = \frac{|\mathcal{C}^1|}{N} \quad (26)$$

is called the *predictive acceptance rate* of the predictions.

**Theorem 2.** For a dataset with the base rate  $r_a$  and corresponding predictions with a predictive acceptance rate of  $\hat{r}_a$ , the accuracy is limited by

$$acc \leq 1 - |\hat{r}_a - r_a|. \quad (27)$$

**Corollary 2.1.** Given a dataset that consists of two subsets  $\mathcal{S}_0$  and  $\mathcal{S}_1$  ( $\mathcal{D} = \mathcal{S}_0 \cup \mathcal{S}_1$ ) where  $p$  is the ratio of  $|\mathcal{S}_0|$  to  $|\mathcal{D}|$  and given corresponding acceptance rates  $r_a^0$  and  $r_a^1$  and predictions with target rates  $\hat{r}_a^0$  and  $\hat{r}_a^1$ , the accuracy is limited by

$$acc \leq 1 - p \cdot |\hat{r}_a^0 - r_a^0| - (1 - p) \cdot |\hat{r}_a^1 - r_a^1|. \quad (28)$$

The proofs are fairly straightforward and can be found in the Supplementary Material.

Corollary 2.1 implies that in the common case where group  $s = 0$  is disadvantaged ( $r_a^0 < r_a^1$ ) and also underrepresented ( $p < \frac{1}{2}$ ), the highest accuracy under demographic parity can be achieved at  $PR_t = r_a^1$  with

$$acc \leq 1 - p \cdot (r_a^1 - r_a^0). \quad (29)$$

However, this means willingly accepting a lower accuracy in the (smaller) subset  $\mathcal{S}_0$  that is compensated by a very good accuracy in the (larger) subset  $\mathcal{S}_1$ . A decidedly “fairer” approach is to aim for the same accuracy in both subsets. This is achieved by using the average of the base acceptance rates for the target rate. As we balance the test set in our experiments, this kind of sacrificing of one demographic group does not work there. We compare the two choices ( $PR_t^{max}$  and  $PR_t^{avg}$ ) in Section 5.

### 3.3 CONDITIONALLY BALANCED DATASET

There is a fairness definition related to demographic parity which allows conditioning on “legitimate” risk factors  $\ell$  when considering how equal the demographic groups are treated Corbett-Davies et al. (2017). This cleanly translates into balanced datasets which are balanced conditioned on  $\ell$ :

$$\mathbb{P}(\bar{y} = 1 | \ell = \ell', s = 0) \stackrel{!}{=} \mathbb{P}(\bar{y} = 1 | \ell = \ell', s = 1). \quad (30)$$

We can interpret this as splitting the data into partitions based on the value of  $\ell$ , where the goal is to have all these partitions be balanced. This can easily be achieved by our method by setting a  $PR_t(\ell)$  for each value of  $\ell$  and computing the transition probabilities for each sample depending on  $\ell$ .

---

**Algorithm 2** Targeting a balanced dataset

---

**Input:** target rate  $PR_t$ , biased acceptance rate  $PR_b^i$

**Output:** transition probabilities  $d_{\bar{y}=j}^{s=i}$

```
1: if  $PR_t > PR_b^i$  then
2:    $\mathbb{P}(\bar{y} = 1 | y = 1, s = i) \leftarrow 1$ 
3: else
4:    $\mathbb{P}(\bar{y} = 1 | y = 1, s = i) \leftarrow \frac{PR_t}{PR_b^i}$ 
5: end if
6: if  $j=0$  then
7:    $\mathbb{P}(\bar{y} = 0 | y = 1, s = i) \leftarrow 1 - \mathbb{P}(\bar{y} = 1 | y = 1, s = i)$ 
8:    $d_{\bar{y}=0}^{s=i} \leftarrow \frac{\mathbb{P}(\bar{y}=0 | y=1, s=i) \cdot PR_b^i}{1 - PR_t}$ 
9: else if  $j=1$  then
10:   $d_{\bar{y}=1}^{s=i} \leftarrow \frac{\mathbb{P}(\bar{y}=1 | y=1, s=i) \cdot PR_b^i}{PR_t}$ 
11: end if
```

---

## 4 RELATED WORK

There are several ways to enforce fairness in machine learning models: as a pre-processing step (Kamiran & Calders, 2012; Louizos et al., 2016; Lum & Johndrow, 2016; Zemel et al., 2013; Chiappa, 2019; Quadrianto et al., 2019), as a post-processing step (Feldman et al., 2015; Hardt et al., 2016), or as a constraint during the learning phase (Calders et al., 2009; Zafar et al., 2017a;b; Donini et al., 2018; Dimitrakakis et al., 2019). Our method enforces fairness during the learning phase (an in-processing approach) but, unlike other approaches, we do not cast fair-learning as a *constrained* optimisation problem. Constrained optimisation requires a customised procedure. In Goh et al. (2016), Zafar et al. (2017a), and Zafar et al. (2017b), suitable majorisation-minimisation/convex-concave procedures (Lanckriet & Sriperumbudur, 2009) were derived. Furthermore, such constrained optimisation approaches may lead to more unstable training, and often yield classifiers with both worse accuracy and more unfair (Cotter et al., 2018).

The approaches most closely related to ours were given by Kamiran & Calders (2012) who present four pre-processing methods: *Suppression*, *Massaging the dataset*, *Reweighting*, and *Sampling*. In our comparison we focus on methods 2, 3 and 4, because the first one simply removes sensitive attributes and those features that are highly correlated with them. All the methods given by Kamiran & Calders (2012) aim only at enforcing demographic parity.

The massaging approach uses a classifier to first rank all samples according to their probability of having a positive label ( $y = 1$ ) and then flips the labels that are closest to the decision boundary such that the data then satisfies demographic parity. This *pre-processing* approach is similar in spirit to our *in-processing* method but differs in the execution. In our method (Section 3.2), “ranking” and classification happen in one step and labels are not explicitly flipped but assigned probabilities of being flipped.

The reweighting method reweights samples based on whether they belong to an over-represented or under-represented demographic group. The sampling approach is based on the same idea but works by resampling instead of reweighting. Both reweighting and sampling aim to effectively construct a balanced dataset, without affecting the labels. This is in contrast to our method which treats the class labels as potentially untrustworthy and allows defying them.

One approach in Calders & Verwer (2010) is also worth mentioning. It is based on a *generative* Naïve Bayes model in which a latent variable  $L$  is introduced which is reminiscent to our target label  $\bar{y}$ . We provide a *discriminative* version of this approach. In discriminative models, parameters capture the conditional relationship of an output given an input, while in generative models, the joint distribution of input-output is parameterised. With this conditional relationship formulation ( $\mathbb{P}(y|\bar{y}, s) = \mathbb{P}(\bar{y}|y, s)\mathbb{P}(y|s)/\mathbb{P}(\bar{y}|s)$ ), we can have detailed control in setting the target rate. Calders & Verwer (2010) focuses only on the demographic parity fairness metric.



## 5 EXPERIMENTS

We compare the performance of our target-label model with other existing models based on two real-world datasets. These datasets have been previously considered in the fairness-aware machine learning literature.

### 5.1 IMPLEMENTATION

The proposed method is compatible with any likelihood-based algorithm. We consider both a nonparametric and a parametric model. The nonparametric model is a Gaussian process model, and Logistic regression is the parametric counterpart. Since our fairness approach is not being framed as a constrained optimisation problem, we can reuse off-the-shelf toolboxes including the GPyTorch library by Gardner et al. (2018) for Gaussian process models. This library incorporates recent advances in scalable variational inference including variational *inducing inputs* and likelihood ratio/REINFORCE estimators. The variational posterior can be derived from the likelihood and the prior. We need just need to modify the likelihood to take into account the target labels (Algorithm 1).

### 5.2 DATA

We run experiments on two real-world datasets. The first dataset is the **Adult Income** dataset (Dheeru & Karra Taniskidou, 2017). It contains 33,561 data points with census information from US citizens. The labels indicate whether the individual earns more ( $y = 1$ ) or less ( $y = 0$ ) than \$50,000 per year. We use the dataset with either *race* or *gender* as the sensitive attribute. The input dimension, excluding the sensitive attributes, is 12 in the raw data; the categorical features are then one-hot encoded. For the experiments, we removed 2,399 instances with missing data and used only the training data, which we split randomly for each trial run. The second dataset is the **ProPublica recidivism** dataset. It contains data from 6,167 individuals that were arrested. The data was collected when investigating the COMPAS risk assessment tool (Angwin et al., 2016). The task is to predict whether the person was rearrested within two years ( $y = 1$  if they were rearrested,  $y = 0$  otherwise). We again use the dataset with either *race* or *gender* as the sensitive attributes.

### 5.3 BALANCING THE TEST SET

Any fairness method that is targeting demographic parity, treats the training set as defective in one way: the acceptance rates are not equal in the training set and this needs to be corrected. As such, it does not make sense to evaluate these methods on a dataset that is equally defective. Predicting at equal acceptance rates is the correct result and the test set should reflect this.

In order to generate a test set which has the property of equal acceptance rates, we subsample the given, imbalanced, test set. For evaluating demographic parity, we discard datapoints from the imbalanced test set such that the resulting subset satisfies  $\mathbb{P}(s = j | y = i) = \frac{1}{2}$  for all  $i$  and  $j$ . This balances the set in terms of  $s$  and ensures  $\mathbb{P}(y, s) = \mathbb{P}(y)\mathbb{P}(s)$ , but does not force the acceptance rate to be  $\frac{1}{2}$ , which in the case of the Adult dataset would be a severe change as the acceptance rate is naturally quite low there. Using the described method ensures that the minimal amount of data is discarded for the Adult dataset. We have empirically observed that all fairness algorithms benefit from this balancing of the test set.

The situation is different for equality of opportunity. A perfect classifier automatically satisfies equality of opportunity on *any dataset*. Thus, an algorithm aiming for this fairness constraint should not treat the dataset as defective. Consequently, for evaluating equality of opportunity we perform no balancing of the test set.

### 5.4 METHOD

We evaluate two versions of our target label model<sup>1</sup>: *FairGP*, which is based on Gaussian Process models, and *FairLR*, which is based on logistic regression. We also train baseline models that do not take fairness into account.

---

<sup>1</sup>The code can be found on GitHub: <https://github.com/predictive-analytics-lab/ethicml-models/tree/master/implementations/fairgp>.

Table 1: Accuracy and fairness (with respect to *demographic parity*) for various methods on the balanced test set of the Adult dataset. Fairness is defined as  $PR_{s=0}/PR_{s=1}$  (a completely fair model would achieve a value of 1.0). Left: using **race** as the sensitive attribute. Right: using **gender** as the sensitive attribute. The mean and std of 10 repeated experiments.

Algorithm	Fair $\rightarrow$ 1.0 $\leftarrow$	Accuracy $\uparrow$	Fair $\rightarrow$ 1.0 $\leftarrow$	Accuracy $\uparrow$
GP	$0.80 \pm 0.07$	$0.888 \pm 0.007$	$0.54 \pm 0.05$	$0.900 \pm 0.006$
LR	$0.83 \pm 0.06$	$0.884 \pm 0.007$	$0.52 \pm 0.03$	$0.898 \pm 0.003$
SVM	$0.89 \pm 0.06$	$0.899 \pm 0.004$	$0.49 \pm 0.05$	$0.913 \pm 0.004$
FairGP (ours)	$0.86 \pm 0.07$	$0.888 \pm 0.006$	$0.87 \pm 0.09$	$0.902 \pm 0.007$
FairLR (ours)	$0.90 \pm 0.06$	$0.874 \pm 0.009$	$0.93 \pm 0.04$	$0.886 \pm 0.012$
ZafarAccuracy	$0.67 \pm 0.17$	$0.808 \pm 0.016$	$0.77 \pm 0.08$	$0.853 \pm 0.017$
ZafarFairness	$0.81 \pm 0.06$	$0.879 \pm 0.009$	$0.74 \pm 0.11$	$0.897 \pm 0.004$
Kamiran & Calders (2012)	$0.87 \pm 0.07$	$0.882 \pm 0.007$	$0.96 \pm 0.03$	$0.900 \pm 0.004$
Agarwal et al. (2018)	$0.86 \pm 0.08$	$0.883 \pm 0.008$	$0.65 \pm 0.04$	$0.900 \pm 0.004$

In both *FairGP* and *FairLR*, our approach is implemented by modifying the likelihood function. First, the unmodified likelihood is computed (corresponding to  $\mathbb{P}(\hat{y} = 1|x, \theta)$ ) and then a linear transformation (dependent on  $s$ ) is applied as given by equation 13. No additional ranking of the samples is needed, because the unmodified likelihood already supplies ranking information.

The fair GP models and the baseline GP model are all based on variational inference and use the same settings. During training, each batch is equivalent to the whole dataset. The number of inducing inputs is 500 on the ProPublica dataset and 2500 on the Adult dataset which corresponds to approximately  $1/8$  of the number of training points for each dataset. We use a squared-exponential (SE) kernel with automatic relevance determination (ARD) and the probit function as the likelihood function. We optimise the hyper-parameters and the variational parameters using the Adam method (Kingma & Ba, 2015) with the default parameters. We use the full covariance matrix for the Gaussian variational distribution. The logistic regression is trained with RAdam (Liu et al., 2019) and uses L2 regularisation. For the regularisation coefficient, we conducted a hyper-parameter search over 10 folds of the data. For each fold, we picked the hyper-parameter which achieved the best fairness among those 5 with the best accuracy scores. We then averaged over the 10 hyper-parameter values chosen in this way and then used this average for all runs to obtain our final results.

In addition to the GP and LR baselines, we compare our proposed model with the following methods: Support Vector Machine (*SVM*), *Kamiran & Calders* (Kamiran & Calders, 2012) (“reweighing” method), *Agarwal et al.* (Agarwal et al., 2018) (using logistic regression as the classifier) and several methods given by Zafar et al. (Zafar et al., 2017b;a), which include maximising accuracy under demographic parity fairness constraints (*ZafarFairness*), maximising demographic parity fairness under accuracy constraints (*ZafarAccuracy*), and removing disparate mistreatment by constraining the false negative rate (*ZafarEqOpp*). Every method is evaluated over 10 repeats that each have different splits of the training and test set.

## 5.5 RESULTS FOR DEMOGRAPHIC PARITY ON ADULT DATASET

Following Zafar et al. (2017b) we evaluate demographic parity on the Adult dataset. Table 1 shows the accuracy and fairness for several algorithms. In the table, and in the following, we use  $PR_{s=i}$  to denote the observed rate of positive predictions per demographic group  $\mathbb{P}(\hat{y} = 1|s = i)$ . Thus,  $PR_{s=0}/PR_{s=1}$  is a measure for demographic parity, where a completely fair model would attain a value of 1.0. This measure for demographic parity is also called “disparate impact” (see e.g. Feldman et al. (2015); Zafar et al. (2017a)). As the results in Table 1 show, FairGP and FairLR are clearly fairer than the baseline GP and LR. We use the mean ( $PR_t^{avg}$ ) for the target acceptance rate. The difference between fair models and unconstrained models is not as large with *race* as the sensitive attribute, as the unconstrained models are already quite fair there. The results of FairGP are characterised by high fairness and high accuracy. FairLR achieves similar results to FairGP, but with generally slightly lower accuracy but better fairness. We used the two step procedure of Donini et al. (2018) to verify that we cannot achieve the same fairness result with just parameter search on LR.

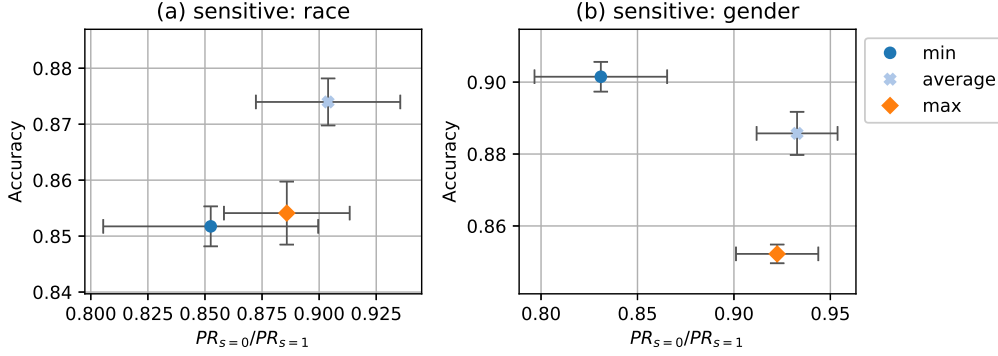


Figure 1: Accuracy and fairness (demographic parity) for various target choices. (a): Adult dataset using race as the sensitive attribute; (b): Adult dataset using gender. Centre of the cross is the mean; height and width of the box encode half of standard deviation of accuracy and disparate impact.

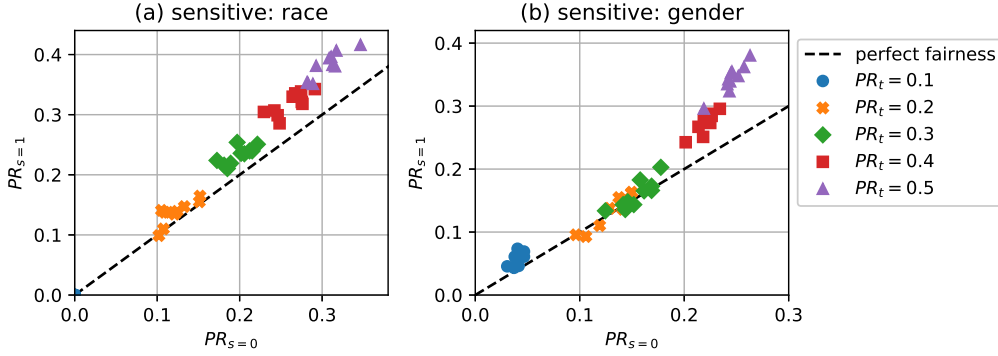


Figure 2: Predictions with different target acceptance rates (demographic parity) for 10 repeats. (a):  $PR_{s=0}$  vs  $PR_{s=1}$  using race as the sensitive attribute; (b):  $PR_{s=0}$  vs  $PR_{s=1}$  using gender.

In Fig. 1, we investigate which choice of target ( $PR_t^{avg}$ ,  $PR_t^{min}$  or  $PR_t^{max}$ ) gives the best result. We use  $PR_t^{avg}$  for all following experiments as this is the fairest choice (cf. Section 3.2). The Fig.1(a) shows results from Adult dataset with *race* as sensitive attribute where we have  $PR_t^{min} = 0.156$ ,  $PR_t^{max} = 0.267$  and  $PR_t^{avg} = 0.211$ .  $PR_t^{avg}$  performs best in term of the trade-off.

Fig. 2(a) and (b) show runs of FairLR where we explicitly set a target acceptance rate,  $PR_t := \mathbb{P}(\bar{y} = 1)$ , instead of taking the mean  $PR_t^{avg}$ . A perfect targeting mechanism would produce a diagonal. The plot shows that setting the target rate has the expected effect on the observed acceptance rate. This tuning of the target rate is the unique aspect of the approach. This would be very difficult to achieve with existing fairness methods; a new constraint would have to be added. The achieved positive rate is, however, usually a bit lower than the targeted rate (e.g. around 0.15 for the target 0.2). This is due to using imperfect classifiers; if TPR and TNR differ from 1, the overall positive rate is affected (see e.g. Forman (2005) for discussion of this).

Fig. 3(a) and (b) show the same data as Fig. 2 but with different axes. It can be seen from this Fig. 3(a) and (b) that the fairness-accuracy trade-off is usually best when the target rate is close to the average of the positive rates in the dataset (which is around 0.2 for both sensitive attribute).

## 5.6 RESULTS FOR EQUALITY OF OPPORTUNITY ON PROPUBLICA DATASET.

For equality of opportunity, we again follow Zafar et al. (2017a) and evaluate the algorithm on the ProPublica dataset. As we did for demographic parity, we define a measure of equality of opportunity

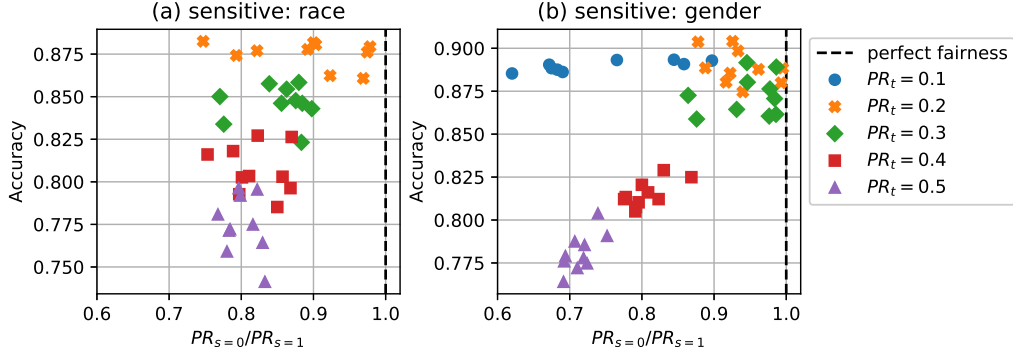


Figure 3: Predictions with different target acceptance rates (demographic parity) for 10 repeats. (a): disparate impact vs accuracy on Adult dataset using race as the sensitive attribute; (b): disparate impact vs accuracy using gender.

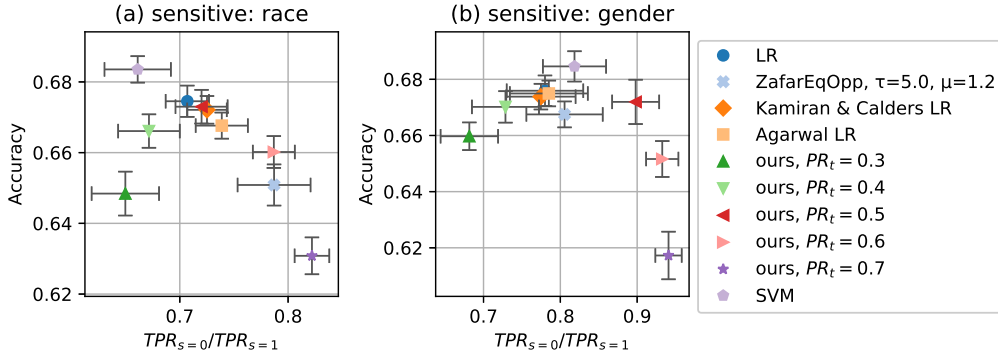


Figure 4: Accuracy and fairness (with respect to *equality of opportunity*) for various methods on ProPublica dataset. (a): using race as the sensitive attribute; (b): using gender. A completely fair model would achieve a value of 1.0 in the x-axis. See Fig. 5(a) and (b) on how these choices of PR setting translate to  $TPR_{s=0}$  vs  $TPR_{s=1}$ .

via the ratio of the true positive rates (TPRs) within the demographic groups. We use  $TPR_{s=i}$  to denote the observed TPR in group  $i$ :  $\mathbb{P}(\hat{y} = 1 | y = 1, s = i)$ , and  $TNR_{s=i}$  for the observed true negative rate (TNR) in the same manner. The measure is then given by  $TPR_{s=0}/TPR_{s=1}$ . A perfectly fair algorithm would achieve 1.0 on the measure.

The results of 10 runs are shown in Fig. 4 and Fig. 5. Fig. 4(a) and (b) show the accuracy-fairness trade-off; Fig. 5(a) and (b) show the achieved TPRs. In the accuracy-fairness plot, varying  $PR_t$  is shown to produce an inverted U-shape: Higher  $PR_t$  still leads to improved fairness, but at a high cost in terms of accuracy.

The latter two plots make clear that the TPR ratio does not tell the whole story: the realisation of the fairness constraint can differ substantially. By setting different target PRs for our method, we can affect TPRs as well, where higher  $PR_t$  leads to higher TPR, stemming from the fact that making more positive predictions increases the chance of making correct positive predictions. Fig. 5 shows that our method can span a wide range of possible TPR values. Tuning these hidden aspects of fairness is the strength of our method.

## 6 DISCUSSION AND CONCLUSION

Fairness is fundamentally not a challenge of algorithms alone, but very much a sociological challenge. A lot of proposals have emerged recently for defining and obtaining fairness in machine learning-

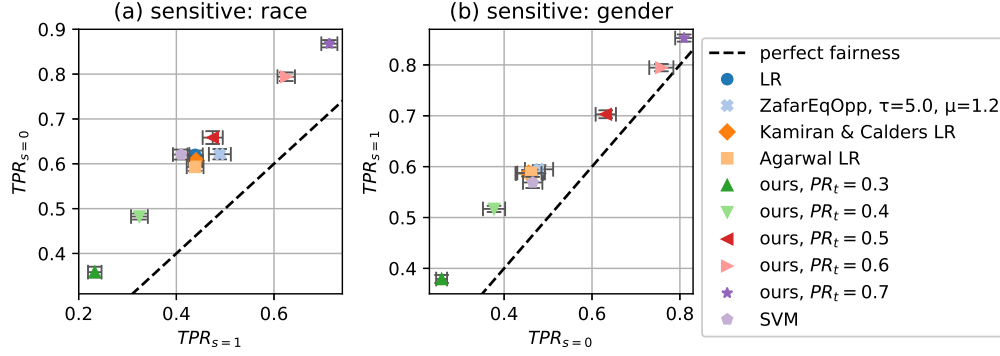


Figure 5: Fairness measure  $TPR_{s=0}$  vs  $TPR_{s=1}$  (equality of opportunity) for different target PRs ( $PR_t$ ). (a): on dataset ProPublica recidivism using race as the sensitive attribute; (b): using gender.

based decision making systems. The vast majority of academic work has focused on two categories of definitions: statistical (group) notions of fairness and individual notions of fairness (see Verma & Rubin (2018) for at least twenty different notions of fairness). Statistical notions are easy to verify but do not provide protections to individuals. Individual notions do give individual protections but need strong assumptions, such as the availability of an agreed-upon similarity metric, which can be difficult in practice. We acknowledge that a proper solution to algorithmic fairness cannot rely on statistics alone. Nevertheless, these statistical fairness definitions can be helpful in understanding the problem and working towards solutions. To facilitate this, at every step, the trade-offs that are present should be made very clear and long-term effects have to be considered as well (Liu et al., 2018; Kallus & Zhou, 2018).

Here, we have developed a machine learning framework which allows us to learn from an implicit balanced dataset, thus satisfying the two most popular notions of fairness (Verma & Rubin, 2018), demographic parity (also known as *avoiding disparate treatment*) and equality of opportunity (or *avoiding disparate mistreatment*). Additionally, we indicate how to extend the framework to cover conditional demographic parity as well. The framework allows us to set a *target rate* to control how the fairness constraint is realised. For example, we can set the target positive rate for demographic parity to be 0.6 for different groups. Depending on the application, it can be important to specify whether non-discrimination ought to be achieved by more positive predictions or more negative predictions. This capability is unique to our approach and can be used as an intuitive mechanism to control the realisation of fairness. Our framework is general and will be applicable for sensitive variables with binary and multi-level values. The current work focuses on a single binary sensitive variable. Future work could extend our tuning approach to other fairness concepts like the closely related predictive parity group fairness (Chouldechova, 2017) or individual fairness (Dwork et al., 2012).

## ACKNOWLEDGEMENTS

Supported by the UK EPSRC project EP/P03442X/1 ‘EthicalML: Injecting Ethical and Legal Constraints into Machine Learning Models’ and the Russian Academic Excellence Project ‘5–100’. We gratefully acknowledge NVIDIA for GPU donations, and Amazon for AWS Cloud Credits. We thank Chao Chen and Songzhu Zheng for their inspiration of our main proof.

## REFERENCES

- Alekh Agarwal, Alina Beygelzimer, Miroslav Dudik, John Langford, and Hanna Wallach. A reductions approach to fair classification. In *ICML*, volume 80, pp. 60–69, 2018.
- Julia Angwin, Jeff Larson, Surya Mattu, and Lauren Kirchner. Machine bias. *ProPublica*, May, 23, 2016.

- Solon Barocas and Andrew D. Selbst. Big data’s disparate impact. *California Law Review*, 104: 671–732, 2016.
- Edwin V Bonilla, Karl Krauth, and Amir Dezfouli. Generic inference in latent gaussian process models. *arXiv preprint arXiv:1609.00577*, 2016.
- Toon Calders and Sicco Verwer. Three naive bayes approaches for discrimination-free classification. *Data Mining and Knowledge Discovery*, 21(2):277–292, 2010.
- Toon Calders, Faisal Kamiran, and Mykola Pechenizkiy. Building classifiers with independency constraints. In *Data mining workshops, 2009. ICDMW’09. IEEE international conference on*, pp. 13–18. IEEE, 2009.
- Silvia Chiappa. Path-specific counterfactual fairness. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pp. 7801–7808, 2019.
- Alexandra Chouldechova. Fair prediction with disparate impact: A study of bias in recidivism prediction instruments. *Big data*, 5(2):153–163, 2017.
- Sam Corbett-Davies, Emma Pierson, Avi Feller, Sharad Goel, and Aziz Huq. Algorithmic decision making and the cost of fairness. In *Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pp. 797–806, 2017.
- Andrew Cotter, Heinrich Jiang, Serena Wang, Taman Narayan, Maya R. Gupta, Seungil You, and Karthik Sridharan. Optimization with non-differentiable constraints with applications to fairness, recall, churn, and other goals. *arXiv preprint arXiv:1809.04198*, 2018.
- Elliot Creager, David Madras, Joern-Henrik Jacobsen, Marissa Weis, Kevin Swersky, Toniann Pitassi, and Richard Zemel. Flexibly fair representation learning by disentanglement. In Kamalika Chaudhuri and Ruslan Salakhutdinov (eds.), *International Conference on Machine Learning (ICML)*, volume 97 of *Proceedings of Machine Learning Research*, pp. 1436–1445. PMLR, 2019.
- Dua Dheeru and Efi Karra Taniskidou. *UCI Machine Learning Repository*, 2017. URL <http://archive.ics.uci.edu/ml>.
- Christos Dimitrakakis, Yang Liu, David C Parkes, and Goran Radanovic. Bayesian fairness. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pp. 509–516, 2019.
- Michele Donini, Luca Oneto, Shai Ben-David, John S Shawe-Taylor, and Massimiliano Pontil. Empirical risk minimization under fairness constraints. In *NeurIPS*, pp. 2796–2806, 2018.
- 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*, pp. 214–226. ACM, 2012.
- Michael Feldman, Sorelle A Friedler, John Moeller, Carlos Scheidegger, and Suresh Venkatasubramanian. Certifying and removing disparate impact. In *Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pp. 259–268. ACM, 2015.
- George Forman. Counting positives accurately despite inaccurate classification. In *European Conference on Machine Learning*, pp. 564–575. Springer, 2005.
- Jacob R Gardner, Geoff Pleiss, David Bindel, Kilian Q Weinberger, and Andrew Gordon Wilson. GPyTorch: Blackbox matrix-matrix gaussian process inference with GPU acceleration. In *NeurIPS*, pp. 7587–7597, 2018.
- Gabriel Goh, Andrew Cotter, Maya Gupta, and Michael P Friedlander. Satisfying real-world goals with dataset constraints. In D. D. Lee, M. Sugiyama, U. V. Luxburg, I. Guyon, and R. Garnett (eds.), *Advances in Neural Information Processing Systems (NIPS)*, pp. 2415–2423, 2016.
- Moritz Hardt, Eric Price, Nati Srebro, et al. Equality of opportunity in supervised learning. In *Advances in neural information processing systems*, pp. 3315–3323, 2016.
- Matthew Joseph, Michael Kearns, Jamie H Morgenstern, and Aaron Roth. Fairness in learning: Classic and contextual bandits. In *NIPS*, pp. 325–333, 2016.

- Nathan Kallus and Angela Zhou. Residual unfairness in fair machine learning from prejudiced data. In *Proceedings of the 35th International Conference on Machine Learning*, volume 80, pp. 2439–2448, 10–15 Jul 2018.
- Faisal Kamiran and Toon Calders. Data preprocessing techniques for classification without discrimination. *Knowledge and Information Systems*, 33(1):1–33, 2012.
- Toshihiro Kamishima, Shotaro Akaho, Hideki Asoh, and Jun Sakuma. Fairness-aware classifier with prejudice remover regularizer. In *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*, pp. 35–50. Springer, 2012.
- Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In Yoshua Bengio and Yann LeCun (eds.), *3rd International Conference on Learning Representations, ICLR 2015*, 2015. URL <http://arxiv.org/abs/1412.6980>.
- Karl Krauth, Edwin V Bonilla, Kurt Cutajar, and Maurizio Filippone. AutoGP: Exploring the capabilities and limitations of Gaussian Process models. *arXiv preprint arXiv:1610.05392*, 2016.
- Gert R. Lanckriet and Bharath K. Sriperumbudur. On the convergence of the concave-convex procedure. In Y. Bengio, D. Schuurmans, J. D. Lafferty, C. K. I. Williams, and A. Culotta (eds.), *Advances in Neural Information Processing Systems (NIPS)*, pp. 1759–1767, 2009.
- Liyuan Liu, Haoming Jiang, Pengcheng He, Weizhu Chen, Xiaodong Liu, Jianfeng Gao, and Jiawei Han. On the variance of the adaptive learning rate and beyond. *arXiv preprint arXiv:1908.03265*, 2019.
- Lydia T. Liu, Sarah Dean, Esther Rolf, Max Simchowitz, and Moritz Hardt. Delayed impact of fair machine learning. In *Proceedings of the 35th International Conference on Machine Learning*, volume 80, pp. 3150–3158, 10–15 Jul 2018.
- Christos Louizos, Kevin Swersky, Yujia Li, Max Welling, and Richard Zemel. The variational fair autoencoder. In *International Conference on Learning Representations (ICLR)*, 2016.
- Kristian Lum and James Johndrow. A statistical framework for fair predictive algorithms. *arXiv preprint arXiv:1610.08077*, 2016.
- Alexandra Olteanu, Carlos Castillo, Fernando Diaz, and Emre Kıcıman. Social data: Biases, methodological pitfalls, and ethical boundaries. *Frontiers in Big Data*, 2:13, 2019.
- Novi Quadrianto and Viktoriia Sharmanska. Recycling privileged learning and distribution matching for fairness. In *Advances in Neural Information Processing Systems*, pp. 677–688, 2017.
- Novi Quadrianto, Viktoriia Sharmanska, and Oliver Thomas. Discovering fair representations in the data domain. In *Computer Vision and Pattern Recognition (CVPR)*, pp. 8227–8236. Computer Vision Foundation / IEEE, 2019.
- Songül Tolan. Fair and unbiased algorithmic decision making: Current state and future challenges. *arXiv preprint arXiv:1901.04730*, 2019.
- Alexander B Tsybakov et al. Optimal aggregation of classifiers in statistical learning. *The Annals of Statistics*, 32(1):135–166, 2004.
- Sahil Verma and Julia Rubin. Fairness definitions explained. In *2018 IEEE/ACM International Workshop on Software Fairness (FairWare)*, pp. 1–7. IEEE, 2018.
- Muhammad Bilal Zafar, Isabel Valera, Manuel Gomez Rodriguez, and Krishna P Gummadi. Fairness beyond disparate treatment & disparate impact: Learning classification without disparate mistreatment. In *Proceedings of the 26th International Conference on World Wide Web*, pp. 1171–1180. International World Wide Web Conferences Steering Committee, 2017a.
- Muhammad Bilal Zafar, Isabel Valera, Manuel Gomez Rogriguez, and Krishna P Gummadi. Fairness constraints: Mechanisms for fair classification. In *Artificial Intelligence and Statistics*, pp. 962–970, 2017b.
- Rich Zemel, Yu Wu, Kevin Swersky, Toni Pitassi, and Cynthia Dwork. Learning fair representations. In *International Conference on Machine Learning*, pp. 325–333, 2013.

## A PROOF OF THEOREM 1

Let  $\eta(x, s) = P(y = 1|x, s)$  be the distribution of the training data. Let  $\bar{\eta}(x, s) = m_s \cdot \eta(x, s) + b_s$ , where

$$\begin{aligned} m_s &= P(\bar{y} = 1|y = 1, s) - P(\bar{y} = 1|y = 0, s) \\ &= 1 - P(\bar{y} = 0|y = 1, s) - P(\bar{y} = 1|y = 0, s) \\ b_s &= P(\bar{y} = 1|y = 0, s) \end{aligned} \quad (31)$$

So,  $\bar{\eta}(x, s) = P(\bar{y} = 1|x, s)$ . Let  $y$  denote the *hard* labels for  $\eta$ :  $y = \mathbb{I}[\eta > \frac{1}{2}]$  and  $\bar{y}$  be the hard labels for  $\bar{\eta}$ :  $\bar{y} = \mathbb{I}[\bar{\eta} > \frac{1}{2}]$ .

**Theorem 3.** *The probability that  $y$  and  $\bar{y}$  disagree ( $y \neq \bar{y}$ ) for any input  $x$  in the dataset is given by:*

$$\mathbb{P}(y \neq \bar{y}|s) = \mathbb{P}\left(|\eta(x, s) - \frac{1}{2}| < t_s\right) \quad (33)$$

where

$$t_s = \left| \frac{m_s + 2b_s - 1}{2m_s} \right|. \quad (34)$$

*Proof.* The decision boundary that lets us recover the true labels is at  $\frac{1}{2}$  (independent of  $s$ ). So, for the shifted distribution,  $\bar{\eta}$ , this threshold to get the true labels would be at  $\frac{1}{2} \cdot m_s + b_s$  (it depends on  $s$  now). If we however use the decision boundary of  $\frac{1}{2}$  for  $\bar{\eta}$ , to make our predictions,  $\bar{y}$ , then this prediction will sometimes not correspond to the true label,  $y \neq \bar{y}$ . When does this happen?

Let  $d_s$  be the new decision boundary:  $d_s = \frac{1}{2} \cdot m_s + b_s$ . There are two possibilities to consider here: either  $\frac{1}{2} < d_s$  or  $\frac{1}{2} > d_s$  (for  $d_s = \frac{1}{2}$ , the decision boundaries are the same and nothing has to be shown). The problem,  $y \neq \bar{y}$ , appears then exactly when the value of  $\bar{\eta}$  is between the two boundaries:

$$\text{if } d_s > \frac{1}{2}: \quad d_s > \bar{\eta}(x, s) > \frac{1}{2} \quad (35)$$

$$\text{if } d_s < \frac{1}{2}: \quad d_s < \bar{\eta}(x, s) < \frac{1}{2} \quad (36)$$

Expressing this in terms of  $\eta$  and simplifying leads to (if  $m_s$  is negative, then the two cases are swapped, but we still get both inequalities):

$$\text{if } d_s > \frac{1}{2}: \quad \frac{1}{2} > \eta(x, s) > \frac{1 - 2b_s}{2m_s} \quad (37)$$

$$\text{if } d_s < \frac{1}{2}: \quad \frac{1}{2} < \eta(x, s) < \frac{1 - 2b_s}{2m_s} \quad (38)$$

This can be summarized as

$$\left| \eta(x, s) - \frac{1}{2} \right| < \left| \frac{1}{2} - \frac{1 - 2b_s}{2m_s} \right|. \quad (39)$$

Let  $t_s$  denote the term on the right side of this inequality (i.e. the “threshold” that determines whether  $y = \bar{y}$  or not). Then

$$t_s = \left| \frac{1}{2} - \frac{1 - 2b_s}{2m_s} \right| = \left| \frac{m_s + 2b_s - 1}{2m_s} \right|. \quad (40)$$

So, we have:  $|\eta(x, s) - \frac{1}{2}| < t_s = \left| \frac{m_s + 2b_s - 1}{2m_s} \right|$ . This leads directly to the statement we wanted to prove:

$$P(y \neq \bar{y}|s) = P\left(\left| \eta(x, s) - \frac{1}{2} \right| < t_s\right). \quad (41)$$

□



## B FINDING MINIMAL $t_s$

We express  $t_s$  in terms of  $PR_b^s$  and  $PR_t$ .

$$t_s = \begin{cases} \frac{1}{2} \frac{PR_b^s - PR_t}{PR_t} & \text{if } PR_t > PR_b^j \\ \frac{1}{2} \frac{PR_t - PR_b^s}{1 - PR_t} & \text{otherwise.} \end{cases} \quad (42)$$

Without loss of generality, we assume  $PR_b^0 < PR_b^1$ . As mentioned in the main text,  $PR_t$  should be between  $PR_b^0$  and  $PR_b^1$  to minimize both  $t_s$ . If that is the case, then we get

$$t_{s=0} = \frac{1}{2} \frac{PR_t - PR_b^0}{1 - PR_t} \quad (43)$$

$$t_{s=1} = \frac{1}{2} \frac{PR_b^1 - PR_t}{PR_t} . \quad (44)$$

If we further assume  $PR_b^1 < \frac{1}{2}$ , then we also have  $PR_t < \frac{1}{2}$  and thus  $PR_t < 1 - PR_t$ . This implies that the denominator of  $t_{s=1}$  is smaller and that, in turn,  $t_{s=1}$  grows faster. This faster growth means that when minimizing  $t_{s=0} + t_{s=1}$ , we have to concentrate on  $t_{s=1}$ . The minimum is then such that  $t_{s=1}$  is 0, i.e.  $PR_t = PR_b^1$ .

## C PROOF OF THEOREM 2

We are given a dataset  $\mathcal{D} = \{(x_i, y_i)\}_i$ , where the  $x_i$  are vectors of features and the  $y_i$  the corresponding labels. We refer to the tuples  $(x, y)$  as the *samples* of the dataset. The number of samples is  $N = |\mathcal{D}|$ .

We assume binary labels ( $y \in \{0, 1\}$ ) and thus can form the (disjoint) subsets  $\mathcal{Y}^0$  and  $\mathcal{Y}^1$  with

$$\mathcal{Y}^j = \{(x, y) \in \mathcal{D} | y = j\} \quad \text{with } j \in \{0, 1\} . \quad (45)$$

Furthermore, we associate each sample with a classification  $\hat{y} \in \{0, 1\}$ . The task of making the classification  $\hat{y} = 0$  or  $\hat{y} = 1$  can be understood as putting each sample from  $\mathcal{D}$  into one of two sets:  $\mathcal{C}^0$  and  $\mathcal{C}^1$ , such that  $\mathcal{C}^0 \cup \mathcal{C}^1 = \mathcal{D}$  and  $\mathcal{C}^0 \cap \mathcal{C}^1 = \emptyset$ .

We refer to the set  $\mathcal{A} = (\mathcal{C}^0 \cap \mathcal{Y}^0) \cup (\mathcal{C}^1 \cap \mathcal{Y}^1)$  as the set of correct (or accurate) predictions. The *accuracy* is given by  $acc = N^{-1} \cdot |\mathcal{A}|$ . From the definition it is clear that  $0 \leq acc \leq 1$ .

**Definition 4.**

$$r_a := \frac{|\mathcal{Y}^1|}{|\mathcal{D}|} = \frac{|\mathcal{Y}^1|}{N} \quad (46)$$

is called the *acceptance rate* of the dataset  $\mathcal{D}$ .

**Definition 5.**

$$\hat{r}_a = \frac{|\mathcal{C}^1|}{|\mathcal{D}|} = \frac{|\mathcal{C}^1|}{N} \quad (47)$$

is called the *target rate* of the predictions.

**Theorem 4.** *For a dataset with the acceptance rate  $r_a$  and corresponding predictions with a target rate of  $\hat{r}_a$ , the accuracy is limited by*

$$acc \leq 1 - |\hat{r}_a - r_a| . \quad (48)$$

*Proof.* We first note that by multiplying by  $N$ , the inequality becomes

$$|\mathcal{A}| \leq N - ||\mathcal{C}^1| - |\mathcal{Y}^1|| . \quad (49)$$

We will choose the predictions  $\hat{y}$  that achieve the highest possible accuracy (largest possible  $\mathcal{A}$ ) and show that this can never exceed  $1 - |\hat{r}_a - r_a|$ . As the set  $\mathcal{Y}^1$  contains all samples that correspond to  $y = 1$ , we try to take as many samples from  $\mathcal{Y}^1$  for  $\mathcal{C}^1$  as possible. Likewise, we take as many indices as possible from  $\mathcal{Y}^0$  for  $\mathcal{C}^0$ .

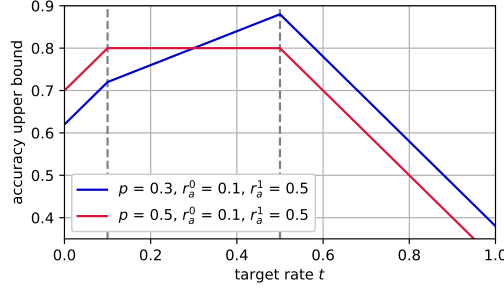


Figure 6: Achievable accuracy for different target values.

We consider three cases:  $\hat{r}_a = r_a$ ,  $\hat{r}_a < r_a$  and  $\hat{r}_a > r_a$ . The first case is trivial; we have  $|\mathcal{C}^1| = |\mathcal{Y}^1|$  and thus are able to set  $\mathcal{C}^1 = \mathcal{Y}^1$ ,  $\mathcal{C}^0 = \mathcal{Y}^0$  and achieve perfect accuracy ( $acc \leq 1$ ).

For  $\hat{r}_a < r_a$ , we have  $|\mathcal{C}^1| < |\mathcal{Y}^1|$  and thus have more samples available with  $y = 1$  than we would optimally need to select for  $\mathcal{C}^1$ . There are two terms to consider that make up the definition of  $\mathcal{A}$ :  $\mathcal{C}^0 \cap \mathcal{Y}^0$  and  $\mathcal{C}^1 \cap \mathcal{Y}^1$ . The intersection of these two terms is empty because  $\mathcal{C}^0 \cap \mathcal{C}^1 = \emptyset$ . Thus,

$$|\mathcal{A}| = |(\mathcal{C}^0 \cap \mathcal{Y}^0) \cup (\mathcal{C}^1 \cap \mathcal{Y}^1)| = |(\mathcal{C}^0 \cap \mathcal{Y}^0)| + |(\mathcal{C}^1 \cap \mathcal{Y}^1)|. \quad (50)$$

Selecting samples from  $\mathcal{Y}^1$  for  $\mathcal{C}^0$  will only *decrease* the first term, so for maximum accuracy, it is fine to take as many samples from  $\mathcal{Y}^1$  for  $\mathcal{C}^1$ . Taking all available samples from  $\mathcal{Y}^1$  such that  $\mathcal{C}^1 \supset \mathcal{Y}^1$ , there is still space left in  $\mathcal{C}^1$  which we will have to fill with samples with  $y = 0$ . Thus, we have  $\mathcal{C}^1 \cap \mathcal{Y}^1 = \mathcal{Y}^1$ . For  $\mathcal{C}^0$ , we have enough  $y = 0$  such that  $\mathcal{C}^0 \subset \mathcal{Y}^0$  and  $\mathcal{C}^0 \cap \mathcal{Y}^0 = \mathcal{C}^0$ . This is the largest we can make these intersections. Putting everything together:

$$\begin{aligned} |\mathcal{A}^{optimal}| &= |(\mathcal{C}^0 \cap \mathcal{Y}^0)| + |(\mathcal{C}^1 \cap \mathcal{Y}^1)| = |\mathcal{C}^0| + |\mathcal{Y}^1| \\ &= N - |\mathcal{C}^1| + |\mathcal{Y}^1| = N - (|\mathcal{C}^1| - |\mathcal{Y}^1|). \end{aligned} \quad (51)$$

For  $\hat{r}_a > r_a$ , the roles of  $\mathcal{C}^0$  and  $\mathcal{C}^1$  are reversed and thus, the signs in the equation are inverted:

$$|\mathcal{A}^{optimal}| = N - (|\mathcal{Y}^1| - |\mathcal{C}^1|). \quad (52)$$

This proves the claim.  $\square$

**Corollary 4.1.** *Given a dataset that consists of two subsets  $\mathcal{S}_0$  and  $\mathcal{S}_1$  ( $\mathcal{D} = \mathcal{S}_0 \cup \mathcal{S}_1$ ) where  $p$  is the ratio of  $|\mathcal{S}_0|$  to  $|\mathcal{D}|$  and given corresponding acceptance rates  $r_a^0$  and  $r_a^1$  and predictions with target rates  $\hat{r}_a^0$  and  $\hat{r}_a^1$ , the accuracy is limited by*

$$acc \leq 1 - p \cdot |\hat{r}_a^0 - r_a^0| - (1 - p) \cdot |\hat{r}_a^1 - r_a^1|. \quad (53)$$

**Example 1.** We consider the case where  $\mathcal{S}_0$  (which could for example be all data points for female individuals) makes up 30% of the dataset; so  $p = 0.3$ . Further, we say that for  $\mathcal{S}_0$  we have an acceptance rate of 10% ( $r_a^0 = 0.1$ ) and for  $\mathcal{S}_1$ , 50% ( $r_a^1 = 0.5$ ). If we then set both target rates to the same value  $t$  ( $\hat{r}_a^0 = \hat{r}_a^1 = t$ ), with  $t = 0.3$ , then the highest accuracy that can be achieved is 0.8 or 80%.

Fig 6 shows the achievable accuracy for different values of  $t$  in blue: We can see that we can achieve the highest accuracy for  $t = r_a^1 = 0.5$ , namely 88%. The plot in orange shows the achievable accuracy for  $p = 0.5$ , i.e., when the two subsets have the same size. In this case, all target rates between  $r_a^0$  and  $r_a^1$  give equal results, namely 80%.

## D ILLUSTRATION OF RESTRICTIONS ON PR

We start by setting a target rate  $r_t$ :

$$P(\bar{y} = 1 | s = 0) \stackrel{!}{=} r_t \quad \text{and} \quad P(\bar{y} = 1 | s = 1) \stackrel{!}{=} r_t \quad (54)$$

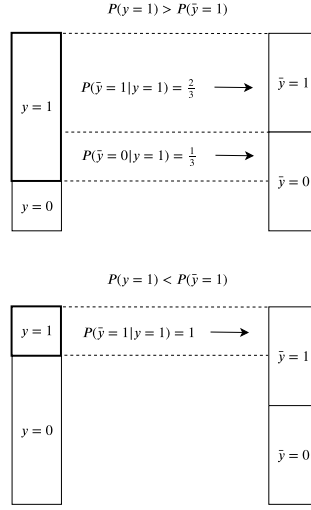


Figure 7: Illustration of demographic parity with target labels. In the situation in the upper part,  $P(\bar{y} = 1|y = 1)$  cannot be set to 1, because there are more samples with  $y = 1$  than there are  $\bar{y} = 1$ . In the situation in the lower part,  $P(\bar{y} = 1|y = 1)$  can be set to 1.

This leads us to the following constraint for  $s' \in \{0, 1\}$ :

$$\begin{aligned} r_t &= P(\bar{y} = 1|s = s') \\ &= \sum_y P(\bar{y} = 1|y, s = s')P(y|s = s') \end{aligned} \quad (55)$$

For  $P(y|s = s')$  we will put in the value at which we want our constraint to hold. We denote  $P(y = 1|s = j)$  as the base rate  $r_b^j$  which we estimate from the training set. Plugging this in, we are left with

$$\begin{aligned} r_t &= P(\bar{y} = 1|y = 0, s = 0) \cdot (1 - r_b^0) \\ &\quad + P(\bar{y} = 1|y = 1, s = 0) \cdot r_b^0 \end{aligned} \quad (56)$$

$$\begin{aligned} r_t &= P(\bar{y} = 1|y = 0, s = 1) \cdot (1 - r_b^1) \\ &\quad + P(\bar{y} = 1|y = 1, s = 1) \cdot r_b^1. \end{aligned} \quad (57)$$

This is a system of linear equations with two equations and four free variables. There is thus still considerable freedom in how we want our constraint to be realized. The freedom that we have here concerns how strongly the accuracy will be affected.

If we set  $P(\bar{y} = 1|y = 1, s)$  to 0.5, then we express the fact that a train label of 1 only implies a target label of 1 in 50% of the cases. In order to minimize the effect on accuracy, we make  $P(\bar{y} = 1|y = 1, s)$  as high as possible and  $P(\bar{y} = 1|y = 0, s)$  as low as possible.

We solve for  $P(\bar{y} = 1|y = 0, s = j)$ :

$$\begin{aligned} &P(\bar{y} = 1|y = 0, s = j) \\ &= \frac{r_b^j}{1 - r_b^j} \left( \frac{r_t}{r_b^j} - P(\bar{y} = 1|y = 1, s = j) \right). \end{aligned} \quad (58)$$

However, we can set  $P(\bar{y} = 1|y = 0, s = j)$  to 0 only if that does not imply  $P(\bar{y} = 1|y = 1, s = j)$  will be greater than 1. This would happen if  $r_t/r_b^j$  were greater than 1.

Figure 7 illustrates this. In the upper part of the figure we have  $r_t/r_b^j$  less than 1. In this case