Journal of Machine Learning Research 25 (2024) 1-42 Submitted 6/23; Revised 5/24; Published 7/24

Interpretable algorithmic fairness in structured and unstructured data

Hari Bandi HBANDI@MIT.EDU
Massachusetts Institute of Technology
77 Massachusetts Avenue, Cambridge, MA 02139, USA

Dimitris Bertsimas DBERTSIM@MIT.EDU
Sloan School of Management, Massachusetts Institute of Technology
77 Massachusetts Avenue, Cambridge, MA 02139, USA

Thodoris Koukouvinos TKOUKOUV@MIT.EDU
Operations Research Center, Massachusetts Institute of Technology
77 Massachusetts Avenue, Cambridge, MA 02139, USA

Sofie Kupiec SKUPIEC@MIT.EDU
Massachusetts Institute of Technology
77 Massachusetts Avenue, Cambridge, MA 02139, USA

Editor: Manuel Gomez-Rodriguez

Abstract

Systemic bias with respect to gender and race is prevalent in datasets, making it challenging to train classification models that are accurate and alleviate bias. We propose a unified method for alleviating bias in structured and unstructured data, based on a novel optimization approach for optimally flipping outcome labels and training classification models simultaneously. In the case of structured data, we introduce constraints on selected objective measures of meritocracy, and present four case studies, demonstrating that our approach often outperforms state-of the art methods in terms of fairness and meritocracy. In the case of unstructured data, we present two case studies on image classification, demonstrating that our method outperforms state-of-the-art approaches in terms of fairness. Moreover, we note that the decrease in accuracy over the nominal model is 3.31% on structured data and 0.65% on unstructured data. Finally, we leverage Optimal Classification Trees (OCTs), to provide insights on which attributes of individuals lead to flipping of their labels and apply it to interpret the flipping decisions on structured data. Utilizing OCTs with auxiliary tabular data as well as Gradient-weighted Class Activation Mapping (Grad-CAM), we provide insights on the flipping decisions for unstructured data.

Keywords: Fair Classification, Bias Alleviation, Mixed Integer Optimization

1. Introduction

In this paper, we consider bias with respect to a sensitive attribute, that can be gender, race or ethnicity. Often classification tasks for structured data including college admissions Wightman (1998) and hiring processes Qin et al. (2018) as well as for unstructured data including face recognition Deng et al. (2019), exhibit a discrimination against people of certain color or gender (Angwin et al. (2016), Wightman (1998)). One reason behind this is the presence of bias in the data. Historical data aggregated in such settings may be biased against certain demographic groups due to systemic bias. Since the datasets used for model training consist of historical data populated with choices made by people, systemic bias may be concealed in them. Consequently, it has been shown that without appropriate intervention during training or evaluation, classification models trained on such datasets can be biased against certain groups of individuals (Angwin et al. (2016); Hardt et al. (2016)). This is due to the fact that during the training process, bias present in the dataset becomes reinforced into the model Bolukbasi et al. (2016). Thus, bias alleviation in machine learning (ML) models has become an increasingly important concern, which we address in this paper.

To address this problem, simple remedies for structured data, such as ignoring the protected attributes, e.g., gender, race, ethnicity, etc., are largely ineffective due to other features being correlated with them Pedreshi et al. (2008). Remedies for unstructured data include the use of boosting methods to replace a deployed deep learning model with a new one that has equal accuracy in different subpopulations of the sensitive attribute Kim et al. (2019), however this approach cannot ensure an equal prediction treatment of individuals in different subpopulations. The given data can be inherently biased in possibly complex ways, making it difficult to alleviate bias. Moreover, it is both unethical and illegal to design a system that makes decisions entirely on the basis of protected demographic attributes, see (Peffer (2009), Barocas and Selbst (2016)) for more details about disparate treatment. Consequently, classification models that are actively trained on such datasets consisting of both human-made and model-made choices can progressively become more biased over time through feedback loops leading to amplified bias against a certain subpopulation. Such feedback loops have been observed for structured data, in predictive policing Lila et al. (2019) and credit markets, while the problem of *disparity amplification* is a possibility in any deployed machine learning model that is, trained on historical data, either structured or unstructured. Therefore, it is critical for any ML model to actively identify and alleviate systemic biases, improving demographic diversity in predicted outcomes.

In the case of structured data, it is important to alleviate bias, while keeping a meritocratic decision making procedure. More precisely, we define *meritocracy* as the practice of selecting people, based on achievement. A desired classification model for structured data, would make fair predictions among the classes of the sensitive attribute, while respecting meritocracy. In our approach, we take both considerations into account when training a classification model on structured data.

To this end, we propose a unified method for improving fairness in structured and unstructured data, leveraging a novel optimization formulation to optimally flip outcome labels, while training classifiers. In the case of structured data, we incorporate additional constraints on selected objective measures of meritocracy. Finally, utilizing OCTs Bertsimas and Dunn (2017) and Grad-CAM Selvaraju et al. (2017), we provide insights on attributes of individuals that lead to flipping of their labels.

Related Work

The literature on fairness in classification and bias alleviation can be categorized into three main approaches: Pre-processing methods that change the data before training, in-processing methods that add constraints or change the objective during training and post-processing methods that adjust the predictions of the classifier after training.

Pre-processing methods: This approach of bias alleviation involves pre-processing the training data. (Calders et al. (2009); Kamiran and Calders (2012); Zliobaite et al. (2011)) introduced several preprocessing techniques for improving fairness in structured data. First, they proposed removing the sensitive attribute and other correlated attributes. Further, they suggested flipping some of the training labels, as decided by a ranker. They also proposed sampling methods, that divide the training data in subgroups and then over-sample / under-sample them. Moreover, Feldman (2015) proposed modifying attributes in the data in order to make them independent of the sensitive attribute. Another body of work involves learning fair data representations. In the case of structured data, Zemel et al. (2013) formulated an optimization problem for learning fair data representations, leveraging clusters that do not carry information about the sensitive attribute. In the case of unstructured data, Louizos et al. (2016) and Quadrianto et al. (2019) proposed autoencoding neural networks for learning fair data representations and Ramaswamy et al. (2021) leveraged generative models to obtain perturbed data and then augment the training set in order to achieve a similar data distribution among the classes of the sensitive attribute.

In-processing methods: This approach of bias alleviation involves modifying the training of a classifier. Much work for structured data is focused on adding fairness constraints or penalties during the training of classifiers in order to improve the overall fairness. Goh et al. (2016) and Zafar et al. (2017) formulated the fairness requirements as linear constraints, which then included in empirical loss minimization for Logistic Regression (LR) and Support Vector Machines (SVM), resulting in convex optimization problems. Moreover (Kearns et al. (2018); Agarwal et al. (2018); Cotter et al. (2019)) extended the approach to the nonconvex setting, by forming the Lagrangian, that is, adding the linear constraints in the objective with Lagrange multipliers, and framing the constrained optimization problem as a two-player game where the first player minimizes the model parameters and the second maximizes the Lagrange multipliers. However, Cotter et al. (2019) showed that training such models can be difficult and convergence to a solution might not be reached. Moreover, Corbett-Davies et al. (2017) and Narasimhan (2018) derived algorithmic solutions for the general empirical risk minimization problem subject to fairness constraints. Another body of work refers to learning data representations that do not contain any information about the sensitive attribute through adversarial training. In this case a min-max optimization problem is formulated by maximizing total accuracy while minimizing the ability of a discriminator to predict the sensitive attribute. Beutel et al. (2017) applied this approach on structured data, (Edwards and Storkey (2016); Wang et al. (2022)) applied it on unstructured data and Zhang et al. (2018) applied it on both. Other works include adding penalties in the objective function to achieve various forms of fairness, refer to (Donini et al. (2018); Komiyama et al. (2018)) for linear models and kernel methods, respectively. Finally, Kamiran and Calders (2012) proposed adding weights to the training data during the training of a classifier in order to mitigate bias and Hashimoto et al. (2018) incorporated Distributionally Robust Optimization (DRO) in the training of a classifier, by minimizing the worst case empirical risk within an appropriately constructed uncertainty set, applicable to both structured and unstructured data.

Post-processing methods: This approach of bias alleviation involves calibrating the output of a trained classifier, which applies to both structured and unstructured data. Hardt et al. (2016) formulated an optimization problem to achieve this, with equality constraints on the true/false positive rates. Furthermore, Pleiss et al. (2017) showed that calibration of the outputs after training can lead to models with a poor accuracy trade-off and also demonstrated that a deterministic solution is only compatible with a single fairness constraint and thus cannot be applied to a group of fairness constraints. In certain special cases, Woodworth et al. (2017) showed that post-processing the outputs of a classifier can be provably suboptimal and in doing so, the resulting classifiers are incompatible with other notions of fairness (Chouldechova (2017); Kleinberg et al. (2016)). Finally, Lohia et al. (2019) proposed modifying the sensitive attribute labels in the test set, in order to improve fairness, utilizing a bias score, and Kim et al. (2019) proposed modifying the predicted labels in the test set, utilizing boosted models on top of trained classifiers.

We note that methods that simply remove the sensitive attribute and related attributes from the training data or under-sample the training data in order to alleviate bias, can worsen the accuracy of the classifier significantly, while not alleviate bias due to chain of correlations with other unobserved attributes, see Calders et al. (2013). On the other hand, in our method we do not throw away data and thus obtain a highly accurate classifier. Moreover, the pre-processing method by Calders et al. (2009), which consists of flipping some of the labels as decided by a ranker, depends heavily on the model utilized as the ranker and can carry out any error it might have. Krasanakis et al. (2018) mentioned that certain data can cause certain biases to different types of classifiers. This cannot be captured by Calders et al. (2009), since the label massaging technique is independent of the model used for the downstream classification task. Krasanakis et al. (2018) further mentioned that it could be more informative to directly observe the effect of biases on the classifier and suitably perform adjustments while training, which is exactly the scope of our method. More precisely, our approach is incorporated in the training of any classifier, for which it identifies the best label flips in order to alleviate bias. We formulate a min-min optimization problem to flip a subset of labels and simultaneously optimize for the parameters of the classification model. Our method is applicable as long as the training data are biased. We remark that another important difference between our approach and the method by Calders et al. (2009) is that in our case we take meritocracy into account. Finally, we note that the label flips from our method only affect the learning of the model parameters and not the decision making. Once the model parameters are learned, they can be used to obtain predictions in the test set, without flipping any labels at that time.

We note that apart from fairness, our approach can also improve out of sample accuracy in certain cases, depending on the amount of bias in the test data. More precisely, if the test data are less biased than the training data, then both fairness and accuracy can be improved, whereas if the test data are at least as biased as the training data we can improve fairness while worsening accuracy, see Wick et al. (2019) for more details.

In summary, our method incorporates the label massaging idea from Calders et al. (2009) in the training of any classifier, allowing us to find the optimal label flips for that classification model. To achieve this, we introduce additional binary variables during training and leverage alternating optimization. In this way, we are able to alleviate bias while not suffer from the shortcomings of the label massaging technique from Calders et al. (2009).

Our approach is computationally efficient, and can be applied to a wide class of ML models trained by stochastic gradient descent, in both structured and unstructured data.

Contributions

Our main contributions can be summarized as follows:

1. We propose a novel optimization approach based on simultaneously optimally flipping labels and training classification models that alleviate systemic bias, that applies to both structured and unstructured data. The formulation is very generic and computationally efficient, leveraging stochastic projected gradient descent. The projection problem is a mixed integer linear optimization (MILO) problem with binary variables equal to the size of training data and therefore it is practically solvable.

2. We apply the proposed framework to structured data, for tabular data classification. We introduce constraints on selected objective measures of meritocracy to restrict distributional differences among datasets with flipped and non-flipped labels. Further, we present case studies on four real-world datasets, the Law School Admission Council (LSAC) dataset, the Crime dataset, the Correctional Offender Management Profiling for Alternative Sanctions (COMPAS) dataset and the German Credit dataset. We demonstrate that our method often outperforms state-of-the-art approaches in terms of fairness and meritocracy.

3. We apply the proposed framework to unstructured data, for image data classification. We present case studies on two widely used datasets for image classification, CelebFaces Attributes (CelebA) and Labeled Faces in the Wild (LFW), demonstrating that our approach outperforms state-of-the-art methods in terms of fairness.

4. We utilize Optimal Classification Trees (OCTs) and Gradient-weighted Class Activation Mapping (Grad-CAM) to provide insights on which attributes lead to flipping of labels, and to help make changes in the current classification processes in a manner understandable by human decision makers. In the case of structured data, we illustrate that the label flips from our method are intuitive and further that our method follows a more meritocratic decision making procedure than the method by Calders et al. (2009). Moreover, in the case of unstructured data, we show that the label flips from our method are also intuitive.

The rest of the paper is structured as follows: In Section 2, we illustrate the building blocks of our method, in Section 3, we define the fairness metrics used for evaluation, in Section 4, we present our numerical results for structured and unstructured data classification, in Section 5, we provide a qualitative analysis of our flipping decisions, in Section 6 we provide a discussion of our method, and finally, we summarize our key findings in Section 7.

The notation that we use is as follows: We use bold faced characters such as $\boldsymbol{x}$ to represent vectors and bold faced capital letters such as $\boldsymbol{X}$ to represent matrices. We define $[n] = \{1, \ldots, n\}$. The $\|\cdot\|_2$ norm of a vector refers to $\|\boldsymbol{x}\|_2 = \sqrt{\sum_{i=1}^{n} x_i^2}$ and the $\|\cdot\|_1$ norm of a vector refers to $\|\boldsymbol{x}\|_1 = \sum_{i=1}^n |x_i|$. Finally, we note that throughout the paper we say that a problem is practically solvable if it can be solved for most real-world datasets.

2. Framework

2.1 Method

We assume that we have training data $\{\boldsymbol{x}_i, y_i\}_{i=1}^n$, where $\boldsymbol{x}_i \in \mathcal{X}$ and can be either structured or unstructured data and $y_i \in \mathcal{Y} = \{-1, 1\}$. Our goal is to learn a classifier parametrized by $\boldsymbol{\theta} \in \Theta$, that minimizes the empirical loss over the training data. Assuming a loss function $\ell(y, \boldsymbol{x}, \boldsymbol{\theta})$, the problem can be formulated as follows:

$$\hat{\boldsymbol{\theta}} = \quad \underset{\boldsymbol{\theta} \in \Theta}{\operatorname{argmin}} \frac{1}{n} \sum_{i=1}^{n} \ell(y_i, \boldsymbol{x}_i, \boldsymbol{\theta}). \tag{1}$$

Apart from being accurate, we also want the classifier to be fair in terms of a sensitive attribute. For this purpose we can flip some of the labels of the training data, while optimizing the model parameters. We introduce binary variables $z_i \in \{0, 1\},\ i \in [n]$ for deciding whether the label for the i-th observation is to be flipped ($z_i = 1$) or not ($z_i = 0$). If the original label is $y_i \in \{-1, 1\}$, then the modified label would be $\tilde{y}_i = y_i(1 - 2z_i)$. Let S denote the binary sensitive attribute and $\mathcal{S}_1, \mathcal{S}_2 \subseteq [n]$ denote the subset of training data belonging to each class, with $|\mathcal{S}_1| = n_1,\ |\mathcal{S}_2| = n_2$ and $n_1 + n_2 = n$. Let p_1, p_2 denote the number of positively labeled observations in each class of the sensitive attribute. Let τ_1, τ_2 denote the proportion of labels that are flipped in $\mathcal{S}_1$ and $\mathcal{S}_2$, respectively. We require that $\boldsymbol{z} \in \mathcal{Z}_{\tau_1, \tau_2}$, where

$$\mathcal{Z}_{\tau_1,\tau_2} = \left\{ \boldsymbol{z} \in \{0,1\}^n : \sum_{i \in \mathcal{S}_1} z_i = \lceil \tau_1 \cdot n_1 \rceil,\ \sum_{i \in \mathcal{S}_2} z_i = \lceil \tau_2 \cdot n_2 \rceil \right\}.$$

Recall that a reason for which a classifier predicts a positive label more often in one class of the sensitive attribute than the other, is a higher rate of positive observations in that class. Therefore, we can alleviate bias by equalizing the rate of positive observations among the classes of the sensitive attribute. Instead of enforcing it as a hard constraint which can be too restrictive and possibly decrease accuracy and meritocracy significantly, we add it as a soft constraint with a tolerance ϵ. We note that in the case of multiple or multi-valued sensitive attributes, equalizing the positive rates among groups might not be possible, see Section 2.4. Without loss of generality, we assume that the rate of positive observations among the two classes is greater in class 1, that is, $p_1/n_1 > p_2/n_2$. Since our objective is to decrease the difference between the rates of positive observations, we decrease the rate of positive observations in $\mathcal{S}_1$ by τ_1 and increase it in $\mathcal{S}_2$ by τ_2. Further, we require that the total ratio of positive labels in the dataset remains unchanged after label flipping, that is, the number of labels flipped among the two subgroups is the same. Therefore, we can derive the following two equations for computing τ_1 and τ_2:

$$\left(\frac{p_1}{n_1} - \tau_1 \right) - \left(\frac{p_2}{n_2} + \tau_2 \right) = \epsilon, \quad \tau_2 n_2 = \tau_1 n_1. \tag{2}$$

After solving the equations we obtain the following:

$$\begin{aligned}\tau_1 &= \frac{n_2 p_1 - p_2 n_1 - n_1 n_2 \epsilon}{n_1(n_1+n_2)},\\ \tau_2 &= \frac{n_2 p_1 - p_2 n_1 - n_1 n_2 \epsilon}{n_2(n_1+n_2)}.\end{aligned} \tag{3}$$

We train a classifier on a modified dataset (through label flipping) by solving a "min-min" optimization problem that decides which labels to flip, while learning the model parameters. The problem is formulated as follows:

$$\hat{\boldsymbol{\theta}} \quad = \quad \operatorname*{argmin}_{\boldsymbol{\theta}\in\Theta} \min_{\boldsymbol{z}\in\mathcal{Z}_{\tau_1,\tau_2}} \frac{1}{n}\sum_{i=1}^{n} \ell(y_i(1-2z_i), \boldsymbol{x}_i, \boldsymbol{\theta}). \tag{4}$$

We note that with the proposed training of a classifier through label flipping, we aim to improve the out of sample demographic parity, see Section 3 for a definition, among the groups of the sensitive attribute.

Problem (4) can be solved efficiently with projected stochastic gradient descent, by alternating between updating the variables $\boldsymbol{\theta}$ and $\boldsymbol{z}$, while also projecting the latter in the set $\mathcal{Z}_{\tau_1,\tau_2}$. Let α, β denote the learning rate for the model parameters and binary variables, respectively. Let f denote the objective function of Problem (4). For a batch of data $C = \{\boldsymbol{x}_i, y_i\}_{i=1}^m$, we have the following update:

$$\begin{aligned}\tilde{\boldsymbol{\theta}} &\longleftarrow \boldsymbol{\theta} - \alpha \nabla_C f(\boldsymbol{\theta}),\\ \tilde{\boldsymbol{z}}_C &\longleftarrow \boldsymbol{z}_C - \beta \nabla f(\boldsymbol{z}_C).\end{aligned}$$

Observe that after each batch update $\boldsymbol{z}$ will no longer be binary at the batch coordinates and as a result at the end of an epoch $\tilde{\boldsymbol{z}} \notin \mathcal{Z}_{\tau_1,\tau_2}$. In order to satisfy the constraints we can project it in the set $\mathcal{Z}_{\tau_1,\tau_2}$ by solving the following optimization problem:

$$\begin{aligned}\operatorname{Proj}_{\mathcal{Z}_{\tau_1,\tau_2}}(\tilde{\boldsymbol{z}}) = \min_{\boldsymbol{z}} \quad & \|\tilde{\boldsymbol{z}} - \boldsymbol{z}\|_1 \\ \text{s.t.} \quad & \sum_{i\in\mathcal{S}_1} z_i = \lceil \tau_1 \cdot n_1 \rceil, \\ & \sum_{i\in\mathcal{S}_2} z_i = \lceil \tau_2 \cdot n_2 \rceil, \\ & z_i \in \{0,1\}.\end{aligned} \tag{5}$$

We measure distance with the ℓ_1 norm and thus obtain a MILO. We remark that in the period 1991–2015, algorithmic advances in MIO coupled with hardware improvements have resulted in an astonishing 450 billion factor speedup in solving MIO problems. As a result Problem (5) is practically solvable for datasets with sizes in the thousands. The proposed training of a classifier is summarized in pseudocode in Algorithm 1. The index t iterates over the epochs and $\mathcal{C}^t$ denotes the partition of training data in batches at epoch t.

Algorithm 1 Fair training of a classifier

```
Input: Training data $(\boldsymbol{X}, \boldsymbol{y})$, sensitive attribute $S$, initial guesses $\boldsymbol{\theta}^0$, $\boldsymbol{z}^0$, parameters $T$, $\alpha, \beta$, $\epsilon$.
Output: Optimal model parameters $\boldsymbol{\theta}^K$.
 1: Compute $\tau_1, \tau_2$ from (3).
 2: Initialize $\boldsymbol{\theta}^1, \boldsymbol{z}^1 = \boldsymbol{\theta}^0,\ \boldsymbol{z}^0$.
 3: Initialize $k = 1$.
 4: for $t = 1 : T$ do
 5:    for $c \in \mathcal{C}^t$ do
 6:       $\boldsymbol{\theta}^{k+1} \longleftarrow \boldsymbol{\theta}^k - \alpha \nabla_c f(\boldsymbol{\theta}^k)$.
 7:       $\tilde{\boldsymbol{z}}_c^{t+1} \longleftarrow \boldsymbol{z}_c^t - \beta \nabla f(\boldsymbol{z}_c^t)$.
 8:    end for
 9:    $\boldsymbol{z}^{t+1} = \mathrm{Proj}_{\mathcal{Z}_{\tau_1,\tau_2}}(\tilde{\boldsymbol{z}}^{t+1})$.
10: end for
11: Return $\boldsymbol{\theta}^K$.
```

2.2 Convergence analysis and modifications

In this section, we examine cases where Algorithm 1 converges to a stationary point of Problem (4). We make the following assumptions/adaptations:

Assumption 1 *The output of the neural network is bounded, that is, $|f_{\boldsymbol{\theta}}(\boldsymbol{x})| \leq M$.*

Adaptation 1 *The learning rate at each iteration of Algorithm 1 is obtained from the the Armijo linesearch algorithm (see Bonettini et al. (2016)).*

Adaptation 2 *At Step 9 of Algorithm 1, $\tilde{\boldsymbol{z}}$ is rounded to the closest integer before solving the projection problem.*

We note that Adaptation 2 is needed in order to obtain a linear objective in the projection problem and therefore optimize over the convex hull of the feasible region, that is,

$$\mathrm{conv}(\mathcal{Z}_{\tau_1,\tau_2}) = \left\{ \boldsymbol{0} \leq \boldsymbol{z} \leq \boldsymbol{1} : \sum_{i \in \mathcal{S}_1} z_i = \lceil \tau_1 \cdot n_1 \rceil,\ \sum_{i \in \mathcal{S}_2} z_i = \lceil \tau_2 \cdot n_2 \rceil \right\}.$$

Notice that in this case Algorithm 1 is a version of cyclic block coordinate descent with gradient projections over a convex set for which Bonettini et al. (2016) have established convergence to a stationary point, under Assumption 1 and Adaptation 1. We summarize the result in Theorem 1.

Theorem 1 (Convergence to stationary point) *Assume that Assumption 1 holds. Then, Algorithm 1 with the logistic loss, that is, $\ell(y, u) = \log(1 + \exp(-yu))$, and Adaptations 1 and 2, converges to a stationary point for Problem (4).*

Proof We first show that the objective is L-smooth in terms of $\boldsymbol{\theta}$. Let $\ell_i(\boldsymbol{\theta}) = \log(1 + \exp(-y_i f_{\boldsymbol{\theta}}(\boldsymbol{x}_i)))$. We have the following

$$\begin{aligned}
\|(\nabla_{\boldsymbol{\theta}} \ell_i(\boldsymbol{\theta}))\| &= \left\| \frac{-y_i \nabla_{\boldsymbol{\theta}} f_{\boldsymbol{\theta}}(\boldsymbol{x}_i) \exp(-y_i f_{\boldsymbol{\theta}}(\boldsymbol{x}_i))}{1 + \exp(-y_i f_{\boldsymbol{\theta}}(\boldsymbol{x}_i))} \right\| \\
&= \left| \frac{-y_i}{1 + \exp(y_i f_{\boldsymbol{\theta}}(\boldsymbol{x}_i))} \right| \|\nabla_{\boldsymbol{\theta}} f_{\boldsymbol{\theta}}(\boldsymbol{x}_i)\| \\
&\leq L,
\end{aligned}$$

where L denotes an upper bound on the Lipschitz constant of the gradient of the neural network. Thus, we obtain

$$\|\nabla_{\boldsymbol{\theta}} \ell(\boldsymbol{\theta})\| = \left\| \sum_{i=1}^{n} \nabla_{\boldsymbol{\theta}} \ell_i(\boldsymbol{\theta}) \right\| \leq \sum_{i=1}^{n} \|\nabla_{\boldsymbol{\theta}} \ell_i(\boldsymbol{\theta})\| \leq nL.$$

We next show that the objective is L-smooth in terms of $\boldsymbol{z}$. We have the following

$$\begin{aligned}
|(\nabla_{\boldsymbol{z}} \ell(\boldsymbol{\theta}, \boldsymbol{z}))_i| &= \left| \frac{2 y_i f_{\boldsymbol{\theta}}(\boldsymbol{x}_i) \exp(-y_i(1 - 2z_i) f_{\boldsymbol{\theta}}(\boldsymbol{x}_i))}{1 + \exp(-y_i(1 - 2z_i) f_{\boldsymbol{\theta}}(\boldsymbol{x}_i))} \right| \\
&= \left| \frac{1}{1 + \exp(y_i(1 - 2z_i) f_{\boldsymbol{\theta}}(\boldsymbol{x}_i))} \right| |2y_i| \, |f_{\boldsymbol{\theta}}(\boldsymbol{x}_i)| \\
&\leq 2M.
\end{aligned}$$

Therefore,

$$\|\nabla_{\boldsymbol{z}} \ell(\boldsymbol{z})\|^2 = \sum_{i=1}^{n} (\nabla_{\boldsymbol{z}} \ell(\boldsymbol{z}))_i^2 \leq \sum_{i=1}^{n} 4M^2 = n4M^2 \implies \|\nabla_{\boldsymbol{z}} \ell(\boldsymbol{z})\| \leq 2M\sqrt{n}.$$

Now observe that the constraints in the projection problem can be written as $\boldsymbol{A}\boldsymbol{z} = \boldsymbol{b}$. Since the indices of $\mathcal{S}_1, \mathcal{S}_2$ are disjoint, the vector containing the differences between the first and second row of $\boldsymbol{A}$ has entries in $\{1, -1\}$. Therefore, from Corollary 3.2 Bertsimas and Weismantel (2005) the matrix $\boldsymbol{A}$ is totally unimodular and as a result the polyhedron is integral. Moreover, for binary $\tilde{\boldsymbol{z}}$ the objective in the projection problem is

$$\sum_{i: \tilde{z}_i = 1} (1 - z_i) + \sum_{i: \tilde{z}_i = 0} z_i.$$

Since the objective is linear and the polyhedron is integral, we can equivalently optimize over $\operatorname{conv}(\mathcal{Z}_{\tau_1, \tau_2})$, which is convex. Therefore, from Theorem 1 in Bonettini et al. (2016), Algorithm 1 with the proposed modifications converges to a stationary point for Problem (4). ■

Herrera et al. (2020) showed that L, an upper bound on the Lipschitz constant of the gradient of the neural network, depends on many parameters including the maximum norm of the network input, the number of neurons on each layer and the maximum norm of the model parameters.

Observe that in the case of structured data, when the classification model is either logistic regression (LR) or support vector machines (SVM), Problem (4) admits an exact mixed integer formulation, which we provide in Appendix B. In both cases the resulting problems are mixed integer nonlinear optimization problems, which become computationally intractable for datasets with sizes in the thousands, whereas Algorithm 1 is still applicable in this case, as the numerical experiments on Section 4 illustrate. In addition, Algorithm 1 can also handle unstructured data, in which case the deployed model can be a deep neural network.

Finally, we propose some additional constraints to ensure that the labels of the less privileged class are being flipped from −1 to +1 and those of the more privileged class the other way around. In order to ensure that the labels in $\mathcal{S}_1$ are flipped from +1 to −1, we add the following constraint:

$$\sum_{i\in\mathcal{S}_1}(y_i+1)/2 \geq \sum_{i\in\mathcal{S}_1}(y_i(1-2z_i)+1)/2.$$

Similarly, in order to ensure that the labels in $\mathcal{S}_2$ are flipped from −1 to +1, we add the following constraint:

$$\sum_{i\in\mathcal{S}_2}(y_i+1)/2 \leq \sum_{i\in\mathcal{S}_2}(y_i(1-2z_i)+1)/2$$

In addition, we require that the total number of positive labels remains the same after flipping, that is,

$$\sum_{i=1}^{n}(y_i+1)/2 = \sum_{i=1}^{n}(y_i(1-2z_i)+1)/2.$$

Although with the additional linear constraints the feasible region in the projection problem is no longer integral and therefore we do not have a convergence guarantee, Algorithm 1 exhibits very good performance in practise.

We note that the proposed framework for structured data has been utilized in a case study for alleviating racial bias in patient discharge disposition classification (home vs. post-acute care (PAC)), see Gebran et al. (2023). The results indicated that without the proposed framework 21.5% of white patients and 12.1% of black patients were discharged to PAC, while with the proposed framework 15.9% of both white and black patients had a recommended discharge to PAC.

2.3 Meritocracy constraints for structured data

Often we want to achieve fairness in terms of the sensitive attribute, while also satisfying constraints on selected objective measures of meritocracy. For example, in the college admission process, we want to increase the number of students admitted from the less represented class, while maintaining a satisfactory average GPA among the admitted students. In practice, these two goals are in conflict and we observe an inherent trade-off between them. Thus, we are in grave need of a systematic approach that alleviate bias without significantly affecting meritocracy. We propose training a classifier based on Algorithm 1, while placing constraints on some selected objective measures of meritocracy $\mathcal{M}$, for example undergraduate GPA, LSAT scores, etc. For each covariate in $\mathcal{M}$, the meritocracy constraints restrict moments of the distribution of positive observations in the modified dataset to change at most by a fraction δ (meritocracy tolerance) of the moments in the original dataset. Recall that p_1, p_2 are the number of positive observations in $\mathcal{S}_1$ and $\mathcal{S}_2$, respectively. Let μ_j denote the first moment of attribute $j \in \mathcal{M}$ among the positive observations in the original dataset, that is,

$$\mu_j = \frac{\sum_{i=1}^n x_{ij}\,(y_i + 1)}{2(p_1 + p_2)}.$$

Note that since $y_i \in \{-1, 1\}$, it follows that $(y_i + 1)/2 \in \{0, 1\}$. Similarly, the first moment of attribute $j \in \mathcal{M}$ among positive observations in the modified dataset is given by

$$\tilde{\mu}_j = \frac{\sum_{i=1}^n x_{ij}\,(y_i(1 - 2z_i) + 1)}{\sum_{i=1}^n (y_i(1 - 2z_i) + 1)}$$

We constraint the first moment to change at most by a fraction δ of the initial value, that is, $|\tilde{\mu}_j - \mu_j| \leq \delta\mu_j$, which is formulated as follows:

$$\left| \frac{\sum_{i=1}^n x_{ij}\,(y_i(1 - 2z_i) + 1)}{\sum_{i=1}^n (y_i(1 - 2z_i) + 1)} - \mu_j \right| \leq \delta\mu_j,\ j \in \mathcal{M}.$$

In order to better match the distribution of positive labels in the original and modified datasets, we also place constraints on the second moments of the merit covariates. Let μ_j^2 denote the second moment of attribute $j \in \mathcal{M}$ among the positive observations in the original dataset, that is,

$$\mu_j^2 = \frac{\sum_{i=1}^n x_{ij}^2\,(y_i + 1)}{2(p_1 + p_2)}$$

Similarly, the second moment of attribute $j \in \mathcal{M}$ among the positive observations in the modified dataset is given by

$$\tilde{\mu}_j^2 = \frac{\sum_{i=1}^n x_{ij}^2\,(y_i(1 - 2z_i) + 1)}{\sum_{i=1}^n (y_i(1 - 2z_i) + 1)}.$$

We then require that the second moment can change at most by a fraction δ of the initial value, that is, $|\tilde{\mu}_j^2 - \mu_j^2| \leq \delta\mu_j^2$, which is formulated as follows:

$$\left| \frac{\sum_{i=1}^n x_{ij}^2\,(y_i(1 - 2z_i) + 1)}{\sum_{i=1}^n (y_i(1 - 2z_i) + 1)} - \mu_j^2 \right| \leq \delta\mu_j^2,\ j \in \mathcal{M}.$$

Observe that both first and second moment constraints are linear in the binary variables $\boldsymbol{z}$, thus the projection problem remains a MILO. Note that it is also possible to impose constraints on the Wasserstrein distance of the distributions of the merit covariates among positive observations between the two datasets. However, in this case the projection problem becomes much harder since the number of variables and constraints increases significantly, see Appendix C for the entire formulation.

2.4 Extensions to multiple and multi-valued sensitive attributes

In this section, we demonstrate how our method can be extended to the case of multiple sensitive attributes as well as one multi-valued sensitive attribute.

Multiple sensitive attributes. We first assume that we have two sensitive attributes $\mathcal{S}^1, \mathcal{S}^2$ and that we want to flip labels in order to simultaneously equalize the rate of positive observations in the two classes of each sensitive attribute. Without loss of generality we assume that class 1 is privileged in both sensitive attributes. Now, observe that we can only flip labels for observations that are positive/negative for both sensitive attributes. Let $\left|\mathcal{S}_1^1 \cap \mathcal{S}_1^2\right| = \overline{n}_1, \left|\mathcal{S}_2^1 \cap \mathcal{S}_2^2\right| = \overline{n}_2$. We obtain the following constraint set:

$$\mathcal{Z}_{\tau_1,\tau_2} = \left\{ \boldsymbol{z} \in \{0,1\}^n : \sum_{i \in \mathcal{S}_1^1 \cap \mathcal{S}_1^2} z_i = \lceil \tau_1 \cdot \overline{n}_1 \rceil, \; \sum_{i \in \mathcal{S}_2^1 \cap \mathcal{S}_2^2} z_i = \lceil \tau_2 \cdot \overline{n}_2 \rceil \right\},$$

where τ_1, τ_2 are obtained from equations (2). Note that $\overline{n}_1 + \overline{n}_2 \neq n$. Observe that by equalizing the ratio of positive observations between the more/less privileged groups we are also equalizing them with respect to the other two subpopulations whose ratio of positive observations is in between. In case we have k sensitive attributes, where $k > 2$, assuming each one with class 1 as the privileged class, we obtain the following constraint set:

$$\mathcal{Z}_{\tau_1,\tau_2} = \left\{ \boldsymbol{z} \in \{0,1\}^n : \sum_{i \in \mathcal{S}_1^1 \cap \ldots \mathcal{S}_1^k} z_i = \lceil \tau_1 \cdot \overline{n}_1 \rceil, \; \sum_{i \in \mathcal{S}_2^1 \cap \ldots \mathcal{S}_2^k} z_i = \lceil \tau_2 \cdot \overline{n}_2 \rceil \right\},$$

where $\left|\mathcal{S}_1^1 \cap \ldots, \mathcal{S}_1^k\right| = \overline{n}_1, \left|\mathcal{S}_2^1 \cap \ldots \mathcal{S}_2^k\right| = \overline{n}_2$. Observe that in this case we have fewer options for label flipping and thus we might not be able to equalize the ratio of positive observations among the classes of a sensitive attribute by as much as only considering that attribute. However, we are able to equalize the ratio of positive observations among the intersection of classes of different sensitive attributes.

Multi-valued sensitive attribute. We further treat the case of one multi-valued sensitive attribute. We first assume that we have 3 classes, $\mathcal{S}_1, \mathcal{S}_2, \mathcal{S}_3$ where $|\mathcal{S}_1| = n_1, |\mathcal{S}_2| = n_2, |\mathcal{S}_3| = n_3$ and that 1 is the privileged class. We decrease the rate of positive observations in class 1 by τ_1 and increase it in classes $2, 3$ by τ_2, τ_3. We obtain the following constraint set:

$$\mathcal{Z}_{\tau_1,\tau_2,\tau_3} = \left\{ \boldsymbol{z} \in \{0,1\}^n : \sum_{i \in \mathcal{S}_1} z_i = \lceil \tau_1 \cdot n_1 \rceil, \; \sum_{i \in \mathcal{S}_2} z_i = \lceil \tau_2 \cdot n_2 \rceil, \; \sum_{i \in \mathcal{S}_3} z_i = \lceil \tau_3 \cdot n_3 \rceil \right\}.$$

We assume that we want to equalize the rate of positive observations between class 2 and class 1 by ϵ_2 and those between class 3 and class 1 by ϵ_3. We can determine τ_1, τ_2, τ_3 from the following equations:

$$\begin{aligned}
&\left(\frac{p_1}{n_1} - \tau_1\right) - \left(\frac{p_2}{n_2} + \tau_2\right) = \epsilon_2, \\
&\left(\frac{p_1}{n_1} - \tau_1\right) - \left(\frac{p_3}{n_3} + \tau_3\right) = \epsilon_3, \\
&\tau_1 n_1 = \tau_2 n_2 = \tau_3 n_3.
\end{aligned}$$

Substituting the values $\tau_1 = \frac{n_2}{n_1}\tau_2$, $\tau_1 = \frac{n_3}{n_1}\tau_3$ in the first and second equation, respectively, we obtain

$$\tau_2 = \frac{n_2 p_1 - n_1 p_2 - n_1 n_2 \epsilon_2}{n_2(n_1+n_2)}, \quad \tau_3 = \frac{n_3 p_1 - n_1 p_3 - n_1 n_3 \epsilon_3}{n_3(n_1+n_3)}.$$

Moreover, we have the equations $\tau_1 = \frac{n_2}{n_1}\tau_2 = \frac{n_3}{n_1}\tau_3$ from which we can obtain $\epsilon_3 = f(\epsilon_2)$, where $f(\cdot)$ denotes a closed form expression. Observe that in this case we only have freedom to choose one of the parameters ϵ_2, ϵ_3.

In the general case we assume that we have k classes, $\mathcal{S}_1, \ldots, \mathcal{S}_k$, where $|\mathcal{S}_1| = n_1, \ldots, |\mathcal{S}_k| = n_k$. Without loss of generality we assume that class 1 is the privileged class. We then decrease the rate of positive observations in class 1 by τ_1 and increase it in class i by τ_i, $i \geq 2$. We obtain the following constraint set:

$$\mathcal{Z}_{\tau_1,\ldots,\tau_k} = \left\{ \boldsymbol{z} \in \{0,1\}^n : \sum_{i\in\mathcal{S}_1} z_i = \lceil \tau_1 \cdot n_1 \rceil, \ \ldots, \ \sum_{i\in\mathcal{S}_k} z_i = \lceil \tau_k \cdot n_k \rceil \right\}.$$

We assume that we want to equalize the rate of positive observations between class i and class 1 by ϵ_i, where $i \geq 2$. We can determine $\tau_1, \ldots, \tau_k$ from the following equations:

$$\begin{aligned}
&\left(\frac{p_1}{n_1} - \tau_1\right) - \left(\frac{p_2}{n_2} + \tau_2\right) = \epsilon_2,\\
&\left(\frac{p_1}{n_1} - \tau_1\right) - \left(\frac{p_3}{n_3} + \tau_3\right) = \epsilon_3,\\
&\ldots,\\
&\left(\frac{p_1}{n_1} - \tau_1\right) - \left(\frac{p_k}{n_k} + \tau_k\right) = \epsilon_k,\\
&\tau_1 n_1 = \tau_2 n_2 = \ldots = \tau_k n_k.
\end{aligned}$$

In this case, we also have freedom to choose only one of the parameters ϵ_i, as Lemma 2 illustrates.

Lemma 2 (Choosing parameters for a multi-valued sensitive attribute) *Assume we have a multi-valued sensitive attribute $\mathcal{S}$, with k classes $\mathcal{S}_1, \ldots, \mathcal{S}_k$, where $|\mathcal{S}_1| = n_1, \ldots, |\mathcal{S}_k| = n_k$ and class 1 is the privileged class. When decreasing the rate of positive observations in class 1 by τ_1 and increasing it in class i, $i \geq 2$, by τ_i in order to equalize the rate of positive observations by ϵ_i, we only have freedom to choose one of the parameters $\epsilon_2, \ldots, \epsilon_k$.*

Proof If we substitute the value $\tau_1 = \frac{n_i}{n_1}\tau_i$ in the i-th equation we obtain the following solutions:

$$\tau_2 = \frac{n_2 p_1 - n_1 p_2 - n_1 n_2 \epsilon_2}{n_2(n_1+n_2)}, \ \ldots, \ \tau_k = \frac{n_k p_1 - n_1 p_k - n_1 n_k \epsilon_k}{n_k(n_1+n_k)}$$

Moreover, from the additional equations $\tau_1 = \frac{n_2}{n_1}\tau_2 = \ldots = \frac{n_k}{n_1}\tau_k$, we obtain:

$$\begin{aligned}
&\frac{n_2}{n_1}\tau_2 = \frac{n_3}{n_1}\tau_3 \implies \epsilon_3 = f_3(\epsilon_2),\\
&\ldots,\\
&\frac{n_2}{n_1}\tau_2 = \frac{n_k}{n_1}\tau_k \implies \epsilon_k = f_k(\epsilon_2),
\end{aligned}$$

where $f_3(\cdot), \ldots, f_k(\cdot)$ denote closed form expressions. Since the choice of ϵ_2 was random, the result holds for any other choice. We thus observe that it if we pick one of the parameters ϵ_i, the other ones are obtained in closed form in order to satisfy the equations. ■

Finally, we note that many existing methods for alleviating bias do not extend to the cases of multiple or multi-valued sensitive attributes, see Table 1 in Zafar et al. (2017), in which case our approach makes a significant contribution.

3. Quantifying fairness and meritocracy

The main quantities used for evaluating fairness are Demographic Parity and Equalized Odds, defined as follows (see Wang et al. (2022)):

Definition 3 (Demographic Parity) *A classifier satisfies demographic parity if the value of the sensitive attribute s cannot influence assigning a positive label, that is,*

$$\mathbb{P}(\hat{y} = 1 \mid s = 1) \ = \ \mathbb{P}(\hat{y} = 1 \mid s = 2).$$

Definition 4 (Equalized Odds) *A classifier satisfies equalized odds if the value of the sensitive attribute s cannot influence assigning a positive label given y, that is,*

$$\mathbb{P}(\hat{y} = 1 \mid y, s = 1) \ = \ \mathbb{P}(\hat{y} = 1 \mid y, s = 2), \ y \in \{-1, 1\}.$$

We measure demographic parity, with Statistical Parity Difference (SPD):

$$\text{SPD } = \mid \mathbb{P}(\hat{y} = 1 \mid s = 1) - \mathbb{P}(\hat{y} = 1 \mid s = 2) \mid,$$

and equalized odds with Difference in Equalized Odds (DEO):

$$\text{DEO } = \mid \mathbb{P}(\hat{y} = 1, \mid y = y, s = 1) - \mathbb{P}(\hat{y} = 1 \mid y = y, s = 2) \mid, \ y \in \{-1, 1\}.$$

For each class s of the sensitive attribute, the True Positive Rate (TPR) / False Positive Rate (FPR), is defined as the probability of predicting a positive label when the true label within s is positive / negative, that is,

$$\begin{aligned} \text{TPR}_s \ &= \ \mathbb{P}(\hat{y} = 1 \mid y = 1, s = s), \ \ s \in \{1, 2\}, \\ \text{FPR}_s \ &= \ \mathbb{P}(\hat{y} = 1 \mid y = -1, s = s), \ \ s \in \{1, 2\}. \end{aligned}$$

We also report the Equal Opportunity Difference (EOD), which measures the difference in TPR among the classes of the sensitive attribute and it is defined as follows:

$$\text{EOD } = \mid \text{TPR}_1 - \text{TPR}_2 \mid .$$

Finally, for structured data $\{\boldsymbol{x}_i, y_i\}_{i=1}^m$ and predicted labels $(\hat{y}_i)_{i=1}^m$, we define the merit metric for attribute k, denoted as Merit(k), as the distributional distance of attribute k among positive observations between the dataset with the true labels and that with the predicted labels. We use the Wasserstrein distance with the ℓ_1 norm to measure distributional distance. More precisely, let $(\alpha_1, \ldots, \alpha_{m_1})$ be the samples of covariate k among positive observations in the dataset with the true labels and $(\beta_1, \dots, \beta_{m_2})$ be those in the dataset with the predicted labels. Then, Merit(k) is defined as the distributional distance between the two samples and can be obtained from the solution of a linear program:

$$\begin{aligned}
\text{Merit}(k) \;=\; \min_{\boldsymbol{\gamma}} \quad & \sum_{i=1}^{m_1}\sum_{j=1}^{m_2} \gamma_{ij}|\alpha_i - \beta_j| & \\
\text{s.t.} \quad & \sum_{j=1}^{m_2} \gamma_{ij} = 1/m_1, & i \in [m_1], \\
& \sum_{i=1}^{m_1} \gamma_{ij} = 1/m_2, & j \in [m_2], \\
& \gamma_{ij} \geq 0, & i \in [m_1], j \in [m_2].
\end{aligned}$$

In the numerical experiments the merit attributes are chosen based on the intuition that their value is indicative for the target label, which is further supported with a statistical hypothesis test showing that the difference of the average value among positive and negative observations in the training set is statistically significant.

4. Numerical experiments

Training Details. For tabular data classification, we use LR as the base architecture and for image classification we use the ResNet-18 He et al. (2016) as the base architecture. Throughout we use the the soft margin loss, along with the Adam optimizer Kingma and Ba (2015) and a batch size of 64. We tune the learning rate and the number of epochs for both our approach, the nominal model and the benchmarks (if applicable) on a validation set based on the SPD metric. Throughout we fix ϵ = 1e-2 for our method. We initialize Algorithm 1 with a random feasible solution $\boldsymbol{z}^0$ obtained as $\boldsymbol{z}^0 = \text{Proj}_{\mathcal{Z}_{\tau_1,\tau_2}}(\mathbf{0})$. All experiments are conducted in Python using Pytorch Paszke et al. (2017) for model training and Gurobi Optimization, Inc. (2017) (Gurobi) for solving Problem (5). We utilize the train/validation/test splits from the Python package ethicML (2022) (EthicML) for structured data and those from Pytorch for unstructured data. All results are averaged over 10 different random seeds.

4.1 Structured data classification

Experiments setting We benchmark our method with the nominal model, that is, a classification model without taking fairness into account, as well as state-of-the art methods for improving fairness in classification. Those include the pre-processing label massaging method from Calders et al. (2009), the Reweighting method from Kamiran and Calders (2012), the in-processing methods from Zafar et al. (2017) and Agarwal et al. (2018) and the more recently developed DRO-based method by Hashimoto et al. (2018). For our approach, we try the values $\alpha, \beta \in \{1\text{e-}3, 1\text{e-}2, 1\text{e-}1\}$ and $T \in \{20, 50, 100\}$. Further, for the benchmarks that include parameters, we tune them on a validation set. We apply the method from Zafar et al. (2017) with fairness objective and accuracy constraints and vice versa and report the best results in terms of out of sample SPD. For the parameter γ, which corresponds to the maximum amount allowed for loss reduction when optimizing for fairness, we try the values $\{0.01, 0.1, 0.5, 1.0, 10\}$. For Hashimoto et al. (2018) we utilize the same values for the learning rate and number of epochs as in our method. We utilize EthicML for all benchmark implementations and the Python package *scipy* for computing the Wasserstrein distance, that defines the merit metric. We evaluate all methods in terms of accuracy, fairness and meritocracy on a held out test set. We refer to Appendix A for the standard deviations. A value of 0.000 in the standard deviation indicates that it was less than 10^{-4}. Algorithm 1 is applied with meritocracy tolerance $\delta \in \{0.1, 0.5\}$, as well as without the meritocracy constraints (No Merit). The value $\delta = 0.1$ indicates a strong meritocracy requirement in the optimization, while the value $\delta = 0.5$ indicates a moderate meritocracy requirement.

We remark that the major contribution of our method is improving demographic parity, while the other fairness measures used in the evaluations are for referential comparisons. The methods by Calders et al. (2009) and Kamiran and Calders (2012) also optimize for demographic parity. Further, Hashimoto et al. (2018) aims to improve the accuracy of the classifier in the minority class of the sensitive attribute, while Zafar et al. (2017) aims to improve the ratio of positive predictions among the classes of the sensitive attribute and as a result optimizes demographic parity. Finally, the method by Agarwal et al. (2018) can improve demographic parity and equalized odds.

4.1.1 LSAC dataset

The first dataset that we consider is the LSAC dataset. This dataset originates from a longitudinal bar passage study by Wightman (1998) from 1991-1997 investigating whether the bar exam taken by law students in the US is biased against ethnic minorities. The dataset contains anonymized historical information about the law students who participated in this study resulting in total 20,798 observations and 11 features including age, LSAT score (*lsat*), first year law school GPA (*zfygpa*), cumulative law school GPA and undergraduate GPA(*ugpa*). The target is *acceptance*, indicating the bar exam decision, the sensitive attribute is *race* and the merit attributes are *lsat* and *ugpa*. From Table 1 we observe that the differences in the average values of the merit attributes among positive and negative observations in the training set are statistically significant.

Table 1: Average values of the merit attributes for the LSAC dataset, among positive/negative observations in the training set and the p-value of the T-test with null hypothesis of equal average values.

	Avg Pos	Avg Neg	p-value
lsat	37.31	32.18	$< 10^{-5}$
ugpa	3.25	3.01	$< 10^{-5}$

The dataset is biased more towards the white subpopulation which has ratio $p_1/n_1 = 0.921$, while the non-white subpopulation has ratio $p_2/n_2 = 0.718$. We utilize 14,569 observations for training, $1,868$ for validation and $4,361$ for testing. Moreover, we have

$$\left|\frac{p_1^{TRAIN}}{n_1^{TRAIN}} - \frac{p_2^{TRAIN}}{n_2^{TRAIN}}\right| = 0.203, \quad \left|\frac{p_1^{TEST}}{n_1^{TEST}} - \frac{p_2^{TEST}}{n_2^{TEST}}\right| = 0.185.$$

The benchmark results are illustrated in Table 2.

Table 2: Out of sample accuracy, fairness and meritocracy for the LSAC dataset, with *race* as the sensitive attribute.

	Acc ↑	SPD ↓	EOD ↓	DEO ↓	Merit(*lsat*) ↓	Merit(*ugpa*) ↓
Nominal	0.886	0.299	0.207	0.282	0.239	0.263
Calders	0.884	0.019	0.019	0.084	0.229	0.248
Kamiran	0.899	0.013	**0.004**	**0.003**	0.239	0.259
Zafar	0.889	0.015	**0.004**	0.022	0.205	0.223
Agarwal	**0.903**	0.109	0.045	0.137	0.183	0.199
Hashimoto	**0.903**	0.151	0.072	0.206	0.181	0.197
Ours($\delta = 0.1$)	0.893	0.072	0.022	0.037	**0.089**	**0.107**
Ours($\delta = 0.5$)	0.892	0.026	0.017	0.056	0.156	0.166
Ours(No Merit)	0.890	**0.011**	0.018	0.068	0.181	0.191

From Table 2, we observe that our approach, when applied without the meritocracy constraints, achieves the best out of sample SPD. Further, when applied with the meritocracy constraints for $\delta = 0.1$, it ranks first in terms of Merit(*lsat*) and Merit(*ugpa*), while for $\delta = 0.5$ it achieves a good trade-off between fairness and meritocracy. We notice an overall improvement in out of sample accuracy from the nominal model, which follows from the fact that the test data are slightly less biased than the training data.

4.1.2 Crime dataset

The second dataset that we consider is the Communities and Crime dataset from the UCI ML Repository Dua and Graff (2017). It contains $1,993$ observations corresponding to communities in the United States described by 136 features, and the per capita crime rate for each community. The communities with a crime rate above the 70-th percentile are labeled as ‘high crime’ and the others as ‘low crime’. The dataset contains many attributes, including the percent of the population under poverty (*pct-under-pov*), the percent of the population that is unemployed (*pct-unemp*), the median income and the number of illegal immigrants. The target is *highCrime*, indicating whether the crime rate in the community is high, the sensitive attribute is *race* and the merit attributes are *pct-under-pov* and *pct-unemployed.* From Table 3 we observe that the differences in the average values of the merit attributes among positive and negative observations in the training set are statistically significant.

Table 3: Average values of the merit attributes for the Crime dataset, among positive/negative observations in the training set and the p-value of the T-test with null hypothesis of equal average values.

	Avg Pos	Avg Neg	p-value
pct-under-pov	0.459	0.224	$< 10^{-5}$
pct-unemp	0.501	0.294	$< 10^{-5}$

The dataset is biased more towards the non-white subpopulation which has ratio $p_1/n_1 = 0.527$, while the white subpopulation has ratio $p_2/n_2 = 0.125$. We utilize $1,594$ observations for training, 119 for validation and 280 for testing. Moreover, we have

$$\left|\frac{p_1^{TRAIN}}{n_1^{TRAIN}} - \frac{p_2^{TRAIN}}{n_2^{TRAIN}}\right| = 0.402, \quad \left|\frac{p_1^{TEST}}{n_1^{TEST}} - \frac{p_2^{TEST}}{n_2^{TEST}}\right| = 0.424.$$

The benchmark results are illustrated in Table 4.

Table 4: Out of sample accuracy, fairness and merit metrics for the Crime dataset, with *race* as the sensitive attribute.

	Acc ↑	SPD ↓	EOD ↓	DEO ↓	Merit(*pct-under-pov*) ↓	Merit(*pct-unemp*) ↓
Nominal	**0.779**	0.296	0.249	0.164	0.026	0.011
Calders	0.732	0.121	0.269	0.213	0.039	0.011
Kamiran	0.775	0.195	0.027	**0.009**	0.037	0.007
Zafar	0.757	0.232	0.026	0.043	0.026	0.012
Agarwal	0.771	0.221	0.043	0.039	0.029	0.006
Hashimoto	0.767	0.398	0.255	0.219	0.045	0.006
Ours($\delta = 0.1$)	0.728	0.213	0.191	0.136	**0.023**	**0.003**
Ours($\delta = 0.5$)	0.689	0.107	0.019	0.041	0.029	0.005
Ours(No Merit)	0.678	**0.093**	**0.017**	0.038	0.037	0.007

From Table 4, we observe that our method, when applied without the meritocracy constraints, achieves the best out of sample SPD and EOD, while it ranks second in terms of DEO. Further, when applied with the meritocracy constraints for $\delta = 0.1$, it ranks first in terms of Merit(*pct-under-pov*) and Merit(*pct-unemp*), while for $\delta = 0.5$ it achieves a good trade-off between fairness and meritocracy. More precisely, we note that for $\delta = 0.5$, our approach outperforms the state-of-the-art methods in terms of SPD and EOD, while only worsening Merit(*pct-under-pov*) by 0.006 compared to $\delta = 0.1$. Finally, we observe that the improvement in fairness in this case comes with a loss in accuracy, which is more pronounced for our method. Observe that in this case, the test data are slightly more biased that the training data.

4.1.3 COMPAS dataset

The third dataset that we consider is the ProPublica's COMPAS dataset Angwin et al. (2016). In this dataset, the task is recidivism classification (high/low), based on criminal history, prison time and demographics. The dataset contains $6,167$ observations with 400 attributes including the defendants' age, race, sex, number of prior convictions (*priors*) and COMPAS assigned risk scores. The target is *highCrime*, indicating whether the risk score for recidivism is high, the sensitive attribute is *race* and the merit attribute is *priors*. From Table 5 we observe that the differences in the average values of the merit attribute among positive and negative observations in the training set are statistically significant.

Table 5: Average value of the merit attribute for the COMPAS dataset, among positive/negative observations in the training set and the p-value of the T-test with null hypothesis of equal average values.

	Avg Pos	Avg Neg	p-value
priors	4.71	1.98	$< 10^{-5}$

The dataset is biased towards the non-white subpopulation which has ratio $p_1/n_1 = 0.48$, while the white subpopulation has ratio $p_2/n_2 = 0.35$. We utilize $4,933$ observations for training, 370 for validation and 864 for testing. Moreover, we have

$$\left|\frac{p_1^{TRAIN}}{n_1^{TRAIN}} - \frac{p_2^{TRAIN}}{n_2^{TRAIN}}\right| = 0.129, \quad \left|\frac{p_1^{TEST}}{n_1^{TEST}} - \frac{p_2^{TEST}}{n_2^{TEST}}\right| = 0.106.$$

The benchmark results are illustrated in Table 6.

Table 6: Out of sample accuracy, fairness and merit metrics for the COMPAS dataset, with *race* as the sensitive attribute.

	Acc ↑	SPD ↓	EOD ↓	DEO ↓	Merit(*priors*) ↓
Nominal	0.652	0.141	0.195	0.129	0.048
Calders	0.544	**0.074**	0.128	**0.069**	0.112
Kamiran	**0.679**	0.118	0.192	0.106	0.073
Zafar	0.554	0.097	0.139	0.095	0.091
Agarwal	0.669	0.159	0.256	0.151	0.075
Hashimoto	0.658	0.115	0.213	0.119	0.086
Ours($\delta = 0.1$)	0.643	0.106	0.158	0.098	**0.034**
Ours($\delta = 0.5$)	0.653	0.101	0.138	0.090	0.059
Ours(No Merit)	0.659	0.098	**0.122**	0.088	0.076

From Table 6, we observe that the label massaging approach from Calders achieves the best performance in terms of SPD and DEO, however it incurs a significant decrease in accuracy. On the other hand, our approach achieves the best performance in terms of EOD, when applied without the meritocracy constraints. Further, when applied with the meritocracy constraints with $\delta = 0.1$, our approach achieves the best performance in terms of Merit(*priors*). We also observe that our method, when applied without the meritocracy constraints, ranks third in terms of SPD and second in terms of DEO. Further, we notice that when applied with the meritocracy constraints for $\delta = 0.5$, our approach ranks second in terms of EOD and at the same time it outperforms all methods apart from the nominal model in terms of Merit(*priors*).

4.1.4 German credit dataset

The final dataset that we consider is the German credit dataset from the UCI ML Repository Dua and Graff (2017). This dataset classifies people as good or bad credit risks. It contains in total $1,000$ observations and 58 features including age, credit history, employment and the current amount of credit (*credit*). The target is *credit-label*, indicating the classification of the person as good or bad credit risk, the sensitive attribute is *gender* and the merit attribute is *credit*. From Table 7 we observe that the difference in the average values of the merit attribute among positive and negative observations in the training set is statistically significant.

Table 7: Average value of the merit attribute for the German credit dataset, among positive/negative observations in the training set and the p-value of the T-test with null hypothesis of equal average values.

	Avg Pos	Avg Neg	p-value
credit	0.28	-0.12	$< 10^{-5}$

The dataset is biased more towards the male subpopulation which has ratio $p_1/n_1 = 0.331$, while the female subpopulation has ratio $p_2/n_2 = 0.291$. We utilize 800 observations for training, 60 for validation and 140 for testing. Moreover, we have

$$\left|\frac{p_1^{TRAIN}}{n_1^{TRAIN}} - \frac{p_2^{TRAIN}}{n_2^{TRAIN}}\right| = 0.040, \ \left|\frac{p_1^{TEST}}{n_1^{TEST}} - \frac{p_2^{TEST}}{n_2^{TEST}}\right| = 0.043.$$

The benchmark results are illustrated in Table 8.

From Table 8, we observe that our approach, when applied without the meritocracy constraints, achieves the best performance in terms of SPD and EOD. Moreover, we notice that for $\delta = 0.1$, our method ranks first in terms of Merit(*credit*), while for $\delta = 0.5$ it achieves a good trade-off between fairness and meritocracy. More precisely, it ranks third in terms of Merit(*credit*) and first in terms of EOD. In this dataset, we observe a decrease in accuracy from the nominal model, which follows from the fact that the test data are slightly more biased that the training data.

Table 8: Out of sample accuracy, fairness and merit metrics for the German credit dataset, with *gender* as the sensitive attribute.

	Acc ↑	SPD ↓	EOD ↓	DEO ↓	Merit(*credit*) ↓
Nominal	**0.732**	0.035	0.036	**0.027**	0.016
Calders	0.716	0.018	0.051	0.037	0.145
Kamiran	0.714	0.012	0.044	**0.027**	0.128
Zafar	0.664	0.041	0.027	0.030	0.033
Agarwal	0.728	0.028	0.076	0.055	0.019
Hashimoto	0.671	0.067	0.087	0.055	0.038
Ours($\delta = 0.1$)	0.729	0.018	0.049	0.037	**0.015**
Ours($\delta = 0.5$)	0.720	0.007	**0.026**	0.036	0.017
Ours(No Merit)	0.721	**0.006**	**0.026**	0.033	0.019

4.1.5 The price of fairness

In this section, we compare the average values of the merit covariates when predicting a positive label in our method in comparison with the nominal model. From Table 9, we observe that the average values of the merit attributes among positive labels do not differ significantly between our method and the nominal model, signifying that the price of fairness is small. More specifically, in the LSAC dataset we observe that the average values of the merit covariates for students with positive outcome labels in the white subpopulation do not change significantly after employing Algorithm 1. Moreover, we notice that the average values of *lsat* for the positively labeled students in the non-white subpopulation decrease, signifying a lowering of thresholds for passing the bar exam for non-white students. Further, in the Crime dataset, we observe that in the non-white subpopulation the threshold for predicting a positive label increases in both merit attributes, while in the white subpopulation it increases for $\delta = 0.1$ but decreases for $\delta = 0.5$. Finally, we notice that as we increase δ, the average values of the merit covariates in both groups decrease, since in this case Algorithm 1 prioritizes fairness over meritocracy.

4.1.6 The effect of δ on fairness and meritocracy

In this section, we illustrate the trade-offs between out of sample fairness and meritocracy. First, we show the behavior of the out of sample merit metric with the meritocracy tolerance δ. For each dataset, we vary $\delta \in [0, 1]$ and report the merit metric on the test set for the merit attributes considered in Section 4.1. From Figure 1, we observe that, as expected, the out of sample merit metric follows a monotonic behavior with δ. It takes the smallest value for $\delta = 0.1$ and then increases as δ also increases. Moreover, we notice that when δ increases from 0.6 to 1, the out of sample merit metric does not change significantly, especially in the Crime and German credit datasets.

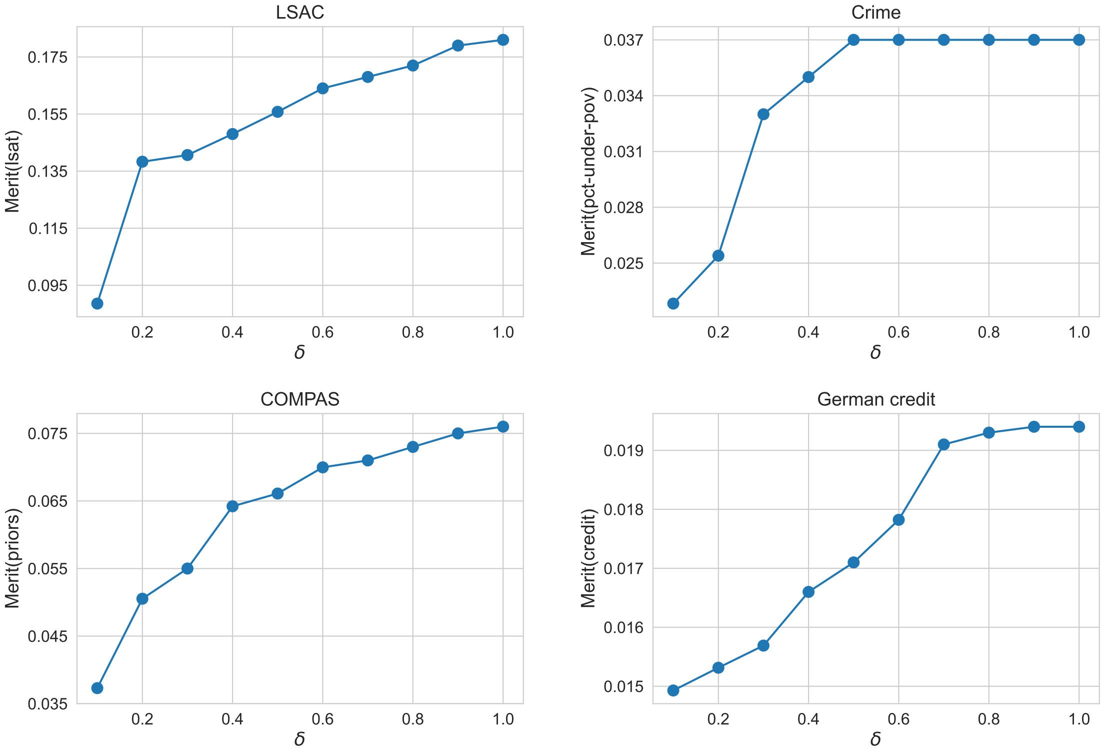

Figure 1: Out of sample merit metric with δ. The y axis represents the merit metric and the x axis the meritocracy tolerance δ.

Table 9: Average values of the merit attributes among positive outcomes, for each class of the sensitive attribute, for the actual test data (Data), the nominal model (Nominal) and Algorithm 1 with meritocracy tolerance δ (Ours(δ)). The majority group corresponds to the white subpopulation in LSAC, to the non-white subpopulation in Crime and COMPAS and to the male subpopulation in German.

Dataset		Majority group				Minority group			
	Attribute	Data	Nominal	Ours($\delta = 0.5$)	Ours($\delta = 0.1$)	Data	Nominal	Ours($\delta = 0.5$)	Ours($\delta = 0.1$)
LSAC	*lsat*	37.701	37.689	37.611	37.656	34.588	35.135	35.077	35.106
	ugpa	3.271	3.264	3.266	3.268	3.094	3.115	3.121	3.125
Crime	*pct-under-pov*	0.390	0.726	0.775	0.787	0.502	0.577	0.529	0.646
	pct-unemployed	0.476	0.821	0.875	0.888	0.512	0.595	0.592	0.643
COMPAS	*priors*	2.655	8.07	7.552	7.145	5.258	11.097	10.109	9.628
German	*credit*	0.316	2.787	1.989	2.102	-0.165	1.565	1.598	1.601

Next, in Figure 2, we illustrate the trade-off between the out of sample merit metric and SPD. The experimental setup is the same as in Figure 1. We observe that the best performing SPD corresponds to the worst performing merit metric and vice versa. Moreover, we notice that in the LSAC and COMPAS datasets the relationship is almost linear.

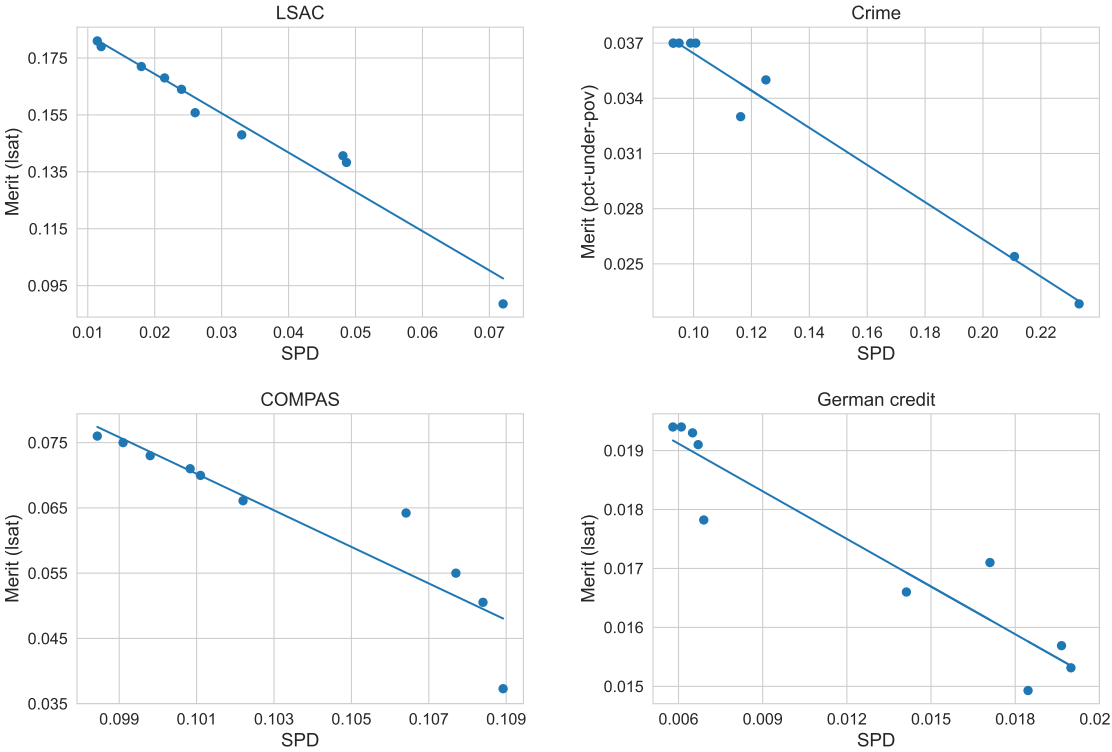

Figure 2: Out of sample merit metric and SPD. The y axis represents the out of sample merit metric and the x axis the out of sample SPD, for various $\delta \in [0, 1]$.

4.2 Unstructured data classification

Experiments setup We compare Algorithm 1 with the nominal model, that is, a classification model without taking fairness into account, FAAP Wang et al. (2022), a recently proposed adversarial training method for achieving fairness in image classification, that has showed promising results and the DRO-based method by Hashimoto et al. (2018). In all cases we use ResNet-18 as the deployed model architecture. Further, we note that FAAP follows the training of a GAN, in which both the discriminator and generator losses are updated every time. The loss of the generator contains a weighted sum of a fairness loss, which includes an entropy regularization term, and a target prediction loss. When applying Algorithm 1, we try the values $\alpha, \beta \in \{1\text{e-}3, 1\text{e-}2, 1\text{e-}1\}$ and $T \in \{20, 50\}$. Further, for FAAP we try the values $\{1\text{e-}3, 1\text{e-}2, 1\text{e-}1\}$ for the learning rate, the values $\{20, 50\}$ for the number of epochs, as well as the values $\{0.1, 0.5, 1\}$ for the weight of the target prediction loss in the objective. For the DRO-based method we try the same values with our approach for both the learning rate and the number of epochs. Finally, we remark that FAAP optimizes for demographic parity, which is also the main goal of our approach.

CelebA

The first dataset that we consider is the CelebA dataset Liu et al. (2015), consisting of $202,599$ images with 40 attributes per image. We utilize 150k images for training, 10k images for validation and 20k images for testing. The sensitive attribute that we consider is *gender* and the targets that we consider are *Brown Hair*, *Blond Hair*, *Wavy Hair*, and *Smiling*. The ratios $p_1/n_1, p_2/n_2$ for each subpopulation of the sensitive attribute, for each target, are illustrated in Table 10 and their difference for the training and test sets is illustrated in Table 11. The results are illustrated in Table 12. We refer to Appendix A for the standard deviations.

Table 10: Ratios $p_1/n_1, p_2/n_2$ on each target, for each subpopulation of *gender* on the CelebA dataset.

Target	Gender Class	
	Male	Female
Brown Hair	0.15	0.24
Blond Hair	0.02	0.24
Wavy Hair	0.14	0.45
Smiling	0.40	0.54

From Table 12, we observe that our approach outperforms the state-of-the-art methods in terms of SPD and EOD. More precisely, it is the best performing method in terms of both SPD and EOD on *Brown Hair*, *Wavy Hair*, and *Smiling*. Further, in terms of DEO it outperforms the other methods on *Brown Hair* and *Wavy Hair*. Moreover, we notice that our method improves accuracy over the nominal model on *Blond Hair*, however it worsens it on the remaining targets. In total the average reduction in accuracy from employing our method over the nominal model is 0.71%.

Table 11: Ratios difference $\left|\frac{p_1}{n_1} - \frac{p_2}{n_2}\right|$ for train data (Train) and test data (Test), for *gender* and *race* on the CelebA dataset.

Target	Gender	
	Train	Test
Brown Hair	0.088	0.103
Blond Hair	0.222	0.209
Wavy Hair	0.304	0.312
Smiling	0.139	0.142

Table 12: Numerical results on the CelebA dataset with *gender* as the sensitive attribute.

(a) Brown Hair

	Acc ↑	SPD ↓	EOD ↓	DEO ↓
Nominal	**0.855**	0.035	0.032	0.028
FAAP	0.853	0.028	0.009	0.024
DRO	0.850	0.022	**0.006**	0.009
Ours	0.851	**0.011**	**0.006**	**0.005**

(b) Blond Hair

	Acc ↑	SPD ↓	EOD ↓	DEO ↓
Nominal	0.917	0.059	0.055	0.033
FAAP	0.908	0.035	0.059	**0.025**
DRO	0.895	**0.032**	**0.045**	0.029
Ours	**0.919**	0.041	0.051	0.029

(c) Wavy Hair

	Acc ↑	SPD ↓	EOD ↓	DEO ↓
Nominal	**0.775**	0.081	0.065	0.045
FAAP	0.771	0.101	0.091	0.055
DRO	0.761	0.076	0.059	0.048
Ours	0.772	**0.055**	**0.041**	**0.025**

(d) Smiling

	Acc ↑	SPD ↓	EOD ↓	DEO ↓
Nominal	**0.825**	0.057	0.046	0.025
FAAP	0.794	0.039	0.021	**0.010**
DRO	0.798	0.045	0.024	0.012
Ours	0.806	**0.031**	**0.016**	0.015

LFW

The second dataset that we consider is the LFW dataset Huang et al. (2007), consisting of 13, 244 images with 73 attributes per image. We utilize 9, 5k images for training, 1k images for validation and 2, 5k images for testing. The sensitive attributes that we consider are *gender* and *race*. The targets that we consider are *Brown Hair*, *Attractive*, *Wavy Hair*, and *Smiling*. The ratios $p_1/n_1, p_2/n_2$ for each subpopulation of the sensitive attributes, for each target, are illustrated in Table 13 and their difference for the training and test sets is illustrated in Table 14. The results for *gender* and *race* are illustrated in Tables 15 and 16, respectively. We refer to Appendix A for the standard deviations.

From Table 15, we observe that our method outperforms the other approaches in terms of the out of sample fairness metrics across the board. More precisely, our method achieves the best performance in terms of SPD, on *Brown Hair*, *Wavy Hair* and *Smiling* and in

Table 13: Ratios $p_1/n_1, p_2/n_2$ on each target, for each subpopulation of *gender* and *race* on the LFW dataset.

Target	Gender Class		Race Class	
	Male	Female	White	Non-White
Brown Hair	0.317	0.489	0.364	0.330
Attractive	0.290	0.659	0.408	0.267
Wavy Hair	0.400	0.549	0.513	0.201
Smiling	0.339	0.683	0.393	0.479

Table 14: Ratios difference $\left|\frac{p_1}{n_1} - \frac{p_2}{n_2}\right|$ for train data (Train) and test data (Test), for *gender* and *race* on the LFW dataset.

Target	Gender		Race	
	Train	Test	Train	Test
Brown Hair	0.172	0.175	0.033	0.021
Attractive	0.369	0.371	0.141	0.117
Wavy Hair	0.149	0.203	0.312	0.277
Smiling	0.344	0.345	0.087	0.066

Table 15: Numerical results on the LFW Dataset with *gender* as the sensitive attribute.

(a) Brown Hair

	Acc ↑	SPD ↓	EOD ↓	DEO ↓
Nominal	**0.775**	0.151	0.113	0.069
FAAP	0.752	0.136	0.107	0.071
DRO	0.745	0.129	0.102	0.068
Ours	0.765	**0.112**	**0.074**	**0.041**

(b) Attractive

	Acc ↑	SPD ↓	EOD ↓	DEO ↓
Nominal	**0.747**	0.301	0.211	0.162
FAAP	0.715	**0.119**	0.133	0.082
DRO	0.669	0.126	0.145	0.088
Ours	0.728	0.164	**0.037**	**0.038**

(c) Wavy Hair

	Acc ↑	SPD ↓	EOD ↓	DEO ↓
Nominal	**0.779**	0.141	0.055	0.036
FAAP	0.767	0.162	0.121	0.067
DRO	0.712	0.122	0.101	0.057
Ours	**0.779**	**0.087**	**0.010**	**0.033**

(d) Smiling

	Acc ↑	SPD ↓	EOD ↓	DEO ↓
Nominal	**0.898**	0.236	0.052	0.042
FAAP	0.878	0.231	**0.051**	**0.033**
DRO	0.835	0.134	0.092	0.049
Ours	0.852	**0.074**	0.086	0.089

Table 16: Numerical results on the LFW dataset with *race* as the sensitive attribute.

(a) Brown Hair

	Acc ↑	SPD ↓	EOD ↓	DEO ↓
Nominal	**0.791**	0.027	**0.035**	0.024
FAAP	0.766	**0.022**	0.042	0.027
DRO	0.711	**0.022**	0.039	**0.021**
Ours	**0.791**	0.025	0.055	0.032

(b) Attractive

	Acc ↑	SPD ↓	EOD ↓	DEO ↓
Nominal	0.748	0.096	0.067	0.051
FAAP	**0.773**	0.090	0.096	0.057
DRO	0.676	**0.033**	0.029	0.026
Ours	0.762	0.064	**0.027**	**0.024**

(c) Wavy Hair

	Acc ↑	SPD ↓	EOD ↓	DEO ↓
Nominal	0.767	0.207	0.128	0.089
FAAP	0.759	0.198	0.181	0.107
DRO	0.645	0.093	0.096	0.063
Ours	**0.772**	**0.192**	**0.079**	**0.060**

(d) Smiling

	Acc ↑	SPD ↓	EOD ↓	DEO ↓
Nominal	0.877	0.032	**0.064**	**0.042**
FAAP	**0.899**	0.018	0.081	0.045
DRO	0.868	0.011	0.086	0.045
Ours	0.893	**0.008**	0.109	0.058

terms of EOD as well as DEO, on *Brown Hair*, *Attractive* and *Wavy Hair*. Further, we notice that our method achieves same accuracy as the nominal model on *Wavy Hair*, while it is less accurate on *Brown Hair*, *Attractive* and *Smiling*. In total the average reduction in accuracy from employing our method over the nominal model is 2.34%.

From Table 16, we observe that our method achieves the best performance in terms of SPD on *Attractive*, *Wavy Hair* and *Smiling* and in terms of EOD as well as DEO, on *Attractive* and *Wavy Hair*. Moreover, our method improves accuracy over the nominal model on *Attractive*, *Wavy Hair* and *Smiling*. In total the average increase in accuracy from employing our method over the nominal model is 1.10%.

4.3 Multiple sensitive attributes

In this section, we apply Algorithm 1 to simultaneously alleviate bias in the case of multiple sensitive attributes. We consider the LSAC dataset for structured data and the LFW dataset with the target *Brown Hair* for unstructured data. In both cases we consider *race* and *gender* as the sensitive attributes. We compare the results of Algorithm 1 when applied to simultaneously alleviate bias for both sensitive attributes, as outlined in Section 2.4, with what we obtain when we apply it separately for each sensitive attribute. We evaluate in terms of SPD for each possible combination of *race* and *gender*. We note that *race* takes the values *black* (B), *white* (W) and *gender* takes the values *male* (M), *female* (F). From Table 17, we observe that SPD with respect to the different subgroups is better, when Algorithm 1 is applied to alleviate bias for the two sensitive attributes jointly. This happens because in this case we only flip labels in the intersection of the two privileged/non-privileged groups, whereas when we apply Algorithm 1 to alleviate bias separately it is possible that a label is flipped negatively for an observation which is privileged for one sensitive attribute but not for the other and as a result make less fair predictions for this subpopulation. When we apply our method to alleviate bias jointly, we only flip labels in the intersection of the privileged/non-privileged groups, which makes the classifier fairer with respect to the remaining groups whose ratio of positive observations is in between.

Table 17: Out of sample accuracy and SPD comparison for our approach on the LSAC (without meritocracy constraints) and LFW datasets, with *race* and *gender* as the sensitive attributes, imposed either jointly or separately. SPD(a,b) denotes SPD with respect to subgroups a,b.

Dataset		Acc ↑	SPD(WM, WF) ↓	SPD(WM, BM) ↓	SPD(WM, BF) ↓	SPD(WF, BM) ↓	SPD(WF, BF) ↓	SPD(BM, BF) ↓
LSAC	Joint	0.884	0.001	0.007	0.003	0.008	0.003	0.005
	Separate	0.890	0.007	0.198	0.281	0.205	0.289	0.083
LFW	Joint	0.779	0.016	0.127	0.145	0.185	0.161	0.033
	Separate	0.790	0.024	0.174	0.152	0.199	0.175	0.058

4.4 Summary of findings

From the numerical experiments on structured data we have the following key findings: First, we observe that our method outperforms the state-of-the-art approaches in terms of out of sample SPD, in 3 out of 4 datasets. Moreover, we observe that when applied with the meritocracy constraints with meritocracy tolerance $\delta = 0.1$, it achieves the best performance in terms of meritocracy, while for $\delta = 0.5$ it achieves a good trade-off between fairness and meritocracy. We also observe that in our method the resulting average values of the merit covariates among positive outcomes are close to those from the nominal model, as can be seen in Table 9, indicating that the price of fairness is low, that is, we do not need to sacrifice much in meritocracy in order to improve fairness. Further, we observe that our method can also improve accuracy if the test data are less biased than the training data and worsen it otherwise. In total the average reduction in accuracy from employing our method over the nominal model, across all structured datasets, is 3.31%. Finally, we note that there does not exist a method that performs uniformly the best in terms of accuracy, fairness, and meritocracy.

From the numerical experiments on unstructured data we have the following key findings: First, we observe that our method outperforms the state-of-the-art approaches in terms of out of sample fairness. Moreover, it improves accuracy over the nominal model if the test data are less biased than the training data and worsens it otherwise. In total the average reduction in accuracy from employing our method over the nominal model, across all different targets and unstructured datasets, is 0.65%. We observe that there does not exist a method that performs uniformly the best in terms of accuracy and fairness, while our method achieves good performance in both objectives across the board.

Finally, we note that, in both structured and unstructured data, the benefit of our method over state-of-the-art approaches is mostly evident in terms of the SPD metric. This observation aligns with the main concept of our method, which is equalizing the rate of positive observations among the classes of the sensitive attribute.

5. Interpreting the flipping decisions

5.1 OCT

In this section, we use OCTs developed by Bertsimas and Dunn (2017), which are highly interpretable and achieve state-of-the-art performance on classification problems to identify and differentiate individuals for whom the outcome label is changed to either a *positive* or *negative* label from individuals with *no change* to the outcome labels. Further, we provide a comparison among the flipping decision from our method and those from the the method by Calders et al. (2009).

In order to train an OCT for this purpose, we first construct a dataset based on the individuals for which we do/do not flip outcome labels using Algorithm 1. We apply Algorithm 1 for the same values of the hyper-parameters as in Section 4.1, with moderate meritocracy tolerance $\delta = 0.5$ in case of structured data, and obtain the best performing ones on a validation set, for which we then obtain the final flipping decisions. Similarly, we obtain the flipping decisions from the method by Calders et al. (2009). Then, each observation in the training dataset is labeled as one of the following: (a) *flip positive* (outcome label changed to a positive label), (b) *flip negative* (outcome label changed to a negative label), or (c) *no flip* (outcome label unchanged) based on Algorithm 1 or the method by Calders et al. (2009). Using this dataset, we train a three-class OCT model with tree-depth chosen using cross-validation with depth one through five for the ease of interpretability and select the model with the highest cross-validation accuracy among them.

In the case of structured data, we present an OCT for interpreting the flipping decisions of the LSAC dataset for both our method as well as the method by Calders et al. (2009) in Figures 3 and 4, respectively. The features that we consider in the OCT are those from the LSAC dataset, see Section 4.1.1. The OCT approximates characteristics of law students for whom the outcome label was changed to either a positive (passing the bar exam) or a negative (failing the bar exam) label. Further, it identifies and differentiates characteristics of non-white students that should be positively labeled, and characteristics of white students whose outcomes should be flipped.

From Figure 3, we observe that our method identifies individuals in the non-white subpopulation with the highest UGPA and LSAT scores and flips their labels positively. Examples of such individuals include the following characteristics: $\{\text{lsat} \geq 46.5, \text{ugpa} \geq 3.25\}$. On the other hand, it identifies the least meritorious students in the white subpopulation, for example individuals with low UGPA and LSAT score, and flips their labels negatively. Examples of such individuals include the following characteristics: $\{\text{ugpa} < 3.25,\ \text{lsat} < 38.25,\ \text{zgpa} < -0.24\}$.

From Figure 4, we observe that the method by Calders et al. (2009) follows in general a less meritocratic procedure for label flipping than our approach. More precisely, we observe that individuals in the non-white subpopulation with labels flipped positively have the following characteristics: $\{\text{lsat} < 33.25,\ \text{zgpa} \geq 0.095\}$. Further, individuals in the white subpopulation with labels flipped negatively have the following characteristics: $\{\text{lsat} \geq 33.25,\ \text{zgpa} < -0.615\}$. We observe that the LSAT score in this case can be higher for individuals with labels flipped negatively than those with labels flipped positively. On the other hand, we note that our approach identifies the strongest individuals in the non-white subpopulation, in terms of LSAT score, and flips their label. More precisely, we observe that

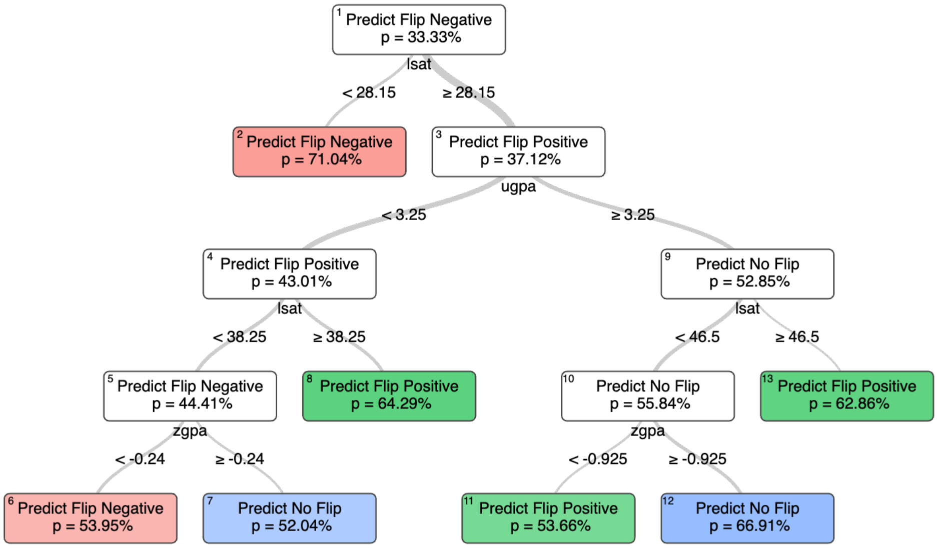

Figure 3: OCT for interpreting the flipping decisions of our method for the *acceptance* target on the LSAC dataset. We note that *Flip Positive* indicates individuals in the non-white subpopulation with label flipped positively, *Flip Negative* indicates individuals in the white subpopulation with label flipped negatively and *No Flip* indicates no label flipping. The value of p corresponds to the percentage of observations in the node corresponding to the predicted class, that is, the class with the highest percentage. The out of sample accuracy of the OCT is 0.937.

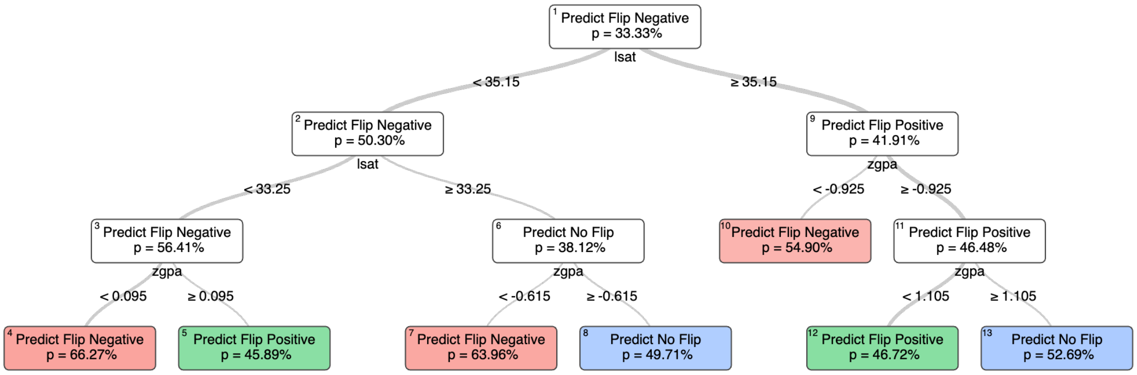

Figure 4: OCT for interpreting the flipping decisions of the method by Calders et al. (2009) for the *acceptance* target on the LSAC dataset. We note that *Flip Positive* indicates individuals in the non-white subpopulation with label flipped positively, *Flip Negative* indicates individuals in the white subpopulation with label flipped negatively and *No Flip* indicates no label flipping. The value of p corresponds to the percentage of observations in the node corresponding to the predicted class, that is, the class with the highest percentage. The out of sample accuracy of the OCT is 0.935.

the highest LSAT threshold for flipping a label positively is 35.15 for the method by Calders et al. (2009), while for our approach it is 46.50. We also observe that our method identifies individuals in the white subpopulation with very low LSAT scores, that is, lsat < 28.15, and flips their labels negatively. Finally, we observe that, unlike our approach, the method from Calders does not leverage UGPA for deciding which labels to flip.

In the case of unstructured data, we present an OCT for interpreting the flipping decisions on the LFW dataset, for the target *Smiling*, in Figure 5. The features that we consider in the OCT are auxiliary tabular data, that are available as other targets of the LFW dataset, including *Strong Nose-Mouth Lines*, *Teeth Not Visible* and *Mouth Closed*. The OCT approximates characteristics of individuals for whom the outcome label was changed to either a positive (smiling) or a negative (non-smiling) label using Algorithm 1. Further, it identifies and differentiates characteristics of males that should be positively labeled, and characteristics of females whose outcomes should be flipped.

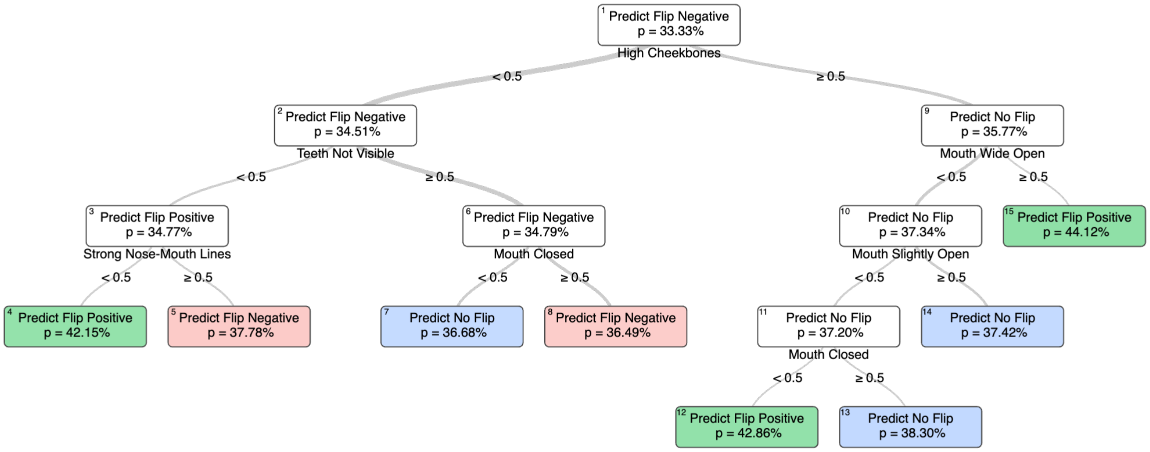

Figure 5: OCT for interpreting the flipping decisions of our method for the *Smiling* target on the LFW dataset. We note that *Flip Positive* indicates individuals in the male subpopulation with label flipped positively, *Flip Negative* indicates individuals in the female subpopulation with label flipped negatively and *No Flip* indicates no label flipping. The value of p corresponds to the percentage of observations in the node corresponding to the predicted class, that is, the class with the highest percentage. The out of sample accuracy of the OCT is 0.885.

As Figure 5 illustrates, our approach identifies males that have characteristics that are likely to correspond to a smiling person and flips their labels positively. More specifically, we observe that males with labels flipped positively have characteristics such as *High Cheekbones* and *Mouth Wide Open* or *Mouth Not Closed*. Further, we observe that our method identifies females with characteristics that are not very likely to correspond to a smiling person, such as *Teeth Not Visible* and *Mouth Closed*, and flips their labels negatively. In either case we notice that Algorithm 1 identifies those individuals that are least likely to be smiling/non-smiling and flips their labels.

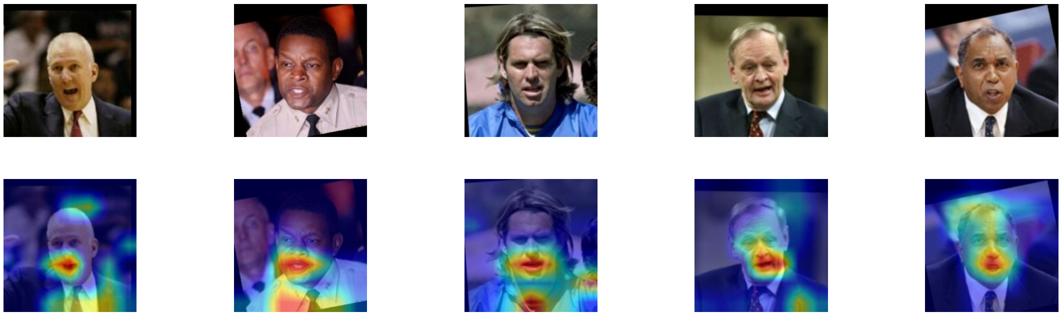

Figure 6: Images of males with labels flipped from non-smiling to smiling. The first row includes the original image and the second row includes the activations from Grad-CAM.

5.2 Grad-CAM

In this section, we utilize the visual explanation method Grad-CAM Selvaraju et al. (2017) in order to further understand the flipping decisions from our method on unstructured data. Grad-CAM is a model explanation method by visualizing the regions of input data that are important for predictions, see Selvaraju et al. (2017) for more details. We focus on the flipping decisions for the target *Smiling*, on the LFW dataset, where we recall that the privileged group is the female subpopulation. We run Algorithm 1 for the same values of the hyper-parameters as in Section 4.2 and obtain the best performing ones on a validation set, for which we then obtain the flipping decisions. Afterwards, we train a new image classification model, with the ResNet-18 architecture, for predicting whether we flip the label of each image or not. We then apply Grad-CAM to see which parts of the input image activate when flipping a label. From Figure 6, we observe that for males with labels flipped from no-smiling to smiling the region of the image that activates includes features such as *Mouth Open*, *Visible Teeth* and *Strong Noise Mouth Lines*. On the other hand, as Figure 7 illustrates, for females with labels flipped from smiling to no-smiling, the region of the image that activates includes features such as *Mouth Closed* or *Mouth Slightly Open* and *Teeth Not Visible*. We observe that the results are in agreement with the observations in Section 5.1.

6. Discussion

Our method applies to structured and unstructured data that are biased in terms of a sensitive attribute. In the case of structured data, both our approach as well as constraint-based approaches for fairness learning are applicable, in which case our approach often performs better in terms of out sample demographic parity, see for example LSAC and Crime datasets. Moreover, our approach can be incorporated in the training of deep neural networks and therefore it also applies to unstructured data, while other constraint-based approaches for fairness learning, such as Zafar et al. (2017), do not apply in this case. Further, note that methods that learn fair representations from training data may not generalize well to unseen data. The average improvement in out of sample accuracy of our method over FAAP is 0.66% in the CelebA dataset and 0.52% in the LFW dataset. In

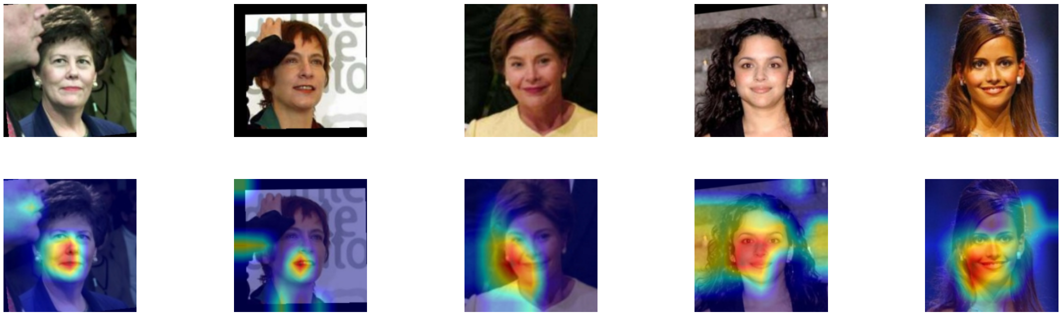

Figure 7: Images of females with labels flipped from smiling to non-smiling. The first row includes the original image and the second row includes the activations from Grad-CAM.

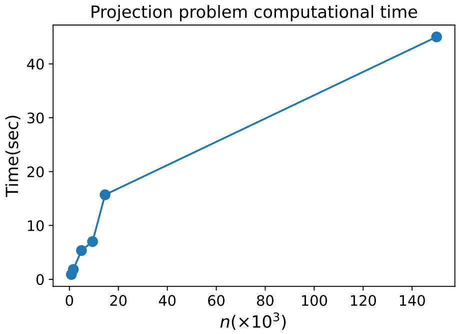

Figure 8: Computational time of Problem (5) with number of training data (n).

addition, with fair data representations there is a loss in the data interpretability, which can complicate the understanding of decisions made in the classification process. On the other hand, our approach, while working with the original features, achieves fairness in an interpretable way, as demonstrated in Section 5.

An important aspect of our approach is the computational time of the projection problem, that is, Problem (5). In Figure 8, we summarize the computational time of the projection problem across all datasets in the numerical experiments. We observe that in all cases the projection problem could be solved in less than a minute. For datasets of dimension up to 10^4 it was solved in less than 10 seconds, while for the largest dataset of dimension 1.5×10^5 it was solved in 45 seconds. We would not expect Problem (5) to scale to datasets of dimension 10^6 or higher, in which case an alternative approach could be used, such as solving the linear relaxation and rounding the optimal solution to the closest integer.

7. Conclusions

To summarize, in this paper we developed a unified method for improving fairness in both structured and unstructured data classification, utilizing a novel optimization approach to train classification models. In the case of structured data, we showed that we can modify the selection processes so as to enhance fairness, resulting in better performance than existing methods. Moreover, our method also improves accuracy over the nominal model in case the test data are less biased than the training data and worsens it otherwise. The average reduction in accuracy is 3.31% on structured data and 0.65% on unstructured data. Further, our approach, when applied to structured data, takes meritocracy into account by placing additional constraints on the moments of selected merit covariates. As a result, our approach outperforms state-of-the-art methods in terms of meritocracy, when applied with a small meritocracy tolerance, while achieving a good trade-off between fairness and meritocracy when applied with a moderate tolerance. We observe that after employing our method, the merit attributes among the positive outcomes do not change significantly from the nominal model, indicating a small price of fairness. In practice, one has the freedom of selecting how much emphasis should be placed on meritocracy, when applying our method to structured data, by adjusting the meritocracy tolerance δ. Finally, by utilizing OCTs and Grad-CAM, we were able to interpret the flipping decisions made by our method. In the case of structured data, we observed that our method makes intuitive changes to the current selection processes in a way that is understandable by human decision makers. Further, in the case of unstructured data, we noticed that the label flips from our method are also intuitive. We believe that the methodology proposed in this paper contributes to alleviating bias in an interpretable and equitable way.

Code availability

A code implementation of our method, including structured and unstructured data examples can be found in the following github repository: `https://github.com/ThKoukouv/Fair_Classification`.

Acknowledgements

We would like to thank the anonymous reviewers for their detailed and constructive comments, which have helped greatly to improve the quality and presentation of the manuscript.

Appendix A

In this section of the Appendix we include the standard deviations for the numerical experiments. Tables 18, 19, 20 and 21 include the standard deviations for the LSAC, Crime, COMPAS and German credit datasets, respectively. Further, Table 22 includes those for CelebA and Tables 23, 24 include those for LFW with *gender* and *race* as the sensitive attribute, respectively.

Appendix B

Mixed integer formulations for LR and SVM In this section we provide the exact mixed integer formulations of Problem (4), when the classification model is either LR or SVM. For LR, we linearize $z_i\boldsymbol{\beta} = \boldsymbol{\gamma}_i$, $z_i\beta_0 = \gamma_{0i}$. Leveraging the fact that $z_i \in \{0,1\}$, for a large M the constraints $z_i\boldsymbol{\beta} = \boldsymbol{\gamma}_i$ are equivalent to $-z_iM \leq \gamma_{ij} \leq z_iM,\ -(1-z_i)M \leq \gamma_{ij} - \beta_j \leq (1-z_i)M$. In a similar way we linearize $z_i\beta_0 = \gamma_{0i}$ and obtain the following

Table 18: Standard deviations for the results in Table 2.

	Acc ↑	SPD ↓	EOD ↓	DEO ↓	Merit(*lsat*) ↓	Merit(*ugpa*) ↓
Nominal	0.886 ± 0.023	0.299 ± 0.059	0.207 ± 0.062	0.282 ± 0.045	0.239 ± 0.143	0.263 ± 0.152
Calders	0.884 ± 0.000	0.019 ± 0.000	0.019 ± 0.000	0.084 ± 0.000	0.229 ± 0.000	0.248 ± 0.000
Kamiran	0.899 ± 0.000	0.013 ± 0.000	$\mathbf{0.004 \pm 0.000}$	$\mathbf{0.003 \pm 0.000}$	0.239 ± 0.000	0.259 ± 0.000
Zafar	0.889 ± 0.000	0.015 ± 0.000	$\mathbf{0.004 \pm 0.000}$	0.022 ± 0.000	0.205 ± 0.000	0.223 ± 0.000
Agarwal	$\mathbf{0.903 \pm 0.002}$	0.109 ± 0.002	0.045 ± 0.002	0.137 ± 0.003	0.183 ± 0.002	0.199 ± 0.002
Hashimoto	$\mathbf{0.903 \pm 0.001}$	0.151 ± 0.009	0.072 ± 0.009	0.206 ± 0.011	0.181 ± 0.010	0.197 ± 0.011
Ours($\delta = 0.1$)	0.893 ± 0.006	0.072 ± 0.022	0.022 ± 0.011	0.037 ± 0.021	$\mathbf{0.089 \pm 0.016}$	$\mathbf{0.107 \pm 0.015}$
Ours($\delta = 0.5$)	0.892 ± 0.007	0.026 ± 0.006	0.017 ± 0.007	0.056 ± 0.021	0.156 ± 0.061	0.166 ± 0.072
Ours(No Merit)	0.890 ± 0.007	$\mathbf{0.011 \pm 0.002}$	0.018 ± 0.004	0.068 ± 0.022	0.181 ± 0.047	0.191 ± 0.033

Table 19: Standard deviations for the results in Table 4.

	Acc ↑	SPD ↓	EOD ↓	DEO ↓	Merit(*pct-under-pov*) ↓	Merit(*pct-unemp*) ↓
Nominal	$\mathbf{0.779 \pm 0.005}$	0.296 ± 0.015	0.249 ± 0.020	0.164 ± 0.014	0.026 ± 0.001	0.011 ± 0.001
Calders	0.732 ± 0.000	0.121 ± 0.000	0.269 ± 0.000	0.213 ± 0.000	0.039 ± 0.000	0.011 ± 0.001
Kamiran	0.775 ± 0.000	0.195 ± 0.000	0.027 ± 0.000	$\mathbf{0.009 \pm 0.000}$	0.037 ± 0.000	0.007 ± 0.000
Zafar	0.757 ± 0.000	0.232 ± 0.000	0.026 ± 0.000	0.043 ± 0.000	0.026 ± 0.000	0.012 ± 0.001
Agarwal	0.771 ± 0.003	0.221 ± 0.007	0.043 ± 0.009	0.039 ± 0.005	0.029 ± 0.001	0.006 ± 0.001
Hashimoto	0.767 ± 0.003	0.398 ± 0.005	0.255 ± 0.009	0.219 ± 0.006	0.045 ± 0.011	0.006 ± 0.000
Ours($\delta = 0.1$)	0.728 ± 0.039	0.213 ± 0.078	0.191 ± 0.080	0.136 ± 0.053	$\mathbf{0.023 \pm 0.003}$	$\mathbf{0.003 \pm 0.000}$
Ours($\delta = 0.5$)	0.689 ± 0.017	0.107 ± 0.008	0.019 ± 0.005	0.041 ± 0.005	0.029 ± 0.001	0.005 ± 0.001
Ours(No Merit)	0.678 ± 0.004	$\mathbf{0.093 \pm 0.003}$	$\mathbf{0.017 \pm 0.003}$	0.038 ± 0.006	0.037 ± 0.001	0.007 ± 0.001

Table 20: Standard deviations for the results in Table 6.

	Acc ↑	SPD ↓	EOD ↓	DEO ↓	Merit(*priors*) ↓
Nominal	0.652 ± 0.007	0.141 ± 0.029	0.195 ± 0.032	0.129 ± 0.027	0.048 ± 0.038
Calders	0.544 ± 0.000	$\mathbf{0.074 \pm 0.000}$	0.128 ± 0.000	$\mathbf{0.069 \pm 0.000}$	0.112 ± 0.000
Kamiran	$\mathbf{0.679 \pm 0.000}$	0.118 ± 0.000	0.192 ± 0.000	0.106 ± 0.000	0.073 ± 0.000
Zafar	0.554 ± 0.000	0.097 ± 0.000	0.139 ± 0.000	0.095 ± 0.000	0.091 ± 0.000
Agarwal	0.669 ± 0.006	0.159 ± 0.009	0.256 ± 0.016	0.151 ± 0.009	0.075 ± 0.004
Hashimoto	0.658 ± 0.014	0.115 ± 0.018	0.213 ± 0.030	0.119 ± 0.008	0.086 ± 0.022
Ours($\delta = 0.1$)	0.643 ± 0.016	0.106 ± 0.034	0.158 ± 0.036	0.098 ± 0.031	$\mathbf{0.034 \pm 0.022}$
Ours($\delta = 0.5$)	0.653 ± 0.044	0.101 ± 0.042	0.138 ± 0.065	0.090 ± 0.033	0.059 ± 0.031
Ours(No Merit)	0.659 ± 0.011	0.098 ± 0.013	$\mathbf{0.122 \pm 0.007}$	0.088 ± 0.011	0.076 ± 0.035

formulation:

$$
\begin{aligned}
\min_{\boldsymbol{\beta},\beta_0,\boldsymbol{z},\boldsymbol{\gamma},\boldsymbol{\gamma}_0} \quad & \sum_{i=1}^{n} \log\left(1+\exp\left(-y_i(\boldsymbol{\beta}^T\boldsymbol{x}_i+\beta_0)+2y_i(\boldsymbol{\gamma}_i^T\boldsymbol{x}_i+\gamma_{0i})\right)\right)\\
\text{s.t} \quad & \sum_{i\in\mathcal{S}_1} z_i = \lceil \tau_1 \cdot n_1 \rceil,\\
& \sum_{i\in\mathcal{S}_2} z_i = \lceil \tau_2 \cdot n_2 \rceil,\\
& -z_iM \le \gamma_{ij} \le z_iM,\ i\in[n], j\in[p],\\
& -z_iM \le \gamma_{0i} \le z_iM,\ i\in[n], j\in[p],
\end{aligned}
$$

Table 21: Standard deviations for the results in Table 8.

	Acc ↑	SPD ↓	EOD ↓	DEO ↓	Merit(*credit*) ↓
Nominal	**0.732 ± 0.010**	0.035 ± 0.007	0.036 ± 0.011	**0.027 ± 0.009**	0.016 ± 0.005
Calders	0.716 ± 0.000	0.018 ± 0.000	0.051 ± 0.000	0.037 ± 0.000	0.145 ± 0.000
Kamiran	0.714 ± 0.000	0.012 ± 0.000	0.044 ± 0.000	**0.027 ± 0.000**	0.128 ± 0.000
Zafar	0.664 ± 0.000	0.041 ± 0.000	0.027 ± 0.000	0.030 ± 0.000	0.033 ± 0.000
Agarwal	0.728 ± 0.002	0.028 ± 0.007	0.076 ± 0.006	0.055 ± 0.005	0.019 ± 0.007
Hashimoto	0.671 ± 0.015	0.067 ± 0.033	0.087 ± 0.069	0.055 ± 0.047	0.038 ± 0.012
Ours($\delta = 0.1$)	0.729 ± 0.010	0.018 ± 0.009	0.049 ± 0.025	0.037 ± 0.012	**0.015 ± 0.007**
Ours($\delta = 0.5$)	0.720 ± 0.008	0.007 ± 0.004	**0.026 ± 0.009**	0.036 ± 0.004	0.017 ± 0.005
Ours(No Merit)	0.721 ± 0.008	**0.006 ± 0.003**	**0.026 ± 0.007**	0.033 ± 0.003	0.019 ± 0.005

Table 22: Standard deviations for the results in Table 12.

(a) Brown Hair

	Acc ↑	SPD ↓	EOD ↓	DEO ↓
Nominal	0.855 ± 0.021	0.035 ± 0.007	0.032 ± 0.006	0.028 ± 0.007
FAAP	0.853 ± 0.024	0.028 ± 0.004	0.009 ± 0.002	0.024 ± 0.008
DRO	0.850 ± 0.026	0.022 ± 0.005	0.006 ± 0.001	0.009 ± 0.001
Ours	0.851 ± 0.024	0.012 ± 0.004	0.006 ± 0.001	0.005 ± 0.001

(b) Blond Hair

	Acc ↑	SPD ↓	EOD ↓	DEO ↓
Nominal	0.917 ± 0.015	0.059	0.055	0.033
FAAP	0.908 ± 0.012	0.035 ± 0.003	0.059 ± 0.006	0.025 ± 0.004
DRO	0.895 ± 0.013	0.032 ± 0.002	0.062 ± 0.006	0.029 ± 0.004
Ours	0.919 ± 0.012	0.041 ± 0.009	0.051 ± 0.011	0.029 ± 0.007

(c) Wavy Hair

	Acc ↑	SPD ↓	EOD ↓	DEO ↓
Nominal	0.775 ± 0.022	0.081 ± 0.012	0.065 ± 0.006	0.045 ± 0.009
FAAP	0.771 ± 0.028	0.101 ± 0.015	0.091 ± 0.012	0.055 ± 0.007
DRO	0.761 ± 0.028	0.076 ± 0.009	0.059 ± 0.008	0.048 ± 0.003
Ours	0.772 ± 0.024	0.055 ± 0.005	0.041 ± 0.003	0.025 ± 0.003

(d) Smiling

	Acc ↑	SPD ↓	EOD ↓	DEO ↓
Nominal	0.825 ± 0.029	0.057 ± 0.009	0.046 ± 0.007	0.025 ± 0.007
FAAP	0.794 ± 0.028	0.039 ± 0.008	0.021 ± 0.003	0.010 ± 0.003
DRO	0.798 ± 0.029	0.045 ± 0.008	0.024 ± 0.005	0.012 ± 0.004
Ours	0.806 ± 0.026	0.031 ± 0.008	0.016 ± 0.003	0.015 ± 0.003

where p denotes the number of features. For SVM, we linearize $z_i\boldsymbol{w} = \boldsymbol{v}_i$, $z_ib = d_i$. Leveraging the fact that $z_i \in \{0, 1\}$, for a large M the constraints $z_i\boldsymbol{w} = \boldsymbol{v}_i$ are equivalent to $-z_iM \leq v_{ij} \leq z_iM$, $-z_iM \leq v_{ij} \leq z_iM$. In a similar way we linearize $z_ib = d_i$ and

Table 23: Standard deviations for the results in Table 15.

(a) Brown Hair

	Acc ↑	SPD ↓	EOD ↓	DEO ↓
Nominal	0.775 ± 0.011	0.151 ± 0.010	0.113 ± 0.005	0.069 ± 0.006
FAAP	0.752 ± 0.021	0.136 ± 0.004	0.107 ± 0.015	0.071 ± 0.012
DRO	0.745 ± 0.026	0.129 ± 0.008	0.102 ± 0.009	0.068 ± 0.003
Ours	0.765 ± 0.023	0.112 ± 0.005	0.074 ± 0.015	0.041 ± 0.008

(b) Attractive

	Acc ↑	SPD ↓	EOD ↓	DEO ↓
Nominal	0.747 ± 0.031	0.301 ± 0.011	0.211 ± 0.041	0.162 ± 0.014
FAAP	0.715 ± 0.009	0.119 ± 0.008	0.133 ± 0.013	0.082 ± 0.006
DRO	0.669 ± 0.030	0.126 ± 0.032	0.145 ± 0.055	0.088 ± 0.038
Ours	0.728 ± 0.031	0.164 ± 0.023	0.037 ± 0.007	0.038 ± 0.012

(c) Wavy Hair

	Acc ↑	SPD ↓	EOD ↓	DEO ↓
Nominal	0.779 ± 0.012	0.141 ± 0.019	0.055 ± 0.031	0.036 ± 0.017
FAAP	0.0767 ± 0.012	0.162 ± 0.020	0.121 ± 0.033	0.067 ± 0.017
DRO	0.712 ± 0.048	0.122 ± 0.039	0.101 ± 0.027	0.057 ± 0.010
Ours	0.779 ± 0.007	0.087 ± 0.012	0.010 ± 0.006	0.033 ± 0.013

(d) Smiling

	Acc ↑	SPD ↓	EOD ↓	DEO ↓
Nominal	0.898 ± 0.012	0.236 ± 0.009	0.052 ± 0.023	0.042 ± 0.008
FAAP	0.878 ± 0.006	0.231 ± 0.009	0.051 ± 0.015	0.033 ± 0.009
DRO	0.835 ± 0.059	0.134 ± 0.047	0.092 ± 0.029	0.049 ± 0.015
Ours	0.852 ± 0.009	0.074 ± 0.021	0.086 ± 0.023	0.089 ± 0.011

Table 24: Standard deviations for the results in Table 16.

(a) Brown Hair

	Acc ↑	SPD ↓	EOD ↓	DEO ↓
Nominal	0.791 ± 0.014	0.027 ± 0.015	0.035 ± 0.028	0.024 ± 0.014
FAAP	0.766 ± 0.022	0.022 ± 0.008	0.042 ± 0.012	0.027 ± 0.006
DRO	0.711 ± 0.044	0.022 ± 0.004	0.039 ± 0.012	0.021 ± 0.014
Ours	0.791 ± 0.010	0.030 ± 0.013	0.055 ± 0.024	0.032 ± 0.011

(b) Attractive

	Acc ↑	SPD ↓	EOD ↓	DEO ↓
Nominal	0.748 ± 0.033	0.096 ± 0.019	0.067 ± 0.037	0.051 ± 0.014
FAAP	0.773 ± 0.019	0.090 ± 0.014	0.096 ± 0.011	0.057 ± 0.009
DRO	0.676 ± 0.012	0.033 ± 0.008	0.029 ± 0.006	0.026 ± 0.006
Ours	0.762 ± 0.015	0.064 ± 0.015	0.027 ± 0.009	0.024 ± 0.006

(c) Wavy Hair

	Acc ↑	SPD ↓	EOD ↓	DEO ↓
Nominal	0.767 ± 0.021	0.207 ± 0.032	0.128 ± 0.051	0.089 ± 0.028
FAAP	0.759 ± 0.006	0.198 ± 0.009	0.181 ± 0.029	0.107 ± 0.013
DRO	0.645 ± 0.041	0.093 ± 0.022	0.096 ± 0.031	0.063 ± 0.021
Ours	0.772 ± 0.019	0.192 ± 0.013	0.079 ± 0.009	0.060 ± 0.011

(d) Smiling

	Acc ↑	SPD ↓	EOD ↓	DEO ↓
Nominal	0.877 ± 0.031	0.032 ± 0.014	0.064 ± 0.024	0.042 ± 0.012
FAAP	0.899 ± 0.004	0.018 ± 0.006	0.081 ± 0.026	0.045 ± 0.011
DRO	0.868 ± 0.047	0.011 ± 0.005	0.086 ± 0.012	0.045 ± 0.008
Ours	0.893 ± 0.005	0.008 ± 0.002	0.109 ± 0.015	0.058 ± 0.007

obtain the following formulation:

$$\begin{aligned}
\min_{\boldsymbol{w},b,\boldsymbol{z},\boldsymbol{v},\boldsymbol{d}} \quad & \frac{1}{2}\|\boldsymbol{w}\|_2^2 + C\sum_{i=1}^{n}\xi_i \\
\text{s.t} \quad & \sum_{i\in\mathcal{S}_1} z_i = \lceil \tau_1 \cdot n_1 \rceil, \\
& \sum_{i\in\mathcal{S}_2} z_i = \lceil \tau_2 \cdot n_2 \rceil, \\
& y_i(\boldsymbol{w}^T\boldsymbol{x}_i - b) - 2y_i(\boldsymbol{v}_i^T\boldsymbol{x}_i - d_i) \geq 1 - \xi_i,\ i \in [n], \\
& -z_iM \leq v_{ij} \leq z_iM,\ i \in [n],\ j \in [p], \\
& -z_iM \leq d_i \leq z_iM,\ i \in [n], \\
& -(1-z_i)M \leq v_{ij} - w_j \leq (1-z_i)M,\ i \in [n],\ j \in [p], \\
& -(1-z_i)M \leq d_i - b \leq (1-z_i)M,\ i \in [n], \\
& (1-\delta)\mu_j \leq \frac{\sum_{i=1}^{n} x_{ij}\left(y_i(1-2z_i)+1\right)}{\sum_{i=1}^{n}\left(y_i(1-2z_i)+1\right)} \leq (1+\delta)\mu_j,\ j \in \mathcal{M}, \\
& (1-\delta)\mu_j^2 \leq \frac{\sum_{i=1}^{n} x_{ij}^2\left(y_i(1-2z_i)+1\right)}{\sum_{i=1}^{n}\left(y_i(1-2z_i)+1\right)} \leq (1+\delta)\mu_j^2,\ j \in \mathcal{M}, \\
& z_i \in \{0,1\},\ i \in [n].
\end{aligned}$$

Appendix C

Wasserstrein distance approach for meritocracy constraints The merit covariate distribution among positive labels in the original and modified datasets is the following:

$$((x_{1j}(y_1+1)/2, \dots, (x_{nj}(y_n+1)/2), \; ((x_{1j}(y_1(1-2z_1)+1)/2, \dots, (x_{nj}(y_n(1-2z_n)+1)/2), \; j \in \mathcal{M}.$$

Let P, Q denote the covariate distribution on positive labels in the original and modified datasets, respectively. We then have the constraint $\mathcal{W}_1(P, Q) \leq \delta$, which can be formulated as follows:

$$\min_{\boldsymbol{\gamma} \in \Gamma} \left\{ \sum_{k,l} \gamma_{kl} \left| x_{kj}(y_i+1)/2 - x_{lj}(y_l(1-2z_l)+1)/2 \right| \right\} \leq \delta,$$

where

$$\Gamma = \left\{ \boldsymbol{\gamma} \in \mathbb{R}_+^{n \times n} : \sum_j \gamma_{ij} = \frac{1}{p_1+p_2} \frac{y_i+1}{2}, \sum_i \gamma_{ij} = \frac{1}{p_1+p_2} \frac{y_j(1-2z_j)+1}{2} \right\}.$$

Linearizing the products $\gamma_{kl} z_l$ with new variables u_{kl} and introducing additional variables θ_{kl} to model absolute values, the Wasserstrein distance requirement can be formulated with the following linear constraints:

$$\begin{aligned}
&\sum_{k,l} \theta_{kl} \leq \delta, \\
&\frac{x_{kj}}{2}(\gamma_{kl} y_k + \gamma_{kl}) - \frac{x_{lj}}{2}(y_l \gamma_{kl} - 2y_l u_{kl} + \gamma_{kl}) \leq \theta_{kl}, \; \forall k, l, \\
&\frac{x_{kj}}{2}(\gamma_{kl} y_k + \gamma_{kl}) - \frac{x_{lj}}{2}(y_l \gamma_{kl} - 2y_l u_{kl} + \gamma_{kl}) \geq -\theta_{kl}, \; \forall k, l, \\
&\sum_j \gamma_{ij} = \frac{1}{p_1+p_2} \frac{y_i+1}{2}, \\
&\sum_i \gamma_{ij} = \frac{1}{p_1+p_2} \frac{y_j(1-2z_j)+1}{2} \\
&-Mz_l \leq u_{kl} \leq Mz_l, \; \forall l, \\
&u_{kl} \leq \gamma_{kl} + M(1-z_l), \; \forall k, l, \\
&u_{kl} \geq \gamma_{kl} - M(1-z_l), \; \forall k, l,
\end{aligned}$$

where M is a large number.

References

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

Julia Angwin, Jeff Larson, Surya Mattu, and Lauren Kirchner. Machine bias. *ProPublica, May*, 23(2016):139–159, 2016.

Solon Barocas and Andrew D Selbst. Big data's disparate impact. *California Law Review*, pages 671–732, 2016.

Dimitris Bertsimas and Jack Dunn. Optimal classification trees. *Machine Learning*, 106(7): 1039–1082, 2017.

Dimitris Bertsimas and Robert Weismantel. *Optimization over integers*, volume 13. Dynamic Ideas Belmont, 2005.

Alex Beutel, Jilin Chen, Zhe Zhao, and Ed H Chi. Data decisions and theoretical implications when adversarially learning fair representations. *arXiv preprint arXiv:1707.00075*, 2017.

Tolga Bolukbasi, Kai-Wei Chang, James Y Zou, Venkatesh Saligrama, and Adam T Kalai. Man is to computer programmer as woman is to homemaker? debiasing word embeddings. *Advances in Neural Information Processing Systems*, 29, 2016.

Silvia Bonettini, Marco Prato, and Simone Rebegoldi. A cyclic block coordinate descent method with generalized gradient projections. *Applied Mathematics and Computation*, 286:288–300, 2016.

Toon Calders, Faisal Kamiran, and Mykola Pechenizkiy. Building classifiers with independency constraints. In *2009 IEEE International Conference on Data Mining Workshops*, pages 13–18. IEEE, 2009.

Toon Calders, Asim Karim, Faisal Kamiran, Wasif Ali, and Xiangliang Zhang. Controlling attribute effect in linear regression. In *2013 IEEE 13th International Conference on Data Mining*, pages 71–80. IEEE, 2013.

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*, pages 797–806, 2017.

Andrew Cotter, Maya Gupta, Heinrich Jiang, Nathan Srebro, Karthik Sridharan, Serena Wang, Blake Woodworth, and Seungil You. Training well-generalizing classifiers for fairness metrics and other data-dependent constraints. In *International Conference on Machine Learning*, pages 1397–1405. PMLR, 2019.

Jiankang Deng, Jia Guo, Niannan Xue, and Stefanos Zafeiriou. Arcface: Additive angular margin loss for deep face recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4690–4699, 2019.

Michele Donini, Luca Oneto, Shai Ben-David, John S Shawe-Taylor, and Massimiliano Pontil. Empirical risk minimization under fairness constraints. *Advances in Neural Information Processing Systems*, 31, 2018.

Dheeru Dua and Casey Graff. UCI Machine Learning Repository, 2017. URL http://archive.ics.uci.edu/ml.

Harrison Edwards and Amos Storkey. Censoring representations with an adversary. 2016.

ethicML. Ethicml documentation, 2022. URL https://wearepal.ai/EthicML/.

Michael Feldman. *Computational fairness: Preventing machine-learned discrimination.* PhD thesis, 2015.

Anthony Gebran, Sumiran S Thakur, Lydia R Maurer, Hari Bandi, Robert Sinyard, Ander Dorken-Gallastegi, Mary Bokenkamp, Mohamad El Moheb, Leon Naar, Annita Vapsi, et al. Development of a machine learning–based prescriptive tool to address racial disparities in access to care after penetrating trauma. *JAMA surgery*, 2023.

Gabriel Goh, Andrew Cotter, Maya Gupta, and Michael P Friedlander. Satisfying real-world goals with dataset constraints. *Advances in Neural Information Processing Systems*, 29, 2016.

Gurobi Optimization, Inc. Gurobi optimizer reference manual, 2017. URL http://www.gurobi.com.

Moritz Hardt, Eric Price, and Nati Srebro. Equality of opportunity in supervised learning. *Advances in Neural Information Processing Systems*, 29, 2016.

Tatsunori Hashimoto, Megha Srivastava, Hongseok Namkoong, and Percy Liang. Fairness without demographics in repeated loss minimization. In *International Conference on Machine Learning*, pages 1929–1938. PMLR, 2018.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016.

Calypso Herrera, Florian Krach, and Josef Teichmann. Estimating full lipschitz constants of deep neural networks. *arXiv preprint arXiv:2004.13135*, 2020.

Gary B. Huang, Manu Ramesh, Tamara Berg, and Erik Learned-Miller. Labeled faces in the wild: A database for studying face recognition in unconstrained environments. Technical Report 07-49, University of Massachusetts, Amherst, October 2007.

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

Michael Kearns, Seth Neel, Aaron Roth, and Zhiwei Steven Wu. Preventing fairness gerrymandering: Auditing and learning for subgroup fairness. In *International Conference on Machine Learning*, pages 2564–2572. PMLR, 2018.

Michael P Kim, Amirata Ghorbani, and James Zou. Multiaccuracy: Black-box post-processing for fairness in classification. In *Proceedings of the 2019 AAAI/ACM Conference on AI, Ethics, and Society*, pages 247–254, 2019.

Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In *3rd International Conference on Learning Representations, ICLR*, 2015.

Jon Kleinberg, Sendhil Mullainathan, and Manish Raghavan. Inherent trade-offs in the fair determination of risk scores. *Proceedings of Innovations in Theoretical Computer Science*, 2016.

Junpei Komiyama, Akiko Takeda, Junya Honda, and Hajime Shimao. Nonconvex optimization for regression with fairness constraints. In *International Conference on Machine Learning*, pages 2737–2746. PMLR, 2018.

Emmanouil Krasanakis, Eleftherios Spyromitros-Xioufis, Symeon Papadopoulos, and Yiannis Kompatsiaris. Adaptive sensitive reweighting to mitigate bias in fairness-aware classification. In *Proceedings of the 2018 World Wide Web Conference*, pages 853–862, 2018.

Marisol Lila, Manuel Martín Fernández, Enrique Gracia Fuster, Juan J López Ossorio, and José L González. Identifying key predictors of recidivism among offenders attending a batterer intervention program: A survival analysis. *Psychosocial Intervention, 2019, vol. 28, num. 3, p. 157-167*, 2019.

Ziwei Liu, Ping Luo, Xiaogang Wang, and Xiaoou Tang. Deep learning face attributes in the wild. In *Proceedings of International Conference on Computer Vision (ICCV)*, December 2015.

Pranay K Lohia, Karthikeyan Natesan Ramamurthy, Manish Bhide, Diptikalyan Saha, Kush R Varshney, and Ruchir Puri. Bias mitigation post-processing for individual and group fairness. In *ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 2847–2851. IEEE, 2019.

Christos Louizos, Kevin Swersky, Yujia Li, Max Welling, and Richard Zemel. The variational fair autoencoder. In *4th International Conference on Learning Representations*, 2016.

Harikrishna Narasimhan. Learning with complex loss functions and constraints. In *International Conference on Artificial Intelligence and Statistics*, pages 1646–1654. PMLR, 2018.

Adam Paszke, Sam Gross, Soumith Chintala, Gregory Chanan, Edward Yang, Zachary DeVito, Zeming Lin, Alban Desmaison, Luca Antiga, and Adam Lerer. Automatic differentiation in pytorch. 2017.

Dino Pedreshi, Salvatore Ruggieri, and Franco Turini. Discrimination-aware data mining. In *Proceedings of the 14th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 560–568, 2008.

Shelly L Peffer. Title vii and disparate-treatment discrimination versus disparate-impact discrimination: The supreme court's decision in ricci v. destefano. *Review of Public Personnel Administration*, 29(4):402–410, 2009.

Geoff Pleiss, Manish Raghavan, Felix Wu, Jon Kleinberg, and Kilian Q Weinberger. On fairness and calibration. *Advances in Neural Information Processing Systems*, 30, 2017.

Chuan Qin, Hengshu Zhu, Tong Xu, Chen Zhu, Liang Jiang, Enhong Chen, and Hui Xiong. Enhancing person-job fit for talent recruitment: An ability-aware neural network approach. In *The 41st International ACM SIGIR Conference on Research & Development in Information Retrieval*, pages 25–34, 2018.

Novi Quadrianto, Viktoriia Sharmanska, and Oliver Thomas. Discovering fair representations in the data domain. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8227–8236, 2019.

Vikram V Ramaswamy, Sunnie SY Kim, and Olga Russakovsky. Fair attribute classification through latent space de-biasing. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9301–9310, 2021.

Ramprasaath R Selvaraju, Michael Cogswell, Abhishek Das, Ramakrishna Vedantam, Devi Parikh, and Dhruv Batra. Grad-cam: Visual explanations from deep networks via gradient-based localization. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 618–626, 2017.

Zhibo Wang, Xiaowei Dong, Henry Xue, Zhifei Zhang, Weifeng Chiu, Tao Wei, and Kui Ren. Fairness-aware adversarial perturbation towards bias mitigation for deployed deep models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10379–10388, 2022.

Michael Wick, Jean-Baptiste Tristan, et al. Unlocking fairness: a trade-off revisited. *Advances in Neural Information Processing Systems*, 32, 2019.

Linda F Wightman. Lsac national longitudinal bar passage study. lsac research report series. *LSAC Research Report Series.*, 1998.

Blake Woodworth, Suriya Gunasekar, Mesrob I Ohannessian, and Nathan Srebro. Learning non-discriminatory predictors. In *Conference on Learning Theory*, pages 1920–1953. PMLR, 2017.

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

Rich Zemel, Yu Wu, Kevin Swersky, Toni Pitassi, and Cynthia Dwork. Learning fair representations. In *International Conference on Machine Learning*, pages 325–333. PMLR, 2013.

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

Indre Zliobaite, Faisal Kamiran, and Toon Calders. Handling conditional discrimination. In *2011 IEEE 11th International Conference on Data Mining*, pages 992–1001. IEEE, 2011.