Debiasing Deep Chest X-Ray Classifiers using Intra- and Post-processing Methods

Ričards Marcinkevičs ricards.marcinkevics@inf.ethz.ch
Ece Ozkan ece.oezkanelsen@inf.ethz.ch
Julia E. Vogt julia.vogt@inf.ethz.ch
Department of Computer Science
ETH Zurich
Zurich, Switzerland

Abstract

Deep neural networks for image-based screening and computer-aided diagnosis have achieved expert-level performance on various medical imaging modalities, including chest radiographs. Recently, several works have indicated that these state-of-the-art classifiers can be biased with respect to sensitive patient attributes, such as race or gender, leading to growing concerns about demographic disparities and discrimination resulting from algorithmic and model-based decision-making in healthcare. Fair machine learning has focused on mitigating such biases against disadvantaged or marginalised groups, mainly concentrating on tabular data or natural images. This work presents two novel intra-processing techniques based on fine-tuning and pruning an already-trained neural network. These methods are simple yet effective and can be readily applied *post hoc* in a setting where the protected attribute is unknown during the model development and test time. In addition, we compare several intra- and post-processing approaches applied to debiasing deep chest X-ray classifiers. To the best of our knowledge, this is one of the first efforts studying debiasing methods on chest radiographs. Our results suggest that the considered approaches successfully mitigate biases in fully connected and convolutional neural networks offering stable performance under various settings. The discussed methods can help achieve group fairness of deep medical image classifiers when deploying them in domains with different fairness considerations and constraints.

1. Introduction

Chest X-ray imaging is an essential tool for screening and diagnosing conditions affecting the chest and its surrounding, requiring special training for an appropriate interpretation. There has been an increasing effort in deploying deep neural networks for image-based screening and computer-aided diagnosis on chest radiographs (Allaouzi and Ahmed, 2019; Cohen et al., 2020; Bressem et al., 2020) from various datasets (Rajpurkar et al., 2017; Wang et al., 2019; Johnson et al., 2019), with some models achieving an expert-level performance (Irvin et al., 2019). However, several works (Larrazabal et al., 2020; Seyyed-Kalantari et al., 2020, 2021) have shown that these classifiers may be biased, raising ethical concerns regarding ML systems involved in high-stakes decisions (Char et al., 2018; Wiens et al., 2019; Obermeyer et al., 2019). For instance, an ICU patient monitoring and management model trained on a dataset containing few patients from minority groups might suffer from under- or over-detection of events in these groups, leading to alarm fatigue among medical staff and disparate patient outcomes (Rajkomar et al., 2018).

Motivated by similar concerns, researchers have provided many solutions for adjusting models' outputs and directly incorporating fairness into the learning process (Kearns, 2017). In this paper, we will assess the fairness of neural networks from the perspective of classification parity (Corbett-Davies and Goel, 2018): a classifier is said to be fair if some derivative of its confusion matrix, for instance, the true positive rate (TPR), is even across the categories of the protected attribute, such as race or gender. A practical scenario of mitigating bias w.r.t. protected attributes could be as follows. Consider deploying a predictive neural-network-based model in several clinical centres with different demographics, e.g. as explored by Zech et al. (2018) for chest X-ray classification. The constraints on the bias and protected attribute of interest might vary across clinical centres due to different population demographics (see Figure 1). Therefore, it might be more practical to debias the original model based on the local data, following an intra- or post-processing approach (Bellamy et al., 2018; Savani et al., 2020). The setting above is even more relevant with the widespread availability and use of pre-trained models (Gupta et al., 2018; Raghu et al., 2019; Rasmy et al., 2021).

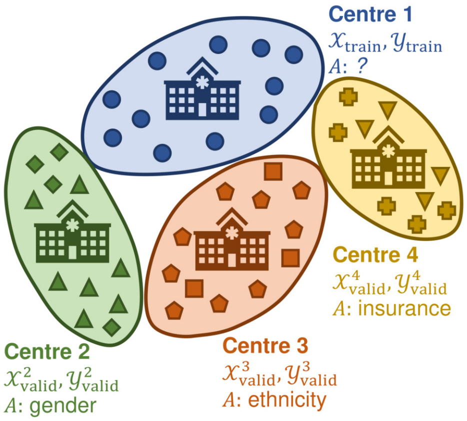

Centre 1
$\mathcal{X}_{\text{train}}, \mathcal{Y}_{\text{train}}$
A: ?
Centre 2
$\mathcal{X}^2_{\text{valid}}, \mathcal{Y}^2_{\text{valid}}$
A: gender
Centre 3
$\mathcal{X}^3_{\text{valid}}, \mathcal{Y}^3_{\text{valid}}$
A: ethnicity
Centre 4
$\mathcal{X}^4_{\text{valid}}, \mathcal{Y}^4_{\text{valid}}$
A: insurance

Figure 1: The intra-processing setting. A model is trained on centre **1**, and debiased on centres **2**, **3**, and **4** that have different fairness constraints (A).

Several prior works on debiasing classifiers, i.e. minimising bias, have mainly concentrated on tabular data or natural images, e.g. by Zafar et al. (2017); Zhang et al. (2018); Kim et al. (2019a); Reimers et al. (2021), and assumed that bias constraints are known before or during training. In that case, the model's inputs could be transformed or reweighed, or bias constraints could be incorporated directly into the loss function. Another line of work has focused on a completely model-agnostic approach (Kamiran et al., 2012; Hardt et al., 2016), adjusting the model's predictions *post hoc* and requiring the knowledge of the protected attribute at test time. On the other hand, the intra-processing scenario emerges naturally when potential biases are unknown or unexplored at the model development time, in other words, when it is impractical or impossible to train a debiased classifier from scratch, as described above, and when the protected attribute is unavailable at test time. Nevertheless, similar to most methods for enforcing classification parity, intra-processing does require the protected attribute *during debiasing*. Although some techniques can be applied when the sensitive attribute is *entirely* unknown, the current work focuses on a different setting.

We propose two novel intra-processing debiasing techniques based on fine-tuning and pruning. Furthermore, we compare several previously proposed debiasing approaches applied to deep chest X-ray classifiers in terms of statistical parity (Besse et al., 2021) and equality of opportunity (Hardt et al., 2016). We believe that this is one of the first works comparing debiasing methods on chest X-ray images. We exploit the publicly available and widely used MIMIC-CXR (Medical Information Mart for Intensive Care – Chest X-Ray) dataset (Johnson et al., 2019). Classifiers trained on this dataset have been shown to be biased w.r.t. various sensitive attributes, such as gender, race, or insurance type (Seyyed-Kalantari et al., 2020, 2021).

Generalizable Insights about Machine Learning in the Context of Healthcare

The main contributions of this work are as follows. (*i*) We consider differentiable proxy functions for statistical parity and equality of opportunity and establish their correspondence to the covariance between the decision boundary of a neural network and the protected attribute. (*ii*) We introduce simple yet effective intra-processing debiasing procedures based on minimising the proxy functions via fine-tuning and pruning an already-trained neural network, which can be effective in a setting where the protected attribute is unknown during the model development and test time. (*iii*) We conduct a comprehensive comparison among the proposed and well-established debiasing approaches on fully connected and convolutional neural networks on several datasets, including chest X-rays, showing that the compared methods can help achieve group fairness when deploying them in domains with different fairness considerations and constraints.

2. Preliminaries

Below, we outline the setting considered throughout the paper. We assume that disjoint training, validation, and test datasets $\mathcal{D} = \left\{(\boldsymbol{x}_i, y_i, a_i)\right\}_i = \mathcal{D}_{\text{train}} \cup \mathcal{D}_{\text{valid}} \cup \mathcal{D}_{\text{test}}$ are given, where $\boldsymbol{x}_i$ are features, e.g. a p-dimensional vector or an image, $y_i \in \{0, 1\}$ is the label, and $a_i \in \{0, 1\}$ is the protected attribute. Attribute a_i may be present among the features in $\boldsymbol{x}_i$ or may be completely exogenous. We will use the capital letters $\boldsymbol{X}$, Y, and A to refer to the corresponding random variables. Furthermore, let $\mathcal{X} = \left\{\boldsymbol{x}_i\right\}_i$, $\mathcal{Y} = \left\{y_i\right\}_i$, and $\mathcal{A} = \left\{a_i\right\}_i$. Let $f_{\boldsymbol{\theta}}(\cdot)$ denote a neural network parameterised by $\boldsymbol{\theta}$ and trained on data points $\left\{(\boldsymbol{x}_i, y_i)\right\}_i$ from $\mathcal{D}_{\text{train}}$. In our experiments (see Section 5), we consider fully connected and convolutional architectures for $f_{\boldsymbol{\theta}}(\cdot)$. If $f_{\boldsymbol{\theta}}(\cdot)$ is a multilayer perceptron, $\boldsymbol{\theta}$ is given by weight matrices $\left\{\boldsymbol{W}^{\text{in}}, \boldsymbol{W}^1, ..., \boldsymbol{W}^L, \boldsymbol{W}^{\text{out}}\right\}$. We will use $\boldsymbol{z}^l\left(\boldsymbol{x}\right)$ for the pre-activations and $\boldsymbol{h}^l(\boldsymbol{x}) = \sigma\left(\boldsymbol{z}^l(\boldsymbol{x})\right)$ for activations in layer $1 \leq l \leq L$, at the input $\boldsymbol{x}$, where $\sigma(\cdot)$ is an activation function. The output of $f_{\boldsymbol{\theta}}(\cdot)$ is given by sigmoid $\left(\boldsymbol{W}^{\text{out}}\boldsymbol{h}^L(\boldsymbol{x})\right)$. For final classification, a threshold $t \in [0,\, 1]$ on the output is chosen by maximising some performance measure, e.g. accuracy, on held-out data $\mathcal{D}_{\text{valid}}$. Thus, for input $\boldsymbol{x}$, the prediction is $\hat{y} = \mathbf{1}_{\left\{f_{\boldsymbol{\theta}(\boldsymbol{x}) \geq t}\right\}}$, where $\mathbf{1}_{\{\cdot\}}$ is the indicator function.

3. Background and Related Work

Classification Parity Many criteria for the fairness of machine learning models have been considered so far (Corbett-Davies and Goel, 2018). The two most common and practical classification parity metrics are statistical parity and equality of opportunity. Statistical parity difference (SPD) (Savani et al., 2020; Besse et al., 2021) is defined as the difference between the probabilities of positive outcomes, i.e. predictions made by the model $f_{\boldsymbol{\theta}}(\cdot)$, across the groups of the protected attribute A:

$$\mathrm{SPD} = \mathbb{P}_{\boldsymbol{X},A}\left(\widehat{Y}=1\middle|A=0\right) - \mathbb{P}_{\boldsymbol{X},A}\left(\widehat{Y}=1\middle|A=1\right). \tag{1}$$

On the other hand, the equal opportunity difference (EOD) (Hardt et al., 2016; Savani et al., 2020) quantifies the discrepancy between the TPRs of the classifier $f_{\boldsymbol{\theta}}(\cdot)$:

$$\mathrm{EOD} = \mathbb{P}_{\boldsymbol{X},Y,A}\left(\widehat{Y}=1\middle|Y=1,A=0\right) - \mathbb{P}_{\boldsymbol{X},Y,A}\left(\widehat{Y}=1\middle|Y=1,A=1\right). \tag{2}$$

In practice, quantities from Equations 1 and 2 can be evaluated using empirical estimators on held-out test data.

Debiasing Minimisation of the SPD or EOD is a solvable technical problem. Debiasing, i.e. the minimisation of bias, often leads to a decrease in the overall predictive performance of the classifier (Reimers et al., 2021). Therefore, ideally, a debiasing algorithm should reduce bias $\mu(\cdot)$, given by the SPD or EOD, without sacrificing performance $\rho(\cdot)$, e.g. balanced accuracy (BA) (Brodersen et al., 2010). One can view this problem as an instance of constrained optimisation (Zafar et al., 2017, 2019; Kim et al., 2019b; Savani et al., 2020), either minimising the bias subject to performance constraints or *vice versa*, maximising performance under bias constraints.

Many debiasing algorithms have been proposed for the setting outlined above. Bellamy et al. (2018) and Savani et al. (2020) provide a practical taxonomy: (*i*) pre-processing algorithms usually reweigh or transform original data, obfuscating protected variables or attenuating group disparities (Kamiran and Calders, 2011; Zemel et al., 2013; Calmon et al., 2017; Celis et al., 2020); (*ii*) in-processing methods incorporate debiasing explicitly into learning, e.g. using an adversarial loss or regularisation (Kamishima et al., 2012; Zafar et al., 2017; Zhang et al., 2018; Reimers et al., 2021); (*iii*) post-processing approaches treat the biased model as a black-box and merely edit its predictions (Kamiran et al., 2012; Hardt et al., 2016; Pleiss et al., 2017); last but not least, (*iv*) intra-processing techniques are inspired by fine-tuning and achieve parity by changing the model's parameters *post hoc* (Savani et al., 2020). An essential difference between post- and intra-processing are assumptions about the access to model parameters and the protected attribute at test time: post-processing adjusts *predictions* based on the given protected attribute value.

All of the methods mentioned above assume the knowledge of the protected attribute at some point in the model's life cycle. Another line of work (Nam et al., 2020a; Lee et al., 2021), beyond the scope of the current paper, focuses on the setting wherein the source of bias is *entirely* unknown, usually resorting to strong assumptions, such as that the bias is easier to learn than other relevant associations.

Pruning In neural networks, parameter pruning usually refers to removing irrelevant weights or entire structural elements (Cheng et al., 2017), e.g. filters in convolutional neural networks. Early works on pruning neural networks, such as optimal brain damage (LeCun et al., 1990) and optimal brain surgeon (Hassibi and Stork, 1993), leveraged criteria based on the second derivative of the error function to prune unimportant weights throughout the training process. Several modern techniques focus on pruning entire structures (Wen et al., 2016; Molchanov et al., 2017; He et al., 2017), e.g. convolutional filters or channels. However, the main principle remains the same: parameters are pruned based on some criterion, and the network is subsequently fine-tuned by backpropagation, if necessary.

Role of Individual Units in Neural Networks Several works have investigated the importance and interpretation of *individual* neurons within deep neural network models, in contrast to the previous research on attribution, which primarily examined input-output relationships (Ancona et al., 2019). For instance, Bau et al. (2020) observed the emergence of single-unit object detectors whose activations are correlated with high-level concepts in discriminative and generative convolutional neural networks (CNN). Leino et al. (2018); Dhamdhere et al. (2019); Srinivas and Fleuret (2019); Nam et al. (2020b) introduce new attribution measures that quantify the influence of individual neurons.

Fairness of Deep Chest X-ray Classifiers Recently, researchers have scrutinised the fairness of deep classifiers trained on well-known and publicly available chest X-ray datasets (Johnson et al., 2019; Wang et al., 2019; Irvin et al., 2019). Larrazabal et al. (2020) reported a consistently lower AUROC for underrepresented genders on imbalanced datasets. In a multi-centre setting, Zech et al. (2018) observed that the performance of chest X-ray classifiers was significantly lower on held-out external data, indicating possible bias due to confounding. Underdiagnosis and TPR disparity were evaluated by Seyyed-Kalantari et al. (2020, 2021) across three large chest X-ray datasets, showing higher underdiagnosis and lower TPRs in underserved patient populations.

4. Methods

We introduce novel intra-processing approaches to debiasing classifiers w.r.t. the SPD and EOD, which build on the work by Savani et al. (2020), who have proposed the intra-processing setting. For an extended comparison with the related works, see Section 7. Our techniques are tailored towards differentiable classifiers and neural networks in particular.

4.1. Classification Parity Proxies

The proposed methods focus on the minimisation of classification disparity. In particular, we minimise the SPD or EOD *directly* without a need for adversarial training using differentiable proxy functions. Given sets of N data points $\mathcal{X}=\{\boldsymbol{x}_i\}_{i=1}^N$, $\mathcal{Y}=\{y_i\}_{i=1}^N$, and $\mathcal{A}=\{a_i\}_{i=1}^N$, the proxy $\tilde{\mu}$ for the SPD is given by

$$\tilde{\mu}_{\mathrm{SPD}}\left(f_{\boldsymbol{\theta}}(\cdot),\,\mathcal{X},\,\mathcal{Y},\,\mathcal{A}\right)=\frac{\sum_{i=1}^N f_{\boldsymbol{\theta}}\left(\boldsymbol{x}_i\right)\left(1-a_i\right)}{\sum_{i=1}^N 1-a_i}-\frac{\sum_{i=1}^N f_{\boldsymbol{\theta}}\left(\boldsymbol{x}_i\right)a_i}{\sum_{i=1}^N a_i}, \tag{3}$$

and, for the EOD, we have

$$\tilde{\mu}_{\mathrm{EOD}}\left(f_{\boldsymbol{\theta}}(\cdot),\,\mathcal{X},\,\mathcal{Y},\,\mathcal{A}\right)=\frac{\sum_{i=1}^N f_{\boldsymbol{\theta}}\left(\boldsymbol{x}_i\right)\left(1-a_i\right)y_i}{\sum_{i=1}^N\left(1-a_i\right)y_i}-\frac{\sum_{i=1}^N f_{\boldsymbol{\theta}}\left(\boldsymbol{x}_i\right)a_iy_i}{\sum_{i=1}^N a_iy_i}. \tag{4}$$

Notably, Equations 3 and 4 are similar to the objective functions considered by Zafar et al. (2017, 2019) in the context of fair logistic regression and SVM models. In Appendix A, we show that Equation 3 corresponds to the empirical estimate $\widehat{\mathrm{Cov}}\left(A,\,f_{\boldsymbol{\theta}}\left(\boldsymbol{X}\right)\right)$. Similarly, Equation 4 corresponds to the empirical estimate of the conditional covariance $\widehat{\mathrm{Cov}}\left(A,\,f_{\boldsymbol{\theta}}\left(\boldsymbol{X}\right)\middle|\,Y=1\right)$. The intuition behind the algorithms described in Sections 4.2 and

4.3 is to fine-tune the given biased neural network and minimise these proxies, thus, reducing the covariance between the protected attribute and decision boundary. The general debiasing procedure is schematically summarised in Figure 2: an already-trained network is debiased on held-out validation data, using the classification parity proxies, and can produce unbiased predictions without the protected attribute at test time.

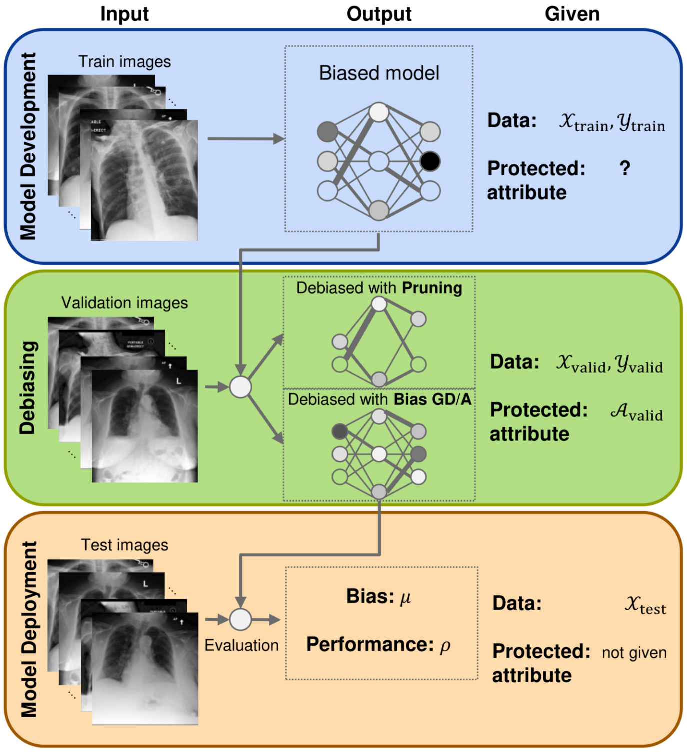


Figure 2: A summary of the debiasing procedure. *(i)* A biased model is trained without the knowledge of the protected attribute. *(ii)* Using the differentiable classification parity proxies, the model is debiased by performing pruning or bias gradient descent/ascent on validation data. *(iii)* The debiased model is evaluated on test data and can produce unbiased predictions *without* the protected attribute.

4.2. Neural Network Pruning for Debiasing

Pruning refers to the procedure of reducing the effective number of parameters in a model. There has been renewed interest in neural network pruning (Cheng et al., 2017; Blalock et al., 2020), mainly for compressing models and reducing computational complexity and energy consumption. Different from the existing work, we propose using pruning to mitigate bias in neural network classifiers. In particular, we introduce a procedure for pruning individual units, or neurons, based on their contributions to classification disparity. In fully connected (FC) layers, a unit is a single component of the (pre-)activation vector; in convolutional layers, it is a component of the three-dimensional tensor. Below we use a one-dimensional index j to enumerate units for both FC and convolutional layers.

Gradient-based Bias Influence Building on the influence-directed explanations proposed by Leino et al. (2018) for measuring the influence of individual neurons in CNNs, we propose a gradient-based statistic for quantifying the influence of units on the classification disparity. For a differentiable bias measure $\tilde{\mu}$, e.g. Equation 3 or 4, the influence of the j-th unit in the l-th layer is given by

$$S_{l,j}=\frac{1}{N}\sum_{i=1}^{N}\frac{\partial\tilde{\mu}\left(f_{\boldsymbol{\theta}}(\cdot),\mathcal{X},\mathcal{Y},\mathcal{A}\right)}{\partial z_j^l(\boldsymbol{x}_i)},\tag{5}$$

where $z_j^l(\boldsymbol{x}_i)$ denotes the unit's pre-activation at input $\boldsymbol{x}_i$. In practice, partial derivatives such as those above can be computed efficiently using automatic differentiation, e.g. PyTorch's Autograd module (Paszke et al., 2017). The measure in Equation 5 can identify the most influential units that need to be pruned. Empirical results in Section 6 suggest that removing influential units effectively reduces the bias.

Pruning Procedure Algorithm 1 outlines the proposed pruning procedure comprising a few simple steps. (*i*) For layer $1 \leq l \leq L$, the influence $S_{l,j}$ (see Equation 5) is evaluated for each unit j on the validation data $\mathcal{D}_{\text{valid}}$. The memory complexity can be reduced by evaluating and averaging the influence across mini-batches rather than the entire validation set. For similar reasons, one may choose to prune only specific layers selectively rather than all L intermediate layers. (*ii*) A specified number, determined by the number of steps $B \geq 1$, of the most influential units are pruned. For FC layers, a unit can be pruned by setting all outgoing weights to 0's, e.g. for the j-th unit in the l-th layer; this amounts to the assignment $\boldsymbol{W}_{j,\cdot}^{l} \leftarrow \mathbf{0}$. For convolutional layers, we implement a dropout-like binary mask applied to pre-activations (Srivastava et al., 2014). Furthermore, note that the order in which units are pruned depends on the sign of the initial model's bias, i.e. whether the bias needs to be driven down or up towards 0. (*iii*) The bias $\mu(\cdot)$ (see Equations 1 and 2) and performance $\rho(\cdot)$ of the pruned network are evaluated on the validation set. (*iv*) Influence $S_{l,j}$ is recomputed for the pruned network, and steps (*ii*)–(*iv*) are repeated. In the end, an optimal sparsity level is chosen by returning a pruned network with the lowest bias and the performance at least $\varrho > 0$, a hyperparameter determined by the user-specified constraint.

The procedure above greedily removes individual units in the intermediate layers of the neural network step-by-step based on the criterion given by Equation 5. It returns a pruned network with minimal bias subject to the performance constraint. We will see that, in practice, it often allows making few changes to the classifier without retraining from scratch and sacrificing the predictive performance while reducing the classification disparity.

4.3. Bias Gradient Descent/Ascent

Since the proxies given by Equations 3 and 4 are differentiable w.r.t. $\boldsymbol{\theta}$, one could reduce the bias directly using gradient descent or ascent, depending on the sign of the bias. Therefore, another approach we propose is fine-tuning the classifier $f_{\boldsymbol{\theta}}(\cdot)$ for a few epochs with a small learning rate, for instance, using mini-batch gradient descent and Equation 3 or 4 as a loss function. Algorithm 2 contains the pseudocode for the bias gradient descent/ascent (GD/A) procedure. This method is at first glance similar to the adversarial debiasing by Zhang et al. (2018), who apply a discriminator to the network's output. However, we perform gradient

```
Input: Held-out validation set $\mathcal{D}_{\text{valid}} = \{(\boldsymbol{x}_i, y_i, a_i)\}_{i=1}^{N_{\text{valid}}}$; neural network $f_{\boldsymbol{\theta}}(\cdot)$ with parameters $\boldsymbol{\theta}$ and $L$ intermediate layers; classification threshold $t \in [0,\,1]$; predictive performance measure $\rho(\cdot)$; bias measure $\mu(\cdot)$; differentiable bias proxy $\tilde{\mu}(\cdot)$; lower bound on performance $\varrho > 0$; number of steps $B \geq 1$
Output: Pruned and debiased network $f_{\tilde{\boldsymbol{\theta}}}(\cdot)$ with parameters $\tilde{\boldsymbol{\theta}}$
$\mu_0 \leftarrow \mu\left(\mathbf{1}_{\{f_{\boldsymbol{\theta}}(\cdot)\geq t\}}, \{\boldsymbol{x}_i\}_{i=1}^{N_{\text{valid}}}, \{y_i\}_{i=1}^{N_{\text{valid}}}, \{a_i\}_{i=1}^{N_{\text{valid}}}\right)$, where $(\boldsymbol{x}_i, y_i, a_i) \in \mathcal{D}_{\text{valid}}$
Initialise $\tilde{\boldsymbol{\theta}} \leftarrow \boldsymbol{\theta}$
Given $\mathcal{D}_{\text{valid}}$ and $\tilde{\mu}(\cdot)$, evaluate $S_{l,j}$ (see Equation 5) for every unit $j$ in layer $1 \leq l \leq L$
for $b = 0$ to $B - 1$ do
    Let $\tau_b \leftarrow q_{1-1/B}\left(\{\mathrm{sgn}\left(\mu_0\right) S_{l,j}\}\right)$, where $q_\alpha(\cdot)$ denotes the empirical $\alpha$-quantile
    Prune unit $j$ in layer $1 \leq l \leq L$ if $\mathrm{sgn}\left(\mu_0\right) S_{l,j} > \tau_b$ and adjust $\tilde{\boldsymbol{\theta}}$ accordingly
    $\tilde{t} \leftarrow \arg\max_{t' \in [0,\,1]} \rho\left(\mathbf{1}_{\{f_{\tilde{\boldsymbol{\theta}}}(\cdot),\geq t'\}}, \{\boldsymbol{x}_i\}_{i=1}^{N_{\text{valid}}}, \{y_i\}_{i=1}^{N_{\text{valid}}}\right)$
    $\mu_b \leftarrow \mu\left(\mathbf{1}_{\{f_{\tilde{\boldsymbol{\theta}}}(\cdot)\geq \tilde{t}\}}, \{\boldsymbol{x}_i\}_{i=1}^{N_{\text{valid}}}, \{y_i\}_{i=1}^{N_{\text{valid}}}, \{a_i\}_{i=1}^{N_{\text{valid}}}\right)$
    $\rho_b \leftarrow \rho\left(\mathbf{1}_{\{f_{\tilde{\boldsymbol{\theta}}}(\cdot)\geq \tilde{t}\}}, \{\boldsymbol{x}_i\}_{i=1}^{N_{\text{valid}}}, \{y_i\}_{i=1}^{N_{\text{valid}}}\right)$
    Reevaluate $S_{l,j}$ for the pruned network $f_{\tilde{\boldsymbol{\theta}}}(\cdot)$
    $\tilde{\boldsymbol{\theta}}_b \leftarrow \tilde{\boldsymbol{\theta}}$
end
$b^* \leftarrow \arg\min_{\substack{0 \leq b \leq B-1 \\ \rho_b \geq \varrho}} |\mu_b|$
return $f_{\tilde{\boldsymbol{\theta}}_{b^*}}(\cdot)$
```

Algorithm 1: Pruning procedure for debiasing neural networks. Individual units are removed greedily based on their influence, and a network with minimal bias subject to the specified performance constraint is returned.

descent/ascent on the differentiable bias proxies *after* the network has been trained and do not require knowledge of the protected attribute during training.

Similar to Algorithm 1, the weight update direction in bias GD/A depends on the sign of the initial bias. Likewise, at the end of the algorithm, a fine-tuned debiased network is returned, minimising the bias with the performance of at least ϱ. This procedure has several additional hyperparameters, namely, learning rate $\eta > 0$, which, in practice, should be chosen sufficiently small, mini-batch size $M \geq 1$, and a maximum number of fine-tuning epochs $E \geq 1$. In our experiments, we observed that, compared to the training of the original model, relatively few fine-tuning epochs suffice to reduce the bias. Although Algorithm 2 is based on the mini-batch gradient descent, other optimisation procedures can be adopted, e.g. batch gradient descent, as long as the procedure supports the evaluation of the differentiable proxy $\tilde{\mu}(\cdot)$ on several data points.

5. Experimental Setup

The purpose of our experiments was twofold: (*i*) test the proposed pruning and bias GD/A methods on tabular and image data for FC and CNN architectures and (*ii*) explore the use of intra- and post-processing to mitigate biases in deep chest X-ray classifiers. Below, we

```
Input: Held-out validation set $\mathcal{D}_{\text{valid}} = \{(\boldsymbol{x}_i, y_i, a_i)\}_{i=1}^{N_{\text{valid}}}$; neural network $f_{\boldsymbol{\theta}}(\cdot)$ with parameters $\boldsymbol{\theta}$; classification threshold $t \in [0, 1]$; predictive performance measure $\rho(\cdot)$; bias measure $\mu(\cdot)$; differentiable bias proxy $\tilde{\mu}(\cdot)$; lower bound on performance $\varrho > 0$; learning rate $\eta > 0$; number of epochs $E \geq 1$; mini-batch size $M \geq 1$
Output: Fine-tuned and debiased network $f_{\tilde{\boldsymbol{\theta}}}(\cdot)$ with parameters $\tilde{\boldsymbol{\theta}}$
$\mu_0 \leftarrow \mu\left(\mathbf{1}_{\{f_{\boldsymbol{\theta}}(\cdot) \geq t\}}, \{\boldsymbol{x}_i\}_{i=1}^{N_{\text{valid}}}, \{y_i\}_{i=1}^{N_{\text{valid}}}, \{a_i\}_{i=1}^{N_{\text{valid}}}\right)$, where $(\boldsymbol{x}_i, y_i, a_i) \in \mathcal{D}_{\text{valid}}$
Initialise $\tilde{\boldsymbol{\theta}} \leftarrow \boldsymbol{\theta}$
for $e = 0$ to $E - 1$ do
    Draw mini-batch $\mathcal{B} = \{(\boldsymbol{x}_i, y_i, a_i)\}_{i=1}^{M}$ without replacement, s.t. $\mathcal{B} \subseteq \mathcal{D}_{\text{valid}}$
    $\tilde{\mu}_e \leftarrow \tilde{\mu}\left(f_{\tilde{\boldsymbol{\theta}}}(\cdot), \{\boldsymbol{x}_i\}_{i=1}^{M}, \{y_i\}_{i=1}^{M}, \{a_i\}_{i=1}^{M}\right)$, where $(\boldsymbol{x}_i, y_i, a_i) \in \mathcal{B}$
    $\tilde{\boldsymbol{\theta}} \leftarrow \tilde{\boldsymbol{\theta}} - \operatorname{sgn}(\mu_0)\, \eta \nabla_{\tilde{\boldsymbol{\theta}}} \tilde{\mu}_e$
    $\tilde{t} \leftarrow \arg\max_{t' \in [0,\,1]} \rho\left(\mathbf{1}_{\{f_{\tilde{\boldsymbol{\theta}}}(\cdot), \geq t'\}}, \{\boldsymbol{x}_i\}_{i=1}^{N_{\text{valid}}}, \{y_i\}_{i=1}^{N_{\text{valid}}}\right)$
    $\mu_e \leftarrow \mu\left(\mathbf{1}_{\{f_{\tilde{\boldsymbol{\theta}}}(\cdot) \geq \tilde{t}\}}, \{\boldsymbol{x}_i\}_{i=1}^{N_{\text{valid}}}, \{y_i\}_{i=1}^{N_{\text{valid}}}, \{a_i\}_{i=1}^{N_{\text{valid}}}\right)$
    $\rho_e \leftarrow \rho\left(\mathbf{1}_{\{f_{\tilde{\boldsymbol{\theta}}}(\cdot) \geq \tilde{t}\}}, \{\boldsymbol{x}_i\}_{i=1}^{N_{\text{valid}}}, \{y_i\}_{i=1}^{N_{\text{valid}}}\right)$
    $\tilde{\boldsymbol{\theta}}_e \leftarrow \tilde{\boldsymbol{\theta}}$
end
$e^* \leftarrow \arg\min_{\substack{0 \leq e \leq E-1 \\ \rho_e \geq \varrho}} |\mu_e|$
return $f_{\tilde{\boldsymbol{\theta}}_{e^*}}(\cdot)$
```

Algorithm 2: Bias gradient descent/ascent procedure for debiasing neural networks. A biased classifier is fine-tuned by performing bias gradient descent or ascent on a differentiable bias proxy function. In the end, a network with minimal bias subject to the specified performance constraint is returned.

briefly summarise the datasets, pre-processing, compared techniques, and the evaluation procedure. Further implementation details can be found in Appendix D.

5.1. Datasets

We compared debiasing techniques on tabular and image data (see Table B.1 in Appendix B). Tabular datasets include several publicly available benchmarks, most of them part of IBM AIF 360 toolkit (Bellamy et al., 2018). We refer the reader to Quy et al. (2022) for a thorough exploratory analysis of these datatsets. Furthermore, we applied debiasing to CNNs trained on the large-scale chest X-ray dataset – MIMIC-CXR (Johnson et al., 2019). In addition, we performed experiments on synthetic data (see Appendix C).

Adult The Adult Census Income data contains 48,842 instances and includes seven categorical, two binary, and six numerical features. The task is to predict whether a person's annual income exceeds 50,000$ (Kohavi, 1996; Quy et al., 2022). In our experiments, we focused on the protected attribute "*sex*". Note that here and below, we use the term "*sex*" to match the reported terminology in the underlying data.

Bank The dataset was collected during phone call marketing campaigns (Moro et al., 2014; Quy et al., 2022) and comprises 45,211 samples with six categorical, four binary, and seven numerical features. The classification task is to predict a deposit subscription by a potential client. We used "*age*" as the protected variable.

COMPAS The Correctional Offender Management Profiling for Alternative Sanctions dataset (Larson et al., 2016; Quy et al., 2022) includes 7,214 samples with 31 categorical, 6 binary, and 14 numerical covariates. The underlying classification problem is predicting the risk of recidivism. The protected attribute is "*race*".

MIMIC-III Medical Information Mart for Intensive Care (MIMIC-III-v1.4) database consists of information on the admissions of patients who stayed in critical care units at a large tertiary care hospital (Johnson et al., 2016). It includes demographics, vital sign measurements, laboratory results, medications, notes, imaging reports, mortality rates, etc. We used pre-processing routine provided by Purushotham et al. (2018) that retains only the first admissions of adult patients ($>$ 15 years). Pre-processed data consist of 17 features from the SAPS-II score. We averaged time-series data for each feature/admission. Our experiments to predict in-hospital mortality focused on the "*age*", "*marital status*", and "*insurance type*" as protected attributes. For "*age*", we grouped subjects $\geq$ 78 years old into one category and the rest into another. For "*marital status*", the two groups comprised *single* and the rest. "*Insurance type*" was dichotomised by grouping *Medicare* and *Medicaid* into one category (*public* health insurance) and the rest into another (*private*), similarly to Meng et al. (2022). Not all protected attribute groupings are clinically meaningful, and debiasing might not be relevant in all cases. For instance, insurance type dichotomisation may be too simplistic since Medicare and Medicaid are distinctively different programs (Altman and Frist, 2015) and should be treated separately in practice. However, we focus on binary-valued protected attributes and contextualise our analysis in the previous work by Meng et al. (2022). In a similar vein, debiasing w.r.t. the SPD with "*age*" as the protected attribute may not be clinically relevant; however, we included these results for completeness.

MIMIC-CXR MIMIC-CXR is a large dataset of chest X-rays from 227,835 studies of 65,379 patients (Johnson et al., 2019). Each study contains one or more images, usually frontal and lateral views. We only used frontal view images, resizing them to 224×224 px. We focused on "*sex*" and "*ethnicity*" as protected variables since the groups of these attributes were previously shown to have disparate classification outcomes (Seyyed-Kalantari et al., 2020, 2021). For each image, one or more labels are reported, comprising 14 binary attributes. For the protected attribute "*sex*", *male* patients formed the privileged and *female* patients the unprivileged group. Since the classifiers trained using the following combinations of disease labels, protected attributes, and privileged/unprivileged groups were shown to have disparate TPRs (Seyyed-Kalantari et al., 2020), we took "*enlarged cardiomediastinum*" (enlarged CM) as the classification label for the attribute "*sex*". For "*ethnicity*", *white* patients were taken as the privileged, whereas patients with *Hispanic/Latino* ethnicity as unprivileged group. For this attribute, we chose "*pneumonia*" as the classification label. Studies with *no findings* were used as the negative class in both cases.

5.2. Debiasing Methods

In addition to the proposed pruning and bias GD/A procedures, we applied several other debiasing methods, focusing on intra- and post-processing approaches. Standard refers to the original, potentially biased classifier $f_{\boldsymbol{\theta}}(\cdot)$ with the classification threshold $t \in [0, 1]$ chosen to maximise the balanced accuracy on the held-out validation data. We used the random perturbation procedure (Random) described by Savani et al. (2020) as a baseline. This method perturbs the parameters of the original network $f_{\boldsymbol{\theta}}(\cdot)$ several times by multiplicative Gaussian noise, distributed as $\mathcal{N}(1, 0.01)$. The procedure returns a perturbed network maximising the bias-constrained objective proposed by Savani et al. (2020) on the validation set. ROC refers to the reject option classification post-processing algorithm (Kamiran et al., 2012) that swaps classification outcomes for the subjects from the underprivileged group who fall within the confidence band around the decision boundary. Eq. Odds is the equalised odds post-processing method (Hardt et al., 2016). This algorithm adjusts output labels probabilistically to balance the odds across the protected attribute categories. Lastly, we considered adversarial fine-tuning (Adv. Intra) (Savani et al., 2020), an intra-processing technique closely related to ours that fine-tunes the biased classifier via adversarial training. In Appendix E.2, we also compare with the adversarial in-processing algorithm by Zhang et al. (2018).

5.3. Classification Models and Debiasing Evaluation

For tabular datasets, we used the same FC architecture and training scheme for the classifier $f_{\boldsymbol{\theta}}(\cdot)$ (see Appendix D), following the experimental setup of Savani et al. (2020). For MIMIC-CXR, we used the VGG-16 (Simonyan and Zisserman, 2015) and ResNet-18 (He et al., 2016) CNN architectures, initialising them with pre-trained weights. All models were trained by minimising the binary cross-entropy loss using the Adam optimiser (Kingma and Ba, 2015). For chest X-ray classifiers, to avoid overfitting, we applied random augmentations during training, such as centre crop, horizontal flip, translation, and rotation.

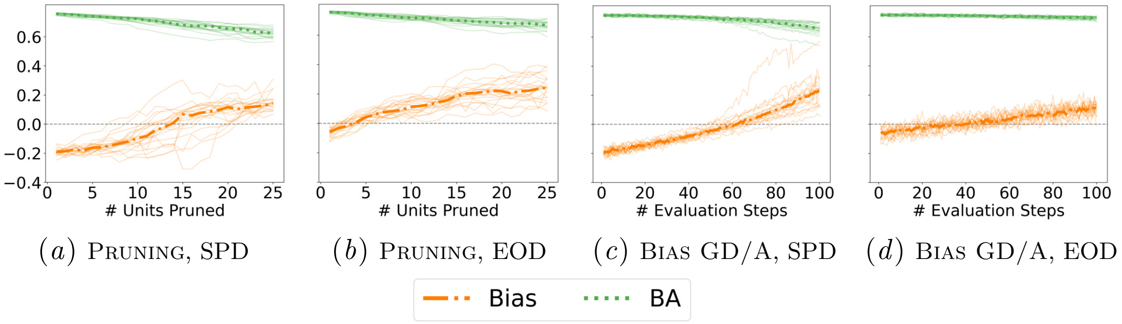
For all compared techniques, debiasing was performed only on the validation set (see Appendix D). The classifiers were trained and debiased repeatedly on the independent replicates of the train-validation-test split in the manner of Monte Carlo cross-validation. Classifiers were evaluated on the test data w.r.t. the bias and performance. We used balanced accuracy to reflect true positive and negative rates equally. For tabular data, the bias was evaluated in terms of the SPD and EOD. For MIMIC-CXR, we focused on the EOD rather than SPD since achieving even positive prediction outcomes across the groups of the protected attributes may not be clinically relevant.

6. Results

In this section, we provide the results of the empirical comparison among several debiasing techniques, including the proposed pruning and bias GD/A intra-processing algorithms. Further results and additional experiments on synthetic data, investigating the stability of pruning and bias GD/A, are discussed in Appendix E.

6.1. Results on Tabular Benchmarks

Tables 1 and 2 contain quantitative results obtained on tabular data: EOD, SPD, and BA before and after debiasing. Compared to other post- and intra-processing techniques, pruning and bias GD/A successfully mitigate biases and tend to sacrifice less accuracy on



(*a*) Pruning, SPD (*b*) Pruning, EOD (*c*) Bias GD/A, SPD (*d*) Bias GD/A, EOD

Figure 3: Changes in the **bias**, given by the SPD (*a, c*) and EOD (*b, d*), and **balanced accuracy** of the neural network during pruning (*a, b*) and bias gradient descent/ascent (*c, d*). The results were obtained on MIMIC-III for predicting in-hospital mortality with the "*insurance type*" as the protected attribute from 20 train-validation-test splits. **Bold** lines correspond to the median across 20 seeds. Note that during the bias GD/A, the model was evaluated three times an epoch.

most datasets. On average, pruning performs slightly worse than GD/A and has larger variability across seeds. The results for the adversarial fine-tuning are in line with those reported in the original paper by Savani et al. (2020). Generally, while this method visibly reduces the bias, it tends to sacrifice the BA more, likely due to minimising a loss function different from that of the bias GD/A. Interestingly, on Adult dataset, both of our procedures drastically reduce the BA of the classifier and perform worse than ROC: we attribute this to the general sensitivity of intra-processing methods (Savani et al., 2020) to initial conditions. In Appendix E, we explore this phenomenon further on synthetic data. In brief, we observed that when the bias of the original classifier is high, proposed techniques reduce the accuracy considerably or fail to reduce the bias, therefore, it may be prudent to retrain the model from scratch using an in-processing approach or resort to post-processing in such cases.

In addition, we examined changes in the bias and balanced accuracy of the neural network throughout the process of pruning and bias GD/A. Figure 3 shows the trajectories of the EOD, SPD, and BA obtained on MIMIC-III data for predicting in-hospital mortality with the "*insurance type*" as the protected attribute. Encouragingly, both methods drive the classification disparity towards zero while not affecting the balanced accuracy of the classifier significantly. We observed that few pruning steps or fine-tuning epochs, compared to the training time of the original model (a maximum of 1,000 epochs), were necessary to reduce the bias, suggesting the viability of fine-tuning an already-trained biased model on the validation set. Generally, the bias and BA trajectories for pruning featured slightly higher variance across seeds. An intuitive explanation could be that pruning, compared to GD/A, explores a relatively limited number of debiased network weight configurations, particularly for smaller architectures, such as the one in our tabular experiments (see Table D.1). We observed similar debiasing dynamics on other tabular benchmarks (see Figure E.1).

Table 1: Bias (*a*) and balanced accuracy (*b*) attained before and after debiasing neural networks trained on nonclinical tabular data. If necessary, debiasing was run twice for each dataset: for the SPD and EOD separately. The results are reported as averages followed by standard deviations across 20 train-validation-test splits. Best results are shown in **bold**, second-best – in *italic*, except for Standard.

(*a*) Bias

Bias Measure	**Method**	**Adult:** ***Sex***	**Bank:** ***Age***	**COMPAS:** ***Race***
SPD	Standard	-0.32±0.02	0.18±0.04	0.19±0.03
	Random	-0.04±0.01	*0.03±0.04*	0.09±0.04
	ROC	-0.04±0.02	0.08±0.04	**-0.01±0.01**
	Eq. Odds	-0.09±0.01	0.06±0.03	0.03±0.06
	Adv. Intra	*-0.03±0.00*	0.05±0.03	0.03±0.03
	Pruning	-0.04±0.05	**0.02±0.04**	*0.02±0.03*
	Bias GD/A	**-0.01±0.04**	0.04±0.05	0.04±0.04
EOD	Standard	-0.14±0.02	0.01±0.04	0.20±0.05
	Random	-0.07±0.03	*0.02±0.04*	0.09±0.04
	ROC	-0.05±0.03	0.04±0.04	**-0.01±0.01**
	Eq. Odds	**-0.01±0.04**	0.04±0.10	*0.03±0.06*
	Adv. Intra	-0.09±0.03	0.03±0.06	0.14±0.07
	Pruning	**-0.01±0.03**	**0.00±0.07**	0.04±0.06
	Bias GD/A	*-0.03±0.03*	*0.02±0.06*	0.06±0.06

(*b*) Balanced accuracy

Bias Measure	**Method**	**Adult:** ***Sex***	**Bank:** ***Age***	**COMPAS:** ***Race***
SPD	Standard	0.82±0.01	0.86±0.01	0.65±0.01
	Random	0.60±0.01	0.60±0.10	0.60±0.03
	ROC	**0.79±0.01**	0.66±0.10	0.50±0.00
	Eq. Odds	*0.73±0.02*	0.70±0.02	0.60±0.01
	Adv. Intra	0.56±0.01	0.61±0.09	0.56±0.04
	Pruning	0.56±0.04	*0.84±0.01*	*0.63±0.02*
	Bias GD/A	0.66±0.01	**0.86±0.01**	**0.64±0.01**
EOD	Standard	0.82±0.01	0.86±0.01	0.65±0.01
	Random	0.78±0.03	**0.86±0.01**	0.61±0.03
	ROC	**0.82±0.01**	**0.86±0.01**	0.50±0.00
	Eq. Odds	0.73±0.02	0.70±0.02	0.60±0.01
	Adv. Intra	*0.78±0.02*	*0.84±0.01*	0.61±0.02
	Pruning	*0.78±0.02*	**0.86±0.03**	*0.62±0.03*
	Bias GD/A	**0.82±0.01**	**0.86±0.01**	**0.64±0.01**

6.2. Chest X-ray Classification

Last but not least, we applied discussed intra- and post-processing techniques to CNNs trained on MIMIC-CXR. Table 3 reports the EOD and BA after debiasing across 20 independent splits for the two pairs of protected attributes and labels and the two architectures. For predicting enlarged CM under the protected attribute "*sex*", the EOD of the original classifier is mild for both VGG and ResNet, and most methods successfully reduce it without affecting the BA. Both pruning and bias GD/A achieve the best results on average alongside the equalised odds post-processing. Surprisingly, adversarial fine-tuning does not reduce the EOD equally well and leads to a lower BA. We attribute its poorer performance

Table 2: Bias and balanced accuracy before and after debiasing neural networks trained on MIMIC-III with *age*, *marital status*, and *insurance type* as protected attributes.

		Age		***Marital Status***		***Insurance Type***	
Bias Measure	**Method**	**Bias**	**BA**	**Bias**	**BA**	**Bias**	**BA**
SPD	Standard	-0.28±0.03	0.76±0.01	0.10±0.02	0.76±0.01	-0.19±0.03	0.75±0.01
	Random	-0.04±0.01	0.64±0.01	0.05±0.01	0.72±0.02	-0.04±0.01	0.67±0.01
	ROC	-0.05±0.01	0.63±0.01	0.03±0.03	**0.75±0.01**	-0.05±0.01	0.68±0.01
	Eq. Odds	*-0.01±0.01*	0.57±0.02	*0.01±0.00*	0.57±0.01	*-0.01±0.00*	0.57±0.01
	Adv. Intra	-0.04±0.01	0.60±0.02	0.04±0.02	0.67±0.04	-0.03±0.01	0.64±0.02
	Pruning	**0.00±0.02**	*0.69±0.02*	**0.00±0.02**	*0.73±0.01*	**0.00±0.02**	*0.69±0.03*
	Bias GD/A	*-0.01±0.02*	**0.72±0.01**	*0.01±0.02*	**0.75±0.01**	*-0.01±0.02*	**0.73±0.01**
EOD	Standard	-0.11±0.04	0.76±0.01	0.08±0.03	0.76±0.01	-0.05±0.04	0.75±0.01
	Random	*-0.05±0.05*	0.72±0.03	0.06±0.04	*0.74±0.03*	-0.04±0.04	**0.75±0.01**
	ROC	*-0.05±0.06*	0.69±0.04	0.03±0.05	**0.75±0.01**	-0.04±0.04	**0.75±0.02**
	Eq. Odds	**0.01±0.04**	0.57±0.02	**0.01±0.04**	0.57±0.01	*0.01±0.04*	0.57±0.01
	Adv. Intra	-0.08±0.03	0.71±0.02	0.06±0.04	0.73±0.01	-0.02±0.03	0.72±0.03
	Pruning	**0.01±0.06**	*0.73±0.01*	*-0.02±0.06*	0.73±0.02	**0.00±0.04**	*0.74±0.01*
	Bias GD/A	**-0.01±0.05**	**0.75±0.01**	*0.02±0.05*	**0.75±0.01**	**0.00±0.04**	**0.75±0.01**

to overfitting on the validation data (see Table E.1 in Appendix E), likely, due to having many learnable parameters in the discriminator network. Nevertheless, it is encouraging that the classifiers trained on the data imbalanced w.r.t. the protected attribute can be debiased *post hoc*, without retraining from scratch.

On the other hand, for predicting pneumonia under the protected attribute "*ethnicity*", the average EOD of the original model is significantly higher for both architectures. The only method which satisfactorily reduces the bias, in this case, is equalised odds. While pruning and bias GD/A do not hurt the performance, the average EOD after debiasing is far from zero. However, both techniques still perform better than the naïve random perturbation baseline. Similar to the results above, adversarial fine-tuning also fails to debias the network. We see several plausible explanations for the poorer performance of the proposed methods: (*i*) overfitting to the small validation set — for pneumonia and "*ethnicity*", it only contains about 1,000 images; (*ii*) the protected attribute "*ethnicity*" is harder to detect from X-rays, compared to "*sex*", and it may be prudent to use a post-processing technique which requires the attribute as input at *test time*; (*iii*) the general sensitivity of the intra-processing to the initial conditions (see Appendix E). Additionally, it must be noted that the attribute "*ethnicity*" is self-reported (Seyyed-Kalantari et al., 2020) and might be noisy and misaligned with the ground truth, introducing another source of variability into the results.

7. Discussion

The intra-processing methods proposed in this work (see Section 4) offer a simple yet effective way of fine-tuning neural networks to mitigate classification disparity. The proposed differentiable bias proxy functions are directly related to the empirical covariance between the decision boundary and protected attribute (see Appendix A), similar to the loss func-

Table 3: Equal opportunity difference and balanced accuracy attained before and after debiasing VGG-16 (*a*,*b*) and ResNet-18 (*c*,*d*) trained on MIMIC-CXR to predict (*a*,*c*) enlarged cardiomediastinum (with the protected attribute "*sex*") and (*b*,*d*) pneumonia (with the protected attribute "*ethnicity*").

(*a*) Enlarged CM, *Sex*; VGG-16

Method	EOD	BA
Standard	-0.05±0.02	0.77±0.01
Random	-0.03±0.03	*0.75±0.01*
ROC	-0.05±0.02	*0.75±0.03*
Eq. Odds	*0.01±0.03*	*0.75±0.01*
Adv. Intra	-0.04±0.03	0.73±0.01
Pruning	**0.00±0.02**	**0.76±0.02**
Bias GD/A	*-0.01±0.04*	**0.76±0.01**

(*b*) Pneumonia, *Ethnicity*; VGG-16

Method	EOD	BA
Standard	-0.14±0.04	0.73±0.02
Random	-0.11±0.06	**0.71±0.02**
ROC	*-0.07±0.06*	0.65±0.06
Eq. Odds	**0.00±0.06**	*0.70±0.01*
Adv. Intra	-0.13±0.05	*0.70±0.02*
Pruning	-0.09±0.05	**0.71±0.03**
Bias GD/A	-0.08±0.06	**0.71±0.02**

(*c*) Enlarged CM, *Sex*; ResNet-18

Method	EOD	BA
Standard	-0.05±0.04	0.76±0.01
Random	**0.00±0.03**	0.73±0.02
ROC	-0.05±0.03	*0.74±0.04*
Eq. Odds	*0.01±0.03*	*0.74±0.01*
Adv. Intra	-0.04±0.04	0.73±0.02
Pruning	*-0.01±0.03*	*0.74±0.02*
Bias GD/A	**0.00±0.03**	**0.76±0.01**

(*d*) Pneumonia, *Ethnicity*; ResNet-18

Method	EOD	BA
Standard	-0.14±0.05	0.73±0.02
Random	*-0.06±0.06*	0.65±0.04
ROC	-0.07±0.04	0.65±0.05
Eq. Odds	**-0.01±0.06**	0.70±0.01
Adv. Intra	-0.14±0.03	*0.71±0.02*
Pruning	-0.11±0.05	0.70±0.02
Bias GD/A	-0.11±0.05	**0.73±0.02**

tions considered by Zafar et al. (2017, 2019) in the context of linear classification. There exist criteria for fairness beyond the SPD and EOD (Corbett-Davies and Goel, 2018). Our approaches can be readily combined with other parity measures, e.g. the average odds difference, by deriving proxies similar to those described in Section 4.1.

The presented debiasing techniques differ in several aspects from the related work. While there have been many efforts to debias neural networks resorting to adversarial training (Zhang et al., 2018; Kim et al., 2019a; Reimers et al., 2021), these works have mainly focused on the in-processing setting, where the source of bias is known during training. Notably, the adversarial in-processing technique by Zhang et al. (2018) is similar to the proposed bias GD/A (see Section 4.3). Next to the main objective, it maximises the cross-entropy term for predicting the protected attribute using an adversary defined on the outputs of the base classifier. The current work offers a different perspective concentrating on the intra-processing scenario and chest X-ray classification. Moreover, to the best of our knowledge, neural network pruning or dropout have never been considered from the perspective of debiasing for group fairness. In particular, it would be interesting to investigate if the proposed pruning procedure may be augmented with other, non-gradient-based criteria for evaluating the influence of neurons (cf. Equation 5).

Although model-agnostic post-processing methods, such as ROC (Kamiran et al., 2012) and equalised odds (Hardt et al., 2016), are applied *post hoc*, they assume access to the protected attribute at test time. On the other hand, the methods introduced by Savani et al. (2020) are most closely related to ours. Random perturbation and layer-wise optimisation (Savani et al., 2020) are based on computationally expensive zeroth-order optimisation techniques. While adversarial fine-tuning, similarly to bias GD/A, minimises a differentiable bias proxy by mini-batch gradient descent, it resorts to adversarial training, has more hyper- and learnable parameters than our methods, and leverages a different loss function. Furthermore, Savani et al. (2020) focus only on the conventional tabular debiasing benchmarks and natural images and do not apply their methods to clinical data.

Empirically, we have comprehensively evaluated our and related intra- and post-processing methods on various tabular and medical image datasets for fully connected and convolutional neural network architectures (see Section 6). In brief, the experiments on tabular data (see Section 6.1) suggest that the proposed intra-processing approaches effectively reduce the bias when it is present and offer improved performance over model-agnostic techniques. We have also demonstrated that pruning and GD/A can reduce the classification disparity in deep chest X-ray classifiers (see Section 6.2) for VGG-16 and ResNet-18 networks. However, when the validation set is too small, and the bias of the initial model is too high, it might be more prudent to retrain the model from scratch or use a post-processing algorithm.

Another contribution of this work is the application to deep chest X-ray classification. A body of previous literature has identified biases within the state-of-the-art models trained on large-scale publicly available datasets (Larrazabal et al., 2020; Seyyed-Kalantari et al., 2020, 2021). However, none of these works has investigated the *mitigation* of the identified biases. We believe that the current paper is a valuable contribution to the discussion on achieving group fairness for medical image classifiers. Moreover, the considered intra-processing setting may become particularly pertinent to healthcare applications of machine learning due to the increasing adoption of transfer learning and pre-trained models.

Limitations The current study has several limitations. In particular, the experimental setup is restricted to few neural network architectures. It would be interesting to explore other CNN models, such as DenseNet (Huang et al., 2017) and SqueezeNet (Iandola et al., 2016). For MIMIC-CXR, we have only focused on two protected-attribute-label pairs, and further investigation is warranted. Furthermore, we have considered binary classification under a binary-valued protected variable. Therefore, for practical use-cases, it is necessary to adapt our methods to the multilevel setting and extend them to the bias measures beyond the SPD and EOD, as explored by Zafar et al. (2019).

8. Conclusion

This work considered differentiable proxy functions for statistical parity and equality of opportunity, showing their correspondence to the covariance between the decision boundary of a neural network and the protected attribute. We proposed two novel intra-processing debiasing procedures based on neural network pruning and fine-tuning that utilise these proxies. Our experimental results on tabular data, including MIMIC-III, with fully connected neural networks indicate the viability of the proposed methods, especially compared to model-agnostic post-processing. Furthermore, we applied our and related techniques to mitigate disparity in chest X-ray classifiers trained on MIMIC-CXR and demonstrated that previously reported biases could be reduced without retraining models from scratch.

Future Work In future work, it would be interesting to consider criteria for pruning beyond the gradient-based influence and study a more general setting with multiple classes and protected attribute categories. For tabular data, it could be helpful to investigate the use of pruning in the input layer to remove "biased" variables directly. The experimental results on MIMIC-CXR should be extended by more labels and protected attributes.

Code and Data Availability

The code is available at https://github.com/i6092467/diff-bias-proxies. All of the datasets in our experiments are publicly available.

Acknowledgements

The authors would like to thank Dr. Mona Azadkia, Dr. Alexander Marx, and Imant Daunhawer for valuable discussions and feedback. RM was supported by the SNSF grant #320038189096, EO was supported by the Hasler Foundation grant #21050.

References

Imane Allaouzi and Mohamed Ben Ahmed. A novel approach for multi-label chest X-ray classification of common thorax diseases. *IEEE Access*, 7:64279–64288, 2019.

Drew Altman and William H. Frist. Medicare and Medicaid at 50 years. *JAMA*, 314(4): 384, 2015.

Marco Ancona, Enea Ceolini, Cengiz Öztireli, and Markus Gross. Gradient-based attribution methods. In *Explainable AI: Interpreting, Explaining and Visualizing Deep Learning*, pages 169–191. Springer International Publishing, 2019.

David Bau, Jun-Yan Zhu, Hendrik Strobelt, Agata Lapedriza, Bolei Zhou, and Antonio Torralba. Understanding the role of individual units in a deep neural network. *Proceedings of the National Academy of Sciences*, 117(48):30071–30078, 2020.

Rachel K. E. Bellamy, Kuntal Dey, Michael Hind, Samuel C. Hoffman, Stephanie Houde, Kalapriya Kannan, Pranay Lohia, Jacquelyn Martino, Sameep Mehta, Aleksandra Mojsilovic, Seema Nagar, Karthikeyan Natesan Ramamurthy, John Richards, Diptikalyan Saha, Prasanna Sattigeri, Moninder Singh, Kush R. Varshney, and Yunfeng Zhang. AI Fairness 360: An extensible toolkit for detecting, understanding, and mitigating unwanted algorithmic bias, 2018. arXiv:1810.01943.

Philippe Besse, Eustasio del Barrio, Paula Gordaliza, Jean-Michel Loubes, and Laurent Risser. A survey of bias in machine learning through the prism of statistical parity. *The American Statistician*, pages 1–11, 2021.

Davis Blalock, Jose Javier Gonzalez Ortiz, Jonathan Frankle, and John Guttag. What is the state of neural network pruning? In *Proceedings of Machine Learning and Systems*, volume 2, pages 129–146, 2020.

Keno K. Bressem, Lisa C. Adams, Christoph Erxleben, Bernd Hamm, Stefan M. Niehues, and Janis L. Vahldiek. Comparing different deep learning architectures for classification of chest radiographs. *Scientific Reports*, 10(1), 2020.

Kay Henning Brodersen, Cheng Soon Ong, Klaas Enno Stephan, and Joachim M. Buhmann. The balanced accuracy and its posterior distribution. In *20th International Conference on Pattern Recognition*. IEEE, 2010.

Flavio Calmon, Dennis Wei, Bhanukiran Vinzamuri, Karthikeyan Natesan Ramamurthy, and Kush R Varshney. Optimized pre-processing for discrimination prevention. In *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc., 2017.

L. Elisa Celis, Vijay Keswani, and Nisheeth Vishnoi. Data preprocessing to mitigate bias: A maximum entropy based approach. In *Proceedings of the 37th International Conference on Machine Learning*, volume 119, pages 1349–1359. PMLR, 2020.

Danton S. Char, Nigam H. Shah, and David Magnus. Implementing machine learning in health care — addressing ethical challenges. *New England Journal of Medicine*, 378(11): 981–983, 2018.

Yu Cheng, Duo Wang, Pan Zhou, and Tao Zhang. A survey of model compression and acceleration for deep neural networks, 2017. arXiv:1710.09282.

Shweta Chopra, Nupur Baghel, Shubham Annadate, and Rajalakshmi D. Shanmugasundaram. Fighting algorithmic bias using adversarial networks, 2020. URL https://github.com/choprashweta/Adversarial-Debiasing/blob/master/CIS_519_Project_Report%20(4).pdf.

Joseph Paul Cohen, Mohammad Hashir, Rupert Brooks, and Hadrien Bertrand. On the limits of cross-domain generalization in automated X-ray prediction. In *Medical Imaging with Deep Learning*, 2020.

Sam Corbett-Davies and Sharad Goel. The measure and mismeasure of fairness: A critical review of fair machine learning, 2018. arXiv:1808.00023.

Kedar Dhamdhere, Mukund Sundararajan, and Qiqi Yan. How important is a neuron? In *International Conference on Learning Representations*, 2019.

Priyanka Gupta, Pankaj Malhotra, Lovekesh Vig, and Gautam M. Shroff. Using features from pre-trained TimeNET for clinical predictions. In *KHD@IJCAI*, 2018.

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

Babak Hassibi and David Stork. Second order derivatives for network pruning: Optimal brain surgeon. In *Advances in Neural Information Processing Systems*, volume 5. Morgan-Kaufmann, 1993.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, 2016.

Yihui He, Xiangyu Zhang, and Jian Sun. Channel pruning for accelerating very deep neural networks. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, 2017.

Gao Huang, Zhuang Liu, Laurens van der Maaten, and Kilian Q. Weinberger. Densely connected convolutional networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017.

Forrest N. Iandola, Song Han, Matthew W. Moskewicz, Khalid Ashraf, William J. Dally, and Kurt Keutzer. SqueezeNet: AlexNet-level accuracy with 50x fewer parameters and $<$ 0.5MB model size, 2016. arXiv:1602.07360.

Jeremy Irvin, Pranav Rajpurkar, Michael Ko, Yifan Yu, Silviana Ciurea-Ilcus, Chris Chute, Henrik Marklund, Behzad Haghgoo, Robyn Ball, Katie Shpanskaya, Jayne Seekins, David A. Mong, Safwan S. Halabi, Jesse K. Sandberg, Ricky Jones, David B. Larson, Curtis P. Langlotz, Bhavik N. Patel, Matthew P. Lungren, and Andrew Y. Ng. CheXpert: A large chest radiograph dataset with uncertainty labels and expert comparison. In *33rd AAAI Conference on Artificial Intelligence, AAAI 2019, 31st Innovative Applications of Artificial Intelligence Conference, IAAI 2019 and the 9th AAAI Symposium on Educational Advances in Artificial Intelligence, EAAI 2019*, pages 590–597, 2019.

Alistair E. W. Johnson, Tom J. Pollard, Lu Shen, Li wei H. Lehman, Mengling Feng, Mohammad Ghassemi, Benjamin Moody, Peter Szolovits, Leo Anthony Celi, and Roger G. Mark. MIMIC-III, a freely accessible critical care database. *Scientific Data*, 3(1), 2016.

Alistair E. W. Johnson, Tom J. Pollard, Nathaniel R. Greenbaum, Matthew P. Lungren, Chih-ying Deng, Yifan Peng, Zhiyong Lu, Roger G. Mark, Seth J. Berkowitz, and Steven Horng. MIMIC-CXR-JPG: A large publicly available database of labeled chest radiographs, 2019.

Faisal Kamiran and Toon Calders. Data preprocessing techniques for classification without discrimination. *Knowledge and Information Systems*, 33(1):1–33, 2011.

Faisal Kamiran, Asim Karim, and Xiangliang Zhang. Decision theory for discrimination-aware classification. In *2012 IEEE 12th International Conference on Data Mining*, pages 924–929, 2012.

Toshihiro Kamishima, Shotaro Akaho, Hideki Asoh, and Jun Sakuma. Fairness-aware classifier with prejudice remover regularizer. In *Machine Learning and Knowledge Discovery in Databases*, pages 35–50. Springer Berlin Heidelberg, 2012.

Michael Kearns. Fair algorithms for machine learning. In *Proceedings of the 2017 ACM Conference on Economics and Computation*, page 1. Association for Computing Machinery, 2017.

Byungju Kim, Hyunwoo Kim, Kyungsu Kim, Sungjin Kim, and Junmo Kim. Learning not to learn: Training deep neural networks with biased data. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9012–9020, 2019a.

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. Association for Computing Machinery, 2019b.

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

Ron Kohavi. Scaling up the accuracy of naive-Bayes classifiers: A decision-tree hybrid. In *Proceedings of the Second International Conference on Knowledge Discovery and Data Mining*, pages 202–207. AAAI Press, 1996.

Agostina J. Larrazabal, Nicolás Nieto, Victoria Peterson, Diego H. Milone, and Enzo Ferrante. Gender imbalance in medical imaging datasets produces biased classifiers for computer-aided diagnosis. *Proceedings of the National Academy of Sciences*, 117(23): 12592–12594, 2020.

Jeff Larson, Surya Mattu, Lauren Kirchner, and Julia Angwin. How we analyzed the COMPAS recidivism algorithm. https://www.propublica.org/article/how-we-analyzed-the-compas-recidivism-algorithm, 2016. Accessed: 2020.11.02.

Yann LeCun, John Denker, and Sara Solla. Optimal brain damage. In *Advances in Neural Information Processing Systems*, volume 2. Morgan-Kaufmann, 1990.

Jungsoo Lee, Eungyeup Kim, Juyoung Lee, Jihyeon Lee, and Jaegul Choo. Learning debiased representation via disentangled feature augmentation. In *Advances in Neural Information Processing Systems*, 2021.

Klas Leino, Shayak Sen, Anupam Datta, Matt Fredrikson, and Linyi Li. Influence-directed explanations for deep convolutional networks. In *IEEE International Test Conference (ITC)*. IEEE, 2018.

Wei-Yin Loh, Luxi Cao, and Peigen Zhou. Subgroup identification for precision medicine: A comparative review of 13 methods. *WIREs Data Mining and Knowledge Discovery*, 9 (5), 2019.

Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. In *7th International Conference on Learning Representations, ICLR*, 2019.

Chuizheng Meng, Loc Trinh, Nan Xu, James Enouen, and Yan Liu. Interpretability and fairness evaluation of deep learning models on MIMIC-IV dataset. *Scientific Reports*, 12 (1), 2022.

Pavlo Molchanov, Stephen Tyree, Tero Karras, Timo Aila, and Jan Kautz. Pruning convolutional neural networks for resource efficient inference. In *5th International Conference on Learning Representations, ICLR*, 2017.

Sérgio Moro, Paulo Cortez, and Paulo Rita. A data-driven approach to predict the success of bank telemarketing. *Decision Support Systems*, 62:22–31, 2014.

Junhyun Nam, Hyuntak Cha, Sungsoo Ahn, Jaeho Lee, and Jinwoo Shin. Learning from failure: De-biasing classifier from biased classifier. In *Advances in Neural Information Processing Systems*, volume 33, pages 20673–20684. Curran Associates, Inc., 2020a.

Woo-Jeoung Nam, Shir Gur, Jaesik Choi, Lior Wolf, and Seong-Whan Lee. Relative attributing propagation: Interpreting the comparative contributions of individual units in deep neural networks. *Proceedings of the AAAI Conference on Artificial Intelligence*, 34 (03):2501–2508, 2020b.

Ziad Obermeyer, Brian Powers, Christine Vogeli, and Sendhil Mullainathan. Dissecting racial bias in an algorithm used to manage the health of populations. *Science*, 366(6464): 447–453, 2019.

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. NIPS 2017 Autodiff Workshop.

Geoff Pleiss, Manish Raghavan, Felix Wu, Jon Kleinberg, and Kilian Q. Weinberger. On fairness and calibration. In *Proceedings of the 31st International Conference on Neural Information Processing Systems*, pages 5684–5693. Curran Associates Inc., 2017.

Sanjay Purushotham, Chuizheng Meng, Zhengping Che, and Yan Liu. Benchmarking deep learning models on large healthcare datasets. *Journal of Biomedical Informatics*, 83: 112–134, 2018.

Tai Le Quy, Arjun Roy, Vasileios Iosifidis, Wenbin Zhang, and Eirini Ntoutsi. A survey on datasets for fairness-aware machine learning. *WIREs Data Mining and Knowledge Discovery*, 12(3), 2022.

Maithra Raghu, Chiyuan Zhang, Jon Kleinberg, and Samy Bengio. Transfusion: Understanding transfer learning for medical imaging. In *Advances in Neural Information Processing Systems*, volume 32. Curran Associates, Inc., 2019.

Alvin Rajkomar, Michaela Hardt, Michael D. Howell, Greg Corrado, and Marshall H. Chin. Ensuring fairness in machine learning to advance health equity. *Annals of Internal Medicine*, 169(12):866–872, 2018.

Pranav Rajpurkar, Jeremy Irvin, Kaylie Zhu, Brandon Yang, Hershel Mehta, Tony Duan, Daisy Ding, Aarti Bagul, Curtis Langlotz, Katie Shpanskaya, et al. Chexnet: Radiologist-level pneumonia detection on chest X-rays with deep learning. *arXiv preprint arXiv:1711.05225*, 2017.

Laila Rasmy, Yang Xiang, Ziqian Xie, Cui Tao, and Degui Zhi. Med-BERT: pretrained contextualized embeddings on large-scale structured electronic health records for disease prediction. *npj Digital Medicine*, 4(1), 2021.

Christian Reimers, Paul Bodesheim, Jakob Runge, and Joachim Denzler. Conditional adversarial debiasing: Towards learning unbiased classifiers from biased data. In *Pattern Recognition*, pages 48–62. Springer International Publishing, 2021.

Yash Savani, Colin White, and Naveen Sundar Govindarajulu. Intra-processing methods for debiasing neural networks. In *Advances in Neural Information Processing Systems*, volume 33, pages 2798–2810. Curran Associates, Inc., 2020.

Laleh Seyyed-Kalantari, Guanxiong Liu, Matthew McDermott, Irene Y Chen, and Marzyeh Ghassemi. CheXclusion: Fairness gaps in deep chest X-ray classifiers. In *BIOCOMPUTING 2021: Proceedings of the Pacific Symposium*, pages 232–243. World Scientific, 2020.

Laleh Seyyed-Kalantari, Haoran Zhang, Matthew B. A. McDermott, Irene Y. Chen, and Marzyeh Ghassemi. Underdiagnosis bias of artificial intelligence algorithms applied to chest radiographs in under-served patient populations. *Nature Medicine*, 27(12):2176–2182, 2021.

Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. In *3rd International Conference on Learning Representations, ICLR*, 2015.

Suraj Srinivas and François Fleuret. Full-gradient representation for neural network visualization. In *Advances in Neural Information Processing Systems*, volume 32. Curran Associates, Inc., 2019.

Nitish Srivastava, Geoffrey Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov. Dropout: A simple way to prevent neural networks from overfitting. *Journal of Machine Learning Research*, 15(1):1929–1958, 2014.

Xiaosong Wang, Yifan Peng, Le Lu, Zhiyong Lu, Mohammadhadi Bagheri, and Ronald M. Summers. ChestX-ray: Hospital-scale chest X-ray database and benchmarks on weakly supervised classification and localization of common thorax diseases. In *Deep Learning and Convolutional Neural Networks for Medical Imaging and Clinical Informatics*, pages 369–392. Springer International Publishing, 2019.

Wei Wen, Chunpeng Wu, Yandan Wang, Yiran Chen, and Hai Li. Learning structured sparsity in deep neural networks. In *Advances in Neural Information Processing Systems*, volume 29. Curran Associates, Inc., 2016.

Jenna Wiens, Suchi Saria, Mark Sendak, Marzyeh Ghassemi, Vincent X. Liu, Finale Doshi-Velez, Kenneth Jung, Katherine Heller, David Kale, Mohammed Saeed, Pilar N. Ossorio, Sonoo Thadaney-Israni, and Anna Goldenberg. Do no harm: a roadmap for responsible machine learning for health care. *Nature Medicine*, 25(9):1337–1340, 2019.

Muhammad Bilal Zafar, Isabel Valera, Manuel Gomez Rogriguez, and Krishna P. Gummadi. Fairness Constraints: Mechanisms for Fair Classification. In *Proceedings of the 20th International Conference on Artificial Intelligence and Statistics*, volume 54, pages 962–970. PMLR, 2017.

Muhammad Bilal Zafar, Isabel Valera, Manuel Gomez-Rodriguez, and Krishna P. Gummadi. Fairness constraints: A flexible approach for fair classification. *Journal of Machine Learning Research*, 20(75):1–42, 2019.

John R. Zech, Marcus A. Badgeley, Manway Liu, Anthony B. Costa, Joseph J. Titano, and Eric Karl Oermann. Variable generalization performance of a deep learning model to detect pneumonia in chest radiographs: A cross-sectional study. *PLOS Medicine*, 15(11): e1002683, 2018.

Rich Zemel, Yu Wu, Kevin Swersky, Toni Pitassi, and Cynthia Dwork. Learning fair representations. In *Proceedings of the 30th International Conference on Machine Learning*, volume 28, 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*. ACM, 2018.

Appendix A. Decision Boundary Covariance

In this appendix, we study the relationship between the differentiable bias proxies in Equations 3 and 4 (see Section 4 of the main text) and the covariance between the decision boundary of the classifier $f_{\boldsymbol{\theta}}(\cdot)$ and the protected attribute A.

Lemma 1 *For $\mathcal{X} = \{\boldsymbol{x}_i\}_{i=1}^{N}$, $\mathcal{Y} = \{y_i\}_{i=1}^{N}$, and $\mathcal{A} = \{a_i\}_{i=1}^{N}$ and some classifier $f_{\boldsymbol{\theta}}(\cdot)$, $-\tilde{\mu}_{\mathrm{SPD}}\left(f_{\boldsymbol{\theta}},\, \mathcal{X},\, \mathcal{Y},\, \mathcal{A}\right) \propto \widehat{\mathrm{Cov}}\left(A,\, f_{\boldsymbol{\theta}}\left(\boldsymbol{X}\right)\right)$.*

Proof Recall that the covariance is given by

$$\mathrm{Cov}\left(A,\, f_{\boldsymbol{\theta}}\left(\boldsymbol{X}\right)\right) = \mathbb{E}\left[A f_{\boldsymbol{\theta}}\left(\boldsymbol{X}\right)\right] - \mathbb{E}\left[A\right]\mathbb{E}\left[f_{\boldsymbol{\theta}}(\boldsymbol{X})\right].$$

Let $K = \sum_{i=1}^{N} a_i$ and $\overline{f_{\boldsymbol{\theta}}\left(\boldsymbol{x}\right)} = \frac{1}{N}\sum_{i}^{N} f_{\boldsymbol{\theta}}\left(\boldsymbol{x}_i\right)$, consider an empirical estimate

$$\widehat{\mathrm{Cov}}\left(A,\, f_{\boldsymbol{\theta}}\left(\boldsymbol{X}\right)\right) = \frac{1}{N}\sum_{i=1}^{N} f_{\boldsymbol{\theta}}\left(\boldsymbol{x}_i\right) a_i - \frac{K}{N^2}\sum_{i=1}^{N} f_{\boldsymbol{\theta}}\left(\boldsymbol{x}_i\right) = \frac{1}{N}\sum_{i=1}^{N} f_{\boldsymbol{\theta}}\left(\boldsymbol{x}_i\right) a_i - \frac{K}{N}\overline{f_{\boldsymbol{\theta}}\left(\boldsymbol{x}\right)}. \tag{A.1}$$

Observe that

$$\begin{aligned} -\tilde{\mu}_{\mathrm{SPD}} =& \frac{\sum_{i=1}^{N} f_{\boldsymbol{\theta}}\left(\boldsymbol{x}_i\right) a_i}{\sum_{i=1}^{N} a_i} - \frac{\sum_{i=1}^{N} f_{\boldsymbol{\theta}}\left(\boldsymbol{x}_i\right)\left(1-a_i\right)}{\sum_{i=1}^{N}\left(1-a_i\right)} = \frac{1}{K}\sum_{i=1}^{N} f_{\boldsymbol{\theta}}\left(\boldsymbol{x}_i\right) a_i - \frac{N}{N-K}\overline{f_{\boldsymbol{\theta}}\left(\boldsymbol{x}\right)} - \\ & \frac{1}{N-K}\sum_{i=1}^{N} f_{\boldsymbol{\theta}}\left(\boldsymbol{x}_i\right) a_i = \frac{N}{K(N-K)}\sum_{i=1}^{N} f_{\boldsymbol{\theta}}\left(\boldsymbol{x}_i\right) a_i - \frac{NK}{K(N-K)}\overline{f_{\boldsymbol{\theta}}\left(\boldsymbol{x}\right)}. \end{aligned} \tag{A.2}$$

Note that (A.1) $\propto$ (A.2) by a factor of $\frac{N^2}{K(N-K)}$, constant in $\boldsymbol{\theta}$. ■

Lemma 2 *For $\mathcal{X} = \{\boldsymbol{x}_i\}_{i=1}^{N}$, $\mathcal{Y} = \{y_i\}_{i=1}^{N}$, and $\mathcal{A} = \{a_i\}_{i=1}^{N}$ and some classifier $f_{\boldsymbol{\theta}}(\cdot)$, $-\tilde{\mu}_{\mathrm{EOD}}\left(f_{\boldsymbol{\theta}},\, \mathcal{X},\, \mathcal{Y},\, \mathcal{A}\right) \propto \widehat{\mathrm{Cov}}\left(A,\, f_{\boldsymbol{\theta}}\left(\boldsymbol{X}\right) \mid Y=1\right)$.*

Proof Recall that, by the law of total covariance,

$$\begin{aligned} \mathrm{Cov}\left(A,\, f_{\boldsymbol{\theta}}\left(\boldsymbol{X}\right) \mid Y=1\right) =& \mathbb{E}\left[\left(A - \mathbb{E}\left[A \mid Y=1\right]\right)\left(f_{\boldsymbol{\theta}}\left(\boldsymbol{X}\right) - \mathbb{E}\left[f_{\boldsymbol{\theta}}\left(\boldsymbol{X}\right) \mid Y=1\right]\right) \mid Y=1\right] = \\ & \mathbb{E}\left[A f_{\boldsymbol{\theta}}\left(\boldsymbol{X}\right) \mid Y=1\right] - \mathbb{E}\left[A \mid Y=1\right]\mathbb{E}\left[f_{\boldsymbol{\theta}}\left(\boldsymbol{X}\right) \mid Y=1\right]. \end{aligned}$$

Let $M = \sum_{i=1}^{N} y_i$, $R = \sum_{i=1}^{N} a_i y_i$, and $\overline{f_{\boldsymbol{\theta}}\left(\boldsymbol{x}\right)} = \frac{1}{N}\sum_{i}^{N} f_{\boldsymbol{\theta}}\left(\boldsymbol{x}_i\right)$, consider an empirical estimate

$$\begin{aligned} \widehat{\mathrm{Cov}}\left(A,\, f_{\boldsymbol{\theta}}\left(\boldsymbol{X}\right) \mid Y=1\right) =& \frac{\sum_{i=1}^{N} f_{\boldsymbol{\theta}}\left(\boldsymbol{x}_i\right) a_i y_i}{\sum_{i=1}^{N} y_i} - \frac{\sum_{i=1}^{N} a_i y_i}{\sum_{i=1}^{N} y_i} \cdot \frac{\sum_{i=1}^{N} f_{\boldsymbol{\theta}}\left(\boldsymbol{x}_i\right) y_i}{\sum_{i=1}^{N} y_i} = \\ & \frac{1}{M}\sum_{i=1}^{N} f_{\boldsymbol{\theta}}\left(\boldsymbol{x}_i\right) a_i y_i - \frac{R}{M^2}\sum_{i=1}^{N} f_{\boldsymbol{\theta}}\left(\boldsymbol{x}_i\right) y_i. \end{aligned} \tag{A.3}$$

Observe that

$$
\begin{aligned}
-\tilde{\mu}_{\mathrm{EOD}} = & \frac{\sum_{i=1}^{N} f_{\boldsymbol{\theta}}\left(\boldsymbol{x}_i\right) y_i a_i}{\sum_{i=1}^{N} y_i a_i} - \frac{\sum_{i=1}^{N} f_{\boldsymbol{\theta}}\left(\boldsymbol{x}_i\right) y_i (1-a_i)}{\sum_{i=1}^{N} y_i (1-a_i)} = \\
& \frac{1}{R}\sum_{i=1}^{N} f_{\boldsymbol{\theta}}\left(\boldsymbol{x}_i\right) y_i a_i - \frac{1}{M-R}\sum_{i=1}^{N} f_{\boldsymbol{\theta}}\left(\boldsymbol{x}_i\right) y_i - \frac{1}{M-R}\sum_{i=1}^{N} f_{\boldsymbol{\theta}}\left(\boldsymbol{x}_i\right) y_i a_i = \\
& \frac{M}{R(M-R)}\sum_{i=1}^{N} f_{\boldsymbol{\theta}}\left(\boldsymbol{x}_i\right) y_i a_i - \frac{1}{M-R}\sum_{i=1}^{N} f_{\boldsymbol{\theta}}\left(\boldsymbol{x}_i\right) y_i .
\end{aligned}
\tag{A.4}
$$

Note that (A.3) $\propto$ (A.4) by a factor of $\frac{M^2}{R(M-R)}$, constant in $\boldsymbol{\theta}$. ■

Appendix B. Datasets

Nonclinical tabular benchmarking datasets used in our experiments (see Section 6.1) are publicly available in the IBM AIF 360 library (Bellamy et al., 2018). Table B.1 below summarises all real-world datasets considered throughout the paper.

Table B.1: Summary of the datasets. N_{train}, N_{valid}, and N_{test} are the sizes of the training, validation, and test sets, respectively; D is the input dimensionality after preprocessing; and A is the protected attribute.

Dataset	N_{train}	N_{valid}	N_{test}	D	A	**Architecture**
Adult	27,133	9,044	9,045	98	*Sex*	FCNN
Bank	18,292	6,098	6,098	57	*Age*	FCNN
COMPAS	3,700	1,233	1,234	401	*Race*	FCNN
MIMIC-III	21,595	7,199	7,199	43	*Age*	FCNN
	21,595	7,199	7,199	44	*Marital Status*	FCNN
	21,595	7,199	7,199	44	*Insurance Type*	FCNN
MIMIC-CXR	5,528	3,368	3,426	224×224	*Sex*	CNN
	3,984	930	1,122	224×224	*Ethnicity*	CNN

Appendix C. Syntehtic Data

In addition to the real-world data (see Section 5 and Appendix B), we conducted experiments on two synthetic datasets adapted from the literature (for the results, see Appendix E.3). Below we summarise their generative process.

Synthetic by Loh et al. (2019) Loh et al. (2019) performed extensive simulation experiments comparing subgroup identification methods. Their simulation models are suitable for benchmarking debiasing algorithms. We adopted one of their synthetic datasets with the following generative procedure. For N independent data points:

1. Randomly draw features with marginal distributions given by $X_{1,\,2,\,3,\,7,\,8,\,9,\,10} \sim \mathcal{N}\left(0,\,1\right)$, $X_4 \sim \mathrm{Exp}(1)$, $X_5 \sim \mathrm{Bernoulli}(\frac{1}{2})$, $X_6 \sim \mathrm{Cat}(10)$ and $\mathrm{corr}\left(X_2, X_3\right) = 0.5$ and $\mathrm{corr}\left(X_j, X_k\right) = 0.5$, for $j,\,k \in \{7,\,8,\,9,\,10\}$, $j \neq k$.

2. Randomly draw the protected attribute $A \sim \text{Bernoulli}\left(\frac{1}{2}\right)$.

3. Let
$$\text{logit} = \log \frac{\mathbb{P}(Y=1)}{\mathbb{P}(Y=0)} = \frac{1}{2}\left(X_1 + X_2 - X_5\right) + 2\alpha A \mathbf{1}_{\{X_6 \pmod 2)=1\}}, \tag{C.1}$$
where $\mathbf{1}_{\{\cdot\}}$ is an indicator function and $\alpha > 0$ is the parameter controlling the magnitude of the correlation between Y and A.

4. Randomly draw the binary classification label $Y \sim \text{Bernoulli}\left(\frac{\exp(\text{logit})}{\exp(\text{logit})+1}\right)$.

Although this dataset is relatively simplistic, the simulation allows controlling the magnitude of classification disparity in the original classifier. In practice, we observe that the higher the value of α, the higher the absolute SPD or EOD of the classifier trained on features $X_{1:10}$ and labels Y.

Synthetic by Zafar et al. (2017) Zafar et al. (2017) proposed another simple simulation model for generating datasets with different degrees of disparity in classification outcomes. We extended their model[1] to higher dimensionality and classes that are not linearly separable. The data generating process is specified below. For N independent data points:

1. Randomly draw the binary classification label $Y \sim \text{Bernoulli}\left(\frac{1}{2}\right)$.

2. If $Y = 0$, randomly draw $\tilde{\boldsymbol{X}} \sim \mathcal{N}_2\left(\begin{bmatrix}-2\\-2\end{bmatrix}, \begin{bmatrix}10 & 1\\1 & 3\end{bmatrix}\right)$;
otherwise $\tilde{\boldsymbol{X}} \sim \mathcal{N}_2\left(\begin{bmatrix}2\\2\end{bmatrix}, \begin{bmatrix}5 & 1\\1 & 5\end{bmatrix}\right)$.

3. Let
$$\tilde{\boldsymbol{X}}' = \begin{bmatrix}\cos(\vartheta) & -\sin(\vartheta)\\ \sin(\vartheta) & \cos(\vartheta)\end{bmatrix}\tilde{\boldsymbol{X}}, \tag{C.2}$$
where ϑ is the rotation angle controlling the correlation between between Y and A.

4. Let
$$\mathbb{P}\left(A=1\right) = \frac{p\left(\tilde{\boldsymbol{X}}' \,|\, Y=1\right)}{p\left(\tilde{\boldsymbol{X}}' \,|\, Y=1\right) + p\left(\tilde{\boldsymbol{X}}' \,|\, Y=0\right)}.$$

5. Randomly draw the protected attribute $A \sim \text{Bernoulli}\left(\mathbb{P}\left(A=1\right)\right)$.

6. Let $g(\tilde{\boldsymbol{x}}) = \boldsymbol{\Theta}_2\text{ReLU}\left(\boldsymbol{\Theta}_1\text{ReLU}\left(\boldsymbol{\Theta}_0\tilde{\boldsymbol{x}} + \boldsymbol{b}_0\right) + \boldsymbol{b}_1\right) + \boldsymbol{b}_2$, where $\boldsymbol{\Theta}_0 \in \mathbb{R}^{h\times 2}$, $\boldsymbol{\Theta}_1 \in \mathbb{R}^{h\times h}$, $\boldsymbol{\Theta}_2 \in \mathbb{R}^{p\times h}$ and $\boldsymbol{b}_0 \in \mathbb{R}^h$, $\boldsymbol{b}_1 \in \mathbb{R}^h$, $\boldsymbol{b}_2 \in \mathbb{R}^p$ are randomly generated matrices and vectors.

7. Let $\boldsymbol{X} = g\left(\tilde{\boldsymbol{X}}\right)$ be a p-dimensional real-valued feature vector.

Similar to the dataset above, this simulation allows controlling the degree of bias by adjusting the parameter ϑ. In practice, values of ϑ closer to zero result in classifiers with higher absolute SPDs and EODs.

[1] https://github.com/mbilalzafar/fair-classification

Appendix D. Implementation Details

In this appendix, we present further implementation details. For a general overview of the experimental setup, see Section 5 of the main text.

D.1. Train-validation-test Split

All tabular datasets were split into 60% train, 20% validation, and 20% test instances. For MIMIC-CXR, we performed a 50%-25%-25% train-validation-test split stratified by *patients* to avoid data leakage. To mitigate other sources of variability in the original models, we balanced the training, validation, and test sets w.r.t. the number of healthy and pathological cases. Since the prior work (Larrazabal et al., 2020) has investigated imbalance w.r.t. the protected attribute as a potential cause of bias in deep chest X-ray classifiers, we sampled the images of patients so that the training set had 75% and 25% from the privileged and unprivileged groups, respectively. For validation and test sets, the ratio of privileged and unprivileged groups was 50%-50%.

D.2. Model Development

All experiments and methods were implemented in PyTorch (v 1.9.1) (Paszke et al., 2017). For all tabular datasets, we used the same architecture and training scheme for the classifier $f_{\boldsymbol{\theta}}(\cdot)$. We trained a fully connected feedforward neural network with ten hidden layers, 32 units each, ReLU activations, dropout ($p = 0.05$), and batch normalisation (see Table D.1). The network was trained for 1,000 epochs with early stopping by minimising the binary cross-entropy loss using the Adam optimiser (Kingma and Ba, 2015) with `ReduceLROnPlateau` learning rate schedule and mini-batch size of 64.

Table D.1: Fully connected neural network architecture used in debiasing experiments on tabular data (see Section 6.1). `nn` stands for `torch.nn`; `F` stands for `torch.nn.functional`; `input_dim` corresponds to the number of features d.

	Classifier

1	`nn.Linear(input_dim, 32)`
	`F.relu()`
	`nn.Dropout(0.05)`
	`nn.BatchNorm1d(32)`

2	`for l in range(10):`
	`   nn.Linear(32, 32)`
	`   F.relu()`
	`   nn.Dropout(0.05)`
	`   nn.BatchNorm1d(32)`

3	`out = nn.Linear(32, 1)`
4	`torch.sigmoid()`

For MIMIC-CXR, we used classifiers based on the VGG-16 (Simonyan and Zisserman, 2015) and ResNet-18 (He et al., 2016) networks. We initialised classifiers with pre-trained weights and trained them using the binary cross-entropy loss and the AdamW optimiser (Loshchilov and Hutter, 2019) with default parameters, a mini-batch size of 32, and an initial learning rate of 10^{-4} with `StepLR` learning rate schedule. The networks were trained for a maximum of 20 epochs with early stopping on the validation set.

D.3. Method Implementation

We used the following implementation of the debiasing algorithms:

- Random and Adv. Intra: we used the original implementation by Savani et al. (2020) available at https://github.com/abacusai/intraprocessing_debiasing.
- ROC and Eq. Odds: we used the implementation available in the AIF 360 toolkit at https://github.com/Trusted-AI/AIF360.
- Pruning and Bias GD/A: the PyTorch (v 1.9.1) (Paszke et al., 2017) implementation is available at https://github.com/i6092467/diff-bias-proxies.

D.4. Hyperparameters

Table D.2 provides hyperparameter values for the pruning and bias GD/A. For both algorithms, the most sensitive hyperparameter is the lower bound on performance ϱ (see Algorithms 1 and 2), which effectively controls the decrease in performance as a result of debiasing. For MIMIC-III, the same hyperparameter configuration was used across all protected attributes ("*age*", "*marital status*", and "*insurance type*"). For MIMIC-CXR experiments with VGG-16, we performed pruning only in the convolutional layers. For ResNet-18, we only pruned the first `conv1` block, comprising a single convolutional layer. During experimentation, we observed little gain from pruning additional downstream layers.

Table D.2: Hyperparameter values used for the (a) pruning and (b) bias GD/A algorithms throughout the experiments. Herein, "# units" corresponds to the number of units pruned per step and relates to the hyperparameter B from Algorithm 1. ϱ_{SPD} and ϱ_{EOD} denote lower bounds on the balanced accuracy for the SPD and EOD experiments, respectively. For the bias GD/A, η denotes the learning rate; M is the mini-batch size; and E is the number of epochs.

(a) Pruning

Dataset	**# Units**	ϱ_{SPD}	ϱ_{EOD}
Adult	1	0.52	0.75
Bank	1	0.80	0.70
COMPAS	1	0.55	0.55
MIMIC-III	1	0.60	0.60
MIMIC-CXR, *Sex*	1,500	—	0.65
MIMIC-CXR, *Ethnicity*	1,500	—	0.55

(b) Bias GD/A

Dataset	η	M	E	ϱ_{SPD}	ϱ_{EOD}
Adult	1.0e-5	256	200	0.62	0.80
Bank	1.0e-5	256	200	0.70	0.70
COMPAS	1.0e-5	256	200	0.61	0.58
MIMIC-III	1.0e-5	256	100	0.60	0.60
MIMIC-CXR, *Sex*	1.0e-5	32	10	—	0.75
MIMIC-CXR, *Ethnicity*	7.5e-6	32	8	—	0.55

For the random perturbation intra-processing, we used multiplicative noise distributed as $\mathcal{N}(1, 0.01)$ and performed 101 perturbations to maximise the constrained objective proposed by Savani et al. (2020) with an upper/lower bound on the bias of ±0.05 and a margin of 0.01. The same bias bounds were used for the ROC post-processing procedure. For adversarial intra-processing, on tabular datasets, we used hyperparameter values from the original work by Savani et al. (2020): a critic network with three hidden layers, a learning rate of 10^{-3}, 16 training epochs, a mini-batch size of 64, $\lambda = 0.75$, 201 and 101 critic and actor training steps, respectively. Similarly to the random perturbation, we utilised the constrained objective with an upper/lower bound on the bias of ±0.05 and a margin of 0.01. Different from the tabular experiments, for MIMIC-CXR and the protected attribute "*sex*", we used a learning rate of 10^{-4}, five training epochs, an upper/lower bound on the bias of ±0.03, and a margin of 0.02. For MIMIC-CXR and "*ethnicity*", we trained networks for four epochs under the constrained objective with an upper/lower bound of ±0.05, and a margin of 0.01. We attempted tuning the number of epochs, critic and actor steps, and λ, however, we did not observe improved results in both MIMIC-CXR experiments.

Appendix E. Further Results

E.1. Further Quantitative Results

Table E.1: Equal opportunity difference and balanced accuracy attained on the *validation* set before and after debiasing VGG-16 (a,b) and ResNet-18 (c,d) trained on MIMIC-CXR to predict (a,c) enlarged cardiomediastinum (with the protected attribute "*sex*") and (b,d) pneumonia (with the protected attribute "*ethnicity*"). Test set results can be found in Table 3, Section 6.2.

(a) Enlarged CM, *Sex*; VGG-16

Method	EOD	BA
Standard	-0.06±0.03	0.77±0.01
Random	-0.02±0.01	0.76±0.01
ROC	-0.05±0.01	0.74±0.04
Eq. Odds	0.00±0.01	0.75±0.01
Adv. Intra	-0.01±0.01	0.97±0.02
Pruning	0.00±0.03	0.76±0.01
Bias GD/A	-0.01±0.02	0.76±0.01

(b) Pneumonia, *Ethnicity*; VGG-16

Method	EOD	BA
Standard	-0.14±0.05	0.75±0.01
Random	-0.07±0.04	0.73±0.02
ROC	-0.05±0.01	0.65±0.06
Eq. Odds	0.00±0.01	0.71±0.02
Adv. Intra	-0.06±0.05	0.93±0.05
Pruning	0.01±0.02	0.72±0.03
Bias GD/A	0.00±0.03	0.73±0.02

(c) Enlarged CM, *Sex*; ResNet-18

Method	EOD	BA
Standard	-0.06±0.04	0.76±0.01
Random	0.00±0.01	0.74±0.02
ROC	-0.04±0.01	0.74±0.04
Eq. Odds	0.00±0.01	0.74±0.01
Adv. Intra	0.00±0.01	0.99±0.02
Pruning	0.00±0.02	0.74±0.02
Bias GD/A	0.00±0.02	0.76±0.01

(d) Pneumonia, *Ethnicity*; ResNet-18

Method	EOD	BA
Standard	-0.13±0.05	0.74±0.01
Random	-0.01±0.01	0.67±0.04
ROC	-0.04±0.01	0.65±0.05
Eq. Odds	0.00±0.01	0.71±0.02
Adv. Intra	0.00±0.00	1.00±0.00
Pruning	0.00±0.03	0.71±0.02
Bias GD/A	0.00±0.03	0.73±0.02

E.2. Comparison with Adversarial In-processing

Since the proposed bias GD/A procedure (see Algorithm 2) bears similarity to the adversarial in-processing method by Zhang et al. (2018), we additionally evaluated models trained from scratch with an adversary for predicting the protected attribute based on the classifier's output as described by Zhang et al. (2018). The evaluation was performed on the same datasets (see Table B.1) and with the same setup as in the experiments from the main body of the paper. Notably, debiased models were trained directly on the training set and *not* on the validation data, as for intra- and post-processing. For the tabular datasets, we used the implementation available in the AIF 360 toolkit (Bellamy et al., 2018). For the MIMIC-CXR, we adapted the publicly available implementation[2] by Chopra et al. (2020).

Table E.2 reports bias and balanced accuracy of the models trained using adversarial in-processing across all dataset-protected-attribute pairs and network architectures. Encouragingly, the method by Zhang et al. (2018) performed comparably or slightly worse on average than pruning and bias GD/A (cf. Tables 1, 2, and 3). For MIMIC-CXR, we observed a pattern similar to intra-processing where the method failed to remove the bias associated with the attribute "*ethnicity*". Thus, in the latter setting, post-processing, which adjusts the model's predictions at test time based on the protected attribute value, is the only effective family of techniques considered.

Table E.2: Test-set bias and balanced accuracy attained by the networks trained using adversarial in-processing. Models were trained separately for the SPD and EOD.

Experiment	**SPD**	**BA**	**EOD**	**BA**
Adult, *Sex*	-0.02±0.00	0.55±0.01	0.03±0.02	0.79±0.01
Bank, *Age*	0.06±0.04	0.69±0.07	-0.04±0.06	0.86±0.01
COMPAS, *Race*	0.05±0.04	0.60±0.03	0.07±0.04	0.62±0.02
MIMIC-III, *Age*	-0.04±0.02	0.68±0.04	0.06±0.03	0.70±0.02
MIMIC-III, *Marital Status*	0.02±0.03	0.74±0.02	0.00±0.04	0.73±0.01
MIMIC-III, *Insurance Type*	-0.02±0.02	0.71±0.03	0.07±0.03	0.72±0.02
MIMIC-CXR, *Sex*, VGG-16	—	—	0.00±0.03	0.75±0.01
MIMIC-CXR, *Sex*, ResNet-18	—	—	-0.01±0.13	0.72±0.03
MIMIC-CXR, *Ethnicity*, VGG-16	—	—	-0.13±0.05	0.71±0.03
MIMIC-CXR, *Ethnicity*, ResNet-18	—	—	-0.13±0.07	0.71±0.02

In summary, the results above suggest that, despite relying on the smaller validation set and resorting to editing the model's parameters *post hoc*, for the considered datasets, intra-processing methods achieve the model's bias and performance that are comparable to those of the network retrained from scratch on the original training set. This experiment further supports the viability of the intra-processing approach adopted by us.

[2] https://github.com/choprashweta/Adversarial-Debiasing

E.3. Sensitivity to Initial Conditions

As observed before (see Section 6.1), the performance of the classifier debiased using pruning or bias GD/A can vary considerably, for instance, for the Adult dataset (see Table 1). To investigate the sensitivity of the proposed methods to initial conditions, particularly, to the degree of bias within the original classifier, we performed further experiments on two synthetic datasets described in Appendix C. We trained and debiased FC neural networks (see Table D.1) while varying the correlation between the label and protected attribute. Intuitively, we expect debiasing to be less effective when the bias of the classifier is high.

For the dataset by Loh et al. (2019), we trained and debiased classifiers under different values of the parameter $\alpha \in [0.0, 2.5]$ (see Equation C.1). The resulting SPD varies between approximately 0.0 to 0.4, and the EOD is between 0.0 and 0.5. Table E.3(a) shows changes in the BA and SPD of the original classifier and the network obtained after pruning and bias GD/A. Notably, both methods exhibit similar patterns. For $\alpha \in [0.0, 1.5]$, debiased classifiers retain a BA of approximately 0.63, which corresponds to an unbiased performance and reduce the bias to zero with low variance. In contrast, for $\alpha > 1.5$, the variance of the disparity across seeds increases considerably, e.g., for $\alpha = 2.5$, pruning yields an SPD of 0.01±0.13. Similar patterns occur when debiasing w.r.t. the EOD (see Table E.3(b)).

For the dataset by Zafar et al. (2017), we varied the value of the parameter $\vartheta \in [0.7, 1.2]$ (see Equation C.2). Tables E.3(c-d) contain the results across the range of rotation angles. Analogously to the synthetic dataset by Loh et al. (2019), we observe either a decrease in the BA or an increase in the residual bias for lower values of the parameter ϑ, i.e. under a higher initial bias. In summary, while proposed techniques successfully mitigate disparity, when the bias of the original classifier is relatively high, debiasing may either fail or lead to a considerable decrease in predictive performance.

Table E.3: Changes in the balanced accuracy and bias of the original and debiased classifier, given by the SPD (a, c) and EOD (b, d) across varying simulation parameters for the synthetic datasets by Loh et al. (2019) (a-b) and Zafar et al. (2017) (c-d). Averages and standard deviations are reported across ten independent simulations.

(a) Synthetic by Loh et al. (2019), SPD

α	**Standard, BA**	**Pruning, BA**	**Bias GD/A, BA**	**Standard, Bias**	**Pruning, Bias**	**Bias GD/A, Bias**
0.1	0.63±0.00	0.63±0.01	0.62±0.01	-0.05±0.02	0.01±0.02	0.00±0.02
0.5	0.64±0.01	0.63±0.02	0.64±0.01	-0.22±0.02	-0.02±0.03	-0.02±0.02
1.0	0.68±0.01	0.63±0.02	0.66±0.01	-0.35±0.03	0.00±0.01	-0.02±0.03
1.5	0.71±0.01	0.63±0.02	0.66±0.03	-0.39±0.03	0.03±0.05	-0.01±0.04
2.0	0.73±0.01	0.63±0.04	0.56±0.07	-0.39±0.03	-0.03±0.10	0.03±0.08
2.5	0.73±0.00	0.65±0.03	0.57±0.07	-0.41±0.03	0.01±0.13	0.04±0.09

(b) Synthetic by Loh et al. (2019), EOD

α	**Standard, BA**	**Pruning, BA**	**Bias GD/A, BA**	**Standard, Bias**	**Pruning, Bias**	**Bias GD/A, Bias**
0.1	0.63±0.00	0.62±0.01	0.62±0.01	-0.04±0.02	0.01±0.02	0.01±0.02
0.5	0.64±0.01	0.63±0.01	0.64±0.01	-0.22±0.02	-0.01±0.03	-0.02±0.02
1.0	0.68±0.01	0.64±0.01	0.66±0.01	-0.39±0.05	-0.01±0.02	-0.03±0.02
1.5	0.71±0.01	0.63±0.01	0.64±0.03	-0.45±0.04	0.02±0.04	-0.01±0.03
2.0	0.73±0.01	0.63±0.03	0.63±0.02	-0.47±0.04	0.01±0.05	0.00±0.04
2.5	0.73±0.00	0.62±0.05	0.64±0.05	-0.50±0.05	0.04±0.11	-0.04±0.08

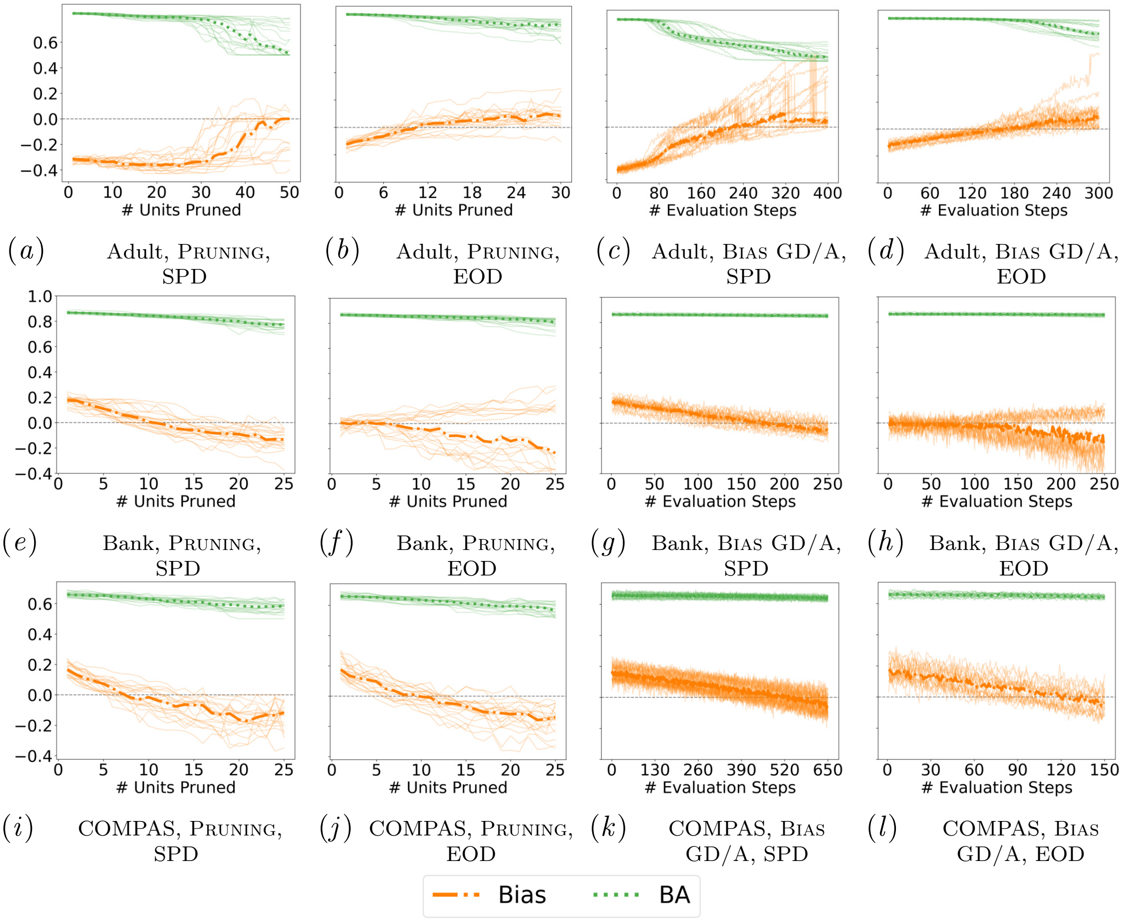
(c) Synthetic by Zafar et al. (2017), SPD

ϑ	**Standard, BA**	**Pruning, BA**	**Bias GD/A, BA**	**Standard, Bias**	**Pruning, Bias**	**Bias GD/A, Bias**
1.2	0.87±0.00	0.84±0.03	0.87±0.00	-0.03±0.01	-0.01±0.02	-0.01±0.01
1.1	0.87±0.00	0.72±0.04	0.83±0.03	-0.13±0.01	0.01±0.02	-0.03±0.04
1.0	0.87±0.00	0.58±0.03	0.74±0.04	-0.23±0.02	-0.09±0.03	-0.02±0.04
0.9	0.87±0.00	0.58±0.04	0.66±0.04	-0.33±0.01	-0.11±0.04	-0.02±0.03
0.8	0.87±0.00	0.62±0.06	0.59±0.02	-0.42±0.01	-0.18±0.08	-0.02±0.03
0.7	0.87±0.00	0.60±0.04	0.57±0.02	-0.49±0.01	-0.18±0.07	-0.03±0.03

(d) Synthetic by Zafar et al. (2017), EOD

ϑ	**Standard, BA**	**Pruning, BA**	**Bias GD/A, BA**	**Standard, Bias**	**Pruning, Bias**	**Bias GD/A, Bias**
1.2	0.87±0.00	0.76±0.06	0.87±0.00	-0.05±0.01	-0.01±0.01	-0.01±0.01
1.1	0.87±0.00	0.76±0.04	0.86±0.02	-0.09±0.01	-0.01±0.01	-0.03±0.03
1.0	0.87±0.00	0.74±0.04	0.86±0.01	-0.12±0.02	0.00±0.03	-0.05±0.03
0.9	0.87±0.00	0.73±0.04	0.84±0.02	-0.16±0.02	-0.03±0.04	-0.05±0.02
0.8	0.87±0.00	0.77±0.06	0.83±0.03	-0.20±0.02	-0.05±0.06	-0.07±0.05
0.7	0.87±0.00	0.75±0.05	0.83±0.03	-0.23±0.02	-0.05±0.07	-0.09±0.03

E.4. Further Qualitative Results



(*a*) Adult, Pruning, SPD

(*b*) Adult, Pruning, EOD

(*c*) Adult, Bias GD/A, SPD

(*d*) Adult, Bias GD/A, EOD

(*e*) Bank, Pruning, SPD

(*f*) Bank, Pruning, EOD

(*g*) Bank, Bias GD/A, SPD

(*h*) Bank, Bias GD/A, EOD

(*i*) COMPAS, Pruning, SPD

(*j*) COMPAS, Pruning, EOD

(*k*) COMPAS, Bias GD/A, SPD

(*l*) COMPAS, Bias GD/A, EOD

Figure E.1: Changes in the bias, given by the SPD (*a, c, e, g, i, k*) and EOD (*b, d, f, h, j, l*), and balanced accuracy of the neural network during pruning (*a, b, e, f, i, j*) and bias gradient descent/ascent (*c, d, g, h, k, l*). The results were obtained on Adult (*top*), Bank (*middle*), and COMPAS (*bottom*) from 20 train-test splits. **Bold** lines correspond to the median across 20 seeds. During the bias GD/A, the model was evaluated several times an epoch. Notably, both procedures reduce bias without a considerable effect on accuracy.