Group Fairness via Group Consensus

Eunice Chan[*]
ecchan2@illinois.edu
University of Illinois,
Urbana-Champaign
Champaign, Illinois, USA

Zhining Liu[*]
liu326@illinois.edu
University of Illinois,
Urbana-Champaign
Champaign, Illinois, USA

Ruizhong Qiu
rq5@illinois.edu
University of Illinois,
Urbana-Champaign
Champaign, Illinois, USA

Yuheng Zhang
yuhengz2@illinois.edu
University of Illinois,
Urbana-Champaign
Champaign, Illinois, USA

Ross Maciejewski
rmacieje@asu.edu
Arizona State University
Tempe, Arizona, USA

Hanghang Tong
htong@illinois.edu
University of Illinois,
Urbana-Champaign
Champaign, Illinois, USA

ABSTRACT

Ensuring equitable impact of machine learning models across different societal groups is of utmost importance for real-world machine learning applications. Prior research in fairness has predominantly focused on adjusting model outputs through pre-processing, in-processing, or post-processing techniques. These techniques focus on correcting bias in *either* the data *or* the model. However, we argue that the bias in the data and model should be addressed in conjunction. To achieve this, we propose an algorithm called GroupDebias to reduce unfairness in the data in a model-guided fashion, thereby enabling models to exhibit more equitable behavior. Even though it is model-aware, the core idea of GroupDebias is independent of the model architecture, making it a versatile and effective approach that can be broadly applied across various domains and model types. Our method focuses on systematically addressing biases present in the training data itself by adaptively dropping samples that increase the biases in the model. Theoretically, the proposed approach enjoys a guaranteed improvement in demographic parity at the expense of a bounded reduction in balanced accuracy. A comprehensive evaluation of the GroupDebias algorithm through extensive experiments on diverse datasets and machine learning models demonstrates that GroupDebias consistently and significantly outperforms existing fairness enhancement techniques, achieving a more substantial reduction in unfairness with minimal impact on model performance.

CCS CONCEPTS

• **Computing methodologies** → *Supervised learning*; Ensemble methods; *Artificial intelligence.*

KEYWORDS

Fairness, Machine Learning, Historical Bias, Sampling

ACM Reference Format:
Eunice Chan, Zhining Liu, Ruizhong Qiu, Yuheng Zhang, Ross Maciejewski, and Hanghang Tong. 2024. Group Fairness via Group Consensus. In *The 2024 ACM Conference on Fairness, Accountability, and Transparency (FAccT '24), June 03–06, 2024, Rio de Janeiro, Brazil.* ACM, New York, NY, USA, 21 pages. https://doi.org/10.1145/3630106.3659006

1 INTRODUCTION

Machine learning is increasingly being used in a variety of application areas with the potential for high societal impact such as filtering job applicants [6] and informing pretrial release decisions [27], studied in the context of varied domains from finance [19, 34] and social sciences [1]. With the increasing adoption of machine learning solutions in society comes the increasing concern for the fairness of such solutions. Being a broad concept, fairness has been defined in a wide variety of ways, the most popular views being (1) individual fairness [5, 14, 16, 29, 32, 36], in which similar individuals receive similar outcomes, and (2) group fairness, which aims to achieve some statistical parity among groups [11].

Ensuring group fairness is particularly important to avoid exacerbating historical injustices. Prior works primarily has predominantly focused on alleviating unfairness in the model [2, 8, 20, 24, 25] or its output [18]. Although there exist efforts to enhance fairness through data pre-processing [10, 22, 39], this aspect remains relatively under-explored in the broader research landscape. It is crucial to recognize that biases in the model output can stem from inherent biases in the data, the design of the model, or a complex interplay between the two [31]. Thus, to address biases in machine learning models, it is not sufficient to individually consider just the data or just the model. A comprehensive approach that tackles biases in both the data and the model is necessary to achieve meaningful progress in improving fairness in a model. Our contribution emphasizes the significance of this holistic perspective, aiming to bridge the gap in current research and provide a more thorough understanding of the multifaceted challenges associated with bias in machine learning models. However, none of the prior work in the field attempts to alleviate unfairness by considering the data and model *in conjunction.*

[*] Both authors contributed equally to this research.

FAccT '24, June 03–06, 2024, Rio de Janeiro, Brazil

ACM ISBN 979-8-4007-0450-5/24/06
https://doi.org/10.1145/3630106.3659006

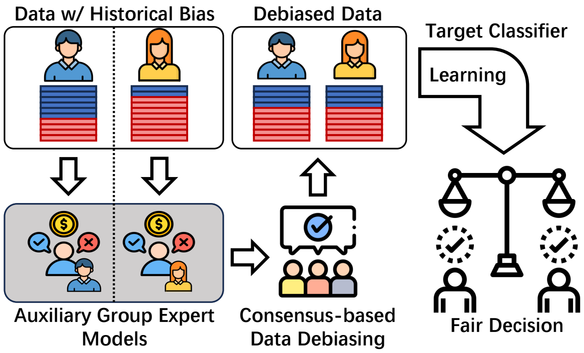


Figure 1: Model-guided, consensus-based data debiasing.

To address this problem, we introduce a model-guided data preprocessing approach to alleviating group fairness in the supervised-binary classification setting, GroupDebias. We motivate our approach with the concept of *expert elicitation*, a scientific consensus methodology used in a wide variety of fields such as social science [33] and economics [12]. As outlined in Figure 1, for each sensitive attribute group, we train an auxiliary group expert model. Then, by utilizing the disagreement among the experts, GroupDebias identifies which samples to remove. Consider a denied female application in a loan approval task. If the expert model trained only on female loan applicants denies the individual, but the expert model trained only on male loan applicants approves the individual, this disagreement suggests this sample may have been a sample that reflects the underlying historical double standard between the two groups in the data which may be amplified by downstream models. This is important as it makes our method versatile: after we leverage group expert models to debias the data, it can be used to train *any* target classifier.

Through comprehensive evaluations on the fairness benchmark datasets COMPAS [27], LSA [1], Adult [3], and Bank [34], we demonstrate that our technique outperforms existing fairness approaches in terms of reducing bias while preserving or even enhancing model performance as well as provide a theoretical analysis bounding the improvement in demographic parity and reduction in balanced accuracy as well as the fairness-utility trade-off. To better measure the trade-off between utility and fairness, we introduce two new metrics: fairness-utility relative gain (FURG) and trade-off ratio (FUTR). Compared with the state-of-the-art, on our benchmarks for logistic regression as shown in Table 3, GroupDebias obtains, on average, a 25.5% increase on the next best model on the FURG metric, and a 321% increase on the FUTR metric. Meanwhile, when it is not the best-performing model, the trade-off does not lag far behind the best trade-off. The average percent reduction from the best model's FURG and FUTR metric values are 16.5% and 4.6% respectively.

The main contributions of this paper are summarized as follows:

- **Perspective.** We propose a novel perspective to bias mitigation by performing model-guided data debiasing, focusing on the data in conjunction with the model using a consensus-based approach.
- **Algorithm.** We propose an efficient and practical model-guided algorithm, GroupDebias, to effectively improve fairness with no assumptions about the target model architecture.
- **Theory.** We provide theoretical guarantees for the efficacy of our algorithm on the improvement in fairness (demographic parity) and the bounded loss of utility (balanced accuracy). More specifically, our theory shows that the fairness-utility trade-off can be linearly controlled by the debias intensity parameter in our algorithm. Our theory is also corroborated by our empirical results.
- **Experiment.** We perform comprehensive experiments on commonly used fairness datasets utilizing a variety of target machine learning models and show that our approach has significantly better fairness-utility gain and trade-off compared to existing methods, consistently ranking in the top half among the tasks and machine learning model combinations we performed our benchmarks upon (Table 4).

For the rest of the paper, we start with a review of prior work: limitations and motivations (Section 2), then introduce the problem we aim to solve (Section 3). Next, we introduce our algorithm and our theoretical analysis (Section 4). We benchmark performance and computational efficiency (Section 5.2). Finally, we conclude with a summary of our findings (Section 6).

2 RELATED WORKS

2.1 Fair Machine Learning

Research in fair machine learning can be split into three categories: pre-processing, in-processing, and post-processing. We discuss a few here and refer the reader to other works for a more comprehensive review [4, 9, 37]. Fair pre-processing approaches transform the data so that the discrimination is removed prior to modeling [13]. This can take the form of reweighing, subsampling, or transforming the data representation to remove sensitive and adjacent attributes [10, 22]. Meanwhile, in-processing techniques modify the model training phase to ensure fairness [13]. These might be more restricted in applicability. Typically, works taking this approach augment the loss function to include some sort of fairness regularizer [2, 25, 39] or are model-specific [8, 20, 24]. There have been relatively fewer works that apply post-processing to improve fairness, which utilizes the model outputs in some way to improve fairness [13, 26, 30]. Overall, many group fairness methods have limited versatility and can only be applied to specific scenarios at a certain stage in the model training pipeline. Furthermore, group fairness techniques often have limitations as to the types of models or tasks to which they could be applied [31].

2.2 Group Experts

Prior work have argued that it is important to explicitly consider the sensitive attribute groups [14]. Although training separate classifiers by domains can be useful, doing it naively, such as with an ensemble, would result in poor performance due to limited training data for each classifier [38]. Meanwhile, some prior work leverage models primarily trained on a sensitive attribute group in the training data [15, 38]. One notable post-processing approach [18] utilizes a holdout set to create a new predictor that takes the sensitive attribute and the initial model's predictions as input. Our work shares a similar spirit in that we utilize auxiliary group expert models trained on subsets of the training data to inform our data debiasing. However, to our best knowledge, none of the prior works has proposed an approach that focused on these auxiliary models nor used it in combination with pre-processing the data.

3 PROBLEM DEFINITION

The main symbols used throughout this paper are summarized in Table 1. Throughout this paper, we use bold upper-case letters to represent matrices (e.g. $\mathbf{X}$), italic upper-case letters to represent vectors (e.g. X), and italic lower-case letters to represent elements in matrices or vectors (e.g. x). We use NumPy indexing convention for indexing of the matrices and vectors. Furthermore, we add a bar to notations to indicate deletion (e.g. $\mathcal{D}$ represents the samples of the dataset and $\bar{\mathcal{D}}$ represents the samples removed from the dataset) and a subscript to indicate it is constructed from the subset $\mathcal{D}_{S=s}$. Furthermore, the advantaged group (defined as $\arg\max_{s\in\mathcal{S}} \Pr[Y = 1|S = s]$) is represented by $s = 1$ while the disadvantaged group is represented by $s = 0$. The favorable positive outcome is represented as $y = 1$ while the negative outcome is $y = 0$. We denote subsets of the dataset as $\mathcal{D}_{\text{cond}}$ where the subscription is the set condition.

Formally, given an unfair input dataset $\mathcal{D} : (\mathbf{X}, S, Y)$, we use $\mathbf{X}[i]/Y[i]/S[i]$ to denote the feature/label/membership of the i-th sample in $\mathcal{D}$. For simplicity, we use $\mathcal{D}_{S=0}$ and $\mathcal{D}_{S=1}$ to represent the subset of $\mathcal{D}$ containing samples from group 0 and 1, respectively. To represent only positive or negative samples from group s, we use $\mathcal{D}_{S=s,Y=1}$, and $\mathcal{D}_{S=s,Y=0}$, respectively.

The key idea of our proposed method is to leverage an auxiliary model (f_0 and f_1) to identify and delete the biased samples in the training dataset, to create a debiased training dataset, which can then be used to train the target classification model (f). With the above notation, our problem can be formally defined as follows.

Problem 1 (Model-guided Training Dataset Debiasing). ***Given:*** *(1) A dataset $\mathcal{D}$ with non-sensitive attributes X, a binary sensitive attribute $s \in \mathcal{S} := \{0, 1\}$, and a binary outcome variable $y \in \mathcal{Y} := \{0, 1\}$, (2) auxiliary classifiers f_0 and f_1, (3) the debias intensity λ, (4) the target positive ratio α, and (5) the consensus drop weight ϵ;* ***Find:*** *A debiased dataset $\mathcal{D} \setminus \bar{\mathcal{D}}$ that is a subset of the original dataset where (1) the sampling probability is controlled by ϵ, f_0, and f_1, and (2) the amount of samples to remove is controlled by λ and α.*

4 DATA DEBIASING VIA GROUP CONSENSUS

In this section, we present the proposed GroupDebias method (Section 4.1) and relevant theoretical analyses (Section 4.2). Our algorithm systematically addresses biases present in the training data by adaptively identifying and discarding samples that carry historical bias. Specifically, drawing upon the idea of expert elicitation, we build expert models for each demographic group and then leverage the consensus among group-specific experts to locate biased samples. Figure 2 shows an illustrative example of the GroupDebias workflow. We further provide theoretical guarantees on the fairness-utility trade-off of GroupDebias, showing that it can achieve improved fairness with a small, bounded utility cost.

4.1 The GroupDebias Algorithm

We now formally describe the GroupDebias algorithm, summarized in Algorithm 1. The core step of our algorithm is to estimate the bias of each training sample by building and consulting the auxiliary group expert models. Subsequently, samples with high biases will be discarded to achieve data debiasing.

Given the learning algorithm, we first use $\mathcal{D}_{S=0}$ and $\mathcal{D}_{S=1}$ to train the group expert models, namely f_0 and f_1 (Steps 1-3). Having the group experts, we derive the consensus vector $C \in \mathbb{R}^{|\mathcal{D}|}$ as follows (Step 4):

Definition 1 (Consensus Vector). *Given a dataset $\mathcal{D}$, the consensus vector $C \in \mathbb{R}^{|\mathcal{D}|}$ describes the prediction consistency between group expert models $f_0(\cdot)$ and $f_1(\cdot)$, where the i-th element corresponding to the i-th sample is*

$$C[i] = \begin{cases} 1 & \text{if } f_0(\mathbf{X}[i]) = f_1(\mathbf{X}[i]), \\ 0 & \text{otherwise.} \end{cases}$$

We then debias the dataset $\mathcal{D}$ by discarding samples without consensus. The number of samples to remove (i.e., deletion budget) for each sensitive attribute group is controlled by two user-specified parameters, including *debias intensity* $\lambda \in [0, 1]$, and *target positive ratio* $\alpha \in [0, 1]$ (Step 7). Formally, the deletion budget of sensitive attribute group $s \in \mathcal{S}$ is calculated as $\bar{n}_s :=$

$$\begin{cases} \max\Big\{0,\ \Big\lfloor \lambda\Big(|\mathcal{D}_{S=s,Y=0}| - \frac{1-\alpha}{\alpha}|\mathcal{D}_{S=s,Y=1}|\Big)\Big\rfloor\Big\} & \text{if } \frac{|\mathcal{D}_{S=s,Y=1}|}{n_s} < \alpha, \\ \max\Big\{0,\ \Big\lfloor \lambda\Big(|\mathcal{D}_{S=s,Y=1}| - \frac{\alpha}{1-\alpha}|\mathcal{D}_{S=s,Y=0}|\Big)\Big\rfloor\Big\} & \text{otherwise.} \end{cases} \tag{1}$$

Eq. (1) ensures that the sub-sampling does not reduce the percentage of positive samples less than the smallest percentage between the sensitive attribute groups. As a result, the representativeness of the dataset is not drastically changed. Intuitively, in Eq. (1), the *target proportion* α specifies the expected ratio of samples that receive favorable outcomes (i.e., positive labels $Y = 1$), while the *debias intensity* $\lambda \in [0, 1]$ indicates how many deletions are allowed in each group to match the positive ratio α. Setting $\lambda = 1$ would make the positive sample ratio exactly α for all sensitive groups, while $\lambda = 0$ implies no allowance for deleting any samples. We note that λ can be viewed as a trade-off parameter between fairness and utility: a larger λ favors better fairness of the data by deleting more biased training samples, but at the expense of potentially more utility loss in terms of predictive performance degradation. In practice, users can adjust λ and α based on the application scenario to achieve optimal data debiasing with GroupDebias.

Next, we discuss how to select the candidate samples for deletion for each group. As described before, our objective is to make the positive ratio for each group approach the user-defined target positive ratio α. The philosophy here is the likelihood of a positive outcome should be the same regardless of whether the person is in the protected (e.g., female) group. To achieve this goal with minimal data removal, for a specific group $s \in \mathcal{S}$, we select its deletion candidates based on its actual positive ratio α_s and the given target positive ratio α. Specifically (Step 8), the deletion candidate set $\mathcal{D}_s^{\text{cand}}$ of sensitive group s is determined by

$$\mathcal{D}_s^{\text{cand}} = \begin{cases} \mathcal{D}_{S=s,Y=0} & \text{if } \frac{|\mathcal{D}_{S=s,Y=1}|}{n_s} < \alpha, \\ \mathcal{D}_{S=s,Y=1} & \text{otherwise.} \end{cases} \tag{2}$$

Table 1: Table of symbols.

Symbol	Definition
$\mathbf{X}$	The non-sensitive attribute matrix.
S	The sensitive attribute group vector.
Y	The classification label vector.
$\mathcal{S}$	The range of values an element $s \in S$ can take.
$\mathcal{Y}$	The range of values an element $y \in Y$ can take.
$\mathcal{D}$	The set of samples that make up the training dataset. Each sample consists of (X, s, y).
$\bar{\mathcal{D}}$	Set of samples that are removed from the training dataset.
f	Target model.
f_s	Auxiliary group expert model trained on $\mathcal{D}_{S=s}$.
n	The total number of samples in $\mathcal{D}$.
n_s	The number of samples in $\mathcal{D}_{S=s}$.
$\bar{n}_s$	The number of deleted samples from $\mathcal{D}_{S=s}$.
C	The binary consensus vector.
W_s	The weight vector that controls the probability of a sample being selected.
Δ	The demographic parity difference. $\Pr[Y=1\|S=1] - \Pr[Y=1\|S=0] > 0$.
π_s	The proportion of positive labels in $\mathcal{D}_{S=s}$. $\Pr[Y=1 \mid S=s]$.
λ	The debias intensity parameter.
α	The target positive ratio parameter.
α_s	The positive label ratio of $\mathcal{D}_{S=s}$.
ϵ	The consensus drop weight parameter.

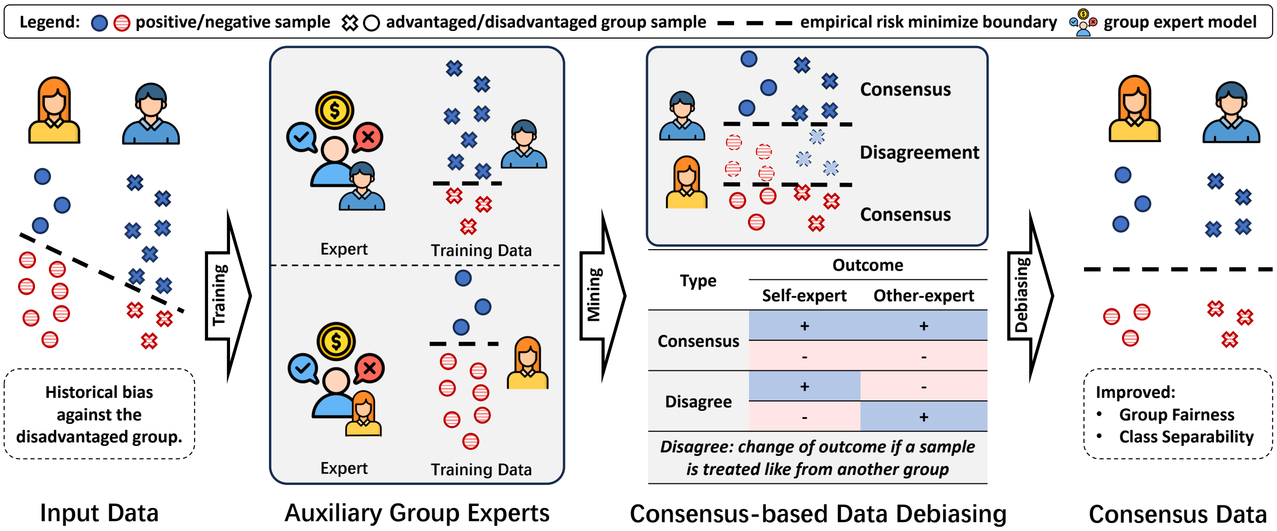

Legend: ● ⊖ positive/negative sample ✖ ○ advantaged/disadvantaged group sample — — — empirical risk minimize boundary group expert model
Historical bias against the disadvantaged group.
Training
Expert
Training Data
Expert
Training Data
Mining
Consensus
Disagreement
Consensus
Type
Outcome
Self-expert
Other-expert
Consensus
+
+
-
-
Disagree
+
-
-
+
Disagree: change of outcome if a sample is treated like from another group
Debiasing
Improved:
Group Fairness
Class Separability
Input Data
Auxiliary Group Experts
Consensus-based Data Debiasing
Consensus Data

Figure 2: An illustrative overview of the proposed consensus-based data debiasing framework. Group-specific experts is trained on sensitive attribute group subsets of the data, and the consensus of the experts on each sample is leveraged to perform a weighted subsampling of the dataset.

In order to take a weighted sample $\bar{\mathcal{D}}_s$ (Step 10), we calculate the weight vector $W_s := [W_s[1], \cdots, W_s[(|\mathcal{D}|)]]$ as follows (Step 9), where $0 \leq \epsilon < 1$ is the consensus drop weight.

$$W_s[i] = \begin{cases} 1 & \text{if } (\mathbf{X}[i], S[i], Y[i]) \in \mathcal{D}_s^{\text{cand}}, C[i] = 0, \\ \epsilon & \text{if } (\mathbf{X}[i], S[i], Y[i]) \in \mathcal{D}_s^{\text{cand}}, C[i] = 1, \\ 0 & \text{otherwise.} \end{cases} \quad (3)$$

Ensemble Variation. We further present an ensemble variation of GroupDebias. The motivation for this variation is that the base GroupDebias algorithm does not make use of the full dataset (i.e., the target model never sees the removed samples $\bar{\mathcal{D}}$). The ensemble variation allows the full dataset to be used in training. To be specific, the target model does not get to leverage the whole training set due to subsampling. This results in a greater variance in performance and potential information loss depending on the random seed used. However, this can be alleviated by training multiple copies of the model on different variations of the dataset. Some of the samples without consensus will be kept at random, so each dataset is likely to keep a different subset of the samples without consensus. Therefore, the ensemble as a whole will see more of the training set than each individual model that makes up the ensemble.

Algorithm 1 The GroupDebias Algorithm

```
Input: Dataset $\mathcal{D}$, Auxiliary Group Expert Classifiers $f_0$ and $f_1$,
    Debias Intensity $\lambda$, Target Positive Ratio $\alpha$, Consensus Drop
    Weight $\epsilon$
for each sensitive group $s \in \mathcal{S}$ do
    Train group expert $f_s(\cdot)$ on $\mathcal{D}_{S=s}$
end for
Create consensus vector $C$ (Def. 1)
Initialize $\bar{\mathcal{D}} \leftarrow \emptyset$
for each sensitive group $s \in \mathcal{S}$ do
    Calculate deletion budget $\bar{n}_s$ (Eq. (1))
    Construct set of deletion candidates $\mathcal{D}_s^{\text{cand}}$ (Eq. (2))
    Create weight vector $W_s$ (Eq. (3))
    Take a weighted sample $\bar{\mathcal{D}}_s$ from $\mathcal{D}$ using weights $W_s$ s.t.
    $|\bar{\mathcal{D}}_s| = \bar{n}_s$
    Update $\bar{\mathcal{D}} \leftarrow \bar{\mathcal{D}} \bigcup \bar{\mathcal{D}}_s$
end for
return Debiased dataset $\mathcal{D} \setminus \bar{\mathcal{D}}$
```

4.2 Theoretical Analysis

We provide a theoretical analysis of the GroupDebias algorithm with respect to the DP (demographic parity) fairness metric. We show that there exists a classifier trained with the GroupDebias algorithm that results in improved fairness and a bounded loss of utility.

To be specific, we establish the following theoretical guarantees for GroupDebias in reducing the demographic parity difference between the sensitive attribute groups:

Theorem 1. *Under the assumptions listed in Appendix C.2, if* $n_0 > \frac{\pi_1}{(1-\lambda\pi_1)\Delta}$ *(i.e., the dataset size of the minority group is not too small), then for any given* $\epsilon > 0$*, with probability at least* $1 - e^{-\Omega(n)}$*, under the target ratio* $\alpha := \frac{|\mathcal{D}_{S=1,Y=1}|}{n_1}$ *(i.e., the positive rate of the majority group), there exists a classifier* $\widehat{f} : \mathcal{X} \times \mathcal{S} \to \mathcal{Y}$ *that minimizes the classification error over the debiased training set* $\mathcal{D} \setminus \bar{\mathcal{D}}$ *and achieves both improved fairness and bounded loss of utility:*

- *Improved fairness (demographic parity):*

$$\left|\Pr[\widehat{f}(\mathbf{X}, S) = 1 \,|\, S = 0] - \Pr[\widehat{f}(\mathbf{X}, S) = 1 \,|\, S = 1]\right| \leq (1-\lambda)\Delta + O\Big(\frac{1}{n}\Big); \tag{4}$$

- *Bounded loss of utility (balanced error rate):*

$$\Pr[\widehat{f}(\mathbf{X}, S) \neq Y \mid S = 0] + \Pr[\widehat{f}(\mathbf{X}, S) \neq Y \mid S = 1] \leq \lambda\Delta + O\Big(\frac{1}{n}\Big). \tag{5}$$

Theorem 1 (full proof in Appendix C.3) gives a probabilistic guarantee to improvement in fairness and the bounded reduction in utility. In Eq. (4), the difference in the proportion of positive $\hat{Y}$ values in each sensitive attribute group is bounded by $(1 - \lambda)\Delta + O\big(\frac{1}{n}\big)$. This improvement in fairness is dependent on the dataset size n, the debias intensity λ, and the original demographic parity difference Δ. With a high debias intensity and large dataset, the demographic parity difference for $\hat{f}$ can get very close to 0, which means a well-learned $\hat{f}$ is a very fair classifier. Meanwhile, in Eq. (5), the balanced error rate is bounded by $\lambda\Delta + O\big(\frac{1}{n}\big)$. This is a similar bound to the bound on fairness in Eq. (4). However, the difference is that we have a $\lambda\Delta$ term rather than a $(1 - \lambda)\Delta$ term. This means the smaller the debias intensity, the smaller the upper bound of the error rate becomes. The implication of these two probabilistic guarantees shows the λ controls for the the fairness-utility trade-off (which we also corroborate empirically in Section 5.2).

Although our algorithm is primarily focused on improving the DP, we also benchmark performance on another fairness metric, EO (equalized odds).

Proposition 1 (Relation between demographic parity and equalized odds). *Let*

$$\mathrm{DP} := \Pr[\widehat{Y} = 1 \mid S = 1] - \Pr[\widehat{Y} = 1 \mid S = 0],$$

$$\mathrm{EO}_{Y=y} := \Pr[\widehat{Y} = 1 \mid Y = y, S = 1] - \Pr[\widehat{Y} = 1 \mid Y = y, S = 0],$$

$$\mathrm{Util}_{S=s} := \Pr[\widehat{Y} = 0 \mid Y = 0, S = s] + \Pr[\widehat{Y} = 1 \mid Y = 1, S = s]$$

denote the demographic parity difference of the prediction $\widehat{Y}$*, equalized odds difference w.r.t. class* y*, and the utility of group* s*, respectively. Then,*

$$\mathrm{DP} = \Delta \cdot (\mathrm{Util}_{S=1} - 1) + \pi_0 \cdot \mathrm{EO}_{Y=1} + (1 - \pi_0) \cdot \mathrm{EO}_{Y=0}.$$

Note: In Theorem 1, utility is defined as the error rate weighing both groups equally while here we define the per-group utility as the accuracy weighing both outcomes (Y=1, Y=0) equally.

Proposition 1 (full proof at Appendix C.4) shows that DP and EO are correlated. Thus, our algorithm is expected to improve EO implicitly although it is primarily designed to improve DP.

5 EXPERIMENTS

5.1 Setting

Datasets. We run empirical evaluations on four datasets comparing against two sensitive attributes each, totalling to eight unique tasks. Table 2 gives the dataset statistics for the tasks we benchmark upon. Further details of the setup for reproducibility are in Appendix A. Code can be found at https://anonymous.4open.science/r/GroupDebias.

Baselines. We use six target models f, including (1) logistic regression, (2) k-neighbors classifier, (3) decision tree classifier, (4) a multi-layer perceptron, (5) AdaBoost, and (6) bagging. For the fair baselines, we select four different algorithms, including (1) one pre-processing baseline, Reweight [22], a model-agnostic reweighing scheme, (2) one in-processing baseline, Reduction (EO/DP) [2], which performs gradient reductions, (3) one post-processing baseline, Threshold (EO/DP) [18], which adjusts the model based on S, Y, and $\hat{Y}$, and (4) one ensemble baseline, AdaFair [20], which uses AdaBoost to reduce unfairness.

Table 2: Statistics of the task settings utilized to benchmark our method. We utilize four different datasets and compare against two sensitive attributes per dataset for a total of eight tasks.

Dataset	Sensitive Attribute	# Features	# Samples			% Y=1	
			Total	S=1	S=0	S=1	S=0
COMPAS [27]	sex	11	5875	4714	1161	49.15%	36.18%
	race			3528	2347	51.33%	39.45%
LSA [1]	gender	9	49900	26183	23717	14.93%	25.41%
	race			37545	12355	23.17%	9.99%
Adult [3]	gender	98	45222	30527	14695	31.25%	11.36%
	race			38903	6319	26.24%	15.84%
Bank [34]	age	57	30488	29624	864	12.35%	23.03%
	marital status			17492	12996	11.75%	13.87%

We compare these results with the *Dummy* model which randomly predicts classes with a probability proportional to the class distribution in the training data and the *Vanilla* model which is the target model trained on the original training dataset without modifications.

Metrics. We use six evaluation metrics: two utility metrics, two fairness metrics, and two trade-off metrics. To quantify utility, we use both the accuracy (Acc.) and, due to some datasets being significantly imbalanced, the balanced accuracy (BAcc.) [7]. To quantify fairness, we utilize both the demographic party (DP) metric [23] and the equalized odds (EO) metric [17]. Because there are no existing metrics for quantifying the trade-off between utility and fairness, we propose two new metrics, Fairness-Utility Relative Gain (FURG) and Fairness-Utility Trade-off Ratio (FUTR), to provide a more comprehensive and intuitive assessment for the fair learning algorithms. In our proposed trade-off metrics, we consider the trade-offs made in UG (utility gain) and unfairness drop (UD).

Formally, we define utility gain (UG) of f_{fair} over f as

$$\text{UG}(f, f_{\text{dummy}}, f_{\text{fair}}) := \frac{m(f_{\text{fair}}) - m(f)}{m(f) - m(f_{\text{dummy}})} \tag{6}$$

where f is the target model, f_{dummy} is the dummy model, and f_{fair} is the target model with a fairness method applied to it. The utility metric function m (e.g., Acc., BAcc) is defined so that higher values corresponds to better model performance. A higher UG means that f_{fair} bears less utility loss (or gains more utility) over f.

A dummy model (f_{dummy}) typically carries no bias as it gives random predictions, thus any model with utility worse than f_{dummy} is strictly worse than random prediction in both utility and fairness. Therefore, we use f_{dummy} as the baseline in Eq. (6) for computing utility drop (instead of 0) to highlight the relative performance drop of fair learning algorithms in a more practical sense.

Meanwhile, unfairness drop (UD) is defined as

$$\text{UD}(f, f_{\text{fair}}) := \frac{1}{|\mathcal{M}|} \sum_{m \in \mathcal{M}} \frac{m(f) - m(f_{\text{fair}})}{m(f)} \tag{7}$$

where we share the same notation f and f_{fair} as above. This metric takes in set of fairness metrics $\mathcal{M}$ where each fairness metric assigns lower values to models with better fairness. A higher UD means that f_{fair} obtains more fairness improvement over f.

Ideally, the best fair learning method should maximize the increase in utility (or, equivalently, minimize the reduction in utility) while also maximize the reduction in unfairness. However, there is often a tension between learning utility and fairness. In order to quantify the fairness-utility trade-off, we further introduce two combined metrics as follows. The first is the Fairness-Utility Relative Gain (FURG) metric, which is defined as the sum of the UG (utility gain) and UD (unfairness drop). The intuition of FURG is to measure the total combined gain in utility and fairness, with the equal importance to both utility and fairness. Another choice is the Fairness-Utility Trade-off Ratio (FUTR) metric, which is defined as the negative ratio between UD (unfairness drop) and UG (utility gain). The intuition of FUTR is to consider the return on investment (ROI), i.e., the unfairness reduction per unit learning utility loss. To avoid 'divide-by-zero' issue in FUTR, we replace UG by $\min(\text{UG}, -0.01)$ in experiments. The intuition is that we assume a minimum utility loss of 0.01 for f_{fair}. Given the tension between the utility and fairness, this is a reasonable treatment since the improvement of the model fairness is often at the expense of the learning utility loss to some degree. For both FURG and FUTR, the larger the metric value, the better the fairness-utility trade-off.

5.2 Results and Discussion

5.2.1 Main Results. Table 3 shows the effectiveness on each dataset using the logistic regression target model. This table shows that with a logistic regression as the target model, for each dataset, our approach has the best trade-off as measured by at least one of the trade-off metrics we use: FURG, or FUTR. In the cases where our approach does not have the best trade-off value, it is close to the best trade-off. Furthermore, the Fairness column of Table 3 provides empirical results which is consistent with Proposition 1 in that a change in fairness according to one fairness metric tends to mean a change in fairness in the same direction according to the other fairness metric.

Table 4 presents the result of the average ranking of each model on the full array of settings (two sensitive attributes for four datasets; to see the results by task, see Appendix D). In this table, each method is ranked with 1 being the best score, 2 being the second best, and so on. This is done for 8 tasks: the 4 datasets compared on 2 sensitive attributes using 6 different target models for a total of 48 different settings. Our method consistently ranks highly compared to the other approaches.

Table 3: Comparison of our method with other FairML techniques (80-20 split, use logistic regression as target classifier). Ours is a single target classifier trained with GroupDebias, no ensemble. Δ represents the utility gain (UG) or unfairness drop (UD) for the utility and fairness metrics respectively. Explanations for select FURG and FUTR values in Appendix D.1. We bold the best FURG/FUTR value for each task group, and underline the second best value.

Task	Method	Utility				Fairness				Combined	
		Acc.	Δ	BAcc.	Δ	DP	Δ	EO	Δ	FURG	FUTR
COMPAS (sex)	Dummy	49.69±2.1	-	49.49±2.2	-	0.97±0.6	-	4.40±4.3	-	-	-
	Vanilla	69.60±0.6	-	69.00±0.6	-	33.62±1.6	-	38.23±3.9	-	-	-
	Reweight	68.89±0.9	-3.59%	67.68±0.9	-6.74%	27.51±2.2	-18.18%	34.27±5.5	-10.36%	9.10	2.76
	Reduction$_{DP}$	68.92±1.0	-3.42%	68.59±1.0	-2.08%	10.48±4.1	-68.82%	8.57±4.3	-77.60%	70.46	26.62
	Reduction$_{EO}$	68.73±0.9	-4.36%	68.23±0.9	-3.93%	12.85±2.8	-61.80%	10.94±3.3	-71.38%	62.44	16.06
	Threshold$_{DP}$	66.98±0.8	-13.16%	66.83±0.8	-11.10%	3.15±1.0	-90.64%	14.56±5.1	-61.92%	64.15	6.29
	Threshold$_{EO}$	62.04±1.0	-37.95%	60.26±1.0	-44.81%	3.22±2.6	-90.42%	6.36±3.9	-83.37%	45.51	2.10
	AdaFair	59.54±3.6	-50.51%	58.10±4.7	-55.87%	19.43±12.4	-42.20%	25.65±16.1	-32.92%	-15.63	0.71
	Ours	69.04±1.2	-2.82%	68.86±1.2	-0.71%	2.71±2.6	-91.94%	7.73±2.3	-79.78%	**84.09**	**48.62**
Adult (sex)	Dummy	62.36±0.4	-	49.59±0.4	-	0.72±0.4	-	2.41±1.4	-	-	-
	Vanilla	84.68±0.2	-	76.51±0.3	-	18.92±1.1	-	12.95±3.6	-	-	-
	Reweight	81.28±0.3	-15.24%	63.68±0.8	-47.66%	8.12±0.5	-57.10%	9.26±1.5	-28.51%	11.35	1.36
	Reduction$_{DP}$	82.01±0.4	-11.98%	69.70±0.8	-25.30%	1.49±0.6	-92.15%	32.20±3.7	+148.64%	-46.88	-1.52
	Reduction$_{EO}$	84.63±0.2	-0.23%	76.34±0.3	-0.64%	18.21±1.2	-3.75%	10.64±4.0	-17.81%	10.35	10.78
	Threshold$_{DP}$	79.37±0.2	-23.79%	73.45±0.3	-11.36%	0.72±0.8	-96.18%	11.67±0.8	-9.90%	35.47	3.02
	Threshold$_{EO}$	81.84±0.3	-12.74%	72.35±0.6	-15.44%	8.99±1.1	-52.47%	3.34±1.8	-74.18%	**49.23**	4.49
	AdaFair	75.21±0.0	-42.41%	50.00±0.0	-98.47%	0.00±0.0	-100.00%	0.00±0.0	-100.00%	29.56	1.42
	Ours	84.39±0.3	-1.29%	77.01±0.4	+1.86%	14.57±1.4	-23.00%	5.35±1.0	-58.68%	41.12	**40.84**
LSA (race)	Dummy	67.88±0.4	-	49.78±0.4	-	0.39±0.3	-	2.28±1.1	-	-	-
	Vanilla	83.51±0.2	-	63.63±0.3	-	10.01±0.3	-	24.78±1.2	-	-	-
	Reweight	80.59±0.1	-18.69%	51.30±0.1	-89.06%	0.67±0.0	-93.35%	2.63±0.3	-89.39%	37.49	1.70
	Reduction$_{DP}$	83.25±0.4	-1.68%	62.30±1.3	-9.62%	4.38±5.0	-56.27%	13.80±10.1	-44.33%	44.65	8.90
	Reduction$_{EO}$	82.91±0.7	-3.85%	61.11±2.4	-18.22%	4.82±4.7	-51.89%	17.42±6.9	-29.71%	29.77	3.70
	Threshold$_{DP}$	81.30±0.2	-14.12%	62.55±0.3	-7.84%	0.48±0.3	-95.19%	17.62±1.8	-28.89%	51.06	5.65
	Threshold$_{EO}$	80.94±0.1	-16.44%	54.12±0.3	-68.69%	0.84±0.6	-91.63%	2.24±1.0	-90.97%	48.74	2.15
	AdaFair	80.09±0.0	-21.87%	50.00±0.0	-98.42%	0.00±0.0	-100.00%	0.00±0.0	-100.00%	39.85	1.66
	Ours	83.25±0.2	-1.64%	64.28±0.3	+4.63%	4.14±0.7	-58.61%	3.04±1.7	-87.72%	**74.66**	**73.16**
Bank (age)	Dummy	77.68±0.3	-	49.90±0.6	-	1.59±1.6	-	5.22±3.7	-	-	-
	Vanilla	89.66±0.3	-	67.81±0.5	-	10.31±3.3	-	14.38±10.6	-	-	-
	Reweight	87.51±0.1	-17.99%	50.98±0.1	-93.96%	0.57±0.3	-94.46%	1.74±1.3	-87.90%	35.21	1.63
	Reduction$_{DP}$	89.57±0.2	-0.74%	66.92±0.5	-4.98%	0.91±0.9	-91.20%	12.74±6.3	-11.42%	48.45	17.93
	Reduction$_{EO}$	89.71±0.3	+0.38%	67.68±0.3	-0.72%	3.91±2.0	-62.09%	7.09±6.8	-50.72%	56.24	**56.41**
	Threshold$_{DP}$	89.52±0.2	-1.18%	66.98±0.4	-4.66%	1.79±0.9	-82.67%	16.65±6.3	+15.76%	30.54	11.47
	Threshold$_{EO}$	87.83±0.3	-15.33%	67.19±0.3	-3.46%	5.08±2.9	-50.72%	10.89±3.4	-24.32%	28.12	3.99
	AdaFair	87.40±1.6	-18.87%	60.24±5.9	-42.29%	8.84±8.2	-14.28%	10.62±7.3	-26.16%	-10.36	0.66
	Ours	89.66±0.4	-0.05%	72.81±1.0	+27.93%	5.47±2.3	-46.97%	5.65±4.1	-60.69%	**67.77**	53.83

Table 5 provides a comparison of our approach against the other fairness methods as well as the target model baselines for all combination of tasks and machine learning models we have benchmarked on with five seeds per run (to see the results by task, see Appendix D). Compared to the other methods and even the target model, on average, our approach has better utility and fairness according to the EO and balanced accuracy metrics.

Table 4: Average rank of different FairML techniques on all 8 tasks (4 datasets x 2 sensitive attributes) with 6 target machine learning models. We further introduce Ours$_{ens}$, an ensemble of multiple target classifiers trained with independently sampled subsets, to validate its ability in reducing the variance brought about by random sampling process. The missing cells represent invalid combinations.

Method	**LR**		**KNN**		**MLP**		**DT**		**ADA**		**BAG**	
	FURG	FUTR	FURG	FUTR	FURG	FUTR	FURG	FUTR	FURG	FUTR	FURG	FUTR
Reweight	6.38$_{\pm 1.22}$	6.75$_{\pm 0.83}$	-	-	-	-	5.38$_{\pm 2.29}$	4.62$_{\pm 2.23}$	5.25$_{\pm 1.92}$	4.25$_{\pm 2.05}$	6.12$_{\pm 2.03}$	4.88$_{\pm 2.71}$
Reduction$_{DP}$	4.38$_{\pm 2.18}$	4.88$_{\pm 1.76}$	-	-	-	-	4.62$_{\pm 2.23}$	3.88$_{\pm 2.37}$	3.75$_{\pm 2.28}$	$\underline{3.75}_{\pm 2.33}$	4.25$_{\pm 1.48}$	3.75$_{\pm 1.09}$
Reduction$_{EO}$	5.25$_{\pm 2.17}$	4.00$_{\pm 1.94}$	-	-	-	-	5.38$_{\pm 2.39}$	5.38$_{\pm 2.74}$	5.38$_{\pm 1.87}$	5.12$_{\pm 2.47}$	6.00$_{\pm 1.94}$	5.38$_{\pm 2.12}$
Threshold$_{DP}$	4.38$_{\pm 1.11}$	4.62$_{\pm 0.86}$	2.62$_{\pm 0.99}$	3.50$_{\pm 0.50}$	3.25$_{\pm 0.83}$	3.38$_{\pm 0.48}$	4.00$_{\pm 1.32}$	4.50$_{\pm 1.12}$	4.00$_{\pm 2.12}$	4.88$_{\pm 1.54}$	4.00$_{\pm 2.24}$	5.88$_{\pm 1.17}$
Threshold$_{EO}$	3.38$_{\pm 2.06}$	5.12$_{\pm 1.27}$	$\underline{2.50}_{\pm 1.32}$	3.50$_{\pm 0.50}$	2.50$_{\pm 1.22}$	3.62$_{\pm 0.48}$	5.88$_{\pm 1.36}$	5.50$_{\pm 1.87}$	5.62$_{\pm 1.11}$	5.25$_{\pm 1.71}$	4.88$_{\pm 1.45}$	4.62$_{\pm 1.73}$
AdaFair	7.25$_{\pm 1.09}$	7.38$_{\pm 0.86}$	-	-	-	-	5.38$_{\pm 2.74}$	6.50$_{\pm 1.41}$	6.12$_{\pm 2.98}$	6.75$_{\pm 1.92}$	6.38$_{\pm 1.58}$	6.88$_{\pm 1.54}$
Ours	**2.38**$_{\pm 1.32}$	**1.38**$_{\pm 0.48}$	**2.38**$_{\pm 0.86}$	$\underline{1.75}_{\pm 0.43}$	**2.00**$_{\pm 1.00}$	**1.38**$_{\pm 0.48}$	$\underline{2.88}_{\pm 1.62}$	$\underline{2.88}_{\pm 1.17}$	$\underline{3.50}_{\pm 1.66}$	$\underline{3.75}_{\pm 1.39}$	$\underline{2.38}_{\pm 0.70}$	$\underline{2.88}_{\pm 1.36}$
Ours$_{ens}$	$\underline{2.62}_{\pm 1.32}$	$\underline{1.88}_{\pm 0.60}$	$\underline{2.50}_{\pm 1.22}$	**1.25**$_{\pm 0.43}$	$\underline{2.25}_{\pm 0.97}$	$\underline{1.62}_{\pm 0.48}$	**2.50**$_{\pm 1.12}$	**2.75**$_{\pm 1.98}$	**2.38**$_{\pm 0.86}$	**2.25**$_{\pm 1.71}$	**2.00**$_{\pm 1.50}$	**1.75**$_{\pm 1.09}$

Table 5: The difference between our approach and other fairness methods as well as the target model baselines for all combination of tasks and machine learning models we have benchmarked on. Our approach has a minimal impact on the learning utility with either a minor decrease in Acc. or even increase in BAcc., and meanwhile it consistently reduces both DP and EO.

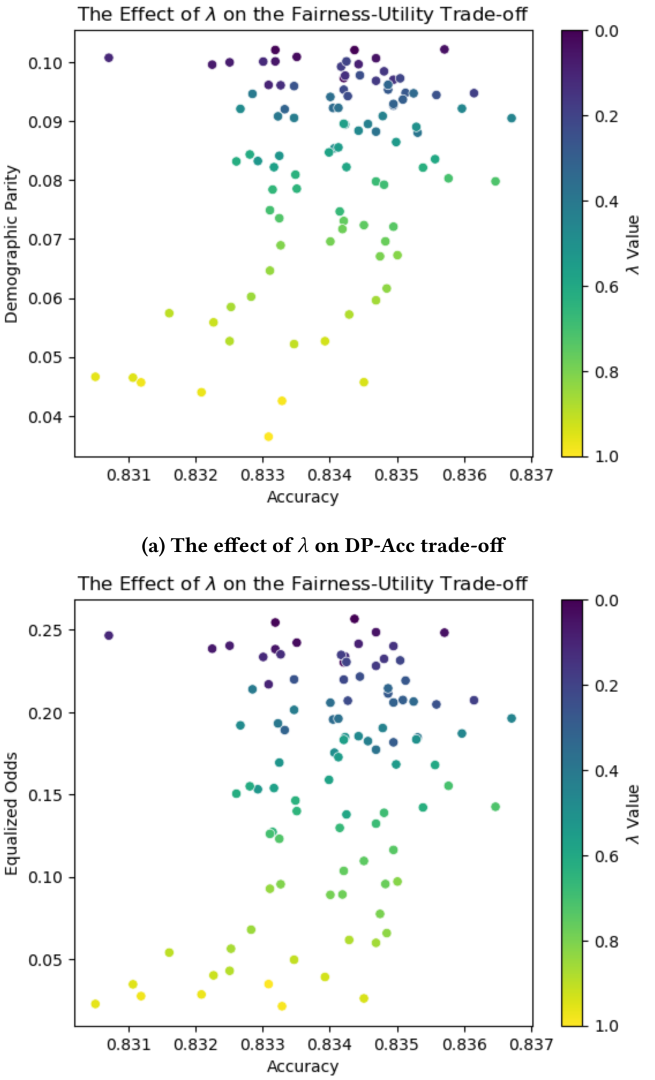
		Utility		**Fairness**	
		Acc.	**BAcc.**	**DP**	**EO**
Ours	**Vanilla**	-0.009$_{\pm 0.01}$	0.004$_{\pm 0.02}$	-0.066$_{\pm 0.07}$	-0.077$_{\pm 0.09}$
	Reweight	-0.005$_{\pm 0.02}$	0.033$_{\pm 0.06}$	-0.036$_{\pm 0.07}$	-0.037$_{\pm 0.08}$
	Reduction$_{DP}$	-0.005$_{\pm 0.02}$	0.01$_{\pm 0.02}$	-0.013$_{\pm 0.05}$	-0.023$_{\pm 0.07}$
	Reduction$_{EO}$	-0.007$_{\pm 0.02}$	0.004$_{\pm 0.02}$	-0.035$_{\pm 0.05}$	-0.033$_{\pm 0.06}$
	Threshold$_{DP}$	0.008$_{\pm 0.02}$	0.024$_{\pm 0.02}$	0.044$_{\pm 0.05}$	-0.04$_{\pm 0.06}$
	Threshold$_{EO}$	0.005$_{\pm 0.02}$	0.027$_{\pm 0.03}$	-0.003$_{\pm 0.04}$	-0.004$_{\pm 0.06}$
	AdaFair	0.005$_{\pm 0.04}$	0.058$_{\pm 0.1}$	-0.037$_{\pm 0.08}$	-0.037$_{\pm 0.08}$
Ours$_{ens}$	**Vanilla**	-0.005$_{\pm 0.01}$	0.008$_{\pm 0.02}$	-0.066$_{\pm 0.08}$	-0.08$_{\pm 0.09}$
	Reweight	0.0$_{\pm 0.02}$	0.039$_{\pm 0.06}$	-0.036$_{\pm 0.07}$	-0.038$_{\pm 0.08}$
	Reduction$_{DP}$	0.0$_{\pm 0.01}$	0.016$_{\pm 0.02}$	-0.012$_{\pm 0.05}$	-0.025$_{\pm 0.07}$
	Reduction$_{EO}$	-0.002$_{\pm 0.02}$	0.01$_{\pm 0.02}$	-0.034$_{\pm 0.05}$	-0.034$_{\pm 0.06}$
	Threshold$_{DP}$	0.012$_{\pm 0.02}$	0.029$_{\pm 0.02}$	0.044$_{\pm 0.05}$	-0.044$_{\pm 0.06}$
	Threshold$_{EO}$	0.009$_{\pm 0.02}$	0.031$_{\pm 0.03}$	-0.003$_{\pm 0.04}$	-0.008$_{\pm 0.05}$
	AdaFair	0.01$_{\pm 0.04}$	0.064$_{\pm 0.1}$	-0.037$_{\pm 0.08}$	-0.038$_{\pm 0.08}$

Overall, as shown in Table 5, for all combination of tasks and machine learning models we have benchmarked on, our approach's change in accuracy is about the middle of the average accuracy change of the fair ML methods we compare against. Furthermore, our approach has, on average, a smaller DP value than all the other approaches except for Threshold$_{DP}$. In terms of fairness-utility trade-off, our method consistently outperforms the rest in terms of the fairness-utility trade-off (Table 3). Table 4 shows that our method has consistently strong performance on all the datasets across diverse model architectures with our approach.

We also show the effectiveness of the debias intensity hyperparameter λ to control the fairness-utility trade-off. The λ controls the amount of subsampling done. The larger the parameter value, the more samples are removed. In Figure 3, the varying debias intensities are plotted for our approach with a logisitic regression target model with the Law School Admissions (race) task. As shown in the figure, the larger the λ, the better (lower) the fairness metric (for both DP and EO) and the worse (lower) the utility metric (accuracy). For a large range of λ, the utility stays in the range of $[0.830, 0.840]$ (x-axis of Figure 3), indicating that our proposed method has a minor impact on the learning utility.

As mentioned earlier, the debiased dataset by our method can be used to train any target model, allowing it to be applied to a wide range of machine learning models. As we can observe from Table 4, our approach has consistently strong performance on a variety of target models. Only one other approach: Threshold$_{DP}$/Threshold$_{EO}$ supports the full range of models our approach can apply to. The other methods we compare against cannot be directly applied to all the models.

5.2.2 Ensemble Variation. Table 4 shows the ensemble variant of our approach further improves performance compared to the single model trained on a single subsampled dataset. Furthermore, Table 5 compares the ensemble version of GroupDebias against the other methods and the target model. It shows that, on average, our approach has better utility and fairness according to the EO and balanced accuracy metrics. For the accuracy metric, the ensemble variation is on the higher end, matching Reweight and Reduction$_{DP}$



(a) The effect of λ on DP-Acc trade-off

(b) The effect of λ on EO-Acc trade-off

Figure 3: The trade-off between fairness (demographic parity and equalized odds) and utility (accuracy) on the Law School Admissions (race) dataset with the logistic regression target model.

while being slightly worse than Reduction$_{EO}$. On the DP fairness metric, performance is about the same as the single model variation. Overall, our ensemble approach performs better than all the other approaches in terms of the fairness-utility trade-off, with the only exception in model performance by Reduction$_{EO}$ and in unfairness reduction by Threshold$_{DP}$.

5.2.3 Computation Efficiency. We compare the runtime of each method on the full combination of each dataset-sensitive attribute task and models used to generate our results (Figure 4). As model training tends to take the most time and our method trains group experts and a final model on a subsampled dataset, we look at the relationship between time and the size of the largest sensitive attribute group. We can see that our method is faster than Reweight and Reduction (Reduction$_{DP}$ and Reduction$_{EO}$) and our ensemble version has performance comparable to AdaFair's. Our single-model version typically has equivalent or slightly longer runtime compared to Threshold (Threshold$_{DP}$ and Threshold$_{EO}$). Generally, the performance of our approach is very lightweight compared to the other approaches while being able to apply to a much broader range of model architectures.

6 CONCLUSION

In this paper, we propose GroupDebias, an approach for reducing group unfairness in machine learning models by sub-sampling the dataset using the prediction of auxiliary group expert models. We provide a novel perspective to bias mitigation which uses the model to guide the data debiasing process. Compared to other fair machine learning methods, we are able to get superb fairness-utility trade-off with minimal assumptions about the target machine learning model. We provide theoretic bounds for the reduction in balanced accuracy and a guarantee for the improvement in the demographic parity metric. We also illustrate the advantage of our approach through comprehensive benchmarks. As machine learning become more integrated in real world applications, model fairness will become increasingly important in a wide variety of scenarios. In the approach outlined in this paper, we focus on rectifying historical bias. Furthermore, our proposed metrics is limited in that the values are highly dependent on the particular specific metrics used. We hope our unique approach based on group experts will provide new insight for developing versatile group fairness methods and inspire others to investigate the effects of consensus-based approaches on different types of biases as well as develop more general fairness-utility trade-off metrics.

ACKNOWLEDGMENTS

This work is supported by NSF (2134079 and 2324770), the NSF Program on Fairness in AI in collaboration with Amazon (1939725), DHS (17STQAC00001-07-00), the C3.ai Digital Transformation Institute, and IBM-Illinois Discovery Accelerator Institute. The content of the information in this document does not necessarily reflect the position or the policy of the Government or Amazon or IBM, and no official endorsement should be inferred. The U.S. Government is authorized to reproduce and distribute reprints for Government purposes notwithstanding any copyright notation here on.

REFERENCES

[1] 2009. Law School Admissions. Project SEAPHE. http://www.seaphe.org/databases.php.

[2] Alekh Agarwal, Alina Beygelzimer, Miroslav Dudík, John Langford, and Hanna Wallach. 2018. A reductions approach to fair classification. In *International conference on machine learning*. PMLR, 60–69.

[3] Barry Becker and Ronny Kohavi. 1996. Adult. UCI Machine Learning Repository. DOI: https://doi.org/10.24432/C5XW20.

[4] 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. 2018. AI Fairness 360: An Extensible Toolkit for Detecting, Understanding, and Mitigating Unwanted Algorithmic Bias. https://arxiv.org/abs/1810.01943

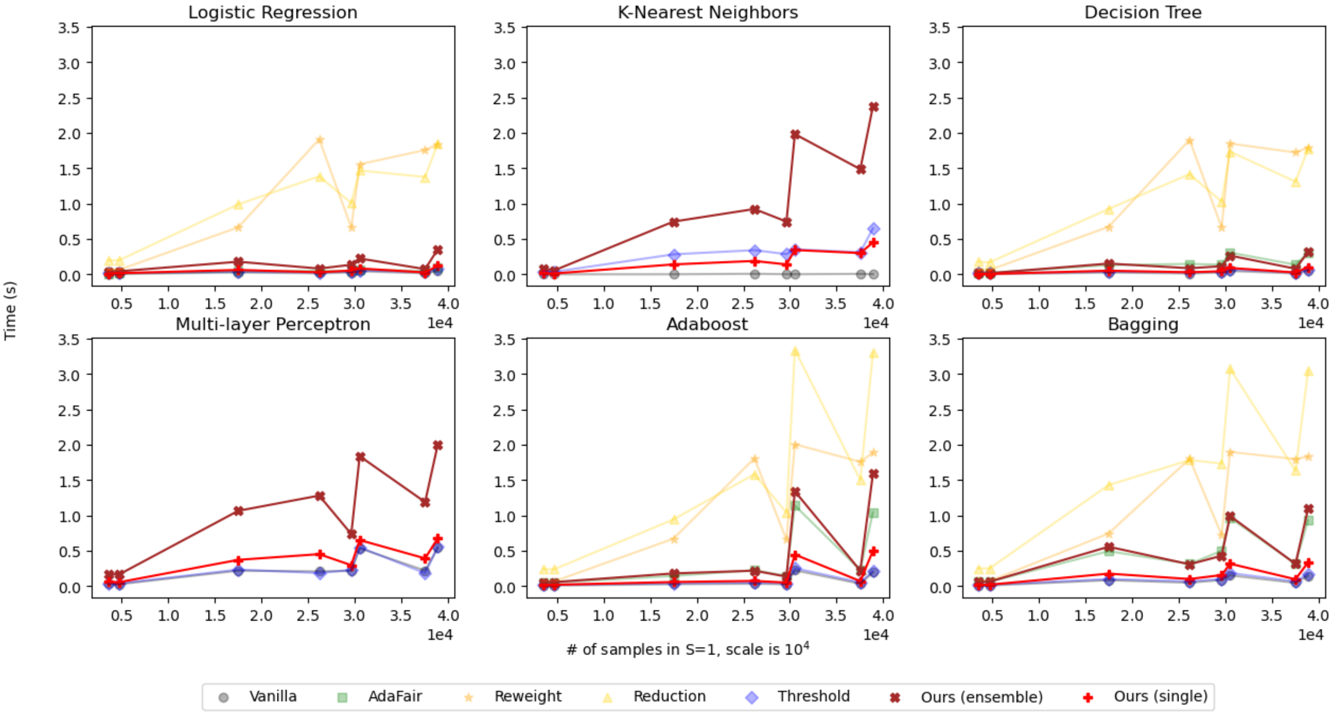


Figure 4: Running time under different number of samples. Best viewed in color.

[5] Reuben Binns. 2020. On the Apparent Conflict between Individual and Group Fairness. In *Proceedings of the 2020 Conference on Fairness, Accountability, and Transparency* (Barcelona, Spain) *(FAT* '20)*. Association for Computing Machinery, New York, NY, USA, 514–524. https://doi.org/10.1145/3351095.3372864

[6] Miranda Bogen and Aaron Rieke. 2018. Help wanted: an examination of hiring algorithms, equity, and bias. https://apo.org.au/node/210071

[7] Kay Henning Brodersen, Cheng Soon Ong, Klaas Enno Stephan, and Joachim M. Buhmann. 2010. The Balanced Accuracy and Its Posterior Distribution. In *2010 20th International Conference on Pattern Recognition*. 3121–3124. https://doi.org/10.1109/ICPR.2010.764

[8] Toon Calders and Sicco Verwer. 2010. Three naive Bayes approaches for discrimination-free classification. *Data Mining and Knowledge Discovery* 21, 2 (01 Sep 2010), 277–292. https://doi.org/10.1007/s10618-010-0190-x

[9] Simon Caton and Christian Haas. 2020. Fairness in Machine Learning: A Survey. *CoRR* abs/2010.04053 (2020). arXiv:2010.04053 https://arxiv.org/abs/2010.04053

[10] Junyi Chai and Xiaoqian Wang. 2022. Self-Supervised Fair Representation Learning without Demographics. In *Advances in Neural Information Processing Systems*, S. Koyejo, S. Mohamed, A. Agarwal, D. Belgrave, K. Cho, and A. Oh (Eds.), Vol. 35. Curran Associates, Inc., 27100–27113. https://proceedings.neurips.cc/paper_files/paper/2022/file/ad991bbc381626a8e44dc5414aa136a8-Paper-Conference.pdf

[11] YooJung Choi, Meihua Dang, and Guy Van den Broeck. 2021. Group Fairness by Probabilistic Modeling with Latent Fair Decisions. *Proceedings of the AAAI Conference on Artificial Intelligence* 35, 13 (May 2021), 12051–12059. https://doi.org/10.1609/aaai.v35i13.17431

[12] Abigail R. Colson and Roger M. Cooke. 2018. Expert Elicitation: Using the Classical Model to Validate Experts' Judgments. *Review of Environmental Economics and Policy* 12, 1 (2018), 113–132. https://doi.org/10.1093/reep/rex022 arXiv:https://doi.org/10.1093/reep/rex022

[13] Brian d'Alessandro, Cathy O'Neil, and Tom LaGatta. 2017. Conscientious Classification: A Data Scientist's Guide to Discrimination-Aware Classification. *Big Data* 5, 2 (2017), 120–134. https://doi.org/10.1089/big.2016.0048 arXiv:https://doi.org/10.1089/big.2016.0048 PMID: 28632437.

[14] Cynthia Dwork, Moritz Hardt, Toniann Pitassi, Omer Reingold, and Richard Zemel. 2012. Fairness through awareness. In *Proceedings of the 3rd innovations in theoretical computer science conference*. 214–226.

[15] Cynthia Dwork, Nicole Immorlica, Adam Tauman Kalai, and Max Leiserson. 2018. Decoupled classifiers for group-fair and efficient machine learning. In *Conference on fairness, accountability and transparency*. PMLR, 119–133.

[16] Sainyam Galhotra, Yuriy Brun, and Alexandra Meliou. 2017. Fairness Testing: Testing Software for Discrimination. In *Proceedings of the 2017 11th Joint Meeting on Foundations of Software Engineering* (Paderborn, Germany) *(ESEC/FSE 2017)*. Association for Computing Machinery, New York, NY, USA, 498–510. https://doi.org/10.1145/3106237.3106277

[17] Moritz Hardt, Eric Price, Eric Price, and Nati Srebro. 2016. Equality of Opportunity in Supervised Learning. In *Advances in Neural Information Processing Systems*, D. Lee, M. Sugiyama, U. Luxburg, I. Guyon, and R. Garnett (Eds.), Vol. 29. Curran Associates, Inc. https://proceedings.neurips.cc/paper_files/paper/2016/file/9d2682367c3935defcb1f9e247a97c0d-Paper.pdf

[18] Moritz Hardt, Eric Price, and Nati Srebro. 2016. Equality of opportunity in supervised learning. *Advances in neural information processing systems* 29 (2016).

[19] Hans Hofmann. 1994. Statlog (German Credit Data). UCI Machine Learning Repository. DOI: https://doi.org/10.24432/C5NC77.

[20] Vasileios Iosifidis and Eirini Ntoutsi. 2019. Adafair: Cumulative fairness adaptive boosting. In *Proceedings of the 28th ACM international conference on information and knowledge management*. 781–790.

[21] Faisal Kamiran and Toon Calders. 2009. Classifying without discriminating. In *2009 2nd International Conference on Computer, Control and Communication*. 1–6. https://doi.org/10.1109/IC4.2009.4909197

[22] Faisal Kamiran and Toon Calders. 2012. Data preprocessing techniques for classification without discrimination. *Knowledge and information systems* 33, 1 (2012), 1–33.

[23] Faisal Kamiran and Toon Calders. 2012. Data preprocessing techniques for classification without discrimination. *Knowledge and Information Systems* 33, 1 (01 Oct 2012), 1–33. https://doi.org/10.1007/s10115-011-0463-8

[24] Faisal Kamiran, Toon Calders, and Mykola Pechenizkiy. 2010. Discrimination Aware Decision Tree Learning. In *2010 IEEE International Conference on Data Mining*. 869–874. https://doi.org/10.1109/ICDM.2010.50

[25] Toshihiro Kamishima, Shotaro Akaho, Hideki Asoh, and Jun Sakuma. 2012. Fairness-Aware Classifier with Prejudice Remover Regularizer. In *Machine Learning and Knowledge Discovery in Databases*, Peter A. Flach, Tijl De Bie, and Nello Cristianini (Eds.). Springer Berlin Heidelberg, Berlin, Heidelberg, 35–50.

[26] Michael P. Kim, Amirata Ghorbani, and James Zou. 2019. Multiaccuracy: Black-Box Post-Processing for Fairness in Classification. In *Proceedings of the 2019 AAAI/ACM Conference on AI, Ethics, and Society* (Honolulu, HI, USA) *(AIES '19)*. Association for Computing Machinery, New York, NY, USA, 247–254. https://doi.org/10.1145/3306618.3314287

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

[28] Tai Le Quy, Arjun Roy, Vasileios Iosifidis, Wenbin Zhang, and Eirini Ntoutsi. 2022. A survey on datasets for fairness-aware machine learning. *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery* 12, 3 (2022), e1452.

[29] Xuran Li, Peng Wu, and Jing Su. 2023. Accurate Fairness: Improving Individual Fairness without Trading Accuracy. *Proceedings of the AAAI Conference on Artificial Intelligence* 37, 12 (Jun. 2023), 14312–14320. https://doi.org/10.1609/aaai.v37i12.26674

[30] Pranay K. Lohia, Karthikeyan Natesan Ramamurthy, Manish Bhide, Diptikalyan Saha, Kush R. Varshney, and Ruchir Puri. 2019. Bias Mitigation Post-processing for Individual and Group Fairness. In *ICASSP 2019 - 2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP).* 2847–2851. https://doi.org/10.1109/ICASSP.2019.8682620

[31] Ninareh Mehrabi, Fred Morstatter, Nripsuta Saxena, Kristina Lerman, and Aram Galstyan. 2021. A survey on bias and fairness in machine learning. *ACM computing surveys (CSUR)* 54, 6 (2021), 1–35.

[32] Alan Mishler, Edward H. Kennedy, and Alex Chouldechova. 2020. Fairness in Risk Assessment Instruments: Post-Processing to Achieve Counterfactual Equalized Odds. In *2021 ACM Conference on Fairness, Accountability, and Transparency.* https://www.microsoft.com/en-us/research/publication/fairness-in-risk-assessment-instruments-post-processing-to-achieve-counterfactual-equalized-odds/

[33] M. Granger Morgan. 2014. Use (and abuse) of expert elicitation in support of decision making for public policy. *Proceedings of the National Academy of Sciences* 111, 20 (2014), 7176–7184. https://doi.org/10.1073/pnas.1319946111 arXiv:https://www.pnas.org/doi/pdf/10.1073/pnas.1319946111

[34] S. Moro, P. Rita, and P. Cortez. 2012. Bank Marketing. UCI Machine Learning Repository. DOI: https://doi.org/10.24432/C5K306.

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

[36] Felix Petersen, Debarghya Mukherjee, Yuekai Sun, and Mikhail Yurochkin. 2021. Post-processing for Individual Fairness. In *Advances in Neural Information Processing Systems*, M. Ranzato, A. Beygelzimer, Y. Dauphin, P.S. Liang, and J. Wortman Vaughan (Eds.), Vol. 34. Curran Associates, Inc., 25944–25955. https://proceedings.neurips.cc/paper_files/paper/2021/file/d9fea4ca7e4a74c318ec27c1deb0796c-Paper.pdf

[37] Andrea Romei and Salvatore Ruggieri. 2014. A multidisciplinary survey on discrimination analysis. *The Knowledge Engineering Review* 29, 5 (2014), 582–638. https://doi.org/10.1017/S0269888913000039

[38] Zeyu Wang, Klint Qinami, Ioannis Christos Karakozis, Kyle Genova, Prem Nair, Kenji Hata, and Olga Russakovsky. 2020. Towards fairness in visual recognition: Effective strategies for bias mitigation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition.* 8919–8928.

[39] Rich Zemel, Yu Wu, Kevin Swersky, Toni Pitassi, and Cynthia Dwork. 2013. Learning Fair Representations. In *Proceedings of the 30th International Conference on Machine Learning (Proceedings of Machine Learning Research, Vol. 28)*, Sanjoy Dasgupta and David McAllester (Eds.). PMLR, Atlanta, Georgia, USA, 325–333. https://proceedings.mlr.press/v28/zemel13.html

Supplementary Materials

For our supplementary materials, we provide the implementation details for reproducing our results (Section A), statistics and details about the datasets we have benchmarked upon (Section B), details supporting our algorithm design including assumptions and proofs supporting our theoretical analysis (Section C), and additional experimental results benchmarking the fair ML techniques (Section D).

A REPRODUCIBILITY

A.1 Base Models

We use four base models: logistic regression, k-neighbors classifier, decision tree classifier (with no limit to the max depth), a multi-layer perceptron (with one layer of 8 neurons trained for up to 50 iterations), AdaBoost with 5 estimators and a decision tree base model, and bagging with 5 estimators and a decision tree base model. All other parameters are set to the default parameters as set in `sklearn`. The parameters are chosen due to the relatively small size of the datasets.

A.2 Our Algorithm

Our algorithm has three hyperparameters: the *debias intensity*, λ, the *target positive sample ratio*, α, and the *consensus drop weight*, ϵ.

In our experiments, we test 0.5 and 1.0 as the debias intensity for λ. We decide to use 1.0 to prioritize fairness over accuracy. For α, we use the group max positive sample rate after comparing the performance of that, the advantaged group positive sample rate, and the overall positive sample rate on the COMPAS (sex) task. Finally, we set ϵ to 0.1 so that the probability of selecting those points is less than the disagreement samples after trying another approach where we select from the samples with disagreement uniformly at random and if there are more points to subsample than the number of disagreement samples, then select from the samples with consensus also uniformly at random.

A.2.1 Ensemble Variant. For the ensemble version of our approach, we use an ensemble of five models trained on five variations of the dataset generated via our method.

A.3 Computing Infrastructure

Our code should be able to run on any modern computer. We have been able to replicate the results on a laptop running Windows 11 Home with 32 GB of RAM without using the GPU and using 13th Gen Intel(R) Core(TM) i9-13900HX 2.20GHz CPU. We use the `sklearn v1.3.0` [35] implementations for the base models and the `aif360 v0.5.0` [4] implementations for fair baselines.

B DATASETS

The input is zero-mean normalized and each task is run five times with five train-test split of the dataset done by shuffling and splitting the full dataset into an 80-20 split.

B.1 COMPAS (ProPublica COMPAS Dataset)

The COMPAS prediction task is to leverage information about an individual's criminal history and demographics to identify whether the individual would re-offend within two years [27].

We look at the COMPAS dataset in terms of the sensitive attribute sex, identified as **COMPAS (sex)**, and the sensitive attribute race, identified as **COMPAS (race)**.

B.2 LSA (Law School Admissions Dataset)

The LSA prediction task is to leverage the numerical credentials of the individual to identify admission decisions [1].

We look at the LSA dataset in terms of the sensitive attribute gender, identified as **LSA (gender)**, and the sensitive attribute race, identified as **LSA (race)**. We use a subset of the dataset where we remove some samples with positive outcome for the disadvantaged groups to exaggerate the inequality between the groups. This version of the dataset is provided with our code.

B.3 Adult (Adult Census Income Dataset)

The Adult prediction task is to leverage demographic information about an individual collected by a census to identify whether the individual makes more than $50K a year [3].

We look at the Adult dataset in terms of the sensitive attribute gender, identified as **Adult (gender)**, and the sensitive attribute race, identified as **Adult (race)**.

B.4 Bank (Bank Marketing Dataset)

The Bank prediction task is to leverage financial and demographic features to predict whether an individual would subscribe a term deposit [34]. We apply a pre-processing to the age attribute as done in Kamiran and Calders [21] to convert it to a binary value where the advantaged group is $25 \leq \text{age} < 60$ and the disadvantaged group is $\text{age} < 25$ or $\text{age} \leq 60$ as suggested in Le Quy et al. [28].

We look at the Bank dataset in terms of the sensitive attribute age, identified as **Bank (age)** and the sensitive attribute marital status, identified as **Bank (marital status)**. For this, we specifically use the **marital=married** attribute.

C ALGORITHM DETAILS APPENDIX

C.1 Consensus

In our algorithm, we use consensus to determine whether there is a discrepancy in the standard. This crucially depends on the assumption that the feature space for both groups are similar. If the feature space is not similar, that represents a dramatic distribution shift from the distribution of samples in one sensitive attribute group to the other and an auxiliary group expert model for one sensitive attribute group would not have good inference on the other group.

We find empirically that the assumption that the feature space is similar is generally true. For the tasks we've applied our method upon, the mean absolute difference between the average feature value for each sensitive attribute group (aside from the sensitive attribute features) is less than 0.01 for the datasets we use for our empirical analysis (Table 6).

C.2 Assumptions

Without loss of generality, we make the following standard assumptions to facilitate our analysis.

We make a standard assumption on the data collection process.

Table 6: The mean absolute difference in average feature values between the two sensitive attribute groups.

Dataset	Group Feature Difference
COMPAS (sex)	0.025
COMPAS (race)	0.037
Adult (sex)	0.035
Adult (race)	0.018
LSA (sex)	0.048
LSA (race)	0.079
Bank (age)	0.076
Bank (marital status)	0.039

Assumption 1 (Data collection). *The original dataset $\mathcal{D}$ consists of two independent groups: the disadvantaged group has $n_0 = \Theta(n)$ i.i.d. samples $(X, 0, y)$ with $(X, y) \sim P_{\mathbf{X},Y|S=0}$, and the advantaged group has $n_1 = \Theta(n)$ i.i.d. samples $(x, 1, y)$ with $(X, y) \sim P_{\mathbf{X},Y|S=1}$. We do not assume that $P_{\mathbf{X},Y|S=0}$ and $P_{\mathbf{X},Y|S=1}$ are identical.*

We assume that features are sufficiently fine-grained such that there are only at most a few repeated samples in the dataset. The intuition behind this assumption is that the dataset size is correlated with feature granularity. When collecting data, a good dataset would have enough samples to provide insight to the underlying trend, but not so much that the dataset is dominated with duplicate samples, because it would be a waste of resources.

Assumption 2 (Feature granularity). *There exists $0 \leq \gamma \leq 1$ such that*

$$\Pr[\boldsymbol{X} = S \mid S = s] \leq \frac{\gamma}{n_s}, \qquad \forall S \in \mathcal{X}, \forall s \in \mathcal{S}.$$

For example, if at least one of the features is continuous, this assumption holds with $\gamma = 0$.

We assume that unfairness exists (in terms of demographic parity).

Assumption 3 (Unfairness). *The advantaged group is more likely to be positively labeled than the disadvantaged group:*

$$\Delta := \Pr[Y = 1 \mid S = 1] - \Pr[Y = 1 \mid S = 0] > 0.$$

We assume that the classification problem is well defined.

Assumption 4 (Well-definedness). *The label Y is a function $Y : \mathcal{X} \times \mathcal{S} \to \mathcal{Y}$ of the features X and the attribute S.*

Since our method applies to various learning algorithms, we do not make assumptions on the learning algorithm in order to avoid the complication of different learning theories for different learning algorithms. Instead, we assume that the group experts are perfect and will show that our debiased training set $\mathcal{D} \setminus \bar{\mathcal{D}}$ gives rise to a classifier that achieves zero classification error over $\mathcal{D} \setminus \bar{\mathcal{D}}$ and is fair without significant loss of utility. As long as the learning algorithm is sufficiently good, we can expect that the learned model is close to this ideal classifier.

Assumption 5 (Group experts). *We assume that the group experts give the ground-truth labels.*

$$f_s(X) = Y(X, s), \qquad \forall X \in \mathcal{X},\ s \in \mathcal{S}.$$

C.3 Proof of Theorem 1

Note that for each $s \in \mathcal{S}$,

$$D_s := |\mathcal{D}_{S=s,Y=0}| = \sum_{(\mathbf{x}_i,s_i,y_i)\in\mathcal{D}_{S=s}} 1_{[y_i=0]} \sim \mathrm{Binomial}(n_s, 1-\pi_s)$$

is a sum of n_s i.i.d. Bernoullis $1_{[y_i=0]}$. Let $\Gamma := \frac{(1-\pi_0)-\lambda\Delta}{\gamma} \geq (1-\pi_0)-\lambda\Delta$. (Note that we are using the convention that $1/0=+\infty$, so the inequality above still holds even when $\gamma=0$.)

Using McDiarmid's inequality w.r.t. the Bernoullis $1_{[y_i=0]}$, since the deviation (i.e., the numerator in McDiarmid's inequality) is

$$\begin{aligned}
&\Gamma n_0 n_1 - n_1 - \mathbb{E}[(1-\lambda)n_1 D_0 + ((\lambda+\Gamma)n_0-1)D_1 - D_0 D_1] \\
&= \Gamma n_0 n_1 - n_1 - [(1-\lambda)n_1 n_0(1-\pi_0) \\
&\quad + ((\lambda+\Gamma)n_0-1)n_1(1-\pi_1) - n_0(1-\pi_0)n_1(1-\pi_1)] \\
&= ((\Gamma\pi_1 - \pi_0 + \pi_0\pi_1)n_0 - \pi_1)n_1 \\
&\geq (((1-\pi_0-\lambda\Delta)\pi_1 - \pi_0 + \pi_0\pi_1)n_0 - \pi_1)n_1 \\
&= ((\pi_1 - \pi_0 - \lambda\pi_1\Delta)n_0 - \pi_1)n_1 \\
&= ((1-\lambda\pi_1)\Delta n_0 - \pi_1)n_1 > 0,
\end{aligned}$$

and the sum of squared difference bounds w.r.t. the Bernoullis (i.e., the denominator in McDiarmid's inequality) is

$$\begin{aligned}
&n_0 \max\{((1-\lambda)n_1 - n_1)^2, ((1-\lambda)n_1 - 0)^2\} \\
&\quad + n_1 \max\{((\lambda+\Gamma)n_0 - 1 - n_0)^2, ((\lambda+\Gamma)n_0 - 1 - 0)^2\} \\
&= n_0 \max\{\lambda^2, (1-\lambda)^2\} n_1^2 + n_1 \max\{((\lambda+\Gamma-1)n_0-1)^2, \\
&\quad ((\lambda+\Gamma)n_0-1)^2\},
\end{aligned}$$

then we have

$$\begin{aligned}
&\Pr[(1-\lambda)n_1 D_0 + ((\lambda+\Gamma)n_0-1)D_1 - D_0 D_1 > \Gamma n_0 n_1 - n_1] \\
&\leq \mathrm{e}^{-\frac{2\left(((1-\lambda\pi_1)\Delta n_0-\pi_1)n_1\right)^2}{n_0\max\{\lambda^2,(1-\lambda)^2\}n_1^2+n_1\max\{((\lambda+\Gamma-1)n_0-1)^2,((\lambda+\Gamma)n_0-1)^2\}}} \\
&= \mathrm{e}^{-\frac{\Theta(n^4)}{\Theta(n^3)}} = \mathrm{e}^{-\Omega(n)}.
\end{aligned}$$

This implies

$$\begin{aligned}
&\Pr\left[(1-\lambda)|\mathcal{D}_{S=0,Y=0}| + \lambda\frac{|\mathcal{D}_{S=1,Y=0}|}{|\mathcal{D}_{S=1,Y=1}|}|\mathcal{D}_{S=0,Y=1}| + 1 > \Gamma n_0\right] \\
&= \Pr\left[(1-\lambda)D_0 + \lambda\frac{D_1}{n_1-D_1}(n_0-D_0) + 1 > \Gamma n_0\right] \\
&\leq \Pr[(1-\lambda)D_0(n_1-D_1) + \lambda D_1(n_0-D_0) + (n_1-D_1) \\
&\qquad > \Gamma n_0(n_1-D_1)] \\
&= \Pr[(1-\lambda)n_1 D_0 + ((\lambda+\Gamma)n_0-1)D_1 - D_0 D_1 > \Gamma n_0 n_1 - n_1] \\
&\leq \mathrm{e}^{-\Omega(n)}.
\end{aligned}$$

Let $\alpha := \frac{|\mathcal{D}_{S=1,Y=1}|}{n_1}$. Then,

$$\begin{aligned}
&\Pr\left[\frac{\gamma}{n_0}|\mathcal{D}_{S=0,Y=0} \setminus \bar{\mathcal{D}}_0| \leq 1-\pi_0-\lambda\Delta\right] \\
&= \Pr\Bigg[|\mathcal{D}_{S=0,Y=0}| - \max\left\{0, \left\lfloor\lambda\left(|\mathcal{D}_{S=0,Y=0}| - \frac{1-\alpha}{\alpha}|\mathcal{D}_{S=0,Y=1}|\right)\right\rfloor\right\} \\
&\qquad \leq \frac{n_0}{\gamma}(1-\pi_0-\lambda\Delta)\Bigg] \\
&= \Pr\Bigg[|\mathcal{D}_{S=0,Y=0}| - \max\Bigg\{0, \Bigg\lfloor\lambda\Bigg(|\mathcal{D}_{S=0,Y=0}| \\
&\qquad - \frac{|\mathcal{D}_{S=1,Y=0}|}{|\mathcal{D}_{S=1,Y=1}|}|\mathcal{D}_{S=0,Y=1}|\Bigg)\Bigg\rfloor\Bigg\} \leq \Gamma n_0\Bigg] \\
&\geq \Pr\Bigg[|\mathcal{D}_{S=0,Y=0}| - \lambda\Bigg(|\mathcal{D}_{S=0,Y=0}| \\
&\qquad - \frac{|\mathcal{D}_{S=1,Y=0}|}{|\mathcal{D}_{S=1,Y=1}|}|\mathcal{D}_{S=0,Y=1}|\Bigg) + 1 \leq \Gamma n_0\Bigg] \\
&= \Pr\left[(1-\lambda)|\mathcal{D}_{S=0,Y=0}| + \lambda\frac{|\mathcal{D}_{S=1,Y=0}|}{|\mathcal{D}_{S=1,Y=1}|}|\mathcal{D}_{S=0,Y=1}| + 1 \leq \Gamma n_0\right] \\
&= 1 - \Pr\left[(1-\lambda)|\mathcal{D}_{S=0,Y=0}| + \lambda\frac{|\mathcal{D}_{S=1,Y=0}|}{|\mathcal{D}_{S=1,Y=1}|}|\mathcal{D}_{S=0,Y=1}| + 1 > \Gamma n_0\right] \\
&\geq 1 - \mathrm{e}^{-\Omega(n)}.
\end{aligned}$$

Under the event above,

$$\begin{aligned}
\Pr[(\mathbf{X},S) \in \mathcal{D}_{S=0,Y=0} \setminus \bar{\mathcal{D}}_0 \mid S=0] &\leq \frac{\gamma}{n_0}|\mathcal{D}_{S=0,Y=0} \setminus \bar{\mathcal{D}}_0| \\
&\leq 1-\pi_0-\lambda\Delta.
\end{aligned}$$

Then, there exists a set A with

$$(\mathcal{D}_{S=0,Y=0} \setminus \bar{\mathcal{D}}_0) \subseteq A \subseteq \{(\mathbf{x},s) \in \mathcal{X}\times\mathcal{S} : Y(\mathbf{x},s)=0\}$$

such that $\Pr[Y=0 \mid (\mathbf{X},S) \in A] = 1$, and

$$|(1-\pi_0-\lambda\Delta) - \Pr[(\mathbf{X},S) \in A \mid S=0]| \leq \frac{\gamma}{n_0} = O\left(\frac{1}{n}\right).$$

Define the classifier $\widehat{f} : \mathcal{X}\times\mathcal{S} \to \mathcal{Y}$ by

$$\widehat{f}(\mathbf{x},s) := \begin{cases} 1_{[(\mathbf{x},s)\notin A]}, & \text{if } s=0, \\ Y(\mathbf{x},s), & \text{if } s=1, \end{cases} \qquad \mathbf{x}\in\mathcal{X},\ s\in\mathcal{S}.$$

It is clear that $\widehat{f}$ achieves zero classification error over $\mathcal{D} \setminus \bar{\mathcal{D}}$. Regarding fairness,

$$\begin{aligned}
&\left|\Pr[\widehat{f}(\mathbf{X},S)=1 \mid S=0] - \Pr[\widehat{f}(\mathbf{X},S)=1 \mid S=1]\right| \\
&= \left|\Pr[(\mathbf{X},S) \notin A \mid S=0] - \Pr[Y=1 \mid S=1]\right| \\
&= \left|\Pr[(\mathbf{X},S) \notin A \mid S=0] - \pi_1\right| \\
&\leq \left|(\pi_0+\lambda\Delta) - \pi_1\right| + \left|\Pr[(\mathbf{X},S) \notin A \mid S=0] - (\pi_0+\lambda\Delta)\right| \\
&= \left|(\pi_0+\lambda\Delta) - \pi_1\right| + \left|(1-\pi_0-\lambda\Delta) - \Pr[(\mathbf{X},S) \in A \mid S=0]\right| \\
&\leq \left|(\pi_0+\lambda\Delta) - \pi_1\right| + O\left(\frac{1}{n}\right) \\
&= \left|\lambda\Delta - \Delta\right| + O\left(\frac{1}{n}\right) \\
&= (1-\lambda)\Delta + O\left(\frac{1}{n}\right).
\end{aligned}$$

Regarding utility, since $A \subseteq \{(\mathbf{x}, s) \in \mathcal{X} \times \mathcal{S} : Y(\mathbf{x}, s) = 0\}$, then

$$\begin{aligned}
&\sum_{s \in \mathcal{S}} \Pr[\widehat{f}(\mathbf{X}, S) \neq Y \mid S = s] \\
&= \Pr[1_{[(\mathbf{X},S)\notin A]} \neq Y \mid S = 0] + \Pr[Y(\mathbf{X}, S) \neq Y \mid S = 1] \\
&= \Pr[1_{[(\mathbf{X},S)\notin A]} \neq Y \mid S = 0] + 0 \\
&= \Pr[(\mathbf{X}, S) \notin A, Y = 0 \mid S = 0] + \Pr[(\mathbf{X}, S) \in A, Y = 1 \mid S = 0] \\
&= \Pr[(\mathbf{X}, S) \notin A, Y = 0 \mid S = 0] + 0 \\
&= \Pr[Y = 0 \mid S = 0] - \Pr[(\mathbf{X}, S) \in A, Y = 0 \mid S = 0] \\
&= (1 - \pi_0) - \Pr[(\mathbf{X}, S) \in A, Y = 0 \mid S = 0] \\
&= (1 - \pi_0) - \Pr[(\mathbf{X}, S) \in A \mid S = 0] \\
&\leq (1 - \pi_0) - \left(1 - \pi_0 - \lambda\Delta - O\left(\frac{1}{n}\right)\right) \\
&= \lambda\Delta + O\left(\frac{1}{n}\right).
\end{aligned}$$

C.4 Proof of Proposition 1

We have that

$$\begin{aligned}
&\Pr[\widehat{Y} = 1, Y = 1 \mid S = 1] - \Pr[\widehat{Y} = 1, Y = 1 \mid S = 0] \\
&= \Pr[Y = 1 \mid S = 1] \Pr[\widehat{Y} = 1 \mid Y = 1, S = 1] \\
&\quad - \Pr[Y = 1 \mid S = 0] \Pr[\widehat{Y} = 1 \mid Y = 1, S = 0] \\
&= \Pr[Y = 1 \mid S = 1] \Pr[\widehat{Y} = 1 \mid Y = 1, S = 1] \\
&\quad - \Pr[Y = 1 \mid S = 0] \Pr[\widehat{Y} = 1 \mid Y = 1, S = 1] \\
&\quad + \Pr[Y = 1 \mid S = 0] \Pr[\widehat{Y} = 1 \mid Y = 1, S = 1] \\
&\quad - \Pr[Y = 1 \mid S = 0] \Pr[\widehat{Y} = 1 \mid Y = 1, S = 0] \\
&= (\Pr[Y = 1 \mid S = 1] \\
&\quad - \Pr[Y = 1 \mid S = 0]) \Pr[\widehat{Y} = 1 \mid Y = 1, S = 1] \\
&\quad + \Pr[Y = 1 \mid S = 0](\Pr[\widehat{Y} = 1 \mid Y = 1, S = 1] \\
&\quad - \Pr[\widehat{Y} = 1 \mid Y = 1, S = 0]) \\
&= \Delta \cdot \Pr[\widehat{Y} = 1 \mid Y = 1, S = 1] + \pi_0 \cdot \mathrm{EO}_{Y=1},
\end{aligned}$$

and that

$$\begin{aligned}
&\Pr[\widehat{Y} = 1, Y = 0 \mid S = 1] - \Pr[\widehat{Y} = 1, Y = 0 \mid S = 0] \\
&= \Pr[Y = 0 \mid S = 1] \Pr[\widehat{Y} = 1 \mid Y = 0, S = 1] \\
&\quad - \Pr[Y = 0 \mid S = 0] \Pr[\widehat{Y} = 1 \mid Y = 0, S = 0] \\
&= \Pr[Y = 0 \mid S = 1] \Pr[\widehat{Y} = 1 \mid Y = 0, S = 1] \\
&\quad - \Pr[Y = 0 \mid S = 0] \Pr[\widehat{Y} = 1 \mid Y = 0, S = 1] \\
&\quad + \Pr[Y = 0 \mid S = 0] \Pr[\widehat{Y} = 1 \mid Y = 0, S = 1] \\
&\quad - \Pr[Y = 0 \mid S = 0] \Pr[\widehat{Y} = 1 \mid Y = 0, S = 0] \\
&= (\Pr[Y = 0 \mid S = 1] \\
&\quad - \Pr[Y = 0 \mid S = 0]) \Pr[\widehat{Y} = 1 \mid Y = 0, S = 1] \\
&\quad + \Pr[Y = 0 \mid S = 0](\Pr[\widehat{Y} = 1 \mid Y = 0, S = 1] \\
&\quad - \Pr[\widehat{Y} = 1 \mid Y = 0, S = 0]) \\
&= ((1 - \Pr[Y = 1 \mid S = 1]) \\
&\quad - (1 - \Pr[Y = 1 \mid S = 0])) \Pr[\widehat{Y} = 1 \mid Y = 0, S = 1] \\
&\quad + (1 - \Pr[Y = 1 \mid S = 0])(\Pr[\widehat{Y} = 1 \mid Y = 0, S = 1] \\
&\quad - \Pr[\widehat{Y} = 1 \mid Y = 0, S = 0]) \\
&= -\Delta \cdot \Pr[\widehat{Y} = 1 \mid Y = 0, S = 1] + (1 - \pi_0) \cdot \mathrm{EO}_{Y=0}.
\end{aligned}$$

Thus,

$$\begin{aligned}
\mathrm{DP} = &\Pr[\widehat{Y} = 1 \mid S = 1] - \Pr[\widehat{Y} = 1 \mid S = 0] \\
&= (\Pr[\widehat{Y} = 1, Y = 1 \mid S = 1] + \Pr[\widehat{Y} = 1, Y = 0 \mid S = 1]) \\
&\quad - (\Pr[\widehat{Y} = 1, Y = 1 \mid S = 0] + \Pr[\widehat{Y} = 1, Y = 0 \mid S = 0]) \\
&= (\Pr[\widehat{Y} = 1, Y = 1 \mid S = 1] - \Pr[\widehat{Y} = 1, Y = 1 \mid S = 0]) \\
&\quad + (\Pr[\widehat{Y} = 1, Y = 0 \mid S = 1] - \Pr[\widehat{Y} = 1, Y = 0 \mid S = 0]) \\
&= \Delta \cdot \Pr[\widehat{Y} = 1 \mid Y = 1, S = 1] + \pi_0 \cdot \mathrm{EO}_{Y=1} \\
&\quad - \Delta \cdot \Pr[\widehat{Y} = 1 \mid Y = 0, S = 1] + (1 - \pi_0) \cdot \mathrm{EO}_{Y=0} \\
&= \Delta \cdot \Pr[\widehat{Y} = 1 \mid Y = 1, S = 1] + \pi_0 \cdot \mathrm{EO}_{Y=1} \\
&\quad + \Delta \cdot (1 - \Pr[\widehat{Y} = 1 \mid Y = 0, S = 1] - 1) + (1 - \pi_0) \cdot \mathrm{EO}_{Y=0} \\
&= \Delta \cdot \Pr[\widehat{Y} = 1 \mid Y = 1, S = 1] + \pi_0 \cdot \mathrm{EO}_{Y=1} \\
&\quad + \Delta \cdot (\Pr[\widehat{Y} = 0 \mid Y = 0, S = 1] - 1) + (1 - \pi_0) \cdot \mathrm{EO}_{Y=0} \\
&= \Delta \cdot (\Pr[\widehat{Y} = 1 \mid Y = 1, S = 1] + \Pr[\widehat{Y} = 0 \mid Y = 0, S = 1] - 1) \\
&\quad + \pi_0 \cdot \mathrm{EO}_{Y=1} + (1 - \pi_0) \cdot \mathrm{EO}_{Y=0} \\
&= \Delta \cdot (\mathrm{Util}_{S=1} - 1) + \pi_0 \cdot \mathrm{EO}_{Y=1} + (1 - \pi_0) \cdot \mathrm{EO}_{Y=0}.
\end{aligned}$$

D EXPERIMENTAL RESULTS

See Table 7 for a full comparison of our method with other fair ML techniques on the COMPAS (sex) dataset.

Table 4 and Table 5 present the average rank of the different FairML techniques on all tasks. For a per-task breakdown of the results aggregated in Table 4, see Table 8 and Table 9. For the breakdown of Table 5, see Table 10, Table 11, Table 12, Table 13.

Table 7: Comparison of our method with other FairML techniques on adaptability to different black-box classifiers (on COMPAS (sex) dataset).

Base Model	Method	Utility				Fairness				Unified	
		Acc.	Δ	BAcc.	Δ	DP	Δ	EO	Δ	FURG	FUTR
Dummy		49.69$_{\pm 2.1}$	-	49.49$_{\pm 2.2}$	-	0.97$_{\pm 0.6}$	-	4.40$_{\pm 4.3}$	-	-	-
LR	Vanilla	69.60$_{\pm 0.6}$	-	69.00$_{\pm 0.6}$	-	33.62$_{\pm 1.6}$	-	38.23$_{\pm 3.9}$	-	-	-
	Reweight	68.89$_{\pm 0.9}$	-3.59%	67.68$_{\pm 0.9}$	-6.74%	27.51$_{\pm 2.2}$	-18.18%	34.27$_{\pm 5.5}$	-10.36%	9.10	2.76
	Reduction$_{DP}$	68.92$_{\pm 1.0}$	-3.42%	68.59$_{\pm 1.0}$	-2.08%	10.48$_{\pm 4.1}$	-68.82%	8.57$_{\pm 4.3}$	-77.60%	70.46	26.62
	Reduction$_{EO}$	68.73$_{\pm 0.9}$	-4.36%	68.23$_{\pm 0.9}$	-3.93%	12.85$_{\pm 2.8}$	-61.80%	10.94$_{\pm 3.3}$	-71.38%	62.44	16.06
	Threshold$_{DP}$	66.98$_{\pm 0.8}$	-13.16%	66.83$_{\pm 0.8}$	-11.10%	3.15$_{\pm 1.0}$	-90.64%	14.56$_{\pm 5.1}$	-61.92%	64.15	6.29
	Threshold$_{EO}$	62.04$_{\pm 1.0}$	-37.95%	60.26$_{\pm 1.0}$	-44.81%	3.22$_{\pm 2.6}$	-90.42%	6.36$_{\pm 3.9}$	-83.37%	45.51	2.10
	AdaFair	59.54$_{\pm 3.6}$	-50.51%	58.10$_{\pm 4.7}$	-55.87%	19.43$_{\pm 12.4}$	-42.20%	25.65$_{\pm 16.1}$	-32.92%	-15.63	0.71
	Ours	69.04$_{\pm 1.2}$	-2.82%	68.86$_{\pm 1.2}$	-0.71%	2.71$_{\pm 2.6}$	-91.94%	7.73$_{\pm 2.3}$	-79.78%	$\underline{84.09}$	$\underline{48.62}$
	Ours$_{ens}$	69.11$_{\pm 1.0}$	-2.48%	68.90$_{\pm 1.1}$	-0.48%	2.45$_{\pm 2.2}$	-92.71%	7.58$_{\pm 2.1}$	-80.18%	**84.96**	**58.40**
KNN	Vanilla	64.97$_{\pm 1.1}$	-	64.54$_{\pm 1.2}$	-	15.53$_{\pm 1.2}$	-	13.98$_{\pm 1.2}$	-	-	-
	Threshold$_{DP}$	63.32$_{\pm 1.4}$	-10.80%	63.14$_{\pm 1.5}$	-9.34%	3.09$_{\pm 3.6}$	-80.10%	9.24$_{\pm 4.8}$	-33.89%	46.92	5.66
	Threshold$_{EO}$	62.25$_{\pm 1.3}$	-17.82%	61.43$_{\pm 1.4}$	-20.70%	2.48$_{\pm 1.4}$	-84.01%	7.24$_{\pm 3.4}$	-48.19%	46.84	3.43
	Ours	64.24$_{\pm 1.3}$	-4.79%	63.99$_{\pm 1.3}$	-3.70%	3.10$_{\pm 1.6}$	-80.04%	5.32$_{\pm 5.7}$	-61.97%	**66.76**	$\underline{16.72}$
	Ours$_{ens}$	64.27$_{\pm 1.3}$	-4.57%	64.02$_{\pm 1.3}$	-3.51%	3.19$_{\pm 2.7}$	-79.49%	5.48$_{\pm 3.4}$	-60.77%	$\underline{66.09}$	**17.37**
MLP	Vanilla	68.63$_{\pm 1.4}$	-	68.00$_{\pm 1.4}$	-	36.12$_{\pm 3.0}$	-	43.06$_{\pm 6.7}$	-	-	-
	Threshold$_{DP}$	65.72$_{\pm 0.8}$	-15.36%	65.68$_{\pm 0.8}$	-12.55%	2.90$_{\pm 2.1}$	-91.96%	15.65$_{\pm 5.5}$	-63.66%	$\underline{63.85}$	5.58
	Threshold$_{EO}$	59.29$_{\pm 0.9}$	-49.33%	57.19$_{\pm 1.0}$	-58.39%	4.88$_{\pm 2.7}$	-86.50%	8.97$_{\pm 4.9}$	-79.16%	28.97	1.54
	Ours	66.88$_{\pm 1.2}$	-9.25%	66.69$_{\pm 1.0}$	-7.09%	13.06$_{\pm 10.1}$	-63.84%	20.66$_{\pm 6.4}$	-52.02%	49.76	$\underline{7.09}$
	Ours$_{ens}$	68.32$_{\pm 0.8}$	-1.62%	68.01$_{\pm 0.8}$	+0.04%	6.23$_{\pm 4.5}$	-82.75%	6.23$_{\pm 4.0}$	-85.53%	**83.35**	**84.14**
DT	Vanilla	60.71$_{\pm 1.3}$	-	60.30$_{\pm 1.3}$	-	8.45$_{\pm 3.7}$	-	9.04$_{\pm 3.5}$	-	-	-
	Reweight	60.51$_{\pm 1.1}$	-1.85%	60.06$_{\pm 1.1}$	-2.20%	8.18$_{\pm 2.8}$	-3.21%	10.08$_{\pm 2.2}$	+11.43%	-6.14	-2.03
	Reduction$_{DP}$	59.66$_{\pm 0.6}$	-9.57%	59.32$_{\pm 0.7}$	-9.05%	5.96$_{\pm 3.8}$	-29.48%	10.16$_{\pm 2.7}$	+12.40%	-0.77	0.92
	Reduction$_{EO}$	60.87$_{\pm 0.7}$	+1.39%	60.41$_{\pm 0.7}$	+1.02%	6.17$_{\pm 1.7}$	-27.07%	9.59$_{\pm 2.9}$	+6.01%	$\underline{11.73}$	$\underline{10.53}$
	Threshold$_{DP}$	59.71$_{\pm 0.8}$	-9.10%	59.43$_{\pm 0.7}$	-8.04%	4.20$_{\pm 2.9}$	-50.38%	10.12$_{\pm 3.9}$	+11.96%	10.64	2.24
	Threshold$_{EO}$	60.65$_{\pm 0.9}$	-0.62%	60.23$_{\pm 1.0}$	-0.68%	7.83$_{\pm 4.2}$	-7.36%	9.67$_{\pm 5.9}$	+6.90%	-0.42	0.23
	AdaFair	61.46$_{\pm 1.4}$	+6.79%	61.04$_{\pm 1.4}$	+6.85%	10.07$_{\pm 2.2}$	+19.09%	12.06$_{\pm 3.4}$	+33.42%	-19.43	-26.25
	Ours	59.90$_{\pm 1.9}$	-7.41%	59.58$_{\pm 1.9}$	-6.61%	3.81$_{\pm 3.3}$	-54.92%	10.73$_{\pm 6.4}$	+18.71%	11.09	2.58
	Ours$_{ens}$	60.44$_{\pm 1.3}$	-2.47%	60.13$_{\pm 1.3}$	-1.57%	1.69$_{\pm 2.2}$	-79.98%	5.57$_{\pm 1.7}$	-38.45%	**57.19**	**29.30**
AdaBoost	Vanilla	60.87$_{\pm 0.8}$	-	60.27$_{\pm 0.7}$	-	8.50$_{\pm 3.6}$	-	10.37$_{\pm 5.8}$	-	-	-
	Reweight	61.26$_{\pm 1.4}$	+3.50%	60.67$_{\pm 1.3}$	+3.70%	7.53$_{\pm 3.3}$	-11.39%	9.52$_{\pm 2.2}$	-8.24%	13.42	9.81
	Reduction$_{DP}$	60.65$_{\pm 0.7}$	-1.98%	60.13$_{\pm 0.6}$	-1.31%	4.70$_{\pm 2.2}$	-44.72%	6.68$_{\pm 3.7}$	-35.62%	$\underline{38.53}$	24.44
	Reduction$_{EO}$	60.73$_{\pm 1.0}$	-1.22%	60.31$_{\pm 0.9}$	+0.35%	4.70$_{\pm 3.3}$	-44.66%	8.97$_{\pm 3.7}$	-13.51%	28.65	$\underline{29.08}$
	Threshold$_{DP}$	59.81$_{\pm 1.0}$	-9.44%	59.41$_{\pm 0.8}$	-7.94%	4.68$_{\pm 4.2}$	-44.90%	11.15$_{\pm 6.2}$	+7.54%	9.99	2.15
	Threshold$_{EO}$	60.77$_{\pm 1.0}$	-0.91%	60.22$_{\pm 0.8}$	-0.45%	7.63$_{\pm 5.5}$	-10.17%	11.58$_{\pm 7.1}$	+11.67%	-1.43	-0.75
	AdaFair	60.83$_{\pm 1.4}$	-0.30%	60.49$_{\pm 1.4}$	+2.09%	12.52$_{\pm 3.6}$	+47.33%	12.49$_{\pm 4.9}$	+20.49%	-33.01	-33.91
	Ours	59.98$_{\pm 0.8}$	-7.91%	59.54$_{\pm 0.8}$	-6.81%	3.89$_{\pm 1.7}$	-54.28%	7.76$_{\pm 4.6}$	-25.14%	32.35	5.39
	Ours$_{ens}$	60.85$_{\pm 0.9}$	-0.15%	60.49$_{\pm 0.9}$	+2.08%	2.86$_{\pm 2.8}$	-66.39%	6.55$_{\pm 3.8}$	-36.81%	**52.57**	**51.60**
Bagging	Vanilla	61.62$_{\pm 0.3}$	-	61.29$_{\pm 0.2}$	-	13.61$_{\pm 2.2}$	-	13.11$_{\pm 1.6}$	-	-	-
	Reweight	62.30$_{\pm 0.9}$	+5.71%	61.97$_{\pm 0.8}$	+5.74%	12.63$_{\pm 4.1}$	-7.25%	12.19$_{\pm 4.2}$	-7.05%	12.87	7.15
	Reduction$_{DP}$	62.50$_{\pm 0.8}$	+7.42%	62.17$_{\pm 0.8}$	+7.46%	5.28$_{\pm 3.2}$	-61.22%	4.90$_{\pm 2.1}$	-62.65%	69.37	61.94
	Reduction$_{EO}$	62.14$_{\pm 0.6}$	+4.42%	61.78$_{\pm 0.6}$	+4.14%	7.07$_{\pm 3.6}$	-48.06%	7.36$_{\pm 3.6}$	-43.88%	50.26	45.97
	Threshold$_{DP}$	61.33$_{\pm 1.1}$	-2.43%	61.25$_{\pm 1.2}$	-0.32%	2.45$_{\pm 1.7}$	-82.00%	8.53$_{\pm 3.3}$	-34.96%	57.11	42.67
	Threshold$_{EO}$	62.08$_{\pm 0.8}$	+3.85%	61.86$_{\pm 0.8}$	+4.88%	6.54$_{\pm 2.3}$	-51.96%	6.99$_{\pm 1.8}$	-46.69%	53.69	49.33
	AdaFair	63.81$_{\pm 1.0}$	+18.40%	63.45$_{\pm 1.0}$	+18.29%	12.53$_{\pm 2.4}$	-7.97%	10.42$_{\pm 3.6}$	-20.53%	32.60	14.25
	Ours	61.86$_{\pm 1.4}$	+2.00%	61.66$_{\pm 1.4}$	+3.13%	3.16$_{\pm 1.8}$	-76.76%	2.56$_{\pm 1.2}$	-80.51%	$\underline{81.20}$	**78.64**
	Ours$_{ens}$	63.37$_{\pm 1.4}$	+14.69%	63.15$_{\pm 1.3}$	+15.80%	3.64$_{\pm 2.5}$	-73.24%	4.02$_{\pm 2.3}$	-69.36%	**86.55**	$\underline{71.30}$

Table 8: Rank of different FairML techniques with 6 target machine learning models separated into the 8 tasks (4 datasets x 2 sensitive attributes). The missing cells represent invalid combinations. Results for the COMPAS (sex), COMPAS (race), Adult (sex) and Adult (race) tasks.

Task	Method	LR		KNN		MLP		DT		ADA		BAG	
		FURG	FUTR	FURG	FUTR	FURG	FUTR	FURG	FUTR	FURG	FUTR	FURG	FUTR
COMPAS (sex)	Reweight	7	6	-	-	-	-	7	7	5	4	8	8
	Reduction$_{DP}$	3	3	-	-	-	-	6	5	2	3	3	3
	Reduction$_{EO}$	5	4	-	-	-	-	2	2	4	2	6	5
	Threshold$_{DP}$	4	5	3	3	2	3	4	4	6	6	4	6
	Threshold$_{EO}$	6	7	4	4	4	4	5	6	7	7	5	4
	AdaFair	8	8	-	-	-	-	8	8	8	8	7	7
	Ours	2	2	1	2	3	2	3	3	3	5	2	1
	Ours$_{ens}$	1	1	2	1	1	1	1	1	1	1	1	2
COMPAS (race)	Reweight	7	7	-	-	-	-	4	2	5	4	8	7
	Reduction$_{DP}$	6	6	-	-	-	-	7	6	6	6	5	3
	Reduction$_{EO}$	5	5	-	-	-	-	6	7	8	8	7	6
	Threshold$_{DP}$	4	3	2	4	4	3	3	3	3	3	2	4
	Threshold$_{EO}$	3	4	1	3	3	4	5	5	4	2	4	2
	AdaFair	8	8	-	-	-	-	8	8	7	7	6	8
	Ours	1	1	3	1	2	2	1	4	2	5	3	5
	Ours$_{ens}$	2	2	4	2	1	1	2	1	1	1	1	1
Adult (sex)	Reweight	6	7	-	-	-	-	6	4	5	2	6	3
	Reduction$_{DP}$	8	8	-	-	-	-	4	3	2	4	5	4
	Reduction$_{EO}$	7	3	-	-	-	-	8	8	7	7	8	6
	Threshold$_{DP}$	4	5	2	4	4	4	3	5	4	5	3	8
	Threshold$_{EO}$	1	4	1	3	1	3	7	7	6	6	4	5
	AdaFair	5	6	-	-	-	-	5	6	8	8	7	7
	Ours	2	1	3	1	2	1	1	2	1	3	2	2
	Ours$_{ens}$	3	2	4	2	3	2	2	1	3	1	1	1
Adult (race)	Reweight	7	7	-	-	-	-	3	3	3	2	2	1
	Reduction$_{DP}$	1	4	-	-	-	-	5	5	5	5	5	4
	Reduction$_{EO}$	6	3	-	-	-	-	8	8	6	6	6	5
	Threshold$_{DP}$	3	5	1	3	2	3	4	4	1	4	1	6
	Threshold$_{EO}$	2	6	4	4	1	4	7	7	7	7	7	7
	AdaFair	8	8	-	-	-	-	6	6	8	8	8	8
	Ours	4	1	3	2	4	2	1	1	4	3	3	2
	Ours$_{ens}$	5	2	2	1	3	1	2	2	2	1	4	3

D.1 Select Table 3 Results

On the Adult (sex) task, Reduction$_{DP}$ has a negative FURG value. Our trade-off metrics equally considers the method's performance on both EO and DP. Although Reduction$_{DP}$ strongly improved DP, EO had significantly deteriorated (+148.64%) which results in an overall poor FURG value.

On the LSA (race) task, AdaFair resulted in a constant classifier. That is, it only classified as the majority class. This can be noticed by the BAcc score which is 50% which implies one class is predicted 100% of the time and the other class is predicted 0% of the time. As a result, BAcc performance has a -98.42% relative loss w.r.t the vanilla and dummy classifiers which yields lower FUTR and FURG values.

Received 22 January 2024; revised 18 March 2024; accepted 29 March 2024

Table 9: Rank of different FairML techniques with 6 target machine learning models separated into the 8 tasks (4 datasets x 2 sensitive attributes). The missing cells represent invalid combinations. Results for the LSA (sex), LSA (race), Bank (age), and Bank (marital) tasks.

Task	Method	LR		KNN		MLP		DT		ADA		BAG	
		FURG	FUTR	FURG	FUTR	FURG	FUTR	FURG	FUTR	FURG	FUTR	FURG	FUTR
LSA (sex)	Reweight	8	8	-	-	-	-	7	4	7	4	7	3
	Reduction$_{DP}$	2	4	-	-	-	-	1	2	1	2	1	2
	Reduction$_{EO}$	1	3	-	-	-	-	2	1	2	1	2	1
	Threshold$_{DP}$	6	6	2	4	3	4	4	7	4	7	4	7
	Threshold$_{EO}$	3	5	1	3	1	3	6	5	6	5	6	6
	AdaFair	7	7	-	-	-	-	8	8	8	8	8	8
	Ours	5	1	3	2	2	1	5	3	5	3	3	5
	Ours$_{ens}$	4	2	4	1	4	2	3	6	3	6	5	4
LSA (race)	Reweight	7	7	-	-	-	-	1	2	2	3	4	2
	Reduction$_{DP}$	5	3	-	-	-	-	4	1	4	1	5	4
	Reduction$_{EO}$	8	5	-	-	-	-	8	8	7	8	8	8
	Threshold$_{DP}$	3	4	3	3	3	3	2	4	1	4	3	5
	Threshold$_{EO}$	4	6	4	4	4	4	7	6	6	6	6	6
	AdaFair	6	8	-	-	-	-	6	7	8	7	7	7
	Ours	2	2	1	2	1	1	5	5	5	5	1	3
	Ours$_{ens}$	1	1	2	1	2	2	3	3	3	2	2	1
Bank (age)	Reweight	5	7	-	-	-	-	8	8	8	8	8	8
	Reduction$_{DP}$	4	4	-	-	-	-	2	1	2	1	6	6
	Reduction$_{EO}$	3	1	-	-	-	-	4	3	5	4	4	4
	Threshold$_{DP}$	6	5	4	3	4	3	6	4	7	7	7	5
	Threshold$_{EO}$	7	6	3	4	3	4	7	7	4	3	2	2
	AdaFair	8	8	-	-	-	-	1	5	1	6	5	7
	Ours	1	2	2	2	1	1	3	2	6	5	3	3
	Ours$_{ens}$	2	3	1	1	2	2	5	6	3	2	1	1
Bank (marital)	Reweight	4	5	-	-	-	-	7	7	7	7	6	7
	Reduction$_{DP}$	6	7	-	-	-	-	8	8	8	8	4	4
	Reduction$_{EO}$	7	8	-	-	-	-	5	6	4	5	7	8
	Threshold$_{DP}$	5	4	4	4	4	4	6	5	6	3	8	6
	Threshold$_{EO}$	1	3	2	3	3	3	3	1	5	6	5	5
	AdaFair	8	6	-	-	-	-	1	4	1	2	3	3
	Ours	2	1	3	2	1	1	4	3	2	1	2	2
	Ours$_{ens}$	3	2	1	1	2	2	2	2	3	4	1	1

Table 10: The difference between our approach and other fairness methods as well as the target model baselines for all combination of tasks and machine learning models we have benchmarked on. Our approach has a minimal impact on the learning utility with either a minor decrease in Acc. or even increase in BAcc., and meanwhile it consistently reduces both DP and EO. Results on the COMPAS dataset.

			Utility		**Fairness**	
			Acc.	**BAcc.**	**DP**	**EO**
COMPAS (sex)	**Ours**	**Vanilla**	$-0.008_{\pm 0.01}$	$-0.005_{\pm 0.01}$	$-0.144_{\pm 0.11}$	$-0.122_{\pm 0.12}$
		Reweight	$-0.005_{\pm 0.01}$	$-0.002_{\pm 0.01}$	$-0.106_{\pm 0.09}$	$-0.093_{\pm 0.12}$
		Reduction$_{DP}$	$-0.002_{\pm 0.01}$	$-0.001_{\pm 0.01}$	$-0.032_{\pm 0.04}$	$-0.004_{\pm 0.04}$
		Reduction$_{EO}$	$-0.004_{\pm 0.01}$	$-0.003_{\pm 0.01}$	$-0.043_{\pm 0.05}$	$-0.020_{\pm 0.05}$
		Threshold$_{DP}$	$0.008_{\pm 0.01}$	$0.008_{\pm 0.01}$	$0.015_{\pm 0.06}$	$-0.024_{\pm 0.08}$
		Threshold$_{EO}$	$0.025_{\pm 0.04}$	$0.032_{\pm 0.04}$	$-0.005_{\pm 0.07}$	$0.007_{\pm 0.07}$
		AdaFair	$0.013_{\pm 0.05}$	$0.016_{\pm 0.06}$	$-0.102_{\pm 0.07}$	$-0.080_{\pm 0.10}$
	Ours$_{ens}$	**Vanilla**	$0.000_{\pm 0.01}$	$0.002_{\pm 0.01}$	$-0.160_{\pm 0.11}$	$-0.154_{\pm 0.14}$
		Reweight	$0.002_{\pm 0.01}$	$0.006_{\pm 0.01}$	$-0.113_{\pm 0.08}$	$-0.106_{\pm 0.10}$
		Reduction$_{DP}$	$0.005_{\pm 0.01}$	$0.006_{\pm 0.01}$	$-0.039_{\pm 0.05}$	$-0.016_{\pm 0.04}$
		Reduction$_{EO}$	$0.003_{\pm 0.01}$	$0.005_{\pm 0.01}$	$-0.050_{\pm 0.05}$	$-0.033_{\pm 0.03}$
		Threshold$_{DP}$	$0.016_{\pm 0.01}$	$0.015_{\pm 0.01}$	$-0.001_{\pm 0.05}$	$-0.056_{\pm 0.06}$
		Threshold$_{EO}$	$0.032_{\pm 0.04}$	$0.039_{\pm 0.04}$	$-0.021_{\pm 0.05}$	$-0.026_{\pm 0.05}$
		AdaFair	$0.020_{\pm 0.05}$	$0.024_{\pm 0.06}$	$-0.110_{\pm 0.07}$	$-0.092_{\pm 0.09}$
COMPAS (race)	**Ours**	**Vanilla**	$-0.021_{\pm 0.02}$	$-0.015_{\pm 0.02}$	$-0.156_{\pm 0.09}$	$-0.148_{\pm 0.10}$
		Reweight	$-0.018_{\pm 0.02}$	$-0.011_{\pm 0.02}$	$-0.127_{\pm 0.07}$	$-0.116_{\pm 0.10}$
		Reduction$_{DP}$	$-0.003_{\pm 0.03}$	$0.004_{\pm 0.03}$	$-0.082_{\pm 0.03}$	$-0.067_{\pm 0.04}$
		Reduction$_{EO}$	$-0.010_{\pm 0.03}$	$-0.003_{\pm 0.03}$	$-0.089_{\pm 0.04}$	$-0.086_{\pm 0.04}$
		Threshold$_{DP}$	$0.005_{\pm 0.02}$	$0.005_{\pm 0.02}$	$0.016_{\pm 0.04}$	$-0.023_{\pm 0.04}$
		Threshold$_{EO}$	$0.000_{\pm 0.03}$	$0.007_{\pm 0.03}$	$-0.027_{\pm 0.06}$	$-0.013_{\pm 0.07}$
		AdaFair	$0.013_{\pm 0.07}$	$0.022_{\pm 0.08}$	$-0.092_{\pm 0.10}$	$-0.070_{\pm 0.09}$
	Ours$_{ens}$	**Vanilla**	$-0.013_{\pm 0.01}$	$-0.008_{\pm 0.01}$	$-0.156_{\pm 0.09}$	$-0.149_{\pm 0.11}$
		Reweight	$-0.008_{\pm 0.01}$	$-0.001_{\pm 0.01}$	$-0.124_{\pm 0.08}$	$-0.115_{\pm 0.10}$
		Reduction$_{DP}$	$0.007_{\pm 0.03}$	$0.014_{\pm 0.03}$	$-0.079_{\pm 0.03}$	$-0.066_{\pm 0.05}$
		Reduction$_{EO}$	$0.000_{\pm 0.03}$	$0.006_{\pm 0.03}$	$-0.086_{\pm 0.05}$	$-0.086_{\pm 0.06}$
		Threshold$_{DP}$	$0.013_{\pm 0.01}$	$0.012_{\pm 0.01}$	$0.016_{\pm 0.03}$	$-0.024_{\pm 0.04}$
		Threshold$_{EO}$	$0.008_{\pm 0.02}$	$0.014_{\pm 0.03}$	$-0.027_{\pm 0.05}$	$-0.014_{\pm 0.06}$
		AdaFair	$0.023_{\pm 0.07}$	$0.032_{\pm 0.08}$	$-0.089_{\pm 0.10}$	$-0.069_{\pm 0.10}$

Table 11: The difference between our approach and other fairness methods as well as the target model baselines for all combination of tasks and machine learning models we have benchmarked on. Our approach has a minimal impact on the learning utility with either a minor decrease in Acc. or even increase in BAcc., and meanwhile it consistently reduces both DP and EO. Results on the Adult dataset.

			Utility		Fairness	
			Acc.	**BAcc.**	**DP**	**EO**
Adult (sex)	**Ours**	**Vanilla**	-0.009$_{\pm 0.01}$	0.000$_{\pm 0.01}$	-0.056$_{\pm 0.01}$	-0.046$_{\pm 0.03}$
		Reweight	-0.002$_{\pm 0.02}$	0.035$_{\pm 0.06}$	-0.021$_{\pm 0.05}$	-0.027$_{\pm 0.02}$
		Reduction$_{DP}$	0.001$_{\pm 0.01}$	0.022$_{\pm 0.03}$	0.008$_{\pm 0.07}$	-0.073$_{\pm 0.12}$
		Reduction$_{EO}$	-0.009$_{\pm 0.01}$	-0.003$_{\pm 0.01}$	-0.056$_{\pm 0.02}$	-0.042$_{\pm 0.03}$
		Threshold$_{DP}$	0.044$_{\pm 0.01}$	0.032$_{\pm 0.01}$	0.124$_{\pm 0.02}$	-0.049$_{\pm 0.02}$
		Threshold$_{EO}$	0.006$_{\pm 0.02}$	0.023$_{\pm 0.02}$	-0.001$_{\pm 0.05}$	-0.002$_{\pm 0.04}$
		AdaFair	0.007$_{\pm 0.05}$	0.057$_{\pm 0.12}$	-0.015$_{\pm 0.09}$	-0.016$_{\pm 0.05}$
	Ours$_{ens}$	**Vanilla**	-0.005$_{\pm 0.00}$	0.005$_{\pm 0.00}$	-0.050$_{\pm 0.01}$	-0.041$_{\pm 0.04}$
		Reweight	0.004$_{\pm 0.02}$	0.041$_{\pm 0.05}$	-0.013$_{\pm 0.05}$	-0.019$_{\pm 0.02}$
		Reduction$_{DP}$	0.007$_{\pm 0.01}$	0.028$_{\pm 0.03}$	0.016$_{\pm 0.07}$	-0.066$_{\pm 0.12}$
		Reduction$_{EO}$	-0.003$_{\pm 0.00}$	0.004$_{\pm 0.01}$	-0.048$_{\pm 0.02}$	-0.034$_{\pm 0.03}$
		Threshold$_{DP}$	0.049$_{\pm 0.01}$	0.037$_{\pm 0.01}$	0.130$_{\pm 0.02}$	-0.043$_{\pm 0.02}$
		Threshold$_{EO}$	0.010$_{\pm 0.01}$	0.028$_{\pm 0.02}$	0.005$_{\pm 0.05}$	0.004$_{\pm 0.04}$
		AdaFair	0.013$_{\pm 0.05}$	0.064$_{\pm 0.12}$	-0.007$_{\pm 0.09}$	-0.009$_{\pm 0.04}$
Adult (race)	**Ours**	**Vanilla**	-0.001$_{\pm 0.00}$	0.003$_{\pm 0.01}$	-0.022$_{\pm 0.01}$	-0.034$_{\pm 0.03}$
		Reweight	0.007$_{\pm 0.02}$	0.037$_{\pm 0.05}$	0.004$_{\pm 0.01}$	-0.015$_{\pm 0.03}$
		Reduction$_{DP}$	-0.001$_{\pm 0.00}$	0.001$_{\pm 0.01}$	-0.003$_{\pm 0.02}$	-0.015$_{\pm 0.01}$
		Reduction$_{EO}$	-0.003$_{\pm 0.00}$	-0.005$_{\pm 0.01}$	-0.020$_{\pm 0.02}$	-0.016$_{\pm 0.02}$
		Threshold$_{DP}$	0.011$_{\pm 0.00}$	0.007$_{\pm 0.01}$	0.064$_{\pm 0.01}$	-0.037$_{\pm 0.02}$
		Threshold$_{EO}$	0.006$_{\pm 0.01}$	0.015$_{\pm 0.02}$	0.005$_{\pm 0.02}$	-0.015$_{\pm 0.03}$
		AdaFair	0.003$_{\pm 0.02}$	0.005$_{\pm 0.03}$	-0.034$_{\pm 0.01}$	-0.044$_{\pm 0.03}$
	Ours$_{ens}$	**Vanilla**	0.002$_{\pm 0.00}$	0.006$_{\pm 0.01}$	-0.022$_{\pm 0.01}$	-0.033$_{\pm 0.03}$
		Reweight	0.010$_{\pm 0.01}$	0.042$_{\pm 0.05}$	0.005$_{\pm 0.01}$	-0.014$_{\pm 0.03}$
		Reduction$_{DP}$	0.002$_{\pm 0.01}$	0.006$_{\pm 0.01}$	-0.003$_{\pm 0.02}$	-0.013$_{\pm 0.01}$
		Reduction$_{EO}$	0.001$_{\pm 0.00}$	0.000$_{\pm 0.01}$	-0.020$_{\pm 0.02}$	-0.015$_{\pm 0.02}$
		Threshold$_{DP}$	0.013$_{\pm 0.00}$	0.010$_{\pm 0.00}$	0.064$_{\pm 0.01}$	-0.036$_{\pm 0.02}$
		Threshold$_{EO}$	0.009$_{\pm 0.01}$	0.018$_{\pm 0.01}$	0.004$_{\pm 0.02}$	-0.014$_{\pm 0.03}$
		AdaFair	0.007$_{\pm 0.02}$	0.010$_{\pm 0.03}$	-0.034$_{\pm 0.01}$	-0.042$_{\pm 0.03}$

Table 12: The difference between our approach and other fairness methods as well as the target model baselines for all combination of tasks and machine learning models we have benchmarked on. Our approach has a minimal impact on the learning utility with either a minor decrease in Acc. or even increase in BAcc., and meanwhile it consistently reduces both DP and EO. Results on the LSA dataset.

			Utility		**Fairness**	
			Acc.	**BAcc.**	**DP**	**EO**
LSA (sex)	**Ours**	**Vanilla**	$-0.017_{\pm0.01}$	$0.003_{\pm0.01}$	$-0.065_{\pm0.01}$	$-0.104_{\pm0.04}$
		Reweight	$-0.018_{\pm0.02}$	$0.046_{\pm0.05}$	$-0.023_{\pm0.04}$	$-0.020_{\pm0.04}$
		Reduction$_{DP}$	$-0.015_{\pm0.01}$	$0.014_{\pm0.02}$	$0.022_{\pm0.02}$	$0.044_{\pm0.02}$
		Reduction$_{EO}$	$-0.023_{\pm0.01}$	$0.012_{\pm0.01}$	$0.010_{\pm0.02}$	$0.044_{\pm0.02}$
		Threshold$_{DP}$	$-0.008_{\pm0.02}$	$0.064_{\pm0.02}$	$0.035_{\pm0.02}$	$0.012_{\pm0.04}$
		Threshold$_{EO}$	$-0.004_{\pm0.02}$	$0.039_{\pm0.03}$	$0.010_{\pm0.03}$	$0.043_{\pm0.05}$
		AdaFair	$-0.006_{\pm0.02}$	$0.028_{\pm0.07}$	$-0.044_{\pm0.06}$	$-0.044_{\pm0.08}$
	Ours$_{ens}$	**Vanilla**	$-0.018_{\pm0.01}$	$0.004_{\pm0.01}$	$-0.062_{\pm0.02}$	$-0.103_{\pm0.03}$
		Reweight	$-0.020_{\pm0.03}$	$0.048_{\pm0.05}$	$-0.023_{\pm0.04}$	$-0.023_{\pm0.04}$
		Reduction$_{DP}$	$-0.018_{\pm0.01}$	$0.015_{\pm0.02}$	$0.021_{\pm0.02}$	$0.040_{\pm0.02}$
		Reduction$_{EO}$	$-0.025_{\pm0.02}$	$0.013_{\pm0.01}$	$0.010_{\pm0.02}$	$0.040_{\pm0.02}$
		Threshold$_{DP}$	$-0.009_{\pm0.02}$	$0.066_{\pm0.02}$	$0.038_{\pm0.03}$	$0.013_{\pm0.04}$
		Threshold$_{EO}$	$-0.005_{\pm0.02}$	$0.041_{\pm0.03}$	$0.012_{\pm0.04}$	$0.044_{\pm0.05}$
		AdaFair	$-0.008_{\pm0.02}$	$0.029_{\pm0.07}$	$-0.045_{\pm0.06}$	$-0.048_{\pm0.08}$
LSA (race)	**Ours**	**Vanilla**	$-0.010_{\pm0.01}$	$0.002_{\pm0.01}$	$-0.053_{\pm0.02}$	$-0.118_{\pm0.09}$
		Reweight	$-0.011_{\pm0.02}$	$0.045_{\pm0.05}$	$0.006_{\pm0.02}$	$0.013_{\pm0.04}$
		Reduction$_{DP}$	$-0.010_{\pm0.01}$	$0.005_{\pm0.01}$	$-0.019_{\pm0.03}$	$-0.041_{\pm0.08}$
		Reduction$_{EO}$	$0.000_{\pm0.01}$	$-0.006_{\pm0.03}$	$-0.077_{\pm0.06}$	$-0.125_{\pm0.07}$
		Threshold$_{DP}$	$0.010_{\pm0.01}$	$0.011_{\pm0.01}$	$0.033_{\pm0.01}$	$-0.060_{\pm0.07}$
		Threshold$_{EO}$	$0.003_{\pm0.02}$	$0.043_{\pm0.05}$	$-0.005_{\pm0.03}$	$-0.041_{\pm0.06}$
		AdaFair	$0.005_{\pm0.02}$	$0.024_{\pm0.07}$	$-0.056_{\pm0.07}$	$-0.071_{\pm0.08}$
	Ours$_{ens}$	**Vanilla**	$-0.006_{\pm0.00}$	$0.004_{\pm0.01}$	$-0.053_{\pm0.01}$	$-0.117_{\pm0.08}$
		Reweight	$-0.006_{\pm0.02}$	$0.048_{\pm0.05}$	$0.001_{\pm0.02}$	$0.009_{\pm0.04}$
		Reduction$_{DP}$	$-0.006_{\pm0.00}$	$0.008_{\pm0.01}$	$-0.024_{\pm0.03}$	$-0.045_{\pm0.08}$
		Reduction$_{EO}$	$0.004_{\pm0.01}$	$-0.003_{\pm0.03}$	$-0.082_{\pm0.06}$	$-0.129_{\pm0.06}$
		Threshold$_{DP}$	$0.013_{\pm0.01}$	$0.013_{\pm0.01}$	$0.033_{\pm0.02}$	$-0.059_{\pm0.07}$
		Threshold$_{EO}$	$0.006_{\pm0.01}$	$0.046_{\pm0.05}$	$-0.005_{\pm0.04}$	$-0.040_{\pm0.06}$
		AdaFair	$0.009_{\pm0.02}$	$0.027_{\pm0.07}$	$-0.061_{\pm0.07}$	$-0.074_{\pm0.08}$

Table 13: The difference between our approach and other fairness methods as well as the target model baselines for all combination of tasks and machine learning models we have benchmarked on. Our approach has a minimal impact on the learning utility with either a minor decrease in Acc. or even increase in BAcc., and meanwhile it consistently reduces both DP and EO. Results on the Bank dataset.

			Utility		Fairness	
			Acc.	**BAcc.**	**DP**	**EO**
Bank (age)	**Ours**	**Vanilla**	$-0.008_{\pm0.01}$	$0.037_{\pm0.01}$	$-0.023_{\pm0.03}$	$-0.036_{\pm0.09}$
		Reweight	$-0.002_{\pm0.01}$	$0.070_{\pm0.09}$	$-0.021_{\pm0.05}$	$-0.025_{\pm0.06}$
		Reduction$_{DP}$	$-0.009_{\pm0.01}$	$0.033_{\pm0.02}$	$0.019_{\pm0.03}$	$-0.017_{\pm0.07}$
		Reduction$_{EO}$	$-0.008_{\pm0.01}$	$0.034_{\pm0.01}$	$0.004_{\pm0.03}$	$-0.007_{\pm0.06}$
		Threshold$_{DP}$	$-0.007_{\pm0.01}$	$0.043_{\pm0.01}$	$0.054_{\pm0.04}$	$-0.107_{\pm0.07}$
		Threshold$_{EO}$	$0.002_{\pm0.01}$	$0.043_{\pm0.03}$	$0.005_{\pm0.02}$	$-0.011_{\pm0.06}$
		AdaFair	$-0.002_{\pm0.02}$	$0.164_{\pm0.10}$	$0.037_{\pm0.08}$	$0.020_{\pm0.07}$
	Ours$_{ens}$	**Vanilla**	$-0.002_{\pm0.00}$	$0.042_{\pm0.01}$	$-0.017_{\pm0.03}$	$-0.040_{\pm0.08}$
		Reweight	$0.007_{\pm0.01}$	$0.079_{\pm0.08}$	$-0.015_{\pm0.05}$	$-0.027_{\pm0.08}$
		Reduction$_{DP}$	$-0.001_{\pm0.00}$	$0.042_{\pm0.01}$	$0.025_{\pm0.03}$	$-0.019_{\pm0.07}$
		Reduction$_{EO}$	$0.000_{\pm0.00}$	$0.042_{\pm0.01}$	$0.010_{\pm0.03}$	$-0.009_{\pm0.04}$
		Threshold$_{DP}$	$-0.001_{\pm0.00}$	$0.048_{\pm0.01}$	$0.060_{\pm0.04}$	$-0.110_{\pm0.07}$
		Threshold$_{EO}$	$0.008_{\pm0.01}$	$0.048_{\pm0.02}$	$0.011_{\pm0.02}$	$-0.015_{\pm0.05}$
		AdaFair	$0.006_{\pm0.01}$	$0.173_{\pm0.10}$	$0.043_{\pm0.07}$	$0.018_{\pm0.08}$
Bank (marital)	**Ours**	**Vanilla**	$0.000_{\pm0.00}$	$0.006_{\pm0.01}$	$-0.008_{\pm0.01}$	$-0.006_{\pm0.03}$
		Reweight	$0.006_{\pm0.01}$	$0.044_{\pm0.08}$	$-0.005_{\pm0.01}$	$-0.012_{\pm0.03}$
		Reduction$_{DP}$	$0.000_{\pm0.00}$	$0.006_{\pm0.01}$	$-0.013_{\pm0.01}$	$-0.014_{\pm0.02}$
		Reduction$_{EO}$	$-0.002_{\pm0.00}$	$0.008_{\pm0.01}$	$-0.010_{\pm0.01}$	$-0.008_{\pm0.03}$
		Threshold$_{DP}$	$0.002_{\pm0.00}$	$0.025_{\pm0.01}$	$0.010_{\pm0.01}$	$-0.035_{\pm0.03}$
		Threshold$_{EO}$	$0.003_{\pm0.00}$	$0.011_{\pm0.01}$	$-0.002_{\pm0.01}$	$0.001_{\pm0.02}$
		AdaFair	$0.005_{\pm0.01}$	$0.149_{\pm0.10}$	$0.008_{\pm0.01}$	$0.011_{\pm0.03}$
	Ours$_{ens}$	**Vanilla**	$0.004_{\pm0.00}$	$0.010_{\pm0.01}$	$-0.007_{\pm0.01}$	$-0.006_{\pm0.03}$
		Reweight	$0.012_{\pm0.01}$	$0.050_{\pm0.08}$	$-0.005_{\pm0.01}$	$-0.011_{\pm0.03}$
		Reduction$_{DP}$	$0.005_{\pm0.00}$	$0.012_{\pm0.01}$	$-0.012_{\pm0.01}$	$-0.013_{\pm0.02}$
		Reduction$_{EO}$	$0.004_{\pm0.00}$	$0.014_{\pm0.01}$	$-0.010_{\pm0.01}$	$-0.007_{\pm0.03}$
		Threshold$_{DP}$	$0.006_{\pm0.00}$	$0.029_{\pm0.01}$	$0.010_{\pm0.01}$	$-0.035_{\pm0.03}$
		Threshold$_{EO}$	$0.007_{\pm0.00}$	$0.014_{\pm0.01}$	$-0.001_{\pm0.01}$	$0.000_{\pm0.02}$
		AdaFair	$0.011_{\pm0.01}$	$0.156_{\pm0.10}$	$0.008_{\pm0.01}$	$0.012_{\pm0.03}$