

Fair Classification with Noisy Protected Attributes

L. Elisa Celis
Yale University

Lingxiao Huang
Yale University

Nisheeth K. Vishnoi
Yale University

June 9, 2020

Abstract

Due to the growing deployment of classification algorithms in various social contexts, developing methods that are fair with respect to protected attributes such as gender or race is an important problem. However, the information about protected attributes in datasets may be inaccurate due to either issues with data collection or when the protected attributes used are themselves predicted by algorithms. Such inaccuracies can prevent existing fair classification algorithms from achieving desired fairness guarantees. Motivated by this, we study fair classification problems when the protected attributes in the data may be “noisy”. In particular, we consider a noise model where any protected type may be flipped to another with some fixed probability. We propose a “denoised” fair optimization formulation that can incorporate very general fairness goals via a set of constraints, mitigates the effects of such noise perturbations, and comes with provable guarantees. Empirically, we show that our framework can lead to near-perfect statistical parity with only a slight loss in accuracy for significant noise levels.

Contents

1	Introduction	1
1.1	Our contributions	1
1.2	Related works	2
2	Our model	2
3	Theoretical results	3
3.1	Denoised fairness constraints and denoised program	3
3.2	Main theorem: Performance of Program (DenoisedFair)	4
3.3	Algorithm for Program (DenoisedFair)	5
4	Proof of Theorem 3.3	5
4.1	Useful notations and useful facts for Theorem 3.3	6
4.2	Proof of Theorem 3.3	6
4.3	Proof of Lemma 4.5: Relation between Program (TargetFair) and (DenoisedFair)	7
4.4	Proof of Lemma 4.6: Bad classifiers are not feasible for Program (2)	9
5	Extention to general $p \geq 2$ and multiple fairness constraints	11
6	Empirical results	14
7	Conclusion, limitations, and future work	16
A	Discussion of initial attempts	19
A.1	Randomized labeling	19
A.2	Replacing S by \hat{S} in Program (TargetFair)	20
B	Other empirical results	26

1 Introduction

Fair classification has been a topic of intense study in ML due to its immense importance. Consequently, a host of fair classification algorithms have been proposed; see [4]. These algorithms crucially assume that one has access to the protected attributes (e.g., race, sex or gender) for training and/or deployment. However, data for protected attributes may be missing or contain inaccuracies.

Data collection is a complex process and may contain recording and reporting errors unintentional or otherwise [33]. Cleaning the data also requires making difficult and political decisions along the way, yet is often necessary especially when it comes to questions of race, gender, or identity [30]. Further, information about protected attributes may be missing entirely [12], something that has been recently brought into the public eye when attempting to measure COVID19 health disparities [2]. In such cases, protected attributes can be predicted from other data, however we know that this process is itself contains errors and biases [29, 7]. All of the above scenarios cause a significant problem for fair classification as existing approaches implicitly assume perfect protected attribute data and may not achieve the same performance on fairness metrics as they would if the data was perfect.

Thus, in order for fair classification techniques to be effective in the above mentioned cases, they must take protected attribute errors into consideration.

1.1 Our contributions

We introduce a flexible framework for fair classification that can mitigate the effect of inaccuracies – modeled as noisy perturbations – in the protected attributes. In particular, the framework we propose can handle:

- **multiple, non-binary** protected attributes;
- **arbitrary** “flipping” noise wherein each protected attribute is switched to another with some probability;
- **multiple** fairness metrics, including statistical parity, equalized odds, false discovery parity, and any formulation in the class of “linear-fractional” fairness constraints [10, Table 1].

Our method learns an approximate optimal fair classifier on the underlying dataset with high probability.

Theoretically, we formulate the fair classification problem with noisy protected attributes (Problem 1), whose goal is to solve the underlying fair program with certain fairness constraints (Program (TargetFair)) while the given dataset contains flipping noises (Definition 2.1). We propose a denoised fair program (Program (DenoisedFair)) whose optimizer f^Δ is provably approximately optimal on the underlying dataset (Theorem 3.3). The interpolation between fairness and accuracy by f^Δ is controlled by an input threshold $\tau \in [0, 1]$. Our program can be extended (Program (Gen-DenoisedFair)), that handles multiple, non-binary protected attributes, arbitrary flipping noises and multiple fairness constraints, with a provable guarantee (Theorem 5.8).

In order to deal with the presence of flipping noises, we develop the following novel technical ideas:

- We formulate denoised fairness constraints (Definition 3.1) that are used to estimate the underlying constraint in Program (TargetFair) (Lemma 4.5). Consequently, an optimal fair classifier f^* of Program (TargetFair) is likely to be feasible for Program (DenoisedFair), which ensures that f^Δ does not lose accuracy compared to f^* .
- For proving the fairness performance of f^Δ , we define a class of “bad classifiers” (Definition 4.1) and provide an upper bound of the probability that all bad classifiers are not feasible for Program (DenoisedFair) (Lemma 4.6), which is sufficient for proving f^Δ is fair (Theorem 3.3). The main idea is to upper bound certain “capacity” of bad classifiers (Definition 4.4), whose scale depends on the VC-dimension of classifiers (Definition 3.2).

We implement our program by logistic regression (Section 3.3) and examine our approach empirically on two benchmark fairness datasets: **Adult** and on the **Compas** (Section 6). We consider sex and race respectively (coded as binary for the purposes of the simulations) as the protected attribute and statistical parity as the fairness metric. The empirical results (Table 1) show that our framework can achieve higher fairness than existing fair classification algorithms **ZVRG** [38] and **GYF** [17], with a slight loss in accuracy.

For instance, for the **Compas** dataset with sex as the protected attribute, our framework can achieve statistical rate 0.88 while the highest rate achieved by **ZVRG** and **GYF** is only 0.61, with an accuracy of 58% (which is 8% less than that of **ZVRG** and **GYF**). By varying the input threshold τ , our framework can smoothly tradeoff between accuracy and fairness (Section B).

1.2 Related works

Fair classification. Fair classification has been well studied in recent years and a large body of work has focused on formulating the fair classification problem as a constrained optimization problem, e.g., constrained to statistical parity [38, 39, 28, 17, 10], or equalized odds [21, 37, 28, 10], and developing algorithms for it. Another class of algorithms for fair classification first learn an unconstrained optimal classifier and then shift the decision boundary according to the fairness requirement, e.g., [15, 21, 18, 32, 36, 14]. Interested readers can see a summary and comparisons of existing fair classification algorithms in [16, 3]. In contrast to this work, the assumption in all of these approaches is that the algorithm is given perfect information about the protected class.

Data correction. There is significant effort to correctly code and/or correct datasets to remove potential biases and inaccuracies. Cleaning raw data is a significant step in the pipeline, and efforts to correct for missing or inaccurately coded attributes have been studied in-depth for protected attributes, e.g., in the context of the census [30]. An alternate approach considers changing the composition of the dataset itself in order to correct for known biases in representation, and popular methods include re-labeling/re-weighting approach of [9, 24, 25], the repair methods of [19, 35], or optimization based methods such as [13]. In either case, the correction process while important is imperfect, and our work can help by starting with these improved yet imperfect datasets in order to build fair classifiers.

Classifiers robust to choice of datasets. Some recent studies have pointed to the brittleness of fair classification algorithms in the presence of noise in the selection of training dataset. For instance, [16] observed that fair classification algorithms may not be stable with respect to variations in the training dataset. Towards this, certain variance reduction or stability techniques have been introduced; see e.g., [23] who investigate how to achieve a fair classifier that is also stable with respect to variation in datasets. Applying their approach to our model, there is no guarantee that the learned classifier is fair over the underlying dataset.

Noise in labels. Recently, there have been some works [6, 5] that study fair classification when the label in the input dataset is noisy. The main difference of [6, 5] from our work is that they consider noisy labels instead of noisy protected attributes, which makes our denoised algorithms very different.

2 Our model

Let $\mathcal{D} = \mathcal{X} \times [p] \times \{0, 1\}$ be a domain. Each sample (X, Z, Y) is drawn from \mathcal{D} where \mathcal{X} is the feature space, $Z \in [p]$ represents a protected attribute,¹ and $Y \in \{0, 1\}$ is the label of (X, Z) that we want to predict. Usually, $\mathcal{X} \subseteq \mathbb{R}^d$ where d is the number of features for each sample.

Let $S = \{s_i = (x_i, z_i, y_i) \in \mathcal{D}\}_{i \in [N]}$ be an (underlying) dataset. Let $\mathcal{F} \subseteq \{0, 1\}^{\mathcal{X}}$ denote a family of all possible classifiers, i.e., the protected attribute Z is not used for prediction. Given a loss function $L(\cdot, \cdot)$ that takes a classifier f and a sample s as arguments, our goal is to learn a classifier $f \in \mathcal{F}$ that minimizes the empirical risk $\frac{1}{N} \sum_{i \in [N]} L(f, s_i)$ over S with certain fairness constraints.

$$\min_{f \in \mathcal{F}} \frac{1}{N} \sum_{i \in [N]} L(f, s_i) \quad \text{s.t.} \quad \Omega(f, S) \geq \tau. \quad (\text{TargetFair})$$

Here, $\Omega(\cdot) : \mathcal{F} \times \mathcal{D}^* \rightarrow \mathbb{R}$ is a function given explicitly for a specific fairness metric, e.g., statistical rate [38, 28, 17, 10], or false positive/negative rate [21, 37, 28, 10]. For instance, let D denote the empirical

¹The domain \mathcal{D} can be generalized to include multiple protected attributes Z_1, \dots, Z_m where $Z_i \in [p_i]$. We discuss a single protected attribute for simplicity. See Section 5 in S.M.

distribution over S and consider the statistical rate of a classifier f on S defined as

$$\gamma(f, S) := \frac{\min_{l \in [p]} \Pr_D [f = 1 \mid Z = l]}{\max_{l \in [p]} \Pr_D [f = 1 \mid Z = l]}. \quad (1)$$

Then an induced fairness constraint could be $\gamma(f, S) \geq 0.8$. Since $\gamma(f, S) \geq 0.8$ is non-convex, sometimes in the literature, one may consider a convex function $\Omega(f, S)$ as an estimate of $\gamma(f, S)$, e.g., $\Omega(f, S)$ is formulated as a covariance-type function in [38], and is formulated as the weighted sum of the logs of the empirical estimate of favorable bias in [17].

If S is observed, we can directly solve Program (TargetFair). However, as discussed earlier, the protected attributes in S may be imperfect and we may only observe a noisy dataset \hat{S} instead of S . We first introduce the noise model on protected attributes considered in this paper. Here, we present it for $p = 2$ and generalize it to non-binary protected attributes (Definition 5.2 in Section 5).

Definition 2.1 (Flipping noises for a binary protected attribute) *Suppose $p = 2$, i.e., $Z \in \{0, 1\}$. Let $\eta_0, \eta_1 \in (0, 0.5)$ be noise parameters. For each $i \in [N]$, we assume that the i th noisy sample $\hat{s}_i = (x_i, \hat{z}_i, y_i)$ is realized as follows:*

- If $z_i = 0$, then $\hat{z}_i = 0$ with probability $1 - \eta_0$ and $\hat{z}_i = 1$ with probability η_0 .
- If $z_i = 1$, then $\hat{z}_i = 0$ with probability η_1 and $\hat{z}_i = 1$ with probability $1 - \eta_1$.

This flipping noise has been considered in prior works [31, 6]. As η_0 or η_1 increase, the observed dataset \hat{S} becomes more noisy. Specifically, if $\eta_0 = \eta_1 = 0.5$, $\hat{Z} = 1$ holds with probability exactly 0.5 and it seems that we can not learn any information of Z from \hat{Z} .

We also make the following assumption on an optimal classifier f^* of Program (TargetFair).

Assumption 1 (Lower bound for the positive predictions of f^*) *There exists a constant $\lambda \in (0, 0.5)$ such that $\min \{\Pr_D [f^* = 1, Z = 0], \Pr_D [f^* = 1, Z = 1]\} \geq \lambda$.*

For instance, if there are 20% of samples with $Z = 0$ and $\Pr_D [f^* = 1 \mid Z = i] \geq 0.5$ ($i \in \{0, 1\}$), we have $\lambda \geq 0.1$. In practice, exact λ is unknown but we can set λ according to the context, e.g., λ can be set higher if $\min \{\Pr_D [Y = 1, Z = 0], \Pr_D [Y = 1, Z = 1]\}$ is large. Making this assumption is not strictly necessary, but the scale of λ affects the performance of our approaches; see Remark 3.4. Note that the term $f^* = 1$ in Assumption 1 is because we consider the statistical rate. We show how to extend Assumption 1 to handle general fairness constraints (Assumption 2 in Section 5).

We first take statistical rate with a binary protected attribute as an example and consider the following problem that aims to solve Program (TargetFair), albeit, with a given noisy dataset \hat{S} . We show how to extend the problem to multiple, non-binary protected attributes and multiple fairness constraints in Section 5.

Problem 1 (Fair classification w.r.t. statistical parity with noisy protected attributes) *Given a binary protected attribute ($p = 2$), a fairness constraint of the form $\gamma(f, S) \geq \tau$, a noisy dataset \hat{S} drawn from the underlying dataset S with flipping noise parameters $\eta_0, \eta_1 \in (0, 0.5)$, and $\lambda \in (0, 0.5)$ for which Assumption 1 holds, the goal is to learn an (approximately) optimal fair classifier $f \in \mathcal{F}$ of Program (TargetFair).*

3 Theoretical results

The main difficulty in solving Problem 1 is to satisfy the fairness constraints for S when S is unknown. Our key idea is to design new constraints over \hat{S} that estimate the underlying fairness constraints of Program (TargetFair). We first show how to design such constraints.

3.1 Denoised fairness constraints and denoised program

Let \hat{D} denote the empirical distribution over \hat{S} . For simplicity, let $\pi_{ij} := \Pr_{D, \hat{D}} [\hat{Z} = i \mid Z = j]$ for $i, j \in \{0, 1\}$, $\mu_i := \Pr_D [Z = i]$ and $\hat{\mu}_i := \Pr_{\hat{D}} [\hat{Z} = i]$ for $i \in \{0, 1\}$. If D and \hat{D} are clear from the context, we denote $\Pr_{D, \hat{D}} [\cdot]$ by $\Pr [\cdot]$.

Definition 3.1 (Denoised fairness constraints and denoised program) *Given a noisy dataset \widehat{S} and a classifier $f \in \{0, 1\}^{\mathcal{X}}$, let*

$$\Gamma_0(f) := \frac{(1 - \eta_1) \Pr[f = 1, \widehat{Z} = 0] - \eta_1 \Pr[f = 1, \widehat{Z} = 1]}{(1 - \eta_1)\widehat{\mu}_0 - \eta_1\widehat{\mu}_1}$$

and

$$\Gamma_1(f) := \frac{(1 - \eta_0) \Pr[f = 1, \widehat{Z} = 1] - \eta_0 \Pr[f = 1, \widehat{Z} = 0]}{(1 - \eta_0)\widehat{\mu}_1 - \eta_0\widehat{\mu}_0}.$$

We define the denoised statistical rate to be $\gamma^\Delta(f, \widehat{S}) := \min \left\{ \frac{\Gamma_0(f)}{\Gamma_1(f)}, \frac{\Gamma_1(f)}{\Gamma_0(f)} \right\}$, and define our denoised fairness constraints to be

$$\begin{cases} (1 - \eta_1) \Pr[f = 1, \widehat{Z} = 0] - \eta_1 \Pr[f = 1, \widehat{Z} = 1] \geq (1 - \eta_0 - \eta_1)\lambda - \delta, \\ (1 - \eta_0) \Pr[f = 1, \widehat{Z} = 1] - \eta_0 \Pr[f = 1, \widehat{Z} = 0] \geq (1 - \eta_0 - \eta_1)\lambda - \delta, \\ \gamma^\Delta(f, \widehat{S}) \geq \tau - \delta, \end{cases} \quad (2)$$

where $\delta \in (0, 1)$ is a fixed constant and τ is the desired lower bound on statistical rate. Our denoised program is as follows:

$$\min_{f \in \mathcal{F}} \frac{1}{N} \sum_{i \in [N]} L(f, \widehat{s}_i) \quad \text{s.t.} \quad \text{Constraints (2)}. \quad (\text{DenoisedFair})$$

The δ in Constraint (2) is used as a relaxation parameter depending on the context. Intuitively, $\Gamma_i(f)$ is designed to estimate $\Pr[f = 1 \mid Z = i]$ for $i \in \{0, 1\}$: its numerator approximates $(1 - \eta_0 - \eta_1) \Pr[f = 1, Z = i]$ and its denominator approximates $(1 - \eta_0 - \eta_1)\mu_i$. Take the denominator as an example; the intuition for the numerator is similar. The main observation is that for $i, j \in \{0, 1\}$, value $\Pr[\widehat{Z} = i \mid Z = j]$ can be well estimated by concentration bounds, e.g., $\Pr[\widehat{Z} = 1 \mid Z = 0] \approx \eta_0$. Then we can represent μ_i ($i \in \{0, 1\}$) by a linear combination of $\widehat{\mu}_0$ and $\widehat{\mu}_1$.

Due to how Γ_0 and Γ_1 are chosen, the first two constraints are designed to estimate the constraint $\min \{\Pr_D[f = 1, Z = 0], \Pr_D[f = 1, Z = 1]\} \geq \lambda$ due to Assumption 1, and the last constraint is designed to estimate $\gamma(f, S) \geq \tau$ by the definition of γ (Lemma 4.5).

3.2 Main theorem: Performance of Program (DenoisedFair)

Our main theorem shows that solving Program (DenoisedFair) finds an approximately fair classifier, which does not increase the empirical risk (compared to f^*) and only slightly violates the fairness constraint. To state our result, we need the following definition that measures the complexity of \mathcal{F} .

Definition 3.2 (VC-dimension of (S, \mathcal{F}) [20]) *Given a subset $A \subseteq [N]$, we define*

$$\mathcal{F}_A := \{\{i \in A : f(s_i) = 1\}\}_{f \in \mathcal{F}}$$

to be the collection of subsets of A that may be shattered by some $f \in \mathcal{F}$. The VC-dimension of (S, \mathcal{F}) is the largest integer t such that there exists a subset $A \subseteq [N]$ with $|A| = t$ and $|\mathcal{F}_A| = 2^t$.

Suppose $\mathcal{X} \subseteq \mathbb{R}^d$ for some integer $d \geq 1$. If $\mathcal{F} = \{0, 1\}^{\mathcal{X}}$, we observe that the VC-dimension is $t = N$. Several commonly used families \mathcal{F} have VC-dimension $O(d)$, including linear threshold functions [20], kernel SVM and gap tolerant classifiers [8]. VC-dimension affects the performance of Program (DenoisedFair); see discussion in Remark 3.4. The main theorem in this paper is as follows.

Theorem 3.3 (Performance of Program (DenoisedFair)) *Suppose the VC-dimension of (S, \mathcal{F}) is t . Given any parameters $\eta_0, \eta_1, \lambda \in (0, 0.5)$ and $\delta \in (0, 1)$, let $f^\Delta \in \mathcal{F}$ denote an optimal solution of Program (DenoisedFair). With probability at least $1 - O\left(e^{-\frac{(1-\eta_0-\eta_1)^2 \lambda^2 \delta^2 n}{5000}} + t \ln\left(\frac{50}{(1-\eta_0-\eta_1)\lambda\delta}\right)\right)$, we have*

- $\frac{1}{N} \sum_{i \in [N]} L(f^\Delta, s_i) \leq \frac{1}{N} \sum_{i \in [N]} L(f^*, s_i);$
- $\gamma(f^\Delta, S) \geq \tau - 3\delta.$

Remark 3.4 Observe that the success probability depends on $1 - \eta_0 - \eta_1$, δ , λ and the VC-dimension t of (S, \mathcal{F}) . If $1 - \eta_0 - \eta_1$ or δ is close to 0, i.e., the protected attributes are very noisy or there is no relaxation for $\gamma(f, S) \geq \tau$ respectively, the success probability guarantee naturally tends to be 0. Now we discuss the remaining parameters λ and t .

Discussion on λ . Intuitively, the success probability guarantee should tends to 0 when λ is close to 0. For instance, suppose there is only one sample s_1 with $f^*(s_1) = 0$ conditioned on $Z = 0$, i.e., $\Pr_D[f^* = 1, Z = 0] = 1/N$. To approximate f^* , we may need to label $f(s_1) = 1$. However, due to the flipping noises, $\hat{z}_1 = 1$ with probability η_0 and there are $\Omega(N)$ samples with noisy protected type $\hat{z}_i = 0$. It is likely that we can not find out the specific sample s_1 and label $f(s_1) = 1$, unless letting $f = 1$. However, $f = 1$ may lead to a large empirical risk (see discussion in Section A.1).

Discussion on t . The success probability also depends on t which captures the complexity of \mathcal{F} . Suppose $\mathcal{X} \subseteq \mathbb{R}^d$ for some integer $d \geq 1$. The worst case is $\mathcal{F} = \{0, 1\}^{\mathcal{X}}$ with $t = N$, which takes the success probability guarantee below 0. On the other hand, if the VC-dimension does not depend on N , e.g., only depends on $d \ll N$, the failure probability is exponentially small on N . For instance, if \mathcal{F} is the collection of all linear threshold functions, i.e., each classifier $f \in \mathcal{F}$ has the form $f(s_i) = \mathbf{I}[\langle x_i, \theta \rangle \geq r]$ for some vector $\theta \in \mathbb{R}^d$ and threshold $r \in \mathbb{R}$. We have $t \leq d + 1$ for an arbitrary dataset S [20].

3.3 Algorithm for Program (DenoisedFair)

As a use case, and used in our empirical results, we show how to solve Program (DenoisedFair) for logistic regression. Let $\mathcal{F}' = \{f'_\theta \mid \theta \in \mathbb{R}^d\}$ be the family of logistic regression classifiers where for each sample $s = (x, z, y)$, $f'_\theta(s) := \frac{1}{1 + e^{-\langle x, \theta \rangle}}$. We first learn a classifier $f'_\theta \in \mathcal{F}'$ and then round each $f'_\theta(\hat{s}_i)$ to $f_\theta(\hat{s}_i) := \mathbf{I}[f(\hat{s}_i) \geq 0.5]$. Consequently, $\Pr[f_\theta = 1, \hat{Z} = i] = \frac{1}{N} \sum_{i \in [N]: \hat{Z}=i} \mathbf{I}[\langle x_i, \theta \rangle \geq 0]$. Then let $\mu'_0 := (1 - \eta_1)\hat{\mu}_0 - \eta_1\hat{\mu}_1$ and $\mu'_1 := (1 - \eta_0)\hat{\mu}_1 - \eta_0\hat{\mu}_0$. Constraint (2) can be written as

$$\left\{ \begin{array}{l} \frac{1-\eta_1}{N} \sum_{i \in [N]: \hat{Z}=0} \mathbf{I}[\langle x_i, \theta \rangle \geq 0] - \frac{\eta_1}{N} \sum_{i \in [N]: \hat{Z}=1} \mathbf{I}[\langle x_i, \theta \rangle \geq 0] \geq (1 - \eta_0 - \eta_1)\lambda - \delta, \\ \frac{1-\eta_0}{N} \sum_{i \in [N]: \hat{Z}=1} \mathbf{I}[\langle x_i, \theta \rangle \geq 0] - \frac{\eta_0}{N} \sum_{i \in [N]: \hat{Z}=0} \mathbf{I}[\langle x_i, \theta \rangle \geq 0] \geq (1 - \eta_0 - \eta_1)\lambda - \delta, \\ ((\tau - \delta)\eta_0\mu'_0 + (1 - \eta_1)\mu'_1) \sum_{i \in [N]: \hat{Z}=0} \mathbf{I}[\langle x_i, \theta \rangle \geq 0] \\ \geq ((\tau - \delta)(1 - \eta_0)\mu'_0 + \eta_1\mu'_1) \sum_{i \in [N]: \hat{Z}=1} \mathbf{I}[\langle x_i, \theta \rangle \geq 0], \\ ((\tau - \delta)\eta_1\mu'_1 + (1 - \eta_0)\mu'_0) \sum_{i \in [N]: \hat{Z}=1} \mathbf{I}[\langle x_i, \theta \rangle \geq 0] \\ \geq ((\tau - \delta)(1 - \eta_1)\mu'_1 + \eta_0\mu'_0) \sum_{i \in [N]: \hat{Z}=0} \mathbf{I}[\langle x_i, \theta \rangle \geq 0]. \end{array} \right. \quad (3)$$

Now we propose the following program that minimizes the logistic loss.

$$\min_{\theta \in \mathbb{R}^d} -\frac{1}{N} \sum_{i \in [N]} (y_i \log f_\theta(s_i) + (1 - y_i) \log(1 - f_\theta(s_i))) \quad s.t. (3). \quad (\text{DenoisedLR})$$

Sometimes, we may append a regularization term $C \cdot \|\theta\|_2^2$ to the above loss function where $C \geq 0$ is a given regularization parameter. We can apply some constrained optimization packages to solve this program, e.g., SLSQP [26].

4 Proof of Theorem 3.3

The main idea is to verify a) f^* is a feasible solution of Program (DenoisedFair); b) Any “unfair” classifier $f \in \mathcal{F}$ violates Constraint (2). If both conditions hold, the empirical risk of f^Δ is guaranteed to be at most that of f^* and f^Δ must be fair over S (Theorem 3.3). The main difficulty is that there may be an infinite number of unfair classifiers and we need to show the probability that all these classifiers violate Constraint (2) is large, say close to 1.

4.1 Useful notations and useful facts for Theorem 3.3

We first define the collection of classifiers that are expected to violate Constraint (2).

Definition 4.1 (Bad classifiers) *Given a family $\mathcal{F} \subseteq \{0, 1\}^{\mathcal{X}}$, we call $f \in \mathcal{F}$ a bad classifier if f belongs to at least one of the following sub-families:*

- $\mathcal{G}_0 := \{f \in \mathcal{F} : \min \{\Pr[f = 1, Z = 0], \Pr[f = 1, Z = 1]\} < \frac{\lambda}{2}\};$
- Let $T = \lceil 232 \log \log \frac{2(\tau-3\delta)}{\lambda} \rceil$. For $i \in [T]$, define $\mathcal{G}_i := \left\{f \in \mathcal{F} \setminus \mathcal{G}_0 : \gamma(f, S) \in \left[\frac{\tau-3\delta}{1.01^{2^i+1}-1}, \frac{\tau-3\delta}{1.01^{2^i-1}-1}\right)\right\}$.

Intuitively, if $f \in \mathcal{G}_0$, it is likely that f violates the first or the second of Constraint (2); if $f \in \mathcal{G}_i$ for some $i \in [T]$, it is likely that $\gamma^\Delta(f, \hat{S}) < \tau - \delta$. Thus, any bad classifier is likely to violate Constraint (2) (Lemma 4.5). We still need to lower bound the total violating probability for all bad classifiers. Towards this, we need the following definition.

Definition 4.2 (ε -nets) *Given a family $\mathcal{F} \subseteq \{0, 1\}^{\mathcal{X}}$ of classifiers and $\varepsilon \in (0, 1)$, we say $F \subseteq \mathcal{F}$ is an ε -net of \mathcal{F} if for any $f, f' \in F$, $\Pr_D[f \neq f'] \geq \varepsilon$; and for any $f \in \mathcal{F}$, there exists $f' \in F$ such that $\Pr_D[f \neq f'] \leq \varepsilon$. We denote $M_\varepsilon(\mathcal{F})$ as the smallest size of an ε -net of \mathcal{F} .*

For instance, it follows from basic coding theory [27] that $M_\varepsilon(\{0, 1\}^{\mathcal{X}}) = \Omega(2^{N-O(\varepsilon N \log N)})$. The size of an ε -net is usually depends exponentially on the VC-dimension.

Theorem 4.3 (Relation between VC-dimension and ε -nets [22]) *Suppose the VC-dimension of (S, \mathcal{F}) is t . For any $\varepsilon \in (0, 1)$, $M_\varepsilon(\mathcal{F}) = O(\varepsilon^{-t})$.*

Next, we define the capacity of bad classifiers based on ε -nets.

Definition 4.4 (Capacity of bad classifiers) *Let $\varepsilon_0 = \frac{(1-\eta_0-\eta_1)\lambda-2\delta}{5}$. Let $\varepsilon_i = \frac{1.01^{2^i-1}\delta}{5}$ for $i \in [T]$ where $T = \lceil 232 \log \log \frac{2(\tau-3\delta)}{\lambda} \rceil$. Given a family $\mathcal{F} \subseteq \{0, 1\}^{\mathcal{X}}$, we denote the capacity of bad classifiers by*

$$\Phi(\mathcal{F}) := 2e^{-2\varepsilon_0^2 n} M_{\varepsilon_0}(\mathcal{G}_0) + 4 \sum_{i \in [T]} e^{-\frac{\varepsilon_i^2 (1-\eta_0-\eta_1)^2 \lambda^2 n}{200}} \cdot M_{\varepsilon_i (1-\eta_0-\eta_1)\lambda/10}(\mathcal{G}_i).$$

Actually, $\Phi(\mathcal{F})$ is shown to be an upper bound for the probability that there exists a bad classifier feasible for Program (DenoisedFair) (Lemma 4.6). Roughly, the factor $2e^{-2\varepsilon_0^2 n}$ is an upper bound of the probability that a bad classifier $f \in \mathcal{G}_0$ violates Constraint (2), and the factor $4e^{-\varepsilon_i^2 \lambda^2 \delta^2 n}$ is an upper bound of the probability that a bad classifier $f \in \mathcal{G}_i$ violates Constraint (2). Intuitively, we prove that if all bad classifiers in the nets of \mathcal{G}_i ($0 \leq i \leq T$) are not feasible for Program (DenoisedFair), all bad classifiers should violate Constraint (2) since they are close to some bad classifiers in the nets which leads to close Γ_0 and Γ_1 . Note that the scale of $\Phi(\mathcal{F})$ depends on the size of ε -nets of \mathcal{F} , which can be upper bounded by Theorem 4.3 and leads to the success probability of Theorem 3.3.

4.2 Proof of Theorem 3.3

We first present the following lemma. Its proof can be found in Section 4.3.

Lemma 4.5 (Relation between Program (TargetFair) and (DenoisedFair)) *Let $f \in \mathcal{F}$ be an arbitrary classifier and $\varepsilon \in (0, 0.5)$. With probability at least $1 - 2e^{-2\varepsilon^2 n}$, we have*

$$\begin{aligned} (1 - \eta_1) \Pr[f = 1, \hat{Z} = 0] - \eta_1 \Pr[f = 1, \hat{Z} = 1] &\in (1 - \eta_0 - \eta_1) \Pr[f = 1, Z = 0] \pm \varepsilon, \\ (1 - \eta_0) \Pr[f = 1, \hat{Z} = 1] - \eta_0 \Pr[f = 1, \hat{Z} = 0] &\in (1 - \eta_0 - \eta_1) \Pr[f = 1, Z = 1] \pm \varepsilon. \end{aligned}$$

Moreover, if $\min_{i \in \{0, 1\}} \Pr[f = 1, Z = i] \geq \frac{\lambda}{2}$, then with probability at least $1 - 4e^{-\frac{\varepsilon^2 (1-\eta_0-\eta_1)^2 \lambda^2 n}{200}}$,

$$\gamma^\Delta(f, \hat{S}) \in (1 \pm \varepsilon) \gamma(f, S).$$

The first part of this lemma shows how to estimate $\Pr[f = 1, Z = i]$ ($i \in \{0, 1\}$) in terms of $\Pr[f = 1, \widehat{Z} = 0]$ and $\Pr[f = 1, \widehat{Z} = 1]$, which motivates the first two constraints of Program (DenoisedFair). The second part of the lemma motivates the last constraint of Program (DenoisedFair). By Assumption 1, we have

$$\min\{\Pr[f^* = 1, Z = 0], \Pr[f^* = 1, Z = 1]\} \geq \lambda.$$

Hence, by the above lemma, f^* is likely to be feasible for Program (DenoisedFair). Consequently, f^Δ has empirical loss at most that of f^* . For the fairness performance, we need the following lemma that lower bounds the total probability that all bad classifiers violate Constraint (2), by the capacity of bad classifiers (Definition 4.4). The proof can be found in Section 4.4.

Lemma 4.6 (Bad classifiers are not feasible for Program (2)) *Assuming $\delta \in (0, 0.1\lambda)$, then with probability at least $1 - \Phi(\mathcal{F})$, any bad classifier violates Constraint (2). Assuming the VC-dimension of (S, \mathcal{F}) is t and $\delta \in (0, 1)$, then with probability at least $1 - O\left(e^{-\frac{(1-\eta_0-\eta_1)^2\lambda^2\delta^2n}{5000} + t \ln(\frac{50}{(1-\eta_0-\eta_1)\lambda\delta})}\right)$, any bad classifier violates Constraint (2).*

Proof: [Proof of Theorem 3.3] We first upper bound the probability that $\gamma^\Delta(f^\Delta, \widehat{S}) \geq \tau - 3\delta$. Let $\mathcal{F}_b = \{f \in \mathcal{F} : \gamma(f, S) < \tau - 3\delta\}$. If all classifiers in \mathcal{F}_b violate Constraint (2), we have that $\gamma^\Delta(f^\Delta, \widehat{S}) \geq \tau - 3\delta$. Note that if $\min_{i \in \{0, 1\}} \Pr[f = 1, Z = i] \geq \frac{\lambda}{2}$, then $\gamma(f, S) \geq \frac{\lambda}{2}$ holds by definition. Also, $\frac{\lambda - 3\delta}{1.01^{2T+1} - 1} \leq \frac{\lambda}{2}$. Thus, we conclude that $\mathcal{F}_b \subseteq \cup_{i=0}^T \mathcal{G}_i$. Then if all bad classifiers violate Constraint (2), we have $\gamma^\Delta(f^\Delta, \widehat{S}) \geq \tau - 3\delta$. By Lemma 4.6, $\gamma^\Delta(f^\Delta, \widehat{S}) \geq \tau - 3\delta$ holds with probability at least $1 - O\left(e^{-\frac{(1-\eta_0-\eta_1)^2\lambda^2\delta^2n}{5000} + t \ln(\frac{50}{(1-\eta_0-\eta_1)\lambda\delta})}\right)$.

Next, we upper bound the probability that f^* is feasible for Program (DenoisedFair), which implies $\frac{1}{N} \sum_{i \in [N]} L(f^\Delta, s_i) \leq \frac{1}{N} \sum_{i \in [N]} L(f^*, s_i)$. Letting $\varepsilon = \delta$ in Lemma 4.5, we have that with probability at least $1 - 2e^{-2\delta^2n} - 4e^{-\frac{(1-\eta_0-\eta_1)^2\lambda^2\delta^2n}{200}}$,

$$\begin{cases} (1 - \eta) \Pr[f^* = 1, \widehat{Z} = 0] - \eta \Pr[f^* = 1, \widehat{Z} = 1] \geq (1 - \eta_0 - \eta_1) \Pr[f^* = 1, Z = 0] - \delta, \\ (1 - \eta) \Pr[f^* = 1, \widehat{Z} = 1] - \eta \Pr[f^* = 1, \widehat{Z} = 0] \geq (1 - \eta_0 - \eta_1) \Pr[f^* = 1, Z = 1] - \delta, \\ \gamma^\Delta(f^*, \widehat{S}) \geq (1 - \delta)\gamma(f, S) \geq \gamma(f, S) - \delta. \end{cases}$$

It implies that f^* is feasible for Program (DenoisedFair) with probability at least $1 - 2e^{-2\varepsilon^2n} - 4e^{-\frac{(1-\eta_0-\eta_1)^2\lambda^2\delta^2n}{200}}$. This completes the proof. \square

4.3 Proof of Lemma 4.5: Relation between Program (TargetFair) and (DenoisedFair)

Proof: We first have the following simple observation.

Observation 4.7 1) $\mu_0 + \mu_1 = 1$, $\widehat{\mu}_0 + \widehat{\mu}_1 = 1$, and $\pi_{0,i} + \pi_{1,i} = 1$ holds for $i \in \{0, 1\}$; 2) For any $i, j \in \{0, 1\}$, $\Pr[Z = i \mid \widehat{Z} = j] = \frac{\pi_{ji}\mu_i}{\widehat{\mu}_j}$; 3) For any $i \in \{0, 1\}$, $\widehat{\mu}_i = \pi_{i,i}\mu_i + \pi_{i,1-i}\mu_{1-i}$.

Similar to Equation (35), we have

$$\begin{aligned} \Pr[f = 1, \widehat{Z} = 0] &= \Pr[\widehat{Z} = 0 \mid f = 1, Z = 0] \cdot \Pr[f = 1, Z = 0] \\ &\quad + \Pr[\widehat{Z} = 0 \mid f = 1, Z = 1] \cdot \Pr[f = 1, Z = 1]. \end{aligned} \tag{4}$$

Similar to the proof of Lemma A.5, by the Chernoff bound (additive form), both

$$\Pr[\widehat{Z} = 1 \mid f = 1, Z = 0] \in \eta_0 \pm \frac{\varepsilon}{2\Pr[f = 1, Z = 0]}, \tag{5}$$

and

$$\Pr \left[\widehat{Z} = 0 \mid f = 1, Z = 1 \right] \in \eta_1 \pm \frac{\varepsilon}{2 \Pr[f = 1, Z = 1]}, \quad (6)$$

hold with probability at least

$$1 - 2e^{-\frac{\varepsilon^2 n}{12\eta \Pr[f=1, Z=0]}} - 2e^{-\frac{\varepsilon^2 n}{12\eta \Pr[f=1, Z=1]}} \stackrel{\eta \leq 0.4}{\geq} 1 - 2e^{-2\varepsilon^2 n}.$$

Consequently, we have

$$\begin{aligned} & \Pr[f = 1, \widehat{Z} = 0] \\ = & \Pr[\widehat{Z} = 0 \mid f = 1, Z = 0] \cdot \Pr[f = 1, Z = 0] \\ & + \Pr[\widehat{Z} = 0 \mid f = 1, Z = 1] \cdot \Pr[f = 1, Z = 1] \quad (\text{Eq. (4)}) \\ \in & \left(1 - \eta_0 \pm \frac{\varepsilon}{2 \Pr[f = 1, Z = 0]}\right) \cdot \Pr[f = 1, Z = 0] \\ & + \left(\eta_1 \pm \frac{\varepsilon}{2 \Pr[f = 1, Z = 1]}\right) \cdot \Pr[f = 1, Z = 1] \quad (\text{Ineqs. (5) and (6)}) \\ \in & (1 - \eta_0) \Pr[f = 1, Z = 0] + \eta_1 \Pr[f = 1, Z = 1] \pm \varepsilon, \end{aligned} \quad (7)$$

and similarly,

$$\Pr[f = 1, \widehat{Z} = 1] \in \eta_0 \Pr[f = 1, Z = 0] + (1 - \eta_1) \Pr[f = 1, Z = 1] \pm \varepsilon. \quad (8)$$

By the above two inequalities, we conclude that

$$\begin{aligned} & (1 - \eta_1) \Pr[f = 1, \widehat{Z} = 0] - \eta_1 \Pr[f = 1, \widehat{Z} = 1] \\ \in & (1 - \eta_1) ((1 - \eta_0) \Pr[f = 1, Z = 0] + \eta_1 \Pr[f = 1, Z = 1] \pm \varepsilon) \\ & - \eta_1 (\eta_0 \Pr[f = 1, Z = 0] + (1 - \eta_1) \Pr[f = 1, Z = 1] \pm \varepsilon) \quad (\text{Ineqs. (7) and (8)}) \\ \in & (1 - \eta_0 - \eta_1) \Pr[f = 1, Z = 0] \pm \varepsilon. \end{aligned}$$

Similarly, we have

$$(1 - \eta_0) \Pr[f = 1, \widehat{Z} = 1] - \eta_0 \Pr[f = 1, \widehat{Z} = 0] \in (1 - \eta_0 - \eta_1) \Pr[f = 1, Z = 1] \pm \varepsilon.$$

This completes the proof of the first conclusion.

Next, we focus on the second conclusion. By assumption, $\min\{\Pr[f = 1, Z = 0], \Pr[f = 1, Z = 1]\} \geq \frac{\lambda}{2}$. Let $\varepsilon' = \frac{\varepsilon(1-\eta_0-\eta_1)\lambda}{20}$. By a similar argument as for the first conclusion, we have the following claim.

Claim 4.8 *With probability at least $1 - 4e^{-2(\varepsilon')^2 n}$, we have*

$$\left\{ \begin{array}{l} (1 - \eta_1) \Pr[f = 1, \widehat{Z} = 0] - \eta_1 \Pr[f = 1, \widehat{Z} = 1] \in (1 - \eta_0 - \eta_1) \Pr[f = 1, Z = 0] \pm \varepsilon', \\ (1 - \eta_0) \Pr[f = 1, \widehat{Z} = 1] - \eta_0 \Pr[f = 1, \widehat{Z} = 0] \in (1 - \eta_0 - \eta_1) \Pr[f = 1, Z = 1] \pm \varepsilon', \\ (1 - \eta_1)\widehat{\mu}_0 - \eta_1\widehat{\mu}_1 \in (1 - \eta_0 - \eta_1)\mu_0 \pm \varepsilon', \\ (1 - \eta_0)\widehat{\mu}_1 - \eta_0\widehat{\mu}_0 \in (1 - \eta_0 - \eta_1)\mu_1 \pm \varepsilon'. \end{array} \right.$$

Now we assume Claim 4.8 holds whose success probability is at least $1 - 4e^{-\frac{\varepsilon^2(1-\eta_0-\eta_1)^2\lambda^2 n}{200}}$ since $\varepsilon' = \frac{\varepsilon(1-\eta_0-\eta_1)\lambda}{20}$. Consequently, we have

$$\begin{aligned} & (1 - \eta_1) \Pr[f = 1, \widehat{Z} = 0] - \eta_1 \Pr[f = 1, \widehat{Z} = 1] \\ \geq & (1 - \eta_0 - \eta_1) \Pr[f = 1, Z = 0] - \varepsilon' \quad (\text{Claim 4.8}) \\ \geq & \frac{(1 - \eta_0 - \eta_1)\lambda}{2} - \varepsilon' \quad (\text{by assumption}) \\ \geq & 0.45 \cdot (1 - \eta_0 - \eta_1)\lambda. \quad (\eta \leq 0.4, \varepsilon' = \frac{\varepsilon(1 - \eta_0 - \eta_1)\lambda}{20}) \end{aligned} \quad (9)$$

Similarly, we can also argue that

$$(1 - \eta_1)\widehat{\mu}_0 - \eta_1\widehat{\mu}_1 \geq 0.45 \cdot (1 - \eta_0 - \eta_1)\lambda. \quad (10)$$

Then we have

$$\begin{aligned}
& \Pr[f = 1 \mid Z = 0] \\
= & \frac{\Pr[f = 1, Z = 0]}{\mu_0} \\
\in & \frac{(1 - \eta_1) \Pr[f = 1, \widehat{Z} = 0] - \eta_1 \Pr[f = 1, \widehat{Z} = 1] \pm \varepsilon'}{(1 - \eta_1)\widehat{\mu}_0 - \eta_1\widehat{\mu}_1 \pm \varepsilon'} \quad (\text{Claim 4.8}) \\
\in & \frac{(1 \pm \frac{\varepsilon'}{0.45 \cdot (1 - \eta_0 - \eta_1)\lambda}) \left((1 - \eta_1) \Pr[f = 1, \widehat{Z} = 0] - \eta_1 \Pr[f = 1, \widehat{Z} = 1] \right)}{(1 \pm \frac{\varepsilon'}{0.45 \cdot (1 - \eta_0 - \eta_1)\lambda}) ((1 - \eta_1)\widehat{\mu}_0 - \eta_1\widehat{\mu}_1)} \quad (11) \\
& (\text{Ineq. (9)}) \\
\in & (1 \pm \frac{\varepsilon}{9})^2 \cdot \Gamma_0(f). \quad (\text{Defns. of } \Gamma_0(f) \text{ and } \varepsilon')
\end{aligned}$$

Similarly, we can also prove that

$$\Pr[f = 1 \mid Z = 1] \in (1 \pm \frac{\varepsilon}{9})^2 \cdot \Gamma_1(f). \quad (12)$$

Overall, we have that with probability at least $1 - 4e^{-\frac{\varepsilon^2(1 - \eta_0 - \eta_1)^2\lambda^2n}{200}}$,

$$\begin{aligned}
\gamma^\Delta(f, S) &= \min \left\{ \frac{\Gamma_0(f)}{\Gamma_1(f)}, \frac{\Gamma_1(f)}{\Gamma_0(f)} \right\} \\
&\in (1 \pm \varepsilon) \cdot \min \left\{ \frac{\Pr[f = 1 \mid Z = 0]}{\Pr[f = 1 \mid Z = 1]}, \frac{\Pr[f = 1 \mid Z = 1]}{\Pr[f = 1 \mid Z = 0]} \right\} \quad (\text{Ineqs. (11) and (12)}) \\
&\in (1 \pm \varepsilon) \cdot \gamma(f, S).
\end{aligned}$$

Combining with Claim 4.8, we complete the proof of the second conclusion. \square

4.4 Proof of Lemma 4.6: Bad classifiers are not feasible for Program (2)

Proof: Suppose $\delta \in (0, 0.1\lambda)$. We discuss \mathcal{G}_0 and \mathcal{G}_i ($i \in [T]$) separately.

Bad classifiers in \mathcal{G}_0 . Let G_0 be an ε_0 -net of \mathcal{G}_0 of size $M_{\varepsilon_0}(\mathcal{G}_0)$. Consider an arbitrary classifier $g \in G_0$. By Lemma 4.5, with probability at least $1 - 2e^{-2\varepsilon_0^2n}$, we have

$$\begin{aligned}
& (1 - \eta_1) \Pr[g = 1, \widehat{Z} = 0] - \eta_1 \Pr[g = 1, \widehat{Z} = 1] \\
\leq & (1 - \eta_0 - \eta_1) \Pr[g = 1, Z = 0] + \varepsilon_0 \quad (13) \\
< & \frac{(1 - \eta_0 - \eta_1)\lambda}{2} + \varepsilon_0, \quad (\text{Defn. of } \mathcal{G}_0)
\end{aligned}$$

and

$$(1 - \eta_0) \Pr[g = 1, \widehat{Z} = 1] - \eta_0 \Pr[g = 1, \widehat{Z} = 0] < \frac{(1 - \eta_0 - \eta_1)\lambda}{2} + \varepsilon_0. \quad (14)$$

By the union bound, all classifiers $g \in G_0$ satisfy Inequalities (13) and (14) with probability at least $1 - 2e^{-2\varepsilon_0^2n}M_{\varepsilon_0}(\mathcal{G}_0)$. Suppose this event happens. We consider an arbitry classifier $f \in \mathcal{G}_0$. W.l.o.g., we assume $\Pr[f = 1, Z = 0] < \frac{\lambda}{2}$. By Definition 4.2, there must exist a classifier $g \in G_0$ such that $\Pr[f \neq g] \leq \varepsilon_0$. Then we have

$$(1 - \eta_1) \Pr[f = 1, \widehat{Z} = 0] - \eta_1 \Pr[f = 1, \widehat{Z} = 1]$$

$$\begin{aligned}
&\leq (1 - \eta_1)(\Pr[g = 1, \widehat{Z} = 0] + \varepsilon_0) - \eta_1(\Pr[g = 1, \widehat{Z} = 1] - \varepsilon_0) & (\Pr[f \neq g] \leq \varepsilon_0) \\
&\leq \frac{(1 - \eta_0 - \eta_1)\lambda}{2} + 2\varepsilon_0 & (\text{Ineq. (13)}) \\
&\leq \frac{(1 - \eta_0 - \eta_1)\lambda}{2} + \frac{(1 - \eta_0 - \eta_1)\lambda - 2\delta}{2} & (\text{Defn. of } \varepsilon_0) \\
&= (1 - \eta_0 - \eta_1)\lambda - \delta,
\end{aligned}$$

Thus, we conclude that all classifiers $f \in \mathcal{G}_0$ violate Constraint (2) with probability at least $1 - 2e^{-2\varepsilon_0^2 n} M_{\varepsilon_0}(\mathcal{G}_0)$.

Bad classifiers in \mathcal{G}_i for $i \in [T]$. We can assume that $\tau - 3\delta \geq \lambda/2$. Otherwise, all \mathcal{G}_i for $i \in [T]$ are empty, and hence, we complete the proof. Consider an arbitrary $i \in [T]$ and let G_i be an ε_i -net of \mathcal{G}_i of size $M_{\varepsilon_i(1-\eta_0-\eta_1)\lambda/10}(\mathcal{G}_i)$. Consider an arbitrary classifier $g \in G_i$. By the proof of Lemma 4.5, with probability at least $1 - 4e^{-\frac{\varepsilon_i^2(1-\eta_0-\eta_1)^2\lambda^2 n}{200}}$, we have

$$\left\{ \begin{array}{l} (1 - \eta_1) \Pr[g = 1, \widehat{Z} = 0] - \eta_1 \Pr[g = 1, \widehat{Z} = 1] \\ \in (1 - \eta_0 - \eta_1) \Pr[g = 1, Z = 0] \pm \frac{\varepsilon_i(1-\eta_0-\eta_1)\lambda}{20}, \\ (1 - \eta_0) \Pr[g = 1, \widehat{Z} = 1] - \eta_0 \Pr[g = 1, \widehat{Z} = 0] \\ \in (1 - \eta_0 - \eta_1) \Pr[g = 1, Z = 1] \pm \frac{\varepsilon_i(1-\eta_0-\eta_1)\lambda}{20}, \\ \gamma^\Delta(f, \widehat{S}) \in (1 \pm \varepsilon_i) \cdot \gamma(f, S). \end{array} \right. \quad (15)$$

Moreover, we have

$$\begin{aligned}
\gamma^\Delta(g, \widehat{S}) &\leq (1 + \varepsilon_i) \cdot \gamma(g, S) \\
&< (1 + \varepsilon_i) \cdot \frac{\tau - 3\delta}{1.01^{2^i - 1}}. \quad (\text{Defn. of } \mathcal{G}_i)
\end{aligned} \quad (16)$$

By the union bound, all classifiers $g \in G_i$ satisfy Inequality (16) with probability at least

$$1 - 4e^{-\frac{\varepsilon_i^2(1-\eta_0-\eta_1)^2\lambda^2 n}{200}} M_{\varepsilon_i(1-\eta_0-\eta_1)\lambda/10}(\mathcal{G}_i).$$

Suppose this event happens. We consider an arbitrary classifier $f \in \mathcal{G}_i$. By Definition 4.2, there must exist a classifier $g \in G_i$ such that $\Pr[f \neq g] \leq \varepsilon_i(1 - \eta_0 - \eta_1)\lambda/10$. By Inequality (15) and a similar argument as that for Inequality (9), we have

$$(1 - \eta_1) \Pr[g = 1, \widehat{Z} = 0] - \eta_1 \Pr[g = 1, \widehat{Z} = 1] \geq 0.45 \cdot (1 - \eta_0 - \eta_1)\lambda. \quad (17)$$

$$\begin{aligned}
&\Gamma_0(f) \\
&= \frac{(1 - \eta_1) \Pr[f = 1, \widehat{Z} = 0] - \eta_1 \Pr[f = 1, \widehat{Z} = 1]}{(1 - \eta)\widehat{\mu}_0 - \eta\widehat{\mu}_1} \\
&\in \frac{(1 - \eta_1) \left(\Pr[g = 1, \widehat{Z} = 0] \pm \frac{\varepsilon_i(1-\eta_0-\eta_1)\lambda}{10} \right) - \eta_1 \left(\Pr[g = 1, \widehat{Z} = 1] \pm \frac{\varepsilon_i(1-\eta_0-\eta_1)\lambda}{10} \right)}{(1 - \eta)\widehat{\mu}_0 - \eta\widehat{\mu}_1} \\
&\quad (\Pr[f \neq g] \leq \varepsilon_i(1 - \eta_0 - \eta_1)\lambda/10) \\
&\in \frac{(1 - \eta_1) \Pr[g = 1, \widehat{Z} = 0] - \eta_1 \Pr[g = 1, \widehat{Z} = 1] \pm \frac{\varepsilon_i(1-\eta_1-\eta_1)\lambda}{5}}{(1 - \eta)\widehat{\mu}_0 - \eta\widehat{\mu}_1} \quad (\eta \leq 0.4) \\
&\in (1 \pm 0.45\varepsilon_i) \cdot \frac{(1 - \eta) \Pr[g = 1, \widehat{Z} = 0] - \eta \Pr[g = 1, \widehat{Z} = 1]}{(1 - \eta)\widehat{\mu}_0 - \eta\widehat{\mu}_1} \quad (\text{Ineq. (17)}) \\
&\in (1 \pm 0.45\varepsilon_i) \cdot \Gamma_0(g).
\end{aligned} \quad (18)$$

Similarly, we can also prove

$$\Gamma_1(f) \in (1 \pm 0.45\varepsilon_i) \cdot \Gamma_1(g). \quad (19)$$

Thus, we conclude that

$$\begin{aligned} \gamma^\Delta(f, \hat{S}) &= \min \left\{ \frac{\Gamma_0(f)}{\Gamma_1(f)}, \frac{\Gamma_1(f)}{\Gamma_0(f)} \right\} \\ &\leq \frac{1 + 0.45\varepsilon_i}{1 - 0.45\varepsilon_i} \cdot \min \left\{ \frac{\Gamma_0(g)}{\Gamma_1(g)}, \frac{\Gamma_1(g)}{\Gamma_0(g)} \right\} \quad (\text{Ineqs. (18) and (19)}) \\ &< \frac{1 + 0.45\varepsilon_i}{1 - 0.45\varepsilon_i} \cdot (1 + \varepsilon_i) \cdot \frac{\tau - 3\delta}{1.01^{2i-1}} \quad (\text{Ineq. (16)}) \\ &\leq \frac{1 + 0.45\varepsilon_1}{1 - 0.45\varepsilon_1} \cdot (1 + \varepsilon_1) \cdot (\tau - 3\delta) \quad (\delta < 0.1\lambda) \\ &\leq \tau - \delta. \quad (\varepsilon_1 = \frac{1.01\delta}{5}) \end{aligned}$$

It implies that all classifiers $f \in \mathcal{G}_i$ violate Constraint (2) with probability at least

$$1 - 4e^{-\frac{\varepsilon_i^2(1-\eta_0-\eta_1)^2\lambda^2n}{200}} M_{\varepsilon_i(1-\eta_0-\eta_1)\lambda/10}(\mathcal{G}_i).$$

By the union bound, we complete the proof of Lemma 4.6 for $\delta \in (0, 0.1\lambda)$.

For general $\delta \in (0, 1)$, each bad classifier violates Constraint (2) with probability at most $4e^{-\frac{\varepsilon_1^2(1-\eta_0-\eta_1)^2\lambda^2n}{200}}$ by the above argument. By Definition 4.1, $|M_{\varepsilon_0}(\mathcal{G}_0)| + \sum_{i \in [T]} |M_{\varepsilon_i(1-\eta_0-\eta_1)\lambda/10}(\mathcal{G}_i)| \leq |M_{\varepsilon_1(1-\eta_0-\eta_1)\lambda/10}(\mathcal{F})|$. Then by the definition of $\Phi(\mathcal{F})$ and Theorem 4.3, the probability that there exists a bad classifier violating Constraint (2) is at most $\Phi(\mathcal{F}) = O\left(e^{-\frac{(1-\eta_0-\eta_1)^2\lambda^2\delta^2n}{5000} + t \ln(\frac{50}{(1-\eta_0-\eta_1)\lambda\delta})}\right)$. This completes the proof of Lemma 4.6. \square

5 Extention to general $p \geq 2$ and multiple fairness constraints

In this section, we show how to extend Problem 1 to multiple, non-binary protected attributes and multiple fairness constraints. We consider a general class of fairness defined in [10], based on the following definition.

Definition 5.1 (Linear-fractional/Linear group performance functions [10]) *Given a classifier $f \in \mathcal{F}$ and $i \in [p]$, we call $q_i(f)$ the group performance of $Z = i$ if $q_i(f) = \Pr[\xi(f) \mid \xi'(f), Z = i]$ for some events $\xi(f), \xi'(f)$ that might depend on the choice of f . Define a group performance function $q : \mathcal{F} \rightarrow [0, 1]^2$ for any classifier $f \in \mathcal{F}$ as $q(f) = (q_1(f), \dots, q_p(f))$.*

*Denote $\mathcal{Q}_{\text{linf}}$ to be the collection of all group performance functions. If ξ' does not depend on the choice of f , q is said to be **linear**. Denote $\mathcal{Q}_{\text{lin}} \subseteq \mathcal{Q}_{\text{linf}}$ to be the collection of all linear group performance functions.*

At a high level, a classifier f is considered to be fair w.r.t. to q if $q_1(f) \approx \dots \approx q_p(f)$. Definition 5.1 is general and contains many fairness metrics. For instance, if $\xi := (f = Y)$ and $\xi' := \emptyset$, we have $q_i(f) = \Pr[f = Y \mid Z = i]$ which is linear and called the accuracy rate. Also, if $\xi := (Y = 0)$ and $\xi' := (f = 1)$, we have $q_i(f) = \Pr[Y = 0 \mid f = 1, Z = i]$ which is linear-fractional and called the false discovery rate. More examples are summarized in [10, Table 1].

Given a group performance function q , we define Ω_q to be

$$\Omega_q(f, S) := \min_{i \in [p]} q_i(f) / \max_{i \in [p]} q_i(f).$$

Note that γ is a special case of the above definition where $q_i(f)$ is the statistical rate. Next, we extend the flipping noises to general $p \geq 2$.

Definition 5.2 (Flipping noises in general) Let $H \in [0, 1]^{p \times p}$ be a non-singular matrix satisfying that $\sum_{j \in [p]} H_{ij} = 1$ for any $i \in [p]$. For each $i \in [N]$, we assume the i th noisy sample is $\hat{s}_i = (x_i, \hat{z}_i, y_i)$ where $\hat{z}_i = j$ with probability $H_{z_i, j}$ for $j \in [p]$.

Note that H may not be symmetric, i.e., it is possible that $H_{ij} \neq H_{ji}$ for $i \neq j$. Also note that Definition 2.1 is a special case of Definition 5.2 by letting $H = \begin{bmatrix} 1 - \eta_0 & \eta_0 \\ \eta_1 & 1 - \eta_1 \end{bmatrix}$. According to the above definitions, we are ready to propose the following extension of Problem 1 to general $p \geq 2$ and multiple protected attributes.

Problem 2 (Fair classification with noisy protected attributes) Given m protected attributes, k group performance functions $q^{(1)}, \dots, q^{(k)}$ where each one is based on some protected attribute, a threshold vector $\tau \in [0, 1]^k$ and a noisy dataset \hat{S} with noise matrix H , the goal is to learn an (approximate) optimal fair classifier $f \in \mathcal{F}$ of the following program:

$$\begin{aligned} \min_{f \in \mathcal{F}} \frac{1}{N} \sum_{i \in [N]} L(f, s_i) \quad & \text{s.t.} \\ \Omega_{q^{(i)}}(f, S) \geq \tau_i, \quad & \forall i \in [k]. \end{aligned} \quad (\text{Gen-TargetFair})$$

We slightly abuse the notation by letting f^* denote an optimal fair classifier of Program (Gen-TargetFair). Similarly, we will design a denoised program for Problem 2. Note that we only need to show how to design denoised fairness constraints for an arbitrary group performance function q with respect to an arbitrary protected attribute, which can be naturally extended to multiple fairness constraints. Thus, we consider the case that $k = 1$ in the following, i.e., Program (TargetFair) for a given function q . Accordingly, Assumption 1 changes to the following.

Assumption 2 (Lower bound for events of f^*) Suppose there exists constant $\lambda \in (0, 0.5)$ such that $\min_{i \in [p]} \Pr[\xi(f^*), \xi'(f^*), Z = i] \geq \lambda$.

By definition, we know that for any $i \in [p]$,

$$q_i(f) = \frac{\Pr[\xi(f), \xi'(f), Z = i]}{\Pr[\xi'(f), Z = i]}.$$

As in Program (DenoisedFair), the main idea is to represent $\Pr[\xi(f), \xi'(f), Z = i]$ or $\Pr[\xi'(f), Z = i]$ by a linear combination of $\{\Pr[\xi(f), \xi'(f), Z = j]\}_{j \in [p]}$ or $\{\Pr[\xi'(f), Z = j]\}_{j \in [p]}$ respectively. For $\Pr[\xi'(f), Z = i]$, we only need to replace $f = 1$ in the argument of statistical rate by $\xi'(f)$. Similarly, we replace $f = 1$ by $(\xi(f), \xi'(f))$ for $\Pr[\xi(f), \xi'(f), Z = i]$.

Next, we provide the details for $\Pr[\xi'(f), Z = i]$ and the argument for $\Pr[\xi(f), \xi'(f), Z = i]$ is similar. Recall that $\pi_{ij} := \Pr[\hat{Z} = i \mid Z = j]$ for $i, j \in [p]$, $\mu_i := \Pr[Z = i]$ and $\hat{\mu}_i := \Pr[\hat{Z} = i]$ for $i \in [p]$. Similar to Equation (4), we have for each $i \in [p]$

$$\Pr[\xi'(f), \hat{Z} = i] = \sum_{j \in [p]} \Pr[\hat{Z} = i \mid \xi'(f), Z = j] \cdot \Pr[\xi'(f), Z = j]. \quad (20)$$

By Definition 5.2 and a similar argument as in the proof of Lemma 4.5, we have the following lemma.

Lemma 5.3 (Relation between $\Pr[\xi'(f), \hat{Z} = i]$ and $\Pr[\xi'(f), Z = j]$) Let $\varepsilon \in (0, 1)$ be a fixed constant. With probability at least $1 - 2pe^{-2\varepsilon^2 n}$, we have for each $i \in [p]$,

$$\Pr[\xi'(f), \hat{Z} = i] \in \sum_{j \in [p]} H_{ji} \cdot \Pr[\xi'(f), Z = j] \pm \varepsilon.$$

We define

$$w(f) := (\Pr[\xi'(f), Z = 1], \dots, \Pr[\xi'(f), Z = p]),$$

and

$$\widehat{w}(f) := \left(\Pr[\xi'(f), \widehat{Z} = 1], \dots, \Pr[\xi'(f), \widehat{Z} = p] \right).$$

Since H is non-singular, $(H^\top)^{-1}$ exists. Let $M := \max_{i \in [p]} \|(H^\top)^{-1}\|_1$ denote the maximum ℓ_1 -norm of a row of $(H^\top)^{-1}$. By Lemma 5.3, we directly obtain the following lemma.

Lemma 5.4 (Approximation of $\Pr[\xi'(f), Z = i]$) *With probability at least $1 - 2pe^{-2\varepsilon^2 n}$, for each $i \in [p]$,*

$$w(f)_i \in (H^\top)_i^{-1} \widehat{w}(f) \pm \varepsilon \|(H^\top)^{-1}\|_1 \in (H^\top)_i^{-1} \widehat{w}(f) \pm \varepsilon M.$$

Thus, we use $(H^\top)_i^{-1} \widehat{w}(f)$ to estimate $\Pr[\xi'(f), Z = i]$. Moreover, to estimate constraint

$$\min_{i \in [p]} \Pr[\xi(f), \xi'(f), Z = i] \geq \lambda,$$

we construct the following constraints:

$$(H^\top)^{-1} \widehat{w}(f) \geq (\lambda - \varepsilon M) \mathbf{1}. \quad (21)$$

Remark 5.5 *We show that the first two constraints of Program (DenoisedFair) is a special case of Constraint (21). Since $H = \begin{bmatrix} 1 - \eta_0 & \eta_0 \\ \eta_1 & 1 - \eta_1 \end{bmatrix}$ in Definition 2.1, we have*

$$(H^\top)^{-1} = \begin{bmatrix} \frac{1 - \eta_1}{1 - \eta_0 - \eta_1} & -\frac{\eta_1}{1 - \eta_0 - \eta_1} \\ -\frac{\eta_0}{1 - \eta_0 - \eta_1} & \frac{1 - \eta_0}{1 - \eta_0 - \eta_1} \end{bmatrix}.$$

Then $M = \frac{1}{1 - \eta_0 - \eta_1}$ and we can verify that the first two constraints of Program (DenoisedFair) are equivalent to Constraint (21) when $\xi'(f) = (f = 1)$.

Similarly, we define

$$u(f) := (\Pr[\xi(f), \xi'(f), Z = 1], \dots, \Pr[\xi(f), \xi'(f), Z = p]),$$

and

$$\widehat{u}(f) := \left(\Pr[\xi(f), \xi'(f), \widehat{Z} = 1], \dots, \Pr[\xi(f), \xi'(f), \widehat{Z} = p] \right).$$

Then we use $(H^\top)_i^{-1} \widehat{u}(f)$ to estimate $\Pr[\xi(f), \xi'(f), Z = i]$. Moreover, to estimate constraint

$$\min_{i \in [p]} \Pr[\xi(f), \xi(f), \xi'(f), Z = i] \geq \lambda,$$

we construct the following constraints:

$$(H^\top)^{-1} \widehat{u}(f) \geq (\lambda - \varepsilon M) \mathbf{1}. \quad (22)$$

Note that $\widehat{u}(f) \leq \widehat{w}(f)$ by definition. Inequality (22) is a sufficient condition for Inequality (21). Let $\delta \in (0, 1)$ be a given relaxed parameter. Now we are ready to define the corresponding denoised fair program.

$$\begin{aligned} & \min_{f \in \mathcal{F}} \frac{1}{N} \sum_{i \in [N]} L(f, \widehat{s}_i) \quad s.t. \\ & (H^\top)^{-1} \widehat{u}(f) \geq (\lambda - \delta) \mathbf{1}, \\ & \min_{i \in [p]} \frac{(H^\top)^{-1} \widehat{u}(f)}{(H^\top)^{-1} \widehat{w}(f)} \geq (\tau - \delta) \cdot \max_{i \in [p]} \frac{(H^\top)^{-1} \widehat{u}(f)}{(H^\top)^{-1} \widehat{w}(f)}. \end{aligned} \quad (\text{Gen-DenoisedFair})$$

Naturally, we define the following general notions of bad classifiers and the corresponding capacity.

Definition 5.6 (Bad classifiers in general) Given a family $\mathcal{F} \subseteq \{0, 1\}^{\mathcal{X}}$, we call $f \in \mathcal{F}$ a bad classifier if f belongs to at least one of the following sub-families:

- $\mathcal{G}_0 := \{f \in \mathcal{F} : \min_{i \in [p]} \Pr[\xi(f), \xi'(f), Z = i] < \frac{\lambda}{2}\};$
- Let $T = \lceil 232 \log \log \frac{2(\tau-3\delta)}{\lambda} \rceil$. For $i \in [T]$, define

$$\mathcal{G}_i := \left\{ f \in \mathcal{F} \setminus \mathcal{G}_0 : \Omega_q(f, S) \in \left[\frac{\tau - 3\delta}{1.01^{2^{i+1}-1}}, \frac{\tau - 3\delta}{1.01^{2^i-1}} \right] \right\}.$$

Note that Definition 4.1 is a special case of the above definition by letting $p = 2$, $M = 10$, $\xi(f) = (f = 1)$ and $\xi'(f) = \emptyset$. Based on Definition 5.6, we propose the following definition.

Definition 5.7 (Capacity of bad classifiers in general) Let $\varepsilon_0 = \frac{\lambda-2\delta}{5M}$. Let $\varepsilon_i = \frac{1.01^{2^i-1}\delta}{5}$ for $i \in [T]$ where $T = \lceil 232 \log \log \frac{2(\tau-3\delta)}{\lambda} \rceil$. Given a family $\mathcal{F} \subseteq \{0, 1\}^{\mathcal{X}}$, we denote the capacity of bad classifiers by

$$\Phi(\mathcal{F}) := 2pe^{-2\varepsilon_0^2 n} M_{\varepsilon_0}(\mathcal{G}_0) + 4p \sum_{i \in [T]} e^{-\frac{\varepsilon_i^2 \lambda^2 n}{200M^2}} \cdot M_{\varepsilon_i \lambda/10M}(\mathcal{G}_i).$$

By a similar argument as in Lemma 4.6, we can prove that $\Phi(\mathcal{F})$ is an upper bound of the probability that there exists a bad classifier feasible for Program (Gen-DenoisedFair). Consequently, we obtain the following theorem as an extension of Theorem 3.3.

Theorem 5.8 (Performance of Program (Gen-DenoisedFair)) Suppose the VC-dimension of (S, \mathcal{F}) is $t \geq 1$. Given any non-singular matrix $H \in [0, 1]^{p \times p}$ with $\sum_{j \in [p]} H_{ij} = 1$ for each $i \in [p]$, $\lambda \in (0, 0.5)$ and $\delta \in (0, 0.1\lambda)$, let $f^\Delta \in \mathcal{F}$ denote an optimal fair classifier of Program (Gen-DenoisedFair). With probability at least $1 - \Phi(\mathcal{F}) - 4pe^{-\frac{\lambda^2 \delta^2 n}{200M^2}}$, the following properties hold

- $\frac{1}{N} \sum_{i \in [N]} L(f^\Delta, s_i) \leq \frac{1}{N} \sum_{i \in [N]} L(f^*, s_i);$
- $\Omega_q(f^\Delta, S) \geq \tau - 3\delta.$

Specifically, if the VC-dimension of (S, \mathcal{F}) is t and $\delta \in (0, 1)$, the success probability is at least $1 - O(pe^{-\frac{\lambda^2 \delta^2 n}{5000M^2} + t \ln(50M/\lambda\delta)})$.

Proof: The proof is almost the same as in Theorem 3.3: we just need to replace $\frac{1}{1-\eta_0-\eta_1}$ by M everywhere. Note that the term $4pe^{-\frac{\lambda^2 \delta^2 n}{200M^2}}$ is an upper bound of the probability that f^* is not feasible for Program (Gen-DenoisedFair). The idea comes from Lemma 5.4 by letting $\varepsilon = \frac{\lambda\delta}{20M}$ such that for each $i \in [p]$,

$$w(f^*)_i \in (1 \pm \frac{\delta}{10})(H^\top)_i^{-1} \hat{w}(f^*) \text{ and } u(f^*)_i \in (1 \pm \frac{\delta}{10})(H^\top)_i^{-1} \hat{u}(f^*).$$

Consequently, $\frac{1}{N} \sum_{i \in [N]} L(f^\Delta, s_i) \leq \frac{1}{N} \sum_{i \in [N]} L(f^*, s_i)$. Since $\Phi(\mathcal{F})$ is an upper bound of the probability that there exists a bad classifier feasible for Program (Gen-DenoisedFair), we complete the proof. \square Finally, for multiple fairness constraints, the success probability of Theorem 5.8 changes to be

$$1 - O(kpe^{-\frac{\lambda^2 \delta^2 n}{5000M^2} + t \ln(50M/\lambda\delta)}).$$

Overall, we show how to solve Problem 2.

6 Empirical results

We implement our denoised algorithm and compare the performance with baseline algorithms on real datasets. The experiments are conducted on a 4-Core desktop CPU with 8GB RAM.

Table 1: performance of algorithms over testing datasets, including the average and standard deviation of accuracy and fairness measure γ . For **DenoisedLR**, we report the performances with parameter $\tau = 0.5, 0.7, 0.9$. The full accuracy-fairness tradeoffs when varying τ can be found in Section B.

	Adult				Compas			
	sex		race		sex		race	
	acc	γ	acc	γ	acc	γ	acc	γ
Unconstrained	.79	.65	.80	.67	.66	.52	.66	0.61
ZVRG	.79(.00)	.65(.02)	.80(.00)	.67(.04)	.66(.01)	.52(.03)	.66(.01)	.61(.04)
GYF	.79(.00)	.65(.02)	.80(.00)	.67(.04)	.66(.01)	.52(.03)	.66(.01)	.61(.04)
FairLR	.77(.02)	.74(.17)	.76(.01)	.86(.18)	.57(.04)	.78(.14)	.57(.04)	.81(.13)
DenoisedLR (.5)	.78(.01)	.73(.11)	.78(.02)	.72(.07)	.62(.04)	.68(.08)	.63(.02)	.75(.06)
DenoisedLR (.7)	.77(.01)	.67(.06)	.78(.02)	.79(.09)	.61(.03)	.71(.04)	.62(.04)	.79(.08)
DenoisedLR (.9)	.76(.01)	.85 (.12)	.76(.01)	.92 (.10)	.53(.02)	.88 (.07)	.58(.04)	.88 (.07)

Datasets. We do experiments on the **Adult** and **Compas** datasets [1], as pre-processed in [3]. The **Adult** dataset consists of 34189 individuals in the training dataset and 14653 individuals in the testing dataset, where each individual has 18 binary features and a label indicating whether the income is greater than 50k USD or not. The **Compas** dataset consists of 3694 individuals in the training dataset and 1684 individuals in the testing dataset, where each individual has 10 binary features and a label indicating whether the individual reoffends. We take sex and race (coded as binary in the processed data) to be the protected attributes whose value is 0 for minority groups. We consider the statistical rate γ over the testing dataset as defined in Equation (1).

Baseline and metrics. We evaluate the effectiveness of our algorithm **DenoisedLR** (Section 3.3). We compare against two state-of-the-art fair classification algorithms designed to ensure statistical parity **ZVRG** [38] and **GYF** [17], both of which have been shown to achieve better fairness than other algorithms, including on these datasets [16, 17]. Additionally, we implement the algorithm **FairLR** which minimizes logistic loss constrained to statistical parity over the given noisy dataset as described in Remark A.2 in Section A.2. Finally, we learn an unconstrained optimal classifier as a baseline.

Implementation details. Given a training dataset S , we generate a noisy dataset \hat{S} with, for illustrative purposes, $\eta_0 = 0.3$ and $\eta_1 = 0.1$ (i.e., the minority group is more likely to contain errors, as would be expected in various applications [30]). We consider other choices of η_0, η_1 in Section B. We run each algorithm on \hat{S} and vary the fairness constraints (e.g., the choice of $\tau \in [0.5, 0.95]$ in **FairLR** and **DenoisedLR**), learn the corresponding fair classifier, and report its accuracy (acc) and statistical rate (γ) over the test dataset. We perform 10 repetitions (i.e., generate the noisy dataset \hat{S} 10 times) and report the mean and standard deviation. We set $\lambda = 0$ for the **Adult** dataset as there are only 24% of samples with label $Y = 1$, and set $\lambda = 0.1$ for the **Compas** dataset as the ratio of samples with label $Y = 1$ is 47%.

Results. Table 1 summarizes the highest fairness metric γ achieved by baseline algorithms over the **Adult** and **Compas** testing datasets. Note that the state-of-the-art fairness algorithms **ZVRG** and **GYF** actually do not improve the fairness metric γ over an unconstrained classifier (i.e., one that seeks only to optimize accuracy with no regard to fairness). In other words, the errors in the protected class in our simulated dataset are enough to eliminate any benefit the fair classifiers would otherwise attain. In contrast, our approach can achieve much higher γ than baseline algorithms. The extent of this improvement varies with the strength of the constraint τ , but comes with a natural, if minimal, tradeoff with accuracy. The simple formulation **FairLR** can achieve higher γ than **ZVRG** and **GYF**, likely due to the fact that the fairness constraint in **FairLR** is directly $\gamma(f, \hat{S}) \geq \tau$ instead of relaxed constraints used in **ZVRG** and **GYF**. However, we observe that **DenoisedLR** can outperform **FairLR** with regard to γ given a large enough τ , and performs similarly with respect to accuracy. Moreover, **DenoisedLR** achieves γ close to or even larger than τ .

7 Conclusion, limitations, and future work

In this paper, we study fair classification with noisy protected attributes. We consider flipping noises and propose a unified framework that constructs an approximate optimal fair classifier over the underlying dataset for multiple, non-binary protected attributes and multiple fairness constraints. Empirically, our denoised algorithm matches our provable guarantees and can achieve higher fairness than existing fair classification algorithms with noisy input datasets.

Our work leaves several interesting futural directions. One direction is to consider other types of noises, e.g., the noisy protected type not only depends on the underlying type but may also depend on other features. There exist several works that design fair classifiers with noisy labels [6, 5]. Another interesting direction is to consider joint noises over both protected attributes and labels. Finally, it seems that our model relates to another setting in which each protected attribute follows from a known distribution. It is interesting to investigate whether our methods can be adapted to this setting.

References

- [1] A. Asuncion and D.J. Newman. UCI machine learning repository. archive.ics.uci.edu/ml/index.php, 2007. University of California, Irvine, School of Information and Computer Sciences.
- [2] Tony Barboza and Joseph Serna. As coronavirus deaths surge, missing racial data worry L.A. county officials. <https://www.latimes.com/california/story/2020-04-06/missing-racial-data-coronavirus-deaths-worries-los-angeles-county-officials>, 2020. Los Angeles Times.
- [3] Rachel K. E. Bellamy, Kuntal Dey, Michael Hind, Samuel C. Hoffman, Stephanie Houde, Kalapriya Kannan, Pranay Lohia, Jacquelyn Martino, Sameep Mehta, Aleksandra Mojsilovic, Seema Nagar, Karthikeyan Natesan Ramamurthy, John Richards, Diptikalyan Saha, Prasanna Sattigeri, Moninder Singh, Kush R. Varshney, and Yunfeng Zhang. AI Fairness 360: An extensible toolkit for detecting, understanding, and mitigating unwanted algorithmic bias, October 2018.
- [4] Rachel KE Bellamy, Kuntal Dey, Michael Hind, Samuel C Hoffman, Stephanie Houde, Kalapriya Kannan, Pranay Lohia, Jacquelyn Martino, Sameep Mehta, Aleksandra Mojsilovic, et al. AI fairness 360: An extensible toolkit for detecting, understanding, and mitigating unwanted algorithmic bias. *arXiv preprint arXiv:1810.01943*, 2018.
- [5] Arpita Biswas and Suvam Mukherjee. Ensuring fairness under prior probability shifts. *ArXiv*, abs/2005.03474, 2020.
- [6] Avrim Blum and Kevin Stangl. Recovering from biased data: Can fairness constraints improve accuracy? *Symposium on the foundations of responsible computing, (FORC) 2020*, 2020.
- [7] Joy Buolamwini and Timnit Gebru. Gender shades: Intersectional accuracy disparities in commercial gender classification. In *Conference on Fairness, Accountability and Transparency, FAT 2018, 23-24 February 2018, New York, NY, USA*, pages 77–91, 2018.
- [8] Christopher J. C. Burges. A tutorial on support vector machines for pattern recognition. *Data Mining and Knowledge Discovery*, 2(2):121–167, 1998.
- [9] Toon Calders, Faisal Kamiran, and Mykola Pechenizkiy. Building classifiers with independency constraints. *2009 IEEE International Conference on Data Mining Workshops*, pages 13–18, 2009.
- [10] L. Elisa Celis, Lingxiao Huang, Vijay Keswani, and Nisheeth K. Vishnoi. Classification with fairness constraints: A meta-algorithm with provable guarantees. In *Proceedings of the Conference on Fairness, Accountability, and Transparency, FAT* 2019, Atlanta, GA, USA, January 29-31, 2019*, pages 319–328, 2019.

- [11] Stevie Chancellor, Shion Guha, Jofish Kaye, Jen King, Niloufar Salehi, Sarita Schoenebeck, and Elizabeth Stowell. The relationships between data, power, and justice in csw research. In *Conference Companion Publication of the 2019 on Computer Supported Cooperative Work and Social Computing*, pages 102–105, 2019.
- [12] Ethnicity Data, Michele Ver Ploeg, and Edward Perrin. Eliminating health disparities: Measurement and data needs. *Washington (DC): National Academies Press (US)*, 2004.
- [13] Flávio du Pin Calmon, Dennis Wei, Bhanukiran Vinzamuri, Karthikeyan Natesan Ramamurthy, and Kush R. Varshney. Optimized pre-processing for discrimination prevention. In *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, 4-9 December 2017, Long Beach, CA, USA*, pages 3992–4001, 2017.
- [14] Cynthia Dwork, Nicole Immorlica, Adam Tauman Kalai, and Mark D. M. Leiserson. Decoupled classifiers for group-fair and efficient machine learning. In *Fairness, Accountability, and Transparency in Machine Learning*, pages 119–133, 2018.
- [15] Benjamin Fish, Jeremy Kun, and Ádám D. Lelkes. A confidence-based approach for balancing fairness and accuracy. In *Proceedings of the 2016 SIAM International Conference on Data Mining, 2016*, pages 144–152. SIAM, 2016.
- [16] Sorelle A. Friedler, Carlos Scheidegger, Suresh Venkatasubramanian, Sonam Choudhary, Evan P. Hamilton, and Derek Roth. A comparative study of fairness-enhancing interventions in machine learning. In *Fairness, Accountability, and Transparency in Machine Learning*, 2019.
- [17] Naman Goel, Mohammad Yaghini, and Boi Faltings. Non-discriminatory machine learning through convex fairness criteria. In *Proceedings of the Thirty-Second AAAI Conference on Artificial Intelligence*, 2018.
- [18] Gabriel Goh, Andrew Cotter, Maya R. Gupta, and Michael P. Friedlander. Satisfying real-world goals with dataset constraints. In *Advances in Neural Information Processing Systems 29: Annual Conference on Neural Information Processing Systems*, pages 2415–2423, 2016.
- [19] Paula Gordaliza, Eustasio del Barrio, Fabrice Gamboa, and Jean-Michel Loubes. Obtaining fairness using optimal transport theory. In *Proceedings of the 36th International Conference on Machine Learning, ICML 2019, 9-15 June 2019, Long Beach, California, USA*, pages 2357–2365, 2019.
- [20] Sarel Har-peled. *Geometric Approximation Algorithms*. American Mathematical Society, USA, 2011.
- [21] Moritz Hardt, Eric Price, and Nati Srebro. Equality of opportunity in supervised learning. In *Advances in Neural Information Processing Systems 29: Annual Conference on Neural Information Processing*, pages 3315–3323, 2016.
- [22] David Haussler. Sphere packing numbers for subsets of the Boolean n -cube with bounded Vapnik-Chervonenkis dimension. *Journal of Combinatorial Theory, Series A*, 69(2):217–232, 1995.
- [23] Lingxiao Huang and Nisheeth K. Vishnoi. Stable and fair classification. In *Proceedings of the 36th International Conference on Machine Learning, ICML 2019, 9-15 June 2019, Long Beach, California, USA*, pages 2879–2890, 2019.
- [24] Faisal Kamiran and Toon Calders. Classifying without discriminating. In *Computer, Control and Communication, 2009. IC4 2009. 2nd International Conference on*, pages 1–6. IEEE, 2009.
- [25] Faisal Kamiran and Toon Calders. Data preprocessing techniques for classification without discrimination. *Knowledge and Information Systems*, 33(1):1–33, 2012.
- [26] D. Kraft. *A Software Package for Sequential Quadratic Programming*. Deutsche Forschungs- und Versuchsanstalt für Luft- und Raumfahrt Köln: Forschungsbericht. Wiss. Berichtswesen d. DFVLR, 1988.

- [27] J. H. Van Lint. *Introduction to Coding Theory*. Springer-Verlag, Berlin, Heidelberg, 3rd edition, 1998.
- [28] Aditya Krishna Menon and Robert C. Williamson. The cost of fairness in binary classification. In *Conference on Fairness, Accountability and Transparency, FAT*, pages 107–118, 2018.
- [29] Vidya Muthukumar, Tejaswini Pedapati, Nalini K. Ratha, Prasanna Sattigeri, Chai-Wah Wu, Brian Kingsbury, Abhishek Kumar, Samuel Thomas, Aleksandra Mojsilovic, and Kush R. Varshney. Understanding unequal gender classification accuracy from face images. *CoRR*, abs/1812.00099, 2018.
- [30] Melissa Nobles. Shades of citizenship: Race and the census in modern politics. *Bibliovault OAI Repository, the University of Chicago Press*, 2000.
- [31] Ryan O’Donnell. *Analysis of Boolean Functions*. Cambridge University Press, 2014.
- [32] Geoff Pleiss, Manish Raghavan, Felix Wu, Jon M. Kleinberg, and Kilian Q. Weinberger. On fairness and calibration. In *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems*, pages 5684–5693, 2017.
- [33] Jose A Saez, Mikel Galar, Julian Luengo, and Francisco Herrera. Tackling the problem of classification with noisy data using multiple classifier systems: Analysis of the performance and robustness. *Information Sciences*, 247:1–20, 2013.
- [34] Hannah Sassaman, Jennifer Lee, Jenessa Irvine, and Shankar Narayan. Creating community-based tech policy: case studies, lessons learned, and what technologists and communities can do together. In *Proceedings of the 2020 Conference on Fairness, Accountability, and Transparency*, pages 685–685, 2020.
- [35] Hao Wang, Berk Ustun, and Flávio P. Calmon. Repairing without retraining: Avoiding disparate impact with counterfactual distributions. In *Proceedings of the 36th International Conference on Machine Learning, ICML 2019, 9-15 June 2019, Long Beach, California, USA*, pages 6618–6627, 2019.
- [36] Blake E. Woodworth, Suriya Gunasekar, Mesrob I. Ohannessian, and Nathan Srebro. Learning non-discriminatory predictors. In *Proceedings of the 30th Conference on Learning Theory, COLT 2017, Amsterdam, The Netherlands, 7-10 July 2017*, pages 1920–1953, 2017.
- [37] Muhammad Bilal Zafar, Isabel Valera, Manuel Gomez-Rodriguez, and Krishna P. Gummadi. Fairness beyond disparate treatment & disparate impact: Learning classification without disparate mistreatment. In *Proceedings of the 26th International Conference on World Wide Web, WWW 2017*, pages 1171–1180, 2017.
- [38] Muhammad Bilal Zafar, Isabel Valera, Manuel Gomez-Rodriguez, and Krishna P. Gummadi. Fairness constraints: Mechanisms for fair classification. In *Proceedings of the 20th International Conference on Artificial Intelligence and Statistics, AISTATS 2017, 20-22 April 2017, Fort Lauderdale, FL, USA*, pages 962–970, 2017.
- [39] Brian Hu Zhang, Blake Lemoine, and Margaret Mitchell. Mitigating unwanted biases with adversarial learning. In *Proceedings of the 2018 AAAI/ACM Conference on AI, Ethics, and Society*, pages 335–340, 2018.

A Discussion of initial attempts

We first discuss two natural ideas including randomized labeling (Section A.1) and solving Program (ConFair) that only depends on \hat{S} (Section A.2). We also discuss their weakness on either the empirical loss or the fairness constraints.

A.1 Randomized labeling

A simple idea is that for each sample $s_i \in S$, i.i.d. draw the label $f(s_i)$ to be 0 with probability α and to be 1 with probability $1 - \alpha$ ($\alpha \in [0, 1]$). This simple idea leads to a fair classifier by the following lemma.

Lemma A.1 (A random classifier is fair) *Let $f \in \{0, 1\}^{\mathcal{X}}$ be a classifier generated by randomized labeling. With probability at least $1 - 2e^{-\frac{\alpha\lambda N}{1.2 \times 10^5}}$, $\gamma(f, S) \geq 0.99$.*

Proof: Let $A = \{i \in [N] : z_i = 0\}$ be the collection of samples with $z_i = 0$. By Assumption 1, we know that $|A| \geq \lambda N$. For $i \in A$, let X_i be the random variable where $X_i = f(s_i)$. By randomized labeling, we know that $\Pr[X_i = 1] = \alpha$. Also,

$$\Pr[f = 1 \mid Z = 0] = \frac{\sum_{i \in A} X_i}{|A|}. \quad (23)$$

Since all X_i ($i \in A$) are independent, we have

$$\begin{aligned} \Pr \left[\sum_{i \in A} X_i \in (1 \pm 0.005) \cdot \alpha |A| \right] &\geq 1 - 2e^{-\frac{0.005^2 \alpha |A|}{3}} \quad (\text{Chernoff bound}) \\ &\geq 1 - 2e^{-\frac{\alpha \lambda N}{1.2 \times 10^5}}. \quad (|A| \geq \lambda N) \end{aligned} \quad (24)$$

Thus, with probability at least $1 - 2e^{-\frac{\alpha \lambda N}{1.2 \times 10^5}}$,

$$\begin{aligned} \Pr[f = 1 \mid Z = 0] &= \frac{\sum_{i \in A} X_i}{|A|} \quad (\text{Eq. (23)}) \\ &\in (1 \pm 0.005) \cdot \frac{\alpha |A|}{|A|} \quad (\text{Ineq. (24)}) \\ &\in (1 \pm 0.005)\alpha. \end{aligned}$$

Similarly, we have that with probability at least $1 - 2e^{-\frac{\alpha \lambda N}{1.2 \times 10^5}}$,

$$\Pr[f = 1 \mid Z = 1] \in (1 \pm 0.005)\alpha.$$

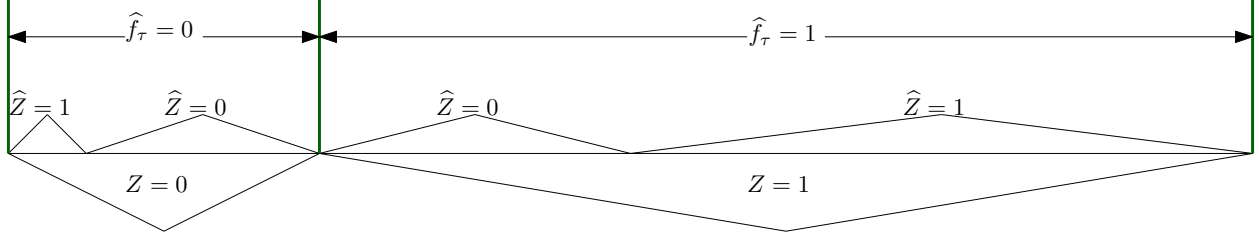
By the definition of $\gamma(f, S)$, we complete the proof. \square

However, there is no guarantee for the empirical risk of randomized labeling. For instance, consider the loss function $L(f, s) := \mathbf{I}[f(s) = y]$ where $\mathbf{I}[\cdot]$ is the indicator function, and suppose there are $\frac{N}{2}$ samples with $y_i = 0$. In this setting, the empirical risk of f^* may be close to 0, e.g., $f^* = Y$. Meanwhile, the expected empirical risk of randomized labeling is

$$\frac{1}{N} \left((1 - \alpha) \cdot \frac{N}{2} + \alpha \cdot \frac{N}{2} \right) = \frac{1}{2},$$

which is much larger than that of f^* .

Figure 1: An example showing that $\gamma(f, S)$ and $\gamma(f, \hat{S})$ can differ by a lot. The detailed explanation can be found in Example A.3.



A.2 Replacing S by \hat{S} in Program (TargetFair)

Another idea is to solve the following program which only depends on \hat{S} , i.e., simply replacing S by \hat{S} in Program (TargetFair).

$$\begin{aligned} \min_{f \in \mathcal{F}} \frac{1}{N} \sum_{i \in [N]} L(f, \hat{s}_i) \quad s.t. \\ \gamma(f, \hat{S}) \geq \tau. \end{aligned} \quad (\text{ConFair})$$

Remark A.2 Similar to Section 3.3, we can design an algorithm that solves Program (ConFair) by logistic regression.

$$\begin{aligned} \min_{\theta \in \mathbb{R}^d} -\frac{1}{N} \sum_{i \in [N]} (y_i \log f_{\theta}(s_i) + (1 - y_i) \log(1 - f_{\theta}(s_i))) \quad s.t. \\ \hat{\mu}_1 \cdot \sum_{i \in [N]: \hat{Z}=0} \mathbf{I}[\langle x_i, \theta \rangle \geq 0] \geq \tau \hat{\mu}_0 \cdot \sum_{i \in [N]: \hat{Z}=1} \mathbf{I}[\langle x_i, \theta \rangle \geq 0], \\ \hat{\mu}_0 \cdot \sum_{i \in [N]: \hat{Z}=1} \mathbf{I}[\langle x_i, \theta \rangle \geq 0] \geq \tau \hat{\mu}_1 \cdot \sum_{i \in [N]: \hat{Z}=0} \mathbf{I}[\langle x_i, \theta \rangle \geq 0]. \end{aligned} \quad (\text{FairLR})$$

Let \hat{f}^* denote an optimal solution of Program (ConFair). Ideally, we want to use \hat{f}^* to estimate f^* . Since Z is not used for prediction, we have that for any $f \in \mathcal{F}$,

$$\sum_{i \in [N]} L(f, s_i) = \sum_{i \in [N]} L(f, \hat{s}_i).$$

Then if \hat{f}^* satisfies $\gamma(\hat{f}^*, S) \geq \tau$, we conclude that \hat{f}^* is also an optimal solution of Program (TargetFair). However, due to the flipping noises, \hat{f}^* may be far from f^* (Example A.3). More concretely, it is possible that $\gamma(\hat{f}^*, S) \ll \tau$ (Lemma A.4). Moreover, we discuss the range of $\Omega(f^*, \hat{S})$ (Lemma A.5). We find that $\Omega(f^*, \hat{S}) < \tau$ may hold which implies that f^* may not be feasible for Program (ConFair). We first give an example showing that \hat{f}^* can perform very bad over S with respect to the fairness metric.

Example A.3 Our example is shown in Figure 1. We assume that $\mu_0 = 1/3$ and $\mu_1 = 2/3$. Let $\eta = 1/3$ be the noise parameter and we assume $\pi_{20} = \pi_{01} = 1/3$. Consequently, we have that

$$\hat{\mu}_0 = 1/3 \times 2/3 + 2/3 \times 1/3 = 4/9.$$

Then we consider the following simple classifier $f \in \{0, 1\}^{\mathcal{X}}$: $\hat{f}^* = Z$. We directly have that $\Pr[\hat{f}^* = 1 \mid Z = 0] = 0$ and $\Pr[\hat{f}^* = 1 \mid Z = 1] = 1$, which implies that $\gamma(\hat{f}^*, S) = 0$. We also have that

$$\begin{aligned} \Pr[\hat{f}^* = 1 \mid \hat{Z} = 0] &= \Pr[Z = 1 \mid \hat{Z} = 0] && (\hat{f}^* = Z) \\ &= \frac{\pi_{01} \cdot \mu_1}{\hat{\mu}_0} && (\text{Observation 4.7}) \\ &= 0.5, \end{aligned}$$

and

$$\begin{aligned} \Pr[\hat{f}^* = 1 \mid \hat{Z} = 1] &= \Pr[Z = 1 \mid \hat{Z} = 1] && (\hat{f}^* = Z) \\ &= \frac{\pi_{11} \cdot \mu_1}{\hat{\mu}_1} && (\text{Observation 4.7}) \\ &= 0.8, \end{aligned}$$

which implies that $\gamma(\hat{f}^*, \hat{S}) = 0.625$. Hence, there is a gap between $\gamma(\hat{f}^*, S)$ and $\gamma(\hat{f}^*, \hat{S})$, say 0.625, in this example. Consequently, \hat{f}^* can be very unfair over S , and hence, is far from f^* .

Next, we give some theoretical results showing the weaknesses of Program (ConFair).

An upper bound for $\gamma(f, S)$. More generally, given a classifier $f \in \{0, 1\}^{\mathcal{X}}$, we provide an upper bound for $\gamma(f, S)$ that is represented by $\gamma(f, \hat{S})$; see the following lemma.

Lemma A.4 (An upper bound for $\gamma(f, S)$) Suppose we have

1. $\Pr[f = 1 \mid \hat{Z} = 0] \leq \Pr[f = 1 \mid \hat{Z} = 1]$;
2. $\Pr[f = 1, Z = 0 \mid \hat{Z} = 0] \leq \alpha_0 \cdot \Pr[f = 1, Z = 1 \mid \hat{Z} = 0]$ for some $\alpha_0 \in [0, 1]$;
3. $\Pr[f = 1, Z = 0 \mid \hat{Z} = 1] \leq \alpha_1 \cdot \Pr[f = 1, Z = 1 \mid \hat{Z} = 1]$ for some $\alpha_1 \in [0, 1]$.

Let $\beta_{ij} = \frac{\hat{\mu}_i}{\mu_j}$ for $i, j \in \{0, 1\}$. The following inequality holds

$$\gamma(f, S) \leq \frac{\alpha_0(1 + \alpha_1)\beta_{00} \cdot \gamma(f, \hat{S}) + \alpha_1(1 + \alpha_0)\beta_{10}}{(1 + \alpha_1)\beta_{01} \cdot \gamma(f, \hat{S}) + (1 + \alpha_0)\beta_{11}} \leq \max\{\alpha_0, \alpha_1\} \cdot \frac{\mu_1}{\mu_0}.$$

The intuition of the first assumption is that the statistical rate for $Z = 0$ is at most that for $Z = 1$ over the noisy dataset \hat{S} . The second and the third assumptions require the classifier f to be less positive when $Z = 0$. Intuitively, f is restricted to induce a smaller statistical rate for $Z = 0$ over both S and \hat{S} . Specifically, if $\alpha_0 = \alpha_1 = 0$ as in Example A.3, we have $\gamma(f, S) = 0$. Even if $\alpha_0 = \alpha_1 = 1$, we have $\gamma(f, S) \leq \frac{\mu_1}{\mu_0}$ which does not depend on $\gamma(f, \hat{S})$.

Proof: [Proof of Lemma A.4] By the first assumption, we have

$$\gamma(f, \hat{S}) = \frac{\Pr[f = 1 \mid \hat{Z} = 0]}{\Pr[f = 1 \mid \hat{Z} = 1]}. \quad (25)$$

By the second assumption, we have

$$\begin{aligned}
& \Pr[f = 1, Z = 1 \mid \widehat{Z} = 0] \\
&= \frac{(1 + \alpha_0) \cdot \Pr[f = 1, Z = 1 \mid \widehat{Z} = 0]}{1 + \alpha_0} \\
&\geq \frac{\Pr[f = 1, Z = 1 \mid \widehat{Z} = 0] + \Pr[f = 1, Z = 0 \mid \widehat{Z} = 0]}{1 + \alpha_0} \\
&= \frac{1}{1 + \alpha_0} \cdot \Pr[f = 1 \mid \widehat{Z} = 0].
\end{aligned} \tag{26}$$

Similarly, we have the following

$$\Pr[f = 1, Z = 0 \mid \widehat{Z} = 0] \leq \frac{\alpha_0}{1 + \alpha_0} \cdot \Pr[f = 1 \mid \widehat{Z} = 0]. \tag{27}$$

Also, by the third assumption, we have

$$\Pr[f = 1, Z = 1 \mid \widehat{Z} = 1] \geq \frac{1}{1 + \alpha_1} \cdot \Pr[f = 1 \mid \widehat{Z} = 1], \tag{28}$$

and

$$\Pr[f = 1, Z = 0 \mid \widehat{Z} = 1] \leq \frac{\alpha_1}{1 + \alpha_1} \cdot \Pr[f = 1 \mid \widehat{Z} = 1]. \tag{29}$$

Then

$$\begin{aligned}
& \Pr[f = 1 \mid Z = 0] \\
&= \Pr[f = 1, \widehat{Z} = 0 \mid Z = 0] + \Pr[f = 1, \widehat{Z} = 1 \mid Z = 0] \\
&= \Pr[f = 1, Z = 0 \mid \widehat{Z} = 0] \cdot \frac{\widehat{\mu}_0}{\mu_0} \\
&\quad + \Pr[f = 1, Z = 0 \mid \widehat{Z} = 1] \cdot \frac{\widehat{\mu}_1}{\mu_0} \\
&= \Pr[f = 1, Z = 0 \mid \widehat{Z} = 0] \cdot \beta_{00} \\
&\quad + \Pr[f = 1, Z = 0 \mid \widehat{Z} = 1] \cdot \beta_{10} \quad (\text{Defn. of } \beta_{00} \text{ and } \beta_{10}) \\
&\leq \frac{\alpha_0 \beta_{00}}{1 + \alpha_0} \cdot \Pr[f = 1 \mid \widehat{Z} = 0] + \frac{\alpha_1 \beta_{10}}{1 + \alpha_1} \cdot \Pr[f = 1 \mid \widehat{Z} = 1]. \quad (\text{Ineqs. (27) and (29)})
\end{aligned} \tag{30}$$

By a similar argument, we have

$$\begin{aligned}
& \Pr[f = 1 \mid Z = 1] \\
&= \Pr[f = 1, Z = 1 \mid \widehat{Z} = 0] \cdot \beta_{01} \\
&\quad + \Pr[f = 1, Z = 1 \mid \widehat{Z} = 1] \cdot \beta_{11} \quad (\text{Defn. of } \beta_{01} \text{ and } \beta_{11}) \\
&\geq \frac{\beta_{01}}{1 + \alpha_0} \cdot \Pr[f = 1 \mid \widehat{Z} = 0] + \frac{\beta_{11}}{1 + \alpha_1} \cdot \Pr[f = 1 \mid \widehat{Z} = 1]. \quad (\text{Ineqs. (26) and (28)})
\end{aligned} \tag{31}$$

Thus, we have

$$\begin{aligned}
\gamma(f, S) &\leq \frac{\Pr[f = 1 \mid Z = 0]}{\Pr[f = 1 \mid Z = 1]} && \text{(Defn. of } \gamma(f, S)) \\
&\leq \frac{\frac{\alpha_0 \beta_{00}}{1 + \alpha_0} \cdot \Pr[f = 1 \mid \widehat{Z} = 0] + \frac{\alpha_1 \beta_{10}}{1 + \alpha_1} \cdot \Pr[f = 1 \mid \widehat{Z} = 1]}{\frac{\beta_{01}}{1 + \alpha_0} \cdot \Pr[f = 1 \mid \widehat{Z} = 0] + \frac{\beta_{11}}{1 + \alpha_1} \cdot \Pr[f = 1 \mid \widehat{Z} = 1]} && \text{(Ineqs. (30) and (31))} \\
&= \frac{\alpha_0(1 + \alpha_1)\beta_{00} \cdot \gamma(f, \widehat{S}) + \alpha_1(1 + \alpha_0)\beta_{10}}{(1 + \alpha_1)\beta_{01} \cdot \gamma(f, \widehat{S}) + (1 + \alpha_0)\beta_{11}} && \text{(Eq. (25))} \\
&\leq \max \left\{ \alpha_0 \cdot \frac{\beta_{00}}{\beta_{01}}, \alpha_1 \cdot \frac{\beta_{10}}{\beta_{11}} \right\} \\
&= \max \{ \alpha_0, \alpha_1 \} \cdot \frac{\mu_1}{\mu_0}, && \text{(Defn. of } \beta_{ij})
\end{aligned}$$

which completes the proof. \square

f^* may not be feasible in Program (ConFair). We consider a simple case that $\eta_1 = \eta_2 = \eta$. Without loss of generality, we assume that $\Pr[f^* = 1 \mid Z = 0] \leq \Pr[f^* = 1 \mid Z = 1]$, i.e., the statistical rate of $Z = 0$ is smaller than that of $Z = 1$ over S . Consequently, we have

$$\gamma(f^*, S) = \frac{\Pr[f^* = 1 \mid Z = 0]}{\Pr[f^* = 1 \mid Z = 1]}.$$

Lemma A.5 (Range of $\Omega(f^*, \widehat{S})$) *Let $\varepsilon \in (0, 0.5)$ be a given constant and let*

$$\Gamma = \frac{\eta\mu_0 + (1 - \eta)(1 - \mu_0)}{(1 - \eta)\mu_0 + \eta(1 - \mu_0)} \cdot \frac{(1 - \eta)\mu_0\gamma(f^*, S) + \eta(1 - \mu_0)}{\eta\mu_0\gamma(f^*, S) + (1 - \eta)(1 - \mu_0)}.$$

With probability at least $1 - 4e^{-\frac{\varepsilon^2 \eta \lambda N}{192}}$, the following holds

$$\gamma(f^*, \widehat{S}) \in (1 \pm \varepsilon) \cdot \min \left\{ \Gamma, \frac{1}{\Gamma} \right\}.$$

For instance, if $\mu_0 = 0.5$, $\gamma(f^*, S) = 0.8 = \tau$ and $\eta = 0.2$, we have

$$\gamma(f^*, \widehat{S}) \approx 0.69 < \tau.$$

Then f^* is not a feasible solution of Program (ConFair). Before proving the lemma, we give some intuitions.

Discussion A.6 *By Definition 2.1, we have that for a given classifier $f^* \in \mathcal{F}$,*

$$\Pr[\widehat{Z} = 1 \mid Z = 0] \approx \Pr[\widehat{Z} = 0 \mid Z = 1] \approx \eta \quad (32)$$

Moreover, the above property also holds when conditioned on a subset of samples with $Z = 0$ or $Z = 1$. Specifically, for $i \in \{0, 1\}$,

$$\Pr[\widehat{Z} = 1 \mid f^* = 1, Z = 0] \approx \Pr[\widehat{Z} = 0 \mid f^* = 1, Z = 1] \approx \eta \quad (33)$$

Another consequence of Property (32) is that for $i \in \{0, 1\}$,

$$\begin{aligned}
\widehat{\mu}_i &= \pi_{i,i}\mu_i + \pi_{i,1-i}\mu_{1-i} && \text{(Observation 4.7)} \\
&\approx (1 - \eta)\mu_i + \eta\mu_{1-i}. && \text{(Property (32))}
\end{aligned} \quad (34)$$

Then we have

$$\begin{aligned}
& \Pr[f^* = 1 \mid \widehat{Z} = 0] \\
= & \Pr[f^* = 1, Z = 0 \mid \widehat{Z} = 0] + \Pr[f^* = 1, Z = 1 \mid \widehat{Z} = 0] \\
= & \Pr[Z = 0 \mid \widehat{Z} = 0] \cdot \Pr[f^* = 1 \mid Z = 0, \widehat{Z} = 0] \\
& + \Pr[Z = 1 \mid \widehat{Z} = 0] \cdot \Pr[f^* = 1 \mid Z = 1, \widehat{Z} = 0] \\
= & \frac{\pi_{00}\mu_0}{\widehat{\mu}_0} \cdot \Pr[f^* = 1 \mid Z = 0, \widehat{Z} = 0] + \frac{\pi_{01}\mu_1}{\widehat{\mu}_0} \cdot \Pr[f^* = 1 \mid Z = 1, \widehat{Z} = 0] \\
& (\text{Observation 4.7}) \\
\approx & \frac{(1-\eta)\mu_0}{(1-\eta)\mu_0 + \eta\mu_1} \cdot \Pr[f^* = 1 \mid Z = 0, \widehat{Z} = 0] \\
& + \frac{\eta\mu_1}{(1-\eta)\mu_0 + \eta\mu_1} \cdot \Pr[f^* = 1 \mid Z = 1, \widehat{Z} = 0] \quad (\text{Properties (32) and (34)}) \\
= & \frac{(1-\eta)\mu_0}{(1-\eta)\mu_0 + \eta(1-\mu_0)} \cdot \frac{\Pr[f^* = 1 \mid Z = 0] \cdot \Pr[\widehat{Z} = 0 \mid f^* = 1, Z = 0]}{\Pr[\widehat{Z} = 0 \mid Z = 0]} \\
& + \frac{\eta\mu_1}{(1-\eta)\mu_0 + \eta(1-\mu_0)} \cdot \frac{\Pr[f^* = 1 \mid Z = 1] \cdot \Pr[\widehat{Z} = 0 \mid f^* = 1, Z = 1]}{\Pr[\widehat{Z} = 0 \mid Z = 1]} \\
\approx & \frac{(1-\eta)\mu_0}{(1-\eta)\mu_0 + \eta(1-\mu_0)} \cdot \Pr[f^* = 1 \mid Z = 0] \\
& + \frac{\eta\mu_1}{(1-\eta)\mu_0 + \eta(1-\mu_0)} \cdot \Pr[f^* = 1 \mid Z = 1]. \quad (\text{Properties (32) and (33)})
\end{aligned}$$

Similarly, we can represent

$$\begin{aligned}
& \Pr[f^* = 1 \mid \widehat{Z} = 1] \\
\approx & \frac{\eta\mu_0}{\eta\mu_0 + (1-\eta)(1-\mu_0)} \Pr[f^* = 1 \mid Z = 0] + \frac{(1-\eta)\mu_1}{\eta\mu_0 + (1-\eta)(1-\mu_0)} \Pr[f^* = 1 \mid Z = 1].
\end{aligned}$$

Applying the approximate values of $\Pr[f^* = 1 \mid \widehat{Z} = 0]$ and $\Pr[f^* = 1 \mid \widehat{Z} = 1]$ to compute $\gamma(f^*, S)$, we have Lemma A.5.

Proof: [Proof of Lemma A.5] By definition, we have

$$\gamma(f^*, \widehat{S}) \leq \frac{\Pr[f^* = 1 \mid \widehat{Z} = 0]}{\Pr[f^* = 1 \mid \widehat{Z} = 1]}.$$

Thus, it suffices to provide an upper bound for $\Pr[f^* = 1 \mid \widehat{Z} = 0]$ and a lower bound for $\Pr[f^* = 1 \mid \widehat{Z} = 1]$. Similar to Discussion A.6, we have

$$\begin{aligned}
\Pr[f^* = 1 \mid \widehat{Z} = 0] &= \frac{\Pr[Z = 0] \cdot \Pr[f^* = 1 \mid Z = 0] \cdot \Pr[\widehat{Z} = 0 \mid f^* = 1, Z = 0]}{\Pr[\widehat{Z} = 0]} \\
&\quad + \frac{\Pr[Z = 1] \cdot \Pr[f^* = 1 \mid Z = 1] \cdot \Pr[\widehat{Z} = 0 \mid f^* = 1, Z = 1]}{\Pr[\widehat{Z} = 0]} \\
&= \frac{\mu_0 \cdot \Pr[f^* = 1 \mid Z = 0] \cdot \Pr[\widehat{Z} = 0 \mid f^* = 1, Z = 0]}{\pi_{00}\mu_0 + \pi_{01}(1 - \mu_0)} \\
&\quad + \frac{\mu_1 \cdot \Pr[f^* = 1 \mid Z = 1] \cdot \Pr[\widehat{Z} = 0 \mid f^* = 1, Z = 1]}{\pi_{00}\mu_0 + \pi_{01}(1 - \mu_0)},
\end{aligned} \tag{35}$$

and

$$\begin{aligned}
\Pr[f^* = 1 \mid \widehat{Z} = 1] &= \frac{\Pr[Z = 0] \cdot \Pr[f^* = 1 \mid Z = 0] \cdot \Pr[\widehat{Z} = 1 \mid f^* = 1, Z = 0]}{\Pr[\widehat{Z} = 1]} \\
&\quad + \frac{\Pr[Z = 1] \cdot \Pr[f^* = 1 \mid Z = 1] \cdot \Pr[\widehat{Z} = 1 \mid f^* = 1, Z = 1]}{\Pr[\widehat{Z} = 1]} \\
&= \frac{\mu_0 \cdot \Pr[f^* = 1 \mid Z = 0] \cdot \Pr[\widehat{Z} = 1 \mid f^* = 1, Z = 0]}{\pi_{11}(1 - \mu_0) + \pi_{20}\mu_0} \\
&\quad + \frac{\mu_1 \cdot \Pr[f^* = 1 \mid Z = 1] \cdot \Pr[\widehat{Z} = 1 \mid f^* = 1, Z = 1]}{\pi_{11}(1 - \mu_0) + \pi_{20}\mu_0},
\end{aligned} \tag{36}$$

We then analyze the right side of the Equation (35). We take the term $\Pr[\widehat{Z} = 0 \mid f^* = 1, Z = 1]$ as an example. Let $A = \{i \in [n] : f^*(s_i) = 1, z_i = 0\}$. By Assumption 1, we have $|A| \geq \lambda N$. For $i \in A$, let X_i be the random variable where $X_i = 1 - \widehat{z}_i$. By Definition 2.1, we know that $\Pr[X_i = 1] = \eta$. Also,

$$\Pr[\widehat{Z} = 0 \mid f^* = 1, Z = 1] = \frac{\sum_{i \in A} X_i}{|A|}. \tag{37}$$

Since all X_i ($i \in A$) are independent, we have

$$\begin{aligned}
\Pr\left[\sum_{i \in A} X_i \in (1 \pm \frac{\varepsilon}{8}) \cdot \eta |A|\right] &\geq 1 - 2e^{-\frac{\varepsilon^2 \eta |A|}{192}} \quad (\text{Chernoff bound}) \\
&\geq 1 - 2e^{-\frac{\varepsilon^2 \eta \lambda N}{192}}. \quad (|A| \geq \lambda N)
\end{aligned} \tag{38}$$

Thus, with probability at least $1 - 2e^{-\frac{\varepsilon^2 \eta \lambda N}{192}}$,

$$\begin{aligned}
\Pr[\widehat{Z} = 0 \mid f^* = 1, Z = 1] &= \frac{\sum_{i \in A} X_i}{|A|} \quad (\text{Eq. (37)}) \\
&\in (1 \pm \frac{\varepsilon}{8}) \cdot \frac{\eta |A|}{|A|} \quad (\text{Ineq. (38)}) \\
&\in (1 \pm \frac{\varepsilon}{8}) \eta.
\end{aligned}$$

Consequently, we have

$$\begin{aligned}
\Pr \left[\widehat{Z} = 1 \mid f^* = 1, Z = 1 \right] &= 1 - \Pr \left[\widehat{Z} = 0 \mid f^* = 1, Z = 1 \right] \\
&\in 1 - (1 \pm \frac{\varepsilon}{8})\eta && \text{(Ineq. (39))} \\
&\in (1 \pm \frac{\varepsilon}{8})(1 - \eta) && (\eta < 0.5)
\end{aligned}$$

Similarly, we can prove that with probability at least $1 - 4e^{-\frac{\varepsilon^2 \eta \lambda N}{192}}$,

- $\pi_{01}, \pi_{20}, \Pr \left[\widehat{Z} = 1 \mid f^* = 1, Z = 0 \right], \Pr \left[\widehat{Z} = 0 \mid f^* = 1, Z = 1 \right] \in (1 \pm \frac{\varepsilon}{8})\eta;$
- $\pi_{00}, \pi_{11}, \Pr \left[\widehat{Z} = 0 \mid f^* = 1, Z = 0 \right], \Pr \left[\widehat{Z} = 1 \mid f^* = 1, Z = 1 \right] \in (1 \pm \frac{\varepsilon}{8})(1 - \eta).$

Applying these inequalities to Equations (35) and (36), we have that with probability at least $1 - 4e^{-\frac{\varepsilon^2 \eta \lambda N}{192}}$,

$$\begin{aligned}
\frac{\Pr \left[f^* = 1 \mid \widehat{Z} = 0 \right]}{\Pr \left[f^* = 1 \mid \widehat{Z} = 1 \right]} &\in (1 \pm \varepsilon) \cdot \frac{\eta \mu_0 + (1 - \eta)(1 - \mu_0)}{(1 - \eta)\mu_0 + \eta(1 - \mu_0)} \cdot \frac{(1 - \eta)\mu_0 \gamma(f^*, S) + \eta(1 - \mu_0)}{\eta \mu_0 \gamma(f^*, S) + (1 - \eta)(1 - \mu_0)} \\
&\in (1 \pm \varepsilon) \cdot \Gamma,
\end{aligned}$$

and

$$\frac{\Pr \left[f^* = 1 \mid \widehat{Z} = 1 \right]}{\Pr \left[f^* = 1 \mid \widehat{Z} = 0 \right]} \in (1 \pm \varepsilon) \cdot \frac{1}{\Gamma}.$$

By the definition of $\gamma(f^*, \widehat{S})$, we complete the proof. \square

B Other empirical results

We first present the performances of our denoised algorithm **DenoisedLR** under different τ ; summarized by Figures 2 and 3. In general, as τ increases, the accuracy of **DenoisedLR** decreases and γ increases up to 1. Sometimes, the average of γ for $\tau = 0.95$ may be smaller than that for $\tau = 0.9$. A possible reason is that $\tau = 0.95$ is a stronger constraint such that the success probability of Theorem 3.3 is smaller. Then there may exist some noisy dataset \widehat{S} such that the leaned classifier f of **DenoisedLR** violates $\gamma(f, S) \geq \tau$.

We also investigate the performances of algorithms w.r.t. varying η_0, η_1 . We consider $\eta_0 = \eta_1 = \eta \in \{0.1, 0.15, 0.2, 0.25, 0.3, 0.35, 0.4\}$. Other settings are the same as in the main text. Both **ZVRG** and **GYF** learn the unconstrained optimal classifier whose γ is at most 0.7. The performances of **FairLR** and **DenoisedLR** are summarized in Figures 4 and 5, where we select $\tau = 0.8$. Observe that **DenoisedLR** always achieves γ close to 0.8, which is higher than that of **FairLR** for different η . Meanwhile, the loss of accuracy for **DenoisedLR** compared to **FairLR** is small, say around 1%-6%.

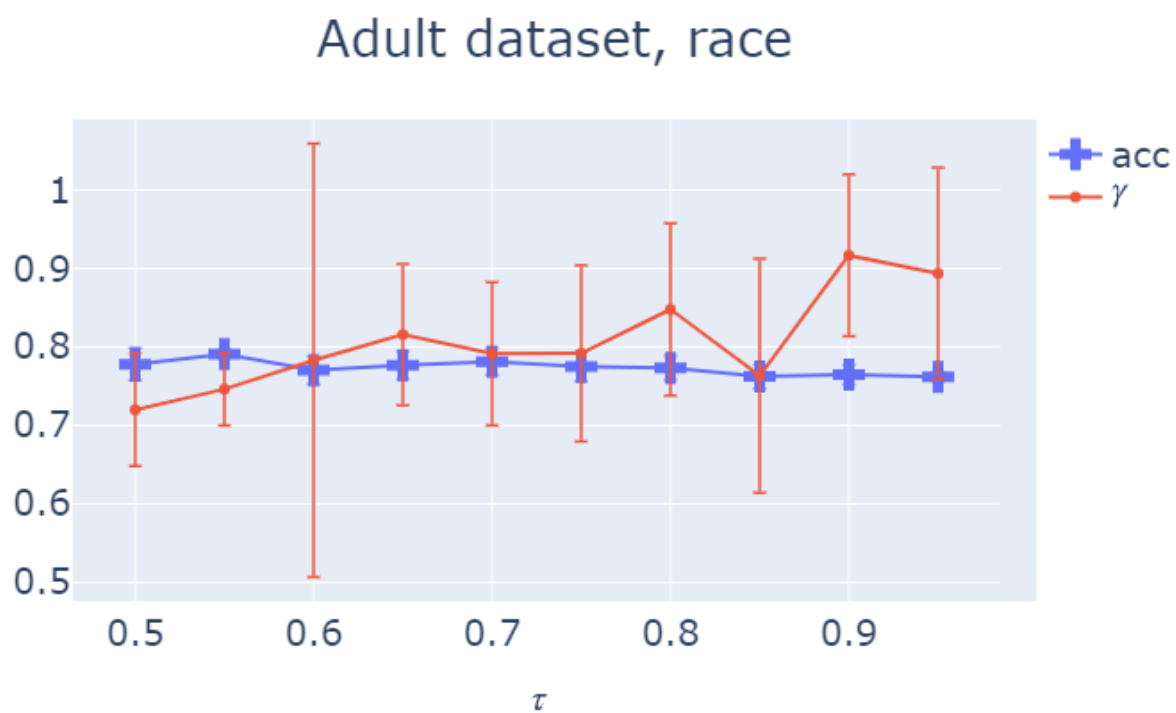
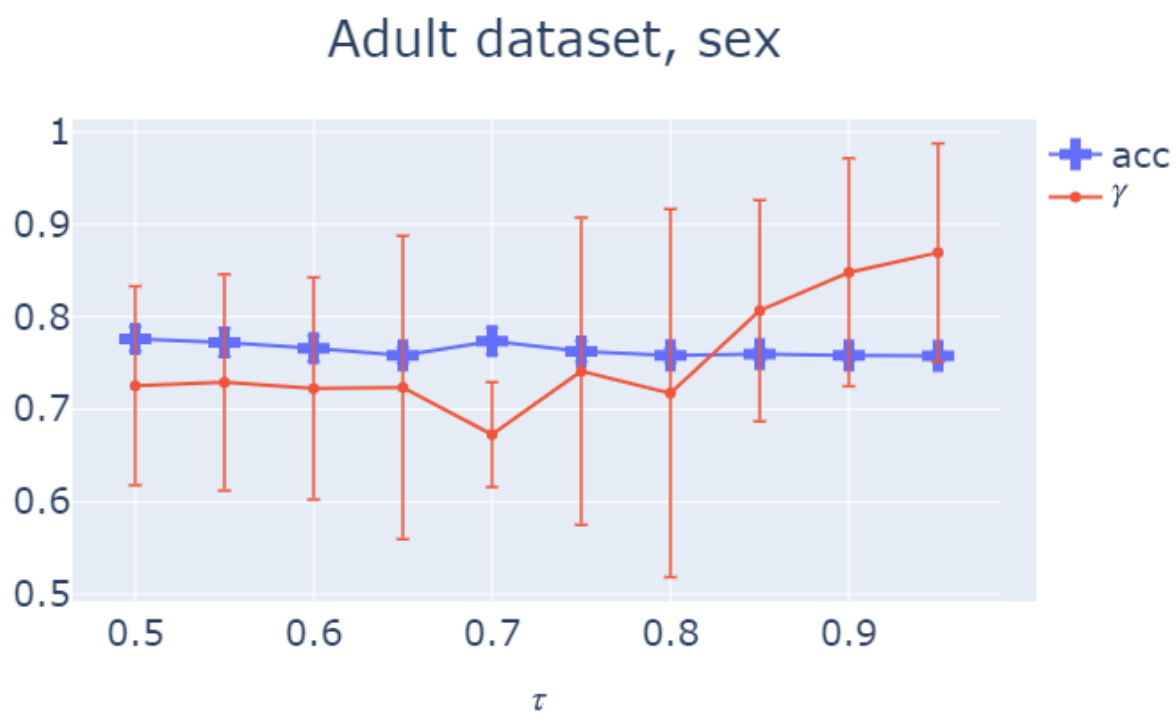


Figure 2: The accuracy and γ of **DenoisedLR** over the **Adult** dataset w.r.t. varying τ .

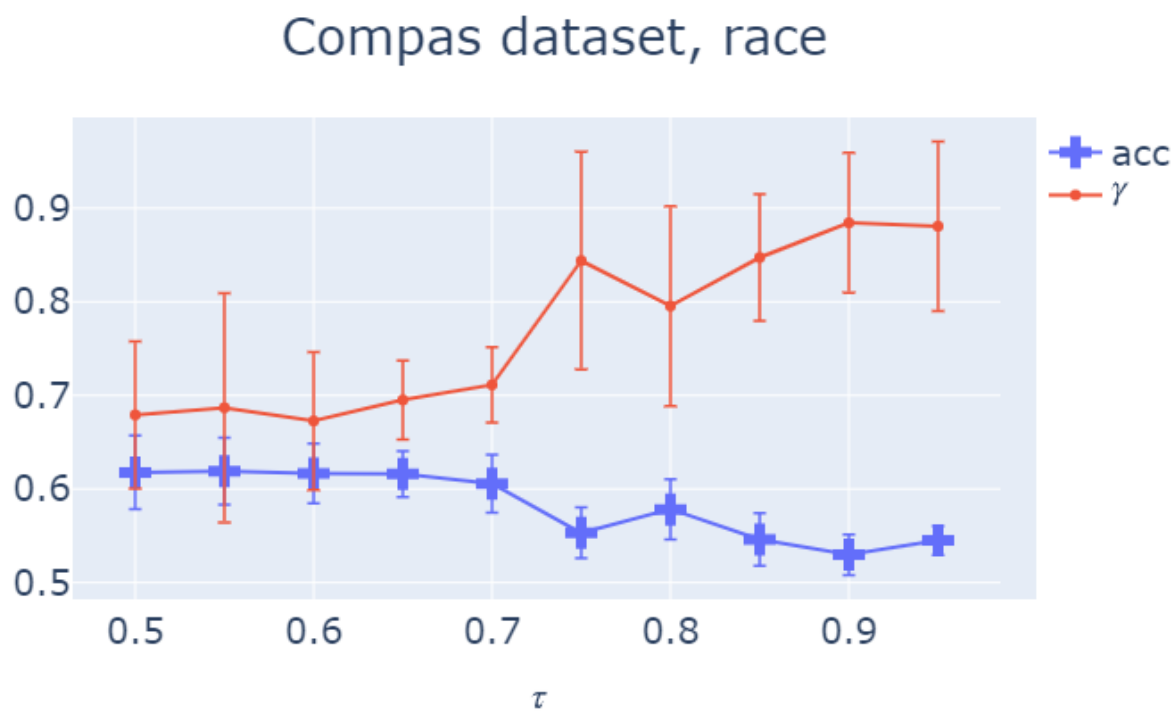
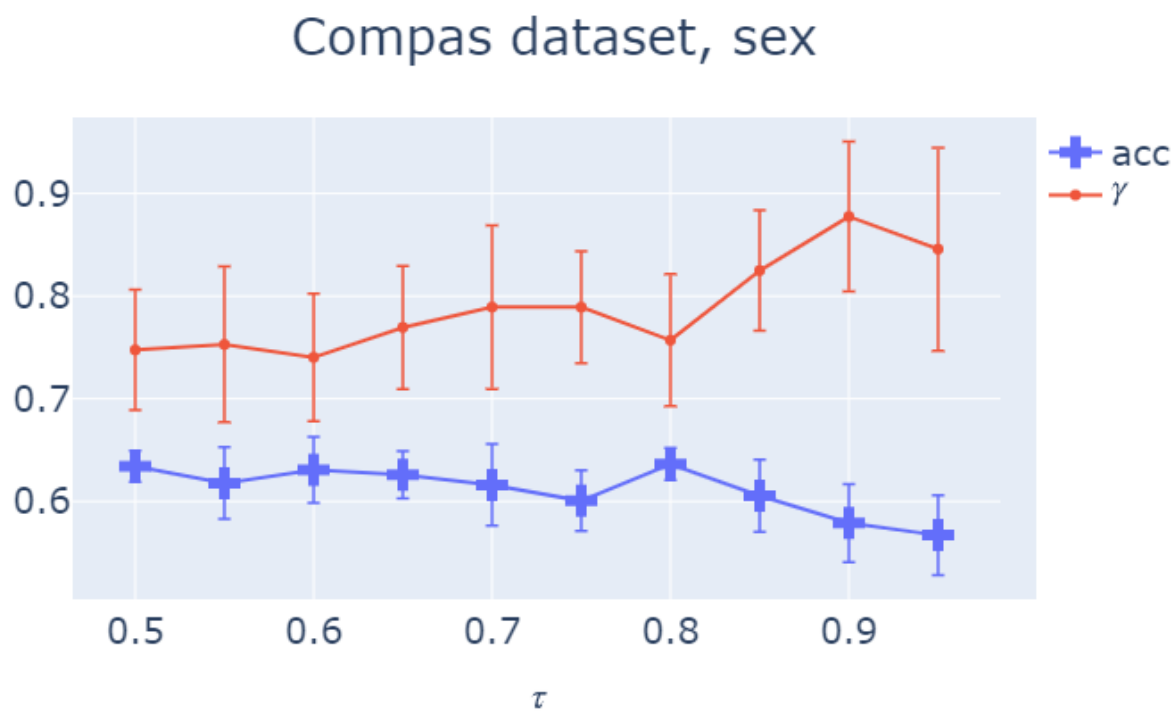


Figure 3: The accuracy and γ of **DenoisedLR** over the **Compas** dataset w.r.t. varying τ .

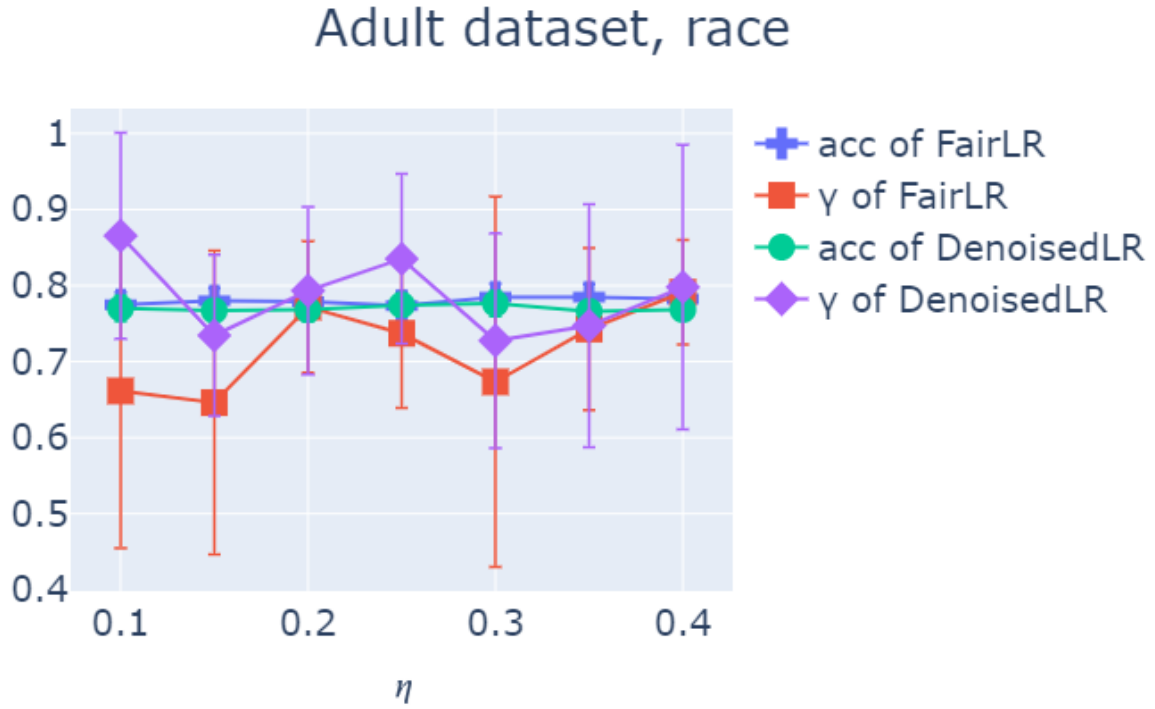
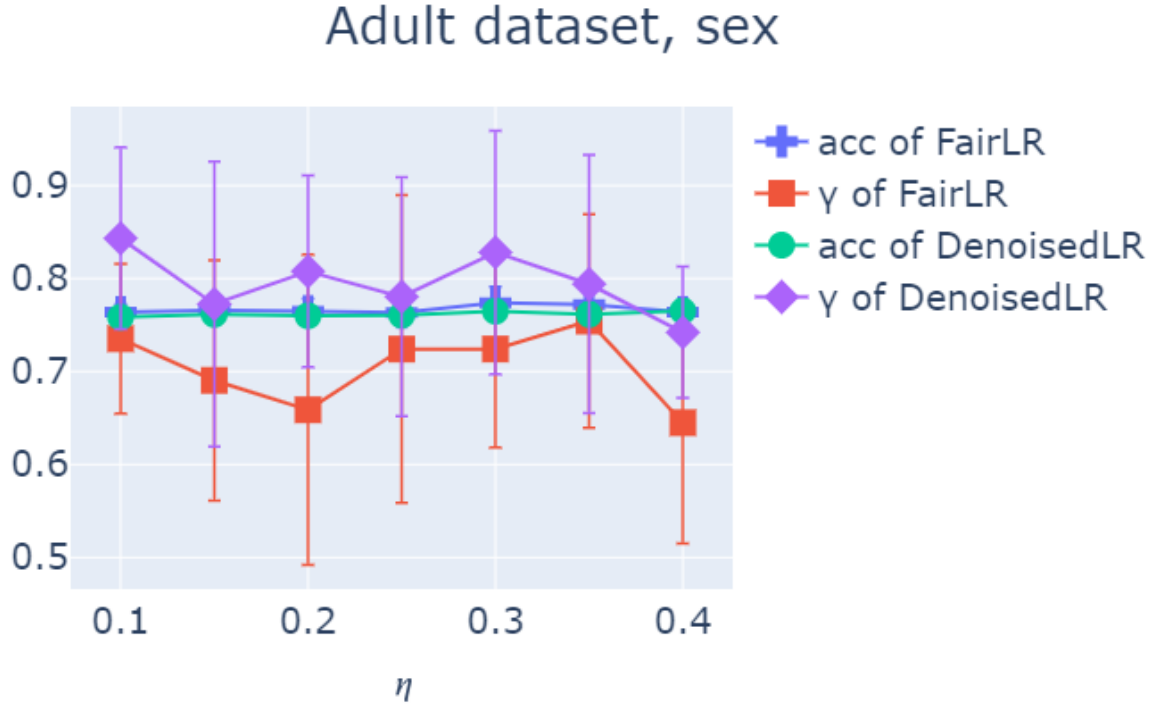
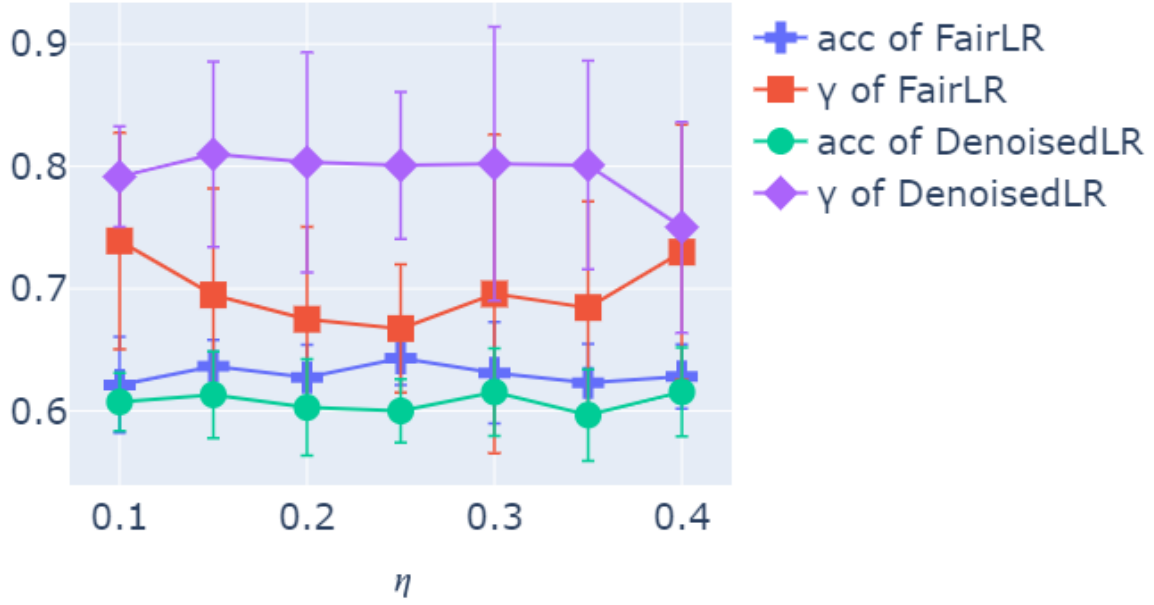


Figure 4: The accuracy and γ of **FairLR** and **DenoisedLR** over the **Adult** dataset w.r.t. varying η for $\tau = 0.8$. When the protected attribute is sex, the unconstrained optimal classifier has accuracy % and $\gamma =$. When the protected attribute is race, the unconstrained optimal classifier has accuracy % and $\gamma =$.

Compas dataset, sex



Compas dataset, race

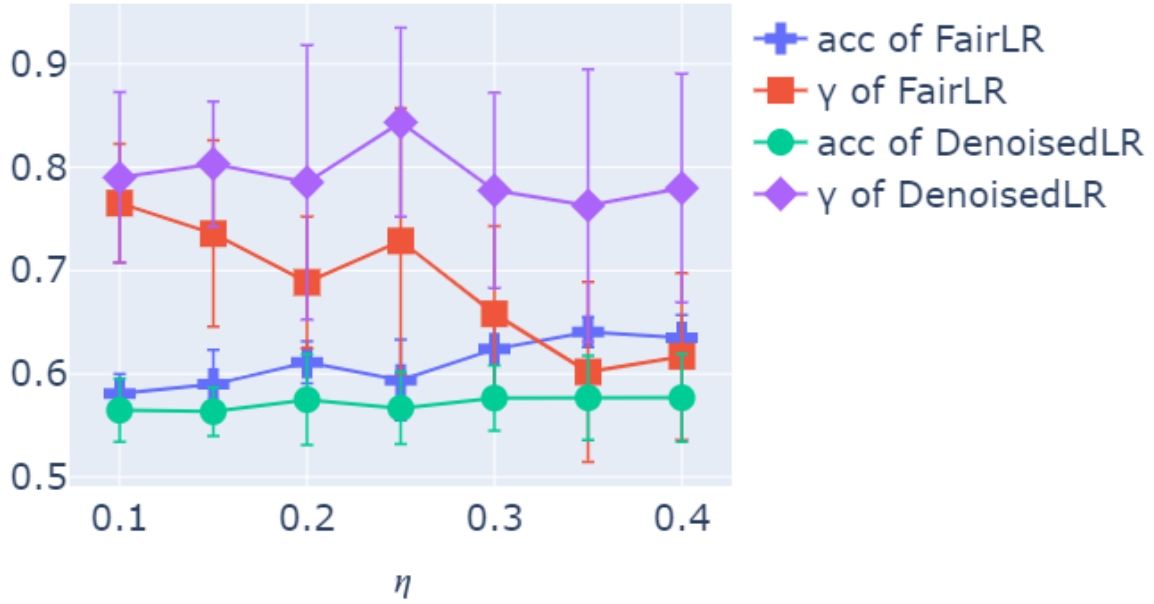


Figure 5: The accuracy and γ of **FairLR** and **DenoisedLR** over the **Compas** dataset w.r.t. varying η for $\tau = 0.8$. When the protected attribute is sex, the unconstrained optimal classifier has accuracy 66% and $\gamma = 0.62$. When the protected attribute is race, the unconstrained optimal classifier has accuracy 66% and $\gamma = 0.53$.