L_q REGULARIZATION FOR FAIRNESS AI ROBUST TO SAMPLING BIAS

Anonymous authors
Paper under double-blind review

ABSTRACT

It is well recognized that historical biases exist in training data against a certain sensitive group (e.g., non-white, women) which are socially unacceptable, and these unfair biases are inherited to trained AI models. Various learning algorithms have been proposed to remove or alleviate unfair biases in trained AI models. In this paper, we consider another type of bias in training data so-called *sampling bias* in view of fairness AI. Here, sampling bias means that training data do not represent well the population of interest. Sampling bias occurs when special sampling designs (e.g., stratified sampling) are used when collecting training data, or the population where training data are collected is different from the population of interest. When sampling bias exists, fair AI models on training data may not be fair in test data. To ensure fairness on test data, we develop computationally efficient learning algorithms robust to sampling bias. In particular, we propose a robust fairness constraint based on the L_q norm which is a generic algorithm to be applied to various fairness AI problems without much hamper. By analyzing multiple benchmark data sets, we show that our proposed robust fairness AI algorithm improves existing fair AI algorithms much in terms of the robustness to sampling bias and has significant computational advantages compared to other robust fair AI algorithms.

1 INTRODUCTION

AI (Artificial Intelligence) is being widely used in various decision-makings directly related to human social life, such as credit scoring, criminal risk assessment, and college admissions (Angwin et al., 2016). However, it is well recognized that there exist historical biases in training data against a certain sensitive group (e.g., non-white, women) which are not socially acceptable due to ethics or regulatory frameworks for fairness, and these unfair biases are inherited to trained AI models (Kleinberg et al., 2018; Mehrabi et al., 2019). A lot of learning algorithms have been proposed to remove or alleviate unfair biases in trained AI models to treat sensitive groups as equally as possible. In general, these algorithms try to search AI models which are not only accurate but also similar between sensitive groups in a certain sense.

In this paper, we consider another type of biases in training data, so-called *sampling bias* in view of fairness AI. Here, we say that sampling bias exists in given training data when the training data do not represent the population of interest well. Possible causes of sampling bias are the usage of special sampling designs other than the simple random sampling (e.g., stratified sampling) and population mismatch (i.e., the population where training data are collected is different from the population of interest, or the population keeps changing as time goes). When sampling bias exists, fair AI models on training data may not be fair on test data. Hence, to ensure fairness on test data, we need a device to learn AI models whose fairness is robust to sampling bias.

Contributions: This paper aims to develop computationally efficient learning algorithms that yield fair prediction models whose fairness is robust to sampling bias. Our main contributions are summarized as follows:

- We propose a *robust fairness constraint* which yields prediction models whose fairness is robust to sampling bias.

- To resolve computational issues, we introduce the L_q-robust fairness constraints and implement the corresponding learning algorithm. Our learning algorithm is flexible enough to apply to various fairness constraints without much hamper.
- We demonstrate by analyzing several benchmark datasets that our learning algorithm provides prediction models which are robust to sampling bias.

Related works: Various learning methods have been proposed to train an accurate and fair model, which are classified by the two notions - definition of fairness and process of learning. Most definitions of fairness can be roughly categorized into two groups - group fairness and individual fairness. Group fairness (Calders et al., 2009; Hardt et al., 2016) requires that a certain quantity of a prediction model should be similar between sensitive groups while individual fairness (Dwork et al., 2012) demands the predictions of two similar individuals should be similar.

Fair learning algorithms are divided into three groups in view of the process of learning. The *pre-processing methods* remove bias in training data (Kamiran & Calders, 2012) before training prediction models. Meanwhile, the *in-processing methods* train a prediction model by minimizing the cost function subject to fairness constraints (Kamishima et al., 2012; Menon & Williamson, 2018). Since most fairness constraints are not differentiable, various surrogate constraints have been proposed (Zafar et al., 2017; 2019; Donini et al., 2018; Wu et al., 2019; Padala & Gujar, 2020). The last one is the *post-processing methods*, which first train a prediction model without any constraints and then transform the trained prediction model for each sensitive group to meet a fairness criterion (Hardt et al., 2016; Jiang et al., 2020; Wei et al., 2020). This paper focuses on the *in-processing methods* and modifies them to be robust to sampling bias.

Several works have been done for fairness AI considering biases other than historical or regulatory biases. Fogliato et al. (2020) took into account a possible noisy measurement bias in the observed label y. Hashimoto et al. (2018); Lamy et al. (2019); Wang et al. (2020) have considered problems where the information of sensitive groups are not available or contaminated by noises.

For sampling bias, Taskesen et al. (2020) implemented a learning algorithm based on the DRO (distributional robust optimization) approach. But, this algorithm is only applicable when both the loss and fairness constraint are convex. Mandal et al. (2020) has developed an interesting optimization algorithm for learning fair prediction models under sampling bias. The algorithm of Mandal et al. (2020), however, requires learning prediction models iteratively and thus is hard to be applied for computationally intensive prediction models such as deep neural networks. Our learning algorithm can be considered as a smooth version of Mandal et al. (2020) to reduce computational burdens.The corresponding optimization algorithms are based on standard (stochastic) gradient descent and so our learning algorithm can be applied to most standard fairness AI algorithms to make them robust to sampling bias without much modification.

2 REVIEW OF LEARNING ALGORITHMS FOR FAIRNESS

Let $(Y, \mathbf{X}, Z)$ be the random vector of a triplet of output, input, and sensitive variable, whose distribution is P. For simplicity, we consider a binary classification problem where $Y \in \{-1, 1\}$ and a binary sensitive variable $Z \in \{0, 1\}$. For a given loss function l and a class $\mathcal{F}$ of prediction models, the aim of supervised learning is to find f^* defined as $f^* = \operatorname{argmin}_{f \in \mathcal{F}} \mathbb{E}\{l(Y, f(\mathbf{X}))\}$. Due to historical biases or social prejudices, the optimal prediction model f^* would not be socially acceptable because it treats individuals or sensitive groups unfairly. Thus, we want to search f, which is fair and at the same time makes the population risk $\mathbb{E}\{l(Y, f(\mathbf{X}))\}$ as small as possible.

Suppose that $\mathcal{F}_{\text{fair}}$ is a subset of $\mathcal{F}$, which consists of all fair prediction models. Then, the goal of fair supervised learning is to find f^*_{fair} defined as $f^*_{\text{fair}} = \operatorname{argmin}_{f \in \mathcal{F}_{\text{fair}}} \mathbb{E}\{l(Y, f(\mathbf{X}))\}$. There are various proposals for $\mathcal{F}_{\text{fair}}$ but most of existing classes $\mathcal{F}_{\text{fair}}$ can be formulated as $\mathcal{F}_{\text{fair}} = \{f \in \mathcal{F} : \phi(f, P) \leq \epsilon\}$, where ϕ is a real valued function so called a fairness constraint function. Below, we give the two fairness constraint functions which we use in the numerical studies.

- Group fairness: Most of between group fairness constraints can be formalized as $\phi(f, P) = |\mathbb{E}(\eta(f, Y, \mathbf{X})|Z = 0) - \mathbb{E}(\eta(f, Y, \mathbf{X})|Z = 1)|$ for a certain functional η. The fairness constraint ϕ becomes the disparity impact Barocas & Selbst (2016) if If

$\eta(f, y, \mathbf{x}) = \mathbb{I}(f(\mathbf{x}) > 0)$. In practice, to avoid computational difficulty in using $\mathbb{I}(\cdot)$, we use the surrogated disparity impact constraint where $\eta(f, y, \mathbf{x}) = \{1 + f(\mathbf{x})\}_+$ (Zafar et al., 2017; Padala & Gujar, 2020). The mean score parity (Coston et al., 2019) is obtained by letting $\eta(f, y, \mathbf{x}) = f(\mathbf{x})$.

- Individual Fairness: In practice, group fairness is not sufficient since a group faired prediction model can be seriously unfair for certain subgroups. To resolve this problem, the notion of individual fairness is considered which requires that similar individuals should be treated similarly Dwork et al. (2012). In this paper, we consider (γ, ξ)-*Uniform Individual Fairness* defined as

$$\phi(f; \gamma, \xi, P) = P\left(\sup_{\mathbf{v}: d(\mathbf{X}, \mathbf{v}) \leq \xi} D(f(\mathbf{X}), f(\mathbf{v})) > \gamma\right),$$

  where $D(\cdot, \cdot)$ is a similarity metric between treatments and $d(\cdot, \cdot)$ is a similarity metric between individuals.

In practice, we do not know P but have training data $(y_1, \mathbf{x}_1, z_1), \ldots, (y_n, \mathbf{x}_n, z_n)$. One of the most popular approaches for estimating f^*_{fair} is a constrained empirical risk minimization approach, which estimates f^*_{fair} by $\hat{f}_{\text{fair}}$ defined as $\hat{f}_{\text{fair}} = \operatorname{argmin}_{f \in \hat{\mathcal{F}}_{\text{fair}}} \sum_{i=1}^n l(y_i, f(\mathbf{x}_i))/n$, where $\hat{\mathcal{F}}_{\text{fair}} = \{f \in \mathcal{F} : \phi(f, P_n) < \epsilon\}$ and P_n is the empirical distribution.

3 Probability model for sampling bias

Let P^{tr} and P^{te} be the two probability measures for training and test datasets, respectively. When $P^{\text{tr}} \neq P^{\text{te}}$, sampling bias exists and standard fair learning algorithms could be problematic.

To model the relation between P^{tr} and P^{te}, let $s : \mathcal{X} \to (0, \infty)$ be a weight function. The probability model for sampling bias assumes that $p^{\text{tr}}(y, \mathbf{x}) \propto s(\mathbf{x}) p^{\text{te}}(y, \mathbf{x})$. Unless $s(\mathbf{x})$ is a constant function, $P^{\text{tr}} \neq P^{\text{te}}$ and sampling bias emerges. We call $s(\cdot)$ the bias function. The following four examples of sampling bias are selected from Mehrabi et al. (2019).

- Sampling bias: Sampling bias is caused by non-random sampling of subgroups, e.g., stratified sampling.
- Self-selection bias: Each datum decides to participate in the training data by her own decision. An example of self-selection bias is a college admission where only the data of applicants are available, and those who do not apply are excluded from the training data. But, the model is required to be fair for all students.
- Population bias: The population where data are collected is different from the target population. For example, population bias can occur on social platforms where the main users are different according to their gender and age.
- Temporal bias: Temporal bias occurs when populations or behaviors change over time. Data collected at different time points may vary according to various criteria, such as system users or system usage.

The interpretation of the bias function s depends on a specific case of sampling bias. If the population is unbiased but training data are biased, which is the case of sampling bias and self-selection bias, the bias function can be interpreted as an inclusion probability. That is, we first sample $(y, \mathbf{x})$ from $P = P^{\text{te}}$ and include it to training data with probability proportional to $s(\mathbf{x})$. On the other hand, the population of interest is different from the population of training data, which is the case for the population bias and temporal bias, a datum $(y, \mathbf{x})$ is generated from $P = P^{\text{tr}}$ and is accepted for test data with probability proportional to $1/s(\mathbf{x})$.

In this paper, we assume that the bias function s depends only on $\mathbf{x}$ but not on y. This assumption is made because we want to separate out unfair (historical or regulatory) bias and sampling bias. Unfair bias exists in $P(y|\mathbf{x})$ which is not affected by the bias function s as long as it depends only on $\mathbf{x}$. That is, the biased probability model shares the same unfair bias and the same Bayes classifier regardless of s. The main purpose of this paper is to study how the sampling bias affects the fairness of estimated prediction models and develop an algorithm that yields prediction models whose fairness is robust to sampling bias.

4 ROBUST FAIRNESS CONSTRAINTS AND CORRESPONDING LEARNING ALGORITHMS

4.1 ROBUST FAIRNESS CONSTRAINT

Suppose that the bias function $s(\cdot)$ is known. Let $w_i = 1/s(\mathbf{x}_i)$ and $\mathbf{w} = (w_1, \ldots, w_n)^\top$. For given f, an unbiased estimator of $\phi(f, P^{\text{te}})$, the fairness function on test data is $\phi(f, P_{n,\mathbf{w}})$, where $P_{n,\mathbf{w}}(\cdot) \propto \sum_{i=1}^n w_i \delta_{(y_i,\mathbf{x}_i,z_i)}(\cdot)$, which is the well known Horvitz-Thompson estimator (H-T estimator) (Horvitz & Thompson, 1952). Then, $\hat{f}_{\text{fair},\mathbf{w}}$ defined as $\hat{f}_{\text{fair},\mathbf{w}} = \text{argmin}_{f \in \hat{\mathcal{F}}_{\text{fair},\mathbf{w}}} \sum_{i=1}^n l(y_i, f(\mathbf{x}_i))/n$, where $\hat{\mathcal{F}}_{\text{fair},\mathbf{w}} = \{f \in \mathcal{F} : \phi(f, P_{n,\mathbf{w}}) < \epsilon\}$ would be a reasonable estimator for fair prediction.

In practice, we do not know the bias function, and in such cases, we may require that $\phi(f, P_{n,\mathbf{w}}) < \epsilon$ for all possible values of $\mathbf{w}$. This requirement, however, would be too restrictive so that only very simple prediction models could satisfy this requirement. A reasonable remedy is to require that $\phi(f, P_{n,\mathbf{w}}) < \epsilon$ only for $\mathbf{w}$ close to the equal weights.

Without loss of generality, we assume that $\sum_{i=1}^n w_i = 1$. When there is no sampling bias (i.e., $s(\cdot)$ is a constant function), we have $w_i = 1/n$. For given $\delta \in (0, 1)$, we let $\mathcal{W}_\delta = \left\{\mathbf{w} : w_i = (1-\delta)\frac{1}{n} + \delta v_i, (v_1, \ldots, v_n)^\top \in \mathcal{S}^n\right\}$, where $\mathcal{S}^n$ is the n dimensional simplex on $\mathbb{R}^n$. We propose the δ-robust fairness constraint as

$$\mathcal{F}_{\text{fair},\delta} = \left\{ f : \sup_{\mathbf{w} \in \mathcal{W}_\delta} \phi(f, P_{n,\mathbf{w}}) \leq \epsilon \right\}. \tag{1}$$

In turn, we propose to estimate f by $\hat{f}_{\text{fair},\delta}$ defined as

$$\hat{f}_{\text{fair},\delta} = \text{argmin}_{f \in \hat{\mathcal{F}}_{\text{fair},\delta}} \sum_{i=1}^n l(y_i, f(\mathbf{x}_i))/n. \tag{2}$$

The constant δ is a regularization parameter that could be proportional to the expected amount of sampling bias in practice.

4.2 L_q ROBUST FAIRNESS CONSTRAINTS

Solving (2) would be difficult even though it is not impossible in particular when $\phi(f, P_{n,\mathbf{w}})$ is nonlinear in $\mathbf{w}$. Note that most group fairness constraint functions are nonlinear. To resolve this obstacle, in this subsection, we propose a relaxed version of (1) such that the corresponding estimator can be obtained computationally easily.

Let $Q^{(n)}$ be a distribution on $\mathcal{S}^n$, which is specified later on. For given $q > 0$ we define the L_q δ-robust fairness constraint as

$$\mathcal{F}_{\text{fair},\delta,q} = \left\{ f : \{\mathbb{E}_{\mathbf{w}} \phi^q(f, P_{n,\mathbf{w}})\}^{1/q} \leq \epsilon \right\}, \tag{3}$$

where $\mathbf{w} = (1-\delta)(1/n, \ldots, 1/n)^\top + \delta \mathbf{v}$ and $\mathbf{v} \sim Q^{(n)}$. Then, we estimate f by $\hat{f}_{\text{fair},\delta,q}$ defined as $\hat{f}_{\text{fair},\delta,q} = \text{argmin}_{f \in \hat{\mathcal{F}}_{\text{fair},\delta,q}} \sum_{i=1}^n l(y_i, f(\mathbf{x}_i))/n$. Theoretically, the L_q robust fairness constraints becomes to the robust fairness constraints as q becomes larger since $\{\mathbb{E}_{\mathbf{w}} \phi^q(f, P_{n,\mathbf{w}})\}^{1/q} \to \sup_{\mathbf{w} \in \mathcal{W}_\delta} \phi(f, P_{n,\mathbf{w}})$ as $q \to \infty$ provided that the support of $Q^{(n)}$ is equal to $\mathcal{S}^n$. In this sense, the robust fairness constraint in (1) can be called the L_∞ robust fairness constraint. Our numerical study indicates that the L_q robust fairness constraint performs well for $q \in [1.5, 2]$ provided that $Q^{(n)}$ is selected carefully.

An important technical issue in the L_q robust fairness constraint is to choose $Q^{(n)}$. Even though the L_q robust fairness constraint converges to the L_∞ robust fairness constraint as $q \to \infty$, the robustness of $\hat{f}_{\text{fair},\delta,q}$ for a finite q strongly depends on the choice of $Q^{(n)}$. For given f, let $\mathbf{w}_f = \text{argmax}_{\mathbf{w} \in \mathcal{W}_\delta} \phi(f, P_{n,\mathbf{w}})$. In most fairness constraint functions, $\mathbf{w}_f$ locates at the boundary of $\mathcal{W}_\delta$ instead of the interior. The proof of this claim for the disparity impact constraint is given in Appendix A.1. Thus, it would be helpful to select $Q^{(n)}$ that puts most of its mass near the boundary of $\mathcal{W}_\delta$.

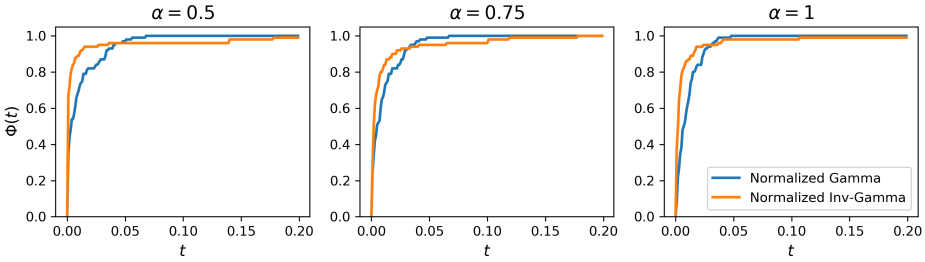


Figure 1: Comparison of the normalized gamma distribution and the normalized inverse-gamma distribution: the distribution functions of w_is.

For this purpose, we consider a normalized distribution defined as follows. Let G be a distribution on $[0, \infty)$ and let $R_1, \ldots, R_n$ be independent random variables following G. Then, the distribution of the random vector $\mathbf{W} = (W_1, \ldots, W_n)^\top$ defined as $W_i = R_i / \sum_{l=1}^n R_l$ is called the normalized G-distribution. If G is a Gamma distribution, then the normalized G-distribution becomes a Dirichlet distribution. The normalized inverse-Gaussian distribution is used for Bayesian analysis by Lijoi et al. (2005). By choosing G appropriately, we can have $Q^{(n)}$ that has desirable properties for our purpose.

For $\mathbf{w}$ to be close to the boundary of $\mathcal{W}_\delta$, most of w_is are close to 0, and the remaining fews are large. This would happen when there are few outliers among $R_1, \ldots, R_n$. That is, if the tail of G becomes fatter, the corresponding $Q^{(n)}$ has more mass near the boundary. Motivated by this observation, we propose to use the inverse-gamma distribution with the parameter (α, β) (i.e. the distribution of $1/D$, where D is a gamma random variable with parameter (α, β)) for G. Since β does not affect $Q^{(n)}$, we let $\beta = 1$. The inverse Gamma distribution has a polynomially decreasing tail, and does not even have the first moment when $\alpha \leq 1$. Figure 1 compares the normalized gamma and normalized inverse-gamma distributions on $\mathcal{S}^n$ with $n = 100$ as follows. We first generate $\mathbf{w}$ from $Q^{(n)}$ and draw the distribution $\Phi(t)$ defined as $\Phi(t) = \sum_{i=1}^n \mathbb{I}(w_i \leq t)$. Figure 1 compares the Φs of the normalized gamma distribution and the normalized inverse-gamma distribution for various values of α. Note that the normalized inverse-gamma distribution has more mass near 0 but also more mass on large values of w.

4.3 Learning with L_q robust fairness constraints

There are three regularization parameters δ, q and α in the L_q robust fairness constraint. The larger the δ is, the more robust the resulting prediction model is. Thus, we can control the fairness-robustness by choosing δ accordingly. On the other hand, even though larger q and smaller α increase fairness-robustness, the impact of q and α diminishes very fast. In addition, a too-large value of q makes the optimization problem numerically unstable. Our numerical studies indicate that the L_q robust fairness constraints with $\alpha \in (0.5, 1)$ and $q \in [1.5, 2]$ yield sufficiently fairness-robust prediction models. On the other hand, numerical instability occurs frequently when $q > 2$.

Once the three regularization parameters are selected, we estimate $\hat{f}_{\text{fair},\delta,q}$ by minimizing

$$\sum_{i=1}^n l(y_i, f(\mathbf{x}_i))/n + \lambda \left\{\mathbb{E}_{\mathbf{w}} \phi^q(f, P_{n,\mathbf{w}})\right\}^{1/q}, \tag{4}$$

where λ is the Lagrangian multiplier corresponding to ϵ. A standard gradient descent algorithm with approximating $\left(\mathbb{E}_{\mathbf{w}} \phi^q(f, P_{n,\mathbf{w}})\right)^{1/q}$ by a Monte-Carlo simulation works well.

4.4 Mini-batch learning algorithm

When training data are large, we frequently resort to a mini-batch learning algorithm, in which case we need a special technique to deal with the L_q robust fairness constraint. For this problem, we propose to modify the learning algorithms as follows. Let $\mathcal{D}$ be a mini-batch which is a subset of $\{1, \ldots, n\}$ with $|\mathcal{D}| = m$. Let $\mathbf{w}_\mathcal{D} = (w_{\mathcal{D},i}, i \in \mathcal{D})$ be a random vector following $Q_\delta^{(m)}$, where $Q_\delta^{(m)}$

is the distribution function of $(1-\delta)(1/m,\ldots,1/m)^\top + \delta\mathbf{v}$ and $\mathbf{v} \sim Q^{(m)}$. At each iteration of the mini-batch learning, we replace the L_q robust fairness constraint by $\{\mathbb{E}_{\mathbf{w}_\mathcal{D}}\phi^q(f, P_{\mathcal{D},\mathbf{w}_\mathcal{D}})\}^{1/q}$, where $P_{\mathcal{D},\mathbf{w}_\mathcal{D}}(\cdot) \propto \sum_{i\in\mathcal{D}} w_{\mathcal{D},i}\delta_{(y_i,\mathbf{x}_i,z_i)}(\cdot)$. Of course, $\{\mathbb{E}_{\mathbf{w}_\mathcal{D}}\phi^q(f, P_{\mathcal{D},\mathbf{w}_\mathcal{D}})\}^{1/q}$ can be approximated by a Monte Carlo simulation. The mini-batch learning algorithm for the L_q robust fairness constraint is summarized in Algorithm 1.

Algorithm 1 Mini-batch learning algorithm

1: **Initialize:**
   Model parameter $\boldsymbol{\theta}^0$.
2: **for** epoch $t = 1, 2, ..$ **do**
3: For given mini-batch $\mathcal{D}_t$ of size m, draw $\mathbf{w}_{\mathcal{D}_t,k} \sim Q_\delta^{(m)}$ for $k = 1,\ldots,K$.
4: Update $\boldsymbol{\theta}$:

$$\boldsymbol{\theta}^{t+1} \leftarrow \boldsymbol{\theta}^t - \eta\nabla_{\boldsymbol{\theta}}\left\{\frac{1}{m}\sum_{i\in\mathcal{D}_t} l(y_i, f_{\boldsymbol{\theta}}(\mathbf{x}_i)) + \lambda\left\{\frac{1}{K}\sum_{k=1}^{K}\phi^q(f_{\boldsymbol{\theta}}, P_{m,\mathbf{w}_{\mathcal{D}_t,k}})\right\}^{1/q}\right\} \tag{5}$$

5: **end for**

It should be noted that the above mini-batch learning algorithm is not a stochastic version of the original optimization problem (4). This is because the gradient in (5) is not an unbiased estimator of the gradient of (4). However, we can think of this mini-batch algorithm as a stochastic version of a new robust fairness constraint as follows. Let $\mathcal{A}$ be the collection of all subsets $\mathcal{D}$ of $\{1,\ldots,n\}$ with $|\mathcal{D}| = m$. Then, we define the mini-batch L_q robust fairness constraint as

$$\phi^{\text{mini}}(f) = \frac{1}{|\mathcal{A}|}\sum_{\mathcal{D}\in\mathcal{A}}\{\mathbb{E}_{\mathbf{w}_\mathcal{D}}\phi^q(f, P_{\mathcal{D},\mathbf{w}_\mathcal{D}})\}^{1/q}. \tag{6}$$

We have confirmed empirically that the mini-batch robust fairness constraint also works well whose details are given in Appendix A.4.

5 Numerical studies

To investigate the impact of sampling bias to the fairness of the trained classifier, we analyze four real-world datasets, which are popularly used in fairness AI research and publicly available: (i) The Adult Income dataset (Adult, Dua & Graff (2017)); (ii) The Bank Marketing dataset (Bank, Dua & Graff (2017)); (iii) The Law School dataset (Law school, Wightman & Ramsey (1998)); (iv) The Compas Propublica Risk Assessment dataset (COMPAS, Larson et al. (2016)). Except for the Adult dataset, which has separate training and test datasets, we obtain training and test datasets by splitting the dataset randomly with 8:2 ratio and repeat the training/test split 5 times for performance evaluation.

We consider linear logistic models for $\mathcal{F}$ and use the binary cross-entropy for the loss function. The results for deep neural networks are provided in Appendix A.4. We train the models by a stochastic gradient descent algorithm using PyTorch. The SGD optimizer is used with a momentum of 0.9 and a learning rate of 0.1. All experiments are conducted on a GPU server with NVIDIA TITAN Xp GPUs.

In this study, we focus on the three fairness constraints introduced in Section 2: the disparate impact (DI), the mean score parity (MSP), and the uniform individual fairness (UIF). We compare fairness-robust prediction models trained by our algorithm with those trained by standard in-processing methods (Goh et al., 2016; Wu et al., 2019) as well as unconstrained methods. For DI, we replace the indicator function in the fairness constraint with the hinge function. For choosing the Lagrange multiplier λ in (4), we first set a specific level ϵ in the fairness constraint (e.g., $\epsilon = 0.03$) and select λ so that the resulting prediction model $\hat{f}$ meets $\mathbb{E}_{\mathbf{w}}\phi^q(f, P_{n,\mathbf{w}}) \approx \epsilon^q$. For δ, the three values $\{0.2, 0.6, 1\}$ are considered. A similar procedure is used for choosing the Lagrangian parameter in the in-processing methods

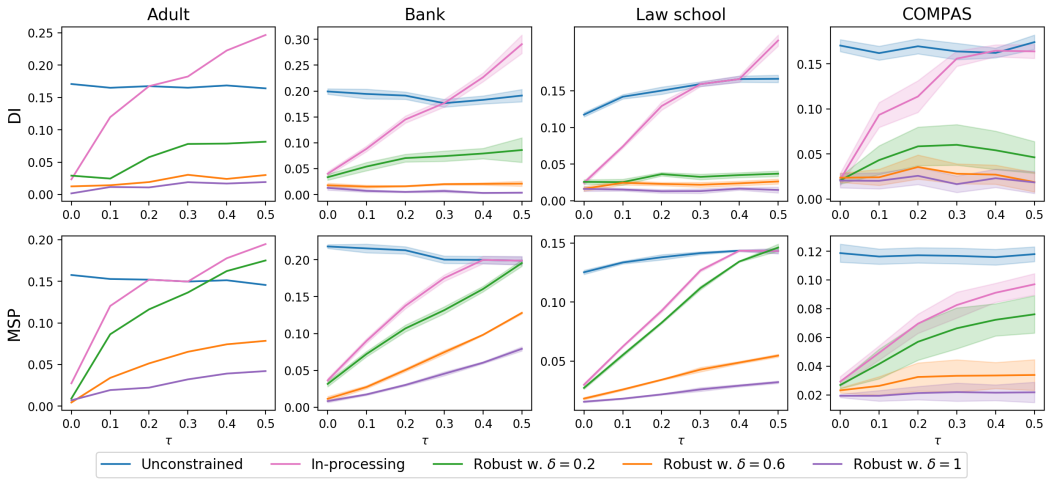


Figure 2: Comparison of the DI and MSP values of trained prediction model when sampling bias exists in training data. The colored bands for each line of the Bank, Law school and COMPAS datasets represent the pointwise 2-se (standard error) confidence intervals.

5.1 BIASED TRAINING DATA

We investigate the impact of sampling bias in training data to fairness by analyzing synthetically generated biased data from the four real-world datasets. We generate a biased training dataset as follows. (i) First, we fit a prediction model $\tilde{f}$ by use of the in-processing method with a specific level of fairness (e.g., $\phi(\tilde{f}, P_n^{\text{tr}}) \approx 0.03$). (ii) We compute the worst case weight $\tilde{\mathbf{w}}$ defined as $\tilde{\mathbf{w}} = \text{argmax}_{\mathbf{w}\in\mathcal{B}}\phi(\tilde{f}, P_{n,\mathbf{w}}^{\text{tr}})$ for a subset $\mathcal{B}$ of $\mathcal{S}^n$. We select $\mathcal{B}$, whose specification is given in Appendix A.2. (iii) For given $\tau \in [0,1]$, we set $\mathbf{w}_\tau = (1-\tau)\mathbf{w}_0 + \tau\hat{\mathbf{w}}$, where $\mathbf{w}_0 = (1/n, \ldots, 1/n)^\top$ and $\hat{\mathbf{w}} \propto 1/\tilde{\mathbf{w}}$. (iv) Finally, we generate a synthetic biased training dataset by sampling from the original training dataset with probability $\mathbf{w}_\tau$. Note that a large value of τ results in a large sampling bias in synthetic training data.

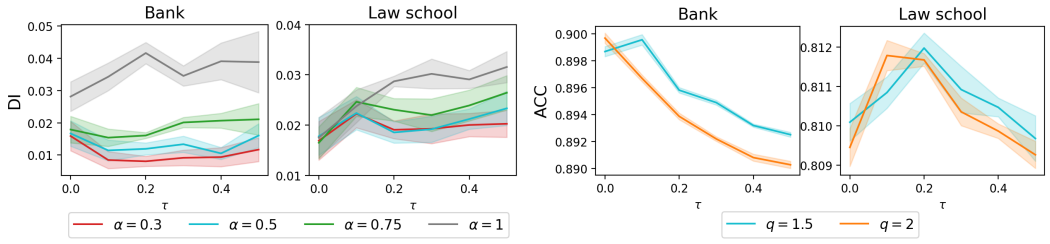
Performance assessment: For a given τ, we train prediction models based on a synthetically biased training data and evaluate the fairness values (i.e., the values of the fairness function of the trained prediction models) and the accuracies on the test data. The results of the fairness values of various trained prediction models for $\tau \in \{0.1, 0.2, \ldots, 0.5\}$ are compared in Figure 2, where the parameter α and q are set to be 0.75 and 1.5, respectively. As we expect, the L_q robust fairness constraint improves the fairness-robustness significantly. A large value of δ improves the fairness-robustness much. Of course, the accuracies, which are reported in Appendix A.4, decreases as δ increases. In practice, we could choose δ that trades off the accuracy and fairness-robustness optimally.

Sensitivity analysis of α and q: We investigate how the choice of α and q affects the fairness-robustness. The left panel of Figure 3 compares the DI values of the trained prediction models with various values of α when δ and q are fixed at 0.6 and 1.5, respectively. The results are similar except the case of $\alpha = 1$. The choice of α being 0.75 or 0.5 looks reasonable.

The right panel of Figure 3 compares the accuracies of the trained models with the L_q robust DI constraint for $q = 1.5$ and $q = 2$ whose DI values on test data are near 0.03, where δ and α are set to be 0.6 and 0.75, respectively. The accuracies with $q = 1.5$ are higher than those with $q = 2$ for the Bank dataset while they are similar for the Law school dataset. Similar results are observed for MSP, which are given in Appendix A.4.

5.2 BIASED TEST DATA

Performance assessment for DI and MSP: We train a model with the original training dataset and evaluate the fairness of the prediction model on the weighted test dataset. For a given prediction



(a) Sensitive analysis of α

(b) Sensitive analysis of q

Figure 3: Sensitive analysis for the L_q robust fairness algorithm. (a) We compare the test DI values for sensitive analysis of α. We fix $q = 1.5$ and $\delta = 0.6$. (b) For sensitive analysis of q, we compare the test accuracy when $q = 1.5$ and $q = 2$. We fix $\delta = 0.6$ and $\alpha = 0.5$. The tuning parameter λ is selected when the test DI value is similar to 0.03. In each figure, the colored bands represent the pointwise 2-se confidence intervals.

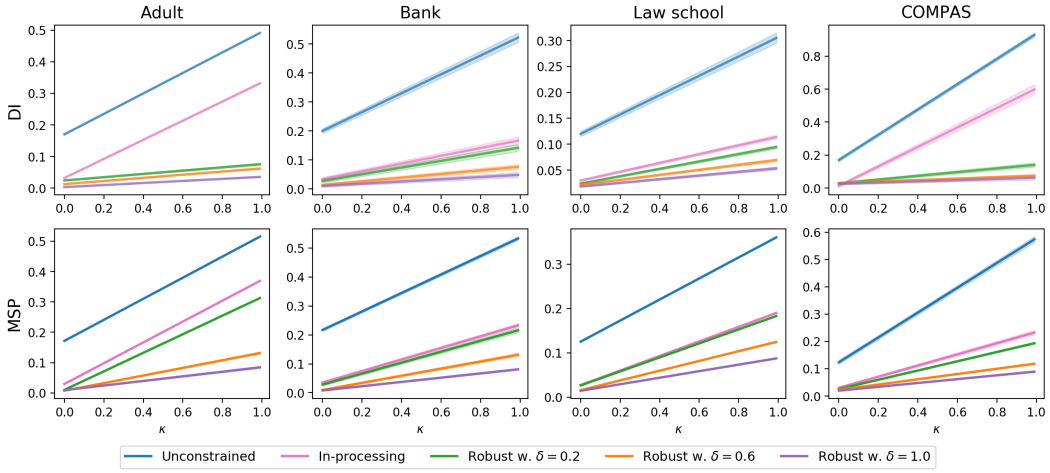
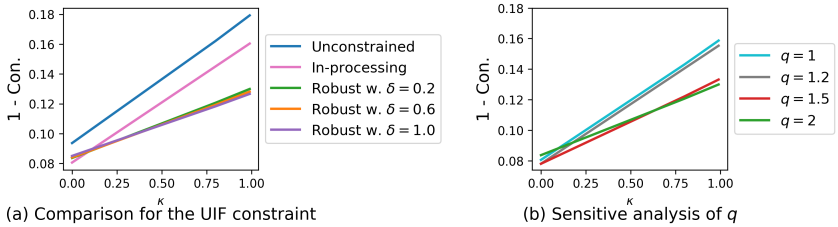


Figure 4: Comparison of the DI and MSP values of trained prediction model when sampling bias exists in test data. The colored bands for each line of the Bank, Law school and COMPAS datasets represent the pointwise 2-se confidence intervals.

model f, we evaluate $\phi(f, P^{\text{te}}_{n,\hat{\mathbf{w}}_\kappa})$ for various values of κ, where $\hat{\mathbf{w}}_\kappa$ is the one which maximizes $\phi(f, P^{\text{te}}_{n,\mathbf{w}})$ among those satisfying $\|\mathbf{w} - \mathbf{w}_0\|_\infty \leq \kappa$. See Appendix A.3 for the detailed procedure to select $\hat{\mathbf{w}}_\kappa$. Figure 4 summarizes the results of DI and MSP for various values of δ with (α, q) being $(0.75, 1.5)$, which amply indicate that the L_q robust fairness constraint performs quite well for the case of sampling bias in test data.

Performance assessment for UIF: The left panel of Figure 5 compares the $1 - \text{Con.}$ values of the robustly trained prediction models with $\alpha = 0.75$ and $q = 2$ as well as trained models by the in-processing and unconstrained methods. Con. (*consistency*) is a measure of the consistency between $f(\mathbf{x})$ and $f(\mathbf{x}')$ when $\mathbf{x}$ and $\mathbf{x}'$ are the same except the sensitive variable. See Yurochkin & Sun (2020); Mukherjee et al. (2020) for the detailed definition of Con. The results confirm that the L_q robust fairness constraint also performs well for UIF. The right panel of Figure 5 investigates the sensitivity of the L_q robust UIF constraint to the choice q while δ and α are fixed at 0.2 and 0.75, respectively. Note that UIF is linear in $\mathbf{w}$ and hence the L_q robust UIF does not work for $q = 1$. The results suggest that $q \in [1.5, 2]$ would be a reasonable choice. More results of numerical studies for UIF are given in Appendix A.4.

Comparison with the L_∞ robust fairness constraint: We compare the L_q and L_∞ constraints when $\delta = 1$. For the L_∞ constraint, we use the algorithm of Mandal et al. (2020). For prediction



(a) Comparison for the UIF constraint (b) Sensitive analysis of q

Figure 5: Comparison of the fairness values and sensitive analysis of q for UIF based on the Adult dataset when sampling bias exists in test data.

models, we consider linear models since the algorithm of Mandal et al. (2020) is only available for linear models.

The results are presented in Appendix A.5. The L_q with $q = 1.5$ and L_∞ robust fairness constraints perform similarly when training data are balanced in the output labels (i.e. $n_0 = n_1$ where n_y is the number of data whose labels are y) while the L_q constraint is sightly superior when training data are imbalanced (e.g. $n_0 = 3n_1$). In addition, computation time of our algorithm is much faster (more than 40 times faster for the Adult dataset) than that of Mandal et al. (2020). These results suggest that the proposed learning algorithm with the L_q robust fairness constraint can be understood as a computationally efficient proxy of the learning algorithm with the L_∞ robust fairness constraint.

6 Conclusion

In this paper, we have focused on fairness-robustness. We do not claim that our method is best in the sense that it achieves the best accuracy among fairness-robust prediction models. Theoretical studies for this direction would be worth pursuing.

When the bias function s depends on the output label y, the situation becomes much more difficult. A representative example of such cases is the case-control design, where we randomly select samples from each class. Within the author's knowledge, not much study about the fairness of y-biased data has been done. We leave this topic as future work.

There are many possible areas where the idea of the L_q robust fairness can be applied. Obviously, we have only considered in-processing methods. Robustifying pre-processing and post-processing methods are also interesting, and the L_q robust fairness could be modified for this purpose. Another area would be Distributionally robust optimization (DRO), where a prediction model is trained by minimizing the worst-case training loss. The L_q robust fairness constraint can be applied to DRO which may yield computationally efficient learning algorithms.

References

Julia Angwin, Jeff Larson, Surya Mattu, and Lauren Kirchner. Machine bias. *ProPublica, May*, 23: 2016, 2016.

Solon Barocas and Andrew D Selbst. Big data's disparate impact. *Calif. L. Rev.*, 104:671, 2016.

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

Amanda Coston, Karthikeyan Natesan Ramamurthy, Dennis Wei, Kush R Varshney, Skyler Speakman, Zairah Mustahsan, and Supriyo Chakraborty. Fair transfer learning with missing protected attributes. In *Proceedings of the 2019 AAAI/ACM Conference on AI, Ethics, and Society*, pp. 91–98, 2019.

Michele Donini, Luca Oneto, Shai Ben-David, John S Shawe-Taylor, and Massimiliano Pontil. Empirical risk minimization under fairness constraints. In *Advances in Neural Information Processing Systems*, pp. 2791–2801, 2018.

Dheeru Dua and Casey Graff. UCI machine learning repository, 2017. URL http://archive.ics.uci.edu/ml.

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

Riccardo Fogliato, Alexandra Chouldechova, and Max G'Sell. Fairness evaluation in presence of biased noisy labels. In *International Conference on Artificial Intelligence and Statistics*, pp. 2325–2336. PMLR, 2020.

Gabriel Goh, Andrew Cotter, Maya Gupta, and Michael P Friedlander. Satisfying real-world goals with dataset constraints. In *Advances in Neural Information Processing Systems*, pp. 2415–2423, 2016.

Moritz Hardt, Eric Price, and Nati Srebro. Equality of opportunity in supervised learning. In *Advances in neural information processing systems*, pp. 3315–3323, 2016.

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

Daniel G Horvitz and Donovan J Thompson. A generalization of sampling without replacement from a finite universe. *Journal of the American statistical Association*, 47(260):663–685, 1952.

Ray Jiang, Aldo Pacchiano, Tom Stepleton, Heinrich Jiang, and Silvia Chiappa. Wasserstein fair classification. In *Uncertainty in Artificial Intelligence*, pp. 862–872. PMLR, 2020.

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

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

Jon Kleinberg, Jens Ludwig, Sendhil Mullainathan, and Ashesh Rambachan. Algorithmic fairness. In *Aea papers and proceedings*, volume 108, pp. 22–27, 2018.

Alexandre Louis Lamy, Ziyuan Zhong, Aditya Krishna Menon, and Nakul Verma. Noise-tolerant fair classification. *arXiv preprint arXiv:1901.10837*, 2019.

Jeff Larson, Surya Mattu, Lauren Kirchner, and Julia Angwin. How we analyzed the COMPAS recidivism algorithm. *ProPublica (5 2016)*, 9(1), 2016.

Antonio Lijoi, Ramsés H Mena, and Igor Prünster. Hierarchical mixture modeling with normalized inverse-gaussian priors. *Journal of the American Statistical Association*, 100(472):1278–1291, 2005.

Debmalya Mandal, Samuel Deng, Suman Jana, Jeannette Wing, and Daniel J Hsu. Ensuring fairness beyond the training data. In H. Larochelle, M. Ranzato, R. Hadsell, M. F. Balcan, and H. Lin (eds.), *Advances in Neural Information Processing Systems*, volume 33, pp. 18445–18456. Curran Associates, Inc., 2020. URL https://proceedings.neurips.cc/paper/2020/file/d6539d3b57159babf6a72e106beb45bd-Paper.pdf.

Ninareh Mehrabi, Fred Morstatter, Nripsuta Saxena, Kristina Lerman, and Aram Galstyan. A survey on bias and fairness in machine learning. *arXiv preprint arXiv:1908.09635*, 2019.

Aditya Krishna Menon and Robert C Williamson. The cost of fairness in binary classification. In *Conference on Fairness, Accountability and Transparency*, pp. 107–118, 2018.

Debarghya Mukherjee, Mikhail Yurochkin, Moulinath Banerjee, and Yuekai Sun. Two simple ways to learn individual fairness metrics from data. In *Proceedings of the 37th International Conference on Machine Learning*, pp. 7097–7107, 2020.

Manisha Padala and Sujit Gujar. FNNC: Achieving Fairness through Neural Networks. pp. 2249–2255, 07 2020. doi: 10.24963/ijcai.2020/311.

Bahar Taskesen, Viet Anh Nguyen, Daniel Kuhn, and Jose Blanchet. A distributionally robust approach to fair classification. *arXiv preprint arXiv:2007.09530*, 2020.

Serena Wang, Wenshuo Guo, Harikrishna Narasimhan, Andrew Cotter, Maya Gupta, and Michael I Jordan. Robust optimization for fairness with noisy protected groups. *arXiv preprint arXiv:2002.09343*, 2020.

Dennis Wei, Karthikeyan Natesan Ramamurthy, and Flavio Calmon. Optimized Score Transformation for Fair Classification. volume 108 of *Proceedings of Machine Learning Research*, pp. 1673–1683, Online, 26–28 Aug 2020. PMLR. URL http://proceedings.mlr.press/v108/wei20a.html.

Linda F Wightman and Henry Ramsey. *LSAC national longitudinal bar passage study*. Law School Admission Council, 1998.

Yongkai Wu, Lu Zhang, and Xintao Wu. On convexity and bounds of fairness-aware classification. In *The World Wide Web Conference*, pp. 3356–3362, 2019.

Mikhail Yurochkin and Yuekai Sun. SenSeI: Sensitive Set Invariance for Enforcing Individual Fairness, 2020.

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

Muhammad Bilal Zafar, Isabel Valera, Manuel Gomez-Rodriguez, and Krishna P Gummadi. Fairness Constraints: A Flexible Approach for Fair Classification. *J. Mach. Learn. Res.*, 20(75):1–42, 2019.

A Appendix

A.1 Location of $\mathbf{w}_f$ for the DI constraint

Recall that

$$\mathbf{w}_f := \mathrm{argmax}_{\mathbf{w}\in\mathcal{W}_\delta}\phi(f, P_{n,\mathbf{w}})$$

as the "worst" weight with respect to the constraint function $\phi(\cdot)$ with given function f and a sample set with size n.

Proposition 1 (Location of $\mathbf{w}_f$)**.** *Suppose that the four sets* $\{i : f(\mathbf{x}_i) > 0, z_i = z\}$ *and* $\{i : f(\mathbf{x}_i) \leq 0, z_i = z\}$ *for* $z \in \{0, 1\}$ *are all nonempty. Then*

$$\mathbf{w}_f \in \partial\mathcal{W}_\delta,$$

for any $\delta > 0$*, where* ∂A *is the boundary of a set* A*.*

Proof. For given $\mathbf{w} \in \mathcal{W}_\delta$, let

$$\tilde{\phi}(f, P_{n,\mathbf{w}}) = \frac{\sum_{i=1}^n w_i\mathbb{I}(f(\mathbf{x}_i) > 0)\mathbb{I}(z_i = 0)}{\sum_{i=1}^n w_i\mathbb{I}(z_i = 0)} - \frac{\sum_{j=1}^n w_j\mathbb{I}(f(\mathbf{x}_j) > 0)\mathbb{I}(z_j = 1)}{\sum_{j=1}^n w_j\mathbb{I}(z_j = 1)}.$$

Note that $\phi(f, P_{n,\mathbf{w}}) = |\tilde{\phi}(f, P_{n,\mathbf{w}})|$.

Suppose that $\mathbf{w}_f$ is an interior point of $\mathcal{W}_\delta$. Without loss of generality, we assume that $\tilde{\phi}(f, P_{n,\mathbf{w}_f}) > 0$. Choose $i_1 \in \{i : f(\mathbf{x}_i) > 0, z_i = 0\}, i_2 \in \{i : f(\mathbf{x}_i) \leq 0, z_i = 0\}$ and $l_1 \in \{i : f(\mathbf{x}_i) > 0, z_i = 1\}, l_2 \in \{i : f(\mathbf{x}_i) \leq 0, z_i = 1\}$. Since $\mathbf{w}_f$ is an interior point of $\mathcal{W}_\delta$, we can choose $\gamma > 0$ such that $\tilde{\mathbf{w}}$ defined as $\tilde{w}_k = w_{f,k}$ for $k \notin \{i_1, i_2, l_1, l_2$ and $\tilde{w}_{i_1} = w_{f,i_1} + \gamma, \tilde{w}_{i_2} = w_{f,i_2} - \gamma, \tilde{w}_{l_1} = w_{f,l_1} - \gamma, \tilde{w}_{l_2} = w_{f,l_2} + \gamma$, also belongs to $\mathcal{W}_\delta$. However, it holds that $\tilde{\phi}(f, P_{n,\tilde{\mathbf{w}}}) > \tilde{\phi}(f, P_{n,\mathbf{w}_f})$, which contradicts the definition of $\mathbf{w}_f$. Thus, $\mathbf{w}_f$ should be located at the boundary of $\mathcal{W}_\delta$. □

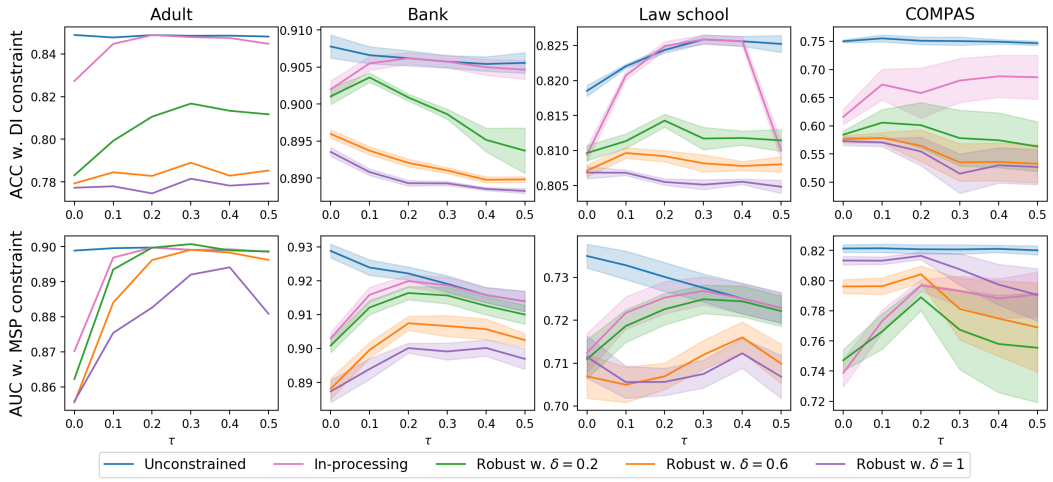


Figure 6: Comparison of the prediction performance when sampling bias exists in training data. Classification accuracies are presented for prediction models trained with the DI constraint, and AUCs are presented for prediction models trained with the MSP constraint. The colored bands for each line of the Bank, Law school and COMPAS datasets represent the pointwise 2-se confidence intervals.

A.2 The set $\mathcal{B}$ used in Section 5.1

Let $\pi_0 = n_0/n$ and $\pi_1 = n_1/n$, where $n_0 = \sum_{i=1}^n \mathbb{I}(z_i = 0)$ and $n_1 = \sum_{i=1}^n \mathbb{I}(z_i = 1)$. Define the set $\mathcal{B}_\kappa$ for $\kappa \in [0, 1]$ as

$$\mathcal{B}_\kappa = \left\{ \mathbf{w} \in \mathcal{S}^n : \|\mathbf{w} - \mathbf{w}_0\| \le \kappa/n, \sum_{i=1}^n w_i \mathbb{I}(z_i = 0) = \pi_0, \sum_{i=1}^n w_i \mathbb{I}(z_i = 1) = \pi_1 \right\}. \quad (7)$$

We let $\mathcal{B} = \mathcal{B}_{0.99}$. The set $\mathcal{B}_\kappa$ is considered by Mandal et al. (2020) for their numerical studies.

A.3 The weight $\hat{\mathbf{w}}_\kappa$ in Section 5.2

The $\hat{\mathbf{w}}_\kappa$ is defined as

$$\hat{\mathbf{w}}_\kappa = \mathrm{argmax}_{\mathbf{w} \in \mathcal{B}_\kappa} \phi(f, P^{\mathrm{te}}_{n,\mathbf{w}}),$$

where $\mathcal{B}_\kappa$ is defined in (7).

A.4 Omitted results

Tradeoff between robust fairness and prediction performance: It was shown that large δ improves fairness-robustness much whose results are presented in Figure 2 and 4. In this subsection, we investigate how the prediction performance is affected by the choice of δ. For the DI constraint, we measure classification accuracy, while AUC is measured for the MSP constraint. Figures 6 and 7 summarize the results for the cases where sampling bias exists in training and test data, respectively. Generally, large δ negatively affects the prediction performances. However, the AUC is less affected and even is improved with a larger δ for the COMPAS dataset. There would be more interesting stories in the relation of fairness-robustness and score estimation.

Sensitivity analysis of α and q for the MSP constraint: We perform the sensitivity analysis for the MSP constraint as is done for the DI constraint in Figure 3. The left panel of Figure 8 presents the MSP values of the trained prediction models with $\alpha \in \{0.3, 0.5, 0.75, 1.\}$. As expected, the prediction model trained with smaller α is more robust. Similarly to the DI constraint, 0.75 or 0.5 looks reasonable for the choice of α. The right panel of Figure 8 presents the AUC values of the trained prediction models with the MSP constraint for $q = 1.5$ and $q = 2$. Here, δ and α are set to be 0.6 and 0.75. For the Bank dataset, the prediction model trained with $q = 1.5$ is more accurate than

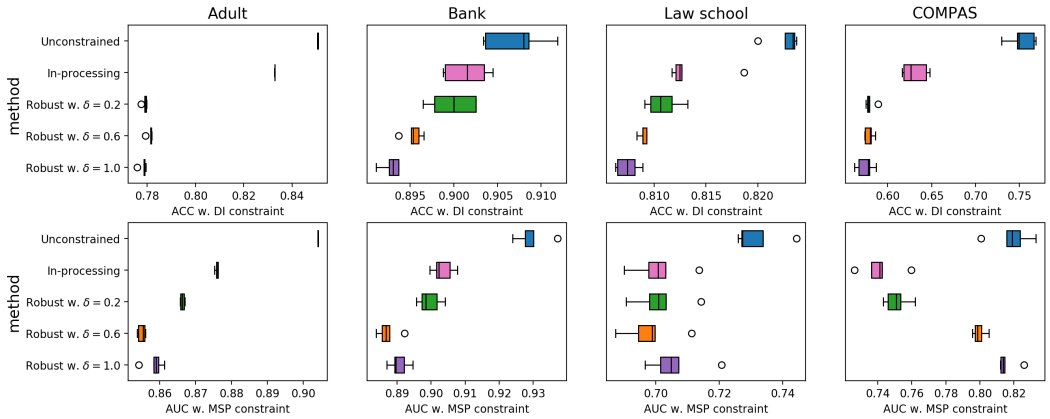
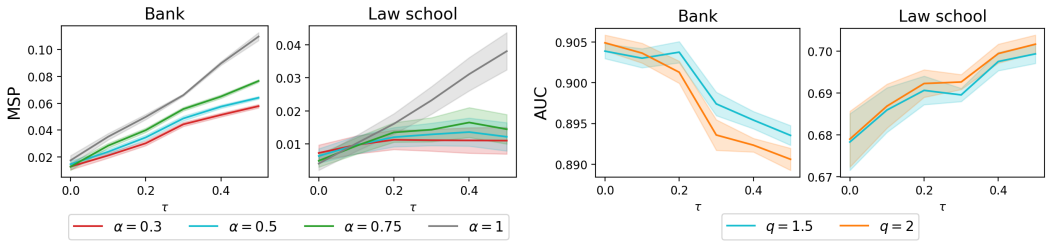


Figure 7: Comparison of the prediction performance when sampling bias exists in test data. Classification accuracies are presented for prediction models trained with the DI constraint, and AUCs are presented for prediction models trained with the MSP constraint.



(a) Sensitive analysis of α

(b) Sensitive analysis of q

Figure 8: Sensitive analysis for the L_q robust fairness algorithm with the MSP constraint. (a) We compare the test MSP values for sensitive analysis of α. We fix $q = 1.5$ and $\delta = 0.6$. (b) For sensitive analysis of q, we compare the test AUC when $q = 1.5$ and $q = 2$. We fix $\delta = 0.6$ and $\alpha = 0.5$. The tuning parameter λ is selected when the test MSP value is similar to 0.03. In each figure, the colored bands represent the pointwise 2-se confidence intervals.

that with $q = 2$, while there is no big difference for the Law school dataset. We believe that $q = 1.5$ would be a reasonable choice for most practical purposes.

More experiments for UIF: Figures 9 and 10 present complementary results for Figure 5. The implications obtained from Figure 5 which is for the Adult dataset are still valid for the other three datasets. Furthermore, when sampling bias exists in training data, we also demonstrate the robustness of our proposed method as shown in Figure 11 and 12.

Comparison of the batch learning and mini-batch learning: Figure 13 compares the results obtained by the batch learning and mini-batch learning with the DI constraint when sampling bias exists in test data. The parameters (α, δ, q) are set to be $(0.75, 0.6, 1.5)$ and the batch size is set to 2000. The DI values and accuracies of the two learning methods are similar indicating that the mini-batch learning algorithm also works well.

Results for deep neural networks: We also investigate the performance of DNN models trained with robust fairness algorithm. We use the DNN model with two hidden layers of the same dimension as the input vector. Figure 14-17 show patterns similar to those for the logistic regression model, but DNN models have better prediction performances in most cases.

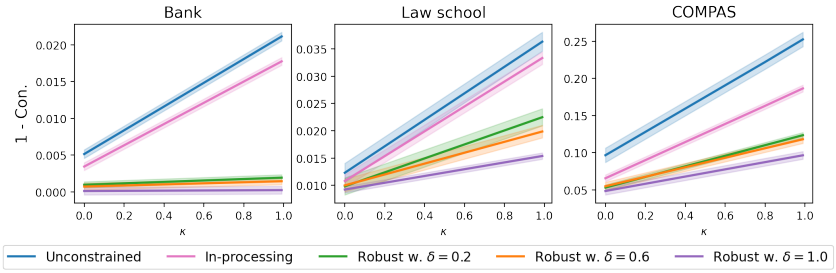


Figure 9: Comparison of the fairness values for UIF based on the Bank, Law school and COMPASS datasets when sampling bias exists in test data. The colored bands for each line of the Bank, Law school and COMPAS datasets represent the pointwise 2-se confidence intervals.

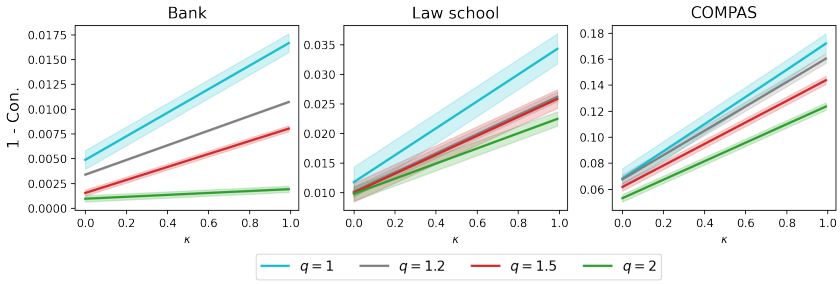


Figure 10: Sensitive analysis of q for UIF based on the Bank, Law school, and COMPAS datasets when sampling bias exists in test data. The colored bands for each line of the Bank, Law school and COMPAS datasets represent the pointwise 2-se confidence intervals.

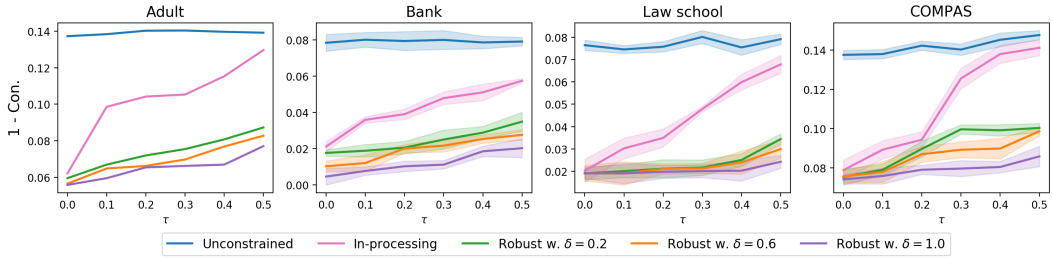


Figure 11: Comparison of the fairness values for UIF based on the Adult, Bank, Law school, and COMPAS dataset when sampling bias exists in training data. The colored bands for each line of the Bank, Law school and COMPAS datasets represent the pointwise 2-se confidence intervals

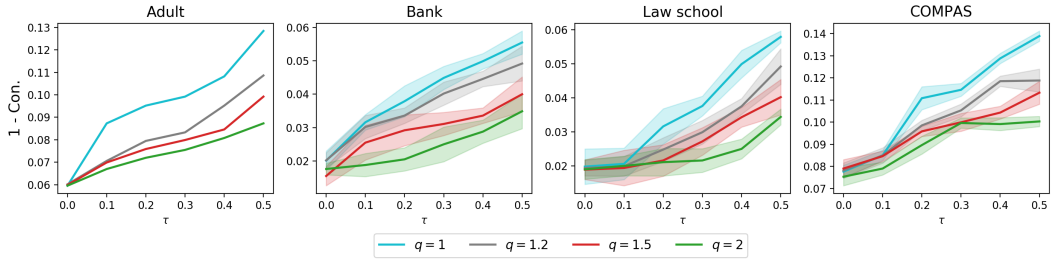


Figure 12: Sensitive analysis of q for UIF based on the Adult, Bank, Law school, and COMPAS dataset when sampling bias exists in training data. The colored bands for each line of the Bank, Law school and COMPAS datasets represent the pointwise 2-se confidence intervals

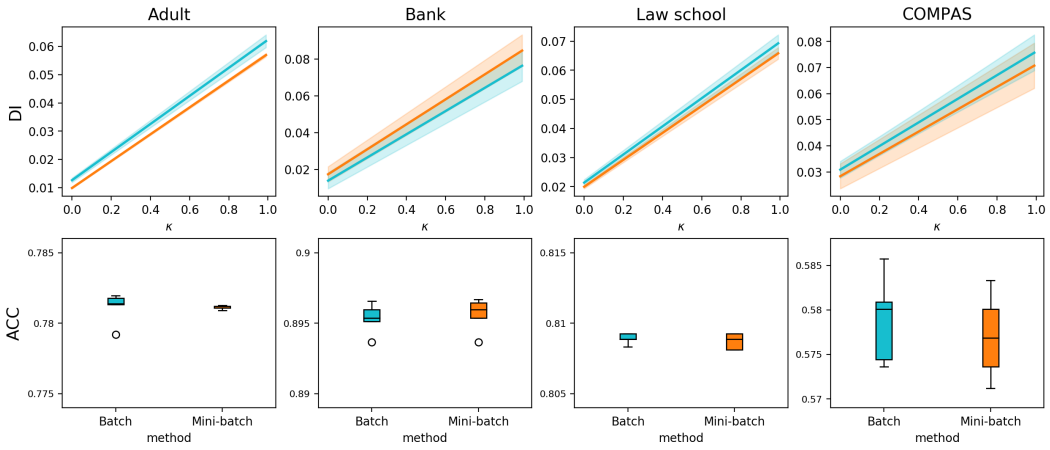


Figure 13: Comparison of the batch learning and mini-batch learning. The colored bands for each line are the pointwise 2-se confidence intervals.

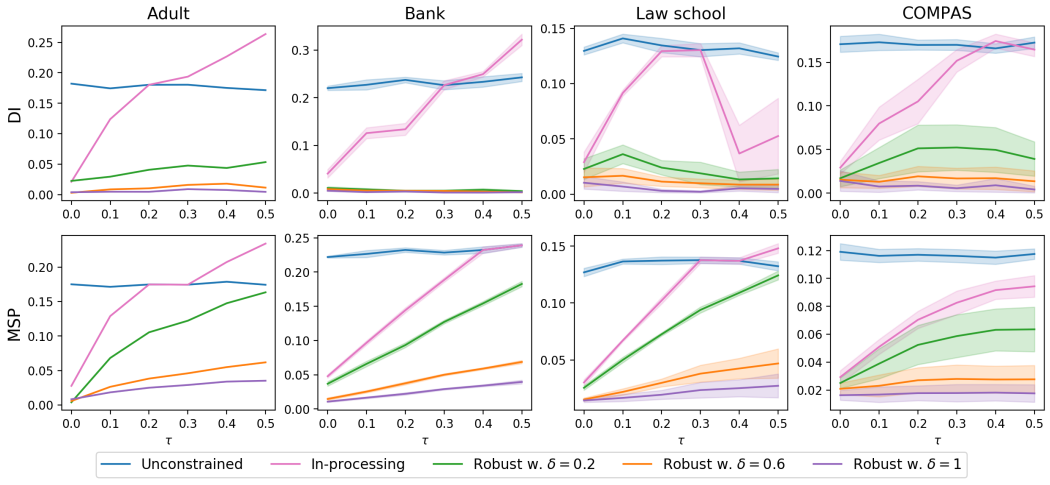


Figure 14: Comparison of the DI and MSP values of trained DNN models when sampling bias exists in training data. The colored bands for each line of the Bank, Law school and COMPAS datasets represent the pointwise 2-se confidence intervals.

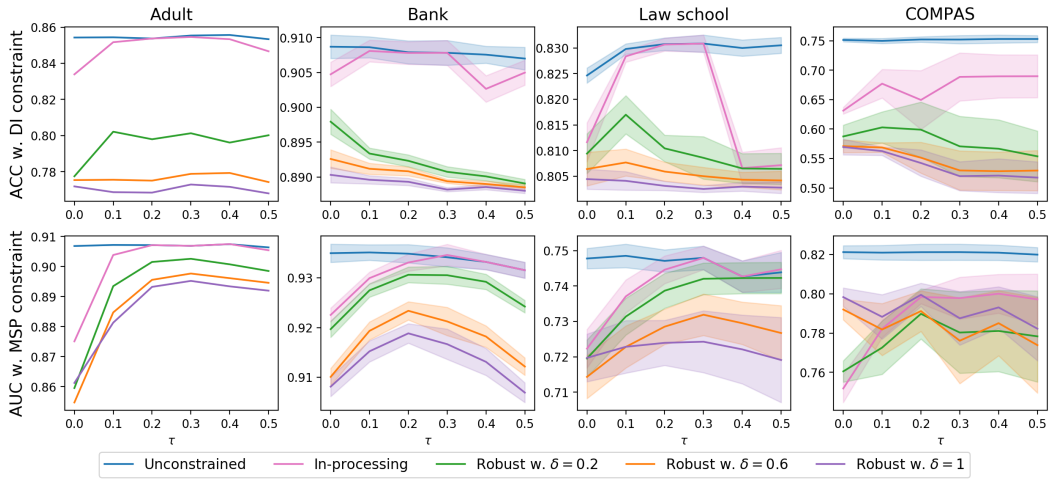


Figure 15: Comparison of the prediction performances of trained DNN models under the robust fairness constraint when sampling bias exists in training data. Classification accuracies are presented for DNN models trained with the DI constraint, and AUCs are presented for DNN models trained with the MSP constraint. The colored bands for each line of the Bank, Law school and COMPAS datasets represent the pointwise 2-se confidence intervals.

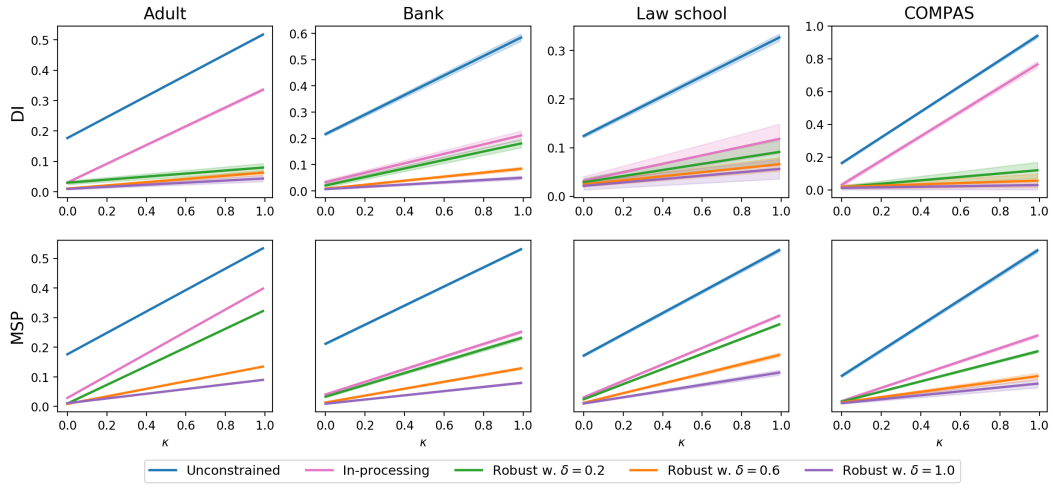


Figure 16: Comparison of the DI and MSP values of trained DNN models when sampling bias exists in test data. The colored bands for each line of the Bank, Law school and COMPAS datasets represent the pointwise 2-se confidence intervals.

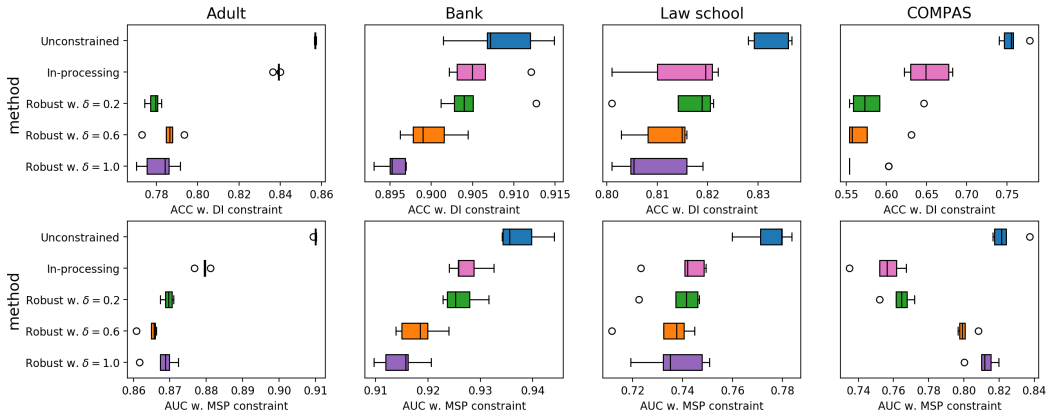


Figure 17: Comparison of the prediction performance of DNN models when sampling bias exists in test data. Classification accuracies are presented for DNN models trained with DI constraint, and AUCs are presented for DNN models trained with MSP constraint. The colored bands for each line of the Bank, Law school and COMPAS datasets represent the pointwise 2-se confidence intervals.

Table 1: The means and standard errors of computation times for the balanced Adult dataset.

Method	**Computation time**
L_∞	318.80 ± 2.45
L_q	7.83 ± 0.09

A.5 Results for comparing the L_q and L_∞ robust fairness constraints

Brief summary of the algorithm of Mandal et al. (2020): The algorithm of Mandal et al. (2020) solves (2) directly when $\delta = 1$. It first considers a Lagrangian form of (2) given as

$$\sum_{i=1}^{n} l(y_i, f(\mathbf{x}_i)) + \sum_{\mathbf{w} \in \mathcal{W}_1} \lambda_{\mathbf{w}} \phi(f, P_{n,\mathbf{w}}). \tag{8}$$

Then, it learns f by minimizing (8) with respect to $f \in \mathcal{F}$ and maximizing (8) with respect to $\lambda_{\mathbf{w}}$ subject to $\sup_{\mathbf{w} \in \mathcal{W}_1} |\lambda_{\mathbf{w}}| \leq B$ for some $B > 0$ iteratively until convergence. Though the algorithm is theoretically well founded, computational burden would be large. First of all, the algorithm requires to learn f iteratively and hence is hard to be applied to computationally intensive models such as deep neural networks. Only the linear logistic regression is considered in Mandal et al. (2020). In addition, maximization of (8) with respect to $\lambda_{\mathbf{w}}$ needs to solve a linear programming problem on $\mathcal{W}_1$, which would be computationally demanding when n is large (and thus $\mathcal{W}_1$ becomes a large dimensional simplex).

Results: We consider two cases for training data - balanced and imbalanced ones with respect to the label distribution. For given ratio $\nu = n_0/n_1$, we sample 2000 many samples from the original training data whose label ratio is equal to ν. We select the regularization parameters as follows. We first tune B in the algorithm of Mandal et al. (2020) so that the trained prediction model has sufficiently small DI values for biased test data with $\kappa = 1$. Then, we tune the regularization parameters α and λ in our algorithm so that the DI values of the trained prediction model on biased test datasets with $\kappa = 0$ and $\kappa = 1$ is similar to those of the model trained by the algorithm of Mandal et al. (2020). Then, we compare the prediction accuracies of the two prediction models on the test dataset with $\kappa = 0$. Figure 18 shows that the accuracies of the two prediction models are similar, while Figure 19 indicates that our algorithm is slightly superior for imbalanced cases. In addition, we compare the computation time of the two algorithms in Table 1 which shows that our robust fairness algorithm is much faster.

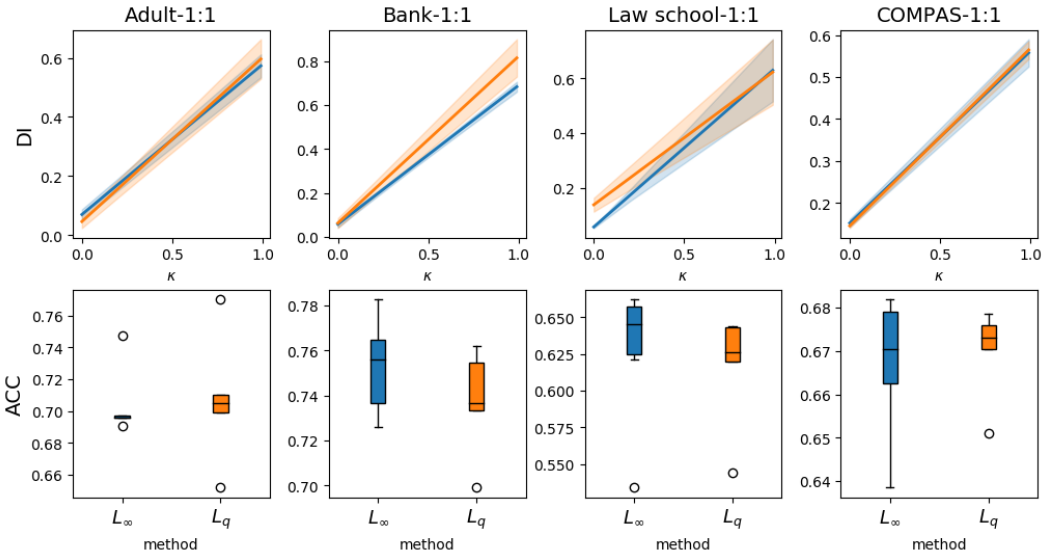


Figure 18: Comparison of the L_q and L_∞ constraints for balanced data cases. The colored bands for each line are the pointwise 2-se confidence intervals.

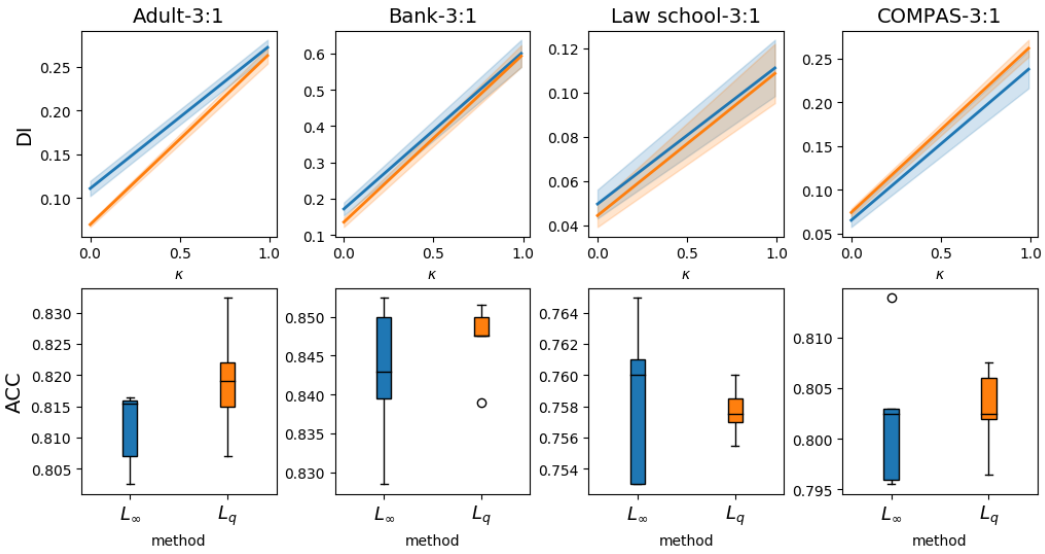


Figure 19: Comparison of the L_q and L_∞ constraints for imbalanced data cases. The colored bands for each line are the pointwise 2-se confidence intervals.