Differentially Private Fair Learning

Matthew Jagielski [1] **Michael Kearns** [2] **Jieming Mao** [2] **Alina Oprea** [1] **Aaron Roth** [2] **Saeed Sharifi-Malvajerdi** [2] **Jonathan Ullman** [1]

Abstract

Motivated by settings in which predictive models may be required to be non-discriminatory with respect to certain attributes (such as race), but even collecting the sensitive attribute may be forbidden or restricted, we initiate the study of fair learning under the constraint of differential privacy. Our first algorithm is a private implementation of the equalized odds post-processing approach of (Hardt et al., 2016). This algorithm is appealingly simple, but must be able to use protected group membership explicitly at test time, which can be viewed as a form of "disparate treatment". Our second algorithm is a differentially private version of the oracle-efficient in-processing approach of (Agarwal et al., 2018) which is more complex but need not have access to protected group membership at test time. We identify new tradeoffs between fairness, accuracy, and privacy that emerge only when requiring all three properties, and show that these tradeoffs can be milder if group membership may be used at test time. We conclude with a brief experimental evaluation.

1. Introduction

Large-scale algorithmic decision making, often driven by machine learning on consumer data, has increasingly run afoul of various social norms, laws and regulations. A prominent concern is when a learned model exhibits discrimination against some demographic group, perhaps based on race or gender. Concerns over such algorithmic discrimination have led to a recent flurry of research on fairness in machine learning, which includes both new tools and methods for designing fair models, and studies of the tradeoffs between predictive accuracy and fairness (ACM, 2019).

At the same time, both recent and longstanding laws and regulations often restrict the use of "sensitive" or protected attributes in algorithmic decision-making. U.S. law prevents the use of race in the development or deployment of consumer lending or credit scoring models, and recent provisions in the E.U. General Data Protection Regulation (GDPR) restrict or prevent even the collection of racial data for consumers. These two developments — the demand for non-discriminatory algorithms and models on the one hand, and the restriction on the collection or use of protected attributes on the other — present technical conundrums, since the most straightforward methods for ensuring fairness generally require knowing or using the attribute being protected. It seems difficult to guarantee that a trained model is not discriminating against (say) a racial group if we cannot even identify members of that group in the data.

A recent line of work (Veale & Binns, 2017; Kilbertus et al., 2018) made these cogent observations, and proposed an interesting solution employing the cryptographic tool of *secure multiparty computation* (commonly abbreviated *MPC*). In this model, we imagine a commercial entity with access to consumer data that excludes race, but this entity would like to build a predictive model for, say, commercial lending, under the constraint that the model be non-discriminatory by race with respect to some standard fairness notion (e.g. equality of false rejection rates). In order to do so, the company engages in MPC with a set of regulatory agencies, which are either trusted parties holding consumers' race data (Veale & Binns, 2017), or hold among them a secret sharing of race data, provided by the consumers themselves (Kilbertus et al., 2018). Together the company and the regulators apply standard fair machine learning techniques in a distributed fashion. In this way the company never directly accesses the race data, but still manages to produce a fair model, which is the output of the MPC. The guarantee provided by this solution is the standard one of MPC — namely, the company learns *nothing more than whatever is implied by its own consumer data, and the fair model returned by the protocol.*

Our point of departure stems from our assertion that MPC is the wrong guarantee to give if our motivation is ensuring that data about an individual's race does not "leak" to the company via the model. In particular, MPC *implies nothing about what individual information can already be inferred*

[1] Northeastern University, Boston, MA, USA [2] University of Pennsylvania, Philadelphia, PA, USA. Correspondence to: Saeed Sharifi-Malvajerdi <saeedsh@wharton.upenn.edu>.

Proceedings of the 36^{th} International Conference on Machine Learning, Long Beach, California, PMLR 97, 2019. Copyright 2019 by the author(s).

from the learned model itself. The guarantee we would prefer is that the company's data and the fair model do not leak anything about an individual's race beyond what can be inferred from "population level" correlations. That is, the fair model should not leak anything beyond inferences that could be carried out *even if the individual in question had declined to provide her racial identity*. This is exactly the type of promise made by *differential privacy* (Dwork et al., 2006b), but not by MPC.

The insufficiency of MPC. To emphasize the fact that concerns over leakage of protected attributes under the guarantee of MPC are more than hypothetical, we describe a natural example where this leakage would actually occur.

Example. An SVM model, trained in the standard way, is represented by the underlying support vectors, which are just data points from the training data. Thus, if race is a feature represented in the training data, an SVM model computed under MPC reveals the race of the individuals represented in the support vectors. This is the case even if race is uncorrelated with all other features and labels, in which case differential privacy would prevent such inferences. We note that there are differentially private implementations of SVMs.

The reader might object that, in this example, the algorithm is trained to use racial data at test time, and so the output of the algorithm is directly affected by race. But there are also examples in which the same problems with MPC can arise *even when race is not an input to the learned model, and race is again uncorrelated with the company's data.* We also note that SVMs are just an extreme case of a learned model fitting, and thus potentially revealing, its training data. For example, points from the training set can also be recovered from trained neural networks (Song et al., 2017).

Our approach: differential privacy. These examples show that cryptographic approaches to "locking up" sensitive information during a training process are insufficient as a privacy mechanism — *we need to explicitly reason about what can be inferred from the output of a learning algorithm*, not simply say that we cannot learn more than such inferences. In this paper we thus instead consider the problem of designing fair learning algorithms that also promise differential privacy with respect to consumer race, and thus give strong guarantees about what can be inferred from the learned model.

We note that the guarantee of differential privacy is somewhat subtle, and does *not* promise that the company will be unable to infer race. For example, it might be that a feature that the company already has, such as zip codes, is perfectly correlated with race, and a computation that is differentially private might reveal this correlation. In this case, the company will be able to infer racial information about its customers. However, differential privacy prevents leakage of individual racial data beyond what can be inferred from population-level correlations.

Like (Veale & Binns, 2017), our approach can be viewed as a collaboration between a company holding non-sensitive consumer data and a regulator holding sensitive data. Our algorithms allow the regulator to build fair models from the combined data set (potentially also under MPC) in a way that ensures the company, or any other party with access to the model or its decisions, cannot infer the race of any consumer in the data much more accurately than they could do from population-level statistics alone. Thus, we comply with the spirit of laws and regulations asking that sensitive attributes not be leaked, while still allowing them to be used to enforce fairness.

1.1. Our Results

We study the problem of learning classifiers from data with protected attributes. More specifically, we are given a class of classifiers $\mathcal{H}$ and we output a randomized classifier in $\Delta(\mathcal{H})$ (i.e. a distribution over $\mathcal{H}$). The training data consists of m individual data points of the form (X, A, Y). Here $X \in \mathcal{X}$ is the vector of unprotected attributes, $A \in \mathcal{A}$ is the protected attribute and Y is the binary label. As discussed above, our algorithms achieve three goals simultaneously:

- **Differential privacy**: Our learning algorithms satisfy *differential privacy* (Dwork et al., 2006b) with respect to protected attributes. (They need not be differentially private with respect to the unprotected attributes X — although sometimes are.)
- **Fairness:** Our learning algorithms guarantee approximate notions of statistical fairness across the groups specified by the protected attribute. The particular statistical fairness notion we focus on is *Equalized Odds* (Hardt et al., 2016), which in the binary classification case reduces to asking that false positive rates and false negative rates be approximately equal, conditional on all values of the protected attribute (but our techniques apply to other notions of statistical fairness as well, including statistical parity).
- **Accuracy:** Our output classifier has error rate comparable to non-private benchmarks in $\Delta(\mathcal{H})$ consistent with the fairness constraints.

We evaluate fairness and error as in-sample quantities. Out-of-sample generalization for both error and fairness follow from standard sample-complexity bounds in learning theory, and so we elide this complication for clarity (but see e.g. the treatment in (Kearns et al., 2018b) for formal generalization bounds).

We start with a simple extension of the *post-processing* ap-

Algorithm	Assumptions on $\mathcal{H}$	Fairness Guarantee	Needs access to A at test time?	Does it guarantee privacy of X as well?	Error	Fairness Violation
DP-postprocessing	None	Equalized Odds	Yes	No	$\tilde{O}\left(\frac{\lvert\mathcal{A}\rvert}{m\epsilon}\right)$ [1]	$\tilde{O}\left(\frac{1}{\min \hat{q}_{ay} m\epsilon}\right)$
DP-oracle-learner	$d_{\mathcal{H}} < \infty$ $d_{\mathcal{H}} := VC(\mathcal{H})$	Equalized Odds	No	No	$\tilde{O}\left(\frac{B}{\min \hat{q}_{ay}}\sqrt{\frac{\lvert\mathcal{A}\rvert d_{\mathcal{H}}}{m\epsilon}}\right)$	$B^{-1} + \tilde{O}\left(\frac{1}{\min \hat{q}_{ay}}\sqrt{\frac{\lvert\mathcal{A}\rvert d_{\mathcal{H}}}{m\epsilon}}\right)$
	$\lvert\mathcal{H}\rvert < \infty$	Equalized Odds	No	Yes	$\tilde{O}\left(\frac{B}{\min \hat{q}_{ay}}\sqrt{\frac{\lvert\mathcal{A}\rvert \ln(\lvert\mathcal{H}\rvert)}{m\epsilon}}\right)$	$B^{-1} + \tilde{O}\left(\frac{1}{\min \hat{q}_{ay}}\sqrt{\frac{\lvert\mathcal{A}\rvert \ln(\lvert\mathcal{H}\rvert)}{m\epsilon}}\right)$
	$\lvert\mathcal{H}\rvert < \infty$, $\mathcal{H}$ has maximally discriminatory classifiers	Equalized False Positive Rate	Yes	Yes	$\tilde{O}\left(\frac{\lvert\mathcal{A}\rvert}{\min \hat{q}_{ay}}\sqrt{\frac{\lvert\mathcal{A}\rvert \ln(\lvert\mathcal{H}\rvert)}{m\epsilon}}\right)$	$\tilde{O}\left(\frac{\lvert\mathcal{A}\rvert}{\min \hat{q}_{ay}}\sqrt{\frac{\lvert\mathcal{A}\rvert \ln(\lvert\mathcal{H}\rvert)}{m\epsilon}}\right)$

Table 1: Summary of Results for Our Differentially Private Fair Learning Algorithms. In this table, m is the training sample size, $\hat{q}_{ay}$ is the fraction of data with $A = a$ and $Y = y$, $|\mathcal{A}|$ is the number of protected groups, and ϵ is the privacy parameter. B is explained in text. For all but the marked error bound, the comparison benchmark is the optimal fair classifier. The marked bound is compared to a weaker benchmark: the outcome of the non-private post-processing procedure.

proach of (Hardt et al., 2016). Their algorithm starts with a possibly unfair classifier $\widehat{Y}$ and derives a fair classifier by mixing $\widehat{Y}$ with classifiers which are based on protected attributes. This involves solving a linear program which takes quantities $\hat{q}_{\hat{y}ay}$ as input. Here $\hat{q}_{\hat{y}ay}$ is the fraction of data points with $\widehat{Y} = \hat{y}, A = a, Y = y$. To make this approach differentially private with respect to protected attributes, we start with $\widehat{Y}$ which is learned without using protected attributes and we use standard techniques to perturb the $\hat{q}_{\hat{y}ay}$'s before feeding them into the linear program, in a way that guarantees differential privacy. We analyze the additional error and fairness violation that results from the perturbation. Detailed results can be found in Section 3.

Although having the virtue of being exceedingly simple, this first approach has two significant drawbacks. First, even without privacy, this post-processing approach does not in general produce classifiers with error that is comparable to that of the best fair classifiers, and our privacy preserving modification inherits this limitation. Second, and often more importantly, this post-processing approach crucially requires that protected attributes can be used at test time, and this isn't feasible (or legal) in certain applications. Even when it is, if racial information is held only by a regulator, although it may be feasible to train a model once using MPC, it probably is not feasible to make test-time decisions repeatedly using MPC.

We then consider the approach of (Agarwal et al., 2018), which we refer to it as *in-processing* (to distinguish it from post-processing). They give an *oracle-efficient* algorithm, which assumes access to a subroutine that can optimally solve classification problems absent a fairness constraint (in practice, and in our experiments, these "oracles" are implemented using simple learning heuristics). Their approach does not have either of the above drawbacks: it does not require that protected features be available at test time, and it is guaranteed to produce the approximately optimal fair classifier. The algorithm is correspondingly more complicated. The main idea of their approach (following the presentation of (Kearns et al., 2018b)) is to show that the optimal fair classifier can be found as the equilibrium of a zero-sum game between a "Learner" who selects classifiers in $\mathcal{H}$ and an "Auditor" who finds fairness violations. This equilibrium can be approximated by iterative play of the game, in which the Auditor plays exponentiated gradient descent and the Learner plays best responses (computed via an efficient cost-sensitive classification oracle). To make this approach private, we add Laplace noise to the gradients used by the Auditor and we let the Learner run the exponential mechanism (or some other private learning oracle) to compute approximate best responses. Our technical contribution is to show that the Learner and the Auditor still converge to an approximate equilibrium despite the noise introduced for privacy. Detailed results can be found in Section 4.

One of the most interesting aspects of our results is an inherent tradeoff that arises between privacy, accuracy, and fairness, that doesn't arise when any two of these desiderata are considered alone. This manifests itself as the parameter "B" in our in-processing result (see Table 1) which mediates the tradeoff between error, fairness and privacy. This parameter also appears in the (non-private) algorithm of (Agarwal et al., 2018)—but there it serves only to mediate a tradeoff between fairness and running time. At a high level, the reason for this difference is that without the need for privacy, we can increase the number of iterations of the algorithm to decrease the error to any desired level. However, when we also need to protect privacy, there is an additional tradeoff, and increasing the number of iterations also requires increasing the scale of the gradient perturbations, which may not always decrease error.

This tradeoff exhibits an additional interesting feature. Recall that as we discussed above, the in-processing approach works even if we can not use protected attributes at test time. But *if we are allowed to use protected attributes at test time*, we are able to obtain a better tradeoff between these quantities — essentially eliminating the role of the variable B that would otherwise mediate this tradeoff. We give details of this improvement in section 4.1 (for this result, we also need to relax the fairness requirement from *Equalized Odds* to *Equalized False Positive Rates*). The main step in the proof is to show that, for small constant B and $\mathcal{H}$ containing certain "maximally discriminatory" classifiers which make decisions *solely* on the basis of group membership, we can give a better characterization of the Learner's strategy at the approximate equilibrium of the zero-sum game.

Finally, we provide evidence that using protected attributes at test time is necessary for obtaining this better tradeoff. In Section 4.2, we consider the sensitivity of computing the error of the optimal classifier subject to fairness constraints. We show that this sensitivity can be substantially higher when the classifier cannot use protected attributes at test time, which shows that higher error must be introduced to estimate this error privately.

1.2. Related Work

The literature on algorithmic fairness is growing rapidly, and is by now far too extensive to exhaustively cover here. See (Chouldechova & Roth, 2018) for a recent survey. Our work builds directly on that of (Hardt et al., 2016), (Agarwal et al., 2018), and (Kearns et al., 2018b). In particular, (Hardt et al., 2016) introduces the "equalized odds" definition that we take as our primary fairness goal, and gave a simple post-processing algorithm that we modify to make differentially private. (Agarwal et al., 2018) derives an "oracle efficient" algorithm which can optimally solve the fair empirical risk minimization problem (for a variety of statistical fairness constraints, including equalized odds) given oracles (implemented with heuristics) for the unconstrained learning problem. (Kearns et al., 2018b) generalize this algorithm to be able to handle infinitely many protected groups. We give a differentially private version of (Agarwal et al., 2018) as well.

Our paper is directly inspired by (Kilbertus et al., 2018), who study how to train fair machine learning models by encrypting sensitive attributes and applying secure multiparty computation (MPC). We share the goal of (Kilbertus et al., 2018): we want to train fair classifiers without leaking information about an individual's race through their participation in the training. Our starting point is the observation that differential privacy, rather than secure multiparty computation, is the right tool for this.

We use differential privacy (Dwork et al., 2006b) as our notion of individual privacy, which has become an influential "solution concept" for data privacy in the last decade. See (Dwork & Roth, 2014) for a survey. We make use of standard tools from this literature, including the Laplace mechanism (Dwork et al., 2006b), the exponential mechanism (McSherry & Talwar, 2007) and composition theorems (Dwork et al., 2006a; 2010).

2. Model and Preliminaries

Suppose we are given a data set of m individuals drawn $i.i.d.$ from an unknown distribution $\mathcal{P}$ where each individual is described by a tuple (X, A, Y). $X \in \mathcal{X}$ forms a vector of *unprotected attributes*, $A \in \mathcal{A}$ is the *protected attribute* where $|\mathcal{A}| < \infty$, and $Y \in \mathcal{Y}$ is a binary label. Without loss of generality, we write $\mathcal{A} = \{0, 1, \ldots, |\mathcal{A}| - 1\}$ and let $\mathcal{Y} = \{0, 1\}$. Let $\widehat{\mathcal{P}}$ denote the empirical distribution of the observed data. Our primary goal is to develop an algorithm to learn a (possibly randomized) *fair* classifier $\widehat{Y}$, with an algorithm that guarantees the *privacy* of the sensitive attribute A. By privacy, we mean differential privacy, and by fairness, we mean (approximate versions of) the *Equalized Odds* condition of (Hardt et al., 2016). Both of these notions are parameterized: differential privacy has a parameter ϵ, and the approximate fairness constraint is parameterized by γ. Our main interest is in characterizing the tradeoff between ϵ, γ, and classification error. Here we provide basic definitions of fairness and differential privacy. See the supplementary file for a detailed discussion of them.

Notations: $\widehat{\mathbb{P}}$ throughout refers to the probability taken w.r.t $\widehat{\mathcal{P}}$. $\hat{q}_{\hat{y}ay} := \widehat{\mathbb{P}}[\widehat{Y} = \hat{y}, A = a, Y = y]$, and $\hat{q}_{ay} := \widehat{\mathbb{P}}[A = a, Y = y]$. $\widehat{\text{FP}}_a(\widehat{Y}) = \widehat{\mathbb{P}}[\widehat{Y} = 1 \,|\, A = a, Y = 0], \widehat{\text{TP}}_a(\widehat{Y}) = \widehat{\mathbb{P}}[\widehat{Y} = 1 \,|\, A = a, Y = 1]$ are the false and true positive rates of $\widehat{Y}$ in the subpopulation $\{A = a\}$. $\Delta\widehat{\text{FP}}_a(\widehat{Y}) = |\widehat{\text{FP}}_a(\widehat{Y}) - \widehat{\text{FP}}_0(\widehat{Y})|$ and $\Delta\widehat{\text{TP}}_a(\widehat{Y}) = |\widehat{\text{TP}}_a(\widehat{Y}) - \widehat{\text{TP}}_0(\widehat{Y})|$ are used to measure $\widehat{Y}$'s false/true positive rate discrepancies across groups. $\widehat{\text{err}}\,(\widehat{Y}) = \widehat{\mathbb{P}}[\widehat{Y} \neq Y]$ is the error of $\widehat{Y}$.

2.1. Fairness

Definition 2.1 (γ-Equalized Odds Fairness)**.** *A classifier $\widehat{Y}$ satisfies the γ-Equalized Odds condition with respect to the attribute A, if for all $a \in \mathcal{A}$, the false and true positive rates of $\widehat{Y}$ in the subpopulations $\{A = a\}$ and $\{A = 0\}$ are within γ of one another. In other words, for all $a \neq 0$, $\Delta FP_a(\widehat{Y}) \leqslant \gamma$ and $\Delta TP_a(\widehat{Y}) \leqslant \gamma$.*

2.2. Differential Privacy

Let $\mathcal{D}$ be a *data universe* from which a database D of size m is drawn and let $M : \mathcal{D}^m \to \mathcal{O}$ be an algorithm that takes a *database* D as input. Differential privacy requires that the addition or removal of a single data entry should have little (distributional) effect on the output of the mechanism. In other words, for every pair of *neighboring* databases $D \sim D' \in \mathcal{D}^m$ that differ in at most one entry, differential privacy requires that the distribution of $M(D)$ and $M(D')$ are "close" to each other, measured by parameters ϵ and δ.

Definition 2.2 ((ϵ, δ)-Differential Privacy (DP) (Dwork et al., 2006b))**.** *A randomized algorithm $M : \mathcal{D}^m \to \mathcal{O}$ is said to be (ϵ, δ)-differentially private if for all pairs of neighboring databases $D, D' \in \mathcal{D}^m$ and all $O \subseteq \mathcal{O}$,*

$$\mathbb{P}[M(D) \in O] \leqslant e^{\epsilon}\, \mathbb{P}[M(D') \in O] + \delta$$

if $\delta = 0$, M is said to be ϵ-differentially private.

Recall that our data universe is $\mathcal{D} = (\mathcal{X}, \mathcal{A}, \mathcal{Y})$, which will be convenient to partition as $(\mathcal{X}, \mathcal{Y}) \times \mathcal{A}$. Given a dataset D of size m, we will write it as a pair $D = (D_I, D_S)$ where $D_I \in (\mathcal{X}, \mathcal{Y})^m$ represent the insensitive attributes and $D_S \in \mathcal{A}^m$ represent the sensitive attributes. We will sometimes incidentally guarantee differential privacy over the entire data universe $\mathcal{D}$ (see Table 1), but our main goal will be to promise differential privacy only with respect to the sensitive attributes. Write $D_S \sim D'_S$ to denote that D_S and D'_S differ in exactly one coordinate (i.e. in one person's group membership). An algorithm is (ϵ, δ)-*differentially private in the sensitive attributes* if for all $D_I \in (\mathcal{X}, \mathcal{Y})^m$ and for all $D_S \sim D'_S \in \mathcal{A}^m$, we have:

$$\mathbb{P}\left[M(D_I, D_S) \in O\right] \leqslant e^{\epsilon}\, \mathbb{P}\left[M(D_I, D'_S) \in O\right] + \delta$$

3. DP Fair Learning: Post-Processing

In this section we will present our first differentially private fair learning algorithm which will be called **DP-postprocessing**. The **DP-postprocessing** algorithm (Algorithm 1) is a private variant of the algorithm introduced in (Hardt et al., 2016).

(Hardt et al., 2016)'s approach starts with a base classifier $\widehat{Y}$ which is learned without using protected attributes. They derive a fair classifier $\widehat{Y}_p$ by mixing $\widehat{Y}$ with classifiers depending on the protected attributes. $\widehat{Y}_p$ is specified by a parameter $p = (p_{\hat{y}a})_{\hat{y},a}$, a vector of probabilities such that $p_{\hat{y}a} := \mathbb{P}[\widehat{Y}_p = 1 \,|\, \widehat{Y} = \hat{y}, A = a]$. Among all fair $\widehat{Y}_p$'s, the one with minimum error can be found by solving a linear program which takes as input the aggregate statistics $\hat{q}_{\hat{y}ay}$ for all $\hat{y}, a, y$.

In Algorithm 1, we make the above approach differentially private. Notice this method depends on the protected attributes only to compute the quantities $\hat{q}_{\hat{y}ay}$. To guarantee differential privacy, Algorithm 1 computes $\tilde{q}_{\hat{y}ay}$ (a noisy version of $\hat{q}_{\hat{y}ay}$) and then feeds $\tilde{q}_{\hat{y}ay}$ into the linear program $\widetilde{\text{LP}}$ (1). In this linear program, terms with tildes (e.g. $\tilde{q}_{ay}$, $\widetilde{\text{err}}$, $\widetilde{\text{FP}}$, $\widetilde{\text{TP}}$) are defined with respect to $\tilde{q}_{\hat{y}ay}$ instead of $\hat{q}_{\hat{y}ay}$.

Algorithm 1 ϵ-DP fair classification: **DP-postprocessing**

```
**Input:** privacy parameter $\epsilon$, confidence parameter $\beta$, fairness violation $\gamma$, training examples $\{(X_i, A_i, Y_i)\}_{i=1}^m$

Train the base classifier $\widehat{Y}$ on $\{(X_i, Y_i)\}_{i=1}^m$.
Compute $\hat{q}_{\hat{y}ay} = \widehat{\mathbb{P}}[\widehat{Y} = \hat{y}, A = a, Y = y]$.
Sample $W_{\hat{y}ay} \overset{i.i.d.}{\sim} \text{Lap}\,(2/m\epsilon)$ for all $\hat{y}, a, y$.
Perturb each $\hat{q}_{\hat{y}ay}$ with noise: $\tilde{q}_{\hat{y}ay} = \hat{q}_{\hat{y}ay} + W_{\hat{y}ay}$.
Solve $\widetilde{\text{LP}}$ (1) to get the minimizer $\tilde{p}^\star$.

**Output:** $\tilde{p}^\star$, the trained classifier $\widehat{Y}$
```

We analyze the performance of Algorithm 1 in Theorem 3.1. Its proof is deferred to the supplementary file. The main step of the proof is to understand how the introduced noise propagates to the solution of the linear program. We also briefly review the approach of (Hardt et al., 2016) in the supplementary file.

$\widetilde{\text{LP}}$: ϵ-Differentially Private Linear Program

$$\begin{aligned}
&\underset{p}{\arg\min} && \widetilde{\text{err}}\,(\widehat{Y}_p) \\
&\text{s.t. } \underset{a \neq 0}{\forall a \in \mathcal{A}} && \Delta\widetilde{\text{FP}}_a(\widehat{Y}_p) \leqslant \gamma + \frac{4\ln\left(4|\mathcal{A}|/\beta\right)}{\min\{\tilde{q}_{a0}, \tilde{q}_{00}\}\, m\epsilon} \\
& && \Delta\widetilde{\text{TP}}_a(\widehat{Y}_p) \leqslant \gamma + \frac{4\ln\left(4|\mathcal{A}|/\beta\right)}{\min\{\tilde{q}_{a1}, \tilde{q}_{01}\}\, m\epsilon} \\
& && 0 \leqslant p_{\hat{y}a} \leqslant 1 \quad \forall \hat{y}, a
\end{aligned} \tag{1}$$

$$\widetilde{\text{err}}\,\left(\widehat{Y}_p\right) := \textstyle\sum_{\hat{y},a} \left(\tilde{q}_{\hat{y}a0} - \tilde{q}_{\hat{y}a1}\right) \cdot p_{\hat{y}a} + \sum_{\hat{y},a} \tilde{q}_{\hat{y}a1}$$

$$\Delta\widetilde{\text{FP}}_a\left(\widehat{Y}_p\right) := \Big|\text{FP}_a(\widehat{Y}) \cdot p_{1a} + \left(1 - \widetilde{\text{FP}}_a(\widehat{Y})\right) \cdot p_{0a} - \widetilde{\text{FP}}_0(\widehat{Y}) \cdot p_{10} - \left(1 - \widetilde{\text{FP}}_0(\widehat{Y})\right) \cdot p_{00}\Big|$$

$$\Delta\widetilde{\text{TP}}_a\left(\widehat{Y}_p\right) := \Big|\widetilde{\text{TP}}_a(\widehat{Y}) \cdot p_{1a} + \left(1 - \widetilde{\text{TP}}_a(\widehat{Y})\right) \cdot p_{0a} - \widetilde{\text{TP}}_0(\widehat{Y}) \cdot p_{10} - \left(1 - \widetilde{\text{TP}}_0(\widehat{Y})\right) \cdot p_{00}\Big|$$

Theorem 3.1 (Error-Privacy, Fairness-Privacy Tradeoffs)**.** *Suppose* $\min_{a,y}\{\hat{q}_{ay}\} > 4\ln\left(4|\mathcal{A}|/\beta\right)/\left(m\,\epsilon\right)$. *Let $\widehat{p}^\star$ be the optimal γ-fair solution of the non-private post-processing algorithm of (Hardt et al., 2016) and let $\widetilde{p}^\star$ be the output of Algorithm 1 which is the optimal solution of $\widetilde{LP}$ (1). With probability at least $1 - \beta$,*

$$\widehat{err}\left(\widehat{Y}_{\widetilde{p}^\star}\right) \leqslant \widehat{err}\left(\widehat{Y}_{\widehat{p}^\star}\right) + \frac{24|\mathcal{A}|\ln\left(4|\mathcal{A}|/\beta\right)}{m\epsilon}$$

and for all $a \neq 0$*,*

$$\Delta\widehat{FP}_a\left(\widehat{Y}_{\widetilde{p}^\star}\right) \leqslant \gamma + \frac{8\ln\left(4|\mathcal{A}|/\beta\right)}{\min\{\hat{q}_{a0}, \hat{q}_{00}\}\, m\epsilon - 4\ln\left(4|\mathcal{A}|/\beta\right)}$$

$$\Delta\widehat{TP}_a\left(\widehat{Y}_{\widetilde{p}^\star}\right) \leqslant \gamma + \frac{8\ln\left(4|\mathcal{A}|/\beta\right)}{\min\{\hat{q}_{a1}, \hat{q}_{01}\}\, m\epsilon - 4\ln\left(4|\mathcal{A}|/\beta\right)}$$

We emphasize that the accuracy guarantee stated in Theorem 3.1 is relative to the non-private post-processing algorithm, *not* relative to the optimal fair classifier. This is because the non-private post-processing algorithm itself has no such optimality guarantees.

4. DP Fair Learning: In-Processing

In this section we will introduce our second differentially private fair learning algorithm: **DP-oracle-learner** (Algorithm 3). It is based on the fair learning algorithm presented in (Agarwal et al., 2018). Essentially, (Agarwal et al., 2018) reduces the γ-fair learning problem into the following Lagrangian min-max problem:

$$\min_{Q \in \Delta(\mathcal{H})} \max_{\boldsymbol{\lambda} \in \Lambda} L(Q, \boldsymbol{\lambda}) := \widehat{\text{err}}\,(Q) + \boldsymbol{\lambda}^\top \hat{\boldsymbol{r}}(Q) \qquad (2)$$

Here $\mathcal{H}$ is a given class of binary classifiers with $d_{\mathcal{H}} = VCD(\mathcal{H}) < \infty$, and $\hat{\boldsymbol{r}}(Q)$ is a vector of fairness violations of the classifier Q across groups, and $\boldsymbol{\lambda} \in \Lambda = \{\boldsymbol{\lambda} : ||\boldsymbol{\lambda}||_1 \leqslant B\}$ is the dual variable. In this work,

$$\hat{\boldsymbol{r}}(Q) := \begin{bmatrix} \widehat{\text{FP}}_a(Q) - \widehat{\text{FP}}_0(Q) - \gamma \\ \widehat{\text{FP}}_0(Q) - \widehat{\text{FP}}_a(Q) - \gamma \\ \widehat{\text{TP}}_a(Q) - \widehat{\text{TP}}_0(Q) - \gamma \\ \widehat{\text{TP}}_0(Q) - \widehat{\text{TP}}_a(Q) - \gamma \end{bmatrix}_{\substack{a \in \mathcal{A} \\ a \neq 0}} \in \mathbb{R}^{4(|\mathcal{A}|-1)}$$

$$\boldsymbol{\lambda} = \left[\lambda_{(a,0,+)}, \lambda_{(a,0,-)}, \lambda_{(a,1,+)}, \lambda_{(a,1,-)}\right]^\top_{\substack{a \in \mathcal{A} \\ a \neq 0}} \in \mathbb{R}^{4(|\mathcal{A}|-1)}$$

The method developed by (Agarwal et al., 2018), in the language of (Kearns et al., 2018b), gives a reduction from finding an optimal fair classifier to finding the equilibrium of a two-player zero-sum game played between a "Learner" (Q-player) who needs to solve an unconstrained learning problem (given access to an efficient cost-sensitive classification oracle) and an "Auditor" ($\boldsymbol{\lambda}$-player) who weights ($\boldsymbol{\lambda}$) the fairness violations. Having the learner play its best response and the auditor play a no-regret learning algorithm guarantees convergence of the average plays to the equilibrium.

In Algorithm 3, to make the above approach private, Laplace noise is added to the gradients used by the Auditor (who plays exponentiated gradient descent with learning rate η) and we let the Learner run the exponential mechanism (or some other private learning oracle) to compute approximate best responses. Subroutine 2 reduces the Learner's best response problem to privately solving a cost sensitive classification problem solved with a private oracle $\text{CSC}_{\epsilon'}(\mathcal{H})$. Here we sketch the main steps of analyzing Algorithm 3. All the proofs and a review of the approach of (Agarwal et al., 2018) are deferred to the supplementary file.

Subroutine 2 $\text{BEST}_h^{\epsilon'}$

```
Input: $\boldsymbol{\lambda}$, training examples $\{(X_i, A_i, Y_i)\}_{i=1}^m$, privacy guarantee $\epsilon'$
for $i = 1, \ldots, m$ do
    $C_i^0 \leftarrow \mathbb{1}\{Y_i \neq 0\}$ (cost of labeling 0)
    $C_i^1 \leftarrow \mathbb{1}\{Y_i \neq 1\} + \frac{\lambda_{(A_i,Y_i,+)} - \lambda_{(A_i,Y_i,-)}}{\hat{q}_{A_iY_i}} \mathbb{1}\{A_i \neq 0\} - \sum_{\substack{a \in \mathcal{A} \\ a \neq 0}} \frac{\lambda_{(a,Y_i,+)} - \lambda_{(a,Y_i,-)}}{\hat{q}_{A_iY_i}} \mathbb{1}\{A_i = 0\}$ (cost of labeling 1)
end for
Call $\text{CSC}_{\epsilon'}(\mathcal{H})$ with $\{X_i, C_i^0, C_i^1\}_{i=1}^m$ to get $h^\star$.
Output: $h^\star$
```

Algorithm 3 (ϵ, δ)-differentially private fair classification: **DP-oracle-learner**

```
Input: privacy parameters $\epsilon, \delta$, bound $B$, VC dimension $d_{\mathcal{H}}$, confidence parameter $\beta$, fairness violation $\gamma$, training examples $\{(X_i, A_i, Y_i)\}_{i=1}^m$
    $T \leftarrow \frac{B\sqrt{\ln(4|\mathcal{A}|-3)}\, m\,\epsilon}{2(2|\mathcal{A}|B+1)\sqrt{\ln(1/\delta)}(d_{\mathcal{H}}\ln(m)+\ln(2/\beta))}$
    $\eta \leftarrow \frac{1}{2}\sqrt{\frac{\ln(4|\mathcal{A}|-3)}{T}}$
$\widetilde{\boldsymbol{\theta}}_1 \leftarrow \mathbf{0} \in \mathbb{R}^{4(|\mathcal{A}|-1)}$
for $t = 1, \ldots, T$ do
    $\widetilde{\lambda}_{t,k} \leftarrow B\frac{\exp(\widetilde{\theta}_{t,k})}{1+\sum_{k'}\exp(\widetilde{\theta}_{t,k'})}$ for all $k$
    $\widetilde{h}_t \leftarrow \text{BEST}_h^{\epsilon'}(\widetilde{\boldsymbol{\lambda}}_t)$ with $\epsilon' = \epsilon/(4\sqrt{T\ln(1/\delta)})$
    Sample $\boldsymbol{W}_t$: $W_{t,k} \overset{i.i.d.}{\sim} \text{Lap}(\frac{8|\mathcal{A}|\sqrt{T\ln(1/\delta)}}{(\min_{a,y}\{\hat{q}_{ay}\}\,m-1)\cdot\epsilon})$
    $\widetilde{\boldsymbol{r}}_t \leftarrow \hat{\boldsymbol{r}}_t(\widetilde{h}_t) + \boldsymbol{W}_t$
    $\widetilde{\boldsymbol{\theta}}_{t+1} \leftarrow \widetilde{\boldsymbol{\theta}}_t + \eta\widetilde{\boldsymbol{r}}_t$
end for
$\widetilde{Q} \leftarrow \frac{1}{T}\sum_{t=1}^T \widetilde{h}_t, \quad \widetilde{\boldsymbol{\lambda}} \leftarrow \frac{1}{T}\sum_{t=1}^T \widetilde{\boldsymbol{\lambda}}_t$
Output: $(\widetilde{Q}, \widetilde{\boldsymbol{\lambda}})$
```

We first bound the regret of the Learner and the Auditor in Lemma 4.1 and 4.2 by understanding how the introduced perturbations affect these regret terms.

Lemma 4.1 (Regret of the Private Learner)**.** *Suppose* $\{\widetilde{h}_t\}_{t=1}^T$ *is the sequence of best responses to* $\{\widetilde{\boldsymbol{\lambda}}_t\}_{t=1}^T$ *by the private Learner over* T *rounds. We have that with prob-*

ability at least $1 - \beta/2$*,*

$$\frac{1}{T}\sum_{t=1}^{T} L(\widetilde{h}_t, \widetilde{\boldsymbol{\lambda}}_t) - \frac{1}{T}\min_{Q \in \Delta(\mathcal{H})} \sum_{t=1}^{T} L(Q, \widetilde{\boldsymbol{\lambda}}_t) \leqslant \frac{8\left(2|\mathcal{A}|B + 1\right)\sqrt{T\ln\left(1/\delta\right)}\left(d_{\mathcal{H}}\ln\left(m\right) + \ln\left(2T/\beta\right)\right)}{\left(\min_{a,y}\{\hat{q}_{ay}\}\, m - 1\right)\cdot \epsilon}$$

Lemma 4.2 (Regret of the Private Auditor)**.** *Let* $\{\widetilde{\boldsymbol{\lambda}}_t\}_{t=1}^{T}$ *be the sequence of exponentiated gradient descent plays (with learning rate* η*) by the private Auditor to given* $\{\widetilde{h}_t\}_{t=1}^{T}$ *of the private Learner over* T *rounds. With probability at least* $1 - \beta/2$*,*

$$\frac{1}{T}\max_{\boldsymbol{\lambda}\in\Lambda} \sum_{t=1}^{T} L(\widetilde{h}_t, \boldsymbol{\lambda}) - \frac{1}{T}\sum_{t=1}^{T} L(\widetilde{h}_t, \widetilde{\boldsymbol{\lambda}}_t) \leqslant$$

$$\frac{B\ln(4|\mathcal{A}| - 3)}{\eta T} + 4\eta B\left(1 + \frac{4|\mathcal{A}|\sqrt{T\ln\left(\frac{1}{\delta}\right)}\ln\left(\frac{8T|\mathcal{A}|}{\beta}\right)}{\left(\min_{a,y}\{\hat{q}_{ay}\}\, m - 1\right)\cdot\epsilon}\right)^2$$

Now in Theorem 4.3, given Lemma 4.1 and 4.2, we can characterize the average plays of both players.

Theorem 4.3. *Let* $(\widetilde{Q}, \widetilde{\boldsymbol{\lambda}})$ *be the output of Algorithm 3. We have that with probability at least* $1 - \beta$*,* $(\widetilde{Q}, \widetilde{\boldsymbol{\lambda}})$ *is a* ν*-approximate solution of the game, i.e.,*

$$\begin{aligned} L(\widetilde{Q}, \widetilde{\boldsymbol{\lambda}}) &\leqslant L(Q, \widetilde{\boldsymbol{\lambda}}) + \nu \quad \textit{for all } Q \in \Delta(\mathcal{H}) \\ L(\widetilde{Q}, \widetilde{\boldsymbol{\lambda}}) &\geqslant L(\widetilde{Q}, \boldsymbol{\lambda}) - \nu \quad \textit{for all } \boldsymbol{\lambda} \in \Lambda \end{aligned}$$

and that

$$\nu = \widetilde{O}\left(\frac{B}{\min\limits_{a,y}\{\hat{q}_{ay}\}}\sqrt{\frac{|\mathcal{A}|\sqrt{\ln\left(1/\delta\right)}\ln\left(m^{d_{\mathcal{H}}}/\beta\right)}{m\,\epsilon}}\right)$$

where we hide further logarithmic dependence on m*,* ϵ*, and* $|\mathcal{A}|$ *under the* $\widetilde{O}$ *notation.*

We are now ready to conclude the **DP-oracle-learner** algorithm's analysis with our main theorem.

Theorem 4.4 (Error-Privacy, Fairness-Privacy Tradeoffs)**.** *Let* ν *be as in Theorem 4.3. Let* $(\widetilde{Q}, \widetilde{\boldsymbol{\lambda}})$ *be the output of Algorithm 3 and let* $Q^\star$ *be the solution to the non-private* γ*-fair ERM problem. We have that with probability at least* $1 - \beta$*,* $\widehat{err}\,(\widetilde{Q}) \leqslant \widehat{err}\,(Q^\star) + 2\nu$ *and for all* $a \neq 0$*,*

$$\Delta\widehat{FP}_a(\widetilde{Q}) \leqslant \gamma + \frac{1 + 2\nu}{B}\ ,\ \Delta\widehat{TP}_a(\widetilde{Q}) \leqslant \gamma + \frac{1 + 2\nu}{B}$$

Remark 4.1. *The bounds above reveal a tradeoff between accuracy and fairness violation by controlling through the parameter* B*. As* B *increases, the upper bound on error gets looser while the one on fairness violation gets tighter.*

Remark 4.2. *We assumed in this section that the protected attribute* A *is not available to the classifiers in* $\mathcal{H}$ *("*A*-blind" classification) and stated all our bounds in terms of* $d_{\mathcal{H}}$*. In the more general setting where classifiers in* $\mathcal{H}$ *could depend on* A *("*A*-aware" classification), similar results hold. The only change to make is to replace* $\ln\left(m^{d_{\mathcal{H}}}\right)$ *with* $\ln\left(|\mathcal{H}|\right)$ *in the bounds. See the supplementary file for more details.*

4.1. Better Tradeoffs for A-aware Classification

In this subsection we show that if we only ask for equalized false positive rates (instead of equalized odds, which also requires equalized true positive rates), and moreover, if we assume $\mathcal{H}$ includes all "maximally discriminatory" classifiers (see Assumption 4.1), the fairness violation guarantees given in Theorems 4.4 can be improved.

Assumption 4.1. $\mathcal{H}$ *includes all group indicator functions:* $\{h_a(X, A) = \mathbb{1}_{A=a},\ \bar{h}_a(X, A) = \mathbb{1}_{A\neq a}\ |a \in \mathcal{A}\} \subseteq \mathcal{H}$*.*

Theorem 4.5 (Error-Privacy, Fairness-Privacy Tradeoffs)**.** *Let* $(\widetilde{Q}, \widetilde{\boldsymbol{\lambda}})$ *be the output of Algorithm 3, and let* $Q^\star$ *be the solution to the non-private* γ*-fair ERM problem. Under assumptions 4.1, and* $B > |\mathcal{A}| - 1$*, with probability at least* $1 - \beta$*,* $\widehat{err}\,(\widetilde{Q}) \leqslant \widehat{err}\,(Q^\star) + 2\nu$ *, and for all* $a \neq 0$*,*

$$\Delta\widehat{FP}_a(\widetilde{Q}) \leqslant \gamma + \frac{2\nu}{B - (|\mathcal{A}| - 1)}$$

where

$$\nu = \widetilde{O}\left(\frac{B}{\min_{a,y}\{\hat{q}_{ay}\}}\sqrt{\frac{|\mathcal{A}|\sqrt{\ln(1/\delta)}\ln(|\mathcal{H}|/\beta)}{m\,\epsilon}}\right)$$

4.2. A Separation: A-blind vs. A-aware Classification

In this subsection we show that the sensitivity of the error of the optimal classifier subject to fairness constraints can be substantially higher if it is prohibited from using sensitive attributes at test time, and thus we need more noise to estimate this error subject to differential privacy. This shows a fundamental tension between the goals of trading off privacy and approximate equalized odds, with the goal of preventing "disparate treatment" (using protected attributes explicitly in classification).

Given a data set D of m individuals, define $f(D)$ to be the optimal error rate of any classifier constrained to have a false positive rate disparity $\leqslant \gamma$. Now consider the following problem instance. Let X be the unprotected attribute taking value in $\mathcal{X} = \{U, V\}$, and let A be the protected attribute taking value in $\mathcal{A} = \{R, B\}$. Suppose $\mathcal{H}$ consists of two classifiers h_0 and h_U where $h_0(X, A) = 0$ and $h_U(X, A) = \mathbb{1}_{X=U}$. Notice that both h_0 and h_U depend only on the unprotected attribute. Let h_R and h_B be two other classifiers that depend on the protected attribute: $h_R(X, A) = \mathbb{1}_{A=R}$ and $h_B(X, A) = \mathbb{1}_{A=B}$.

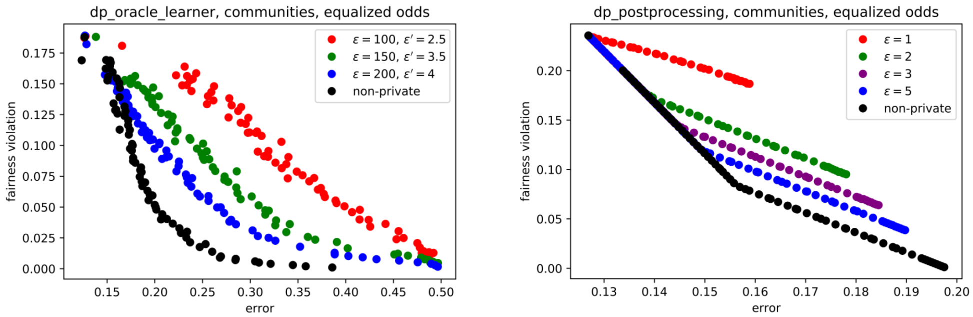

Figure 1: Left figure shows the Pareto frontier of error and (equalized odds) fairness violation for the DP-oracle-learner algorithm on the Communities dataset across different privacy parameters . Right figure shows the corresponding Pareto curves for the DP-postprocessing algorithm. Each point on the private curves is averaged over many rounds to reduce the effect of noise variance. See text for details.

Theorem 4.6. *Consider $\gamma > 1/m$ and datasets with $\min_a \hat{q}_{a0} \geqslant C$ for some constant $C > 0$. If $\mathcal{H} = \{h_0, h_U\}$, the sensitivity of f is $\Omega\left(1/(\gamma m)\right)$. If the "maximally discriminatory" classifiers h_R and h_B are also included in $\mathcal{H}$, i.e. $\mathcal{H} = \{h_0, h_U, h_R, h_B\}$, the sensitivity of f is $O\left(1/m\right)$.*

5. Experimental Evaluation

As a proof of concept, we empirically evaluate our two algorithms on a common fairness benchmark dataset: the Communities and Crime dataset [2] from the UC Irvine Machine Learning Repository. We refer the reader to (Kearns et al., 2018a) for an outline of potential fairness concerns present in the dataset. We clean and preprocess the data identically to (Kearns et al., 2018a). Our main experimental goal is to obtain, for both algorithms, the Pareto frontier of error and fairness violation tradeoffs for different levels of differential privacy. To elaborate, for a given setting of input parameters, we start with the target fairness violation bound $\gamma = 0$ and then increase it over a rich pre-specified subset of $[0, 1]$ while recording for each γ the error and the (realized) fairness violation of the classifier output by the algorithm. We take $\mathcal{H}$ to be the class of linear threshold functions, $\beta = 0.05$, and $\delta = 10^{-7}$.

Logistic regression is used as the base classifier of the **DP-postprocessing** algorithm in our experiments. To implement the Learner's cost-sensitive classification oracle used in the **DP-oracle-learner** algorithm, following (Kearns et al., 2018a), we build a regression-based linear predictor for each vector of costs (C_0 and C_1), and classify a point according to the lowest predicted cost. We made this private following the method of (Smith et al., 2017): computing each regression as $(X^T X)^{-1} X^T C_b$, and adding appropriately scaled Laplace noise to both $X^T X$ and $X^T C_b$. Note when the sensitive attribute A is not included in X (the A-blind case, as in our experiments) noise need not be added to $X^T X$ as we only need to guarantee the privacy of A.

The theory is ambiguous in its predictions about which algorithm should perform better: the "privacy cost" is higher for the in-processing algorithm, but the benchmark that the post-processing algorithm competes with is weaker. We would generally expect therefore that on sufficiently large datasets, the in-processing algorithm would obtain better tradeoffs, but on small datasets, the post-processing algorithm would.

Our experimental results appear in Fig. 1. Indeed, on our relatively small dataset ($m \approx$ 2K), the post-processing algorithm can obtain good tradeoffs between accuracy and fairness at meaningful levels of ϵ, whereas the in-processing algorithm cannot. Nevertheless, we can empirically obtain the "shape" of the Pareto curve trading off accuracy and fairness for unreasonable levels of ϵ using our algorithm. This is still valuable, because the value of ϵ obtained by our algorithms predictably decreases as the dataset size m increases without otherwise changing the dynamics of the algorithm. For example, if we "upsampled" our dataset by a factor of 10 (i.e. taking 10 copies of the dataset), the result would be a reasonably sized dataset of $m \approx$ 20K. Our algorithm run on this upsampled dataset would obtain the same tradeoff curve but now with meaningful values of ϵ. In the left panel of Fig. 1, ϵ is the actual privacy parameter used in the experiments; while ϵ' is the value that the privacy parameter would take on the upsampled dataset.

Recall that the post-processing approach requires the use of the protected attribute at test time, but the in-processing approach does not. Our results therefore suggest that the requirement that we *not* use the protected attribute at test time (i.e. that we be avoid "disparate treatment") might be extremely burdensome if we also want the protections of differential privacy and have only small dataset sizes. In contrast, it can be overcome with the in-processing algorithm at larger dataset sizes.

[2] Briefly, each record in this dataset summarizes aggregate socioeconomic information about both the citizens and police force in a particular U.S. community, and the problem is to predict whether the community has a high rate of violent crime.

Acknowledgements

AR is supported in part by NSF grants AF-1763307 and CNS-1253345. JU is supported by NSF grants CCF-1718088, CCF-1750640, and CNS-1816028, and a Google Faculty Research Award. Alina Oprea was partially supported by the Combat Capabilities Development Command Army Research Laboratory under Cooperative Agreement Number W911NF-13-2-0045 (ARL Cyber Security CRA). The views and conclusions contained in this document are those of the authors and should not be interpreted as representing the official policies, either expressed or implied, of the Combat Capabilities Development Command Army Research Laboratory or the U.S. Government. The U.S. Government is authorized to reproduce and distribute reprints for Government purposes not withstanding any copyright notation here on.

References

ACM. ACM Conference on Fairness, Accountability and Transparency. 2019. URL https://fatconference.org/index.html.

Agarwal, A., Beygelzimer, A., Dudik, M., Langford, J., and Wallach, H. A reductions approach to fair classification. In Dy, J. and Krause, A. (eds.), *Proceedings of the 35th International Conference on Machine Learning*, volume 80 of *Proceedings of Machine Learning Research*, pp. 60–69, Stockholmsmässan, Stockholm Sweden, 10–15 Jul 2018. PMLR. URL http://proceedings.mlr.press/v80/agarwal18a.html.

Chouldechova, A. and Roth, A. The frontiers of fairness in machine learning. 2018. URL arXiv:1810.08810.

Dwork, C. and Roth, A. The algorithmic foundations of differential privacy. *Foundations and Trends® in Theoretical Computer Science*, 9(3–4):211–407, 2014. ISSN 1551-305X. doi: 10.1561/0400000042. URL http://dx.doi.org/10.1561/0400000042.

Dwork, C., Kenthapadi, K., McSherry, F., Mironov, I., and Naor, M. Our data, ourselves: Privacy via distributed noise generation. In Vaudenay, S. (ed.), *Advances in Cryptology - EUROCRYPT 2006*, pp. 486–503, Berlin, Heidelberg, 2006a. Springer Berlin Heidelberg. ISBN 978-3-540-34547-3.

Dwork, C., McSherry, F., Nissim, K., and Smith, A. Calibrating noise to sensitivity in private data analysis. In Halevi, S. and Rabin, T. (eds.), *Theory of Cryptography*, pp. 265–284, Berlin, Heidelberg, 2006b. Springer Berlin Heidelberg. ISBN 978-3-540-32732-5.

Dwork, C., Rothblum, G. N., and Vadhan, S. Boosting and differential privacy. In *Proceedings of the 2010 IEEE 51st Annual Symposium on Foundations of Computer Science*, FOCS '10, pp. 51–60, Washington, DC, USA, 2010. IEEE Computer Society. ISBN 978-0-7695-4244-7. doi: 10.1109/FOCS.2010.12. URL http://dx.doi.org/10.1109/FOCS.2010.12.

Hardt, M., Price, E., , and Srebro, N. Equality of opportunity in supervised learning. In Lee, D. D., Sugiyama, M., Luxburg, U. V., Guyon, I., and Garnett, R. (eds.), *Advances in Neural Information Processing Systems 29*, pp. 3315–3323. Curran Associates, Inc., 2016. URL http://papers.nips.cc/paper/6374-equality-of-opportunity-in-supervised-learning.pdf.

Kearns, M., Neel, S., Roth, A., and Wu, Z. S. An empirical study of rich subgroup fairness for machine learning. 2018a. URL arXiv:1808.08166v1.

Kearns, M., Neel, S., Roth, A., and Wu, Z. S. Preventing fairness gerrymandering: Auditing and learning for subgroup fairness. In Dy, J. and Krause, A. (eds.), *Proceedings of the 35th International Conference on Machine Learning*, volume 80 of *Proceedings of Machine Learning Research*, pp. 2564–2572, Stockholmsmässan, Stockholm Sweden, 10–15 Jul 2018b. PMLR. URL http://proceedings.mlr.press/v80/kearns18a.html.

Kilbertus, N., Gascón, A., Kusner, M. J., Veale, M., Gummadi, K. P., and Weller, A. Blind justice: Fairness with encrypted sensitive attributes. 2018. URL arXiv:1806.03281v1.

McSherry, F. and Talwar, K. Mechanism design via differential privacy. In *Proceedings of the 48th Annual IEEE Symposium on Foundations of Computer Science*, FOCS '07, pp. 94–103, Washington, DC, USA, 2007. IEEE Computer Society. ISBN 0-7695-3010-9. doi: 10.1109/FOCS.2007.41. URL http://dx.doi.org/10.1109/FOCS.2007.41.

Smith, A., Thakurta, A., and Upadhyay, J. Is interaction necessary for distributed private learning? In *Security and Privacy (SP), 2017 IEEE Symposium on*, pp. 58–77. IEEE, 2017.

Song, C., Ristenpart, T., and Shmatikov, V. Machine learning models that remember too much. In *Proceedings of the 2017 ACM SIGSAC Conference on Computer and Communications Security*, pp. 587–601. ACM, 2017.

Veale, M. and Binns, R. Fairer machine learning in the real world: Mitigating discrimination without collecting sensitive data. *Big Data & Society*, 4(2):2053951717743530, 2017.