Differentially Private Fair Learning

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

[1] *University of Pennsylvania*
[2] *Northeastern University*

February 6, 2019

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. We design two learning algorithms that simultaneously promise *differential privacy* and *equalized odds*, a "fairness" condition that corresponds to equalizing false positive and negative rates across protected groups. 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] that can be used to find the *optimal* fair classifier, given access to a subroutine that can solve the original (not necessarily fair) learning problem. This algorithm 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 paper [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 their 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 trusted party, such as a regulatory agency, who does have access to the race data for the same consumers. Together the company and the regulator 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 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 [Kilbertus et al., 2018], 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 $Y \in \{0, 1\}$ 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* approach 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.

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.

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 and 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 (SMC). 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 and 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 and Talwar, 2007] and composition theorems [Dwork et al., 2006a, Dwork et al., 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 attributes 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.

2.1 Notations

- $\mathbb{P}$ and $\mathbb{E}$ refer to the probability and expectation operators taken with respect to the true underlying distribution $\mathcal{P}$. $\widehat{\mathbb{P}}$ and $\widehat{\mathbb{E}}$ are the corresponding empirical versions.

- We will use notation $\text{FP}_a(\widehat{Y})$ and $\text{TP}_a(\widehat{Y})$ to refer to the false and true positive rates of $\widehat{Y}$ on the subpopulation $\{A = a\}$.

$$\text{FP}_a(\widehat{Y}) = \mathbb{P}\left[\widehat{Y} = 1 \mid A = a, Y = 0\right] \quad , \quad \text{TP}_a(\widehat{Y}) = \mathbb{P}\left[\widehat{Y} = 1 \mid A = a, Y = 1\right]$$

  $\widehat{\text{FP}}_a(\widehat{Y})$ and $\widehat{\text{TP}}_a(\widehat{Y})$ are used to refer to the empirical false and true positive rates. $\Delta\text{FP}_a(\widehat{Y}) = |\text{FP}_a(\widehat{Y}) - \text{FP}_0(\widehat{Y})|$ and $\Delta\text{TP}_a(\widehat{Y}) = |\text{TP}_a(\widehat{Y}) - \text{TP}_0(\widehat{Y})|$ are used to measure $\widehat{Y}$'s false and true positive rate discrepancies across groups. $\Delta\widehat{\text{FP}}_a(\widehat{Y})$ and $\Delta\widehat{\text{TP}}_a(\widehat{Y})$ are the corresponding empirical versions.

- $\hat{q}_{\hat{y}ay} = \widehat{\mathbb{P}}[\widehat{Y} = \hat{y}, A = a, Y = y]$ is the empirical fraction of the data with $\widehat{Y} = \hat{y}, A = a$, and $Y = y$. With slight abuse of notation, we will use $\hat{q}_{ay} = \widehat{\mathbb{P}}[A = a, Y = y] = \hat{q}_{1ay} + \hat{q}_{0ay}$ to denote the empirical fraction of the data with $A = a$ and $Y = y$. We will see that $\min_{a,y} \hat{q}_{ay}$ shows up in our analyses and plays a role in the performance of our algorithms.

- $\widehat{\mathrm{err}}\,(\widehat{Y}) = \widehat{\mathbb{P}}[\widehat{Y} \neq Y]$ is the training error of the classifier $\widehat{Y}$. Given a randomized classifier $Q \in \Delta(\mathcal{H})$, $\widehat{\mathrm{err}}\,(Q) = \mathbb{E}_{h\sim Q}\left[\widehat{\mathbb{P}}[h(X) \neq Y]\right]$.

2.2 Fairness

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

$$\left| FP_a(\widehat{Y}) - FP_{a'}(\widehat{Y}) \right| \leqslant \gamma \quad , \quad \left| TP_a(\widehat{Y}) - TP_{a'}(\widehat{Y}) \right| \leqslant \gamma$$

The above constraint involves quadratically many inequalities in $|\mathcal{A}|$. It will be more convenient to instead work with a slightly different formulation of γ-Equalized Odds in which we constrain the difference between false and true positive rates in the subpopulation $\{A = a\}$ and the corresponding rates for $\{A = 0\}$ to be at most γ for all $a \neq 0$. The choice of group 0 as an anchor is arbitrary and without loss of generality. The result is a set of only linearly many constraints. For all $a \in \mathcal{A}$:

$$\Delta FP_a(\widehat{Y}) = \left| FP_a(\widehat{Y}) - FP_0(\widehat{Y}) \right| \leqslant \gamma \quad , \quad \Delta TP_a(\widehat{Y}) = \left| TP_a(\widehat{Y}) - TP_0(\widehat{Y}) \right| \leqslant \gamma$$

Since the distribution $\mathcal{P}$ is not known, we will work with empirical versions of the above quantities, in which all the probabilities will be taken with respect to the empirical distribution of the observed data $\widehat{\mathcal{P}}$. Since we will generally be dealing with this definition of fairness, we will use the shortened term "γ-*fair*" throughout the paper to refer to "γ-*Equalized Odds fair*".

2.3 Differential Privacy

Let $\mathcal{D}$ be a *data universe* from which a database D of size m is drawn and let M be an algorithm that takes the database D as input and outputs $M(D) \in \mathcal{O}$. Informally speaking, 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 where closeness are measured by the privacy 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 $O \subseteq \mathcal{O}$,*

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

where $\mathbb{P}$ is taken with respect to the randomness of M. if $\delta = 0$, M is said to be ϵ-DP.

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 $D_I \in (\mathcal{X}, \mathcal{Y})^m$ represents the insensitive attributes and $D_S \in \mathcal{A}^m$ represents 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 $D_S \sim D'_S \in \mathcal{A}^m$ and for all $O \subseteq \mathcal{O}$, we have:

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

Differentially private mechanisms usually work by deliberately injecting perturbations into quantities computed from the sensitive data set, and used as part of the computation. The injected perturbation is sometimes "explicitly" in the form of a (zero-mean) noise sampled from a known distribution, say Laplace or Gaussian, where the scale of noise is calibrated to the sensitivity of the query function to the input data. However, in some other cases, the noise is "implicitly" injected by maintaining a distribution over a set of possible outcomes for the algorithm and outputting a sample from that distribution. The *Laplace* or *Gaussian* mechanisms which are two standard techniques to achieve differential privacy follow the former approach by adding Laplace or Gaussian noise of appropriate scale to the outcome of computation, respectively. The *Exponential* mechanism instead falls into the latter case and is often used when an object, say a classifier, with optimal utility is to be chosen privately. In the setting of this paper, to guarantee the privacy of the sensitive attribute A in our algorithms, we will be using the Laplace and the Exponential Mechanisms which are briefly reviewed below. See [Dwork and Roth, 2014] for a more detailed discussion and analysis.

Let's start with the Laplace mechanism which, as stated before, perturbs the given query function f with zero-mean Laplace noise calibrated to the ℓ_1-sensitivity of the query function. The ℓ_1-sensitivity of a function is essentially how much a function would change in ℓ_1 norm if one changed at most one entry of the database.

Definition 2.3 (ℓ_1-sensitivity of a function)**.** *The ℓ_1-sensitivity of $f : \mathcal{D}^m \to \mathbb{R}^k$ is*

$$\Delta f = \max_{\substack{D, D' \in \mathcal{D}^m \\ D \sim D'}} \left\| f(D) - f(D') \right\|_1$$

Definition 2.4 (Laplace Mechanism [Dwork et al., 2006b])**.** *Given a query function $f : \mathcal{D}^m \to \mathbb{R}^k$, a database $D \in \mathcal{D}^m$, and a privacy parameter ϵ, the Laplace mechanism outputs:*

$$\widetilde{f}_{\epsilon}\left(D\right) = f\left(D\right) + \left(W_1, \ldots, W_k\right)$$

where W_i's are i.i.d. random variables drawn from Lap$\left(\Delta f / \epsilon\right)$.

Keep in mind that besides having privacy, we would like the privately computed query $\widetilde{f}_{\epsilon}(D)$ to have some reasonable accuracy. The following theorem which uses standard tail bounds for a Laplace random variable formalizes the tradeoff between privacy and accuracy for the Laplace mechanism.

Theorem 2.1 (Privacy vs. Accuracy of the Laplace Mechanism [Dwork et al., 2006b])**.** *The Laplace mechanism guarantees ϵ-differential privacy and that with probability at least $1-\delta$,*

$$||\widetilde{f}_{\epsilon}\left(D\right)-f\left(D\right)||_{\infty}\leqslant\ln\left(\frac{k}{\delta}\right)\cdot\left(\frac{\Delta f}{\epsilon}\right)$$

While the Laplace mechanism is often used when the task at hand is to calculate a bounded numeric query (e.g. mean, median), the Exponential mechanism is used when the goal is to output an object (e.g. a classifier) with maximum utility (i.e. minimum loss). To formalize the exponential mechanism, let $\ell:\mathcal{D}^{m}\times\mathcal{H}\to\mathbb{R}$ be a loss function that given an input database $D\in\mathcal{D}^{m}$ and $h\in\mathcal{H}$, specifies the loss of h on D by $\ell\left(D,h\right)$. Without a privacy constraint, the goal would be to output $\arg\min_{h\in\mathcal{H}}\ell\left(D,h\right)$ for the given database D, but when privacy is required, the private algorithm must output $\arg\min_{h\in\mathcal{H}}\ell\left(D,h\right)$ with some "perturbation" which is formalized in the following definition. Let $\Delta\ell$ be the sensitivity of the loss function ℓ with respect to the database argument D. In other words,

$$\Delta\ell=\max_{h\in\mathcal{H}}\max_{\substack{D,D'\in\mathcal{D}^{m}\\ D\sim D'}}\left|\ell\left(D,h\right)-\ell\left(D',h\right)\right|$$

Definition 2.5 (Exponential Mechansim [McSherry and Talwar, 2007])**.** *Given a database $D\in\mathcal{D}^{m}$ and a privacy parameter ϵ, output $h\in\mathcal{H}$ with probability proportional to $\exp\left(-\epsilon\ell(D,h)/2\Delta\ell\right)$.*

Theorem 2.2 (Privacy vs. Accuracy of the Exponential Mechanism [McSherry and Talwar, 2007])**.** *Let $h^{\star}=\arg\min_{h\in\mathcal{H}}\ell\left(D,h\right)$ and $\widetilde{h}_{\epsilon}\in\mathcal{H}$ be the output of the Exponential mechanism. We have that $\widetilde{h}_{\epsilon}$ is ϵ-DP and that with probability at least $1-\delta$,*

$$|\ell\left(D,\widetilde{h}_{\epsilon}\right)-\ell\left(D,h^{\star}\right)|\leqslant\ln\left(\frac{|\mathcal{H}|}{\delta}\right)\cdot\left(\frac{2\Delta\ell}{\epsilon}\right)$$

We will discuss some important properties of differential privacy such as *post-processing* and *Composition Theorems* in Appendix A.

3 Differentially Private 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 is a private variant of the fair learning algorithm introduced in [Hardt et al., 2016] where decisions made by an arbitrary base classifier $\widehat{Y}$ have their false and true positive rates equalized across different groups $\{A=a\}$ in a post-processing step. Due to the desire for privacy of the sensitive attribute A, we assume the base classifier $\widehat{Y}$ is trained only on the unprotected attributes X and that A is used only for the post-processing step.

The proposed algorithm of [Hardt et al., 2016] derives 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 $p_{\hat{y}a}:=\mathbb{P}\left[\widehat{Y}_p=1\,|\,\widehat{Y}=\hat{y},A=a\right]$. Among all fair $\widehat{Y}_p$'s, the one with minimum error can be found by solving a linear program whose coefficients depend only on the $\hat{q}_{\hat{y}ay}$ quantities, and thus privacy will be achieved if these quantities are calculated privately using the Laplace mechanism. Once we do this, the differential privacy guarantees of the algorithm will follow from the post-processing property (Lemma A.1). While the approach is straightforward and simply implementable, the privately learned classifier will need to explicitly take as input the sensitive attribute A at test time which is not feasible (or legal) in all applications.

We have the **DP-postprocessing** algorithm written in Algorithm 1. Notice as discussed above, to guarantee differential privacy of the protected attribute, 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 $\widetilde{\mathrm{LP}}$ (1). In this linear program, terms with tildes (e.g. $\tilde{q}_{ay}$, $\widetilde{\mathrm{err}}$, $\widetilde{\mathrm{FP}}$, $\widetilde{\mathrm{TP}}$) are defined with respect to $\tilde{q}_{\hat{y}ay}$ instead of $\hat{q}_{\hat{y}ay}$. We analyze the performance of Algorithm 1 in Theorem 3.1. Its proof is deferred to Appendix B.2. The main step of the proof is to understand how the introduced noise propagates to the solution of the linear program. We aill also briefly review the fair learning approach of [Hardt et al., 2016] in Appendix B.1.

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

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

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

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

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

Algorithm 1: ϵ-differentially private 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 $\{(X_i, Y_i)\}_{i=1}^m$.
  ➔ Calculate $\hat{q}_{\hat{y}ay} = \widehat{\mathbb{P}}[\widehat{Y} = \hat{y}, A = a, Y = y]$.
  ➔ Sample $W_{\hat{y}ay} \overset{i.i.d.}{\sim} \mathrm{Lap}\,(2/m\epsilon)$ for all $\hat{y}, a, y$.
  ➔ Perturb each $\hat{q}_{\hat{y}ay}$: $\tilde{q}_{\hat{y}ay} = \hat{q}_{\hat{y}ay} + W_{\hat{y}ay}$.
  ➔ Solve $\widetilde{\mathrm{LP}}$ (1) to get the minimizer $\tilde{p}^\star$.
Output: $\tilde{p}^\star$, the trained classifier $\widehat{Y}$
```

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* $\tilde{p}^\star$ *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: its main virtue is simplicity. In the next section, we analyze a more complicated algorithm that is competitive with the optimal fair classifier.

4 Differentially Private Fair Learning: In-processing

γ-fair ERM Problem

$$\begin{aligned} \min_{Q \in \Delta(\mathcal{H})} \quad & \widehat{\text{err}}\left(Q\right) \\ \text{s.t. } \forall \underset{a\neq 0}{a \in \mathcal{A}}: \quad & \Delta\widehat{\text{FP}}_a(Q) \leqslant \gamma \\ & \Delta\widehat{\text{TP}}_a(Q) \leqslant \gamma \end{aligned} \tag{2}$$

In this section we will introduce our second differentially private fair learning algorithm which will be called **DP-oracle-learner** and is based on the algorithm presented in [Agarwal et al., 2018]. Essentially, [Agarwal et al., 2018] reduces the γ-fair learning problem (2) into the following Lagrangian min-max problem:

$$\min_{Q \in \Delta(\mathcal{H})} \quad \max_{\boldsymbol{\lambda} \in \Lambda} \quad L(Q, \boldsymbol{\lambda}) := \widehat{\text{err}}\left(Q\right) + \boldsymbol{\lambda}^\top \hat{\boldsymbol{r}}\left(Q\right) \tag{3}$$

Here $\mathcal{H}$ is a given class of binary classifiers with $d_{\mathcal{H}} = VCD(\mathcal{H}) < \infty$ and $\Delta(\mathcal{H})$ is the set of all randomized classifiers that can be obtained by functions in $\mathcal{H}$. $\hat{\boldsymbol{r}}\left(Q\right)$ 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 where the bound B is chosen to ensure convergence. In this work,

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

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 finds fairness violations. In an iterative framework, having the learner play its best response and the auditor play a no-regret learning algorithm (we use exponentiated gradient descent, or "multiplicative weights") guarantees convergence of the average plays to the equilibrium ([Freund and Schapire, 1996]).

In Algorithm 3, to make the above approach differentially private, Laplace mechanism is used by the Auditor when computing the gradients and we let the Learner run the exponential mechanism (or some other private learning oracle) to compute approximate best responses. This is the differentially private equivalent of assuming access to a perfect oracle, as is done in [Agarwal et al., 2018, Kearns et al., 2018b]. In practice, the exponential mechanism would be substituted for a computationally efficient private learner with heuristic accuracy guarantees. 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 of this section, as well as a brief review of [Agarwal et al., 2018]'s approach for the fair learning problem without privacy constraints, will appear in Appendix C.

We assume in this section that the VC dimension of $\mathcal{H}$ $(= d_{\mathcal{H}})$ is finite, in which case the set of strategies for the Learner reduces to $\Delta(\mathcal{H}(S))$, where $\mathcal{H}(S)$ is the set of all possible labellings induced on $S := \{X_i\}_{i=1}^m$ by $\mathcal{H}$. In other words, $\mathcal{H}(S) = \{(h(X_1), \ldots, h(X_m)) | h \in \mathcal{H}\}$ and recall that $|\mathcal{H}(S)| \leqslant O(m^{d_{\mathcal{H}}})$ by Sauer's Lemma. Note that since the privacy of the protected attribute A is required, we need A to be excluded from the domain of functions in $\mathcal{H}$ ("A-blind classification") and accordingly, from the set S. Because otherwise there might be some privacy loss of A through using $\mathcal{H}(S)$ as the range of the exponential mechanism for the private Learner. This assumption is of course not necessary if one is willing to instead assume $|\mathcal{H}| < \infty$. We will have a discussion later where we state our guarantees assuming $|\mathcal{H}| < \infty$ instead of $d_{\mathcal{H}} < \infty$. Note that having $\mathcal{H}(S)$ as the range of the exponential mechanism used by the private Learner implies the privacy of the unprotected attributes X is *not* guaranteed. However, in the more general setting where $|\mathcal{H}| < \infty$ is assumed, the privacy of the unprotected attributes comes for free as there will be no reduction of $\mathcal{H}$ to $\mathcal{H}(S)$.

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
    $C_i^0 \leftarrow \mathbb{1}\{Y_i \neq 0\}$
    $C_i^1 \leftarrow \mathbb{1}\{Y_i \neq 1\} + \frac{\lambda_{(A_i,Y_i,+)} - \lambda_{(A_i,Y_i,-)}}{\hat{q}_{A_i Y_i}} \mathbb{1}\{A_i \neq 0\} - \sum_{\substack{a \in \mathcal{A} \\ a \neq 0}} \frac{\lambda_{(a,Y_i,+)} - \lambda_{(a,Y_i,-)}}{\hat{q}_{A_i Y_i}} \mathbb{1}\{A_i = 0\}$
end
Call $\text{CSC}_{\epsilon'}(\mathcal{H})$ with $\{X_i, C_i^0, C_i^1\}_{i=1}^m$ to get $h^\star$.
Output: $h^\star$
```

We first bound the regret of the Learner and the Auditor in Lemma 4.1 and 4.2 by understanding how the introduced noise affect these regrets. Proofs of these Lemmas follow from the "sensitivity" and "accuracy" of the private players which are all stated and proved in Appendix C.2.

```
Algorithm 3: $(\epsilon, \delta)$-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 $\{(X_i, A_i, Y_i)\}_{i=1}^{m}$

    $T \leftarrow \frac{B\sqrt{\ln(4|\mathcal{A}|-3)}\, m\, \epsilon}{2\,(2|\mathcal{A}|B+1)\sqrt{\ln(1/\delta)}\,(d_{\mathcal{H}}\ln(m)+\ln(2/\beta))}, \quad \eta \leftarrow \frac{1}{2}\sqrt{\frac{\ln(4|\mathcal{A}|-3)}{T}}$

$\widetilde{\boldsymbol{\theta}}_1 \leftarrow \mathbf{0} \in \mathbb{R}^{4(|\mathcal{A}|-1)}$
for $t = 1, \dots, T$ do
    $\widetilde{\lambda}_{t,k} \leftarrow B\frac{\exp(\widetilde{\theta}_{t,k})}{1+\sum_{k'}\exp(\widetilde{\theta}_{t,k'})}$ for $1 \leqslant k \leqslant 4(|\mathcal{A}|-1)$
    $\widetilde{h}_t \leftarrow \mathrm{BEST}_h^{\epsilon'}(\widetilde{\boldsymbol{\lambda}}_t)$ with $\epsilon' = \epsilon/(4\sqrt{T\ln(1/\delta)})$
    Sample $\boldsymbol{W}_t \in \mathbb{R}^{4(|\mathcal{A}|-1)}$ where $W_{t,k} \overset{i.i.d.}{\sim} \mathrm{Lap}\,(\frac{8|\mathcal{A}|\sqrt{T\ln(1/\delta)}}{(\min_{a,y}\{\hat{q}_{ay}\}\, m-1)\cdot\epsilon})$
    $\widetilde{\boldsymbol{r}}_t \leftarrow \widehat{\boldsymbol{r}}_t\,(\widetilde{h}_t) + \boldsymbol{W}_t$
    $\widetilde{\boldsymbol{\theta}}_{t+1} \leftarrow \widetilde{\boldsymbol{\theta}}_t + \eta\widetilde{\boldsymbol{r}}_t$
end
$\widetilde{Q} \leftarrow \frac{1}{T}\sum_{t=1}^{T}\widetilde{h}_t, \quad \widetilde{\boldsymbol{\lambda}} \leftarrow \frac{1}{T}\sum_{t=1}^{T}\widetilde{\boldsymbol{\lambda}}_t$
Output: $(\widetilde{Q}, \widetilde{\boldsymbol{\lambda}})$
```

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

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

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

$$\frac{1}{T}\max_{\boldsymbol{\lambda}\in\Lambda}\sum_{t=1}^{T} L(\widetilde{h}_t, \boldsymbol{\lambda}) - \frac{1}{T}\sum_{t=1}^{T} L(\widetilde{h}_t, \widetilde{\boldsymbol{\lambda}}_t) \leqslant \frac{B\ln(4|\mathcal{A}|-3)}{\eta T} + 4\eta B\left(1 + \frac{4|\mathcal{A}|\sqrt{T\ln(1/\delta)}\ln(8T|\mathcal{A}|/\beta)}{(\min_{a,y}\{\hat{q}_{ay}\}\, m-1)\cdot\epsilon}\right)^2$$

Now in Theorem 4.3, given the regret bounds of Lemma 4.1 and 4.2, we can characterize the average plays of both players. This theorem provides a formal guarantee that the output $(\widetilde{Q}, \widetilde{\boldsymbol{\lambda}})$ of Algorithm 3 forms a "ν-approximate equilibrium" of the game between the Learner and the Auditor (where ν is specified in the theorem). This property essentially means neither play would gain more than ν if they palyed an strategy other than the ones output by the Algorithm.

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.,*

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

and that

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

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 the main theorem of this subsection that provides high probability bounds on the accuracy and fairness violation of the output $\widetilde{Q}$ of Algorithm 3. These bounds can be viewed as revealing the inherent tradeoff between privacy of the algorithm and accuracy or fairness of the output classifier where a stronger privacy guarantee (i.e. smaller ϵ and δ) will lead to weaker accuracy and fairness guarantees.

Theorem 4.4 (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 2. We have that with probability at least $1-\beta$,*

$$\widehat{err}\,(\widetilde{Q}) \leqslant \widehat{err}\,(Q^\star) + 2\nu$$

and for all $a \neq 0$,

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

where

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

Remark 4.1. *Notice the bounds stated above reveal a tradeoff between accuracy and fairness violation that we may control through the parameter B. As B gets increased, the upper bound on error will get looser while the one on fairness violation gets tighter. We will consider a setting in the next subsection where we can remove this extra tradeoff and choose B as small as possible — at the cost of requiring that the classifiers be able to use protected attributes at test time.*

We assumed so far 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 $\ln(m^{d_{\mathcal{H}}})$ with $\ln(|\mathcal{H}|)$ in Algorithm 3 (when computing the number of iterations T) and in the bounds. See Theorem 4.5 for this generalization.

Theorem 4.5 (Error-Privacy, Fairness-Privacy Tradeoffs)**.** *Suppose $|\mathcal{H}| < \infty$ and let $(\widetilde{Q}, \widetilde{\boldsymbol{\lambda}})$ be the output of Algorithm 3 that runs for*

$$T = \frac{B\sqrt{\ln(4|\mathcal{A}|-3)}\,m\,\epsilon}{2\,(2|\mathcal{A}|B+1)\,\sqrt{\ln(1/\delta)}\,(\ln(|\mathcal{H}|)+\ln(2/\beta))}$$

iterations, and let $Q^\star$ be the solution to the non-private γ-fair ERM problem 2. We have that with probability at least $1-\beta$,

$$\widehat{err}\,(\widetilde{Q}) \leqslant \widehat{err}\,(Q^\star) + 2\nu$$

and for all $a \neq 0$,

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

where

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

4.1 An Extension: 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 Theorem 4.5 can be improved. As a consequence, the tradeoff discussed in Remark 4.1 will be no longer an issue. Thus, in this subsection, we are interested in solving the γ-fair ERM Problem 4 which now only has false positive parity constraints.

γ-fair ERM Problem

$$\begin{aligned} \min_{Q\,\in\,\Delta(\mathcal{H})} \quad & \widehat{\mathrm{err}}\,(Q) \\ \text{s.t. } \forall \underset{a\neq 0}{a \in \mathcal{A}}: \quad & \Delta\widehat{\mathrm{FP}}_a(Q) \leqslant \gamma \end{aligned} \tag{4}$$

Assumption 4.1. *$\mathcal{H}$ includes all maximally discriminatory classifiers (i.e. 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 4.6 (Error-Privacy, Fairness-Privacy Tradeoffs)**.** *Suppose $|\mathcal{H}| < \infty$, $B > |A| - 1$, and let Assumption 4.1 hold. Let $(\widetilde{Q}, \widetilde{\boldsymbol{\lambda}})$ be the output of Algorithm 3, and let $Q^\star$ be the solution to the γ-fair ERM problem 4. We have that with probability at least $1-\beta$,*

$$\widehat{err}\,(\widetilde{Q}) \leqslant \widehat{err}\,(Q^\star) + 2\nu$$

and for all $a \neq 0$,

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

As an immediate consequence of Theorem 4.6, we have the following Corollary where $B = |\mathcal{A}|$ can be chosen to get bounds which are now free of B.

Corollary 4.6.1. *Under assumptions stated in Theorem 4.6, one can choose* $B = |\mathcal{A}|$ *in Algorithm 3, in which case with probability at least* $1-\beta$*,*

$$\widehat{err}\,(\widetilde{Q}) \leqslant \widehat{err}\,(Q^\star) + 2\nu$$

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

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

where

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

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

In this subsection we show that the sensitivity of the accuracy of the optimal classifier subject to fairness constraints can be substantially higher if it is prohibited from using sensitive attributes at test time. This implies that higher error must be introduced when estimating this accuracy 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. Given a data set D of m individuals, define $f(D)$ to be the optimal error rate in the γ-fair ERM problem 4 which is constrained to have a false positive rate disparity of at most γ.

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. Consider two other classifiers h_R and h_B that depend on the protected attribute: $h_R(X, A) = \mathbb{1}_{A=R}$ and $h_B(X, A) = \mathbb{1}_{A=B}$.

Theorem 4.7. *Consider* $\gamma > 1/m$ *and data sets 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* $\Omega(1/(\gamma m))$*. If the "maximally discriminatory" classifier* h_R *and* h_B *are included in* $\mathcal{H}$ *as well, i.e.* $\mathcal{H} = \{h_0, h_U, h_R, h_B\}$*, the sensitivity of* f *is* $O(1/m)$*.*

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

[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.

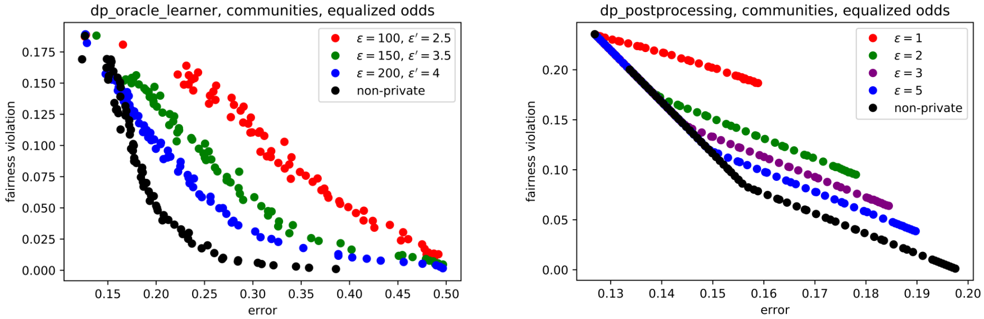

Figure 1: Left figure shows the Pareto frontier of error and (equalized odds) fairness violation for the DP-oracle-learner algorithm on the Communities dataset across different privacy parameters . Right figure shows the corresponding Pareto curves for the DP-postprocessing algorithm. Each point on the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

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 ($m \approx 2$K), 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 $m \approx 20$K. 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 dataset that was upsampled by a factor of 10.

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.

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.

References

[ACM, 2019] ACM (2019). ACM Conference on Fairness, Accountability and Transparency.

[Agarwal et al., 2018] Agarwal, A., Beygelzimer, A., Dudik, M., Langford, J., and Wallach, H. (2018). A reductions approach to fair classification. arXiv:1803.02453v3.

[Chouldechova and Roth, 2018] Chouldechova, A. and Roth, A. (2018). The frontiers of fairness in machine learning.

[Dwork et al., 2006a] Dwork, C., Kenthapadi, K., McSherry, F., Mironov, I., and Naor, M. (2006a). Our data, ourselves: Privacy via distributed noise generation. In Vaudenay, S., editor, *Advances in Cryptology - EUROCRYPT 2006*, pages 486–503, Berlin, Heidelberg. Springer Berlin Heidelberg.

[Dwork et al., 2006b] Dwork, C., McSherry, F., Nissim, K., and Smith, A. (2006b). Calibrating noise to sensitivity in private data analysis. In Halevi, S. and Rabin, T., editors, *Theory of Cryptography*, pages 265–284, Berlin, Heidelberg. Springer Berlin Heidelberg.

[Dwork and Roth, 2014] Dwork, C. and Roth, A. (2014). The algorithmic foundations of differential privacy. *Foundations and Trends® in Theoretical Computer Science*, 9(3–4):211–407.

[Dwork et al., 2010] Dwork, C., Rothblum, G. N., and Vadhan, S. (2010). Boosting and differential privacy. In *Proceedings of the 2010 IEEE 51st Annual Symposium on Foundations of Computer Science*, FOCS '10, pages 51–60, Washington, DC, USA. IEEE Computer Society.

[Freund and Schapire, 1996] Freund, Y. and Schapire, R. E. (1996). Game theory, on-line prediction and boosting. In *Proceedings of the Ninth Annual Conference on Computational Learning Theory*, COLT '96, pages 325–332, New York, NY, USA. ACM.

[Hardt et al., 2016] Hardt, M., Price, E., , and Srebro, N. (2016). Equality of opportunity in supervised learning. In Lee, D. D., Sugiyama, M., Luxburg, U. V., Guyon, I., and Garnett, R., editors, *Advances in Neural Information Processing Systems 29*, pages 3315–3323. Curran Associates, Inc.

[Kearns et al., 2018a] Kearns, M., Neel, S., Roth, A., and Wu, Z. S. (2018a). An empirical study of rich subgroup fairness for machine learning.

[Kearns et al., 2018b] Kearns, M., Neel, S., Roth, A., and Wu, Z. S. (2018b). Preventing fairness gerrymandering: Auditing and learning for subgroup fairness.

[Kilbertus et al., 2018] Kilbertus, N., Gascón, A., Kusner, M. J., Veale, M., Gummadi, K. P., and Weller, A. (2018). Blind justice: Fairness with encrypted sensitive attributes. arXiv: 1806.03281v1.

[McSherry and Talwar, 2007] McSherry, F. and Talwar, K. (2007). Mechanism design via differential privacy. In *Proceedings of the 48th Annual IEEE Symposium on Foundations of Computer Science*, FOCS '07, pages 94–103, Washington, DC, USA. IEEE Computer Society.

[Shalev-Shwartz, 2012] Shalev-Shwartz, S. (2012). Online learning and online convex optimization. *Foundations and Trends® in Machine Learning*, 4(2):107–194.

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

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

A Appendix for Models and Preliminaries: Differential Privacy

An important property of differential privacy is that it is robust to *post-processing*. The post-processing of an (ϵ, δ)-DP algorithm output remains (ϵ, δ)-DP.

Lemma A.1 (Post-Processing [Dwork et al., 2006b]). *Let $M : \mathcal{D}^m \to \mathcal{O}$ be a (ϵ, δ)-DP algorithm and let $f : \mathcal{O} \to \mathcal{R}$ be any randomized function. We have that the algorithm $f \, o \, M : \mathcal{D}^m \to \mathcal{R}$ is (ϵ, δ)-DP.*

Another important property of differential privacy is that DP algorithms can be composed adaptively with a graceful degradation in their privacy parameters.

Theorem A.2 (Composition [Dwork et al., 2010]). *Let M_t be an (ϵ_t, δ_t)-DP algorithm for $t \in [T]$. We have that the composition $M = (M_1, \ldots, M_T)$ is (ϵ, δ)-DP where $\epsilon = \sum_t \epsilon_t$ and $\delta = \sum_t \delta_t$.*

Following the Composition Theorem A.2, if for instance, an iterative algorithm that runs in T iterations is to be made private with target privacy parameters ϵ and $\delta = 0$, each iteration must be made ϵ/T-DP. This may lead to a huge amount of per iteration noise if T is too large. The Advanced Composition Theorem A.3 instead allows the privacy parameter at each step to scale with $O(\epsilon/\sqrt{T})$.

Theorem A.3 (Advanced Composition [Dwork et al., 2010]). *Suppose $0 < \epsilon < 1$ and $\delta > 0$ are target privacy parameters. Let M_t be a (ϵ', δ')-DP algorithm for all $t \in [T]$. We have that the composition $M = (M_1, \ldots, M_T)$ is $(\epsilon, T\delta' + \delta)$-DP where $\epsilon = 2\epsilon'\sqrt{2T \ln(1/\delta)}$.*

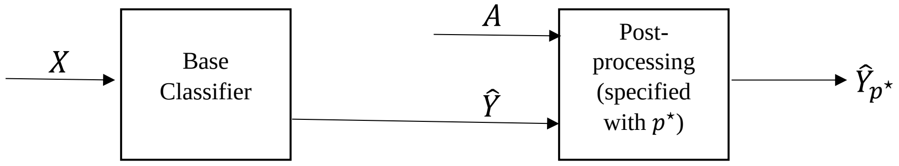

Figure 2: The post-processing technique. In the training phase, training examples are used to train the base classifier and find the optimal $p^\star$ by solving LP (5).

B Appendix for DP Fair Learning: Post-processing

B.1 Fair Learning Approach of [Hardt et al., 2016]

We briefly review the fair learning approach of [Hardt et al., 2016] in this subsection. Suppose there is an arbitrary base classifier $\widehat{Y}$ which is trained on the set of training examples $\{(X_i, Y_i)\}_{i=1}^m$. The goal is to make the classifications of the base classifier γ-fair with respect to the sensitive attribute A by post-processing the predictions given by $\widehat{Y}$. With slight abuse of notation, let $\widehat{Y}_p$ denote the derived optimal γ-fair randomized classifier where $p = (p_{\hat{y}a})_{\hat{y},a}$ is a vector of probabilities describing $\widehat{Y}_p$ and 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 the optimization problem LP (5). Once the optimal solution $p^\star$ is found, one would then use this vector of probabilities, along with the estimate $\widehat{Y}$ given by the base classifier and the sensitive attribute A, to make further predictions. See Fig. 2 for a visual presentation of the adopted model.

LP: Linear Program

$$\begin{aligned}
\underset{p}{\arg\min} \quad & \operatorname{err}\left(\widehat{Y}_p\right) \\
\text{s.t. } \underset{a \neq 0}{\forall a \in \mathcal{A}} \quad & \Delta\mathrm{FP}_a\left(\widehat{Y}_p\right) \leqslant \gamma \\
& \Delta\mathrm{TP}_a\left(\widehat{Y}_p\right) \leqslant \gamma \\
& 0 \leqslant p_{\hat{y}a} \leqslant 1 \quad \forall \hat{y}, a
\end{aligned} \tag{5}$$

Since the true underlying distribution $\mathcal{P}$ is not known, in practice the empirical distribution $\widehat{\mathcal{P}}$ is used to estimate the quantities appearing in LP (5). Using simple probability techniques, one can expand the empirical quantities $\widehat{\operatorname{err}}\,(\widehat{Y}_p)$, $\Delta\widehat{\mathrm{FP}}_a(\widehat{Y}_p)$, and $\Delta\widehat{\mathrm{TP}}_a(\widehat{Y}_p)$ in a linear form in p with coefficients being a function of $\hat{q}_{\hat{y}ay}$ and $\hat{q}_{ay}$ quantities (see $\widehat{\mathrm{LP}}$ (6)).

$\widehat{\text{LP}}$: Empirical Linear Program

$$\begin{aligned} \arg\min_{p} \quad & \widehat{\text{err}}\left(\widehat{Y}_p\right) \\ \text{s.t. } \forall \underset{a\neq 0}{a \in \mathcal{A}} \quad & \Delta\widehat{\text{FP}}_a\left(\widehat{Y}_p\right) \leqslant \gamma \\ & \Delta\widehat{\text{TP}}_a\left(\widehat{Y}_p\right) \leqslant \gamma \\ & 0 \leqslant p_{\hat{y}a} \leqslant 1 \quad \forall \hat{y}, a \end{aligned} \tag{6}$$

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

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

$$\Delta\widehat{\text{TP}}_a\left(\widehat{Y}_p\right) = \left|\widehat{\text{TP}}_a\left(\widehat{Y}\right)\cdot p_{1a} + \left(1 - \widehat{\text{TP}}_1\left(\widehat{Y}\right)\right)\cdot p_{0a} - \widehat{\text{TP}}_0\left(\widehat{Y}\right)\cdot p_{10} - \left(1 - \widehat{\text{TP}}_0\left(\widehat{Y}\right)\right)\cdot p_{00}\right|$$

B.2 Proof of Theorem 3.1

The proof of Theorem 3.1 relies on some facts which are stated here.

Claim B.1 (ℓ_1-Sensitivity of $\hat{\boldsymbol{q}}$ to A)**.** *Let $\hat{\boldsymbol{q}} = \left[\hat{q}_{\hat{y}ay}\right]_{\hat{y},a,y}$ be the empirical distribution of $\{\widehat{Y}, A, Y\}$ and let $\Delta\hat{\boldsymbol{q}}$ be the ℓ_1-sensitivity of $\hat{\boldsymbol{q}}$ to A. We have that*

$$\Delta\hat{\boldsymbol{q}} = \max_{\substack{A, A' \in \mathcal{A}^m \\ A \sim A'}} \left\| \hat{\boldsymbol{q}}\left(A\right) - \hat{\boldsymbol{q}}\left(A'\right)\right\|_1 = \frac{2}{m}$$

Lemma B.2. *Suppose $\min_{a,y}\{\hat{q}_{ay}\} > 4\ln\left(4|\mathcal{A}|/\beta\right)/\left(m\,\epsilon\right)$. we have that with probability $\geqslant 1-\beta$,*

1. $\left|\widetilde{err}\left(\widehat{Y}_p\right) - \widehat{err}\left(\widehat{Y}_p\right)\right| \leqslant \frac{12|\mathcal{A}|\ln(4|\mathcal{A}|/\beta)}{m\epsilon} \quad ;\forall\, p.$
2. $\tilde{q}_{ay} > 0 \quad ;\forall\, a, y.$
3. $\left|\widetilde{FP}_a\left(\widehat{Y}\right) - \widehat{FP}_a\left(\widehat{Y}\right)\right| \leqslant \frac{2\ln(4|\mathcal{A}|/\beta)}{\tilde{q}_{a0}\,m\epsilon}, \quad \left|\widetilde{TP}_a\left(\widehat{Y}\right) - \widehat{TP}_a\left(\widehat{Y}\right)\right| \leqslant \frac{2\ln(4|\mathcal{A}|/\beta)}{\tilde{q}_{a1}\,m\epsilon} \quad ;\forall\, a.$
4. $\left|\Delta\widetilde{FP}_a\left(\widehat{Y}_p\right) - \Delta\widehat{FP}_a\left(\widehat{Y}_p\right)\right| \leqslant \frac{4\ln(4|\mathcal{A}|/\beta)}{\min\{\tilde{q}_{a0},\tilde{q}_{00}\}\,m\epsilon}, \left|\Delta\,\widetilde{TP}_a\left(\widehat{Y}_p\right) - \Delta\,\widehat{TP}_a\left(\widehat{Y}_p\right)\right| \leqslant \frac{4\ln(4|\mathcal{A}|/\beta)}{\min\{\tilde{q}_{a1},\tilde{q}_{01}\}\,m\epsilon}\ ;\forall\, a, p.$
5. $\hat{\text{p}}^\star$*, the optimal solution of* $\widehat{LP}$ *(6), is feasible in* $\widetilde{LP}$ *(1).*

Proof of Lemma B.2. By Claim B.1 and Theorem 2.1, we have that with probability at least $1-\beta$, $||\hat{\boldsymbol{q}} - \tilde{\boldsymbol{q}}||_\infty \leqslant \ln\left(4|\mathcal{A}|/\beta\right)\cdot\left(2/m\epsilon\right)$. Hence with probability $\geqslant 1-\beta$,

1. $\forall\, p$,

$$\left|\widetilde{\text{err}}\,(\widehat{Y}_p) - \widehat{\text{err}}\,(\widehat{Y}_p)\right| \leqslant \sum_{\hat{y},a,y}\left|\tilde{q}_{\hat{y}ay} - \hat{q}_{\hat{y}ay}\right| + \sum_{\hat{y},a}\left|\tilde{q}_{\hat{y}a1} - \hat{q}_{\hat{y}a1}\right| \leqslant \frac{12|\mathcal{A}|\ln(4|\mathcal{A}|/\beta)}{m\epsilon}$$

2. For all a, y,

$$\begin{aligned}|\tilde{q}_{ay} - \hat{q}_{ay}| &= |\tilde{q}_{1ay} + \tilde{q}_{0ay} - \hat{q}_{1ay} - \hat{q}_{0ay}| \\ &\leqslant |\tilde{q}_{1ay} - \hat{q}_{1ay}| + |\tilde{q}_{0ay} - \hat{q}_{0ay}| \\ &\leqslant \frac{4\ln(4|\mathcal{A}|/\beta)}{m\epsilon}\end{aligned}$$

But by the stated assumption, $\hat{q}_{ay} > \frac{4\ln(4|\mathcal{A}|/\beta)}{m\epsilon}$ implying that $\tilde{q}_{ay} > 0$.

3. $\forall\, a$,

$$\begin{aligned}\left|\widetilde{\mathrm{FP}}_a(\widehat{Y}) - \widehat{\mathrm{FP}}_a(\widehat{Y})\right| &= \left|\frac{\tilde{q}_{1a0}}{\tilde{q}_{1a0} + \tilde{q}_{0a0}} - \frac{\hat{q}_{1a0}}{\hat{q}_{1a0} + \hat{q}_{0a0}}\right| \\ &= \left|\frac{\tilde{q}_{1a0}\,\hat{q}_{0a0} - \hat{q}_{1a0}\,\tilde{q}_{0a0}}{(\tilde{q}_{1a0} + \tilde{q}_{0a0})(\hat{q}_{1a0} + \hat{q}_{0a0})}\right| \\ &= \left|\frac{\hat{q}_{0a0}(\tilde{q}_{1a0} - \hat{q}_{1a0}) - \hat{q}_{1a0}(\tilde{q}_{0a0} - \hat{q}_{0a0})}{(\tilde{q}_{1a0} + \tilde{q}_{0a0})(\hat{q}_{1a0} + \hat{q}_{0a0})}\right| \\ &\leqslant \frac{2\ln(4|\mathcal{A}|/\beta)}{|\tilde{q}_{a0}|\, m\epsilon} \\ &= \frac{2\ln(4|\mathcal{A}|/\beta)}{\tilde{q}_{a0}\, m\epsilon} \quad \text{(by Part 2 of this Lemma)}\end{aligned}$$

And similarly,

$$\left|\widetilde{\mathrm{TP}}_a(\widehat{Y}) - \widehat{\mathrm{TP}}_a(\widehat{Y})\right| \leqslant \frac{2\ln(4|\mathcal{A}|/\beta)}{\tilde{q}_{a1}\, m\epsilon}$$

4. Observe that $\forall\, a, p$,

$$\begin{aligned}&\left|\Delta\widetilde{\mathrm{FP}}_a(\widehat{Y}_p) - \Delta\widehat{\mathrm{FP}}_a(\widehat{Y}_p)\right| \\ &\leqslant \Big|\widetilde{\mathrm{FP}}_a(\widehat{Y}) \cdot p_{1a} + (1 - \widetilde{\mathrm{FP}}_a(\widehat{Y})) \cdot p_{0a} - \widetilde{\mathrm{FP}}_0(\widehat{Y}) \cdot p_{10} - (1 - \widetilde{\mathrm{FP}}_0(\widehat{Y})) \cdot p_{00} \\ &\quad - \widehat{\mathrm{FP}}_a(\widehat{Y}) \cdot p_{1a} - (1 - \widehat{\mathrm{FP}}_a(\widehat{Y})) \cdot p_{0a} + \widehat{\mathrm{FP}}_0(\widehat{Y}) \cdot p_{10} + (1 - \widehat{\mathrm{FP}}_0(\widehat{Y})) \cdot p_{00}\Big| \\ &\leqslant \left|\widetilde{\mathrm{FP}}_a(\widehat{Y}) - \widehat{\mathrm{FP}}_a(\widehat{Y})\right| \cdot |p_{1a} - p_{0a}| + \left|\widetilde{\mathrm{FP}}_0(\widehat{Y}) - \widehat{\mathrm{FP}}_0(\widehat{Y})\right| \cdot |p_{10} - p_0| \\ &\leqslant \frac{4\ln(4|\mathcal{A}|/\beta)}{\min\{\tilde{q}_{a0}, \tilde{q}_{00}\}\, m\epsilon} \quad \text{(by part 3 of this Lemma)}\end{aligned}$$

A similar argument holds for $\left|\Delta\widetilde{\mathrm{TP}}_a(\widehat{Y}_p) - \Delta\widehat{\mathrm{TP}}_a(\widehat{Y}_p)\right| \leqslant \frac{4\ln(4|\mathcal{A}|/\beta)}{\min\{\tilde{q}_{a1},\tilde{q}_{01}\}\, m\epsilon}$.

5. We will show that $\widehat{p}^\star$ satisfies the first constraint of $\widetilde{\mathrm{LP}}$ (1) for all $a \in \mathcal{A}$. Satisfying the second constraint can be similarly shown and the third is trivial. We have that

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

by part 4 of this Lemma and the fact that $\left|\Delta\widehat{\mathrm{FP}}_a(\widehat{Y}_{\widehat{p}^\star})\right| \leqslant \gamma$ (see $\widehat{\mathrm{LP}}$ (6)).

□

Proof of Theorem 3.1. Following Lemma B.2, with probability at least $1-\beta$

$$\begin{aligned}
\widehat{\mathrm{err}}\,(\widehat{Y}_{\widetilde{p}^\star}) &\leqslant \widetilde{\mathrm{err}}\,(\widehat{Y}_{\widetilde{p}^\star}) + \frac{12|\mathcal{A}|\ln(4|\mathcal{A}|/\beta)}{m\epsilon} && \text{(part 1 of Lemma B.2)}\\
&\leqslant \widetilde{\mathrm{err}}\,(\widehat{Y}_{\widehat{p}^\star}) + \frac{12|\mathcal{A}|\ln(4|\mathcal{A}|/\beta)}{m\epsilon} && \text{(part 5 of Lemma B.2)}\\
&\leqslant \widehat{\mathrm{err}}\,(\widehat{Y}_{\widehat{p}^\star}) + \frac{24|\mathcal{A}|\ln(4|\mathcal{A}|/\beta)}{m\epsilon} && \text{(part 1 of Lemma B.2)}
\end{aligned}$$

Also, for all $a \neq 0$,

$$\begin{aligned}
\Delta\widehat{\mathrm{FP}}_a\,(\widehat{Y}_{\widetilde{p}^\star}) \;&\leqslant\; \Delta\widetilde{\mathrm{FP}}_a(\widehat{Y}_{\widetilde{p}^\star}) + \frac{4\ln(4|\mathcal{A}|/\beta)}{\min\{\tilde{q}_{a0},\tilde{q}_{00}\}\,m\epsilon} && \text{(part 4 of Lemma B.2)}\\
&\leqslant\; \gamma + \frac{8\ln(4|\mathcal{A}|/\beta)}{\min\{\tilde{q}_{a0},\tilde{q}_{00}\}\,m\epsilon} && \text{(see } \widetilde{\mathrm{LP}} \text{ (1))}\\
&\leqslant\; \gamma + \frac{8\ln(4|\mathcal{A}|/\beta)}{\min\{\hat{q}_{a0},\hat{q}_{00}\}\,m\epsilon - 4\ln(4|\mathcal{A}|/\beta)}
\end{aligned}$$

The last inequality follows from the fact that $|\tilde{q}_{ay} - \hat{q}_{ay}| \leqslant 4\ln(4|\mathcal{A}|/\beta)/m\epsilon$ for all a, y. It follows similarly that,

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

□

C Appendix for DP Fair Learning: In-processing

C.1 Fair Learning Approach of [Agarwal et al., 2018]

Suppose given a class of binary classifiers $\mathcal{H}$, the task is to find the optimal γ-fair classifier in $\Delta(\mathcal{H})$, where $\Delta(\mathcal{H})$ is the set of all randomized classifiers that can be obtained by functions in $\mathcal{H}$. [Agarwal et al., 2018] provided a reduction of the learning problem with only the fairness constraint to a two-player zero-sum game and introduced an algorithm that achieves the lowest empirical error. In this section we mainly discuss their reduction approach which forms the basis of our differentially private fair learning algorithm: **DP-oracle-learner**. Although [Agarwal et al., 2018] considers a general form of a constraint that captures many existing notions of fairness, in this paper, we focus on the Equalized Odds notion of fairness described in Definition 2.1. Our techniques, however, generalize beyond this. To begin with, the γ-fair classification task can be modeled as the constrained optimization problem 7.

γ-fair Learning Problem

$$\begin{aligned}
\min_{Q\in\Delta(\mathcal{H})} \quad & \mathrm{err}\,(Q)\\
\text{s.t. } \forall \underset{a\neq 0}{a \in \mathcal{A}}: \quad & \Delta\mathrm{FP}_a(Q) \leqslant \gamma \qquad (7)\\
& \Delta\mathrm{TP}_a(Q) \leqslant \gamma
\end{aligned}$$

As the data generating distribution $\mathcal{P}$ is unknown, we will be dealing with the Fair Empirical Risk Minimization (ERM) problem 8. In this empirical version, all the probabilities and expectations are taken with respect to the empirical distribution of the data $\widehat{\mathcal{P}}$.

γ-fair ERM Problem

$$\begin{aligned} \min_{Q \in \Delta(\mathcal{H})} \quad & \widehat{\text{err}}\,(Q) \\ \text{s.t. } \forall \underset{a \neq 0}{a \in \mathcal{A}}: \quad & \Delta\widehat{\text{FP}}_a(Q) \leqslant \gamma \\ & \Delta\widehat{\text{TP}}_a(Q) \leqslant \gamma \end{aligned} \tag{8}$$

Toward deriving a fair classification algorithm, the above fair ERM problem 8 will be rewritten as a two-player zero-sum game whose equilibrium is the solution to the problem. Let $\widehat{\boldsymbol{r}}(Q) \in \mathbb{R}^{4(|\mathcal{A}|-1)}$ store all fairness violations of the classifier Q.

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

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

$$L(Q, \boldsymbol{\lambda}) = \widehat{\text{err}}\,(Q) + \boldsymbol{\lambda}^\top \widehat{\boldsymbol{r}}(Q)$$

be the Lagrangian of the optimization problem. We therefore have that the Fair ERM Problem 8 is equivalent to

$$\min_{Q \in \Delta(\mathcal{H})} \quad \max_{\boldsymbol{\lambda} \in \mathbb{R}_+^{4|\mathcal{A}|}} \quad L(Q, \boldsymbol{\lambda})$$

In order to guarantee convergence, we further constrain the ℓ_1 norm of $\boldsymbol{\lambda}$ to be bounded. So let $\Lambda = \{\boldsymbol{\lambda} \in \mathbb{R}_+^{4(|\mathcal{A}|-1)} :\ ||\boldsymbol{\lambda}||_1 \leqslant B\}$ be the feasible space of the dual variable $\boldsymbol{\lambda}$ for some constant B. Hence, the primal and the dual problems are as follows.

$$\begin{aligned} &\text{primal problem:} \quad \min_{Q \in \Delta(\mathcal{H})} \quad \max_{\boldsymbol{\lambda} \in \Lambda} \quad L(Q, \boldsymbol{\lambda}) \\ &\text{dual problem:} \quad \max_{\boldsymbol{\lambda} \in \Lambda} \quad \min_{Q \in \Delta(\mathcal{H})} \quad L(Q, \boldsymbol{\lambda}) \end{aligned}$$

The above primal and dual problems can be shown to have solutions that coincide at a point $(Q^\star, \boldsymbol{\lambda}^\star)$ which is the saddle point of L. From a game theoretic perspective, the saddle point can be viewed as an equilibrium of a zero-sum game between a Learner (Q-player) and an Auditor ($\boldsymbol{\lambda}$-player) where $L(Q, \boldsymbol{\lambda})$ is how much the Learner must pay to the Auditor. Algorithm 5, developed by [Agarwal et al., 2018], proceeds iteratively according to a no-regret dynamic where in each iteration, the Learner plays the best response (BEST_h) to the given play of the Auditor and the Auditor plays exponentiated gradient descent. The average play of both players over T rounds are then taken as the output of the algorithm, which can be shown to converge to the saddle point $(Q^\star, \boldsymbol{\lambda}^\star)$ ([Freund and Schapire, 1996]). [Agarwal et al., 2018] shows how BEST_h

can be solved efficiently having access to the cost-sensitive classification oracle for $\mathcal{H}$ (CSC($\mathcal{H}$)) and we have their reduction for our Equalized Odds notion of fairness written in Subroutine 4.

Assumption C.1 (Cost-Sensitive Classification Oracle for $\mathcal{H}$)**.** *It is assumed that the proposed algorithm has access to CSC$(\mathcal{H})$ which is the cost-sensitive classification oracle for $\mathcal{H}$. This oracle takes as input a set of individual-level attributes and costs $\{X_i, C_i^0, C_i^1\}_{i=1}^m$, and outputs $\arg\min_{h\in\mathcal{H}} \sum_{i=1}^m h(X_i)C_i^1 + (1-h(X_i))\,C_i^0$. In practice, these oracles are implemented using learning heuristics.*

Note that the Learner finds $\arg\min_{Q\in\Delta(\mathcal{H})} L(Q,\boldsymbol{\lambda})$ for a given $\boldsymbol{\lambda}$ of the Auditor and since the Lagrangian L is linear in Q, the minimizer of $L(Q,\boldsymbol{\lambda})$ can be chosen to put all the probability mass on a single classifier $h \in \mathcal{H}$. Additionally, our reduction in Subroutine 4 looks different from the one derived in Example 4 of [Agarwal et al., 2018] since we have our Equalized Odds fairness constraints formulated a bit differently from how it is formulated in [Agarwal et al., 2018].

Subroutine 4: BEST$_h$

```
Input: $\boldsymbol{\lambda}$, training examples $\{(X_i, A_i, Y_i)\}_{i=1}^m$
for $i = 1, \dots, m$ do
    $C_i^0 \leftarrow \mathbb{1}\{Y_i \neq 0\}$
    $C_i^1 \leftarrow \mathbb{1}\{Y_i \neq 1\} + \frac{\lambda_{(A_i,Y_i,+)} - \lambda_{(A_i,Y_i,-)}}{\hat{q}_{A_i Y_i}} \mathbb{1}\{A_i \neq 0\} - \sum_{\substack{a\in\mathcal{A} \\ a\neq 0}} \frac{\lambda_{(a,Y_i,+)} - \lambda_{(a,Y_i,-)}}{\hat{q}_{A_i Y_i}} \mathbb{1}\{A_i = 0\}$
end
Call CSC$(\mathcal{H})$ to find $h^\star = \underset{h\in\mathcal{H}}{\arg\min} \sum_{i=1}^m h(X_i)C_i^1 + (1-h(X_i))\,C_i^0$
Output: $h^\star$
```

Algorithm 5: exp. gradient reduction for fair classification ([Agarwal et al., 2018])

```
Input: fairness violation $\gamma$
       bound B, learning rate $\eta$, number of rounds $T$
       training examples $\{(X_i, A_i, Y_i)\}_{i=1}^m$
$\boldsymbol{\theta}_1 \leftarrow \mathbf{0} \in \mathbb{R}^{4(|\mathcal{A}|-1)}$
for $t = 1, \dots, T$ do
    $\lambda_{t,k} \leftarrow B\frac{\exp(\theta_{t,k})}{1+\sum_{k'}\exp(\theta_{t,k'})}$ for $1 \leqslant k \leqslant 4(|\mathcal{A}|-1)$
    $h_t \leftarrow \text{BEST}_h(\boldsymbol{\lambda}_t)$
    $\boldsymbol{\theta}_{t+1} \leftarrow \boldsymbol{\theta}_t + \eta\,\widehat{\boldsymbol{r}}_t(h_t)$
end
$\widehat{Q} \leftarrow \frac{1}{T}\sum_{t=1}^T h_t, \quad \widehat{\boldsymbol{\lambda}} \leftarrow \frac{1}{T}\sum_{t=1}^T \boldsymbol{\lambda}_t$
Output: $(\widehat{Q}, \widehat{\boldsymbol{\lambda}})$
```

[Agarwal et al., 2018] shows for any $\nu > 0$, and for appropriately chosen η and T, Algorithm 5 under Assumption C.1 returns a pair $(\widehat{Q}, \widehat{\boldsymbol{\lambda}})$ for which

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

that corresponds to a ν-*approximate* equilibrium of the game and it implies neither player can gain more than ν by changing their strategy (see Theorem 1 of [Agarwal et al., 2018]). They further show that any ν-approximate equilibrium of the game achieves an error close to the best error one would hope to get and the amount by which it violates the fairness constraints is reasonably small (see Theorem 2 of [Agarwal et al., 2018]).

C.2 Missing Lemmas and Proofs of Section 4

Lemma C.1 (Sensitivity of the Private Players to A)**.** *Let $\Delta\widehat{\boldsymbol{r}}_t$ and $\Delta\ell_t$ be the sensitivity of $\widehat{\boldsymbol{r}}_t$ (of the Auditor) and ℓ_t (of the Learner) respectively. We have that for all $t \in [T]$,*

$$\Delta\widehat{\boldsymbol{r}}_t = \max_{\substack{A,A' \in \mathcal{A}^m \\ A \sim A'}} ||\widehat{\boldsymbol{r}}_t(A) - \widehat{\boldsymbol{r}}_t(A')||_1 \leqslant \frac{2|\mathcal{A}|}{\min_{a,y}\{\hat{q}_{ay}\}\, m - 1}$$

$$\Delta\ell_t = \max_{h \in \mathcal{H}} \max_{\substack{A,A' \in \mathcal{A}^m \\ A \sim A'}} |\ell_t(h;A) - \ell_t(h;A')| \leqslant \frac{2|\mathcal{A}|B + 1}{\min_{a,y}\{\hat{q}_{ay}\}\, m - 1}$$

Proof of Lemma C.1. Recall that at round t, the private $\boldsymbol{\lambda}$-player is given some $h_t \in \mathcal{H}$ and wants to calculate

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

privately, where for all $a \in \mathcal{A}$, we have that

$$\widehat{\mathrm{FP}}_a(h_t) = \frac{\hat{q}_{1a0}}{\hat{q}_{a0}} = \frac{\hat{q}_{1a0}}{\hat{q}_{1a0} + \hat{q}_{0a0}} \qquad \widehat{\mathrm{TP}}_a(h_t) = \frac{\hat{q}_{1a1}}{\hat{q}_{a1}} = \frac{\hat{q}_{1a1}}{\hat{q}_{1a1} + \hat{q}_{0a1}}$$

Having modified one of the records in $A \in \mathcal{A}^m$, say $A_j = a$ is changed to $A'_j = a'$ for some $j \in [m]$, $\hat{q}_{\hat{y}_j a y_j}$ will then decrease by $1/m$ and $\hat{q}_{\hat{y}' a' y_j}$ will increase by $1/m$ where $\hat{y}'$ may or may not be equal to $\hat{y}_j$. Thus, depending on the value of y_j, it is then the case that

- if $y_j = 0$: $\widehat{\mathrm{FP}}_a(h_t)$ and $\widehat{\mathrm{FP}}_{a'}(h_t)$ will change by at most $1/\left(\min_{a,y}\{\hat{q}_{ay}\}\, m - 1\right)$.
- if $y_j = 1$: $\widehat{\mathrm{TP}}_a(h_t)$ and $\widehat{\mathrm{TP}}_{a'}(h_t)$ will change by at most $1/\left(\min_{a,y}\{\hat{q}_{ay}\}\, m - 1\right)$.

Therefore, since each $\widehat{\mathrm{FP}}_a$ ($\widehat{\mathrm{TP}}_a$) appears twice in $\widehat{\boldsymbol{r}}_t(h_t)$ if $a \neq 0$ and $2(|\mathcal{A}| - 1)$ times if $a = 0$, we have that

$$\Delta\widehat{\boldsymbol{r}}_t \leqslant \frac{2|\mathcal{A}|}{\min_{a,y}\{\hat{q}_{ay}\}\, m - 1}$$

Let's move on to the sensitivity of ℓ_t of the private Q-player. Recall that at round t, the Q-player is given some $\boldsymbol{\lambda}_t \in \Lambda$ and wants to find $\arg\min_{h \in \mathcal{H}} \ell_t(h) \equiv L(h, \boldsymbol{\lambda}_t) = \widehat{\mathrm{err}}\,(h) + \boldsymbol{\lambda}_t^\top \widehat{\boldsymbol{r}}_t(h)$ privately.

It is then obvious that since $||\boldsymbol{\lambda}_t||_1 \leqslant B$,

$$\Delta\ell_t \leqslant \frac{1}{m} + \frac{2|\mathcal{A}|B}{\min_{a,y}\{\hat{q}_{ay}\}\, m - 1} \leqslant \frac{2|\mathcal{A}|B + 1}{\min_{a,y}\{\hat{q}_{ay}\}\, m - 1}$$

□

Lemma C.2 (Accuracy of the Private Players)**.** *At round t of Algorithm 3, let $\widehat{\boldsymbol{r}}_t = \widetilde{\boldsymbol{r}}_t - \boldsymbol{W}_t$ be the noiseless version of $\widetilde{\boldsymbol{r}}_t$ and $h_t^\star$ be the classifier given by the noiseless subroutine $BEST_h(\widetilde{\boldsymbol{\lambda}}_t)$. We have that*

$$w.p. \geqslant 1 - \beta/2T, \quad ||\widetilde{\boldsymbol{r}}_t - \widehat{\boldsymbol{r}}_t||_\infty \leqslant \frac{8|\mathcal{A}|\sqrt{T\ln(1/\delta)}\ln\left(8T|\mathcal{A}|/\beta\right)}{(\min_{a,y}\{\hat{q}_{ay}\}\, m - 1)\cdot\epsilon}$$

$$w.p. \geqslant 1 - \beta/2T, \quad L(\widetilde{h}_t, \widetilde{\boldsymbol{\lambda}}_t) \leqslant L(h_t^\star, \widetilde{\boldsymbol{\lambda}}_t) + \frac{8\left(2|\mathcal{A}|B + 1\right)\sqrt{T\ln\left(1/\delta\right)}\left(d_{\mathcal{H}}\ln\left(m\right) + \ln\left(2T/\beta\right)\right)}{(\min_{a,y}\{\hat{q}_{ay}\}\, m - 1)\cdot\epsilon}$$

Proof of Lemma C.2. Results follow from Lemma C.1, Theorem 2.1 and Theorem 2.2 of this paper. Recall that $|\mathcal{H}(S)| \leqslant O(m^{d_{\mathcal{H}}})$ by Sauer's Lemma. □

Proof of Lemma 4.1. This result follows directly from the accuracy of the private Q-player given in Lemma C.2. □

Proof of Lemma 4.2. We follow the proof given for Theorem 1 of [Agarwal et al., 2018] and modify where necessary. Let $\Lambda' = \{\boldsymbol{\lambda}' \in \mathbb{R}_+^{4|\mathcal{A}|-3} : ||\boldsymbol{\lambda}'||_1 = B\}$. Any $\boldsymbol{\lambda} \in \Lambda$ is associated with a $\boldsymbol{\lambda}' \in \Lambda'$ which is equal to $\boldsymbol{\lambda}$ on the first $4(|\mathcal{A}| - 1)$ coordinates and has the remaining mass on the last one. Let $\widetilde{\boldsymbol{r}}'_t \in \mathbb{R}^{4|\mathcal{A}|-3}$ be equal to $\widetilde{\boldsymbol{r}}_t$ on the first $4(|\mathcal{A}| - 1)$ coordinates and zero in the last one. We have that for any $\boldsymbol{\lambda}$ and its associated $\boldsymbol{\lambda}'$, and particularly $\widetilde{\boldsymbol{\lambda}}_t$ and $\widetilde{\boldsymbol{\lambda}}'_t$ of Algorithm 3, and all t

$$\boldsymbol{\lambda}^\top \widetilde{\boldsymbol{r}}_t = (\boldsymbol{\lambda}')^\top \widetilde{\boldsymbol{r}}'_t \quad , \quad \widetilde{\boldsymbol{\lambda}}_t^\top \widetilde{\boldsymbol{r}}_t = (\widetilde{\boldsymbol{\lambda}}'_t)^\top \widetilde{\boldsymbol{r}}'_t \tag{9}$$

Observe that with probability at least $1-\beta/2T$, $||\widetilde{\boldsymbol{r}}'_t||_\infty = ||\widetilde{\boldsymbol{r}}_t||_\infty \leqslant 2 + \frac{8|\mathcal{A}|\sqrt{T\ln(1/\delta)}\ln(8T|\mathcal{A}|/\beta)}{(\min_{a,y}\{\hat{q}_{ay}\}\, m-1)\cdot\epsilon}$ (see Lemma C.2). Thus, by Corollary 2.14 of [Shalev-Shwartz, 2012], we have that with probability at least $1 - \beta/2$, for any $\boldsymbol{\lambda}' \in \Lambda'$,

$$\sum_{t=1}^{T}(\boldsymbol{\lambda}')^\top \widetilde{\boldsymbol{r}}'_t \leqslant \sum_{t=1}^{T}(\widetilde{\boldsymbol{\lambda}}'_t)^\top \widetilde{\boldsymbol{r}}'_t + \frac{B\ln(4|\mathcal{A}| - 3)}{\eta} + 4\eta B\left(1 + \frac{4|\mathcal{A}|\sqrt{T\ln(1/\delta)}\ln(8T|\mathcal{A}|/\beta)}{(\min_{a,y}\{\hat{q}_{ay}\}\, m - 1)\cdot\epsilon}\right)^2 T$$

Consequently, by Equation 9, we have that with probability at least $1 - \beta/2$, for any $\boldsymbol{\lambda} \in \Lambda$,

$$\sum_{t=1}^{T}\boldsymbol{\lambda}^\top \widetilde{\boldsymbol{r}}_t \leqslant \sum_{t=1}^{T}\widetilde{\boldsymbol{\lambda}}_t^\top \widetilde{\boldsymbol{r}}_t + \frac{B\ln(4|\mathcal{A}| - 3)}{\eta} + 4\eta B\left(1 + \frac{4|\mathcal{A}|\sqrt{T\ln(1/\delta)}\ln(8T|\mathcal{A}|/\beta)}{(\min_{a,y}\{\hat{q}_{ay}\}\, m - 1)\cdot\epsilon}\right)^2 T \tag{10}$$

which completes the proof. □

Proof of Theorem 4.3. Let

$$R_Q := \frac{8(2|\mathcal{A}|B+1)\sqrt{T\ln(1/\delta)}\,(d_{\mathcal{H}}\ln(m)+\ln(2T/\beta))}{(\min_{a,y}\{\hat{q}_{ay}\}\,m-1)\cdot\epsilon}$$

and

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

be the regret bounds of the private Q and $\boldsymbol{\lambda}$ players respectively, and let $\nu := R_Q + R_{\boldsymbol{\lambda}}$. We have that for any $Q \in \Delta(\mathcal{H}(S))$, with probability at least $1-\beta$,

$$\begin{aligned}
L(Q,\widetilde{\boldsymbol{\lambda}}) &= \frac{1}{T}\sum_{t=1}^{T} L(Q,\widetilde{\boldsymbol{\lambda}}_t) \quad \text{(by linearity of } L\text{)}\\
&\geqslant \frac{1}{T}\sum_{t=1}^{T} L(\widetilde{h}_t,\widetilde{\boldsymbol{\lambda}}_t) - R_Q \quad \text{(by Lemma 4.1)}\\
&\geqslant \frac{1}{T}\sum_{t=1}^{T} L(\widetilde{h}_t,\widetilde{\boldsymbol{\lambda}}) - R_{\boldsymbol{\lambda}} - R_Q \quad \text{(by Lemma 4.2)}\\
&= L(\widetilde{Q},\widetilde{\boldsymbol{\lambda}}) - \nu
\end{aligned}$$

Now for any $\boldsymbol{\lambda} \in \Lambda$, with probability at least $1-\beta$,

$$\begin{aligned}
L(\widetilde{Q},\boldsymbol{\lambda}) &= \frac{1}{T}\sum_{t=1}^{T} L(\widetilde{h}_t,\boldsymbol{\lambda}) \quad \text{(by linearity of } L\text{)}\\
&\leqslant \frac{1}{T}\sum_{t=1}^{T} L(\widetilde{h}_t,\widetilde{\boldsymbol{\lambda}}_t) + R_{\boldsymbol{\lambda}} \quad \text{(by Lemma 4.2)}\\
&\leqslant \frac{1}{T}\sum_{t=1}^{T} L(\widetilde{Q},\widetilde{\boldsymbol{\lambda}}_t) + R_{\boldsymbol{\lambda}} + R_Q \quad \text{(by Lemma 4.1)}\\
&= L(\widetilde{Q},\widetilde{\boldsymbol{\lambda}}) + \nu
\end{aligned}$$

Therefore, with probability at least $1-\beta$,

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

and that

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

Plugging in the proposed values of T and η in Algorithm 3 results in

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

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

The following two lemmas are taken from [Agarwal et al., 2018] and are used in the proof of Theorem 4.4 and Theorem 4.6.

Lemma C.3 (Empirical Error Bound [Agarwal et al., 2018])**.** *Let $(\widetilde{Q}, \widetilde{\boldsymbol{\lambda}})$ be any ν-approximate solution of the game described in section 4,i.e.,*

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

For any Q satisfying the fairness constraints of the fair ERM problem, we have that

$$\widehat{err}\,(\widetilde{Q}) \leqslant \widehat{err}\,(Q) + 2\nu$$

Lemma C.4 (Empirical Fairness Violation [Agarwal et al., 2018])**.** *Let $(\widetilde{Q}, \widetilde{\boldsymbol{\lambda}})$ be any ν-approximate solution of the game described in section 4, 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 suppose the fairness constraints of the fair ERM problem are feasible. Then the distribution $\widetilde{Q}$ satisfies

$$\begin{aligned} \max_{a \in \mathcal{A}} \left| \widehat{FP}_a(\widetilde{Q}) - \widehat{FP}_0\left(\widetilde{Q}\right) \right| &\leqslant \gamma + \frac{1 + 2\nu}{B} \\ \max_{a \in \mathcal{A}} \left| \widehat{TP}_a(\widetilde{Q}) - \widehat{TP}_0\left(\widetilde{Q}\right) \right| &\leqslant \gamma + \frac{1 + 2\nu}{B} \end{aligned}$$

Proof of Theorem 4.4. The results follow from Theorem 4.3, Lemma C.3, and Lemma C.4. □

Proof of Theorem 4.6. The stated bound on $\widehat{\text{err}}\,(\widetilde{Q})$ follows from Lemma C.3. Let's now prove the bound on fairness violation. Let, for all $a \in \mathcal{A}$, $\beta_a := (\widehat{\text{FP}}_0(\widetilde{Q}) - \widehat{\text{FP}}_a(\widetilde{Q}) - \gamma)_+$ and $\bar{\beta}_a := (\widehat{\text{FP}}_a(\widetilde{Q}) - \widehat{\text{FP}}_0(\widetilde{Q}) - \gamma)_+$. Notice at most one of β_a and $\bar{\beta}_a$ can be positive.

We are going to construct some deviating strategies: Q and $\boldsymbol{\lambda}$. As shown in the previous subsection, we know $(\widetilde{Q}, \widetilde{\boldsymbol{\lambda}})$ is a ν-approximate equilibrium of the zero-sum game. It implies

$$L(\widetilde{Q}, \boldsymbol{\lambda}) - \nu \leqslant L(\widetilde{Q}, \widetilde{\boldsymbol{\lambda}}) \leqslant L(Q, \widetilde{\boldsymbol{\lambda}}) + \nu.$$

Define $Q = \frac{1}{1+\sum_{a \in \mathcal{A}}(\beta_a + \bar{\beta}_a)}(\widetilde{Q} + \sum_a \beta_a h_a + \hat{\beta}_a \hat{h}_a)$. It is easy to see that, for all $a \in \mathcal{A}$,

$$\Delta\widehat{\text{FP}}_a(Q) \preceq \gamma.$$

Then we have

$$
\begin{aligned}
& L(Q,\widetilde{\boldsymbol{\lambda}}) + \nu \\
\leqslant & \widehat{\text{err}}\,(Q) + \nu \\
\leqslant & \widehat{\text{err}}\,\left(\frac{1}{1+\sum_{a\in\mathcal{A}}(\beta_a+\bar{\beta}_a)}(\widetilde{Q}+\sum_a \beta_a h_a + \hat{\beta}_a \hat{h}_a)\right) + \nu \\
\leqslant & \frac{1}{1+\sum_{a\in\mathcal{A}}(\beta_a+\bar{\beta}_a)}\widehat{\text{err}}\,(\widetilde{Q}) + \frac{\sum_{a\in\mathcal{A}}(\beta_a+\bar{\beta}_a)}{1+\sum_{a\in\mathcal{A}}(\beta_a+\bar{\beta}_a)} + \nu \\
\leqslant & \widehat{\text{err}}\,(\widetilde{Q}) + \sum_{a\in\mathcal{A}}(\beta_a+\bar{\beta}_a) + \nu \\
\leqslant & \widehat{\text{err}}\,(\widetilde{Q}) + (|\mathcal{A}|-1)\cdot(\max_{a\in\mathcal{A}}|\widehat{\text{FP}}_a(\widetilde{Q}) - \widehat{\text{FP}}_0(\widetilde{Q})| - \gamma)_+ + \nu.
\end{aligned}
$$

Define $\boldsymbol{\lambda}$ to have B in the coordinate which corresponds to $\arg\max_{a\in\mathcal{A}}|\widehat{\text{FP}}_a(\widetilde{Q}) - \widehat{\text{FP}}_0(\widetilde{Q})|$ and 0 in other coordinates. Then we have

$$
L(\widetilde{Q},\boldsymbol{\lambda}) - \nu = \widehat{\text{err}}\,(\widetilde{Q}) + B(\max_{a\in\mathcal{A}}|\widehat{\text{FP}}_a(\widetilde{Q}) - \widehat{\text{FP}}_0(\widetilde{Q})| - \gamma) - \nu
$$

To sum up, we get

$$
\widehat{\text{err}}\,(\widetilde{Q}) + B(\max_{a\in\mathcal{A}}|\widehat{\text{FP}}_a(\widetilde{Q}) - \widehat{\text{FP}}_0(\widetilde{Q})| - \gamma) - \nu \leqslant \widehat{\text{err}}\,(\widetilde{Q}) + (|\mathcal{A}|-1)\cdot(\max_{a\in\mathcal{A}}|\widehat{\text{FP}}_a(\widetilde{Q}) - \widehat{\text{FP}}_0(\widetilde{Q})| - \gamma)_+ + \nu.
$$

This implies

$$
\max_{a\in\mathcal{A}}|\widehat{\text{FP}}_a(\widetilde{Q}) - \widehat{\text{FP}}_0(\widetilde{Q})| \leqslant \gamma + \frac{2\nu}{B-(|\mathcal{A}|-1)}.
$$

which completes the proof. □

Proof of Theorem 4.7. First consider the case where $\mathcal{H} = \{h_0, h_U\}$. Choose data set D of size m as follows: $m/2$ individuals with $(A = R, X = V, Y = 0)$; $m/4$ individuals with $(A = B, X = U, Y = 1)$, $m(1-\gamma)/4$ individuals with $(A = B, X = V, Y = 0)$ and $m\gamma/4$ individuals with $(A = B, X = U, Y = 0)$. For this data set, it is easy to check that h_U has error $\gamma/4$ and h_U satisfies the fairness constraint. So $f(D) \leqslant \gamma/4$. Now consider D's neighboring data set D' by changing one individual with $(A = B, X = V, Y = 0)$ to $(A = B, X = U, Y = 0)$. For D', the classifier which satisfies the fairness constraint and has the minimum error rate is $\frac{1}{4+\gamma m}(4h_0 + \gamma m h_U)$. Therefore

$$
f(D') = \frac{1}{4+\gamma m}\left(4\cdot\frac{1}{4} + \gamma m\cdot\frac{m\gamma/4+1}{m}\right) = \frac{\gamma}{4} + \frac{1}{4+\gamma m}.
$$

implying that $|f(D) - f(D')| = \Omega(1/(\gamma m))$ and the sensitivity of f is $\Omega(1/(\gamma m))$.

Now consider the case where $\mathcal{H} = \{h_0, h_U, h_R, h_B\}$. It suffices to show that $f(D') \leqslant f(D) + O(1/m)$ for any neighboring data sets D and D'. Let Q^* be the classifier with minimum error rate on data set D. We have $f(D) = \widehat{\text{err}}\,(Q^*, D)$ and we know $|\widehat{\text{FP}}_R(Q^*, D) - \widehat{\text{FP}}_B(Q^*, D)| \leqslant \gamma$ (we put D into the arguments of $\widehat{\text{err}}$ and $\widehat{\text{FP}}$ as we are talking about two different data sets). For data set D', there are two cases.

- The case when $|\widehat{\mathrm{FP}}_R(Q^*, D') - \widehat{\mathrm{FP}}_B(Q^*, D')| \leqslant \gamma$: In this case, we have

$$f(D') \leqslant \widehat{\mathrm{err}}\,(Q^*, D') \leqslant \widehat{\mathrm{err}}\,(Q^*, D) + 1/m = f(D) + 1/m.$$

- The case when $|\widehat{\mathrm{FP}}_R(Q^*, D') - \widehat{\mathrm{FP}}_B(Q^*, D')| > \gamma$: Wlog let's assume $\widehat{\mathrm{FP}}_R(Q^*, D') - \widehat{\mathrm{FP}}_B(Q^*, D') > \gamma$. And let $\alpha = \widehat{\mathrm{FP}}_R(Q^*, D') - \widehat{\mathrm{FP}}_B(Q^*, D') - \gamma$. We know $\alpha > 0$ and we also have

  $$\alpha = \widehat{\mathrm{FP}}_R(Q^*, D') - \widehat{\mathrm{FP}}_B(Q^*, D') - \gamma \leqslant \widehat{\mathrm{FP}}_R(Q^*, D) - \widehat{\mathrm{FP}}_B(Q^*, D) - \gamma + 2/(Cm) \leqslant 2/(Cm).$$

  Now define $Q' = \frac{1}{1+\gamma+\alpha}\left((1+\gamma)Q^* + \alpha h_B\right)$. We have

  $$\widehat{\mathrm{FP}}_R(Q', D') - \widehat{\mathrm{FP}}_B(Q', D') = \frac{1}{1+\gamma+\alpha}\Big((1+\gamma)(\widehat{\mathrm{FP}}_R(Q^*, D') - \widehat{\mathrm{FP}}_B(Q^*, D')) - \alpha\Big) = \gamma.$$

  Therefore

  $$\begin{aligned} f(D') &\leqslant \widehat{\mathrm{err}}\,(Q', D') \\ &\leqslant \frac{1}{1+\gamma+\alpha}\big((1+\gamma)\,\widehat{\mathrm{err}}\,(Q^*, D') + \alpha\,\widehat{\mathrm{err}}\,(h_B, D')\big) \\ &\leqslant f(D) + 1/m + \alpha \\ &\leqslant f(D) + O(1/m) \end{aligned}$$

$\square$