Preventing Fairness Gerrymandering: Auditing and Learning for Subgroup Fairness

Michael Kearns [1] **Seth Neel** [1] **Aaron Roth** [1] **Zhiwei Steven Wu** [2]

Abstract

We introduce a new family of fairness definitions that interpolate between statistical and individual notions of fairness, obtaining some of the best properties of each. We show that checking whether these notions are satisfied is computationally hard in the worst case, but give practical oracle-efficient algorithms for learning subject to these constraints, and confirm our findings with experiments.

1. Introduction

As machine learning is being deployed in increasingly consequential domains (including policing (Rudin, 2013), criminal sentencing (Barry-Jester et al., 2015), and lending (Koren, 2016)), the problem of ensuring that learned models are *fair* has become urgent.

Approaches to fairness in machine learning can coarsely be divided into two kinds: *statistical* and *individual* notions of fairness. Statistical notions typically fix a small number of protected demographic groups $\mathcal{G}$ (such as racial groups), and then ask for (approximate) parity of some statistical measure across all of these groups. One popular statistical measure asks for equality of false positive or negative rates across all groups in $\mathcal{G}$ (this is also sometimes referred to as an *equal opportunity* constraint (Hardt et al., 2016)). Another asks for equality of classification rates (also known as *statistical parity*). These statistical notions of fairness are the kinds of fairness definitions most common in the literature (see e.g. (Kamiran & Calders, 2012; Hajian & Domingo-Ferrer, 2013; Kleinberg et al., 2017; Hardt et al., 2016; Friedler et al., 2016; Zafar et al., 2017; Chouldechova, 2017)).

One main attraction of statistical definitions of fairness is that they can in principle be obtained and checked without making any assumptions about the underlying population, and hence lead to more immediately actionable algorithmic approaches. On the other hand, individual notions of fairness ask for the algorithm to satisfy some guarantee which binds at the individual, rather than group, level. Individual notions of fairness have attractively strong semantics, but their main drawback is that achieving them seemingly requires more assumptions to be made about the setting under consideration.

The semantics of statistical notions of fairness would be significantly stronger if they were defined over a large number of *subgroups*, thus permitting a rich middle ground between fairness only for a small number of coarse pre-defined groups, and the strong assumptions needed for fairness at the individual level. Consider the kind of *fairness gerrymandering* that can occur when we only look for unfairness over a small number of pre-defined groups:

Example 1.1. *Imagine a setting with two binary features, corresponding to race (say black and white) and gender (say male and female), both of which are distributed independently and uniformly at random in a population. Consider a classifier that labels an example positive if and only if it corresponds to a black man, or a white woman. Then the classifier will appear to be equitable when one considers either protected attribute alone, in the sense that it labels both men and women as positive 50% of the time, and labels both black and white individuals as positive 50% of the time. But if one looks at any* conjunction *of the two attributes (such as black women), then it is apparent that the classifier maximally violates the statistical parity fairness constraint. Similar examples for classification are easily constructed.*

We remark that the issue raised by this toy example is not merely hypothetical. In our experiments in Section 5, we show that similar violations of fairness on subgroups of the pre-defined groups can result from the application of standard machine learning methods applied to real datasets. To avoid such problems, we would like to be able to satisfy a fairness constraint not just for the small number of protected groups defined by single protected attributes, but for a combinatorially large or even infinite collection of structured

[*]Equal contribution [1]University of Pennsylvania, Philadelphia, PA, USA [2]Mcrosoft Research, New York City, NY, USA. Correspondence to: Michael Kearns <mkearns@cis.upenn.edu>, Seth Neel <sethneel@wharton.upenn.edu>, Aaron Roth <aaroth@cis.upenn.edu>, Zhiwei Steven Wu <steven7woo@gmail.com>.

Proceedings of the 35^{th} *International Conference on Machine Learning*, Stockholm, Sweden, PMLR 80, 2018. Copyright 2018 by the author(s).

subgroups definable over protected attributes.

In this paper, we consider the problem of *auditing* binary classifiers for equal opportunity and statistical parity, and the problem of *learning* classifiers subject to these constraints, when the number of protected groups is large. There are exponentially many ways of carving up a population into subgroups, and we cannot necessarily identify a small number of these *a priori* as the only ones we need to be concerned about. At the same time, we cannot insist on any notion of statistical fairness for *every* subgroup of the population: for example, any imperfect classifier could be accused of being unfair to the subgroup of individuals defined ex-post as the set of individuals it misclassified. This simply corresponds to "overfitting" a fairness constraint.

Our investigation focuses on the computational challenges, both in theory and in practice.

1.1. Our Results

Briefly, our contributions are: 1) Formalization of the problem of auditing and learning classifiers for fairness with respect to rich classes of subgroups $\mathcal{G}$. 2) Results proving (under certain assumptions) the computational equivalence of auditing $\mathcal{G}$ and (weak) agnostic learning of $\mathcal{G}$. 3) Provably convergent algorithms for learning classifiers that are fair with respect to $\mathcal{G}$, based on a formulation as a two-player zero-sum game between a Learner (the primal player) and an Auditor (the dual player). We provide two different algorithms, both of which are based on solving for the equilibrium of this game. The first provably converges in a polynomial number of steps; the second is only guaranteed to converge asymptotically but is computationally simpler. 4) An implementation and extensive empirical evaluation of the simpler algorithm demonstrating its effectiveness on a real dataset in which subgroup fairness is a concern.

1.2. Related Work

Independent of our work, (Hébert-Johnson et al., 2017) also consider a related and complementary notion of fairness that they call "multicalibration". For a real-valued predictor, calibration informally requires that for every value $v \in [0, 1]$ predicted by an algorithm, the fraction of individuals who truly have a positive label in the subset of individuals on which the algorithm predicted v should be approximately equal to v. Multicalibration asks for approximate calibration on every set defined implicitly by some circuit in a set $\mathcal{G}$. (Hébert-Johnson et al., 2017) give an algorithmic result that is broadly similar to the one we give for learning subgroup fair classifiers, but for this different fairness notion. Our techniques differ from theirs significantly.

Thematically, the most closely related piece of prior work is (Zhang & Neill, 2016), who also aim to audit classification algorithms for discrimination in subgroups that have not been pre-defined. Our work differs from theirs in a number of important ways. First, we audit the algorithm for common measures of statistical unfairness, whereas (Zhang & Neill, 2016) design a new measure compatible with their particular algorithmic technique. Second, we give a formal analysis of our algorithm. Most importantly we give actionable algorithms for learning subgroup fair classifiers, whereas (Zhang & Neill, 2016) restrict attention to auditing.

Technically, the most closely related piece of work (and from which we take inspiration for our algorithm in Section 4) is (Agarwal et al., 2017), who show that given access to an agnostic learning oracle for a class $\mathcal{H}$, there is an efficient algorithm to find the lowest-error distribution over classifiers in $\mathcal{H}$ subject to equalizing false positive rates across polynomially many subgroups. Their algorithm can be viewed as solving the same zero-sum game that we solve, but in which the "subgroup" player plays gradient descent over his pure strategies, one for each sub-group. This ceases to be an efficient or practical algorithm when the number of subgroups is large, as is our case.

There is also other work showing computational hardness for fair learning problems. Most notably, (Woodworth et al., 2017) show that finding a linear threshold classifier that approximately minimizes hinge loss subject to equalizing false positive rates across populations is computationally hard (assuming that refuting a random k-XOR formula is hard). In contrast, we show that even *checking* whether a classifier satisfies a false positive rate constraint on a particular data set is computationally hard (if the number of subgroups on which fairness is desired is too large to enumerate).

2. Model and Preliminaries

We model each individual as being described by a tuple $((x, x'), y)$, where $x \in \mathcal{X}$ denotes a vector of *protected attributes*, $x' \in \mathcal{X}'$ denotes a vector of *unprotected attributes*, and $y \in \{0, 1\}$ denotes a label. Note that in our formulation, an auditing algorithm not only may not see the unprotected attributes x', it may not even be aware of their existence. For example, x' may represent proprietary features or consumer data purchased by a credit scoring company.

We will write $X = (x, x')$ to denote the joint feature vector. We assume that points (X, y) are drawn i.i.d. from an unknown distribution $\mathcal{P}$. Let D be a decision making algorithm, and let $D(X)$ denote the (possibly randomized) decision induced by D on individual (X, y). We restrict attention in this paper to the case in which D makes a binary classification decision: $D(X) \in \{0, 1\}$. Thus we alternately refer to D as a classifier. When *auditing* a fixed classifier D, it will be helpful to make reference to the distribution over examples (X, y) together with their induced classification $D(X)$. Let $\mathcal{P}_{\text{audit}}(D)$ denote the induced *target joint distribution* over the tuple $(x, y, D(X))$ that results from sampling $(x, x', y) \sim \mathcal{P}$, and providing x, the true label y, and the classification $D(X) = D(x, x')$ but not the unprotected attributes x'. Note that the randomness here is over both the randomness of $\mathcal{P}$, and the potential randomness of the classifier D.

We will be concerned with learning and auditing classifiers D satisfying two common statistical fairness constraints: equality of classification rates (also known as statistical parity), and equality of false positive rates (also known as equal opportunity). Auditing for equality of false negative rates is symmetric and so we do not explicitly consider it. Each fairness constraint is defined with respect to a set of protected groups. We define sets of protected groups via a family of indicator functions $\mathcal{G}$ for those groups, defined over protected attributes. Each $g : \mathcal{X} \to \{0, 1\} \in \mathcal{G}$ has the semantics that $g(x) = 1$ indicates that an individual with protected features x is in group g.

Definition 2.1 (Statistical Parity (SP) Subgroup Fairness)**.** *Fix any classifier D, distribution $\mathcal{P}$, collection of group indicators $\mathcal{G}$, and parameter $\gamma \in [0, 1]$. For each $g \in \mathcal{G}$, define*

$$\alpha_{SP}(g, \mathcal{P}) = \Pr_{\mathcal{P}}[g(x) = 1]$$

$$\beta_{SP}(g, D, \mathcal{P}) = |\mathrm{SP}(D) - \mathrm{SP}(D, g)|,$$

where $\mathrm{SP}(D) = \Pr_{\mathcal{P},D}[D(X) = 1]$ *and* $\mathrm{SP}(D, g) = \Pr_{\mathcal{P},D}[D(X) = 1 | g(x) = 1]$ *denote the overall acceptance rate of D and the acceptance rate of D on group g respectively. We say that D satisfies γ-*statistical parity (SP) Fairness *with respect to $\mathcal{P}$ and $\mathcal{G}$ if for every $g \in \mathcal{G}$*

$$\alpha_{SP}(g, \mathcal{P})\ \beta_{SP}(g, D, \mathcal{P}) \leq \gamma.$$

We will sometimes refer to $\mathrm{SP}(D)$ *as the* SP base rate.

Remark 2.2. *Note that our definition references two approximation parameters, both of which are important. We are allowed to ignore a group g if it (or its complement) represent only a small fraction of the total probability mass. The parameter α governs how small a fraction of the population we are allowed to ignore. Similarly, we do not require that the probability of a positive classification in every subgroup is exactly equal to the base rate, but instead allow deviations up to β. Both of these approximation parameters are necessary from a statistical estimation perspective. We control both of them with a single parameter γ.*

Definition 2.3 (False Positive (FP) Subgroup Fairness)**.** *Fix any classifier D, distribution $\mathcal{P}$, collection of group indicators $\mathcal{G}$, and parameter $\gamma \in [0, 1]$. For each $g \in \mathcal{G}$, define*

$$\alpha_{FP}(g, \mathcal{P}) = \Pr_{\mathcal{P}}[g(x) = 1, y = 0]$$

$$\beta_{FP}(g, D, \mathcal{P}) = |\mathrm{FP}(D) - \mathrm{FP}(D, g)|$$

where $\mathrm{FP}(D) = \Pr_{D,\mathcal{P}}[D(X) = 1 \mid y = 0]$ *and* $\mathrm{FP}(D, g) = \Pr_{D,\mathcal{P}}[D(X) = 1 \mid g(x) = 1, y = 0]$ *denote the overall false-positive rate of D and the false-positive rate of D on group g respectively.*

*We say D satisfies γ-*False Positive (FP) Fairness *with respect to $\mathcal{P}$ and $\mathcal{G}$ if for every $g \in \mathcal{G}$*

$$\alpha_{FP}(g, \mathcal{P})\ \beta_{FP}(g, D, \mathcal{P}) \leq \gamma.$$

We will sometimes refer to $\mathrm{FP}(D)$ *FP-base rate.*

For either SP and FP fairness, if the algorithm D fails to satisfy the γ-fairness condition, then we say that D is γ-*unfair* with respect to $\mathcal{P}$ and $\mathcal{G}$. We call any subgroup g which witnesses this unfairness an γ-*unfair certificate* for $(D, \mathcal{P})$.

An *auditing algorithm* for a notion of fairness is given sample access to $\mathcal{P}_{\text{audit}}(D)$ for some classifier D. It will either deem D to be fair with respect to $\mathcal{P}$, or will else produce a certificate of unfairness.

Definition 2.4 (Auditing Algorithm)**.** *Fix a notion of fairness (either SP or FP fairness), a collection of group indicators $\mathcal{G}$ over the protected features, and any $\delta, \gamma, \gamma' \in (0, 1)$ such that $\gamma' \leq \gamma$. A (γ, γ')-auditing algorithm for $\mathcal{G}$ with respect to distribution $\mathcal{P}$ is an algorithm $\mathcal{A}$ such that for any classifier D, when given access the distribution $\mathcal{P}_{\text{audit}}(D)$, $\mathcal{A}$ runs in time $\mathrm{poly}(1/\gamma', \log(1/\delta))$, and with probability $1 - \delta$, outputs a γ'-unfair certificate for D whenever D is γ-unfair with respect to $\mathcal{P}$ and $\mathcal{G}$. If D is γ'-fair, $\mathcal{A}$ will output "fair".*

As we will show, the notion of auditing is closely related to weak agnostic learning.

Definition 2.5 (Weak Agnostic Learning (Kearns et al., 1994; Kalai et al., 2008))**.** *Let Q be a distribution over $\mathcal{X} \times \{0, 1\}$ and let $\varepsilon, \varepsilon' \in (0, 1/2)$ such that $\varepsilon \geq \varepsilon'$. We say that the function class $\mathcal{G}$ is $(\varepsilon, \varepsilon')$-*weakly agnostically learnable *under distribution Q if there exists an algorithm L such that when given sample access to Q, L runs in time $\mathrm{poly}(1/\varepsilon', 1/\delta)$, and with probability $1 - \delta$, outputs a hypothesis $h \in \mathcal{G}$ such that*

$$\min_{f \in \mathcal{G}} err(f, Q) \leq 1/2 - \varepsilon \implies err(h, Q) \leq 1/2 - \varepsilon'.$$

where $err(h, Q) = \Pr_{(x,y) \sim Q}[h(x) \neq y]$.

Cost-Sensitive Classification. In this paper, we will also give reductions to *cost-sensitive classification (CSC)* problems. Formally, an instance of a CSC problem for the class $\mathcal{H}$ is given by a set of n tuples $\{(X_i, c_i^0, c_i^1)\}_{i=1}^n$ such that c_i^ℓ corresponds to the cost for predicting label ℓ on point X_i. Given such an instance as input, a CSC oracle finds a hypothesis $\hat{h} \in \mathcal{H}$ that minimizes the total cost:

$$\hat{h} \in \operatorname*{argmin}_{h \in \mathcal{H}} \sum_{i=1}^{n} [h(X_i)c_i^1 + (1 - h(X_i))c_i^0] \qquad (1)$$

Remark 2.6. *Cost-sensitive classification is polynomially equivalent to agnostic learning (Zadrozny et al., 2003). Sometimes one definition will be more convenient to work with than the other.*

3. Auditing ⇔ Weak Agnostic Learning

In this section, we give a reduction from the problem of auditing both statistical parity and false positive rate fairness, to the problem of agnostic learning, and vice versa. This has two implications. The main implication is that, from a worst-case analysis point of view, auditing is computationally hard in almost every case (since it inherits this pessimistic state of affairs from agnostic learning). However, worst-case hardness results in learning theory have not prevented the successful practice of machine learning, and there are many heuristic algorithms that in real-world cases successfully solve "hard" agnostic learning problems. Our reductions also imply that these heuristics can be used successfully as auditing algorithms, and we exploit this in the development of our algorithmic results and their experimental evaluation.

We will think about these as the target distributions for a learning problem: i.e. the problem of learning to predict $D(X)$ from only the protected features x. We will relate the ability to agnostically learn on these distributions, to the ability to audit D given access to the original distribution $\mathcal{P}_{\text{audit}}(D)$.

Statistical Parity Fairness We give our reduction first for SP subgroup fairness. The reduction for FP subgroup fairness will follow as a corollary, since auditing for FP subgroup fairness can be viewed as auditing for statistical parity fairness on the subset of the data restricted to $y = 0$.

Theorem 3.1. *Fix any distribution $\mathcal{P}$, and any set of group indicators $\mathcal{G}$. Then for any $\gamma, \varepsilon > 0$, the following relationships hold:*

- *If there is a $(\gamma/2, (\gamma/2 - \varepsilon))$ auditing algorithm for $\mathcal{G}$ for all D such that $\mathrm{SP}(D) = 1/2$, then the class $\mathcal{G}$ is $(\gamma, \gamma/2 - \varepsilon)$-weakly agnostically learnable under $\mathcal{P}^D$.*
- *If $\mathcal{G}$ is $(\gamma, \gamma - \varepsilon)$-weakly agnostically learnable under marginal distribution $\mathcal{P}^D$ on $(x, D(X))$ for all D such that $\mathrm{SP}(D) = 1/2$, then there is a $(\gamma, (\gamma - \varepsilon)/2)$ auditing algorithm for $\mathcal{G}$ for SP fairness under $\mathcal{P}$.*

False Positive Fairness A corollary of the above reduction is an analogous equivalence between auditing for FP subgroup fairness and agnostic learning. This is because a FP fairness constraint can be viewed as a statistical parity fairness constraint on the subset of the data such that $y = 0$. Therefore, Theorem 3.1 implies the following:

Corollary 3.2. *Fix any distribution $\mathcal{P}$, and any set of group indicators $\mathcal{G}$. The following two relationships hold:*

- *If there is a $(\gamma/2, (\gamma/2 - \varepsilon))$ auditing algorithm for $\mathcal{G}$ for all D with $\mathrm{FP}(D) = 1/2$, then $\mathcal{G}$ is $(\gamma, \gamma/2 - \varepsilon)$-weakly agnostically learnable under $\mathcal{P}^D_{y=0}$.*
- *If $\mathcal{G}$ is $(\gamma, \gamma - \varepsilon)$–weakly agnostically learnable under the conditional distribution $\mathcal{P}^D_{y=0}$ of (X, y) conditioned on the event that $D(X) = 1$ for all D with $\mathrm{FP}(D) = 1/2$, then there is a $(\gamma, (\gamma - \varepsilon)/2)$ auditing algorithm for FP subgroup fairness for $\mathcal{G}$ under distribution $\mathcal{P}$.*

Worst-Case Intractability of Auditing While we shall see in subsequent sections that the equivalence given above has positive algorithmic and experimental consequences, from a purely theoretical perspective the reduction of agnostic learning to auditing has strong negative worst-case implications. More precisely, we can import a long sequence of intractability results for agnostic learning to obtain:

Theorem 3.3. *Under standard complexity-theoretic intractability assumptions, for $\mathcal{G}$ the classes of conjunctions of boolean attributes, linear threshold functions, or bounded-degree polynomial threshold functions, there exist distributions P such that the auditing problem cannot be solved in polynomial time, for either SP or FP fairness.*

While Theorem 3.3 shows that certain natural subgroup classes $\mathcal{G}$ yield intractable auditing problems in the worst case, in the rest of the paper we demonstrate that effective heuristics for this problem on specific (non-worst case) distributions can be used to derive an effective and practical learning algorithm for subgroup fairness.

4. ERM Subject to Fairness Constraints $\mathcal{G}$

In this section, we present an algorithm for training a classifier that satisfies false-positive subgroup fairness simultaneously for all protected subgroups specified by a family of group indicator functions $\mathcal{G}$. All of our techniques also apply to a statistical parity or false negative rate constraint.

Let S denote a set of n labeled examples $\{z_i = (x_i, x_i'), y_i)\}_{i=1}^n$, and let $\mathcal{P}$ denote the empirical distribution over this set of examples. Let $\mathcal{H}$ be a hypothesis class defined over both the protected and unprotected attributes, and let $\mathcal{G}$ be a collection of group indicators over the protected attributes. We assume that $\mathcal{H}$ contains a constant classifier (which implies that there is at least one fair classifier to be found, for any distribution).

Our goal will be to find the distribution over classifiers from $\mathcal{H}$ that minimizes classification error subject to the fairness constraint over $\mathcal{G}$. We will design an iterative algorithm that, when given access to a CSC oracle, computes an optimal randomized classifier in polynomial time.

Let D denote a probability distribution over $\mathcal{H}$. Consider the following *Fair ERM (Empirical Risk Minimization)* problem:

$$\min_{D \in \Delta_{\mathcal{H}}} \mathop{\mathbb{E}}_{h \sim D} [err(h, \mathcal{P})] \tag{2}$$

$$\text{s.t. } \forall g \in \mathcal{G} \qquad \alpha_{FP}(g, \mathcal{P})\, \beta_{FP}(g, D, \mathcal{P}) \le \gamma. \tag{3}$$

where $err(h, \mathcal{P}) = \Pr_{\mathcal{P}}[h(x, x') \neq y]$, and the quantities α_{FP} and β_{FP} are defined in Definition 2.3. We will write OPT to denote the objective value at the optimum for the Fair ERM problem, that is the minimum error achieved by a γ-fair distribution over the class $\mathcal{H}$.

Our main theoretical result is a computationally efficient oracle-based algorithm for solving the Fair ERM problem.

Theorem 4.1. *Fix any $\nu, \delta \in (0, 1)$. Then given an input of n data points and accuracy parameters ν, δ and access to oracles $\mathrm{CSC}(\mathcal{H})$ and $\mathrm{CSC}(\mathcal{G})$, there exists an algorithm runs in time $\mathrm{poly}(1/\nu, \log(1/\delta))$, and with probability at least $1 - \delta$, output a randomized classifier $\hat{D}$ such that $err(\hat{D}, \mathcal{P}) \le \mathrm{OPT} + \nu$, and for any $g \in \mathcal{G}$, the fairness constraint violations satisfies*

$$\alpha_{FP}(g, \mathcal{P})\, \beta_{FP}(g, \hat{D}, \mathcal{P}) \le \gamma + O(\nu).$$

Overview of our solution.

First, we rewrite the Fair ERM problem as a linear program with finitely many decision variables and constraints even when $\mathcal{H}$ and $\mathcal{G}$ are infinite. To do this, we take advantage of the fact that Sauer's Lemma lets us bound the number of labellings that any hypothesis class $\mathcal{H}$ of bounded VC dimension can induce on any fixed dataset. The LP has one variable for each of these possible labellings, rather than one variable for each hypothesis. Moreover, again by Sauer's Lemma, we have one constraint for each of the finitely many possible subgroups induced by $\mathcal{G}$ on the fixed dataset, rather than one for each of the (possibly infinitely many) subgroups definable over arbitrary datasets. This step is important — it will guarantee that strong duality holds.

We then derive the partial Lagrangian of the LP, and note that computing an approximately optimal solution to this LP is equivalent to finding an approximate minmax solution for a corresponding zero-sum game, in which the payoff function U is the value of the Lagrangian. The pure strategies of the primal or "Learner" player correspond to classifiers $h \in \mathcal{H}$, and the pure strategies of the dual or "Auditor" player correspond to subgroups $g \in \mathcal{G}$. Intuitively, the Learner is trying to minimize the sum of the prediction error and a fairness penalty term (given by the Lagrangian), and the Auditor is trying to penalize the fairness violation of the Learner by first identifying the subgroup with the greatest fairness violation and putting all the weight on the dual variable corresponding to this subgroup. In order to reason about convergence, we restrict the set of dual variables to lie in a bounded set: C times the probability simplex. C is a parameter that we have to set in the proof of our theorem to give the best theoretical guarantees — but it is also a parameter that we will vary in the experimental section.

We observe that given a mixed strategy for the Auditor, the best response problem of the Learner corresponds to a CSC problem. Similarly, given a mixed strategy for the Learner, the best response problem of the Auditor corresponds to an auditing problem (which can be represented as a CSC problem). Hence, if we have oracles for solving CSC problems, we can compute best responses for both players, in response to arbitrary mixed strategies of their opponents.

Finally, we show that the ability to compute best responses for each player is sufficient to implement dynamics known to converge quickly to equilibrium in zero-sum games. Our algorithm has the Learner play *Follow the Perturbed Leader (FTPL)* (Kalai & Vempala, 2005), which is a no-regret algorithm, against an Auditor who at every round best responds to the learner's mixed strategy. By the seminal result of Freund & Schapire (1996), the average plays of both players converge to an approximate equilibrium. In order to implement this in polynomial time, we need to represent the loss of the learner as a low-dimensional linear optimization problem. To do so, we first define an appropriately translated CSC problem for any mixed strategy λ by the Auditor, and cast it as a linear optimization problem.

5. Experimental Evaluation

We now describe an experimental evaluation of our proposed algorithmic framework on a dataset in which fairness is a concern, due to the preponderance of racial and other sensitive features. We also demonstrate that for this dataset, our methods are empirically necessary to avoid fairness gerrymandering.

While the no-regret-based algorithm described in the last section enjoys provably polynomial time convergence, for the experiments we instead implemented a simpler yet effective algorithm based on *Fictitious Play* dynamics. We first describe and discuss this modified algorithm.

5.1. Solving the Game with Fictitious Play

Like the algorithm given in the last section, the algorithm we implemented works by simulating a game dynamic that converges to Nash equilibrium in the zero-sum game that we derived, corresponding to the Fair ERM problem. Rather than using a no-regret dynamic, we instead use a simple iterative procedure known as *Fictitious Play* (Brown, 1949). Fictitious Play dynamics has the benefit of being more practical to implement: at each round, both players simply need to compute a single best response to the empirical play of their opponents, and this optimization requires only a single call to a CSC oracle. In contrast, the FTPL dynamic we gave in the previous section requires making many calls to a CSC oracle per round — a computationally expensive process — in order to find a sparse approximation to the Learner's mixed strategy at that round. Fictitious Play also has the benefit of being deterministic, unlike the randomized sampling required in the FTPL no-regret dynamic, thus eliminating a source of experimental variance.

The disadvantage is that Fictitious Play is only known to converge to equilibrium in the limit (Robinson, 1951), rather than in a polynomial number of rounds (though it is conjectured to converge quickly under rather general circumstances; see (Daskalakis & Pan, 2014) for a recent discussion). As we will show, it performs well on real data, despite the fact that it has weaker theoretical guarantees.

Fictitious play proceeds in rounds, and in every round each player chooses a best response to his opponent's empirical history of play across previous rounds, by treating it as the mixed strategy that randomizes uniformly over the empirical history. Pseudocode for the implemented algorithm is given in the full version.

5.2. Description of Data

The dataset we use is known as the "Communities and Crime" (C&C) dataset, available at the UC Irvine Data Repository.[1] Each record in this dataset describes the aggregate demographic properties of a different U.S. community; the data combines socio-economic data from the 1990 US Census, law enforcement data from the 1990 US LEMAS survey, and crime data from the 1995 FBI UCR. The total number of records is 1994, and the number of features is 122. The variable to be predicted is the rate of violent crime in the community.

While there are larger and more recent datasets in which subgroup fairness is a potential concern, there are properties of the C&C dataset that make it particularly appealing for our experimental evaluation. Foremost among these is the relatively high number of sensitive or protected attributes, and the fact that they are real-valued (since they represent aggregates in a community rather than specific individuals). This means there is a very large number of protected subgroups that can be defined over them. We obtain a set 18 real-valued protected attributes, most of which are related to race (e.g. the percentage and the average per capita incomes of multiple racial groups in the communities). We note that the C&C dataset has numerous other features that arguably could or should be protected as well (such as gender features), which would raise the dimensionality of the protected subgroups even further. [2]

We convert the real-valued rate of violent crime in each community to a binary label indicating whether the community is in the 70th percentile of that value, indicating that it is a relatively high-crime community. Thus the strawman baseline that always predicts 0 (lower crime) has error approximately 30% or 0.3 on this classification problem. We chose the 70th percentile since it seems most natural to predict the highest crime rates.

As in the theoretical sections of the paper, our main interest and emphasis is on the effectiveness of our proposed algorithm **FairFictPlay** on a given dataset, including:[3]

- Whether the algorithm in fact converges, and does so in a feasible amount of computation. Recall that formal convergence is only guaranteed under the assumption of oracles that do not exist in practice, and even then is only guaranteed asymptotically.
- Whether the classifier learned by the algorithm has nontrivial accuracy, as well as strong subgroup fairness properties.
- Whether the algorithm and dataset permits nontrivial tuning of the trade-off between accuracy and subgroup fairness.

5.3. Algorithm Implementation

The main details in the implementation of **FairFictPlay** are the identification of the model classes for Learner and Auditor, the implementation of the cost sensitive classification oracle and auditing oracle, and the identification of the protected features for Auditor. For our experiments, at each round Learner chooses a linear threshold function over all 122 features. We implement the cost sensitive classification oracle via a two stage regression procedure. In particular, the inputs to the cost sensitive classification oracle are cost vectors c_0, c_1, where the i^{th} element of c_k is the cost of predicting k on datapoint i. We train two linear regression models r_0, r_1 to predict c_0 and c_1 respectively, using all 122 features. Given a new point x, we predict the cost of

[1] http://archive.ics.uci.edu/ml/datasets/Communities+and+Crime

[2] Ongoing experiments on other datasets where fairness is a concern will be reported on in a forthcoming experimental paper.

[3] In the full version, we provide a generalization error bound on the fairness violations as a function of the VC dimensions of the Learner's hypothesis class $\mathcal{H}$ and the Auditor's subgroup class $\mathcal{G}$. Thus for simplicity, we report all results here on the full C&C dataset of 1994 points, treating it as the true distribution of interest.

classifying x as 0 and 1 using our regression models: these predictions are $r_0(x)$ and $r_1(x)$ respectively. Finally we output the prediction $\hat{y}$ corresponding to lower predicted cost: $\hat{y} = \text{argmin}_{i \in \{0,1\}} r_i(x)$.

Auditor's model class consists of all linear threshold functions over just the 18 aforementioned protected race-based attributes. As per the algorithm, at each iteration t Auditor attempts to find a subgroup on which the false positive rate is substantially different than the base rate, given the Learner's randomized classifier so far. We implement the auditing oracle by treating it as a weighted regression problem in which the goal is find a linear function (which will be taken to define the subgroup) that on the negative examples, can predict the Learner's probabilistic classification on each point. We use the same regression subroutine as Learner does, except that Auditor only has access to the 18 sensitive features, rather than all 122.

5.4. Results

Particularly in light of the gaps between the idealized theory and the actual implementation, the most basic questions about **FairFictPlay** are whether it converges at all, and if so, whether it converges to "interesting" with both nontrivial classification error (much better than the 30% or 0.3 baserate), and nontrivial subgroup fairness (much better than ignoring fairness altogether). We shall see that at least for the C&C dataset, the answers to these questions is strongly affirmative.

We begin by examining the evolution of the error and unfairness of Learner's model. In the left panel of Figure 1 we show the error of the model found by Learner vs. iteration for values of γ ranging from 0 to 0.029. Several comments are in order.

First, after an intial period in which there is a fair amount of oscillatory behavior, by 6,000 iterations most of the curves have largely flattened out, and by 8,000 iterations it appears most but not all have reached approximate convergence. Second, while the top-to-bottom ordering of these error curves is approximately aligned with decreasing γ — so larger γ generally results in lower error, as expected — there are many violations of this for small t, and even a few at large t. Third, and as we will examine more closely shortly, the converged values at large t do indeed exhibit a range of errors.

In the right panel of Figure 1, we show the corresponding unfairness γ_t of the subgroup found by the Auditor at each iteration t for the same runs and values of the parameter γ (indicated by horizontal dashed lines), with the same color-coding as for the left panel. Now the ordering is generally reversed — larger values of γ generally lead to higher γ_t curves, since the fairness constraint on the Learner is weaker. We again see a great deal of early oscillatory behavior, with most γ_t curves then eventually settling at or near their corresponding input γ value, as Learner and Auditor engage in a back-and-forth struggle for lower error for Learner and γ-subgroup fairness for Auditor.

For any choice of the parameter γ, and each iteration t, the two panels of Figure 1 yield a pair of realized values $\langle \varepsilon_t, \gamma_t \rangle$ from the experiment, corresponding to a Learner model whose error is ε_t, and for which the worst subgroup the Auditor was able to find had unfairness γ_t. The set of all $\langle \varepsilon_t, \gamma_t \rangle$ pairs across all runs or γ values thus represents the different trade-offs between error and unfairness found by our algorithm on the data. Most of these pairs are of course Pareto-dominated by other pairs, so we are primarily interested in the undominated frontier.

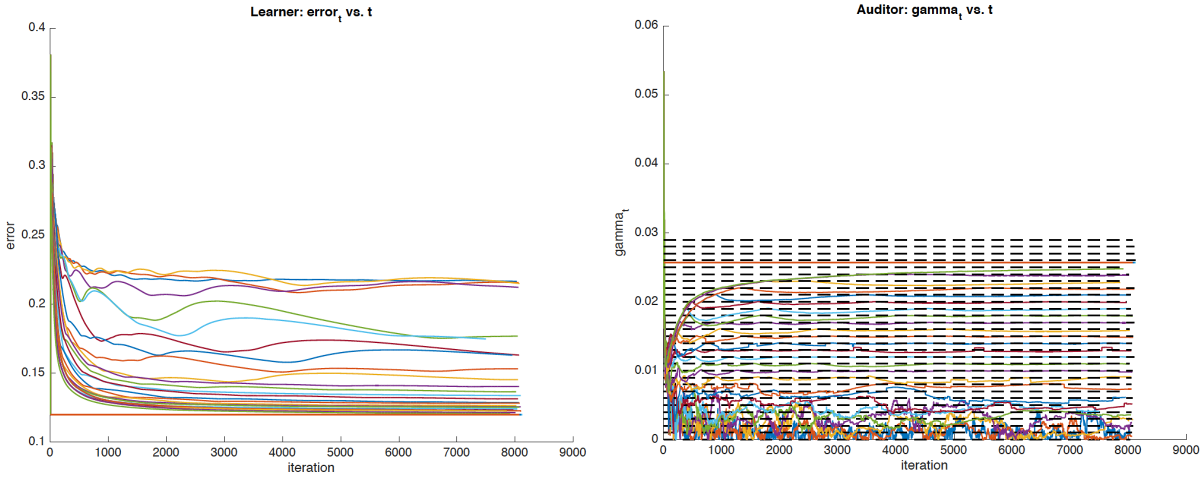
In the left panel of Figure 2, for each value of γ we show the Pareto-optimal pairs, color-coded for the value of γ. Each value of γ yields a set or cloud of undominated pairs that are usually fairly close to each other, and as expected, as γ is increased, these clouds generally move leftwards and upwards (lower error and higher unfairness).

We anticipate that the practical use of our algorithm would, as we have done, explore many values of γ and then pick a model corresponding to a point on the aggregated Pareto frontier across all γ, which represents the collection of all undominated models and the overall error-unfairness trade-off. This aggregate frontier is shown in the right panel of Figure 2, and shows a relatively smooth menu of options, ranging from error about 0.21 and no unfairness at one extreme, to error about 0.12 and unfairness 0.025 at the other, and an appealing assortment of intermediate trade-offs. Of course, in a real application the selection of a particular point on the frontier should be made in a domain-specific manner by the stakeholders or policymakers in question.

5.5. Protecting Marginal Subgroups is Not Sufficient

It is intuitive that one can construct (as we did in the introduction) artificial examples in which classifiers which equalize false positive rates across groups defined only with respect to individual protected binary features can exhibit unfairness in more complicated subgroups. However, it might be the case that on real-world datasets, enforcing false positive rate fairness only in marginal subgroups, using previously known algorithms (like (Agarwal et al., 2017)), would already provide at least approximate fairness in the combinatorially many subgroups defined by a simple (e.g. linear threshold) function over the protected features.

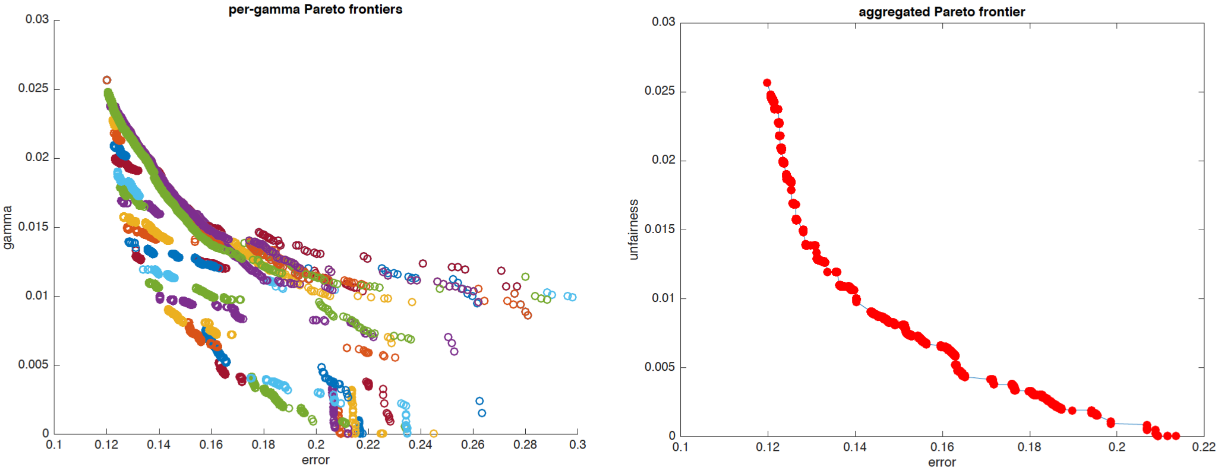
To explore this possibility, we implemented the algorithm of (Agarwal et al., 2017), which employs a similar optimization framework. We used the same Communities and Crime dataset with the same 18 protected features. Our 18 protected attributes are real valued. In order to come up with



(a)

(b)

Figure 1. Evolution of the error and unfairness of Learner's classifier across iterations, for varying choices of γ. (a) Error ε_t of Learner's model vs iteration t. (b) Unfairness γ_t of subgroup found by Auditor vs. iteration t, as measured by Definition 2.3. See text for details.



(a)

(b)

Figure 2. (a) Pareto-optimal error-unfairness values, color coded by varying values of the input parameter γ. (b) Aggregate Pareto frontier across all values of γ. Here the γ values cover the same range but are sampled more densely to get a smoother frontier. See text for details.

a small number of protected groups, we threshold each real-valued attribute at its mean, and define 36 protected groups: each one corresponding to one of the protected attributes lying either above or below its mean.

We then ran the algorithm from (Agarwal et al., 2017).The algorithm converges to a lowest achieved marginal false positive rate of 0.0249, meaning that each marginal-only subgroup was rather well-protected, as intended by the (heuristic) optimization.

However, upon auditing the resulting classifier with respect to the richer class of linear threshold functions on the continuously-valued protected features, we find that there is a subgroup whose γ value is 0.007 a significant multiple of the value of 0.002 achieved by our algorithm at the same error. This demonstrates empirically that merely minimizing marginal unfairness will not generally result in more refined subgroup fairness "for free, and in fact may fail badly in this regard. While perhaps not surprising from a theoretical perspective, since the two methods are attempting to optimize different objectives, it is reassuring to see on a data set with a large number of protected features.

References

Agarwal, A., Beygelzimer, A., Dudík, M., and Langford, J. A reductions approach to fair classification. *Fairness, Accountability, and Transparency in Machine Learning (FATML)*, 2017.

Barry-Jester, A. M., Casselman, B., and Goldstein, D. The new science of sentencing. *The Marshall Project*, August 8 2015. Retrieved 4/28/2016.

Brown, G. W. Some notes on computation of games solutions, Jan 1949.

Chouldechova, A. Fair prediction with disparate impact: A study of bias in recidivism prediction instruments. *arXiv preprint arXiv:1703.00056*, 2017.

Daskalakis, C. and Pan, Q. A counter-example to Karlin's strong conjecture for fictitious play. In *Foundations of Computer Science (FOCS), 2014 IEEE 55th Annual Symposium on*, pp. 11–20. IEEE, 2014.

Freund, Y. and Schapire, R. E. Game theory, on-line prediction and boosting. In *Proceedings of the Ninth Annual Conference on Computational Learning Theory, COLT 1996, Desenzano del Garda, Italy, June 28-July 1, 1996.*, pp. 325–332, 1996. doi: 10.1145/238061.238163.

Friedler, S. A., Scheidegger, C., and Venkatasubramanian, S. On the (im) possibility of fairness. *arXiv preprint arXiv:1609.07236*, 2016.

Hajian, S. and Domingo-Ferrer, J. A methodology for direct and indirect discrimination prevention in data mining. *IEEE transactions on knowledge and data engineering*, 25(7):1445–1459, 2013.

Hardt, M., Price, E., and Srebro, N. Equality of opportunity in supervised learning. *Advances in Neural Information Processing Systems*, 2016.

Hébert-Johnson, Ú., Kim, M. P., Reingold, O., and Rothblum, G. N. Calibration for the (computationally-identifiable) masses. *arXiv preprint arXiv:1711.08513*, 2017.

Kalai, A. T. and Vempala, S. Efficient algorithms for online decision problems. *J. Comput. Syst. Sci.*, 71(3):291–307, 2005. doi: 10.1016/j.jcss.2004.10.016.

Kalai, A. T., Mansour, Y., and Verbin, E. On agnostic boosting and parity learning. In *Proceedings of the 40th Annual ACM Symposium on Theory of Computing, Victoria, British Columbia, Canada, May 17-20, 2008*, pp. 629–638, 2008. doi: 10.1145/1374376.1374466.

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

Kearns, M. J., Schapire, R. E., and Sellie, L. M. Toward efficient agnostic learning. *Machine Learning*, 17(2-3): 115–141, 1994.

Kleinberg, J., Mullainathan, S., and Raghavan, M. Inherent trade-offs in the fair determination of risk scores. In *Proceedings of the 2017 ACM Conference on Innovations in Theoretical Computer Science, Berkeley, CA, USA, 2017*, 2017.

Koren, J. R. What does that web search say about your credit? *Los Angeles Times*, July 16 2016. Retrieved 9/15/2016.

Robinson, J. An iterative method of solving a game. *Annals of Mathematics*, pp. 10–2307, 1951.

Rudin, C. Predictive policing using machine learning to detect patterns of crime. *Wired Magazine*, August 2013. Retrieved 4/28/2016.

Woodworth, B., Gunasekar, S., Ohannessian, M. I., and Srebro, N. Learning non-discriminatory predictors. *arXiv preprint arXiv:1702.06081*, 2017.

Zadrozny, B., Langford, J., and Abe, N. Cost-sensitive learning by cost-proportionate example weighting. In *Proceedings of the 3rd IEEE International Conference on Data Mining (ICDM 2003), 19-22 December 2003, Melbourne, Florida, USA*, pp. 435, 2003. doi: 10.1109/ICDM.2003.1250950.

Zafar, M. B., Valera, I., Gomez-Rodriguez, M., and Gummadi, K. P. Fairness beyond disparate treatment & disparate impact: Learning classification without disparate mistreatment. In *Proceedings of the 26th International Conference on World Wide Web, WWW 2017, Perth, Australia, April 3-7, 2017*, pp. 1171–1180, 2017. doi: 10.1145/3038912.3052660.

Zhang, Z. and Neill, D. B. Identifying significant predictive bias in classifiers. *arXiv preprint arXiv:1611.08292*, 2016.