Probably Approximately Metric-Fair Learning

Guy N. Rothblum [*1] **Gal Yona** [*1]

Abstract

The seminal work of Dwork *et al.* [ITCS 2012] introduced a metric-based notion of individual fairness: given a task-specific similarity metric, their notion required that every pair of similar individuals should be treated similarly. In the context of machine learning, however, individual fairness does not generalize from a training set to the underlying population. We show that this can lead to computational intractability even for simple fair-learning tasks. With this motivation in mind, we introduce and study a relaxed notion of *approximate metric-fairness*: for a random pair of individuals sampled from the population, with all but a small probability of error, if they are similar then they should be treated similarly. We formalize the goal of achieving approximate metric-fairness simultaneously with best-possible accuracy as Probably Approximately Correct and Fair (PACF) Learning. We show that approximate metric-fairness *does* generalize, and leverage these generalization guarantees to construct polynomial-time PACF learning algorithms for the classes of linear and logistic predictors.

1. Introduction

Machine learning is increasingly used to make consequential classification decisions about individuals. Examples range from predicting whether a user will enjoy a particular article, to estimating a felon's recidivism risk, to determining whether a patient is a good candidate for a medical treatment. Automated classification comes with great benefits, but it also raises substantial societal concerns (cf. (O'Neil, 2016) for a recent perspective). One prominent concern is that these algorithms might discriminate against individuals or groups in a way that violates laws or social and ethical norms. This might happen due to biases in the training data or due to biases introduced by the algorithm. To address these concerns, and to truly unleash the full potential of automated classification, there is a growing need for frameworks and tools to mitigate the risks of algorithmic discrimination. A growing literature attempts to tackle these challenges by exploring different fairness criteria.

Discrimination can take many guises. It can be difficult to spot and difficult to define. Imagine a protected minority population P (defined by race, gender identity, etc). A natural approach for protecting the members of P from discrimination is to make sure that they are not mistreated *on average*. For example, that on average members of P and individuals outside of P are classified in any particular way with roughly the same probability. This is a *"group-level"* fairness notion, sometimes referred to as *statistical parity*.

Pointing out several weakness of group-level notions of fairness, the seminal work of (Dwork et al., 2012) introduced a notion of *individual fairness*. Their notion relies on a *task-specific similarity metric* that specifies, for every two individuals, how similar they are with respect to the specific classification task at hand. Given such a metric, similar individuals should be treated similarly, i.e. assigned similar classification distributions (their focus was on probabilistic classifiers, as will be ours). In this work, we refer to their fairness notion as *perfect metric-fairness*.

Given a good metric, perfect metric-fairness provides powerful protections from discrimination. Furthermore, the metric provides a vehicle for specifying social norms, cultural awareness, and task-specific knowledge. While coming up with a good metric can be challenging, metrics arise naturally in prominent existing examples (such as credit scores and insurance risk scores), and in natural scenarios (a metric specified by an external regulator). Dwork *et al.* studied the goal of finding a (probabilistic) classifier that minimizes utility loss (or maximizes accuracy), subject to satisfying the perfect metric-fairness constraint. They showed how to phrase and solve this optimization problem for a given collection of individuals.

1.1. This Work

Building on these foundations, we study *metric-fair machine learning*. Consider a learner that is given a similarity metric and a training set of labeled examples, drawn from

[*]Equal contribution [1]Weizmann Institute of Science, Rehovot, Israel. Correspondence to: Guy N. Rothblum <rothblum@alum.mit.edu>, Gal Yona <gal.yona@gmail.com>.

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

an underlying population distribution. The learner should output a *fair* classifier that (to the extent possible) accurately classifies the underlying population.

This goal departs from the scenario studied in (Dwork et al., 2012), where the focus was on guaranteeing metric-fairness and utility for the dataset at hand. *Generalization* of the fairness guarantee is a key difference: we focus on guaranteeing fairness not just for the (training) data set at hand, but also for the underlying population from which it was drawn. We note that perfect metric-fairness does not, as a rule, generalize from a training set to the underlying population. This presents computational difficulties for constructing learning algorithms that are perfectly metric-fair for the underlying population. Indeed, we exhibit a simple learning task that, while easy to learn without fairness constraints, becomes computationally infeasible under the perfect metric-fairness constraint (given a particular metric).[1] See below and in Section 6 for further details.

We develop a relaxed *approximate metric-fairness* framework for machine learning, where fairness does generalize from the sample to the underlying population, and present polynomial-time fair learning algorithms in this framework. We proceed to describe our setting and contributions.

Problem setting. A metric-fair learning problem is defined by a domain $\mathcal{X}$ and a similarity metric d. A metric-fair learning algorithm gets as input the metric d and a sample of labeled examples, drawn i.i.d. from a distribution $\mathcal{D}$ over labeled examples from $(\mathcal{X} \times \pm 1)$, and outputs a classifier h. To accommodate fairness, we focus on probabilistic classifiers $h : \mathcal{X} \to [0, 1]$, where we interpret $h(x)$ as the probability of label 1 (the probability of -1 is thus $(1 - h(x))$). We refer to these probabilistic classifiers as *predictors*.

Approximate Metric-Fairness. Taking inspiration from Valiant's celebrated PAC learning model (Valiant, 1984), we allow a small fairness error, which opens the door to generalization. We require that for two individuals sampled from the underlying population, with all but a small probability, if they are similar then they should be treated similarly. Similarity is measured by the statistical distance between the classification distributions given to the two individuals (we also allow a small additive slack in the similarity measure). We refer to this condition as *approximate metric-fairness (MF)*. Similarly to PAC learning, we also allow a small probability of a complete fairness failure.

Given a well-designed metric, approximate metric-fairness guarantees that almost every individual gets fair treatment compared to almost every other individual. In particular, it provides discrimination-protections to *every* group P that is not too small. However, this guarantee also has limitations: particular individuals and even small groups might encounter bias and discrimination. There are certainly settings in which this is problematic, but in other settings protecting all groups that are not too small is an appealing guarantee. The relaxation is well-motivated because approximate fairness opens the door to fairness-generalization bounds, as well as efficient learning algorithms for a rich collection of problems (see below). We elaborate on these choices in Section 2.

Competitive accuracy. Turning our attention to the accuracy objective, we follow (Dwork et al., 2012) in considering fairness to be a hard constraint (e.g. imposed by a regulator). Given the fairness constraint, what is a reasonable accuracy objective? Ideally, we would like the predictor's accuracy to approach (as the sample size grows) that of the most accurate approximately MF predictor. This is analogous to the accuracy guarantee pioneered in (Dwork et al., 2012). A *probably approximately correct and fair (PACF)* learning algorithm guarantees both approximate MF and "best-possible" accuracy. A more relaxed accuracy benchmark is approaching the accuracy of the best classifier that is approximately MF for a tighter (more restrictive) fairness-error. We refer this as a *relaxed* PACF learning algorithm (looking ahead, our efficient algorithms achieve this relaxed accuracy guarantee). We note that even relaxed PACF guarantees that the classifier is (at the very least) competitive with the best *perfectly* metric-fair classifier. We elaborate in Section 3.

Generalization bounds. A key issue in learning theory is that of generalization: to what extent is a classifier that is accurate on a finite sample $S \sim \mathcal{D}^m$ also guaranteed to be accurate w.r.t the underlying distribution? We develop strong generalization bounds for approximate metric-fairness, showing that for any class of predictors with bounded Rademacher complexity, approximate MF on the sample S implies approximate MF on the underlying distribution (w.h.p. over the choice of sample S). The use of Rademacher complexity guarantees fairness-generalization for finite classes and also for many infinite classes. Proving that approximate metric-fairness generalizes well is a crucial component in our analysis: it opens the door to polynomial-time algorithms that can focus on guaranteeing fairness (and accuracy) on the sample. Generalization also implies information-theoretic sample-complexity bounds for PACF learning, similar to those known for PAC learning (without fairness constraints). We elaborate in Section 4.

Efficient algorithms. We construct polynomial-time (relaxed) PACF algorithms for linear and logistic regression. Recall that (for fairness) we focus on regression problems:

[1] We remark that perfect metric-fairness can always be obtained trivially by outputting a constant classifier that treats all individuals identically, the challenge is achieving metric-fairness together with non-trivial accuracy.

learning predictors that assign a probability in $[0, 1]$ to each example. For linear predictors, the probability is a linear function of an example's distance from a hyperplane. Logistic predictors compose a linear function with a sigmoidal transfer function. This allows logistic predictors to exhibit sharper transitions from low predictions to high predictions. In particular, a logistic predictor can better approximate a classifier that labels examples that are below a hyperplane by -1, and examples that are above the hyperplane by 1. Linear and logistic predictors can be more powerful than they first seem: by embedding a learning problem into a higher-dimensional space, linear functions (over the expanded space) can capture the power of many of the function classes that are known to be PAC learnable (Hellerstein & Servedio, 2007). We overview these results in Section 5. We note that a key challenge in efficient metric-fair learning is that the fairness constraints are neither Lipschitz nor convex (even when the predictor is linear). This is also a challenge for proving generalization and sample complexity bounds.

Perfect metric-fairness is hard. Under mild cryptographic assumptions, we exhibit a learning problem and a similarity metric where: (i) there exists a *perfectly fair and perfectly accurate* simple (linear) predictor, but (ii) any polynomial-time perfectly metric-fair learner can only find a trivial predictor, whose error approaches 1/2. In contrast, (iii) there *does* exist a polynomial-time (relaxed) PACF learning algorithm for this task. This is an important motivation for our study of *approximate* MF. We elaborate in Section 6.

Organization. In the remainder of this paper, we go on to provide a detailed overview of our contributions. In **Section 1.2** we review related work. **Section 2** details and discusses the definition of approximate metric-fairness. Accurate and fair (PACF) learning is discussed in **Section 3**. We state and prove fairness-generalization bounds in **Section 4**. Our polynomial-time PACF learning algorithms for linear and logistic regression are in **Section 5**. **Section 6** elaborates on the hardness of *perfectly* metric-fair learning.

1.2. Related Work

There is a growing body of work attempting to study the question of algorithmic discrimination. This literature is characterized by an abundance of definitions, each capturing different discrimination concerns and notions of fairness. One high-level distinction can be drawn between *group* and *individual* notions of fairness.

Group fairness notions assume the existence of a protected attribute (e.g gender, race), which induces a partition of the instance space into some small number of groups. A fair classifier is one that achieves parity of some statistical measure across these groups. Some prominent measures include classification rates (statistical parity, see e.g (Feldman et al., 2015)), calibration, and false positive or negative rates (Kleinberg et al., 2016; Chouldechova, 2017; Hardt et al., 2016). It has been established that some of these notions are inherently incompatible with each other, in all but trivial cases (Kleinberg et al., 2016; Chouldechova, 2017). The work of (Woodworth et al., 2017) takes a step towards incorporating the fairness notion of (Hardt et al., 2016) into a statistical and computational theory of learning, and considers a relaxation of the fairness definition to overcome the computational intractability of the learning objective. The work of (Dwork et al., 2017) proposes an efficient framework for learning different classifiers for different groups in a fair manner.

Individual fairness (Dwork et al., 2012) posits that "similar individuals should be treated similarly". This powerful guarantee is formalized via a Lipschitz condition (with respect to an existing task-specific similarity metric) on the classifier mapping individuals to distributions over outcomes. Recent works (see e.g (Joseph et al.)) study different individual-level fairness in the contexts of reinforcement and online learning.

Our notion of approximate metric-fairness can be interpreted as staking a middle-ground between individual- and group-fairness. In this sense, it is similar to recent works that protect large collections of sufficiently-large groups (Hébert-Johnson et al., 2017; Kearns et al., 2017; Kim et al., 2018). A distinction from these works is in protecting *every* sufficiently-large group, rather than a large collection of groups that is fixed a priori. (Kim et al., 2018) consider a (computational) relaxation of individual fairness, focusing on settings where the metric itself is not fully known.

Finally, several works have studied fair regression (Kamishima et al., 2012; Calders et al., 2013; Zafar et al., 2017; Berk et al., 2017). The main differences in our work are a focus on metric-based fairness, a strong rigorous fairness guarantee, and proofs of competitive accuracy (both stated with respect to the underlying distribution).

2. Approximate Metric-Fairness

We require that metric-fairness holds for all but a small α fraction of pairs of individuals. That is, with all but α probability over a choice of two individuals from the underlying distribution, if the two individuals are similar then they get similar classification distributions. We think of $\alpha \in [0, 1)$ as a small constant, and note that setting $\alpha = 0$ recovers the definition of *perfect* metric-fairness (thus, setting α to be a small constant larger than 0 is indeed a relaxation). Similarity is measured by the statistical distance between the classification distributions given to the two individuals, where we also allow a small additive slack γ in the similarity measure. The larger γ is, the more "differently" similar individuals might be treated. We think of γ as a small constant, close to 0.

Definition 2.1 *A predictor h is (α, γ) approximately metric-fair (MF) with respect to a similarity metric d and a data distribution $\mathcal{D}$ if:*

$$\mathcal{L}_\gamma^F \overset{def}{=} \Pr_{x,x'\sim\mathcal{D}}[|h(x) - h(x')| > d(x, x') + \gamma] \leq \alpha \quad (1)$$

Similarly to the PAC learning model, we also allow a small δ probability of failure. This probability is taken over the choice of the training set and over the learner's coins. For example, δ bounds the probability that the randomly sampled training set is not representative of the underlying population. We think of δ as very small or even negligible. A learning algorithm is *probably approximately metric-fair* if with all but δ probability over the sample (and the learner's coins), it outputs a classifier that is (α, γ)-approximately MF. Further details are in Appendix A.2 in the full version (see supplementary material).

Given a well-designed metric, approximate metric-fairness (for sufficiently small α, γ) guarantees that almost every individual gets fair treatment compared to almost every other individual (see Appendix A.3 for a quantitative discussion). *Every* protected group P of fractional size significantly larger than α is protected in the sense that, on average, members of P are treated similarly to similar individuals outside of P. We note, however, that this guarantee does not protect single individuals or small groups (see the discussion in Section 1.1).

3. Accurate and Fair Learning

Our goal is to obtain learning algorithms that are probably approximately metric-fair, and that simultaneously guarantee non-trivial accuracy. Recall that fairness, on its own, can always be obtained by outputting a constant classifier that ignores its input and treats all individuals identically. It is the combination of the fairness and the accuracy objectives that makes for an interesting task. As discussed above, we follow (Dwork et al., 2012) in focusing on finding a classifier that maximizes accuracy, subject to the approximate metric-fairness constraint. This is a natural formulation, as we think of fairness as a hard requirement (imposed, for example, by a regulator), and thus fairness cannot be traded off for better accuracy.

As discussed above, we focus on the setting of binary classification. A *learning problem* is defined by an instance domain $\mathcal{X}$ and a class $\mathcal{H}$ of predictors (probabilistic classifiers) $h : \mathcal{X} \rightarrow [0, 1]$. A *fair* learning problem also includes a similarity metric $d : \mathcal{X}^2 \rightarrow [0, 1]$. The learning algorithm gets as input the metric d and a sample of labeled examples, drawn i.i.d. from a distribution $\mathcal{D}$ over labeled examples from $(\mathcal{X} \times \pm 1)$, and its goal is to output a predictor that is both fair and as accurate as possible. A *proper* learner outputs a predictor in the class $\mathcal{H}$, whereas an *improper* learner's output is unconstrained (but $\mathcal{H}$ is used as a benchmark for accuracy). For a learned (real-valued) predictor h, we use $err_D(h)$ to denote the expected ℓ_1 error of h (the absolute loss) on a random sample from $\mathcal{D}$.[2]

Accuracy guarantee: PACF learning. As discussed above, the goal in metric-fair and accurate learning is optimizing the predictor's accuracy subject to the fairness constraint. Ideally, we aim to approach (as the sample size grows) the error rate of the most accurate classifier that satisfies the fairness constraints. A more relaxed benchmark is guaranteeing (α, γ)-approximate metric-fairness, while approaching the accuracy of the best classifier that is (α', γ')-approximately metric-fair, for $\alpha' \in [0, \alpha]$ and $\gamma' \in [0, \gamma]$. Our efficient learning algorithms will achieve this more relaxed accuracy goal (see below). We note that even relaxed competitiveness means that the classifier is (at the very least) competitive with the best *perfectly* metric-fair classifier.

These goals are captured in the following definition of *probably approximately correct and fair (PACF) learning*. Crucially, both fairness and accuracy goals are stated with respect to the (unknown) underlying distribution.

Definition 3.1 (PACF Learning) *A learning algorithm $\mathcal{A}$ PACF-learns a hypothesis class $\mathcal{H}$ if for every metric d and population distribution $\mathcal{D}$, every required fairness parameters $\alpha, \gamma \in [0, 1)$, every failure probability $\delta \in (0, 1)$, and every error parameters $\epsilon, \epsilon_\alpha, \epsilon_\gamma \in (0, 1)$, there exists a sample complexity $m = \text{poly}\left(\frac{\log|\mathcal{X}|\cdot\log(1/\delta)}{\alpha\cdot\gamma\cdot\epsilon.\,\epsilon_\alpha\cdot\epsilon_\gamma}\right)$ and constants $\alpha', \gamma' \in [0, 1)$ (specified below), such that with all but δ probability over an i.i.d. sample of size m and $\mathcal{A}$'s coin tosses, the output predictor h satisfies the following two conditions:*

1. **Fairness***: h is (α, γ)-approximately metric-fair w.r.t. the metric d and the distribution $\mathcal{D}$.*
2. **Accuracy***: Let $\mathcal{H}'_F$ be the subclass of hypotheses in $\mathcal{H}$ that are $(\alpha' - \epsilon_\alpha, \gamma' - \epsilon_\gamma)$-approximately MF, then:*

$$err_D(h) \leq \min_{h' \in \mathcal{H}'_F} err_D(h') + \epsilon$$

We say that $\mathcal{A}$ is efficient *if it runs in time* $\text{poly}(m)$*. If accuracy holds for $\alpha' = \alpha$ and $\gamma' = \gamma$, then we stay that $\mathcal{A}$ is a* strong *PACF learning algorithm. Otherwise, we say that $\mathcal{A}$ is a* relaxed *PACF learning algorithm.*

See Appendix B and Definitions B.2 and B.3 for a full treatment. Note that the accuracy guarantee is *agnostic*: we

[2] All results also translate to ℓ_2 error (the squared loss).

make no assumptions about the way the training labels are generated. Agnostic learning is particularly well suited to our setting: since we make no assumptions about the metric d, even if the labels are generated by $h \in \mathcal{H}$, it might be the case that d does not allow for accurate predictions, in which case a fair learner cannot compete with h's accuracy.

4. Generalization

Generalization is a key issue in learning theory. We develop strong generalization bounds for approximate metric-fairness, showing that with high probability, guaranteeing *empirical* approximate MF on a training set also guarantees approximate MF on the underlying distribution (w.h.p. over the choice of sample S). This generalization bound opens the door to polynomial-time algorithms that can focus on guaranteeing fairness (and accuracy) on the sample and effectively rules out the possibility of creating a "false facade" of fairness (i.e, a classifier that appears fair on a random sample, but is not fair w.r.t new individuals).

Towards proving generalization, we define the empirical fairness loss on a sample S (a training set). Fixing a fairness parameter γ, a predictor h and a pair of individuals x, x' in the training set, consider the MF loss on the "edge" between x and x' (recall that the MF loss is 1 if the "internal" inequality of Equation (1) holds, and 0 otherwise). Observe that the losses on the $\binom{|S|}{2}$ edges are not independent random variables (over the choice of S), because each individual $x \in S$ affects many edges. Thus, rather than count the empirical MF loss over all edges, we restrict ourselves to a "matching" $M(S)$ in the complete graph whose vertices are S: a collection of edges, where each individual is involved in exactly one edge. The empirical MF loss of h on S is defined as the average MF loss over edges in $M(S)$.[3] Note that, since we restricted our attention to a matching, the MF losses on these edges are now independent random variables (over the choice of S). A classifier is *empirically* (α, γ)-approximately MF if its empirical MF loss is at most α. We are now ready to state our generalization bound:

Theorem 4.1 *Let $\mathcal{H}$ be a hypothesis class with Rademacher complexity $R_m(\mathcal{H}) = (r/\sqrt{m})$. For every $\delta \in (0,1)$ and every $\epsilon_\alpha, \epsilon_\gamma \in (0,1)$, there exists a sample complexity $m = O\left(\frac{r^2 \cdot \ln(1/\delta)}{\epsilon_\alpha^2 \cdot \epsilon_\gamma^2}\right)$, such that with probability at least $1-\delta$ over an i.i.d sample $S \sim \mathcal{D}^m$, simultaneously for every $h \in \mathcal{H}$: if h is (α, γ)-approximately metric-fair on the sample S, then h is also $(\alpha + \epsilon_\alpha, \gamma + \epsilon_\gamma)$-approximately metric-fair on the underlying distribution $\mathcal{D}$.*

See Appendix A.4 and Theorem A.9 for a full statement and discussion (and see Definition A.8 for a definition of Rademacher complexity). Rademacher complexity differs from the celebrated VC-dimension in several respects: first, it is defined for any class of real-valued functions (making it suitable for our setting of learning probabilistic classifiers); second, it is data-dependent and can be measured from finite samples (indeed, Theorem 4.1 can be stated w.r.t. the *empirical* Rademacher complexity on a given sample); third, it often results in tighter uniform convergence bounds (see, e.g, (Koltchinskii & Panchenko, 2002)). We note that for every finite hypothesis class $\mathcal{H}$ whose range is $[0,1]$, the Rademacher complexity is bounded by $O(\sqrt{\log |\mathcal{H}|/m})$.

Technical Overview of Theorem 4.1. For any class of (bounded) real-valued functions $\mathcal{F}$, the maximal difference (over all functions $f \in \mathcal{F}$) between the function's empirical average on a randomly drawn sample, and the function's true expectation over the underlying distribution, can be bounded in terms of the Rademacher complexity of the class (as well as the sample size and desired confidence). For a hypothesis class $\mathcal{H}$ and a loss function ℓ, applying this result for the class $\mathcal{L}(\mathcal{H}) = \{\ell_h\}_{h \in \mathcal{H}}$ yields a bound on the maximal difference (over all hypotheses $h \in \mathcal{H}$) between the true loss and the empirical loss, in terms of the Rademacher complexity of the composed class $\mathcal{L}(\mathcal{H})$. If the loss function ℓ is G-Lipschitz, this can be converted to a bound in terms of the Rademacher complexity of $\mathcal{H}$ using the fact that $R(\mathcal{L}(\mathcal{H})) \leq G \cdot R(\mathcal{H})$.

Turning our attention to generalization of the fairness guarantee, we are faced with the problem that our "0-1" MF loss function is *not Lipschitz*. We resolve this by defining an approximation ℓ' to the MF loss that is a piece-wise linear and G-Lipschitz function. The approximation ℓ' *does* generalize, and so we conclude that the empirical MF loss is close to the empirical value of ℓ', which is close to the *true* value of ℓ', which in turn is close to the *true* MF loss. The approximation incurs a $1/G$ additive slack in the fairness guarantee. The larger G is, the more accurately ℓ' approximates the MF loss, but this comes at the price of increasing the Lipschitz constant (which hurts generalization). The generalization theorem statement above reflects a choice of G that trades off these conflicting concerns.

4.1. Information-Theoretic Sample Complexity

The fairness-generalization result of Theorem 4.1 implies that, from a sample-complexity perspective, any hypothesis class is strongly PACF learnable, with sample complexity comparable to that of standard PAC learning.

Theorem 4.2 *Let $\mathcal{H}$ be a hypothesis class with Rademacher complexity $R_m(\mathcal{H}) = (r/\sqrt{m})$. Then $\mathcal{H}$ is information-theoretically strongly PACF learnable with sample complex-*

[3] The choice of *which* matching is used does not affect any of the results. Note that we could also choose to average over *all* the edges in the graph induced by S. Generalization bounds still follow, but the rate of convergence is not faster.

ity $m = O\left(\frac{r^2 \ln(1/\delta)}{(\epsilon')^2}\right)$, *for* $\epsilon' = \min\{\epsilon, \epsilon_\alpha, \epsilon_\gamma\}$.

5. Efficient Fair Learning

One of our primary contributions is the construction of polynomial-time relaxed-PACF learning algorithms for expressive hypothesis classes. We focus on linear classification tasks, where the labels are determined by a separating hyperplane. Learning linear classifiers, also referred to as halfspaces or linear threshold functions, is a central tool in machine learning. By embedding a learning problem into a higher-dimensional space, linear classifiers (over the expanded space) can capture surprisingly strong classes, such as polynomial threshold functions (see, for example, the discussion in (Hellerstein & Servedio, 2007)). The "kernel trick" (see, e.g, (Shalev-Shwartz & Ben-David, 2014)) can allow for efficient solutions even over very high (or infinite) dimensional embeddings. Many of the known (distribution-free) PAC learning algorithms can be derived by learning linear threshold functions (Hellerstein & Servedio, 2007).

Recall that in *metric-fair* learning, we aim to learn a probabilistic classifier, or a *predictor*, that outputs a real value in $[0, 1]$. We interpret the output as the probability of assigning the label 1. We are thus in the setting of *regression*. We show polynomial-time relaxed-PACF learning algorithms for *linear regression* and for *logistic regression*. See Appendix D.2 for full and formal details.

5.1. Linear Regression

Linear regression, the task of learning linear predictors, is an important and well-studied problem in the machine learning literature. In terms of accuracy, this is an appealing class when we expect a linear relationship between the probability of the label being 1 and the distance from a hyperplane. Taking the domain $\mathcal{X}$ to be the unit ball, we define the class of linear predictors as:

$$H_{lin} \stackrel{\text{def}}{=} \{\mathbf{x} \mapsto (1 + \langle \mathbf{w}, \mathbf{x} \rangle)/2 : \ \|\mathbf{w}\| \leq 1\} .$$

We restrict w to the unit ball to guarantee that $\langle \mathbf{w}, \mathbf{x} \rangle \in [-1, 1]$. We then invoke a linear transformation so that the final prediction is in $[0, 1]$, as required. Restricting the predictor's output to the range $[0, 1]$ is important. In particular, it means that a linear predictor must be $(1/2)$-Lipschitz, which might not be appropriate for certain classification tasks (see the discussion of logistic regression below).

We show a relaxed PACF learning algorithm for H_{lin}:

Theorem 5.1 *H_{lin} is relaxed PACF learnable with sample and time complexities of poly*$(\frac{1}{\epsilon_\gamma}, \frac{1}{\epsilon_\alpha}, \frac{1}{\epsilon}, \log\frac{1}{\delta})$. *For every $\gamma' \in [0, 1)$ and $\alpha' = (\alpha \cdot \gamma - \gamma')$, the accuracy of the learned predictor approaches (or beats) the most accurate (α', γ')-approximately MF predictor.*

Algorithm overview. Since the Rademacher complexity of (bounded) linear functions is small (Kakade et al., 2009), Theorem 4.1 implies that empirical approximate metric-fairness on the training set generalizes to the underlying population. Thus, given the metric and a training set, our task is to find a linear predictor that is as accurate as possible, conditioned on the *empirical* fairness constraint. We use $H = H_{lin}$ to denote the class of linear predictors defined above. Fixing desired fairness parameters $\alpha, \gamma \in (0, 1)$, let $\widehat{H}^{\alpha,\gamma} \subseteq H$ be the subset of linear functions that are also (α, γ)-approximately MF on the training set. Given a training set S of m labeled examples, we would like to solve the following optimization problem:

$$\operatorname*{argmin}_{h \in H} err_S(h) \text{ subject to } h \in \widehat{H}^{\alpha,\gamma}$$

Observe, however, that $\widehat{H}^{\alpha,\gamma}$ is not a convex set. This is a consequence of the "0/1" metric-fairness loss. Thus, we do not know how to solve the above optimization problem efficiently. Instead, we will further constrain the predictor h by bounding its ℓ_1 *MF loss*. For a predictor h let its (empirical) ℓ_1 MF violation $\xi_S(h)$ be given by:

$$\xi_S(h) = \sum_{(x,x') \in M(S)} \max\left(0, |h(x) - h(x')| - d(x, x')\right) .$$

For $\tau \in [0, 1]$, we take $\widehat{H}^{\tau}_{\ell_1} \subset H$ to be the set of linear predictors h s.t. $\xi_S(h) \leq \tau$. For any fixed τ, this is a convex set, and we can find the most (empirically) accurate predictor in $\widehat{H}^{\tau}_{\ell_1}$ in polynomial time. For fairness, we show that small ℓ_1 fairness loss also implies the standard notion of approximate metric-fairness (with related parameters α, γ). For accuracy, we also show that approximate metric-fairness (with smaller fairness parameters) implies small ℓ_1 loss. Thus, optimizing over predictors whose ℓ_1 loss is bounded gives a predictor that is competitive with (a certain class of) approximately MF predictors. In particular for $\tau, \sigma \in [0, 1)$ we have: $\widehat{H}^{\tau-\sigma,\sigma} \subseteq \widehat{H}^{\tau}_{\ell_1} \subseteq \widehat{H}^{\frac{\tau}{\gamma},\gamma}$. Thus, by picking $\tau = \alpha \cdot \gamma$ we guarantee (empirical) (α, γ)-approximate metric-fairness. Moreover, for any choice of σ, the set over which we optimize contains all of the predictors that are $((\alpha\gamma - \sigma), \sigma)$-approximately MF. Thus, our (empirical) accuracy is competitive with all such predictors, and we obtain a relaxed PACF algorithm. The empirical fairness and accuracy guarantees generalize beyond the training set by Theorem 4.1 (fairness-generalization) and a standard uniform convergence argument for accuracy.

5.2. Logistic Regression

Logistic regression is another appealing class. Here, the prediction need not a be a linear function of the distance from a hyperplane. Rather, we allow the use of a sigmoid function $\phi_\ell : [-1, 1] \to [0, 1]$ defined as $\phi_\ell(z) = \frac{1}{1+\exp(-4\ell \cdot z)}$

(which is continuous and ℓ-Lipschitz). The class of logistic predictors is formed by composing a linear function with a sigmoidal transfer function:

$$H_{\phi,L} \overset{\text{def}}{=} \{\mathbf{x} \mapsto \phi_\ell\left(\langle w, \mathbf{x}\rangle\right) : \|\mathbf{w}\| \leq 1, \ell \in [0, L]\} \quad (2)$$

The sigmoidal transfer function gives the predictor the power to exhibit sharper transitions from low predictions to high predictions around a certain distance (or decision) threshold. For example, suppose a distance from the hyperplane provides a quality score for candidates with respect to a certain task. Suppose also that an employer wants to hire candidates whose quality scores are above some threshold $\eta \in [-1, +1]$. The class $H_{\phi,L}$ can give probabilities close to 0 to candidates whose quality scores are under $\eta - 1/L$, and probabilities close to 1 to candidates whose quality scores are over $\eta + 1/L$. Linear predictors, on the other hand, need to be $(1/2)$-Lipschitz (since we restrict their output to be in $[0, 1]$, see Section 5.1). Logistic predictors seem considerably better-suited to this type of scenario. Indeed, the class $H_{\phi,L}$ can achieve good accuracy on linearly separable data whose margin (i.e. the expected distance from the hyperplane) is larger than $1/L$. Moreover, similarly to linear threshold functions, logistic regression can be applied after embedding the learning problem into a higher-dimensional space. For example, in the "quality score" example above, the score could be computed by a low-degree polynomial.

Our primary technical contribution is a polynomial-time relaxed PACF learner for $H_{\phi,L}$ where L is constant.

Theorem 5.2 *For every constant $L > 0$, $H_{\phi,L}$ is relaxed PACF learnable with sample and time complexities of poly$(\frac{1}{\epsilon_\gamma}, \frac{1}{\epsilon_\alpha}, \frac{1}{\epsilon}, \log \frac{1}{\delta})$. For every $\gamma' \in [0, 1)$ and $\alpha' = (\alpha \cdot \gamma - \gamma')$, the learned predictor's accuracy approaches the best (α', γ')-approximately MF predictor.*

More generally, our algorithm is exponential in the parameter L. Recall that we expect to have good accuracy on linearly separable data whose margins are larger than $(1/L)$. Thus, one can interpret the algorithm as having runtime that is exponential in the reciprocal of the (expected) margin.

Algorithm overview. We note that fair learning of logistic predictors is considerably more challenging than the linear case because the sigmoidal transfer function specifies non-convex fairness constraints. In standard logistic regression, polynomial-time learning is achieved by replacing the standard loss with a convex logistic loss. In metric-fair learning, however, it not clear how to replace the sigmoidal transfer function by a convex surrogate.

To overcome these barriers, we use *improper* learning. We embed the linear problem at hand into a higher-dimensional space, where logistic predictors and their fairness constraints can be approximated by convex expressions. To do so, we use a beautiful result of Shalev-Schwartz *et al.* (Shalev-Shwartz et al., 2011) that presents a particular infinite-dimensional kernel space where our fairness constraints can be made convex.

In particular, we replace the problem of PACF learning $H_{\phi,L}$ with the problem of PACF learning H_B, a class of linear predictors with norm bounded by B in a RHKS defined by Vovk's infinite-dimension polynomial kernel, $k(x, x') = (1 - \langle x, x'\rangle)^{-1}$. We learn the linear predictor in this RHKS using the result of Theorem 5.1 to obtain a relaxed PACF algorithm for H_B. We use the kernel trick to argue that the sample complexity is $m = O(B/(\epsilon')^2)$, where $\epsilon' = \min(\epsilon, \epsilon_\alpha, \epsilon_\gamma)$, and the time complexity is *poly*(m).

For every $B \geq 0$, we can thus learn a linear predictor (in the above RHKS) that is (empirically) sufficiently fair, and whose (empirical) accuracy is competitive with all the linear predictors with norm bounded by B that are $((\alpha\gamma - \sigma), \sigma)$-approximately MF, for any choice of σ. To prove PACF learnability of $H_{\phi,L}$, we build on the polynomial approximation result of Shalev-Schwartz *et al.* (Shalev-Shwartz et al., 2011) to show that taking B to be sufficiently large ensures that the accuracy of the set of (α, γ)-AMF predictors in $H_{\phi,L}$ is comparable to the accuracy of the set of (α, γ)-AMF predictors in H_B. This requires a choice of B that is $\exp(O(L \cdot \ln(L/\epsilon')))$, which is where the exponential dependence on L comes in.

6. Hardness of Perfect Metric-Fairness

As discussed above, perfect metric-fairness *does not generalize* from a training set to the underlying population. For example, consider a very small subset of the population that isn't represented in the training set. A classifier that discriminates against this small subset might be perfectly metric-fair *on the training set*. The failure of generalization poses serious challenges to constructing learning algorithms. Indeed, we show that perfect metric-fairness can make simple learning tasks computationally intractable (with respect to a particular metric).

We present a natural learning problem and a metric where, even though a *perfectly fair and perfectly accurate* simple (linear) classifier exists, it cannot be found by any polynomial-time learning algorithm that is perfectly metric-fair. Indeed, any such algorithm can only find trivial classifiers with error rate approaching 1/2 (not much better than random guessing). The learner can tell that a particular (linear) classifier is *empirically* perfectly fair (and perfectly accurate). However, even though the classifier is perfectly fair on the underlying distribution, the (polynomial-time) learner cannot certify that this is the case, and thus it has to settle for outputting a trivial classifier. We note that there does exist an *exponential-time* perfectly metric-fair learning algorithm with a competitive accuracy guarantee (see footnote 5 in the full version). The issue is the computational complexity of this task. In contrast, the relaxed notion of approximate metric-fairness does allow for *polynomial-time* relaxed-PACF learning algorithms that obtain competitive accuracy for this task (as it does for a rich class of learning problems, see Section 5).

We present an overview of the hard learning task and discuss its consequences below. See Appendix E and Theorem E.1 for a more formal description. Since we want to argue about computational intractability, we need to make computational assumptions (in particular, if $P = NP$, then perfectly metric-fair learning would be tractable). We will make the *minimal* cryptographic hardness assumption that one-way functions exist (see, e.g, (Goldreich, 2001)).

Simplified construction. For this sketch, we take a uniform distribution $\mathcal{D}$ over a domain $\mathcal{X} = \{\pm 1\}^n$. For an item (or individual) $x \in \mathcal{X}$, its label will be given by the linear classifier $w(x) = x_1$. Note that the linear classifier w indeed is perfectly accurate.[4]

To argue that fair learning is intractable, we construct two metrics d_U and d_V that are *computationally indistinguishable*: no polynomial-time algorithm can tell them apart (even given the explicit description of the metric).[5] We construct these metrics so that d_U does not allow *any* non-trivial accuracy, whereas d_V essentially imposes no fairness constraints. Thus, w is a perfectly fair and perfectly accurate classifier w.r.t. d_V. Now, since a polynomial-time learning algorithm $\mathcal{A}$ cannot tell d_U and d_V apart, it has to output the same (distribution on) classifiers given either of these two metrics. If $\mathcal{A}$, given d_U, outputs a classifier with non-trivial accuracy, then it violates perfect metric-fairness. Thus, when given d_U, $\mathcal{A}$ must (with high probability) output a classifier with error close to $1/2$. This remains the case even when $\mathcal{A}$ is given the metric d_V (by indistinguishability), despite the fact perfect metric-fairness under d_V allows for *perfect accuracy*.

We construct the metrics as follows. The metric d_V gives *every* pair of individuals $x, x' \in \mathcal{X}$ distance 1. The metric d_U, on the other hand, partitions the items in $\mathcal{X}$ into disjoint pairs (x, x') where the label of x is 1, the label of x' is -1, but the distance between x and x' is 0.[6] Thus, the metric d_U assigns to each item $x \in X$ a "hidden counterpart" x' that is *identical* to x, but has the opposite label. The distance between any two distinct elements that are not "hidden counterparts" is 1 (as in d_V). The metric d_U specifies that hidden counterparts (x, x') are *identical*, and thus any perfectly metric-fair classifier h must treat them identically. Since x and x' have opposing labels, h's average error on the pair must be $1/2$. The support of $\mathcal{D}$ is partitioned into disjoint hidden counterparts, and thus we conclude that $err_{\mathcal{D}}(h) = 1/2$. Note that this is true regardless of h's complexity (in particular, it also rules out improper learning). We construct the metrics using a cryptographic pseudorandom generator (PRG), which specifies the hidden counterparts (in d_U) or their absence (in d_V). See the full version for details.

Discussion. We make several remarks about the above result. First, note that the data distribution is fixed, and the optimal classifier is linear and simple: it only considers a single coordinate. This makes the hardness result sharper: without fairness, the learning task is trivial (indeed, since the classifier is fixed there is nothing to learn). It is the fairness constraint (and only the fairness constraint) that leads to intractability. The computational hardness of perfectly fair learning applies also to improper learning. Finally, the metrics for which we show hardness are arguably contrived (though we note they do obey the triangle inequality). This rules out perfectly metric-fair learners that work for *any* given metric. A natural direction for future work is restricting the choice of metric, which may make perfectly metric-fair learning feasible.

[4] Note that the expected margin in this distribution is small compared to the norms of the examples. This is for simplicity and readability. The full hardness result is shown (in a very similar manner) for data where the margins are large. In particular, this means that the class of predictors $H_{\phi,L}$ can achieve good accuracy with constant L. See Appendix E.

[5] More formally, we construct two *distribution* on metrics, such that no polynomial-time algorithm can tell whether a given metric was sampled from the first distribution or from the second. For readability, we mostly ignore this distinction in this sketch.

[6] Formally, d_U is a pseudometric, since it has distinct items at distance 0. We can make d_U be a true metric by replacing the distance 0 with an arbitrarily small positive quantity. The hardness result is essentially unchanged.

Acknowledgements

We thank Cynthia Dwork and Omer Reingold for invaluable and illuminating conversations. This work was supported by the ISRAEL SCIENCE FOUNDATION (grant No. 5219/17).

References

Berk, R., Heidari, H., Jabbari, S., Joseph, M., Kearns, M., Morgenstern, J., Neel, S., and Roth, A. A convex framework for fair regression. *arXiv preprint arXiv:1706.02409*, 2017.

Calders, T., Karim, A., Kamiran, F., Ali, W., and Zhang, X. Controlling attribute effect in linear regression. In *Data Mining (ICDM), 2013 IEEE 13th International Conference on*, pp. 71–80. IEEE, 2013.

Chouldechova, A. Fair prediction with disparate impact: A study of bias in recidivism prediction instruments. *Big data*, 5(2):153–163, 2017.

Dwork, C., Hardt, M., Pitassi, T., Reingold, O., and Zemel, R. S. Fairness through awareness. In *Innovations in Theoretical Computer Science (ITCS)*, pp. 214–226, 2012.

Dwork, C., Immorlica, N., Kalai, A. T., and Leiserson, M. Decoupled classifiers for fair and efficient machine learning. *CoRR*, abs/1707.06613, 2017. URL http://arxiv.org/abs/1707.06613.

Feldman, M., Friedler, S. A., Moeller, J., Scheidegger, C., and Venkatasubramanian, S. Certifying and removing disparate impact. In *Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pp. 259–268. ACM, 2015.

Goldreich, O. *The Foundations of Cryptography - Volume 1, Basic Techniques*. Cambridge University Press, 2001. ISBN 0-521-79172-3.

Hardt, M., Price, E., and Srebro, N. Equality of opportunity in supervised learning. *CoRR*, abs/1610.02413, 2016. URL http://arxiv.org/abs/1610.02413.

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

Hellerstein, L. and Servedio, R. A. On PAC learning algorithms for rich boolean function classes. *Theor. Comput. Sci.*, 384(1):66–76, 2007. doi: 10.1016/j.tcs.2007.05.018. URL https://doi.org/10.1016/j.tcs.2007.05.018.

Joseph, M., Kearns, M., Morgenstern, J., Neel, S., and Roth, A. Better fair algorithms for contextual bandits.

Kakade, S. M., Sridharan, K., and Tewari, A. On the complexity of linear prediction: Risk bounds, margin bounds, and regularization. In *Advances in neural information processing systems*, pp. 793–800, 2009.

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

Kearns, M., Neel, S., Roth, A., and Wu, Z. S. Preventing fairness gerrymandering: Auditing and learning for subgroup fairness. *arXiv preprint arXiv:1711.05144*, 2017.

Kim, M. P., Reingold, O., and Rothblum, G. N. Fairness through computationally-bounded awareness. *CoRR*, abs/1803.03239, 2018. URL http://arxiv.org/abs/1803.03239.

Kleinberg, J., Mullainathan, S., and Raghavan, M. Inherent trade-offs in the fair determination of risk scores. *arXiv preprint arXiv:1609.05807*, 2016.

Koltchinskii, V. and Panchenko, D. Empirical margin distributions and bounding the generalization error of combined classifiers. *Annals of Statistics*, pp. 1–50, 2002.

O'Neil, C. *Weapons of Math Destruction: How Big Data Increases Inequality and Threatens Democracy*. Crown, New York, 2016.

Shalev-Shwartz, S. and Ben-David, S. Understanding machine learning: From theory to algorithms. chapter 16, pp. 215–226. Cambridge university press, 2014.

Shalev-Shwartz, S., Shamir, O., and Sridharan, K. Learning kernel-based halfspaces with the 0-1 loss. *SIAM Journal on Computing*, 40(6):1623–1646, 2011.

Valiant, L. G. A theory of the learnable. *Commun. ACM*, 27 (11):1134–1142, 1984. doi: 10.1145/1968.1972. URL http://doi.acm.org/10.1145/1968.1972.

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

Zafar, M. B., Valera, I., Gomez Rodriguez, M., and Gummadi, K. P. Fairness constraints: Mechanisms for fair classification. *arXiv preprint arXiv:1507.05259*, 2017.