Probably Approximately Metric-Fair Learning

Guy N. Rothblum*
Weizmann Institute of Science

Gal Yona†
Weizmann Institute of Science

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.

*rothblum@alum.mit.edu. Research supported by the ISRAEL SCIENCE FOUNDATION (grant No. 5219/17).
†gal.yona@gmail.com. Research supported by the ISRAEL SCIENCE FOUNDATION (grant No. 5219/17).

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'N16] 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, political affiliation, 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"* notion of fairness, sometimes referred to as *statistical parity*.

Pointing out several weakness of group-level notions of fairness, the seminal work of [DHP+12] 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: Approximately Metric-Fair Machine Learning

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 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 [DHP+12], 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 training set 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 [Val84], 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 and their consequences in Section 2.

Competitive accuracy. Turning our attention to the accuracy objective, we follow [DHP$^+$12] in considering fairness to be a hard constraint (e.g. imposed by a regulator). Given the fairness constraint, what is a reasonable accuracy objective? Ideally, we would like the predictor's accuracy to approach (as the sample size grows) that of the most accurate approximately MF predictor. This is analogous to the accuracy guarantee pioneered in [DHP$^+$12]. 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.

[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.

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 that are similar to those known for PAC learning (without any 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: 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 [HS07]. 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. Berk *et al.* [BHJ$^+$17] also study fair regression and formulate a measure of individual fairness loss, albeit in a different setting without a metric (see Section 7).

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* metric-fairness. We elaborate in Section 6.

Organization. In the remainder of this section we provide an overview of our contributions. **Section 2** details and discusses the definition of approximate metric-fairness and its relationship to related works. 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. Further related work is discussed in **Section 7**.

Full and formal details are in **Sections A through E**. Conclusions and a discussion of future directions are in Section **F**.

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:*

$$\mathcal{L}_{\gamma}^{F} \triangleq \mathop{\mathbf{Pr}}_{x,x' \sim \mathcal{D}}[|h(x) - h(x')| > d(x, x') + \gamma] \leq \alpha \tag{1}$$

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 Section A.

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 Section 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).

Between group and individual fairness: related works. Recent works [HJKRR17, KNRW17] study fairness notions that aim to protect large collections of sufficiently-large groups. Similarly to our work, these can be viewed as falling between individual and group notions of fairness. A distinction from these works is that approximate metric-fairness protects *every* sufficiently-large group, rather than a large collection of groups that is fixed a priori. Recent works [GJKR18, KRR18] extend the study of metric fairness to settings where the metric is not known (whereas we focus on a setting where the metric is fixed and known in its entirety), and consider relaxed fairness notions that allow individual fairness to be violated.

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 predictor that ignores its input and treats all individuals identically (indeed, such a classifier is *perfectly* metric-fair). It is the combination of the fairness and the accuracy objectives that makes for an interesting task. As discussed above, we follow [DHP$^+$12] in focusing on finding a predictor 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) $h : \mathcal{X} \to [0, 1]$. A *fair* learning problem also includes a similarity metric $d : \mathcal{X}^2 \to [0, 1]$. 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 $\epsilon, \epsilon_\alpha, \epsilon_\gamma \in (0, 1)$ the following holds:*

There exists a sample complexity $m = \text{poly}\left(\frac{\log|\mathcal{X}|\cdot\log(1/\delta)}{\alpha\cdot\gamma\cdot\epsilon\cdot\epsilon_\alpha\cdot\epsilon_\gamma}\right)$ 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 $\mathcal{H}'_F$ denote the subclass of hypotheses in $\mathcal{H}$ that are $(\alpha' - \epsilon_\alpha, \gamma' - \epsilon_\gamma)$-approximately metric-fair, 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 Section B and Definitions B.2 and B.3 for a full treatment. Note that the accuracy guarantee is *agnostic*: we make no assumptions about the way the training labels are generated. Agnostic learning is particularly well suited to our metric-fairness setting: since we make no assumptions

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

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 the following holds:*

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 Section A.5 and Theorem A.12 for a full statement and discussion (and see Definition A.11 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, [KP02]). 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

[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 than restricting our attention to a matching.

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.

4.0.1 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. An exponential-time PACF learning algorithm simply finds the predictor in $\mathcal{H}$ that minimizes the empirical error, while also satisfying empirical approximate metric-fairness.

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 complexity $m = O\left(\frac{r^2 \ln(1/\delta)}{(\epsilon')^2}\right)$, 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 [HS07]). The "kernel trick" (see, e.g, [SSBD14]) 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 [HS07].

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 Section D 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:

$$H_{lin} \stackrel{\text{def}}{=} \left\{ \mathbf{x} \mapsto \frac{1 + \langle \mathbf{w}, \mathbf{x} \rangle}{2} : \ \|\mathbf{w}\| \leq 1 \right\},$$

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 [KST09], 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:

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

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).[4] 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 accuracy of the learned predictor approaches (or beats) the most accurate (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

[4]This might, at first glance, seem like a technicality. After all, why not simply consider linear predictors whose output can be in a larger range? The problem is that it isn't clear how to plug these larger values into the fairness constraints in a way that keeps the optimization problem convex and also has competitive accuracy.

constraints. In standard logistic regression, where fairness is not a concern, 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.* [SSSS11] 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.* [SSSS11] 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,[5] the issue is the computational complexity of this task. In

[5]For example, an exponential-time algorithm could learn by enumerating all possible classifiers, discarding all the ones that are not perfectly metric-fair (using a brute-force search over all pairs of individuals for each candidate classifier), and then output the most-accurate classifier among the perfectly metric-fair ones. It is important to note that this algorithm doesn't try to guarantee *empirical* perfect metric-fairness, which we know does not generalize. Rather, the learner has to consider the fairness behavior over all pairs of individuals.

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 Section 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 [Gol01] for further background.

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.[6]

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).[7] 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.[8] 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 hardness result. First, note that the data distribution is fixed, and the optimal classifier is linear and very simple: it only considers a single

[6]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 Section E.

[7]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.

[8]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.

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.

7 Further Related Work

There is a growing body of work attempting to study the question of algorithmic discrimination, particularly through the lens of machine learning. This literature is characterized by an abundance of definitions, each capturing different discrimination concerns and notions of fairness. This literature is vast and growing, and so we restrict our attention to the works most relevant to ours.

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 [FFM+15]), calibration, and false positive or negative rates [KMR16, Cho17, HPS16]. It has been established that some of these notions are inherently incompatible with each other, in all but trivial cases [KMR16, Cho17]. The work of [WGOS17] takes a step towards incorporating the fairness notion of [HPS16] 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 [DIKL17] proposes an efficient framework for learning different classifiers for different groups in a fair manner.

Individual fairness [DHP+12] 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 [JKMR16, JKM+] study different individual-level fairness guarantees in the contexts of reinforcement and online learning. The work of [ZWS+13] aims to learn an intermediate "fair" representation that best encodes the data while successfully obfuscating membership in a protected group. See also the more recent work [BCZC17].

Several works have studied fair regression [KAAS12, CKK+13, ZVGRG17, BHJ+17]. The main differences in our work are a focus on metric-based individual fairness, a strong rigorous fairness guarantee, and proofs of competitive accuracy (both stated with respect to the underlying population distribution).

A Metric Fairness Definitions

A.1 (Perfect) Metric-Fairness

Dwork *et al.* [DHP+12] introduced individual fairness, a similarity-based fairness notion in which a probabilistic classifier is said to be fair if it assigns similar distributions to similar individuals.

Definition A.1 (Perfect Metric-Fairness)**.** *A probabilistic classifier $h : \mathcal{X} \to [0,1]$ is said to be perfectly metric-fair w.r.t a distance metric $d : \mathcal{X} \times \mathcal{X} \to [0,1]$, if for every $x, x' \in \mathcal{X}$,*

$$\Lambda\left(h(x), h(x')\right) \leq d(x, x') \tag{3}$$

where $h(x)$ is interpreted as the probability h will assign the label $+1$ to $x \in \mathcal{X}$, Λ is a distance measure between distributions and d is a task-specific distance metric that is assumed to be known in advance. Throughout this work we take Λ to be the statistical distance, yielding $\Lambda\left(h(x), h(x')\right) = |h(x) - h(x')|$.

In the setting considered by [DHP+12], a finite set of individuals V should be assigned outcomes from a set A. Under the assumption that d is known, they demonstrated that the problem of minimizing an arbitrary loss function $L : V \times A \to R$, subject to the individual fairness constraint can be formulated as an LP and thus can be solved in time $\text{poly}(|A|, |V|)$.

A.2 Approximate Metric-Fairness

We consider a learning setting in which the goal is learning a classifier h that satisfies the fairness constraint in Equation (3) with respect to some unknown distribution $\mathcal{D}$ over $\mathcal{X}$, after observing a finite sample $S \sim \mathcal{D}^m$. To this end, we introduce a metric-fairness loss function that, for a given classifier h and a pair of individuals in $\mathcal{X}$, assigns a penalty of 1 if the fairness constraint is violated by more than a γ additive term.

Definition A.2. *For a metric d and $\gamma \geq 0$, the metric-fairness loss on a pair $(x, x') \in \mathcal{X}$ is*

$$\ell_{\gamma,d}(h, \left(x, x'\right)) = \begin{cases} 1 & \Lambda\left(h(x), h(x')\right) > d(x, x') + \gamma \\ 0 & \Lambda\left(h(x), h(x')\right) \leq d(x, x') + \gamma \end{cases} \tag{4}$$

The overall metric-fairness loss for a hypothesis h is the expected violation for a random pair according to $\mathcal{D}$.

Definition A.3 (Metric-Fairness Loss)**.** *For a metric d and $\gamma \geq 0$,*

$$\mathcal{L}^F_{\mathcal{D},d,\gamma}(h) = \mathbb{E}_{x,x' \sim \mathcal{D}}\left[\ell_{\gamma,d}(h, (x, x'))\right] \tag{5}$$

We go on to define the empirical fairness loss, a data-dependent quantity designed to estimate the unknown $\mathcal{L}^F_{\mathcal{D},d,\gamma}(h)$. To this end, we think of a sample $S \sim \mathcal{D}^m$ as defining a complete weighed graph, denoted $G(S)$, whose vertices are S and whose edges are weighed by $w(e) = w(x_i, x_j) = d(x_i, x_j)$. Now, observe that when S is sampled i.i.d from $\mathcal{D}$, any matching $M \subseteq G(S)$[9] is an i.i.d sample from $\mathcal{D} \times \mathcal{D}$. We now define the empirical loss by replacing the expectation over $\mathcal{D}$ in Equation (5) with the expectation over some matching $M \subseteq G(S)$.

[9]Note that from the structure of $G(S)$, it has exactly m matchings, each of size $\frac{m-1}{2}$ (w.l.o.g, we assume m is odd).

Definition A.4 (Empirical Metric-Fairness loss)**.**

$$\mathcal{L}^F_{S,d,\gamma}(h) = \frac{2}{m-1} \cdot \sum_{(x,x')\in M(S)} \left[\ell_{\gamma,d}(h,(x,x'))\right] \tag{6}$$

Finally, we will say that a classifier h is (α,γ)-fair w.r.t $\mathcal{D}$ (respectively, S) and d if its respective metric-fairness loss is at most α.

Definition A.5 ((α,γ)-Metric-Fairness)**.** *A probabilistic classifier $h : \mathcal{X} \to [0,1]$ is said to be (α,γ)-fair w.r.t a metric d and $\mathcal{D}$ (respectively, S) if $\mathcal{L}^F_{\mathcal{D},d,\gamma}(h) \leq \alpha$ (respectively, $\mathcal{L}^F_{S,d,\gamma}(h) \leq \alpha$).*

Notation When S and d are clear from context, we will use the more succinct notation $\mathcal{L}^F_\gamma(h)$ for the true fairness loss and $\widehat{\mathcal{L}}^F_\gamma(h)$ for the empirical fairness loss. When dealing with a hypothesis class $\mathcal{H}$, we use $\mathcal{H}^{\alpha,\gamma} \subseteq \mathcal{H}$ to denote all the (α,γ)-fair hypotheses in $\mathcal{H}$ (w.r.t $\mathcal{D}$), and $\widehat{H}^{\alpha,\gamma} \subseteq \mathcal{H}$ for those which are (α,γ)-fair w.r.t S.

A.3 Approximate Metric Fairness: Interpretation

An α-fair classifier (for $\alpha > 0$) no longer holds any guarantee for any single individual. To interpret the guarantee it *does* give, we consider the following definition.

Definition A.6. *A probabilistic classifier $h : \mathcal{X} \to [0,1]$ is said to be $(\alpha_1,\alpha_2;\gamma)$ metric-fair w.r.t $d,\mathcal{D}$ if*

$$Pr_{x\sim\mathcal{D}}\left[Pr_{x'\sim\mathcal{D}}\left[\Lambda\left(h(x),h(x')\right) > d(x,x') + \gamma\right] > \alpha_2\right] \leq \alpha_1 \tag{7}$$

Definition A.6 is very similar to A.5 but it lends itself to a more intuitive interpretation of fairness for groups. Informally, we will say that an individual feels α_2-discriminated against by h if the proportion of individuals with whom his constraint is violated (think: individuals who are equally qualified to him but receive different treatment) exceeds α_2; now, (α_1,α_2)-fairness ensures that the proportion of individuals who find h to be α_2-discriminatory does not exceed α_1. Hence, this is a guarantee for groups: an (α_1,α_2)-fair classifier cannot cause an entire group of fractional mass α_1 to be discriminated against. The strength of this guarantee is that it holds for *any* such group (even for those formed ex-ante). In this sense, (α_1,α_2)-fairness represents a middle-ground between the strict notion of individual fairness and the loose notions of group-fairness.

Finally, we show that the two definitions are related: any α-fair classifier is also (α_1,α_2)-fair, for every α_1,α_2 for which $\alpha_1 \cdot \alpha_2 \geq \alpha$. This demonstrates that optimizing for accuracy under an α-fairness constraint is a flexible way of achieving interpretable fairness guarantees for a range of desired α_1,α_2 values.

Claim A.7. *For every $\alpha,\gamma \in (0,1)$, and $\alpha_1,\alpha_2 \in (0,1)$ for which $\alpha_1 \cdot \alpha_2 \geq \alpha$, if h is (α,γ)-fair then it is also $(\alpha_1,\alpha_2;\gamma)$-fair.*

Proof of Claim A.7. For simplicity, we define the following indicator function,

$$\mathbb{1}_{\gamma,h}(x,x') = \begin{cases} 1 & \Lambda\left(h(x),h(x')\right) > d(x,x') + \gamma \\ 0 & o.w \end{cases}$$

Let $\alpha, \gamma, \alpha_1, \alpha_2 \in (0,1)$ such that $\alpha_1 \cdot \alpha_2 \geq \alpha$, and assume that h is (α, γ)-fair w.r.t $\mathcal{D}$. Assume for contradiction that h is not $(\alpha_1, \alpha_2; \gamma)$-fair w.r.t $\mathcal{D}$. That means that

$$Pr_{x\sim\mathcal{D}}\left[Pr_{x'\sim\mathcal{D}}\left[\mathbb{1}_{\gamma,h}(x,x')=1\right] > \alpha_2\right] > \alpha_1$$

If we denote the subset of "α_2-discriminated" individuals as B,

$$B \triangleq \left\{x \in \mathcal{X} \,:\, Pr_{x'\sim\mathcal{D}}\left[\mathbb{1}_{\gamma,h}(x,x')=1\right] > \alpha_2\right\}$$

then the assumption is equivalent to $Pr_{x\sim\mathcal{D}}[x \in S] > \alpha_1$. We now obtain:

$$\begin{aligned}
\alpha &\geq Pr_{x,x'\sim\mathcal{D}}\left[\mathbb{1}_{\gamma,h}(x,x')=1\right] \\
&= Pr_{x\sim\mathcal{D}}[x\in S]\cdot Pr_{x'\sim\mathcal{D}}\left[\mathbb{1}_{\gamma,h}(x,x')=1 | x\in S\right] + \underbrace{Pr_{x\sim\mathcal{D}}[x\notin S]\cdot Pr_{x'\sim\mathcal{D}}\left[\mathbb{1}_{\gamma,h}(x,x')=1 | x\notin S\right]}_{\geq 0} \\
&\geq Pr_{x\sim\mathcal{D}}[x\in S]\cdot Pr_{x'\sim\mathcal{D}}\left[\mathbb{1}_{\gamma,h}(x,x')=1 | x\in S\right] \\
&> \alpha_1 \cdot \alpha_2
\end{aligned}$$

where the first transition is from the assumption that h is (α, γ)-fair, and the final transition is from the assumption that h is not $(\alpha_1, \alpha_2; \gamma)$-fair. We therefore have that $\alpha > \alpha_1 \cdot \alpha_2$, which contradicts our assumption. □

A.4 ℓ_1-Metric-Fairness

In this work we focus on approximate metric-fairness and the (α, γ) metric fairness loss (Definition A.2). We find that this notion provides appealing and *interpretable* protections from discrimination: as discussed above, for small enough α, γ, every sufficiently large group is protected from blatant discrimination (see Section A.3). However, in turning to design efficient metric-fair learning algorithms, working directly with this definition presents difficulties (see Section D). In particular, the "0/1" nature of the metric fairness loss means that the set $\widehat{H}^{\alpha,\gamma}$ is not a convex set. Trying to learn an empirically (α, γ) metric-fair that optimizes accuracy is a non-convex optimization problem, and it isn't clear how to optimize using convex-optimization tools.

In light of this difficulty, we introduce a different metric-fairness loss definition. It overcomes the non-convexity by replacing the bound on the expected *number* of fairness violations with a bound on the expected *sum* of the fairness violations.

Definition A.8 (ℓ_1 MF loss)**.** *For a metric d, the ℓ_1 metric-fairness loss on a pair $(x, x') \in \mathcal{X}$ is*

$$\ell_d^1\left(h, (x,x')\right) = \max\left(0, \left|h(x) - h(x')\right| - d(x,x')\right)$$

Similarly to the regular metric-fairness loss, the loss for a hypothesis h is the expected violation for a random pair according to $\mathcal{D}$, and the empirical loss replaces the expectation over $\mathcal{D}$ with the expectation over some matching $M \subseteq G(S)$.

Definition A.9 (τ ℓ_1-Metric-Fairness)**.** *A probabilistic classifier h is said to be τ ℓ_1-metric-fair w.r.t a metric d and w.r.t $\mathcal{D}$ (respectively, S) if its respective ℓ_1 MF loss is bounded by τ.*

When $S, \mathcal{D}, d$ are clear from context we use the notation $H^{\tau}_{\ell_1}$ (respectively, $\widehat{H}^{\tau}_{\ell_1}$) for the subset of hypotheses from H which are τ ℓ_1-MF w.r.t $\mathcal{D}$ (respectively, S). We use $H^{\alpha,\gamma}_{\ell_0}$ (previously $H^{\alpha,\gamma}$) to emphasize that fairness is calculated w.r.t the standard MF loss.

The main advantage of ℓ_1 metric-fairness is that it induces convex constraints and tractable optimization problems. We note that ℓ_1 metric-fairness also generalizes from a sample (indeed, in this case the fairness loss is Lipschitz and thus it's easier to prove generalization bounds). The main disadvantage of ℓ_1 metric-fairness is in interpreting the guarantee: it is less immediately obvious how bounding the sum of "fairness deviations" translates into protections for groups or for individuals. Nonetheless, we show that τ ℓ_1-metric-fairness implies $(\tau/\gamma, \gamma)$ approximate metric fairness for every $\gamma > 0$. Thus, when we optimize over the set of τ ℓ_1-MF predictors, we are guaranteed to output an approximately metric-fair predictor. Moreover, since approximate MF also implies ℓ_1-MF, any solution to the (ℓ_1) optimization problem will also be competitive with a certain class of approximately metric-fair classifiers.

The connection between approximate and ℓ_1 metric fairness is quantified in Lemma A.10 below. We note that there is a gap between these upper and lower bounds. This is the "price" we pay for relaxing from approximate to ℓ_1 metric fairness. We also note that in proving that ℓ_1-MF implies approximate metric-fairness, it is essential to use a non-zero additive γ violation term.

Lemma A.10. *For every sample S, matching $M(S)$ and every $\tau, \gamma \in (0,1)$, $\widehat{H}^{\tau-\gamma,\gamma}_{\ell_0} \subseteq \widehat{H}^{\tau}_{\ell_1} \subseteq \widehat{H}^{\frac{\tau}{\gamma},\gamma}_{\ell_0}$.*

Proof of Lemma A.10. We begin by defining the *induced violation vector* of a classifier $h \in H$. For a sample S, a matching $M \subseteq G(S)$ and a value $\gamma \in (0,1)$, the induced violation vector $\xi_\gamma(h) \in \{0,1\}^{|M|}$ is defined as:

$$[\xi_\gamma(h)]_i = \max\left\{0, \left|h(\mathbf{x}) - h(\mathbf{x}')\right| - d(\mathbf{x}, \mathbf{x}') - \gamma\right\}$$

where $(\mathbf{x}, \mathbf{x}')$ is the i-th edge in the matching M.

We first prove that $\widehat{H}^{\tau}_{\ell_1} \subseteq \widehat{H}^{\frac{\tau}{\gamma},\gamma}_{\ell_0}$. If $h \in \widehat{H}^{\tau}_{\ell_1}$, then this means that $\|\xi_0(h)\|_1 \leq \tau$. Assume for contradiction that for some $\gamma > 0$ we have $\|\xi_\gamma(h)\|_0 > \frac{\tau}{\gamma}$. But this implies that $\|\xi_0(h)\|_1 \geq \|\xi_\gamma(h)\|_0 \cdot \gamma > \frac{\tau}{\gamma} \cdot \gamma = \tau$, which is a contradiction. We therefore have that $\|\xi_\gamma(h)\|_0 \leq \frac{\tau}{\gamma}$ from which we conclude that $h \in \widehat{H}^{\frac{\tau}{\gamma},\gamma}_{\ell_0}$.

Next, we prove that $\widehat{H}^{\tau-\gamma,\gamma}_{\ell_0} \subseteq \widehat{H}^{\tau}_{\ell_1}$. To do so, we'll prove that $\widehat{H}^{\tau-\gamma,\gamma}_{\ell_0} \subseteq \widehat{H}^{\frac{\tau-\gamma}{1-\gamma},\gamma}_{\ell_0} \subseteq \widehat{H}^{\tau}_{\ell_1}$. Observe that for every $\tau, \gamma \in (0,1)$ we have that $\widehat{H}^{\tau-\gamma,\gamma}_{\ell_0} \subseteq \widehat{H}^{\frac{\tau-\gamma}{1-\gamma},\gamma}_{\ell_0}$ because $\frac{\tau-\gamma}{1-\gamma} \geq \tau - \gamma$. To see that $\widehat{H}^{\frac{\tau-\gamma}{1-\gamma},\gamma}_{\ell_0} \subseteq \widehat{H}^{\tau}_{\ell_1}$, recall that $h \in \widehat{H}^{\frac{\tau-\gamma}{1-\gamma},\gamma}_{\ell_0}$ implies that $\|\xi_\gamma(h)\|_0 \leq \frac{\tau-\gamma}{1-\gamma}$. Thus:

$$\|\xi_0(h)\|_1 \leq \frac{\tau-\gamma}{1-\gamma} \cdot 1 + \left(1 - \frac{\tau-\gamma}{1-\gamma}\right) \cdot \gamma = \tau$$

which implies $h \in \widehat{H}^{\tau}_{\ell_1}$, as required. □

A.5 Generalization

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? In this section, we work to develop similar generalization bounds for our metric-fairness loss function.

Proving that fairness can generalize well is a crucial component in our analysis - it effectively rules out the possibility of creating a "false facade" of fairness (i.e, a classifier that only appears fair on a sample, but is not fair w.r.t new individuals).

The generalization bounds will be based on proving uniform convergence of the empirical estimates (in our case, $\widehat{\mathcal{L}}^F_\gamma(h)$) to the fairness loss, simultaneously for every $h \in \mathcal{H}$, in terms of the Rademacher complexity of the hypotheses class $\mathcal{H}$. Rademacher complexity differs from celebrated VC-dimension complexity measure in three aspects: 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; third, it often results in tighter uniform convergence bounds (see, e.g, [KP02]).

Definition A.11 (Rademacher complexity)**.** *Let Z be an input space, $\mathcal{D}$ a distribution on Z, and $\mathcal{F}$ a real-valued function class defined on Z. The empirical Rademacher complexity of $\mathcal{F}$ with respect to a sample $S = \{z_1 \dots z_m\}$ is the following random variable:*

$$\widehat{\mathcal{R}}_m(\mathcal{F}) = \mathbb{E}_\sigma \left[\sup_{f \in \mathcal{F}} \left(\frac{1}{m} \sum_{i=1}^{m} \sigma_i f(z_i) \right) \right] \tag{8}$$

The expectation is taken over $\sigma = (\sigma_1, \dots, \sigma_m)$ where the σ_i's are independent uniformly random variables taking values in $\{\pm 1\}$. The Rademacher complexity of $\mathcal{F}$ is defined as the expectation of $\widehat{\mathcal{R}}_m(\mathcal{F})$ over all samples of size m:

$$\mathcal{R}_m(\mathcal{F}) = \mathbb{E}_{S \sim \mathcal{D}^m} \left[\widehat{\mathcal{R}}_m(\mathcal{F}) \right] \tag{9}$$

In our case, the fact that our fairness loss $\ell_{\gamma,d}$ is a 0-1 style-loss means that for infinite hypothesis classes, the generalization argument is involved and we incur an extra approximation term in γ.

Theorem A.12 (Rademacher-Based Uniform Convergence of the Metric-Fairness Loss)**.** *Let $\mathcal{H}$ be a hypotheses class with Rademacher complexity $R_m(\mathcal{H})$. For every $\delta, \gamma \in (0,1)$, every $G \geq 1$ and every $m \geq 0$ (w.l.o.g assume m is odd), with probability at least $1-\delta$ over an i.i.d sample $S \sim \mathcal{D}^m$, simultaneously for every $h \in \mathcal{H}$:*

$$\mathcal{L}^F_{\gamma + \frac{1}{G}}(h) - \Delta_m \leq \widehat{\mathcal{L}}^F_\gamma(h) \leq \mathcal{L}^F_{\gamma - \frac{1}{G}}(h) + \Delta_m \tag{10}$$

where $\Delta_m = 2G \cdot \left(4\widehat{R}_{\frac{m-1}{2}}(\mathcal{H}) + \frac{4+17\sqrt{\ln(4/\delta)}}{\sqrt{m-1}} \right)$.

Before proving the theorem, we note that an immediate corollary is that our metric-fairness loss is capable of generalizing well for any hypothesis class that has a small Rademacher complexity. In particular, we will be using the following result for the class of linear classifiers with norm bounded by C in a RHKS $\mathbb{V}$ whose inner products are implemented by a kernel K (i.e, exists a mapping $\psi_K : \mathcal{X} \to \mathbb{V}$ such that $\langle \psi(\mathbf{x}), \psi(\mathbf{x}') \rangle = K(\mathbf{x}, \mathbf{x}')$):

$$H_{\psi,C} \stackrel{\text{def}}{=} \{ \mathbf{x} \mapsto \langle \mathbf{v}, \psi(\mathbf{x}) \rangle : \quad \mathbf{v} \in \mathbb{V}, \, \|\mathbf{v}\| \leq C \}$$

Corollary A.13. *Let $H_{\psi,C}$ as above, for any $C \geq 0$ and kernel K. For every $\delta, \gamma \in (0,1)$, every $G \geq 1$ and every $m \geq 0$, w.p at least $1-\delta$, simultaneously for every $h \in H_{\psi,C}$*

$$\mathcal{L}^F_{\gamma + \frac{1}{G}}(h) - \Delta_m \leq \widehat{\mathcal{L}}^F_\gamma(h) \leq \mathcal{L}^F_{\gamma - \frac{1}{G}}(h) + \Delta_m \tag{11}$$

where $M = \sup K(\mathbf{x}, \mathbf{x}')$ *and* $\Delta_m = 2G \cdot \frac{4+4\sqrt{2}\sqrt{CM}+17\sqrt{\ln(4/\delta)}}{\sqrt{m-1}}$.

Proof of Corollary A.13. The proof follows from Theorem A.12 and the fact that the Rademacher complexity $R_m(H_{\psi,C})$ is bounded by $\sqrt{\frac{C \cdot M}{m}}$ (see [KST09]).

□

To set the stage for proving Theorem A.12, we state some useful properties of the Rademacher complexity notion, see [BM02].

Theorem A.14 (Two-sided Rademacher-based uniform convergence)**.** *Consider a set of functions* $\mathcal{F}$ *mapping* Z *to* $[0,1]$*. For every* $\delta > 0$*, with probability at least* $1-\delta$ *over a random draw of a sample* S *of size* m*, every* $f \in F$ *satisfies*

$$\left|\mathbb{E}_D\left[f(z)\right] - \widehat{\mathbb{E}}\left[f(z)\right]\right| \leq 2R_m(F) + \sqrt{\frac{\log\frac{2}{\delta}}{2m}} \tag{12}$$

$$\left|\mathbb{E}_D\left[f(z)\right] - \widehat{\mathbb{E}}\left[f(z)\right]\right| \leq 2\widehat{R}_m(F) + 3\sqrt{\frac{\log\frac{4}{\delta}}{2m}} \tag{13}$$

Lemma A.15. *Let* F *be a class of real functions. Then, for any sample* S *of size* m*:*

1. *For any function* h*,* $\widehat{R}_m(F+h) \leq \widehat{R}_m(F) + 2\sqrt{\frac{\widehat{\mathbb{E}}[h^2]}{m}}$.
2. *If* $A : \mathbb{R} \to \mathbb{R}$ *is* L*-Lipschitz and satisfies* $A(0) = 0$*, then* $\widehat{R}_m(A \circ F) \leq 2L \cdot \widehat{R}_m(F)$.
3. *For every* $\delta \in (0,1)$*, w.p at least* $1-\delta$ *over the choice of* S*,*

$$-2\sqrt{\frac{\log\frac{2}{\delta}}{2m}} \leq R_m(F) - \widehat{R}_m(F) \leq 2\sqrt{\frac{\log\frac{2}{\delta}}{2m}}$$

Proof of Theorem A.12.
Consider the input space $Z = \mathcal{X} \times \mathcal{X}$ and consider the functions $\mathcal{F} = \left\{\ell_\gamma^h : \ h \in \mathcal{H}\right\}$, where $\ell_\gamma^h : Z \to \mathbb{R}$ is the fairness loss defined as

$$\ell_\gamma^h(z) = \ell_h(x,x') = \mathbb{1}\left[\left|h(x) - h(x')\right| > d(x,x') + \gamma\right] \triangleq \begin{cases} 1 & |h(x) - h(x')| > d(x,x') + \gamma \\ 0 & o.w \end{cases}$$

For a given sample $S \sim \mathcal{D}^m$ (w.l.o.g, assume m is odd), let $M = \left\{z_1, \ldots, z_{\frac{m-1}{2}}\right\} \subseteq Z$ be any matching in the graph induced by S. Observe that M is indeed an i.i.d sample from $\mathcal{D} \times \mathcal{D}$ and recall that $|M| = \frac{m-1}{2} \triangleq \tilde{m}$. For the sake of simplicity, we hereby denote $\widehat{R}_{\tilde{m}}(F)$ as the empirical Rademacher complexity with respect to M induced by a random sample S.
Denote the threshold function at γ:

$$\sigma_\gamma(x) = \begin{cases} 1 & x > \gamma \\ 0 & x \leq \gamma \end{cases}$$

Hence,

$$\begin{aligned}\mathcal{F} &= \sigma_\gamma \circ \mathcal{G} \\ \mathcal{G} &= \left\{(x,x') \mapsto \left|h(x) - h(x')\right| - d(x,x')\right\}_{h\in\mathcal{H}}\end{aligned}$$

Observe that $\mathcal{G}$ can be further decomposed as

$$\mathcal{G} = \mathrm{abs} \circ \mathcal{H}' + f$$

where $\mathcal{H}' = \{(x,x') \mapsto h(x) - h(x')\}_{h\in\mathcal{H}} \triangleq \{g_h\}_{h\in\mathcal{H}}$, $\mathrm{abs}(\cdot)$ is the absolute value function (which is 1-Lipschitz), and $f = f(x,x') = -d(x,x')$.

Claim A.16. *$R_{\widetilde{m}}(\mathcal{H}') \le 2R_{\widetilde{m}}(\mathcal{H})$.*

Proof of Claim A.16. Denote $M(S) = \{z_1,\ldots,z_{\widetilde{m}}\}$,where $z_i = \left(x_i^1, x_i^2\right)$. By definition,

$$\begin{aligned}R_{\widetilde{m}}(\mathcal{H}') &= \mathbb{E}_{S\sim\mathcal{D}^m}\left[\widehat{R}_{\widetilde{m}}(\mathcal{H}')\right] \\ &= \mathbb{E}_{S,\sigma}\left[\sup_{g_h\in\mathcal{H}'}\left(\frac{1}{\widetilde{m}}\sum_{i=1}^{m}\sigma_i g_h(z_i)\right)\right] \\ &= \mathbb{E}_{S,\sigma}\left[\sup_{g_h\in\mathcal{H}'}\left(\frac{1}{\widetilde{m}}\sum_{i=1}^{m}\sigma_i\left(h(x_i^1) - h(x_i^2)\right)\right)\right] \\ &\le \mathbb{E}_{S,\sigma}\left[\sup_{g_h\in\mathcal{H}'}\left(\frac{1}{\widetilde{m}}\sum_{i=1}^{m}\sigma_i h(x_i^1)\right) + \sup_{g_h\in\mathcal{H}'}\left(\frac{1}{\widetilde{m}}\sum_{i=1}^{m}-\sigma_i h(x_i^2)\right)\right] \\ &= \mathbb{E}_{S,\sigma}\left[\sup_{h\in\mathcal{H}}\left(\frac{1}{\widetilde{m}}\sum_{i=1}^{m}\sigma_i h(x_i^1)\right) + \sup_{h\in\mathcal{H}}\left(\frac{1}{\widetilde{m}}\sum_{i=1}^{m}-\sigma_i h(x_i^2)\right)\right] \\ (\star) &= \mathbb{E}_{S,\sigma}\left[\sup_{h\in\mathcal{H}}\left(\frac{1}{\widetilde{m}}\sum_{i=1}^{m}\sigma_i h(x_i^1)\right)\right] + \mathbb{E}_{S,\sigma}\left[\sup_{h\in\mathcal{H}}\left(\frac{1}{\widetilde{m}}\sum_{i=1}^{m}\sigma_i h(x_i^2)\right)\right] \\ &= 2R_{\widetilde{m}}(\mathcal{H})\end{aligned}$$

where the transition marked by $(\star)$ is due to the fact that negating a Rademacher variable does not change its distribution. □

Claim A.17. *$R_{\widetilde{m}}(\mathcal{G}) \le 4R_{\widetilde{m}}(\mathcal{H}) + \frac{2}{\sqrt{\widetilde{m}}}$*

Proof of Claim A.17.

$$\begin{aligned}
R_{\widetilde{m}}(\mathcal{G}) &\equiv \mathbb{E}_{S\sim\mathcal{D}^m}\left[\widehat{R}_{\widetilde{m}}(\mathcal{G})\right] \\
&\equiv \mathbb{E}_{S\sim\mathcal{D}^m}\left[\widehat{R}_{\widetilde{m}}(\mathrm{abs}\circ\mathcal{H}'+f)\right] \\
\text{(Fact 1 in A.15)} &\leq \mathbb{E}_{S\sim\mathcal{D}^m}\left[\widehat{R}_{\widetilde{m}}(\mathrm{abs}\circ\mathcal{H}')+2\sqrt{\frac{\widehat{\mathbb{E}}[f^2]}{\widetilde{m}}}\right] \\
&\leq \mathbb{E}_{S\sim\mathcal{D}^m}\left[\widehat{R}_{\widetilde{m}}(\mathrm{abs}\circ\mathcal{H}')\right]+\frac{2}{\sqrt{\widetilde{m}}} \\
\text{(Fact 2 in A.15)} &\leq \mathbb{E}_{S\sim\mathcal{D}^m}\left[2\widehat{R}_{\widetilde{m}}(\mathcal{H}')\right]+\frac{2}{\sqrt{\widetilde{m}}} \\
&= 2R_{\widetilde{m}}(\mathcal{H}')+\frac{2}{\sqrt{\widetilde{m}}} \\
\text{(Claim A.16)} &\leq 4R_{\widetilde{m}}(\mathcal{H})+\frac{2}{\sqrt{\widetilde{m}}}
\end{aligned}$$

□

Now, had the threshold function σ_γ been Lipschitz, we could again use Fact 2 in Lemma A.15 to finish the proof. Unfortunately, σ_γ is not Lipschitz. We therefore instead approximate it using a piecewise linear function with Lipschitz constant G, which we denote with τ_γ^G:

$$\tau_\gamma^G(x)=\begin{cases}0 & x\leq\gamma\\ G\,(x-\gamma) & \gamma<x<\gamma+\frac{1}{G}\\ 1 & x\geq\gamma+\frac{1}{G}\end{cases}$$

Proposition A.18. *For every $G\geq 0$, every $\gamma\in(0,1)$ and every function h,*

$$\mathcal{L}_\sigma(h)\leq\mathcal{L}_\gamma^{\tau^G}(h)\leq\mathcal{L}^\sigma_{\gamma+\frac{1}{G}}(h)$$

Proof of Proposition A.18. The proof follows directly from the fact that from the construction of τ_γ^G, it holds that for every u, $\sigma_{\gamma+\frac{1}{G}}(u)\leq\tau_\gamma^G(u)\leq\sigma_\gamma(u)$. □

Now, letting

$$\widetilde{\mathcal{F}}=\tau_\gamma^G\circ\mathcal{G}$$

We can use the Lipschitzness of τ_γ^G to obtain

$$\begin{aligned}
R_{\widetilde{m}}(\widetilde{\mathcal{F}}) &= \mathbb{E}_{S\sim\mathcal{D}^m}\left[\widehat{R}_{\widetilde{m}}(\widetilde{\mathcal{F}})\right] \\
\text{(Fact 2 in Lemma A.15)} &\leq 2G\cdot\mathbb{E}_{S\sim\mathcal{D}^m}\left[\widehat{R}_{\widetilde{m}}(\mathcal{G})\right] \\
&= 2G\cdot R(\mathcal{G}) \\
\text{(Claim A.17)} &\leq 2G\left[4R_{\widetilde{m}}(\mathcal{H})+\frac{2}{\sqrt{\widetilde{m}}}\right] \\
&= 8G\cdot R_{\widetilde{m}}(\mathcal{H})+\frac{4G}{\sqrt{\widetilde{m}}}
\end{aligned}$$

Define the following loss notations:

$$\ell_\gamma^\sigma(h;x,x') = \sigma_\gamma\left(\left|h(x)-h(x')\right| - d(x,x')\right)$$

$$\ell_\gamma^{\tau^G}(h;x,x') = \tau_\gamma^G\left(\left|h(x)-h(x')\right| - d(x,x')\right)$$

and let $\mathcal{L}_\gamma^{\tau^G}(h)$ and $\widehat{\mathcal{L}}_\gamma^{\tau^G}(h)$ (respectively, $\mathcal{L}_\gamma^{\sigma}(h)$ and $\widehat{\mathcal{L}}_\gamma^{\sigma}(h)$) denote the average with respect to $\mathcal{D}$ and $M(S)$ of $\ell_\gamma^{\tau^G}(h;x,x')$ (respectively, $\ell_\gamma^{\sigma}(h;x,x')$).
By plugging $\widetilde{\mathcal{F}}$ into Theorem A.14 (Equation 12), we'd have that with probability at least $1-\frac{\delta}{2}$ over random draws of samples of size $\widetilde{m}$, every $h \in H$ satisfies

$$\begin{aligned}\left|\mathcal{L}_\gamma^{\tau^G}(h) - \widehat{\mathcal{L}}_\gamma^{\tau^G}(h)\right| &\le 2R_{\widetilde{m}}(\widetilde{\mathcal{F}}) + \sqrt{\frac{\ln(4/\delta)}{2\widetilde{m}}} \\ &\le 8G\cdot R_{\widetilde{m}}(\mathcal{H}) + \frac{4G}{\sqrt{\widetilde{m}}} + \sqrt{\frac{\ln(4/\delta)}{2\widetilde{m}}}\end{aligned}$$

Since we eventually want a data-dependent bound, we'll Fact 3 in Lemma A.15 to deduce that w.p at least $1-\frac{\delta}{2}$,

$$-2\sqrt{\frac{\ln(4/\delta)}{2\widetilde{m}}} \le R_{\widetilde{m}}(\mathcal{H}) - \widehat{R}_{\widetilde{m}}(\mathcal{H}) \le 2\sqrt{\frac{\ln(4/\delta)}{2\widetilde{m}}} \tag{14}$$

Now, note that

$$\begin{aligned}\widehat{\mathcal{L}}_\gamma^{F}(h) &\triangleq \widehat{\mathcal{L}}_\gamma^{\sigma}(h) \\ &\ge \widehat{\mathcal{L}}_\gamma^{\tau^G}(h) \\ &\ge \mathcal{L}_\gamma^{\tau^G}(h) - \left(8G\cdot R_{\widetilde{m}}(\mathcal{H}) + \frac{4G}{\sqrt{\widetilde{m}}} + \sqrt{\frac{\ln(2/\delta)}{2\widetilde{m}}}\right) \\ \left(\text{Equation (14), w.p } 1-\tfrac{\delta}{2}\right) &\ge \mathcal{L}_\gamma^{\tau^G}(h) - \left(8G\widehat{R}_{\widetilde{m}}(\mathcal{H}) + \frac{4G}{\sqrt{\widetilde{m}}} + 16G\cdot\sqrt{\frac{\ln(4/\delta)}{2\widetilde{m}}} + \sqrt{\frac{\ln(2/\delta)}{2\widetilde{m}}}\right) \\ &\ge \mathcal{L}_\gamma^{\tau^G}(h) - \left(8G\widehat{R}_{\widetilde{m}}(\mathcal{H}) + \frac{4G+17G\cdot\sqrt{\ln(4/\delta)}}{\sqrt{\widetilde{m}}}\right) \\ \left(\text{Proposition A.18}\right) &\ge \mathcal{L}_{\gamma+\frac{1}{G}}^{\sigma}(h) - G\cdot\left(8\widehat{R}_{\widetilde{m}}(\mathcal{H}) + \frac{4+17\sqrt{\ln(4/\delta)}}{\sqrt{\widetilde{m}}}\right) \\ &\triangleq \mathcal{L}_{\gamma+\frac{1}{G}}^{F}(h) - G\cdot\left(8\widehat{R}_{\widetilde{m}}(\mathcal{H}) + \frac{4+17\sqrt{\ln(4/\delta)}}{\sqrt{\widetilde{m}}}\right)\end{aligned}$$

Similarly,

$$\widehat{\mathcal{L}}_\gamma^{F}(h) \le \mathcal{L}_{\gamma-\frac{1}{G}}^{F}(h) + G\cdot\left(8\widehat{R}_{\widetilde{m}}(\mathcal{H}) + \frac{4+17\sqrt{\ln(4/\delta)}}{\sqrt{\widetilde{m}}}\right)$$

Combining both the previous inequalities and using the union bound, we obtain that for every $G \ge 0$, with probability at least $1-\delta$, simultaneously $\forall h \in \mathcal{H}$

$$\mathcal{L}_{\gamma+\frac{1}{G}}^{F}(h) - \Delta_{\widetilde{m}} \le \widehat{\mathcal{L}}_\gamma^{F}(h) \le \mathcal{L}_{\gamma-\frac{1}{G}}^{F}(h) + \Delta_{\widetilde{m}}$$

where $\Delta_m = 2G \cdot \left(4\widehat{R}_{\frac{m-1}{2}}(\mathcal{H}) + \frac{4+17\sqrt{\ln(4/\delta)}}{\sqrt{m-1}}\right)$, as required.

□

B (Fair and Accurate) Learning Objectives

B.1 Preliminaries

We will consider a learning problem as given by a 5-tuple $\left(\mathcal{X}, (\mathcal{Y}, \mathcal{Y}'), \mathcal{H}, \left(\mathcal{L}^U, \mathcal{L}^F\right)\right)$. In this notation, $\mathcal{X}$ is the set of instances; $\mathcal{Y}$ is the set of possible labels assigned to instances; $\mathcal{Y}' \supseteq \mathcal{Y}$ is the set of labels the learner is allowed to return; $\mathcal{H}$ is a fixed family of predictors $h : \mathcal{X} \to \mathcal{Y}'$ which we require the learner to compete with (in terms of the returning a classifier with loss comparable to the best loss in $\mathcal{H}$); $\mathcal{L}^U$ is a univariate ("utility") loss function used to measure the discrepancy between the predicted outputs and the true labels; and finally, $\mathcal{L}^F$ is a bivariate ("fairness") loss function. We denote $\mathcal{L}^U_{\mathcal{D}}(\mathcal{H}) = \min_{h\in\mathcal{H}}\left(\mathcal{L}^U_{\mathcal{D}}(h)\right)$.

For the remainder of this work, we will focus on the setting of of binary classification, i.e $\mathcal{Y} = \{\pm 1\}$. To accommodate probabilistic classifiers, we will consider $\mathcal{Y}' = [0,1]$, such that we interpret a classifier $h : \mathcal{X} \to \mathcal{Y}' = [0,1]$ as predicting the label 1 for an instance x with probability $h(x)$, and -1 otherwise. For utility we will mostly use the absolute loss: $\ell_{abs}(h,(x,y)) = |h(x) - y|$. For fairness we use the approximate metric-fairness loss defined in Section A.2.

B.2 PAC Learnability under a fairness constraint

From a learning perspective, we say that a learning algorithm is fair if w.h.p, it returns a sufficiently-fair classifier.

Definition B.1 ((α,γ)-fair learning algorithm)**.** *A learning algorithm A is (α,γ)-fair for a given metric d if there exists a function $m_A(\alpha,\gamma,\delta) : (0,1)^3 \to \mathbb{N}$ such that for any distribution $\mathcal{D}$ and for every $\delta \in (0,1)$, if S is an i.i.d sample from $\mathcal{D}$ of size $m \geq m_A(\alpha,\gamma,\delta)$, then w.p greater than $1-\delta$, the output of $A(S)$ is (α,γ)-fair w.r.t $\mathcal{D}, d$.*

Note that any learning algorithm A that completely disregards the sample and simply outputs a constant predictor (e.g, $h(x) \equiv 1$) satisfies the condition in Equation A.1 and is therefore a (0,0)-fair learning algorithm. In other words, fair learning is, in itself, trivial. It is the combination of a fairness and an accuracy objective that makes for an interesting task. In this work, we focus on the objective of maximizing utility subject to a constraint on the fairness loss. This is a natural formulation, because we think of fairness as a hard (often externally imposed) requirement that cannot necessarily be traded off for better accuracy. The objective of finding the most accurate sufficiently-fair hypothesis is summarized in the following definition. Crucially, both fairness and accuracy goals are stated w.r.t the unknown underlying distribution.

Definition B.2 (PACF Learnability: Probably-Approximately Correct and Fair)**.** *A hypothesis class $\mathcal{H}$ is PACF learnable with respect to a univariate utility loss function $\ell^U : \mathcal{H} \times Z \to \mathbb{R}_+$ and a bivariate fairness loss function $\ell^F_d : \mathcal{H} \times Z \times Z \to \mathbb{R}_+$ if there exists a function $m_{\mathcal{H}}(\alpha,\gamma,\delta,\epsilon,\epsilon_\alpha,\epsilon_\gamma) : (0,1)^6 \to \mathbb{N}$ (polynomial in $\alpha,\gamma,\epsilon,\epsilon_\alpha,\epsilon_\gamma,\log\frac{1}{\delta}$) and a learning algorithm with the following property: For every required fairness parameters $\alpha,\gamma \in (0,1)$, failure probability $\delta \in (0,1)$ and error*

parameters $\epsilon, \epsilon_\alpha, \epsilon_\gamma \in (0,1)$ *and for every distribution* $\mathcal{D}$ *over* $Z = \mathcal{X} \times \mathcal{Y}$*, when running the learning algorithm on* $m \geq m_{\mathcal{H}}(\alpha, \gamma, \delta, \epsilon, \epsilon_\alpha, \epsilon_\gamma)$ *i.i.d. examples generated by* $\mathcal{D}$*, the algorithm returns a hypothesis* h *such that, with probability of at least* $1-\delta$ *(over the choice of the* m *training examples),*

$$\mathcal{L}^F_{\mathcal{D},\gamma,d}(h) \leq \alpha \tag{15}$$

$$\mathcal{L}^U_{\mathcal{D}}(h) \leq \mathcal{L}^U_{\mathcal{D}}\left(\mathcal{H}^{\alpha-\epsilon_\alpha,\gamma-\epsilon_\gamma}\right) + \epsilon \tag{16}$$

where $\mathcal{L}^U_{\mathcal{D}}(h) = \mathbb{E}_{z\sim\mathcal{D}}\left[\ell^U(h,z)\right]$ *and* $\mathcal{L}^F_{\mathcal{D}}(h) = \mathbb{E}_{z,z'\sim\mathcal{D}}\left[\ell^F_\gamma(h,z,z')\right]$*.*

We consider the above definition as *strong* PACF learnability, and also define a relaxed notion, in which the learner must still output an hypothesis that is α fair, but in terms of accuracy is only required to compete with $\mathcal{H}^{\alpha'}$, for some $0 \leq \alpha' \leq \alpha$, that does not need to approach α as $m \to \infty$. We formalize this using a function $g : [0,1]^2 \to [0,1]^2$ that captures the degradation in the accuracy guarantee.

Definition B.3 ($g(\cdot)$-relaxed PACF Learnability)**.** *A hypothesis class* $\mathcal{H}$ *is* $g(\cdot)$*-relaxed PACF learnable with respect to a function* $g : [0,1]^2 \to [0,1]^2$*, a univariate utility loss function* $\ell^U : \mathcal{H} \times Z \to \mathbb{R}_+$ *and a bivariate fairness loss function* $\ell^F : \mathcal{H} \times Z \times Z \to \mathbb{R}_+$ *if there exists a function* $m_{\mathcal{H}}(\alpha, \gamma, \delta, \epsilon, \epsilon_\alpha, \epsilon_\gamma) : (0,1)^6 \to \mathbb{N}$ *(polynomial in* $\alpha, \gamma, \epsilon, \epsilon_\alpha, \epsilon_\gamma, \log\frac{1}{\delta}$*), and a learning algorithm with the following property: For every required fairness parameters* $\alpha, \gamma \in (0,1)$*, failure probability* $\delta \in (0,1)$ *and accuracy parameters* $\epsilon, \epsilon_\alpha, \epsilon_\gamma, \in (0,1)$ *and for every distribution* $\mathcal{D}$ *over* $Z = \mathcal{X} \times \mathcal{Y}$*, when running the learning algorithm on* $m \geq m_{\mathcal{H}}(\alpha, \gamma, \epsilon, \epsilon_\alpha, \epsilon_\gamma, \delta)$ *i.i.d. examples generated by* $\mathcal{D}$*, the algorithm returns a hypothesis* h *such that, with probability of at least* $1-\delta$ *(over the choice of the* m *training examples),*

$$\mathcal{L}^F_{\mathcal{D},\gamma,d}(h) \leq \alpha \tag{17}$$

$$\mathcal{L}^U_{\mathcal{D}}(h) \leq \mathcal{L}^U_{\mathcal{D}}\left(\mathcal{H}^{g(\alpha,\gamma)-(\epsilon_\alpha,\epsilon_\gamma)}\right) + \epsilon \tag{18}$$

C Information Theoretic PACF Learnability

Using the generalization result from Theorem A.12, we can now derive the sample complexity for *strong* PACF learnability in the information theoretic setting.

Theorem C.1 (Information theoretic PACF learnability)**.** *Suppose* $\mathcal{H}$ *is PAC learnable with sample complexity* $m_{PAC}(\epsilon, \delta)$*. Then it is PACF learnable with sample complexity*

$$m\left(\alpha, \gamma, \epsilon, \epsilon_\alpha, \epsilon_\gamma, \delta\right) = \max\left\{ m_{PAC}(\epsilon, \delta),\ \left(\frac{8 + 34\sqrt{\ln(4/\delta)}}{\epsilon_\alpha\epsilon_\gamma - 8\widehat{R}_{\frac{m-1}{2}}(\mathcal{H})}\right)^2 + 1 \right\}$$

Proof of Theorem C.1.
Let $\alpha, \gamma \in (0,1)$ required fairness parameters, $\delta \in (0,1)$ required failure probability and $\epsilon, \epsilon_\alpha, \epsilon_\gamma \in (0,1)$ error parameters. Let m, G be parameters to be determined later. Set $\widetilde{\gamma} = \gamma - \frac{1}{G}$ and $\widetilde{\alpha} = \alpha - \Delta_m$, for $\Delta_m = 2G \cdot \left(4\widehat{R}_{\frac{m-1}{2}}\left(\mathcal{H}\right) + \frac{4+17\sqrt{\ln(4/\delta)}}{\sqrt{m-1}}\right)$.
Let $h^\star$ denote the solution to the fairness-constrained ERM,

$$h^\star = \arg\min_{h\in\mathcal{H}} \widehat{\mathcal{L}}^U(h) \text{ subject to } h \in \widehat{H}^{\widetilde{\alpha},\widetilde{\gamma}}$$

Let $m_F = \left(\frac{8+34\sqrt{\ln(4/\delta)}}{\epsilon_\alpha \epsilon_\gamma - 8\widehat{R}_{\frac{m-1}{2}}(\mathcal{H})}\right)^2 + 1$ and $m_{PAC} = m_{PAC}(\epsilon,\delta)$.,

For fairness, we use Theorem A.12 to prove that for every m, G, w.p at least $1-\frac{\delta}{2}$ over the choice of $S \sim \mathcal{D}^m$, $h^\star$ is (α,γ)-fair w.r.t $\mathcal{D}$:

$$\begin{aligned}
\mathcal{L}^F_\gamma(h^\star) &\leq \widehat{\mathcal{L}}^F_{\gamma-\frac{1}{G}}(h^\star) + \Delta_m \\
&= \widehat{\mathcal{L}}^F_{\widetilde{\gamma}}(h^\star) + \Delta_m \\
\left(h^\star \in \widehat{H}^{\widetilde{\alpha},\widetilde{\gamma}}\right) &\leq \widetilde{\alpha} + \Delta_{\widetilde{m}} \\
&= \alpha
\end{aligned}$$

For accuracy, we note that the known equivalence of learnability and uniform convergence in the regression setting [ABDCBH97] implies that for $m \geq m_{PAC}$, w.p at least $1-\delta/2$, for every $h \in \mathcal{H}$, $\mathcal{L}^U(h) \leq \widehat{\mathcal{L}}^U(h) + \epsilon$. Using this we obtain that w.p at least $1-\delta/2$, for $m \geq \max\{m_{PAC}, m_{PACF}\}$:

$$\begin{aligned}
\mathcal{L}^U(h^\star) &\leq \widehat{\mathcal{L}}^U(h^\star) + \epsilon \\
&= \widehat{\mathcal{L}}^U(\widehat{H}^{\widetilde{\alpha},\widetilde{\gamma}}) + \epsilon \\
&= \widehat{\mathcal{L}}^U(\widehat{H}^{\alpha-\Delta_{\widetilde{m}},\gamma-\frac{1}{G}}) + \epsilon \\
&\leq \widehat{\mathcal{L}}^U(\widehat{H}^{\alpha-\epsilon_\alpha,\gamma-\epsilon_\gamma}) + \epsilon
\end{aligned}$$

We now use the union bound to conclude that setting $m \geq \max\{m_{PAC}, m_F\}$ yields that with probability at least $1-\delta$ over the choice of $S \sim \mathcal{D}^m$, h satisfies all the conditions in Definition B.2, and is therefore (exact) PACF learnable.

□

D Efficient relaxed-PACF Learnability of Linear Predictors

In this section, we present our main result, which is that the classes of (linear and logistic) predictors[10] are efficiently $g(\cdot)$-relaxed PACF-learnable, with $g(\alpha,\gamma) = (\alpha\cdot\gamma - \gamma^\star, \gamma^\star)$, for every $\gamma^\star \in (0,1)$. We begin with the proof for the class of linear classifiers, which demonstrates the idea of a relaxation that allows for efficient solving of the optimization problem associated with the relaxed PACF learnability requirement. We then proceed to the case of logistic predictors, which is more involved due to the non-convexity of the fairness objective induced by a logistic transfer function. We overcome this difficulty using improper learning.

[10]Since we interpret the output of the linear and logistic regression models are probabilities for a classification model, we refer to them as linear and logistic *classifiers*, rather than regressors.

D.1 Setting

For the remainder of this section, we consider the problem of efficiently PACF learning the problem $\left(\mathcal{X}, \left(\mathcal{Y}, \mathcal{Y}'\right), H, \left(\mathcal{L}^U, \mathcal{L}^F\right)\right)$, where: $\mathcal{X}$ is a compact subset of an RKHS, which w.l.o.g. will be taken to be the unit ball around the origin; $\mathcal{Y} = \{\pm 1\}$ and $\mathcal{Y}' = [0, 1]$, since we're interested in performing classification using probabilistic classifiers; the utility loss $\mathcal{L}^U$ is the absolute value loss function and $\mathcal{L}^F$ is the approximate metric-fairness loss w.r.t some known metric d. We define hypothesis classes H_{lin} and H_ϕ, corresponding to linear and logistic regression predictors, as follows. H_{lin} is the class of linear predictors, [11]

$$H_{lin} \stackrel{\text{def}}{=} \left\{ \mathbf{x} \mapsto \frac{1 + \langle \mathbf{w}, \mathbf{x} \rangle}{2} : \ \|\mathbf{w}\| \leq 1 \right\}, \tag{19}$$

And $H_{\phi,L}$ is the class of logistic predictors, formed by composing a linear function with a sigmoidal transfer function:

$$H_{\phi,L} \stackrel{\text{def}}{=} \{ \mathbf{x} \mapsto \phi_\ell \left(\langle w, \mathbf{x} \rangle \right) : \ \mathbf{w} \in \mathcal{X}, \ell \in [0, L] \} \tag{20}$$

where $\phi_\ell : [-1, 1] \to [0, 1]$ is a continuous and ℓ-Lipschitz sigmoid function, defined as $\phi_\ell(z) = \frac{1}{1+\exp(-4\ell \cdot z)}$.

D.2 Linear Regression

Theorem D.1. *For every $\gamma^\star \in (0, 1)$, H_{lin} is relaxed PACF learnable with $g(\alpha, \gamma) = (\alpha \cdot \gamma - \gamma^\star, \gamma^\star)$ and sample and time complexities of poly$(\frac{1}{\epsilon_\gamma}, \frac{1}{\epsilon_\alpha}, \frac{1}{\epsilon}, \log \frac{1}{\delta})$*

Proof of Theorem D.1.
The proof is structured as follows. First, we discuss why the immediate optimization problem associated with PACF learning H_{lin} is not efficiently solvable. We then show a convex problem whose solution can be shown to meet the fairness and competitiveness requirements in Definition B.3, but with respect to the sample. We conclude the proof by proving generalization of both the fairness and the competitiveness requirements.
For simplicity, we use $H = H_{lin}$ throughout the proof. Fix a sample S. The straight-forward approach to PACF learn $\left(\mathcal{X}, \left(\mathcal{Y}, \mathcal{Y}'\right), H, \left(\mathcal{L}^U, \mathcal{L}^F\right)\right)$ would be to search H for an hypothesis that minimizes the empirical risk, subject to being sufficiently fair w.r.t the pairs from $M(S)$ (a random matching induced by the sample S). This is equivalent to solving the following optimization problem, for some appropriate setting of $\alpha, \gamma \in (0, 1)$:

$$\min_{\|\mathbf{w}\| \leq 1} \sum_{i=1}^{m} |\langle \mathbf{w}, \mathbf{x}_i \rangle - y_i| \ \text{ subject to } \ \mathbf{w} \in \widehat{H}_{\ell_0}^{\alpha,\gamma} \tag{21}$$

where $\widehat{H}_{\ell_0}^{\alpha,\gamma}$ denotes the set of hypotheses in H that are (α, γ)-fair w.r.t $M(S)$. We use the ℓ_0 notation because:

$$\mathbf{w} \in \widehat{H}_{\ell_0}^{\alpha,\gamma} \iff \begin{cases} |\langle \mathbf{w}, \mathbf{x} \rangle - \langle \mathbf{w}, \mathbf{x}' \rangle| \leq d(\mathbf{x}, \mathbf{x}') + \gamma + \xi_e(\mathbf{w}) & \forall e = (\mathbf{x}, \mathbf{x}') \in M(S) \\ \xi_e(\mathbf{w}) \in \{0, 1\} & \forall e \in M(S) \\ \|\xi(\mathbf{w})\|_0 \leq \alpha \cdot |M(S)| \end{cases} \tag{22}$$

[11] The norm bound on $\mathbf{w}$ is so to ensure that $h(x) \in [-1, 1] = \mathcal{Y}$

here, $\xi_e(\mathbf{w}) \in 0,1$ is an indicator for whether the metric-fairness constraint on the sample pair $e = (\mathbf{x}, \mathbf{x}')$ is violated by more than an additive γ term or not, $\xi(\mathbf{w}) \in \{0,1\}^{|M(S)|}$ is the vector of violations on $M(S)$ induced by the linear classifier parametrized by $\mathbf{w}$, and $\|\xi(\mathbf{w})\|_0 \leq \alpha \cdot |M(S)|$ ensures that the overall fraction of such violations does not exceed α. We formalize the definition of a fairness-violation vector as follows:

Definition D.2. *For every $\mathbf{w} \in \mathcal{X}$ and $\gamma \in (0,1)$, the induced violation vector $\xi^\gamma(\mathbf{w}) \in \{0,1\}^{|M(S)|}$ is defined as*

$$[\xi_\gamma(\mathbf{w})]_i = d(\mathbf{x}, \mathbf{x}') + \gamma - \left|\langle \mathbf{w}, \mathbf{x}\rangle - \left\langle \mathbf{w}, \mathbf{x}'\right\rangle\right|$$

Unfortunately, the constraint $w \in \widehat{H}^{\alpha,\gamma}_{\ell_0}$ is not convex (since $\widehat{H}^{\alpha,\gamma}_{\ell_0}$ is not a convex set), and hence the optimization problem associated with PACF learning H (Equation 21) is not convex. We overcome this by replacing the ℓ_0 constraint on the norm of $\xi^\gamma(\mathbf{w})$ with an ℓ_1 constraint on the norm of $\xi^0(\mathbf{w})$ (note that we took here $\gamma = 0$). We denote this ℓ_1-norm-constrained version of $\widehat{H}^{\alpha,\gamma}$ as $\widehat{H}^{\alpha}_{\ell_1}$ (see Definition A.8 and the discussion that follows it):

$$\mathbf{w} \in \widehat{H}^{\alpha}_{\ell_1} \iff \begin{cases} |\langle \mathbf{w}, \mathbf{x}\rangle - \langle \mathbf{w}, \mathbf{x}'\rangle| \leq d(\mathbf{x}, \mathbf{x}') + \xi_e(\mathbf{w}) & \forall e = (\mathbf{x}, \mathbf{x}') \in M(S) \\ 0 \leq \xi_e(\mathbf{w}) \leq 1 & \forall e \in M(S) \\ \|\xi(\mathbf{w})\|_1 \leq \alpha \end{cases}$$

Lemma D.3. *For every $\alpha \in (0,1)$*

$$\min_{\|\mathbf{w}\| \leq 1} \sum_{i=1}^{m} |\langle \mathbf{w}, \mathbf{x}_i\rangle - y_i| \;\; \textit{subject to} \;\; \mathbf{w} \in \widehat{H}^{\alpha}_{\ell_1} \tag{23}$$

is a convex optimization problem.

Proof of Lemma D.3. It's sufficient to prove that $\widehat{H}^{\alpha}_{\ell_1}$ is a convex set. Let $\mathbf{w}_1, \mathbf{w}_2 \in \widehat{H}^{\alpha}_{\ell_1}$, we wish to show that $\forall t \in [0,1]$, $\mathbf{w} = t\mathbf{w}_1 + (1-t)\mathbf{w}_2$ is also in $\widehat{H}^{\alpha}_{\ell_1}$.
Let $e = (\mathbf{x}, \mathbf{x}') \in M(S)$. Then,

$$\begin{aligned} \left|\langle \mathbf{w}, \mathbf{x}\rangle - \left\langle \mathbf{w}, \mathbf{x}'\right\rangle\right| &= \left|\left\langle t\mathbf{w}_1 + (1-t)\mathbf{w}_2, \mathbf{x} - \mathbf{x}'\right\rangle\right| \\ &\leq t\left|\left\langle \mathbf{w}_1, \mathbf{x} - \mathbf{x}'\right\rangle\right| + (1-t)\left|\left\langle \mathbf{w}_2, \mathbf{x} - \mathbf{x}'\right\rangle\right| \\ \left(\mathbf{w}_1, \mathbf{w}_2 \in \widehat{H}^{\alpha}_{\ell_1}\right) &\leq t \cdot \left(d(\mathbf{x}, \mathbf{x}') + \xi_e(\mathbf{w}_1)\right) + (1-t) \cdot \left(d(\mathbf{x}, \mathbf{x}') + \xi_e(\mathbf{w}_2)\right) \\ &= d(\mathbf{x}, \mathbf{x}') + t \cdot \xi_e(\mathbf{w}_1) + (1-t) \cdot \xi_e(\mathbf{w}_2) \end{aligned}$$

this implies that for every $e \in M(S)$, $\xi_e(\mathbf{w}) = t \cdot \xi_e(\mathbf{w}_1) + (1-t) \cdot \xi_e(\mathbf{w}_2)$. Observe that since $t \in [0,1]$ and $0 \leq \xi_e(\mathbf{w}_1), \xi_e(\mathbf{w}_2) \leq 1$, we also have that $0 \leq \xi_e(\mathbf{w}) \leq 1$. Finally, the third constraint also holds, since we have that:

$$\begin{aligned}
\|\xi(\mathbf{w})\|_1 &= \sum_{e\in M(S)} \xi_e(\mathbf{w}) \\
&= \sum_{e\in M(S)} [t\cdot\xi_e(\mathbf{w}_1) + (1-t)\cdot\xi_e(\mathbf{w}_2)] \\
&= t\cdot\sum_{e\in M(S)} \xi_e(\mathbf{w}_1) + (1-t)\cdot\sum_{e\in M(S)} \xi_e(\mathbf{w}_2) \\
\left(\mathbf{w}_1,\mathbf{w}_2 \in \widehat{H}^{\alpha}_{\ell_1}\right) &\leq t\cdot\alpha\cdot|M(S)| + (1-t)\cdot\alpha\cdot|M(S)| \\
&= \alpha\cdot|M(S)|
\end{aligned}$$

Thus, we conclude that $\mathbf{w}\in\widehat{H}^{\alpha}_{\ell_1}$.

□

At this point, we have an efficient algorithm (the program from Equation 23) that can produce an hypothesis which is empirically ℓ_1-MF and is also competitive with the class of all such empirically ℓ_1-MF hypotheses. Lemma A.10 implies that this algorithm can be used to produce an hypothesis that is sufficiently *approximate metric-fair* and is also competitive in accuracy with a certain (smaller) class of approximately metric-fair classifiers (where in both cases, fairness is calculated w.r.t the sample).

To arrive at satisfying the conditions of relaxed metric-fair learning, it's left to prove that these guarantees can be extended to also hold for the underlying distribution $\mathcal{D}$. To this end, we employ generalization arguments for both the utility and fairness loss. For utility (Claim D.4), we use a standard Rademacher-based uniform convergence result, stated for the general setting of a class of linear predictors with bounded norm in a RHKS; for fairness (Claim D.5) we use the generalization result from Corollary A.13.

Claim D.4. *Let $H_{B,M}$ denote the class of linear predictors with norm bounded by B in a RHKS with a reproducing kernel $K:\mathcal{X}\times\mathcal{X}\to\mathbb{R}$ that satisfies $M=\sup K(\mathbf{x},\mathbf{x}')$. For every $\epsilon,\delta\in(0,1)$, setting $m\geq\frac{BM}{\epsilon^2}\left(2+9\sqrt{\ln(8/\delta)}\right)$ yields that w.p at least $1-\delta$ over an i.i.d sample $S\sim\mathcal{D}^m$, every $h\in H_{B,M}$ satisfies $\widehat{\mathcal{L}}^U(h)\leq\mathcal{L}^U(h)+\epsilon$.*

Claim D.5. *For every $\alpha,\gamma\in(0,1)$, $G\geq 1$ and $\delta\in(0,1)$, with probability at least $1-\frac{\delta}{2}$ over an i.i.d sample $S\sim\mathcal{D}^m$,*

$$H^{\alpha-\rho,\gamma-\frac{1}{G}}_{\ell_0}\subseteq\widehat{H}^{\alpha,\gamma}_{\ell_0}\subseteq H^{\alpha+\rho,\gamma+\frac{1}{G}}_{\ell_0}$$

where $\rho=2G\cdot\frac{4+4\sqrt{2}+17\sqrt{\ln(4/\delta)}}{\sqrt{m-1}}$.

Proof of Claim D.5. Let $G\geq 1$ and $\gamma\in(0,1)$. From Corollary A.13 (recall that in our setting, $C=M=1$), we obtain that w.p at least $1-\frac{\delta}{2}$ over an i.i.d sample $S\sim\mathcal{D}^m$,

$$\widehat{\mathcal{L}}^F_{\gamma+\frac{2}{G}}(\mathbf{w})-\rho\leq\mathcal{L}^F_{\gamma+\frac{1}{G}}(\mathbf{w})\leq\widehat{\mathcal{L}}^F_{\gamma}(\mathbf{w})+\rho$$

This implies that with all but $\frac{\delta}{2}$ probability, for every $\alpha\in(0,1)$, the following holds:

$$\mathbf{w}\in\widehat{H}^{\alpha,\gamma}_{\ell_0}\Rightarrow\widehat{\mathcal{L}}^F_{\gamma}(\mathbf{w})\leq\alpha\Rightarrow\mathcal{L}^F_{\gamma+\frac{1}{G}}(\mathbf{w})\leq\alpha+\rho\Rightarrow\mathbf{w}\in H^{\alpha+\rho,\gamma+\frac{1}{G}}_{\ell_0}$$

$$\mathbf{w}\in H^{\alpha-\rho,\gamma-\frac{1}{G}}_{\ell_0}\Rightarrow\mathcal{L}^F_{\gamma-\frac{1}{G}}(\mathbf{w})\leq\alpha-\rho\Rightarrow\widehat{\mathcal{L}}^F_{\gamma}(\mathbf{w})\leq\alpha\Rightarrow\mathbf{w}\in\widehat{H}^{\alpha,\gamma}_{\ell_0}$$

which implies $H_{\ell_0}^{\alpha-\rho,\gamma-\frac{1}{G}} \subseteq \widehat{H}_{\ell_0}^{\alpha,\gamma} \subseteq H_{\ell_0}^{\alpha+\rho,\gamma+\frac{1}{G}}$, as required. □

We are now prepared to prove relaxed PACF learnability of H. Let $\alpha, \gamma \in (0,1)$ be the required fairness parameters, $\delta \in (0,1)$ required failure probability and $\epsilon, \epsilon_\alpha, \epsilon_\gamma \in (0,1)$ error parameters. Let G, m be parameters to be determined later. Define $\widetilde{\alpha} = (\alpha - \rho) \cdot \widetilde{\gamma}$ and $\widetilde{\gamma} = \gamma - \frac{1}{G}$, and denote with $\mathbf{w}$ the solution to the program in Equation 23 using the parameter $\widetilde{\alpha}$.

Proposition D.6. *with probability at least* $1 - \frac{\delta}{2}$, $\mathbf{w}$ *is* (α, γ)*-fair w.r.t* $\mathcal{D}$.

Proof of Proposition D.6.

$$\begin{aligned}
\mathcal{L}_\gamma^F(\mathbf{w}) &\leq \mathcal{L}^F\left(\widehat{H}_{\ell_1}^{\widetilde{\alpha}}\right) \\
\text{(Lemma A.10, with } \widetilde{\gamma}) &\leq \mathcal{L}_\gamma^F\left(\widehat{H}_{\ell_0}^{\frac{\widetilde{\alpha}}{\widetilde{\gamma}},\widetilde{\gamma}}\right) \\
\left(\text{Claim D.5, w.p } 1 - \tfrac{\delta}{2}\right) &\leq \mathcal{L}_\gamma^F\left(H_{\ell_0}^{\frac{\widetilde{\alpha}}{\widetilde{\gamma}}+\rho,\widetilde{\gamma}+\frac{1}{G}}\right) \\
&= \mathcal{L}_\gamma^F\left(H_{\ell_0}^{\alpha,\gamma}\right) \\
&= \alpha
\end{aligned}$$

□

Proposition D.7. *Setting*

$$m \geq \max\left\{\left(\frac{\sqrt{2}+\sqrt{\ln(8/\delta)}}{\sqrt{2}\epsilon}\right)^2, \left(\frac{4\left(4+4\sqrt{2}+17\sqrt{\ln(4/\delta)}\right)}{(1-\alpha)\cdot\epsilon_\alpha\cdot\min\left\{\epsilon_\alpha, \frac{\epsilon_\gamma}{2}\right\}}\right)^2\right\}$$

ensures that with probability at least $1 - \frac{\delta}{2}$ *over an i.i.d sample* $S \sim \mathcal{D}^m$*, for every* $\gamma^\star$*,*

$$\mathcal{L}^U(\mathbf{w}) \leq \mathcal{L}^U\left(H^{\alpha\gamma-\gamma^\star-\epsilon_\alpha,\gamma^\star-\epsilon_\gamma}\right) + \epsilon$$

Proof of Proposition D.7.

First, let $m^1 = \left(\frac{2\cdot\left(\sqrt{2}+\sqrt{\ln(8/\delta)}\right)}{\sqrt{2}\epsilon}\right)^2$.

$$\begin{aligned}
\mathcal{L}^U(\mathbf{w}) &= \\
\left(\text{Claim D.4, w.p } 1 - \tfrac{\delta}{8} \text{ and } m \geq m^1\right) &\leq \widehat{\mathcal{L}}^U(\mathbf{w}) + \tfrac{\epsilon}{2} \\
&= \widehat{\mathcal{L}}^U(\widehat{H}_{\ell_1}^{\widetilde{\alpha}}) + \tfrac{\epsilon}{2} \\
\text{(Lemma A.10 with } \widetilde{\alpha},\gamma^\star) &\leq \widehat{\mathcal{L}}^U(\widehat{H}_{\ell_0}^{\widetilde{\alpha}-\gamma^\star,\gamma^\star}) + \tfrac{\epsilon}{2} \\
\left(\text{Claim D.5, w.p } 1 - \tfrac{\delta}{8}\right) &\leq \widehat{\mathcal{L}}^U(H_{\ell_0}^{\widetilde{\alpha}-\gamma^\star-\rho,\gamma^\star-\frac{1}{G}}) + \tfrac{\epsilon}{2} \\
\left(\text{Claim D.4 w.p } 1 - \tfrac{\delta}{4}, \text{ and } m \geq m^1\right) &\leq \mathcal{L}^U(H_{\ell_0}^{\widetilde{\alpha}-\gamma^\star-\rho,\gamma^\star-\frac{1}{G}}) + \tfrac{\epsilon}{2} + \tfrac{\epsilon}{2} \\
&= \mathcal{L}^U(H_{\ell_0}^{(\alpha-\rho)\left(\gamma-\frac{1}{G}\right)-\gamma^\star-\rho,\gamma^\star-\frac{1}{G}}) + \epsilon
\end{aligned}$$

It is left to show that there is some setting of G and a sufficiently large sample size m for which it holds that

$$\begin{cases} (\alpha-\rho)\left(\gamma-\frac{1}{G}\right)-\gamma^\star-\rho \geq \alpha\gamma-\gamma^\star-\epsilon_\alpha \\ \gamma^\star-\frac{1}{G} \geq \gamma^\star-\epsilon_\gamma \end{cases}$$

Denote $\epsilon' = \min\left\{\epsilon_\alpha, \frac{\epsilon_\gamma}{2}\right\}$ and set $G = \frac{1}{\epsilon'}$. The second equation is now satisfied, because $\frac{1}{G} = \epsilon' \leq \frac{\epsilon_\gamma}{2} \leq \epsilon_\gamma$. Plugging this into the first equation and simplifying, we obtain:

$$(\alpha-\rho)(\gamma-\epsilon')-\gamma^\star-\rho \geq \alpha\gamma-\gamma^\star-\epsilon_\alpha \iff \rho \leq \frac{\epsilon_\alpha-\alpha\cdot\epsilon'}{1+\gamma-\epsilon'}$$

For simplicity, denote $\rho = \frac{G\overbrace{\left(8+8\sqrt{2}+34\sqrt{\ln(4/\delta)}\right)}^{z(\delta)}}{\sqrt{m-1}} = \frac{z(\delta)}{\epsilon'\cdot\sqrt{m-1}}$ and note that $\frac{\epsilon_\alpha-\alpha\cdot\epsilon'}{1+\gamma-\epsilon'} \geq \frac{(1-\alpha)\cdot\epsilon_\alpha}{2}$. It therefore suffices to choose m such that:

$$\frac{z(\delta)}{\epsilon'\cdot\sqrt{m-1}} \leq \frac{(1-\alpha)\cdot\epsilon_\alpha}{2} \iff m \geq \left(\frac{4\left(4+4\sqrt{2}+17\sqrt{\ln(4/\delta)}\right)}{(1-\alpha)\cdot\epsilon_\alpha\cdot\min\left\{\epsilon_\alpha,\frac{\epsilon_\gamma}{2}\right\}}\right)^2+1$$

We can now use the union bound to conclude that for m stated in the statement of Proposition D.12, w.p at least $1-\frac{\delta}{2}$ over an i.i.d sample $S\sim\mathcal{D}^m$, it holds that

$$\mathcal{L}^U(\mathbf{w}) \leq \mathcal{L}^U(H_{\ell_0}^{(\alpha-\rho)\left(\gamma-\frac{1}{G}\right)-\gamma^\star-\rho,\gamma^\star-\frac{1}{G}})+\epsilon \leq \mathcal{L}^U(H_{\ell_0}^{\alpha\gamma-\gamma^\star-\epsilon_\alpha,\gamma^\star-\epsilon_\gamma})+\epsilon$$

as required.

□

To conclude the proof of Theorem D.1, we note that by Propositions D.6 and D.7, we have that w.p at least $1-\delta$ over an i.i.d sample $S\sim\mathcal{D}^m$ all the conditions in Definition B.3 are met for $g(\alpha,\gamma)=(\alpha\cdot\gamma-\gamma^\star,\gamma^\star)$ and m as in the definition of Proposition D.7. This implies that for this $g(\cdot)$, H is $g(\cdot)$-relaxed PACF learnable with sample complexity m and in time $poly(m)$, as required.

□

D.3 Logistic Regression

Theorem D.8. *For every constant $L>0$ and for every $\gamma^\star\in(0,1)$, $H_{\phi,L}$ is relaxed PACF learnable with $g(\alpha,\gamma)=(\alpha\cdot\gamma-\gamma^\star,\gamma^\star)$, with sample and time complexities that are polynomial in $(\frac{1}{\epsilon_\gamma},\frac{1}{\epsilon_\alpha},\frac{1}{\epsilon},\log\frac{1}{\delta})$.*

Proof of Theorem D.8.
The optimization problem that we used to efficiently-learn H_{lin} in the proof of Theorem D.1 (Equation 23) is no longer convex in the case of H_ϕ, due to the addition of the sigmoidal transfer function ϕ:

$$\begin{aligned} \underset{\mathbf{w},\xi(\mathbf{w})}{\text{minimize}} \quad & \sum_{i=1}^m |\langle\mathbf{w},\mathbf{x}_i\rangle-y_i| \\ \text{subject to} \quad & \left|\phi\left(\langle\mathbf{w},\mathbf{x}\rangle\right)-\phi\left(\left\langle\mathbf{w},\mathbf{x}'\right\rangle\right)\right| \leq d(\mathbf{x},\mathbf{x}')+\xi_e(\mathbf{w}), \quad \forall e=(\mathbf{x},\mathbf{x}')\in M(S) \\ & 0\leq\xi_e(\mathbf{w})\leq 1 \qquad\qquad \forall e\in M(S) \\ & \|\xi(\mathbf{w})\|_1\leq\alpha \end{aligned}$$

This implies that convex optimization techniques can't be used to solve the above formulation efficiently. We overcome this using improper learning. Namely, instead of attempting to directly PACF learn H_ϕ, we PACF learn some other large class of hypotheses, where the fairness constraint *is* convex. We proceed to define this class, previously introduced in [SSSS11]. Define the following kernel function $K : \mathcal{X} \times \mathcal{X} \to \mathbb{R}$:

$$K\left(\mathbf{x}, \mathbf{x}'\right) \stackrel{\text{def}}{=} \frac{1}{1 - \frac{1}{2}\langle \mathbf{x}, \mathbf{x}' \rangle}$$

Since this is a positive definite kernel function, there exists some mapping $\psi : \mathcal{X} \to \mathbb{V}$, where $\mathbb{V}$ is a RHKS with $\langle \psi(\boldsymbol{x}), \psi(\boldsymbol{x}') \rangle = K(\boldsymbol{x}, \boldsymbol{x}')$. Denote $\mathbb{V}_B = \left\{ \mathbf{v} \in \mathbb{V} \mid \|\mathbf{v}\|^2 \leq B \right\} \subseteq \mathbb{V}$ and consider the class of linear classifiers parametrized by vectors in $\mathbb{V}_B$:

$$H_B \stackrel{\text{def}}{=} \{\mathbf{x} \mapsto \langle v, \psi(\mathbf{x}) \rangle : \quad \mathbf{v} \in \mathbb{V}_B\}$$

We show that the optimization problem associated with PACF learning H_B can be solved efficiently, and that its solution satisfies the requirements for relaxed PACF learning of H_ϕ.

Claim D.9. *H_B is efficiently relaxed-PACF learnable.*

Proof of Claim D.9. PACF learning H_B requires solving the following program:

$$\begin{aligned}
\underset{\xi(\mathbf{v}),\, \mathbf{v} \in \mathbb{V}_B}{\text{minimize}} \quad & \sum_{i=1}^{m} |\langle \mathbf{v}, \psi(\mathbf{x}_i) \rangle - y_i| \\
\text{subject to} \quad & \left| \langle \mathbf{v}, \psi(\mathbf{x}) \rangle - \left\langle \mathbf{v}, \psi(\mathbf{x}') \right\rangle \right| \leq d(\mathbf{x}, \mathbf{x}') + \xi_e(\mathbf{v}), \quad \forall e = (\mathbf{x}, \mathbf{x}') \in M(S) \\
& 0 \leq \xi_e(\mathbf{v}) \leq 1 \qquad \forall e \in M(S) \\
& \|\xi(\mathbf{v})\|_1 \leq \alpha
\end{aligned}$$

Or its equivalent re-parametrization[12], which we refer to as Program 2:

$$\begin{aligned}
\underset{\mathbf{v} \in \mathbb{V}\,, \xi(\mathbf{v})}{\text{minimize}} \quad & \sum_{i=1}^{m} |\langle \mathbf{v}, \psi(\mathbf{x}_i) \rangle - y_i| + \lambda_B \|\mathbf{v}\|_2^2 \\
\text{subject to} \quad & \left| \langle \mathbf{v}, \psi(\mathbf{x}) \rangle - \left\langle \mathbf{v}, \psi(\mathbf{x}') \right\rangle \right| \leq d(\mathbf{x}, \mathbf{x}') + \xi_e(\mathbf{v}), \quad \forall e = (\mathbf{x}, \mathbf{x}') \in M(S) \\
& 0 \leq \xi_e(\mathbf{v}) \leq 1 \qquad \forall e \in M(S) \\
& \|\xi(\mathbf{v})\|_1 \leq \alpha
\end{aligned}$$

Note that while the constraints are now linear, the mapping ψ induced by the kernel K is possibly infinite dimensional, and hence it is not obvious that this optimization problem can be solved efficiently. Fortunately, we can use The Representer Theorem [Wah90] to reduce Program 2 to a finite dimensional optimization problem.

Lemma D.10. *There exists a solution to Program 2 of the form $\mathbf{v}^* = \sum_{\ell=1}^{m} \beta_\ell \psi(\mathbf{x}_\ell)$, where for every $\ell \in [m]$, $\beta_\ell \in [0, 1]$.*

[12]The two formulations are equivalent in the sense that any solution to the first program (for some desired B) can be obtained by solving Program 2 (for an appropriate value λ_B), see Theorem 1 in [ORA16].

Proof of Lemma D.10.
According to the Representer Theorem, any problem of the form

$$\min_{\mathbf{w}} \left\{ f\left(\langle \mathbf{w}, \psi(\mathbf{x}_1)\rangle, \ldots, \langle \mathbf{w}, \psi(\mathbf{x}_m)\rangle\right) + R\left(\|\mathbf{w}\|\right)\right\}$$

for an arbitrary function $f : \mathbb{R}^m \to \mathbb{R}$ and a monotonically nondecreasing function $R : \mathbb{R}_+ \to \mathbb{R}$, has a solution of the form $\mathbf{v}^* = \sum_{i=1}^m \beta_i \psi(\mathbf{x}_i)$. Hence, it's sufficient to prove that the optimization problem in Program 2 is an instance of the problem $\min_{\mathbf{w}} \left\{ f\left(\langle \mathbf{w}, \psi(\mathbf{x}_1)\rangle, \ldots, \langle \mathbf{w}, \psi(\mathbf{x}_m)\rangle\right) + R\left(\|\mathbf{w}\|\right)\right\}$, for some "permissible" choice of f and R. We show this by taking $a_i = \langle \mathbf{w}, \psi(\mathbf{x}_i)\rangle$ and defining $R(a) = \lambda a^2$ and

$$f(a_1, \ldots, a_m) = \begin{cases} \frac{1}{m}\sum_{i=1}^m |a_i - y_i| & \text{if } \exists \xi \in [0,1]^{\mathbb{R}^{|M(S)|}} \text{ s.t } \|\xi\|_1 \le \alpha \\ & \text{and } \forall e = (x_i, x_j) \in M(S),\ \ |a_i - a_j| \le d(x_i, x_j) + \xi_e \\ \infty & \text{otherwise} \end{cases}$$

□

Lemma D.10 implies we can instead optimize over $\beta_1 \ldots \beta_m$. This yields the following optimization problem:

$$\begin{aligned} \underset{\xi,\beta}{\text{minimize}} \quad & \sum_{i=1}^m \left| \sum_{\ell=1}^m \beta_\ell K(\mathbf{x}_\ell, \mathbf{x}_i) - y_i \right| + \lambda_B \sum_{i,j=1}^m \beta_i \beta_j K(\mathbf{x}_i, \mathbf{x}_j) \\ \text{subject to} \quad & \left| \sum_{\ell=1}^m \beta_\ell K(\mathbf{x}_\ell, \mathbf{x}) - \sum_{\ell=1}^m \beta_\ell K(\mathbf{x}_\ell, \mathbf{x}') \right| \le d(\mathbf{x}, \mathbf{x}') + \xi_e(\beta), \quad \forall e = (\mathbf{x}, \mathbf{x}') \in M(S) \\ & 0 \le \xi_e(\beta) \le 1 \qquad \forall e \in M(S) \\ & \|\xi(\beta)\|_1 \le \alpha \end{aligned}$$

We can now conclude the proof of Claim D.9, since this is a convex optimization problem in $O(m)$ variables and therefore can be solved in time $poly(m, \log B, \log \frac{1}{\alpha})$ using standard optimization tools. □

For convenience, we hereby denote the solution to the above program when instantiated with parameters B and α as $\mathbf{w}_{B,\alpha}$. We define a learning algorithm A that given parameters $B^\star, \alpha^\star$ returns $\mathbf{w}_{B^\star,\alpha^\star}$.

Now, let $\alpha, \gamma \in (0,1)$ be the required fairness parameters, $\delta \in (0,1)$ required failure probability, $\epsilon, \epsilon_\alpha, \epsilon_\gamma \in (0,1)$ error parameters and $\gamma^\star \in (0,1)$. Recall (Definition B.3) that in order to prove that H_ϕ is $g(\cdot)$-relaxed PACF learnable for $g(\alpha, \gamma) = (\alpha \cdot \gamma - \gamma^\star, \gamma^\star)$, we must prove that there is some setting of $B^\star, \alpha^\star$ and a sample size $m \le poly(\frac{1}{\alpha}, \frac{1}{\gamma}, \frac{1}{\epsilon}, \frac{1}{\epsilon_\alpha}, \frac{1}{\epsilon_\gamma}, \log \frac{1}{\delta})$, for which running A on a sample $S \sim \mathcal{D}^m$ yields $\mathbf{w} = A(B^\star, \alpha^\star)$ that satisfies:

$$\begin{cases} \mathcal{L}_\gamma^F(\mathbf{w}) \le \alpha & \text{(fairness)} \\ \mathcal{L}^U(\mathbf{w}) \le \mathcal{L}^U\left(H_\phi^{\alpha\gamma - \gamma^\star - \epsilon_\alpha, \gamma^\star - \epsilon_\gamma}\right) & \text{(accuracy)} \end{cases}$$

Let G, B parameters to be defined later, and define:

$$\rho = 2G \cdot \frac{4 + 8\sqrt{B} + 17\sqrt{\ln(4/\delta)}}{\sqrt{m-1}}$$
$$\widetilde{\gamma} = \gamma - \frac{1}{G}$$
$$\widetilde{\alpha} = (\alpha - \rho) \cdot \widetilde{\gamma}$$

Proposition D.11. *For every $B > 0$, w.p at least $1 - \frac{\delta}{2}$ over a sample $S \sim \mathcal{D}^m$, $\mathbf{w} = A(B, \widetilde{\alpha})$ is (α, γ)-fair w.r.t $\mathcal{D}$.*

Proof of Proposition D.11. The argument follows directly from the same arguments in the proof of Proposition D.6 in Theorem D.1, since H_B is a class of linear classifiers. □

Proposition D.12. *Setting*

$$\epsilon^\star = \min\left\{\epsilon, \epsilon_\alpha, \tfrac{\epsilon_\gamma}{2}\right\}$$
$$B^\star = 6L^4 + \exp\left(9L \log\left(\frac{4L}{\epsilon^\star}\right) + 5\right)$$
$$m \geq m^\star = \max\left\{\frac{2B^\star \cdot \left(2 + 9\sqrt{\ln(8/\delta)}\right)}{\epsilon^2}, \quad \left(\frac{4\left(4 + 8\sqrt{B^\star} + 17\sqrt{\ln(4/\delta)}\right)}{(1-\alpha)\cdot\epsilon_\alpha \cdot \epsilon^\star}\right)^2 + 1\right\}$$

ensures that for every $\gamma^\star \in (0,1)$ and every $\delta, \alpha, \gamma, \epsilon_\alpha, \epsilon_\gamma, \epsilon \in (0,1)$, w.p at least $1 - \frac{\delta}{2}$ over the choice of a sample $S \sim \mathcal{D}^m$, $\mathbf{w} = A(B^\star, \alpha^\star)$ satisfies

$$\mathcal{L}^U(\mathbf{w}) \leq \mathcal{L}^U\left(H_\phi^{\alpha\gamma - \gamma^\star - \epsilon_\alpha, \gamma^\star - \epsilon_\gamma}\right) + \epsilon$$

Proof of Proposition D.12.
The proof of Proposition D.12 will be based on the fact that for sufficiently large B, $H_\phi^{\alpha,\gamma}$ is approximately contained (in terms of accuracy) in $H_B^{\alpha,\gamma}$. This is formalized in the following claim.

Claim D.13. *For every $\alpha, \gamma, \epsilon \in (0,1)$ and $L \geq 3$, setting $B = 6L^4 + \exp\left(9L \log\left(\frac{4L}{\epsilon}\right) + 5\right)$ ensures that $\mathcal{L}^U(H_B^{\alpha,\gamma+\epsilon}) \leq \mathcal{L}^U(H_\phi^{\alpha,\gamma}) + \frac{\epsilon}{2}$.*

Proof of Claim D.13.
From Lemma 2.5 in [SSSS11], for every $\epsilon' \in (0,1)$, setting $B = 6L^4 + \exp\left(9L \log\left(\frac{2L}{\epsilon}\right) + 5\right)$ yields that for any $h \in H_\phi$ there exists $h_B \in H_B$ such that

$$\forall \mathbf{x} \in \mathcal{X}, \ \ |h_B(\mathbf{x}) - h(\mathbf{x})| \leq \epsilon'$$

Let $\alpha, \gamma, \epsilon \in (0,1)$. Now, by applying the above with $\epsilon' = \frac{1}{2}\epsilon$, we conclude that for every $h \in H_\phi^{\alpha,\gamma}$ there exists $h_B \in H_B$ such that $\forall \mathbf{x} \in \mathcal{X}, \ \ |h_B(\mathbf{x}) - h(\mathbf{x})| \leq \epsilon'$. This implies that $\mathcal{L}^U(h_B) \leq \mathcal{L}^U(h) + \epsilon'$.
To conclude the proof, it suffices to prove that $h_B \in H_B^{\alpha,\gamma+2\epsilon'}$.

Indeed:

$$\begin{aligned}
\mathcal{L}^F_{\gamma+2\epsilon'}(h_B) &= \|\xi^{\gamma+2\epsilon'}(h_B)\|_0 \\
&= \sum_{(x,x')\in M(S)} 1\left[\left|h_B(x)-h_B(x')\right| > d(x,x')+\gamma+2\epsilon'\right] \\
&= \sum_{(x,x')\in M(S)} 1\left[\left|h_B(x)-h_B(x')\right| > d(x,x')+\gamma+2\epsilon'\right] \\
\star \leq & \sum_{(x,x')\in M(S)} 1\left[\left|h(x)-h(x')\right| + \cancel{2\epsilon'} > d(x,x')+\gamma+\cancel{2\epsilon'}\right] \\
&= \sum_{(x,x')\in M(S)} 1\left[\left|h(x)-h(x')\right| > d(x,x')+\gamma\right] \\
&= \|\xi^{\gamma}(h)\|_0 \\
\left(h\in H^{\alpha,\gamma}_{\phi}\right) &\leq \alpha
\end{aligned}$$

where in $\star$ we used the fact that from the triangle inequality,

$$\forall \mathbf{x}\in\mathcal{X},\ |h_B(\mathbf{x})-h(\mathbf{x})|\leq\epsilon' \Longrightarrow \forall \mathbf{x},\mathbf{x}'\in\mathcal{X},\ \left|h_B(\mathbf{x})-h_B(\mathbf{x}')\right| \leq \left|h(\mathbf{x})-h(\mathbf{x}')\right| + 2\epsilon'$$

□

We can now prove Proposition D.12. Let $\gamma^\star\in(0,1)$ and $\mathbf{w}=A(B^\star,\alpha^\star)$, for $B^\star$ and $\alpha^\star$ as in the proposition's statement.
First, let $m^1=\frac{2B}{\epsilon^2}\left(2+9\sqrt{\ln(8/\delta)}\right)$ and $\epsilon'=\min\left\{\epsilon,\epsilon_\alpha,\frac{\epsilon_\gamma}{2}\right\}$. Now:

$$\begin{aligned}
&\mathcal{L}^U(\mathbf{w}) \\
\left(\text{Claim D.4, w.p } 1-\tfrac{\delta}{4} \text{ and } m\geq m^1\right) &\leq \widehat{\mathcal{L}}^U(\mathbf{w})+\frac{\epsilon}{4} \\
&= \widehat{\mathcal{L}}^U\left(\widehat{H}^{\widetilde{\alpha}}_{B,\ell_1}\right)+\frac{\epsilon}{4} \\
\left(\text{Lemma A.10 with } \widetilde{\alpha},\gamma^\star\right) &\leq \widehat{\mathcal{L}}^U\left(\widehat{H}^{\widetilde{\alpha}-\gamma^\star,\gamma^\star}_{B,\ell_0}\right)+\frac{\epsilon}{4} \\
&\triangleq \widehat{\mathcal{L}}^U\left(\widehat{H}^{\widetilde{\alpha}-\gamma^\star,\gamma^\star}_{B}\right)+\frac{\epsilon}{4} \\
\left(\text{Claim D.5, w.p } 1-\tfrac{\delta}{2}\right) &\leq \widehat{\mathcal{L}}^U\left(H^{\widetilde{\alpha}-\gamma^\star-\rho,\gamma^\star-\frac{1}{G}}_{B}\right)+\frac{\epsilon}{4} \\
\left(\text{Claim D.4, w.p } 1-\tfrac{\delta}{4} \text{ and } m\geq m^1\right) &\leq \mathcal{L}^U\left(H^{\widetilde{\alpha}-\gamma^\star-\rho,\gamma^\star-\frac{1}{G}}_{B}\right)+\frac{\epsilon}{2} \\
\left(\text{Claim D.13 for } B\geq B^\star \text{ and } \epsilon'\right) &\leq \mathcal{L}^U\left(H^{\widetilde{\alpha}-\gamma^\star-\rho,\,\gamma^\star-\frac{1}{G}-\epsilon'}_{\phi}\right)+\frac{\epsilon}{2}+\frac{\epsilon'}{2} \\
&\leq \mathcal{L}^U\left(H^{(\alpha-\rho)\cdot\left(\gamma-\frac{1}{G}\right)-\gamma^\star-\rho,\,\gamma^\star-\frac{1}{G}-\epsilon'}_{\phi}\right)+\epsilon
\end{aligned}$$

It is left to show that there is some setting of G and a sufficiently large sample size m for which it also holds that

$$\begin{cases}
(\alpha-\rho)\left(\gamma-\frac{1}{G}\right)-\gamma^\star-\rho \geq \alpha\gamma-\gamma^\star-\epsilon_\alpha \\
\gamma^\star-\frac{1}{G}-\epsilon' \geq \gamma^\star-\epsilon_\gamma
\end{cases}$$

We set $G = \frac{1}{\epsilon'}$. The second inequality is satisfied, because $\frac{1}{G} = \epsilon' = 2\epsilon' - \epsilon' \leq \epsilon_\gamma - \epsilon'$. Plugging this into the first equation and simplifying, we obtain:

$$(\alpha - \rho)(\gamma - \epsilon') - \gamma^\star - \rho \geq \alpha\gamma - \gamma^\star - \epsilon_\alpha \iff \rho \leq \frac{\epsilon_\alpha - \alpha \cdot \epsilon'}{1 + \gamma - \epsilon'}$$

which is satisfied for any $m \geq \left(\frac{4\left(4+8\sqrt{B}+17\sqrt{\ln(4/\delta)}\right)}{(1-\alpha)\cdot\epsilon_\alpha \cdot \min\left\{\epsilon, \epsilon_\alpha, \frac{\epsilon_\gamma}{2}\right\}} \right)^2 + 1$.

We can now use the union bound to conclude that for the sample size m specified in the Proposition's statement, w.p at least $1 - \frac{\delta}{2}$ over an i.i.d sample $S \sim \mathcal{D}^m$, it holds that

$$\mathcal{L}^U(\mathbf{w}) \leq \mathcal{L}^U(H_\phi^{(\alpha-\rho)\left(\gamma-\frac{1}{G}\right)-\gamma^\star-\rho, \gamma^\star-\frac{1}{G}-\epsilon}) + \epsilon \leq \mathcal{L}^U(H_{\ell_0}^{\alpha\gamma-\gamma^\star-\epsilon_\alpha, \gamma^\star-\epsilon_\gamma}) + \epsilon$$

as required. □

To conclude the proof of Theorem D.8, we note that by Propositions D.11 and D.12, we have that w.p at least $1 - \delta$ over an i.i.d sample $S \sim \mathcal{D}^m$ all the conditions in Definition B.3 are met for $g(\alpha, \gamma) = (\alpha \cdot \gamma - \gamma^\star, \gamma^\star)$ and $m \geq m^\star$ as in the definition of Proposition D.12. This implies that for this $g(\cdot)$, $H_{\phi,L}$ is $g(\cdot)$-relaxed PACF learnable with sample complexity m and in time $poly(m)$, as required. □

E Intractability of Perfectly Metric-Fair Learning

We show that perfect metric-fairness can make simple learning tasks computationally intractable. Towards this, we exhibit a simple learning task that becomes intractable under a perfect metric-fairness constraint (for a particular metric). We note that this task *can* be solved in polynomial time under the approximate metric-fairness relaxation. See the discussion in Section 6.

Theorem E.1 (Intractability of perfectly metric-fair learning)**.** *Assume that one-way functions exist and let $\mathcal{X}$ be the unit ball in $\mathcal{R}^n$. There exist: (i) a fixed distribution $\mathcal{D}$ over $(\mathcal{X} \times \pm 1)$, (ii) a linear classifier $w : \mathcal{X} \to \pm 1$ that perfectly labels $\mathcal{D}$ ($err_{\mathcal{D}}(w) = 0$), and (iii) two efficiently sampleable distributions U and V on metrics $d : \mathcal{X}^2 \to [0, 1]$, where:*

1. *For* every *metric d drawn from U,* every *perfectly metric-fair classifier $h : \mathcal{X} \to [0, 1]$ (in any hypothesis class) has error $err_{\mathcal{D}}(h) = 1/2$.*
2. *With overwhelming probability over a metric d drawn from V, the (linear) classifier w is perfectly metric-fair.*
3. *For every polynomial-time learning algorithm $\mathcal{A}$ there exists a constant $\alpha \in [0, 1]$ such that one of the following two conditions holds:*
   (a) *Given a metric sampled from U, $\mathcal{A}$ outputs a classifier that violates perfect metric-fairness with probability almost α.*
   (b) *Given a metric sampled from V, $\mathcal{A}$ outputs a classifier h whose error is almost $1/2$ with probability at least $(1 - \alpha)$.*

Moreover, the linear classifier w not only labels examples in $\mathcal{D}$ correctly, it also has large margins (greater than $1/2$) on every example in $\mathcal{D}$'s support.

Proof of Theorem 6. For the sake of readability, we choose to be somewhat informal in our treatment of asymptotics (i.e. we refer to a single distribution on learning problems and metrics, rather than an ensemble of distributions that grows with n). For an introduction to the foundations of cryptography, including pseudorandom generators, indistinguishability, and negligible quantities, we refer the reader to Goldreich [Gol01] .

We begin by specifying the distribution $\mathcal{D}$ and the classifier w. $\mathcal{D}$ will be the uniform distribution on the unit ball $\mathcal{X}$, conditioned on the n-th coordinate being either $1/2$ or $-1/2$. We restrict the last coordinate to ensure large margins (which are important for efficient PACF learnability). The linear classifier w simply outputs the sign of the last coordinate, and each example $x \in \mathcal{X}$ gets the label $w(x)$. Note that indeed the linear classifier w has perfect accuracy and large margins.

Metric distributions. To describe the metric, we use a cryptographic pseudo-random generator (PRG) as follows. Recall that a PRG is a function such-that no polynomial-time algorithm can distinguish between a uniformly random string in $\{0,1\}^{2n}$, and the output of G on a random string in $\{0,1\}^{n-1}$. Note that this is the case even though only a negligible fraction of the strings in $\{0,1\}^{2n}$ are in G's image. A celebrated result of Hastad *et al.* [HILL99] shows that PRGs can be constructed from any one-way function.

A metric d in U or in V is described by a string $y \in \{0,1\}^{2n}$. The only difference between U and V will be in the distribution of y: in U the vector y will be pseudorandom (in the image of the generator), in V the vector y will be truly random (and with overwhelming probability not in the image of the generator). Given y, the distance between two distinct individuals $x, x' \in \mathbb{R}^n$ is determined as follows:

1. If x and x' get the same label, i.e. if $sign(x[n]) = sign(x'[n])$, then $d(x,x') = 1$.
2. Otherwise, let $\Delta^{x,x'} \in \{0,1\}^{n-1}$ be computed as $\Delta_i^{x,x'} = \frac{1-(sign(x_i)\cdot sign(x'_i))}{2}$. I.e. $\Delta_i^{x,x'}$ is 0 if the signs of x_i and x'_i are identical, and 1 if they are different.

   If $G(\Delta^{x,x'}) = y$, then $d(x,x') = 0$.
3. Otherwise, $d(x,x') = 1$.

Note that for any choice of y, the above construction is indeed a pseduometric. The distance from x to itself is defined to be 0, and distances are symmetric by symmetry of the $\oplus$ operation. Finally, for every $x, x', x'' \in X$ we have that:

$$d(x,x') \leq d(x,x'') + d(x'',x').$$

To see this, observe that if $d(x,x') = 0$ then the LHS is bounded by the RHS. On the other hand, if $d(x,x') = 1$, then x and x' are distinct, and if x'' is also distinct from them, then it cannot be the case that both $d(x,x'') = 0$ and $d(x'',x') = 0$: one of these two distances must be 1.

We remark that in this construction we allow the distance between distinct points to be 0, and thus we get a distribution on pseudometrics. By defining the distance between examples x and x' s.t. $G(\Delta^{x,x'}) = y$ to be a small positive quantity rather than 0 we would obtain a true metric, and the results are essentially unchanged.

Turning to prove the claimed properties, we have:

Property 1. For metrics in U, y is pseudorandom, where $G(s) = y$ for some $s \in \{0,1\}^{n-1}$. We can divide the examples in $\mathcal{D}$'s support into (disjoint) pairs (x, x') s.t. $\Delta^{x,x'} = s$, where the label of x is 1 and the label of x' is -1. Let h be any perfectly metric-fair classifier. Since the $d(x, x') = 0$, h has to treat x and x' identically. Thus, the sum of h's errors on these two examples must be exactly 1. Since we can partition the support of $\mathcal{D}$ into disjoint pairs of this form, we conclude that h's overall error must be exactly $1/2$.

Property 2. For metrics in U, y is truly random. With overwhelming probability over the choice of y, there does not exist *any* $s \in \{0,1\}^{n-1}$ such that $G(s) = y$. When no such s exists, the metric d assigns distance 1 to *every* pair of distinct individuals. Thus, *every* hypothesis is perfectly metric-fair and in particular the classifier w is a perfectly metric-fair classifier with error 0.

Property 3. Finally, since G is a pseudorandom generator, no polynomial-time learner can distinguish between a metric (i.e. y) drawn from U and a metric (i.e. y) drawn from V (except with negligible advantage). For a learning algorithm $\mathcal{A}$, let α be the probability that $\mathcal{A}$, given a metric sampled from V, outputs a classifier whose error is noticeably less than $1/2$ (e.g. the error is no greater than $(1/2 - 1/n)$). Then, by the PRG's indistinguishability property, when given a metric sampled from U, the learner $\mathcal{A}$ must also output a classifier whose error is noticeably less than $1/2$ with probability almost α (e.g. at least $(\alpha - 1/n)$). But by Property 1, whenever this is the case, $\mathcal{A}$ is also violating perfect metric-fairness. □

F Conclusions and Future Directions

We conclude with several directions for future exploration:

The source of the similarity metric. Throughout our work, we assumed that the similarity metric is known to the learner. This is a natural assumption in the scenario that the metric is used as a vehicle for knowledgeably correcting biases in the training data, or in domains where such metrics naturally exist (such as credit scores and insurance risk scores). In other settings, however, relying on the existence of a metric is clearly a limitation (as also noted by [DHP+12]). One potential avenue for future work is investigating the use of machine learning to recover a similarity metric from fairly labeled data.

Assumptions on the metric. We make no assumptions about the similarity metric. In particular, it can be completely incompatible with accuracy and cryptographically contrived. Studying fairness-accuracy trade-offs imposed by particular similarity metrics is an interesting direction for future research direction and could also be supplemented by empirical studies. Another interesting question is whether there exist natural classes of metrics for which the hardness results for perfect metric-fairness do not hold.

Other efficient fair algorithms. The main challenge in *efficient* metric-fair learning is that the metric-fairness constraints are specified in terms of the hypotheses themselves. This means that when the hypotheses are not convex (e.g, the class of logistic predictors), the fairness constraints specify a non-convex set. In the case of logistic regression we overcame these barriers using *improper*

learning. However, computational tractability came at the cost of an increase in the sample complexity, and the resulting algorithm was only polynomial so long as the Lipschitz constant L of the sigmoidal transfer function was small. In particular, we only expect good accuracy in cases where data is linearly separable with large (expected) margins. The hardness result in [SSSS11] suggest that a polynomial dependence on L cannot be achieved using this method. A natural question, therefore, is whether other approaches can be used to construct metric-fair learning algorithms which are efficient and can be accurate even for data separated by smaller margins.

G Acknowledgements

We thank Cynthia Dwork and Omer Reingold for invaluable and illuminating conversations.

References

[ABDCBH97] Noga Alon, Shai Ben-David, Nicolo Cesa-Bianchi, and David Haussler. Scale-sensitive dimensions, uniform convergence, and learnability. *Journal of the ACM (JACM)*, 44(4):615–631, 1997.

[BCZC17] Alex Beutel, Jilin Chen, Zhe Zhao, and Ed H. Chi. Data decisions and theoretical implications when adversarially learning fair representations. *CoRR*, abs/1707.00075, 2017.

[BHJ+17] Richard Berk, Hoda Heidari, Shahin Jabbari, Matthew Joseph, Michael Kearns, Jamie Morgenstern, Seth Neel, and Aaron Roth. A convex framework for fair regression. *arXiv preprint arXiv:1706.02409*, 2017.

[BM02] Peter L Bartlett and Shahar Mendelson. Rademacher and gaussian complexities: Risk bounds and structural results. *Journal of Machine Learning Research*, 3(Nov):463–482, 2002.

[Cho17] Alexandra Chouldechova. Fair prediction with disparate impact: A study of bias in recidivism prediction instruments. *Big data*, 5(2):153–163, 2017.

[CKK+13] Toon Calders, Asim Karim, Faisal Kamiran, Wasif Ali, and Xiangliang Zhang. Controlling attribute effect in linear regression. In *Data Mining (ICDM), 2013 IEEE 13th International Conference on*, pages 71–80. IEEE, 2013.

[DHP+12] Cynthia Dwork, Moritz Hardt, Toniann Pitassi, Omer Reingold, and Richard S. Zemel. Fairness through awareness. In *Innovations in Theoretical Computer Science (ITCS)*, pages 214–226, 2012.

[DIKL17] Cynthia Dwork, Nicole Immorlica, Adam Tauman Kalai, and Max Leiserson. Decoupled classifiers for fair and efficient machine learning. *CoRR*, abs/1707.06613, 2017.

[FFM+15] Michael Feldman, Sorelle A Friedler, John Moeller, Carlos Scheidegger, and Suresh Venkatasubramanian. Certifying and removing disparate impact. In *Proceedings*

of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pages 259–268. ACM, 2015.

[GJKR18] Stephen Gillen, Christopher Jung, Michael J. Kearns, and Aaron Roth. Online learning with an unknown fairness metric. *CoRR*, abs/1802.06936, 2018.

[Gol01] Oded Goldreich. *The Foundations of Cryptography - Volume 1, Basic Techniques*. Cambridge University Press, 2001.

[HILL99] Johan Håstad, Russell Impagliazzo, Leonid A. Levin, and Michael Luby. A pseudorandom generator from any one-way function. *SIAM J. Comput.*, 28(4):1364–1396, 1999.

[HJKRR17] Ursula Hébert-Johnson, Michael P Kim, Omer Reingold, and Guy N Rothblum. Calibration for the (computationally-identifiable) masses. *arXiv preprint arXiv:1711.08513*, 2017.

[HPS16] Moritz Hardt, Eric Price, and Nathan Srebro. Equality of opportunity in supervised learning. *CoRR*, abs/1610.02413, 2016.

[HS07] Lisa Hellerstein and Rocco A. Servedio. On PAC learning algorithms for rich boolean function classes. *Theor. Comput. Sci.*, 384(1):66–76, 2007.

[JKM$^+$] Matthew Joseph, Michael Kearns, Jamie Morgenstern, Seth Neel, and Aaron Roth. Better fair algorithms for contextual bandits.

[JKMR16] Matthew Joseph, Michael Kearns, Jamie H Morgenstern, and Aaron Roth. Fairness in learning: Classic and contextual bandits. In *Advances in Neural Information Processing Systems*, pages 325–333, 2016.

[KAAS12] Toshihiro Kamishima, Shotaro Akaho, Hideki Asoh, and Jun Sakuma. Fairness-aware classifier with prejudice remover regularizer. In *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*, pages 35–50. Springer, 2012.

[KMR16] Jon Kleinberg, Sendhil Mullainathan, and Manish Raghavan. Inherent trade-offs in the fair determination of risk scores. *arXiv preprint arXiv:1609.05807*, 2016.

[KNRW17] Michael Kearns, Seth Neel, Aaron Roth, and Zhiwei Steven Wu. Preventing fairness gerrymandering: Auditing and learning for subgroup fairness. *arXiv preprint arXiv:1711.05144*, 2017.

[KP02] Vladimir Koltchinskii and Dmitry Panchenko. Empirical margin distributions and bounding the generalization error of combined classifiers. *Annals of Statistics*, pages 1–50, 2002.

[KRR18] Michael P. Kim, Omer Reingold, and Guy N. Rothblum. Fairness through computationally-bounded awareness. *CoRR*, abs/1803.03239, 2018.

[KST09] Sham M Kakade, Karthik Sridharan, and Ambuj Tewari. On the complexity of linear prediction: Risk bounds, margin bounds, and regularization. In *Advances in neural information processing systems*, pages 793–800, 2009.

[O'N16] Cathy O'Neil. *Weapons of Math Destruction: How Big Data Increases Inequality and Threatens Democracy*. Crown, New York, 2016.

[ORA16] Luca Oneto, Sandro Ridella, and Davide Anguita. Tikhonov, ivanov and morozov regularization for support vector machine learning. *Machine Learning*, 103(1):103–136, 2016.

[SSBD14] Shai Shalev-Shwartz and Shai Ben-David. Understanding machine learning: From theory to algorithms. chapter 16, pages 215–226. Cambridge university press, 2014.

[SSSS11] Shai Shalev-Shwartz, Ohad Shamir, and Karthik Sridharan. Learning kernel-based halfspaces with the 0-1 loss. *SIAM Journal on Computing*, 40(6):1623–1646, 2011.

[Val84] Leslie G. Valiant. A theory of the learnable. *Commun. ACM*, 27(11):1134–1142, 1984.

[Wah90] Grace Wahba. *Spline models for observational data*. SIAM, 1990.

[WGOS17] Blake Woodworth, Suriya Gunasekar, Mesrob I Ohannessian, and Nathan Srebro. Learning non-discriminatory predictors. *arXiv preprint arXiv:1702.06081*, 2017.

[ZVGRG17] Muhammad Bilal Zafar, Isabel Valera, Manuel Gomez Rodriguez, and Krishna P Gummadi. Fairness constraints: Mechanisms for fair classification. *arXiv preprint arXiv:1507.05259*, 2017.

[ZWS+13] Rich Zemel, Yu Wu, Kevin Swersky, Toni Pitassi, and Cynthia Dwork. Learning fair representations. In *International Conference on Machine Learning*, pages 325–333, 2013.