Preventing Disparate Treatment in Sequential Decision Making

Hoda Heidari[1] and **Andreas Krause**[1]
[1]ETH Zürich
hheidari@inf.ethz.ch, krausea@ethz.ch

Abstract

We study fairness in sequential decision making environments, where at each time step a learning algorithm receives data corresponding to a new individual (e.g. a new job application) and must make an *irrevocable* decision about him/her (e.g. whether to hire the applicant) based on observations made so far. In order to prevent cases of *disparate treatment*, our time-dependent notion of fairness requires algorithmic decisions to be *consistent*: if two individuals are similar in the feature space and arrive during the same time epoch, the algorithm must assign them to similar outcomes. We propose a general framework for post-processing predictions made by a black-box learning model, that guarantees the resulting sequence of outcomes is consistent. We show theoretically that imposing consistency will not significantly slow down learning. Our experiments on two real-world data sets illustrate and confirm this finding in practice.

1 Introduction

With the rise of big data, the use of algorithmic decision making systems has become widespread in a broad range of social domains—examples include employment [Miller, 2015], credit lending [Petrasic *et al.*, 2017], and criminal justice [Barry-Jester *et al.*, 2015]. Algorithmic decisions made in this fashion directly impact people's lives and may potentially affect certain individuals or social groups negatively [Podesta *et al.*, 2014]. In recent years, numerous studies (see e.g. [Sweeney, 2013; Barocas and Selbst, 2016]) and media articles [Mann and O'Neil, 2016; Angwin *et al.*, 2016; Levin, 2016] have illustrated and cautioned against algorithmic unfairness. This has led to an active area of research into quantifying and guaranteeing fairness for machine learning [Kamishima *et al.*, 2012; Dwork *et al.*, 2012; Kleinberg *et al.*, 2016; Hardt *et al.*, 2016]. Most existing notions of algorithmic fairness, such as demographic parity or equality of opportunity, seek to prevent a particular form of discrimination, known as *disparate impact* in legal texts—disparate impact refers to practices that collectively allocate a more favorable outcome to one socially salient group compared to another. Moreover, the vast majority of existing studies address fairness considerations for *supervised batch learning*—where the entire training data is available ahead of time, and the same predictive model is applied to every new instance—and little attention has been given to *online* settings in which the learner receives individuals' data over time, and makes irrevocable decisions about them in a sequential fashion. In practice—e.g. when it comes to credit lending and employment decisions—the latter is often the case.

In this work, we consider an online setting where at each time step the learner receives data corresponding to a new individual, and assigns an irrevocable outcome/label to him/her. The learner then observes the true label for that individual, and incorporates the recent observation into decision making for future rounds. In contrast to previous work, our goal here is to prevent cases of *disparate treatment*—disparate treatment refers to unequal treatment of similarly situated individuals due to their protected characteristics (e.g. race or sex). Our notion of fairness requires algorithmic decisions to be *consistent* over time. That is, if two individuals are similar in the feature space and arrive during the same time epoch, the algorithm must assign them to similar outcomes. Our definition of fairness can be thought of as a *time-dependent* variant of [Dwork *et al.*, 2012].

Our formulation of disparate impact is motivated by the following observation: in the United States legal system, the most common method for establishing a case of disparate treatment is to present *circumstantial* evidence that one has been treated unfavorably compared to *similarly situated individuals* (see the Civil Rights Act of 1964, Title VII, Equal Employment Opportunities). More specifically in the context of employment decisions such as hiring, promotion, or salary, a qualified individual who belongs to a protected class can establish a prima facie case of disparate treatment by proving that the employer treated more favorably similarly situated individuals who do not share the protected characteristic. For example, *"the employer rejected their job application, but continued to solicit applicants with equal qualifications"* (the wage project, http://www.wageproject.org/files/pdispimp.php). We remark that thus far the fair-ML community has largely interpreted disparate treatment as the explicit incorporation of a protected characteristic as input to the learning algorithm (see e.g. [Lipton *et al.*, 2017]). While the explicit use of a sensitive feature can clearly be seen as a *direct* evidence of disparate treatment, such evidence is very easy to avoid—by removing the sensitive feature from training data—and is rarely available in real-world scenarios. In this work, our focus is on a more nuanced, but more common method of showing disparate treatment, that is, by providing *indirect/circumstantial* evidence that one has been treated unfavorably compared to similarly situated individuals.

Armed with this broader understanding of disparate treatment, an employer may wish to prevent its decision making algorithm from treating similarly-situated individuals differently. This motivates the definition of fairness we propose in this work: While the eventual goal of the learner is to find a hypothesis with low prediction error, the labels it assigns to individuals along the way must be consistent with one another. It is easy to see that requiring consistency to hold over the entire span of decision making severely restricts the learner's ability to incorporate new information into its predictive model. Therefore, we require consistency to hold among recent observations only. Our key contribution is presenting a general framework for post-processing predictions made by a black-box learning model, so that the resulting sequence of outcomes is consistent. Our theoretical analysis shows that imposing consistency does not significantly slow down learning. Our experiments on two real-world data sets further illustrate and confirm this finding in practice.

1.1 Related Work

Existing notions of algorithmic (un)fairness can be divided into two main categories: *individual fairness* and *group fairness*. Most existing studies of algorithmic (un)fairness focus on statistical or group notions. Statistical parity [Kleinberg *et al.*, 2016; Dwork *et al.*, 2012; Corbett-Davies *et al.*, 2017], disparate impact [Zafar *et al.*, 2017; Feldman *et al.*, 2015], equality of opportunity [Hardt *et al.*, 2016], and calibration [Kleinberg *et al.*, 2016] are important formulations belonging to this category. Statistical notions of fairness suffer from several drawbacks—perhaps most importantly they fail to guarantee fairness at the individual level; see [Berk *et al.*, 2017] for detailed examples.

The notion of individual fairness was first proposed by Dwork et al. [2012] for classification in batch learning environments. The notion requires that two individuals who are similar with respect to the task at hand, receive similar probability distributions over class labels. Our definition of fairness can be thought of as a *time-dependent* variant of [Dwork *et al.*, 2012]. For regression tasks, our notion of consistency replaces their statistical distance between distributions with the difference between actual regression labels—we argue that individuals are mainly concerned with their realized outcomes, not the probability distribution from which those outcome are sampled. Unfortunately, this requirement can render learning impossible when it comes to classification tasks. Guaranteeing consistency among label *distributions*, however, can still be done readily by our algorithm (see Section 4).

Existing mechanisms to guarantee fairness for learning algorithms can be divided into three categories: preprocessing (see e.g. [Pedreschi *et al.*, 2008]), in-processing (see e.g. [Dwork *et al.*, 2012]), and post-processing (see e.g. [Hardt *et al.*, 2016]). Our algorithm belongs to the third category: it does not interfere with the inner workings of the learning procedure; rather treats it as a black-box and adjusts the algorithmic predictions so as to maintain consistency.

A number of recent studies have initiated the study of fairness for online learning [Joseph *et al.*, 2016; 2017; Jabbari *et al.*, 2017]. Joseph et al. [2016] study fairness in the multi-armed bandit setting, where arms correspond to socially salient groups (e.g. racial groups) and pulling an arm corresponds to choosing an individual from that group (e.g. to allocate a loan to). Simply put, an algorithm is considered fair when it never prefers one arm to another if the chosen arm has lower expected reward than the unchosen one. This probabilistic definition of fairness does not make any comparison between algorithmic decisions made over time. Jabbari et al. [2017] study fairness in reinforcement learning. Their definition of fairness is similar to that proposed by Joseph et al. [2016]: a fair algorithm must never prefer one action to another if the long-term discounted reward of the latter is higher. In both of these models, guaranteeing fairness may impose a high (and in worst-case exponential) cost on learning.

Our learning framework is closely related to PAC learning and (stochastic) online learning. In particular, similar to the PAC framework, our goal is to bound the number of samples a learning algorithm needs to observe before reaching a certain level of accuracy with high probability. But unlike PAC learning and similar to online learning, our framework is *time-dependent*: instances arrive sequentially, and the algorithm has to make decisions on the go, and before seeing all training instances. In contrast to previous work on (stochastic) online learning, we don't focus on regret minimization; rather we aim at achieving a sufficient level of accuracy while fulfilling certain constraints (in particular, consistency) on the labels generated over time.

2 Model and Preliminaries

We start by introducing our *Sequential Decision Making* framework. We focus our presentation on regression tasks, but as we will discuss later in Section 4, our work can accommodate classification, as well.

At each round $t = 0, 1, 2, \cdots$, the learner $\mathcal{A}$ receives a new context $\mathbf{x}_t \in \mathcal{X}$, and makes a prediction $\bar{y}_t \in \mathcal{Y} = [0, 1]$. The learner then observes the true label y_t for $\mathbf{x}_t$. There exists an underlying distribution $\mathcal{D}$ on $\mathcal{X} \times \mathcal{Y}$ such that for every $t = 0, 1, 2, \cdots$, $(\mathbf{x}_t, y_t)$ is an i.i.d. sample from $\mathcal{D}$. The goal of the learner is to eventually find a hypothesis $h \in \mathcal{H}$ to label instances, such that h has bounded error with high probability. Unlike the traditional PAC learning framework, the learner may face additional constraints on the sequence of labels it produces along the way.

In this work, our focus is on constraints imposed by fairness considerations. At a high level, we would like the algorithm to assign to every new instance $\mathbf{x}_t$ a label $\bar{y}_t$ that is not too far away from labels assigned to similar contexts in recent history. More precisely, we assume the existence of a distance *metric* d among contexts in $\mathcal{X}$:

$$d : \mathcal{X} \times \mathcal{X} \to [0, \infty)$$

Note that a metric satisfies non-negativity, symmetry, identity of indiscernibles, and most importantly for the purposes of this work, the *triangle inequality*:

$$\forall \mathbf{x}_i, \mathbf{x}_j, \mathbf{x}_k \in \mathcal{X} : d(\mathbf{x}_i, \mathbf{x}_j) + d(\mathbf{x}_j, \mathbf{x}_k) \geq d(\mathbf{x}_i, \mathbf{x}_k).$$

We define consistency with respect to d, and as follows:

Definition 1 (Relative and Pairwise Fairness) *Two labeled contexts* $(\mathbf{x}_i, \bar{y}_i), (\mathbf{x}_j, \bar{y}_j) \in \mathcal{X} \times \mathcal{Y}$ *are called* γ-relatively fair *if:*

$$|\bar{y}_i - \bar{y}_j| \leq d(\mathbf{x}_i, \mathbf{x}_j) + \gamma \qquad (1)$$

where $\gamma \in [0, 1]$ *is a constant. A set* S *of labeled contexts is called* γ-pairwise fair *if (1) holds for any* $(\mathbf{x}_i, \bar{y}_i), (\mathbf{x}_j, \bar{y}_j) \in S$.

Ideally, we would like $\mathcal{A}$ to always produce pairwise fair labels; that is, for any time step T we would like $\{(\mathbf{x}_t, \bar{y}_t)\}_{t=0}^{T}$ to be γ-pairwise fair. However, to avoid perpetuating mistakes $\mathcal{A}$ might make early on, we can only require this to hold among recent observations (see Example 1). Formally, we define our notion of time-*consistency* as follows:

Definition 2 (Consistency) *Given constants* $\gamma > 0$ *and* $K \in \mathbb{N}$, *a sequence* $\{(\mathbf{x}_t, \bar{y}_t)\}_{t=0}^{\infty}$ *of labeled contexts is called* (γ, K)-consistent *if for any* $t = 0, 1, \cdots,$ *the set* $S_t = \{(\mathbf{x}_i, \bar{y}_i)\}_{i=t}^{t+K}$ *is* γ*-pairwise fair.*

Equivalently the sequence is (γ, K)-consistent, if for any $t, s \in \mathbb{N} \cup \{0\}$ with $|t - s| \leq K$, $(\mathbf{x}_t, \bar{y}_t), (\mathbf{x}_s, \bar{y}_s)$ are γ-relatively fair.

Definition 3 (Consistent Sequential (CS) Learnability) *A hypothesis class* $\mathcal{H}$ *is* (γ, K)-CS learnable *if there exists an algorithm* $\mathcal{A}$ *operating in the sequential decision making framework such that for any* $\epsilon, \delta \in (0, 1)$ *and any distribution* $\mathcal{D}$,

- $\mathcal{A}$ *produces labels* $\bar{y}_0, \bar{y}_1, \bar{y}_2, \cdots$ *that are* (γ, K)*-consistent.*
- *After observing* $\mathcal{N}(\gamma, K, \epsilon, \delta)$ *instances along with their true labels, with probability at least* $(1 - \delta)$ $\mathcal{A}$ *can find and thereafter follow a hypothesis* $h^* \in \mathcal{H}$ *such that*

$$L_{\mathcal{D}}(h^*) \leq \min_{h \in \mathcal{H}} L_{\mathcal{D}}(h) + \epsilon$$

where L *is the loss function.*

The function $\mathcal{N}(\gamma, K, \epsilon, \delta)$ specifies the sample complexity of CS-learning.

Our definition of CS-learnability requires $\mathcal{A}$ to eventually commit to a hypothesis $h^* \in \mathcal{H}$; that is, for $t > \mathcal{N}(\gamma, K, \epsilon, \delta)$, $\bar{y}_t = h^*(\mathbf{x}_t)$. Given this requirement, in order for $\mathcal{A}$'s sequence of labels to remain consistent throughout, we need to make the following assumption:

Assumption 1 *Every* $h \in \mathcal{H}$ *is* η*-fair with* $0 \leq \eta \leq \gamma$*:*

$$\forall \mathbf{x}_i, \mathbf{x}_j \in \mathcal{X} : |h(\mathbf{x}_i) - h(\mathbf{x}_j)| \leq d(\mathbf{x}_i, \mathbf{x}_j) + \eta.$$

To simplify the statement of our results, throughout we assume $\eta = 0$, but as we discuss in Section 4, our theory readily extends to $0 < \eta < \gamma$.

How strong is Assumption 1? First, note that for any given hypothesis class $\mathcal{H}$ and distance metric d, Assumption 1 holds if η is taken to be sufficiently large—for instance, it trivially holds for $\eta \geq \max_{h,\mathbf{x}} h(\mathbf{x})$ regardless of the distance metric d. Second, for a particular choice of η, one can apply Dwork *et al.*'s pairwise constraints ahead of time in order to restrict $\mathcal{H}$ to η-fair hypotheses only. In Section 5, we empirically illustrate the impact of enforcing these pairwise constraints on accuracy. Lastly, in Section 5, we show empirically that our proposed algorithm performs well even if Assumption 1 is violated.

We end this section with two remarks. First, as mentioned earlier for CS-learnability to be possible, K has to be finite. The following example illustrates the impossibility of CS-learning when $K = \infty$.

Example 1 *Let* $\mathcal{H}$ *consists of two hypotheses only:* h_0 *which always predicts* 0 *and* h_1 *that always predicts* 1*. Consider the realizable setting, and suppose* $\mathcal{D}$ *is a degenerate distribution with all its mass on a particular context* $\mathbf{x} \in \mathcal{X}$*, so that for all* $t = 0, 1, \cdots$, $\mathbf{x}_t = \mathbf{x}$*. Let* $K = \infty$ *and* $\gamma < 1$*. Any algorithm has to make a mistake in predicting* y_0 *with probability at least* 0.5 *(e.g.* $y_0 = 1$*, but the algorithm predicts* $\bar{y}_0 = 0$*).*

After observing the true label for $\mathbf{x}_0$ *even though the algorithm can accurately predict the true label for all upcoming instances (due to the realizability assumption and* $\mathcal{D}$ *being degenerate), for all* $t \geq 1$*, the* γ*-relative fairness between* $\bar{y}_0, \bar{y}_t$ *forces it to make a prediction with error at least* $(1 - \gamma)$ *for* $\mathbf{x}_t$*:* $\bar{y}_t - \bar{y}_0 \leq d(\mathbf{x}_t, \mathbf{x}_0) + \gamma = \gamma$*. Therefore for* $\delta < 0.5$ *and* $\epsilon < 1 - \gamma$*, the CS-learnablity conditions cannot be satisfied.*

Second, we note that a CS-learning algorithm can indefinitely continue generating (γ, K)-consistent labels, without ever getting stuck a situation where no consistent label exists for the new context.

Proposition 1 *Suppose* $S = \{(\mathbf{x}_i, \bar{y}_i)\}_{i=0}^{K-1}$ *is* γ*-pairwise fair. Then for any* $\mathbf{x}_K \in \mathcal{X}$ *there exists a non-empty interval* I *of length at least* γ*, such that any* $\bar{y}_K \in I$ *is a* γ*-relatively fair label for* $\mathbf{x}_K$ *with respect to* S*.*

Proof In order for $\bar{y}_K$ to be γ-relatively fair with respect to $\bar{y}_i$ $(i = 0, \cdots, K - 1)$ it must be the case that

$$\bar{y}_K \in [\bar{y}_i - d(\mathbf{x}_K, \mathbf{x}_i) - \gamma, \bar{y}_i + d(\mathbf{x}_K, \mathbf{x}_i) + \gamma] \qquad (2)$$

Call the above interval I_i. If $I = \bigcap_{i=1}^{K-1} I_i \neq \emptyset$, then any $\bar{y}_K \in \bigcap_{i=1}^{K-1} I_i$ is a γ-relatively fair label with respect to every element of S. Therefore, the statement of the proposition is equivalent to $I \neq \emptyset$ and $|I| \geq \gamma$. To prove this holds, we apply the following lemma:[1]

Lemma 1 *Consider* m *intervals* $I_1, I_2, \cdots, I_m$ *where* $I_i = [l_i, u_i]$*. Suppose that for any* $1 \leq i < j \leq m$, $|I_i \cap I_j| \geq \ell$*, where* $\ell \geq 0$ *is a constant. Then*

- $\bigcap_{i=1}^{m} I_i = [\max_i l_i, \min_i u_i]$.
- $|\bigcap_{i=1}^{m} I_i| \geq \ell$.

Note that for any $0 \leq i, j \leq K - 1$, $|I_i \cap I_j| \geq \gamma$. Suppose not, and there exists $0 \leq i, j \leq K - 1$ such that $|I_i \cap I_j| < \gamma$.

[1] The proof is straightforward and is omitted due to space constraints.

That means either $\bar{y}_i - d(\mathbf{x}_K, \mathbf{x}_i) > \bar{y}_j + d(\mathbf{x}_K, \mathbf{x}_j) + \gamma$ or $\bar{y}_j - d(\mathbf{x}_K, \mathbf{x}_j) > \bar{y}_i + d(\mathbf{x}_K, \mathbf{x}_i) + \gamma$. Consider the first case (the analysis for the second case is identical). We have that

$$\begin{aligned} & \bar{y}_i - d(\mathbf{x}_K, \mathbf{x}_i) > \bar{y}_j + d(\mathbf{x}_K, \mathbf{x}_j) + \gamma \\ \Rightarrow \quad & \bar{y}_i - \bar{y}_j > d(\mathbf{x}_K, \mathbf{x}_i) + d(\mathbf{x}_K, \mathbf{x}_j) + \gamma \\ \Rightarrow \quad & \bar{y}_i - \bar{y}_j > d(\mathbf{x}_i, \mathbf{x}_j) + \gamma \end{aligned} \tag{3}$$

where the last line follows from the triangle inequality. But (3) is in contradiction with γ-relative fairness of $\bar{y}_i, \bar{y}_j$. Therefore for any $0 \leq i, j \leq K-1$, $|I_i \cap I_j| \geq \gamma$. Now applying Lemma 1, we obtain that $\bigcap_{i=1}^{K-1} I_i \neq \emptyset$ and $|\bigcap_{i=1}^{K-1} I_i| \geq \gamma$. This finishes the proof. ∎

3 Algorithm

In this Section, we propose a general CS-learning algorithm, called *Consistently Follow The Leader (CFTL)*. Our algorithm is compatible with any PAC-learnable hypothesis class, and requires only *blackbox* access to the corresponding learning algorithm. We show that the CS-sample complexity of CFTL has polynomial dependency on K and $\frac{1}{\gamma}$, and furthermore, this dependency is tight.

Let $\mathcal{H}$ be a PAC-learnable hypothesis class with sample complexity specified by $\mathcal{N}(\epsilon, \delta)$. CFTL requires black-box access to the PAC learning algorithm for $\mathcal{H}$, denoted by $PAC_{\mathcal{H}}$. $PAC_{\mathcal{H}}$ receives a training data set of size $\mathcal{N}(\epsilon, \delta)$ and outputs a hypothesis $h \in \mathcal{H}$ whose loss is bounded by ϵ with probability at least $(1-\delta)$.

At a high level, CFTL works as follows: at each round T, it feeds the data observed up to T to $PAC_{\mathcal{H}}$ and obtains a predictive model h. Given a new instance $\mathbf{x}_T$, it utilizes h to make a prediction about the corresponding label of $\mathbf{x}_T$. This predicted label will not necessarily be consistent with the labels CFTL has previously generated (note that even though we assume all hypotheses in $\mathcal{H}$ are consistent, the predictive model, h, changes over time, so labels predicted over time need not be consistent). Among the set of all (γ, K)-consistent labels for $\mathbf{x}$ (which according to Proposition 1 is a non-empty interval), CFTL picks the one closest to $h(\mathbf{x}_T)$ as $\bar{y}_T$.

Next, we show that it takes CFTL at most $\mathcal{N}(\epsilon, \delta) + \frac{K}{\gamma}$ steps to be able to follow a hypothesis h^* whose error is bounded by ϵ with probability at least $1-\delta$. Note that while CFTL can find such hypothesis sooner—after exactly $\mathcal{N}(\epsilon, \delta)$ steps—it cannot immediately start generating labels according to h^* due to the (γ, K)-consistency constraints.

Theorem 1 *Suppose the sample complexity of PAC learning a hypothesis class $\mathcal{H}$ is specified by $\mathcal{N}(\epsilon, \delta)$. Then the sample complexity of CS-learning for $\mathcal{H}$ is at most $\mathcal{N}(\epsilon, \delta) + \frac{K}{\gamma}$. Furthermore, there exist examples for which this bound is tight.*

Proof First, consider the case for $K = 1$. According to Proposition 1, CFTL always produces (γ, K)-consistent labels. Also given the sample complexity of PAC learning for class $\mathcal{H}$ we know that at time $\mathcal{N}(\epsilon, \delta)$ the algorithm has enough samples to find a hypothesis h^* whose error is bounded by ϵ with probability at least $1-\delta$. For $t > \mathcal{N}(\epsilon, \delta)$, define the potential function p_t as the difference between the output label (i.e. $\bar{y}_t$) and predicted label (i.e. $h^*(\mathbf{x}_t)$) for $\mathbf{x}_t$:

$$p_t \equiv |\bar{y}_t - h^*(\mathbf{x}_t)|.$$

Algorithm 1: CFTL

```
input: $K, \gamma, \epsilon, \delta$
$\mathcal{S} = \emptyset$.                 // Training data set
$h \leftarrow h_0$.       // Pick an arbitrary $h_0 \in \mathcal{H}$.
for $T = 0, 1, 2, \cdots$ do
    Receive a new context $\mathbf{x}_T \in \mathcal{X}$.
    $l \leftarrow \max_{t \in \{T-K, \cdots, T-1\}} \bar{y}_t - d(\mathbf{x}_T, \mathbf{x}_t) - \gamma$.
    $u \rightarrow \min_{t \in \{T-K, \cdots, T-1\}} \bar{y}_t + d(\mathbf{x}_T, \mathbf{x}_t) + \gamma$.
    Let $I = [l, u]$.       // $I$ is the set of all
     consistent labels for $\mathbf{x}_T$
    if $h(\mathbf{x}_T) \in I$ then
        $\bar{y}_T = h(\mathbf{x}_T)$.
    else if $\bar{y}_T < l$ then
        $\bar{y}_T = l$
    else
        $\bar{y}_T = u$
    end
    Observe $y_T$ for $\mathbf{x}_T$ and $\mathcal{S} \leftarrow \mathcal{S} \cup \{(\mathbf{x}_T, y_T)\}$.
    if $T \leq \mathcal{N}(\epsilon, \delta)$ then
        $h \leftarrow PAC_{\mathcal{H}}(\mathcal{S})$
    end
end
```

Obviously $p_t \geq 0$ for all $t > \mathcal{N}(\epsilon, \delta)$. Note that if at time step t, $p_t = 0$, then CFTL can follow h^* thereafter. This is because h^* is fair, and by generating all future labels via h^*, CFTL remains $(\gamma, 1)$-consistent. If not and $p_t > 0$, then we claim $p_{t+1} - p_t \geq \gamma$—that is, the potential function shrinks by a margin of at least γ every round. Assuming this claim holds, it is immediate that it takes CFTL at most $\frac{1}{\gamma}$ additional instances (starting from $t = \mathcal{N}(\epsilon, \delta) + 1$) to be able to follow h^*, finishing the proof for $K = 1$.

It only remains to establish the above claim for cases where $p_t > 0$. Suppose without loss of generality that $\bar{y}_t > h^*(\mathbf{x}_t)$ (the analysis for the case of $\bar{y}_t < h^*(\mathbf{x}_t)$ is analogous). Three cases are possible for $h^*(\mathbf{x}_{t+1})$:

- $h^*(\mathbf{x}_{t+1}) \in [\bar{y}_t - d(\mathbf{x}_t, \mathbf{x}_{t+1}) - \gamma, \bar{y}_t + d(\mathbf{x}_t, \mathbf{x}_{t+1}) + \gamma]$. This means $p_{t+1} = 0$.
- $h^*(\mathbf{x}_{t+1}) > \bar{y}_t + d(\mathbf{x}_t, \mathbf{x}_{t+1}) + \gamma$. Re-write this as

  $$\begin{aligned} & h^*(\mathbf{x}_{t+1}) - \bar{y}_t > d(\mathbf{x}_t, \mathbf{x}_{t+1}) + \gamma \\ \Rightarrow \quad & h^*(\mathbf{x}_{t+1}) - h^*(\mathbf{x}_t) > d(\mathbf{x}_t, \mathbf{x}_{t+1}) + \gamma \end{aligned}$$

  where the last line follows because $\bar{y}_t \geq h^*(\mathbf{x}_t)$. This is a contradiction with the fact that h^* is 0-fair. So this case is impossible.
- $h^*(\mathbf{x}_{t+1}) < \bar{y}_t - d(\mathbf{x}_t, \mathbf{x}_{t+1}) - \gamma$. In this case, we have that $\bar{y}_{t+1} = \bar{y}_t - d(\mathbf{x}_t, \mathbf{x}_{t+1}) - \gamma$. If $p_t - p_{t+1} < \gamma$ (or equivalently $p_{t+1} > p_t - \gamma$), we have that

  $$\bar{y}_{t+1} - h^*(\mathbf{x}_{t+1}) > \bar{y}_t - h^*(\mathbf{x}_t) - \gamma.$$

  Plugging in $\bar{y}_{t+1} = \bar{y}_t - d(\mathbf{x}_t, \mathbf{x}_{t+1}) - \gamma$ we have

  $$\bar{y}_t - d(\mathbf{x}_t, \mathbf{x}_{t+1}) - \gamma - h^*(\mathbf{x}_{t+1}) > \bar{y}_t - h^*(\mathbf{x}_t) - \gamma$$

  or equivalently

  $$h^*(\mathbf{x}_t) - h^*(\mathbf{x}_{t+1}) > d(\mathbf{x}_t, \mathbf{x}_{t+1}) \tag{4}$$

But (4) is a contradiction with the fact that h^* is 0-fair. So it must be the case that $p_{t+1} \leq p_t - \gamma$. This finishes the proof.

Next, consider the case for $K > 1$. Define the potential function q_t as follows: $q_t \equiv \max_{s=t-K,\cdots,t} p_s$. We claim that q_t decreases by a margin of at least γ every K rounds. Assuming this claim holds, it is immediate that it takes CFTL at most $\frac{K}{\gamma}$ additional instances (starting from $t = \mathcal{N}(\epsilon, \delta)+1$) to be able to follow h^*.

It only remains to prove the above claim. CFTL chooses $\bar{y}_{t+1}$ so that it is γ-relatively fair with respect to previous K labels. So according to Proposition 1, we know this means one of the following is the case:

- $p_{t+1} = 0$;
- there exists $i \in \{t, \cdots, t-K+1\}$ such that $h^*(\mathbf{x}_{t+1}) > \bar{y}_i + d(\mathbf{x}_{t+1}, \mathbf{x}_i) + \gamma$;
- or there exists $j \in \{t, \cdots, t-K+1\}$ such that $h^*(\mathbf{x}_{t+1}) < \bar{y}_j - d(\mathbf{x}_{t+1}, \mathbf{x}_j) - \gamma$.

For the latter two cases, following the argument we presented above for $K = 1$, we obtain that there exists $i \in \{t, \cdots, t-K+1\}$ such that $p_{t+1} \leq p_i - \gamma \leq q_t - \gamma$. So $q_{t+1} \leq q_t$. Repeating this for the next K contexts, we have that $q_{t+K} \leq q_t - \gamma$.

The following example shows the bound is tight: Consider the setting in Example 1. Any algorithm can be made to mistakenly predict y_0 (e.g. $y_0 = 1$, but the algorithm predicts $\bar{y}_0 = 0$) with probability at least 0.5. It is easy to see that in this case the algorithm is subsequently forced to predict $\bar{y}_t = \gamma$ for the first K instances ($t = 1, \cdots, K$), $\bar{y}_t = 2\gamma$ for the second K instances ($t = K+1, \cdots, 2K$), and so on, until predicting the true label is consistent with the previous K decisions. Therefore, it takes any algorithm $\frac{K}{\delta}$ additional steps before it can follow h^*. ■

4 Extensions and Discussion

In this Section, we discuss some of the extensions and limitations of our analysis.

Adjusting K and γ over time Our analysis allows for K and γ to be adjusted over time. In particular, it is possible to make the sequence of generated labels gradually more and more consistent as follows: Suppose at time $T > \mathcal{N}(\gamma, K, \epsilon, \delta)$, we wish to increase K to $K' > K$ and decrease γ to $\gamma' < \gamma$. For the analysis to remain valid all we need is that the last K' predictions are γ'-fair. This can be satisfied by predicting the same label for $\mathbf{x}_{T+1}, \cdots, \mathbf{x}_{T+K'}$. Continue with CFTL$(\gamma', K', \epsilon, \delta)$ thereafter.

Alternative methods for label prediction Our analysis is independent of how labels are predicted in the first $\mathcal{N}(\epsilon, \delta)$ rounds. We chose to make prediction at each time step using the hypothesis that best models the data observed so far, but any alternative label prediction method (e.g. standard online learning algorithms) can replace this without affecting the analysis for $t > \mathcal{N}(\epsilon, \delta)$. Also, instead of updating the hypothesis every round, one may choose to update it periodically with lower frequency (e.g. once every 100 new instances).

Analysis for $0 < \eta < \gamma$ The only part of the analysis that makes use of Assumption 1 with $\eta = 0$ is equation 4. If $0 < \eta < \gamma$ a similar argument shows that the potential function p_t decreases by a margin of at least $(\gamma - \eta)$ in each round, making the sample complexity $\mathcal{N}(\epsilon, \delta) + \frac{K}{\gamma - \eta}$. Note, however, that this bound is uninformative if $\eta = \gamma$. If $\eta > \gamma$, $\mathcal{H}$ is not (γ, K)-CS-learnable, and there remains always a gap between the accuracy of CFTL and h^*. As we will see in Section 5, in practice this gap is fairly small.

Classification Our analysis relies on the label set $\mathcal{Y}$ being a compact interval—as is the case in regression—but the argument readily extends to *binary* classification as follows: As opposed to working directly with the actual labels $y_i \in \mathcal{Y} = \{0, 1\}$, let z_i specify the probability of $\mathbf{x}_i$ being positive. Note that $z_i \in \mathcal{Z} = [0, 1]$, so by replacing y_i's with z_i's, the compactness is restored and the rest of the analysis goes through without any modification. Note, however, that similar to [Dwork *et al.*, 2012] this only guarantees (γ, K)-consistency among probability *distributions* over labels, and *not* among *realized* labels themselves.

To extend our work to multi-class classification, we propose two approaches. Suppose $|\mathcal{Y}| = M$. Let $z_i \in \Delta(\mathcal{Y})$ represent a probability distribution that specifies the probability with which $\mathbf{x}_i$ belongs to each of the M classes in $\mathcal{Y}$. The first approach assigns a utility to each distribution and compares them indirectly via their corresponding utilities; whereas the second approach directly compares two distributions using a distance metric defined over $\Delta(\mathcal{Y})$.

- *Utility-based approach:* Let $u : \Delta(\mathcal{Y}) \rightarrow \mathbb{R}$ be a continuous utility function that quantifies the degree of social desirability of each distribution (e.g. $u(z) = \sum_{o=1}^{M} z_o u_o$ where u_o is the utility of outcome $o \in \mathcal{Y}$ and z_o is its probability under distribution z). Modify the definition of relative fairness as follows:

  $$|u(\bar{z}_i) - u(\bar{z}_j)| \leq d(\mathbf{x}_i, \mathbf{x}_j) + \gamma$$

  And define the potential function as follows: $p_t \equiv |u(h^*(\mathbf{x}_t)) - u(\bar{z}_t)|$. The rest of the analysis closely follows the one presented in Section 3 and is omitted due to space constraint.

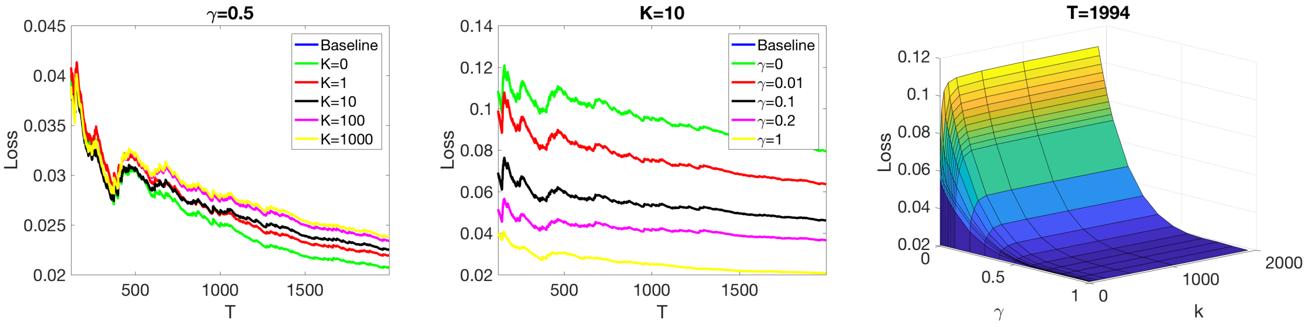
- *Distance metric between distributions:* Let D be a distance metric defined over the space of probability distributions on $\mathcal{Y}$ (e.g. D could be the L_∞-norm). One can modify the definition of relative fairness (Equation 1) as follows:

  $$D(\bar{z}_i, \bar{z}_j) \leq d(\mathbf{x}_i, \mathbf{x}_j) + \gamma$$

  and subsequently change the definition of the potential function to $p_t \equiv D(h^*(\mathbf{x}_t), \bar{z}_t)$. The complete analysis of this approach requires further elaboration, and is beyond the scope of this work.

5 Experiments

The theory presented in Sections 3, 4 is only applicable to cases where Assumption 1 holds and $\eta < \gamma$. In this Section, we first illustrate the effect of guaranteeing Assumption 1 on accuracy loss (measured in terms of Mean Squared Error).



(a) (b) (c)

Figure 1: Regression on the Crime and Communities data set. Here $\eta \geq 0.8$.

Second, for a setting in which the assumption is violated and $\eta > \gamma$, we empirically evaluate the performance of our algorithm in terms of accuracy loss, and compare it with the unconstrained baseline.

Datasets We ran lasso regression on the normalized *Crime and Communities data set* [Dheeru and Karra Taniskidou, 2017]. The data consists of 1994 observations each made up of 122 predictive features, and it contains socio-economic, law enforcement, and crime data from the 1995 FBI UCR. Community type (e.g. urban vs. rural), average family income, and the per capita number of police officers in the community are a few examples of the variables included in the dataset. The target variable is the “Per Capita Violent Crimes”. The data has been normalized so that for both the target variable and all the explanatory variables the range is between 0 and 1. We replaced all missing values with 0. We also ran logistic regression on a classification data set—the *Adult Income data set* [Dheeru and Karra Taniskidou, 2017]—and observed very similar trends.

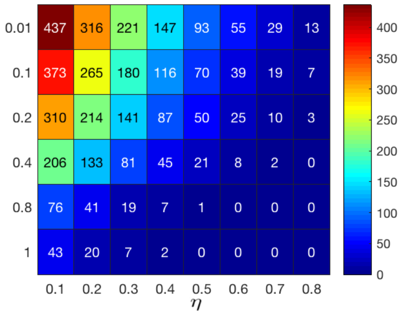
Accuracy loss when Assumption 1 is enforced. In order to guarantee Assumption 1 for a particular value of η, we optimized for accuracy subject to pairwise constraints of the form:

$$\forall i, j : |\bar{y}_i - \bar{y}_j| \leq c \times d(\mathbf{x}_i, \mathbf{x}_j) + \eta, \tag{5}$$

where c is a normalization parameter that adjusts the range of the distance between any two instances. Figure 2 shows the impact of c and η on the percentage of increase in accuracy loss. As expected, smaller values of c and η result in greater loss.

Accuracy loss when Assumption 1 does not hold. Next, we empirically evaluate the performance of our algorithm when Assumption 1 is violated, and compare its loss with an unconstrained baseline that works as follows: At time T it outputs $h_T(\mathbf{x}_T)$ where $h_T = PAC_{\mathcal{H}}(\{(\mathbf{x}_t, y_t)\}_{t=0}^{T-1})$. We tested the baseline and CFTL both on a regression and a classification data set. For the regression task, we ran LASSO with regularization coefficients $\lambda = 0.01$. λ was chosen by performing a 10-fold cross validation on the entire data set. We then used the same value of λ at all time steps.

Figure 1 illustrates the performance of CFTL on the Crime and Communities data set. We observe that for a fixed value of γ (i.e. $\gamma = 0.5$), CFTL's performance degrades as K increases (see Figure 1a). Similarly, for a fixed value of K (i.e. $K = 10$), CFTL's performance degrades as $1/\gamma$ increases (see Figure 1b). The rate of decay in accuracy is much faster for $1/\gamma$ as opposed to K (see Figure 1c). As expected, the baseline coincides with CFTL when $K = 0$ and when γ is sufficiently large (given the range of labels in this data set, $\gamma = 1$ suffices for this to be the case for any K). Importantly, for the particular sequence of instances, the baseline is not $(\gamma, 1)$-consistent unless $\gamma > 0.8$. In summary, our simulations indicate that the performance of CFTL is comparable with that of the baseline even when $\gamma > \eta$, and consistency does not considerably slow down learning.



c \ η	0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.8
0.01	437	316	221	147	93	55	29	13
0.1	373	265	180	116	70	39	19	7
0.2	310	214	141	87	50	25	10	3
0.4	206	133	81	45	21	8	2	0
0.8	76	41	19	7	1	0	0	0
1	43	20	7	2	0	0	0	0

Figure 2: Percentage of increase in MSE as the result of imposing pairwise constraints for different values of c and η.

6 Conclusion

In this work, we presented a natural model of fairness-aware sequential decision making. We showed that imposing time-dependant consistency constraints on the sequence of predicted labels does not significantly affect the speed of learning. Interesting directions for future work include, but are not limited to:

- *Alternative definitions of consistency:* In order for learning to be possible, any viable definition of time-consistency has to limit the impact of decisions made in distant past on future ones. We fulfilled this by taking K to be finite. Other, more complicated modeling choices (e.g. introducing a discount factor) are imaginable.

- *Alternative definitions of fairness:* Our focus in this work was on preventing cases of disparate treatment through "treating most recent similarly-situated individuals similarly". We leave the study of bounding other notions of individual fairness (e.g. algorithmic inequality [Speicher *et al.*, 2018]) among most recent observations as an open direction for future work.

Acknowledgments

This work by supported by CTI grant no. 27248.1 PFES-ES.

References

[Angwin *et al.*, 2016] Julia Angwin, Jeff Larson, Surya Mattu, and Lauren Kirchner. Machine bias. *Propublica*, 2016.

[Barocas and Selbst, 2016] Solon Barocas and Andrew D. Selbst. Big data's disparate impact. *California Law Review*, 2016.

[Barry-Jester *et al.*, 2015] Anna Barry-Jester, Ben Casselman, and Dana Goldstein. The new science of sentencing. *The Marshall Project*, August 2015. Retrieved 4/28/2016.

[Berk *et al.*, 2017] Richard Berk, Hoda Heidari, Shahin Jabbari, Michael Kearns, and Aaron Roth. Fairness in criminal justice risk assessments: The state of the art. *arXiv preprint arXiv:1703.09207*, 2017.

[Corbett-Davies *et al.*, 2017] Sam Corbett-Davies, Emma Pierson, Avi Feller, Sharad Goel, and Aziz Huq. Algorithmic decision making and the cost of fairness. *arXiv preprint arXiv:1701.08230*, 2017.

[Dheeru and Karra Taniskidou, 2017] Dua Dheeru and Efi Karra Taniskidou. UCI machine learning repository, 2017.

[Dwork *et al.*, 2012] Cynthia Dwork, Moritz Hardt, Toniann Pitassi, Omer Reingold, and Richard Zemel. Fairness through awareness. In *Proceedings of the Innovations in Theoretical Computer Science Conference*, pages 214–226. ACM, 2012.

[Feldman *et al.*, 2015] Michael Feldman, Sorelle A Friedler, John Moeller, Carlos Scheidegger, and Suresh Venkatasubramanian. Certifying and removing disparate impact. In *Proceedings of the International Conference on Knowledge Discovery and Data Mining*, pages 259–268. ACM, 2015.

[Hardt *et al.*, 2016] Moritz Hardt, Eric Price, and Nati Srebro. Equality of opportunity in supervised learning. In *Proceedings of Advances in Neural Information Processing Systems*, pages 3315–3323, 2016.

[Jabbari *et al.*, 2017] Shahin Jabbari, Matthew Joseph, Michael Kearns, Jamie Morgenstern, and Aaron Roth. Fairness in reinforcement learning. In *Proceedings of the International Conference on Machine Learning*, pages 1617–1626, 2017.

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

[Joseph *et al.*, 2017] Matthew Joseph, Michael Kearns, Jamie Morgenstern, Seth Neel, and Aaron Roth. Fair algorithms for infinite contextual bandits. *arXiv preprint arXiv:1610.09559*, 2017.

[Kamishima *et al.*, 2012] Toshihiro Kamishima, Shotaro Akaho, Hideki Asoh, and Jun Sakuma. Fairness-aware classifier with prejudice remover regularizer. *Machine Learning and Knowledge Discovery in Databases*, pages 35–50, 2012.

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

[Levin, 2016] Sam Levin. A beauty contest was judged by AI and the robots didn't like dark skin. *The Guardian*, 2016.

[Lipton *et al.*, 2017] Zachary C. Lipton, Alexandra Chouldechova, and Julian McAuley. Does mitigating ml's disparate impact require disparate treatment? *arXiv preprint arXiv:1711.07076*, 2017.

[Mann and O'Neil, 2016] Gideon Mann and Cathy O'Neil. Hiring algorithms are not neutral. *Harvard Business Review*, 2016.

[Miller, 2015] Clair Miller. Can an algorithm hire better than a human? *The New York Times*, June 25 2015. Retrieved 4/28/2016.

[Pedreschi *et al.*, 2008] Dino Pedreschi, Salvatore Ruggieri, and Franco Turini. Discrimination-aware data mining. In *Proceedings of the International Conference on Knowledge Discovery and Data Mining*, pages 560–568. ACM, 2008.

[Petrasic *et al.*, 2017] Kevin Petrasic, Benjamin Saul, James Greig, and Matthew Bornfreund. Algorithms and bias: What lenders need to know. *White & Case*, 2017.

[Podesta *et al.*, 2014] John Podesta, Penny Pritzker, Ernest Moniz, John Holdren, and Jeffrey Zients. Big Data: Seizing Opportunities, Preserving Values. *Executive Office of the President. The White House.*, 2014.

[Speicher *et al.*, 2018] Till Speicher, Hoda Heidari, Nina Grgic-Hlaca, Krishna P. Gummadi, Adish Singla, Adrian Weller, and Muhammad Bilal Zafar. A unified approach to quantifying algorithmic unfairness: Measuring individual and group unfairness via inequality indices. In *Proceedings of the International Conference on Knowledge Discovery and Data Mining*, 2018.

[Sweeney, 2013] Latanya Sweeney. Discrimination in online ad delivery. *Queue*, 11(3):10, 2013.

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