Meritocratic Fairness for Infinite and Contextual Bandits

Matthew Joseph, Michael Kearns, Jamie Morgenstern, Seth Neel, and Aaron Roth
University of Pennsylvania

Abstract

We study fairness in linear bandit problems. Starting from the notion of meritocratic fairness introduced in Joseph et al. (2016), we carry out a more refined analysis of a more general problem, achieving better performance guarantees with fewer modelling assumptions on the number and structure of available choices as well as the number selected. We also analyze the previously-unstudied question of fairness in infinite linear bandit problems, obtaining instance-dependent regret upper bounds as well as lower bounds demonstrating that this instance-dependence is necessary. The result is a framework for meritocratic fairness in an online linear setting that is substantially more powerful, general, and realistic than the current state of the art.

1 Introduction

The problem of repeatedly making choices and learning from choice feedback arises in a variety of settings, including granting loans, serving ads, and hiring. Encoding these problems in a *bandit* setting enables one to take advantage of a rich body of existing bandit algorithms. UCB-style algorithms, for example, are guaranteed to yield no-regret policies for these problems.

Joseph et al. (2016), however, raises the concern that these no-regret policies may be *unfair*: in some rounds, they will choose options with lower expected rewards over options with higher expected rewards, for example choosing less qualified job applicants over more qualified ones. Consider a UCB-like algorithm aiming to hire all qualified applicants in every round. As time goes on, any no-regret algorithm must behave unfairly for a vanishing fraction of rounds, but the total number of *mistreated* people – in hiring, people who saw a less qualified job applicant hired in a round in which they themselves were not hired – can be large (see Figure 1).

Joseph et al. (2016) then design no-regret algorithms which minimize mistreatment and are fair in the following sense: their algorithms (with high probability) never at any round place higher selection probability on a less qualified applicant than on a more qualified applicant. However, their analysis assumes that there are k well-defined groups, each with its own mapping from features to expected rewards; at each round exactly one individual from each group arrives; and exactly one individual is chosen in each round. In the hiring setting, this equates to assuming that a company receives one job applicant from each group and must hire exactly one (rather than m or all qualified applicants) introducing an unrealistic element of competition and unfairness both between applicants and between groups.

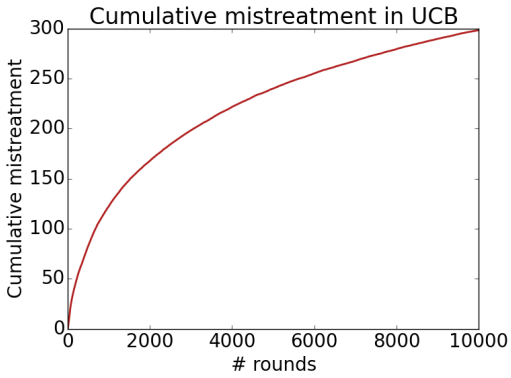

Figure 1: Cumulative mistreatments for UCB. See the experiments section in the full technical version for details and additional experimental evaluation of the structure of mistreatment.

The aforementioned assumptions are unrealistic in many practical settings; our work shows they are also *unnecessary*. Meritocratic fairness can be defined without reference to groups, and algorithms can satisfy the strictest form of meritocratic fairness without any knowledge of group membership. Even without this knowledge, we design algorithms which are fair with respect to *any* possible group structure over individuals. In Section 2, we present this general definition of fairness. The definition further allows for the number of individuals arriving in any round to vary, and is sufficiently flexible to apply to settings where algorithms can select $m \in [k]$ individuals in each round.Since the definition makes no reference to groups, the model makes no assumptions about how many individuals arriving at time t belong to any group. A company can then consider a large pool of applicants, not necessarily stratified by race or gender, with an

arbitrary number of candidates from any one of these populations, and hire one, m, or even every qualified applicant.

We then present a framework for designing meritocratically fair online linear contextual bandit algorithms. In Section 3, we show how to design fair algorithms to pick one of at most k individuals arriving in each round (the linear contextual bandits problem (Abe, Biermann, and Long, 2003; Auer, 2002)), as well as when m individuals may be chosen in each round ("multiple play" introduced and studied absent fairness in Anantharam, Varaiya, and Walrand (1987)). We therefore study a much more general model than Joseph et al. (2016) and, in Section 3, substantially improve upon their black-box regret guarantees for linear bandit problems using a technical analysis specific to the linear setting.

However, these regret bounds still scale (polynomially) with k, the maximum number of individuals seen in any given round. This may be undesirable for large k, thus motivating the investigation of fair algorithms for the *infinite* bandit setting (the online linear optimization with bandit feedback problem (Flaxman, Kalai, and McMahan, 2005)).[1] In Section 4 we provide such an algorithm via an adaptation of our general confidence interval-based framework that takes advantage of the fact that optimal solutions to linear programs must be *extreme points* of the feasible region. We then prove, subject to certain assumptions, a regret upper bound that depends on Δ_{gap}, an instance-dependent parameter based on the distance between the best and second-best extreme points in a given choice set.

In Section 5 we show that this instance dependence is almost tight by exhibiting an infinite choice set satisfying our assumptions for which *any* fair algorithm must incur regret dependent polynomially on Δ_{gap}, separating this setting from the online linear optimization setting absent a fairness constraint. In Section 6 we justify our assumptions on the choice set by exhibiting a set that both violates our assumptions and admits *no* fair algorithm with nontrivial regret guarantees. A condensed presentation of our methods and results appears in Figure 2.

Finally, we note that our algorithms share an overarching logic for reasoning about fairness. These algorithms all satisfy fairness by *certifying optimality*, never giving preferential treatment to x over y unless the algorithm is *certain* that x has higher reward than y. The algorithms accomplish this by computing confidence intervals around the estimated rewards for individuals. If two individuals have overlapping confidence intervals, we say they are *linked*; if x can be reached from y using a sequence of linked individuals, we say they are *chained*.

1.1 Related Work in Fairness

Fairness in machine learning has seen substantial recent growth as a subject of study, and many different definitions of fairness exist. We provide a brief overview here; see e.g. Berk et al. (2017) and Corbett-Davies et al. (2017) for detailed descriptions and comparisons of these definitions.

Many extant fairness notions are predicated on the existence of *groups*, and aim to guarantee that certain groups are not unequally favored or mistreated. In this vein, Hardt, Price, and Srebro (2016) introduced the notion of *equality of opportunity*, which requires that a classifier's predicted outcome should be independent of a protected attribute (such as race) conditioned on the true outcome, and they and Woodworth et al. (2017) have studied the feasibility and possible relaxations thereof. Similarly, Zafar et al. (2017) analyzed an equivalent concurrent notion of (un)fairness they call *disparate mistreatment*. Separately, Kleinberg, Mullainathan, and Raghavan (2017) and Chouldechova (2017) showed that different notions of group fairness may (and sometimes must) conflict with one another.

This paper, like Joseph et al. (2016), departs from the work above in a number of ways. We attempt to capture a particular notion of *individual* and *weakly meritocratic* fairness that holds *throughout the learning process*. This was inspired by Dwork et al. (2012), who suggest fair treatment equates to treating "similar" people similarly, where similarity is defined with respect to an assumed pre-specified task-specific metric. Taking the fairness formulation of Joseph et al. (2016) as our starting point, our definition of fairness does not promise to correct for past inequities or inaccurate or biased data. Instead, it assumes the existence of an accurate mapping from features to true quality for the task at hand[2] and promises fairness while learning and using this mapping in the following sense: any *individual* who is currently more qualified (for a job, loan, or college acceptance) than another individual will always have at least as good a chance of selection as the less qualified individual.

The one-sided nature of this guarantee, as well as its formulation in terms of quality, leads to the name *weakly meritocratic* fairness. Weakly meritocratic fairness may then be interpreted as a minimal guarantee of fairness: an algorithm satisfying our fairness definition cannot favor a worse option but is not required to favor a better option. In this sense our fairness requirement encodes a necessary variant of fairness rather than a completely sufficient one. This makes our upper bounds (Sections 3 and 4) relatively weaker and our lower bounds (Sections 5 and 6) relatively stronger.

We additionally note that our fairness guarantees require fairness *at every step of the learning process*. We view this as an important point, especially for algorithms whose learning processes may be long (or even continuous). Furthermore, while it may seem reasonable to relax this requirement to allow a small fraction of unfair steps, it is unclear how to do so without enabling discrimination against a correspondingly small population.

Finally, while our fairness definition draws from Joseph et al. (2016), we work in what we believe to be a significantly more general and realistic setting. In the finite case we allow for a variable number of individuals in each round from a variable number of groups and also allow selection of a variable number of individuals in each round, thus dropping

[1] We note that both the finite and infinite settings have infinite numbers of potential candidates: the difference arises in how many choices an algorithm has in a given round.

[2] Friedler, Scheidegger, and Venkatasubramanian (2016) provide evidence that providing fairness from bias-corrupted data is quite difficult.

# selected each round	# options each round	Technique	Notes	Regret
Exactly $j \leq k$	$\leq k$	Play all of chains in descending order, randomizing over last chain as necessary to pick exactly j	Requires randomness	$\tilde{O}\left(dkj\sqrt{T}\right)$
Unconstrained	$\leq k$	Select all in every chain with highest UCB > 0	Deterministic	$\tilde{O}\left(dk^2\sqrt{T}\right)$
Exactly 1	∞ bounded convex set $\Delta_{\text{gap}} > 0$	Play uniquely best point or UAR from entire set	Requires randomness	$\tilde{O}\left(c \cdot \log(T)/\Delta_{\text{gap}}^2\right)$ $\tilde{\Omega}(1/\Delta_{\text{gap}})$ $\Omega(T)$ for $\Delta_{\text{gap}} = 0$

Figure 2: Settings in which our framework provides fair algorithms. In all cases, fairness can be imposed only across pairs for any partitioning of the input space; the bounds here assume they bind across all pairs, and are thus worst-case upper bounds. See Section 4 for a full explanation of the distribution-dependent constant c in the regret bound for the infinite case.

several assumptions from Joseph et al. (2016). We also analyze the previously unstudied topic of fairness with infinitely many choices.

2 Model

Fix some $\beta \in [-1, 1]^d$, the underlying linear coefficients of our learning problem, and T the number of rounds. For each $t \in [T]$, let $C_t \subseteq D = [-1, 1]^d$ denote the set of available choices in round t. We will consider both the "finite" action case, where $|C_t| \leq k$, and the infinite action case. An algorithm $\mathcal{A}$, facing choices C_t, picks a subset $P_t \subseteq C_t$, and for each $x_t \in P_t$, $\mathcal{A}$ observes reward $y_t \in [-1, 1]$ such that $\mathbb{E}[y_t] = \langle \beta, x_t \rangle$, and the distribution of the noise $\eta_t = y_t - \langle \beta, x_t \rangle$ is sub-Gaussian.

Refer to all observations in round t as $Y_t \in [-1, 1]^{|P_t|}$ where $Y_{t,i} = y_{t,i}$ for each $x_{t,i} \in P_t$. Finally, let $\mathbf{X}_t = [X_1; \ldots; X_t], \mathbf{Y}_t = [Y_1; \ldots; Y_t]$ refer to the design and observation matrices at round t.

We are interested in settings where an algorithm may face size constraints on P_t. We consider three cases: the standard linear bandits problem ($|P_t| = 1$), the multiple choice linear bandits problem ($|P_t| = m$), and the heretofore unstudied (to the best of the authors' knowledge) case in which the size of P_t is unconstrained. For short, we refer to these as 1-bandit, m-bandit, and k-bandit.

Regret The notion of regret we will consider is that of pseudo-regret. Facing a sequence of choice sets $C_1, \ldots, C_T$, suppose $\mathcal{A}$ chooses sets $P_1, \ldots, P_T$.[3] Then, the expected reward of $\mathcal{A}$ on this sequence is $\text{Rew}(\mathcal{A}) = \mathbb{E}\left[\sum_{t \in [T]} \left[\sum_{x_t \in P_t} y_t\right]\right]$.

Refer to the sequence of feasible choices[4] which maximizes expected reward as $P_{*,1} \subseteq C_1, \ldots, P_{*,T} \subseteq C_T$, defined with full knowledge of β.

Then, the **pseudo-regret** of $\mathcal{A}$ on a sequence is defined as

$$\text{Rew}(P_{*,1}, \ldots, P_{*,T}) - \text{Rew}(\mathcal{A}) = R(T).$$

The **pseudo-regret** of $\mathcal{A}$ refers to the maximum pseudo-regret $\mathcal{A}$ incurs on any sequence of choice sets and any $\beta \in [-1, 1]^d$. If $R(T) = o(T)$, then $\mathcal{A}$ is said to be **no-regret**. If, for any input parameter $\delta > 0$, $R(T)$ upper-bounds the expectation of the rewards of the sequence chosen by $\mathcal{A}$ with probability $1 - \delta$, then we call this a *high-probability* regret bound for $\mathcal{A}$.

Fairness Consider an algorithm $\mathcal{A}$, which chooses a sequence of *probability distributions* $\pi_1, \pi_2, \ldots, \pi_T$ over feasible sets to pick, $\pi_t \in \Delta(2^{C_t})$. Note that distribution π_t depends upon $C_1, \ldots, C_t$, the choices $P_1, \ldots, P_{t-1}$, and $Y_1, \ldots, Y_{t-1}$.

We now give a formal definition of fairness of an algorithm for the 1-bandit, m-bandit, and k-bandit problems. We adapt our fairness definition from Joseph et al. (2016), generalizing from discrete distributions over finite action sets to mixture distributions over possibly infinite action sets. We slightly abuse notation and refer to the probability density and mass functions of an element $x \in C_t$: this refers to the marginal distribution of x being chosen (namely, the probability that x belongs to the set picked according to the distribution π_t).

Definition 1 (Weakly Meritocratic Fairness)**.** We say that an algorithm $\mathcal{A}$ is *weakly meritocratic* if, for any input $\delta \in (0, 1]$ and any β, with probability at least $1 - \delta$, at every round t, for every $x, x' \in C_t$ such that $\langle \beta, x \rangle \geq \langle \beta, x' \rangle$:

- If π_t is a discrete distribution: For $g_t(x) = \pi_t(x)$ (the probability mass function)
$$g_t(x) \geq g_t(x').$$
- If π_t is a continuous distribution: For $g_t(x) = f_t(x)$ (the probability density function)
$$g_t(x) \geq g_t(x').$$

[3] If these are randomized choices, the randomness of $\mathcal{A}$ is incorporated into the expected value calculations.

[4] We assume these have the appropriate size for each problem we consider: singletons in the 1-bandit problem, size at most m in the m-bandit problem, and arbitrarily large in the k-bandit problem.

- If π_t can be written as a mixture distribution: $\sum_i \alpha_i \pi_{ti}, \sum_i \alpha_i = 1$, such that each constituent distribution $\pi_{ti} \in \Delta(2^{C_t})$ is either discrete or continuous and satisfies one of the above two conditions.

For brevity, as we consider only this fairness notion in this paper, we will refer to weakly meritocratic fairness as "fairness". We say $\mathcal{A}$ is **round-fair** at time t if π_t satisfies the above conditions.

This definition can be easily generalized over any partition $\mathcal{G}$ of D, by requiring this weak monotonicity hold *only for pairs x, x' belonging to different elements of the partition G, G'*. The special case above of the singleton partition is the most stringent choice of partition. We focus our analysis on the singleton partition as a minimal worst-case framework, but this model easily relaxes to apply only across groups, as well as to only requiring "one-sided" monotonicity, where monotonicity is required only for pairs where the more qualified member belongs to group G rather than G'.

Remark 1. In the k-bandit setting, Definition 1 can be simplified to require, with probability $1 - \delta$ over its observations, an algorithm *never* select a less-qualified individual over more-qualified one in any round, and can be satisfied by deterministic algorithms.

3 Finite Action Spaces: Fair Ridge Regression

In this section, we introduce a family of fair algorithms for linear 1-bandit, m-bandit, and the (unconstrained) k-bandit problems. Here, an algorithm sees a slate of at most k distinct individuals each round and selects some subset of them for reward and observation. This lets us encode settings where an algorithm repeatedly observes a new pool of k individuals, each represented by a vector of d features, then decides to give some of those individuals loans based upon those vectors, observes the quality of the individuals to whom they gave loans, then updates the loan allocation rule. The regret of these algorithms scales polynomially in k and d as the algorithm gets tighter estimates of β.

All of the algorithms are based upon the following template. They maintain an estimate $\hat{\beta}_t$ of β from observations, along with confidence intervals around the estimate. They use $\hat{\beta}_t$ to estimate the rewards for the individuals on day t and the confidence interval around $\hat{\beta}_t$ to create a confidence interval around each of these estimated rewards.

Any two individuals whose intervals overlap on day t will be picked with the same probability by the algorithm. Call any two individuals whose intervals overlap on day t *linked*, and any two individuals belonging to the transitive closure of the linked relation *chained*. Since any two linked individuals will be chosen with the same probability, any two chained individuals will also be chosen with the same probability.

An algorithm constrained to pick exactly $m \in [k]$ individuals each round will pick them in the following way. Order the chains by their highest upper confidence bound. In that order, select all individuals from each chain (with probability 1), while that results in taking fewer than m individuals. When the algorithm arrives at the first chain for which it does not have capacity to accept every individual in the chain, it fills its remaining capacity uniformly at random from that chain's individuals. If the algorithm can pick any number of individuals, it will pick all individuals chained to any individual with positive upper confidence bound. The full pseudocode for RIDGEFAIR$_m$ is given in Figure 3. We now present the regret guarantees for fair 1-bandit, m-bandit, and k-bandit using this framework.

Theorem 1. *Suppose, for all t, η_t is 1-sub-Gaussian, $C_t \subseteq [-1, 1]^d$, and $||x_t||_2 \leq 1$ for all $x_t \in C_t$, and $||\beta|| \leq 1$. Then,* RIDGEFAIR$_1$, RIDGEFAIR$_m$, *and* RIDGEFAIR$_{\leq k}$ *are fair algorithms for the 1-bandit, m-bandit, and k-bandit problems, respectively. With probability $1 - \delta$, for $j \in \{1, m, k\}$, the regret of* RIDGEFAIR$_j$ *is*

$$R(T) = O\left(dkj\sqrt{T}\log\left(\frac{T}{\delta}\right)\right) = \tilde{O}(dkj\sqrt{T}).$$

We pause to compare our bound for 1-bandit to that found in Joseph et al. (2016). Their work supposes that each of k groups has an independent d-dimensional linear function governing its reward and provides a fair algorithm regret upper bound of $\tilde{O}\left(\min\{T^{\frac{4}{5}}k^{\frac{6}{5}}d^{\frac{3}{5}}, k^3\}\right)$. To directly encode this setting in ours, one would need to use a single dk-dimensional linear function, yielding a regret bound of $\tilde{O}\left(dk^2\sqrt{T}\right)$. This is an improvement on their upper bound for all values of T for which the bounds are nontrivial (recalling that the bound from Joseph et al. (2016) becomes nontrivial for $T > d^3k^6$, while the bound here becomes nontrivial for $T > d^2k^4$). We also briefly observe that RIDGEFAIR$_{\leq k}$ satisfies an additional "fairness" property: with high probability, it always selects *every* available individual with positive expected reward.

Each of these algorithms will use ℓ_2-regularized least-squares regressor to estimate β. Given a design matrix $\mathbf{X}$, response vector $\mathbf{Y}$, and regularization parameter $\gamma \geq 1$ this is of the form $\hat{\beta} = (\mathbf{X}^T\mathbf{X} + \gamma I)^{-1}\mathbf{X}^T\mathbf{Y}$. Valid confidence intervals (that contain β with high probability) are nontrivial to derive for this estimator (which might be biased); to construct them, we rely on martingale matrix concentration results (Abbasi-Yadkori, Pál, and Szepesvári, 2011).

We now sketch the proof of Theorem 1 (the full proof of this and all other results are in the full technical version of this paper). We first establish that, with probability $1 - \delta$, for all rounds t, for all $x_{t,i} \in C_t$, that $y_{t,i} \in [\ell_{t,i}, u_{t,i}]$ (i.e. that the confidence intervals being used are valid). Using this fact, we establish that the algorithm is fair. The algorithm plays any two actions which are linked with equal probability in each round, and any action with a confidence interval above another action's confidence interval with weakly higher probability. Thus, if the payoffs for the actions lie anywhere within their confidence intervals, RIDGEFAIR is fair, which holds as the confidence intervals are valid.

Proving a bound on the regret of RIDGEFAIR requires some non-standard analysis, primarily because the widths of the confidence intervals used by the algorithm do not shrink uniformly. The sum of the widths of the intervals of our *selected* (and therefore observed) actions grows sublinearly in

procedure RIDGEFAIR$_m(\delta, T, k, \gamma \geq 1$, ExactBool)
  for $t \geq 1, 1 \leq i \leq k$ **do**
    Let $\mathbf{X}_t, \mathbf{Y}_t$ = design matrix, observed payoffs before round t
    Let C_t be the choice set in round t
    Let $\bar{V}_t = {\mathbf{X}_t}^T \mathbf{X}_t + \gamma I$
    Let $\hat{\beta}_t = (\bar{V}_t)^{-1} {\mathbf{X}_t}^T \mathbf{Y}_t$ ▷ regularized LSE
    Let $\hat{y}_{t,i} = \langle \hat{\beta}_t, x_{t,i} \rangle$ for each $x_{t,i} \in C_t$
    Let $w_{t,i} = ||x_{t,i}||_{(\bar{V}_t)^{-1}} (\sqrt{2d \log(\frac{1+t/\gamma}{\delta})} + \sqrt{\gamma})$
    Let $[\ell_{t,i}, u_{t,i}] = [\hat{y}_{t,i} - w_{t,i}, \hat{y}_{t,i} + w_{t,i}]$
      ▷ Conf. int. for $\hat{y}_{t,i}$
    if ExactBool **then**
      PICK $(m, \{(x_{t,i}, [\ell_{t,i}, u_{t,i}])\})$
    else PICK$_\leq$ $(m, \{(x_{t,i}, [\ell_{t,i}, u_{t,i}])\})$
    $\mathbf{X}_{t+1} = \mathbf{X}_t :: X_t, \mathbf{Y}_{t+1} = \mathbf{Y}_t :: Y_t$.
      ▷ Update design matrices
procedure PICK$(m, (x_{t,1}, [\ell_{t,1}, u_{t,1}]), \ldots, (x_{t,k}, [\ell_{t,k}, u_{t,k}]))$
  Let $M = C_t$
  Let $P_t = \emptyset$
  while $|P_t| < m$ **do**
    Let $x_{t,\hat{i}} = \text{argmax}_{x_{t,i} \in M} u_{t,i}$
      ▷ Highest UCB not yet selected
    Let S_t be the set of actions in C_t chained to $x_{t,\hat{i}}$
      ▷ Highest chain not yet selected
    if $|S_t| \leq m - |P_t|$ **then**
      $P_t = P_t \cup S_t$
        ▷ Take the chain with probability 1
      $M = M \setminus S_t$
    else
      Let Q_t be $m - |P_t|$ actions chosen UAR from S_t
      Let $P_t = P_t \cup Q_t$
      ▷ fill remaining capacity UAR from the chain
  Play P_t
procedure PICK$_\leq(m, (x_{t,1}, [\ell_{t,1}, u_{t,1}]), \ldots, (x_{t,k}, [\ell_{t,k}, u_{t,k}]))$
  Let $P_t = \{$all actions chained to any $x_{t,i} \in C_t : u_{t,i} > 0\}$
  Let $M = C_t$
  Let $P_t = \emptyset$
  while $|P_t| < m \wedge u_{t,x_{t,\hat{i}}} > 0$ for $x_{t,\hat{i}} = \text{argmax}_{x_{t,i} \in M} u_{t,i}$ **do**
    Let S_t be the set of actions in C_t chained to $x_{t,\hat{i}}$
      ▷ Highest chain not yet selected
    if $|S_t| \leq m - |P_t|$ **then**
      $P_t = P_t \cup S_t$
        ▷ Take the chain with probability 1
      $M = M \setminus S_t$
    else
      Let Q_t be $m - |P_t|$ actions chosen UAR from S_t
      Let $P_t = P_t \cup Q_t$
      ▷ fill remaining capacity UAR from the chain
  Play P_t

Figure 3: RIDGEFAIR$_m$, a fair no-regret algorithm for picking $\leq m$ actions whose payoffs are linear.

t. UCB variants, by virtue of playing an action a with highest upper confidence bound, have regret in round t bounded by a's confidence interval width. RIDGEFAIR, conversely, suffers regret equal to the *sum* of the confidence widths of the chained set, while only receiving feedback for the action it actually takes. We overcome this obstacle by relating the sum of the confidence interval widths of the linked set to the sum of the widths of the selected actions.

4 Fair algorithms for convex action sets

In this section we analyze linear bandits with infinite choice sets in the 1-bandit setting.[5] We now provide a fair algorithm with an instance-dependent sublinear regret bound for infinite convex choice sets; Section 5 shows that instance dependence is necessary for fair algorithms in an infinite setting.

A naive adaptation of RIDGEFAIR to an infinite setting requires maintenance of infinitely many confidence intervals and is therefore impractical. We instead assume that our choice sets are convex bodies and exploit the resulting geometry: since our underlying function is linear, it is maximized at an *extremal* point. This simplifies the problem, since we need only reason about the relative quality of extremal points. The relevant quantity is Δ_{gap}, a notion adapted from Dani, Hayes, and Kakade (2008) that denotes the difference in reward between the best and second-best extremal points in the choice set. When Δ_{gap} is large we can identify the optimal choice more quickly, then select it deterministically without violating fairness. When Δ_{gap} is small, we need more observations to determine which of the top two points is best – and before we make this determiniation, deterministically selecting any action violates fairness for any points infinitesimally close to the true best point (and we are forced to play randomly from the entire choice set).

Our resulting fair algorithm, FAIRGAP, proceeds as follows: in each intervals around the two extreme points with highest estimated reward round it uses its current estimate of β to construct confidence and selects the higher one if these intervals do not overlap; otherwise, it selects uniformly at random from the entire convex body. We prove fairness and bound regret by analyzing the rate at which random exploration shrinks our confidence intervals and relating it to the frequency of exploitation, a function of Δ_{gap}, defined below.

Definition 2 (Gap, adapted from Dani, Hayes, and Kakade (2008))**.** Given sequence of action sets $C = (C_1, \ldots, C_T)$, define Ω_t to be the set of extremal points of C_t, i.e. the points in C_t that cannot be expressed as a proper convex combination of other points in C_t, and let $x_t^* = \max_{x \in C_t} \langle \beta, x \rangle$. The *gap* of C_t is

$$\Delta_{\text{gap}} = \min_{1 \leq t \leq T} \left(\inf_{x_t \in \Omega_t, x_t \neq x_t^*} \langle \beta, x_t^* - x_t \rangle \right).$$

Δ_{gap} is a lower bound on difference in payoff between the optimal action and any other extremal action in any C_t.

[5] Note that no-regret guarantees are in general impossible for infinite choice sets in m-bandit and k-bandit settings, since the continuity of the infinite choice sets we consider makes selecting multiple choices while satisfying fairness impossible without choosing uniformly at random from the entire set.

When $\Delta_{\text{gap}} > 0$, this implies the existence of a unique optimal action in each C_t. Our algorithm (implicitly) and our analysis (explicitly) exploits this quantity: a larger gap enables us to confidently identify the optimal action more quickly. We now present the regret and fairness guarantees for FAIRGAP.

Theorem 2. *Given sequence of action sets $C = (C_1, \ldots, C_T)$ where each C_t has nonzero Lebesgue measure and is contained in a ball of radius r and feedback with R-sub-Gaussian noise,* FAIRGAP *is fair and achieves*

$$\text{REGRET}\,(T) = O\left(\frac{r^6 R^2 \ln(2T/\delta)}{\kappa^2 \lambda^2 \Delta_{gap}^2}\right)$$

where $\kappa = 1 - r\sqrt{\frac{2\ln\left(\frac{2dT}{\delta}\right)}{T\lambda}}$ *and* $\lambda = \min_{1\le t\le T}\left[\lambda_{\min}(\mathbb{E}_{x_t\sim_{UAR}C_t}[x_t {x_t}^T])\right]$

A full proof of FAIRGAP's fairness and regret bound, as well as pseudocode, appears in the full technical version. We sketch the proof here: our proof of fairness proceeds by bounding the influence of noise on the confidence intervals we construct (via matrix Chernoff bounds) and proving that, with high probability, FAIRGAP constructs correct confidence intervals. This requires reasoning about the spectrum of the covariance matrix of each choice set, which is governed by λ, a quantity which, informally, measures how quickly we learn from uniformly random actions. [6]. With correct confidence intervals in hand, fairness follows almost immediately, and to bound regret we analyze the rate at which these confidence intervals shrink.

The analysis above implies identical regret and fairness guarantees when each C_t is finite. For comparison, the results of Section 3 guarantee $\text{REGRET}\,(T) = O(dk\sqrt{T})$. This result, in comparison, enjoys a regret independent of k which is especially useful in cases with large k.

Finally, our analysis so far has elided any computational efficiency issues arising from sampling randomly from C. We note that it is possible to circumvent this issue by relaxing our definition of fairness to *approximate fairness* and obtain similar regret bounds for an efficient implementation. We achieve this using results from the broad literature on sampling and estimating volume in convex bodies, as well as recent work on finding "2nd best" extremal solutions to linear programs. Full details appear in the appendix of the full technical version.

5 Instance-dependent Lower Bound for Fair Algorithms

We now present a lower bound instance for which any fair algorithm *must* suffer gap-dependent regret. More formally, we show that when each choice set is a square, i.e. $C_t = [0,1]^2$ for all t, for any fair algorithm $\text{REGRET}\,(T) = \tilde{\Omega}(1/\Delta_{\text{gap}})$ with probability at least $1-\delta$. This also implies the weaker result that no fair algorithm enjoys an instance-independent sub-linear regret bound $o(T)$ holding uniformly over all β. We therefore obtain a clear separation between fair learning and the unconstrained case (Dani, Hayes, and Kakade, 2008), and show that an instance-dependent upper bound like the one in Section 4 is unavoidable. Our arguments establish fundamental constraints on fair learning with large choice sets and quantify through the Δ_{gap} parameter how choice set geometry can affect the performance of fair algorithms. The lower bound employs a Bayesian argument resembling that in Joseph et al. (2016) but with a novel "chaining" argument suited to infinite action sets; we defer its proof to the full technical version of this paper.

Theorem 3. *For all t let $C_t = [-1,1]^d$, $\beta \in [-1,1]^d$, and $y_t = \langle x_t, \beta\rangle + \eta_t$, where $\eta_t \sim U[-1,1]$. Let $\mathcal{A}$ be any fair algorithm. Then for every gap Δ_{gap}, there is a distribution over instances with gap $\Omega(\Delta_{gap})$ such that any fair algorithm has regret $\text{REGRET}\,(T) = \tilde{\Omega}(1/\Delta_{gap})$ with probability $1-\delta$.*

We note that this impossibility result only holds for $d \ge 2$. When $d = 1$, the choice set reduces to $[-1,1]$, and similarly $\beta \in [-1,1]$. Thus, the optimal action is $\text{sign}(\beta)$). It takes $O(1/\beta^2)$ observations to determine the sign of β. A fair algorithm may play randomly from $[-1,1]$ until it has determined $\text{sign}(\beta)$, and then play $\text{sign}(\beta)$ for every round thereafter. As the maximum per-round regret of any action is $O(\beta)$, and because the maximum cumulative regret obtained by the algorithm is with high probability $O(\beta \cdot 1/\beta^2) = O(1/\beta)$, the regret of this simple algorithm over T rounds is $O(\min(\beta \cdot T, 1/\beta^2))$. Taking the worst case over β, we see that this quantity is bounded uniformly by $O(\sqrt{T})$, a sublinear parameter independent regret bound.

6 Zero Gap: Impossibility Result

Section 4 presents an algorithm for which the sublinear regret bound has dependence $1/\Delta_{\text{gap}}^2$ on the instance gap. Section 5 exhibits an choice set C with a $\tilde{\Omega}(1/\Delta_{\text{gap}})$ dependence on the gap parameter. We now exhibit a choice set C for which $\Delta_{\text{gap}} = 0$ for every β, and for which no fair algorithm can obtain non-trivial regret for any value of β. This precludes even instance-dependent fair regret bounds on this action space, in sharp contrast with the unconstrained bandit setting.

Theorem 4. *For all t let $C_t = S^1$, the unit circle, and $\eta_t \sim$ Unif$(-1,1)$. Then for any fair algorithm $\mathcal{A}$, $\forall \beta \in S^1, \forall T \ge 1$, we have*

$$\mathbb{E}_\beta[\text{REGRET}\,(T)] = \Omega(T).$$

S^1 makes fair learning difficult for the following reasons: since S^1 has no extremal points, there is no finite set of points which for any β contains the uniquely optimal action, and for any point in S^1, and any finite set of observations, there is another point in S^1 for which the algorithm cannot confidently determine relative reward. Since this property holds for *every* point, the fairness constraint transitively requires that the algorithm play every point uniformly at random, at every round. The formal argument again relies on a Bayesian analysis of chaining, as well as a basic fact about the topology of S^1.

[6] λ can be computed for finite C_t or approximated by any positive lower bound for infinite C_t and substituted into our results.

References

Abbasi-Yadkori, Y.; Pál, D.; and Szepesvári, C. 2011. Improved algorithms for linear stochastic bandits. In *Advances in Neural Information Processing Systems*, 2312–2320.

Abe, N.; Biermann, A. W.; and Long, P. M. 2003. Reinforcement learning with immediate rewards and linear hypotheses. *Algorithmica* 37(4):263–293.

Anantharam, V.; Varaiya, P.; and Walrand, J. 1987. Asymptotically efficient allocation rules for the multiarmed bandit problem with multiple plays – part i: I.i.d. rewards. *IEEE Transactions on Automatic Control* AC-32(Nov):968–976.

Auer, P. 2002. Using confidence bounds for exploitation-exploration trade-offs. *Journal of Machine Learning Research* 3(Nov):397–422.

Berk, R.; Heidari, H.; Jabbari, S.; Kearns, M.; and Roth, A. 2017. Fairness in criminal justice risk assessments: The state of the art. *arXiv preprint arXiv:1703.09207*.

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

Corbett-Davies, S.; Pierson, E.; Feller, A.; Goel, S.; and Huq, A. 2017. Algorithmic decision making and the cost of fairness. *arXiv preprint arXiv:1701.08230*.

Dani, V.; Hayes, T. P.; and Kakade, S. M. 2008. Stochastic linear optimization under bandit feedback. In *COLT*, 355–366.

Dwork, C.; Hardt, M.; Pitassi, T.; Reingold, O.; and Zemel, R. 2012. Fairness through awareness. In *Proceedings of ITCS 2012*, 214–226. ACM.

Flaxman, A. D.; Kalai, A. T.; and McMahan, H. B. 2005. Online convex optimization in the bandit setting: gradient descent without a gradient. In *Proceedings of the sixteenth annual ACM-SIAM symposium on Discrete algorithms*, 385–394. Society for Industrial and Applied Mathematics.

Friedler, S. A.; Scheidegger, C.; and Venkatasubramanian, S. 2016. On the (im)possibility of fairness. In *arXiv*, volume abs/1609.07236.

Hardt, M.; Price, E.; and Srebro, N. 2016. Equality of opportunity in supervised learning. In *Advances in Neural Information Processing Systems 29: Annual Conference on Neural Information Processing Systems 2016, December 5-10, 2016, Barcelona, Spain*, volume abs/1610.02413.

Joseph, M.; Kearns, M.; Morgenstern, J. H.; and Roth, A. 2016. Fairness in learning: Classic and contextual bandits. In *Advances in Neural Information Processing Systems*, 325–333.

Kleinberg, J.; Mullainathan, S.; and Raghavan, M. 2017. Inherent trade-offs in the fair determination of risk scores. In *ITCS*.

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

Zafar, M. B.; Valera, I.; Rodriguez, M. G.; and Gummadi, K. P. 2017. Fairness beyond disparate treatment and disparate impact: Learning classification without disparate mistreatment. In *Proceedings of World Wide Web Conference*.