Fairness Is Not Static: Deeper Understanding of Long Term Fairness via Simulation Studies

Alexander D'Amour*
Google Research
alexdamour@google.com

Hansa Srinivasan*
Google Research
hansas@google.com

James Atwood
Google Research
atwoodj@google.com

Pallavi Baljekar
Google Research
pbaljeka@google.com

D. Sculley
Google Research
dsculley@google.com

Yoni Halpern
Google Research
yhalpern@google.com

ABSTRACT

As machine learning becomes increasingly incorporated within high impact decision ecosystems, there is a growing need to understand the *long-term* behaviors of deployed ML-based decision systems and their potential consequences. Most approaches to understanding or improving the fairness of these systems have focused on static settings without considering long-term dynamics. This is understandable; long term dynamics are hard to assess, particularly because they do not align with the traditional supervised ML research framework that uses fixed data sets. To address this structural difficulty in the field, we advocate for the use of simulation as a key tool in studying the fairness of algorithms. We explore three toy examples of dynamical systems that have been previously studied in the context of fair decision making for bank loans, college admissions, and allocation of attention. By analyzing how learning agents interact with these systems in simulation, we are able to extend previous work, showing that static or single-step analyses do not give a complete picture of the long-term consequences of an ML-based decision system. We provide an extensible open-source software framework for implementing fairness-focused simulation studies and further reproducible research, available at https://github.com/google/ml-fairness-gym.

CCS CONCEPTS

• **Computing methodologies** → **Simulation environments**; **Supervised learning by classification**.

ACM Reference Format:
Alexander D'Amour, Hansa Srinivasan, James Atwood, Pallavi Baljekar, D. Sculley, and Yoni Halpern. 2020. Fairness Is Not Static: Deeper Understanding of Long Term Fairness via Simulation Studies. In *Conference on Fairness, Accountability, and Transparency (FAT* '20), January 27–30, 2020, Barcelona, Spain.* ACM, New York, NY, USA, 10 pages. https://doi.org/10.1145/3351095.3372878

*Both authors contributed equally to this research.

FAT '20, January 27–30, 2020, Barcelona, Spain*

ACM ISBN 978-1-4503-6936-7/20/02.
https://doi.org/10.1145/3351095.3372878

1 INTRO: BEYOND STATIC FAIRNESS

Much of the literature on fairness in machine learning is motivated by the concern that high impact decisions made by machine-learned systems may have negative consequences for vulnerable populations [23, e.g., reviewed in]. However, much of this prior work has focused on the fairness implications of decisions made in a static or one-shot context in which long-term effects and system level dynamics are not considered. Despite recent work that has shown that long-term implications can be at odds with fairness objectives optimized in the static setting [16, 21, 22], long-term implications remain relatively under-studied.

In this work, we propose that simulation studies can serve as a framework for systematically exploring the *long-term* implications of deploying a machine learning based decision system (henceforth, a *learning agent*). Simulation studies address a gap between currently-favored evaluation of fairness policies on static, real-world datasets, and more recent forays into simple, analytically tractable models of dynamics [16, 20–22]. Simulations allow access to counterfactual information about how the data would have varied if a different data collection or decision-making policy had been in place, a dimension that is missing from static datasets. For example, the COMPAS [25] and German Credit [5] datasets do not provide counterfactual information about how the data would have looked if a different data collection or decision-making policy had been in place. In addition, simulations allow for concrete exploration of system level dynamics, feedback loops, and other long-term effects that may be intractable to analyze in closed form or to demonstrate empirically in static settings.

To demonstrate the utility of simulations, we perform expanded analyses of three canonical scenarios that have been treated in previous fairness papers. In the first demonstration, we consider the dynamic lending scenario studied in Liu et al. [21], where the credit score of loan applicants can change in response to the agent's decision to lend or reject. Here, we perform a long-term analysis of a dynamic scenario where only short-term analyses had been performed before.

In the second demonstration, we consider the attention allocation problem presented in Ensign et al. [7] and Elzayn et al. [6], where an agent is tasked with allocating finite units of attention across different sites with the goal of discovering harmful incidents. Here, we add dynamics to this scenario, which was previously treated under an assumption of stationary rates.

Finally, we consider a college admissions scenario studied in Hu et al. [16] and Milli et al. [22], where applicants are able to manipulate their scores (at a cost) to obtain a desirable decision from the agent. Here, we examine equilibria that are realized from repeated play of a two-player game where previous work only considered the one-shot setting. In each of these scenarios, we find that the concrete, long-term view offered by simulation supports qualitatively different (though not incompatible) fairness conclusions from those obtained before.

1.1 Contributions

This work makes several contributions, with the goal of raising the profile of simulation studies in the ML fairness community.

Our primary contribution is to demonstrate in several settings how feedback dynamics complicate the analysis of long term fairness consequences. We show that, in each of these settings, the long-run dynamics depart substantially from those that are measured in one-shot contexts. Although these simulations may be too stylized to draw substantive conclusions about each of the decision problems that they model, they nonetheless demonstrate key complications in framing and measuring fairness in dynamic environments.

Along the way, we show that fairness-oriented simulations can fit into the standard framework of Markov Decision Processes (MDPs), which are commonly used in a number of subfields, including robotics and reinforcement learning. This framing is flexible, and puts dynamical analyses of fairness into a language that is more familiar to many ML researchers than the economic concepts highlighted in previous work.

Finally, along with code to reproduce the results in this paper, we provide a general library `ml-fairness-gym` for specifying new simulation environments and agents with a unified interface for easy extensibility and development at https://github.com/google/ml-fairness-gym.

1.2 Related Work

The simulation approach that we advocate for in this paper is meant to complement several lines of work in fair machine learning. First, simulation complements the bulk of the fairness literature that focuses on classification in static settings [e.g., 8, 14, 19, 29]. These static approaches to fairness are often evaluated on real datasets. This approach has a number of virtues, chief among them being a relevance to real problems, as opposed to the "toy" nature of simulated environments. However, as discussed above, the scope of fairness questions that can be addressed in the static setting is limited, so we believe that both sorts of investigations are valuable. Secondly, simulation can be used to expand analytical investigations into fairness with dynamics that have recently appeared in papers on lending [21], resource allocation [6, 7], and college admissions [16, 20, 22]. Finally, the simulation approach that we use here complements work in fair reinforcement learning [e.g., 17, 18]. In particular, the simulation framework that we propose would constitute an ideal testbed for these methods.

More broadly, this work is meant to take a small step toward incorporating greater social context in fairness analyses of machine learning systems. Social scientists have long drawn attention to the fact that decision-making technologies, and the concepts that underpin them, not only react to, but actively shape the social environment in which they are deployed. For example, in a long line of work, Ian Hacking identifies the "looping effects" [11] that result from categorizing people into "kinds" as an act of "making up people" [10, 12]. More recently, a number of critical evaluations of the algorithmic fairness literature draw attention to the ways that machine learning systems interact with, and often reinforce, the power structures that generate the inequities that this literature is meant to address [3, 15, 28]. By these accounts, notions of fairness and justice can only be addressed if this constructivist feedback is taken into account. Approaches such as systems dynamics [9, 30] have been proposed before to model and simulate some of these social dynamics. Aspirationally, despite the simplicity of the examples we present here, we hope our framework can extend these approaches in a way that allows practitioners to integrate their machine learning procedures and draw broad lessons about how they might interact with a social environment.

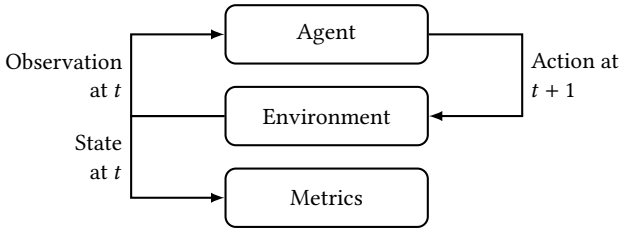

Figure 1: Schematic of agent-environment interaction loop used in simulations. The agent-environment loop is an MDP. The observation at t is a function of the environment's state at t. The environment's transition to the next state depends on its current state and the agent's action. The environment exposes its state to metrics for evaluation. In our implementation, metrics examine the history of environment states offline.

Our open-source library `ml-fairness-gym` is built on the OpenAI Gym framework [1] which simulates a Markov Decision Process (MDP) between *agents* and *environments*. Our choice of this framing is discussed in more detail in the next section.

2 A FRAMEWORK FOR SIMULATIONS: ENVIRONMENT, METRICS, AGENTS

To implement all of our simulations, we developed an open-source library `ml-fairness-gym` that extends OpenAI's Gym [1]. In the Gym framework, *agents* interact with simulated *environments* in an alternating loop. At each step, the agent chooses an action which affects the environment's state. The environment then reveals an *observation* and the agent, which then uses that observation to inform its next choice of action. This loop repeats indefinitely (Figure 1), or until the environment reaches an end state. In contrast to the traditional supervised machine learning framing that uses a training set of i.i.d. labeled examples and a test set drawn from the same distribution, in the agent-environment framework, every action by the agent affects the environment state. In this way, training and testing are interleaved, decisions can cause the population

to shift, and decisions at time T affect the decision to be made by the agent at time $T + 1$.

The agent-environment framing modeled as a Markov Decision Process (MDP) is common in robotics [e.g., 32] and reinforcement learning [31]. We choose to adopt it here because it naturally encodes the idea that the decisions (classifications) of the learner have *consequences* beyond those that can be summarized in terms of prediction error. In addition, the MDP framework expands the role of the learning algorithm to include choices made in data collection as well (the "dataset" of observations available to the agent depend on the agent's sequence of actions). This corresponds to a common setting in machine learning where a perfectly sampled dataset is not simply available for training, but the data is often accumulated through experience applying a particular policy (e.g., a bank learns how to identify good loan applicants through the process of lending).

We evaluate long-term fairness questions in our framework with *metrics* that characterize the realized consequences of an agent's policy for different subpopulations by summarizing the the environment's state over time. [1] This does not always paint a uniform picture (a policy can be good for a group in one way and bad in another), but gives a nuanced understanding of how policies play out. In general, the metrics for a given policy are most interpretable when compared to a baseline.

In one significant departure from the OpenAI Gym framework and the standard formulation of reinforcement learning, our environments do *not* encode a particular goal for the agent (this is often indicated by a "reward signal"). Instead, the designer of the *agent* is responsible for defining their own objective in relation to the observations from the environment. This decision reflects our belief that fair machine learning encompasses all parts of the design process including choosing appropriate goals in the first place.

3 LONG-TERM CONSEQUENCES IN BINARY DECISION MAKING: LENDING

For our first demonstration, we examine a setting where an agent makes binary decisions that cause the underlying population of individuals to evolve. In particular, we consider the lending scenario introduced in Liu et al. [21], where an agent representing a bank makes decisions about whether to approve or reject applications for loans from a stream of individuals. Liu et al. [21] explicate some one-step implications of policies that maximize profit for the bank, as well as policies that are subject to so-called equality of opportunity constraints [14]. [2] Here, we extend this analysis over many steps via simulation.

We highlight two main takeaways from this demonstration. First, despite being theoretically compatible, simulation and tightly-scoped analytical exploration can yield qualitatively different stories about the relative strengths and weaknesses of agent policies.

[1] We choose to focus on the realized consequences for evaluation because simulation adds the most value for these assessments. In the language of Hardt et al. [14], these are "oblivious" or "black-box" evaluations. For a more complete picture of fairness, other evaluations, such as process-based evaluations that probe the *agent's* internal state, may be desirable.

[2] Throughout this section we use the term "equality of opportunity" as in Hardt et al. [14] to narrowly refer to the constraint that a binary classifier should have equal true positive rates (TPR) across groups.

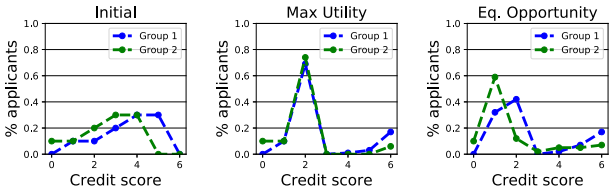

Figure 2: Initial credit score distributions of the two groups (far left) and final states after 20K steps of the environment using a max-util agent (center) and EO agent (right). The credit distributions start with group 2 slightly disadvantaged, but the groups converge to the similar distributions under the max-util agent, while the EO agent maintain unequal credit distributions between groups.

Figure 3: (Top) Group-conditional decision thresholds of the EO agent change with each step of the simulation. Fractional decision thresholds are achieved by sampling adjacent integral thresholds with appropriate probabilities (e.g., a threshold of 3.1 means sampling from 3, 4 with probabilities 0.9, 0.1 respectively). Large fluctuations in the learned threshold occur on steps where there are no applicants with score 4, allowing the threshold to move arbitrarily between 3 and 4 with no consequence. (Bottom) The recall values associated with the thresholds at each step.

In particular, the qualitative stories about the consequences of max-util and EO policies that emerge from our simulations are quite different from the stories that one might take from the results in Liu et al. [21]. Second, the simulation reveals that, when dynamics are introduced, agent policies and standard metrics can become misaligned. Here, we show that even when the EO agent equalizes true positive rates between groups at each time step, it will not, in general, equalize the standard true positive rate metric computed across the entire simulation, even in expectation. Although these insights could have been achieved through analytical approaches alone, they occurred initially as surprising (to us) simulation results that we were later able to characterize more formally.

3.1 Environment

We use the same lending environment specification as Liu et al. [21]. In this environment, each loan applicant has an observable group membership variable A and a discrete credit score $C \in 1, \cdots, C_{max}$.

There is a finite pool of loan applicants from each group with C values distributed according to an initial group-specific distribution $p_0^A(C)$. Applicants are sampled uniformly with replacement from the pool of applicants and the agent chooses to approve or decline the loan. If the applicant defaults, the agent's profit decreases by r_- and the applicant's C value is decreased by c_-. If the applicant pays back, the agent's profit is increased by r_+ and the applicant's C value is increased by c_+.

In this simulation, probability of repaying is a deterministic function of credit score $\pi(C)$; when an applicant's score increases or decreases, so, too, does their probability of repaying. Adding noise to this relationship could create a more nuanced simulation. If the noise were applied to all groups equivalently, it would not erase the underlying dynamics discussed here. However, other noise models, such as differential measurement error, could induce different interesting dynamics (see Section 4.2 of Liu et al. [21] for discussion of a model where disadvantaged group credit scores are systematically underestimated).

3.2 Metrics

To evaluate the consequences of deploying an agent we look at how it affects the average credit of each group, as measured by the overall change in credit score distributions, as well as changes in the group conditional probability of repayment, and the cumulative number of loans. We also evaluate the agent's aggregated true positive rate by computing the ratio of successful loans given to the number of applicants who would have repaid a loan over the course of the simulation.

3.3 Agents

We consider two agents. First, we consider an agent whose policy maximizes profits (without any future discounting) for the bank (*max-util agent*). Liu et al. [21] prove that such an agent will employ a threshold classifier that chooses some threshold τ, then deterministically accepts any applicant with $\pi(C) \geq \tau$ and rejects all others. The threshold depends only on the conditional probability of repayment given C, which is constant over time, and thus the max-util agent's decision policy remains fixed over the course of the simulation.

Second, we consider an agent that myopically maximizes utility subject to equality of opportunity [14] constraints at every step (*EO agent*). Specifically, the agent's decision rule is constrained to equalize the true positive rate (TPR) between the two groups. Because of the discrete scores in this setting, equalizing TPR between groups is not always feasible with a deterministic decision policy. Thus, we consider randomized policies that probabilistically interpolate between thresholds that represent adjacent points on the convex hull of the ROC curve. As discussed in Hardt et al. [14], while the max-util agent has a single threshold across all groups, the EO agent generally employs a different policy for each group.

Because it is constrained to equalize TPR, the EO agent's policy at each time step depends on the population distributions of the two groups (illustrated in appendix A.3). Thus, unlike the max-util agent, the EO agent's policy can change over the course of the simulation in response to the changing credit score distributions in each group of applicants.

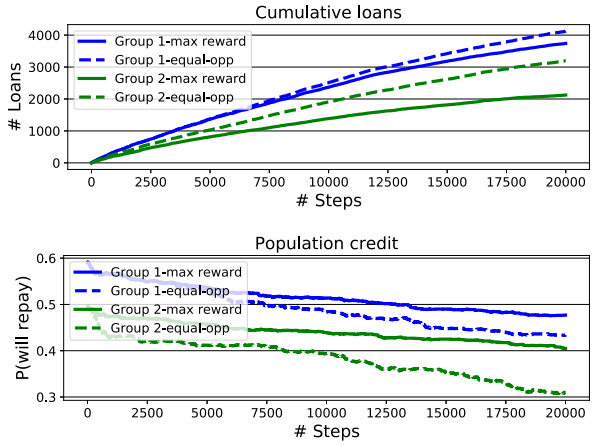

Figure 4: (Top) Cumulative loans granted by the max-util and EO agents stratified by the group identity of the applicant. (Bottom) Group credit (quantified by group-conditional probability of repayment) as the simulation progresses. The EO agent increases access to loans for group 2, but also widens the credit gap between the groups.

3.4 Experiments and results

We now describe a set of simulations in this environment that compare maximum utility agents to equality of opportunity agents. Some aspects of deploying a maximum utility agent are straightforward to understand analytically (Appendix A). However, the consequences of deploying an EO agent, as well as the long-term population-level effects of max-util and opportunity equalizing policies, are considerably more complex, and benefit from analysis via simulation. Our results tell a qualitatively different story from Liu et al. [21], and highlight a mismatch between the EO agent and the agent's aggregate true positive rate that does not arise in static settings.

We simulate a population with two groups with shifted credit score distributions such that group 2 starts out disadvantaged compared to group 1 (Figure 2). We simplify the EO agent's task and grant it oracle access to the population distribution and exact values of repayment probabilities π. This removes any difficulty of estimation allowing us to focus solely on the problem of making fair decisions.

Diverging narratives from one-step analysis. Figure 4 shows the primary outcomes from our experiment. Some of these results are surprising in light of the results in Liu et al. [21]. First, we arrive at similar conclusions about the impact of EO policies on group-wise credit scores, but find the welfare implications for the disadvantaged group to be more ambiguous than was originally suggested. Consistent with results in Liu et al. [21], Figure 4 shows that the EO agent "over-lends" to the disadvantaged group by, at

times, applying a lower decision threshold than the max-util agent (Figure 3). This results in a lower average credit trajectory for the disadvantaged group, and widens the credit-gap between the groups compared to the max-util agent. However, it is not clear that the EO agent leaves the disadvantaged group worse-off. As the top panel of Figure 4 shows, despite the poor credit score trajectory, the EO agent grants a larger number of loans to the disadvantaged group. Depending on context, one might take a group's overall credit, or the number of loans the group receives, to be the better indicator of the group's welfare. In any case, the simulation highlights the importance of considering this trade-off.

Second, we arrive at qualitatively different conclusions about the impact of the max-util agent. Specifically, in the bottom panel of Figure 4, the average credit of both groups is decreasing under the max-util agent, which seems to disagree with the result in Liu et al. [21] stating that, under max-util policies, group average credit scores are non-decreasing. In fact, these results are compatible, and this surface disagreemnt points to the fact that the analytical results do not cover all of the cases necessary to specify a full simulation. In particular, the anlaytical theory only covers circumstances where the institution's individual utility function is more stringent than the expected score changes [Assumption 1 in Section 3 of 21], and so the conclusions do not apply to cases where individuals already have maximum credit C_{max} and are unable to increase their score further. When such edge effects are present, applicants' scores must eventually drop below the agent's lending threshold, where they remain for the remainder of the simulation (see Appendix A). Here, the simulation highlights the fact that edge cases can flip the qualitative implications of theoretical results when they are transferred to real settings, even though the simulation itself is highly stylized.

EO agents and aggregated TPR. Interestingly, enforcing equal TPRs between groups at every step does not succeed at equalizing TPR when aggregating over the course of the full simulation (!). Figure 5 shows how the TPR-gap between the two groups does not converge to zero in the same way that it would in a static population where credit score does not change as a function of loan repayment. This counter-intuitive property can be thought of as an instance of the well-studied Simpson's paradox (See e.g., Ross [27] for an example of this phenomenon in calculating batting averages in baseball). In Appendix A.4, we show analytically why we would not expect equality of opportunity to be preserved in aggregate doing a simple two-step analysis. This finding suggests that, in dynamics environments where populations are shifting, estimates of aggregate TPR may not be useful for auditing agents that apply EO policies at each point in time.

4 DYNAMIC INCIDENT RATES IN ATTENTION ALLOCATION

Next, we consider the problem of *attention allocation*, in which an agent is tasked with spot checking or monitoring several sites for incidents, but does not have sufficient resources to do so exhaustively at every time step. Real world examples of this setting may include food inspection, child services, and pest control.[3] Our analysis extends the dynamic attention allocation settings considered in Ensign et al. [7] and Elzayn et al. [6]. In both of these papers, authors derive unbiased estimates of incident rates despite missing observations. Elzayn et al. [6] additionally propose an algorithm to allocate in a way that equalizes discovery probability from these rate estimates.

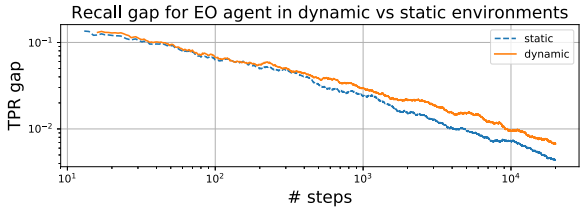

Figure 5: TPR gap between the two groups for the equality of opportunity agent, averaged over 100 simulations. For comparison, we show how the TPR gap is reduced over the course of a simulation without any dynamics (static line). The TPR gap in the dynamic environment does not converge in the same way.

Our demonstration in this section is more exploratory—rather than comparing directly to previous results, we consider how adding dynamics to this problem changes the implications of deployed policies, and their corresponding fairness considerations. This simulation highlights some failure modes that can occur when an agent fails to model the dynamics of the environment. Specifically, we examine how adding feedback between the allocation scheme and incident rates affects long-term outcomes.[4] Considering the scenarios of pest control or food inspections, it seems realistic that the rate of incidents could change in response to interventions that result from allocating attention of inspectors. We find that these dynamics break certain equivalences and trade-offs that are present in the setting with stationary rates.

4.1 Environment

In the allocation environment, the agent distributes N discrete units of attention across K sites at each time step. Each unit of attention is able to discover a single incident. Let $a_{k,t}$ be the amount of attention allocated to site k at time t. At each time step, for each site k, the total number of incidents is sampled $y_{kt} \sim \text{Poisson}(r_{k,t})$. The agent then discovers $\hat{y}_{kt} := \min\{a_{kt}, y_{kt}\}$ incidents (precision discovery model of Elzayn et al. [6]). The rates $r_{k,t}$ change in response to attention

$$r_{k,t+1} = \begin{cases} r_{k,t} + d & \text{if } a_{k,t} = 0 \\ r_{k,t} - d a_{k,t} & \text{else} \end{cases}, \tag{1}$$

where d is a parameter that controls how dynamic the environment is.

[3] We explicitly do not consider the problem of predictive policing in this paper, due in part to concerns such as those raised by [26].
[4] Elzayn et al. [6] noted this extension as a potential area of future work.

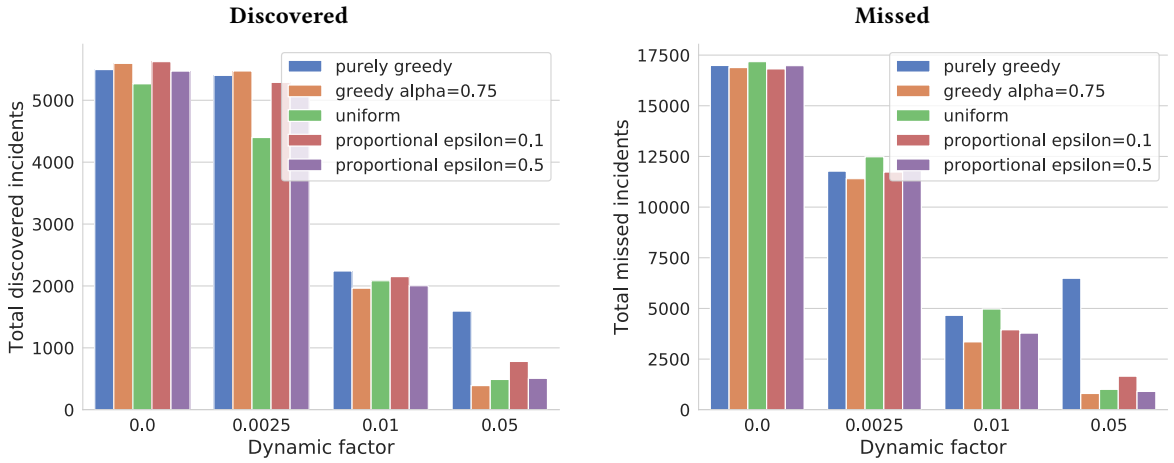

Figure 6: Incidents discovered (left) and missed (right) for different agents. The dynamic factor, d, controls the amount that allocation affects future incident generation. While the greedy agent seems to be the very successful based on number of discovered incidents under dynamics, it is also one of the agents missing the most incidents.

4.2 Metrics

We track several metrics to quantify the welfare of individuals at different sites, and the fairness of agent actions. Because incidents are considered harmful, we track both *total discovered incidents* and *total missed incidents* to measure population welfare. We note that the simulation provides a unique opportunity to track missed incidents, which are often not measured in real-world settings, where administrative data only record discovered incidents.

To assess fairness, we implement a metric that empirically measures departures from a criterion that Elzayn et al. [6] call "equality of discovery probability." Equality of discovery probability implies that incidents that occurred at each site have equal probability of being discovered by the allocation policy. We measure the gap in empirical discovery probabilities as the maximum discrepancy in caught to occurred incidents between sites:

$$\Delta = \max_{k,k'} \left| \frac{\sum_t \hat{y}_{kt}}{\sum_t y_{kt}} - \frac{\sum_t \hat{y}_{k't}}{\sum_t y_{k't}} \right|. \tag{2}$$

This is easily calculated in terms of discovered and missed incident counts. We calculate this gap as an aggregate over the history of the actions and environment's state because we want to assess how the agent performed and affected the environment overall.

4.3 Agents

We evaluate several agents in this simulation study. As a baseline, we consider a *uniform agent* that allocates attention uniformly at random across sites.

In addition, we consider *proportional agents*, motivated by Ensign et al. [7]'s notion of proportional allocation as a fair allocation strategy, which allocate units of attention with probabilities proportional to their estimates of incident rates $\hat{r}$. Finally, we consider fairness-constrained *greedy agents* proposed by Elzayn et al. [6]. These agents allocate attention sequentially, maximizing the probability that the next unit of attention will result in a discovery subject to the constraint that the maximum gap in discovery probabilities between sites is less than α. When $\alpha = 1$, the agent is purely greedy, with effectively no fairness constraints, while $\alpha = 0$ requires exact equality of likelihood of incident discovery across all locations.

Both the greedy and proportional agents rely on estimates $\hat{r}$. Following Elzayn et al. [6], each agent estimates the rates that maximize the likelihood of the observed incident counts $\hat{y}$ at each site under the censored Poisson model (maximum likelihood estimation). Importantly, this internal agent model assumes that the incident rates at each site are fixed in time (the model is misspecified under dynamics). To effectively estimate the rates at each site, the agents must employ some form of exploration. We consider several epsilon-greedy versions of these agents parameterized by exploration parameter ϵ. These allocate attention uniformly with probability ϵ and follow their ordinary policy with probability $1 - \epsilon$.

4.4 Experiments and results

With the environment, agent and metrics specified, we can explore how the dynamics in the incident rates in response to attention (parameterized by d in (1)) affects the long-term outcomes when the agents are deployed. In particular, we examine how the effectiveness and fairness properties of misspecified agents change at the dynamics are made more intense.

To evaluate the performance of agents under the simple dynamics of the environment we run experiments with 5 different dynamic factor values: $d = [0.0, 0.0025, 0.01, 0.05]$, using 5 agents: uniform, proportional $\epsilon = 0.1$, proportional $\epsilon = 0.5$, greedy $\alpha = 0.75$ (fairness constrained), and greedy $\alpha = 1.0$ (unconstrained). Each experiment consists of 50 runs of the 1000 step simulation averaged together. The simulations are run on an environment with 5 locations having rate of $[8, 6, 4, 3, 1.5]$ and 6 units of attention. The fairness parameter $\alpha = 0.75$ for the fairness-constrained greedy agent is relatively high as a consequence of the α-fairness constraint being unsatisfiable for certain combinations of low numbers of the rates and attention units.

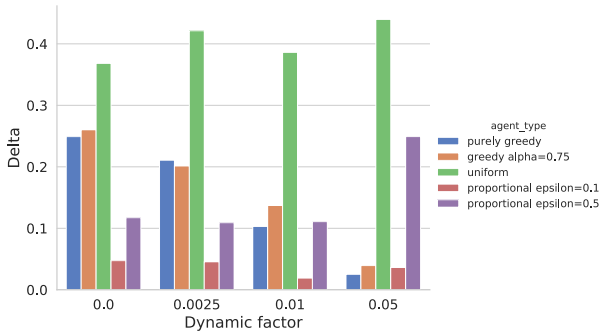

Figure 7: The largest delta across locations of incidents discovered over incidents occurred for each agent and each dynamic factor, as described in Equation 2. Higher deltas correspond to larger disparities in treatment between locations. The purely greedy agent is one of the least fair in the static ($d = 0.0$) scenario but is one of the most fair in dynamic ($d > 0$) scenarios. The proportional (epsilon=0.1) agent performs fairly throughout.

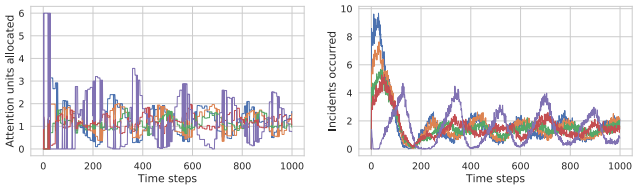

Figure 8: Attention allocations and incident occurrences over time for the purely greedy agent with $d = 0.05$, with each color representing a different site. The agent's allocations lag slightly behind the true rates.

Effectiveness under dynamics. First, we examine the agents' effectiveness at discovering and controlling incidents. Figure 6 summarizes the incidents that are discovered and missed by each agent type. Without dynamics ($d = 0.0$), all agents perform relatively similarly, catching and missing similar numbers of incidents. However, when dynamics are introduced with $d \neq 0.0$, for larger values of d, the unconstrained greedy agent stands out in that it both *catches and misses* more incidents than any other agent. Because uncaught incidents are considered harmful in this scenario, this corresponds to the worst outcomes among all agents.

We illustrate how these poor outcomes play out in Figure 8, which illustrates the behavior of the greedy agent alongside the incident dynamics that its policy induces. By trying to maximize incidents discovered the agent over-allocates to sites where incidents were observed recently, resulting in increasing incident rates in other sites that are under-allocated. By contrast, the other agents are able to cause overall incident rates to decrease, thus avoiding the run-away increase of incidents the greedy agent causes. Here, by optimizing the wrong objective too aggressively, the greedy agent *causes* more incidents to occur.

As Figure 6 makes clear, one of the key difference between the settings with and without dynamics is the relationship between discovered and missed incidents. With static incident rates, maximizing discovered incidents is equivalent to minimizing missed incidents, so to minimize harm from uncaught incidents, it suffices to maximize discovered incidents. However, when incident rates respond dynamically to allocations, this equivalency breaks, and discovering more incidents is no longer an indicator of an agent performing as expected. This has a number of implications for how agents are evaluated, and how we expect agent performance to transfer from static to dynamic settings.

Most importantly, the non-equivalence between high caught incidents and low missed incidents is that, under many realistic data collection mechanisms, the failure of the greedy agent would not be detected by an auditor. As noted above, in many real settings, only discovered incidents are recorded by the agent and available to auditors. If the agents in this simulation were evaluated on discovered incidents alone, the agents inducing poor all-around outcomes would be deemed the most successful. This calls for great care when transferring policies and evaluation strategies from stationary to dynamic settings.

In addition, this non-equivalence suggests that we should not expect policies that are optimized in the static setting to remain optimal or near-optimal, even when small dynamics are introduced. This is because maximizing caught incidents corresponds to the wrong objective under dynamics. Indeed, in examining Figure 6, we see that, under dynamics, fairness constrained policies generally outperform unconstrained policies in terms of missed incidents the long run.

Fairness under dynamics. In addition to agent effectiveness, we also measure the fairness of each agent in this simulation using the discovery probability discrepancy metric defined in (2). Here, we find that strategies that achieve similar effectiveness under dynamics can have widely varying fairness properties. These results are summarized in Figure 7. Most strikingly, allocation schemes that incorporate substantial uniform randomness (uniform and proportional $\epsilon = 0.5$) perform starkly worse in terms of Δ as the dynamic factor d increases. On the other hand, the greedy $\alpha = 0.75$ agent ensures fairer outcomes the larger the dynamic factor d, despite being misspecified. The divergent fairness properties of these agents is somewhat surprising given their largely comparable performance in controlling missed incidents. In fact, among these three strategies, greedy $\alpha = 0.75$ is both the most effective and the fairest. Finally, the pure greedy and proportional $\epsilon = 0.1$ strategies also appear to become fairer as d increases, but this is somewhat less interesting, given that this comes, in part, from a trade-off with effectiveness, which is particularly stark in the case of the pure greedy strategy.

Extensions. This work can be extended to evaluate the performance of agents under a variety of environment changes. Any parameter of the environment can be iterated upon across simulations to explore how resulting metrics change. An interesting avenue of future work for the attention allocation environment is to vary aspects of the environment that the agents make modeling assumptions about. The agents presented involve two key modeling decisions in the likelihood functions, assuming the incidents follow a Poisson distribution and incidents are discovered under a

precision discovery model. This is an ideal scenario in our experiments because the environment does indeed operate under these models. Real world environments, however, may have harder to model discovery functions and incident distributions that can only ever be approximately modeled. A future extension of this work could explore the vulnerability of these modeling decisions in these agents when the environment operates with a different distribution or discovery model.

5 REALIZED EQUILIBRIA IN A STRATEGIC MANIPULATION SETTING: COLLEGE ADMISSIONS

For our final demonstration, we consider a strategic classification scenario [13], in which individuals are able to pay a cost to manipulate their features in order to obtain a desired decision from the agent. We implement this scenarios as a stylized model of college admissions, where applicants are aware of the agent's decision rule, and can pay to manipulate or "game" their observable features to obtain their preferred decision (e.g., by investing in test prep courses). This setting is a special case of a sequential two-player game called the Stackleberg game [2].

In a strategic classification setting, the agent can anticipate feature manipulation, and employ a *robust* decision rule. Hardt et al. [13] showed that if the agent has knowledge of the cost functions so that it can determine how much applicants must pay to manipulate their scores, then the agent can learn a best-response decision rule that nearly recovers the accuracy of the optimal decision rule on unmanipulated scores. Generally, this robust strategy employs a more conservative decision threshold that forces some qualified candidates to manipulate their scores, but is too costly for unqualified candidates to reach.

Robustness, however, imposes a burden on applicants. Both Hu et al. [16] and Milli et al. [22] point out that deploying a robust decision rule in this setting has important fairness implications. In the college admissions scenario, they show that robust strategies can impose disproportionate burdens on qualified applicants from disadvantaged groups. This sets up a trade-off between the agent's utility, which is increasing in the classifier's accuracy, and applicant utility, which is decreasing in the cost that qualified applicants must pay to be accepted by the agent. The implication of this work is that responsible actors should consider this trade-off before deploying robust policies.

Using simulation, we extend these analyses in two ways. First, we compare the behavior of one-shot agents considered in previous work against the behavior of an agent that is able to retrain its classifier across many rounds of decisions, but remains unaware of gaming behavior. Second, we consider how noise in the relationship between an applicant's unmanipulated score and their true label affects the continuously retrained classifier (previous work considered a noiseless relationship). We find that with and without noise, the continuously retrained agent behaves strategically (i.e., implements a robust strategy) by raising its threshold to some extent in the presence of gaming behavior, and we find that noise compounds this incidental strategic behavior. This suggests that the fairness implications of robust policies need to be considered in continuous retraining contexts, even if the agent is not designed to anticipate strategic manipulation.

5.1 Environment

At each round the agent[5], representing a college, publishes its admission threshold score τ. The environment then generates a set of applicants with a set of ground truth test scores in $[0, 1]$ that determine the true eligibility of each applicant. The applicants then choose whether to pay a cost to manipulate their scores in order to pass the admissions bar published in τ. The cost is group-specific, and depends on the difference between the true scores and manipulated scores. Applicants play rationally and only pay to change their score if it will change an unfavorable decision to a favorable one, and if the cost does not exceed the benefit of a favorable decision. The environment emits these manipulated features to the agent, which attempts to classify these applicants.

To examine the fairness in this environment, we follow the terminology used in Milli et al. [22] and report the *social burden*, which is the cost that *all* eligible candidates in a group would have to pay to get favorable decisions. We also consider applicants belonging to two groups, one of which faces the disadvantage of paying a higher cost to improve their score by the same amount.

5.2 Agents

For this set of simulations, we consider the following policies for agents, all trying to maximize accuracy. First, we consider a *static agent* that implements a naïve, one-shot classification strategy. This is implemented as a policy that accepts all applicants for a fixed number of rounds, then trains a fixed classifier on the unmanipulated (score, label) pairs that it has observed. Secondly, we consider a *robust agent* that implements a similar one-shot policy, but uses the robust classification algorithm from Hardt et al. [13] for training. Finally, we consider a *continuous agent* that gathers an initial set of unmanipulated applicants, then continuously retrains a non-robust classifier based on the subsequent manipulated scores and labels that it observes. We consider the continuous agent to be a reasonable model of deployed machine learning systems.

5.3 Experiments and results

Our key finding is that the continuously retraining agent compensates for strategic manipulation to varying extents, even though it is not explicitly designed to do so, and that this behavior is exacerbated by noise.

We illustrate the results of our simulations in Figure 9. The plots on the left are from the setting where unmanipulated scores can perfectly classify applicants; the plots on the right are from the setting where there is noise in the score-label relationship. The top row of plots shows the accuracy that an agent could achieve within each group, and overall, if it were to fix its threshold at a particular value.[6] The decision thresholds that each agent arrives at after many rounds are shown with vertical lines; as predicted by previous

[5] In the strategic classification literature, applicants are often referred to as agents and the college as a *jury*. In our setting where we consider agents and environments, it is simpler to think of the the college as the agent whose action is publishing a threshold score and the actions of the applicants as the environment's response.

[6] A key property of the strategic classification game is that these curves do not map on to the empirical risk curve that one would use to train a standard classifier.

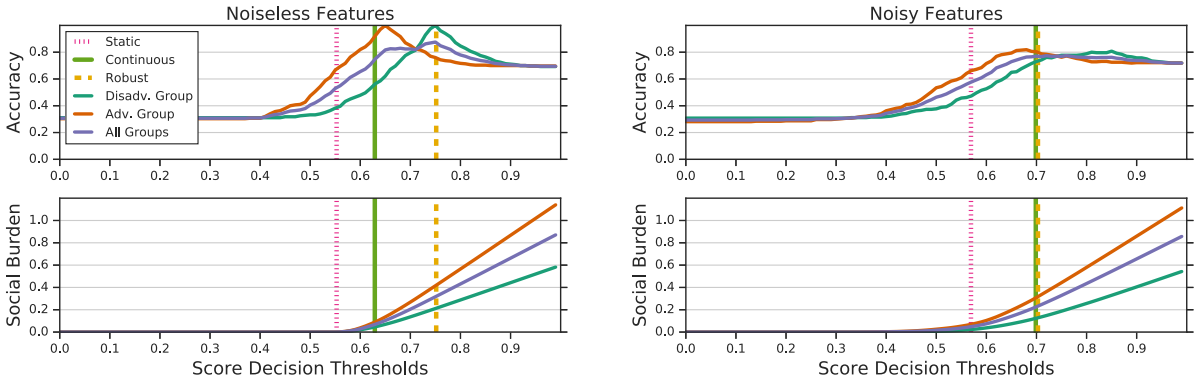

Figure 9: Illustration of limiting decision thresholds and their implications in the college admission scenario. The columns show outcomes in two different noise regimes, and the rows show accuracy (agent utility) and social burden (costs borne by applicants) outcomes. Vertical lines show long-run thresholds reached by each agent type, and curves show outcomes for the whole population and stratified by group. The naïve continuously retrained agent incorporates no knowledge of score manipulation, but still sets a higher threshold at equilibrium than the naïve static agent, achieving a higher accuracy, but inducing a higher social burden. In the high-noise regime, this agent's threshold converges to the manipulation-robust threshold.

work, the robust threshold is always larger than the static one, and optimizes overall accuracy. The bottom row shows the social burden incurred by each group, which is increasing in the decision threshold; this is the social cost of robust classification discussed in previous work [16, 22]. The continuous threshold settles between the static and robust thresholds. In the noisy case, it climbs all the way to the robust threshold after many rounds. These findings are consistent with results in Milli et al. [22], who show that Nash equilibria can occur in this strategic classification setting, and these equilibria always lie between the static naïve and robust thresholds. Because repeated games settle into Nash equilibria, this simulation is useful for characterizing the particular equilibrium that a learning agent reaches, and how the noise regime influences that particular equilibrium.

The results suggest that, if manipulation is occurring, practitioners should consider the fairness implications of robust classification even if they are not deploying a robust agent.

6 DISCUSSION

6.1 Simulations complement experiments with real data

The simulations in this paper are all *extremely simple*. This should not be interpreted as a claim that the world is actually simple; rather, this is meant to highlight that *even* with these simplified dynamics, the consequences of using learning agents are not immediately obvious and require simulation to uncover. Building and verifying realistic simulations of processes as complex as e.g., college admissions and how they influence and are influenced by educational systems and cultural frames, is a laborious process and achieving a high level of realism may be impossible. Striking the right balance of highlighting the important dynamics abstractly in simulation, and effectively using real data and possibly even small experiments to ensure relevance of the results to the real world is an ongoing tension and we expect to see more work in that space.

6.2 Policy search via reinforcement learning

Rather than using simulations to evaluate an agent's fairness in the long term, we can also invert the problem, and use simulation as a way to suggest new fair algorithms, using reinforcement learning [31] to search for policies that optimize for positive long term consequences. The simulation library that accompanies this paper (https://github.com/google/ml-fairness-gym) uses the standard reinforcement learning API of OpenAI Gym [1], and is thus compatible with a many modern reinforcement learning tools.

We note that this search is extremely sensitive to characterization of the *rewards* to be optimized. For example, in the dynamic attention allocation problem described in Section 4, maximizing *discovered* incidents is not the same as minimizing missed incidents. In early experiments with reinforcement learning agents, we observe that a DQN (Deep Q-Network) agent [24] that receives rewards for every discover incident learns to "neglect" locations for long enough for the rates to rise so that it is likely to make a discovery, rather than keeping incidents rates low (which results in fewer discoveries). Code to reproduce this experiment using the Dopamine reinforcement learning framework [4] is available with the rest of the experiments code for this paper. Designing rewards that do not lead to this kind of "gaming the system" behavior is left as a direction for future work.

6.3 Other interesting directions

By expanding the frame of the learning problem to include the concurrent data collection and decision making of an agent (i.e., online learning), we open a number of avenues for exploration including optimal data collection for fairness, and determining whether it is possible to detect unfair actions of a (possibly adversarial) agent whose decisions affect what data it collects.

The simulation framework used here can also be extended to scenarios where multiple agents interact competitively or cooperatively and examine the fairness implications that emerge.

The framework also sets up simulation experiments to be easily reproducible and extendable, which we hope to see become standard in the fairness community. We are excited to see how algorithms can be designed to make fair decisions in dynamic environments and strongly believe that simple simulations are a first step in establishing understanding and tools to address this challenging problem.

ACKNOWLEDGMENTS

We thank William Isaac, Donald Martin, Meredith Whittaker, Andrew Smart, Vinodkumar Prabhakaran, Alex Hanna, Emily Denton, X Eyee, Ed Chi, Zelda Mariet, and the participants at the KDD XAI workshop for in-depth feedback about the focus, implementation, and framing of this manuscript and the `ml-fairness-gym` project.

REFERENCES

[1] Greg Brockman, Vicki Cheung, Ludwig Pettersson, Jonas Schneider, John Schulman, Jie Tang, and Wojciech Zaremba. 2016. Openai gym. *arXiv preprint arXiv:1606.01540* (2016).

[2] Michael Brückner and Tobias Scheffer. 2011. Stackelberg games for adversarial prediction problems. In *Proceedings of the 17th ACM SIGKDD international conference on Knowledge discovery and data mining*. ACM, 547–555.

[3] Alex Campolo, Madelyn Sanfilippo, Meredith Whittaker, and Kate Crawford. 2017. AI now 2017 report. *AI Now Institute at New York University* (2017).

[4] Pablo Samuel Castro, Subhodeep Moitra, Carles Gelada, Saurabh Kumar, and Marc G. Bellemare. 2018. Dopamine: A Research Framework for Deep Reinforcement Learning. (2018). http://arxiv.org/abs/1812.06110

[5] Dheeru Dua and Casey Graff. 2017. UCI Machine Learning Repository. http://archive.ics.uci.edu/ml

[6] Hadi Elzayn, Shahin Jabbari, Christopher Jung, Michael Kearns, Seth Neel, Aaron Roth, and Zachary Schutzman. 2019. Fair algorithms for learning in allocation problems. In *Proceedings of the Conference on Fairness, Accountability, and Transparency*. ACM, 170–179.

[7] Danielle Ensign, Sorelle A Friedler, Scott Neville, Carlos Scheidegger, and Suresh Venkatasubramanian. 2018. Runaway Feedback Loops in Predictive Policing. In *Conference on Fairness, Accountability and Transparency*. ACM, 160–171.

[8] Michael Feldman, Sorelle A Friedler, John Moeller, Carlos Scheidegger, and Suresh Venkatasubramanian. 2015. Certifying and removing disparate impact. In *Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. ACM, 259–268.

[9] Jay W Forrester. 2007. System dynamicsâĂŤa personal view of the first fifty years. *System Dynamics Review: The Journal of the System Dynamics Society* 23, 2-3 (2007), 345–358.

[10] Ian Hacking. 1986. Making Up People. In *Reconstructing individualism: Autonomy, individuality, and the self in Western thought*, Thomas C Heller, Morton Sosna, and David E Wellberry (Eds.). Stanford University Press.

[11] Ian Hacking. 1995. The looping effects of human kinds. (1995).

[12] Ian Hacking, Jan Hacking, et al. 1999. *The social construction of what?* Harvard university press.

[13] Moritz Hardt, Nimrod Megiddo, Christos Papadimitriou, and Mary Wootters. 2016. Strategic classification. In *Proceedings of the 2016 ACM conference on innovations in theoretical computer science*. ACM, 111–122.

[14] Moritz Hardt, Eric Price, and Nathan Srebro. 2016. Equality of opportunity in supervised learning. In *Proceedings of the 30th International Conference on Neural Information Processing Systems*. Curran Associates Inc., 3323–3331.

[15] Anna Lauren Hoffmann. 2019. Where fairness fails: data, algorithms, and the limits of antidiscrimination discourse. *Information, Communication & Society* 22, 7 (2019), 900–915.

[16] Lily Hu, Nicole Immorlica, and Jennifer Wortman Vaughan. 2019. The disparate effects of strategic manipulation. In *Proceedings of the Conference on Fairness, Accountability, and Transparency*. ACM, 259–268.

[17] Shahin Jabbari, Matthew Joseph, Michael Kearns, Jamie Morgenstern, and Aaron Roth. 2017. Fairness in reinforcement learning. In *Proceedings of the 34th International Conference on Machine Learning-Volume 70*. JMLR. org, 1617–1626.

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

[19] Faisal Kamiran and Toon Calders. 2009. Classifying without discriminating. In *2009 2nd International Conference on Computer, Control and Communication*. IEEE, 1–6.

[20] Sampath Kannan, Aaron Roth, and Juba Ziani. 2019. Downstream effects of affirmative action. In *Proceedings of the Conference on Fairness, Accountability, and Transparency*. ACM, 240–248.

[21] Lydia T Liu, Sarah Dean, Esther Rolf, Max Simchowitz, and Moritz Hardt. 2018. Delayed Impact of Fair Machine Learning. In *Proceedings of the 35th International Conference on Machine Learning*.

[22] Smitha Milli, John Miller, Anca D Dragan, and Moritz Hardt. 2019. The Social Cost of Strategic Classification. In *Proceedings of the Conference on Fairness, Accountability, and Transparency*. ACM, 230–239.

[23] Shira Mitchell, Eric Potash, and Solon Barocas. 2018. Prediction-based decisions and fairness: A catalogue of choices, assumptions, and definitions. *arXiv preprint arXiv:1811.07867* (2018).

[24] Volodymyr Mnih, Koray Kavukcuoglu, David Silver, Andrei A Rusu, Joel Veness, Marc G Bellemare, Alex Graves, Martin Riedmiller, Andreas K Fidjeland, Georg Ostrovski, et al. 2015. Human-level control through deep reinforcement learning. *Nature* 518, 7540 (2015), 529.

[25] ProPublica. [n.d.]. compas-analysis. https://github.com/propublica/compas-analysis/

[26] Rashida Richardson, Jason Schultz, and Kate Crawford. 2019. Dirty Data, Bad Predictions: How Civil Rights Violations Impact Police Data, Predictive Policing Systems, and Justice. *New York University Law Review Online, Forthcoming* (2019).

[27] Ken Ross. 2007. *A mathematician at the ballpark: Odds and probabilities for baseball fans*. Penguin.

[28] Andrew D Selbst, Danah Boyd, Sorelle A Friedler, Suresh Venkatasubramanian, and Janet Vertesi. 2019. Fairness and abstraction in sociotechnical systems. In *Proceedings of the Conference on Fairness, Accountability, and Transparency*. ACM, 59–68.

[29] Till Speicher, Hoda Heidari, Nina Grgic-Hlaca, Krishna P Gummadi, Adish Singla, Adrian Weller, and Muhammad Bilal Zafar. 2018. A Unified Approach to Quantifying Algorithmic Unfairness: Measuring Individual &Group Unfairness via Inequality Indices. In *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*. ACM, 2239–2248.

[30] John D Sterman. 2001. System dynamics modeling: tools for learning in a complex world. *California management review* 43, 4 (2001), 8–25.

[31] Richard S Sutton and Andrew G Barto. 2018. *Reinforcement learning: An introduction*.

[32] Sebastian Thrun, Wolfram Burgard, and Dieter Fox. 2005. *Probabilistic robotics*.