Decision making with limited feedback: Error bounds for recidivism prediction and predictive policing

Danielle Ensign
University of Utah

Sorelle A. Friedler
Haverford College

Scott Neville
University of Utah

Carlos Scheidegger
University of Arizona

Suresh Venkatasubramanian
University of Utah

ABSTRACT

When models are trained for deployment in decision-making in various real-world settings, they are typically trained in batch mode. Historical data is used to train and validate the models prior to deployment. However, in many settings, *feedback* changes the nature of the training process. Either the learner does not get full feedback on its actions, or the decisions made by the trained model influence what future training data it will see. We focus on the problems of recidivism prediction and predictive policing, showing that both problems (and others like these) can be abstracted into a general reinforcement learning framework called partial monitoring. We then design algorithms that yield provable guarantees on regret for these problems, and discuss the policy implications of these solutions.

CCS CONCEPTS

• **Theory of computation** → **Reinforcement learning**; *Regret bounds*; • **Computing methodologies** → *Machine learning algorithms*;

KEYWORDS

Feedback loops, predictive policing, recidivism prediction, partial monitoring

ACM Reference format:
Danielle Ensign, Sorelle A. Friedler, Scott Neville, Carlos Scheidegger, and Suresh Venkatasubramanian. 2017. Decision making with limited feedback: Error bounds for recidivism prediction and predictive policing. In *Proceedings of FAT/ML 2017, Halifax, Nova Scotia, Canada, August 2017,* 5 pages.
https://doi.org/10.1145/nnnnnnn.nnnnnnn

1 INTRODUCTION

Machine learning models are increasingly being used to make real-world decisions, such as who to hire, who should receive a loan, where to send police, and who should receive parole. These deployed models mostly use traditional batch-mode machine learning, where decisions are made and observed results supplement the training data for the next batch.

However, the problem of *feedback* makes traditional batch learning frameworks both inappropriate and incorrect. Hiring algorithms only receive feedback on people who were hired, predictive policing algorithms only observe crime in neighborhoods they patrol, and so on. Secondly, decisions made by the system influence the data that is fed to it in the future. For example, once a decision has been made to patrol a certain neighborhood, crime from *that* neighborhood will be fed into the training apparatus for the next round of decision-making.

In this paper, we model these problems in a reinforcement learning setting, and derive algorithms with provable error bounds. Interestingly, these algorithms also translate into concrete procedures that differ from current practice in the problems under study.

The problems. We will focus on two problems that are of particular societal importance: recidivism prediction and predictive policing. These problems are at the core of the algorithmic pipeline in criminal justice through which automated decision-making has a huge material impact on society. In addition, we focus on these problems because they serve as archetypal problems through which we can gain an understanding of generalizable issues faced in deployment. Another motivating factor in the choice of these questions is that systems for solving these problems are already in place and issues with these processes are already documented, making the discussion of remedies more urgent. The problems with recidivism prediction have been documented in the well-publicized and Pulitzer-prize finalist work by ProPublica [1], although interestingly the problems of limited feedback have not been discussed. PredPol, a predictive policing system, has been shown to produce inaccurate feedback loops when deployed in batch mode [16], so that police are repeatedly sent back to the same neighborhoods, even though the underlying crime rate would suggest a different deployment.

Definition 1.1 (Recidivism Prediction). Given an inmate up for parole, use a model of *re-offense* (whether the individual will reoffend within a fixed time period after being released) to determine whether they should be granted parole.

Definition 1.2 (Predictive Policing). Given historical crime data for a collection of regions, decide how to allocate patrol officers to areas to detect crime.

This research is funded in parts by the NSF under grants IIS-1633387, IIS-1633724 and IIS-1513651.

FAT/ML 2017, August 2017, Halifax, Nova Scotia, Canada

ACM ISBN 978-x-xxxx-xxxx-x/YY/MM…$15.00
https://doi.org/10.1145/nnnnnnn.nnnnnnn

Results. We formalize recidivism prediction and predictive policing in the context of reinforcement learning, providing regret bounds in both strong and weak settings. Specifically, we present an $O(T^{2/3})$ regret bound for predictive policing in the setting where we have multiple officers patrolling two neighborhoods, and $O(\sqrt{T})$ mistake bounds for recidivism prediction.

2 RELATED WORK

While our work does not directly address issues of fairness, accountability and transparency, it fits into the larger framework of the social implications of the use of algorithmic decision-making. The narrower question of defining notions of fairness in *sequential* learning settings such as the ones we describe has been studied extensively, primarily in the setting of bandits (regular, contextual and linear) and Markov decision processes[15, 14, 12, 13]. There, the primary goal is to understand how to define fairness in such a process, and how ensuring fairness might affect the ability to learn an accurate model.

We note that the perspective from Markov decision processes (and POMDPs) has much to offer: however, the problems of limited feedback relate more directly to the area of *partial monitoring* [9] which we employ heavily in this paper. There are a number of systems currently in place for recidivism prediction and predictive policing. While the details of the actual implementations (such as COMPAS [18]) remain proprietary, [7] provide a comprehensive review of the methods used in this area. There has been important *empirical* work [16] demonstrating the consequences of feedback loops in simulation in the predictive policing setting (specifically the system known as PREDPOL [17].

3 PARTIAL MONITORING

The reinforcement learning framework we will be using to evaluate the above problems is the well-known *partial monitoring*[19],[9, Chapter 6] framework. Formally, a partial monitoring problem $P = (A, Y, H, L)$ consists of a set of n actions $A = \{a_1, a_2, ..., a_n\}$ and a set of m outcomes (adversary actions) $Y = \{y_1, y_2, ..., y_m\}$. There is a feedback function (also called a feedback matrix) $H\colon A \times Y \to \Sigma$ that takes in a learner action and an outcome and outputs some symbol $\sigma \in \Sigma$ denoting information that the learner receives. Finally there is a hidden loss function (also called a loss matrix) $L\colon A \times Y \to \mathbb{R}$ that takes in an action and an outcome and outputs a loss (which is usually assumed to be positive). We denote $h(a_t, y_t) \in \Sigma$ as the value of H given an action and an outcome, and $\ell(a_t, y_t) \in \mathbb{R}$ as the value of L given an action and an outcome. The learner and adversary are told what L and H are before the learning begins.

As usual, an algorithm consists of sequence of actions, and its quality is measured in terms of regret bounds, (either weak, strong or stochastic). Standard multi-arm bandits [8] can be captured in this setting by setting the feedback matrix H to be equal to the loss matrix L.

In general, proving regret bounds for partial monitoring is hard because the feedback matrix H might bear no relation to the true loss matrix L. Thus, results in this area take two forms. One class of results look at general bounds on partial monitoring under assumptions about the relation between H and L[6] and another class of results look at special sub-cases that are more amenable to analysis (such as the vast literature on bandits[8]).

Context. In certain scenarios, the algorithm might be provided with *context* to help it decide an appropriate action. The intuition behind context is that it provides a signal for what the outcome might be. Formally, fix a class of functions $\mathcal{F}$ from the set of contexts $\mathcal{X}$ to distributions over the space of outcomes. This class is known to both algorithm and adversary. Before the interaction starts, the adversary secretly picks some $f \in \mathcal{F}$. At the start of each round, the algorithm is supplied with *context* $x_t \in \mathcal{X}$. The adversary is then constrained to pick an outcome as a random draw from the distribution $f(x_t)$ [4].

Regret and Mistake Bounds. For any partial monitoring algorithm, let the algorithm actions be $a_1, a_2, \ldots, a_T$ with corresponding outcomes $o_1, o_2, \ldots, o_T$. Note that the actions might be random variables. Then the (weak) regret of the algorithm is its loss compared to the loss of any fixed action:

$$R_T = \sum_{i \in T} \ell(a_i, o_i) - \min_{a \in A} \sum_{t \leq T} \ell(a, o_i)$$

and the *expected* weak regret is $E[R_T]$. We in turn maximize this over all outcomes, yielding the worst-case expected (weak) regret. Alternately, we can measure algorithm performance in terms of *mistake bounds*. A mistake is an action-outcome pair for which $\ell(a, o) > 0$, and the mistake bound of an algorithm is the number of mistakes. Note that mistake bounds are not relative with respect to some fixed action.

4 RECIDIVISM PREDICTION

We now formalize the problem of recidivism prediction in the context of partial monitoring. Recall from Section 1 that recidivism prediction is the problem of determining if someone convicted of a crime will reoffend if released (often measured based on rearrest within a fixed time period, say, 2 years). Such predictions are then used to determine if parole should be granted. The *action* here is the decision to grant parole, and the *outcome* is whether a crime is subsequently committed or not. Formally, we will assume two actions, `keep` and `release` and two outcomes, `c` ("crime") and `¬c` ("no crime"). We can then define a *feedback* matrix and a corresponding loss matrix

$$L = \begin{matrix} \\ \texttt{keep} \\ \texttt{release} \end{matrix} \begin{matrix} \begin{matrix} \texttt{c} & \texttt{¬c} \end{matrix} \\ \begin{pmatrix} 0 & b \\ c & d \end{pmatrix} \end{matrix}, H = \begin{matrix} \\ \texttt{keep} \\ \texttt{release} \end{matrix} \begin{matrix} \begin{matrix} \texttt{c} & \texttt{¬c} \end{matrix} \\ \begin{pmatrix} - & - \\ c & d \end{pmatrix} \end{matrix}$$

In what follows, we will assume that $c = b = 1$ and $d = 0$. However in general, one might assign different values to b, c and d if one had different risk associated with incorrect release (c) versus an unfair (d) or valid (b) incarceration. In recidivism

prediction, *context* consists of profile information about the individual being evaluated.

4.1 A connection to apple tasting

Apple tasting is a well known example of partial monitoring that can be solved with good regret bounds. In the apple tasting problem, the goal is to test apples in an orchard prior to selling them. If we test an apple by tasting it, we discover if it is bad or good, (but we cannot then sell it and incur a loss if it was good). If we sell the apple, then we receive a loss if it is bad and no loss if it is good. In this setting the *partial* monitoring comes from the algorithm only receiving feedback if it decides to taste. However, the algorithm incurs a hidden loss if it tastes a good apple or sells a bad one.

Formally, we can encode this as partial monitoring with the following loss and feedback matrices:

$$L = \begin{matrix} \\ \texttt{sell} \\ \texttt{taste} \end{matrix} \begin{matrix} \begin{matrix} \texttt{bad} & \texttt{good} \end{matrix} \\ \begin{pmatrix} 0 & b \\ c & d \end{pmatrix} \end{matrix}, H = \begin{matrix} \\ \texttt{sell} \\ \texttt{taste} \end{matrix} \begin{matrix} \begin{matrix} \texttt{bad} & \texttt{good} \end{matrix} \\ \begin{pmatrix} - & - \\ c & d \end{pmatrix} \end{matrix},$$

The *context* here is provided by the description of the apple: its color, texture and so on. The key observation we make here is that: *apple tasting is equivalent to recidivism prediction*. This leads us to a regret bound for recidivism prediction.

LEMMA 4.1 (VIA ANTOS ET AL. [2]). *There exists a minimax $O(\sqrt{T})$ weak regret algorithm for recidivism prediction where T is the number of time steps, and this can be achieved using the* EXP3 *algorithm of Auer et al. [3]*

4.1.1 Mistake Bounds. The particular structure of the apple tasting problem allows for a stronger analysis. Helmbold et al. [11] presented algorithm that achieves a mistake bound of $\sqrt{T}$ for apple tasting.Moreover, their bounds apply even when we have *context*. As before, this immediately yields a similar bound for recidivism prediction.

To state the result we must first assume the existence of an online binary classifier that makes a total of M_+ false positive and M_- false negative errors in T steps. As Helmbold et al. [11] show, such a classifier can be obtained from related results [10]. The bound for recidivism prediction can then be stated as follows.

LEMMA 4.2 (VIA HELMBOLD ET AL. [11]). *There exists an algorithm for recidivism prediction whose expected mistake bound upper bounded by $M_+ + 2\sqrt{TM_-}$.*

Algorithms. The results above come with algorithms that achieve the desired error bounds. In the interest of space we focus on the stronger result from Lemma 4.2. The key insight here (and in many such methods) is that in order to defeat the adversary, the algorithm must decide *at random* when to request an evaluation (i.e release an inmate). More formally, the algorithm asks the online classifier to make a prediction. If the classifier predicts $\neg$c, the recidivism predictor does the same. If not, the predictor tosses a coin and with probability roughly $\sqrt{M/T}$ decides to release the inmate anyway and obtains feedback. Over time, the probability of overriding the classifier decreases (as its accuracy increases). We refer the reader to Helmbold et al. [11] for details of the proof.

Policy Implications. The algorithm presented above suggests that inmates should be released *at random* in order to build an effective model for recidivism prediction. The practical remedy for this problem has been to observe that judges are assigned to cases in a random way. Suppose each judge j is modeled by a predictor $p_j(x)\colon \mathcal{X} \to [0,1]$ that takes a personal profile x and releases the person with probability $p_j(x)$. If cases are assigned uniformly at random to one of k judges then the probability of an individual being released is $\frac{1}{k}\sum_j p_j(x)$. As long as individual judge bias (captured in $p_j(x)$) is distributed across the range, this effectively corresponds to releasing an individual uniformly at random.

The claim that current methodology captures the random labeling recommended by Lemma 4.2 rests on a key assumption: for any input x, $\frac{1}{k}\sum_j p_j(x)$ is close to $1/2$. Once we stratify by crime, this might not be true at all: for example, most judges may be less likely to grant convicted murderers parole. In this case, $\frac{1}{k}\sum_j p_j(x) \ll 1/2$. We leave for future research the question of how stable the $\sqrt{T}$-mistake bound algorithm is under such weaker definitions of randomness.

5 PREDICTIVE POLICING

We now turn to the problem of predictive policing. As in the previous section, we will frame this problem in the context of partial monitoring in order to derive the desired regret bounds.

Single officer, two areas. We start with the simple model of one officer patrolling two neighborhoods A and B. The officer actions are `to-A` and `to-B` and in each area either crime is committed or not. Our loss matrix and the corresponding feedback matrix are

$$L = \begin{matrix} \\ \texttt{to-A} \\ \texttt{to-B} \end{matrix} \begin{matrix} \begin{matrix} \texttt{A,B} & \neg\texttt{A,B} & \texttt{A},\neg\texttt{B} & \neg\texttt{A},\neg\texttt{B} \end{matrix} \\ \begin{pmatrix} 1 & 1 & 0 & 0 \\ 1 & 0 & 1 & 0 \end{pmatrix} \end{matrix}$$

$$H = \begin{matrix} \\ \texttt{to-A} \\ \texttt{to-B} \end{matrix} \begin{matrix} \begin{matrix} \texttt{A,B} & \neg\texttt{A,B} & \texttt{A},\neg\texttt{B} & \neg\texttt{A},\neg\texttt{B} \end{matrix} \\ \begin{pmatrix} 1 & 0 & 1 & 0 \\ 1 & 1 & 0 & 0 \end{pmatrix} \end{matrix}.$$

where `A` denotes the event that crime occurs in A and `B` denotes crime in B. Note that the officer only gets feedback from the region she visits.

We first observe that setting $K = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}$ we can write $L = KH$. This allows us to invoke a general theorem from [9, Chapter 6] that yields the following result.

LEMMA 5.1. *Predictive policing with one officer and two neighborhoods admits a solution with $O(T^{2/3})$ worst-case weak regret.*

Multiple officers, two areas. We now turn to the more general scenario with multiple officers monitoring two areas. When we allow multiple officers to patrol an area, we have to modify our definition of loss and feedback. We will assume that

each officer can detect one crime, officers detect crime independently, and the *feedback* only reveals the number of crimes caught. Assume that a precinct has k police officers are available to patrol the regions A and B. We model this allocation as an action $\mathbf{i} = (i, k-i)$. An actual randomized strategy will choose among these with some probability.

The outcome $\mathbf{o} = (a, b)$ generated by the adversary represents the actual crime in each area. The (hidden) loss associated with this action-outcome pair is given by $\ell(\mathbf{i}, \mathbf{o}) = \max(0, a-i) + \max(0, b+i-k)$ and the corresponding feedback is $h(\mathbf{i}, \mathbf{o}) = \sum_i \min(a_i, o_i)$. We would like to bound the weak regret for such a game with respect to worst-case adversaries.

THEOREM 5.2. *There is an algorithm for predictive policing with two neighborhoods that for a T-step game has $O(T^{2/3})$ weak regret against worst-case adversaries.*

We now present a high-level sketch of the proof. The loss and feedback functions defined above yield loss and feedback *matrices* indexed by actions and outcomes, i.e $H_{\mathbf{i},\mathbf{o}} = h(\mathbf{i}, \mathbf{o})$. Let the total number of outcomes be denoted by M (we shall see shortly that M is bounded by $(k+1)^2$).

Definition 5.3 (Signal Matrix[6]). Let s_i be the number of distinct symbols in the i^{th} row of H and let $\sigma_1, \sigma_2, \ldots, \sigma_{s_i}$ be an enumeration of those symbols. Then the *signal matrix* $S^{\mathbf{i}} \in \{0,1\}^{s_i \times M}$ of action i is defined as $(S^{\mathbf{i}})_{k,\mathbf{o}} = \mathbb{I}\{H_{\mathbf{i},\mathbf{o}} = \sigma_k\}$.

Let Im A denote the column space of a matrix A. Let ℓ_i be a column vector formed as the transpose of the i^{th} row of L. Let $A \oplus B$ denote a direct sum of matrices A, B i.e the matrix formed by concatenating corresponding rows (assuming they each have the same number of rows). Then the key relationship between L and H is captured in the following definition:

Definition 5.4 ([19]). The pair (L, H) is said to be *globally observable* if $\ell_i - \ell_j \in \text{Im} \oplus_{\mathbf{i}} S^{\mathbf{i}^\top}$

This is important because as Bartók et al. [6] show, any globally observable system admits a protocol with $O(T^{2/3})$ weak regret[1]. Thus, all that remains is to show that the pair (L, H) associated with predictive policing is globally observable.

It turns out that using the stated loss function makes it hard to prove global observability. Thus, we consider a different loss function. Suppose a_t, b_t crimes committed respectively at time t and assume that i_t officers are placed in A at time t. Instead of $\ell'(\mathbf{i}, (a,b)) = \max(a-i, 0) + \max(b-k+i, 0)$ we will use the loss $\ell(\mathbf{i}, (a,b)) = \min(a+b, k) - \min(a, i) - \min(b, k-i)$. Let s_t be the number of 'successful' officers at time t, the number of officers who caught some crime at that timestep. The key observation is that for any timestep t, ℓ and ℓ' differ by an amount that depends only on a_t, b_t and k, and thus the strategies under these two losses are identical (since k is fixed and the algorithms don't get to see a_t, b_t). Further, the regret depends only on s_i:

$$\begin{aligned}\ell'(s) - \ell'(s') &= \sum_t (a_t + b_t) - \sum_t (s_t) - (\sum_t (a_t + b_t) - \sum_t (s'_t)) \\ &= \ell(s) - \ell(s')\end{aligned}$$

Without loss of generality, we can assume that the outcomes (a_t, b_t) are such that $a_t, b_t \le k$. Thus the space of outcomes is $\{(a,b) \mid 0 \le a \le k, 0 \le b \le k\}$ of size $(k+1)^2$. Therefore, we can represent each row of the loss matrix as a $k+1 \times k+1$ matrix with rows and columns indexed by a_t, b_t (note the 0 indexing), and it has a nice form.

$$\ell_{\mathbf{i}} = \begin{pmatrix} 0 & 0 & \cdots & & 0 & 1 & 2 & \cdots & c-1 & c \\ 0 & 0 & \cdots & & 0 & 1 & 2 & \cdots & c-1 & c-1 \\ \vdots & \vdots & \ddots & & \vdots & \vdots & \vdots & & & \vdots \\ 0 & 0 & \cdots & & 0 & 1 & 2 & \cdots & 2 & 2 \\ 0 & 0 & \cdots & & 0 & 1 & 1 & \cdots & 1 & 1 \\ 0 & 0 & \cdots & & 0 & 0 & & \cdots & 0 & 0 \\ 1 & \cdots & & 1 & 0 & \vdots & & & \vdots & \vdots \\ 2 & \cdots & 2 & \vdots & & & & & & \\ \vdots & & & & & & & & & \\ d-1 & d-1 & & & & & & & & \\ d & d-1 & \cdots & 1 & 0 & 0 & \cdots & & & 0 \end{pmatrix}$$

where c, d are constants. Explicitly,

$$\ell_{\mathbf{i}}(a,b) = \ell(i,(a,b)) = \begin{cases} \min(i-a, b-(k-i)) & \text{if } a < i \text{ and } b > k-i \\ \min(a-i, (k-i)-b) & \text{if } a > i \text{ and } b < k-i \\ 0 & \text{otherwise} \end{cases} \quad (1)$$

This follows from a straightforward case analysis.

At a high level the rest of our proof now works as follows. We show (based on the interpretation of the rows of the loss matrix above) that the associated entries in the signal matrix also have a simple representation. We then use these representations to construct a vector of weights $\mathbf{x}$ such that

$$\ell_i - \ell_j = (\oplus_{\mathbf{i}} S_{\mathbf{i}}^\top)\mathbf{x}$$

establishing Theorem 5.2.

The Algorithm. The algorithm that yields the regret bound promised by the above structural result is an exponentially-weighted forecaster developed by Piccolboni and Schindelhauer [19]. Roughly speaking, it maintains weights on each action (in this case, the desired assignment of officers to areas), and based on the feedback updates the weights. Global observability implies that these updates can be done correctly, and all actions retain some fixed likelihood of being chosen, regardless of past information.

This is in sharp contrast to current approaches to predictive policing, which maintain estimates of crimes seen in the two areas based on past policing activity, and use these estimates to make new decisions. As Lum and Isaac [16] have shown in prior work, such an approach can lead to runaway feedback where the system encourages officers to patrol only the region perceived as having more crime.

[1] The theorem also requires a condition called *nondegeneracy*. Our system satisfies this property, but in the interest of space we defer this proof to an extended version of the paper.

6 CONCLUSION

In this paper we present a model for decision-making with limited feedback in terms of the partial monitoring framework from machine learning, and show that this yields algorithms with provable guarantees. Note that these bounds are not necessarily optimal: it might be possible to obtain $\sqrt{T}$ regret bounds for predictive policing.

REFERENCES

[1] Julia Angwin, Jeff Larson, Surya Mattu, and Lauren Kirchner. Machine bias. *ProPublica*, May 23, 2016.

[2] András Antos, Gábor Bartók, Dávid Pál, and Csaba Szepesvári. Toward a classification of finite partial-monitoring games. *Theoretical Computer Science*, 473:77–99, 2013.

[3] Peter Auer, Nicolo Cesa-Bianchi, Yoav Freund, and Robert E Schapire. The nonstochastic multiarmed bandit problem. *SIAM journal on computing*, 32(1):48–77, 2002.

[4] Gábor Bartók and Csaba Szepesvári. Partial monitoring with side information. In *International Conference on Algorithmic Learning Theory*, pages 305–319. Springer, 2012.

[5] Gábor Bartók, Dávid Pál, and Csaba Szepesvári. Minimax regret of finite partial-monitoring games in stochastic environments. In *COLT*, volume 2011, pages 133–154, 2011.

[6] Gábor Bartók, Dean P Foster, Dávid Pál, Alexander Rakhlin, and Csaba Szepesvári. Partial monitoring—classification, regret bounds, and algorithms. *Mathematics of Operations Research*, 39(4):967–997, 2014.

[7] Richard A. Berk and Justin Bleich. Statistical procedures for forecasting criminal behavior. *Criminology & Public Policy*, 12(3):513–544, 2013. ISSN 1745-9133. doi: 10.1111/1745-9133.12047. URL http://dx.doi.org/10.1111/1745-9133.12047.

[8] Sébastien Bubeck, Nicolo Cesa-Bianchi, et al. Regret analysis of stochastic and nonstochastic multi-armed bandit problems. *Foundations and Trends® in Machine Learning*, 5 (1):1–122, 2012.

[9] Nicolo Cesa-Bianchi and Gábor Lugosi. *Prediction, learning, and games*. Cambridge university press, 2006.

[10] David P. Helmbold, Nicholas Littlestone, and Philip M. Long. On-line learning with linear loss constraints. *Information and Computation*, 161(2):140 – 171, 2000. ISSN 0890-5401. doi: http://dx.doi.org/10.1006/inco.2000.2871. URL http://www.sciencedirect.com/science/article/pii/S0890540100928712.

[11] David P Helmbold, Nicholas Littlestone, and Philip M Long. Apple tasting. *Information and Computation*, 161(2): 85–139, 2000.

[12] Shahin Jabbari, Matthew Joseph, Michael Kearns, Jamie Morgenstern, and Aaron Roth. Fair learning in markovian environments. *CoRR*, abs/1611.03071, 2016. URL http://arxiv.org/abs/1611.03071.

[13] Matthew Joseph, Michael Kearns, Jamie Morgenstern, Seth Neel, and Aaron Roth. Rawlsian fairness for machine learning. *arXiv preprint arXiv:1610.09559*, 2016.

[14] Matthew Joseph, Michael Kearns, Jamie H Morgenstern, and Aaron Roth. Fairness in learning: Classic and contextual bandits. In D. D. Lee, M. Sugiyama, U. V. Luxburg, I. Guyon, and R. Garnett, editors, *Advances in Neural Information Processing Systems 29*, pages 325–333. Curran Associates, Inc., 2016. URL http://papers.nips.cc/paper/6355-fairness-in-learning-classic-and-contextual-bandits.pdf.

[15] Sampath Kannan, Michael Kearns, Jamie Morgenstern, Mallesh Pai, Aaron Roth, Rakesh Vohra, and Z Steven Wu. Fairness incentives for myopic agents. *arXiv preprint arXiv:1705.02321*, 2017.

[16] Kristian Lum and William Isaac. To predict and serve? *Significance*, pages 14 – 18, October 2016.

[17] George O. Mohler, Martin B. Short, Sean Malinowski, Mark Johnson, George E. Tita, Andrea L. Bertozzi, and P. Jeffrey Brantingham. Randomized controlled field trials of predictive policing. *Journal of the American Statistical Association*, 110(512):1399 – 1411, 2015.

[18] Inc. NorthPointe. Compas. http://www.northpointeinc.com/files/downloads/FAQ_Document.pdf.

[19] Antonio Piccolboni and Christian Schindelhauer. Discrete prediction games with arbitrary feedback and loss. In *International Conference on Computational Learning Theory*, pages 208–223. Springer, 2001.