Algorithm Games and Rational Play with Strategic Inference

In-Koo Cho* and Jonathan Libgober**

*Emory University
**University of Southern California

November 23, 2021

Abstract. We develop an approach, inspired by ideas from machine learning, to study how algorithms can induce rational play in settings featuring sequential moves and exogenous uncertainty. An algorithm designer seeks to prescribe actions for a sequence of second movers using only historical data, to maximize their average long run payoffs. While constraints on feasible algorithms may prevent as-if rational play from emerging, we describe how these constraints can be endogenously circumvented to effectively guide behavior across a rich set of possible environments using limited details.

1. Introduction

When might "as-if" rational play emerge in strategic settings when full rationality is not assumed? This paper addresses this classic question within a class of sequential move games where the action of a first mover could strategically convey information about a (payoff-relevant) state to a second mover. In our model, the first mover's strategy is chosen with commitment as a function of the state (as in, e.g., Kamenica and Gentzkow (2011)). Meanwhile, an algorithm prescribes a reply to (a sequence of) second movers, using the interaction history to determine future recommendations. Our interest in this paper is when and how rationality can be thus achieved.

Contact. icho30@emory.edu, libgober@usc.edu . This paper was previously circulated under the title "Machine Learning for Strategic Inference." We thank Juan Carrillo, Jason Hartline, Kevin He, Navin Kartik, David Kempe, Roger Moon, Xiaosheng Mu, Guofu Tan, Ashesh Rambachan, Joel Sobel, Erik Strand, and especially Grigory Franguridi, for helpful conversations and comments, and seminar audiences at AMETS, Emory, the Econometric Society World Congress, the World Congress of the Game Theory Society, the NSF/NBER/CEME Conference on Mathematical Economics, Rochester, UC-San Diego, and USC. This project started when the first author was visiting the University of Southern California. We are grateful for hospitality and support from USC. Financial support from the National Science Foundation is gratefully acknowledged.

Our paper differs from past work addressing the above question due to our focus on sequential move settings with strategic inference. This focus, in turn, will motivate two distinguishing features of our exercise. First, inducing rational play for the second mover will require the estimation of an exogenous distribution over payoff relevant states. Second, the specification of the strategy set for the second-mover will be endogenous—that is, determined by the algorithm.

The combination of these two features is economically relevant and timely. Examples abound. Consider the problem of a recommendation platform desiring that buyers who frequent it be able to safely trust their recommendations blindly—that is, to prescribe actions coinciding with the behavior of a well-informed rational economic agent. Buyer-seller interactions are frequently modelled as sequential move games, and they will fall within our framework (as we will discuss extensively). Different products may feature different distributions over quality, and sellers may react to the algorithm when deciding how to sell. Thus, whether buying is rational depends on both endogenous choices (i.e., prices chosen by sellers) and exogenous variables (i.e., the distribution over product quality). Our first distinguishing feature emerges since the distribution over exogenous variables will influence whether the algorithm should recommend buyers purchase. Furthermore, if the algorithm is to be portable across a wide variety of products and categories, then the platform may wish the recommendation strategy to be *as simple as possible*. But the "as possible" will ultimately depend upon what strategies sellers end up implementing. Thus, the algorithm must determine when to increase the sophistication of the recommendation policy provided to buyers. This *endogenous simplicity* is the basis of our second distinguishing feature.

In sequential move settings with strategic inference such as the platform's problem above, it is typical for the distribution over states (product quality) to influence the optimal second-mover strategy. Thus, the need to learn about this distribution in order to determine rational replies—our first distinguishing feature—will be immediate. It may be less obvious why the second distinguishing feature will matter. To see this, note that if there are two possible second-mover *actions* and a continuum of first-mover actions, then the set of possible second-mover *strategies* has cardinality equal to the power set of the continuum. Our view is that it would be intractible for an algorithm to determine an optimal strategy from a set this large. Still, some readers might object that in many settings, permissive assumptions can imply that rational second-mover strategies always belong to a smaller, tractable subset. Thus, a natural conjecture is that, instead of endogenizing the strategy set for the second mover, one could employ the following procedure:

- First, determine the set of optimal first-mover strategies, assuming a rational second mover.
- Then, specify the second-mover strategies to be those which are optimal against some first-mover strategy (and have the algorithm choose the best performing among these).

With *exogenously fixed* first-mover strategies, such a scheme would achieve our ultimate goal.

Unfortunately, we identify a form of *algorithm exploitation* which implies this scheme could fail to produce a rational reply. The key economic force was first described (to our knowledge) by Rubinstein (1993). We use the main model from that paper as a running example, and build on its conclusions as well. That paper studied a buyer-seller setting and showed that if a buyer is restricted to use a single-threshold strategy—i.e., one that makes the same decision on a given side of a fixed threshold—then the seller can benefit from a particular form of randomization which "fools" these buyers into making a decision which is suboptimal given the realized price.[1]

In the construction of Rubinstein (1993), a two-threshold strategy (i.e., one that prescribes the same action depending on which of three intervals the price belongs to) would prevent the seller's exploitation. So, one might conjecture that simply requiring the algorithm to consider these strategies as well would be enough to induce rational play. We extend his result to highlight a *horse race* that emerges more generally—the first mover might always seek to "one up" the second mover; if the algorithm were to consider any 83-threshold strategy as possible, the first mover could exploit it by using a strategy which would require 84 thresholds to achieve optimality.

To emphasize the horse race possibility, we will assume throughout that the first mover's strategy is chosen rationally. As an aside, this will also allow us to avoid complications associated with interacting algorithms.[2] But more importantly, since the exploitation we study corresponds to the first mover's departures from the strategy she would follow were the second-mover rational, the focus on a rational first mover will allow us to highlight this directly.[3]

The purpose of this paper is to show that algorithm design provides a way out of the horse race. Toward this end, we introduce the structure of an *algorithm game* to articulate the incentives underlying the choice of algorithm. In an algorithm game, an algorithm designer seeks a recommendation strategy that maximizes the average second-mover payoff. While algorithms cannot condition on either the distribution over the state or the first mover's strategy, data on past interactions enable the algorithm to potentially prescribe rational replies (at least approximately), across a wide set of parameters—or more precisely, across possible distributions over the state.

The closest precedent for the structure of algorithm games comes from the literature on "machine games," pioneered by Rubinstein (1986) and Abreu and Rubinstein (1988). These papers

[1] The reasoning behind this result is as follows. First, the optimally chosen single-threshold classifier can do strictly better than simply randomizing the guess, implying that the seller can exploit the incentives of the buyer in order to manipulate the decision rule. On the other hand, it is impossible for threshold rules to implement the optimal decision with probability 1 when this rational rule is non-monotone in the price. The first point implies the buyer trades off against errors; the second implies that the tradeoff falls short of the fully rational response. As a result, the seller can force a different decision than the rational one for these buyers (with arbitrarily high probability).

[2] Past work shows that outcomes may depart from Nash predictions when multiple algorithms interact, in particular when algorithms can condition their strategies on the past actions of other algorithms; Calvano, Calzolari, Denicolò, and Pastorello (2019) and Brown and MacKay (2021) show this can lead to collusion in pricing settings.

[3] See 7.2 for more discussion of this assumption.

introduced machine games to study whether as-if rational play (appropriately defined) might emerge in two-player repeated games when strategies are executed by automata. Automata, in turn, are reductive descriptions of strategies, and in a machine game they are evaluated in terms of the payoffs they induce in the repeated game (net of any costs associated with executing them). However, to the best of our knowledge, past work in this area has not considered our first distinguishing feature, namely the need to estimate a persistent distribution to determine rational replies. The need to incorporate this motivates the new structure of an algorithm game.[4]

How will the introduction of algorithm games enable us to exit the horse race, given the apparent unbounded complexity necessary to cover the set of all possible strategies? We distinguish two sources of complexity: the complexity of the initially endowed strategies, and the complexity of the algorithm's updating rule. Specifically, the algorithms we evaluate are those that are initially endowed with a set of strategies—which we refer to as *baseline strategies*—and the ability to find the "best performing strategy" from this set (according to an arbitrary metric, which could depend on the history of the interaction). We posit that an algorithm that involves a larger set of baseline strategies is more complex, as finding the best performing one becomes more computationally demanding. However, we *will* allow algorithms to construct richer strategies which fall outside of the set of baseline strategies, if specified how—the constraints on and complexity of algorithms will hinge on which strategies the algorithm can *optimize* among.

Constraints in the form of restrictions on baseline strategies have been studied extensively in the machine learning literature, which typically treats the data generating process as exogenous. Our goal, however, is to perform a similar algorithm design exercise, but where the "data" is generated by endogenous strategic choices. The endogeneity will influence the realized, on-path performance of algorithms, even though we will need to consider off-path performance as well to determine optimal first-mover choices. That said, the constraints we study are similar to those that emerge in typical machine learning problems, where simple predictions are often sought for an observation among a very large set of possibilities. Restricting baseline strategies in this paper is analogous to restricting the initial set of classifiers in those applications. Aside from drawing inspiration from machine learning to formalize algorithmic constraints, we also draw inspiration to define "good performance" of an algorithm. The requirement is that the algorithm can achieve a *Probably Approximately Correct (or PAC)* guarantee. We present this notion in Section 3.1, where we describe why it corresponds to successful implementation of rational behavior in the long run.

Our main results highlight the following tension, the main message of the paper. On the

[4] As we discuss more thoroughly in the literature review, a large literature has studied *particular* algorithms in simultaneous-move games (e.g., fictitious play, regret-matching etc.), and asked what kind of behavior emerges in the long run when they are used by players. But since the possible strategy space induced by these algorithms is exogenously given, it is less clear how they speak to our second distinguishing feature.

one hand, due to the problem of algorithm exploitation, it is generally not possible to ensure rationality if the set of strategies the algorithm can prescribe coincides with the set of baseline strategies as defined above. Nevertheless, it is possible to induce rationality if algorithms can endogenously expand the set of strategies they can implement, even without enriching the set of baseline strategies. Algorithm games provide a way out of the horse race.

How are these richer strategies constructed? As a starting point, we observe that the Adaptive Boosting algorithm (also known as AdaBoost; Schapire and Freund (2012)) can construct (under an appropriate condition) an arbitrarily accurate strategy by considering a "weighted combination of baseline strategies," with the weights determined by the algorithm. This algorithm requires the ability to (repeatedly) find an optimal response to an arbitrary conjecture of the first mover's play. We are able to achieve our goal, then, by specifying how this algorithm can be applied to the economic setting at hand. Aside from the need to verify that the condition for this algorithm to work[5] are satisfied under permissive assumptions, there are two main technical difficulties we face toward achieving that end. These form the bulk of our technical contribution.

The first technical contribution is due to our focus on a strategic inference problem. In a buyer-selling setting, it may be that "low quality" is observed at some price, but that "high quality" is in fact more likely and that correspondingly a rational buyer would choose a "buying" action. More generally, our exercise requires the payoff-maximizing decision to be *inferred* by the algorithm. One of our main results is that this added difficulty does not change the qualitative desirable properties of the algorithm. This result uses techniques from large deviations theory, allowing us to find a rate at which we can guarantee an approximation of rational (second-mover) strategies.

These results require the first mover only use a finite (though otherwise arbitrary) number of actions. Without this, we cannot guarantee sufficient data is observed for a given first-mover action to infer the rational reply following that action. The second technical contribution shows this restriction can be relaxed if the first mover's strategy is subject to potentially arbitrarily small implementation error. These shocks essentially reduce the algorithm's problem to one where the previous restrictions can be imposed, with minimal changes to incentives.

To summarize, the answer to our theoretical question is that rationality can be ensured with the ability to (a) find an optimal strategy among the set of single-threshold strategies, and (b) combine strategies in a particular (pre-specified) way. Our hypothetical recommendation platform could eventually guarantee as-good-as rational recommendations. In the theoretical setting we identify, as-if rational play can emerge even without explicit rationality from the second movers.

[5] The requirement on the set of baseline strategies for this algorithm to work is known as *weak learnability*. This requirement is defined formally in Section 6.1, and roughly speaking says that the optimal baseline strategy uniformly outperforms a random guess. This requirement is significantly less demanding than requiring that the strategy always produce the optimal reply (which rationality requires). We show how to check weak learnability straightforwardly when resorting to single-threshold classifiers (which have natural interpretations, even beyond Rubinstein (1993)).

2. Environment

Our model consists of two components: a stage game and a supergame. Strategies are chosen in the latter and then executed in the former. We call our particular supergame an *algorithm game*, as this is where the choice of algorithm is made. We defer a discussion of how we represent algorithms, as well as the desirable features of our proposed algorithm, until Section 3.

2.1. Stage Games

2.1.1. Actions and Parameters

The stage game is a Sender-Receiver game in which an informed Sender makes the first move. We also call the Sender *the (informed) principal*, and the Receiver *the agent*, as in Maskin and Tirole (1992), though our model also describes a Sender-Receiver game with Sender commitment, as in Kamenica and Gentzkow (2011).

Let Θ denote a set of types endowed with a prior distribution π, where $\pi(\theta)$ is the probability that type $\theta \in \Theta$ is realized. This type is potentially payoff relevant to both the Sender and the Receiver. Throughout the paper, we only consider π with finite support. We also assume throughout that π is known by the Sender; we take π to be (commonly) known by the Receiver only in the benchmark where he is (assumed to be) rational, though most of the paper is instead concerned with the case where the Receiver is not rational and cannot (directly) condition their strategy on π.

Conditioned on the realized value of $\theta \in \Theta$, the Sender takes an action $p \in \mathcal{P} \subset \mathbb{R}^n$ where $\mathcal{P}$ is compact. The strategy of the Sender is:

$$\sigma : \Theta \to \Delta(\mathcal{P}),$$

where $\Delta(X)$ denotes the set of probability distributions over a set X. We let Σ denote the set of all possible σ. The choice of $\sigma \in \Sigma$ is determined in the algorithm game described below in Section 2.2.1. At the start of the stage game, p is drawn (as per σ) and observed by the Receiver. Conditioned on p (but not θ), the Receiver chooses $a \in A$ according to a strategy:

$$r : \mathcal{P} \to \Delta(A).$$

We assume$|A| < \infty$; when describing our methods in the main text we primarily focus on the case of$|A| = 2$, discussing how to generalize the results when$|A| > 2$ in Section 6.3.1.[6] Stage game payoffs of the Sender and the Receiver following (θ, p, a) are $u(\theta, p, a)$ and $v(\theta, p, a)$, respectively.

[6] We also present an application with more than two possible Receiver actions in Section 5.2.

The timing of the moves in the stage game is as follows (where we recall the strategies for each player will be chosen in the algorithm game, described below):

S_1. The state $\theta \in \Theta$ is realized according to π, with only the Sender observing θ.

S_2. The Sender's action $p \in \mathcal{P}$ is realized according to $\sigma(p : \theta)$.

S_3. The Receiver takes action $a \in A$ conditioned on p (but not θ).

S_4. Payoffs are realized according to $u(\theta, p, a)$ and $v(\theta, p, a)$.

Though special, this stage game framework is very rich and covers many different previously studied applications. For instance, If we interpret $p = (p_1, \dots, p_n)$ as a contract, and $a \in A = \{-1, 1\}$ as "reject" ($a = -1$) or "accept" ($a = 1$), the stage game is a model of the informed principal (Maskin and Tirole (1992)). If p is interpreted as a message sent by a worker, and $a \in A$ as the wage paid by the firm, then the stage game becomes a signaling game (Spence (1973)). Both of these applications are discussed in more detail in Section 5. For now, we place no further restrictions on $u(\theta, p, a)$ and $v(\theta, p, a)$, though these are often implicit.

2.1.2. Payoffs and the Rational Benchmark

The outcomes of the above interaction when both players are rational are familiar. In that case, Receiver's optimization problem is:

$$\max_{a \in A} \sum_{\theta \in \mathsf{supp}\pi} v(\theta, p, a)\pi(\theta : p)$$

where $\pi(\theta : p)$ is the posterior probability assigned to θ conditioned on p. If p is used with a positive probability by σ, then $\pi(\theta : p)$ is computed by Bayes rule:

$$\pi(\theta : p) = \frac{\sigma(p : \theta)\pi(\theta)}{\sum_{\theta'} \sigma(p : \theta')\pi(\theta')}.$$

We will refer to the *rational label*, denoted $y^R : \Sigma \times \mathcal{P} \to A$, as the solution to the following optimization problem:[7]

$$\sum_{\theta \in \mathsf{supp}\pi} v(\theta, p, y^R(\sigma, p))\pi(\theta : p) \geq \sum_{\theta \in \mathsf{supp}\pi} v(\theta, p, a)\pi(\theta : p) \qquad \forall a \in A,$$

[7] We use the term "label" to distinguish the strategy of the Receiver, as both Sender and Receiver will use strategies. We also view the label terminology as helping connect our work to the machine learning literature, where we derive a lot of the motivation for our exercise, as it is consistent with the usage there—for instance, as we discuss in Section 6.1, there is a connection between our exercise and the literature on *label noise* in machine learning.

where $\pi(\theta : p)$ is computed via Bayes rule whenever $\sum_\theta \sigma(p : \theta)\pi(\theta) > 0$.[8] Let $\mathcal{H}^R$ denote the set of all y^R which emerge under some $\pi \in \Pi$.

We will let σ^R denote the best response of the Sender against a Bayesian rational Receiver with perfect foresight:

$$\sum_{\theta,p,a} u(\theta, p, a)\sigma^R(p : \theta)y^R(\sigma^R, p)\pi(\theta) \geq \sum_{\theta,p,a} u(\theta, p, a)\sigma(p : \theta)y^R(\sigma, p)\pi(\theta) \qquad \forall \sigma \in \Sigma.$$

Note that (σ^R, y^R) constitutes a perfect Bayesian equilibrium in the stage-game with a rational Receiver. Of course, if the Receiver were not rational, then it correspondingly might not be optimal for a rational Sender to choose σ^R.

2.1.3. Running Example: Monopoly Market (Rubinstein 1993)

As discussed in the introduction, we build on the main observation of Rubinstein (1993) to highlight the possibility of a horse race between the first and second mover when algorithm capabilities are restricted. We briefly describe his model, commenting on differences until after we present the definition of an algorithm game. In Rubinstein (1993), a monopolist chooses a single price to sell to two different kinds of buyers—and only one buyer type is guided by the algorithm, the other being rational. The algorithm, by contrast, decides whether its relevant buyers should buy (i.e., choose $a = 1$) or not buy (i.e., choose $a = -1$), given the observed price. The quality of the seller's product is given by $\theta \in \{L, H\}$, and the buyer's value for the product is v_θ, where $v_H > v_L$. This is a standard lemons setting. Notice that:

$$y^R(\sigma, p) = 1 \Leftrightarrow \mathbb{E}[v_\theta \mid p, \sigma] \geq p.$$

Importantly, the seller's payoff is not simply 0 if the buyer chooses not to buy, since the price is constrained to be the same for both buyer types. Rubinstein (1993) further assumes:

- $u(L, p, -1) < u(L, p, 1)$
- $u(H, p, 1) < u(H, p - 1)$
- $\arg\max_p u(L, p, 1) = v_L < \arg\max_p u(H, p, -1) \leq v_H$

The first condition says that when the product quality is low, the seller would rather the buyer buy (e.g., if production costs are 0). The second says that the seller would rather have the buyers

[8] For a fixed σ, $y^R(\sigma, \cdot) : \mathcal{P} \to A$ is a strategy of the agent, satisfying sequential rationality.

not buy when the product quality is high (e.g., if the buyers serviced by the algorithm are costly to serve, at least on average).

The last condition suggests how the seller would *like* the algorithm to err—ideally, the seller would prefer (1) the buyer purchase when $\theta = L$, charging a low price, and (2) the buyer *not* purchase when $\theta = H$, while choosing a high price. Importantly, such a policy is not implementable as a unique equilibrium with rational buyers—in that case, the high price would reveal high quality.

To conclude, we note that one can show that σ^R randomizes over at most two prices. $\mathcal{H}^R$ consists of the set of all *increasing single-threshold strategies*:

$$h_{\overline{v},k} = \begin{cases} 1 & p \geq \overline{v} \\ -1 & p < \overline{v} \end{cases}, \text{ where } k \in \{-1, 1\}.$$

The set of single-threshold strategies will play a role in our main results below. For now, we simply note that this set contains all rational replies the buyer may use, to any optimal seller strategy.

2.2. Algorithm Game (i.e., the Supergame)

Our interest is in whether rational replies can emerge as a long run outcome in repeated play of the stage game. We dub the corresponding supergame an *algorithm game*. Specifically, we consider a repetition of the stage game interaction, played over discrete time $t = 1, 2, \ldots$, where the stage game interactions occur at every $t \geq 1$. We will take the true state θ to be drawn IID across periods according to π; we let (p_t, a_t) denote the actions of the Sender and Receiver, respectively.

2.2.1. Defining Algorithms

Throughout this paper, we assume that the Sender (principal) is fully rational, but the strategic choice of the Receiver (agent) must be delegated to an algorithm. The algorithm will take in the set of *histories*, a sequence of outcomes that occurred during the game. We denote the set of histories by $\mathcal{D}$, and denote the particular history of outcomes until time t by D_t. Our main results assume:

$$D_t = \{(p_1, \theta_1), \ldots, (p_t, \theta_t)\}$$

However, as an intermediate step, we will discuss the case where $y^R(\sigma, p)$, the rational reply to σ given price p, is observed instead of θ.

The assumption that θ is observed is less strong than it may initially appear; a seemingly more realistic assumption could be that the algorithm only observes the Receiver's own payoffs, possibly without revealing the whole state θ itself. While in some cases it may be reasonable to assume the state is observed ex-post, our main purpose in assuming θ is observed is to avoid considering experimentation motives for the algorithm designer, which would significantly complicate our main message. Furthermore, richer model setups could accommodate this case more easily. For instance, if N Receivers arrived in every period, then the algorithm could recommend $|A|$ of them to always select some fixed acton, and recommend an optimal reply to all others. Doing this would make $v(\theta, p, a)$ observable for all a, conveying the same information as observing θ itself. Of course, numerous other questions might emerge regarding how to optimally learn about each period's θ; however, these questions would take us too far afield.

We now define the algorithm designer's strategy space, $\mathcal{T}$, and choices $\tau \in \mathcal{T}$:

Definition 1. *Let Γ be some fixed set of possible (Receiver) strategies. An algorithm is a function*

$$\tau : \mathcal{D} \to \Gamma.$$

where $\mathcal{D}$ is a set of histories.

We let $\mathcal{T}$ denote the set of feasible algorithms (i.e., a subset of the set of functions from $\mathcal{D}$ into Γ); thus $\mathcal{T}$ represents the choice set of the algorithm designer, perhaps given constraints on possible algorithms. In many cases, $\mathcal{T}$ may be retricted, or only implicitly defined[9]; while we consider economically meaningful restrictions on algorithms, we defer a complete description of them until our discussion of algorithm desiderata.

Our main interest in this paper is in understanding which kinds of Γ and $\mathcal{T}$ enable the Receiver to approximate the rational label, y^R.

2.2.2. Timing and Objectives

An algorithm game is a *simultaneous move* game between the (rational) Sender and an *algorithm designer*, in which the strategies dictating play at times $t = 1, 2, \ldots$ are determined:

A_{-1}. According to some prior distribution, Nature selects the distribution π of the underlying game from a set Π, where $\Pi \subset \Delta(\Theta)$ only includes distributions with finite support.[10]

A_0. Conditioned on π, the Sender commits to some $\sigma \in \Sigma$. The Algorithm Designer commits to an algorithm $\tau \in \mathcal{T}$ without observing the realized $\pi \in \Pi$. These are chosen simultaneously.

[9] For instance, below we define *recursive ensemble algorithms*, which is one such implicit restriction $\mathcal{T}$

[10] Notice that we do not necessarily assume that any pair $\pi_1, \pi_2 \in \Pi$ have intersecting or even overlapping support (though this is also certainly allowed). Correspondingly, we emphasize we do not assume Θ is itself finite, even though all $\pi \in \Pi$ have finite support. If $|\Theta| = \infty$, clearly Π will need to be a strict subset of $\Delta(\Theta)$; if $|\Theta| < \infty$, Π may or may not be a strict subset of $\Delta(\Theta)$.

A_1. The stage game is played at $t = 1, 2, \ldots$, with (θ_t, p_t) drawn IID across periods, where $\theta_t \sim \pi$ and $p_t \sim \sigma(\cdot \mid \theta_t)$. The Receiver's strategy in each period t is $\tau(D_t)$, yielding his action $\tau(D_t)(p)$. The observation (i.e., p_t and θ_t) is added to the dataset each period.

We conclude by specifying payoffs in the algorithm game. Given a sequence (θ_t, p_t, a_t), both the rational Sender and the algorithm designer are simply the long run average expected payoffs:

$$\mathcal{U}_s(\sigma, \tau) = \lim_{T \to \infty} \frac{1}{T} \sum_{t=1}^{T} \mathbf{E} u(\theta_t, p_t, a_t), \quad \text{and}\, \mathcal{U}_r(\sigma, \tau) = \lim_{T \to \infty} \frac{1}{T} \sum_{t=1}^{T} \mathbf{E} v(\theta_t, p_t, a_t).$$

where (p_t, a_t) is generated by (σ, τ) in period t and the expectation is otherwise conditioned only on $\pi \in \Pi$ (recalling that θ is taken to be drawn IID). The objective of the rational Sender is to maximize $\mathcal{U}_s$ by choosing σ, conditioned on $\pi \in \Pi$.

2.2.3. Revisiting the Running Example and the Possibility of Algorithm Exploitation

Before we turn to a more complete description of how we describe algorithms and $\mathcal{T}$, we continue our running example to illustrate the horse race described in the introduction. A natural conjecture about how to specify Γ, as simply as possible, would be the following:

1. Find a simple class of strategies which includes every possible equilibrium buyer strategy.
2. Pick the strategy within this class that minimizes the probability of making a mistake.

If the seller's strategy were exogenously given by σ^R, such a scheme would work. In this case, recall that the set of single-threshold strategies contains all rational replies of the Receiver. However, with endogenous σ^R, such a scheme is susceptible to exploitation:

Proposition 1. *Suppose that Γ consists of the set of single-threshold strategies. There exists u, v, Θ and $\pi \in \Delta(\Theta)$ such that no algorithm can construct an optimal reply to the seller's optimal on-path strategy in the Monopoly Model (Section 2.1.3).*

The intuition behind the result is simple. If the buyer can only use a single-threshold strategy, the seller can choose some σ such that the buyer's best reply would fall outside of this class. This is done by introducing an "intermediate price" where a large loss would be incurred, though this price only emerges with low probability. This introduction makes the the buyer unable to approximate the rational reply; and in particular, this inability resolves by having the buyer not purchase at a high price, which Section 2.1.3 indicated would be the seller's preferred outcome.

In the construction used in the previous proposition, the rational reply would require two thresholds—buying at extreme intervals, and not in an intermediate interval. So perhaps the algorithm can simply enrich Γ? Say, to include two-threshold strategies? While this would succeed against the particular seller strategy used to exploit single-threshold buyers, the seller still might "one-up" the buyer to make sure the rational reply falls *just outside* of the set under consideration, at least when some other distributions π are possible:[11]

Proposition 2. *Suppose the buyer is restricted to using an n-threshold strategy, for n arbitrary but finite.*[12] *Then there exists u, v, Θ and $\pi \in \Delta(\Theta)$ such that no algorithm can construct an optimal reply to the seller's optimal on-path strategy in the Monopoly Model (Section 2.1.3).*

The previous two propositions suggests the possibility of a "horse race" between the seller and the algorithm designer—the algorithm designer might simply try to expand the set of possible strategies, but the seller could always use a strategy which necessitates a class of strategies that including one more threshold. This general principle motivates our exercise and our main results. In order to ensure that the buyer always chooses a rational strategy, the previous propositions suggests that it's necessary to add more and more strategies to the set of feasible strategies. Even if "on-path" a rational Receiver uses a simple strategy (e.g., given by a single-threshold), for the rational outcome to be obtained, optimal actions must emerge *both on- and off- path*. Thus, in principle, the set of strategies needed to obtain rationality could be *much* larger than the set of strategies that would actually be used; sometimes, so dramatically that one could not hope to approximate a rational reply given any finite amount of data.

There is a way out, however, and it uses the structure of the algorithm game. Instead of seeking to *exogenously* increase the set of strategies which could be optimal, instead we have the strategy set *endogenously* expand in response to the seller's actions. In other words, we will show how to specify an algorithm so that the set of strategies possible *reacts* to the seller's strategy, rather than the other way around (as in the previous propositions). In order to precisely describe how this is done and why this is a less demanding alternative than simply directly considering richer strategy spaces as possible, we need some additional notions which we introduce in the next section.[13]

[11] The environment of Rubinstein (1993) is actually *not* sufficient to deliver this strengthening; instead, in the proof we use a slight modification of it to show that the horse race always emerges, although our construction requires that *two* distributions for π might emerge.

[12] An n-threshold strategy is defined by n numbers, say $v_1, \ldots, v_n$, such that the buyer uses the same action at all $p \in (v_i, v_{i+1})$, taking $v_0 = -\infty$ and $v_{n+1} = \infty$

[13] Aside from our accommodation of more general environments than Rubinstein (1993), the key distinction with that paper is its assumption that (1) the set of strategies a buyer can use is limited, but (2) given the seller's strategy, the optimal decision rule is automatically chosen. By contrast, our exercise allows (1) algorithms to *construct* more elaborate strategies, despite only being able to *find* an optimal rule within a limited set (as we clarify in the next section), and (2) the optimal rule to be determined using historical data, and not by fiat.

To conclude, we briefly record that, given some specifications, it may indeed be possible to design an algorithm which *outperforms* rationality:

Proposition 3. *There exists a specification of the monopoly model (i.e., a choice of u, v, Θ and $\pi \in \Delta(\Theta)$) such that the second mover can outperform $y^R(\sigma^R, p)$ by using $\Gamma \subset \mathcal{H}^R$.*

The specifications which deliver this are ones where the algorithm only needs to find an optimal reply in a single environment. In these cases, the algorithm can outperform rationality by committing to a particular decision rule which would not itself be a best reply. However, these proposals would be susceptible to details of the enviornment; provided a sufficiently rich set of strategies might potentially be necessary, such restrictions may very well backfire for some particular $\pi \in \Pi$.

3. Algorithm Desiderata

The substance of our exercise is to reconcile two *competing* objectives:

- The algorithm should approach the rational benchmark sufficiently quickly, and
- The algorithm is not too computationally demanding.

The tension between these objectives is clarified by our running example (in Sections 2.1.3 and 2.2.3). In that application, a computationally simple algorithm might restrict Γ to be the set of single-threshold classifiers; but as Propositions 1 and 2 showed, this is insufficient to avoid the horse race and approximate rationality.

We now formally describe each of these targets, and describe how these desiderata relate to one another. In doing so, we provide more details of how we specify algorithms τ and the set of algorithms $\mathcal{T}$.

3.1. PAC Learnability

The first desideratum of an algorithm relates to the amount of data necessary for the algorithm to achieve a close approximation. To motivate this, suppose the algorithm does, in fact, provide an incorrect recommendation. One could attribute this to two different kinds of failings:

- The particular history realized did not enable an approximation of the right strategy, or
- When drawing $p \sim \sigma$ to evaluate performance, the particular p realization was not one for which the produced strategy gave the correct recommendation.

Our first desideratum essentially states that the probability of both of these events is small; both converge to 0 at a rate that is exponential in the amount of data available:

Definition 2. *A strategy set Γ is PAC-Learnable*[14] *if there exists an algorithm τ such that the following inequality holds:*

$$\lim_{t \to \infty} \mathbb{P}_{D_t}[\mathbb{P}_{p \sim \sigma}[\tau(D_t)(p) \neq y(p) \mid D_t] < \epsilon] > 1 - \delta,$$

for any $y : \mathcal{P} \to A$, $\sigma \in \Delta(\mathcal{P})$, $\epsilon \in [0, 1/2)$ and $\delta \in [0, 1)$. In this case, we say the algorithm achieves a PAC-guarantee, and we call the algorithm τ a PAC-learning algorithm. We say Γ is efficient PAC-learnable if (1) there exists a polynomial in $1/\epsilon$ and $1/\delta$, say $T(1/\epsilon, 1/\delta)$, such that whenever $t \geq T(1/\epsilon, 1/\delta)$, the previous condition holds, and (2) the number of computations of the algorithm τ is polynomial as well.

According to this definition, the output of the algorithm would be "approximately" correct—in that it gives the optimal prediction—with high probability, given suffficient data. The efficiency part of the previous definition essentially requires a particular (polynomial) rate on the amount of data necessary for an algorithm to provide a good answer.

We emphasize that, the *realized* convergence rate of the algorithm will depend only on what the Sender does on-path. While we seek some guarantee for every possible $\pi \in \Pi \subset \Delta(\Theta)$, the particular rate of convergence may depend on the problem (and, of course, endogeneity of σ). The fact that the dataset $(D_t)_{t \in \mathbb{N}}$ will depend on algorithm capabilities makes the realized rate of the algorithm endogenously determined, a distinguishing feature of our exercise. However, notice that whether or not the PAC-condition holds does *not* depend on which Sender strategy is chosen on-path—this is because we require the inequality to hold *conditional on the data generating process, for all data generating processes*—and, therefore, all possible σ^R (in other words, *all* possible Sender choices, regardless of whether or not they are chosen).[15]

Along these lines, a useful observation for our subsequent analysis is that the PAC requirement essentially strengthens rationality—that is, PAC learnability is a sufficient condition for the algorithm game to have a Nash equilibrium which approximates $(\sigma^R, y^R(\sigma^R, p))$:

Proposition 4. *If τ is a PAC learning algorithm, the Sender's optimal strategy is σ^R and the Receiver's long run payoff is as if $y^R(\sigma^R, p)$ were played in every period.*

[14] For more background on this notion, see Shalev-Shwartz and Ben-David (2014), for instance.

[15] Notice also this definition assumes it is possible to actually approximate $y^R(\sigma, p)$ using some element from Γ. A related concept which does not use this assumption is *agnostic learnability*; as this plays no role in our analysis, however, we do not discuss it further, but see Shalev-Shwartz and Ben-David (2014) for more details.

Proof. If τ is a PAC learning algorithm, then the Receiver learns σ accurately in the long run, for any possible Sender choice σ. Thus, the long run average expected payoff of the Sender is

$$\mathcal{U}(\sigma, \tau) = \mathbf{E}_\theta u(\theta, \sigma, y^R(\sigma, \sigma(\theta)))$$

By the definition,

$$\sigma^R \in \arg\max \mathbf{E}_\theta u(\theta, \sigma, y^R(\sigma, \sigma(\theta))).$$

By PAC learnability,

$$\lim_{t \to \infty} \mathbb{P}[\tau(D_t)(p) \neq y^R(\sigma^R, p)] = 0,$$

implying that $\mathbb{E}[v(\theta_t, p_t, a_t)] \to \mathbb{E}[v(\theta_t, p_t, y^R(\sigma^R, p)]$ as $t \to \infty$; the implication holds due to the assumption that π has finite support, meaning that the payoff is bounded. This implies the long run payoffs are equal to those obtained against a rational player, as desired. □

3.2. Computational Constraints

We now describe a particular way of specifying $\mathcal{T}$ which will allow us to describe our notion of algorithm complexity. We posit that the algorithm can solve the following problem (e.g., via a computer program), for fixed set of strategies $\mathcal{H}$ and arbitrary $d \in \Delta(\mathcal{P})$ and $y : \mathcal{P} \to A$:

$$\max_{h \in \mathcal{H}} \quad \sum_p \mathbf{1}[h(p) = y(p)]d(p), \tag{3.1}$$

We refer to this step as finding the *best fitting strategy*, and treat it as a black box. As alluded to above, we refer to the set $\mathcal{H}$ to be the set of *baseline strategies*.

The larger the set $\mathcal{H}$, the harder it is to solve this maximization problem. The crux behind our notion of computational simplicity is that we desire the set $\mathcal{H}$ to be minimal—it does contain enough strategies to find a rational reply, but also does not contain more strategies than it needs to do so.[16] Put differently, our notion of complexity is one whereby an algorithm is more complex if it has the ability to solve more difficult versions of (3.1), where the added difficulty emerges due to more possible maximizing candidates to check.

The simplest set of baseline strategies we will generally be able to consider is the set of *hyperplane strategies*:

[16] In this paper, to maintain focus we only define *relative* complexity of $\mathcal{H}$ using subset inclusion. We do not discuss formal measures of the size of $\mathcal{H}$, though past versions of this paper appealed to the notion of VC-dimension to do so. We refer interested readers to Al-Najjar (2009) and Basu and Echenique (2019) for more on this concept.

Definition 3. *A single threshold (linear) strategy is a mapping*

$$h : \mathcal{P} \to A$$

where $\exists a_+, a_- \in A$*,* $\lambda \in \mathbb{R}^n$ *and* $\omega \in \mathbb{R}$ *such that*

$$h(p) = \begin{cases} a_+ & \text{if } p \in \{\tilde{p} \in \mathbb{R}^n \ : \ \lambda\tilde{p} \geq \omega\} \\ a_- & \text{if } p \in \{\tilde{p} \in \mathbb{R}^n \ : \ \lambda\tilde{p} < \omega\}. \end{cases}$$

These strategies essentially generalize the single-threshold strategies discussed in the context of our running example in Sections 2.1.3 and 2.2.3. For environments like this one, this is the *simplest class of strategies* whereby it will be possible to approximate the rational reply. Notice that, provided the utility functions satisfy an appropriate notion of single-crossing given the particular environment, increasing single-threshold strategies will be optimal. The fact that we also will require "decreasing" single-threshold strategies is perhaps less obvious; however, as we discuss below, in order for our algorithm to work, we require *closedness under action permutation*,[17] which is satisfied by this strategy set.

Our analysis in Sections 2.1.3 and 2.2.3 shows that simply solving (3.1) for the set of hyperplane strategies is insufficient ensure rationality approximately obtains. We therefore specify how to endogenously expand the set of implementable second-mover strategies, without otherwise adding excessive complexity demands. We focus on the following class of strategies derived from $\mathcal{H}$:

Definition 4. *Strategy* H *is an ensemble of* $\mathcal{H}$ *if* $\exists h_1, \ldots, h_K \in \mathcal{H}$ *and* $\alpha_1, \ldots, \alpha_K \geq 0$ *such that*

$$H(\sigma, p) = \arg\max_{a} \sum_{k=1}^{K} \alpha_k \mathbf{1}[a = h_k(\sigma, p)]$$

We say that an algorithm is an ensemble algorithm if it produces an ensemble of $\mathcal{H}$*.*

We can interpret H as a weighted majority vote of $h_1, \ldots, h_K$, thus constructing a strategy as a *convex combination* of elements of $\mathcal{H}$.[18] Since the final strategy is constructed through a basic arithmetic operation, one can easily construct an more elaborate one from rudimentary strategies.[19]

Of course, one could try to add ensembles of $\mathcal{H}$ to $\mathcal{H}$ itself, but this would involve solving more difficult versions of (3.1). Instead, we focus on cases where (3.1) is solved for a fixed set $\mathcal{H}$,

[17] More formally, this requirement states that if $G : A \to A$ is a permutation, and $h(p) \in \mathcal{H}$, then $G(h(p)) \in \mathcal{H}$.

[18] Without loss of generality, we can assume that $\sum_{k=1}^{K} \alpha_k = 1$, since if not we can simply divide by this sum and obtain the same strategy.

[19] Ensemble algorithms have been remarkably successful in real world applications (Dietterich (2000)).

but the possible strategies to implement are the outcome of a *recursive ensemble algorithm*:

Definition 5. *An algorithm τ is a* ***recursive ensemble algorithm*** *using $\mathcal{H}$ if:*

- *The final strategy produced is an ensemble of $\mathcal{H}$, and*
- *Each strategy $h_k \in \mathcal{H}$ is found by solving (3.1), using some distribution $d_k(p) \in \Delta(\mathcal{P})$, and*
- *Either $k = 1$, or $\{d_k(p)\}_{p\in\mathcal{P}}$ is a function of $y^R(\sigma, \cdot), h_{k-1}, \alpha_{k-1}$, and $\{d_{k-1}(p)\}_{p\in\mathcal{P}}$ alone.*

3.3. Adaptive Boosting as a Simple Recursive Ensemble Algorithm

Before proceeding to our main results, we review the Adaptive Boosting algorithm (see also Schapire and Freund (2012)), henceforth referred to as AdaBoost. This will both illustrate the notion of a recursive ensemble algorithm, and specify the workhorse algorithm we will use for our main results. While the algorithm can be generalized, below we focus on the case where $A = \{-1, 1\}$.

As per Definition 5, we take d_1 to be the uniform distribution over $\mathcal{P}$, with the algorithm specified as follows:

1. Given a distribution d_k over $\mathcal{P}$, let $h_k(p)$ be the strategy which solves (3.1), and define

$$\epsilon_k = \mathbb{P}_{d_k}\left(h_k(p) \neq y^R(\sigma, p)\right) \tag{3.2}$$

as the probability that the optimal strategy h_k at k misclassifies p under d_k.

2. If $\epsilon_k = 0$, then the algorithm has returned a perfect forecast, and thus the algorithm will return output h_k as the final forecasting rule.

3. Otherwise, $\epsilon_k > 0$, and thus set:

$$\alpha_k = \frac{1}{2}\log\frac{1-\epsilon_k}{\epsilon_k} \tag{3.3}$$

4. Finally, for each p in the support of σ,

$$d_{k+1}(p) = \frac{d_k(p)\exp(-\alpha_k y^R(\sigma, p)h_k(p))}{Z_k}$$

where

$$Z_k = \sum_p d_k(p)\exp(-\alpha_k y^R(\sigma, p)h_k(p)).$$

Provided the algorithm never terminates at step 2 above, then it terminates at some fixed K, specified in advance, and the final Receiver strategy produced by the algorithm is:

$$\tau_A(D_k)(p) = \arg\max_{a \in A} \sum_{t=1}^{k} \alpha_t \mathbf{1}(h_t(p) = a).$$

Given some termination rule, this algorithm produces an ensemble of $\mathcal{H}$. While this ensemble will typically not be an element of $\mathcal{H}$, it does not require solving a problem any more computationally difficult than (3.1).

4. Statements of the Main Results

We now exhibit an equilibrium of the algorithm game where second mover's reply is rational, and the first mover's strategy is chosen as-if the second mover were rational. The algorithm we highlight is computationally simple, in the sense described in Section 3.2, and achieves a PAC guarantee as well. In fact, the backbone of our proposal is AdaBoost, as specified in the previous Section 3.3. Recall that by Proposition 4, showing that an algorithm is a PAC-learning algorithm implies that the strategy that is produced in the long run is indeed $y^R(\sigma, p)$.

As a warmup, we first illustrate that AdaBoost would achieve our goal if $y^R(\sigma, p)$ were observed and $|\mathcal{P}| < \infty$. Recall that Σ is the set of feasible Sender strategies:

Lemma 1. *Suppose the values of $y^R(\sigma, p)$ are observed by the algorithm, $\forall(\sigma, p)$, and that Σ only contains strategies with finite support. Then there exists an algorithm which*

- *Is a recursive ensemble algorithm,*
- *Uses $\mathcal{H}$ equal to the set of single-threshold classifiers, and*
- *achieves a PAC-guarantee,*

We denote the corresponding algorithm by τ_A.

The case where $y^R(\sigma, p)$ is observed without noise is the dominant case in machine learning classification problems (though we discuss this more thoroughly in Section 6.2). The novel feature of *our* problem is that the algorithm needs to infer $y^R(\sigma, p)$ from past observations, due to the presence of strategic inference. This will yield an algorithm $\tau_{\hat{A}}$, which coincides with τ_A with the added step of inferring the rational replies. This requires a slightly stronger assumption on the stage games, but nevertheless we can still show that we obtain an analogous result as would be obtained with observed optimal $y^R(\sigma, p)$.

Theorem 1. *Suppose Σ only contains strategies with finite support,*[20] *that $y^R(\sigma, p)$ is a strict best response $\forall \sigma$. Then there exists an algorithm which*

- *Is a recursive ensemble algorithm,*
- *Uses $\mathcal{H}$ equal to the set of single-threshold classifiers, and*
- *achieves a PAC-guarantee,*

Denoting this algorithm by $\tau_{\hat{A}}$, $(\sigma^R, \tau_{\hat{A}})$ thereby forms a Nash equilibrium of the algorithm game, which achieves identical long run payoffs as (σ^R, y^R).

The strict best response condition ensures that the unique value of $y^R(\sigma, p)$ *must* be learned in order to achieve a strict best reply. In the applications we study below, we verify that it indeed does hold. If multiple values of $y^R(\sigma, p)$ are optimal, then it may be that the Sender deviates from σ in order to break the Receiver's indifference in a particular way. The following example illustrates the difficulty that might emerge if this assumption does not hold, and why an improvement is likely not possible:

Example 1. *Consider the following version of the ultimatum game, and let us take for granted that the Receiver uses an algorithm which computes an optimal reply to the empirical distribution (as we show exists). The Sender offers the Receiver a payment, say $\mathcal{P} = \{0, 1/10, 2/10, \ldots, 1\}$. The Receiver can accept ($a = 1$) or reject ($a = 0$) the offer. The state θ is equally likely to be -1 or 1. The Sender's payoff is $u(\theta, p, a) = 1 - p$, while Receiver's payoff is $v(\theta, p, 0) = 0$ and $v(\theta, p, 1) = \theta + p$.*

When the distribution over θ is known, backwards induction shows that the Sender must choose $p = 0$ with probability 1, with the Receiver accepting the offer. This is the unique subgame perfect equilibrium outcome. However, the Receiver's payoff would be unchanged when rejecting the offer, since the rational Receiver would be indifferent.

When the distribution over θ is not known, however, notice that for all finite time horizons, it is equally likely that there are more $\theta = 1$ observations than $\theta = -1$ observations. Therefore, in expectation, the Receiver will choose $a = 1$ half the time and $a = -1$ half the time in any odd-numbered period; if the Receiver always chose $a = 1$ with an even number of periods (when a tie could conceivably emerge), the long run payoff of the sender would be $3/4$, making this an upper bound from σ^R.

In this case, the Sender would have a strictly profitable deviation—by choosing $p = 1/10$ with probability 1, given a sufficiently large amount of data, the Receiver will find it strictly optimal to choose $a = 1$ (eventually). Therefore, the Sender obtains long run payoff of $9/10$, strictly larger than what they obtain by playing σ^R.

[20] Note that while we require the support to be finite, we do not require a uniform bound over the set of all strategies.

More generally, the example shows that whenever there are multiple best replies, subgame perfection might require that one of them is never taken in the case of indifference, in order to induce particular behavior from the first mover. As the example shows, this could lead to quite a different outcome when the distribution over θ needs to be estimated.

In addition to the strict best response condition, the previous result also requires a restriction on the first mover's strategy space. There are two issues. First, we need to be able to ensure that enough observations for a given $p \in \mathcal{P}$ in order to derive reliable estimates of the probability distribution over Θ given p. For continuous probability distributions, the probability that algorithm observes $p_t = p_s$ for $s \neq t$ is 0. The second issue is that the technical conditions necessary to apply Lemma 1 need not be satisfied in this case either.

Using some of the economic features of the environment we are interested in, we deal with this possibility in the following way: First, we imagine that the Sender's strategy is subject to a possibly arbitrarily small implementation shock, which prevents them from using an arbitrary strategy $\sigma \in \Delta(\mathcal{P})$ exactly.[21] On the other hand, given a sufficiently small shock, they still approximate the original distribution arbitrarily well. However, the presence of these implementation shocks allows us to specify how to group observations together, by ensuring that the expected payoffs from a certain action do not change arbitrarily quickly in p.

Theorem 2. *Let $(z_1, \dots, z_n) \sim \phi_\eta$, where ϕ_η is a continuously differentiable density supported in the η-ball around 0.*[22] *Suppose the first mover's action is generated as $p + z$, where $p \sim \sigma(\cdot \mid \theta) \in \Delta(\mathcal{P})$ is either continuous or discrete and $z \sim \phi_\eta$. Then, given any ε, a PAC-guarantee of the ε-optimal reply can be obtained by partitioning $\mathcal{P}$ into finitely many regions and restricting the second mover's actions to be constant within each region (using a recursive ensemble algorithm with $\mathcal{H}$ as the set of single-threshold clasifiers, as in Theorem 1).*

Note that it follows from this result if $y^R(\sigma, p)$ is a strict best reply for all σ, the same conclusion of Theorem 1 holds as well.

Before we turn to some additional applications and an overview of the proofs of the main results, we briefly comment that these results actually understate how quickly convergence may be obtained. For instance, in our running example, σ^R is a deterministic function. For this reason, however, it is reassuring that we do *not* need to make the set $\mathcal{H}$ in (3.1) excessively large, even though Section 2.2.3 suggested that we might. In other words, an attractive property of

[21] The use of such shocks meant to smooth out a potentially ill-behaved distribution was previously studied in a general equilibrium context by Anderson and Sonnenschein (1982) and Anderson and Sonnenschein (1985). The context is quite different and their role is distinct as well—here, they emerge due to the need to satisfy weak learnability as well as ensure sufficient data on θ is aggregated.

[22] For instance, $\phi_\eta(z) = \frac{1}{K} \exp\left(-\frac{1}{1-|z/\eta|^2}\right) \frac{1}{\eta^{|A|-1}}$, satisfies these properties.

our algorithm is that it *minimizes* the set of baseline strategies. This conclusion contrasts to Propositions 1 and 2, which suggested that richer strategies were necessary to achieve rationality. Rationality can be achieved without increasing the computational difficulty of (3.1).

5. Applications

Before turning to the proofs of these results, we describe some other interactions which our model speaks to. This will articulate the added richness relative to the Rubinstein (1993) setting, and also show the usefulness of our results. Indeed, in order for our algorithm to work, the only necessary condition is on $y^R(\sigma, p)$ being a strict best reply. In these cases, if σ is simple (e.g., is deterministic), then the algorithm will achieve a much better rate relative to when σ is arbitrary.

5.1. Wrapping up the Running Example

We briefly record that the strict best response condition is satisfied in the monopoly market example in Sections 2.1.3 and 2.2.3.

Lemma 2. *Consider the monopoly market model as specified by Rubinstein (1993).*[23] *Fix σ which assigns $p > v_L$ with positive probability, satisfying*

$$\mathbf{E}_\theta v(\theta, p, 1) \geq 0. \tag{5.4}$$

Then, the ex-ante expected profit of the principal against $\tau_{\hat{A}}$ from σ is strictly smaller than from σ':

$$\mathcal{U}(\sigma^R, \tau_{\hat{A}}) > \mathcal{U}(\sigma, \tau_{\hat{A}}).$$

This Lemma shows that, given the algorithm $\tau_{\hat{A}}$, the outcome of the interaction will (at least eventually) coincide with the rational outcome, overturning the implication of Rubinstein (1993). The proof essentially amounts to checking the strict best response condition of Theorem 1.

5.2. Labor Markets

For an application with more than two possible Receiver actions,[24] we us consider a labor market signaling model. Here, the Receiver takes the role of the firm and the Sender takes the role of the worker from the Spence signaling model (see also Maskin and Tirole (1992)). The true state is the

[23] This proof uses the precise payoff specification of Rubinstein (1993), which our prior exposition does not include. For completeness, this can be found in the Appendix.

[24] While our results are stated for the $|A| = 2$ case, we show how to construct algorithms for this case in Section 6.3.1.

productivity of the worker $\theta \in \Theta = \{H, L\}$. Conditioned on θ, a worker chooses p which we interpret as education level. Her strategy is

$$\sigma : \Theta \to \mathcal{P} \subset \mathbb{R}_+.$$

The payoff function of the Sender (worker) is

$$u(\theta, p, a) = a - \frac{p}{\theta + 1}$$

We abstract away the competition among multiple firms in the labor market. Conditioned on p, the labor market wage is determined according to the expected productivity $\mathbf{E}(\theta : p)$ conditioned on p. The firm has to pay the worker the equal amount of the expected productivity because of (un-modeled) competition among firms. The Receiver's goal is to make an accurate forecast about the expected productivity of the worker. The payoff of the Receiver is

$$v(\theta, p, a) = -(\theta - a)^2$$

If the support of $\sigma(p : H)$ is disjoint from the support of $\sigma(p : L)$, σ is a separating strategy. If a separating strategy is an equilibrium strategy, then the equilibrium is called a separating equilibrium. The literature often focuses on the Riley outcome, which maximizes the ex-ante expected payoff of the principal among all separating equilibria.

5.2.1. Analysis

The firm seeks to predict worker quality. Hence if $A = \mathbb{R}$, then

$$y^R(\sigma, p) = \arg\max_{a \in A} \mathbf{E}_\theta \left[v(\theta, p, a) : p, \sigma \right]$$

where the posterior distribution over θ is calculated via Bayes rule from σ and the prior over θ. Strict concavity of v implies that $y^R(\sigma, p)$ is a strict best response $\forall \sigma, p$.

Since there are multiple actions in this example, single-threshold decision rules are parameterized by (a^+, a^-, p^0):

$$h(p) = \begin{cases} a^+ & \text{if } p \geq p^0 \\ a^- & \text{if } p < p^0. \end{cases}$$

In each round of the algorithm τ_A from Lemma 1, h_t solves

$$\max_{h \in \mathcal{H}} \mathbf{E}_\theta \left[v(\theta, p, a) : p, \sigma \right].$$

In the algorithm outlined in Theorem 1, we estimate the posterior distribution of σ conditioned on each p to construct $\tau_{\hat{A}}$. If the worker learns $y^R(\sigma, p)$ eventually $\forall \sigma, p$, then the principal's choice σ^R maximizes

$$\mathbf{E}[u(\theta, p, y^R(\sigma, p)) : \sigma, p] = \sum_{\theta} \sum_{p} u(\theta, p, y^R(\sigma, p))\sigma(p : \theta)\pi(\theta).$$

If σ^R entails separation by the high productivity worker, then the Riley outcome is the solution, that generates the largest ex-ante expected surplus for the principal among all separating equilibria. In order to satisfy the incentive constraint among different types of the principal, the principal with $\theta = H$ incurs the signaling cost. If the signaling cost outweighs the benefit of separation, then σ^R is the pooling equilibrium where both worker types take the minimal signal.

Theorem 1 requires $y^R(\sigma, p)$ to be a strict best response $\forall \sigma, p$. In general, this may not hold some σ and p in this example. Let us assume that

$$A = \{a_0 = 0, a_1, \ldots, a_J\}$$

and $a_i - a_{i-1} = \Delta > 0$ and $a_F = 1 + \sup \Theta > 0$. Although $y^R(\sigma, p)$ may not be a strict best response for some σ and p, the set of best responses contains at most 2 elements, which differ by $\Delta > 0$. Abusing notation, let $y^R(\sigma, p)$ be the set of best responses, if the agent has multiple best responses at p.

Therefore, applying the convergence result with this modification, we have $\exists T$ such that, $\forall t \geq T$,

$$\mathbb{P}\left(\exists y \in y^R(\sigma, p), y = \tau_{\hat{A}}(D_t)(p)\right) < e^{-\rho t}.$$

For a sufficiently small $\Delta > 0$, σ^R is either a strategy close to the Riley outcome, or the pooling equilibrium where both types of the principals choose the smallest value of p.

5.3. Insurance

The following is borrowed from Maskin and Tirole (1992). Suppose that the principal (Sender) is a shipping company seeking to purchase insurance from an insurance company, an agent (Receiver) that is seeking to delegate the decision of whether to offer the terms put forth by the shipping company. The principal seeks insurance every period, but faces risk (e.g., due to the location of shipping demand) that is idiosyncratic every period.

In this case, we imagine the principal choose terms within some compact set $\mathcal{P} \subset \mathbb{R}^2$, where $p = (x, q)$ denotes a policy which provides a payment x in the event of a loss, and costs an amount q. If $\theta \in \{L, H\}$ (with $L < H$) denotes the probability of a loss, then the principal's utility is:

$$u(\theta, p, a) = \begin{cases} (1-\theta) f(I-q) + \theta f(I-q-L+x) & a = 1 \\ (1-\theta) f(I) + \theta f(I-L) & a = -1 \end{cases},$$

for some concave f. The agent's utility is:

$$v(\theta, p, a) = \begin{cases} q - \theta x & a = 1 \\ 0 & a = -1 \end{cases}$$

It is natural to consider $\mathcal{P}$ whereby, against a rational buyer, the principal would seek a high level of insurance when risk is high (i.e., $\theta = H$), and avoid insurance when risk is low (i.e., $\theta = L$). In contrast, the agent's payoff may be decreasing in the quantity of insurance when $\theta = H$, while increasing in the quantity of insurance when $\theta = L$.

We now verify the implications of this observation on the Sender behavior in our particular examples, showing that this results in the Sender-preferred Stackleberg outcome is emerging. This requires us to verify the unique strict best reply condition for this setting.

5.3.1. Analysis

The decision problem of the agent is to identify each pair (x, q) of payment x and cost q as an acceptable contract $(a = 1)$ or not $(a = -1)$. The added difficulty in this example is that the action space is multidimensional. However, we still specify the baseline strategy space to be the set of single-threshold classifiers. That is, as per our algorithm, we assume access to strategies induced by hyperplanes:

$$h(x, q) = \begin{cases} 1 & \text{if } (x, q) \in \{(x, q) : \lambda_x x + \lambda_q q \geq \omega\} \\ -1 & \text{otherwise.} \end{cases}$$

We can construct $\tau_{\hat{A}}$ by estimating $\mathbf{E} v(\theta, p, a)$ for each (p, a).

To apply Theorem 1, it is necessary to show the strict best reply condition. This turns out to hold in our setting:

Lemma 3. *Suppose that σ assigns a positive probability to (x, q) where*

$$\mathbf{E}(q - \theta x : (q, x)) = 0$$

for $x > 0$. Then σ is not a best response to $\tau_{\hat{A}}$.

Following the same logic as in the previous example, we conclude that if σ is a best response to $\tau_{\hat{A}}$, then

$$\tau_{\hat{A}}(D_t)(q,x) = y^R(\sigma,(x,q))$$

with probability 1. A best reply σ to $\tau_{\hat{A}}$ yields a long-run outcome of $(\sigma^R, y^R(\sigma^R,p))$.

6. Overview of the Proofs of the Main Results

This section discusses the techniques used to prove the results from the previous section, with details being relegated to the Appendix. We start with the proof of Lemma 1; this essentially describes the condition necessary for the specification of AdaBoost to work, *weak learnability*. The main innovations in our framework relate to Theorems 1 and 2; the former emerges due to our focus on uncertainty in the first mover's type, a fundamental feature of settings featuring inference. The second emerges as we seek to relax finiteness restrictions on the Sender's strategy.

6.1. Specifying the Algorithm with observed $y^R(\sigma,p)$ (Lemma 1)

The sufficient condition for an arbitrary strategy to be approximatable by combining single-threshold strategies is *weak learnability*. Roughly speaking, weak learnability says that the optimally chosen strategy outperforms someone who had some very minimal knowledge of the rational reply. That is, it must be that using only the strategy set, one can do better than someone who made a random guess, provided this guess would be made correct with some arbitrarily small probability. While this may seem permissive—and indeed, it is certainly less stringent than requiring the ability to approximate the truth with high probability—the difficulty in achieving it is the fact that this guarantee must be uniform over all possible distributions. Formally:

Definition 6. *If $|A| = 2$, a strategy set $\mathcal{H}$ is weakly learnable if, for every distribution d over observations $p \in P(\sigma)$ and actions $y(p)$, the best fitting strategy satisfies:*

$$\sum_{p \in P(\sigma)} D(p)(\mathbf{1}[y(p) \neq h(p)] - \mathbf{1}[y(p) = h(p)]) \geq \rho.$$

If $|A| > 2$, a strategy set $\mathcal{H}$ is weakly learnable if, for every distribution d over observations $p \in P(\sigma)$ and actions $y(p)$, the best fitting strategy satisfies:

$$\sum_{p \in P(\sigma)} \mathbf{1}[\overline{h}(p) \neq y(p)]d(p) \leq \sum_{p \in P(\sigma)} \mathbb{E}_{\tilde{y} \sim B}[(1-\rho)\mathbf{1}[\tilde{y} \neq y(p)]]d(p),$$

for some $\rho > 0$ and some distribution B over A.

The second condition is a generalization of the first, though the first is perhaps more familiar from the machine learning literature (as most attention has focused on the case where there are at most two possible choices). This condition reflects the idea that the strategy randomly guesses the action according to some distribution B, but is "flipped to being correct" with probability ρ. For the $|A| > 2$ case, the right hand side describes the expected error in such a case, and the left hand side describes the error from the optimally chosen element of $\mathcal{H}$. If weak learnability fails, then *no* recursive ensemble algorithm can be built to approximate $y(p)$ based on $\mathcal{H}$ alone.[25] Perhaps more surprising is that it is tight, in that weak learnability is all that is required.

It turns out the set of single-threshold classifiers is weakly learnable:

Proposition 5. *The set of single-threshold classifiers satisfies the weak learnability condition of Definition 6.*

Proof. See Appendix A. □

The proof of the Appendix actually proves a stronger theorem which may be of independent interest, and a version of it is used later when we consider the $|A| > 2$ case as well: Any hypothesis class that *contains all action permutations* can at least match the random guess guarantee. The proof of this intermediate lemma uses a duality argument in order to show that no distribution can lead to a lower payoff when the "closed under action permutations" condition is satisfied. Importantly, however, this is true for *any* hypothesis class, including the trivial one. This observation allows us to show that the added richness of single-threshold classifiers provides the additional gain over random guessing.

Since the weak learnability condition is satisfied, following Schapire and Freund (2012), we can show that, for AdaBoost as specified in Section 3.3:

$$\mathbb{P}\left(\tau_A(D_t)(p) = y^R(\sigma, p)\right) \geq 1 - e^{-t\rho(G)} \tag{6.5}$$

for any mixed strategy σ, where ρ is some function (determined in the proof) and G is the number of elements in the support of σ.[26]

6.2. Inferring $y^R(\sigma, p)$ from $\{v(\theta_t, p_t, a)\}$ (Theorem 1)

Next, we drop the assumption that the Receiver observes $y^R(\sigma, p)$, and instead uses an estimate, which we denote by $\hat{y}_t(p)$. This modification introduces the possibility that $\hat{y}_t(p) \neq y^R(\sigma, p)$—the

[25] For example, imagine $\mathcal{H}$ only consists of trivial strategies (i.e., those that always choose a single action). A corollary of a result in Appendix A is that these strategies can do equally well as a random guesser. However, it is clear that they cannot do strictly better, as they are restricted to giving the same guess to all possible p, unlike a random guesser who is correct with an added probability ρ.

[26] A sketch of the proof is in Appendix B.

machine learning literature refers to this possibility as *classification noise* or *label noise*. Boosting algorithms like the ones we propose using are known to perform quite poorly in the presence of classification noise. While the reasons for this are somewhat technical, very briefly they relate to how the particular coefficients α_k above are chosen. Notice that ϵ_k in (3.2) requires the $y^R(\sigma, p)$ to be observed. Without label noise, the choices of α_k specified in Section 3.3 emerge from minimizing a particular convex function of the ϵ_ks which achieve the same minimum as the (non-convex) objective function that the algorithm would like to minimize. When label noise is present, this replacement function itself need not be convex (see, for instance, Long and Servedio (2010) or Freund (2009); Frenay and Verleysen (2014) provides a survey of the issue and proposed solutions).

What is our solution, and why does it get around this problem? In the papers referenced above, label noise is modelled as the possibility that there is some fixed probability ρ according to which the correct action is not chosen. By contrast, in our setting, label noise emerges because the algorithm would like to find the action a which maximizes:

$$\sum_{\theta} v(\theta, p, a)\pi(\theta : p) \tag{6.6}$$

where π is computed via Bayes rule. A natural proposal is to simply replace π by its sample analog, and indeed, this will be our proposal as well. Our proposed algorithm, $\tau_{\hat{A}}$ coincides with the one in the previous section, but where first $\hat{y}_t$ is estimated and $y^R(\sigma, p)$ is replaced with $\hat{y}_t(p)$. More precisely, we make three changes:

- Replace $y^R(\sigma, p)$ with $\hat{y}_t(p) = \arg\max \sum_{\theta} v(\theta, p, a)\hat{\pi}_t(\theta : p)$, where $\hat{\pi}_t(\theta : p)$ denotes the fraction of times state θ is observed when the Sender's choice was p,
- Choose h_k to solve:

  $$\max_{h \in \mathcal{H}} \sum_{p} h(p) d_t(p)[1 \cdot f_t^y(p) - 1 \cdot (1 - f_t^y(p))]$$

  instead of (3.1), where f^y is the empirical probability that $\hat{y}_t(p) = 1$ at the beginning of period t. Thus, $\hat{y}_t(p) = -1$ with probability $1 - f_t^y(p)$.
- Replace the errors ϵ_t with $\hat{\epsilon}_t$, where

  $$\hat{\epsilon}_t = \sum_{p} d_t(p) \left[f_t^y(p) \mathbf{1}(h(p) = 1) + (1 - f_t^y(p)) \mathbf{1}(h(p) = -1) \right].$$

The key feature that distinguishes our setting from other settings with classification noise is that for us the noise vanishes in the limit as $t \to \infty$. Of course, this property is only sensible in our setting due to the *origin* of the classification noise—namely, that the Sender might be randomizing their strategy, and making the ex-post payoff following some p possibly different from the expected payoff following p. Now, the observation that label noise vanishes by itself is not enough to eliminate this problem. Our goal is a PAC-guarantee, which requires not just that the noise in $\hat{y}_t(p)$ vanishes, but also that this vanishes quickly, *uniformly* over p. For our modification to work, $\hat{y}_t(p)$ must satisfy the large deviation property (LDP):

Definition 7. *$\hat{y}_t(p)$ satisfies large deviation properties (LDP) if $\exists \lambda > 0$ such that, $\forall p$ in the support of σ,*

$$\limsup_{t\to\infty} -\frac{1}{t}\log \mathbb{P}\left(y^R(\sigma,p) \neq \hat{y}_t(p)\right) \leq \lambda. \tag{6.7}$$

If an estimator satisfies LDP, the tail portion of the forecating error vanishes at an exponential rate, as the sample average of i.i.d. random variables converges to the population mean. If an estimator fails to satisfy LDP, the finite sample property of the estimator tends to be extremely erratic (Meyn (2007)). Most estimators in economics satisfy LDP.

With $\tau_{\hat{A}}$, we can construct action recommendations from data, and that for the hypothesis class of interest the weak learnability condition is satisfied. The last step to show the algorithm works, in the case where the set of possible p has finite support, is that the output of the algorithm will indeed converge to the rational reply, provided the weights are specified correctly.

Proposition 6. *Suppose that $\hat{y}_t$ satisfies uniform LDP and that $y^R(\sigma,p)$ is a strict best response $\forall p$. Then, $\forall \sigma$ that randomizes over G elements of $\mathcal{P}$, $\exists T$ and $\exists \rho(G) > 0$ such that*

$$\mathbb{P}\left(\tau_{\hat{A}}(D_t)(p) = y^R(\sigma,p)\,\forall t \geq T\right) \geq 1 - e^{-t\rho(G)}.$$

Proof. See Appendix B. □

6.3. Dropping Restrictions on Sender's Strategy (Theorem 2)

Our second main result shows how we can relax the restrictions on the seller's strategy space, at the expense of requiring an arbitrary small implementation error in the seller's strategy. We first describe the idea for the case where $y(\sigma,p)$ is observable, as this conveys the main intuition for the result. The case where $y(\sigma,p)$ is not observed and has to be inferred follows the same logic as Theorem 1, which we briefly comment on as well.

Letting $\tilde{\sigma}_\eta$ denote the distribution of $p+z$. The proof of Theorem 2 proceeds in the following steps:

- Step 1: Show that the expected value conditional on price under $\tilde{\sigma}_\eta$, in the image of the Sender's possible strategies, is uniformly equicontinuous.

- Step 2: Show that the same action recommended when the price is $p + z_i$ would be recommended when the price is p, with high probability.
- Step 3: Verify that the change in recommendation due to discarding "low density prices" occurs with vanishing probability.

Putting these together shows that the change in the expectation can be made arbitrarily small, as can the probability that small density observations are drawn. The condition that the distribution over p be either discrete or continuous is stronger than necessary; what is necessary is continuity of the conditional expectation as a function of price, which can be satisfied if the discrete portions and continuous portions are separated, for instance. The Theorem highlights that the only restrictions on the set of possible sender strategies are very mild in order to obtain this.

Thus, while it may be difficult to ensure rationality against an *arbitrary* strategy $\sigma \in \Delta(\mathcal{P})$, this can be done if instead the Receiver focuses on finding a rational response to $\tilde{\sigma}_\eta$. Since η can be arbitrarily small, we interpret this as saying arbitrarily small implementation errors could allow us to recover the main conclusion of Theorem 1, without restricting the Sender's strategy space.

When $y(\sigma, p)$ is not observed, it becomes necessary to determine it. Notice, however, that if the optimal reply at some p is to change such that some other action that is ε-optimal at p', then the change in p must be sufficiently large by uniform equicontinuity. Using this observation, we can essentially pool prices together so that we are ensured to get a large sample of θ observations. This leads to an estimate of $\pi(\theta : P_i)$ for some region P_i, rather than $\pi(\theta : p)$; nevertheless, uniform equicontinuity together with the relaxed requirement that the produced strategy be ε-optimal ensures that this expansion suffices.

6.3.1. On $|A| > 2$ Actions

The Adaptive Boosting algorithm, as introduced by Schapire and Freund (2012), guides how we combine classifiers, with our main results showing to extend this algorithm to address our economic application. The original Adaptive Boosting algorithm only applies to the case of $|A| = 2$. To handle the case of $|A| > 2$, we appeal to a generalization introduced by Mukherjee and Schapire (2013). The basic idea behind this algorithm is largely the same, with one minor drawback, which is that algorithm depends on the constant ρ in Definition 6, and thus this must be computed in advance. While our work shows an algorithm exists, the computation of the learnability constant is more indirect and hence explicitly finding a parameter that works is more difficult. The arguments for these proofs follow from results in the machine learning literature (see Schapire and Freund (2012)), which we can apply to show that this algorithm can yield a response for which the misclassification probability vanishes.

As for Proposition 6, its proof is stated for the general case. The proof reveals that the rate at which the probability of misclassfication vanishes is determined entirely by the number of Sender actions in the support of σ. Thus, the algorithm is efficient (in the sense of Definition 2) in that it maintains an exponential rate of convergence (Shalev-Shwartz and Ben-David (2014)).

7. Conclusion

7.1. Literature

This paper takes the framework of PAC learnability, familiar from machine learning, and applies it to a strategic setting. Within economics, this agenda is most closely related to the literature on learning in games when behavior depends on a statistical method. The single-agent problem is a particular special case, and this case is the focus of Al-Najjar (2009) and Al-Najjar and Pai (2014). However, since we are focused on a strategic setting, the data the algorithm receives is *endogenous* in our setting. In contrast, their benchmarks correspond to the case of exogenous data. This problem is also studied in Spiegler (2016), who focuses on causality and defines a solution concept for behavior that arises from individuals fitting a directed acyclic graph to past observations. Eliaz and Spiegler (Forthcoming) study the problem of a statistician estimating a model in order to help an agent take an action, motivated as we are by issues involved with the interaction between rational players and statistical algorithms. More recently, Zhao, Ke, Wang, and Hsieh (2020) take a decision-theoretic approach in a single-agent setting with lotteries, showing how a relaxation of the independence axiom leads to a neural-network representation of preferences.

As mentioned in the introduction, our structure of an algorithm is very close to the structure of a "machine game" used to study the outcomes that might emerge when automata implement strategies on behalf of individuals in repeated games. The introduction of automata is often motivated by a desire to compare the complexity of different strategies players might use, with strategies being dubbed more complex if they require more automata to implemented. Rubinstein (1986) studied the prisoner's dilemma in particular and Abreu and Rubinstein (1988) studied more general two-player games; however, a key difference between these papers is whether the added complexity is viewed as a flow cost (as it is in the former) or a fixed cost (as it is in the latter). Other important papers following these include Aumann and Sorin (1989) and Ben-Porath (1993), who extended and qualified some of their results; Piccione and Rubinstein (1993) consider repeated extensive form games, as we do. The structure of an algorithm game is similar, in the sense that both look at repeated interactions—several papers in this literature also focus on undiscounted payoffs, as we do here. On the one hand, we introduce exogenous uncertainty and require the algorithm to learn about it—introducing a novel inference element which is absent from these papers and a substantial technical focus. On the other hand, our restriction on the first mover's strategy to be rational and IID across periods is a constraint on the problem necessary for us to be able to make this inference problem well-behaved. We also mention that our notion of complexity is different, instead determined by the set of baseline classifiers (as opposed to the number of states). Despite these differences, we view the motivation as quite similar and also view a key contribution in extending this framework to study new economic questions, while simultaneously importing tools from machine learning in order to answer them.

Classic contributions to the literature on algorithm approximations of Nash behavior include Hart and Mas-Colell (2000), Foster and Vohra (1997) and Fudenberg and Levine (1995); see also Fudenberg and Levine (1998) for a textbook treatment of this area. Yet even recently, the literature has still for the most part focused on settings where the interactions between players is *static*, ruling out the main environments we are interested in here. In contrast, our setting is a simple, two-player (and two-move) sequential game. Cherry and Salant (2019) discuss a procedure whereby players' behavior arises from a statistical rule estimated by sampling past actions. This leads to an endogeneity issue similar to the one present in our environment, i.e., an interaction between the data generating process and the statistical method used to evaluate it. Liang (2018) also focuses on games of incomplete information, asking when a class of learning rules leads to rationalizable behavior. Studying model selection in econometrics, Olea, Ortoleva, Pai, and Prat (2019) consider an auction model and ask which statistical models achieve the highest confidence in results as a function of a particular dataset.[27] By contrast, we show that a version of the weak learnability condition in settings with two possible Receiver actions also applies to settings with an arbitrary finite number of actions. Our ability to handle this in our problem suggests our proposals are of broader interest. We believe that this extension is important, as it shows our conclusions do not hinge on other artificial limitations on the environment.

The literature on learning in *extensive* form games has typically assumed that agents experiment optimally, and hence embeds notion of rationality on the part of agents which we dispense with in this paper. Significant contributions include Fudenberg and Kreps (1995), Fudenberg and Levine (1993) and Fudenberg and Levine (2006). Most of this literature has focused on cases where there is no exogenous uncertainty regarding a player's type, and asking whether self-confirming behavior emerges as the outcome. An important exception is Fudenberg and He (2018), who study the steady-state outcomes from experimentation in a signaling game (see also Fudenberg and Clark (2021)). While a rational agent in our game would need to form an expectation over an exogenous random variable, issues related to off-path beliefs do not arise because our Sender has commitment.

[27] On the question of algorithms in particular, one concern is that the algorithm design problem may be susceptible to bias or induce unwanted discrimination when implemented, relative to rationality. See Rambachan, Kleinberg, Mullainathan, and Ludwig (2020) for an analysis of these issues and how they may be overcome.

Perhaps closest in motivation is the computer science literature studying algorithm perform in strategic situations. Braverman, Mao, Schneider, and Weinberg (2018) consider optimal pricing of a seller repeatedly selling to a single buyer who repeatedly uses a no-regret learning algorithm. They show that, on the one hand, while a particular class of learning algorithms (i.e., those that are *mean-based*) are susceptible to exploitation, others would lead to the seller's optimal strategy simply being to use the Myersonian optimum. Deng, Schneider, and Sivan (2019) also study strategies against no-regret learners in a broad class of games without uncertainty, and consider whether a strategic player can guarantee a higher payoff than what would be implied by first-mover advantage. Blum, Hajiaghayi, Ligett, and Roth (2008) consider the Price of Anarchy (i.e., the ratio between first-best welfare and worst-case equilibrium welfare), and show in a broad class of games that this quantity is the same whether players use Nash strategies or regret-minimizing ones. Nekipelov, Syrgkanis, and Tardos (2015) assume players in a repeated auction use a no-regret learning algorithm, making similar behavioral assumptions as we do here. Their interest is in inferring the set of rationalizable actions from data.

Though we seek to incorporate several aspects of this literature's conceptual framework, our problem has three notable differences. First, this literature typically assumes particular algorithms or principal objectives (such as no-regret learning) which differ from traditional Bayesian rationality. In contrast, we maintain a Bayesian rational objective for the seller, and also focus on an algorithm designer seeking to maximize expected payoffs. Second, we focus on relating the incentives of the rational player and *the algorithm's capabilities*, and study the extent to which different assumptions on the algorithm design problem influence the task of approximating rationality. Our main result articulates how different action spaces for the algorithm designer yield different results regarding whether and when the outcome will approximate the rational benchmark. Lastly, our general framework focuses on settings with strategic inference—that is, where the payoffs following a given first-mover action are state-dependent—and thus covers a set of single-agent applications which extend beyond particular pricing settings, where most (though admittedly not all) of this literature has focused. In particular, the settings discussed in this literature typically do not cover lemons markets settings, to the best of our knowledge.[28] As a

[28] An important exception is Camara, Hartline, and Johnsen (2020), who study an environment covering many of our same applications such as Bayesian Persuasion. However, they still maintain the other two distinguishing features, focusing on a regret objective for the principal, as well as particular no-regret assumptions for the agent. Still, we emphasize that both our paper and theirs focuses on environments where the principal/Sender chooses a state-dependent strategy. This leads to the aforementioned endogeneity between the data generating process (induced by the principal) and the choices of the algorithm/learner—this emerges due to the fact that the same Sender action may induce two distinct replies from the algorithm following two distinct Sender strategies. In their setting, this endogeneity motivates the use of "policy-regret" as an objective for the principal (due to their reinforcement learning approach to the principal's problem). While we do not use a regret objective for the principal, see Arora, Dekel, and Tewari (2012) and Arora, Dinitz, Marinov, and Mohri (2018) for more on the differences between these

result, new technical issues (e.g., dealing with residual uncertainty in the correct actions) is not addressed in these papers. Despite these differences, our hope is that this paper inspires further connection between the economics literature on decisionmakers as statisticians and the computer science literature on strategic choices against classes of algorithms. It appears to us that these results from computer science have not yet been fully appreciated in economics.

7.2. Discussion of Model Assumptions

We discuss a number of our key assumptions made above and highlight the role they played in the analysis.

Sender Rationality. Above, we assume that the stage game's first mover (Sender) is rational, whereas the second mover (Receiver) is algorithmic. This assumption makes the algorithm design question hardest in the sense that the set of possible Sender strategies is as large as possible. But this assumption is also useful since it allows us to ignore questions related to convergence of the *Sender's* strategy to equilibrium, and eliminates any corresponding noise or time dependence in the data generating process used by the (Receiver's) algorithm. Since there is a fundamental asymmetry between players in the class of games we study, addressing first-mover algorithms raises many orthogonal issues which would obscure our main message. For instance, for the question of inducing Sender rationality to be interesting, one should likely study a different move protocol—since, for us, the Receiver has no knowledge which the Sender would seek to adapt to, since the Sender observes π (and the Receiver does not).

Intertemporal Preference. We assume that the Sender is long-lived, while the algorithm acts on behalf of a sequence of short-lived Receivers, with both maximizing undiscounted payoffs. Prior versions of this paper considered the case where future payoffs were discounted at rate $\delta < 1$; the main lessons remain valid for δ sufficiently large, although there are some added technical difficulties in the analysis related to Theorem 2 this introduces.

Notice that if the Sender were instead short lived, then she would not necessarily benefit from exploiting the algorithm as in Sections 2.1.3 and 2.2.3.[29] Notice that we could have instead considered the case where there is also one long-lived Receiver, but which an algorithm makes choices on behalf of. For instance, an insurance company may seek to automate the approval/rejection

notions.

[29] The added difficulty with short-lived Senders is actually that, as discussed above, the short-term best reply would be different depending on how close the algorithm has approximated rationality—although, for the short-lived Senders to adapt to the algorithm choice, they would need to somehow know how much (and what) data the algorithm received, which require additional elements in the main model.

of contract offers, an application similar to what we discuss in Section 5.3.1. Whether there is a single Receiver or a sequence of Receivers is expositional, and which makes more sense will depend on the application.

Observed Data. We assume that the data at the end of each period is perfectly observed by the Receiver. For instance, Etsy or Amazon might be able to ask consumers exactly how much they liked a recommended product. Given sufficient consumer sophistication regarding seller strategies, it may be possible for the platform to determine y^R exactly, though this strikes us as demanding (and hence not required for our main results). Otherwise, the requirement is that consumers give accurate reports. The assumption that the payoffs following *all* actions are observed avoids the need to consider experimentation (so that Etsy would not need to suggest possibly suboptimal products to determine counterfactual payoffs).

For buyer-seller interactions, the assumption that data is observed by the algorithm is appropriate if consumers always reported to the algorithm after their purchase. But as mentioned above, our model also applies to cases where there is a single long-lived Receiver, in which case the assumption would simply be that the algorithm observes own-payoffs ex-post (as with the problem of granting approval, for instance, in an insurance or loan settings[30]). As discussed above, noise in the correct action ex-post is known to severely inhibit the performance of our class of algorithms, though this does not emerge in our setting since $u(\theta, a, p)$ is observed without noise.

Once-and-for-all Choices (and IID data). Our setting also assumes that the Sender makes a once-and-for-all choice of strategy σ. While the optimal strategy choice may be endogenous to the algorithm—in that the Sender would choose different σ if they anticipated a different τ—once σ is chosen, no further endogeneity issue emerges.

For our purposes, this assumption provides the cleanest comparison with the benchmark where the stage game is played once with rational players (where the stage game is as in Kamenica and Gentzkow (2011), possibly allowing Sender messages to influence Receiver utility). Our main message is that we can replicate this benchmark when the Receiver is algorithmic. How Senders can achieve commitment in Bayesian Persuasion is a well-known theoretical issue, which would take us too far afield to address (see, for instance, Best and Quigley (2020)). If the Sender could adjust their strategy over time, it is less clear what the appropriate benchmark comparison would be. This assumption also dramatically simplifies our analysis by ensuring that the algorithm's data is IID. The notion of PAC learnability is usually defined assuming this, and hence a more involved

[30] An active policy debate relates to the use of algorithms in determining whether defendants should be held before trial, without resorting to cash bail. Though the assumption of a long-lived Sender seems less plausible for this application, it seems interesting to potentially adapt the framework here to speak to this application.

notion would be required to relax this assumption. While we suspect that this assumption can be relaxed somewhat, the main difficulty is in finding a meaningful relaxation where tractability and a sensible benchmark comparison remain.

7.3. Final Remarks

This paper introduced the struture of an algorithm game to rigorously answer the question of whether (and how) as-if rational behavior can emerge when full rationality is not assumed. We articulated the following tradeoff in the design of statistical algorithms to mimic rationality: on the one hand, simply fitting a single-threshold classifier to data will fall short of rational play and be exploited. On the other hand, it may not be clear why this is the end of the story. By adding the ability to fit strategies repeatedly and combine them in particular ways, we show how the rational benchmark can be restored. Here, we have taken as a black box the ability to fit single-threshold classifiers. But given this, we exhibit an algorithm that specifies exactly how to put them together in order to construct a strategy which can mimic rationality arbitrarily well. Our main result is that this algorithm emerges as an equilibrium choice in an algorithm game, providing a way out of the horse race.

Part of our hopeful contribution is to illustrate how ideas from machine learning can answer questions of economic significance. Nevertheless, our setting also required some new technical innovations which emerge due to our focus on the economic setting at hand. First, since we study sequential move games with strategic inference, the algorithm only observes a noisy signal of the rational reply. Second, we formulate a notion of implementation error which allows us to relax assumptions on finiteness of the strategy space for the first mover. Both of these emerged due to the motivation of our theoretical applications. These results formed the main technical innovations of the paper, whereas our conceptual innovation are the introduction of algorithm games and the corresponding algorithmic desiderata outlined in Section 3.

We have focused on a simple yet general setting where the comparison to the rational benchmark is most transparent. We have also done so in a class of models which reflects a high degree of economic relevance; we emphasized this in our discussion of various applicaitons in Section 5. And we believe the rising importance of algorithms in the modern economy makes our question relevant and timely. Still, we believe that many issues highlighted by the machine learning literature regarding the design of algorithms can speak to questions of interest to economic theorists. Given how productive the machine learning literature has been in terms of designing algorithms for the purposes of classification, we hope that our work will inspire further analysis of how these algorithms behave in strategic settings.

A. Weak Learnability Proofs

The proof of Proposition 5 uses the following Lemma:

Lemma 4. *Let $\mathcal{H}$ be an arbitrary hypothesis class with the property that for every $h \in \mathcal{H}$ and every permutation $\pi : A \to A$, the composition $\pi \circ h$ is contained in $\mathcal{H}$. Then this hypothesis class can do at least as well as a uniform random guesser.*

Proof. Let Π be the set of all possible permutations on A, noting that$|\Pi| = k!$. Fix an arbitrary $h \in \mathcal{H}$, and define $h^{\pi} = \pi \circ h$. Let $c_{j,y}$ be the cost of assigning action y to p_j. Define

$$\sum_{\pi \in \Pi} c_{j,h^{\pi}(p_j)} = \overline{c}_j.$$

In particular, note that this is invariant to the true optimal action of j. As a result, the random guesser's expected payoff on observation j is is $\overline{c}_j/k!$. To see this, note that $h(p_j)$ gives some fixed guess regarding the label of p_j. Then randomizing over permutations is equivalent to randomizing over actions, as there are an equal number of permutations which flip the recommendation according to $h(p_j)$ and every other label.

We therefore obtain the following matrix equation, for an arbitrary $\rho \in (0, \infty)$, where the number of columns is $k!$ and the number of rows is the number of possible prices.

$$\left(\begin{array}{c|cc|c} \begin{matrix} c_{j,h(p)} \\ -\overline{c}_j/k! \end{matrix} & \cdots & \cdots & \begin{matrix} c_{j,h^{\pi}(p)} \\ -\overline{c}_j/k! \end{matrix} \end{array} \right) \cdot \begin{pmatrix} \frac{\rho}{k!} \\ \vdots \\ \frac{\rho}{k!} \end{pmatrix} = \mathbf{0}$$

Also note that:

$$(1/\rho, \cdots, 1/\rho) \cdot \begin{pmatrix} \frac{\rho}{k!} \\ \vdots \\ \frac{\rho}{k!} \end{pmatrix} = 1$$

So as long as $\rho > 0$, by the theorem of the alternative, we therefore cannot have that a vector $\mathbf{x}$ exists with:

$$\left(\begin{array}{cc} \multicolumn{2}{c}{c_{j,h(p)} - \overline{c}_j/k!} \\ \hline \vdots & \vdots \\ \vdots & \vdots \\ \hline \multicolumn{2}{c}{c_{j,h^{\pi}(p)} - \overline{c}_j/k!} \end{array} \right) \cdot \mathbf{x} \geq \begin{pmatrix} \frac{1}{\rho} \\ \vdots \\ \frac{1}{\rho} \end{pmatrix}.$$

Let $D(p)$ be an arbitrary distribution. Since $\sum_{p \in P} D(p) = 1$, this implies we can find some π such that:

$$\left(\sum_{p_j \in P} D(p_j)(c_{j,h^{\pi}(p_j)} - \frac{\overline{c}_j}{k!}) \right) < \frac{1}{\rho}.$$

Taking $\rho \to \infty$ and rearranging gives:

$$\left(\mathbb{E}_{p\sim D}[c_{j,h^{\pi}(p_j)}]\right) \leq \mathbb{E}_{j\sim D}\left[\frac{\overline{c}_j}{k!}\right]$$

Recalling again that the right hand side of this inequality is the payoff of the random guesser, we have shown that for every possible distribution over prices, we can find some permutation which delivers a cost bounded above by the random guesser. This proves the Lemma. □

Proof of Proposition 5. Let $\mathcal{H}$ be the set of hyperplane classifiers. We prove this by contradiction. If there were no universal lower bound on the error, then we would have, for all ρ, a distribution D_ρ and cost $c^{\rho}_{j,y}$ (without loss normalized to be on the unit sphere themselves) with the property that:

$$\max_{h\in\mathcal{H}} \sum_{p\in P} D_\rho(p) c^{\rho}_{j,h(p)} < U^{\rho}_{c},$$

where U^{ρ}_{c} is the payoff of the uniform random guesser who is correct with added probability ρ. Taking $\rho \to 0$ and passing to a subsequence if necessary, compactness of the unit sphere implies that we can find a distribution D^* and cost function c^* such that:

$$\max_{h\in\mathcal{H}} \sum_{p\in P} D^*(p) c^{*}_{j,h(p)} = U^{0}_{c},$$

where we note by Lemma 4 that at least this bound can be obtained by permutation the labels if necessary. We will arrive at a contradiction by exhibiting a single-hyerplane classifier that achieves a strictly better accuracy, given D^*. Note that $\mathcal{H}$ contains the set of "trivial" classifiers, which give all menus the same label. Also note that the only non-trivial case to consider is when there are at least two prices in the support of D^*; if there were only one price, then simply choosing the prediction corresponding to the label on that price would yield a perfect fit. Since, by assumption, no classifier does better than random guessing, it must be the case in particular that each trivial classifier cannot exceed the random-guess bound. On the other hand, by our previous result, we know there *does* exist a trivial classifier which achieves at least this bound, for *any* D supported on P.

Let $P = \{p_1, \ldots, p_k\}$ be the set of prices supporting D^*, and let $\tilde{p} \in P$ be a price in that is also an extreme point of the convex hull of P. Without loss of generality, assume that $\tilde{p}$ is nontrivial, in the sense that it does not give the same cost to all labels. Note that indeed, this is without loss, since for any such price, the choice of classification is irrelevant.[31] Note that $\tilde{p}$ is not in the convex hull of $P\backslash\{\tilde{p}\}$. Therefore, by the separating hyperplane theorem, we can find an $h \in \mathcal{H}$ which (strictly) separates $\tilde{p}$ from $P\backslash\{\tilde{p}\}$. Denote such a hyperplane by h^*, and note that the set of hyperplane classifiers contains classifiers which assign *any* two labels (possibly the same label) to prices depending on which side of h^* they lie on.

Also note that, again by our previous result, a trivial classifier supported on $P\backslash\{\tilde{p}\}$ can achieve the random guess guaranatee if p is distributed according to the conditional distribution on this set. In other words, our prior lemma implies that there exists $y^* \in A$ such that:

$$\sum_{p_j\in P\backslash\tilde{p}} \frac{D^*(p_j)}{\sum_{q\in P\backslash\tilde{P}} D^*(q)} c^{*}_{j,y^*} = U^{0}_{c^*}.$$

On the other hand, a classifier which separates $p_{\tilde{j}}$ from the other prices can fit $p_{\tilde{j}}$ perfectly. Thus we must have

[31] If all prices are trivial, then we will achieve a contradiction, because that implies that the classifier does do at least as well as the edge-over-random guesser, since all classifiers achieve the same payoff.

$$c_{\tilde{j},y_{\tilde{j}}} < \mathbb{E}_{\hat{y}\sim\text{Unif}}[c_{\tilde{j},\hat{y}}].$$

So consider the hyperplane classifier which predicts $\tilde{y}$ for $\tilde{p}$, and y^* for $p \in P\backslash\{\tilde{p}\}$, i.e., depending on which side of h^* they are on (acknowledging that this may be a trivial classifier). Denote the resulting classifier by h. For this single-hyperplane classifier, we have

$$\sum_{p_j \in P} D^*(p_j)c_{j,h(p_j)} = D^*(p_{\tilde{j}})c_{\tilde{j},y_{\tilde{j}}} + \left(\sum_{q\in P\backslash\{p_{\tilde{j}}\}} D^*(q)\right) \sum_{p_k \in P\backslash\{p_{\tilde{j}}\}} \frac{D^*(p_k)}{\sum_{q\in P\backslash\{p_{\tilde{j}}\}} D^*(q)} c_{k,y^*} > U^0_{c^*},$$

where the inequality holds since the single-threshold classifier does strictly better on some non-trivial price, and as well on all other prices. This completes the proof. □

B. Proof of Proposition 6.

B.1. Convergence of τ_A

B.1.1. The $|A| = 2$ case

We replicate the proof in Schapire and Freund (2012) for reference. Define

$$F_t(p) = \sum_{k=1}^{t} \alpha_k h_k(p).$$

Following the same recursive process described in Schapire and Freund (2012), we have

$$d_{t+1}(p) = \frac{d_1(p)\exp\left(-y(\sigma,p)\sum_{k=1}^{t}\alpha_k h_k(p)\right)}{\prod_{k=1}^{t} Z_k} = \frac{d_1(p)\exp(-y(\sigma,p)F_t(p))}{\prod_{k=1}^{t} Z_k}. \tag{B.8}$$

Following Schapire and Freund (2012), we can show that

$$\mathbb{P}\left(H_t(p) \neq y(\sigma,p)\right) = \mathbf{E}\sum_p d_1(p)\mathbf{1}(H_t(p)\neq y(\sigma,p)) \leq \mathbf{E}\sum_p d_1(p)\exp(-y(\sigma,p)F_t(p)),$$

and

$$\mathbb{P}(H_t(p) \neq y(\sigma,p)) = \mathbf{E}\prod_{k=1}^{t} Z_k.$$

Note

$$Z_k = \sum_p d_k(p)\exp\left(-y(\sigma,p)\alpha_k h_k(p)\right).$$

The rest of the proof follows from Schapire and Freund (2012), which we copy here for later reference.

$$\begin{aligned}
Z_t &= \sum_p d_t(p) \exp\left(-y(\sigma,p)\alpha_t h_t(p)\right) \\
&= \sum_{y(\sigma,p)h_t(p)=1} d_t(p) \exp\left(-\alpha_t\right) + \sum_{y(\sigma,p)h_t(p)=-1} d_t(p) \exp\left(-\alpha_t\right) \\
&= e^{-\alpha_t}(1-\epsilon_t) + e^{\alpha_t}\epsilon_t \\
&= e^{-\alpha_t}\left(\frac{1}{2}+\gamma_t\right) + e^{\alpha_t}\left(\frac{1}{2}-\gamma_t\right) \\
&= \sqrt{1-4\gamma_t^2}
\end{aligned}$$

where

$$\gamma_t = \frac{1}{2} - \epsilon_t.$$

By weak learnability, we know that γ_t is uniformly bounded away from 0: $\exists \gamma > 0$ such that

$$\gamma_t \geq \gamma \qquad \forall t \geq 1.$$

Recall that the maximum number of the elements in the support of σ is N. Thus,

$$d_{t+1}(p) = d_1(p) \prod_{k=1}^{t} \sqrt{1-4\gamma_t^2} \leq \frac{1}{N}\left(1-4\gamma^2\right)^{\frac{t}{2}} \leq \frac{1}{N}e^{-2\gamma^2 t}$$

where the right hand side converges to 0 at the exponential rate uniformly over p.

B.2. The $|A| > 2$ case

The specification of the algorithm can be found in Mukherjee and Schapire (2013). The proof provided below fills in some details to show that convergence holds in a self-contained way.

First, initialize $F_y^0(x_i) = 0$.

- From previous stage, take F_y^t.
- At stage t, find the $h \in \mathcal{H}$ solving:

$$\min_{h\in\mathcal{H}} \frac{1}{m}\sum_{i=1}^{m} \mathbf{1}[h_t(x_i) = y_i]\left((e^{-\eta}-1)\sum_{\tilde{y}\neq y_i} e^{\eta(F_{\tilde{y}}^{t-1} - F_{y_i}^{t-1})}\right) + \mathbf{1}[h_t(x_i) \neq y_i](e^{\eta}-1)e^{\eta(F_{h_t(x_i)}^{t-1} - F_y^{t-1}(x_i))}.$$

- Define $F_y^t(x_i) = \sum_{s=1}^{t} \mathbf{1}[h_t(x_i) = y]$.

The final prediction is $H_t(x_i) = \arg\max_{\tilde{y}} \sum_{t=1}^{T} \mathbf{1}[h_t(x_i) = \tilde{y}]$.

The weak learnability condition says that the hypothesis class can outperform a random guesser that does better than some γ, where we allow for a potentially asymmetric cost of making different errors.

We now show convergence to the rational rule:

Step 1: Bounding The Mistakes: This step is as previous. We have

$$\sum_{i=1}^{m} \mathbf{1}[H_t(x_i) \neq y_i] \leq \sum_{i=1}^{m} \sum_{\tilde{y} \neq y_i} e^{\eta(F^t_{\tilde{y}}(x_i) - F^t_{y_i}(x_i))}.$$

Indeed, the exponential is positive, so this inequality holds when y_i is labelled correctly, and if the label is incorrect, then that means that some $\tilde{y}_i$ satisfies $F^t_{\tilde{y}_i}(x_i) > F^t_{y_i}(x_i)$. Since all exponential terms are positive, and furthermore the exponent is positive if x_i is labelled incorrectly, meaning the right hand side is greater than 1 if mislabeled.

Step 2: Recursive Formulation of the Loss We now show that the right hand side goes to 0 at an exponential rate. We define the loss function to be:

$$L_t(x_i) = \sum_{\tilde{y} \neq y_i} e^{\eta(F^t_{\tilde{y}}(x_i) - F^t_{y_i}(x_i))}, \tilde{L}_t = \frac{1}{m} \sum_{i=1}^{m} L_t(x_i).$$

We first express $\tilde{L}_{t+1}$ as a function of $\tilde{L}_t$. Note that $F^{t+1}_{\tilde{y}}(x_i) = F^t_{\tilde{y}}(x_i)$ for all $\tilde{y} \neq h_t(x_i)$, and $F^{t+1}_{\tilde{y}}(x_i) = F^t_{\tilde{y}}(x_i) + 1$ for $\tilde{y} = h_t(x_i)$. The loss from a given x_i changes depending on whether or not it is correctly classified. For any observation that is classified correctly at the $t+1$th stage, we multiply that observation's loss by a factor of $e^{-\eta}$. On the other hand, for any observation that is classified incorrectly as $\tilde{y}$, we *add* the following:

$$e^{\eta(F^t_{\tilde{y}}(x_i) - F^t_{y_i}(x_i))}(e^{\eta} - 1).$$

So:

$$\tilde{L}_{t+1} = \frac{1}{m} \left(\sum_{i: h_{t+1}(x_i) = y_i} e^{-\eta} L_t(x_i) + \sum_{i: h_{t+1}(x_i) \neq y_i} \left(L_t(x_i) + e^{\eta(F^t_{h_{t+1}(x_i)}(x_i) - F^t_{y_i}(x_i))}(e^{\eta} - 1) \right) \right).$$

Note that if we subtract $\tilde{L}_t$ from both sides, and substitute in for $L_t(x_i)$ above, we obtain:

$$\tilde{L}_{t+1} - \tilde{L}_t = \frac{1}{m} \left(\sum_{i: h_{t+1}(x_i) = y_i} (e^{-\eta} - 1) \sum_{\tilde{y} \neq y_i} e^{\eta(F^t_{\tilde{y}}(x_i) - F^t_{y_i}(x_i))} + \sum_{i: h_{t+1}(x_i) \neq y_i} e^{\eta(F^t_{h_{t+1}(x_i)}(x_i) - F^t_{y_i}(x_i))}(e^{\eta} - 1) \right).$$

Step 3: Weak Learnability By the above, h_{t+1} is chosen to solve:

$$\min_{h \in \mathcal{H}} \frac{1}{m} \sum_{i=1}^{m} \mathbf{1}[h(x_i) = y_i] \left((e^{-\eta} - 1) \sum_{\tilde{y} \neq y_i} e^{\eta(F^t_{\tilde{y}}(x_i) - F^t_{y_i}(x_i))} \right) + \mathbf{1}[h(x_i) \neq y_i](e^{\eta} - 1) e^{\eta(F^t_{h(x_i)}(x_i) - F^t_{y}(x_i))}.$$

In fact, using the previous step, we see that this can equivalently be expressed as $\tilde{L}_{t+1} - \tilde{L}_t$. On the other hand, someone who is random guessing, but is correct with extra probability γ, will be correct with probability $\frac{1-\gamma}{k} + \gamma$, and guess an incorrect label $\tilde{y}$ with probability $\frac{1-\gamma}{k}$. Furthermore, the hypothesis class ensures a weakly lower error (as measured by this cost) than the random guessing. Hence this expression is bounded above by:

$$\frac{1}{m}\sum_{i=1}^{m}\left((\frac{1-\gamma}{k}+\gamma)(e^{-\eta}-1)L_t(x_i) + \frac{1-\gamma}{k}\sum_{\tilde{y}\neq y_i}(e^{\eta}-1)e^{\eta(F^t_{\tilde{y}}(x_i)-F_y(x_i))}\right)$$

Again substituting in for $L_t(x_i)$ and rearranging, we obtain:

$$\left((\frac{1-\gamma}{k}+\gamma)(e^{-\eta}-1) + \frac{1-\gamma}{k}(e^{\eta}-1)\right)\tilde{L}_t.$$

Putting this together, we have this is an upper bound of $\tilde{L}_{t+1}-\tilde{L}_t$, and therefore:

$$\tilde{L}_{t+1} \leq \left(1+\left((\frac{1-\gamma}{k}+\gamma)(e^{-\eta}-1) + \frac{1-\gamma}{k}(e^{\eta}-1)\right)\right)\tilde{L}_t.$$

Step 4: Specifying η We are done if we can ensure $\tilde{L}_t \to 0$ as $t \to \infty$, since Step 1 shows that this implies that the number of misclassifications approaches 0 as well. To complete the argument, we must specify an η which delivers the exponential convergence. However, first note that if $\eta = 0$, the coefficient on $\tilde{L}_t$ in the previous inequality is 1, and the derivative with respect to η is $-\gamma$ at 0, so that this expression is less than 1, for some $\eta > 0$. Setting $\eta = \log(1+\gamma)$, the above coefficient on $\tilde{L}_t$ reduces to:

$$\overbrace{1+\left((\frac{1-\gamma}{k}+\gamma)(\frac{1}{1+\gamma}-1) + \frac{1-\gamma}{k}\gamma\right)}^{z_k(\gamma)}.$$

Note that $z_k(\gamma)$ is bounded above by $\tilde{z}(\gamma) = e^{-\gamma^2/2}$. Indeed, this expression is decreasing in k, with $z_k(0) = 1 = \tilde{z}(0)$, and $z_2(\gamma) = 1 - \frac{\gamma^2}{2} < e^{-\gamma^2/2} = \tilde{z}(\gamma)$. Since $\tilde{L}_0 = (k-1)$, we therefore have that:

$$\tilde{L}_t \leq (k-1)e^{-\gamma t^2/2},$$

as desired.

B.3. Convergence of $\tau_{\hat{A}}$

B.3.1. Preliminaries

Let $\pi(v:p)$ be the posterior distribution of v conditioned on p, and note that v is drawn from a finite set, then $\pi(v:p)$ is a multinomial distribution. Let $\hat{\pi}_t(v:p)$ be the sample average for $\pi(v:p)$. We know that the rate function of $\hat{\pi}_t(v:p)$ is the relative entropy of $\hat{\pi}_t$ with respect to π (Dembo and Zeitouni (1998))

$$I_\pi = \sum_v \hat{\pi}_t(v:p)\log\frac{\hat{\pi}_t(v:p)}{\pi(v:p)},$$

from which we derive λ in (6.7): $\forall\epsilon > 0$, let $N_\epsilon(\pi)$ be the ϵ neighborhood of $\pi(v:p)$, and

$$\lambda = \inf_{\hat{\pi}_t \notin N_\epsilon(\pi)} I_\pi.$$

Note that

$$y^R(\sigma, p) \neq \hat{y}_t(p)$$

only if π and $\hat{\pi}_t$ prescribe different actions. Since $\hat{\pi}_t$ is a consistent estimator of π, the probability of two probability distributions prescribing two different actions vanishes. The large deviation property of $\hat{\pi}_t$ implies that $\hat{y}_t(p)$ satisfies (6.7), if $y^R(\sigma, p)$ is a strict best response.

By the concavity of the logarithmic function, I_π is minimized if π is a uniform distribution and

$$\inf_\pi I_\pi > 0.$$

If$|P| < \infty$ and$|A| < \infty$, we obtain the uniform version of (6.7) with respect to the true probability distribution. We state the result without proof for later reference.

Lemma 5. *Suppose that $\hat{y}_t(p)$ is a consistent*[32] *estimator of $y^R(\sigma, p)$ and satisfies* (6.7)*. Then, $\exists\lambda > 0$ such that*

$$\limsup_{t\to\infty} -\frac{1}{t}\log \mathbb{P}\left(y^R(\sigma, p) \neq \hat{y}_t(p)\ \ \forall p \textit{ in the support of } \sigma\right) \leq \lambda. \tag{B.9}$$

We construct algorithm $\tau_{\hat{A}}$ by replacing $y(\sigma, p)$ by $\hat{y}_t(p)$ in τ_A constructed in the previous section. More precisely, let $f_t^y(p)$ be the empirical probability that $\hat{y}_t(p) = 1$ at the beginning of period t. Thus, $\hat{y}_t(p) = -1$ with probability $1 - f_t^y(p)$. Given $\{d_t(p), \hat{y}_t(p)\}_p$, h_t solves

$$\max_{h\in\mathcal{H}} \sum_p h(p)d_t(p)[1 \cdot f_t^y(p) - 1 \cdot (1 - f_t^y(p))]$$

and

$$\hat{\epsilon}_t = \sum_p d_t(p)\left[f_t^y(p)\mathbf{1}(h(p) = 1) + (1 - f_t^y(p))\mathbf{1}(h(p) = -1)\right].$$

Using weak learnability, we can show that $\exists\rho > 0$ such that

$$\hat{\epsilon}_t \leq \frac{1}{2} - \rho.$$

Since $\hat{y}_t(p)$ has the full support over $\{-1, 1\}$ $\forall t \geq 1$,

$$\hat{\epsilon}_t > 0.$$

Given an algorithm τ_A with observed labels, we can therefore replace it with $\tau_{\hat{A}}$ which involves inferring the labels $y^R(\sigma, \cdot)$, setting them equal to $\hat{y}_t(\cdot)$, for all $t \geq 1$.

B.3.2. Main Proof

Under the assumption that $y^R(\sigma, p)$ is a strict best response,

$$\lim_{t\to\infty} \hat{y}_t(p) = y^R(\sigma, p)$$

almost surely. Since $\hat{y}_t(p)$ satisfies the uniform LDP, $\forall\epsilon > 0$, $\exists\rho(\epsilon, \sigma) > 0$ and $T(\epsilon, \sigma)$ such that

$$\mathbb{P}\left(\exists t \geq T(\epsilon, \sigma), \hat{y}_t(p) \neq y^R(\sigma, p)\right) \leq e^{-t\rho(\epsilon,\sigma)}.$$

[32] An estimator $\hat{y}_t(p)$ is consistent if $\hat{y}_t(p)$ converges to $y^R(\sigma, p)$ in probability as $t \to \infty$.

Since the support of σ contains a finite number of p, the empirical the multinomial probability distribution over θ.

Let $\hat{\pi}_t(\theta : p)$ be the empirical probability distribution over Θ following t rounds of observations. By the law of large numbers, $\hat{\pi}_t(\theta : p) \to \pi(\theta : p)$ computed via Bayes rule from the prior distribution over θ and σ. Write $\Theta = (\theta_1, \ldots, \theta_{|\Theta|})$. Given $\epsilon = (\epsilon, \ldots, \epsilon) \in \mathbb{R}^{|\Theta|}$, the rate function of the multinomial distribution is

$$\sum_{i=1}^{|\Theta|} \epsilon \log \frac{\epsilon}{p(\theta)}$$

where $p(\theta)$ is the probability that θ is realized. Since $\sum_\theta p(\theta) = 1$,

$$\sum_{i=1}^{|\Theta|} \epsilon \log \frac{\epsilon}{p(\theta)} \geq \prod_{i=1}^{|\Theta|} \epsilon \log \frac{\epsilon}{1/|\Theta|} = \prod_{i=1}^{|\Theta|} \epsilon \log \epsilon |\Theta| > 0.$$

Note that the right hand side is independent of σ, which is the rate function of the uniform distribution over Θ. Thus, we can choose $\rho(\epsilon) \leq \rho(\epsilon, \sigma)$ uniformly over σ, which is strictly increasing with respect to $\epsilon > 0$. We choose $T(\epsilon)$ independently of σ as well.

Define an event

$$\mathcal{L} = \left\{ \hat{y}_t(p) = y^R(\sigma, p) \qquad \forall t \geq T(\epsilon) \right\}$$

We know that

$$\mathbb{P}(\mathcal{L}) \geq 1 - e^{-t\rho(\epsilon)}.$$

Fix $t > T(\epsilon)$. We have

$$\begin{aligned} & \mathbb{P}\left(\tau_{\hat{A}}(D_t)(p) \neq y^R(\sigma, p)\right) \\ = \; & \mathbb{P}\left(\tau_{\hat{A}}(D_t)(p) \neq y^R(\sigma, p) : \mathcal{L}\right) \mathbb{P}(\mathcal{L}) + \mathbb{P}\left(\tau_{\hat{A}}(D_t)(p) \neq y^R(\sigma, p) : \mathcal{L}^c\right) \mathbb{P}(\mathcal{L}^c) \\ \leq \; & \mathbb{P}\left(\tau_{\hat{A}}(D_t)(p) \neq y^R(\sigma, p) : \mathcal{L}\right) + \mathbb{P}(\mathcal{L}^c) \\ \leq \; & \mathbb{P}\left(\tau_{\hat{A}}(D_t)(p) \neq y^R(\sigma, p) : \mathcal{L}\right) + e^{-t\rho(\epsilon)}. \end{aligned}$$

Following the same logic as in the proof of Proposition 6, we can show that $\exists \gamma(G) > 0$ such that

$$\hat{Z}_t \leq 1 - \gamma(G) \qquad \forall t \geq 1 \tag{B.10}$$

under $\tau_{\hat{A}}$.

Recall that

$$F_a(p) = \sum_{s=1}^{t} \alpha_s \mathbf{1}(h_s(p) = a).$$

Similarly, we define

$$\hat{F}_a(p) = \sum_{s=1}^{t} \hat{\alpha}_s \mathbf{1}(h_s(p) = a).$$

Following the same logic as in the proof of Proposition 6, we know that if $\tau_{\hat{A}}(D_t)(p) \neq y^R(p)$,

$$\hat{F}_{y^R(\sigma,p)}(p) + \sum_{a \neq y^R(\sigma,p)} \hat{F}_a(p) > 0.$$

Thus,

$$\begin{aligned} \mathbf{1}(\tau_{\hat{A}}(D_t)(p) \neq y^R(\sigma,p)) &\leq \mathbf{1}\left(\hat{F}_{y^R(\sigma,p)}(p) + \sum_{a \neq y^R(\sigma,p)} \hat{F}_a(p)\right) \\ &\leq \exp\left(\hat{F}_{y^R(\sigma,p)}(p) + \sum_{a \neq y^R(\sigma,p)} \hat{F}_a(p)\right). \end{aligned}$$

Conditioned on event $\mathcal{L}$,

$$\hat{y}_t(p) = y^R(\sigma,p) \qquad \forall t \geq T(\epsilon).$$

We can write for $t \geq T(\epsilon)$,

$$\begin{aligned} d_{t+1}(p) &= \frac{\hat{d}_t(p)\exp(\alpha_t(\mathbf{1}(h_t(p) \neq \hat{y}_t(p)) - \mathbf{1}(h_t(p) = \hat{y}_t(p))))}{\hat{Z}_t} \\ &= \frac{\hat{d}_t(p)\exp(\alpha_t(\mathbf{1}(h_t(p) \neq y^R(\sigma,p)) - \mathbf{1}(h_t(p) = y^R(\sigma,p))))}{\hat{Z}_t} \\ &= \frac{d_{T(\epsilon)}(p)\exp(\sum_{s=T(\epsilon)}^t \alpha_s(\mathbf{1}(h_s(p) \neq y^R(\sigma,p)) - \mathbf{1}(h_s(p) = y^R(\sigma,p))))}{\prod_{s=T(\epsilon)}^t \hat{Z}_t}. \end{aligned}$$

Thus,

$$\begin{aligned} &\prod_{s=T(\epsilon)}^t \hat{Z}_t \\ =\ & \sum_p d_{T(\epsilon)}(p)\exp\left[\sum_{s=T(\epsilon)}^t \alpha_s(\mathbf{1}(h_s(p) \neq y^R(\sigma,p)) - \mathbf{1}(h_s(p) = y^R(\sigma,p)))\right] \\ \geq\ & \left(\min_{p \in \mathcal{P}(\sigma)} d_{T(\epsilon)}(p)\right)\sum_p \exp\left[\sum_{s=T(\epsilon)}^t \alpha_s(\mathbf{1}(h_s(p) \neq y^R(\sigma,p)) - \mathbf{1}(h_s(p) = y^R(\sigma,p)))\right]. \end{aligned}$$

Since $d_1(p)$ is the uniform distribution over $\mathcal{P}(\sigma)$,

$$\min_{p \in \mathcal{P}(\sigma)} d_{T(\epsilon)}(p) > 0.$$

We can write

$$
\begin{aligned}
& \prod_{s=1}^{t} \hat{Z}_t = \prod_{s=T(\epsilon)}^{t} \hat{Z}_t \prod_{s=1}^{T(\epsilon)-1} \hat{Z}_t \\
\geq \quad & \left(\min_{p\in\mathcal{P}(\sigma)} d_{T(\epsilon)}(p)\right) \sum_p \exp(\sum_{s=T(\epsilon)}^{t} \hat{\alpha}_s(\mathbf{1}(h_s(p) \neq y^R(\sigma,p)) - \mathbf{1}(h_s(p) = y^R(\sigma,p)))) \prod_{s=1}^{T(\epsilon)-1} \hat{Z}_t \\
= \quad & \frac{\left(\min_{p\in\mathcal{P}(\sigma)} d_{T(\epsilon)}(p)\right) \prod_{s=1}^{T(\epsilon)-1} \hat{Z}_t}{\sum_p \exp\left[\sum_{s=1}^{T(\epsilon)-1} \hat{\alpha}_s(\mathbf{1}(h_s(p) \neq y^R(\sigma,p)) - \mathbf{1}(h_s(p) = y^R(\sigma,p)))\right]} \\
& \qquad \times \sum_p \exp\left[\sum_{s=1}^{t} \hat{\alpha}_s(\mathbf{1}(h_s(p) \neq y^R(\sigma,p)) - \mathbf{1}(h_s(p) = y^R(\sigma,p)))\right]
\end{aligned}
$$

over $\mathcal{L}$. Define

$$
M(\epsilon) = \frac{\left(\min_{p\in\mathcal{P}(\sigma)} d_{T(\epsilon)}(p)\right) \prod_{s=1}^{T(\epsilon)-1} \hat{Z}_t}{\sum_p \exp(\sum_{s=1}^{T(\epsilon)-1} \hat{\alpha}_s(\mathbf{1}(h_s(p) \neq y^R(\sigma,p)) - \mathbf{1}(h_s(p) = y^R(\sigma,p))))}
$$

which is bounded away from 0.

Recall that

$$
\begin{aligned}
& \mathbb{P}(\tau_{\hat{A}}(D_t)(p) \neq y^R(\sigma,p)) \\
\leq \quad & \sum_p d_1(p) \exp(\sum_{s=1}^{t} \hat{\alpha}_s(\mathbf{1}(h_s(p) \neq y^R(\sigma,p)) - \mathbf{1}(h_s(p) = y^R(\sigma,p)))) \\
\leq \quad & \frac{\prod_{s=1}^{t} \hat{Z}_t}{M(\epsilon)} \leq \frac{(1-\gamma(G))^t}{M(\epsilon)} \leq \frac{e^{-t\gamma(G)}}{M(\epsilon)}.
\end{aligned}
$$

Combining the probabilities over $\mathcal{L}$ and $\mathcal{L}^c$, we have that $\forall \epsilon$, $\forall \sigma \in \Sigma^G \subset \Sigma$, $\exists T(\epsilon)$, $\rho(\epsilon)$ and $\gamma(G)$ such that

$$
\mathbb{P}\left(\exists t \geq T(\epsilon),\ \tau_{\hat{A}}(D_t)(p) \neq y^R(\sigma,p)\right) \leq \frac{e^{-t\gamma(G)}}{M(\epsilon)} + e^{-t\rho(\epsilon)}.
$$

We can choose $T > T(\epsilon)$ and $\overline{\rho}$ such that $\forall t \geq T$,

$$
\frac{e^{-t\gamma(G)}}{M(\epsilon)} + e^{-t\rho(\epsilon)} \leq e^{-\overline{\rho} t}
$$

which proves the proposition.

C. Proof of Theorem 2

As stated in the main text, the proof follows three main steps, showing that the modification of the first mover's strategy leads to one where $\mathbb{E}[v(\theta,p,a) \mid \tilde{\sigma}, p]$ is uniformly equicontinuous in p, and maintains the approximate optimality of the reply. After showing each part of this argument, we describe how the Theorem follows under an appropriate discretization.

C.1. Step One

We first show that $\mathbb{E}[v_\theta \mid \tilde{\sigma}_\eta, p]$ is Lipschitz in p uniformly of $\tilde{\sigma}_\eta$, noting that we are restricting to prices where $\tilde{\sigma}_\eta(p) > \gamma$.

$$\tilde{\sigma}'_\eta(p \mid \theta) = \int \phi'_\eta(p - \tilde{p})\sigma(\tilde{p} \mid \theta)d\tilde{p} \leq \max \phi'_\eta := \overline{\phi'}.$$

Furthermore, we have:

$$\frac{d}{dp}\mathbb{P}_{\tilde{\sigma}_\eta}[\theta \mid p] = \frac{\tilde{\sigma}'_\eta(p \mid \theta)\mathbb{P}[\theta]}{\sum_{\tilde{\theta}} \tilde{\sigma}_\eta(p \mid \tilde{\theta})\mathbb{P}[\tilde{\theta}]} - \frac{\tilde{\sigma}_\eta(p \mid \theta)\mathbb{P}[\theta](\sum_{\tilde{\theta}} \sigma'_\eta(p \mid \tilde{\theta})\mathbb{P}[\tilde{\theta}])}{(\sum_{\tilde{\theta}} \tilde{\sigma}_\eta(p \mid \tilde{\theta})\mathbb{P}[\tilde{\theta}])^2},$$

so:

$$\left|\frac{d}{dp}\mathbb{P}_{\tilde{\sigma}_\eta}[\theta \mid p]\right| \leq \overline{\phi'}\mathbb{P}[\theta] \cdot \left(\frac{1}{\sum_{\tilde{\theta}} \tilde{\sigma}_\eta(p \mid \tilde{\theta})\mathbb{P}[\tilde{\theta}]}\right) + \overline{\phi'}\left(\frac{\tilde{\sigma}_\eta(p \mid \theta)\mathbb{P}[\theta]}{(\sum_{\tilde{\theta}} \tilde{\sigma}_\eta(p \mid \tilde{\theta})\mathbb{P}[\tilde{\theta}])^2}\right) \leq \overline{\phi'}\mathbb{P}[\theta]\left(\frac{1}{\gamma} + \frac{M(\eta)}{\gamma^2}\right),$$

where $M(\eta)$ is a bound on $\tilde{\sigma}_\eta(p \mid \theta)\mathbb{P}[\theta]$, which exists since σ and ϕ_η have bounded densities. Hence we see that the conditional probability has a derivative that is uniformly bounded in p, and is hence Lipschitz continuous in p. Importantly, the bound only depends on η and γ (and $\mathbb{P}[\theta]$), and is therefore uniform over all strategies in the image of the augmentation. Hence we can ensure that Lipschitz continuity is mainted for all prices in the support of $\tilde{\sigma}_\eta$.

In fact, recall that the Lipschitz constant is equal to the L^∞ norm of the derivative. Hence Lipschitz continuity depends only on γ, $M(\eta)$ and $\overline{\phi'_\eta}$, meaning that the Lipschitz constant holds uniformly over the image of the distributions emerging under the algorithm. It follows that the image of the seller's strategy under the transformation is uniformly equicontinuous.

C.2. Step Two

Note that since $\mathbb{E}[v_\theta \mid \sigma, p]$ is continuous on $S = \cup_\theta \text{ Supp } \sigma(\cdot \mid \theta)$, $\mathbb{E}[v_\theta \mid \sigma, p]$ is uniformly continuous on any compact $K \subset S$. Define:

$$K_\gamma = \{p : \sum_\theta \sigma(p \mid \theta)\mathbb{P}[\theta] \geq \gamma\}.$$

Using that mollifiers converge uniformly on compact sets, we have that $\tilde{\sigma}_\eta \to \sigma$ uniformly on K_γ. We therefore have that, for any $\tilde{\varepsilon}$, we can find some $\overline{\eta}$ such that if $\eta < \overline{\eta}$ and $p \in K_\gamma$, then $\left|\tilde{\sigma}_\eta(p \mid \theta) - \sigma(p \mid \theta)\right| < \tilde{\varepsilon}$ for all θ, and $\left|\sum_\theta \tilde{\sigma}_\eta(p \mid \theta)\mathbb{P}[\theta] - \sum_\theta \sigma(p \mid \theta)\mathbb{P}[\theta]\right| < \tilde{\varepsilon}$.

Furthermore, since σ is uniformly continuous on K_γ, we have:

$$\left|\sigma(p \mid \theta) - \tilde{\sigma}(p' \mid \theta)\right| = \left|\int \phi_\eta(p' - \tilde{p})(\sigma(p \mid \theta) - \sigma(\tilde{p} \mid \theta))d\tilde{p}\right| \leq \tilde{\varepsilon},$$

using the uniform continuity of σ on K_γ.

So for any $p \in K_\gamma$, and η sufficiently small, we have (letting $\overline{v} = \max_\theta v_\theta$):

$$\begin{aligned}
\left|\mathbb{E}[v_\theta \mid \sigma, p] - \mathbb{E}[v_\theta \mid \tilde{\sigma}_\eta, p']\right| &= \left|\frac{\sum_\theta v_\theta \sigma(p \mid \theta)\mathbb{P}[\theta] \sum_{\tilde{\theta}} \tilde{\sigma}_\eta(p' \mid \tilde{\theta})\mathbb{P}[\tilde{\theta}] - \sum_\theta v_\theta \tilde{\sigma}_\eta(p' \mid \theta)\mathbb{P}[\theta] \sum_{\tilde{\theta}} \sigma(p \mid \tilde{\theta})\mathbb{P}[\tilde{\theta}]}{\left(\sum_\theta \sigma(p \mid \theta)\mathbb{P}[\theta]\right)\left(\sum_\theta \tilde{\sigma}_\eta(p' \mid \theta)\mathbb{P}[\theta]\right)}\right| \\
&\leq \frac{1}{\sigma(p) \cdot (\gamma - \tilde{\varepsilon})} \left| \sum_\theta v_\theta \sigma(p \mid \theta)\mathbb{P}[\theta] \sum_{\tilde{\theta}} (\tilde{\sigma}_\eta(p' \mid \tilde{\theta}) - \sigma(p \mid \tilde{\theta}))\mathbb{P}[\tilde{\theta}] \right. \\
&\left. + \sum_\theta v_\theta(\sigma(p \mid \theta) - \tilde{\sigma}_\eta(p' \mid \theta))\mathbb{P}[\theta] \sum_{\tilde{\theta}} \sigma(p \mid \tilde{\theta})\mathbb{P}[\tilde{\theta}] \right| \\
&\leq \frac{1}{\sigma(p) \cdot (\gamma - \tilde{\varepsilon})} \left(\overbrace{\left| \sum_\theta v_\theta(\sigma(p \mid \theta) - \tilde{\sigma}_\eta(p' \mid \theta)) \sum_{\tilde{\theta}} \sigma(p \mid \tilde{\theta})\mathbb{P}[\tilde{\theta}] \right|}^{\leq \overline{v}\tilde{\varepsilon}\sigma(p)} \right. \\
&\left. + \overbrace{\left| \sum_\theta v_\theta \sigma(p \mid \theta)\mathbb{P}[\theta] \sum_{\tilde{\theta}} (\tilde{\sigma}_\eta(p' \mid \tilde{\theta}) - \sigma(p \mid \tilde{\theta}))\mathbb{P}[\tilde{\theta}] \right|}^{\leq \overline{v} \cdot \tilde{\varepsilon} \cdot \sigma(p)} \right) \\
&\leq \frac{2\overline{v}\tilde{\varepsilon}}{\gamma - \tilde{\varepsilon}}.
\end{aligned}$$

The first inequality follows from adding and subtracting $\sum_\theta v_\theta \sigma(p \mid \theta)\mathbb{P}[\theta] \sum_{\tilde{\theta}} \sigma(p \mid \tilde{\theta})\mathbb{P}[\tilde{\theta}]$ to the numerator inside the absolute value (as well as the lower bound on $\tilde{\sigma}_\eta(p)$), and the second inequality is from the triangle inequality, and the overbraced expression follows from $v_\theta \leq \overline{v}$ and uniform convergence of $\tilde{\sigma}_\eta$ to σ.

So for any fixed γ, we can find some some η such that whenever $\eta < \overline{\eta}$, we can ensure that

$$\left|\mathbb{E}[v_\theta \mid \tilde{\sigma}_\eta, p] - \mathbb{E}[v_\theta \mid \sigma, p]\right| < \varepsilon^*, \text{for all} p \in K_\gamma,$$

by choosing $\tilde{\varepsilon}$ sufficiently small so that $\frac{2\tilde{\varepsilon}}{\gamma(\gamma - \tilde{\varepsilon})} < \varepsilon^*$. It follows that if the Receiver's classifier converges to a rule that is ε-optimal under $\tilde{\sigma}_\eta$, it converges to a rule that is $\varepsilon + \varepsilon^*$ optimal under σ. The probability that this fails to occur is simply the probability that the price is outside of K_γ, which can be made arbitrarily small by taking $\gamma \to 0$, since we can approximate the support of σ arbitrarily well.

C.3. Step Three

Note that, for an arbitrary continuous distribution f, if $p \sim f$ we have (for any compact K):

$$\mathbb{P}_f[L_\gamma] = \int_K \mathbf{1}[p : f(p) \leq \gamma] f(p) dp \leq \int_K \mathbf{1}[p : f(p) \leq \gamma] \gamma dp \leq \mu(K) \cdot \gamma,$$

where μ is Lebesgue measure. Taking L_γ to be the complement of K_γ, it follows that the probability that $p \in L_\gamma$ is small if γ is small, and furthermore that this probability can be made small uniformly, using only γ.

C.4. Putting it together

Fixing γ and $\tilde{\sigma}_\eta$ generated as the distribution of $p + z$, and given some ε, we can partition $\mathcal{P}$ into a collection of sets $P_1, \ldots, P_n$ such that $\max_{p,p' \in P_i} \left|p - p'\right| < \tilde{\delta}$, where $\tilde{\delta}$ ensures that $\left|\mathbb{E}[v(\theta, p, a) \mid \tilde{\sigma}, p] - \mathbb{E}[v(\theta, p, a) \mid \tilde{\sigma}, p]\right| < \varepsilon/2$ whenever $p \notin K_\gamma$ (noting that this will hold for any fixed γ, η whenever δ is sufficiently small). Theorem 1 then shows that the algorithm with this modification converges, at an exponential rate, to a rule which chooses:

$$\arg\max_{a} \int_{\tilde{p}\in P_i} v(a,\theta,\tilde{p})\pi(\theta:\tilde{p})d\tilde{\sigma}(\tilde{p}).$$

There are two cases to consider. First, suppose there is no $p \in P_i$ such that $v(a,\theta,p)$ is an ε best reply; in that case, the choice of second-mover action will not influence the condition of the theorem. On the other hand, if there is some p^* such that $\tilde{a}$ delivers ε higher than all other actions, then at any $p \in P_i$, $\tilde{a}$ delivers $\varepsilon/2$ higher payoff than all other actions. In particular, the action assigned to P_i is the optimal reply at p^*, and hence again we have the $\varepsilon-$optimal reply is achieved at this p^* (in the limit, as per the Proof of Theorem 1. Since this holds for every γ, and since we can take $\gamma \to 0$, we conclude that we can maintain the PAC-requirement. Specifically, note that:

$$\begin{aligned}\mathbb{P}_{p\sim\sigma}[\tau(D_t)(p) \neq y(p) \mid D_t] &\leq \mathbb{P}_{p\sim\sigma}[\tau(D_t)(p) \neq y(p) \mid D_t, p \in K_\gamma]\mathbb{P}_{p\sim\sigma}[p \in K_\gamma] + \mathbb{P}_{p\sim\sigma}[p \notin K_\gamma] \\ &\leq \overbrace{\mathbb{P}_{p\sim\sigma}[\tau(D_t)(p) \neq y(p) \mid D_t, p \in K_\gamma]}^{(1)} + \overbrace{\mu(\mathcal{P})\gamma}^{(2)}\end{aligned}$$

So, the PAC-requirement will be satisfied if we replacing the ϵ in Definition 2 with $\epsilon/2$ (noting that if the requirement holds for some ϵ it must also hold for $\epsilon/2$), so that (1) is less than $\epsilon/2$, and and by choosing $\gamma < \epsilon/(2\mu(\mathcal{P}))$, so that (2) is less than $\epsilon/2$, and hence $\mathbb{P}_{p\sim\sigma}[\tau(D_t)(p) \neq y(p) \mid D_t] < \epsilon$. In other words, fixing ϵ and δ, we choose $\gamma < \epsilon/(2\mu(\mathcal{P}))$, and a partition of $\mathcal{P}$ such that $|p - p'| < \tilde{\delta}$ for any p, p' in the same partition (where $\tilde{\delta}$ comes from the uniform equicontinuity condition, together with the choice of γ). Then, as long as $t \geq T(2/\epsilon, 1/\delta)$, we have the PAC-condition is satisfied, following the Proof of Theorem 1, finishing the proof.

D. Proofs for Examples

Proof of Proposition 1. The ideas in this proof are largely borrowed from Rubinstein (1993), accommodating two additional features of our enviroment: (a) need to infer the strategy from observed data and (b) the generalized setting, but we provide the proof for completeness (while also highlighting which general properties of the utility functions drive the result). We construct a strategy σ^* for the Sender that generates higher payoff than the equilibrium strategy σ^R, thus deriving the contradiction that σ^R is a best response to τ in the long run. More precisely, define $(p^*_\theta, a(\theta))$ to be the Sender payoff-maximizing strategy. We show that the Sender can induce the Receiver to choose $a(\theta) \neq y_R(p^*_\theta)$.

Fix $\epsilon > 0$ small, and suppose $\Theta = \{\theta_L, \theta_H\}$. First suppose $v(\theta_L, p^*_L, a_1) = v(\theta_L, p^*_L, a_0)$. Let $\tilde{p} \in (p_L, p_H)$ satisfies $v(\theta_L, \tilde{p}, 1) < v(\theta_L, \tilde{p}, 0)$. (If p is multidimensional, we can take $\tilde{p}$ to be on the line segment connecting p^*_L and p^*_H) Set $\eta = v(\theta_L, \tilde{p}, 0) - v(\theta_L, \tilde{p}, 1) > 0$. We then choose $\epsilon, \epsilon_H, \epsilon_L > 0$ to satisfy

$$\pi(H)\epsilon_H < \pi(L)\epsilon_L, \tag{D.11}$$

and such that

$$\frac{\epsilon_L}{\epsilon_L + \eta} < \epsilon < \frac{\pi(L)\epsilon_L - \pi(H)\epsilon_H}{\pi(L)\epsilon_L}. \tag{D.12}$$

Under the increasing differences assumption, we can find $p_i(\epsilon_i)$ such that

$$\epsilon_i = v(\theta_i, p_i(\epsilon_i), 1) - v(\theta_i, p_i(\epsilon_i), 0).$$

Consider the following randomized pricing rule σ^* of the Sender: in state H, $\tilde{p}_H(\epsilon_H)$ is chosen with probability 1. In state L, $p_L(\epsilon_L)$ is chosen with probability $1-\epsilon$ and $\tilde{p}$ with probability ϵ.

Under this strategy, the optimal response following $\tilde{p}$ is 0, and this does not vanish as all other parameters tend to 0. However, the ex-post optimal decisions are 1 for both $\tilde{p}_L(\epsilon_L)$ and $\tilde{p}_H(\epsilon_H)$. Nevertheless, (D.12) implies first, the decisionmaker prefers to choose $a=1$ if and only if $\tilde{p}_L(\epsilon_L)$ than choose $a=1$ if and only if $\tilde{p}_H(\epsilon_H)$; and second, that the loss from choosing $a=1$ following $\tilde{p}$ is larger than the loss from choosing $a=0$ at $\tilde{p}_L(\epsilon_L)$. Putting this together, and taking $\epsilon, \epsilon_L, \epsilon_H \to 0$ shows this policy approximates the Sender's optimum, as desired.

The case of $v(\theta_L, p_L^*, a_1) > v(\theta_L, p_L^*, a_0)$ is even more straightforward, since in this case the gain from choosing a_1 is non-vanishing, meaning that we can set $\varepsilon_L = 0$.

The verification that the optimal rule converges to this threshold when emerging from data is straightforward; any recursive learning algorithm generates $\{\phi_t\}$ which converges to $\phi \in \left(v_L - \epsilon_L, \frac{v_H+v_L}{2}\right)$ to achieve the best response of type 1 buyer against σ. Thus, the long run average payoff against such algorithm should be bounded from below by $\mathcal{U}_p^* - \epsilon$. □

Proof of Proosition 2. First, we describe an environment where, when the buyer has access to two-threshold rules, the seller can devise a strategy which delivers higher payoff than the rational benchmark, even though the rational benchmark is a two theshold rule. In other words, we show how the argument in Proposition 1 can deliver the same conclusion when the buyer can use double threshold rules, albeit under a modified environment. Specifically, let $\Theta = \{\theta_0, \theta_L, \theta_H\}$; payoffs are exactly as above when $\theta \in \{\theta_L, \theta_H\}$. When $\theta = \theta_0$, however, we have $v(\theta_0, p, 0) >> v(\theta_0, p, 1)$ with $\arg\max_p v(\theta_0, p, 0) = 0$, while $u(\theta_0, p, 0) > u(\theta_0, p, 1)$ for all possible prices (e.g., high production costs and low quality). We also take $\pi(0)$ (the prior probability of state θ_0) small but fixed. In this case, the rational benchmark is the same as in the case of Proposition 1, with the addition of the seller charging a price of 0 and the buyer not purchasing in state θ_0.

For this environment, the argument from Proposition 1 carries over unchanged, except where the seller always charges 0 in state θ_0. Specifically, using the same pricing strategy in states θ_L and θ_H as outlined above requires the buyer having access to a three-threshold strategy; as $\epsilon, \epsilon_L, \epsilon_H \to 0$, the buyer approaches indifference across each action, whereas the benefit of choosing 0 in state θ_0 does not vanish. Therefore, now the argument implies that this strategy induces the buyer uses a two threshold rule, whereby they choose to not purchase at prices $p \in \{0, p_H(\epsilon_H)\}$ and to purchase at price $p = p_L(\epsilon_L)$.

To complete the proof of the proposition, we now uses these two environments—i.e., the one described in the previous paragraphs as well as the one described in Proposition 1—to show that if the buyer has access to n thresholds, the seller can profitably deviate by using a strategy such that $y^R(\sigma, p)$ requires $n+1$ thresholds; the environment under consideration will be as described in the first part of this proof if n is even, and as described in the proof of Proposition 1 if n is odd. Below we describe the proof for n odd, noting it is an identical argument when n is even (using the slighlty different environment).

Choose ϵ, ϵ_L and ϵ_H as in the above proposition. Let $\{\tilde{p}_k\}_{k=1,\ldots,n}$ be such that $p_L < \tilde{p}_1 < \cdots < p_H$ satisfy $v(\theta_L, \tilde{p}_k, 1) < v(\theta_L, \tilde{p}_k, 1)$ for k odd and $v(\theta_H, \tilde{p}_k, 1) > v(\theta_H, \tilde{p}_k, 1)$ for k even. Let η_k be the absolute value of the differences of these inequalities; choose ϵ_L, ϵ_H and ϵ, ϵ' such that (D.11) is satisfied, as well as:

$$\frac{\epsilon_L}{\epsilon_L + 2\eta_k/(n+1)} < \epsilon < \frac{\pi(L)\epsilon_L - \pi(H)\epsilon_H(1-\epsilon')}{\pi(L)\epsilon_L}, \tag{D.13}$$

and (assuming $n > 1$)

$$\frac{\epsilon_H}{\epsilon_H + 2\eta_k/(n-1)} < \epsilon' \tag{D.14}$$

That this system of inequalities can be satisfied follows from noting that when $\epsilon_H = 0$, the conditions reduce to $\frac{\epsilon_L}{\epsilon_L + 2\eta_k/(n+1)} < \epsilon$, and in particular ϵ' can be taken to be arbitrary; since the right hand side of (D.12) is continuous in ϵ_H, we can find a value for it such that the inequality is maintained.

Again choosing $p_i(\epsilon_i)$ in the same way as in the proof of Proposition 1, we consider the following pricing rule:

- In state H, $\tilde{p}_H(\epsilon_H)$ is chosen with probability $1 - \epsilon'$; with complementary probability, the price is $\tilde{p}_k$ for k even, each with equal probability.

- In state L, $p_L(\epsilon_L)$ is chosen with probability $1 - \epsilon$; with complementary probability, the price is $\tilde{p}_k$ for k odd, each with equal probability.

As before, the payoff gain from choosing after $\tilde{p}_i$ does not vanish as other parameters tend to 0, for all i. On the other hand, the inequlaities (D.11), (D.13) and (D.14) imply: First, the buyer does worse by erring on $\tilde{p}_L(\epsilon_L)$ than $\tilde{p}_H(\epsilon_H)$; and second, the loss from erring on any $\tilde{p}_k$ is larger than the loss from erring at either $\tilde{p}_L(\epsilon_L)$ or $\tilde{p}_H(\epsilon_H)$. Putting these observations together, we have the optimal decision rule is always an n-threshold rule that errs at $p_H(\epsilon_H)$, where the buyer chooses action 0. Again taking $\epsilon, \epsilon_L, \epsilon_H, \epsilon' \to 0$ approximates the seller's optimum, as desired. □

Proposition 3 follows immediately from the following result, which we call Proposition 3'.

Proposition 3' *Suppose $u(\theta, p, 1) - u(\theta, p, 0)$ is constant in θ and weakly concave in p. Suppose further that $u(\theta, p^*, 1) > u(\theta, p^*, 0)$. Then there exists a single-threshold classifier which the algorithm could commit to using which ensures the strategic player chooses p^* with probability 1.*

Proof of Proposition 3'. Concave differences implies that the set

$$K = \{p : u(\theta, p, 1) \geq u(\theta, p, 0)\}$$

is a convex set; if $u(\theta, p_i, 1) - u(\theta, p_i, 0) \geq 0$ for $i = 1, 2$, then the same conclusion holds for $\alpha p_1 + (1-\alpha)p_2$ for all $\alpha \in [0, 1]$. Therefore, given any p^* on the boundary, the supporting hyperplane theorem implies that we can find a linear hyperplane (λ, ω) tangent to this set at p^*.

Suppose the algorithm designer prescribes that the Receiver choose $a = 1$ at any menu p such that $\lambda \cdot p \leq \omega$ and $a = 0$ otherwise. Note that having the Receiver choose $a = 1$ therefore requires choosing p where the Sender would rather the Receiver choose action $a = 0$, by definition of K. Therefore, the strategic player cannot do any better than choosing $\sigma(p \mid \theta)$ which is a point mass at p^*. □

Proof of Lemma 2. It suffices to show that if $p > v_L$ and $\mathbf{E}(v|p) - p \geq 0$, then the expected profit from p is strictly less than $\pi_L v_L$. We write the proof in Rubinstein (1993) for the later reference. For any price p satisfying

$$\mathbb{P}(H|p)v_H + \mathbb{P}(L|p)v_L \geq p,$$

the revenue cannot exceed

$$\mathbb{P}(H|p)v_H + \mathbb{P}(L|p)v_L$$

but the cost is

$$\mathbb{P}(H|p)(1-r)c_2 + \mathbb{P}(H|p)rc_1.$$

Thus, the Sender's expected payoff is at most

$$\mathbb{P}(L|p)v_L + \mathbb{P}(H|p)((1-r)(v_H - c_2) + r(v_H - c_1))$$

Because of the lemon's problem,

$$(1-r)(v_H - c_2) + r(v_H - c_1) < 0$$

and

$$\mathbb{P}(H|p) > 0$$

to satisfy

$$\mathbb{P}(H|p)v_H + \mathbb{P}(L|p)v_L \geq p > v_L.$$

Integrating over p, we conclude that the ex-ante profit is strictly less than $\pi_L v_L$.

□

Proof of Lemma 3. Let (x, q) be some offer such that the agent is indifferent between accepting and rejecting, so that:

$$q - \mathbb{E}[\theta : (x, q)]x = 0$$

The principal's expected payoff is found by taking the expectation of $u(\theta, (x, q), a)$ over all realizations of x, q. By the law of iterated expectations, this occurs if and only if the principal's payoff is maximized following *each* realization of (x, q). We claim the principal is *not* indifferent between actions following any such (x, q). Indeed, letting $\mathbb{E}[\theta : (x, q)] = r$, indifference implies:

$$(1-r)f(I - rx) + rf(I - L + (1-r)x) = (1-r)f(I) + rf(I - L).$$

Note that equality holds if $f(y) = y$. This implies that both lotteries, whether or not the principal accepts, have the same expected values. However, if f is concave, then since $I > I - rx > I - L + (1-r)x > I - L$, it must be that the left hand side is strictly greater than the right hand side.

It follows that if indifference holds, the principal strictly prefers the agent accept the offer by slightly reducing x. □

References

Abreu, D., and A. Rubinstein (1988): "The Structure of Nash Equilibrium in Repeated Games with Finite Automata," *Econometrica*, 56(6), 1259–1281.

Al-Najjar, N. I. (2009): "Decision Makers as Statisticians: Diversity, Ambiguity and Learning," *Econometrica*, 77(5), 1371–1401.

Al-Najjar, N. I., and M. M. Pai (2014): "Coarse decision making and overfitting," *J. Economic Theory*, 150, 467–486.

Anderson, R., and H. Sonnenschein (1982): "On the Existence of Rational Expectations Equilibrium," *Journal of Economic Theory*, 26, 261–278.

——— (1985): "Rational Expectations Equilibrium with Econometric Models," *Review of Economic Studies*, 52(3), 359–369.

Arora, R., O. Dekel, and A. Tewari (2012): "Online Bandit Learning against an Adaptive Adversary: from Regret to Policy Regret," in *Proceedings of the 29th international coference on international conference on machine learning*, pp. 1747–1754.

Arora, R., M. Dinitz, T. Marinov, and M. Mohri (2018): "Policy Regret in Repeated Games," in *Proceedings of the 32nd international conference on neural information processing systems.*

Aumann, R., and S. Sorin (1989): "Bounded Rationality and Cooperation," *Games and Economic Behavior*, 1, 5–39.

Basu, P., and F. Echenique (2019): "Learnability and Models of Decision Making under Uncertainty," forthcoming in Theoretical Economics.

Ben-Porath, E. (1993): "Repeated Games with Finite Automata," *Journal of Economic Theory*, 59, 17–32.

Best, J., and D. Quigley (2020): "Persuasion for the Long Run," Discussion paper, Carnegie Mellon University and University of Oxford.

Blum, A., M. Hajiaghayi, K. Ligett, and A. Roth (2008): "Regret minimization and the price of total anarchy," in *Proceedings of the fortieth annual ACM symposium on Theory of computing*, pp. 373–382.

Braverman, M., J. Mao, J. Schneider, and M. Weinberg (2018): "Selling to a No-Regret Buyer," in *ACM Conf. on ACM Conference on Economics and Computation (ACM EC)*, pp. 523–538.

Brown, Z., and A. MacKay (2021): "Competition in Pricing Algorithms," .

Calvano, E., G. Calzolari, V. Denicolò, and S. Pastorello (2019): "Artificial Intelligence, Algorithmic Pricing and Collusion," .

Camara, M., J. Hartline, and A. Johnsen (2020): "Mechanisms for a No-Regret Agent: Beyond the Common Prior," *FOCS*.

Cherry, J., and Y. Salant (2019): "Statistical Inference in Games," Northwestern University.

Dembo, A., and O. Zeitouni (1998): *Large Deviations Techniques and Applications*. Springer-Verlag, New York, 2nd edn.

Deng, Y., J. Schneider, and B. Sivan (2019): "Strategizing against No-regret Learners," Discussion paper.

Dietterich, T. G. (2000): "Ensemble Methods in Machine Learning," in *Multiple Classifier Systems*, pp. 1–15, Berlin, Heidelberg. Springer Berlin Heidelberg.

Eliaz, K., and R. Spiegler (Forthcoming): "The Model Selection Curse," *American Economic Review: Insights*.

Foster, D., and R. Vohra (1997): "Calibrated Learning and Correlated Eequilibrium," *Games and Economic Behavior*, 21, 40–55.

Frenay, B., and M. Verleysen (2014): "Classification in the presence of label noise: a survey," *IEEE transactions on neural networks and learning systems,*, 25(5), 845–869.

Freund, Y. (2009): "A More Robust Boosting Algorithm," Discussion paper, University of California, San Diego.

Fudenberg, D., and D. Clark (2021): "Justified Communication Equilibrium," *American Economic Reivew*, Forthcoming.

Fudenberg, D., and K. He (2018): "Learning and Type Compatibility in Signaling Games," *Econometrica*, 86(4), 1215–1255.

Fudenberg, D., and D. M. Kreps (1995): "Learning in Extensive Form Games I: Self-confirming Equilibria," *Journal of Economic Theory*, 8(1), 20–55.

Fudenberg, D., and D. Levine (1998): *Learning in Games.* M.I.T. Press.

Fudenberg, D., and D. K. Levine (1993): "Steady State Learning and Nash Equilibrium," *Econometrica*, 61(3), 547–573.

——— (1995): "Consistency and Cautious Fictitious Play," *Journal of Economic Dynamics and Control*, 19, 1065–1089.

——— (2006): "Superstition and Rational Learning," *American Economic Reivew*, 96, 630–651.

Hart, S., and A. Mas-Colell (2000): "A Simple Adaptive Procedure Leading to Correlated Equilibrium," *Econometrica*, 68(5), 1127–1150.

Kamenica, E., and M. Gentzkow (2011): "Bayesian Persuasion," *American Economic Reivew*, 101(6), 2590–2615.

Liang, A. (2018): "Games of Incomplete Information Played by Statisticians," Discussion paper, University of Pennsylvania.

Long, P., and R. Servedio (2010): "Random Classification noise defeats all convex potential boosters," *Machine Learning*, 78(3), 287–304.

Maskin, E., and J. Tirole (1992): " The Principal-Agent Relationship with an Informed Principal, II: Common Values," *Econometrica*, 60(1), 1–42.

Meyn, S. P. (2007): *Control Techniques for Complex Networks.* Cambridge University Press.

Mukherjee, I., and R. E. Schapire (2013): "A Theory of Multiclass Boosting," *Journal of Machine Learning Research*, 14, 437–497.

Nekipelov, D., V. Syrgkanis, and E. Tardos (2015): "Econometrics for Learning Agents," in *Proceedings of the Sixteenth ACM Conference on Economics and Computation*, pp. 1–18.

Olea, J. L. M., P. Ortoleva, M. M. Pai, and A. Prat (2019): "Competing Models," Columbia University, Princeton University and Rice University.

Piccione, M., and A. Rubinstein (1993): "Finite Automata Play a Repeated Extensive Game," *Journal of Economic Theory*, 61, 160–168.

Rambachan, A., J. Kleinberg, S. Mullainathan, and J. Ludwig (2020): "An Economic Approach to Regulating Algorithms," Discussion paper, Harvard Universitiy, Cornell University, and University of Chicago.

Rubinstein, A. (1986): "Finite Automata Play Repeated Prisoners Dilemma," *Journal of Economic Theory*, 39(1), 83–96.

——— (1993): "On Price Recognition and Computational Complexity in a Monopolistic Model," *Journal of Political Economy*, 101(3), 473–484.

Schapire, R. E., and Y. Freund (2012): *Boosting: Foundations and Algorithms.* MIT Press.

Shalev-Shwartz, S., and S. Ben-David (2014): *Understanding Machine Learning: From Theory to Algorithms*. Cambridge University Press.

Spence, A. M. (1973): “Job Market Signaling,” *Quarterly Journal of Economics*, 87(3), 355–374.

Spiegler, R. (2016): “ Bayesian Networks and Boundedly Rational Expectations *,” *The Quarterly Journal of Economics*, 131(3), 1243–1290.

Zhao, C., S. Ke, Z. Wang, and S.-L. Hsieh (2020): “Behavioral Neural Networks,” Discussion paper.