Strategic Behavior Prediction

Kevin G. A. Waugh

CMU-CS-22-144
August 2022

Computer Science Department
School of Computer Science
Carnegie Mellon University
Pittsburgh, PA 15213

Thesis Committee:
Dr. J. Andrew Bagnell (Chair)
Dr. Geoffrey J. Gordon
Dr. Ariel D. Procaccia
Dr. Michael L. Littman (Brown)

*Submitted in partial fulfillment of the requirements
for the degree of Doctor of Philosophy.*

This research was sponsored by a NSERC Fellowship, NDSEG Fellowship, and the Office of Naval Research under award number N000140911052. The views and conclusions contained in this document are those of the author and should not be interpreted as representing the official policies, either expressed or implied, of any sponsoring institution, the U.S. government or any other entity.

Keywords: Behavior Prediction, Equilibrium Computation, Game Theory, Maximum Entropy, Regret Estimation

To my parents, Betty-Anne and Gerry, my sister, Jennifer, my wife, Monica and my children, Benjamin and Cassidy.

Abstract

Predicting strategic goal-oriented multi-agent behavior from observations of play is a ubiquitous task that spans not only computer science, but economics, finance, decision theory, and psychology. Unlike in the single-agent setting, agents cannot myopically maximize their utility. Instead, they must reason about the intentions and goals of the others' to compete for common resources or to cooperate to achieve a common goal. This complicates matters dramatically, as the natural solution concepts are computationally intractable and do not fail gracefully when even a single player deviates from the equilibrium. Furthermore, the behavior we observe is typically non-smooth and often intentionally deceptive to improve robustness and incorporate aspects of information hiding, *e.g.*, bluffing, misleading or slow-playing.

In this thesis, we advocate estimating dual variables, *i.e.*, utilities and regrets, instead of working with the primal behavior directly. We argue these dual variables are easier to estimate, in both theory and practice, than the primal behavior. In particular, these utilities are often relatively smooth, and exhibit nice problem-specific structure. Additionally, in some cases these dual estimates are all that are available—primal procedures are simply inapplicable.

We consider behavior prediction in normal-form games and introduce the inverse correlated equilibria (ICE) polytope. This polytope contains the joint-strategies that have no more internal regret than the demonstrated behavior under a potentially unknown utility function. This is similar in spirit, and inspired by, single-agent imitation learning/inverse reinforcement learning methods like Abbeel and Ng [1] and Ziebart et al. [97], which guarantee the expected reward of the prediction matches the demonstrated behavior. Our method, MaxEnt ICE, predicts the maximum entropy joint-strategy from the ICE polytope. This results in a convex objective that we efficiently optimize with simple gradient-based algorithms as well as strong statistical guarantees on the quality of the resulting prediction. This approach provides robust game-theoretic behavior estimates even when observations are scarce. We experiment with our approach on human behavior from psychological studies, as well as real-world firm market entry behavior.

To conclude, we tie our estimation approach to the large body of work on game abstraction and equilibrium computation in zero-sum extensive-form games. In particular, a crucial step of equilibrium computation is estimating the utilities induced by a strategy. The chosen abstraction serves as a model class providing computational tractability and improving generalization. Following this insight, we introduce regression counterfactual regret minimization (RCFR), a generalization of CFR, an algorithm for zero-sum equilibrium computation. In essence, RCFR drastically improves flexibility in abstraction design.

Acknowledgments

First and foremost, I am incredibly thankful to have had the opportunity to work with and be mentored by Drew Bagnell. I cannot begin to express how fortunate I feel in this regard. Quite simply, I would have not completed the program without his continuing support. Thank you for your patience and guidance.

Thank you to my committee, Geoff Gordon, Michael Littman, and Ariel Procaccia. Your thorough criticism and helpful insights have been invaluable, and your continued patience is greatly appreciated. In particular, thank you to Michael for your flexibility and for hosting me in Providence. My time there meeting with you, Amy Greenwald, and your students was invaluable and thought provoking.

A special thanks to Michael Bowling, Martin Müller, Piotr Rudnicki, Jonathan Schaeffer, Dale Schuurmans, and Duane Szafron, all of whom were instrumental in and supportive of my choice to partake in graduate studies at the University of Alberta and continuing at Carnegie Mellon. Thank you all for your continued support.

Thank you to all the members and friends of the LAIRLAB, past and present. In particular, Shreyansh Daftry, Debadeepta Dey, Felix Duvallet, Allie Del Giorno, Alex Grubb, Hanzhang Hu, Shervin Javdani, Tommy Liu, Kris Kitani, Dan Munoz, Nick Rhinehart, Stephane Ross, Shaurya Shankar, Wen Sun, Arun Venkatraman, Paul Vernaza, Jiaji Zhou, and Brian Ziebart. Your help critiquing posters and presentations was invaluable and greatly appreciated. It was enjoyable and enlightening to watch your research germinate and ultimately come to fruition.

Thank you to my friends from Carnegie Mellon. I could not have made it through this without your support. Our social outings were a delightful way to escape from the long hours in the Gates-Hillman Center. In particular, thanks to my roommates, Jayant Krishnamurthy, Shiva Kaul and Alex Limpaecher; my hallmates, John Dickerson, Sam Ganzfried, Christian Kroer, Jamie Morgenstern, Abe Othman, Erik Zawadzki; my teammates, Severin Hacker, Jiri Simsa; and Byron Boots, Chris Dellin, Anca Dragan, Miro Dudik, Elie Kravat, Akshay Krishnamurthy, Jackie Libby, Anders Oland, Cyrus Omar, Brenden Meeder, Aaditya Ramdas, Matt Stanton, Ekaterina Taralova, Catherine and Gabe Weisz, and Jing Xiang.

Thank you to my friends from afar. It is true that you do not know what you are missing until it is gone. I missed you all since day one and I am grateful to have the opportunity to come home while continuing to do what I love. Thanks to the Computer Poker Research Group, Nolan Bard, Neil Burch, Joshua Davidson, John Hawkin, Michael Johanson, Morgan Kan, Dustin Morrill, Nick Abou Risk, Marty Zinkevich; and others I met along the way, Jessica Enright, Sam Bou Fahkreddine, Jeff Grajkowski, Andy and Stacey Hiew, John Jamieson, Jennifer Krause, Kurt and Chloe McMillan, Tom and Laura Mortimer, Mark Trommelen, Jeff Siegel, Matt Stephure, Christian Smith, Eric Sprenger, and Terry White.

Finally, thank you to my family: my mother Betty-Anne, my father Gerry, my sister Jennifer, my wife Monica, my children Benjamin and Cassidy, and my in-laws Alison, Ashley, Cara, Keith, Michael, Nathan, and Wayne.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

The task of predicting purposeful agent behavior is well-studied and of interest across numerous disciplines. In the econometrics field, it is common to use publicly available information, like demographic and regulatory indices along with tax and sales records, to uncover consumer preferences or infer hidden production costs [63, 74]. For example, it was determined that there is no price fixing in the ready-to-eat cereal market despite the consumer prices vastly exceeding production costs [69]. Through analysis of sales and market data, researchers determined the unaccounted for revenue was being spent on advertising. Other studies have examined the midscale hotel market in Texas to determine the effects of government regulation [91] and market entry costs [17]; as well as substitution effects in the automobile market [4].

In the decision theory community, the same overarching task is viewed from a different angle. It is common knowledge that game-theoretic solution concepts fail to accurately predict human behavior. Some argue that this is due to a modeling problem—the theory is correct, but crucial information is lost or simply unavailable when describing a decision-making scenario as a mathematical object. Not surprisingly, a wide range of exogenous factors influence a person's utility and behavior. For example, people are typically risk averse, overly sensitive to low probability outcomes, and easily influenced by framing effects [26, 67]. Additionally, interactions are often repeated and thus issues relating to reputation beyond the model come into play [26]. Even when maintaining a reputation is unnecessary, behavior is often altruistic. Furthermore, when this goodwill is taken advantage of one might be spiteful in future interactions, even with an unrelated individual. Laboratory studies using small games are well-suited for isolating and validating these effects, but understanding how they all combine into a single number, the player's utility, is challenging and not at all well-understood.

Others believe that humans are simply irrational either due to the inability to fully reason about a situation or that their behavior is subject to errors in execution. For example, even if one's only goal is to win a Chess match, flawless play by a human is surely unattainable in such a complex game. Such reasoning argues that the standard game-theoretic solution concepts, such as the Nash equilibrium, are not sufficient to describe human behavior. We need to explicitly model our limitations, like bounded memory and the inability to forecast beyond a short horizon [13, 25]. This has led to the introduction of new equilibrium concepts, like the quantal response equilibrium [64]. Here, players are expected to err with more costly mistakes being less likely than minor ones. Like the Nash equilibrium, these new solution concepts are accompanied by a host of computational issues and hyperparameters dramatically hindering their ability to scale beyond toy games.

Roughly, econometricians are most interested in the underlying causes and intentions behind observed market behavior. The particular workings of the market, like its pricing rules, are fixed *a priori*. In contrast, the decision theory community aims to understand operationally how humans make decisions. Machine learning applications, on the other hand, consider predictive performance the primary goal. In single-agent settings, inverse optimal control (IOC) techniques assume that the observed behavior is an approximate solution to a parameterized decision problem [1, 70, 75, 76, 99]. By identifying the problem's parameters, an off-the-shelf planner or another suitable decision-making algorithm can imitate the original behavior in novel scenarios. Successful applications include robotic path planning [75, 82] and foot placement [76]; crowd navigation [44] and pedestrian movement prediction [54]; personalized route planning [97]; taxi cab route prediction [98]; and video car racing [78].

Though much of the work from the machine learning community has focused on the single-agent setting, some have delved into strategic behavior prediction. Ortiz et al. suggest that one predict the maximum entropy correlated equilibrium in the absence of observations. Though this optimization is convex, and thus tractable, it requires known utility functions, assumes completely rational players, and makes no use of observations of play [72]. Gao and Pfeffer learn both the game's structure and the players' strategies simultaneously by reduction to a constraint satisfaction problem [34]. Wright and Leyton-Brown use local search techniques to learn the best fitting hyperparameters of a number of equilibrium concepts from observations of human play [96]. Both the latter methods are computationally intensive and have trouble scaling beyond toy problems.

In this thesis, we tackle strategic behavior prediction from the machine learning viewpoint. Our goal is to construct **accurate predictions** of the players' behavior using few observations of play. We aim primarily for strong predictive performance as it is **measurable**, unlike cause or intention upon which we can only speculate. Low sample complexity is of particular importance in multi-agent scenarios as typically the size of a game scales exponentially in the number of players. Consequently, our procedures must be robust to a large amount of noise.

The main contributions of this thesis are:

- A family of computationally tractable methods for estimating multi-agent behavior from regret-feature estimates, MaxEnt ICE (Inverse Correlated Equilibrium) and friends.
- Experimental validation advocating for and demonstrating the efficacy of MaxEnt ICE.
- A generalized view of game abstraction combined with a new method for equilibrium computation in zero-sum extensive-form games, Regression CFR (RCFR).

In Chapter 2, we introduce the overarching theme of our thesis—rather than estimating the agents' behavior directly, the so-called primal problem, we instead first estimate their regret, the dual problem. This is akin to methods for single-agent settings that estimate the player's utility. We present a computationally efficient procedure based on the principle of maximum entropy to convert these regret estimates into the primal behavior.

We argue that estimating regrets (or in the zero-sum case, utility given to the opponent), is much easier than estimating the primal behavior directly. There are at least three reasons for this. First, when there are three or more agents with (potentially) correlated behavior there is simply fewer regrets to estimate than primal joint-actions. Second, when behavior is strategic the regrets are often simpler and smooth in ways that the primal behavior is not. For example, in poker it is common to bluff, *e.g.*, bet with a poor hand hoping your opponent folds a better hand. This results in discontinuities in the primal strategy, *e.g.*, checking with medium strength hands and betting with both weak and strong hands. Here, despite the jagged primal strategy, the utility of each hand increases smoothly with rank. Third, in games of imperfect information, we typically do not completely observe the players' primal strategy even in the limit of infinite observations. For example, in poker we do not observe an opponents' hand—his private information—when he folds. We know our own private cards. Thus, we completely observe the utility the player provides us, *i.e.*, the dual variables that we advocate estimating.

An additional selling-point of our approach is that our recovery procedure is, in a sense, game-theoretic and robust to errors in our regret estimates. In sequential decision-making scenarios, we often do not observe play in all situations that may arise. For example, a chess grandmaster will rarely blunder his queen, thus we cannot know how he will continue playing after such a mistake. Both primal and dual procedures must somehow cope with these unobserved situations. Our maximum entropy recovery procedure is robust to errors in regret estimates. That is, in unobserved situations, or rarely observed situations, we do not require our prediction to as closely match our regret estimates. In these situations, it will simply predict "good" equilibrium-like behavior consistent with the remainder of the strategy.

We conclude Chapter 2 with experimental results on a number of small matrix games played by humans for social science experiments in laboratory settings. We compare our approach to a variety of machine learning and game-theoretic approaches.

In Chapter 3, we extend our approach to situations where the agents' utility is not known. Here, we assume instead that the agents' utility is an unknown linear function of known utility features. This is particularly important from a modelling perspective as players' utility is often a combination of a number of factors and subject to personal taste. For example, consider a player choosing between taking public transit or driving to work. These two choices trade off between time, money and comfort. These are features we can (to an extent) observe, but how they combine into a single number, the player's utility, cannot be known a priori and will vary from person to person.

We conclude Chapter 3 with experimental results predicting where mid-scale hotels will be built in the state of Texas. Here, we observe a number of demographic and regulatory features of each county as well as public tax records from which we determine when and where hotels are built. That is, we do not observe each hotel's profit or their true utility (which likely is not purely profit). In this setting, prior game-theoretic approaches cannot be applied, both because they do not scale to games of this size and because the utility cannot be specified.

Finally in Chapter 4, we extend our dual estimation approach to zero-sum extensive-form games with known utility. Additionally, we integrate our regret estimation technique with a modern equilibrium-finding approach, counterfactual regret minimization (CFR). This integration allows us to draw parallels between our estimation technique and modern game abstraction techniques [80]. In particular, we can view modern game abstraction as defining a piecewise constant model class. During equilibrium computation, we model the players' regrets by a function in this class, which we update each iteration. From this viewpoint, it is clear that it is not necessary to use a piecewise constant estimator—we can choose to use any other functional form.

Together, these contributions demonstrate that we can **accurately** and **tractably** estimate strategic behavior in a plethora of multi-agent scenarios. These estimates are theoretically sound backed by both game theory and information theory. Furthermore, they are empirically validated by a wide range of applications from decision theory, economics and game playing.

Chapter 2

Behavior Prediction with Known Utility

2.1 Definitions

Normal-form games, or matrix games, are the canonical tool used by game-theorists to study competitive and cooperative behavior [73]. Well known examples include the famous "Prisoner's Dilemma", the "Dove and Hawk" game used to model mutually assured destruction during the Cuban missile criss, and the "Battle of the Sexes". They are complete in the sense that they can represent any finite interaction, but often not in the most concise fashion. Like those before us, we begin our journey on behavior prediction here.

Definition 1 (based on Osborne and Rubinstein [73])**.** *Formally, a vector-valued **normal-form game** or **matrix game** is a tuple $\Gamma = (N, \mathcal{A}, u, w^*)$ where*

- *$[N]$ is the set of **players**,*
- *$\mathcal{A} = \times_{i \in [N]} A_i$ is the set of **joint-actions** or **outcomes**,*
- *A_i is the set of player i's **actions**,*
- *$u_i : \mathcal{A} \to \mathbb{R}^K$ is player i's **utility feature function**, and*
- *$w^* \in \mathbb{R}^K$ is the **true utility function**, a linear function.*

We let $A = \max_{i \in [N]} |A_i|$ be the maximum number of actions available to any player. In general the description of such a game is of size $|\mathcal{A}| = O(A^N)$.

Unlike the standard definition, the games we consider are *vector-valued*—each outcome is endowed with a whole vector of utility features. Each feature measures a quantity of the outcome that may correlate with the agents' utilities, like money spent, time taken, fuel used, oranges consumed, *etc*. We assume that the true utility is formed by a common linear function of these utility features, $w^* \in \mathbb{R}^K$. For now, we assume that w^* is known to us. We will relax this assumption in future chapters. Note that we can facilitate player-specific utility functions by simply expanding the feature space.

A matrix game is a one-shot interaction between the N players. Simultaneously and without knowledge of the others' choices, each player $i \in [N]$ chooses an action a_i from its action set A_i. These actions form the game's outcome, a tuple $a = (a_1, a_2, \ldots, a_N)$, and each player receives **utility** $\langle u_i(a), w^* \rangle$, a cardinal measure of happiness or individual well-being.

A player draws its action from its **mixed strategy**, $\sigma_i \in \Delta_{A_i}$, a probability distribution over its actions. A **strategy profile**, $\sigma = (\sigma_1, \sigma_2, \ldots, \sigma_N)$, is a tuple of strategies, one for each player.

We use σ_{-i} to denote the tuple of strategies from profile σ excluding player i.

Given a strategy profile σ, player i's **expected utility features** are defined as

$$\begin{aligned} u_i(\sigma) &\doteq \mathbb{E}_{a\sim\sigma}\left[u_i(a)\right] \\ &= \sum_{a\in\mathcal{A}} u_i(a) \prod_{i\in[N]} \sigma_i(a_i), \end{aligned}$$

and its **expected utility** is $\langle u_i(\sigma), w^* \rangle$.

If player i knows all the strategies for the other players, σ_{-i}, its **best response** is any strategy maximizing its utility:

$$\sigma_i^{\text{BR}} \in \operatorname*{argmax}_{\sigma_i\in\Delta_{A_i}} \langle u_i(\sigma_i, \sigma_{-i}), w^* \rangle$$

Unlike in the single agent setting, players cannot myopically maximize their own utility, *i.e.*, play a best response, because they are not privileged to the others' strategies. They must reason about the others' intentions and act accordingly, *e.g.*, to coordinate to achieve a common goal or to strategically battle for a shared resource. For this reason, we generally need an alternative criteria to evaluate the optimality of the entire strategy profile.

A strategy profile is *stable*, or in equilibrium if no player can benefit by deviating from it. Said another way, each player's strategy in profile σ is a best response. We say that a strategy profile is an **ε-Nash equilibrium** if no player can benefit more than ε by deviating:

$$\left\langle u_i(\sigma_i^{\text{BR}}, \sigma_{-i}), w^* \right\rangle \leq \langle u_i(\sigma), w^* \rangle + \varepsilon \qquad\qquad \forall i \in [N]$$

John Nash's celebrated theorem, for which he won a Nobel prize, demonstrates non-constructively that such an equilibrium exists in any game [68]. Unfortunately, in general computing a Nash equilibrium is computationally hard; specifically, it is PPAD-complete [20]. There is no known polynomial time algorithm to compute such a profile. Additionally, many decision problems associated with Nash equilibria are NP-hard, such as determining if a game has a pure strategy Nash equilibrium, or determining if an action is in the support of any equilibrium [16]. Furthermore, there are no known dynamics that converge to a Nash equilibrium. We will elaborate further on these deficiencies of this solution concept in future sections.

2.1.1 Zero-Sum Games

We call a game **two-player zero-sum**, or just **zero-sum**, if $N = 2$ and

$$\langle u_2(x, y), w^* \rangle = -\langle u_1(x, y), w^* \rangle . \qquad\qquad \forall x \in A_1, y \in A_2$$

That is, what one player wins the other player loses. Zero-sum games are purely strategic—there is no way to benefit by cooperating with the opponent.

Zero-sum games are an important special-case. Many popular games played by humans are zero-sum, *e.g.*, chess, checkers, backgammon, go and some forms of poker. Furthermore, unlike general matrix-games, there are efficient approximation algorithms for computing ε-Nash equilibrium in large games. Additionally, there are also simple dynamics that converge towards a Nash equilibrium [29, 61].

Nash equilibria correspond directly with the saddle-points of the convex minimax problem:

$$\max_{\sigma_1 \in \Delta_{A_1}} \min_{\sigma_2 \in \Delta_{A_2}} \quad \langle u_1(\sigma), w^* \rangle \, .$$

There are many algorithms to tackle this minimax problem directly. For example, Newton's method, the fixed point iteration, or two no-regret learners in self-play. Alternatively, it is insightful to convert the problem into the linear program:

$$\begin{aligned} (\sigma_1^{\text{NE}}, v^*) \in \operatorname*{argmax}_{\sigma_1 \in \Delta_{A_1}, v} \; v \quad & \text{subject to:} \\ \langle u_1(\sigma_1, y), w^* \rangle \geq v. \qquad & \qquad \qquad \forall y \in A_2 \end{aligned}$$

We call v^* the value of game Γ. This value is unique, and the first player can always attain it by playing a Nash equilibrium strategy. That is, playing a Nash equilibrium in a zero-sum game is minimax optimal. This is known as von Neumann's minimax theorem [92]. Unlike in general-sum games, there is no equilibrium selection problem. All Nash equilibrium are *exchangeable*. It is safe to play an equilibrium against an unknown opponent.

A further consequence of the linear programming formulation is that we can write the set of ε-Nash equilibrium strategies for player one as:

$$\Sigma_1^{\varepsilon\text{-NE}} = \{ \sigma_1 \in \Delta_{A_1} \mid \langle u_1(\sigma_1, y), w^* \rangle \geq v^* - \varepsilon, \forall y \in A_2 \} \, .$$

It is convenient to define player one's **strategy value function**, $v_1 : \Delta_{A_1} \to \mathcal{R}$ as

$$v_1(\sigma_1) = \min_{\sigma_2 \in \Delta_{A_2}} \quad \langle u_1(\sigma), w^* \rangle \, .$$

Note that we have $v_1(\sigma_1) \leq v^*$ for all $\sigma_1 \in \Delta_{A_1}$.

We also define player one's **exploitability** function $\varepsilon_1 : \Delta_{A_1} \to \mathcal{R}$ as

$$\varepsilon_1(\sigma_1) = v^* - v_1(\sigma_1).$$

With this notation, we can write the set of ε equilibrium strategies more concisely,

$$\Sigma_1^{\varepsilon\text{-NE}} = \{ \sigma_1 \in \Delta_{A_1} \mid \varepsilon_1(\sigma_1) \leq \varepsilon \} \, .$$

We will focus on behavior prediction in zero-sum games. This will allow us to derive computationally tractable algorithms with the ability to scale to large games. We will soon argue that this restriction is not a major hindrance, *i.e.*, we will be able to apply our techniques in non-zero-sum games as well. Specifically, we will attain this by considering the correlated equilibrium solution concept.

2.2 Problem Statement

Initially, imagine we wish to estimate the strategy of player one in a zero-sum game from a small set of observations of play. We assume that the observations are drawn from a static strategy $\sigma_1 \in \Delta_{A_1}$, *i.e.*, player one is not learning or adapting. We call σ_1 the **true behavior**. We call our estimator, $\hat{\sigma}_1$, the **predicted behavior**. We take the role of an outside observer—we take no actions in the game and we aim for strong predictive performance. We do not try maximize utility in the game, by say constructing a counter-strategy.

There are two ways that we can formally define the set of observations we recieve. The first and most obvious set of observations we call the **primal view**. In the primal view, we observe a sequence of T actions from player one in game Γ, $\{x^t \in A_1\}_{t=1}^T$. When privileged to primal observations, it is as if we are watching the game from over the shoulder of player one. Recall, we are not privileged to player one's actual strategy; only to samples from it.

The second set of observations we call the **dual view**. Here, we observe a sequence of T utility feature functions for player *two*, $\left\{u_2^t : A_2 \to \mathbb{R}^K\right\}_{t=1}^T$, where $\mathbb{E}[u_2^t(y)] = u_2(x^t, y)$ for all $y \in A_2$. That is, $u_2^t(y)$ is an unbiased estimate of the utility for player two at time t. Dual observations are akin to watching the game from over the shoulder of player two.

Note that the dual view is strictly weaker than the primal view. Given primal observations, we can construct dual observations. The converse is not true. For example, in a hand of poker where player one folds, player two does not know with certainty what hand his opponent held, *i.e.*, player two does not completely observe player one's private information. As a consequence, the dual estimation methods are more broadly applicable than primal methods.

In many applications, such as opponent modelling and opponent exploitation, we are not privileged to primal observations. Despite this, most prior techniques require the more natural primal observations [3, 31, 47, 48]. Thus, these methods are in theory unimplementable for almost all imperfect information games. In practice, one typically employs dubious heuristics to fake full primal observations from the available partial information [30].

We reiterate that when we observe the game from player two's point-of-view, we will take no actions in the game. Unlike in sequential settings, this is inconsequential in normal-form games, as exploratory actions are not necessary to examine the entire space of strategies.

We will measure the quality of our prediction by the Kullback-Leibler divergence (KL) to the true behavior [59],

$$\mathrm{KL}(\sigma_1 \| \hat{\sigma}_1) \doteq \mathbb{E}_{a \sim \sigma_1}\left[\log_2 \sigma_1(a) - \log_2 \hat{\sigma}_1(a)\right].$$

The KL divergence measures the number of extra bits required to code σ using $\hat{\sigma}$. That is, it is zero if and only if the two distributions are the same. It is not a metric, as it is not symmetric and does not satisfy the triangle inequality.

Of primary concern is the statistical complexity (*i.e.*, sample complexity) of an approach. That is, how many samples T are necessary to accurately predict the agents' behavior. For example, in multi-player games if we naively view the joint behavior as a high-dimensional multinomial distribution we may need $O(A^N)$ observations to estimate its parameters. The hope is that we can leverage the fact that the behavior is purposeful, or goal driven, to derive tractable estimation methods.

A secondary concern is the computational complexity of an estimator. Many game-theoretic solution concepts, like Nash equilibria in general games, are computationally hard to compute. These concepts are inapplicable beyond toy scenarios and the motivation that they represent how imperfect agents reason is dubious at best. Using these intractable solution concepts as the foundation of an estimator unnecessarily limits scalability or forces us to make modeling concessions.

2.3 Dual Strategy Estimation

Before we introduce our dual strategy estimation technique, let us review the correlated equilibrium solution concept. This will allow us to predict behavior in any game, *e.g.*, with more than two players, by predicting a single player's behavior in a zero-sum game.

2.3.1 Correlated Equilibria and Zero-sum Games

If the players are able to communicate prior to play they can agree on joint-strategy, $\sigma \in \Delta_{\mathcal{A}}$, a distribution over the game's joint-actions that governs how they will play. This allows the players to coordinate and cooperate should it be advantageous to do so.

Operationally, coordination can occur in a number of ways. For example, a neutral third party can sample an outcome $a \sim \sigma$ and privately tell each player i its action a_i. Without any loss of generality, this is the viewpoint commonly taken in theoretical works. Alternatively, the players may simply all observe a common random signal, *e.g.*, the color of a traffic signal, or simply a stream of random bits. Less obviously, in repeated settings the players can communicate and coordinate through their action histories [28, 40]—no explicit communication, *i.e.*, through a side channel, is necessary to achieve correlated behavior.

Given a joint-strategy $\sigma \in \Delta_{\mathcal{A}}$, we write player i's expected utility features as

$$u_i(\sigma) = \mathbb{E}_{a \sim \sigma}\left[u_i(a)\right].$$

In the simultaneous one-shot setting, the players have no additional information available to them when they are to choose their action. Here, they know their portion of the sampled joint-action. When choosing whether or not to deviate, they may condition on this information [5]. We define a switch deviation as

$$\text{switch}_i^{x \to y}(a) = \begin{cases} y & \text{if } a = x \\ a & \text{otherwise} \end{cases}$$

We write the set of switch deviations

$$\Phi^{\text{int}} = \left\{\text{switch}_i^{x \to y} \mid i \in [N], x, y \in A_i\right\}.$$

In general, a deviation for player i is any function $f_i : A_i \to A_i$. We overload $f_i : \mathcal{A} \to \mathcal{A}$ to be the function that modifies player i's action by deviation f_i and leaves all other players' actions unchanged. Note, there is no additional power gained by considering stochastic deviations.

The regret concept corresponding the set of switch deviations is **internal regret**. We write the internal regret features with respect to switch deviation where player i changes x to y as

$$r_i^{x \to y}(\sigma) = u_i(\text{switch}_i^{x \to y}(\sigma)) - u_i(\sigma).$$

An encompassing regret concept is swap regret, which corresponds to the set of **swap deviations**. Swap deviations allow a player to arbitrarily map any action to any other action. That is, a player may choose a switch deviation for every available action.

A joint-strategy is stable, or in equilibrium, if none of the players have any incentive to deviate through any combination of switch deviations.

Definition 2 (from Aumann [2]). *A joint-strategy is an* ***ε-correlated equilibrium*** *if it has no more than ε swap regret.*

$$\sum_{x \in A_i} \max_{y \in A_i} \ \langle r_i^{x \to y}(\sigma), w^* \rangle \leq \varepsilon. \qquad\qquad \forall i \in [N]$$

The number of swap deviations is exponential in the number of actions. Fortunately, swap regret is bounded by internal regret, which has only a polynomial number of deviations.

Lemma 1. *If a joint-strategy is an* ***ε-correlated equilibrium*** *then*

$$\langle r_i^{x \to y}(\sigma), w^* \rangle \leq \varepsilon. \qquad\qquad \forall i \in [N], x, y \in A_i$$

Proof.

$$\begin{aligned} \max_{y \in A_i} \ \langle r_i^{x \to y}(\sigma), w^* \rangle &\geq \langle r_i^{x \to x}(\sigma), w^* \rangle = 0 && \forall i \in [N], x \in A_i \\ \max_{x, y \in A_i} \ \langle r_i^{x \to y}(\sigma), w^* \rangle &\leq \sum_{x \in A_i} \max_{y \in A_i} \ \langle r_i^{x \to y}(\sigma), w^* \rangle \leq \varepsilon \end{aligned}$$

□

Lemma 2. *If a joint-strategy has ε internal regret, then it is an $A\varepsilon$-correlated equilibrium.*

Proof.

$$\begin{aligned} \max_{i \in [N]} \sum_{x \in A_i} \max_{y \in A_i} \ \langle r_i^{x \to y}(\sigma), w^* \rangle &\leq \max_{i \in [N]} \sum_{x \in A_i} \varepsilon \\ &\leq A\varepsilon \end{aligned}$$

□

We will now demonstrate a reduction from an N-player game to a two-player zero-sum game whose ε-Nash equilibrium correspond to joint-strategies with 2ε internal regret. This will ease our exposition as we will need only to concern ourselves with behavior prediction techniques for two-player zero-sum games. Additionally, it may provide some operational insight into the workings of a correlated equilibrium.

Let $\Gamma = (N, \mathcal{A}, u, w^*)$ denote the original N-player game and $\Gamma^{\text{zero}} = ([2], \mathcal{A}^{\text{zero}}, u^{\text{zero}}, w^*)$ our corresponding zero-sum matrix game. The first player of our new game will be the *moderator*, and the second will represent all N players simultaneously.

The moderator in the zero-sum game chooses a joint-action in the N-player game, $a \in A_1^{\text{zero}} = \mathcal{A}$. The opponent represents all the original games' players. Simultaneously and without knowledge of the moderators choice, the opponent chooses a switch deviation, $(i, x, y) \in A_2^{\text{zero}} \cong \Phi^{\text{int}}$. The opponent is then compensated for the incurred regret: $u_2^{\text{zero}}(a, i, x, y) = \langle r_i^{x \to y}(a), w^* \rangle$.

We can view this construction as the moderator simply paying the players to act in a particular way. The players need not concern themselves with deviating in the actual game as the moderator will subsidize them appropriately for any inaccuracy that it asks them to perform.

We will now show that the set of ε-Nash equilibrium strategies for the moderator correspond to the set of joint-strategies with ε internal regret. This is easy barring one subtlety; we must show that the value of this game is zero.

First, note the value of Γ^{zero} is at most zero. The second player can always receive zero by choosing not to deviate, *e.g.*, by choosing $(1, x, x)$ for any $x \in A_1$. Phrased another way, the moderator never receives payment from the players if they deviate optimally. Now let us construct a matching upper bound.

Let $\sigma^{\varepsilon-\text{CE}} \in \Delta_{\mathcal{A}}$ be any ε-correlated equilibrium. By definition,

$$\left\langle r_i^{x \to y}(\sigma^{\varepsilon-\text{CE}}), w^* \right\rangle \leq \varepsilon \qquad\qquad \forall i \in [N], x, y \in A_i$$

In particular, for any $\lambda \in \Delta_{\Phi^{\text{int}}}$,

$$u_2^{\text{zero}}(\sigma^{\varepsilon-\text{CE}}, \lambda) = \sum_{\substack{i \in [N] \\ x, y \in A_i}} \lambda_i^{x \to y} \left\langle r_i^{x \to y}(\sigma^{\varepsilon-\text{CE}}), w^* \right\rangle \leq \sum_{\substack{i \in [N] \\ x, y \in A_i}} \lambda_i^{x \to y} \varepsilon = \varepsilon.$$

If the moderator plays an ε-correlated equilibrium, it never pays the players more than ε. By choosing $\varepsilon = 0$, we have matching lower and upper bounds, *i.e.*, the value of Γ^{zero} is zero.

Now we can show that if $\sigma \in \Delta_{\mathcal{A}}, \lambda \in \Delta_{\Phi^{\text{int}}}$ is an ε-Nash equilibrium of Γ^{zero} then σ has no more than 2ε internal regret in Γ. From the premise, we have for all $i \in [N], x, y \in A_i$,

$$\begin{aligned} u_2^{\text{zero}}(\sigma, i, x, y) - u_2^{\text{zero}}(\sigma, \lambda) &\leq \varepsilon \\ \langle r_i^{x \to y}(\sigma), w^* \rangle &\leq \varepsilon + u_2^{\text{zero}}(\sigma, \lambda) \leq 2\varepsilon. \end{aligned}$$

The last inequality follows since the expected utility of an ε-Nash equilibrium is no more than ε from the game's value.

From this construction, we see that estimating the joint behavior of all the players in a multi-player game is equivilent to estimating the behavior of the moderator in our zero-sum game. In particular, if we take the dual view, we will observe estimates of the players' *internal regrets*, *i.e.*, the utility that the moderator bestows onto the players.

2.3.2 Rational Behavior

Now we are ready to build our dual behavior estimation technique. Roughly speaking, there are two steps. First, we will derive a set of necessary conditions—player rationality assumptions—that our predictor should satisfy. These "game-theoretic" conditions will bias our prediction towards purposeful, or goal-oriented, behavior. With these conditions we hope to make a successful bias-variance trade-off. That is, even when our assumptions do not hold exactly we hope to make more accurate predictions under low sample sizes.

We will use an information-theoretic criteria to select amongst behavior satisfying these conditions. In a sense, we aim to make our necessary conditions sufficient as well. This criteria, the principle of maximum entropy, will result in a convex optimization that we can efficiently solve. This will allow our technique to scale to larger games than prior approaches. Additionally, we will inherit its strong theoretic predictive guarantees.

In single-agent settings, we say an agent is rational if they maximize their utility. This assumption provides the foundation for a host of single-agent imitation learning techniques [1, 75, 97]. Following the same spirit, we require an analogous statement for zero-sum games. Intuitively, we say a player is rational if it maximizes its utility *against an optimal adversary*. That is, if he achieves the game's value.

More formally, we say the player is **strongly rational**, or simply **rational**, if

$$v_1(\sigma_1) = v^*,$$

or equivalently,

$$\langle u_1(\sigma_1, y), w^* \rangle \geq v^*. \qquad \forall y \in A_2$$

That is, the player achieves the value of the game against a worst-case adversary, *i.e.*, he plays an equilibrium strategy, $\sigma_1 \in \Sigma_1^{\text{NE}}$.

Relaxing the assumption slightly, we say the player is **ε-strongly rational** if

$$\langle u_1(\sigma_1, y), w^* \rangle \geq v^* - \varepsilon. \qquad \forall y \in A_2$$

That is, $\sigma_1 \in \Sigma_1^{\varepsilon\text{-NE}}$.

Our approach will first estimate $\hat{\varepsilon}$ from our dual observations. Then, we will require that our estimator, $\hat{\sigma}_1$, is $\hat{\varepsilon}$-rational. This results in the desirable quality that our estimation of the player's behavior will be at least as good (in terms of exploitability) as the true behavior.

More formally, so long as our estimator $\hat{\varepsilon}$ is consistent, *i.e.*, $\hat{\varepsilon} \to \varepsilon$, we will have

$$v_1(\hat{\sigma}_1) \to v_1(\sigma_1).$$

That is, the value of our prediction approaches the value of the player's true behavior.

2.3.3 Exploitability Estimation

Now we derive a consistent estimator of ε from dual observations and analyze its properties.

First, from the dual observations we estimate the opponent's utility feature function by simply averaging the observed functions,

$$\tilde{u}_2(y) = \frac{1}{T} \sum_{t=1}^{T} u_2^t(y). \qquad \forall y \in A_2$$

Next, we write the definition of exploitability in terms of u_2 due to the zero-sum property of the game.

$$\begin{aligned}
\varepsilon &= v^* - v_1(\sigma_1) \\
&= v^* - \min_{y \in A_2} u_1(\sigma_1, y) \\
&= \max_{y \in A_2} v^* - u_1(\sigma_1, y) \\
&= \max_{y \in A_2} v^* + u_2(\sigma_1, y) \\
&\geq v^* + u_2(\sigma_1, y). && \forall y \in A_2
\end{aligned}$$

Substituting in our estimate of the opponent's utility in place of the truth, we arrive our estimator of the first player's suboptimality,

$$\begin{aligned}
\varepsilon &\geq v^* + u_2(\sigma_1, y) \approx v^* + \langle \tilde{u}_2(y), w^* \rangle && \forall y \in A_2 \\
\hat{\varepsilon} &= \max_{y \in A_2} v^* + \langle \tilde{u}_2(y), w^* \rangle \,.
\end{aligned}$$

By Hoeffding's inequality and the union bound we have the following theorem that bounds the error in our estimator as a function of the number of observations.

Theorem 1. *For all $y \in A_2$, we have $|\,\langle \tilde{u}_2(y), w^* \rangle - \langle u_2(\sigma_1, y), w^* \rangle\,| \leq \epsilon$ with probability at least $1 - \delta$ so long as $T \geq (2\Delta^2/\epsilon^2)\log(2|A_2|/\delta)$.*

Here, Δ is a bound on the magnitude of the game's maximum utility.

Proof.

By Hoeffding's inequality, we have for any particular $y \in A_2$ the probability of failure is bounded

$$P\left(|\,\langle \tilde{u}_2(y), w^* \rangle - \langle u_2(\sigma_1, y), w^* \rangle\,| \geq \epsilon\right) \leq 2\exp\left(\frac{-T\epsilon^2}{4\Delta^2}\right)$$

Applying the union bound we have the total probability of failure is bounded

$$P\left(\bigcup_{y \in A_2} |\,\langle \tilde{u}_2(y), w^* \rangle - \langle u_2(\sigma_1, y), w^* \rangle\,| \geq \epsilon\right) \leq 2|A_2|\exp\left(\frac{-T\epsilon^2}{4\Delta^2}\right) \leq \delta$$

Rearranging to isolate T we have

$$T \geq \frac{2\Delta^2}{\epsilon^2}\log\frac{2|A_2|}{\delta}$$

□

As a simple corollary, we can also bound the error in our estimate of ε by the same quantity.

Corollary 1. *$|\hat{\varepsilon} - \varepsilon| \leq \epsilon$ with probability at least $1 - \delta$ so long as $T \geq (2\Delta^2/\epsilon^2)\log(2|A_2|/\delta)$.*

Notice that our estimator has a logarithmic dependence on the number of action's available to player two. Recall that in order to estimate player one's strategy directly from primal observations, we require $O(|A_1|)$ observations. That is, we require exponentially fewer observations to estimate the utility to player two than to estimate the strategy of player one if both players have equal sized strategy spaces.

2.3.4 Principle of Maximum Entropy

We are now able to estimate the player's suboptimality, and thus we have an estimate of the set of ε-rational behavior—$\Sigma_1^{\hat{\varepsilon}\text{-NE}}$. Next, we must devise an efficient strategy for selecting a predictor from this set of behavior. Our selection must resolve the remaining ambiguity in a way to guarantee strong predictive performance.

The principle of maximum entropy states that given known conditions or constraints on a distribution, $\sigma \in \mathcal{X}$, one should predict the distribution of maximum entropy [45]. Entropy, defined as

$$H(\sigma) = \mathbb{E}_{a\sim\sigma}\left[-\log\sigma(a)\right] = -\sum_{a\in A_1}\sigma(a)\log\sigma(a),$$

is a measure of "energy" or randomness. By choosing the most random distribution that satisfies our constraints, we aim to impose no further dependencies or correlations in the distribution. In a sense, we aim to find the distribution where our conditions are both necessary and sufficient.

When the constraints are convex, *e.g.*, they are linear equality and convex inequality constraints, then computing the maximum entropy distribution is a convex optimization problem.

$$\begin{aligned}\sigma^{\text{MaxEnt}} = \underset{\sigma\in\Delta}{\text{argmax}}\ \ & H(\sigma) \quad \text{subject to:}\\ & A\sigma = b\\ & f(\sigma) \leq 0\end{aligned}$$

The solutions to these optimization problems can be approximated to an arbitrary degree efficiently, *e.g.*, by gradient methods or second-order methods.

The principle of maximum entropy is the underlying assumption for many popular machine learning methods, such as logistic regression and conditional random fields. As we will touch on in a few sections, it has also been employed for behavior prediction in games.

A result due to Grünwald and Dawid provides some insight into the predictive properties of the maximum entropy distribution.

Theorem 2 (Grünwald and Dawid [38])**.** *The maximum entropy distribution minimizes the maximum log-loss subject to the known constraints.* i.e.*,*

$$\sigma^{\textit{MaxEnt}} = \underset{\sigma\in\mathcal{X}}{\text{argmin}}\max_{\sigma'\in\mathcal{X}}\mathbb{E}_{a\in\sigma}\left[-\log\sigma'(a)\right]$$

The principle of maximum entropy is prevalent beyond machine learning. It is used in economics and finance in the study of market makers, *e.g.*, the logarithmic market scoring rule [39]. It also underlies the Kelly criterion for optimizing portfolio growth [53].

2.3.5 Maximum Entropy ε-Equilibrium Strategy

Our set of ε-rational behavior is indeed convex—in fact it is a polytope. Thus, choosing $\mathcal{X} = \Sigma_1^{\hat{\varepsilon}\text{-NE}}$ and using the principle of maximum entropy involves a convex optimization problem.

The primal maximum entropy ε-equilibrium strategy convex program is:

$$\begin{aligned} &\max_{\sigma_1 \in \Delta_{A_1}} \; H(\sigma_1), \quad \text{subject to:} \\ &\langle u_1(\sigma_1, y), w^* \rangle \geq v^* - \varepsilon. \qquad\qquad \forall y \in A_2 \end{aligned}$$

We can use standard optimization techniques to solve the primal directly, but it is advantageous to consider the dual program:

$$\min_{\lambda \geq 0} \; (v^* - \varepsilon) \langle e, \lambda \rangle + \log Z(\lambda), \text{ where}$$

$$Z(\lambda) = \sum_{x \in A_1} \exp\left(\sum_{y \in A_2} \lambda(y) \langle u_1(x, y), w^* \rangle \right).$$

Where e is the all-ones vector. The derivation of the dual follows directly from (A.2) and (A.4). The dual objective is smooth and the optimization domain, the positive orthant, is simple. These qualities are particularly appealing as they enable the use of the projected gradient method, an incredibly simple optimization technique.

Note that the dual multipliers are similar to a strategy for the opponent with the exception that they do not normalize to one, *i.e.*, λ is not necessarily a probability distribution. This has a strong and insightful connection to imperfect information game subgame re-solving first introduced in Burch et al. [12]. We will elaborate on this connection in the conclusions and future work chapter.

The constraints in the primal problem are linear and feasible, as $\hat{\varepsilon} \geq 0$, therefore by Slater's condition strong duality holds—there is no duality gap.

Since the primal is strongly convex, we can recover the unique solution from the dual solution,

$$\sigma_1^\lambda(x) = \exp\left(\sum_{y \in A_2} \lambda(y) \langle u_1(x, y), w^* \rangle \right) / Z(\lambda).$$

Note that σ_1^λ is a *soft*, or smoothed, best response to λ. That is, it is a best response where the first player plays according to the softmax.

The gradient of the dual program is simple and of a familiar form:

$$\frac{\partial}{\partial \lambda(y)} = v^* - \varepsilon - \mathbb{E}\left[\langle u_1(\sigma_1^\lambda, y), w^* \rangle \right].$$

In words, the gradient is the difference between the value of the player's observed strategy and the value should the opponent play y. If this difference is negative, then the opponent should play y more often.

Algorithmically, we can apply projected gradient descent to optimize for λ. Starting from an arbitrary λ, *e.g.*, $\lambda = 0$, we iterate

$$\lambda'(y) = \left[\lambda(y) - \alpha \left(v^* - \varepsilon - \mathbb{E}\left[\langle u_1(\sigma_1^\lambda, y), w^* \rangle \right] \right) \right]_+$$

Here, $[x]_+ = \max\{x, 0\}$ and α is a positive step-size. In practice, we choose α to either be a constant or by using a line search.

2.3.6 Relation to Maximum Entropy Correlated Equilibrium

In the absence observations of play, Ortiz *et. al.* recommend predicting that the agents will play the maximum entropy correlated equilibrium. When we reduce a normal-form game to a zero-sum game via the correlated equilibrium reduction from Section 2.3.1, their procedure is equivalent to our dual estimation procedure using $\hat{\varepsilon} = 0$.

2.3.7 Maximum Entropy Opponent Utility Preserving Strategy

Our dual strategy estimation algorithm uses a single statistic to quantify the player's behavior, its suboptimality, ε. In order to estimate this parameter, though, we must estimate the opponent's utility *for all* possible opponent actions uniformly well. In a sense we have more information available to us than we are using to determine our prediction. In this section, we derive the analogous maximum entropy estimator that uses the entire utility estimate, $\tilde{u}_2$.

Like before, we start with the primal estimation problem. Here, we replace the constraint that the behavior be $\hat{\varepsilon}$ suboptimal with a *utility matching* constraint. That is, our behavior estimate should bestow the same utilities onto the opposing player.

$$\begin{aligned} &\max_{\sigma_1 \in \Delta_{A_1}} \quad H(\sigma_1), \quad \text{subject to:} \\ &\langle u_2(\sigma_1, y), w^* \rangle = \langle \tilde{u}_2(y), w^* \rangle \qquad\qquad\qquad \forall y \in A_2 \end{aligned}$$

The dual is similar in form to the previous dual. Here, the domain is unconstrained, as opposed to the positive orthant.

$$\begin{aligned} &\min_{\lambda} \sum_{y \in A_2} \lambda(y) \langle \hat{u}_2, w^* \rangle + \log Z(\lambda), \text{ where} \\ &Z(\lambda) = \sum_{x \in A_1} \exp\left(- \sum_{y \in A_2} \lambda(y) \langle u_2(x,y), w^* \rangle \right). \end{aligned}$$

This derivation of the dual follows directly from (A.1) and (A.4).

Again, there is no duality gap, and we can recover the primal solution from a dual solution.

$$\sigma_1^{\lambda}(x) = \exp\left(\sum_{y \in A_2} \lambda(y) - \langle u_2(x,y), w^* \rangle \right) / Z(\lambda).$$

The gradient of the objective is

$$\frac{\partial}{\partial \lambda(y)} = \langle \hat{u}_2(y), w^* \rangle - \mathbb{E}\left[\langle u_2(\sigma_1^{\lambda}, y), w^* \rangle \right].$$

It is the difference between the estimate of the opponent's utility and the opponent's utility of the behavior defined by λ. Note that this quantity is zero only if the prediction bestows the same utility to the opponent as our estimate, *i.e.*, the primal constraints are satisfied.

Again, we use gradient descent to optimize this objective. Unlike before, no projection step is necessary as we are optimizing over the entire vector space.

In practice, it is important to add a regularization term to our objective to avoid overfitting our prediction to the noise in our utility estimate. We incorporate both $L1$ and $L2$ regularization:

$$\min_{\lambda} \sum_{y \in A_2} \lambda(y) \langle \hat{u}_2, w^* \rangle + \log Z(\lambda) + \kappa \|\lambda\|_2^2 + \mu \|\lambda\|_1$$

These regularization terms can be dealt with explicitly using generalized gradient descent.

By adding the regularization terms in the dual, we change the corresponding primal program. In particular, the altered primal program no longer requires that the utility matching constraints be satisfied exactly. Instead, the constraint residual need only be small in $L1$ and $L2$ norm, the degree of which is controlled by κ and μ [24].

2.3.8 Experimental Results on Kuhn Poker

Kuhn Poker is a small poker game that was introduced and analyzed by Harold W. Kuhn. It consists of a three card deck, and a single round of betting. The game is far too small to be interesting to humans. Despite this, optimal play still exhibits interesting game-theoretic concepts, such as domination and bluffing.

The game is played as follows. First, each of the two players antes a single chip into the pot and is dealt a single private card from the shuffled deck—either a jack, a queen or a king. The first player has the option of checking or betting a single chip. If the first player checks, the second player may bet a single chip, moving action back to the first player, or check behind, leading to a showdown. When either player is facing a bet, they may call, matching the bet and ending play with a showdown, or fold, forfeiting the pot to the other player.

At a showdown, both players reveal their private cards and the player with the highest card wins all the chips in the pot. In Kuhn poker there are no ties, as each card is unique. As a consequence, if dealt a king a player knows they have the strongest hand, and thus should never fold. Similarly, a player dealt a jack knows they cannot possibly win at a showdown. In optimal play, the first player loses at a rate of $1/18$ chips per hand.

We will also experiment with a generalization of Kuhn poker, one card poker, or N-card poker, for fixed $N \geq 3$ [37]. Here, the betting remains the same, but the deck now contains N distinct ranks. This allows us to freely scale the game.

Representing Kuhn poker as a matrix game, the first player has $3^3 = 27$ actions and the second player has $4^3 = 64$ actions. We will compute a strategy in Kuhn poker and use it to play 4-card poker. Four card Kuhn poker is of size 81 by 256.

To start, we generate 0.1-Nash equilibrium strategy for the first player using regret-matching in self-play in Kuhn poker. We then translate this strategy into 4-card poker by mapping the strategy for the middle two ranks to that of the queen in the Kuhn strategy. This is our ground truth behavior. Note that our original strategy is suboptimal in Kuhn poker due to stopping the equilibrium computation early. Additionally, the using abstraction further degrades the quality of the strategy. That is, our ground truth is not *rational*.

We compare our dual strategy estimation technique to the standard multinomial estimator as well as the restricted Nash estimator [48], both of which require primal observations. We will now briefly describe these baselines.

The primal multinomial estimator is simply an average of the primal observations mixed with the uniform distribution (to ensure full support, and finite loss).

$$\sigma_1(a) \propto \alpha + \frac{1}{T}\sum_{t=1}^{T} I(x^t = a)$$

This is equivalent to maximizing the likelihood of the primal observations subject to a Dirichlet prior. We choose the strength of the prior, $\alpha > 0$, using leave-one-out cross-validation.

The restricted Nash technique is technically not designed to estimate a player's behavior, but instead to robustly exploit it using primal observations [48]. It is operationally quite simple. Similar to computing an equilibrium, two no-regret learners play against one-and-other in self-play. There is one additional twist—the player we wish to exploit mixes in the observed behavior with probability ϵ. As with the multinomial, we choose ϵ by cross-validation.

This mixing has a two-fold effect. First, the no-regret learner corresponding to the exploitee learns a game-theoretic strategy that aims to mask or make up for the weaknesses in the observed behavior. Second, as the exploitee is able to somewhat adapt to the exploiter, the exploiter learns a robust counterstrategy. As the ϵ parameter varies from zero to one, the counterstrategy interpolates non-linearly between a best response to an equilibrium strategy.

Though not originally designed to be predictive, we use the strategy that the exploitee learns as a baseline to compare against our dual strategy estimation technique.

The performance of the behavior prediction algorithms is displayed in Figure 2.1. First, we note that our dual estimator, and the dual utility estimator do much better than the multinomial under low sample sizes. Furthermore, the performance of the dual utility estimator is almost invariant to sample size. We can conclude that either the utility estimates more rapidly converge to the true utilities, or that the induced behavior is stable with respect to noisy utility estimates.

Second, the multinomial and the restricted Nash estimators eventually overtake the dual utility estimator. This is to be expected, as they are consistent—they will ultimately converge to the true behavior. The dual estimators do not have this guarantee. This leads us to believe that it may be fruitful to consider primal-dual estimators, which somehow mix between the two estimators at a rate that depends on the number of observations.

In Figure 2.2, we show the error of the maximum entropy ε-Nash equilibrium. Note that the loss hits its minimum of 1.69 at $\varepsilon = 0.016$. This is much lower than the loss of the dual estimator, which hovers around 2.00. Upon inspection, it appears that in this case, shrinking the value estimate towards the game's value improves performance. This has the effect of placing a prior on the agents' performing rationally under lower sample sizes. This is similar in spirit to the idea of [72], where it is assumed the agents are completely rational.

In Figure 2.3 we show both an optimal strategy for the first player as well as the corresponding value for each rank in one card poker with a deck of thirteen cards. This figure is compelling for two reasons. First, note that the optimal strategy is not at all smooth as a function of rank. In particular, the first player bets with its low cards as bluffs, and its high cards for value. This illustrates our claim that strategic behavior is jagged and complicated. Second, the value of each card is a smooth function of rank. In this case it is fit nicely by a simple quadratic. Together, these imply that we can more easily estimate the utility of each rank than the optimal strategy

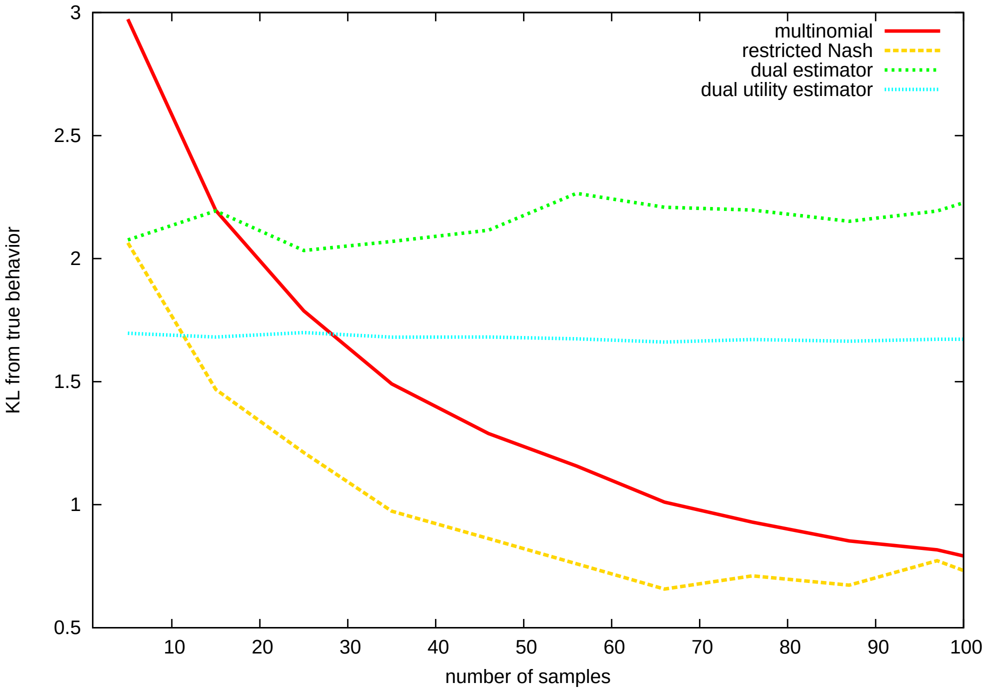

Figure 2.1: Kullback-Leibler divergence of behavior prediction algorithms in one card poker.

itself. For example, if we assume a quadratic form for the value function, we need only estimate three parameters, not thirteen which would be required to estimate each individually.

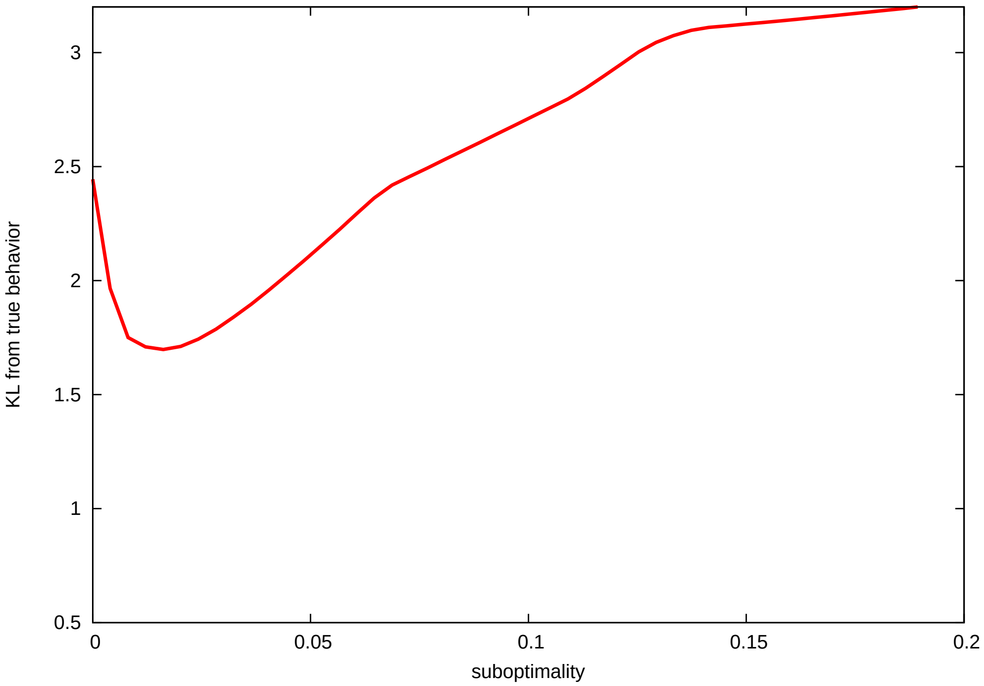

Figure 2.2: Kullback-Leibler divergence of maximum entropy ε-Nash equilibrium in one card poker.

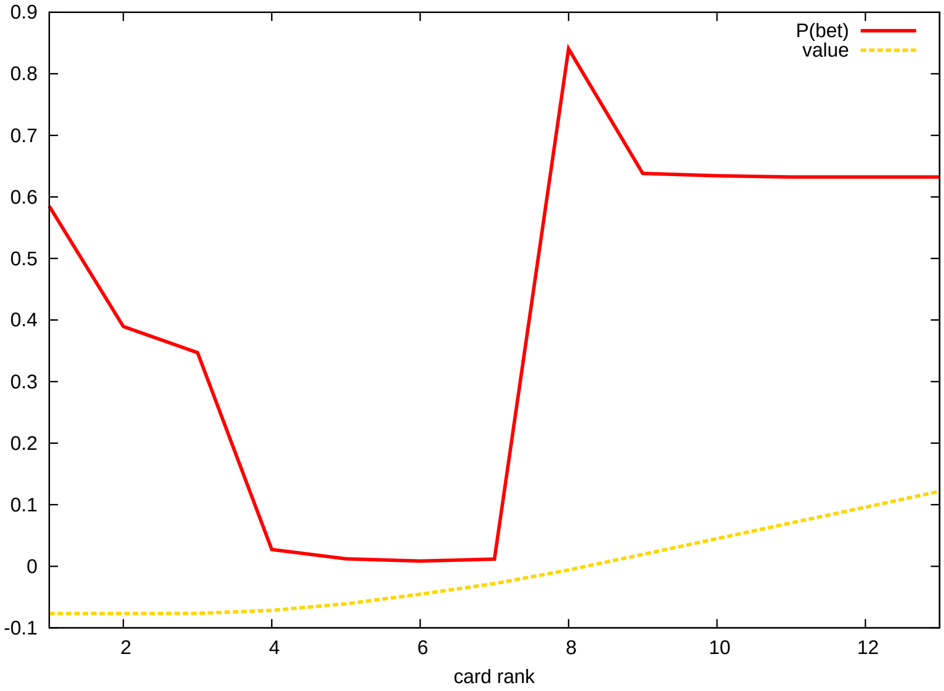

Figure 2.3: Optimal strategy and value of each rank in one card poker to the first player with a deck size of thirteen.

Chapter 3

Behavior Prediction with Unknown Utility

3.0.1 Rationality and the ICE Polytope

Let $\{a^{(t)}\}_{t=1}^{T}$ be a sequence of T independent observations of behavior in game Γ distributed according to σ, the players' **true behavior**. We call the empirical distribution of the observations, $\tilde{\sigma}$, the **demonstrated behavior**.

We aim to learn a distribution $\hat{\sigma}$, called the **predicted behavior**, an estimation of the true behavior from these demonstrations. Moreover, we would like our learning procedure to extract the motives for the behavior so that we may imitate the players in similarly structured, but unobserved games. Initially, let us consider just the estimation problem. While deriving our method, we will assume we have access to the players' true behavior. Afterwards, we will analyze the error introduced by approximating from the demonstrations.

Imitation appears hard barring further assumptions. In particular, if the agents are unmotivated or their intentions are not coerced by the observed game, there is little hope of recovering principled behavior in a new game. Thus, we require a form of rationality.

Proposition 1. *The players in a game are* ***rational*** *with if they prefer joint-strategy σ over joint strategy σ' when*

$$\mathrm{Regret}(\sigma, w^*) < \mathrm{Regret}(\sigma', w^*)$$

Our rationality assumption states that the players are driven to minimize their regret. It is not necessarily the case that they indeed have low or no regret, but simply that they can evaluate their preferences and that they prefer joint strategies with low regret. Through this assumption, we will be able to reason about the players' behavior solely through the game's features; this is what leads to the improved statistical properties of our approach.

As agents' true preferences w^* are unknown, we consider an encompassing assumption that requires that estimated behavior satisfy this property for all possible utility weights. A prediction $\hat{\sigma}$ is **strongly rational** if

$$\forall w \in \mathbb{R}^K, \;\; \mathrm{Regret}(\hat{\sigma}, w) \leq \mathrm{Regret}(\sigma, w).$$

Note that this does *not* assume that the players are completely rational, *i.e.*, they have no regret, with respect to all utility functions.

This assumption is similar in spirit to the utility matching assumption employed by single-agent inverse optimal control techniques. We also have a similar *if and only if* guarantee relating rationality and strong rationality [1, 97].

Theorem 3. *If a prediction $\hat{\sigma}$ is strongly rational and the players are rational, then they do not prefer σ over $\hat{\sigma}$.*

This is immediate as $w^* \in \mathbb{R}^K$.

Phrased another way, a strongly rational prediction is no worse than the true behavior.

Corollary 2. *If a prediction $\hat{\sigma}$ is strongly rational and the true behavior is an ε-correlated equilibrium utility function $w^* \in \mathbb{R}^K$, then $\hat{\sigma}$ is also an ε-equilibrium.*

Again, the proof is immediate as $\mathrm{Regret}(\hat{\sigma}, w^*) \leq \mathrm{Regret}(\sigma, w^*) \leq \varepsilon$.

Conversely, if we are uncertain about the true utility function we **must** assume strong rationality or we risk predicting less desirable behavior.

Theorem 4. *If a prediction $\hat{\sigma}$ is not strongly rational and the players are rational, then there exists a $w^* \in \mathbb{R}^K$ such that σ is preferred to $\hat{\sigma}$.*

The proof follows from the negation of the definition of strong rationality.

By restricting our attention to strongly rational behavior, at worst agents acting according to their unknown preferences will be indifferent between our predictive distribution and their behavior. That is, strong rationality is necessary and sufficient requirement on us, the observer, if the players are rational and we have no knowledge of their true utility function.

Note that requiring a strongly rational prediction is not as restrictive as it may initially seem. The players' true behavior is strongly rational. Thus, the set of strongly rational predictions is always non-empty. Furthermore, any guarantees we can make regarding the quality of a strongly rational prediction with respect to the entire set of strongly rational behavior will also hold with respect to the truth. That is, though the players themselves consider only a single utility function, we lose nothing by considering them all.

Unfortunately, a direct translation of the strong rationality requirement into constraints on the distribution $\hat{\sigma}$ leads to an intractable optimization problem as it has an infinite number of constraints that it involves products of utility vectors with the behavior to be estimated. Fortunately, we can provide an equivalent concise convex description of the constraints on $\hat{\sigma}$ that ensures any feasible distribution satisfies strong rationality. We denote this set of equivalent constraints as the *Inverse Correlated Equilibria* (ICE) polytope.

Definition 3 (Standard ICE Polytope)**.**

$$\begin{aligned} r^f(\hat{\sigma}) &= \sum_{g \in \Phi} \eta_g^f r^g(\sigma), && \forall f \in \Phi \\ \eta^f &\in \Delta_\Phi, && \forall f \in \Phi \\ \hat{\sigma} &\in \Delta_{\mathcal{A}}. \end{aligned}$$

The following corollary equates strong rationality and the standard ICE polytope.

Corollary 3. *A prediction $\hat{\sigma}$ is strongly rational if and only if for all $f \in \Phi$ there exists $\eta^f \in \Delta_\Phi$ such that $\hat{\sigma}$ and η satisfy the standard ICE polytope.*

We now show a more general result that implies Corollary 3. We start by generalizing the notion of strong rationality by restricting w^* to be in a known set $\mathcal{K} \subseteq \mathbb{R}^K$. We say a prediction

$\hat{\sigma}$ is $\mathcal{K}$-strongly rational if

$$\forall w \in \mathcal{K}, \ \ \mathrm{Regret}(\hat{\sigma}, w) \leq \mathrm{Regret}(\sigma, w).$$

If $\mathcal{K}$ is convex with non-empty relative interior and $0 \in \mathcal{K}$, we derive the $\mathcal{K}$-ICE polytope.
Definition 4 ($\mathcal{K}$-ICE Polytope)**.**

$$\begin{aligned} r^f(\hat{\sigma}) - \sum_{g \in \Phi} \eta^f_g r^g(\sigma) &\in -\mathcal{K}^* && \forall f \in \Phi \\ \eta^f &\in \Delta_\Phi, && \forall f \in \Phi \\ \hat{\sigma} &\in \Delta_\mathcal{A}. \end{aligned}$$

Note that the above constraints are linear in $\hat{\sigma}$ and η, and $\mathcal{K}^*$, the dual cone, is convex. The following theorem shows the $\mathcal{K}$-ICE polytope coincides with $\mathcal{K}$-strongly rational joint-strategies.
Theorem 5. *A prediction $\hat{\sigma}$ is $\mathcal{K}$-strongly rational if and only if for all $f \in \Phi$ there exists $\eta^f \in \Delta_\Phi$ such that $\hat{\sigma}$ and η satisfy the $\mathcal{K}$-ICE polytope.*

The proof is provided in Appendix B.

By choosing $\mathcal{K} = \mathbb{R}^K$, then $\mathcal{K}^* = \{0\}$ and the polytope reduces to the standard ICE polytope. Thus, Corollary 3 follows directly from Theorem 5. By choosing $\mathcal{K}$ to be the positive orthant, $\mathcal{K} = \mathcal{K}^* = \mathbb{R}^K_+$, the polytope reduces to the following inequalities. Here, we explicitly assume the utility is positively correlated with the features.
Definition 5 (Positive ICE Polytope)**.**

$$\begin{aligned} r^f(\hat{\sigma}) &\leq \sum_{g \in \Phi} \eta^f_g r^g(\sigma) && \forall f \in \Phi \\ \eta^f &\in \Delta_\Phi, && \forall f \in \Phi \\ \hat{\sigma} &\in \Delta_\mathcal{A}. \end{aligned}$$

Predictive behavior within the ICE polytope will retain the quality of the demonstrations provided. The following corollaries formalize this guarantee.
Corollary 4. *If the true behavior is an ε-correlated equilibrium under w^*, then a prediction $\hat{\sigma}$ that satisfies the standard ICE polytope is an $A\varepsilon$-correlated equilibrium.*

This follows immediately from Lemma 2.

Geometeric Interpretation of the Standard ICE Polytope

Next, we take a brief aside to understand the geometry of the Standard ICE Polytope. The Standard ICE Poltype is made up of a set of constraints of the form,

$$\begin{aligned} r^f(\hat{\sigma}) &= \sum_{g \in \Phi} \eta^f_g r^g(\sigma), \\ \eta^f &\in \Delta_\Phi, \\ \hat{\sigma} &\in \Delta_\mathcal{A}. \end{aligned}$$

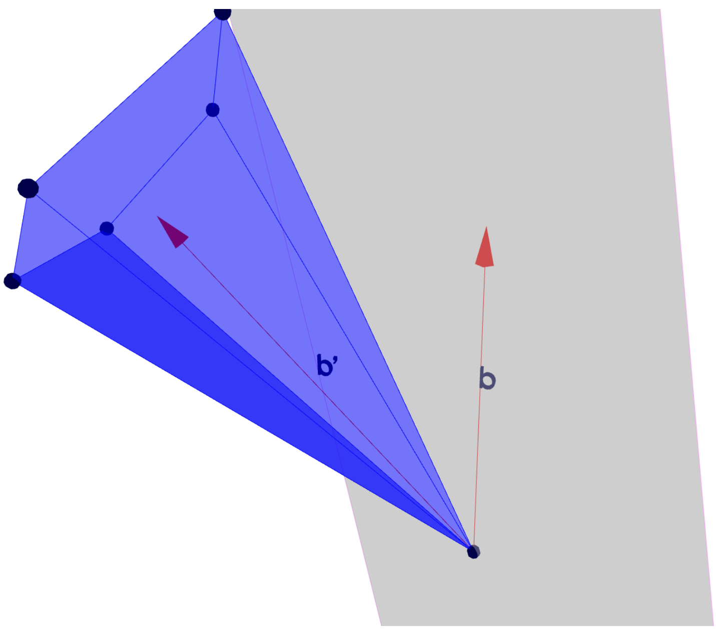

Figure 3.1: A Geometric Visualization of the ICE Standard Polytope.

one for each deviation $f \in \Phi$.

Consider the following linear program that aims to determine if b can be written as a convex combination of the columns of A:

$$\min_{x \in \Delta} \; 0, \text{ subject to: } Ax = b$$

Its dual linear program is

$$\max_{w,\delta} \; \delta - b \cdot w, \text{ subject to: } A^T w \geq \delta e$$

and it can written as the unconstrained convex program

$$\max_{w} \; \min A^T w - b \cdot w$$

By duality and construction, if the primal program is feasible then for any w

$$\min A^T w - b \cdot w \leq 0$$

as the dual's objective value can never exceed the primal's objective value.

Also due to duality and the fact that the dual program is feasible, if the primal program is infeasible then there must exist a w such that

$$A^T w - b \cdot w > 0$$

Said in other words, we have proven a theorem of alternatives: either b can be written as a convex combination of the columns of A, or there exists a hyperplane with normal $w \neq 0$ that separates the columns of A from the vector b. This is shown visually in Figure 3.1. In Figure 3.1, the columns of A are the vertices of the pointed polytope. Vector b is separated from the polytope by the grey hyperplane, but the vector b' is contained in the cone so it cannot be separated.

In the case of the Standard ICE Polytope, the columns of the matrix A correspond to the $r^g(\sigma)$–the demonstrated regret feature vectors for each deviation. The vector b corresponds to a particular $r^f(\hat{\sigma})$, the regret feature vector for the predicted behavior under some deviation $f \in \Phi$. The constraints require that all the predicted regret feature vectors, one for each deviation $f \in \Phi$, must be contained within the demonstrated regret vector polytope.

3.0.2 The Principle of Maximum Entropy

As we are interested in the problem of statistical prediction of strategic behavior, we must find a mechanism to resolve the ambiguity remaining after accounting for the rationality constraints. The **principle of maximum entropy**, due to Jaynes [45], provides a well-justified method for choosing such a distribution. This choice leads to not only statistical guarantees on the resulting predictions, but to efficient optimization.

The **Shannon entropy** of a joint-strategy σ is

$$H(\sigma) = \mathbb{R}_{a \sim \sigma}\left[-\log \sigma(a)\right],$$

and the principle of maximum entropy advocates choosing the distribution with maximum entropy subject to known constraints [45]. That is,

$$\begin{aligned} \sigma_{\text{MaxEnt}} = \operatorname*{argmax}_{\sigma \in \Delta_{\mathcal{A}}} H(\sigma), \quad \text{subject to:}& \\ \sigma \in \mathcal{X}.& \end{aligned}$$

The constraint set $\mathcal{X}$ is typically chosen to capture the important or most salient characteristics of the distribution. When it is convex, finding this distribution is an easy optimization problem. The resulting log-linear family of distributions (*e.g.*, logistic regression, Markov random fields, conditional random fields) are widely used within statistical machine learning.

In the context of multi-agent behavior, the principle of maximum entropy has been employed to obtain correlated equilibria with predictive guarantees in normal-form games when the utilities are known *a priori* [72]. We will now leverage its power with our rationality assumption to select predictive distributions in games where the utilities are unknown, but the important features that define them are available.

For our problem, the constraints are precisely that the distribution is in the ICE polytope, ensuring that whatever we predict has no more regret than the demonstrated behavior.

Definition 6. *The primal maximum entropy ICE optimization problem is*

$$\begin{aligned} &\max_{\hat{\sigma},\eta} \; H(\hat{\sigma}) \quad \textit{subject to:} && \\ r^f(\hat{\sigma}) - \sum_{g \in \Phi} \eta^f_g r^g(\sigma) &\in -\mathcal{K}^* && \forall f \in \Phi \\ \eta^f &\in \Delta_\Phi, && \forall f \in \Phi \\ \hat{\sigma} &\in \Delta_{\mathcal{A}}. && \end{aligned}$$

This program is convex, feasible, and bounded. That is, it has a solution and is efficiently solvable using simple techniques in this form.

Importantly, the maximum entropy prediction enjoys the following guarantee:

Algorithm 1 Dual MaxEnt ICE Subgradient

```
**Input:** Let $\hat{\sigma}$ be the prediction given the current dual weights, $\lambda$, as from Equation (3.1).
for $f \in \Phi$ do
    $g^{\max} \leftarrow \operatorname{argmax}_{g \in \Phi} \left\langle r^g(\sigma), \lambda^f \right\rangle$
    $\frac{\partial}{\partial \lambda^f} L(\lambda) \leftarrow r^{g^{\max}}(\sigma) - r^f(\hat{\sigma})$
end for
return $\frac{\partial}{\partial \lambda} L(\lambda)$
```

Lemma 3. *The maximum entropy ICE distribution minimizes over all strongly rational distributions the worst-case log-loss, $\mathbb{E}_{a \sim \sigma}\left[-\log_2 \hat{\sigma}(a)\right]$, when σ is chosen adversarially but subject to strong rationality.*

The proof of Lemma 3 follows immediately from the result of Grünwald and Dawid [38].

3.0.3 Dual Optimization

In this section, we will derive and describe a procedure for optimizing the dual program for solving the MaxEnt ICE optimization problem. We will see that the dual multipliers can be interpreted as utility vectors and that optimization in the dual has computational advantages. We begin by presenting the dual program.

Theorem 6. *The dual maximum entropy ICE optimization problem is the following non-smooth, but convex program:*

$$\min_{\lambda^f \in \mathcal{K}} \; L(\lambda) \doteq \sum_{f \in \Phi} \operatorname{Regret}(\sigma, \lambda^f) + \log Z(\lambda), \textit{ where}$$

$$Z(\lambda) = \sum_{a \in \mathcal{A}} \exp\left(-\sum_{f \in \Phi} \left\langle r^f(a), \lambda^f \right\rangle\right).$$

We derive the dual in Appendix B

As the dual's feasible set has non-empty relative interior, strong duality holds by Slater's condition—there is no duality gap. We can also use a dual solution to recover $\hat{\sigma}$.

Lemma 4. *Strong duality holds for the maximum entropy ICE optimization problem and given optimal dual weights λ^*, the maximum entropy ICE joint-strategy $\hat{\sigma}$ is*

$$\hat{\sigma}(a) \propto \exp\left(-\sum_{f \in \Phi} \left\langle r^f(a), \lambda^{*,f} \right\rangle\right) \tag{3.1}$$

The dual formulation of our program has important inherent computational advantages. First, so long as $\mathcal{K}$ is simple, the optimization is particularly well-suited for gradient-based optimization, a trait not shared by the primal program. Second, the number of dual variables, $|\Phi|K$, is typically much fewer than the number of primal variables, $|\mathcal{A}| + |\Phi|^2$. Though the work per iteration is still a function of $|\mathcal{A}|$ (to compute the partition function), these two advantages together let us scale to larger problems than if we consider optimizing the primal objective. Computing the expectations necessary to descend the dual gradient can leverage recent advances in the structured, compact game representations: in particular, any graphical game with low-treewidth or finite horizon Markov game [51, 52] enables these computations to be performed in time that scales only polynomially in the number of decision makers.

Algorithm 1 describes the dual subgradient computation. This can be incorporated with any non-smooth gradient method, such as the projected subgradient method [81], to approach the optimal dual weights.

3.1 Behavior Estimation in Parameterized Matrix Games

To account for stochastic, or varying environments, we now consider *distributions over* games. For example, rain may affect travel time, make certain modes of transportation less desirable, or even unavailable. Operationally, nature samples a game prior to play from a distribution known to the players. The players then as a group determine a joint strategy conditioned on the particular game and an outcome is drawn by a coordination device. We let $\mathcal{G}$ denote our class of games and ξ be chance's distribution over $\mathcal{G}$.

As before, we observe a sequence of T independent observations of play, but now in addition to an outcome we also observe nature's choice at each time t. Let $\{(\Gamma^{(t)}, a^{(t)})\}_{t=1}^{T}$ be the sequence of observations drawn from ξ and σ, the **true behavior**. The empirical distribution of the observations, $\tilde{\xi}$ and $\tilde{\sigma}$, together are the **demonstrated behavior**.

Now we aim to learn a **predictive behavior** distribution, $\hat{\sigma}$, for *any* $\Gamma \in \mathcal{G}$, even ones we have not yet observed. Clearly, we must leverage the observations across the entire family to achieve good predictive accuracy. We continue to assume that the players' utility is an unknown linear function, w^*, of the games' features and that this function is fixed across $\mathcal{G}$. Next, we amend our notion of regret and our rationality assumption.

3.1.1 Behavior Estimation through Conditional ICE

Ultimately, we wish to simply employ an additional expectation over the game distribution when reasoning about the regret and regret features. To do this, our notion of a deviation needs to account for the fact that it may be executed in games with different structures. Operationally, one way to achieve this is by having a deviation not modify the player's action in games with a different structure, which increases the size of Φ by a factor of $|\mathcal{G}|$. If the actions, and in turn the deviations, have similar semantic meanings across our entire family of games, one can simply share the deviations across all games. This allows for one to achieve transfer over an infinitely large class. Given such a decision, we write the expected regret features under deviation f as

$$r^f(\sigma) = \mathbb{E}_{\Gamma \sim \xi}\left[\mathbb{E}_{a \sim \sigma^\Gamma}\left[r^f(a)\right]\right],$$

and the expected regret under utility function w as

$$\langle r^f(\sigma), w\rangle = \mathbb{E}_{\Gamma \sim \xi}\left[\mathbb{E}_{a \sim \sigma^\Gamma}\left[\langle r^f(a), w\rangle\right]\right],$$

Again, we quantify the stability of a set of joint strategies using this new notion of expected regret with respect to the deviation set Φ,

$$\operatorname{Regret}(\sigma, w) = \max_{f \in \Phi} \mathbb{E}_{\Gamma \sim \xi}\left[\langle r^f(\sigma), w \rangle\right],$$

which, in turn, entails a notion of an ε-equilibrium for a set of joint strategies, a modified rationality assumption, and a slight modification to the $\mathcal{K}$-ICE polytope,

Definition 7 (Conditional $\mathcal{K}$-ICE Polytope)**.**

$$\begin{aligned} \mathbb{E}_{\Gamma \sim \xi}\left[r^f(\hat{\sigma}^\Gamma) - \sum_{g \in \Phi} \eta_g^f r^g(\sigma^\Gamma)\right] &\in -\mathcal{K}^* & \forall f \in \Phi \\ \eta^f &\in \Delta_\Phi, & \forall f \in \Phi \\ \hat{\sigma}^\Gamma &\in \Delta_{\mathcal{A}}. & \forall \Gamma \in \mathcal{G} \end{aligned}$$

All that remains is to adjust our notion of entropy to take into account a distribution over games. In particular, we choose to maximize the expected entropy of our prediction, which is conditioned on the game sampled by chance.

Definition 8. *The conditional Shannon entropy of a set of strategies σ when games are distributed according to ξ is*

$$H(\sigma) = \mathbb{E}_{\Gamma \sim \xi}\left[H(\sigma^\Gamma)\right].$$

The modified dual optimization problem has a familiar form. We now use the new notion of regret and take the expected value of the log partition function.

Theorem 7. *The dual conditional maximum entropy ICE optimization problem is*

$$\min_{\lambda^f \in \mathcal{K}} \sum_{f \in \Phi} \operatorname{Regret}(\sigma, \lambda^f) + \mathbb{E}_{\Gamma \sim \xi}\left[Z^\Gamma(\lambda)\right].$$

The proof of this is a straightforward extension of Theorem 5 and Theorem 6. To recover the predicted behavior, we use the same exponential family form as before.

As with any machine learning technique, it is advisable to employ some form of complexity control on the resulting predictor to prevent over-fitting. As we now wish to generalize to unobserved games, we too should take the appropriate precautions. In our experiments, we employ $L1$ and $L2$ regularization terms to the dual objective for this purpose. Regularization of the dual weights effectively alters the primal constraints by allowing them to hold approximately, leading to higher entropy solutions [24].

3.2 Sample Complexity

In practice, we do not have full access to the agents' true behavior—if we did, prediction would be straightforward and we would not require our estimation technique. Instead, we may only approximate the desired expectations by averaging over a finite number of observations,

$$\langle r^f(\sigma), w \rangle \approx \frac{1}{T} \sum_{t=1}^{T} \langle r^f(a^{(t)}), w \rangle$$

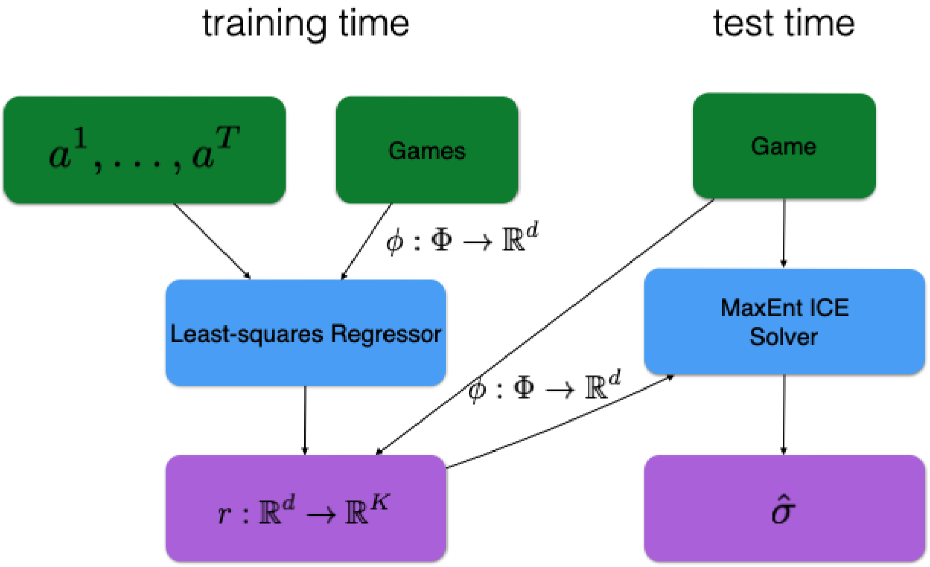

Figure 3.2: Visual interpretation of the Regression ICE algorithm.

In real applications there are costs associated with gathering these observations and, thus, there are inherent limitations on the quality of this approximation. Next, we will analyze the sensitivity of our approach to these types of errors.

First, although $|\mathcal{A}|$ is exponential in the number of players, our technique only accesses σ through expected regret features of the form $r^f(\sigma)$. That is, we need only approximate these features accurately, not the distribution σ. For finite-dimensional vector spaces, we can bound how well the regrets match in terms of $|\Phi|$ and the dimension of the space.

Theorem 8. *With probability at least $1-\delta$, for any w, by observing $T \geq \frac{1}{2\epsilon^2} \log \frac{2|\Phi|K}{\delta}$ outcomes we have for all deviations $\langle r^f(\tilde{\sigma}), w \rangle \leq \langle r^f(\sigma), w \rangle + \epsilon \Delta \|w\|_1$.*

Where Δ is the maximum possible regret over all basis directions. The proof is an application of the union bound and Hoeffding's inequality much like the proof of Theorem 1.

Alternatively, we can bound how well the regrets match independently of the space's dimension by considering each utility function separately.

Theorem 9. *With probability at least $1-\delta$, for any w, by observing $T \geq \frac{1}{2\epsilon^2} \log \frac{|\Phi|}{\delta}$ outcomes we have for all deviations $\langle r^f(\tilde{\sigma}), w \rangle \leq \langle r^f(\sigma), w \rangle + \epsilon \Delta(w)$.*

Where $\Delta(w)$ is the maximum possible regret under w.

Both of the above bounds imply that, so long as the true utility function is not too complex, with high probability we need only logarithmic many samples in terms of $|\Phi|$ and K to closely approximate the regrets and avoid a large violation of our rationality condition. That is, when using internal regret the sample complexity is logarithmic in terms of all quantities of interest, the number of actions, players and features.

Theorem 10. *If for all $f \in \Phi$, $\langle r^f(\tilde{\sigma}), w \rangle \leq \langle r^f(\sigma), w \rangle + \gamma$, then* $\text{Regret}(\tilde{\sigma}, w) \leq \text{Regret}(\sigma, w) + \gamma$.

Proof. For all deviations, $f \in \Phi$, $\langle r^f(\tilde{\sigma}), w \rangle \leq \langle r^f(\sigma), w \rangle + \gamma \leq \text{Regret}(\sigma, w) + \gamma$. In particular, this holds for the deviation that maximizes the demonstrated regret. □

3.3 Experimental Results

3.3.1 Human Laboratory Studies

The games we consider are taken from a number of psychological studies designed to study various aspects of strategic human decision making [18, 19, 36, 41, 42, 77, 86, 87, 88]. Study participants were shown a matrix of utilities defining a two-player game and asked to choose an action for the first player. They were not shown the outcome, *i.e.*, the opponent's choice of action, or their reward—they received no feedback during the study. Upon completion, participants were rewarded with a monetary sum proportional to their overall performance against the other participants. These studies were aggregated and examined previously by Wright and Leyton-Brown [96]. In total, there are 123 games of size 3×3 to 5×5, and $11,771$ observations of play. Here, the observations are actions taken by a single player, not joint-actions reached by a pair of players. Now we will briefly discuss the design and purpose of each study.

Cooper and Van Huyck [18] examine human behavior in 2×2 extensive-form games. The games are all structured such that the first player chooses either left or right. The second player acts only if the first player takes the left action. All games of this form are isomorphic to one of thirty six canonical games. Half of these canonical games result in a dominant strategy for the first player, while in the other half the first player must deduce how the second will act to choose the best action. Of these thirty six, eight games were selected for 187 student participants from Texas A&M University to play in a laboratory setting. Of these eight, six required the players to speculate about their opponent's decision. The games were presented to some students in extensive-form and to others in normal-form. Though the behavior differed based on the representation, surprisingly the difference could not be explained by the ability to do backwards induction, *i.e.*, the ability to reason about how the second player would act. Instead the authors conclude that when presented the extensive-form the participants were more likely to choose the action that lead to the second player's decision.

Costa-Gomes et al. [19] study the strategic sophistication of human decision-making. That is, to what extent does a player reason about the other's behavior and how does this reasoning effect their decision. The authors posit that the player population is composed of a number of types of players and aim to infer the proportions of these types from the observations of play. Some examples of types include: a uniform random player, also known as a level zero player; an altruistic player, who defaults to behavior that benefits the other; a pessimistic; a naive player or a level one player, who best responds to a uniform player; a level two player, who best responds to a naive player; and so on. The study examined eighteen games of size 2×2 to 4×4 played by 144 students from the University of Arizona. The authors concluded that level one and level two players accounted for between 67% and 89% of the population.

Stahl and Haruvy [86], Stahl and Wilson [87], Stahl and Wilson [88], Haruvy and Stahl [41], and Haruvy et al. [42] all originated from the University of Texas and followed very similar experimental procedures. Each examined human behavior in 3×3 symmetric games. All together, 107 games were considered and data was collected from 440 undergraduate participants. In one study, the authors conclude that humans are good at avoiding dominated strategies, but are skeptical regarding others' ability to do the same. For example, when presented with a coordination game where it is clear how both players should cooperate to achieve the maximum reward, many players will pay a small cost to opt out of playing rather than trust the other player will deduce the mutually beneficial action. Haruvy et al. [42] examine diversity within player types. They posit these intra-type deviations are due to execution errors on the part of the player. These errors are modelled with a logit model closely related to the quantal response model, which we will discuss in the future.

Goeree and Holt [36] consider ten coordination games that each show in a particular way that the Nash equilibrium solution concept fails to accurately predict human behavior. For example, Kreps' game, which exhibits two pure strategy Nash equilibria, is examined in the study. To avoid the equilibrium selection problem in this game, humans tend to prefer a slightly suboptimal alternative action. Another game includes a non-Nash action that dampens the games utility. Humans, being risk averse, often select this action despite it never arising in optimal play. This study collected play from 50 participants from the University of Virginia.

Rogers et al. [77] is the final study in this collection. Overall, it includes 17 games played by 74 participants. The authors assume that play follows a quantal response equilibrium, an model that explicitly incorporates execution error to account for bounded player rationality. Each player has an unknown type, which represents the magnitude of this error. The study aims to determine the type distribution of the player population. This study is very closely related to those out of the University of Texas and University of Arizona that aim to model the population with heterogeneous types of players. There, the types come from a discrete set and are related to the levels in the cognitive hierarchy. Here, the execution error is a continuous variable.

Fictitious Play

Fictitious play is a multi-agent learning algorithm introduced by Brown [7]. It is conceptually simple and operationally it mimics how humans often reason in multi-agent scenarios.

The algorithm proceeds iteratively. Throughout, the players maintain a model of how the others will act. On each round, the players best respond to the current behavior. To incorperate the responses into the model, we simply averaging them in.

Formally, we let σ_i^t denote player i's response at time t and let $\bar{\sigma}_i^T$ be the average of i's responses up to and including T. That is,

$$\sigma_i^t \in \operatorname*{argmax}_{\sigma_i \in \Delta_{A_i}} u_i(\sigma_i, \bar{\sigma}_{-i}^{t-1}), \text{ and}$$

$$\bar{\sigma}_i^T = \frac{1}{T} \sum_{t=1}^{T} \sigma_i^t.$$

Occationally, we choose to weight the rounds differently, *i.e.*, we define the weighted average strategy

$$\bar{\sigma}_i^T = \sum_{t=1}^{T} \alpha^t \sigma_i^t / \sum_{t=1}^{T} \alpha^t$$

If fictitious play converges, then it will converge to a Nash equilibrium. In general, though, it is not guaranteed to converge. There are some criteria under which we can guarantee convergence, *e.g.*, if a game is two-player and zero-sum, or if it is solvable by iteratively removing dominated strategies.

Fictitious play and its variants form the basis of many behavior prediction schemes. In particular, it affords the inductive notion of a "level". We say a player is on level zero if they play uniformly. We say a player is on level $k+1$ if they best respond to players level k and below. That is, a level k player plays the strategy profile σ^k.

It is thought that humans do not often reason beyond level two.

Wright and Leyton-Brown use a sophistocated local search procedure to learn a model of human population's level distribution. This model is then combined with fictitious play to compute predictive strategies in small matrix games.

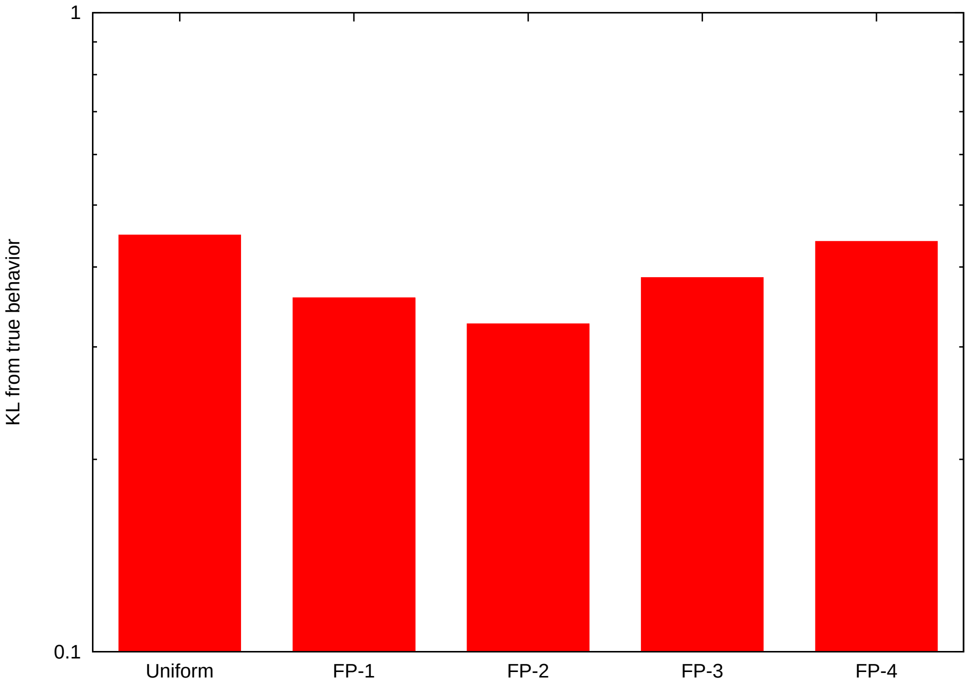

Figure 3.3: Fictitious Play KL divergence from true behavior varying number of iterations.

Iterative Sampling and Reweighting

Sample weighted averaging (SAW) and inertia sample weighted averaging (I-SAW) are decision-theoretic tweaks to fictitious play.

The SAW algorithm explores, *i.e.*, chooses a uniform action, with probability p. The remainder of the time it proceeds like fictitious play—choosing the most profitable past action. The difference is in how SAW estimates the value of the actions. Fictitious play simply estimates these values with the mean utility observed so far. In SAW, we also keep a seperate mean to estimate the value of each action over a short history, say the last three iterations. This history component results in behavior switching when the recent past is *surprising* or different than the overall play.

The I-SAW algorithm adds one further mode to SAW that captures the notion of interia. That is, players are more likely to choose the same action given little changes between iterations. In particular, the algorithm defines a surprise factor by summing the difference between the action's value with its expected value. When this factor is low, the algorithm is likely to repeat its last choice.

Quantal Response

The quantal response equilibrium (QRE) is a generalization of the Nash equilibrium introduced by McKelvey and Palfrey [64]. It allows the players to err, with more costly mistakes being less likely than minor ones.

The logit QRE is its most common instantiation. It introduces an additional parameter, $\lambda \in \mathbb{R}_+$, that controls an additional degree of irrationality by the players. Mathematically, a strategy profile is a λ-QRE if

$$\sigma_i^t(x) \propto \exp\left(\lambda u_i(x, \bar{\sigma}_{-i}^{t-1})\right).$$

That is, each strategy is a *soft best response* to the others, as opposed to a Nash equilibrium where each strategy must be a best response. As $\lambda \to \infty$ the λ-QRE approaches a unique Nash equilibrium. A λ-QRE always exists, and is unique. That is, it does not suffer from the equilibrium selection problem.

Finding a quantal response equilibrium is computationally hard and there are no known dynamics to reach one. This is not surprising as it well approximates a Nash equilibrium for high values of λ. Typically, one uses a homotopy method to trace the path of QREs. That is, starting with $\lambda = 0$, the uniform strategy profile is the unique QRE. One then slightly increases λ and uses Newton's root finding method to correct the strategy profile. We must take care not to increase λ too much to ensure the previous strategy profile is a good starting point for Newton's method.

We can use the quantal response equilibrium itself to predict behavior in small matrix games, or we can use quantal fictitious play.

3.3.2 Market Entry Game

We next evaluate our approach against a number of baselines on data gathered for the Market Entry Prediction Competition [27]. The game has four players and is repeated for fifty trials and is meant to simulate a firm's decision to enter into a market. On each round, all four players simultaneous decide whether or not to open a business. All players who enter the market receive

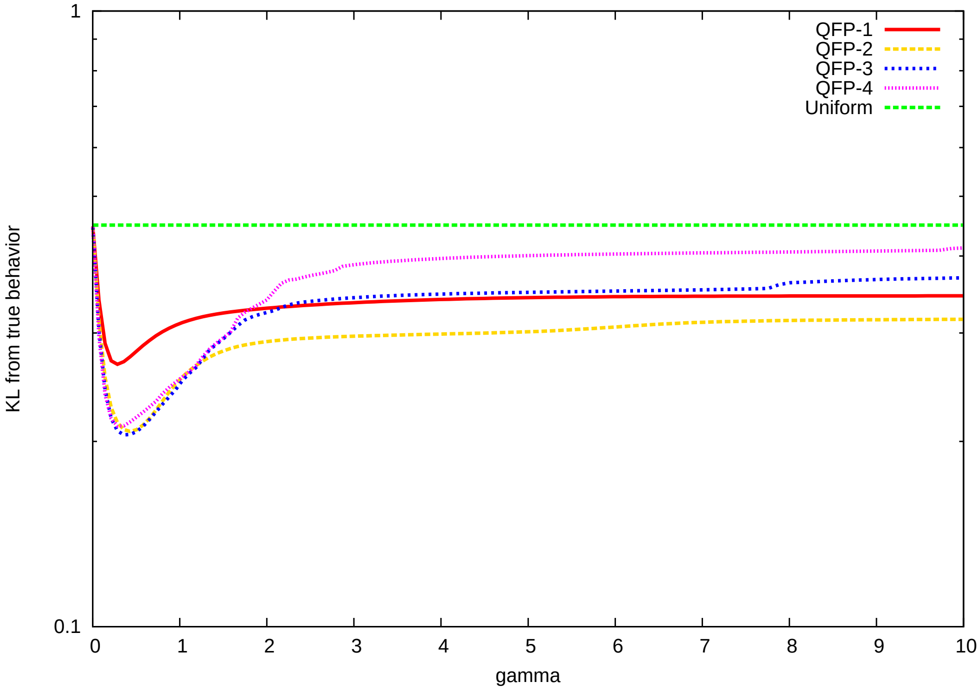

Figure 3.4: γ-Quantal Fictitious Play KL divergence from true behavior varying γ.

a stochastic payoff centered at $10 - kE$, where k is a fixed parameter unknown to the players and E is the number of players who entered. Players who do not enter the market receive a stochastic payoff with zero mean. After each round, each player is shown its reward, as well as the reward it would have received by choosing the alternative.

Observations of human play were gathered by the CLER lab at Harvard [27]. Each student involved in the experiment played ten games lasting fifty rounds each. The students were incentivized to play well through a monetary reward proportional to their cumulative utility. The parameter k was randomly selected in a fashion so that the Nash equilibrium had an entry rate of 50% in expectation. In total, $30,000$ observations of play were recorded. The intent of the competition was to have teams submit programs that would play in a similar fashion to the human subjects. That is, the data was used at test time to validate performance. In contrast, our experiments use actual observations of play at training time to build a predictive model of the human behavior. As we are interested in stationary behavior, we train and test on only the last twenty five trials of each game.

We compared against two baselines. The first baseline, labeled *Multinomial* in the figures, is a smoothed multinomial distribution trained to minimize the leave-one-out cross validation loss. This baseline does not make use of any features of the games. That is, if the players indeed play according to the Nash equilibrium we would expect this baseline to learn the uniform distribution.

The second baseline, labeled *Logistic Regression* in the figures, simply uses regularized logistic regression to learn a linear classification boundary over the outcomes of the game using the same features presented to our method. Operationally, this is equivalent to using MaxEnt Inverse Optimal Control in a single-agent setting where the utility is summed across all the players. This baseline has similar representational power to our method, but lacks an understanding of the strategic elements of the game.

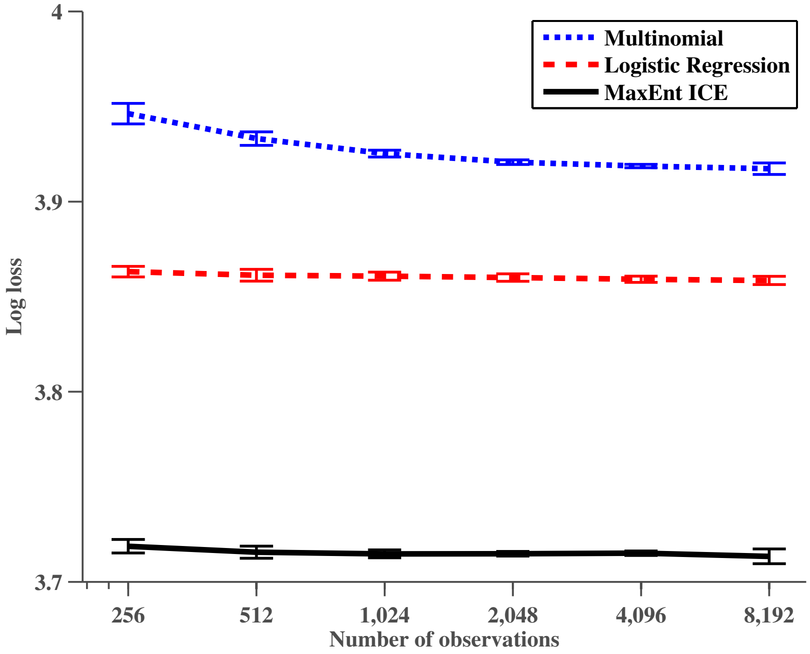

Figure 3.5: Test log loss using only the game's expected utility as a feature in the market entry experiment.

In Figure 3.5, we see a comparison of our method against the baselines when only the game's true expected utility is used as the only feature. We see that our method outperforms both baselines across all sample sizes. We also observe the multinomial distribution performs slightly better than the uniform distribution, which attains a log loss of 4, though substantially worse than logistic regression and our method, indicating that the human players are not particularly well-modeled by the Nash equilibrium. Our method substantially outperforms logistic regression, indicating that there is indeed a strategic interaction that is not captured in the utility features alone.

In Figure 3.6, we see a comparison of our method against the baselines using a variety of predictive features. In particular, we summarize a round using the observed action frequencies, average reward, and reward variance up to that point in the round. To weigh recent observations more strongly, we also employ exponentially-weighted averages. We observe that the use of these features substantially improves the predictive power of the feature-based methods. Interestingly, we also note that the addition of these summary features also narrows the gap between logistic regression and MaxEnt ICE. Under low sample sizes, the logistic model performs the best, but our method overtakes it as more data is made available for training. It appears that in

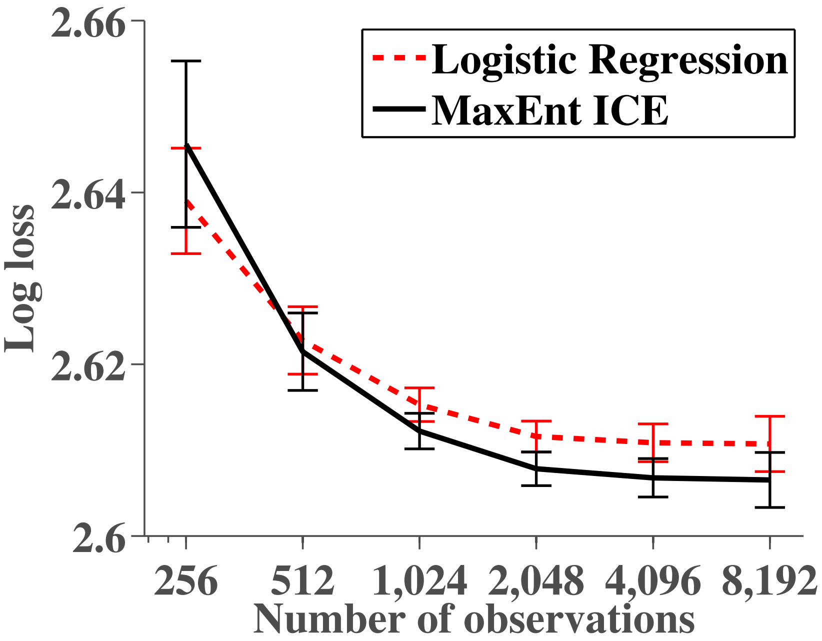

Figure 3.6: Test log loss using a number of history summary features in the market entry experiment.

this scenario, much of the strategic behavior demonstrated by the participants can be captured by these history features.

3.3.3 Mid-scale Hotel Market Entry

For our final experimental evaluation, we considered the task of predicting the behavior of mid-scale hotel chains, like Holiday Inn and Ramada, in the state of Texas. Given demographic and regulatory features of a county, we wish to predict if each chain is likely to open a new hotel or to close an existing one. The observations of play are derived from quarterly tax records over a fifteen year period from forty counties, amounting to a total of $2,205$ observations. The particular counties selected had records of all of the demographic and regulatory features, had at least four action observations, and none was a chain's flagship county. Figure 3.7 highlights the counties and visualizes their regulatory practices.

The demographic and regulatory features were aggregated from various sources and generously provided to us by Prof. Junichi Suzuki (2010). The demographic features for each county include quantities such as size of its population and its area, employment and household income statistics, as well as the presence or absence of an interstate, airport or national park. The regulatory features are indices measuring quantities such as commercial land zoning restrictions, tax rates and building costs. In addition to these noted features, which are fixed across all time periods, there are time-varying features such as the number of hotels and rooms for each chain and the aggregate quarterly income.

We model each quarterly decision as a parameterized simultaneous-move game with six players. Each player, a mid-scale hotel chain, has the action set $\{\text{Close}, \text{NoAction}, \text{Open}\}$, resulting

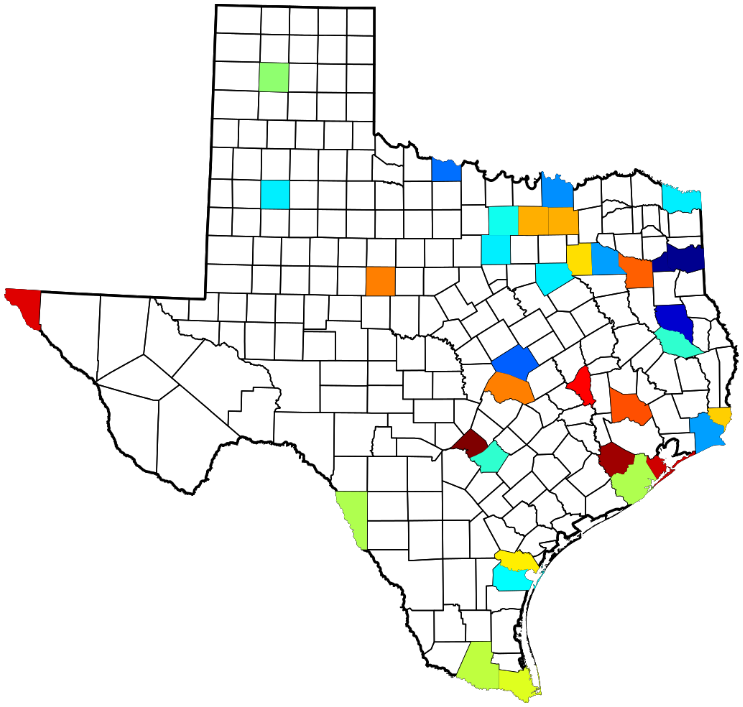

Figure 3.7: Regulatory index values for select counties in Texas. Blue means little regulation and lower costs to enter the market. Red means higher costs.

in 729 total outcomes. For the game's utility, we allocated the county's features to each player in proportion to how many hotels they owned. That is, if a player operated 3 out of 10 hotels, the features associated with utility at that outcome would be the county's feature vector scaled by 0.3. We included bias features associated with each action to account for fixed costs associated with opening or closing a hotel.

In the observation data, there are a small number of instances where a chain opens or closes more than one hotel during a quarter. These events are mapped to Open and Close respectively. Though the outcome-space is quite large, the outcome distribution is extremely biased and the actions of the chains are highly correlated. In particular, over 80% of time the time no action is taken, around 17% of the time a single chain acts, and less than 3% of the time more than one chain acts. As a result, one expects the featureless multinomial estimator to have reasonable performance despite a large number of classes.

For experimentation, we evaluated four algorithms: a smoothed multinomial distribution trained to minimize the leave one out cross-validation loss, MaxEnt inverse optimal control trained once for all players, multi-class logistic regression over the joint action space, and regret-matching ICE with utility matching constraints. As the resulting optimizations for the latter two algorithms are smooth, we employed the L-BFGS quasi-Newton method with L2-regularization for training [71]. As a substitute for L1-regularization, we selected the 23, out of the 63, best features based on their reduction in training error when using logistic regression. Of the top 23 features selected, 11 were regulatory indices.

For the logistic regression and ICE predictors, we only used utility features on the 13 high probability outcomes (no firms build, and one firm acting). The remaining outcomes had only bias features associated with them to help prevent overfitting. We experimented with a number of types of bias features, for example, 4 bias features (one for no firms build, one for a single firm builds, one for a single firm closes and one for all remaining outcomes), as well as 729 bias features (one for each outcome). We found that, though on their own the different bias features had varied predictive performance, when combined with utility and regret features they were quite similar given the appropriate regularization. In the best performing model, which we present here, we used 729 bias features resulting in $1,028$ parameters to the logistic regression model.

In the ICE predictor, we tied together the weights for each deviation across all the players to reduce the number of model parameters. For example, all players shared the same dual parameters for the NoAction $\rightarrow$ Open deviation. Effectively, this alters the rationality assumption such that the *average* regret across all players is the quantity of interest, instead of the maximum regret. Operationally, this is implemented as summing each deviation's gradient in the dual. This treats the players anonymously, thus we implicitly and incorrectly assume that conditioned on the county's parameters each firm is identical. Due to the use parameter tying, the ICE predictor has an additional 156 model parameters.

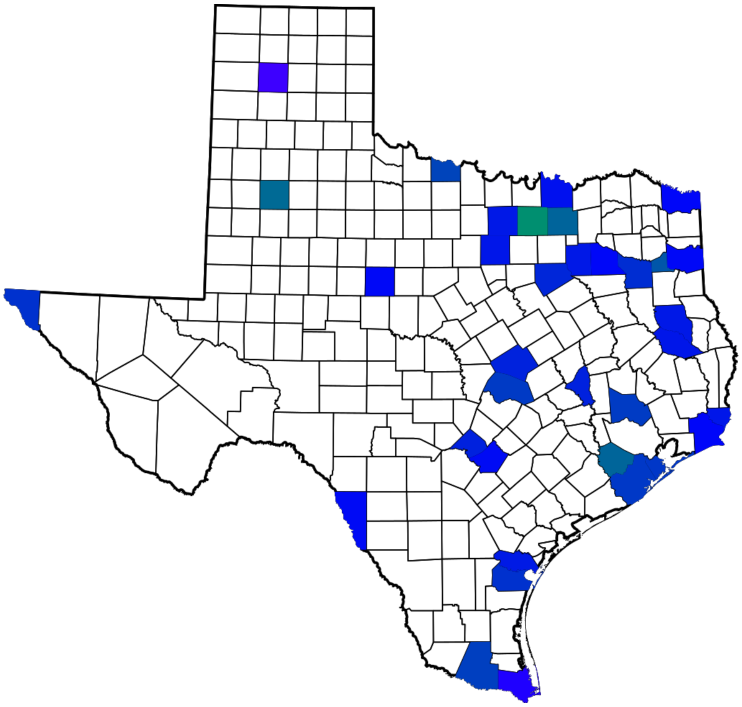

Figure 3.8: The marginalized probability that a chain will build a hotel in Spring 1996 predicted by MaxEnt ICE. Brighter shades of green denote higher probabilities.

The test losses reported were computed using ten-fold cross validation. To fit the regularization parameters for logistic regression, MaxEnt IOC and MaxEnt ICE, we held out 10% of the training data and performed a parameter sweep. For logistic regression, a separate parameter sweep and regularization was used for the bias and utility features. For MaxEnt ICE, an

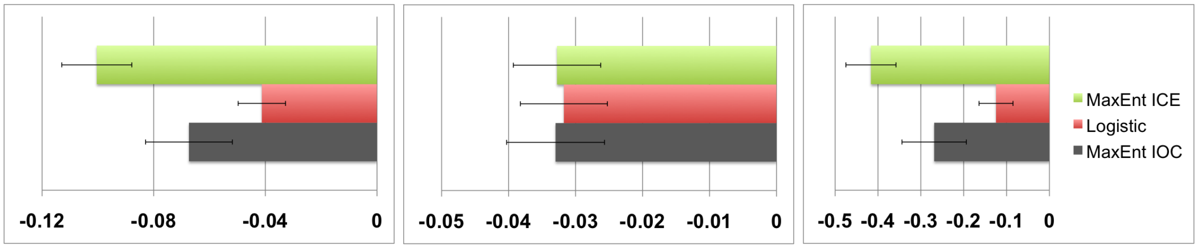

Figure 3.9: (Left) Test log loss on the full outcome space relative to the smoothed multinomial, which has log loss 1.58234 ± 0.058088. (Center) Test log loss no build vs. build outcomes only. Loss is relative multinomial, with log loss 0.721466 ± 0.016539. (Right) Test log loss conditioned on build outcomes only. Loss is relative multinomial, with log loss 6.5911 ± 0.116231.

additional regularization parameter was selected for the regret parameters. A sample of the predictions from MaxEnt ICE are shown in Figure 3.8.

In the left of Figure 3.9, we present the test errors of the three parameterized methods in terms of their offset from that of the featureless multinomial. This quantity has lower variance than the absolute errors, allowing for more accurate comparisons. We see that the addition of the regret features more than doubles the improvement of logistic regression from 2.6% to 6.3%, where as the inverse optimal control method only sees a 4.3% improvement.

In the center of Figure 3.9, we show the test log-loss when the methods are only required to predict if any firm acts. Here, the models are still trained over their complete outcome spaces and their predictions are marginalized. We see that all three methods are equal within noise. That is, the differences in the predictive performances come solely from each method's ability to predict *who* acts. We additionally performed this experiment without the use of regulatory features and found that the logistic regression method achieved a relative loss of -0.027300. Using a paired comparison between the two methods, we note that this difference of 0.004443 is significant with error 0.001886. This echoes Suzuki's conclusions the regulatory environment in this industry affect firms' decisions to build new hotels [91], measured here by improvements in predictive performance.

In the right of Figure 3.9, we demonstrate the test log loss conditioned on at least one firm acting—the portion of the loss that differentiates the methods. The logistic regression method with only utility features performs the worst with a 1.8% improvement over the multinomial base line, the individual inverse optimal control method improves by 4.1% and MaxEnt ICE performs the best with a 6.3% improvement. That is, the addition of regret features, and hence accounting for the strategic aspects of the game, have a significant effect on the predictive performance in this setting. We note that replacing the regulatory features in the regret portion of the MaxEnt ICE model actually slightly improves performance to -0.471763, though not by a significant margin. This implies that the regulatory features have little or no bearing on predicting exactly the firm that will act, which suggests the regulatory practices are unbiased.

Chapter 4

Zero-sum Equilibrium Computation

The contributions from this chapter were done in collaboration with Michael Bowling and Dustin Morrill.

4.1 Regression Regret Matching

Let us begin by presenting a new, yet simple, algorithm for the standard online learning framework. Let A be the set of N **actions** or **experts**. On each time step $t \in [T] \equiv \{1, \ldots, T\}$, the online algorithm chooses $x^t \in \Delta_A$, a distribution over the experts, then observes u^t, a bounded reward vector where $\|u^t\|_\infty \leq L$. The algorithm receives reward $(x^t \cdot u^t)$ and then updates its prediction.

An algorithm's **external regret** compares its performance to the best expert in hindsight. To be **no-regret** is to have regret grow sublinearly in T,

$$R^{\text{ext}} = \max_{x^* \in \Delta_A} \sum_{t=1}^{T} (x^* \cdot u^t) - (x^t \cdot u^t) \in o(T).$$

That is, its average regret goes to zero in the worst-case.

A simple algorithm with this property is regret-matching.

Definition 9 (Regret-matching)**.** *Define the vector of* ***cumulative regret*** *as*

$$R^T \equiv \sum_{t=1}^{T} u^t - (x^t \cdot u^t)e$$

where e is the vector of all ones. Choose $x^t \propto \max\{0, R^{t-1}\}$.

Theorem 11 (Hart and Mas-Colell [40])**.** *Regret-matching is no-regret. In particular, $R^{ext} \leq L\sqrt{NT}$.*

We often have structure between the available actions, *e.g.*, when an action corresponds to betting a particular amount or choosing an acceleration. It is common to model such situations by discretizing the range of available actions and using a single action per interval. In this case, the structure linking actions is completely lost after this abstraction.

Algorithm 2 Regret-matching with Regret Regression

```
$X \leftarrow [], y \leftarrow []$
for $t \in [T]$ do
    $f \leftarrow$ TrainRegressor$(X, y)$
    for $a \in A$ do
        $z(a) \leftarrow f(\varphi(a))$
    end for
    $x^t \propto \max\{0, z^t\}$
    Observe $u^t$
    for $a \in A$ do
        $X \leftarrow X \cup \varphi(a)$
        $y \leftarrow y \cup (u^t(a) - (x^t \cdot u^t))$
    end for
end for
```

In more complicated stateful settings, we can describe each action $a \in A$ with a **feature vector**, $\varphi(a) \in \mathcal{X}$. In poker, there are numerous ways to quantify the strength of a particular hand. For example, its expected value against different possible opponent holdings as well as potential-aware statistics, like the variance of the expected value. Current techniques use eight to ten such statistics to describe the portion of the state space representing the player's hand.

To reduce the action space in these settings, prior work uses unsupervised clustering techniques on the actions in feature space. This can essentially be thought of as an informed discretization of the space. The hope is that actions with similar features can be collapsed, or made indistinguishable, without incurring too much loss [80]. We use this feature vector to retain the structure *during learning*. To reduce to the standard framework, we can simply use an indicator feature for each action.

Algorithm 2 contains the pseudocode for our approach. It employs an arbitrary mean-squared-error minimizing regressor to estimate the cumulative regret. In particular, the training procedure is expected to produce f where $f(\varphi(a)) \approx R^t(a)/t$, or equivalently, the algorithm's regret estimate is $\tilde{R}^t(a) = tf(\varphi(a))$.

We consider the error of our regressor with the l_2 distance between the approximate and true cumulative regret vectors: $\|R^t - \tilde{R}^t\|_2$. Note that this quantity is related to the representational power and recall of the regressor. In particular, if we cannot represent the true cumulative regret then it will be non-zero due to this bias. The regret of Algorithm 2 can then be bounded in terms of this representational power.

Theorem 12. *If for each $t \in [T]$, $\|R^t - \tilde{R}^t\|_2 \leq \epsilon$, then regression regret-matching has regret bounded by $\sqrt{TNL^2 + 2TL\epsilon}$.*

The proof is in Appendix C.

This is a worst-case bound. That is, so long as ϵ is small regression regret-matching cannot be much worse than normal regret-matching. It is possible that there are cases where regression regret-matching can do better than regret-matching if the structure is favorable.

The first subtlety we must note is that the regressor minimizes the mean-squared error of the immediate regrets, *i.e.*

$$\left\{\left(\varphi(a), u^i(a) - u^i \cdot x^i\right) \mid \forall a \in A, i \in [t]\right\}$$

forms the training set. If the hypothesis class can represent the regrets, then $tf(\varphi(a)) = R^t(a)$ obtains the minimum mean-squared error. Note that this error is likely not zero even in the realizable case!

In words, the theorem states that if the error of the regressor decreases like $O(1/T)$ then the algorithm obtains a $O(\sqrt{T})$ regret bound. Note that the cumulative regret can grow like $O(T)$, so a fixed ϵ across time implies a decreasing error rate. The algorithm remains no-regret so long as the bias goes to zero and recall goes to one, *i.e.*, it is asymptotically unbiased. If the bias is constant then there is a constant term in the regret bound. When solving a game, this constant term constitutes the error introduced by a lossy abstraction.

Note that it is sufficient to make the estimator asymptotically unbiased by including indicator features for each action with proper regularization. In a sense, this allows the algorithm to move from an estimate of the true regrets as time increases, *i.e.*, as the true regrets stabilize.

Next, we aim to use our algorithm for sequential decision-making and to relate it to current abstraction techniques. Before we can accomplish this, we must review the extensive-form game framework.

4.2 Regression CFR

We are now ready to put the pieces together to form our new regret algorithm for sequential decision-making scenarios: Regression CFR (RCFR). The algorithm is simple. We will minimize counterfactual regret at every information set using estimates of the counterfactual regret that comes from a single common regressor shared across all information sets. The common regressor uses features, $\varphi(I, a)$, that are a function of the information-set/action pair. This allows the regressor to generalize over similar actions and similar situations in building its regret estimates.

As with CFR, we can derive a regret bound, which in turn implies a worst-case bound on the quality of an approximate equilibrium resulting from self-play.

Theorem 13. *If for each $t \in [T]$ at every information set $I \in \mathcal{I}_i$, $\|R^t(\cdot|I) - \tilde{R}^t(\cdot|I)\|_2 \leq \epsilon$, then RCFR has external regret bounded by $|\mathcal{I}_i|\sqrt{TNL^2 + 2TL\epsilon}$.*

The proof combines Theorem 12 together with the CFR convergence proof of Zinkevich et al. [101].

Now that we have presented RCFR, let us examine how it relates to modern abstraction techniques.

4.3 Generalized Game Abstraction

As noted in the introduction, often the problem we wish to solve, when viewed without any structure, is intractably large. For example, representing a behavioral strategy profile in no-limit

Texas Hold'em[1] requires $8.2 \cdot 10^{160}$ entries [49, p. 12]. To surmount this, we typically first distill the game to a tractably-sized abstract game. Then, after solving it, we map its solution into the full game. The hope is that the abstract game and its solution roughly maintain the important strategic aspects of the full game. Though this turns out to be false [95], it appears to not occur with any significance in the large games of interest [3, 50].

The common way to abstract a game is to simply group together similar information sets [35, 46]. That is, we take situations in the full game that the player can differentiate and make them indistinguishable in the abstract game. If the two situations have similar behavior in equilibrium, then at the very least we have not lost representational power in doing so. This form of abstraction provides a function f mapping full game information sets to abstract game information sets.

To create such an abstraction requires some notion of similarity and compatibility of information sets, and a clustering algorithm, like k-means. Ultimately, the size of the abstract game is a function of the number of clusters—how the information sets collapse together. Let us assume that we have a function $\varphi : \mathcal{I} \times A \to \mathcal{X}$ that maps information-set/action pairs to domain-specific features in space $\mathcal{X}$. We can use such a function to define the similarity of compatible information sets I and I' by, for example, the cumulative inner product, $\sum_{a \in A} \langle \varphi(I, a), \varphi(I', a) \rangle$, which can be passed to k-means to cluster the information sets (often subject to constraints such as preserving perfect recall).

In order to compare the approach to RCFR, consider a single iteration of the counterfactual regret update in both the original game and the abstract game where the players' strategies are fixed. In particular, we have

$$\tilde{r}_i^t(a|I^{\text{abstract}}) = \sum_{I^{\text{full}} \in f^{-1}(I^{\text{abstract}})} r_i^t(a|I^{\text{full}}).$$

That is, the regret at information set I^{abstract} in the abstract game, $\tilde{r}_i^t(\cdot|I^{\text{abstract}})$ is the sum of the regrets in the full game of all information sets that map to it, $f^{-1}(I^{\text{abstract}})$. Taking this view-point in reverse, using CFR on the abstract game is operationally equivalent to solving the full game where we maintain and update the regrets $\tilde{r}_i^t(\cdot|f(I^{\text{full}}))$, *i.e.*, we approximate the true regrets with a tabular regressor $r_i^t(a|I^{\text{full}}) \approx \tilde{r}_i^t(a|f(I^{\text{full}}))$. Thus, abstraction can be thought of as a special-case of RCFR with a particular choice for how to approximate the cumulative counterfactual regret.

RCFR is, of course, not restricted to tabular regressors and so can capture structure that goes beyond traditional abstraction. For example, using linear or kernel regression provides a sort of "soft" abstraction. That is, the regret of two distinct (in feature space) actions can effect one another without making the two completely indistinguishable, which is the only option available to traditional "hard" abstraction.

Unlike traditional abstraction where f is chosen a priori, RCFR is also able to learn the structure of the abstraction online upon seeing actual data. And since the regressor is trained after each time step, RCFR can effectively re-abstract the game on-the-fly as necessary. Furthermore, the abstraction is informed by the game *in a way that is compatible with the learner*. Imperfect information games have an interesting property in that portions of the game that are never reached

[1] We consider the game with 50-100 blinds with 20,000 chip stacks as played in the Annual Computer Poker Competition.

in optimal play have strategic effects. For example, if one player was to behave suboptimally, the other might benefit by deviating into an otherwise undesirable space. Without the possibility of deviating there may be no way to punish the poor play. Practically, this presents as a rather annoying hurdle for abstraction designers. In particular, seemingly ideal abstractions tailored perfectly to the structure of an equilibrium may actually lead to poor solutions due to missing important, but seemingly unnecessary, parts of the strategy space. RCFR avoids all of this as the regressor tunes the abstraction to the current policy, not the solution.

4.4 Leduc Hold'em Experimental Results

In order to illustrate the practicality of RCFR, we test its performance in Leduc Hold'em, a simplified poker game. Our goal is to compare the strategies found by RCFR with varying regressors to strategies generated with conventional abstraction techniques. In addition, we examine the iterative behaviour of RCFR compared to CFR.

Leduc Hold'em is a poker game based on Kuhn poker [84]. It provides a good testbed as common operations, like best response and equilibrium computations, are tractable and exact.

The game has two betting rounds, the preflop and flop. At the beginning of the game both players ante a single chip into the pot and are dealt a single private card from a shuffled deck of six cards—two jacks, two queens and two kings. Then begins the preflop betting round where the first player can either check or bet. If the first player checks, passing their turn, then the second player can end the betting round by checking as well, or continue by betting. When facing a bet; the player can raise by placing two chips into the pot; call by matching the bet in the pot and ending the round; or fold by forfeiting the pot to the opponent. There is a maximum of two wagers per round, *i.e.*, one bet and one raise. A single public card is dealt face up for both players to see at the beginning flop. If the flop betting ends without either player folding, a showdown occurs and the player with the best hand takes the pot. A player that pairs, *i.e.*, their card matches the public card, always has the best hand no matter the rank of the paired card or the opponent's card. If neither player pairs, the one with the highest rank card wins. In the event of a tie, the pot is split. The size of a wager preflop is two chips, and is doubled to four chips on the flop.

Leduc Hold'em has 672 sequences. At equilibrium, the first player is expected to lose 0.08 chips per hand. We show results in milliblinds/antes per hand (mb/h), a thousandth of a chip, *i.e.*, optimally the first player loses 80 mb/h.

We use a regression tree aiming to minimize mean-squared error as our regressor. When training, we examine all candidate splits on a single feature and choose the one that results in the best immediate error reduction. The data is then partitioned according to this split and we recursively train both sets. If the error improvement at a node is less than a threshold, or no improvement can be made by any split, a leaf is inserted that predicts the average. It is this error threshold that we manipulate to control the complexity of the regressor—the size of the tree. All the training data is kept between iterations, as in Algorithm 2.

Eight features were chosen such that the set of features would be small, thus allowing fast regression tree training, which is done on every RCFR iteration, but still descriptive enough to have a unique feature expansion for every sequence: **1)** the expected hand strength ($E[HS]$), or the probability of winning the hand given the available information, and marginalized over a uniform distribution of opponent hands and possible future board card; **2)** the rank of the board card, or zero on the preflop; **3)** the pot size; **4)** the pot relative size of the wager being faced, or zero if not facing a wager; **5)** the number of actions this hand; **6)** the number of wagers this hand; **7)** an indicator on whether or not the next action would be a fold action; and **8)** the pot relative size of the wager that would be made by taking the next action, or zero if the next action would be a check or call.

Features **(1)** and **(2)** refer to private and public card information, **(3)** through **(6)** are public chip and action information, while **(7)** and **(8)** fully describe the next potential action in the sequence[2].

We evaluate the empirical performance of RCFR here according to three metrics: **1)** convergence properties, **2)** exploitability, and **3)** one-on-one competitions.

Strategies were computing using RCFR and four different error threshold values. Each threshold was chosen so that RCFR's regressor would have similar complexity to that of a conventional abstraction, or that of the full game. In Leduc Hold'em, a typical abstraction is one that groups together cards on the preflop and only distinguishes between pairing and non-pairing board cards on the flop. These hand-crafted abstractions are analogous to the $E[HS]$ based abstractions commonly used in Texas Hold'em. Abstractions are denoted, for example, `J.QK` to describe the abstraction that can distinguish between a jack and a queen or king, but cannot distinguish between a queen and king. The remaining three abstractions are then `JQK`, `JQ.K`, and `J.Q.K`. One may also note that `J.Q.K` is a strict refinement of the other three abstractions, and `J.QK` and `JQ.K` both are strict refinements of `JQK`. To generate strategies, chance sampling CFR [100] was run for $100,000$ iterations to solve each abstract game.

Each different RCFR strategy is denoted, for example, `RCFR-22%`, to describe RCFR using a regressor 22% the size of a strategy in the full game. `RCFR-22%`, `RCFR-47%`, and `RCFR-66%` correspond to `JQK`, `J.QK`/`JQ.K`, and `J.Q.K`, respectively, in terms of complexity. `RCFR-96%` corresponds to `FULL`, which denotes no abstraction, and it was made by setting the error threshold to zero, so the regressor was free to split on every feature and become as large as the full game. RCFR and CFR were run for $100,000$ iterations to generate the set of RCFR strategies and a `FULL` strategy, respectively.

Figure 4.1 shows that all RCFR strategies improve at the same rate as an unabstracted CFR strategy (`FULL`) until a plateau is reached, the height of which is determined by the error threshold parameter of that RCFR instance. These plateaus are essentially the exploitability cost incurred by estimating regrets instead of computing and storing them explicitly. Larger thresholds, reducing regressor complexity, incur a greater cost and thus have a higher plateau. As expected, when the error threshold is set to zero, as in the case of `RCFR-96%`, RCFR's progression mimics unabstracted CFR for the full set of $100,000$ iterations.

Figure 4.2 shows that RCFR, given complexity restrictions equivalent to those of conventional abstractions, finds significantly less exploitable strategies. `RCFR-66%`'s regressor is 2% smaller than the size of the `J.Q.K` abstract game, yet `J.Q.K` is sixteen times more exploitable! The closest corresponding strategies in terms of exploitability are `RCFR-47%` and `JQ.K` where

[2] No explicit check/call feature is necessary because it is implicitly encoded by features **(7)** and **(8)** in combination. The action would be a check/call if and only if both are zero (the action would not be a fold nor a wager, and the game has only three action types).

	RCFR-22%	JQK	RCFR-47%	J.QK	JQ.K	RCFR-66%	J.Q.K	RCFR-96%	FULL	Mean
RCFR-22%		319.50 ± 1.81	−97.61 ± 1.31	58.25 ± 1.45	−100.70 ± 1.26	−123.70 ± 1.30	−88.90 ± 1.34	−135.89 ± 1.34	−139.90 ± 1.33	−38.62 ± 1.39
JQK	−319.50 ± 1.81		−453.35 ± 1.85	−405.96 ± 1.97	−363.66 ± 1.70	−452.82 ± 1.85	−369.79 ± 1.83	−495.46 ± 1.85	−486.06 ± 1.86	−418.33 ± 1.84
RCFR-47%	97.61 ± 1.31	453.35 ± 1.85		140.32 ± 1.44	14.19 ± 1.27	−34.91 ± 1.26	18.07 ± 1.29	−44.91 ± 1.26	−47.98 ± 1.27	74.47 ± 1.37
J.QK	−58.25 ± 1.45	405.96 ± 1.97	−140.32 ± 1.44		−107.66 ± 1.35	−147.86 ± 1.42	−94.31 ± 1.45	−149.31 ± 1.41	−156.26 ± 1.43	−56.00 ± 1.49
JQ.K	100.70 ± 1.26	363.66 ± 1.70	−14.19 ± 1.27	107.66 ± 1.35		−46.71 ± 1.26	−33.16 ± 1.30	−53.90 ± 1.27	−56.77 ± 1.27	45.91 ± 1.33
RCFR-66%	123.70 ± 1.30	452.82 ± 1.85	34.91 ± 1.26	147.86 ± 1.42	46.71 ± 1.26		33.65 ± 1.26	−10.17 ± 1.25	−9.41 ± 1.24	102.51 ± 1.35
J.Q.K	88.90 ± 1.34	369.79 ± 1.83	−18.07 ± 1.29	94.31 ± 1.45	33.16 ± 1.30	−33.65 ± 1.26		−36.82 ± 1.26	−37.69 ± 1.25	57.49 ± 1.37
RCFR-96%	135.89 ± 1.34	495.46 ± 1.85	44.91 ± 1.26	149.31 ± 1.41	53.90 ± 1.27	10.17 ± 1.25	36.82 ± 1.26		0.00 ± 1.25	115.81 ± 1.36
FULL	139.90 ± 1.33	486.06 ± 1.86	47.98 ± 1.27	156.26 ± 1.43	56.77 ± 1.27	9.41 ± 1.24	37.69 ± 1.25	0.00 ± 1.25		116.76 ± 1.36

Table 4.1: One-on-one competition crosstable. Each cell is the bankroll for the row player in mb/h and 95% confidence interval of playing the row agent against the column agent for 30,000,000 hands.

JQ.K is *only* three and a half times more exploitable.

Another useful practical property of RCFR is that it appears to avoid non-monotonicities that have been observed in hand-crafted abstractions [95]. That is, increasing the complexity of the regressor appears to improve the full game exploitability of the resulting strategy. Table 4.1 is the one-on-one competition crosstable between each of the agents. Against almost every opponent, each RCFR variant outperforms its corresponding strategy. The exceptions, for example, JQ.K wins 100.70 mb/h against RCFR-22% while RCFR-47% wins only 97.61 mb/h against this same opponent, are small margins. In addition, each RCFR strategy defeats or, in the case of RCFR-96%, ties its counterparts. RCFR-22% and RCFR-47% even win against larger abstract strategies J.QK and J.Q.K, respectively. Dividing the agents into an RCFR team and a conventional agents team, the RCFR team wins 2033.34 mb/h in aggregate.

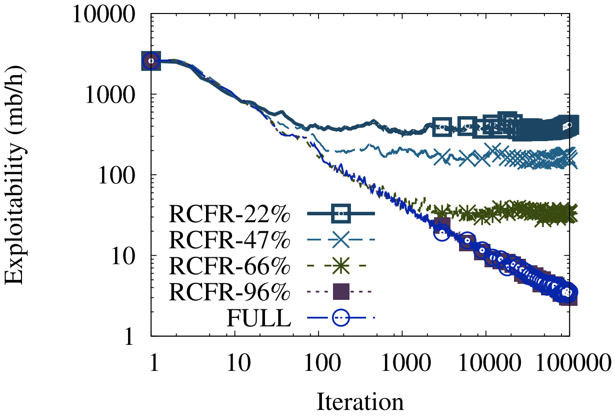

Figure 4.1: Convergence of RCFR using various error thresholds (complexity limitations) compared with CFR on the unabstracted game (`FULL`).

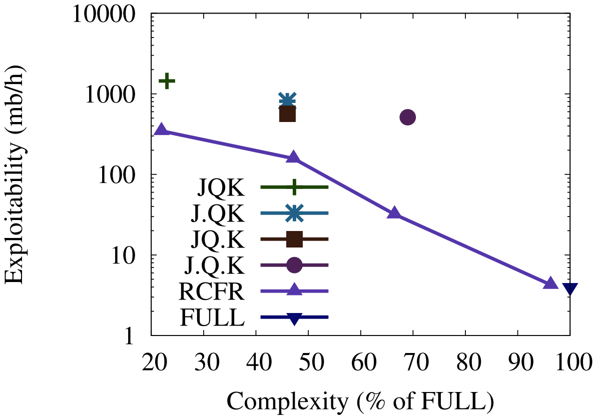

Figure 4.2: Exploitability of final strategies after $100,000$ iterations of RCFR, unabstracted CFR (`FULL`), or chance sampling CFR (`JQK`, `J.QK`, `JQ.K`, and `J.Q.K`). The horizontal axis shows the complexity of the solution method as a percentage of the size of the unabstracted game. RCFR's complexity is the size of the regression tree while the complexity of a conventional abstraction is the size of its abstract game.

Chapter 5

Connections, Future Work and Conclusions

5.1 Future Applications

A recent paper by Koster et al. [55] demonstrates an exciting and promising application of deep reinforcement learning to a mechanism design problem. In particular, they study a scenario where four human players repeatedly invest as a group and the learned-mechanism redistributes the investment returns to the group depending on the players' total capital and the amount they chose to invest. For example, the strict egalitarian mechanism would divide the returns equally regardless of each players' investment amount, and the liberal egalitarian mechanism divides the returns according to the fraction of the total investment amount. Neither of these example mechanisms account for a players' initial advantage relative to the others, which can can lead to poor outcomes if there is a large disparity in capital at the beginning of the game. Koster et al. [55] trained their agent to imitate humans that redistribute investment returns and found that their agent was preferred over a number of baselines.

There are a number of limitations of this experimental setting that could potentially be overcome by incorporating an ICE-like approach into the mechanism learner. One such limitation is that there is a single resource–investment capital–that is considered. An important consequence of this is that the mechanism is privileged to the players' true utility. When there are multiple resources then an individual player may prefer one resource to another and this individual preference cannot be known a priori by the others. A similar consequence could also arise from having multiple investment actions as well, e.g., choosing between a high risk/high reward investment or a safer one that either has a lower reward or a longer term.

Multiple resources can also be used to represent situations where the players' have competitive advantages over one-and-other, e.g., a farmer can grow wheat to trade to a baker for bread. Finally, in this experimental setup the players themselves do not really need to act strategically, that is, they do not need to carefully consider and change their actions based on the actions and preferences of the others. This strategic reasoning can take the form of either collaboration or competition. For example, the farmer and the baker cooperate, but the baker and the chef both demand flour and the goods they produce are somewhat substitutable. As above, a reinforcement learning approach may struggle in a richer experimental setup where the players' actions are not obviously utility maximizing due to others.

5.2 Single Utility Function

During the thesis proposal, it was suggested that one utility function is all that should be necessary, as opposed to one per deviation. It is unclear how to construct a rationality constraint from the primal problem that uses a single utility function, but we can consider using a single utility function from the MaxEnt ICE dual problem.

If for all $f \in \Phi$ we choose $\lambda^f = \lambda$, we have the dual problem:

$$\min_{\lambda \in \mathcal{K}} \sum_{f \in \Phi} \mathrm{Regret}(\sigma, \lambda) + \log Z(\lambda), \text{ where}$$
$$Z(\lambda) = \sum_{a \in \mathcal{A}} \exp\left(- \sum_{f \in \Phi} \langle r(a), \lambda \rangle \right).$$

Using (A.3) and (A.4) and after simplifying, we observe that this corresponds to the primal problem:

$$\max_{\hat{\sigma} \in \Delta_{\mathcal{A}}, \eta \in \Delta_{\Phi}} \quad H(\hat{\sigma}) \quad \text{subject to:}$$
$$\sum_{f \in \Phi} \left[r^f(\hat{\sigma}) - \sum_{g \in \Phi} \eta_g r^g(\sigma) \right] \in -\mathcal{K}^*$$

In particular, if we consider no restrictions on the utility function, *i.e.*, for $\mathcal{K} = \mathbb{R}^n$ then $\mathcal{K}^* = \{0\}$:

$$\sum_{f \in \Phi} r^f(\hat{\sigma}) = |\Phi| \sum_{g \in \Phi} \eta_g r^g(\sigma)$$

which corresponds to the rationality constraint

$$\sum_{f \in \Phi} \left\langle r^f(\sigma), w \right\rangle = |\Phi| \left\langle \mathrm{Regret}(\sigma), w \right\rangle \qquad \forall w \in \mathbb{R}^n$$

In words, this restricts the prediction's average regret over all deviations must equal the demonstrated behavior's overall regret under all utility functions. This is slightly awkward in that the predicted behavior and the demonstrated behavior are using different notions of regret.

This scheme does not generally have any computational or statistical benefits. The computation of the partition function still requires $O(|\mathcal{A}|)$ operations. The storage requirement and weight update are reduced by a factor of $|\Phi|$. Furthermore, it still requires accurately estimating all of the deviation regret features.

5.3 Scaling to Sequential Decision-Making Scenarios

There are numerous issues that have yet to be overcome when moving from behavior prediction in normal-form games to sequential decision-making scenarios.

First, it may simply not be generally possible to accurately predict multi-agent behavior with only trajectories sampled from the true behaviour, especially with unknown utility functions. That is, we may need additional ways to query the demonstrated behavior. For example, even in single-agent scenarios with known utility Ross et al. [78] show in Theorem 2.2 that the expected loss grows linearly with respect to the trajectory length. Roughly speaking, if the predicted behavior makes even a slight error early in the trajectory then that error can easily compound, *e.g.*, by moving to state where little or no training data has been gathered to support it. Ross et al. [78] show that by iteratively gathering training data using the predicted behavior as the exploration policy can correct for these issues.

In multi-agent behavior, these types of errors can occur for additional reasons. For example, when learning a minimax optimal strategy in zero-sum games, the agents must have certain dominated strategies available to them, as though these strategies are never played at equilibrium they can counter or dominate certain opponent strategies. That is, even with a priori knowledge that an action does not support any equilibrium it cannot be safely pruned from the strategy space during learning. This is still true with full knowledge of the support for both players one cannot prune the strategy space. Said another way, why an agent acts in a particular fashion in an observed state often depends on how they reason about states that are never visited at equilibrium.

Second, typically when considering sequential decision-making scenarios one typically decomposes the problem in a fashion where individual decisions can be considered separately. For example, the counterfactual regret decomposition reduces minimizing overall regret over the entire strategy space to minimizing external regret at individual decisions [101]. More precisely, counterfactual regret allows one to consider deviating from one action to another at a single decision as opposed to deviating from an action to an entire strategy for the remainder of the game. When considering games with unknown utility functions, as we have in the normal-form case, this leads to to building sets of possible future utility features, as opposed to a single utility value. Loosely speaking, a best response simply takes the max utility of all actions at a decision and propagates it to the parent decision. As we do not know the true utility we must propagate the information required to perform that max once the true utility is specified to the parent decision. This may lead to an exponentially-sized description of the future utility feature set. To combat this, one can consider working with the primal problem directly and somehow performing constraint generation on the utility functions. Another possibility is to consider a more restrictive type of game, like a Markov game, where one could consider using MaxEnt ICE inside an algorithm like Minimax-Q [62].

Third, as discussed in the thesis proposal, extensive-form correlated equilibria are particularly challenging even to solve for with known utility. Since the proposal, there has been advances in this area. Particularly, Celli et al. [14] present a counterfactual-regret-like online learning algorithm that converges to an extensive-form correlated equilibrium. Of particular interest, this algorithm demonstrates that the agents can converge to correlated behavior without explicit communication in sequential decision-making scenarios. Despite this result, there are still a number of challenges regarding extensive-form correlated equilibria that make behavior prediction challenging. Specifically, the number of constraints defining the set of extensive-form correlated equilibria is in general exponential. Second, the representation of an arbitrary joint-strategy is also in general exponentially-sized in the game description. Celli et al. [14] does not avoid the latter issue as they must store all the strategies encountered and they do not provide a bound on the convergence, *i.e.*, they never form an explicit joint-strategy and their algorithm is potentially exponential in both computation and memory. A further consequence of these issues is that it may not be possible to optimize for/select a particular extensive-form correlated equilibrium in polynomial time. One would hope and need to argue that with a bounded number of trajectories from a extensive-form correlated equilibrium that the representation of the predicted behavior also be of bounded size and require bounded computation to find. In particular, Dudík and Gordon [23] show that one can efficiently approximate the maximum entropy extensive-form correlated equilibrium in polynomial time, giving hope that a maximum entropy approach to behavior prediction is possible at least with known utility.

5.4 Safe Subgame Resolving

Shortly after we introduced MaxEnt ICE, Burch et al. [12] introduced an algorithm, CFR-D, which allows one to decompose a zero-sum equilibrium computation in an imperfect information game. Specifically, the game is partitioned into subgames along its *public tree*. There are a number of specific conditions as to what a valid public tree decomposition is, but for the purposes of this exposition one can think of it as requiring all histories in the transitive closure over both players' information to be grouped together, *e.g.*, in poker the betting tree along with any public cards forms the public tree.

The magic of CFR-D is that one can summarize the equilibrium strategy for a player in a subgame by two vectors–the probability that the player reaches the subgame for every one of the player's initial decisions in the subgame, and the value to the opponent of every one of their initial decisions. With this information, one can perform what has become known as a *safe re-solve computation* to recover the equilibrium strategy.

The re-solve computation itself is an equilibrium computation on a slightly modified game, known as a *gadget* game. Specifically, upon the opponent reaching their initial decision, they may choose to opt-out of playing the subgame and instead receive the value of that decision. Upon solving the gadget game, the player's (and not the opponent's) strategy will approximate an equilibrium strategy in that subgame. Since the original paper, others have determined other ways to formulate the gadget game [65].

Recalling the discussion of the dual estimation procedure for zero-sum games in Chapter 2, we advocate instead of estimating the strategy for one player that one estimate the regrets of the opponent, just as is being done here. Further, inspection of the re-solve computation itself elucidates that we can interpret it as a behavior prediction program with known utility and without any additional selection criteria. Specifically, the strategy set for the player is the set of strategies bestowing the at least the known values to the opponent at their initial decisions.

Safe re-solving is a critical component in a number of recent high profile results: DeepStack [66], a expert-level no-limit poker agent; Libratus, the first no-limit poker agent to beat strong professionals [8, 9, 10]; and Player of Games, a generalization of AlphaZero to imperfect information games [79, 83].

A significant issue with AlphaZero and Player of Games is the sheer volume of data, and thus compute, required to train a strong agent. For example, AlphaZero required 44 million training games played over nine hours on expensive specialized hardware to learn to play Chess. To make it possible for a graduate student to reproduce such results we will need to either reduce the number of samples required by the algorithm, or reduce the dependency on specialized hardware. By viewing this problem with a behavior prediction lens it may be possible to make progress on both of these fronts.

5.5 Beyond Regression CFR

Regression CFR has spearheaded a line of research on computing Nash equilibria in zero-sum games by employing machine learning techniques to learn strategy representations [11, 22, 43, 60, 85, 89, 90]. Unfortunately, R-CFR as well as its successors have not yet been as successful as the search-based methods like AlphaZero, DeepStack, and Player of Games [66, 79, 83].

There at least two issues holding back these methods. First, it can be hard for these methods to get started. Specifically, when starting from a random strategy the regrets do not exhibit as much structure as might be required to learn a good predictor. This can make it hard or impossible to move away from the initial random strategy. This is less of a problem for the search-based methods as the search itself, given that it is deep enough, imposes some structure on the targets.

Second, when computing a Nash equilibrium in an imperfect information game using a self-play method like CFR or fictitious play, it is the average policy that converges to the equilibrium, not the final policy of the players. That is, these methods must somehow record or be able to recover the average strategy at test time. Our work on R-CFR did not tackle this problem at all, but subsequent work has gravitated towards recording trajectories of play during the self-play computation in a large replay buffer and then learning an average strategy predictor at the end of self-play. That is, they are performing behavior prediction in the way that DAgger tells us can fail catastrophically [78]. Additionally, the behavior prediction problem here involves only a single-agent. That is, this presents a perfect opportunity to apply DAgger, or some other hybrid re-solve-based method to push the state-of-the-art in this area.

5.6 Additional Publications

Below is a list of related publications that I have co-authored since starting at Carnegie Mellon that are not presented in full detail in this dissertation.

- M. Schmid, M. Moravčík, N. Burch, R. Kadlec, J. Davidson, K. Waugh, N. Bard, F. Timbers, M. Lanctot, Z. Holland, E. Davoodi, A. Christianson, and M. Bowling. Player of games, 2021. URL `https://arxiv.org/abs/2112.03178`.
- C. Kroer, K. Waugh, F. Kılınç-Karzan, and T. Sandholm. Faster algorithms for extensive-form game solving via improved smoothing functions. *Mathematical Programming*, 179 (1):385–417, 2020.

- T. Davis, K. Waugh, and M. Bowling. Solving large extensive-form games with strategy constraints. In *Conference on Artificial Intelligence (AAAI)*, 2019.
- M. Moravčík, M. Schmid, N. Burch, V. Lisý, D. Morrill, N. Bard, T. Davis, K. Waugh, M. Johanson, and M. Bowling. Deepstack: Expert-level artificial intelligence in heads-up no-limit poker. *Science*, 356:508–513, 2017.
- C. Kroer, K. Waugh, F. Kılınç-Karzan, and T. Sandholm. Theoretical and practical advances on smoothing for extensive-form games. In *Conference on Economics and Computation (EC)*, 2017.
- C. Kroer, K. Waugh, F. Kılınç-Karzan, and T. Sandholm. Faster first-order methods for extensive-form game solving. In *Conference on Economics and Computation (EC)*, 2015.
- K. Waugh and J. A. Bagnell. A unified view of large-scale zero-sum equilibrium computation. In *Workshop on Computer Poker and Imperfect Information (AAAI)*, 2014.
- K. Waugh. A fast and optimal hand isomorphism algorithm. In *Workshop on Computer Poker and Imperfect Information (AAAI)*, 2013.
- S. Ganzfried, T. Sandholm, and K. Waugh. Strategy purification and thresholding: Effective non-equilibrium approaches for playing large games. In *Autonomous Agents and Multi-agent Systems (AAMAS)*, 2012.
- S. Ganzfried, T. Sandholm, and K. Waugh. Strategy purification. In *Workshop on Computer Poker and Imperfect Information (AAAI)*, 2011.
- Michael Johanson, Michael Bowling, Kevin Waugh, and Martin Zinkevich. Accelerating best response calculation in large extensive games. In *Proceedings of the Twenty-Second International Joint Conference on Artificial Intelligence (IJCAI)*, pages 258–265, 2011.

Appendix A

Behavior Prediction with Known Utility

To begin, we recall some useful definitions and properties of convex functions will aid us in our proofs. For more details see Boyd and Vandenberghe [6].

Recall the definition of the **convex conjugate** of a function $f : \mathbb{R}^n \to \mathbb{R}$:

$$f^*(\lambda) = \max_x \langle \lambda, x \rangle - f(x)$$

Consider minimizing f subject to linear equality constraints:

$$\min_x \;\; f(x), \text{ subject to: } Ax = b$$

Introducing Lagrange multiplier λ we have

$$= \min_x \max_\lambda \;\; f(x) + \langle \lambda, b - Ax \rangle$$

So long as the constraints are satisfiable and the minimum exists

$$\begin{aligned}
&= \max_\lambda \min_x \;\; f(x) + \langle \lambda, b - Ax \rangle \\
&\quad = \max_\lambda \;\; \langle \lambda, b \rangle + \min_x f(x) - \langle \lambda, Ax \rangle \\
&\quad = \max_\lambda \;\; \langle \lambda, b \rangle - \max_x \langle \lambda, Ax \rangle - f(x) \\
&\quad = \max_\lambda \;\; \langle \lambda, b \rangle - f^*(A^T \lambda)
\end{aligned} \tag{A.1}$$

i.e., we can write the dual problem in terms of the convex conjugate.

A similar dual results from when minimizing f subject to linear inequality constraints, again subject to the constraints being satisfiable and the minimum existing:

$$\begin{aligned}
&\min_x \;\; f(x), \text{ subject to: } Ax \geq b \\
&= \max_{\lambda \geq 0} \;\; \langle \lambda, b \rangle - f^*(A^T \lambda)
\end{aligned} \tag{A.2}$$

Unlike previously, here λ is restricted to the non-negative orthant.

Generalizing further, let $\mathcal{K} \subseteq \mathbb{R}^m$ be a non-empty convex set. Recall, that the **dual cone** $K^* = \{y \mid \langle y, x \rangle \geq 0, \forall x \in \mathcal{K}\}$. If we minimize f subject to the convex inclusion constraint:

$$\min_x \ f(x), \text{ subject to: } Ax - b \in \mathcal{K}$$

So long as there exists x such that $Ax - b \in \operatorname{int}(\mathcal{K})$ and the minimum exists

$$\begin{aligned}
&= \min_x \max_{\lambda \in \mathcal{K}^*} \ f(x) - \langle \lambda, Ax - b \rangle \\
&= \max_{\lambda \in \mathcal{K}^*} \min_x \ f(x) - \langle \lambda, Ax - b \rangle \\
&= \max_{\lambda \in \mathcal{K}^*} \ \langle \lambda, b \rangle - \max_x \langle \lambda, Ax - b \rangle - f(x) \\
&= \max_{\lambda \in \mathcal{K}^*} \ \langle \lambda, b \rangle - f^*(A^T \lambda) \qquad \text{(A.3)}
\end{aligned}$$

The constraint qualification above is sufficient for strong duality to hold for arbitrary $\mathcal{K}$, but it is not always necessary. For example, when $\mathcal{K} = \{0\}$ we recover the first dual we derived, which requires simply that the constraints are satisfiable.

Finally, recall the convex conjugate of the negative entropy fuction is:

$$\begin{aligned}
-H(x) &= \sum_{i=1}^{N} x_i \log x_i \text{ subject to: } x \in \Delta \\
-H^*(g) &= \log \sum_{i=1}^{N} \exp(g_i) \qquad \text{(A.4)}
\end{aligned}$$

Appendix B

Behavior Prediction with Unknown Utility

Lemma 5. $\forall x \in \mathbb{R}$ *and* $\forall a_1, a_2, ..., a_n \in \mathbb{R}$*,*

$$\begin{aligned} & x \leq \max_{i \in [n]} a_i \\ \iff & \exists \eta \in \Delta_n \textit{ s.t. } x \leq \sum_{i=1}^{n} \eta_i a_i \end{aligned}$$

Proof. Let $j = \operatorname{argmax}_{i \in [n]} a_i$. If $x \leq \max_{i \in [n]} a_i$ then choose $\eta_j = 1$,

$$x \leq a_j = \sum_{i=1}^{n} \eta_i a_i$$

If there exists $\eta \in \Delta_n$ such that

$$\begin{aligned} x &\leq \sum_{i=1}^{n} \eta_i a_i, \text{ then} \\ &\leq \sum_{i=1}^{n} \eta_i a_j = a_j = \max_{i \in [n]} a_i \end{aligned}$$

□

Proof of Theorem 5. For any $\hat{\sigma} \in \Delta_{\mathcal{A}}$:

$$\begin{array}{llrl}
 & \forall w \in \mathcal{K}, & \text{Regret}(\hat{\sigma}, w) & \leq \text{Regret}(\sigma, w) \\
\iff & \forall w \in \mathcal{K}, & \max_{f \in \Phi} \left\langle r^f(\hat{\sigma}), w \right\rangle & \leq \max_{g \in \Phi} \left\langle r^g(\sigma), w \right\rangle \\
\iff & \forall w \in \mathcal{K}, f \in \Phi, & \left\langle r^f(\hat{\sigma}), w \right\rangle & \leq \max_{g \in \Phi} \left\langle r^g(\sigma), w \right\rangle
\end{array}$$

Applying Lemma 5

$$\begin{array}{llrl}
\iff & \forall w \in \mathcal{K}, f \in \Phi \;\; \exists \eta^f \in \Delta_\Phi, & \left\langle r^f(\hat{\sigma}), w \right\rangle & \leq \sum_{g \in \Phi} \left\langle \eta^f_g r^g(\sigma), w \right\rangle \\
\iff & \forall w \in \mathcal{K}, f \in \Phi \;\; \exists \eta^f \in \Delta_\Phi, & \left\langle r^f(\hat{\sigma}) - \sum_{g \in \Phi} \eta^f_g r^g(\sigma), w \right\rangle & \leq 0
\end{array}$$

By the definition of the dual cone $\mathcal{K}^*$

$$\begin{array}{llrl}
\iff & \forall f \in \Phi \;\; \exists \eta^f \in \Delta_\Phi, & r^f(\hat{\sigma}) - \sum_{g \in \Phi} \eta^f_g r^g(\sigma) & \in -\mathcal{K}^*
\end{array}$$

□

Proof of Theorem 6. Recall the primal MaxEnt ICE optimization problem:

$$\max_{\hat{\sigma}, \eta} \;\; H(\hat{\sigma}) \quad \text{subject to:}$$

$$\begin{aligned}
r^f(\hat{\sigma}) - \sum_{g \in \Phi} \eta^f_g r^g(\sigma) &\in -\mathcal{K}^* & \forall f \in \Phi \\
\eta^f &\in \Delta_\Phi, & \forall f \in \Phi \\
\hat{\sigma} &\in \Delta_{\mathcal{A}}.
\end{aligned}$$

Momentarily just considering the objective

$$\begin{aligned}
\max_{\hat{\sigma}, \eta} H(\hat{\sigma}) &= \max_{\eta} \max_{\hat{\sigma}} H(\hat{\sigma}) \\
&= \max_{\eta} - \min_{\hat{\sigma}} -H(\hat{\sigma})
\end{aligned}$$

Now applying (A.3) using the definition of the negative entropy's conjugate function (A.4)

$$\begin{aligned}
&\max_{\eta, \lambda} \;\; \sum_{f \in \Phi} \sum_{g \in \Phi} \eta^f_g \left\langle r^g(\sigma), \lambda^f \right\rangle + \log Z(\lambda), \text{ where} \\
&Z(\lambda) = \sum_{a \in \mathcal{A}} \exp\left(- \sum_{f \in \Phi} \left\langle r^f(a), \lambda^f \right\rangle \right), \text{ subject to:} \\
&\eta^f \in \Delta_\Phi, \lambda^f \in \mathcal{K}. \qquad \forall f \in \Phi
\end{aligned}$$

Finally, we eliminate η by applying Lemma 5 to get the stated MaxEnt ICE dual problem

$$\max_{\lambda^f \in \mathcal{K}} \;\; \sum_{f \in \Phi} \max_{g \in \Phi} \left\langle r^g(\sigma), \lambda^f \right\rangle + \log Z(\lambda), \text{ where}$$

$$Z(\lambda) = \sum_{a \in \mathcal{A}} \exp\left(- \sum_{f \in \Phi} \left\langle r^f(a), \lambda^f \right\rangle \right).$$

□

Appendix C

Zero-sum Equilibrium Computation

Proof of 12. This proof is based on the proof of the polynomially weighted forecaster from [15].

Define $\Phi(R) = \sum_{i=1}^{N} \|R_+\|^2$, where R is a vector of N regrets.

Suppose that for any $\|u^t\|_\infty \leq L$ we have

$$\begin{aligned}\left\langle \nabla\Phi(R^t), u^t - u^t \cdot x^{t+1} e \right\rangle &= \left\langle \nabla\Phi(R^t), r^{t+1} \right\rangle \\ &\leq \epsilon^t\end{aligned}$$

By the Lipschitz continuity of Φ we have

$$\begin{aligned}\Phi(R^{t+1}) &\leq \Phi(R^t) + 2\left\langle \Phi(R^t), R^{t+1} - R^t \right\rangle + \|R^{t+1} - R^t\|_2^2 \\ &= \Phi(R^t) + 2\left\langle \Phi(R^t), r^{t+1} \right\rangle + \|r^{t+1}\|_2^2\end{aligned}$$

By our assumption

$$\begin{aligned}&\leq \Phi(R^t) + 2\epsilon^t + \|r^{t+1}\|_2^2 \\ &\leq \Phi(R^t) + 2\epsilon^t + NL^2\end{aligned}$$

Iterating the inequality

$$\leq (t+1)NL^2 + 2\sum_{k=1}^{t} \epsilon^k$$

Finally, we bound the overall regret by the regularized regret

$$\begin{aligned}\max_{i\in[N]} R_i^T \leq \max_{i\in[N]} (R_i^T)_+ &= \|(R_i^T)_+\|_\infty \\ &\leq \|(R_i^T)_+\|_2 = \sqrt{\Phi(R^T)} \\ &\leq \sqrt{TNL^2 + 2\sum_{t=1}^{T} \epsilon^t}\end{aligned}$$

□

Bibliography

[1] P. Abbeel and A. Y. Ng. Apprenticeship learning via inverse reinforcement learning. In *International Conference on Machine Learning (ICML)*, 2004. (document), 1, 2.3.2, 3.0.1

[2] R Aumann. Subjectivity and correlation in randomized strategies. *Journal of Mathematical Economics*, 1:67–96, 1974. 2

[3] N. Bard, M. Johanson, N. Burch, and M. Bowling. Online implicit agent modelling. In *Autonomous Agents and Multiagent Systems (AAMAS)*, 2013. 2.2, 4.3

[4] S. Berry, J. Levinsohn, and A. Pakes. Automobile prices in market equilibrium. *Econometrica*, 63(4):841–890, July 1995. 1

[5] A. Blum and Y. Mansour. *Algorithmic Game Theory*, chapter Learning, Regret Minimization and Equilibria, pages 79–102. Cambridge University Press, 2007. 2.3.1

[6] S. Boyd and L. Vandenberghe. *Convex Optimization*. Cambridge University Press, 2004. A

[7] G. W. Brown. Iterative solution of games by fictitious play. In T. C. Koopmans, editor, *Activity Analysis of Production and Allocation*. Wiley, New York, 1951. 3.3.1

[8] N. Brown and T. Sandholm. Safe and nested subgame solving for imperfect-information games. In *Neural Information Processing Systems (NeurIPS)*, 2017. 5.4

[9] N. Brown and T. Sandholm. Libratus: The superhuman ai for no-limit poker. In *International Joint Conference on Artificial Intelligence (IJCAI)*, 2017. 5.4

[10] N. Brown and T. Sandholm. Superhuman ai for heads-up no-limit poker: Libratus beats top professionals. *Science*, 359:418–424, 2018. 5.4

[11] N. Brown, A. Lerer, S. Gross, and T. Sandholm. Deep counterfactual regret minimization. In *International Conference on Machine Learning (ICML)*, 2019. 5.5

[12] N. Burch, M. Johanson, and M. Bowling. Solving imperfect information games using decomposition. In *Conference on Artificial Intelligence (AAAI)*, 2014. 2.3.5, 5.4

[13] C. Camerer and T. H. Ho. Experience weighted attraction learning in normal form games. *Econometrica*, 67:827—874, 1999. 1

[14] A. Celli, A. Marchesi, G. Farina, and N. Gatti. No-regret learning dynamics for extensive-form correlated equilibrium. In *Neural Information Processing Systems (NeurIPS)*, 2020. 5.3

[15] G. Cesa-Bianchi, N. Lugosi. *Prediction, learning, and games.* Cambridge University

Press, 2006. C

[16] V. Conitzer and T. Sandholm. Complexity results about nash equilibria. In *International Joint Conference on Artificial Intelligence (IJCAI)*, 2003. 2.1

[17] M. Conlin and V. Kadiyali. Entry-deterring capacity in the texas lodging industry. *Journal of Economics and Management Strategy*, pages 167–185, 2006. 1

[18] D. Cooper and J. Van Huyck. Evidence on the equivalence of the strategic and extensive form representation of games. *Journal of Economic Theory*, 110(2):290–308, 2003. 3.3.1

[19] M. Costa-Gomes, V. Crawford, and B. Broseta. Cognition and behavior in normal-form games: an experimental study. Discussion paper 22, University of California San Diego, 1998. 3.3.1

[20] C. Daskalakis, P. W. Goldberg, and C. H. Papadimitriou. The complexity of computing a nash equilibrium. In *ACM Symposium on Theory of Computing*, STOC '06, pages 71–78, 2006. 2.1

[21] T. Davis, K. Waugh, and M. Bowling. Solving large extensive-form games with strategy constraints. In *Conference on Artificial Intelligence (AAAI)*, 2019.

[22] R. D'Orazio, D. Morrill, J. Wright, and M. Bowling. Alternative function approximation parameterizations for solving games: An analysis of f-regression counterfactual regret minimization. In *Autonomous Agents and Multi-agent Systems (AAMAS)*, 2020. 5.5

[23] M. Dudík and G. Gordon. A sampling-based approach to computing equilibria in succinct extensive-form games. In *Uncertainty in Artificial Intelligence (UAI)*, 2009. 5.3

[24] M. Dudík, S. J. Phillips, and R. E. Schapire. Maximum entropy density estimation with generalized regularization and an application to species distribution modeling. *Journal of Machine Learning Research*, 8:1217–1260, 2007. 2.3.7, 3.1.1

[25] I. Erev and G. Barron. On adaptation, maximization and reinforcement learning among cognitive strategies. *Psychological Review*, 112:912–931, 2005. 1

[26] I. Erev, E. Ert, and E. Yechiam. Loss aversion, diminishing sensitivity, and the effect of experience on repeated decisions. *Journal of Behavioral Decision Making*, 21:575–597, 2008. 1

[27] I. Erev, E. Ert, and A. E. Roth. A choice prediction competition for market entry games: An introduction. *Games*, 1:117–136, 2010. 3.3.2

[28] D. Foster and R. Vohra. Calibrated learning and correlated equilibrium. *Games and Economic Behavior*, 21:40–55, 1996. 2.3.1

[29] Y. Freund and R. E. Schapire. A decision-theoretic generalization of on-line learning and an application to boosting. *Journal Computer and System Sciences*, 55(1):119–139, 1997. 2.1.1

[30] S. Ganzfried and T. Sandholm. Game theory-based opponent modeling in large imperfect-information games. In *Autonomous Agents and Multiagent Systems (AAMAS)*, 2011. 2.2

[31] S. Ganzfried and T. Sandholm. Safe opponent exploitation. *ACM Transactions on Economics and Computation (TEAC)*, 3(2):8:1–28, 2015. 2.2

[32] S. Ganzfried, T. Sandholm, and K. Waugh. Strategy purification. In *Workshop on Computer Poker and Imperfect Information (AAAI)*, 2011.

[33] S. Ganzfried, T. Sandholm, and K. Waugh. Strategy purification and thresholding: Effective non-equilibrium approaches for playing large games. In *Autonomous Agents and Multi-agent Systems (AAMAS)*, 2012.

[34] X. A. Gao and A. Pfeffer. Learning game representations from data using rationality constraints. In *Uncertainty in Artificial Intelligence (UAI)*, 2010. 1

[35] A. Gilpin, T. Sandholm, and T. Sorensen. Potential-aware automated abstraction of sequential games, and holistic equilibrium analysis of texas hold'em poker. In *Conference on Artificial Intelligence (AAAI)*, 2007. 4.3

[36] J. K. Goeree and C. A. Holt. Ten little treasures of game theory and ten intuitive contradictions. *American Economic Review*, 91(5):1402–1422, 2001. 3.3.1

[37] G. Gordon. No-regret algorithms for online convex programs. In *Neural Information Processing Systems (NIPS)*, 2007. 2.3.8

[38] P. D. Grünwald and A. P. Dawid. Game theory, maximum entropy, minimum discrepancy, and robust bayesian decision theory. *Annals of Statistics*, 32:1367–1433, 2003. 2, 3.0.2

[39] R. Hanson. Combinatorial information market design. *Information Systems Frontiers*, 5 (1):107–119, 2003. 2.3.4

[40] S. Hart and A. Mas-Colell. A simple adaptive procedure leading to correlated equilibrium. *Econometrica*, 68(5):1127–1250, 2000. 2.3.1, 11

[41] E. Haruvy and D. Stahl. Equilibrium selection and bounded rationality in symmetric normal-form games. *Journal of Economic Behavior and Organization*, 62(1):98–119, 2007. 3.3.1

[42] E. Haruvy, D. Stahl, and P. Wilson. Modeling and testing for heterogeneity in observed strategic behavior. *Review of Economics and Statistics*, 83(1):146–157, 2001. 3.3.1

[43] J. Heinrich and D. Silver. Deep reinforcement learning from self-play in imperfect-information games, 2016. URL `https://arxiv.org/abs/1603.01121`. 5.5

[44] P. Henry, C. Vollmer, B. Ferris, and D. Fox. Learning to navigate through crowded environments. In *Robotics and Automation*, 2010. 1

[45] E. T. Jaynes. Information theory and statistical mechanics. *Physical Review*, 106(4): 620–630, May 1957. 2.3.4, 3.0.2

[46] M. Johanson. Robust strategies and counter-strategies: Building a champion level computer poker player. Master's thesis, University of Alberta, 2007. 4.3

[47] M. Johanson and M. Bowling. Data biased robust counter strategies. In *Proceedings of the International Conference on Artificial Intelligence and Statistics (AISTATS)*, 2009. 2.2

[48] M. Johanson, M. Zinkevich, and M. Bowling. Computing robust counter-strategies. In *Neural Information Processing Systems (NIPS)*, pages 1128–1135, 2008. 2.2, 2.3.8

[49] Michael Johanson. Measuring the size of large no-limit poker games. Technical Report TR13-01, Department of Computing Science, University of Alberta, 2013. 4.3

[50] Michael Johanson, Michael Bowling, Kevin Waugh, and Martin Zinkevich. Accelerating best response calculation in large extensive games. In *Proceedings of the Twenty-Second International Joint Conference on Artificial Intelligence (IJCAI)*, pages 258–265, 2011. 4.3

[51] S. Kakade, M. Kearns, J. Langford, and L. Ortiz. Correlated equilibria in graphical games. In *Proceedings of Electronic Commerce (EC)*, pages 42–47, 2003. 3.0.3

[52] M. Kearns, M. Littman, and S. Singh. Graphical models for game theory. In *Uncertainty in Artificial Intelligence (UAI)*, UAI '01, 2001. 3.0.3

[53] J. L. Kelly. A new interpretation of information rate. *Bell System Technical Journal*, 35 (4):917–926, 1956. 2.3.4

[54] K. Kitani, B. D. Ziebart, J. A. Bagnell, and M. Hebert. Activity forecasting. In *European Conference on Computer Vision (ECCV)*, 2012. 1

[55] R. Koster, J. Balaguer, A. Tacchetti, A. Weinstein, T. Zhu, Hauser. O, W. Duncan, L. Campbell-Gillingham, P. Thacker, M. Botvinick, and C. Summerfield. Human-centred mechanism design with democratic ai. *Nature Human Behaviour*, 2022. 5.1

[56] C. Kroer, K. Waugh, F. Kılınç-Karzan, and T. Sandholm. Faster first-order methods for extensive-form game solving. In *Conference on Economics and Computation (EC)*, 2015.

[57] C. Kroer, K. Waugh, F. Kılınç-Karzan, and T. Sandholm. Theoretical and practical advances on smoothing for extensive-form games. In *Conference on Economics and Computation (EC)*, 2017.

[58] C. Kroer, K. Waugh, F. Kılınç-Karzan, and T. Sandholm. Faster algorithms for extensive-form game solving via improved smoothing functions. *Mathematical Programming*, 179 (1):385–417, 2020.

[59] S. Kullback and R. A. Leibler. On information and sufficiency. *Annals of Mathematical Statistics*, 22(1):79–86, 1951. 2.2

[60] H. Li, K. Hu, S. Zhang, Y. Qi, and L. Song. Double neural counterfactual regret minimization. In *International Conference on Learning Representations (ICLR)*, 2020. 5.5

[61] N. Littlestone and M. Warmuth. The weighted majority algorithm. *Information and Computation*, 108:212–261, 1994. 2.1.1

[62] M. Littman. Markov games as a framework for multi-agent reinforcement learning. In *International Conference on Machine Learning (ICML)*, 1994. 5.3

[63] D. McFadden. The measurement of urban travel demand. *Journal of Public Economics*, 3(4):303–328, 1974. 1

[64] R. McKelvey and T. Palfrey. Quantal response equilibria for normal form games. *Games and Economic Behavior*, 10:6–38, 1995. 1, 3.3.1

[65] M. Moravčík, M. Schmid, K. Ha, M. Hladik, and S. Gaukrodger. Refining subgames in large imperfect information games. In *Conference on Artificial Intelligence (AAAI)*, 2016. 5.4

[66] M. Moravčík, M. Schmid, N. Burch, V. Lisý, D. Morrill, N. Bard, T. Davis, K. Waugh,

M. Johanson, and M. Bowling. Deepstack: Expert-level artificial intelligence in heads-up no-limit poker. *Science*, 356:508–513, 2017. 5.4, 5.5

[67] J. L. Myers and E. Sadler. Effects of range of payoffs as a variable in risk taking. *Journal of Experimental Psychology*, 60:306–309, 1960. 1

[68] J. Nash. Noncooperative games. *Annals of Mathematics*, 54:289–295, 1951. 2.1

[69] A. Nevo. Measuring market power in the ready-to-eat cereal industry. *Econometrica*, 69 (2):307–342, March 2001. 1

[70] A. Ng and S. Russell. Algorithms for inverse reinforcement learning. In *International Conference on Machine Learning (ICML)*, 2000. 1

[71] J. Nocedal. Updating quasi-newton matrices with limited storage. *Mathematics of Computation*, 35(151):773—782, 1980. 3.3.3

[72] L. E. Ortiz, R. E. Shapire, and S. M. Kakade. Maximum entropy correlated equilibrium. In *International Conference on Artificial Intelligence and Statistics (AISTAT)*, pages 347–354, 2007. 1, 2.3.8, 3.0.2

[73] M. J. Osborne and A. Rubinstein. *A Course in Game Theory*. The MIT Press, 1994. ISBN 0262650401. 2.1, 1

[74] A. Petrin. Quantifying the benefits of new products: The case of the minivan. *Journal of Political Economy*, 110(4), 2002. 1

[75] N. Ratliff, J. A. Bagnell, and M. Zinkevich. Maximum margin planning. In *International Conference on Machine Learning (ICML)*, 2006. 1, 2.3.2

[76] N. D. Ratliff, D. Silver, and J. A. Bagnell. Learning to search: Functional gradient techniques for imitation learning. *Autonomous Robots*, 27(1):25–53, 2009. 1

[77] B. W. Rogers, T. R. Palfrey, and C. F. Camerer. Heterogeneous quantal response equilibrium and cognitive hierarchies. *Journal of Economic Theory*, 144(4):1440–1467, 2009. 3.3.1

[78] S. Ross, G. J. Gordon, and J. A Bagnell. A reduction of imitation learning and structured prediction to no-regret online learning. In *International Conference on Artificial Intelligence and Statistics (AISTATS)*, 2011. 1, 5.3, 5.5

[79] M. Schmid, M. Moravčík, N. Burch, R. Kadlec, J. Davidson, K. Waugh, N. Bard, F. Timbers, M. Lanctot, Z. Holland, E. Davoodi, A. Christianson, and M. Bowling. Player of games, 2021. URL `https://arxiv.org/abs/2112.03178`. 5.4, 5.5

[80] J. Shi and M. Littman. Abstraction methods for game theoretic poker. In *International Conference on Computers and Games (CG)*, 2002. 1, 4.1

[81] N. Z. Shor. *Minimization methods for non-differentiable functions*. Springer-Verlag New York, Inc., 1985. ISBN 0-387-12763-1. 3.0.3

[82] D. Silver, J. A. Bagnell, and A. Stentz. Learning from demonstration for autonomous navigation in complex unstructured terrain. *International Journal of Robotics Research*, 29(1):1565–1592, October 2010. 1

[83] D. Silver, T. Hubert, J. Schrittwieser, I. Antonoglou, M. Lai, A. Guez, M. Lanctot, L. Sifre,

D. Kumaran, T. Graepel, T. Lillicrap, K. Simonyan, and D. Hassabis. A general reinforcement learning algorithm that masters chess, shogi, and go through self-play. *Science*, 362 (6419):1140–1144, 2018. 5.4, 5.5

[84] F. Southey, M. Bowling, B. Larson, C. Piccione, N. Burch, D. Billings, and C. Rayner. Bayes' bluff: Opponent modelling in poker. In *Conference on Uncertainty in AI (UAI)*, 2005. 4.4

[85] S. Srinivasan, M. Lanctot, V. Zambaldi, J. Pèrloat, K. Tuyls, R. Munos, and M. Bowling. Actor-critic policy optimization in partially observable multiagent environments. In *Neural Information Processing Systems (NeurIPS)*, 2018. 5.5

[86] D. Stahl and E. Haruvy. Level-n bounded rationality and dominated strategies in normal-form games. *Journal of Economic Behavior and Organization*, 66(2):226–232, 2008. 3.3.1

[87] D. Stahl and P. Wilson. Experimental evidence on players' models of other players. *Journal of Economic Behavior and Organization*, 25(3):309–327, 1994. 3.3.1

[88] D. Stahl and P. Wilson. On players' models of other players: Theory and experimental evidence. *Games and Economic Behavior*, 10(1):218–254, 1995. 3.3.1

[89] E. Steinberger. Single deep counterfactual regret minimization. In *Deep Reinforcement Learning Workshop (NeurIPS)*, 2019. 5.5

[90] E. Steinberger, A. Lerer, and N. Brown. Dream: Deep regret minimization with advantage baselines and model-free learning, 2020. URL `https://arxiv.org/abs/2006.10410`. 5.5

[91] J. Suzuki. Land use regulation as a barrier to entry: Evidence from the texas lodging industry. *International Economic Review*, 2010. 1, 3.3.3, 3.3.3

[92] J. von Neumann. Zur theorie der gesellshaftsspiele. *Mathematische Annalen*, 100:295–320, 1928. 2.1.1

[93] K. Waugh. A fast and optimal hand isomorphism algorithm. In *Workshop on Computer Poker and Imperfect Information (AAAI)*, 2013.

[94] K. Waugh and J. A. Bagnell. A unified view of large-scale zero-sum equilibrium computation. In *Workshop on Computer Poker and Imperfect Information (AAAI)*, 2014.

[95] K. Waugh, D. Schnizlein, M. Bowling, and D. Szafron. Abstraction pathologies in extensive games. In *International Conference on Autonomous Agents and Multiagent Systems (AAMAS)*, 2008. 4.3, 4.4

[96] J. Wright and K. Leyton-Brown. Beyond equilibrium: Predicting human behavior in normal-form games. In *Conference on Artificial Intelligence (AAAI)*, 2010. 1, 3.3.1

[97] B. D. Ziebart, A. Maas, J. A. Bagnell, and A. Dey. Maximum entropy inverse reinforcement learning. In *Conference on Artificial Intelligence (AAAI)*, 2008. (document), 1, 2.3.2, 3.0.1

[98] B. D. Ziebart, A. Maas, A. K. Dey, and J. A. Bagnell. Navigate like a cabbie: Probabilistic reasoning from observed context-aware behavior. In *International Conference on*

Ubiquitous Computing, 2008. 1

[99] B. D. Ziebart, J. A. Bagnell, and A. K. Dey. Modeling interaction via the principle of maximum causal entropy. In *International Conference on Machine Learning (ICML)*, 2010. 1

[100] M. Zinkevich, M. Johanson, M. Bowling, and C. Piccione. Regret minimization in games with incomplete information. Technical Report TR07-14, Department of Computing Science, University of Alberta, 2007. 4.4

[101] M. Zinkevich, M. Johanson, M. Bowling, and C. Piccione. Regret minimization in games with incomplete information. In *Neural Information Processing (NIPS)*, pages 905–912, 2008. 4.2, 5.3