Predicting and Understanding Initial Play*

Drew Fudenberg† Annie Liang‡

First version: November 14, 2017
This version: January 4, 2018

Abstract

We take a machine learning approach to the problem of predicting initial play in strategic-form games, with the goal of uncovering new regularities in play and improving the predictions of existing theories. The analysis is implemented on data from previous laboratory experiments, and also a new data set of 200 games played on Mechanical Turk. We use two approaches to uncover new regularities in play and improve the predictions of existing theories. First, we use machine learning algorithms to train prediction rules based on a large set of game features. Examination of the games where our algorithm predicts play correctly, but the existing models do not, leads us to introduce a risk aversion parameter that we find significantly improves predictive accuracy. Second, we augment existing empirical models by using play in a set of training games to predict how the models' parameters vary across new games. This modified approach generates better out-of-sample predictions, and provides insight into how and why the parameters vary. These methodologies are not special to the problem of predicting play in games, and may be useful in other contexts.

1 Introduction

In most game theory experiments, the distribution of play the first time participants play a game is not well approximated by the predictions of equilibrium analysis. Initial play does however have some regularities as, for example, shown by the fact that level-k thinking (Stahl and Wilson, 1994), the Poisson Cognitive Hierarchy model (Camerer, Ho and Chong, 2004), and the related models surveyed in Crawford, Costa-Gomes and Iriberri (2013) fit

*We are grateful to Vincent Crawford and Emanuel Vespa for very helpful comments and suggestions, and to Microsoft Research and National Science Foundation grant 1643517 for financial support.

†MIT

‡University of Pennsylvania

initial play reasonably well in many one-shot simultaneous-move games. We use machine learning to try to uncover new regularities, and to develop modifications that improve the predictions of existing theories.

Our approach is twofold: First, we construct a set of game features based on payoff matrices, and use machine learning algorithms to train prediction rules based on these features. By examining the games where our machine learning algorithm predicts play correctly but the existing models do not, we are able to identify regularities in play that are not captured by the existing theories. These regularities lead us to add a single parameter to the level-k model, which we find significantly improves predictive accuracy.

Second, we augment existing models by predicting parameter variation across games. For example, the Poisson Cognitive Hierarchy model (hereafter, the PCHM) has a single free parameter τ, which (loosely) describes the distribution of player sophistication. In the baseline model, the value of τ is assumed to be constant across games. We develop an extension of this model that groups games into classes based on their best-fit values of τ, and then uses training data to identify structural features of the game that predict its class. When given a new game, we then first predict the class of that game, and then use the corresponding class-specific value of τ for prediction of play. This modified approach generates better out-of-sample predictions than the underlying baseline for each of three models that we consider. These improvements on existing theories of initial play are of interest in their own right, but our two methodologies for use of machine learning to improve the predictiveness of theories are more general. Their success here suggests that that may be useful in other problem domains as well.

This paper considers two datasets of game play. The first is play in 86 lab games from six past lab experiments. Since the games in this data set were designed for certain experimental goals, their payoff matrices tend to possess "strategically interesting" features. In order to determine the robustness of our results to these features, we augment this set of games with a new dataset of play by Mechanical Turk subjects in 200 games with randomly drawn payoffs. Compared to the lab games, the games with random payoffs turn out to be more likely to be dominance solvable, more likely to include a strictly dominated action, and more likely to contain an action profile that is clearly best for both players. From prior work, we expect these differences to make initial play in the new games somewhat easier to predict, and indeed, this is what we find in the subsequent analysis.

We study two sorts of predictions tasks: predicting the action played—where error is minimized by predicting the modal action in each game—and predicting the distribution of actions. In both cases, we evaluate the performance of various prediction rules by their *completeness*, which we take to be the percentage of the possible improvement over random guessing (see Peysakhovich and Naecker (2017) and Kleinberg, Liang and Mullainathan

(2017)).

We begin with the task of predicting the realized action, which is a classification problem, and has the advantage that the associated algorithms are easier to interpret. The modal action turns out to be level 1 in 72% of the lab games; as a result, simply predicting the level 1 action performs well, achieving 80% of the attainable improvement over random guessing. We find that the best-performing version of PCHM, which extends the level k model by assuming that types best respond to a Poisson distribution over lower level types, is equivalent to the level 1 model when its free parameter τ is estimated from training data. Thus, the PCHM achieves the same performance as level 1 on this task. In our set of random games, the modal action is level 1 in 87% of the games, and again the level 1 model and PCHM make the same predictions; here, they achieve a completeness measure of 88%.

We then create a large set of game features, including indicators for whether each action satisfies certain strategic properties (level 1, level 2, part of a Nash equilibrium, etc.), and train decision trees to use these features to predict play. When predicting the lab data, we find that the best decision tree with two decision nodes reproduces the level 1 prediction, but the best out-of-sample predictions are made by a tree with three decision nodes. We then examine the 9 (out of 86) lab games where the modal action is not level 1, but is correctly predicted by the best decision tree. It turns out that in each of these games, there is an action whose average payoffs closely approximate the level 1 action, and which additionally yields lower variation in possible payoffs. Players are more likely to choose this "almost" level 1 action over over the actual level 1 action.

One explanation for this behavior is that players maximize a concave function over game payoffs. Motivated by this finding, we modify the level 1 prediction by assuming that agents have utility function for money payoffs of $u(x) = x^\alpha$. We find that estimating the single parameter α substantially improves out-of-sample prediction error, and improves upon the predictive accuracy of our feature-based prediction rule. This suggests that atheoretical prediction rules fit by machine learning algorithms can be used to help craft interpretable parametric models that fit better than the current state of the art. Extending the level 1 model in this way also generates better predictions in the random games (as compared to the benchmark of the level 1 model and PCHM), although here the decision tree model performs slightly worse.

We then turn to the (more frequently studied) problem of predicting the distribution of play. As a naive baseline, we consider prediction of uniform play, where each action is played 1/3 of the time.[1] The PCHM achieves a significant improvement over this baseline, achieving 50% of the possible improvement in predicting the lab data set, and 78% of the possible

[1] We also consider prediction of uniform play over the actions consistent with Nash equilibrium; this turns out not to improve upon the naive baseline.

improvement in predicting the Mechanical Turk data set. Several proposed variations do better still: adding a risk aversion parameter improves the predictive accuracy of the PCHM, as does replacing the assumption of exact maximization with logit responses, as in Stahl and Wilson (1994, 1995) and Leyton-Brown and Wright (2014). The latter approach (which we refer to as LPCHM) is improved further if we assign probability zero to level 0 players in the population (while allowing them to exist in the perceptions of players of higher levels). Notice that although adding additional parameters always improves in-sample fit, it need not reduce out-of-sample predictive error.[2] These methods attain 61-77% of the achievable improvement over guessing at random, and 78-84% of the possible improvement in predicting the Mechanical Turk data set

We then explore a new way to use game features for prediction. Specifically, we note that the distributional predictions of the PCHM and its variants are sensitive to the choice of the parameter τ (the Poisson rate parameter). Moreover, as has been noted in prior work (Camerer, Ho and Chong, 2004), the best-fitting value of τ varies substantially from game to game. Variation in the best-fit value of τ across games suggests that better predictions can be made by allowing for heterogeneity in τ. To accommodate parameter heterogeneity without eliminating the PCHM's predictive content, we propose a way to estimate the appropriate value of τ for a given game from other data. Specifically, we learn a predictive function from game features to best-fit values of τ, and use this function to predict heterogeneity in values of τ in the out-of-sample games. This technique turns out to appreciably improve prediction for both datasets we consider. Moreover, examining a simple decision tree (with two decision nodes) used to predict τ helps us to understand which features correlate with variation of the best-fit values of τ. We find that τ tends to be high in games that have a relatively "obvious" action, meaning that either the level 1 action is a much better response to the uniform distribution than any other action, or that some action profile gives a particularly high payoff. We repeat this procedure of introducing parameter heterogeneity to the LPCHM and the LPCHM without level 0 players, and again obtain improvements in predictive accuracy.

Finally we consider a new kind of data for making predictions: We incentivize MTurk participants to predict play by other individuals. Specifically, subjects were shown a set of games and asked, for each game, to pick the action that they thought was most frequently played. We find that in most cases the "naive crowd prediction rule"—which predicts in the two tasks, respectively, the modal crowd prediction and the distribution of crowd

[2]Throughout, when we say *in-sample*, we mean that the data used for training and testing are the same. Increasing the flexibility of a model always allows for higher in-sample fit. When we say *out-of-sample*, we mean that different data is used for training and for testing. Increasing the flexibility of a model need not result in higher out-of-sample fit; in particular, more complex models are prone to overfitting to the training data.

predictions—does better than the PCHM. Notice that the payoff matrices are not an input into the crowd rule: any information about the game itself is filtered through the perception of the crowd. These results point to the potential for alternative feature sets, such as human inputs, to further improve prediction.

1.1 Background Information and Related Work

As the Crawford, Costa-Gomes and Iriberri (2013) survey shows, there is an extensive literature on initial play in matrix games. Most if not all of these papers use some variant of "cognitive hierarchies" in that their starting point is the specification of a "level 0" or unsophisticated player, who most often is assumed to play a uniform distribution. The various cognitive hierarchy models then use the level 0 type to build up a richer specification of play. The simplest such model is "level 1," which assumes that the whole population plays a best response to level 0. This is too stark a model to be a good fit for the observed distribution of play in most games, but we we will see it does a fairly good job of predicting the most likely (i.e. modal) action.

Work on initial play has had the twin goals of offering an alternative to Nash equilibrium as a way of predicting play and of providing a model of how people think in these settings. Our paper focuses on the first of these goals, so we search for models that are relatively simple and portable, which led us to focus on variants of the PCHM. For this reason, this paper is closest to the improvements of the PCHM proposed by Chong, Ho and Camerer (2016) and Leyton-Brown and Wright (2014): Chong, Ho and Camerer (2016) defines the level 0 player to randomize only over actions that are "never-worst" and Leyton-Brown and Wright (2014) replaces the specification of level 0 from uniform play to a weighted linear model based on five game features. Sgroi and Zizzo (2009) and Hartford, Wright and Leyton-Brown (2016) use deep learning to predict play, but do not share our focus on deriving conceptual lessons that can improve existing models. Our paper is also similar in spirit to Fragiadakis, Knoepfle and Niederle (2016), which tries to identify the subjects that have regularities in play not captured by cognitive hierarchies.

There is also an extensive literature on the prediction of play in repeated interactions with feedback, where learning plays an important role; see e.g. Erev and Roth (1999), Crawford (1995), Cheung and Friedman (1997) and Camerer and Ho (1999). In this paper, we consider only initial play, leaving open the question of how machine learning methods can contribute to our understanding of play in repeated settings.[3]

Costa-Gomes and Weizsacker (2007) compare elicited beliefs over play with play itself, and find that players both approximately act like level 1 players and also believe others to

[3] Camerer, Nave and Smith (2017) uses machine learning to predict play in a repeated bargaining game.

act like level 1 players. This is related to our section 6.1 on crowd predictions, where we show that reported beliefs can be used as inputs into predicting play. Finally, the ability of untrained human subjects to predict economic behaviors is demonstrated by DellaVigna and Pope (2017) in a different context (forecasting the efficacy of different experimental incentives).

2 Data and Experiments

Throughout the paper we consider only three-by-three matrix games. The set of payoff matrices is identified with $G = \mathbb{R}^{18}$, and we use g to mean a typical payoff matrix. The set of row player actions is A_{row}, the set of column player actions is A_{col}, and the set of action profiles is $A = A_{\text{row}} \times A_{\text{col}}$. Finally, we use $u_{\text{row}} : A \to \mathbb{R}$ and $u_{\text{col}} : A \to \mathbb{R}$ to mean the row player's and the column player's payoffs respectively.

Below, we describe two data sets of game play. Our first data set, presented in Section 2.1, consists of play in several past lab experiments. Since the games in this data set were designed for certain experimental goals, they are not a random sample but tend to possess strategically interesting features. In order to determine the robustness of our results to these designs, we augment the lab games with a novel data set, described in Section 2.2, that consists of play by Mechanical Turk subjects in games with randomly drawn payoffs.

2.1 Lab Data

Our data on play in laboratory experiments consists of all 3x3 matrix games in the data set collected (and used) by Kevin Leyton-Brown and James Wright (see e.g. Leyton-Brown and Wright (2014)). This data includes 40-147 observations of play in each of 86 symmetric three-by-three normal-games. Table 1 below lists the number of games, as well as the number of observations of play, from each paper.

Paper	Games	Total # of Observations
Stahl and Wilson (1994)	10	400
Stahl and Wilson (1995)	12	576
Haruvy, Stahl and Wilson (2001)	15	869
Haruvy and Stahl (2007)	20	2940
Stahl and Haruvy (2008)	18	1288
Rogers, Palfrey and Camerer (2009)	17	1210
Total	**86**	**6887**

Table 1: Original sources for the lab play data

The subject pool and payoff scheme differ across the six papers, but all of them use anonymous random matching without feedback: participants play each game only once, are not informed of their partner's play, and do not learn their own payoffs until the end of the session. Since our data set is limited to symmetric games, we label all observed actions (whether chosen by a column player or a row player) as row-player actions.

There is substantial variation in the distribution of play across games. For example, the fraction of subjects who chose the most frequently chosen action (the modal action) varies from 39.19% to 94.56%. Relatedly, there is large variation in how far the observed distribution differs from a uniform distribution over actions. The entropy of the observed distribution of play[4] varies from 0.5 (close to degenerate) to 1.09 (close to uniform). See below for the distributions of both measures.

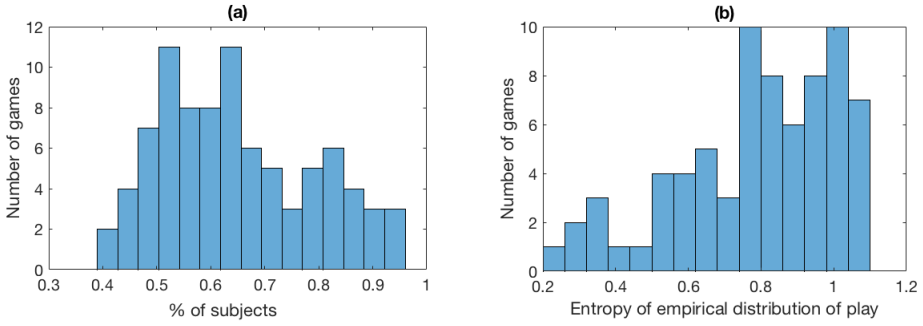


Figure 1: In our lab games: (a) % subjects who chose the modal action; (b) entropy of the distribution of play.

As an illustration, we show below the two games that achieved highest and lowest values according to these measures. The game with the lowest modal frequency (also the game whose distribution has the highest entropy) is the following:

	a_1	a_2	a_3	Frequency
a_1	21,21	93,13	45,29	32.43%
a_2	13,93	69,69	53,53	28.38%
a_3	29,45	53,53	61,61	39.19%

Although most subjects chose action a_3, play is close to uniform. In contrast, in the game with the highest modal frequency (also the game whose distribution has the lowest entropy), 95% of subjects chose the same action:

[4]The entropy of frequency vector (p_1, p_2, p_3) is given by $\sum_{i=1}^{3} p_i \cdot \log(p_i)$.

	a_1	a_2	a_3	Frequency
a_1	35,35	35,25	70,0	2.72%
a_2	25,35	55,55	100,0	94.56%
a_3	0,70	0,100	60,60	2.72%

We point out that each of these 86 games was designed for certain experimental goals. For example, Stahl and Wilson (1994) write that: "Ten symmetric (3×3 games) were selected for a variety of characteristics: three were strict dominance solvable, two were weak dominance solvable, six had unique pure-strategy symmetric Nash equilibrium, while two had unique mixed-strategy Nash equilibria." To determine the robustness of our findings to design features such as these, we augment the laboratory data with a large data set of play in games with randomly generated payoffs, which we now describe below.

2.2 Random Games

We randomly generated two hundred payoff matrices from a uniform distribution over $\{10, 20, \ldots, 90\}$[18]. This scale was chosen to match the lab experiments described above, although unlike in the previous section, the randomly generated games are not symmetric. We presented each of 551 Mechanical Turk subjects with a random subset of fifteen games, and asked them to play as the row player.[5]

Subjects were incentivized by the following payment scheme. On top of a base payment of \$0.35, subjects were told that one of the fifteen games would be chosen at random, and their action would be matched with another subject who had been asked to play as the column player.Their joint moves determined payoffs that were multiplied by \$0.01 to determine the subject's bonus winnings (ranging from \$0.10 to \$0.90). Subjects spent on average 7 minutes on the task, and the average payment was \$0.93, or \$8.14 an hour.[6] The minimum payment was \$0.45 and the maximum payment was \$1.25; the standard deviation of payments was \$0.23. The complete set of instructions can be found in Appendix D.[7]

To understand how the randomly generated games differ from the lab games, we compare various summary statistics of the two sets of payoff matrices. Specifically, we consider: the *number of pure strategy Nash equilibria* and the *number of undominated (row player) actions*. Figure 2 shows that relative to the random games, the games played in lab experiments are more likely to have a higher number of pure-strategy Nash equilibria and a higher number of rationalizable actions. These differences are large, suggesting that the set

[5]Each game was shown to 10-32 subjects, and the average number of responses per game was 20.02.

[6]This is a typical hourly wage for MTurk.

[7]In addition to eliciting play, we asked for subjects to volunteer a free-form description of how they made their decisions. Example answers can be found in Section C of the Online Appendix.

of lab games is indeed different from what we would expect in a random sample.

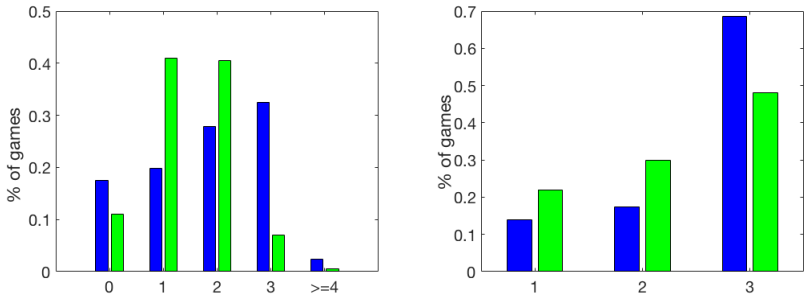


Figure 2: (a) Percentage of games with zero, one, two, three, or at least four pure strategy Nash equilibria (*blue*—lab games; *green*—random games); (b) Percentage of games with one, two, or three actions surviving iterated elimination of (pure-strategy) dominated actions (*blue*—lab games; *green*—random games).

We consider several additional summary statistics in Appendix A, which further reinforce this point. Relative to the randomly generated games, lab games have payoffs with larger variation (higher variance, a larger maximum payoff, and a smaller minimum payoff). In addition, lab games are less likely to be dominance solvable, less likely to include a strictly dominated action, and less likely to contain an action profile that is clearly best for both players (by various measures that we define). From prior work, we expect these differences to make initial play in the new games somewhat easier to predict, and indeed, this is what we find in the subsequent analysis. (Ideally we would use a sample of games that corresponds to the distribution of games that people face in the field, but we do not know what that distribution would be.)

As we saw with the lab data, there is substantial variation in how subjects played across the different (randomly generated) games. The percentage of subjects who chose the modal action varies from 36.84% to 1, and the entropy of the observed distributions of play varies from 0.22 to 1.09. See Figure 3 for the distributions of both measures across games.

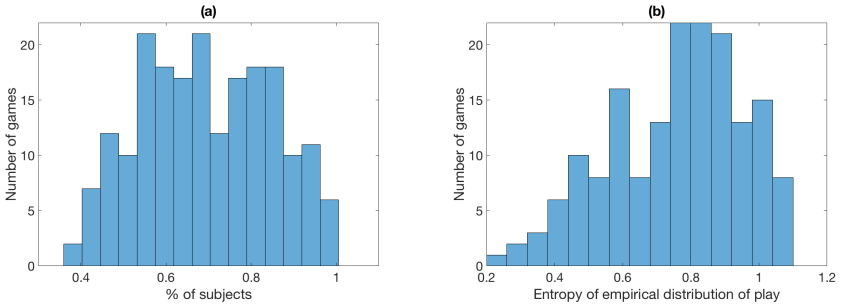


Figure 3: In our random games: (a) % of subjects who chose the modal action; (b) entropy of the observed distribution of play.

3 Prediction Tasks and Measures of Performance

We use two prediction tasks to evaluate how well can we predict play. The first prediction task is a classification problem: given a specific instance of play of a fixed game, we seek to predict which action the row player chose. For this problem, a prediction rule is a mapping from games to row player actions

$$f : G \to A_{\text{row.}}$$

An observation is a pair (g_i, a_i) where g_i is the game played in observation i and a_i is the action that the row player chose. Given a set of n observations $\{(g_i, a_i)\}_{i=1}^{n}$, we measure the error of prediction rule f using the *misclassification rate*

$$\frac{1}{n}\sum_{i=1}^{n} \mathbb{1}(\mathbf{a}_i \neq f(g_i)),$$

i.e. the fraction of observations in which the predicted action is not the action that was chosen in the given instance of play.

The *ideal* prediction rule for a given test set assigns to each game the (actual) modal action in that game. Note that this rule will be be imperfect (i.e. have a strictly positive misclassification rate) unless the empirical distribution is a point mass in all games. The error achieved by this ideal rule is a lower bound on the best achievable error, but it need not be a tight bound. This is because the prediction rule uses knowledge of the data to be predicted, and so its error is not out-of-sample. The *naive* prediction rule guesses uniformly at random; this guarantees a misclassification rate of 2/3, which is also the best possible rate when the empirical distributions are uniform.

In the second prediction task, we seek to predict the distribution over (row player)

actions in each game. A prediction rule for this problem is a mapping from payoff matrices to distributions over row player actions:

$$h : G \to \Delta(A_{\text{row}}).$$

An observation is a pair $(g_i, \mathbf{p}_i)$, where g_i is a (distinct) game and $\mathbf{p}_i$ is the distribution over (row-player) actions observed in that game. Error is assessed using *mean-squared error*

$$\frac{1}{n}\sum_{i=1}^{n}\frac{1}{3}\|\mathbf{p}_i - \hat{\mathbf{p}}_i\|^2,$$

where $\mathbf{p}_i$ is the frequency vector over actions observed for game i, $\hat{\mathbf{p}}_i$ is the predicted frequency vector, the norm is the Euclidean norm, and n is the number of games.

In this problem, the *ideal* prediction rule predicts the actual realized frequency vector for each game, and so has zero mean-squared error; the *naive* prediction of the uniform distribution has a mean-squared error that depends on the data. Again, the ideal rule is a lower bound that may not be attainable as it uses knowledge of the test data. In what follows, we use the gap between the errors of the naive and ideal prediction rules to calibrate the success of various prediction rules. Throughout, we separate the instances of play in lab games and random games and assess error for each dataset separately.

All of the errors that we report (unless explicitly stated otherwise) are tenfold cross-validated prediction errors. For the first prediction task, this means that we divide the games into ten folds, use all observations of play associated with games in nine of the folds for training, and use the observations of play associated with games in the remaining fold for testing. The reported error is averaged across the different choices of test fold. For the second prediction task, we divide the games into ten folds, use all games in nine of the folds for training, and use all games in the remaining fold for testing; again, we report the average prediction error across choices of test set.[8]

The cross-validation approach described above is a more stringent test than a related cross-validation exercise, in which observations are pooled across games before being subdivided into folds (see e.g. Leyton-Brown and Wright (2014)). With this alternative method, it is very likely that the training data contains observations of play from every game in the data set, since there are only 86 games but 6887 observations of play. It is then substantially easier to learn the modal action in each game in our data.[9] In contrast, when

[8] All standard errors for the cross-validated exercises are computed as the standard deviation of prediction errors across choices of test sets, divided by $\sqrt{K}$, where $K = 10$ is the number of iterations. See Hastie, Tibshirani and Friedman (2009) for a reference.

[9] The prediction rule learned in this way may, however, lead to poorer prediction of games outside of our data set.

using the cross-validation approach above, the test data and training data are restricted to include observations of play in different games. We do report results for this alternative "observation-level" cross-validation in Section A.2 of the Online Appendix. As expected, all prediction errors are higher; we find, however, that the qualitative results that we report below are unchanged.

4 Predicting the Action Played

4.1 Approaches

We evaluate and compare several approaches for the problem of predicting the realized action in a given instance of play. We first consider approaches based on Nash equilibrium, the level-k models of Stahl and Wilson (1995), and the Poisson Cognitive Hierarchy model of Camerer, Ho and Chong (2004).

Uniform Nash. We evaluate a prediction rule based on the hypothesis that play is a uniform distribution over the set of actions that are consistent with a pure-strategy Nash equilibrium.[10] Formally, define the set of actions $a_i \in A_{\text{row}}$ such that (a_i, a_j) is a Nash equilibrium for some $a_j \in A_{\text{col}}$, and predict at random from this set.

Level 1. Following Stahl and Wilson (1994, 1995), define a player to be "level 0" if he randomizes uniformly over his actions, so that his distribution of play is given by

$$P_0(a_i) = 1/3 \qquad \forall\ i \in 1, 2, 3$$

The level 1 player best responds to a level 0 player, and the level 1 prediction rule assigns to each game its level 1 action. When the level 1 prediction is not unique, we randomize over the set of level 1 actions.[11]

Poisson Cognitive Hierarchy Model (PCHM). Following Camerer, Ho and Chong (2004), define level 0 and level 1 as above and define the play of level k players, $k \geq 2$, to be the best responses to a perceived distribution

$$g_k(h) = \frac{\pi_\tau(h)}{\sum_{l=0}^{k-1} \pi_\tau(l)} \qquad \forall\ h \in \mathbb{N} \tag{1}$$

[10] This prediction rule is considered previously in Leyton-Brown and Wright (2014).
[11] The level 1 prediction is unique in all of the lab games, but not in all of the random games.

over opponent levels, where π_τ is the Poisson distribution with rate parameter τ. The predicted distribution over actions is based on the assumption that the actual proportion of level k players in the population is $\pi_\tau(k)$. We predict the modal action according to this aggregated distribution. Throughout, we take τ to be a free parameter and estimate it from the training data, allowing different values of τ for each dataset.

Prediction rules based on game features. In addition to the methods described above, we introduce prediction rules based on features that describe strategic properties of the game matrix. Specifically, for each action, we define an indicator variable for whether the action has each of the following properties: whether it is part of a Nash equilibrium, whether it is part of an action profile that maximizes the sum of player payoffs (*altruistic* in Costa-Gomes, Crawford and Broseta (2001) and *efficiency* in Leyton-Brown and Wright (2014)), whether it is part of a Pareto dominant Nash equilibrium, whether it is level k (for each $k \in \{1, 2, \ldots, 7\}$), and whether it allows for the highest possible row player payoff (*optimistic* in Costa-Gomes, Crawford and Broseta (2001) and *maxmax* in Leyton-Brown and Wright (2014)). We include additionally a score feature for how many of the above properties each action satisfies. The higher an action's score, the more compelling a choice it is.

We use a *decision tree algorithm* to learn predictive functions from these features to outcomes. Decision trees recursively partition the feature space and learn a (best) constant prediction for each partition element. We consider trees which use only a single feature to determine the split at each node, and use the standard approach of building up the decision tree one node at a time using a greedy algorithm. Thus, the first node is the best single split, the second node is the best second split conditional on the first, and so forth.

4.2 Results

Table 2 reports the misclassification rates and completeness measures for these prediction rules when predicting the distribution of play in the lab data, where the error attained by the naive prediction rule is set to 0 and the error attained by the ideal rule is set to 1.

When evaluating the PCHM, the best-performing τ (estimated from training data) turns out to correspond to predicting the level 1 action; thus, we report the performance of these two models together.[12,13]

[12] We find that prediction error is minimized at all values of τ in the interval $(0, 1.25]$. The values of τ in this range all yield prediction of the level 1 action for the games in our data sets.

[13] PCHM (and other variants we consider) better fit the *distribution* of actions, but we defer this discussion to Section 5.

	Error	Completeness
Naive benchmark (guess at random)	0.6667	0%
Best constant prediction (always guess a_1)	0.6013	18.96%
	(0.0797)	
Uniform Nash	0.5507	33.66%
	(0.0055)	
Level 1/PCHM	0.3889	80.55%
	(0.0079)	
Prediction based on game features	0.3652	87.42%
	(0.0057)	
Ideal prediction	0.3218	100%

Table 2: Predicting the realized action in lab data

We find that the uniform Nash prediction rule improves slightly over guessing at random, and achieves a completeness measure of approximately 34%. The level 1 model achieves a substantially larger improvement, increasing completeness to 81%. Finally, the prediction rule based on game features does better still, achieving a completeness of 87%.

We now ask what the prediction rule based on game features looks like, and why this decision tree outperforms the level 1 model. As a first pass, we examine the best decision tree under a severe parsimony constraint: use of only two decision nodes. This "2-split" decision tree turns out to reproduce the level 1 model: the predicted action is the action that best responds to a uniform distribution over column player actions. In this sense, the level 1 model is the best "simple" prediction model.[14,15]

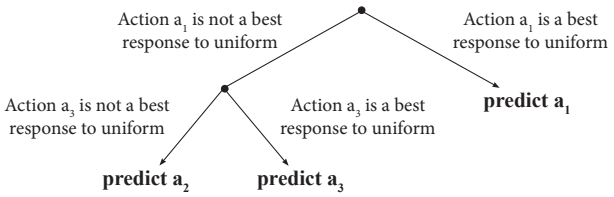


Figure 4: Best 2-split decision tree for predicting the realized action in lab data

[14]This statement should be interpreted with respect to the features that we have defined. It may be that there is a new feature, outside of our set, that would allow for an even more predictive 2-split decision tree.

[15]The selection of level 1 features is robust to the choice of prediction task: in the problem of predicting the realized distribution of play in lab data, the best 2-split decision tree again picks out the level 1 features (see Section A.1 of the Online Appendix).

As we allow for additional complexity by increasing the number of decision nodes n, the best n-split decision tree builds on the level 1 model. Large values of n quickly result in overfitting. The decision tree with the best out-of-sample prediction (shown below) has $n = 3$, and appends a single additional criterion to the level 1 model: it agrees with the level 1 model except that even if action a_1 is level 1, it is not predicted if the number of reasons to choose a_2 are sufficiently large. In this case, action a_2 is predicted instead.

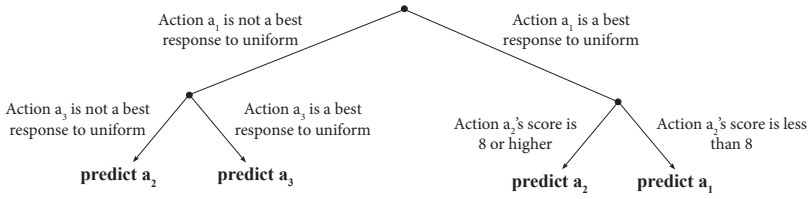


Figure 5: Best decision tree for predicting the realized action in lab data

There turn out to be nine games in which the modal action is correctly predicted by the decision tree above, but incorrectly predicted by the level 1 model. These games are displayed in Figure 6 where the level 1 action is represented in bold and the prediction of the game-based decision tree is represented in italics.

Studying these games reveals a common feature: In each of the games, some action that is not level 1 yields an expected payoff against uniform play that is comparable to the level 1 payoff, and moreover has lower variation in possible row payoffs. Consider for example the first game in Figure 6. Action a_3 is the level 1 action in this game, but the expected payoff to action a_1 is not much smaller (42 vs. 48.33), and choice of action a_1 yields significantly lower variation in possible row player payoffs.[16] In our data, more subjects choose action a_1 than action a_3. In fact, this behavior appears in all of the nine games shown above: subjects preferred actions that were "almost level 1" when those actions yielded lower variation in payoffs.

With knowledge of this regularity, we can modify the level 1 model to take it into account. Specifically, because the departure from level 1 behavior is consistent with a risk averse utility function over payoffs, we suppose that dollar payoffs u are transformed under $f(u) = u^\alpha$, which adds one parameter to PCHM. The standard assumption that players are Expected Utility maximizers is nested as $\alpha = 1$. This revised model has two free parameters (τ, α), and as before, we can estimate the these free parameters on training

[16]Depending on which action the column player takes, the row player will receive any of 43, 91, and 11 if he (the row player) chooses a_3, compared to 47, 51 and 28 if he (the row player) chooses a_1

	a_1	a_2	a_3
a_1	*47,47*	*51,44*	*28,43*
a_2	44,51	11,11	43,91
a_3	**43,28**	**91,43**	**11,11**

	a_1	a_2	a_3
a_1	*45,45*	*50,41*	*21,40*
a_2	41,50	0,0	40,100
a_3	**40,21**	**100,40**	**0,0**

	a_1	a_2	a_3
a_1	**0,0**	**35,55**	**100,30**
a_2	*55,35*	*40,40*	*20,0*
a_3	30,100	0,20	0,0

	a_1	a_2	a_3
a_1	15,15	0,0	0,100
a_2	*0,41*	*90,90*	*10,0*
a_3	**100,0**	**0,21**	**20,20**

	a_1	a_2	a_3
a_1	**20,20**	**30,40**	**100,30**
a_2	*40,30*	*40,40*	*60,0*
a_3	30,100	0,60	40,40

	a_1	a_2	a_3
a_1	1,1	0,10	0,100
a_2	*10,0*	*90,90*	*10,5*
a_3	**100,0**	**5,10**	**20,20**

	a_1	a_2	a_3
a_1	**35,35**	**39,47**	**95,40**
a_2	*47,15*	*51,51*	*67,15*
a_3	40,100	15,67	47,47

	a_1	a_2	a_3
a_1	10,10	10,15	10,100
a_2	*15,10*	*80,80*	*15,0*
a_3	**100,10**	**0,15**	**30,30**

	a_1	a_2	a_3
a_1	**25,25**	**30,40**	**100,31**
a_2	*40,30*	*45,45*	*65,0*
a_3	31,100	0,65	40,40

Figure 6: The most frequently played action (in *italics*) is predicted by the decision tree. The level 1 action is in **bold**.

data and evaluate the estimated model out-of-sample.

Table 3 compares the prediction error of this modified PCHM with the original model.[17] We find that introduction of risk aversion reduces prediction error substantially, in fact improving upon the prediction error of the best decision tree.

Table 4 presents the analogous results for MTurk. The absolute prediction errors are lower for prediction of the MTurk data under each of the approaches, as anticipated in Section 2.2, but most results are qualitatively similar. Specifically, we again find that the PCHM and level 1 predictions coincide, and that the best 2-split decision tree generates the level 1 prediction. In contrast to the lab data, here the level 1 model outperforms the best decision tree (and also achieves a relatively high completeness measure of 88%). [18] Despite

[17] The estimated values of the free parameters are $\tau = 1$ and $\alpha = 0.6250$.

[18] Notice that although the level 1 model can always be reproduced by the decision tree algorithm given the set of features we have defined, the estimated tree varies depending on the training data. Table 4 thus

	Error	Completeness
Level 1/PCHM	0.3889	80.55%
	(0.0079)	
Prediction rule based on game features	0.3652	87.42%
	(0.0057)	
Level 1 with Risk Aversion	0.3642	87.71%
	(0.0093)	

Table 3: Introduction of risk aversion reduces prediction error.

the strong performance of the level 1 model, adding a single parameter for risk aversion again yields an improvement in prediction: the level 1 model with risk aversion attains 91% of the achievable improvement over random guessing.

	Error	Completeness
Naive benchmark (guess at random)	0.6667	0%
Best constant prediction (always guess a_2)	0.6482	4.87%
	(0.0137)	
Uniform Nash	0.4722	51.21%
	(0.0075)	
Level 1/PCHM	0.3323	88.05%
	(0.0065)	
Prediction based on game features	0.3430	85.23%
	(0.0050)	
Level 1 with Risk Aversion	0.3211	91%
	(0.0046)	
Ideal prediction	0.2869	100%

Table 4: Predicting the realized action in MTurk data

5 Predicting the Distribution of Play

Now we turn from the task of predicting the most likely action to the harder problem of predicting the empirical distribution over actions.

says that it would be better to simply force the decision tree to be the level 1 model, instead of giving it the flexibility to learn alternative models from our feature set. Note also that there may well be other feature sets and other learning algorithms that would do better than the level 1 model here.

5.1 Approaches

Our baseline model here is the one-parameter PCHM described in Section 4.1. We consider several variations on this model in Section 5.1.1 that add one or two parameters, as well as simpler models that assume only level 1 behavior. Our main innovation, which we discuss in Section 5.1.2, is to permit heterogeneity in the PCHM parameter τ across games. We use structural features of the payoff matrix to predict whether the best-fit value of τ for that game is likely to be high or low, and estimate PCHM separately for the two classes of games (high- and low-τ).

5.1.1 Variations on Existing Approaches

We consider several modifications of existing theories of initial play.

PCHM with Risk Aversion. Motivated by the success of risk aversion in improving the prediction of level 1, we add risk aversion to the PCHM by transforming payoffs to $f(u) = u^\alpha$. The standard version of PCHM is returned for parameter choice $\alpha = 1$. This approach has two free parameters τ and α, which we estimate on the training data and test out-of-sample.

PCHM with Logit Best Response (LPCHM). Following Stahl and Wilson (1994, 1995) and Leyton-Brown and Wright (2014), we replace the assumption of exact maximization with a logit best response. As before, define

$$P_0^r(a_i) = 1/3 \quad \forall\, a_i \in A_{\text{row}} \qquad\qquad P_0^c(a_j) = 1/3 \quad \forall\, a_j \in A_{\text{col}}$$

to be the distribution of play by a level 0 row player and a level 0 column player (respectively). Then for each level $k \geq 1$, recursively define

$$q_k^r(a_j) = \frac{1}{\sum_{h=0}^{k-1} \pi_\tau(h)} \sum_{h=0}^{k-1} \pi_\tau(h) P_h^c(a_j) \quad \forall\, a_j \in A_{\text{col}} \tag{2}$$

$$U_k^r(a_i) = \sum_{j=1}^{3} q_k^r(a_j) u_{\text{row}}(a_i, a_j) \quad \forall\, a_j \in A_{\text{row}} \tag{3}$$

$$P_k^r(a_i) = \frac{e^{\lambda U_k^r(a_i)}}{\sum_{a_i' \in A_{\text{row}}} e^{\lambda U_k^r(a_i')}} \quad \forall\, a_j \in A_{\text{row}} \tag{4}$$

and symmetrically define q_k^c, U_k^c, and P_k^c. The object q_k^r in (2) is the perceived distribution over opponent actions by a row player of level k; the expression for $U_k^r(a_i)$ in (3) is the expected row payoff to action a_i when the column player is level k; and the object P_k^r in (4) is the distribution over actions chosen by a row player of level k.

The key difference from PCHM is that players do not choose the action that maximizes payoffs with probability 1, but instead put decreasing but positive weight on actions that yield successively lower expected payoffs. The parameter λ controls the "accuracy" of the best response: as $\lambda \to \infty$, play converges to probability 1 on the best response (returning the PCHM), and as $\lambda \to 0$, play converges to a uniform distribution over actions. This model, like the PCHM, assume that players correctly forecast the play of all lower types. We suppose that the logit parameter λ is constant across all players.

As before, the predicted distribution over actions is found by supposing that the true proportion of level k players is given by $\pi_\tau(k)$, and aggregating the corresponding distributions of play. This approach has two free parameters τ and λ, which we estimate on the training data and test out-of-sample.

LPCHM, No Level-0 Players. When fitting the standard PCHM to data, level 0 players not only determine the play of the level 1 players but also capture any randomizations (e.g. from errors or from payoff shocks). We separated these roles in the LPCHM by assuming logit best replies, which explicitly builds in imperfect maximization by higher types. Nevertheless, motivated by the suggestion that there are no true level 0 players in the population (see e.g. Crawford, Costa-Gomes and Iriberri (2013)), we consider a further modification to remove level 0 players altogether.

Specifically, we construct the behavior of level-k players as in LPCHM, but fix the proportion of level 0 players in the population to be 0, reweighting the proportion of level $k \geq 1$ players to $\frac{\pi_\tau(k)}{\sum_{h\geq 1}\pi_\tau(h)}$. Thus, by assumption, there are no level 0 players in the actual population, although they may exist in the perceptions of other players. This approach has two free parameters τ and λ, which we estimate on the training data and test out-of-sample. Note that the "random play" that might be attributed to level 0 players in the baseline PCHM will here be attributed to the error caused by the logit response.[19]

Next, motivated by the preponderance of level 1 behavior, as observed in the previous section, we consider the following models which assume only level 1 behavior. As above, we

[19]Note also that if we modified the baseline PCHM (with exact best responses) in the same way, by supposing that there are no actual level 0 players, then the model would assign probability 1 to the level 1 action in any game in which the level 1 action is unique and is played in a Nash equilibrium. This is a stark prediction that we do not expect to predict well, so we did not try to fit that variant to the data.

assume that players use logit best replies.[20]

Level 1 with Logit Best Response. Let u_i be the expected row payoff to playing action a_i against a uniform distribution over column player actions. Predict action a_i with probability $e^{\lambda u_i} / \sum_{j=1}^{3} e^{\lambda u_j}$, where again the parameter λ controls the degeneracy of the best response. This approach has a single free parameter λ, which we estimate on the training data and test out-of-sample.

5.1.2 Heterogeneous τ

In addition to the variations on existing models we introduced above, we introduce a new way of using parametric models to make predictions.

Consider for example the PCHM, which has the single free parameter τ. In the standard application of this model, a single value of τ is learned for predicting play in all games. The value of τ that best fits the observed distribution of play, however, varies significantly across the games in our data sets (as they did also across the games studied in Camerer, Ho and Chong (2004)). Below we show the distribution of best-fit values of τ for each of the 86 lab games and 200 random games in our data set. The estimates of τ are reported separately for the PCHM (see (a) for lab data and (c) for MTurk data), and for the LPCHM (see (b) for lab data and (d) for MTurk data).[21] Notice that in both cases, the distribution of best-fit values of τ shifts right when logit best replies are assumed in place of the exact best replies of the standard PCHM. We believe this to be because lower values of τ are used to capture "random play" in the baseline PCHM, while this behavior is absorbed into the logit best replies in the LPCHM.

[20] The assumption of logit best responses produces a non-degenerate prediction over actions.

[21] The higher density of $\tau \approx 0$ estimates in the lab data may reflect the differences between the lab games and the random ones we used on MTurk. In particular, the games in the lab experiments tend to be more strategically complex, which might lead more participants to play in a way that does do not fit the PCHM. When forced to match this kind of game play with a value of τ, the best fit can be the uniform distribution (τ=0).

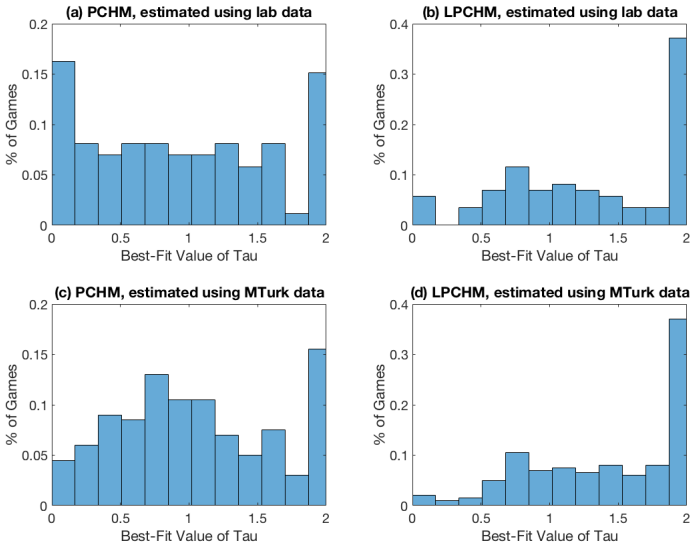


Figure 7: The best-fit value of τ varies substantially across games.

The variation seen in Figure 7 suggests that there are potential gains to prediction by allowing for game-specific values of τ. The best-fit values of τ found above, however, are estimated using the actual observed game play. When predicting play in a new game, we need to anticipate these values from properties of the game matrix alone.

Unfortunately, the reasons that games have play that is better fit by high or low values of τ are not obvious. In particular, we cannot simply interpret high values of τ as corresponding to greater levels of strategic sophistication, and low values of τ to random play. For example, consider the game below:

	a_1	a_2	a_3	Frequency
a_1	25,25	30,60	100,95	22.92%
a_2	60,30	31,31	51,30	41.67%
a_3	95,100	30,51	0,0	35.42%

where the frequencies of play of each action are included in the final column. Here, action a_2 is the modal action, but there there is no value of τ under which the LPCHM gives this action the highest weight. This is loosely because action a_2 is not level-k (in the Stahl and Wilson (1995) sense) for any choice of k. The best-fit value of τ turns out to be 0 in this game, not because play is particularly well approximated by a uniform distribution (which corresponds to $\tau = 0$), but because all other values of τ result in distributions that fit worse.

In contrast, consider the following game:

	a_1	a_2	a_3	Frequency
a_1	50,50	0,50	20,40	0
a_2	50,0	10,10	100,40	67.50%
a_3	40,20	40,100	40,40	32.50%

Actions a_2 and a_3 are level k for different values of k, and action a_1 (which is never played) is never level k. The realized distribution of play can be nearly perfectly approximated by a large choice of τ (roughly 13), which yields the prediction $(0, 0.674, 0.326)$.

Our approach for prediction of τ is to posit a set of game features that describe strategic properties of the game matrix, and then use decision tree algorithms to train predictive functions from the set of game features to values of τ. The features that we use are described in Appendix B.2, and primarily track different notions of "simplicity" of the payoff matrix. For example, we include as features the number of Nash equilibria in the game, the number of actions that are part of an action profile that maximizes the total sum of payoffs, and the number of actions that are level k for some value k. The more actions there are that fulfill each of these criteria, the less likely it is that there is an "obviously best" action. We include also several continuous measures for how much better the "best" action (according to different notions of "best") is than the next best. For example, we look at the difference between the expected payoff of the level 1 action against uniform play, and the expected payoff of the next best action.

We use the features described above to predict whether τ will be "low" or "high" in the following way:

Heterogeneous-PCHM. For each game in our training data, we first find the value of τ that best fits the observed distribution of play. We then learn a function that takes as input the features described above, and outputs a classification for the game as low or high τ. (Somewhat arbitrarily, we define low τ as $\tau < 1$ and high τ as $\tau \geq 1$. In the lab data, 54% of games have best-fit values of τ that are less than 1, and in the set of random games, 46% of games have best-fit values of τ that are less than 1. See Appendix B.2 for robustness checks to other split points.)

We pool all training games with a low value of τ, and learn the best single value τ_{low} for predicting play in these "low-τ" games. We similarly pool all training games with a high value of τ and estimate τ_{high} for predicting play in these high-τ games. We then train a decision tree to classify games into low- or high-τ based on their payoff matrices. When presented with a new game, our approach is to first predict whether τ is low or high (using

the decision tree), and then predict play using PCHM with the corresponding estimate of τ (τ_{low} or τ_{high}). This approach is illustrated in Figure 8.

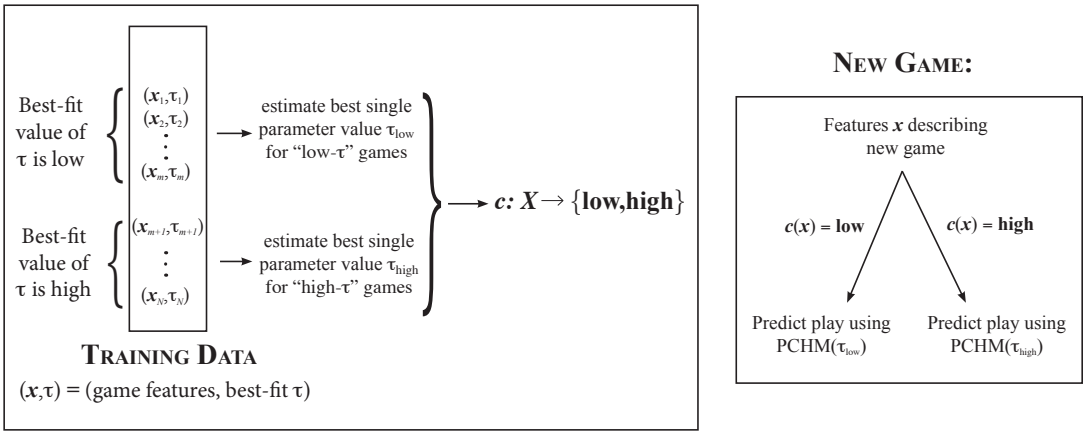


Figure 8: Illustration of our approach for prediction of parameter heterogeneity in PCHM

Note that this approach generates predictions for τ using only game features and not observations of play. Since all reported errors are cross-validated, the additional flexibility that comes from permitting heterogeneity does not guarantee an improvement in prediction.

Heterogeneous-LPCHM. As above, we use the training data to learn a function that predicts whether the best-fit value of τ is low ($\tau < 1$) or high ($\tau \geq 1$) based on the payoff matrix.[22] We then estimate a best-fit pair of parameters $(\tau_{low}, \lambda_{low})$ for the set of low τ games and $(\tau_{high}, \lambda_{high})$ for the set of high τ games. Out of sample, we first predict whether τ is low or high, and then predict using LPCHM with the corresponding (τ, λ) pair.

5.2 Results

Below we evaluate each of the proposed models of Section 5.1 on our two data sets. All prediction errors are tenfold cross-validated, and we additionally report a "completeness" measure as before.[23]

[22] The best-fit τ is less than 1 in 32% of the lab games and 25% of the randomly generated games.

[23] See Section B.1 in the Online Appendix for variations on the Nash prediction rule, and see Appendix C.2 for the estimated values of free parameters.

We begin by considering the data set of lab play. Table 5 reports the prediction errors of the approaches described in Section 5.1.1, ordered from least to most predictive.

Method	Prediction Error	Completeness
Naive benchmark	0.0687	0%
Uniform Nash	0.0828	<0%
PCHM	0.0333	51.53%
	(0.0042)	
Level 1 with Logit BR	0.0265	61.43%
	(0.0040)	
PCHM with Risk Aversion	0.0259	62.30%
	(0.0028)	
LPCHM	0.0175	74.53%
	(0.0014)	
LPCHM, No Level-0 Players	0.0161	76.56%
	(0.0034)	
Ideal prediction	0	100%

Table 5: Predicting the distribution of play in lab data

Our prediction rule based on predicting uniformly at random from Nash actions does not improve upon the naive baseline. In contrast, the PCHM achieves a significant improvement over this baseline, attaining a completeness measure of 50%. The proposed variations do better still, attaining 61-77% of the achievable improvement over guessing at random. The best performance is achieved by LPCHM under the assumption that there are no level 0 players in the population—this approach turns out to achieve more than a 20% improvement in our completeness measure over the classic PCHM.

We show next that we can continue to improve upon these performances by predicting heterogeneity in parameter values (see Table 6). We demonstrate these improvements specifically for three models: the PCHM, the LPCHM, and the adaptation of the LPCHM that drops level 0 players. For example, the Heterogeneous-PCHM improves the PCHM prediction error from 0.0333 to 0.0262 (resulting in an increase of completeness from 52% to 62%). This improvement is achieved by using the following class-specific estimates of τ: $\tau_{low} = 0.44$ and $\tau_{high} = 1.44$ (see Appendix C.2 for more on parameter estimates).

Method	Prediction Error	Completeness
PCHM	0.0333	51.53%
	(0.0042)	
Heterogeneous-PCHM	0.0262	61.86%
	(0.0019)	
LPCHM	0.0175	74.53%
	(0.0014)	
Heterogeneous-LPCHM	0.0165	75.98%
	(0.0030)	
LPCHM, No Level-0 Players	0.0161	76.56%
	(0.0034)	
Heterogeneous-LPCHM, No Level-0	0.0157	77.15%
	(0.0024)	

Table 6: Predicting heterogeneity in τ improves accuracy in predicting play in lab data

All of our results extend to prediction of play in the MTurk games. The performances of the models in 5.1.1 (without parameter heterogeneity) are below shown in Table 7, again ordered from least to most predictive. Again, the prediction rule based on predicting uniformly at random from Nash actions does worse than guessing at random, while the other models we consider attain significant improvements.

There are a few minor differences in Table 7 relative to the corresponding results for the lab data. First, all absolute prediction errors are lower when we predict the MTurk data. This suggests that the distributions of play in the random games are easier to predict, consistent with our observations in Section 2.2. Additionally, we see in Table 7 a marked improvement in the performance of the level 1 model with logit best replies.[24] We see two potential reasons for the performance of this model. First, the Mechanical Turk subjects may be less sophisticated than lab subjects and thus more likely to play the level 1 action. Second, as noted in Section 2.2, the randomly generated games are strategically simpler than the lab games.

In Appendix C.1, we seek to distinguish between these two explanations by supplementing our two main data sets of play with a third, in which 256 MTurk subjects were each asked to act as the row player in 15 games from our lab data set. Subjects were told that their choices would be matched with those of "other subjects," but we were not explicit

[24]This is consistent with our earlier observation that the modal action is more often level 1 in the random games data (88% vs. 72%).

Method	Prediction Error	Completeness
Naive	0.0838	0%
Uniform Nash	0.1283	<0%
PCHM	0.0186	77.80%
	(0.0038)	
PCHM with Risk Aversion	0.0173	79.36%
	(0.0014)	
LPCHM	0.0153	81.74%
	(0.0018)	
Level 1 with Logit BR	0.0134	84.01%
	(0.0008)	
LPCHM, No Level-0 Players	0.0133	84.13%
	(0.0009)	
Ideal prediction	0	100%

Table 7: Predicting the distribution of play in MTurk data

about who those subjects were. On top of a base payment, participants received a payoff bonus, depending on the actions they chose in the game.[25] We find that the level 1 models outperform PCHM variations in predicting this new data, as they did for the random games; this suggests the subject-based explanation as the primary reason for the differences.

Finally, Table 8 shows that introduction of heterogeneity in τ again yields improvements in prediction, although the sizes of these improvements are smaller than for the lab data. For example, Heterogeneous-PCHM improves the PCHM prediction error from 0.0186 to 0.0159 (resulting in an increase of completeness from 78% to 81%).

5.3 Understanding the Classification of τ

Our results above show that the predictions of the PCHM and related models can be improved by using training data to predict parameter heterogeneity. What do these atheoretical prediction rules look like, and is it possible to approximate their performance using more interpretable models? One way to understand these black box classification algorithms is to constrain the the decision tree model to use only two splits, as in Figure 5 of Section 4. . By examining the resulting classification rule, we can better understand the reasons for variation in best-fit value of τ. Below, we focus on application of Heterogeneous-PCHM and Heterogeneous-LPCHM to prediction of the lab data.

[25] We matched subjects' choices with the modal actions chosen by the lab subjects, and paid them according to the corresponding payoff profile.

Method	Prediction Error	Completeness
PCHM	0.0186	77.80%
	(0.0038)	
Heterogeneous-PCHM	0.0159	81.03%
	(0.0006)	
LPCHM	0.0153	81.74%
	(0.0018)	
Heterogeneous-LPCHM	0.0150	82.10%
	(0.0016)	
LPCHM, No Level-0 Players	0.0133	84.13%
	(0.0009)	
Heterogeneous-LPCHM, No Level-0 Players	0.0131	84.37%
	(0.0010)	

Table 8: Predicting heterogeneity in τ improves accuracy in predicting play in MTurk data

The best 2-split tree for classification of τ turns out to use two features, both of which track different measures of the "obviousness" of the best action. We describe these features now. First, for each action a_i, define v_i to be the row player's expected payoff when the column player randomizes uniformly over his actions. The difference between the expected payoff under the level 1 action and the expected payoff under the next best action is then:

$$max_{i\in\{1,2,3\}} v_i - max\left\{ v_j : j \neq \underset{i\in\{1,2,3\}}{\operatorname{argmax}}\, v_i \right\}$$

Below, we call this the *expected payoff gap*. Second, for each action $a_i \in A_1$, let $m_i = \max_{a_2 \in A_{\text{col}}} u_{\text{row}}(a_i, a_2)$ be the highest payoff that the row player can receive if he chooses action a_i. Define

$$\max_{i\in\{1,2,3\}} m_i - \max\left\{ m_j : j \neq \underset{i\in\{1,2,3\}}{\operatorname{argmax}}\, m_i \right\}$$

to be the difference between the highest payoff that the row player can receive when choosing a "max-max" action, versus the action that allows for the next highest possible payoff. Below, we call this feature the *salience gap*. These features are combined for classification of τ in the following way:

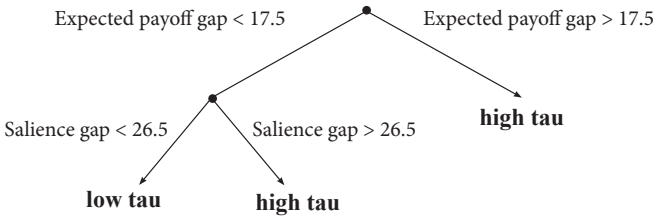


Figure 9: Best 2-split decision tree for classifying τ in PCHM

This decision tree says the following: if either the expected payoff gap is sufficiently large (at least 17.5), or if the expected payoff gap is low but the salience gap is large (at least 26.5), then predict that the game has a high value of τ. Otherwise—that is, if both the expected payoff gap and the row sum gap are low—predict that the game has a low value of τ. Intuitively, a large expected payoff gap and a large salience gap both make the "best" action more compelling. In contrast, if all actions have similar expected payoffs against uniform play, and all allow for similar "best possible" payoffs, then there may not be an obviously best action.

We can directly use this tree to classify games in the following way. For each game in our training data, split games into low- or high- τ categories based on the decision tree above, and learn best-fit values τ_{low} and τ_{high} for games in either class. Out of sample, we first predict whether τ is low or high based on the tree, and then predict the distribution of play using PCHM with the corresponding value of τ.

We can repeat a similar exercise for classification of τ in LPCHM. Figure 10 shows the best decision tree for classifying τ in this model, under the constraint that only two decision nodes can be used. In addition to the salience gap feature already described above, this tree uses a feature for the number of actions that are level k for some value of k.

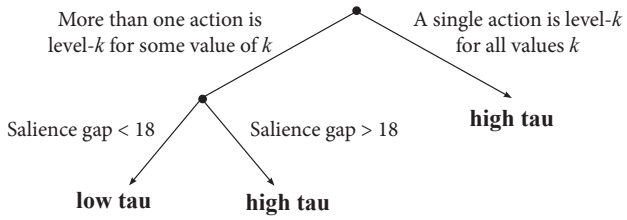


Figure 10: Best 2-split decision tree for classifying τ in LPCHM

The tree is different from the previous one because τ plays a different role in the PCHM and the LPCHM, and the games that have low and high values of τ are not the same across the two models. Nevertheless, the classification trees have some similarities. For example, notice that the presence of a single action that is level k for all orders k is again an indication of strategic simplicity. The classification tree above predicts that τ is high in games with a single action that is level k for all k, and additionally, in games with multiple level k actions and a large salience gap. In contrast, games with multiple level k actions and a small salience gap are predicted to have a low value of τ; as before, these are games in which there may not be an obviously best action.

We compare below the prediction errors using these simpler and more interpretable classification rules, against the baseline model, and against Heterogeneous-PCHM and Heterogeneous-LPCHM (where we do not impose artificial constraints on the complexity of the rule used to predict τ). We find that the 2-split models described above do improve over the baseline model, and achieve between 40-70% of the improvement of the best decision tree. We leave open the question of whether there are other simple models for classification of τ that can perform better yet.

Method	Prediction Error	Completeness
PCHM	0.0333	51.53%
	(0.0042)	
Heterogeneous-PCHM, 2-split	0.0303	55.90%
	(0.0031)	
Heterogeneous-PCHM, unconstrained	0.0262	61.86%
	(0.0019)	
LPCHM	0.0175	74.53%
	(0.0014)	
Heterogeneous-LPCHM, 2-split	0.0165	75.98%
	(0.0025)	
Heterogeneous-LPCHM, unconstrained	0.0161	76.56%
	(0.0030)	

Table 9: Simple classification rules for τ improve beyond the baseline model, and achieve a substantial fraction of the improvement of the best classification rules.

Note that there are two reasons why these improved methods would not predict perfectly even if the test data were generated by an underlying deterministic process. One source of

error is imperfect classification of τ, and the other is limitations of the model class—, even with knowledge of the category of τ, the proposed approaches will not predict perfectly. Appendix B.3 decomposes the prediction errors shown above, providing an estimate of how much of the error is due to either cause. We find that our out-of-sample predictions of τ achieve 40-80% of the achievable improvement with perfect knowledge of the class of τ. These results suggest that significant improvements on prediction beyond those shown above will require use of more flexible model classes (and correspondingly, more data to estimate the free parameters).

6 Identifying New Feature Sets: Crowd Predictions

Our results so far have illustrated the potential for feature-based prediction rules to improve prediction of play beyond existing approaches. Ultimately, these feature-based approaches are only as powerful as the features that we use. An interesting question for subsequent work is then what additional features might be predictive.

We conclude with an extended discussion of one very different approach for feature construction that does not explicitly use the payoff matrix. These features are instead based on human inputs—specifically, predictions of play by untrained human subjects.

6.1 Crowd Prediction Data

We asked human subjects on Mechanical Turk to predict play in 15 games from the 286 games described in Section 2. To the best of our knowledge, these subjects are untrained: the initial part of our experiment consisted of an introduction to matrix games, and we allowed subjects to proceed to the main experiment only after correctly answering a set of comprehension questions.[26]

In the main part of our experiment, each subject was shown a random subset of either fifteen of the lab games (Section 2.1) or fifteen of the random games (Section 2.2). We informed subjects that these games had been played by real people, and asked them to predict the action that was *most likely to be chosen* by the row player. To incentivize effort, we told subjects that on top of their base payment of \$0.25, they would receive an additional \$0.10 for every question they answered correctly. Figure 11 shows a typical question prompt presented to subjects, and the complete set of instructions can be found in Appendix D.

A total of 250 subjects participated in the lab game prediction experiment, and 540 subjects participated in the random game prediction experiment. On average, approximately

[26]The comprehension questions consisted of reporting the payoff to either player for a fixed action pair in two example games (see Appendix D). All subjects eventually answered both comprehension questions correctly.

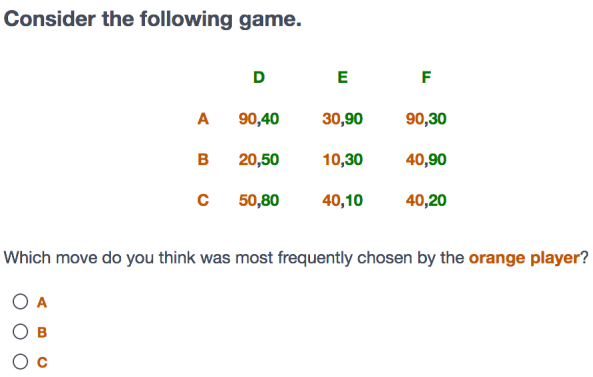


Figure 11: A typical question prompt presented to Mechanical Turk subjects in the single action treatment. The "orange player" is the row player.

40 crowd predictions were observed for each game.

6.2 Predictions and Results

We consider a straightforward use of these crowd predictions for prediction of play. For every game g_i, let x_k^i be the fraction of subjects who predicted action a_k in game g_i, so that the feature vector identified with game g_i is

$$(x_1^i, x_2^i, x_3^i) \in [0,1]^3. \tag{5}$$

In the problem of predicting the realized action, we predict play of action

$$\underset{k\in\{1,2,3\}}{\operatorname{argmax}}\, x_k^i$$

in game g_i, and in the problem of predicting the distribution, we predict the distribution

$$(x_1^i, x_2^i, x_3^i)$$

in game g_i. These naive crowd rules use only the perception of payoffs by (untrained) participants. Nevertheless, we find that they perform extremely well in both prediction problems and for both data sets; see Tables 10 and 11 below.[27] Specifically, the naive crowd

[27] Below, standard errors for the crowd prediction rule are bootstrapped standard errors with 100 resamples.

rule improves upon the PCHM in three of the four prediction problems,[28] and outperforms our most predictive model (LPCHM without level 0 players) in the MTurk data. The naive crowd rule does not, however, improve upon our best model-based approaches for predicting the lab data.

	Lab Data		MTurk Data	
	Error	Completeness	Error	Completeness
Naive benchmark	0.6667	0%	0.6667	10%
Uniform Nash	0.5507	33.66%	0.4722	51.21%
PCHM	0.3838	82.02%	0.3159	92.36%
	(0.0197)		(0.0217)	
Crowd	0.3965	78.34%	0.3091	94.15%
	(0.0056)		(0.0067)	
Ideal prediction	0.3218	100%	0.2869	100%

Table 10: Prediction of the action played

	Lab Data		MTurk Data	
	Error	Completeness	Error	Completeness
Naive benchmark	0.0687	0%	0.0838	0%
Uniform Nash	0.0828	<0%	0.1283	<0%
PCHM	0.0333	51.53%	0.0186	77.80%
	(0.0042)		(0.0038)	
LPCHM, No Level-0 Players	0.0161	76.56%	0.0133	84.13%
	(0.0034)		(0.0009)	
Crowd	0.0285	58.52%	0.0091	89.14%
	(0.0033)		(0.0008)	
Ideal prediction	0	100%	0	100%

Table 11: Prediction of the distribution of play

6.3 Do Subjects Predict Their Own Play?

A potential explanation for the performance of the naive crowd rule predictions is that subjects simply predict the actions that they themselves would choose. This hypothesis would imply that each prediction is equivalent to an observation of play, so that with

[28] The naive crowd rule performs slightly worse than the PCHM in the problem of predicting the realized action in lab data, but its completeness measure is comparable.

sufficiently many predictions, the distribution of crowd predictions would approximate the distribution of play arbitrarily well.

To evaluate this hypothesis, we compare the distributions of play with the distributions of crowd predictions. Specifically, we test the null hypothesis that our samples of game play and samples of crowd predictions are drawn from the same distribution. We use a Kolmogorov-Smirnov test to determine a p-value for each game. Under the hypothesis that crowd predictions and game play are indeed drawn from the same distribution, we would expect these p-values to follow a uniform distribution. We find instead that for both the lab games and the random games, the observed distribution of p-values is statistically different from the uniform distribution (see Figure 12 below).[29]

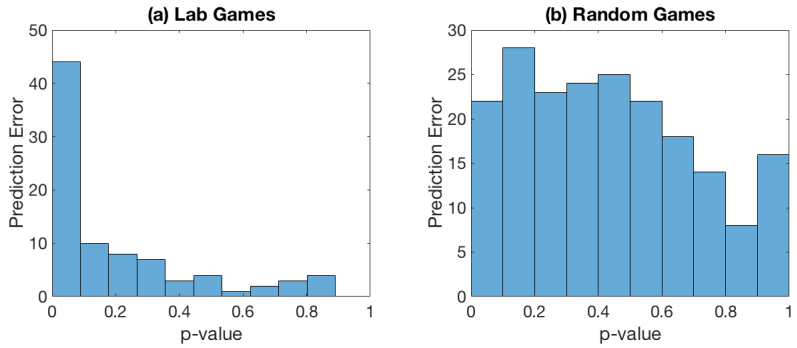


Figure 12: We reject the hypothesis that crowd predictions and game play come from the same distributions for all games.

The difference is especially large for the lab data. One reason for this may be that the populations generating the predictions and game play are different—predictions are made by MTurk subjects, while game play is chosen by lab subjects. We therefore return to a supplementary data set mentioned previously in Section 5.2 (and described in more detail in Appendix C.1), in which 40 Mechanical Turk subjects were asked to play each of the lab games. We repeat the analysis above, this time comparing the observed distribution of play by *MTurk subjects* in the lab games with the crowd predictions. The resulting distribution of p-values (shown below) is less skewed than in the left panel of Figure 12. Nevertheless, this distribution is again statistically different from the uniform distribution, and its departure from the uniform distribution is larger than the one found in the right panel of Figure 12. This provides further evidence that crowd predictions are not identical to game play, and in particular that the crowd systematically predicts some games better

[29] We reject that the distribution of p-values for the lab games is uniform with $p \approx 10^{-17}$ under a Kolmogorov-Smirnov test, and reject that the distribution of p-values for the random games is uniform with $p = 0.0027$.

than others.[30]

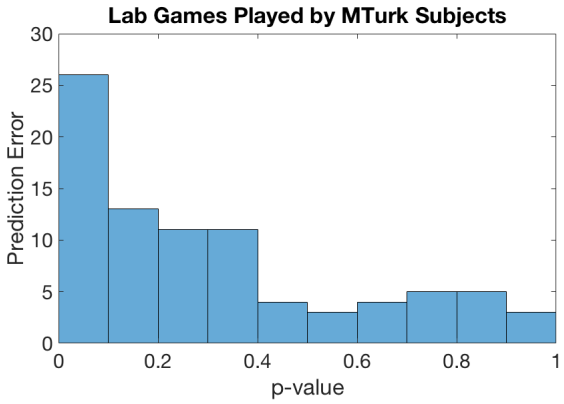


Figure 13: The distribution of p-values remains statistically different from uniform when we compare crowd predictions with MTurk play of lab games.

These results suggest that there is potential to use of human inputs for improved prediction; we leave open the questions of what kind of games are most amenable to crowd prediction, and how human inputs might be more usefully leveraged than the naive aggregation rule considered here.

7 Conclusion

Although significant advances have been made in the prediction of initial play, we still do not have a complete model for initial play in the simple setting of 3×3 one-shot matrix games. We show in this paper that atheoretical prediction rules built on a large set of game-based features can improve predictive accuracy beyond existing models, and we may expect that more complicated rules built on richer feature sets, and trained on more data, should predict better still.[31]

However, there is a tradeoff here, as in other settings, between the predictiveness and the interpretability of the model. Our focus is not on predictive accuracy alone, and as we have shown, machine learning can be used not only to improve predictions of play, but also to improve our understanding of it, and to develop simple and portable improvements on existing models.

[30] We conjecture that the reason for this is because the random games are more likely to have "obviously best" actions which coordinate both play and prediction.

[31] See for example the deep learning techniques used in Hartford, Wright and Leyton-Brown (2016).

One way we do this is to study trained models under strict parsimony constraints. For example, when a sufficiently high parsimony constraint is imposed in the problem of predicting the chosen action, we find that the best decision tree returns the well-known behavioral model of level 1 thinking. As we relax this constraint, the trained model builds on the level 1 framework, appending additional features for prediction. Examination of games in which the machine learning algorithms predicted play better than the existing theories helped us to realize that adding a risk aversion parameter to the level 1 model generates better out-of-sample predictions.

As a second approach, we use machine learning methods to predict heterogeneity in the parameters of existing models. Specifically, we use game features to predict the best value of τ in (variations of) the Poisson Cognitive Hierarchy model. This approach emphasizes that the best model of play may differ systematically depending on the strategic structure of the game, and shows that machine learning methods may help us to identify the forces behind this heterogeneity. Along with papers such as Leyton-Brown and Wright (2014), these results suggest potential for interplay between machine learning methods and theory models to improve prediction and understanding of play in games. Beyond our present setting of predicting play, we expect also that the approach of predicting parameters from underlying features can be used to improve the out-of-sample performance of other economic models.

Appendix

A Supplementary Material to Section 2.2

We compare below the following summary statistics for the two sets of games:

- whether there exists an action profile a that is *best for both players*: a maximizes payoffs for both players:

$$a \in \underset{a' \in A}{\operatorname{argmax}}\, u_{\text{row}}(a') \quad \text{and} \quad a \in \underset{a' \in A}{\operatorname{argmax}}\, u_{\text{col}}(a')$$

- the *number of games with an action that is (pure-strategy) strictly dominated*
- the *number of games that are (pure strategy) dominance-solvable*
- the *variance* of the payoffs $\frac{1}{18}\sum_{i=1}^{18}(g_i - \bar{g})^2$
- the *maximum individual payoff* $\max_i g_i$
- the *minimum individual payoff* $\min_i g_i$
- the *maximum total payoff* $\max_{a \in A} u_{\text{row}}(a) + u_{\text{col}}(a)$
- the *minimum total payoff* $\min_{a \in A} u_{\text{row}}(a) + u_{\text{col}}(a)$
- the *correlation between player payoffs*

	Lab Games	Random Games
Dominance-solvable	0.1395	0.22
≥ 1 strictly dominated action	0.31	0.48
"Best for both" profile	0.1279	0.275
Variance of payoffs	901.7684	652.67
Max payoff	95.4070	88.3000
Min payoff	3.1977	11.8000
Max sum of payoffs	151.1512	153.1000
Min sum of payoffs	22.7209	46
Correlation between players' payoffs	0.074	0.029
Observations	**86**	**200**

Table 12: Comparison of summary statistics for the lab games and random games.

B Complete List of Features

B.1 Features Describing Specific Actions

For each action a_i, we include an indicator variable for:

- whether that action is part of a *pure-strategy Nash equilibrium*
- whether that action is part of an action profile that *maximizes the sum of player payoffs*; that is, whether there exists an action $a_2 \in A_{\text{col}}$ such that

$$u_{\text{row}}(a_1, a_2) + u_{\text{col}}(a_1, a_2) = \max_{a \in A}(u_{\text{row}}(a) + u_{\text{col}}(a)).$$

- whether that action is part of a *Pareto dominant Nash equilibrium*
- whether that action is *"max-max"*; that is, whether there exists some action $a_2 \in A_{\text{col}}$ such that

$$(a_i, a_2) \in \operatorname*{argmax}_{a \in A} u_{\text{row}}(a).$$

- whether that action is *level k* for each $k = 1, 2, \ldots, 7$

Additionally, we include a *score* feature for each action, which is the number of above properties that it satisfies.

B.2 Features Describing Properties of the Game

Recall that $g \in \mathbb{R}^{18}$ describes the payoff matrix. We include the following properties of the payoff matrix:

- The *number of pure strategy Nash equilibria*
- The *number of actions that are strictly dominated by a pure strategy*
- The *number of level 1 actions*; that is, the size of the set $\operatorname{argmax}_{a_1 \in A_{\text{row}}} u_{\text{row}}(a_1, \alpha)$ where α is the uniform distribution.
- The *number of actions that are "max-max."*
- The *number of actions that maximize total payoffs.*.
- The *number of actions that are both level 1 and "max-max."*
- The *number of actions that are both level 1 and maximize total payoffs.*

- The *number of actions that are both "max-max" and maximize total payoffs.*
- The *number of actions that are level 1, "max-max," and maximize total payoffs.*
- For each action a_i, let o_i be the number of properties it satisfies from the following list: level 1, best-for-both, and max-max. We include as a feature

$$max_k o_k - max\left\{o_l : l \neq argmax_k o_k\right\} \tag{6}$$

  The larger this gap, the more salient the "best" action relative to the next best action.
- We include also the feature $max_{k\in\{1,2,3\}} o_k$, where o_k is as defined above.
- The *expected payoff gap*, as defined in Section 5.3.
- The *salience gap*, as defined in Section 5.3.
- For each action $a_i \in A_{\text{row}}$, let $y_i = max_{a_2 \in A_{\text{col}}} u_{\text{row}}(a_i, a_2) + u_{\text{col}}(a_i, a_2)$ be the largest possible total payoff when the row player chooses a_i. Define the feature

$$\max_{i\in\{1,2,3\}} y_i - \max\left\{y_j : j \neq argmax_{i\in\{1,2,3\}} y_i\right\}$$

  to be the difference between the highest achievable total payoff when the row player chooses a best-for-both action, and when the row player chooses the action that allows for the next highest achievable total payoff.

C Supplementary Material to Section 5

C.1 Prediction of Mechanical Turk Play on Lab Games

We supplement our two main data sets with a third, in which 256 subjects were each asked to act as the row player in 15 games from our lab data set. There are in total 45 observations of play for each game. This new data set allows us to better understand the differences in play across our two main data set, and in particular, which differences are due to *different subject pools* and which differences are due to the games having *different strategic structures.* We consider the methods for prediction described in Section 5.1, and again report tenfold cross-validated prediction errors.

Method	Prediction Error	Completeness
Naive	0.0382	1
Uniform Nash	0.0642	<0
PCHM	0.0242	36.65%
	(0.0023)	
PCHM with Risk Aversion	0.0202	47.12%
	(0.0017)	
PCHM with Logit BR	0.0153	59.95%
	(0.0048)	
PCHM with Logit BR, No Level 0 Players	0.0149	60.99%
	(0.0015)	
Level 1 with Logit BR	0.0147	61.52%
	(0.0015)	
Level 1 with Logit BR and Risk Aversion	0.0125	67.28%
	(0.0013)	
Ideal prediction	0	0

Table 13: Prediction of play of lab games by Mechanical Turk subjects

Results are qualitatively similar to Table 5, with the exception that the Level 1 models do much better.

C.2 Parameter Estimates (Pooling All Games)

First, we show below the prediction error of PCHM as a function of τ, where we separately predict the full set of lab play and the full set of MTurk play. This allows us to see how sharply prediction error varies with τ.

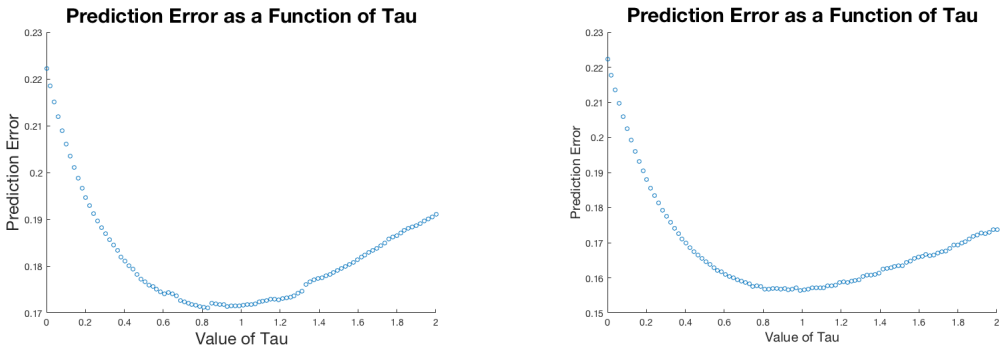


Figure 14: *Left*: prediction of play of lab games; *Right*: prediction of play of random games.

We find that (in-sample) prediction error is minimized at $\tau \approx 0.81$ for prediction of play of lab games, and $\tau \approx 0.94$ for prediction of play in random games.

Next, we report estimated parameters for each of the approaches considered in the main text. We report separately the parameter estimates for each of our data sets of play: lab games played by lab subjects (as introduced in Section 2.1), lab games played by Mechanical Turk subjects (as introduced above in Section C.1), and random games played by Mechanical Turk subjects (as introduced in Section 2.2). Parameter estimates are averaged across the multiple iterations of training.

	Lab Games		Random Games
	Lab Subjects	MTurk Subjects	
PCHM	$\tau = 0.81$	$\tau = 0.33$	$\tau = 0.94$
LPCHM	$\tau = 1.54$	$\tau = 1$	$\tau = 1.25$
	$\lambda = 0.17$	$\lambda = 0.11$	$\lambda = 0.17$
Risk-PCHM	$\tau = 0.75$	$\tau = 0.33$	$\tau = 0.90$
	$\alpha = 0.67$	$\alpha = 0.22$	$\alpha = 0.67$
LPCHM, No Level 0 Players	$\tau = 1.46$	$\tau = 0.35$	$\tau = 0.44$
	$\lambda = 0.14$	$\lambda = 0.05$	$\lambda = 0.09$
Logit Level 1	$\lambda = 0.02$	$\lambda = 0.02$	$\lambda = 0.03$
Logit-Risk Level 1	$\lambda = 0.08$	$\lambda = 0.09$	$\lambda = 0.08$
	$\alpha = 0.71$	$\alpha = 0.62$	$\alpha = 0.81$

Table 14: Parameter Estimates

When estimating Heterogeneous-PCHM on the lab data set, we find that $\tau_{low} = 0.44$ and $\tau_{high} = 1.44$. The same model, estimated using the MTurk dataset, yields $\tau_{low} = 0.67$ and $\tau_{high} = 1.55$. When estimating Heterogeneous-LPCHM on the lab data set, we find $(\tau_{low}, \lambda_{low}) = (0.83, 0.16)$ and $(\tau_{high}, \lambda_{high}) = (1.67, 0.21)$. The same model, estimated using the Mechanical Turk dataset, yields $(\tau_{low}, \lambda_{low}) = (0.67, 0.37)$ and $(\tau_{high}, \lambda_{high}) = (1.82, 0.11)$

C.3 Best-Fit Values of Parameters for Individual Games

A key step in the analysis of Section 5 is to learn best-fit values of τ for each game in the training data. This allows us to train predictive models to classify τ based on the payoff matrix. We report below the distribution of best-fit values of τ in each of our two data sets.

We first note that our parameter estimates for the games in Stahl and Wilson (1995) are similar to the estimates reported in Camerer, Ho and Chong (2004) (Table III):[32]

	Our estimate of τ	*Estimate of τ reported in Camerer, Ho and Chong (2004)*
Game 1	*3.23*	*2.93*
Game 2	*0*	*0*
Game 3	*1.21*	*1.40*
Game 4	*2.82*	*2.34*
Game 5	*2.02*	*2.01*
Game 6	*0*	*0*
Game 7	*8.08*	*5.37*
Game 8	*0*	*0*
Game 9	*1.21*	*1.35*
Game 10	*6.46*	*11.33*
Game 11	*8.48*	*6.48*
Game 12	*1.61*	*1.71*

Table 15: Comparison of our parameter estimates for lab games from Stahl and Wilson (1995) with the estimates reported in Camerer, Ho and Chong (2004).

In both data sets, there is substantial heterogeneity in the best-fit value of τ across different games. Panel (a) in Figure 15 below shows a histogram of best-fit values of τ across the games in our lab data set. In our main analysis in Section 5, we remove the right tail of estimates (as these increase the variance in the output of the prediction algorithm) by imposing a constraint that $\tau \leq 2$. Under this constraint, the median value of τ is 0.8889 and the variance is 0.4433 (For the unconstrained values, the median value of τ is 1.2121, and the variance is 6.4569.) The distribution of best-fit values of τ in this range is shown below in panel (b) of Figure 15. The median value of τ in lab games is 1, and the variance is 0.2102 .

[32]These parameter estimates should be interpreted with caution, since prediction error as a function of τ is a poorly behaved function with large discontinuities. Moreover, for some games, rather different values of τ turn out to yield prediction errors that closely approximate the global minimum. These curiosities explain some of the small differences in the parameter estimates below.

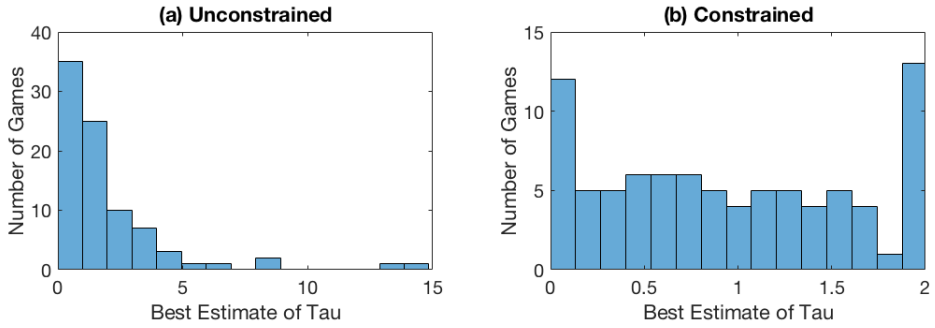
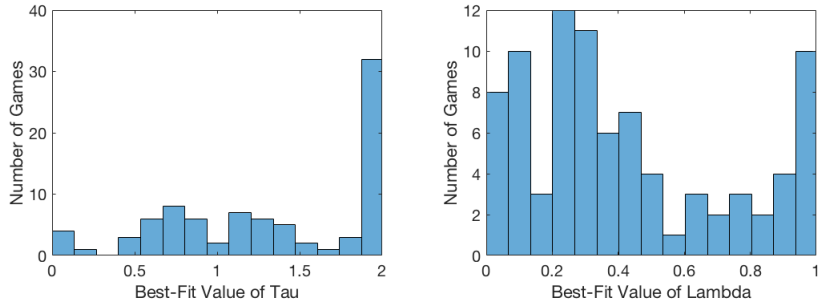
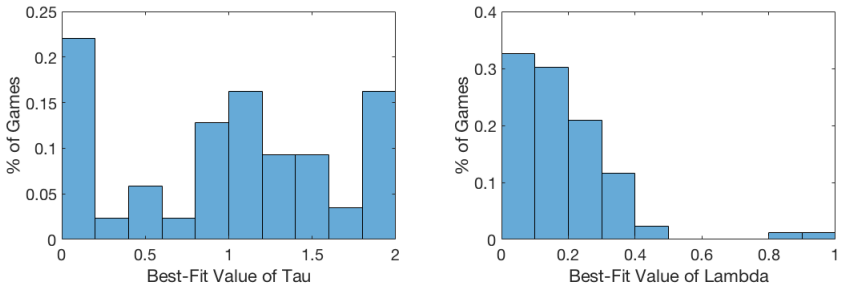


Figure 15: The best-fit τ varies substantially across lab games: in (a) we report the best-fit values of τ, in (b) we report the best-fit values of τ under the constraint that $\tau \leq 2$.

We additionally show the distributions of best-fit values of τ and λ across lab games for the LPCHM. The distribution of best-fit values of τ shifts right relative to Figure 15; this likely reflects that many of the games in which the best-fit value of τ was low had "noisier" distributions of play, which can alternatively be captured with for lower values of λ.



Finally, we show the distributions of best-fit values of τ and λ for the LPCHM with level 0 players removed. Note that unlike in the PCHM, here the parameter value $\tau = 0$ does not correspond to probability 1 of level 0 play.



D Experimental Instructions

The instructions provided to Mechanical Turk subjects in the experiments described in Sections 2.1 and 6.1 can be found below. With a few exceptions, instructions that were repeated across these experiments are only presented once.

D.1 Playing Random Games (Section 2.1)

D.1.1 Initial Instructions

We are researchers interested in how people play a simple kind of game.

Rules of the game

There are two players. Each player is assigned to one of two roles: **orange** and **green**. Both players move only once, and they move at the same time. The orange player's move is to choose one of

A **B** **C**

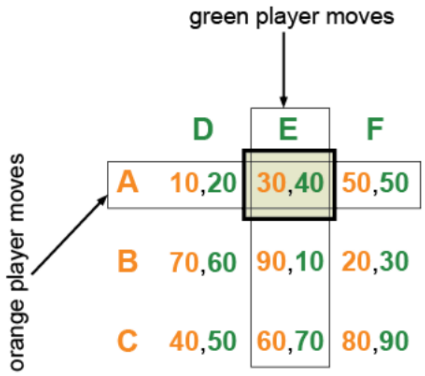
and the green player's move is to choose one of

D **E** **F**

Depending on which moves are chosen, each player wins a certain number of points. These points are displayed in a table like this one:

	D	E	F
A	10,20	30,40	50,50
B	70,60	90,10	20,30
C	40,50	60,70	80,90

To read this table, look at the row marked with the orange player's move, and the column marked with the green player's move. This determines a pair of numbers. For example, if the orange player moves **A** and the green player moves **E**, then you should look at **30**,**40**.



The **first number** is the number of points that the orange player wins, and the **second number** is the number of points that the green player wins.

Easy? Let us ask you a few questions to make sure you got it.

D.1.2 Comprehension Questions

Comprehension Question 1/2

	D	E	F
A	50,40	90,30	20,70
B	30,10	40,90	20,60
C	60,10	50,80	80,40

You are the **orange player**. If you choose **A** and your partner chooses **F**, how many points will you win?

Comprehension Question 2/2

	D	E	F
A	90,90	40,30	70,30
B	70,60	30,30	40,70
C	50,40	80,10	90,30

You are the **green player**. If you choose **D** and your partner chooses **B**, how many points will you win in this game?

D.1.3 Explanation of Payment

Great! You answered both questions correctly. Now let's move on to your main task.

Your task

We will show you fifteen games like the one described above. You will be asked to play the **orange player** in each of these games.

How you are paid

You will be paid a **base rate of $0.35** for completing the HIT. In addition, one of the fifteen games you play will be chosen at random. We will match you with another subject who has been asked to play as the orange player, and we will use your joint moves to determine the number of points you win. You will then receive **a bonus** of:

$0.01 x the number of points you won in that game

This bonus will range from $0.10-$0.90. Please allow up to a week to receive this.

We are almost ready to begin the exercise.

Please read through the following information and indicate your consent before continuing.

D.1.4 Typical Question

Consider the following game.

	D	E	F
A	50,80	10,20	50,50
B	50,50	20,30	90,20
C	40,20	50,70	10,20

You are the **orange player**. What move do you choose?

- ○ A
- ○ B
- ○ C

D.2 Predicting the Most Likely Action (Section 6.1)

D.2.1 Initial Instructions

How well can you guess how people will play in games?

We are researchers interested in whether you can predict how people play in a simple kind of game. Real people were matched with a partner and asked to play the following two-player game:

Rules of the game

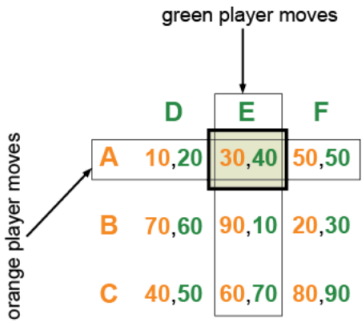
There are two players. Each player is assigned to one of two roles: **orange** and **green**. Both players move only once, and they move at the same time. The yellow player's move is to choose one of

A B C

and the green player's move is to choose one of

D E F

Depending on which moves are chosen, each player wins a certain number of points. These points are displayed in a table like this one:



The number of points the orange player wins is the **first number**, and the number of points the green player wins is the **second number**.

Easy? Let us ask you a few questions to make sure you got it.

D.2.2 Payment Explanation

Great! You answered both questions correctly. Now let's move on to your main task.

The challenge
Real people were asked to play games like the ones you just looked at. In each round of this HIT, we will show you the points table for one of these games, and ask you to guess which move was most frequently chosen by the **orange player**. There are fifteen total games.

How you are paid

You will receive **$0.25** no matter what for completing this HIT. But you will receive **$0.05** more for every round in which you correctly guess the move most frequently chosen. This means that you will win **a bonus of up to 0.75**. Please allow up to a week for the bonus to arrive.

You may only complete this HIT once. If you complete this HIT multiple times, you will be rejected.

We are almost ready to begin the exercise. Please read through the following information and indicate your consent before continuing.

D.2.3 Typical Question

Consider the following game.

	D	E	F
A	45,45	50,41	21,40
B	41,50	0,0	40,100
C	40,21	100,40	0,0

Which move do you think was most frequently chosen by the **orange player**?

- ○ A
- ○ B
- ○ C

References

Camerer, Colin, and Teck-Hua Ho. 1999. "Experienced-Weighted Attraction Learning in Normal Form Games." *Econometrica*. 1.1

Camerer, Colin, Gideon Nave, and Alec Smith. 2017. "Dynamic unstructured bargaining with private information: theory, experiment, and outcome prediction via machine learning." Working Paper. 3

Camerer, Colin, Teck-Hua Ho, and Juin-Kuan Chong. 2004. "A Cognitive Hierarchy Model of Games." *The Quarterly Journal of Economics*. 1, 4.1, 4.1, 5.1.2, C.3, 15

Cheung, Yin-Wong, and Daniel Friedman. 1997. "Individual Learning in Normal Form Games: Some Laboratory Results." *Games and Economic Behavior*. 1.1

Chong, Juin-Kuan, Teck-Hua Ho, and Colin Camerer. 2016. "A Generalized Cognitive Hierarchy Model of Games." *Games and Economic Behavior*. 1.1

Costa-Gomes, Miguel, and Georg Weizsacker. 2007. "Stated Beliefs and Play in Normal-Form Games." *Review of Economic Studies*. 1.1

Costa-Gomes, M., V. Crawford, and B. Broseta. 2001. "Cognition and behavior in normal-form games: an experimental study." *Econometrica*. 4.1

Crawford, Vincent. 1995. "Adaptive Dynamics in Coordination Games." *Econometrica*. 1.1

Crawford, Vincent, Miguel Costa-Gomes, and Nagore Iriberri. 2013. "Structural Models of Nonequilibrium Strategic Thinking: Theory, Evidence, and Applications." *Journal of Economic Literature*. 1, 1.1, 5.1.1

DellaVigna, Stefano, and Devin Pope. 2017. "Predicting Experimental Results: Who Knows What?" *Journal of Political Economy*. 1.1

Erev, Ido, and Alvin Roth. 1999. "Predicting How People Play Games: Reinforcement Learning in Experimental Games with Unique, Mixed Strategy Equilibria." *American Economic Review*. 1.1

Fragiadakis, Daniel E., Daniel T. Knoepfle, and Muriel Niederle. 2016. "Who is Strategic?" Working Paper. 1.1

Hartford, Jason, James Wright, and Kevin Leyton-Brown. 2016. "Deep Learning for Predicting Human Strategic Behavior." 1.1, 31

Haruvy, E., and D. Stahl. 2007. "Equilibrium selection and bounded rationality in symmetric normal-form games." *Journal of Economic Behavior and Organization.* 2.1

Haruvy, E., D. Stahl, and P. Wilson. 2001. "Modeling and testing for heterogeneity in observed strategic behavior." *Review of Economic and Statistics.* 2.1

Hastie, Trevor, Robert Tibshirani, and Jerome Friedman. 2009. *The Elements of Statistical Learning.* Springer. 8

Kleinberg, Jon, Annie Liang, and Sendhil Mullainathan. 2017. "The Theory is Predictive, but is it Complete? An Application to Human Perception of Randomness." Working Paper. 1

Leyton-Brown, Kevin, and James Wright. 2014. "Level-0 Meta-Models for Predicting Human Behavior in Games." *ACM Conference on Economics and Computation (ACM-EC).* 1, 1.1, 2.1, 3, 10, 4.1, 5.1.1, 7, A.2

Morris, Stephen, Rafael Rob, and Hyun Song Shin. 1995. "p-Dominance and Belief Potential." *Econometrica.* 34

Peysakhovich, Alex, and Jeff Naecker. 2017. "Using Methods from Machine Learning to Evaluate Models of Human Choice Under Uncertainty." Forthcoming. 1

Rogers, B.W., T.R. Palfrey, and C.F. Camerer. 2009. "Heterogeneous quantal response equilibrium and cognitive hierarchies." *Journal of Economic Theory.* 2.1

Sgroi, Daniel, and Daniel John Zizzo. 2009. "Learning to play 3x3 games: Neural networks as bounded-rational players." *Journal of Economic Behavior and Organization.* 1.1

Stahl, Dale O., and Paul W. Wilson. 1995. "On players' models of other players: Theory and experimental evidence." *Games and Economic Behavior.* 1, 2.1, 4.1, 4.1, 5.1.1, 5.1.2, C.3, 15

Stahl, D., and E. Haruvy. 2008. "Level-n bounded rationality and dominated strategies in normal-form games." *Journal of Economic Behavior and Organization.* 2.1

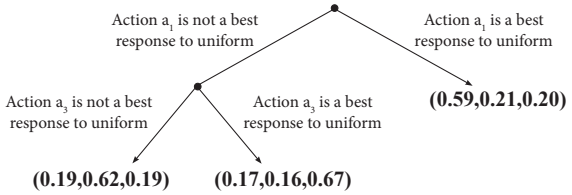
Stahl, D., and P. Wilson. 1994. "Experimental evidence on players' models of other players." *Journal of Economic Behavior and Organization.* 1, 2.1, 2.1, 4.1, 5.1.1

Online Appendix

A Supplementary Material to Section 4

A.1 The Best 2-Split Decision Tree for Predicting Distribution of Play Also Uses Level 1 Features

Below, we show the best 2-split decision tree for the task of predicting the distribution of play. This tree closely resembles the one shown in Figure 5. In particular, the most predictive two features are again the level 1 features.



When action a_1 is level 1, the tree predicts a distribution that places majority weight on a_1, and similar for actions a_2 and a_3.

A.2 Cross-Validation at the Observation Level

In the main text, we reported tenfold cross-validated errors, where the method of cross-validation was to divide the games into ten folds, use all observations of play associated with games in nine of the folds for training, and use the observations of play associated with games in the remaining fold for testing. We consider below an alternative approach to cross-validation, where we pool all of the observations of play and randomly split this pooled data into folds. This is the approach used in Leyton-Brown and Wright (2014).

Table 16 presents misclassification rates for both the lab data and the MTurk data.

	Lab Data		MTurk Data	
	Error	Completeness	Error	Completeness
Naive benchmark	0.6667	0	0.6667	1
Uniform Nash	0.5507	33.66%	0.4722	51.21%
	(0.0055)		(0.0075)	
Level 1/PCHM	0.3838	82.02%	0.3159	92.36%
	(0.0197)		(0.0217)	
Prediction rule based on game features	0.3360	95.88%	0.2984	96.97%
	(0.0056)		(0.0095)	
Ideal prediction	0.3218	1	0.2869	0

Table 16: Predicting the realized action using "observation-level" cross-validation

B Supplementary Material to Section 5

Below we consider predictive models based on the set of game-based features described in Appendix B, and built using Lasso regression.[33] We report the error of the better-performing algorithm below, which does not improve on the PCHM:

	Lab Games		Random Games	
	Error	Completeness	Error	Completeness
Naive benchmark	0.2222	0	0.2222	1
Uniform Nash	0.2529	<0	0.2121	12%
	(0.0020)		(0.0016)	
Feature-Based Prediction Rules	0.1825	51%	0.1630	71%
	(0.0252)		(0.0158)	
PCHM	0.1721	64%	0.1611	74%
	(0.0087)		(0.0205)	
Ideal prediction	0.1443	0	0.1391	1

Table 17: Predicting the distribution of play

B.1 Alternative Nash Predictions

In the main text, we consider uniform prediction over all actions that are part of a Nash equilibrium. This is not the only possible prediction model based on Nash equilibrium. For

[33] Here the outcome to be predicted is a frequency vector, and fit is evaluated using the mean-squared distance between the predicted frequency vector and the actual frequency vector.

example, we can use stricter standards, such as predicting only actions that are part of a Pareto-dominant or a risk-dominant Nash equilibrium.[34]

We consider below the following approaches:

1. **Predict actions that are part of a Pareto-dominant and risk-dominant Nash equilibrium:** here we predict an action with probability 1 if it is part of a Pareto-dominant and risk-dominant Nash equilibrium, and otherwise predict the uniform distribution.

2. **Predict actions that are part of a Pareto-dominant Nash equilibrium:** here we predict an action with probability 1 if it is part of a Pareto-dominant Nash equilibrium, and otherwise predict the uniform distribution.

3. **Predict actions that are part of a risk-dominant Nash equilibrium:** here we predict an action with probability 1 if it is part of a risk-dominant Nash equilibrium, and otherwise predict the uniform distribution.

Method	Lab Games	Random Games
(a)	0.0906	0.0813
(b)	0.1641	0.1238
(c)	0.0906	0.0845

Table 18: Alternative prediction models based on Nash equilibrium

These errors are comparable to those of the uniform Nash prediction rule, and in particular are substantially worse than the PCHM.

B.2 Other Split Points

We consider different ways of classifying games: specifically, defining low-τ and high-τ with alternative split points, and allowing for three categories (low-, medium-, and high-τ). The two alternative split points we consider are the median value of the best-fit τ (in the full lab data set), and the mean value of the best-fit τ (again in the full lab data set). For determining three classes, we use the 33rd and 66th percentile as split points. The results shown below are similar to the main text.

[34]For our setting of 3x3 games, we consider specifically (2/3)-dominance (Morris, Rob and Shin, 1995): (a_1, a_2) is a (2/3)-dominant Nash equilibrium if a_1 is a best response when the column player chooses a_2 with probability at least 2/3, and vice versa.

Method	Prediction Error	Parameter Estimates
Heterogeneous-PCHM split at $\tau = 0.8889$ (median)	0.0230 (0.0024)	$\tau_{low} = 0.3333$ $\tau_{high} = 1.3333$
Heterogeneous-PCHM split at $\tau = 0.9557$ (mean)	0.0248 (0.0032)	$\tau_{low} = 0.4444$ $\tau_{high} = 1.3333$
Heterogeneous-PCHM 3 categories (split at 33 and 66 percentile)	0.0258 (0.0021)	$\tau_{low} = 0.2222$ $\tau_{med} = 0.8888$ $\tau_{high} = 1.7778$

Table 19: Classify τ based on other split points

B.3 Decomposing the Prediction Error

We seek below to separate two causes for prediction error using the Heterogeneous-PCHM and the Heterogeneous-LPCHM. First, our prediction of τ is imperfect, so there may be residual variation in τ even after conditioning on the features we have written down.[35] A second and distinct source of error is model misspecification: for example, even if we could perfectly classify τ, application of the heterogeneous-PCHM will not lead to perfect prediction of play. Error of the first kind can be reduced by improving prediction of τ, whereas error of the second kind can only be reduced by increasing the flexibility of the model class.

To understand how much of the prediction error is due to either source, we remove the second source of error by reporting in-sample prediction errors. Specifically, when given a new game, we use the *actual* best-fit value of τ to classify the game. This approach provides a lower bound for what we could hope to achieve when the class of τ is predicted as in the main text.

For prediction of play in the lab games, we find that the in-sample error of the PCHM with two classes of τ is 0.0213, achieving a completeness measure of 69%. The gap between the out-of-sample error of 62% and the in-sample error of 69% represents room for better prediction of τ, while the much larger gap between 69% and perfect prediction represents room for improved prediction by extending the PCHM. This suggests that our method for predicting τ allows for approximately half of what we would achieve with perfect classification of τ. For the set of random games, we find that the PCHM with out-of-sample prediction of τ improves completeness from 77% to 81%, relative to the best-possible performance 82%. Thus, the feature set that we use for prediction of τ achieves most of the

[35] Additionally, we are naturally constrained by the number of games in the training data.

possible improvement (73%) that is attainable by the proposed method.

	Lab Data		MTurk Data	
	Error	Completeness	Error	Completeness
PCHM	0.0333	51.53%	0.0186	77.80%
	(0.0042)		(0.0038)	
Heterogeneous-PCHM	0.0262	61.86%	0.0159	81.03%
	(0.0019)		(0.0006)	
In Sample	0.0213	69%	0.0149	82.22%

Table 20: Decomposing the prediction error of Heterogeneous-PCHM

The table below repeats this exercise for the LPCHM with qualitatively similar results.

	Lab Data		MTurk Data	
	Error	Completeness	Error	Completeness
LPCHM	0.0175	74.53%	0.0153	81.74%
	(0.0014)		(0.0018)	
Heterogeneous-LPCHM	0.0165	75.98%	0.0150	82.10%
	(0.0030)		(0.0016)	
In Sample	0.0153	77.73%	0.0147	82.46%

Table 21: Decomposing the prediction error of Heterogeneous-LPCHM

C Explanation of Choices in Experiments

Subjects were asked to explain how they made their choices in a (free-form) text box. We show below selected answers from our experiments in which players were asked to choose an action:

- "I chose based on mutually beneficial numbers, followed by singular benificial [sic] numbers, and finished with whatever was left over."
- "Except the first question. I added the orange in each row(A,B,C) Then put it in order from highest to the least. I'm hoping I did this right :o)"
- "i count each value quickly. It is easy for me. Good game"
- "I assumed Green was aquisitive [sic] and non-sharing"

- "Without knowing what sort of patterns the partner displayed it's mostly guesswork. I assumed orange would avoid choosing rows where zero payoff was possible, and that green would similarly prefer not to bet on columns with a zero payoff. I assumed both would think the same way and be trying to achieve a good payoff, not just selecting the row or column with the highest possible payoff. Wheels within wheels."
- "i tried to figure out if there is obvious worst of all, then eliminate it"
- "I looked at what Green would probably pick and then based on that decided what Orange would pick when thinking about what the Green letter would likely be."

We show below selected answers from our experiments in which players were asked to predict the play of others:

- "I picked the lines that had the biggest looking numbers. People like big numbers."
- "I chose mostly the midrange digits for most and varied the low and high for mid and least."
- "I looked at the highest numbers and whether there were any zeroes in the line, because I figured that would be a huge deterrent for someone."
- "I chose the route of either placing the orange player in a strict profit maximizing role without taking into account the decisions of the green player, or I chose the best scenario for both the orange and green player."
- "I just picked what felt right at the particular game"
- "i was aware that the best way to choose orange move was relative to the best move for green but i don't think people that took this study was smart enough for considering that and they would choose first the move that had the biggest number."
- "i just tried to be logical"