# Modeling Human Behavior in Strategic Settings

by

James Robert Wright

B.Sc., Simon Fraser University, 2000

M.Sc., The University of British Columbia, 2010

A DISSERTATION
SUBMITTED IN PARTIAL FULFILLMENT OF THE
REQUIREMENTS FOR THE DEGREE OF

**Doctor of Philosophy**

in

THE FACULTY OF GRADUATE AND POSTDOCTORAL STUDIES

(Computer Science)

The University of British Columbia
(Vancouver)

August 2016

© James Robert Wright, 2016

# Abstract

Increasingly, electronic interactions between individuals are mediated by specialized algorithms. One might hope to optimize the relevant algorithms for various objectives. An aspect of online platforms that complicates such optimization is that the interactions are often *strategic*: many agents are involved, all with their own distinct goals and priorities, and the outcomes for each agent depend both on their own actions, and upon the actions of the other agents.

**My thesis is that human behavior can be predicted effectively in a wide range of strategic settings by a single model that synthesizes known deviations from economic rationality.** In particular, I claim that such a model can predict human behavior better than the standard economic models. Economic mechanisms are currently designed under behavioral assumptions (i.e., full rationality) that are known to be unrealistic. A mechanism designed based on a more accurate model of behavior will be more able to achieve its goal.

In the first part of the dissertation, we develop increasingly sophisticated data-driven models to predict human behavior in strategic settings. We begin by applying machine learning techniques to compare many existing models from behavioral game theory on a large body of experimental data. We then construct a new family of models called quantal cognitive hierarchy (QCH), which have even better predictive performance than the best of the existing models. We extend this model with a richer notion of *nonstrategic* behavior that takes into account features such as fairness, optimism, and pessimism, yielding further performance improvements. Finally, we perform some initial explorations into applying tech-

niques from deep learning in order to automatically learn features of strategic settings that influence human behavior.

A major motivation for modeling human strategic behavior is to improve the design of practical mechanisms for real-life settings. In the second part of the dissertation, we study an applied strategic setting (peer grading), beginning with an analysis of the question of how to optimally apply teaching assistant resources to incentivize students to grade each others' work accurately. We then report empirical results from using a variant of this system in a real-life undergraduate class.

# Preface

The research described in this dissertation was performed in collaboration with other researchers.

Chapter 2 was co-authored with Kevin Leyton-Brown. An early version was published at the Conference of the Association for the Advancement of Artificial Intelligence [Wright and Leyton-Brown, 2010]. I did the data collection, software implementation, literature review, and wrote the majority of the paper. Kevin provided guidance throughout, and contributed a significant amount of writing and editing. The methodology, much of the analysis, and all of the figures in this chapter have been updated since publication.

Chapters 3 and 4 were co-authored with Kevin Leyton-Brown. Early versions of portions of the two chapters were published together in the International Conference on Autonomous Agents and Multiagent Systems [Wright and Leyton-Brown, 2012]. An updated version was submitted to a journal, and is currently under revision. I did the software implementation and experimental evaluation, and wrote the majority of the paper. Kevin provided guidance and helped write the text.

Chapter 5 was co-authored with Kevin Leyton-Brown. An early version was published in the ACM Conference on Economics and Computation [Wright and Leyton-Brown, 2014]. Kevin and I worked together to design the various alternative model specifications. I did the software implementation and experimental evaluation, and wrote the majority of the paper. Kevin provided guidance and helped write the text. This chapter has been heavily updated since publication.

Chapter 6 was co-authored with Kevin Leyton-Brown and Jason Hartford; it is as-yet unpublished. Jason, Kevin, and I worked together to devise the architecture described in the chapter. I did some early software implementation, but Jason did the bulk of the implementation. Jason wrote most of the text; I provided guidance and helped with the text throughout. Kevin also provided guidance and helped write the text.

Chapter 7 was co-authored with Xi Alice Gao and Kevin Leyton-Brown; it is as-yet unpublished. Alice, Kevin, and I worked together to devise the model that we evaluate in the chapter. Alice and I worked together on all the proofs. The text in Section 7.3 is primarily due to me; Alice wrote most of the remaining text, with help from me. Kevin also provided guidance and helped write the text.

Chapter 8 was co-authored with Kevin Leyton-Brown and Chris Thornton. Jessica Dawson also provided very valuable help with the literature review. An early version was published at the ACM Technical Symposium on Computer Science Education [Wright et al., 2015]. Kevin, Chris, and I worked together to devise the mechanism. Chris did the early implementation of the software described, and I added the later enhancements described in the chapter. I wrote the majority of the paper. Kevin provided guidance and helped write the paper.

# Table of Contents

**Abstract** . . . . . **ii**

**Preface** . . . . . **iv**

**Table of Contents** . . . . . **vi**

**List of Tables** . . . . . **xi**

**List of Figures** . . . . . **xii**

**List of Abbreviations** . . . . . **xiv**

**Acknowledgments** . . . . . **xvi**

**Dedication** . . . . . **xviii**

**1 Introduction** . . . . . **1**

1.1 Behavioral Game Theory as a Machine Learning Problem . . . . . 4

1.2 Application Domain: Peer Grading . . . . . 6

1.3 The Way Forward . . . . . 7

**I Behavioral Game Theory as a Machine Learning Problem . . . . . 8**

**2 Prediction Performance of Behavioral Game Theoretic Models . . . 9**
2.1 Introduction . . . 9
2.2 Models for Predicting Human Play of Simultaneous-Move Games 12
2.2.1 Quantal Response Equilibrium . . . 13
2.2.2 Level-k . . . 14
2.2.3 Cognitive Hierarchy . . . 15
2.2.4 Quantal Level-k . . . 16
2.2.5 Noisy Introspection . . . 18
2.3 Comparing Models . . . 19
2.3.1 Prediction Framework . . . 19
2.3.2 Assessing Generalization Performance . . . 21
2.4 Experimental Setup . . . 22
2.4.1 Data . . . 22
2.4.2 Comparing to Nash Equilibrium . . . 25
2.4.3 Computational Environment . . . 26
2.5 Model Comparisons . . . 27
2.5.1 Comparing Behavioral Models . . . 27
2.5.2 Comparing to Nash Equilibrium . . . 30
2.5.3 Dataset Composition . . . 32
2.6 Related Work . . . 35
2.7 Conclusions . . . 37

**3 Parameter Analysis of Behavioral Game Theoretic Models . . . . . 39**
3.1 Introduction . . . 39
3.2 Methods . . . 41
3.2.1 Posterior Distribution Derivation . . . 41
3.2.2 Posterior Distribution Estimation . . . 41
3.2.3 Visualizing Multi-Dimensional Distributions . . . 43

3.3 Analysis . . . 43
3.3.1 Poisson-CH . . . 44
3.3.2 Nash Equilibrium . . . 46
3.3.3 QLk . . . 46
3.4 Conclusions . . . 48

**4 Model Variations . . . 50**
4.1 Construction . . . 50
4.2 Simplicity Versus Predictive Performance . . . 52
4.3 Parameter Analysis . . . 54
4.4 Spike-Poisson . . . 55
4.5 Conclusions . . . 56

**5 Models of Level-0 Behavior . . . 58**
5.1 Introduction . . . 58
5.2 Level-0 Model . . . 59
5.2.1 Level-0 Features . . . 59
5.2.2 Combining Feature Values . . . 64
5.2.3 Feature Transformations . . . 65
5.3 Model Selection . . . 66
5.3.1 Forward Selection . . . 66
5.3.2 Bayesian Optimization . . . 67
5.3.3 Extended Model Performance . . . 69
5.3.4 Parameter Analysis . . . 71
5.4 Related Work . . . 75
5.5 Conclusions . . . 76

**6 Deep Learning for Human Strategic Modeling . . . 78**
6.1 Introduction . . . 78
6.2 Related Work . . . 79
6.3 Modeling Human Strategic Behavior with Deep Networks . . . 80

6.3.1 Feature Layers . . . . . 82
6.4 Experimental Setup . . . . . 88
6.5 Experimental Results . . . . . 89
6.6 Regular Neural Network Performance . . . . . 91
6.7 Conclusions . . . . . 92

**II Application Domain: Peer Grading . . . . . 94**

**7 Incentivizing Evaluation via Limited Access to Ground Truth . . . . 95**
7.1 Introduction . . . . . 96
7.2 Peer-Prediction Mechanisms . . . . . 99
7.3 Impossibility of Pareto-Dominant, Truthful Elicitation . . . . . 104
7.4 Combining Elicitation with Limited Access to Ground Truth . . . 106
7.5 When Does Peer-Prediction Help? . . . . . 108
7.6 Conclusions . . . . . 112
7.7 Proofs . . . . . 114
7.7.1 Proof of Lemma 1 . . . . . 114
7.7.2 Proof of Lemma 2 . . . . . 115
7.7.3 Proof of Lemma 3 . . . . . 117
7.7.4 Proof of Theorem 3 . . . . . 118
7.7.5 Proof of Lemma 4 . . . . . 119
7.7.6 Proof of Corollary 1 . . . . . 121
7.7.7 Proof of Corollary 2 . . . . . 125

**8 Application: Mechanical TA . . . . . 129**
8.1 Introduction . . . . . 129
8.2 Peer Evaluation Model . . . . . 132
8.2.1 Supervised and Independent Reviewers . . . . . 133
8.2.2 Calibration . . . . . 133
8.3 Evolution of our Design . . . . . 134
8.3.1 Calibration Setup . . . . . 135

8.3.2 Independent Reviewers . . . . . 135
8.3.3 Exam Performance . . . . . 137
8.4 Analysis of our Current Design . . . . . 138
8.4.1 Review Quality . . . . . 138
8.4.2 Calibration Performance . . . . . 140
8.5 Related Work . . . . . 142
8.6 Conclusions . . . . . 144

**Bibliography . . . . . 146**

**A CPSC 430 2014 grading rubric . . . . . 160**

# List of Tables

Table 2.1 Names and contents of each dataset. . . . . 25
Table 2.2 Previous estimates of level-0 proportions . . . . . 29
Table 2.3 Datasets conditioned on various game features . . . . . 33
Table 2.4 Existing work in model comparison . . . . . 38

Table 4.1 Model variations with prediction performance . . . . . 51

# List of Figures


Figure 2.1 Likelihoods of model predictions . . . . . 28
Figure 2.2 Likelihoods on "treasure" and "contradiction" treatments . . . 31
Figure 2.3 Likelihoods on feature-based datasets . . . . . 35

Figure 3.1 Cumulative posterior distributions for Poisson-CH . . . . . 44
Figure 3.2 Distributions for Poisson-CH and QRE by dominance-solvability 45
Figure 3.3 Posterior distributions for NEE . . . . . 47
Figure 3.4 Posterior distributions for QLk on ALL10 . . . . . 48

Figure 4.1 Model simplicity vs. prediction performance . . . . . 52
Figure 4.2 Posterior precision distributions for `ah-QCH3` and `ah-QCH4` . . 53
Figure 4.3 Posterior distributions for `ah-QCH3` and `ah-QCH4` . . . . . 54
Figure 4.4 Model simplicity vs. prediction performance for efficient models, QLk, and Spike-Poisson QCH . . . . . 57

Figure 5.1 Prediction performance with binary features . . . . . 67
Figure 5.2 Training and test performance with binary features . . . . . 68
Figure 5.3 Prediction performance for Bayesian optimization incumbents 69
Figure 5.4 Prediction performance for Poisson-CH, Lk, and Spike-Poisson QCH with `linear8`, `linear4`, and uniform level-0 specifications. . . . . 70
Figure 5.5 Posterior distributions of levels for `linear4`, `linear8`, and uniform . . . . . 71

Figure 5.6 Posterior precision distributions for `linear8`, `linear4`, and uniform . . . . . 72
Figure 5.7 Posterior distribution of weights for `linear4` . . . . . 73
Figure 5.8 Posterior distribution of weights for `linear8` . . . . . 74

Figure 6.1 A schematic representation of our architecture . . . . . 82
Figure 6.2 Graphical explanation of pooling units . . . . . 85
Figure 6.3 Prediction performance for GameNet vs. QCH, varying number of hidden units and layers. . . . . 89
Figure 6.4 Prediction performance for GameNet vs. QCH, no pooling units. 90
Figure 6.5 Prediction performance for GameNet vs. QCH, varying number action response layers. . . . . 91
Figure 6.6 Prediction performance of a feed forward neural network on fixed-size games with and without data augmentation . . . . . 92

Figure 8.1 Proportion of independent reviewers . . . . . 136
Figure 8.2 Distributions of final and midterm exam marks . . . . . 137
Figure 8.3 Mean review quality distributions . . . . . 141
Figure 8.4 Deviations from gold standard reviews on calibrations vs. number of calibrations after promotion . . . . . 142

# List of Abbreviations

**BGT** Behavioral Game Theory

**BTS** Bayesian Truth Serum

**CDF** cumulative density function

**CH** cognitive hierarchy

**CMA-ES** Covariance Matrix Adaptation Evolution Strategy

**CPR** Calibrated Peer Review

**Lk** level-$k$

**MAP** Maximum a Posteriori

**MCMC** Markov Chain Monte Carlo

**MLE** maximum likelihood estimation

**MLP** multi-layer perceptron

**NEE** Nash Equilibrium with Error

**NI** noisy introspection

**QCH** quantal cognitive hierarchy

**QLk** quantal level-$k$

**QRE** quantal response equilibrium

**TA** teaching assistant

# Acknowledgments

Many people deserve recognition for their contributions to this work.

First and foremost, I would like to thank my advisor Kevin Leyton-Brown. It is impossible to overstate the importance of his guidance. He has a rare talent for getting right to the crux of the matter, and has saved me from so many blind alleys that I cannot count them. He has taught me more about the presentation of ideas and the workings of the research world than anyone I know. Kevin has been incredibly generous. He has gone to considerable effort to secure amazing opportunities for me, as well as the financial support to make them feasible and the guidance to ensure that I made the most of them. Grad school was a wonderful experience for me, and having Kevin as an advisor was a major reason why.

Over the years, the Game Theory and Decision Theory reading group was always one of my favorite parts of the week. I could always count on insightful discussions, whether research-related or not. I am indebted to my collaborators and co-authors, especially Alice Gao and Jason Hartford. I have learned a great deal by working with them.

I am grateful to my supervisory committee members, Yoram Halevy and Holger Hoos, for their guidance. Before he was on my committee, Yoram's class on behavioral economics early in my degree was of critical value. I am also grateful to many others in the research community for their help and influence. I would particularly like to mention Colin Camerer, Kate Larson, and David Parkes.

Finally, none of this would have been possible without the patience and support of my family, especially Sarah. She has been unfailingly supportive in more

ways than I can name.

This work was funded in part by a Canada Graduate Scholarship from the Natural Sciences and Engineering Research Council of Canada, a Four Year Fellowship from the University of British Columbia, and a Collaborative Research and Development Grant funded by the Natural Sciences and Engineering Research Council of Canada and Google Inc. This work was completed in part while I was visiting the Simons Institute for the Theory of Computing.

For Sarah, Miles, and Edith

# Chapter 1

# Introduction

Increasingly, electronic interactions between individuals are mediated by specialized algorithms. For example, someone seeking short term accommodation might previously have searched free-form text ads on craigslist, whereas now a site such as Airbnb provides specific procedures for finding, booking, and paying for accommodations. Other platforms facilitate interactions that were not previously practical online, such as Uber and Lyft's matching of drivers to passengers, or the use of peer grading in large online courses.

All of these interactions generate data. One might hope to use this data to optimize the relevant algorithms in terms of various objectives. Indeed, many platforms use *A/B testing* for just such a purpose, using a modified algorithm for a random sample of their users and the original algorithm for the rest, and comparing the outcomes to determine which performs better. However, A/B testing has drawbacks. Only a low-dimensional space of variations can realistically be explored, and along the way many users will potentially be exposed to designs that are worse than the original.

One aspect of online platforms that makes interaction optimization especially difficult is that the interactions are often *strategic*. A strategic interaction has two main hallmarks. First, many agents are involved, all with their own distinct goals and priorities; in the online platform case the agents correspond to the users and

the platform designer. Second, the outcomes for each agent depend both on their own actions, and upon the actions of the other agents. For example, the outcome in an auction depends not only on how the winning bidder bids, but also on how the losing bidders bid, and on the auction's rules. Hence in order to act effectively in a strategic setting, an agent must reason not only about his own actions, but also about the beliefs and actions of the other agents, who are in turn reasoning about all the other agents' beliefs and action as well. In strategic settings, it can turn out that straightforward A/B testing does not accurately evaluate *either* of the options under test [Chawla et al., 2014].

An alternative approach is to learn a model of how people will react to an algorithm. One can then optimize among different designs by evaluating their counterfactual performance based on the model. This allows the modeler to explore a larger design space at a lower cost. However, such models are rare. This dissertation describes a research program that aims to construct a general model that can be applied to many strategic settings.

**My thesis is that human behavior can be predicted effectively in a wide range of strategic settings by a single model that synthesizes known deviations from economic rationality.** In particular, I claim that such a model can predict human behavior better than the standard economic models. Economic mechanisms are currently designed under behavioral assumptions (i.e., full rationality) that are known to be unrealistic. A mechanism designed based on a more accurate model of behavior will be more able to achieve its goal, whether that goal is social welfare, revenue, or any other aim. This approach of synthesizing many behavioral anomalies into a single model contrasts with a common approach in economics, where a model is constructed that explains or "rationalizes" a single anomaly [e.g., Gilboa and Schmeidler, 1989, for ambiguity aversion] or a small number of anomalies [e.g., Tversky and Kahneman, 1992, for loss aversion and distorted probability judgments], without necessarily evaluating the predictive strength of the model.

In the rest of the dissertation, we develop data-driven models to predict human

*strategic behavior*; that is, behavior in settings where each participant's rewards depend partially on the actions of other participants. This has important differences from most other machine learning problems. Firstly, each participant's behavior is strongly influenced by that of the others, and by their own forecasts of others' behavior. Second, participants' strategic behavior can be strongly influenced by counterfactuals; i.e., what would have happened had they, or other participants, behaved differently. Furthermore, different algorithm designs often involve differences in the set of choices that are available to the participants. Hence, a model used for answering counterfactual questions about alternate designs must be able to predict in a setting that has a completely different dimensionality than the setting that produced its training data. Thus, prediction in these settings cannot be framed as a classification problem in which one of a fixed set of labels is predicted. Similarly, the need to generalize between datasets with different dimensions of both inputs and outputs means that deep learning techniques cannot be straightforwardly applied in these settings.

Game theory is the standard mathematical framework for understanding strategic interactions. Game theoretic models assume that the participants in an interaction are idealized, perfectly rational agents. This is clearly an unrealistic assumption for individuals, and indeed, we know from both experimental and observational data that standard game theoretic models describe human behavior very poorly [Goeree and Holt, 2001; Rabin, 2000]. The interdisciplinary field of *behavioral game theory* (BGT) investigates deviations from the standard models of game theory, and proposes new models of human behavior by taking account of human cognitive biases and limitations [Camerer, 2003]. These models can be understood as parametric functions that can be applied to inputs of arbitrary dimension.

The dissertation is divided into two parts: behavioral game theory as a machine learning problem, and the peer grading application domain.

## 1.1 Behavioral Game Theory as a Machine Learning Problem

My long-term research agenda is to build a general theory for optimally designing algorithms that mediate interactions between humans, rather than between idealized agents. The BGT literature has proposed a great many models to describe human strategic behavior. However, synthesizing them into a usable predictive model for algorithm design requires computational tools that are not typically available to behavioral game theorists. For example, a natural question for a machine learning specialist would be, which of the many BGT models has the best out-of-sample prediction performance? The standard BGT technique for comparing models is to perform statistical tests comparing a general model's in-sample performance to a specialized version; this makes it impossible to compare non-nested models (i.e., models where neither is a strict generalization of the other) and is prone to preferring models that overfit to the training data.

In Chapter 2, we compare the prediction performance of six well known BGT models according to the standards of the machine learning community. Using a large body of experimental data collected from multiple sources in the literature, we performed a cross-validated comparison of the out-of-sample prediction performance of the models. This comparison involved computing maximum likelihood estimates of extremely nonlinear models, which required considerable computational expertise and resources, including over one year of CPU time on a high-performance computing cluster. Remarkably, we found that one particular model performed better than all of the others in most individual datasets, as well as in the combined dataset. This model incorporates two crucial elements: first, agents do not choose optimally according to their beliefs. Rather, they *quantally respond*, with every action being chosen with positive probability, but more optimal actions being chosen proportionally more often. Second, it is an *iterative* model: each agent is assumed to have a level representing the number of steps of strategic reasoning that it is capable of. Level-$0$ agents do not reason about the other agents; level-$1$ agents believe that all of their opponents are level-$0$ and

respond accordingly; and so forth.

One advantage of *structural models*—models whose parameters have a causal interpretation—is that the parameter values, if known, can help researchers understand reasons and mechanisms behind observed behavior. Thus, good parameter estimates are valuable both for optimizing a model's prediction performance, and also for providing scientific insight in their own right. Maximum likelihood estimation chooses parameters in a sensible way for comparing model performance, but it is less valuable for providing insight. In Chapter 3, we introduce a Bayesian framework for estimating a behavioral model's full posterior parameter distribution. We used this framework to analyze several models from Chapter 2, including the best-performing one. This produced insights that we build upon in Chapter 4 to construct a family of models that is both more parsimonious and more robust, and also predicts the data better, while requiring fewer parameters to be learned.

In any iterative model, agents reason about the behavior of their opponents starting from a specification of nonstrategic (level-0) behavior. Despite the pivotal role that it plays in determining the actions of higher-level agents, modeling level-0 behavior has received virtually no attention in the literature; in practice, almost all existing work specifies this behavior as a uniform distribution over actions. In most games it is not plausible that even nonstrategic agents would choose an action uniformly at random, nor that other agents would expect them to do so. In Chapter 5, we construct a richer model of level-0 behavior that can be plugged into any iterative model, in which level-0 agents choose actions that are in some way *salient* (e.g., by having the maximal best case, or by forming part of the fairest outcome). This level-0 model dramatically improved the performance of several iterative models.

The properties of actions that people might find salient—and which might thus be favored by nonstrategic agents—have thus far been discovered primarily by asking "How might I reason about playing this specific game?" Rather than relying solely on introspection and domain knowledge, we might hope to derive such properties directly from data. Deep learning [e.g., Bengio, 2009] has shown

success in a wide range of domains for automatically discovering features. In Chapter 6, we take the first steps toward adapting deep learning techniques to the strategic prediction domain with the goal of discovering new representations of salient game characteristics.

The invention of the pooling and convolution operators was a major advance in the application of deep neural networks to vision tasks [LeCun et al., 1998]. These operators exploit invariances and local structure in the domain to allow for vastly more efficient training of deep networks. Strategic games have a very different structure. A game is essentially the same game if the action labels are permuted, which means that local structure is less exploitable. Additionally, different games, even in the same domain, can have very different dimensionality depending on how many actions are available to each participant and how many participants there are. Intuitively speaking, the approach that we take in Chapter 6 aims to devise the equivalent of pooling and convolution operators for strategic modeling. This direction has proven extremely promising; the model that we present already achieves significant improvements in prediction performance over any of the structural models that we study, albeit at the expense of completely sacrificing interpretability. In future work, we hope to combine the interpretability of a structural model with the superior prediction performance of a deep model.

## 1.2 Application Domain: Peer Grading

The first part of the dissertation aims to construct a model that can be applied to any one-shot strategic setting involving human participants. In the second part of the dissertation, we shift from this general approach to focus in on a specific setting of this kind: peer grading, in which an instructor wishes to incentivize students to honestly and diligently evaluate each others' work.

We begin by performing a theoretical analysis of a more general problem that includes peer grading as a special case: eliciting truthful evaluations of arbitrary objects. There is a large literature on the problem of *peer prediction*, in which agents are incentivized to truthfully report their observations of a ground truth to

which the mechanism designer has no access whatsoever. The essential contribution of Chapter 7 is to analyze a more realistic setting, in which the mechanism designer has access to this ground truth, but only at a cost, which the designer wishes to minimize or avoid entirely. The peer prediction literature relies heavily on a particular assumption—that agents can coordinate *only* through the information that the mechanism wishes to elicit. We first show that when this assumption is relaxed, agents are almost surely better off not reporting their observations truthfully. This demonstrates that the assumption is not merely innocuous or technical. Second, we show that in the presence of costly access to ground truth, a simple dominant-strategy mechanism can elicit truthful reports better (in the sense of requiring less ground truth) than any of the peer prediction mechanisms of which we are aware.

Finally, in Chapter 8, we describe the outcomes of using a variant of the dominant-strategy mechanism from Chapter 7 in a real undergraduate class. This peer grading platform eventually formed the cornerstone of the class. The platform is now freely available for download, and has since been used at multiple other universities.

## 1.3 The Way Forward

As almost every aspect of our lives is increasingly mediated by algorithms, improving these algorithms has become increasingly valuable. But such improvement requires accurate models of how people will respond to the algorithms, and to each others' responses. With the growing availability of data, there is now an unprecedented opportunity to understand and predict human strategic behavior using a principled machine learning approach. My hope is that this dissertation will prove to be an important step along the way to achieving this goal.

# Part I

# Behavioral Game Theory as a Machine Learning Problem

# Chapter 2

# Prediction Performance of Behavioral Game Theoretic Models

## 2.1 Introduction

In strategic settings, it is common to assume that agents will adopt Nash equilibrium strategies, behaving so that each optimally responds to the others. This solution concept has many appealing properties; e.g., under any other strategy profile, one or more agents will regret their strategy choices. However, experimental evidence shows that Nash equilibrium often fails to describe human strategic behavior [see, e.g., Goeree and Holt, 2001]—even among professional game theorists [Becker et al., 2005].

The relatively new field of *behavioral game theory* extends game-theoretic models to account for human behavior by accounting for human cognitive biases and limitations [Camerer, 2003]. Experimental evidence is the foundation of behavioral game theory, and researchers have developed many models of how humans behave in strategic situations based on such data. This multitude of models presents a practical problem, however: which model should we use to predict human behavior?

Existing work in behavioral game theory does not directly answer this ques-

tion, for two reasons. First, it has tended to focus on explaining (fitting) in-sample behavior rather than predicting out-of-sample behavior. This means that models are vulnerable to *overfitting* the data: the most flexible model can be chosen instead of the most accurate one. Second, behavioral game theory has tended not to compare multiple behavioral models, instead either exploring elaborations of a single model or comparing only to one other model (typically Nash equilibrium). In this chapter we perform rigorous—albeit computationally intensive—comparisons of many different models and model variations on a wide range of experimental data, leading us to believe that ours is the most comprehensive study of its kind.

Our focus is on the most basic of strategic interactions: unrepeated (initial) play in simultaneous move games. In the behavioral game theory literature, five key paradigms have emerged for modeling human decision making in this setting: quantal response equilibrium [QRE; McKelvey and Palfrey, 1995]; the noisy introspection model [NI; Goeree and Holt, 2004]; the cognitive hierarchy model [CH; Camerer et al., 2004]; the closely related level-$k$ [Lk; Costa-Gomes et al., 2001; Nagel, 1995] models; and what we dub quantal level-$k$ [QLk; Stahl and Wilson, 1994] models. Although there exist studies exploring different variations of these models [e.g., Stahl and Wilson, 1995; Ho et al., 1998; Weizsäcker, 2003; Rogers et al., 2009], the overwhelming majority of behavioral models of initial play of normal-form games fall broadly into this categorization.

The first contribution of our work is methodological: we demonstrate broadly applicable techniques for comparing and analyzing behavioral models. We illustrate the use of these techniques via an extensive meta-analysis based on data published in ten different studies, rigorously comparing Lk, QLk, CH, NI, and QRE to each other and to a model based on Nash equilibrium. The findings that result from this meta-analysis both demonstrate the usefulness of the approach and constitute our second contribution.

All of these models depend upon exogenous parameters. Most previous work has focused on models' ability to *describe* human behavior, and hence has sought

parameter values that best explain observed experimental data, or more formally that maximize a dataset's probability. (All of the models that we consider make probabilistic predictions; thus, we must score models according to how much probability mass they assign to observed events, rather than assessing accuracy.) We depart from this descriptive focus, seeking to find models, and hence parameter values, that are effective for *predicting* previously unseen human behavior. Thus, we follow a different approach taken from machine learning and statistics. We begin by randomly dividing the experimental data into a training set and a test set. We then set each model's parameters to values that maximize the likelihood of the training dataset, and finally score each model according to the (disjoint) test dataset's likelihood. To reduce the variance of this estimate without biasing its expected value, we employ cross-validation [e.g., Bishop, 2006], systematically repeating this procedure with different test and training sets.

Our meta-analysis has led us to draw three qualitative conclusions. First, and least surprisingly, Nash equilibrium is less able to explain human play than behavioral models. Second, two high-level themes that underlie the five behavioral models (which we dub "cost-proportional errors" and "limited iterative strategic thinking") appear to model independent and predictively useful phenomena. Third, and building on the previous conclusion, the quantal level-$k$ model of Stahl and Wilson [1994] (QLk)—which combines both of these themes—made the most accurate predictions. Specifically, QLk substantially outperformed all other models on a new dataset spanning all data in our possession, and also had the best or nearly the best performance on each individual dataset. Our findings were quite robust to variation in the games played by human subjects. We broke down model performance by game properties such as dominance structure and number/types of equilibria, and obtained essentially the same results as on the combined dataset. We do note that our datasets consisted entirely of two-player games. Previous work suggests that human subjects reason about $n$-player games as if they were two-player games, failing to fully account for the independence of the other players' actions [Ho et al., 1998; Costa-Gomes et al., 2009]; we might thus expect to

observe qualitatively similar results in the $n$-player case. Nevertheless, empirically confirming this expectation is an important future direction.

In the next section, we define the models that we study. Section 2.3 lays out the formal framework within which we work, and Section 2.4 describes our data, methods, and the Nash-equilibrium-based model to which we compare the behavioral models. Section 2.5 presents the results of our comparisons. In Section 2.6 we survey related work from the literature and explain how our own work contributes to it. We conclude in Section 2.7.

## 2.2 Models for Predicting Human Play of Simultaneous-Move Games

Formally, a normal-form game $G$ is a tuple $(N, A, u)$, where $N$ is a finite set of *agents*; $A = \prod_{i \in N} A_i$ is the set of possible *action profiles*; $A_i$ is the finite set of *actions* available to agent $i$; $u = \{u_i\}_{i \in N}$ is a set of *utility functions* $u_i : A \to \mathbb{R}$, each of which maps from an action profile to a utility for agent $i$. Let $\Delta(X)$ denote the set of probability distributions over a finite set $X$. Overloading notation, we represent the expected utility of a profile of mixed strategies $s \in S = \prod_{i \in N} \Delta(A_i)$ by $u_i(s)$. We use the notation $a_{-i}$ to refer to the joint actions of all agents except for $i$.

A behavioral model is a mapping from a game $G$ and a vector of parameters $\theta$ to a predicted distribution over each action profile $a \in A$, which we denote $\Pr(a \mid G, \theta)$. In what follows, we define five prominent behavioral models of human play in unrepeated, simultaneous-move games.[1]

[1]We focus here on models of behavior in general one-shot normal-form games. We omit models of learning in repeated normal-form games such as impulse-balance equilibrium [Selten and Buchta, 1994], payoff-sampling equilibrium [Osborne and Rubinstein, 1998], action-sampling equilibrium [Selten and Chmura, 2008], and experience-weighted attraction [Camerer and Hua Ho, 1999], and models restricted to a single game class (e.g., symmetric games) such as cooperative equilibrium [Capraro, 2013]. We also omit variants and generalizations of the models we study, such as those introduced by Rogers et al. [2009], Weizsäcker [2003], and Cabrera et al. [2007]; but see Chapter 4, where we systematically explored a particular space of variants.

### 2.2.1 Quantal Response Equilibrium

One important idea from behavioral economics is that people become more likely to make errors as those errors become less costly; we call this making *cost-proportional errors*. This can be modeled by assuming that agents best respond *quantally*, rather than via strict maximization.

**Definition 1** (Quantal best response)**.** Let $u_i(a_i, s_{-i})$ be agent $i$'s expected utility in game $G$ when playing action $a_i$ against strategy profile $s_{-i}$. Then a *(logit) quantal best response* $QBR_i^G(s_{-i}; \lambda)$ by agent $i$ to $s_{-i}$ is a mixed strategy $s_i$ such that

$$s_i(a_i) = \frac{\exp[\lambda \cdot u_i(a_i, s_{-i})]}{\sum_{a_i'} \exp[\lambda \cdot u_i(a_i', s_{-i})]}, \tag{2.1}$$

where $\lambda$ (the *precision* parameter) indicates how sensitive agents are to utility differences, with $\lambda = 0$ corresponding to uniform randomization and $\lambda \to \infty$ corresponding to best response. When its value is clear from context, we will omit the precision parameter. Note that unlike best response, which is a set-valued function, quantal best response always returns a unique mixed strategy. □

The notion of quantal best response gives rise to a generalization of Nash equilibrium known as the *quantal response equilibrium* ("QRE") [McKelvey and Palfrey, 1995].

**Definition 2** (QRE)**.** A *quantal response equilibrium* with precision $\lambda$ is a mixed strategy profile $s^*$ in which every agent's strategy is a quantal best response to the strategies of the other agents. That is, $s_i^* = QBR_i^G(s_{-i}^*; \lambda)$ for all agents $i$. □

A QRE is guaranteed to exist for any normal-form game and non-negative precision [McKelvey and Palfrey, 1995]. However, it is not guaranteed to be unique. As is standard in the literature, we select the (unique) QRE that lies on the principal branch of the QRE homotopy at the specified precision. The principal branch has the attractive feature of approaching the risk-dominant equilibrium (as $\lambda \to \infty$) in $2 \times 2$ games with two strict equilibria [Turocy, 2005].

Although Equation (2.1) is translation-invariant, it is not scale invariant. That is, while adding some constant value to the payoffs of a game will not change its QRE, multiplying payoffs by a positive constant will. This is problematic because utility functions are only unique up to affine transformations [Von Neumann and Morgenstern, 1944]; hence, equivalent utility functions that have been multiplied by different constants will induce different QREs. The QRE concept nevertheless makes sense if human players are believed to play games differently depending on the magnitudes of the payoffs involved.

### 2.2.2 Level-k

Another key idea from behavioral economics is that humans can perform only a limited number of *iterations of strategic reasoning*. The level-$k$ model [Costa-Gomes et al., 2001] captures this idea by associating each agent $i$ with a level $k_i \in \{0, 1, 2, \ldots\}$, corresponding to the number of iterations of reasoning the agent is able to perform. A *level-*$0$ *agent* plays randomly, choosing uniformly from his possible actions. A *level-*$k$ *agent*, for $k \geq 1$, best responds to the strategy played by level-$(k-1)$ agents. If a level-$k$ agent has more than one best response, he mixes uniformly over them.

Here we consider a particular level-$k$ model, dubbed Lk, which assumes that all agents belong to levels 0,1, and 2.[2] Each agent with level $k > 0$ has an associated probability $\epsilon_k$ of making an "error," i.e., of playing an action that is not a best response to the level-$(k-1)$ strategy. Agents are assumed not to account for these errors when forming their beliefs about how lower-level agents will act.

**Definition 3** (Lk model)**.** Let $A_i$ denote player $i$'s action set, and $BR_i^G(s_{-i})$ denote the set of $i$'s best responses in game $G$ to the strategy profile $s_{-i}$. Let $IBR_{i,k}^G$ denote the *iterative best response set* for a level-$k$ agent $i$, with $IBR_{i,0}^G = A_i$ and $IBR_{i,k}^G = BR_i^G(IBR_{-i,k-1}^G)$. Then the distribution $\pi_{i,k}^{Lk} \in \Pi(A_i)$ that the Lk

[2] We here model only level-$k$ agents, unlike Costa-Gomes et al. [2001] who also modeled other decision rules.

model predicts for a level-$k$ agent $i$ is defined as

$$\pi^{Lk}_{i,0}(a_i) = |A_i|^{-1},$$
$$\pi^{Lk}_{i,k}(a_i) = \begin{cases} (1-\epsilon_k)/|IBR^G_{i,k}| & \text{if } a_i \in IBR^G_{i,k}, \\ \epsilon_k/(|A_i| - |IBR^G_{i,k}|) & \text{otherwise.} \end{cases}$$

The overall predicted distribution of actions is a weighted sum of the distributions for each level:

$$\Pr(a_i \mid G, \alpha_1, \alpha_2, \epsilon_1, \epsilon_2) = \sum_{\ell=0}^{2} \alpha_\ell \cdot \pi^{Lk}_{i,\ell}(a_i),$$

where $\alpha_0 = 1 - \alpha_1 - \alpha_2$. This model thus has 4 parameters: $\{\alpha_1, \alpha_2\}$, the proportions of level-1 and level-2 agents, and $\{\epsilon_1, \epsilon_2\}$, the error probabilities for level-1 and level-2 agents. □

### 2.2.3 Cognitive Hierarchy

The cognitive hierarchy model [Camerer et al., 2004], like level-$k$, models agents with heterogeneous bounds on iterated reasoning. It differs from the level-$k$ model in two ways. First, according to this model agents do not make errors; each agent always best responds to its beliefs. Second, agents of level-$m$ best respond to the full distribution of agents at the lower levels 0–$(m-1)$, rather than only to level-$(m-1)$ agents. More formally, every agent has an associated level $m \in \{0, 1, 2, \ldots\}$. Let $f$ be a probability mass function describing the distribution of the levels in the population. Level-0 agents play uniformly at random. Level-$m$ agents ($m \geq 1$) best respond to the strategies that would be played in a population described by the truncated probability mass function $f(j \mid j < m)$.

Camerer et al. [2004] advocate a single-parameter restriction of the cognitive hierarchy model called *Poisson-CH*, in which $f$ is a Poisson distribution.

**Definition 4** (Poisson-CH model)**.** Let $\pi^{PCH}_{i,m} \in \Pi(A_i)$ be the distribution over actions predicted for an agent $i$ with level $m$ by the Poisson-CH model. Let $f(m) = \text{Poisson}(m; \tau)$. Let $BR^G_i(s_{-i})$ denote the set of $i$'s best responses in

game $G$ to the strategy profile $s_{-i}$. Let

$$\pi^{PCH}_{i,0:m} = \sum_{\ell=0}^{m} f(\ell) \frac{\pi^{PCH}_{i,\ell}}{\sum_{\ell'=0}^{m} f(\ell')}$$

be the truncated distribution over actions predicted for an agent conditional on that agent's having level $0 \leq \ell \leq m$. Then $\pi^{PCH}$ is defined as

$$\pi^{PCH}_{i,0}(a_i) = |A_i|^{-1},$$
$$\pi^{PCH}_{i,m}(a_i) = \begin{cases} |BR^G_i(\pi^{PCH}_{i,0:m-1})|^{-1} & \text{if } a_i \in BR^G_i(\pi^{PCH}_{i,0:m-1}), \\ 0 & \text{otherwise.} \end{cases}$$

The overall predicted distribution of actions is a weighted sum of the distributions for each level,

$$\Pr(a_i \,|\, G, \tau) = \sum_{\ell=0}^{\infty} f(\ell) \cdot \pi^{PCH}_{i,\ell}(a_i).$$

The mean of the Poisson distribution, $\tau$, is thus this model's single parameter. □

Rogers et al. [2009] note that cognitive hierarchy and QRE often make similar predictions. One possible explanation for this is that cost-proportional errors are adequately captured by cognitive hierarchy (and other iterative models), even though they do not explicitly model this effect. Alternatively, these phenomena could be sufficiently distinct that explicitly modeling both limited iterative strategic thinking and cost-proportional errors yields improved predictions.

### 2.2.4 Quantal Level-k

Stahl and Wilson [1994] propose a rich model of strategic reasoning that combines elements of the QRE and level-$k$ models; we refer to it as the QLk model (for quantal level-$k$). In QLk, agents have one of three levels, as in Lk.[3] Each agent

[3]Stahl and Wilson [1994] also consider an extended version of this model that adds a type that plays the equilibrium strategy. In order to avoid the complication of having to specify an equilibrium selection rule, we do not consider this extension (as many of the games in our dataset

responds to its beliefs quantally, as in QRE.

A key difference between QLk and Lk is in the error structure. In Lk, higher-level agents believe that all lower-level agents best respond perfectly, although in fact every agent has some probability of making an error. In contrast, in QLk, agents are aware of the quantal nature of the lower-level agents' responses, but have (possibly incorrect) beliefs about the lower-level agents' precision. That is, level-1 and level-2 agents use potentially different precisions ($\lambda$'s), and furthermore level-2 agents' beliefs about level-1 agents' precision can be wrong.

**Definition 5** (QLk model)**.** The probability distribution $\pi_{i,k}^{QLk} \in \Pi(A_i)$ over actions that QLk predicts for a level-$k$ agent $i$ is

$$\begin{aligned}
\pi_{i,0}^{QLk}(a_i) &= |A_i|^{-1}, \\
\pi_{i,1}^{QLk} &= QBR_i^G(\pi_{-i,0}^{QLk}; \lambda_1), \\
\pi_{i,1(2)}^{QLk} &= QBR_i^G(\pi_{-i,0}^{QLk}; \lambda_{1(2)}), \\
\pi_{i,2}^{QLk} &= QBR_i^G(\pi_{i,1(2)}^{QLk}; \lambda_2),
\end{aligned}$$

where $\pi_{i,1(2)}^{QLk}$ is a mixed-strategy profile representing level-2 agents' prediction of how other agents will play. This can be interpreted either as the level-2 agents' beliefs about the behavior of level-1 agents alone, or it can be understood as modeling level-2 agents' beliefs about both level-1 and level-0 agents, with the presence of additional level-0 agents being captured by a lower precision $\lambda_{1(2)}$. Stahl and Wilson [1994] advocate the latter interpretation. The overall predicted distribution of actions is the weighted sum of the distributions for each level,

$$\Pr(a_i \,|\, G, \alpha_1, \alpha_2, \lambda_1, \lambda_2, \lambda_{1(2)}) = \sum_{k=0}^{2} \alpha_k \pi_{i,k}^{QLk}(a_i),$$

where $\alpha_0 = 1 - \alpha_1 - \alpha_2$. The QLk model thus has five parameters: $\{\alpha_1, \alpha_2, \lambda_1, \lambda_2, \lambda_{1(2)}\}$. □

have multiple equilibria). See Section 2.4.2 for bounds on the performance of Nash equilibrium predictions on our dataset.

### 2.2.5 Noisy Introspection

Goeree and Holt [2004] propose a model called *noisy introspection* that combines cost-proportional errors and an iterative view of strategic cognition in a different way. Rather than assuming a fixed limit on the number of iterations of strategic thinking, they instead model cognitive bounds by injecting noise into iterated beliefs about others' beliefs and decisions, with the effect that deeper levels of reasoning are assumed to be noisier. They then show that this process of noise injection converges to a unique prediction after a finite number of iterations, which for most games is relatively small.

Goeree and Holt also introduce a concrete version of this model, in which deeper levels of reasoning are exponentially noisier. We refer to this restricted version as the NI model.

**Definition 6** (NI model)**.** Define $\pi_{i,k}^{NI,n}$ as

$$\pi_{i,k}^{NI,n} = \begin{cases} QBR_i^G(\pi_{-i,k+1}^{NI,n}; \lambda_0/t^k) & \text{if } k < n, \\ QBR_i^G(p_0; \lambda_0/t^n) & \text{otherwise,} \end{cases}$$

where $p_0$ is an arbitrary mixed profile, $\lambda_0 \geq 0$ is a precision, and $t > 1$ is a "telescoping" parameter that determines how quickly noise increases with depth of reasoning. Then the NI model predicts that each agent will play according to

$$\pi_i^{NI} = \lim_{n \to \infty} \pi_{i,0}^{NI,n}.$$

For a fixed game $G$, precision $\lambda_0$, and telescoping parameter $t$, this converges to a unique strategy profile regardless of the choice of $p_0$ (since in the limit the precision becomes low enough to bring any profile arbitrarily close to the uniform distribution.) □

## 2.3 Comparing Models

### 2.3.1 Prediction Framework

How do we determine whether a behavioral model is well supported by experimental data? An experimental dataset $\mathcal{D} = \{(G_i, \{a_{ij} \mid j = 1, \ldots, J_i\}) \mid i = 1, \ldots, I\}$ is a set containing $I$ elements. Each element is a tuple containing a game $G_i$ and a set of $J_i$ pure actions $a_{ij}$, each played by a human subject in $G_i$. For symmetric games, we treat all actions as being played by the first player. For non-symmetric games, the player is implicit in the action being chosen (that is, $J_i$ contains a separate entry for each of the first and second players' actions). There is no reason to maintain the pairing of the play of a human player with that of his opponent, as games are unrepeated. Recall that a behavioral model is a mapping from a game $G_i$ and a vector of parameters $\theta$ to a predicted distribution over each action $a_i$ in $G_i$, which we denote $\Pr(a_i \mid G_i, \theta)$.

A behavioral model can only be used to make predictions when its parameters are instantiated. How should we set these parameters? Our goal is a model that produces accurate probability distributions over the actions of human agents, rather than simply determining the single action most likely to be played. This means that we cannot score different models (or, equivalently, different parameter settings for the same model) using a criterion such as a 0–1 loss function (accuracy), which asks how many actions were accurately predicted. (For example, the 0–1 loss function evaluates models based purely upon which action is assigned the highest probability, and does not take account of the probabilities assigned to the other actions.) Instead, we evaluate a given model on a given dataset by *likelihood*. That is, we compute the probability of the observed actions according to the distribution over actions predicted by the model. The higher the probability of the actual observations according to the prediction output by a model, the better the model predicted the observations. This takes account of the full predicted distribution; in particular, for any given observed distribution, the prediction that

maximizes the likelihood score is the observed distribution itself.[4]

Assume that there is some true set of parameter values, $\theta^*$, under which the model outputs the true distribution $\Pr(a \mid G, \theta^*)$ over action profiles, and that $\theta^*$ is independent of $G$. The maximum likelihood estimate of the parameters based on $\mathcal{D}$,

$$\hat{\theta} = \arg\max_{\theta} \Pr(\mathcal{D} \mid \theta),$$

is a point estimate of the true set of parameters $\theta^*$, whose variance decreases as $I$ grows. We then use $\hat{\theta}$ to evaluate the model:

$$\Pr(a \mid G, \mathcal{D}) = \Pr(a \mid G, \hat{\theta}). \tag{2.2}$$

The likelihood of a single datapoint $d_{ij} = (G_i, a_{ij})$ is

$$\Pr(d_{ij} \mid \theta) = \Pr(G_i, a_{ij} \mid \theta).$$

By the chain rule of probabilities, this[5] is equivalent to

$$\Pr(d_{ij} \mid \theta) = \Pr(a_{ij} \mid G_i, \theta) \Pr(G_i \mid \theta),$$

and by independence of $G$ and $\theta$ we have

$$\Pr(d_{ij} \mid \theta) = \Pr(a_{ij} \mid G_i, \theta) \Pr(G_i). \tag{2.3}$$

The datapoints are independent, so the likelihood of the dataset is just the product

[4] Although the likelihood is what we are interested in, in practice we operate on the log of the likelihood to avoid range problems. Since log likelihood is a monotonic function of likelihood, a model that has higher likelihood than another model will always also have higher log likelihood, and vice versa.

[5] To those unfamiliar with Bayesian analysis, quantities such as $\Pr(\mathcal{D})$, $\Pr(G_i)$, and $\Pr(G_i \mid \theta)$ may seem difficult to interpret or even nonsensical. It is common practice in Bayesian statistics to assign probabilities to any quantity that can vary, such as the games under consideration or the complete dataset that has been observed. Regardless of how they are interpreted, these quantities all turn out to be constant with respect to $\theta$, and so have no influence on the outcome of the analysis.

of the likelihoods of the datapoints,

$$\Pr(\mathcal{D} \mid \theta) = \prod_{i=1}^{I} \prod_{j=1}^{J_i} \Pr(a_{ij} \mid G_i, \theta) \Pr(G_i). \tag{2.4}$$

The probabilities $\Pr(G_i)$ are constant with respect to $\theta$, and can therefore be disregarded when maximizing the likelihood:[6]

$$\arg\max_{\theta} \Pr(\mathcal{D} \mid \theta) = \arg\max_{\theta} \prod_{i=1}^{I} \prod_{j=1}^{J_i} \Pr(a_{ij} \mid G_i, \theta).$$

### 2.3.2 Assessing Generalization Performance

Each of the models that we consider depends on parameters that are estimated from the data. In such settings, one must be careful to avoid the problem of overfitting the data, where the most flexible model can be preferred to the most accurate model. We avoid this problem by estimating parameters on a dataset containing observations from a subset of the games in our dataset (the *training data*) and then evaluating the resulting model by computing likelihood scores on the observations associated with the remaining, disjoint *test data*. That is, every model's performance is evaluated entirely based on *games* that were not used for estimating parameters.

Randomly dividing our experimental data into training and test sets introduces variance into the prediction score, since the exact value of the score depends partly upon the random division. To reduce this variance, we perform 10 rounds of 10-fold *cross-validation*. Specifically, for each round, we randomly partition the games into 10 parts of approximately equal size. For each of the 10 ways of selecting 9 parts from the 10, we compute the maximum likelihood estimate of the model's parameters based on the observations associated with the games of those 9

[6]That is, the values of $\Pr(G_i)$ do not vary as $\theta$ varies. The same number of $\Pr(G_i)$ get multiplied at the end of Equation (2.4) regardless of the value of $\theta$, and hence amount to a single constant that can be disregarded.

parts. We then determine the likelihood of the remaining part given the prediction. We call the average of this quantity across all 10 parts the *cross-validated likelihood*. The average (across rounds) of the cross-validated likelihoods is distributed according to a Student's-$t$ distribution [see, e.g., Witten and Frank, 2000]. We compare the predictive power of different behavioral models on a given dataset by comparing the average cross-validated likelihood of the dataset under each model. We say that one model predicts significantly better than another when the $95\%$ confidence intervals for the average cross-validated likelihoods do not overlap.

## 2.4 Experimental Setup

In this section we describe the data and methods that we used in our model evaluations. We also describe a baseline model based on Nash equilibrium.

### 2.4.1 Data

As described in detail in Section 2.6, we conducted an exhaustive survey of papers that make use of the five behavioral models we consider. We thereby identified ten large-scale, publicly available sets of human-subject experimental data [Stahl and Wilson, 1994, 1995; Costa-Gomes et al., 1998; Goeree and Holt, 2001; Haruvy et al., 2001; Cooper and Van Huyck, 2003; Haruvy and Stahl, 2007; Costa-Gomes and Weizsäcker, 2008; Stahl and Haruvy, 2008; Rogers et al., 2009]. We study all ten[7] of these datasets in this paper, and describe each briefly in what follows.

[7] We identified an additional dataset [Costa-Gomes and Crawford, 2006] which we do not include due to a computational issue. The games in this dataset had between $200$ and $800$ actions per player, which made it intractable to compute many solution concepts. As with Nash equilibrium, the main bottleneck in computing behavioral solution concepts is computing expected utilities. Each epoch of training for this dataset requires taking expected utility over up to $640,000$ outcomes per game, in contrast to between $9$ and approximately $14,000$ outcomes per game in the ALL10 dataset. We attempted to evaluate a coarse version of this data by binning similar actions; however, binning in this way results in games that are not strategically equivalent to the originals (e.g., when multiple iterations of best response would result in the same binned action in the coarsened games but different unbinned actions in the original games). The best way of addressing this computational problem would be to represent the games *compactly* [e.g., Kearns et al., 2001; Koller and Milch, 2001; Jiang et al., 2011], such that expected utility can be computed efficiently

In Stahl and Wilson [1994] experimental subjects played 10 normal-form games for points, where every point represented a 1% chance (per game) of winning $\$2.50$. Participants stood to earn between $\$0.25$ and $\$25.00$ based on their play in the games.

In Stahl and Wilson [1995], subjects played 12 normal-form games, where each point gave a 1% chance (per game) of winning $\$2.00$. Participants stood to earn between $\$0.00$ and $\$24.00$ based on their play in the games.

In Costa-Gomes et al. [1998] subjects played 18 normal-form games, with each point of payoff worth 40 cents. However, subjects were paid based on the outcome of only one randomly-selected game. Participants stood to earn between $\$7.84$ and $\$36.16$ based on their play in the games. Goeree and Holt [2001] presented 10 games in which subjects' behavior was close to that predicted by Nash equilibrium, and 10 other small variations on the same games in which subjects' behavior was *not* well-predicted by Nash equilibrium. The payoffs for each game were denominated in pennies. We included the 10 games that were in normal form. Participants stood to earn between $\$ - 1.02$ and $\$23.30$ based on their play in these 10 games.

In Cooper and Van Huyck [2003], agents played the normal forms of 8 games, followed by extensive form games with the same induced normal forms; we include only the data from the normal-form games. Payoffs were denominated in 10 cent units. Participants stood to earn between $\$0.80$ and $\$4.80$ based on their play in the games.

In Haruvy et al. [2001], subjects played 15 symmetric $3 \times 3$ normal form games. The payoffs were points representing a percentage chance of winning $2.00 for each game. Participants stood to earn between $\$0.00$ and $\$30.00$ based on their play in the games.

In Costa-Gomes and Weizsäcker [2008], subjects played 14 games, and were paid $0.15 per point in one randomly-chosen game. Participants stood to earn between $\$1.83$ and $\$14.13$ based on their play in the games.

over even a very large action space.

In Haruvy and Stahl [2007], subjects played 20 games, again for payoff points representing a percentage chance of winning \$2.00 per game. Participants stood to earn between $\$1.05$ and $\$17.40$ based on their play in the games.

Stahl and Haruvy [2008] present new data on 15 games that contain strategies that are dominated in ways that are "obvious" to varying degrees, again for percentage chances of winning \$2.00 per game. Participants stood to earn between $\$0.00$ and $\$17.55$ based on their play in the games.

Finally, in Rogers et al. [2009], subjects played 17 normal-form games, with payoffs denominated in pennies. Participants stood to earn between $\$2.31$ and $\$13.33$ based on their play in the games.

We represent the data for each game $G_i$ as a pair $(G_i, \{a_{ij}\})$ containing the game itself and a set of observed actions in the game, as in Section 2.3.1. All games had two players, so each single play of a game generated two observations. We built one such dataset for each study, as listed in Table 2.1. We also constructed a combined dataset, dubbed ALL10, containing data from all the datasets. The datasets contained very different numbers of observations, ranging from 400 [Stahl and Wilson, 1994] to 2992 [Cooper and Van Huyck, 2003]. To ensure that each fold had approximately the same population of subjects and games, we evaluated ALL10 using *stratified* cross-validation: we performed the game partitioning and selection process separately for each of the contained source datasets, thereby ensuring that the number of games from each source dataset was approximately equal in each partition element.

The QRE and QLk models depend on a precision parameter that is not scale invariant. E.g., if $\lambda$ is the correct precision for a game whose payoffs are denominated in cents, then $\lambda/100$ would be the correct precision for a game whose payoffs are denominated in dollars. To ensure consistent estimation of precision parameters, especially in the ALL10 dataset where observations from multiple studies were combined, we normalized the payoff values for each game to be in expected cents. As described earlier, in some datasets, payoff points were worth a certain number of cents; in others, points represented percentage chances of win-

**Table 2.1:** Names and contents of each dataset. Units are in expected value, in US dollars.

| Name | Source | Games | $n$ | Units |
|---|---|---|---|---|
| SW94 | Stahl and Wilson [1994] | 10 | 400 | \$0.025 |
| SW95 | Stahl and Wilson [1995] | 12 | 576 | \$0.02 |
| CGCB98 | Costa-Gomes et al. [1998] | 18 | 1566 | \$0.022 |
| GH01 | Goeree and Holt [2001] | 10 | 500 | \$0.01 |
| CVH03 | Cooper and Van Huyck [2003] | 8 | 2992 | \$0.10 |
| HSW01 | Haruvy et al. [2001] | 15 | 869 | \$0.02 |
| HS07 | Haruvy and Stahl [2007] | 20 | 2940 | \$0.02 |
| CGW08 | Costa-Gomes and Weizsäcker [2008] | 14 | 1792 | \$0.0107 |
| SH08 | Stahl and Haruvy [2008] | 18 | 1288 | \$0.02 |
| RPC08 | Rogers et al. [2009] | 17 | 1210 | \$0.01 |
| ALL10 | Union of above | 142 | 13863 | per source |

ning a certain sum, or were otherwise in expected units. Table 2.1 lists the number of expected cents that we deemed each payoff point to be worth for the purposes of normalization.

### 2.4.2 Comparing to Nash Equilibrium

It is desirable to compare the predictive performance of our behavioral models to that of Nash equilibrium. However, such a comparison is not as simple as one might hope, because any attempt to use Nash equilibrium for prediction must extend the solution concept to address two problems. The first problem is that many games have multiple Nash equilibria; in these cases, the Nash prediction is not well defined. The second problem is that Nash equilibrium frequently assigns probability zero to some actions. Indeed, in 82% of the games in our ALL10 dataset *every* Nash equilibrium assigned probability 0 to actions that were actually taken by one or more experimental subjects. This is a problem because we

assess the quality of a model by how well it explains the data; unmodified, Nash equilibrium model considers our experimental data to be *impossible*, and hence receives a likelihood of zero.

We addressed the second problem by augmenting the Nash equilibrium solution concept to say that with some probability, each player chooses an action uniformly at random; this prevents the solution concept from assessing any experimental data as impossible. This probability is a free parameter of the model; as we did with behavioral models, we fit this parameter using maximum likelihood estimation on a training set. We thus call the model Nash Equilibrium with Error (NEE). We sidestepped the first problem by assuming that agents always coordinate to play an equilibrium and by reporting statistics across different equilibria. Specifically, we report the performance achieved by choosing the equilibrium that respectively best and worst fit the *test* data, thereby giving upper and lower bounds on the test-set performance achievable by any Nash-based prediction. (Note that because we "cheat" by choosing equilibria based on test-set performance, these models are not able to generalize to new data, and hence cannot be used in practice.) Finally, we also reported the prediction performance on the test data, averaged over all of the Nash equilibria of the game.[8]

### 2.4.3 Computational Environment

We performed computation using WestGrid (www.westgrid.ca), primarily on the `orcinus` cluster, which has $9600$ 64-bit Intel Xeon CPU cores. We used GAMBIT [McKelvey et al., 2007] to compute QRE and to enumerate the Nash equilibria of games, and computed maximum likelihood estimates using the Covariance Matrix Adaptation Evolution Strategy (CMA-ES) algorithm [Hansen and Ostermeier,

[8]One might wonder whether the $\epsilon$-equilibrium solution concept [e.g., Shoham and Leyton-Brown, 2008, Section 3.4.7] solves either of these problems. It does not. First, $\epsilon$-equilibrium can still assign probability 0 to some actions, unlike NEE which will always assign at least $\epsilon$. Second, relaxing the equilibrium concept only increases the number of equilibria; indeed, every game has infinitely many $\epsilon$-equilibria for any $\epsilon > 0$. Furthermore, to our knowledge, no algorithm for characterizing this set exists, making equilibrium selection impractical.

2001].

## 2.5 Model Comparisons

In this section we describe the results of our experiments comparing the predictive performance of the five behavioral models from Section 2.2 and of the Nash-based models of Section 2.4.2. Figure 2.1 compares our behavioral and Nash-based models. For each model and each dataset, we give the factor by which the dataset was judged more likely according to the model's prediction than it was according to a uniform random prediction. Thus, for example, the ALL10 dataset was found to be approximately $10^{90}$ times more likely according to Poisson-CH's prediction than according to a uniform random prediction. For the Nash Equilibrium with Error model, the error bars show the upper and lower bounds on predictive performance obtained by selecting an equilibrium to maximize or minimize test-set performance, and the main bar shows the expected predictive performance of selecting an equilibrium uniformly at random. For other models, the error bars indicate 95% confidence intervals across cross-validation partitions; in most cases, these intervals are imperceptibly narrow.

### 2.5.1 Comparing Behavioral Models

Poisson-CH and Lk had very similar performance in most datasets. In one way this is an intuitive result, since the models are very similar to each other. Poisson-CH and Lk had very similar performance in most datasets. In one way this is an intuitive result, since the models are very similar to each other. On the other hand, it suggests that the distinction between reasoning about just one lower level versus reasoning about the distribution of all lower levels, and the distinct error models, does not make much difference, which is perhaps less obvious.

QRE and NI tended to perform well on the same datasets. On all but two datasets (HSW01 and CGW08), the ordering between QRE and the iterative models was the same as between NI and the iterative models. We found this

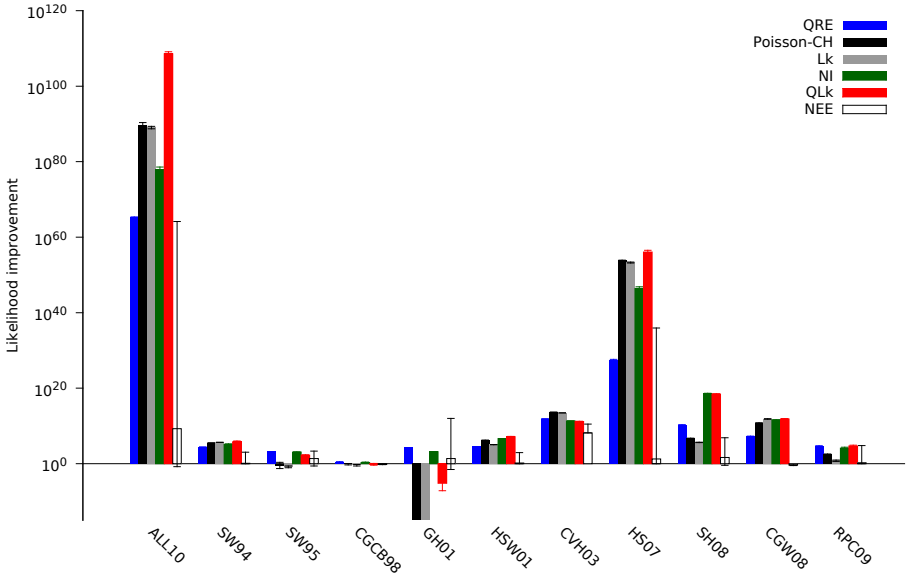


**Figure 2.1:** Average likelihood ratios of model predictions to random predictions, with $95\%$ confidence intervals. Error bars for NEE show upper and lower bounds on performance depending upon equilibrium selection; the main bar for NEE shows the average performance over all equilibria. Note that relative differences in likelihood are not meaningful across datasets, as likelihood drops with growth in the dataset's number of samples and underlying games' numbers of actions. Relative differences in likelihood *are* meaningful within datasets.

result more surprising, since the two models appear quite different. However, cost-proportional errors are a key element of each, and they both assume that all agents will be playing from the same distribution, unlike the iterative models, which assume that different agents will reason to different depths. Further, although NI is not explicitly a fixed-point model, it does assume an unlimited depth of reasoning, like QRE (albeit typically converging after a relatively small number of iterations).

In five datasets, the models based on cost-proportional errors (QRE and NI) predicted human play significantly better than the two models based on bounded iterated reasoning (Lk and Poisson-CH). However, in five other datasets, includ-

**Table 2.2:** Proportion of level-0 agents estimated by previous studies. Burchardi and Penczynski [2011] estimated the proportions of level-0 agents both by fitting a level-$k$ model, and by directly eliciting subjects' strategies; we list both estimates.

| Study | Estimated proportion of level-0 |
|---|---|
| Stahl and Wilson [1994] | 0% |
| Stahl and Wilson [1995] | 6–30% |
| Haruvy et al. [2001] | 6–16% |
| Burchardi and Penczynski [2011] | 37% (by fitting) |
| Burchardi and Penczynski [2011] | 20–42% (by direct elicitation) |

ing ALL10, the situation was reversed, with Lk and Poisson-CH outperforming QRE and NI. In the remaining two datasets, NI outperformed the iterative models, which outperformed QRE. This mixed result is consistent with earlier, less exhaustive comparisons of QRE with these two models [Chong et al., 2005; Crawford and Iriberri, 2007a; Rogers et al., 2009, see also Section 2.6], and suggests to us that, in answer to the question posed in Section 2.2.3, there may be value to modeling both bounded iterated reasoning and cost-proportional errors explicitly. If this hypothesis is true, we might expect that our remaining model, which incorporates both components, would predict better than models that are based on only one component. This was indeed the case: QLk generally outperformed the single-component models. Overall, QLk was the strongest behavioral model; no model predicted significantly better in the majority of datasets. The exceptions were CVH03, SW95, CGCB98, and GH01; we discuss the latter in detail below, in Section 2.5.2.

**Level-0**

Earlier studies found support for quite variable proportions of level-0 agents; see Table 2.2. Our fitted parameters for the Lk and QLk models estimate proportions of level-0 agents that are toward the high end of this range (8% and 34% respectively on the ALL10 dataset). However, note that our estimate for QLk is very

similar to the fitted estimate of Burchardi and Penczynski [2011], and comfortably within the range that they estimated by directly evaluating subjects' elicited strategies in a single game. We analyze the full distributions of parameter values in Chapter 3. In contrast to our estimates, the number of level-0 agents in the population is typically assumed to be negligible in studies that use an iterative model of behavior. Indeed, some studies [e.g., Crawford and Iriberri, 2007b] fix the number of level-0 agents to be 0. Thus, one possible interpretation of our higher estimates of level-0 agents is as evidence of a misspecified model. For example, Poisson-CH uses level-0 agents as the only source of noisy responses. However, we estimated substantial proportions of level-0 agents even for models (Lk and QLk) that include explicit error structures. We thus believe that the alternative—the possibility that these results point to a substantial frequency of nonstrategic behavior[9]—must be taken seriously.

### 2.5.2 Comparing to Nash Equilibrium

It is already widely believed that Nash equilibrium—especially without refinements—is a poor description of humans' initial play in normal-form games [e.g., Goeree and Holt, 2001]. Nevertheless, for the sake of completeness, we also evaluated the predictive power of Nash equilibrium with error on our datasets. Referring again to Figure 2.1, we see that NEE's predictions were worse than those of every behavioral model on every dataset except SW95 and CGCB98. NEE's upper bound—using the post-hoc best equilibrium—was significantly worse than QLk's performance on every dataset except SW95, CGCB98, RPC09, and GH01.

NEE's strong performance on SW95 was surprising; it may have been a result of the unusual subject pool, which consisted of fourth and fifth year undergraduate finance and accounting majors. In contrast, it is unsurprising that NEE performed well on GH01, since this distribution was deliberately constructed so that human

[9] Although the models we present here typically specify level-0 behavior as uniform random choice, applications sometimes specify other nonstrategic level-0 behavior [e.g., Crawford and Iriberri, 2007a; Arad and Rubinstein, 2012].

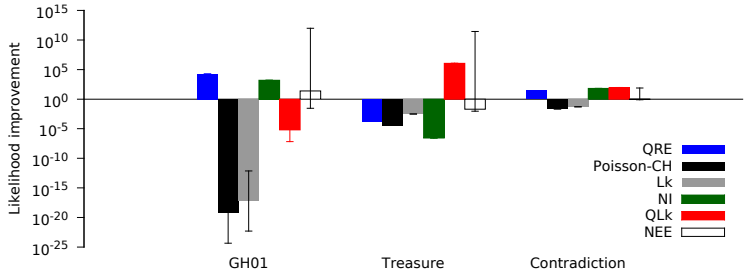


**Figure 2.2:** Average likelihood ratios of model predictions to random predictions, with $95\%$ confidence intervals, on GH01 data separated into "treasure" and "contradiction" treatments. The results in this figure are from cross-validation performed separately on the two treatments. Error bars for NEE show upper and lower bounds on performance depending upon equilibrium selection; the main bar for NEE shows the average performance over all equilibria. Note that relative differences in likelihood are not meaningful across datasets, as likelihood drops with growth in the dataset's number of samples and underlying games' numbers of actions. Relative differences in likelihood *are* meaningful within datasets.

play on half of its games (the "treasure" conditions) would be relatively well described by Nash equilibrium.[10] Figure 2.2 separates GH01 into its "treasure" and "contradiction" treatments and compares the performance of the behavioral and Nash-based models on these separated datasets. In addition to the fact that the "treasure" games were deliberately selected to favor Nash predictions, many of GH01's games have multiple equilibria. This offers an advantage to our NEE model's upper bound, because it gets to pick the equilibrium with best test-set performance on a per-instance basis (see Section 2.5.3). Note that although NEE thus had a higher upper bound than QLk on the "treasure" treatment, its average performance was still quite poor.

NEE has a free parameter, $\epsilon$, that describes the probability of an agent choosing

[10]Of course, GH01 was also constructed so that human play on the other half of its games would be poorly described by Nash equilibrium. However, this is still a difference from the other datasets, in which Nash equilibrium seems to have been a poor description of the *majority* of games.

an action uniformly at random. If Nash equilibrium were a good tool for predicting human behavior, we would expect to find this parameter set to a low value; in contrast, the values of $\epsilon$ that maximize NEE's performance were extremely high. On the ALL10 dataset, a value of $\epsilon = 0.87$ maximized NEE's average-case performance. Even best-case performance, which is computed by choosing the post-hoc performance-maximizing equilibrium for each game, was optimized by $\epsilon = 0.64$. Thus, the fact that well over half of NEE's prediction consists of the uniform noise term provides a strong argument against using Nash equilibrium to predict initial play. This is especially true as the agents within a Nash equilibrium do not take others' noisiness into account, which makes it difficult to interpret $\epsilon$ as a measure of level-0 play rather than of model misspecification.

### 2.5.3 Dataset Composition

As we have already seen in the case of GH01, model performance was sensitive to choices made by the authors of our various datasets about which games to study. One way to control for such choices is to partition our set of games according to important game properties, and to evaluate model performance in each partition. In this section we describe such an analysis.

Overall, our datasets spanned 142 games. The vast majority of these games are matrix games, deliberately lacking inherent meaning in order to avoid framing effects. (Indeed, some studies [e.g., Rogers et al., 2009] even avoid focal payoffs like 0 and 100.) For the most part, these games were chosen to vary according to dominance solvability and equilibrium structure. In particular, most dataset authors were concerned with (1) whether a game could be solved by iterated removal of dominated strategies (either strict or weak) and with how many steps of iteration were required; and (2) the number and type of Nash equilibria that each game possesses.[11]

[11] There were two exceptions. The first was Goeree and Holt [2001], who chose games that had both equilibria that human subjects find intuitive and strategically equivalent variations of these games whose equilibria human subjects find counterintuitive. The second exception was Cooper and Van Huyck [2003], whose normal form games were based on an exhaustive enumeration of

**Table 2.3:** Datasets conditioned on various game features. The column headed "games" indicates how many games of the full dataset meet the criterion, and the column headed "$n$" indicates how many observations each feature-based dataset contains. Observe that the game features are not all mutually exclusive, and so the "games" column does not sum to 142.

| Name | Description | Games | $n$ |
|---|---|---|---|
| D1 | Weak dominance solvable in one round | 2 | 748 |
| D1S | Strict dominance solvable in one round | 0 | 0 |
| D2 | Weak dominance solvable in two rounds | 38 | 5058 |
| D2S | Strict dominance solvable in two rounds | 23 | 2000 |
| DS | Weak dominance solvable in any number of rounds | 52 | 6470 |
| DSS | Strict dominance solvable in any number of rounds | 35 | 3312 |
| ND | Not dominance solvable | 90 | 7393 |
| PSNE1 | Single Nash equilibrium, which is pure | 51 | 4687 |
| MSNE1 | Single Nash equilibrium, which is mixed | 21 | 1387 |
| MULTI-EQM | Multiple Nash equilibria | 70 | 7789 |

We thus constructed subsets of the full dataset based on their dominance solvability and the nature of their Nash equilibria, as described in Table 2.3. We computed cross-validated MLE fits for each model on each of the feature-based datasets of Table 2.3. The results are summarized in Figure 2.3. In two respects, the results across the feature-based datasets mirror the results of Section 2.5.1 and Section 2.5.2. First, QLk significantly outperformed the other behavioral models on the majority of datasets; the exceptions are D1, D2, and D2S (but not DS); and MSNE1. Second, a majority of behavioral models significantly outperformed NEE in all but three datasets: D1, ND and MULTI-EQM. In these three datasets, the upper and lower bounds on NEE's performance contained the performance of either two or all three of the single-factor behavioral models (but not necessarily

the payoff orderings possible in generic 2-player, 2-action extensive-form games.

QLk). It is unsurprising that NEE's upper and lower bounds were widely separated on the MULTI-EQM dataset, since the more equilibria a game has, the more variation there can be in these equilibria's post-hoc performance; NEE's strong best-case performance on this dataset should similarly reflect this variation. It turns out that 55 of the 90 games (and 4731 of the 7393 observations) in the ND dataset are from the MULTI-EQM dataset, which likely explains NEE's high upper bound in that dataset as well. Indeed, this analysis helps to explain some of our previous observations about the GH01 dataset. NEE contains all other models in its performance bounds in this dataset, and in addition to the fact that half the dataset's games (the "treasure" treatments) that were chosen for consistency with Nash equilibrium, some of the other games (the "contradiction" treatments) turn out to have multiple equilibria. Overall, the overlap between GH01 and MULTI-EQM is 5 games out of 10, and 250 observations out of 500.

Unlike in the per-dataset comparisons of Section 2.5.1, both of our iterative single-factor models (Poisson-CH and Lk) significantly outperformed QRE in almost every feature-based dataset, with D2S and DSS as the only exceptions; in D2S, QRE outperformed all other models, and in DSS QRE was significantly outperformed by Lk but not Poisson-CH. One possible explanation is that the filtering features are all biased toward iterative models. However, it seems unlikely that, e.g., *both* dominance-solvability and dominance-nonsolvability are biased toward iterative models. Another possibility is that iterative models are a better model of human behavior, but the cost-proportional error model of QRE is sufficiently superior to the respectively simple and non-existent error models of Lk and Poisson-CH that it outperforms on many datasets that mix game types. Or, similarly, iterative models may fit very differently on dominance solvable and non-dominance solvable games; in this case, they would perform very poorly on mixed data. We explore this last possibility in more detail in Section 3.3.1.

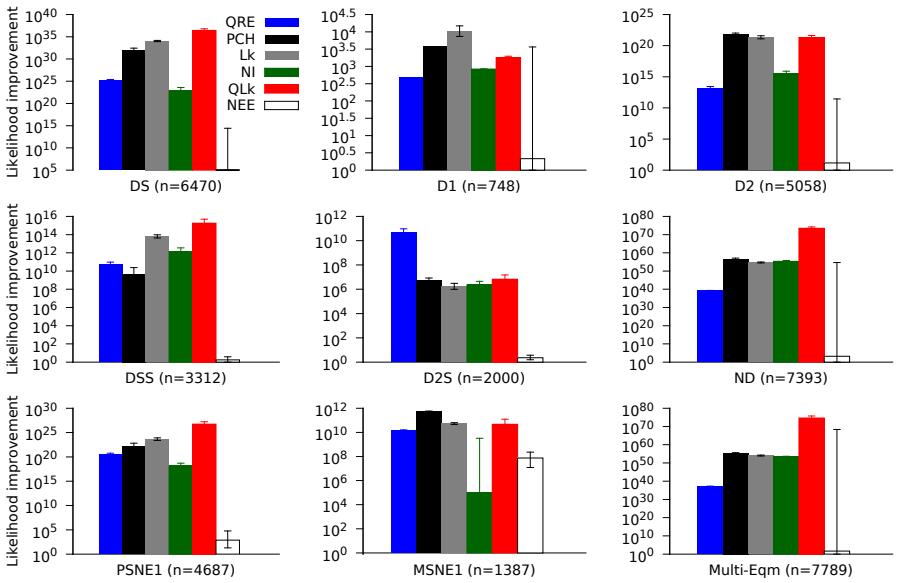


**Figure 2.3:** Average likelihood ratios of model predictions to random predictions, with 95% confidence intervals, on feature-based datasets. For NEE the main bar shows performance averaged over all equilibria, and error bars show post-hoc upper and lower bounds on equilibrium performance.

## 2.6 Related Work

This chapter has been motivated by the question, "What model is best for predicting human behavior in general, simultaneous-move games?" Before beginning our study, we conducted an exhaustive literature survey to determine the extent to which this question had already been answered. Specifically, we used Google Scholar to identify all (1805) citations to the papers introducing the QRE, CH, Lk, NI, and QLk models [McKelvey and Palfrey, 1995; Camerer et al., 2004; Costa-Gomes et al., 2001; Nagel, 1995; Goeree and Holt, 2004; Stahl and Wilson, 1994], and manually checked every reference. We discarded superficial citations, papers that simply applied one of the models to an application domain, and papers that studied repeated games. This left us with a total of 24 papers (including the six with which we began), which we summarize in Table 2.4. Overall, we found no

paper that compared the predictive performance of all six models. Indeed, there are two senses in which the literature focuses on different issues. First, it appears to be more concerned with *explaining* behavior than with *predicting* it. Thus, comparisons of out-of-sample prediction performance were rare. Here we describe the only exceptions that we found:

- Stahl and Wilson [1995] evaluated prediction performance on 3 games using parameters fit from the other games;
- Morgan and Sefton [2002] and Hahn et al. [2010] evaluated prediction performance using held-out test data;
- Camerer et al. [2004] and Chong et al. [2005] computed likelihoods on each individual game in their datasets after using models fit to the $n-1$ remaining games;
- Crawford and Iriberri [2007a] compared the performance of two models by training each model on each game in their dataset individually, and then evaluating the performance of each of these $n$ trained models on each of the $n-1$ other individual games; and
- Camerer et al. [2011] evaluated the performance of QRE and cognitive hierarchy variants on one experimental treatment using parameters estimated on two separate experimental treatments.

Second, most of the papers compared a single one of the five models (often with variations) to Nash equilibrium. Indeed, only nine of the 25 studies (see the bottom portion of Table 2.4) compared more than one of the six key models, and none of these considered QLk. Only three of these studies explicitly compared the prediction performance of more than one of the six models [Chong et al., 2005; Crawford and Iriberri, 2007a; Camerer et al., 2011]; the remaining six performed comparisons in terms of training set fit [Camerer et al., 2001; Goeree and Holt, 2004; Costa-Gomes and Weizsäcker, 2008; Costa-Gomes et al., 2009; Rogers et al., 2009; Breitmoser, 2012].

Rogers et al. [2009] proposed a unifying framework that generalizes both Poisson-CH and QRE, and compared the fit of several variations within this framework. Notably, their framework allows for quantal response within a cognitive hierarchy model. Their work is thus similar to our own search over a system of QLk variants in Chapter 4, but there are several differences. First, we compared out-of-sample prediction performance, not in-sample fit. Second, Rogers et al. restricted the distributions of types to be grid, uniform, or Poisson distributions, whereas we considered unconstrained discrete distributions over levels. Third, they required different agent types to have different precisions, while we did not. Finally, we considered level-$k$ beliefs as well as cognitive hierarchy beliefs, whereas they considered only cognitive hierarchy belief models (although their framework in principle allows for both).

One line of work from the computer science literature also meets our criteria of predicting action choices and modeling human behavior [Altman et al., 2006]. This approach learns association rules between agents' actions in different games to predict how an agent will play based on its actions in earlier games. We did not consider this approach in our study, as it requires data that identifies agents across games, and cannot make predictions for games that are not in the training dataset. Nevertheless, other machine-learning-based methods can clearly be extended to apply to our setting; we explore such an extension in Chapter 6.

## 2.7 Conclusions

To our knowledge, ours is the first study to address the question of which of the QRE, level-$k$, cognitive hierarchy, noisy introspection, and quantal level-$k$ behavioral models is best suited to predicting unseen human play of normal-form games. We explored the prediction performance of these models, along with several modifications. We found that bounded iterated reasoning and cost-proportional errors are both valuable ingredients in a predictive model of human game theoretic behavior: the best-performing model that we studied (QLk) combines both of these elements.

**Table 2.4:** Existing work in model comparison. 'f' indicates comparison of training sample fit only; 't' indicates statistical tests of training sample performance; 'p' indicates evaluation of out-of-sample prediction performance.

| Paper | Nash | QLk | Lk | CH | NI | QRE |
|---|---|---|---|---|---|---|
| Stahl and Wilson [1994] | t | t | | | | |
| McKelvey and Palfrey [1995] | f | | | | | f |
| Stahl and Wilson [1995] | f | p | | | | |
| Costa-Gomes et al. [1998] | f | | f | | | |
| Haruvy et al. [1999] | | t | | | | |
| Costa-Gomes et al. [2001] | f | | f | | | |
| Haruvy et al. [2001] | | t | | | | |
| Morgan and Sefton [2002] | f | | | | | p |
| Weizsäcker [2003] | t | | | | | t |
| Camerer et al. [2004] | f | | | p | | |
| Costa-Gomes and Crawford [2006] | f | | f | | | |
| Stahl and Haruvy [2008] | | t | | | | |
| Rey-Biel [2009] | t | | t | | | |
| Georganas et al. [2010] | f | | f | | | |
| Hahn et al. [2010] | | | | p | | |
| Camerer et al. [2001] | | | | f | | f |
| Goeree and Holt [2004] | f | | | | f | f |
| Chong et al. [2005] | f | | | p | | p |
| Crawford and Iriberri [2007a] | p | | p | | | p |
| Costa-Gomes and Weizsäcker [2008] | f | | f | | f | f |
| Costa-Gomes et al. [2009] | f | | f | f | f | f |
| Rogers et al. [2009] | f | | | f | | f |
| Camerer et al. [2011] | | | | p | | p |
| Breitmoser [2012] | | | t | t | t | t |

# Chapter 3

# Parameter Analysis of Behavioral Game Theoretic Models

## 3.1 Introduction

Making good predictions from behavioral models depends upon obtaining good estimates of model parameters. These estimates can also be useful in themselves, helping researchers to understand both how people behave in strategic situations and whether a model's behavior aligns or clashes with its intended economic interpretation. Unfortunately, the method we have used so far—maximum likelihood estimation, i.e., finding a single set of parameters that best explains the training set—is not a good way of gaining this kind of understanding. The problem is that we have no way of knowing how much of a difference it would have made to have set the parameters differently, and hence how important each parameter setting is to the model's performance. If some parameter is completely uncorrelated with predictive accuracy, the maximum likelihood estimate will set it to an arbitrary value, from which we would be wrong to draw economic conclusions.[1]

[1] We can gain local information about a parameter's importance from the confidence interval around its maximum likelihood estimate: locally important parameters will have narrow confidence intervals, and locally irrelevant parameters will have wide confidence intervals. However,

For example, in the previous chapter we noted that our parameter estimates for QLk implied a much larger proportion of level-$0$ agents than is conventionally expected. We also interpreted the large estimated value of the noise parameter $\epsilon$ as indicating that Nash equilibrium fits the data poorly. However, if there are multiple, very different ways of configuring these models to make good predictions, we should not draw firm conclusions about how people reason based on a single configuration.

An alternative is to use Bayesian analysis to estimate the entire posterior distribution over parameter values, rather than estimating only a single point. This allows us to identify the most likely parameter values; how wide a range of values are argued for by the data (equivalently, how strongly the data argues for the most likely values); and whether the values that the data argues for are plausible in terms of our intuitions about parameters' meanings. In Section 3.2 we derive an expression for the posterior distribution, and describe methods for constructing posterior estimates and using them to assess parameter importance. In Section 3.3 we will apply these methods to study QLk, NEE, and Poisson-CH: the first because it achieved such reliably strong performance; the second to cross check our interpretation of its parameter fit in Chapter 2; and the last because it is the model about which the most explicit parameter recommendation was made in the literature. Camerer et al. [2004] recommended setting Poisson-CH's single parameter, which represents agents' mean number of steps of strategic reasoning, to $1.5$. Our own analysis sharply contradicts this recommendation, placing the 99% credible interval roughly a factor of two lower, on the range $[0.70, 0.76]$. We devote most of our attention to QLk, however, due to its extremely strong performance.

this does not tell us anything outside the neighborhood of the estimate.

## 3.2 Methods

### 3.2.1 Posterior Distribution Derivation

We derive an expression for the posterior distribution $\Pr(\theta \mid \mathcal{D})$ by applying Bayes' rule, where $p_0(\theta)$ is the prior distribution:

$$\Pr(\theta \mid \mathcal{D}) = \frac{p_0(\theta) \Pr(\mathcal{D} \mid \theta)}{\Pr(\mathcal{D})}. \tag{3.1}$$

Substituting in Equation (2.4), which gave an expression for the likelihood of the dataset $\Pr(\mathcal{D} \mid \theta)$, we obtain

$$\Pr(\theta \mid \mathcal{D}) = \frac{p_0(\theta) \prod_{i=1}^{I} \prod_{j=1}^{J_i} \Pr(a_{ij} \mid G_i, \theta) \Pr(G_i)}{\Pr(\mathcal{D})}. \tag{3.2}$$

In practice $\Pr(G_i)$ and $\Pr(\mathcal{D})$ are constants, and so can be ignored:

$$\Pr(\theta \mid \mathcal{D}) \propto p_0(\theta) \prod_{i=1}^{I} \prod_{j=1}^{J_i} \Pr(a_{ij} \mid G_i, \theta). \tag{3.3}$$

Note that by commutativity of multiplication, this is equivalent to performing iterative Bayesian updates one datapoint at a time. Therefore, iteratively updating this posterior neither over- nor underprivileges later datapoints.

### 3.2.2 Posterior Distribution Estimation

We estimate the posterior distribution as a set of samples. When a model has a low-dimensional parameter space, like Poisson-CH, we generate a large number of evenly-spaced, discrete points (so-called *grid sampling*). This has the advantage that we are guaranteed to cover the whole space, and hence will not miss large, important regions. However, this approach does not work when a model's parameter space is large, because evenly-spaced grids require a number of samples exponential in the number of parameters. Luckily, we do not care about hav-

ing good estimates of the whole posterior distribution—what matters is getting good estimates of regions of high probability mass. This can be achieved by sampling parameter settings in proportion to their likelihood, rather than uniformly. A wide variety of techniques exist for performing this sort of sampling. For models such as QLk with a multidimensional parameter space, we used *Metropolis-Hastings sampling* to estimate the posterior distribution. The Metropolis-Hastings algorithm is a Markov Chain Monte Carlo (MCMC) algorithm [e.g., Robert and Casella, 2004] that computes a series of values from the support of a distribution. Although each value depends upon the previous value, the values are distributed as if from an independent sample of the distribution after a sufficiently large number of iterations. MCMC algorithms (and related techniques, e.g., annealed importance sampling [Neal, 2001]) are useful for estimating multidimensional distributions for which a closed form of the density is unknown. They require only that a value *proportional* to the true density be computable (i.e., an unnormalized density). This is precisely the case with the models that we seek to estimate.

We use a flat prior for all parameters.[2] Although this prior is improper on unbounded parameters such as precision, it results in a correctly normalized posterior distribution[3]; the posterior distribution in this case reduces to the likelihood [e.g., Gill, 2002]. For Poisson-CH, where we grid sample an unbounded parameter, we grid sampled within a bounded range ($[0, 10]$), which is equivalent to assigning probability $0$ to points outside the bounds. In practice, this turned out not to matter, as the vast majority of probability mass was concentrated near $0$.

[2]For precision parameters, another natural choice might have been to use a flat prior on the log of precision. In this work, we wanted to avoid artificially preferring precision estimates closer to zero, since it is common for iterative models to assume agents best respond nearly perfectly to lower levels (that is, have infinitely high precisions). Our posterior precision estimates tended to be concentrated near zero regardless.

[3]That is, for the posterior, $\int \cdots \int_{-\infty}^{\infty} \Pr(\theta \mid \mathcal{D})\, d\theta = 1$, even though for the prior $\int \cdots \int_{-\infty}^{\infty} p_0(\theta)\, d\theta$ diverges.

### 3.2.3 Visualizing Multi-Dimensional Distributions

In the sections that follow, we present posterior distributions as cumulative marginal distributions. That is, for every parameter, we plot the cumulative density function (CDF)—the probability that the parameter should be set less than or equal to a given value—averaging over values of all other parameters. Plotting cumulative density functions allows us to visualize an entire continuous distribution without having to estimate density from discrete samples, thus sparing us manual decisions such as the width of bins for a histogram. Plotting marginal distributions allows us to examine intuitive two-dimensional plots about multi-dimensional distributions. Interaction effects between parameters are thus obscured; luckily, in separate tests we have found that for our data these were not a major factor.[4]

## 3.3 Analysis

In this section we analyze the posterior distributions of the parameters for three of the models compared in Section 2.5: Poisson-CH, QRE, and QLk. For Poisson-CH, we computed the likelihood for each value of $\tau \in \{0.01k \mid k \in \mathbb{N}, 0 \leq 0.01k \leq 10\}$, and then normalized by the sum of the likelihoods. For QRE, we computed the likelihood for each value of $\lambda \in \{0.001k \mid k \in \mathbb{N}, 0 \leq 0.001k \leq 1\}$. For QLk, we combined the samples from $4$ independent Metropolis-Hasting chains, each of which computed $220,000$ samples, discarding the first $20,000$ samples as a "burn-in" period to allow the Markov chain to converge. We used the PyMC software package to generate the samples [Patil et al., 2010]. Computing the posterior distribution for a single model in this way typically required approximately 200 CPU-hours.

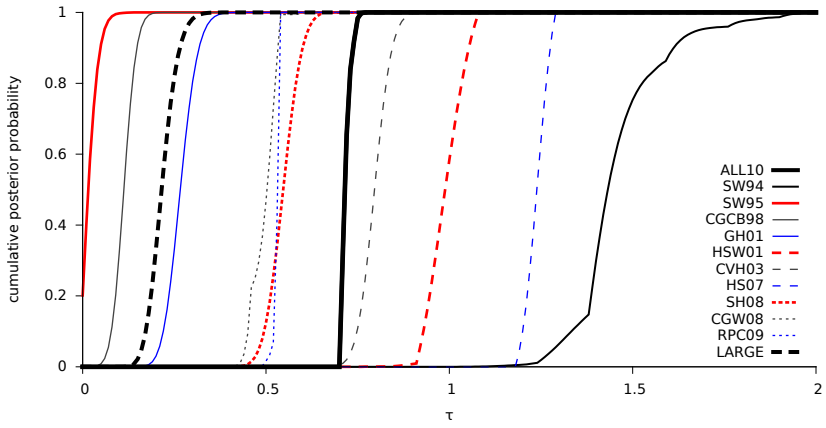


**Figure 3.1:** Cumulative posterior distributions for Poisson-CH's $\tau$ parameter. Bold solid trace is the combined dataset; solid black trace is the outlier Stahl and Wilson [1994] source dataset; bold dashed trace is a subset containing all large games (those with more than 5 actions per player).

### 3.3.1 Poisson-CH

In an influential recommendation from the literature, Camerer et al. [2004] suggest[5] setting the $\tau$ parameter of the Poisson-CH model to $1.5$. Our Bayesian analysis techniques allow us to estimate CDFs for this parameter on each of our datasets (see Figure 3.1). Overall, our analysis strongly contradicts Camerer et al.'s recommendation. On ALL10, the posterior probability of $0.70 \leq \tau \leq 0.76$ is more than $99\%$. On the ALL10 dataset, the likelihood ratio between the posterior median value $\tau = 0.71$ and $\tau = 1.5$ is $10^{402}$; that is, $0.71$ is $10^{402}$ times more likely to be the true value of $\tau$ than $1.5$.

Every other source dataset had a wider $99\%$ *credible interval* (the Bayesian counterpart to confidence intervals) for $\tau$ than ALL10, as indicated by the higher slope of ALL10's cumulative density function (since smaller datasets lead to less

[4] In particular, we found little in the way of interaction effects between parameters.

[5] Although Camerer et al. phrase their recommendation as a reasonable "omnibus guess," it is often cited as an authoritative finding [e.g., Carvalho and Santos-Pinto, 2010; Frey and Goldstone, 2011; Choi, 2012; Goodie et al., 2012].

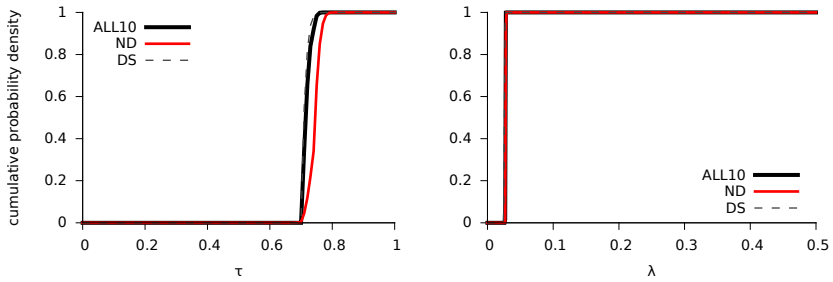


**Figure 3.2:** Distributions for the Poisson-CH and QRE models on the combined dataset (ALL10), non-dominance-solvable games only (ND), and dominance-solvable games only (DS). Left: cumulative distribution of $\tau$ parameter of Poisson-CH; Right: cumulative distribution of $\lambda$ parameter of QRE.

confident predictions). Nevertheless, all but two of the source datasets had median values less than 1.0. Only the Stahl and Wilson [1994] dataset (SW94) supports Camerer et al.'s recommendation (median 1.43). However (as we have observed before), SW94 appears to be an outlier; its credible interval is wider than that of the other distributions, and the distribution is very multimodal, possibly due to the dataset's small size.

Many of the games in our dataset have small action spaces. For example, 108 out of the 142 games in ALL10 have exactly 3 actions per player. One might worry that the estimated average cognitive level in Figure 3.1 is artificially low, since it is impossible to distinguish higher numbers of levels than the number of actions available to each player. We check this by performing the same posterior estimation on a subset of the data consisting only of the 20 large games (i.e., those with more than 5 actions available to each player). As Figure 3.1 shows, the estimated average cognitive level in these large games is even lower than the overall estimate, with a median of 0.22.

As we speculated in Section 2.5.3, Poisson-CH does indeed appear to treat dominance-solvable games differently from non-dominance-solvable games. The left panel of Figure 3.2 compares the cumulative distribution for $\tau$ on the combined dataset to CDFs for non-dominance-solvable games only and for dominance-

solvable games only. The $\tau$ parameter has a nearly identical credible interval for dominance-solvable games as for the full combined dataset. For non-dominance-solvable games the $\tau$ parameter's $99\%$ credible interval is somewhat larger, at $[0.70, 0.82]$. In contrast, the fits for QRE's precision ($\lambda$) parameter are very small for all three game types. This explains how iterative models could outperform QRE on almost every feature-based dataset in Section 2.5.3, in spite of being frequently outperformed by QRE in Section 2.5.1: the features used to separate games in Section 2.5.3 tend to separate dominance-solvable and non-dominance-solvable games, and the iterative models can adapt their predictions accordingly, whereas QRE's predictions are less influenced by a game's dominance solvability or lack thereof.

### 3.3.2 Nash Equilibrium

In Section 2.5.2, we observed that the Nash equilibrium with error model performs best when its noise parameter ($\epsilon$) is set to an extraordinarily large value ($0.87$). We interpreted this as evidence against Nash equilibrium as a good predictive model of human behavior. In this section we estimate the full posterior distribution for $\epsilon$; see Figure 3.3. By doing so we are able to confirm that in both ALL10 and its component source datasets, the posterior distribution for $\epsilon$ is very concentrated around very large values of $\epsilon$.

### 3.3.3 QLk

Figure 3.4 gives the marginal cumulative posterior distributions for each of the parameters of the QLk model. (That is, we computed the five-dimensional posterior distribution, and then extracted from it the five marginal distributions shown here.) Overall, the parameters appear to be well-identified, in the sense of having unimodal distributions with relatively narrow credible intervals.

As in the MLE fits in Chapter 2—of both QLk and the other iterative models—the posterior frequency of level-$0$ agents is surprisingly high. The posterior medians for the proportion of level-$0$, level-$1$, and level-$2$ agents are $0.32$, $0.42$, and

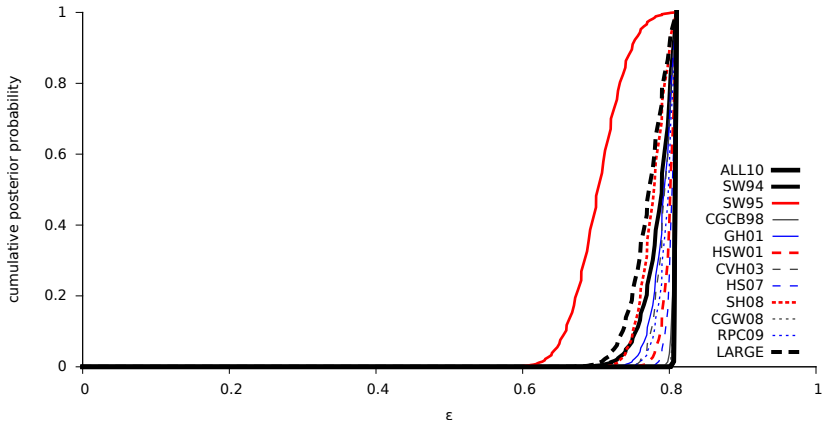


**Figure 3.3:** Cumulative posterior distributions for NEE's $\epsilon$ parameter. Bold solid trace is the combined dataset; bold dashed trace is a subset containing all large games (those with more than 5 actions per player).

$0.26$, respectively. The MLE estimate of level-$0$ proportions ($0.33$) in Chapter 2 was extremely close to the median posterior value. The MLE estimates for level-$1$ and level-$2$ proportions were somewhat lower and higher than the posterior medians, respectively ($0.33$ and $0.33$).

Agents are all attributed rather small quantal response precisions. The posterior median precisions for level-$1$ agents, level-$2$ agents, and the belief of level-$2$ agents about level-$1$ agents are $0.16$, $0.56$, and $0.05$ respectively. The belief of the level-$2$ agents that the level-$1$ agents have a much smaller precision than their actual precision is particularly strongly identified. That is, the ALL$10$ dataset assigns the highest posterior probability to parameter settings in which the level-$2$ agents ascribe a smaller than accurate quantal response precision to the level-$1$ agents. This may indicate that two-level strategic reasoning causes a high cognitive load, which makes agents more likely to make mistakes in their predictions of others' behavior. The main appeal of this explanation is that it allows us to accept the QLk model's strong performance at face value. Alternately, we might worry that QLk fails to capture some crucial aspect of experimental subjects' strategic reasoning. For example, the low value of $\lambda_{1(2)}$ might reflect level-$2$ agents' reason-

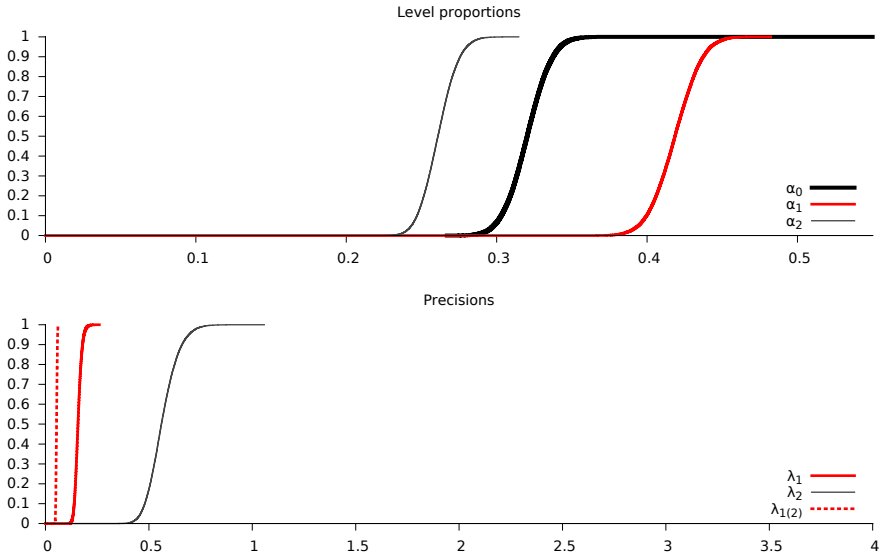


**Figure 3.4:** Marginal cumulative posterior distribution functions for the level proportion parameters ($\alpha_0, \alpha_1, \alpha_2$; top panel) and precision parameters ($\lambda_1, \lambda_2, \lambda_{1(2)}$; bottom panel) of the QLk model on ALL10. $\alpha_0$ is defined implicitly by $\alpha_1$ and $\alpha_2$.

ing about all lower levels rather than just one level below themselves: ascribing a low precision to level-1 agents approximates a mixture of level-1 agents and uniformly randomizing level-0 agents. That is, the low value of $\lambda_{1(2)}$ may be a way of simulating a cognitive hierarchy style of reasoning within a level-$k$ framework. In Chapter 4, we will explore this possibility as part of an evaluation of systematic variations of QLk's modeling assumptions.

## 3.4 Conclusions

Bayesian parameter analysis is a valuable technique for investigating the behavior and properties of models, particularly because it is able to make quantitative recommendations for parameter values. We showed how Bayesian parameter analysis can be applied to derive concrete recommendations for the use of an existing model, Poisson-CH, differing substantially from widely cited advice in the litera-

ture.

Our parameter estimates for all of the iterative models included a substantial proportion of level-$0$ agents. The level-$0$ model is important for predicting the behavior of all agents in an iterative model; both the level-$0$ agents themselves, and the higher-level agents whose behavior is grounded in a model of level-$0$ behavior. Motivated by this, we introduce a richer specification of level-$0$ behavior in Chapter 5, which allows for significant performance improvements.

# Chapter 4

# Model Variations

QLk incorporates a number of modeling assumptions. In this chapter, we investigate the properties of the QLk model by evaluating the predictive power of a family of models that systematically vary a set of these assumptions. In the end, we identify a simpler model that dominated QLk on our data.

## 4.1 Construction

We constructed a broad family of models by modifying the QLk model along four different axes. First, QLk assumes a maximum level of 2; we considered maximum levels of 1 and 3 as well. Second, QLk assumes *inhomogeneous precisions* in that it allows each level to have a different precision; we varied this by also considering *homogeneous precision* models. Third, QLk allows *general precision beliefs* that can differ from lower-level agents' true precisions; we also constructed models that make the simplifying assumption that all agents have *accurate precision beliefs* about lower-level agents.[1] Finally, in addition to *Lk* beliefs (where all other agents are assumed by a level-$k$ agent to be level-$(k-1)$), we also constructed models with *CH* beliefs (where agents believe that the population consists of the true, truncated distribution over the lower levels). We evaluated each

[1] This is in the same spirit as the simplifying assumption made in cognitive hierarchy models that agents have accurate beliefs about the proportions of lower-level agents.

**Table 4.1:** Model variations with prediction performance on the ALL10 dataset. The models with maximum level of $*$ used a Poisson distribution. Models are named according to precision beliefs, precision homogeneity, population beliefs, and type of level distribution. E.g., `ah-QCH3` is the model with accurate precision beliefs, homogeneous precisions, cognitive hierarchy population beliefs, and a discrete distribution over levels 0–3.

| Name | Max Level | Population Beliefs | Precision Beliefs | Precisions | Parameters | Log likelihood vs. u.a.r. |
|---|---|---|---|---|---|---|
| QLk1 | 1 | n/a | n/a | n/a | 2 | $87.37 \pm 1.04$ |
| gi-QLk2 | 2 | Lk | general | inhomo. | 5 | $108.66 \pm 0.56$ |
| ai-QLk2 | 2 | Lk | accurate | inhomo. | 4 | $103.33 \pm 1.75$ |
| gh-QLk2 | 2 | Lk | general | homo. | 4 | $107.96 \pm 0.46$ |
| ah-QLk2 | 2 | Lk | accurate | homo. | 3 | $104.84 \pm 0.58$ |
| gi-QCH2 | 2 | CH | general | inhomo. | 5 | $107.78 \pm 0.88$ |
| ai-QCH2 | 2 | CH | accurate | inhomo. | 4 | $106.76 \pm 0.92$ |
| gh-QCH2 | 2 | CH | general | homo. | 4 | $109.43 \pm 0.58$ |
| ah-QCH2 | 2 | CH | accurate | homo. | 3 | $106.67 \pm 0.41$ |
| gi-QLk3 | 3 | Lk | general | inhomo. | 9 | $113.17 \pm 1.46$ |
| ai-QLk3 | 3 | Lk | accurate | inhomo. | 6 | $109.62 \pm 1.21$ |
| gh-QLk3 | 3 | Lk | general | homo. | 7 | $113.48 \pm 1.46$ |
| ah-QLk3 | 3 | Lk | accurate | homo. | 4 | $107.12 \pm 0.46$ |
| gi-QCH3 | 3 | CH | general | inhomo. | 10 | $113.01 \pm 0.93$ |
| ai-QCH3 | 3 | CH | accurate | inhomo. | 6 | $111.34 \pm 0.59$ |
| gh-QCH3 | 3 | CH | general | homo. | 8 | $113.08 \pm 0.83$ |
| ah-QCH3 | 3 | CH | accurate | homo. | 4 | $110.42 \pm 0.46$ |
| ai-QLk4 | 4 | Lk | accurate | inhomo. | 8 | $110.30 \pm 0.93$ |
| ah-QLk4 | 4 | Lk | accurate | homo. | 5 | $106.63 \pm 0.71$ |
| ah-QLk5 | 5 | Lk | accurate | homo. | 6 | $107.18 \pm 0.57$ |
| ah-QLk6 | 6 | Lk | accurate | homo. | 7 | $106.57 \pm 0.68$ |
| ah-QLk7 | 7 | Lk | accurate | homo. | 8 | $106.50 \pm 0.69$ |
| ah-QLkp | * | Lk | accurate | homo. | 2 | $106.89 \pm 0.28$ |
| ai-QCH4 | 4 | CH | accurate | inhomo. | 8 | $111.54 \pm 0.62$ |
| ah-QCH4 | 4 | CH | accurate | homo. | 5 | $110.88 \pm 0.33$ |
| ah-QCH5 | 5 | CH | accurate | homo. | 6 | $111.22 \pm 0.39$ |
| ah-QCH6 | 6 | CH | accurate | homo. | 7 | $111.26 \pm 0.44$ |
| ah-QCH7 | 7 | CH | accurate | homo. | 8 | $111.42 \pm 0.41$ |
| ah-QCHp | * | CH | accurate | homo. | 2 | $110.48 \pm 0.25$ |

combination of axis values; the 17 resulting models[2] are listed in the top part of Table 4.1. In addition to the 17 exhaustive axis combinations for models with maximum levels in $\{1, 2, 3\}$, we also evaluated (1) 12 additional axis combinations that have higher maximum levels and 8 parameters or fewer: `ai-QCH4` and

[2]When the maximum level is 1, all combinations of the other axes yield identical predictions. Therefore there are only 17 models instead of $3(2^3) = 24$.

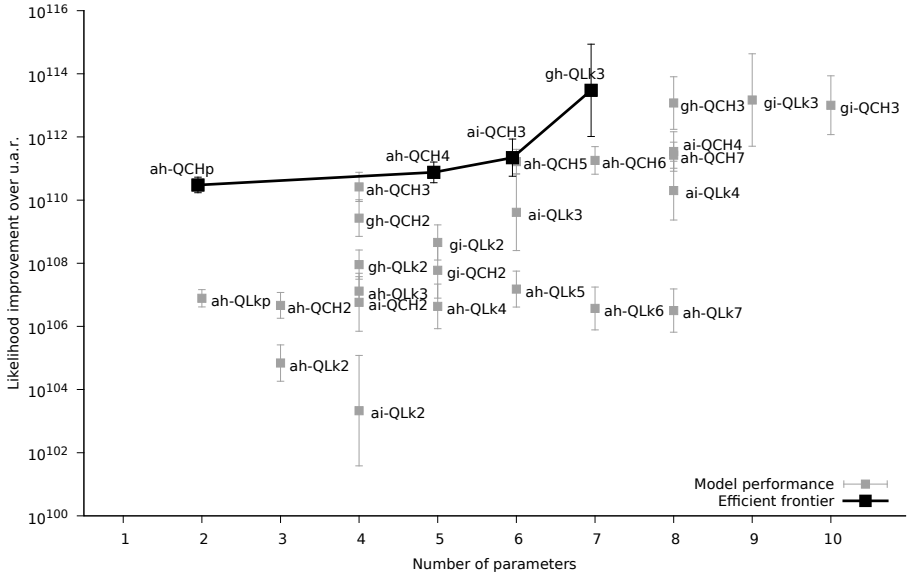


**Figure 4.1:** Model simplicity vs. prediction performance on the ALL10 dataset. `QLk1` is omitted because its far worse performance ($\sim 10^{87}$) distorts the figure's scale.

`ai-QLk4`; `ah-QCH` and `ah-QLk` variations with maximum levels in $\{4, 5, 6, 7\}$; and (2) `ah-QCH` and `ah-QLk` variations that assume a Poisson distribution over the levels rather than using an explicit tabular distribution.[3] These additional models are listed in the bottom part of Table 4.1.

## 4.2 Simplicity Versus Predictive Performance

We evaluated the predictive performance of each model on the ALL10 dataset using 10-fold cross-validation repeated 10 times, as in Section 2.5. The results are given in the last column of Table 4.1 and plotted in Figure 4.1.

All else being equal, a model with higher performance is more desirable, as is a model with fewer parameters. We can plot an *efficient frontier* of those models that achieved the best performance for a given number of parameters or fewer; see Figure 4.1. The original QLk model (`gi-QLk2`) is *not* efficient in this sense;

[3]The `ah-QCHp` model is identical to the CH-QRE model of Camerer et al. [2011].

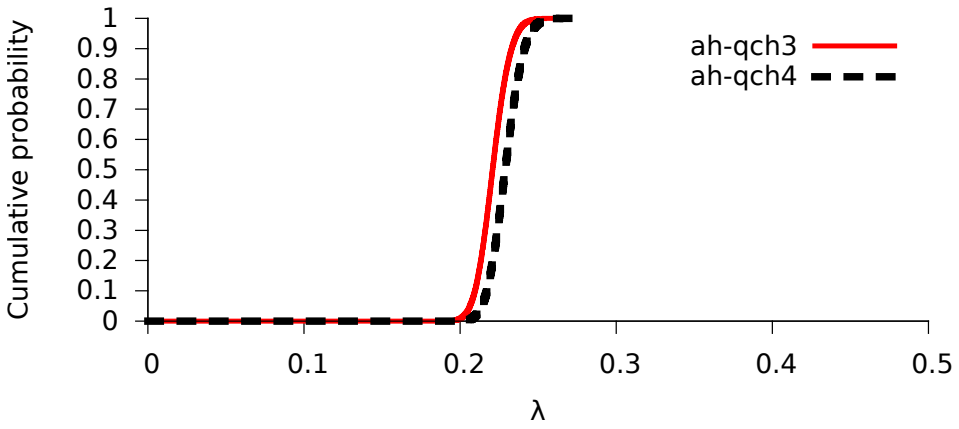


**Figure 4.2:** Marginal cumulative posterior distributions for the precision parameter ($\lambda$) of the `ah-QCH3` and `ah-QCH4` models on ALL10.

it is dominated by, e.g., `ah-QCH3`, which has both significantly better predictive performance and fewer parameters (because it restricts agents to homogeneous precisions and accurate beliefs).

There is a striking pattern among the efficient models with 6 parameters or fewer: every such model has accurate precision beliefs, cognitive hierarchy population beliefs, and, with the exception of `ai-QCH3`, homogeneous precisions. Furthermore, `ai-QCH3`'s performance was not significantly better than that of `ah-QCH5`, which did have homogeneous precisions. This suggests that the most parsimonious way to model human behavior in normal-form games is to use a model of this form.

Adding flexibility by modeling general beliefs about precisions did improve performance; the four best-performing models all incorporated general precision beliefs. However, these models also had much larger variance in their prediction performance on the test set. This may indicate that the models are overly flexible, and hence prone to overfitting.

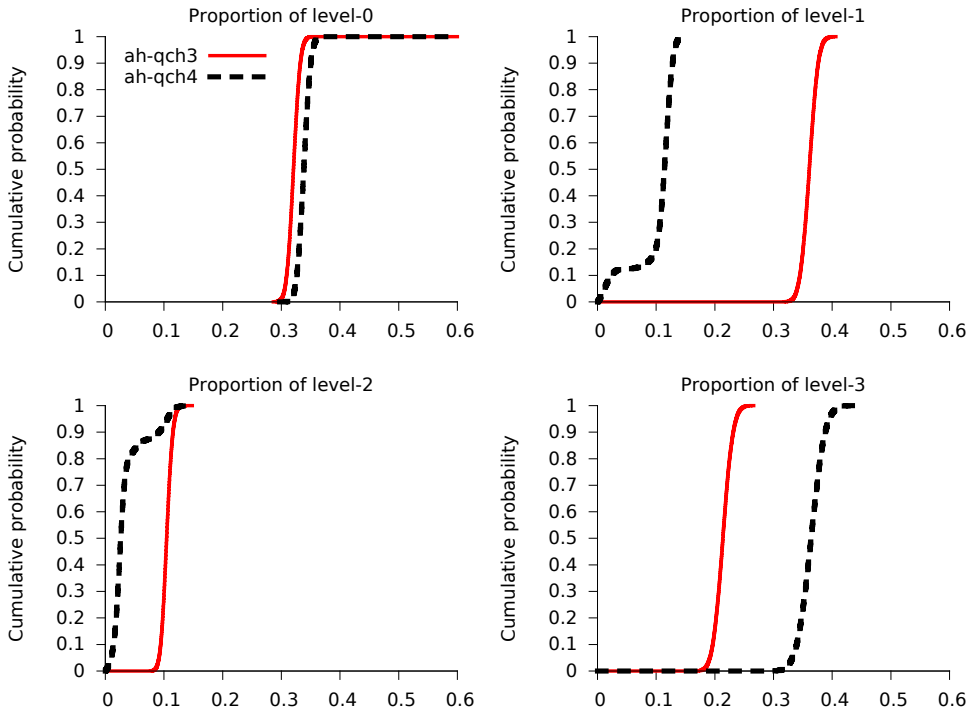


**Figure 4.3:** Marginal cumulative posterior distributions for the level proportion parameters ($\alpha_0, \alpha_1, \alpha_2, \alpha_3$) of the `ah-QCH3` and `ah-QCH4` models on ALL10. Solid lines are `ah-QCH3`; dashed lines are `ah-QCH4`. $\alpha_0$ is defined implicitly by $\alpha_1, \alpha_2, \alpha_3$, and (for `ah-QCH3`) $\alpha_4$.

## 4.3 Parameter Analysis

In this section we examine the marginal posterior distributions of two models from the accurate, homogeneous QCH family (see Figure 4.2 and Figure 4.3). We computed the posterior distribution of the models' parameters using the procedure described in Sections 3.2.2 and 3.3. The posterior distribution for the precision parameter $\lambda$ is concentrated around $0.20$, somewhat greater than the QLk model's estimate for $\lambda_1$. This suggests that QLk's much lower estimate for $\lambda_{1(2)}$ may indeed have been the closest that the model could get to having the level-$2$ agents best respond to a mixture of level-$0$ and level-$1$ agents (as in cognitive hierarchy).

Our robust finding in Chapters 2 and 3 of a large proportion of level-$0$ agents

is confirmed by these models as well. Indeed, the number of level-$0$ agents is nearly the only point of close agreement between the two models with respect to the distribution of levels.

## 4.4 Spike-Poisson

We saw evidence in Section 4.2 that the `ah-QCH` family of models had good performance, with relatively low variance. Another group of general precision models had higher performance, but higher variance suggestive of overfitting. In Section 4.3, we saw that the `ah-QCH` family broadly agreed on a proportion of level-$0$ agents of about $0.30$, but ascribed varying proportions to the other levels. It seems that the data argue very consistently for a particular proportion of level-$0$ agents, and less consistently for the proportions of other levels. At the same time, we saw in Section 4.2 that performance improved within the `ah-QCH` family as the number of modeled levels increased.

In this section, we describe a low-parameter model that specifies the proportion of agents using a mixture of a deterministic fraction of level-$0$ agents and a standard Poisson distribution. This allows for the precise targeting of the level-$0$ proportion, while also allowing a single parameter to specify the proportions of higher levels. We refer to this mixture as a *Spike-Poisson* distribution. Because we will end up recommending its use, we define the full Spike-Poisson QCH model here.

**Definition 7** (Spike-Poisson Quantal Cognitive Hierarchy (QCH) model)**.** Let $\pi^{SP}_{i,m} \in \Pi(A_i)$ be the distribution over actions predicted for an agent $i$ with level $m$ by the Spike-Poisson QCH model. Let

$$f(m) = \begin{cases} \epsilon + (1-\epsilon)\text{Poisson}(m;\tau) & \text{if } m = 0, \\ (1-\epsilon)\text{Poisson}(m;\tau) & \text{otherwise.} \end{cases}$$

Let

$$\pi^{SP}_{i,0:m} = \sum_{\ell=0}^{m} f(\ell) \frac{\pi^{SP}_{i,\ell}}{\sum_{\ell'=0}^{m} f(\ell')}$$

be the truncated distribution over actions predicted for an agent conditional on that agent's having level $0 \leq \ell \leq m$. Then $\pi^{SP}$ is defined as

$$\pi^{SP}_{i,0}(a_i) = |A_i|^{-1},$$
$$\pi^{SP}_{i,m}(a_i) = QBR^{G}_{i}(\pi^{SP}_{i,0:m-1}).$$

The overall predicted distribution of actions is a weighted sum of the distributions for each level:

$$\Pr(a_i \mid \tau, \epsilon, \lambda) = \sum_{\ell=0}^{\infty} f(\ell) \pi^{SP}_{i,\ell}(a_i).$$

The model thus has three parameters: the mean of the Poisson distribution $\tau$, the spike probability $\epsilon$, and the precision $\lambda$. □

Figure 4.4 compares the performance of Spike-Poisson QCH to the efficient models of Section 4.2; for reference, QLk is also included. The three-parameter Spike-Poisson QCH model did not have significantly worse performance than any efficient model, with the exception of `gh-QLk3`. However, its variance was considerably smaller than that of `gh-QLk3`, indicating that it is likely less prone to overfitting.

## 4.5 Conclusions

QLk (`gi-qlk2`) provides substantial flexibility in specifying the beliefs and precisions of different types of agents. In this chapter, we found that this flexibility tends to hurt generalization performance more than it helps. In a systematic search of model variations, we identified a new model family (the accurate precision belief, homogeneous-precision QCH models) that contained the efficient (or nearly-efficient) model for every number of parameters smaller than 7. Based on further analysis of this model family, we constructed a particular model specifica-

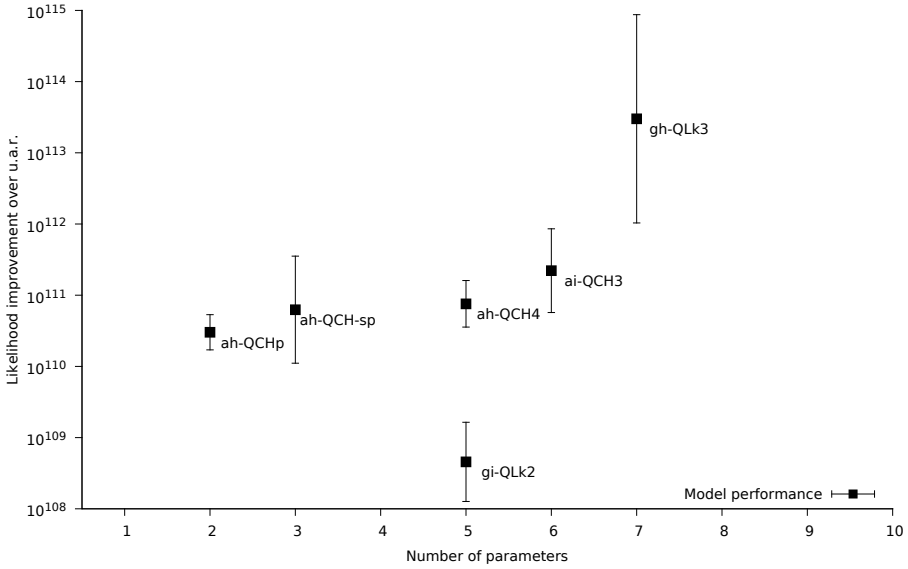


**Figure 4.4:** Model simplicity (number of parameters) versus prediction performance on the ALL10 dataset, comparing the efficient models from Section 4.2, QLk, and and Spike-Poisson QCH.

tion (Spike-Poisson QCH) which offers excellent generalization performance in spite of having only three parameters.

In Chapter 5, we will build further on this model, constructing a model that predicts better than even `gh-QLk3`, while still having fewer parameters and lower variance. We thus recommend the use of Spike-Poisson by researchers wanting to predict human play in (unrepeated) normal-form games, in conjunction with the level-0 extensions of Chapter 5.

# Chapter 5

# Models of Level-0 Behavior

## 5.1 Introduction

In quantal cognitive hierarchy models, such as the Spike-Poisson QCH model from Chapter 4, agents do not reason arbitrarily deeply about their opponents' beliefs about beliefs about beliefs. Instead, they start from a simple nonstrategic strategy[1] (the *level*-0 behavior), and then reason for some fixed number of iterations about responses to that strategy (e.g., a *level*-2 agent quantally best responds to the combined behaviors of level-1 and level-0 agents).

Thus, in order to make use of a quantal cognitive hierarchy model one must first commit to a specification of level-0 behavior. Indeed, this is true of iterative models in general, such as cognitive hierarchy [Camerer et al., 2004] and level-$k$ [Stahl and Wilson, 1994; Nagel, 1995; Costa-Gomes et al., 2001]. It is important to get this specification right, for two reasons. First, there is growing evidence that a substantial fraction of human players do act nonstrategically [Burchardi and Penczynski, 2012; Chapter 3 of this dissertation]. Second, a level-0 model also drives predictions that are made about strategic agents: higher-level agents

[1] In this work, we refer to agents that form explicit beliefs about the behavior of other agents as "strategic," and agents that do not reason about other agents in this way as "nonstrategic". Nonstrategic should not be taken as a synonym for unsophisticated or thoughtless. Some of the level-0 behavior that we describe below is rather sophisticated.

are assumed to act by responding strategically to lower-level agents' behavior.

Almost all work in the literature that uses iterative models adopts the specification that level-$0$ agents play a uniform distribution over actions. (In Section 5.4 we discuss the few exceptions of which we are aware, each of which is based on an explicitly encoded intuition about a specific setting of interest.) The uniform-distribution approach has the advantage that it does not require insight into a game's structure, and hence can be applied to any game. However, in many games it is not plausible that an agent would choose an action uniformly at random, nor that any other agent would expect them to do so. For example, consider a dominated action that always yields very low payoffs for all players.

In this chapter we consider the question of how to do better. Specifically, we investigate general rules that can be used to induce a level-$0$ specification from the normal-form description of an arbitrary game.

## 5.2 Level-0 Model

In this section we present the components from which we will construct models for computing level-$0$ distributions of play. We first describe features computed for each of an agent's actions, followed by options for combining feature values to obtain a level-$0$ prediction.

### 5.2.1 Level-0 Features

Most applications of iterative models specify that level-$0$ agents choose their actions uniformly, thus implicitly identifying nonstrategic behavior with uniform randomization. The core idea of this chapter is that nonstrategic behavior need not be uniform. How then might a nonstrategic agent behave? We argue that agents consider simple rules (*features*) that recommend one or more actions, to greater or lesser degrees. We consider both binary features with range $\{0, 1\}$ and real-valued features with range $\mathbb{R}^+$.

To be eligible for inclusion in our level-$0$ specification, we require that fea-

tures not depend on beliefs about how other agents will attempt to maximize their own utility functions. E.g., the maxmax payoff feature (see below) could be interpreted as a belief that the other agents will act in such a way that the level-0 agent can maximize its own payoff; however, this belief does not take the other agents' utility function into account at all. The minmin unfairness feature (see below) could be interpreted as a belief that the other agents will act to minimize unfairness; however, it does not model the other agents as attempting to maximize their own utility at all.

We restrict our attention to features that can be computed directly from the normal form of the game, and which do not depend on presentation details such as the units in which payoffs are expressed or the order in which actions are presented. This allows for more accurate analysis of strategic models, even when details of presentation are unknown or not yet known. We do not claim that the features that we investigated comprise an exhaustive list of factors that could influence nonstrategic agents' actions. In Chapter 6, we will consider a more automated but less interpretable procedure for learning nonstrategic features.

For each feature, we briefly describe its motivation and then formally define it. Many of our features have been investigated in both the classical and behavioral game theory literature in other contexts. In particular, the maxmax payoff, maxmin payoff, and maxmax welfare features correspond to the *Optimistic*, *Pessimistic*, and *Altruistic* nonstrategic types in Costa-Gomes et al. [2001]. Other features, such as the max-symmetric feature, were influenced by introspection about paradigmatic games such as the Traveler's Dilemma.

For each feature, we define both a binary version and a real-valued version. Unlike a binary feature, where a criterion must be maximized in order to be recommended, with a real-valued feature an action will be recommended to the *degree* that it maximizes a criterion. This addresses the intuition that two very high payoff actions may both be attractive, even if one offers marginally higher payoff than the other.

Some real-valued features represent quantities that an agent would wish to

minimize, rather than maximizing. We apply the inv transformation to these features, where inv is defined differently depending upon how features will be combined. If feature values will be combined linearly, then $\text{inv}(x) = 1/x$. If feature values will be combined with a logit function, then $\text{inv}(x) = -x$.

*Maxmin payoff.* A maxmin action for agent $i$ is the action with the best worst-case guarantee. That is,

$$f^{\text{maxmin}}(a_i) = \begin{cases} 1 & \text{if } a_i \in \arg\max_{a'_i \in A_i} \min_{a_{-i} \in A_{-i}} u_i(a'_i, a_{-i}), \\ 0 & \text{otherwise.} \end{cases}$$

This is the safest single action to play against a hostile agent.[2] The real-valued version of this feature returns the worst-case payoff for an action:

$$f^{\text{min}}(a_i) = \min_{a_{-i} \in A_{-i}} u_i(a_i, a_{-i}).$$

*Maxmax payoff.* In contrast, a maxmax action for agent $i$ is the action with the best best case. That is,

$$f^{\text{maxmax}}(a_i) = \begin{cases} 1 & \text{if } a_i \in \arg\max_{a'_i \in A_i} \max_{a_{-i} \in A_{-i}} u_i(a'_i, a_{-i}), \\ 0 & \text{otherwise.} \end{cases}$$

An agent who aims to maximize his possible payoff will play a maxmax action. The real-valued version of this feature returns the best-case payoff for an action:

$$f^{\text{max}}(a_i) = \max_{a_{-i} \in A_{-i}} u_i(a_i, a_{-i}).$$

[2]Often, a mixed strategy will be safer still against a hostile agent. However, in this application we are not actually trying to find a safest strategy for the agent. Rather, we are trying to specify features of individual actions that might make them attractive to nonstrategic agents.

*Minimax regret.* Savage [1951] proposed the *minimax regret* criterion for making decisions in the absence of probabilistic beliefs. In a game theoretic context, it works as follows. For each action profile, an agent has a possible *regret*: how much more utility could the agent have gained by playing the best response to the other agents' actions? Each of the agent's actions is therefore associated with a vector of possible regrets, one for each possible profile of the other agents' actions. A minimax regret action is an action whose maximum regret (in the vector of possible regrets) is minimal. That is, if

$$r(a_i, a_{-i}) = u_i(a_i, a_{-i}) - \max_{a_i^* \in A_i} u_i(a_i^*, a_{-i})$$

is the regret of agent $i$ in action profile $(a_i, a_{-i})$, then

$$f^{\text{mmr}}(a_i) = \begin{cases} 1 & \text{if } a_i \in \arg\min_{a_i' \in A_i} \max_{a_{-i} \in A_{-i}} r(a_i, a_{-i}), \\ 0 & \text{otherwise.} \end{cases}$$

The real-valued version of this feature returns the worst-case regret for playing an action:

$$f^{\text{mr}}(a_i) = \text{inv}\left[\max_{a_{-i} \in A_{-i}} r(a_i, a_{-i})\right].$$

Higher max regret is less desirable than lower max regret, explaining our use of the inv transformation.

*Minmin unfairness.* Concern for the fairness of outcomes is a common feature of human play in strategic situations, as has been confirmed in multiple behavioral studies, most famously in the Ultimatum game [Thaler, 1988; Camerer and Thaler, 1995]. Let the unfairness of an action profile be the difference between the maximum and minimum payoffs among the agents under that action profile:

$$d(a) = \max_{i,j \in N} u_i(a) - u_j(a).$$

Then a *fair* outcome minimizes this difference in utilities. The *minmin unfairness* feature selects every action which is part of a minimally unfair action profile.

$$f^{\text{fair}}(a_i) = \begin{cases} 1 & \text{if } a_i \in \arg\min_{a'_i \in A_i} \min_{a_{-i} \in A_{-i}} d(a'_i, a_{-i}), \\ 0 & \text{otherwise.} \end{cases}$$

The real-valued version of this feature returns the minimum unfairness that could result from playing a given action:

$$f^{\text{unfair}}(a_i) = \text{inv}\left[\min_{a_{-i} \in A_{-i}} d(a_i, a_{-i})\right].$$

Unfairness is a quantity to be minimized, so we apply the inv transformation.

*Max symmetric.* People often reason about what would happen if the other agent acted as they did.[3] A max-symmetric action is simply the best such action:

$$f^{\text{maxsymm}}(a_i) = \begin{cases} 1 & \text{if } a_i \in \arg\max_{a'_i \in A_i} u(a'_i, \ldots, a'_i), \\ 0 & \text{otherwise.} \end{cases}$$

The real-valued version of this feature returns the symmetric payoff of an action:

$$f^{\text{symm}}(a_i) = u(a_i, \ldots, a_i).$$

*Maxmax welfare.* Finally, one reason that a nonstrategic agent might find an action profile desirable is that it produces the best overall benefit to the pair of agents. The *maxmax welfare* feature selects every action that is part of some

[3] This is a concept that only applies to symmetric games, in which agents have identical action sets, and each agent's payoff matrix is the transpose of the other.

action profile that maximizes the sum of utilities:

$$f^{\text{efficient}}(a_i) = \begin{cases} 1 & \text{if } a_i \in \arg\max_{a'_i \in A_i} \max_{a_{-i} \in A_{-i}} \sum_{j \in N} u_j(a'_i, a_{-i}), \\ 0 & \text{otherwise.} \end{cases}$$

The real-valued version of this feature returns the maximum welfare that could result from playing a given action:

$$f^{\text{welfare}}(a_i) = \max_{a_{-i} \in A_{-i}} \sum_{j \in N} u_j(a_i, a_{-i}).$$

### 5.2.2 Combining Feature Values

Once a set of features have been computed for each of a set of actions, their values must be combined to yield a single distribution over actions. There is an infinite number of ways to perform such a combination. We considered two functional forms, inspired by linear regression and logit regression respectively.

Both specifications accept a set of features and a set of weights. Let $F$ be a set of features mapping from an action to $\mathbb{R}^+$. For each feature $f \in F$, let $w_f \in [0, 1]$ be a weight parameter. Let $\sum_{f \in F} w_f \leq 1$, and let $w_0 = 1 - \sum_{f \in F} w_f$.

The first functional form produces a level-0 prediction over actions for a given agent by taking a weighted sum of feature outputs for each action and then normalizing to produce a distribution.

**Definition 8** (Weighted linear level-0 specification)**.** The *weighted linear level-0 specification* predicts the following distribution of actions for level-0 agents:

$$\pi_{i,0}^{\text{linear},F}(a_i) = \frac{w_0 + \sum_{f \in F} w_f f(a_i)}{\sum_{a'_i \in A_i} \left[ w_0 + \sum_{f \in F} w_f f(a'_i) \right]}.$$

The second functional form assigns a level-0 probability proportional to the exponential of a weighted sum of feature values.

**Definition 9** (Logit level-0 specification)**.** The *logit level-0 specification* predicts

the following distribution of actions for level-$0$ agents:

$$\pi_{i,0}^{\text{logit},F}(a_i) = \frac{\exp(w_0 + \sum_{f \in F} w_f f(a_i))}{\sum_{a_i' \in A_i} \exp(w_0 + \sum_{f \in F} w_f f(a_i))}.$$

### 5.2.3 Feature Transformations

In addition to two functional forms for combining the feature values, we also evaluated two transformations to feature values. These transformations may be applied to each feature value before they are weighted and combined.

The first transformation zeroes out features that have the same value for every action, which we call *uninformative*. The intuition behind this transformation is that informative features should have a greater influence on the prediction precisely when the other features are less informative.

**Definition 10** (Informativeness feature transformation)**.** A feature $f$ is *informative* in a game $G$ if there exists $a_i', a_i'' \in A_i$ such that $f(a_i') \neq f(a_i'')$. The informativeness transformation $I(f)$ of a feature $f$ returns the feature's value when it is informative, and zero otherwise:

$$I(f)(a_i) = \begin{cases} f(a_i) & \text{if } \exists a_i', a_i'' \in A_i : \ f(a_i') \neq f(a_i''), \\ 0 & \text{otherwise.} \end{cases}$$

The second transformation normalizes feature values to $[0, 1]$. This limits the degree to which one real-valued feature can overwhelm other features.

**Definition 11** (Normalized activation feature transformation)**.** The normalized activation transformation $N(f)$ constrains a feature $f$ to take nonnegative values that sum to 1 across all of a game's actions:

$$N(f)(a_i) = \frac{f(a_i)}{\sum_{a_i' \in A_i} f(a_i')}.$$

## 5.3 Model Selection

We took two approaches to constructing a model from the candidate features, functional forms, and transformations described in the previous section. First, we performed *forward selection* of binary features, using a linear functional form and informativeness transformation. We chose this functional form based on a manual evaluation we performed in the conference version of this chapter [Wright and Leyton-Brown, 2014], in which a linear functional form and normalized-activation- and informativeness-transformed binary features yielded good performance. Second, we performed *Bayesian optimization* to automatically evaluate combinations from the full set of candidate features, functional forms, and transformations.

### 5.3.1 Forward Selection

We performed forward selection using the following procedure. We evaluated the test performance of the Spike-Poisson QCH model, extended by a linear, normalized-activation- and informativeness-transformed level-0 model using every one- and two-element subset of the binary features from Section 5.2.1. The best such model used the minmin unfairness and max symmetric features. We then evaluated every combination of features that contained those two features. The results are shown in Figure 5.1.

The best performing linear model found by forward selection contained four features: maxmax payoff, maxmin payoff, minmin unfairness, and max symmetric. We will refer to this model henceforth as `linear4`. Adding further features did not improve prediction performance. Figure 5.2 shows the training and test performance for the best-performing model at each number of features. Notice that the training performance increased with every additional feature, whereas after the four-feature model the test performance decreased. This confirms that overfitting was the cause of the performance decrease.[4]

[4]It might seem obvious that a drop in test performance for more general models must imply overfitting. However, it could also indicate *underfitting*, where we simply do a worse job of

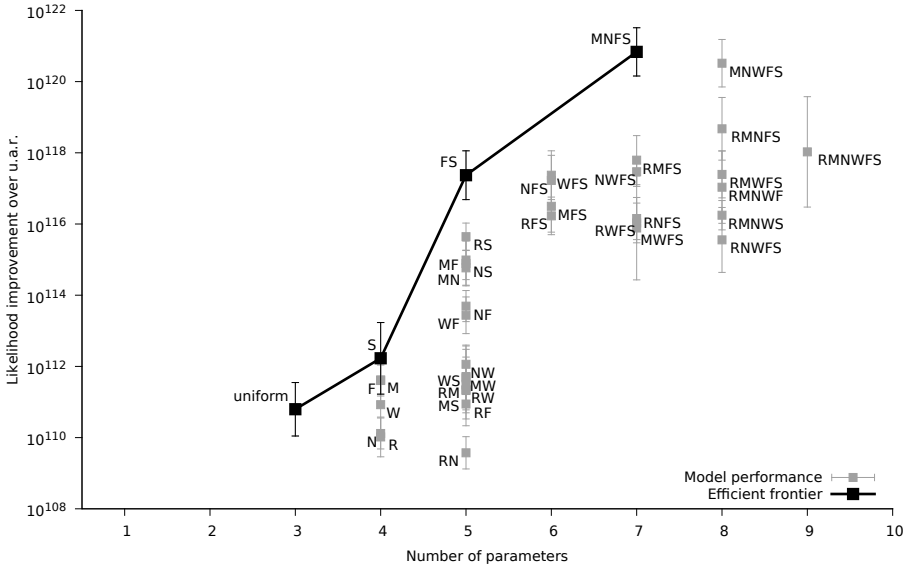


**Figure 5.1:** Prediction performance with 95% confidence intervals for Spike-Poisson QCH extended by binary features. Points are labeled by a code indicating which features were included: (M) maxmax payoff; (N) maxmin payoff; (R) minmax regret; (W) maxmax welfare; (F) minmin unfairness; (S) max symmetric.

### 5.3.2 Bayesian Optimization

We performed Bayesian optimization using SMAC [Hutter et al., 2010, 2011, 2012], a software package for optimizing the configuration of algorithms. SMAC evaluates each configuration on a randomly-chosen *instance* (i.e., input to the algorithm); it then updates a random forest model of predicted performance for configurations. It determines which configurations to evaluate based on the performance model.

We ran 16 parallel SMAC processes for 1200 hours each. The processes shared the results of each run they performed. In the context of choosing a level-0 specification, a configuration is a set of features, a set of feature transformations, and a functional form choice. An instance is a *subfold*: a seed used to randomly

optimizing the more complex models. It is this latter possibility that Figure 5.2 rules out.

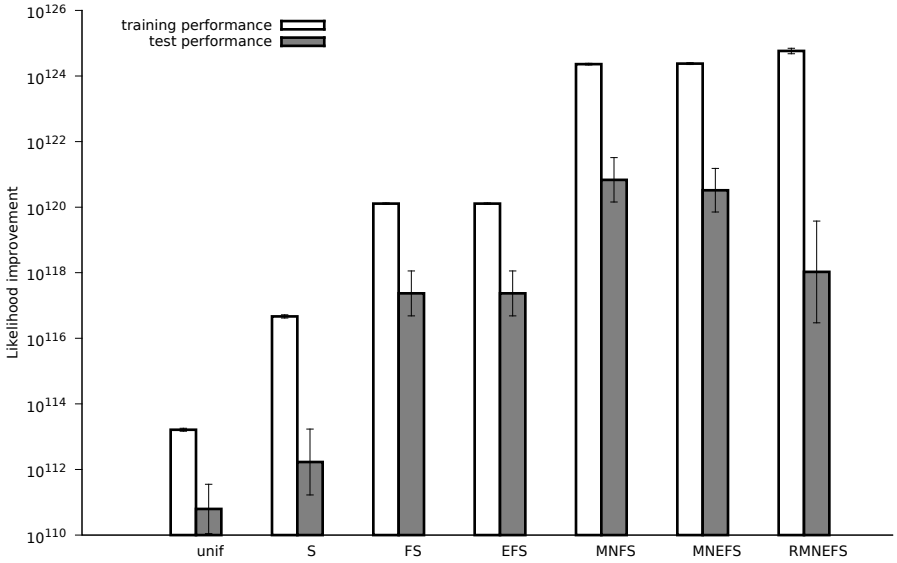


**Figure 5.2:** Training and test performance with 95% confidence intervals for Spike-Poisson QCH extended by binary features, for the best performing set of features at each number of features.

divide the ALL10 dataset into 10 folds; an index indicating which fold is the test fold; and an index indicating which subdivision of the training folds to use as the validation fold. The specified configuration was trained on the training data minus the validation fold; the performance of the trained model on the validation fold was then output to SMAC. The test fold was ignored. In this way, we attempted to avoid overfitting our dataset during model selection, by never using the test fold that we used for our final model evaluations to evaluate candidate configurations.

Figure 5.3 shows the results of the Bayesian optimization process. The best-performing model found by SMAC was a 13-parameter model that contained the same four binary features as `linear4`, plus *all* of the real-valued features described in Section 5.2.1. It combined the features linearly, with both the normalized-activation and informativeness transformations. We refer to this model as `smac`.

We were intrigued that SMAC's best-performing model was essentially `linear4`, augmented by real-valued features. We hypothesized that `linear4` augmented

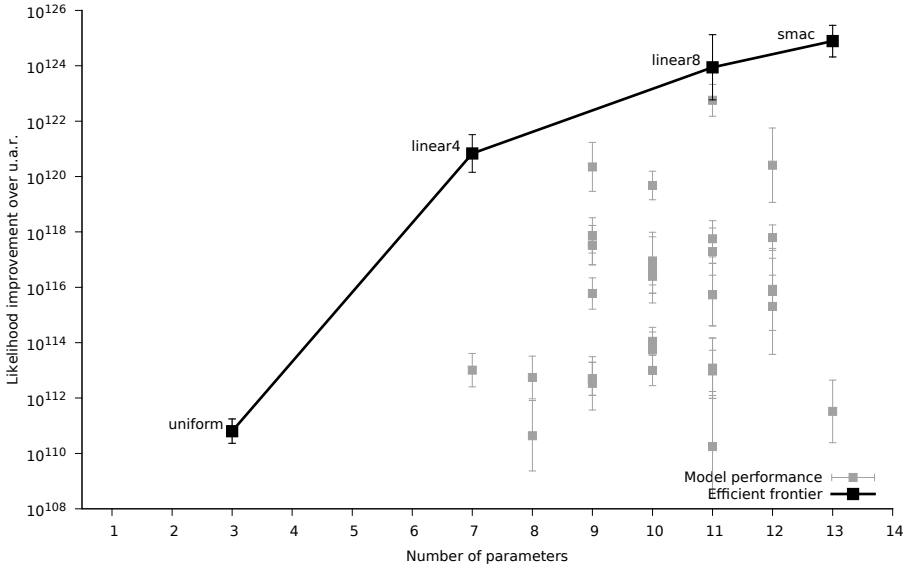


**Figure 5.3:** Prediction performance with 95% confidence intervals for Spike-Poisson QCH extended by features, functional form, and feature transformations selected by Bayesian optimization. We show only models that were at any point "incumbent" (i.e., the best found by SMAC at some point in time).

by real-valued versions of its four binary features only (i.e., excluding the welfare and max-regret features) would perform as well or better as `smac`. This model, which we refer to as `linear8`, was not checked by SMAC, and so we checked it manually. As shown in Figure 5.3, there was no significant difference between the performance of `linear8` and `smac` (although `smac` did insignificantly outperform `linear8`), even though `linear8` has two fewer features. For the remainder of the chapter, we focus our attention on the `linear4` and `linear8` models.

### 5.3.3 Extended Model Performance

We compared the predictive performance of three iterative models using three different specifications of level-0 behavior. The results are displayed in Figure 5.4. The $y$-axis gives the average ratio of the cross-validated likelihood of the extended

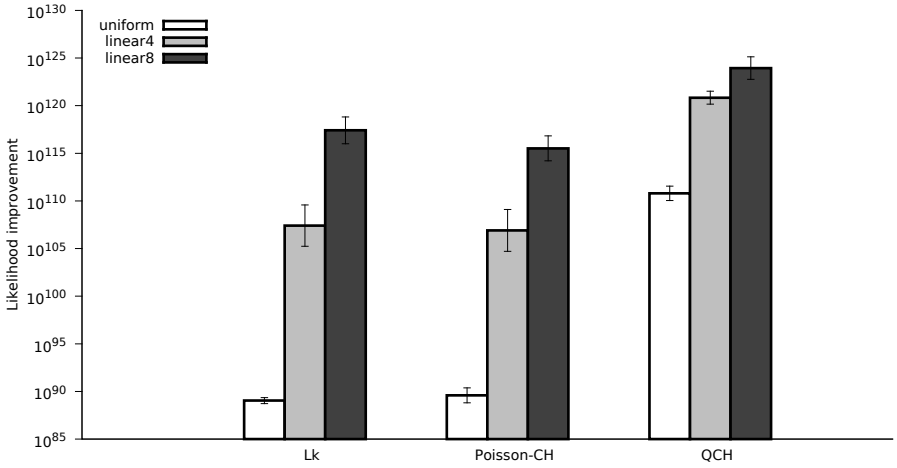


**Figure 5.4:** Average likelihood ratios of model predictions to random predictions, with $95\%$ confidence intervals. Results are shown for three different iterative models (Poisson cognitive hierarchy [Camerer et al., 2004], level-$k$ [Costa-Gomes et al., 2001], and Spike-Poisson quantal cognitive hierarchy [see Section 4.4][see Section 4.4]) using three different level-$0$ specifications (uniform randomization, `linear4` from Section 5.3.1, and `linear8` from Section 5.3.2).

models' predictions divided by the likelihood of a uniform random prediction. Overall, the `linear4` specification yielded a large performance improvement, both on Spike-Poisson QCH and also on the two other iterative models. The `linear8` specification yielded an additional, smaller performance improvement. In fact, the two other iterative models benefited disproportionately from the improved level-$0$ specifications. Spike-Poisson QCH performed better than the other two models under all level-$0$ specifications, but the three models had much more similar (and improved) performance under the `linear4` and `linear8` specifications. Adding the `linear4` level-$0$ model to Lk improved its performance by a factor of about $10^{17}$. For context, this was nearly as large as the performance gap of $10^{19}$ between Lk and QLk in Chapter 2. This is especially interesting given that the level-$0$ model was selected based solely on the degree to which it improved Spike-Poisson QCH's performance.

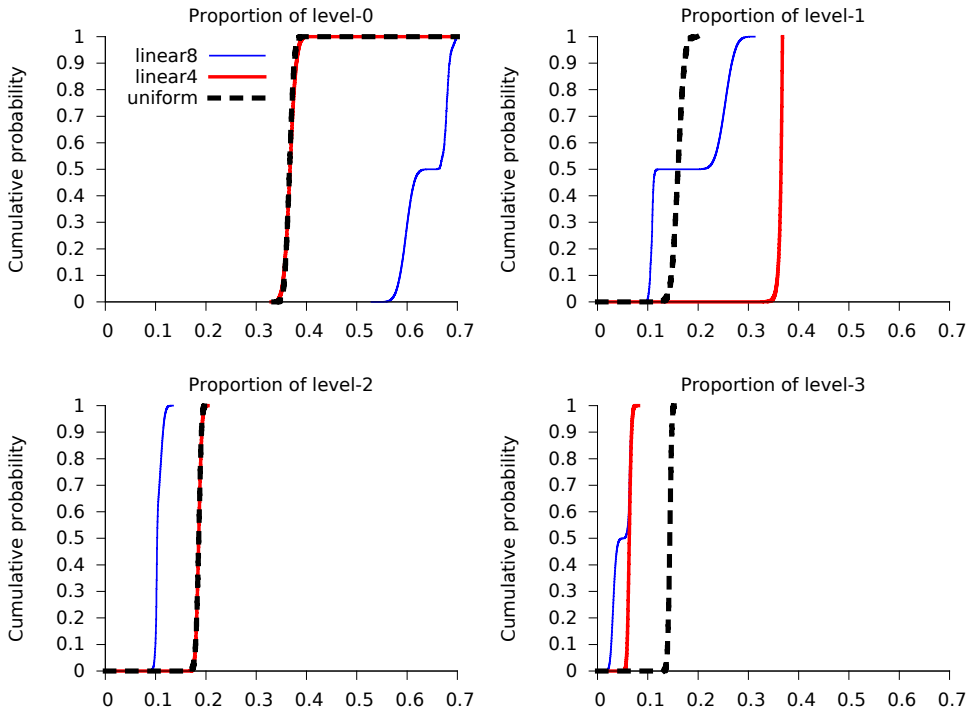


**Figure 5.5:** Marginal cumulative posterior distributions of levels of reasoning in the ALL10 dataset, for Spike-Poisson QCH with `linear8`, `linear4`, and uniform specifications.

### 5.3.4 Parameter Analysis

We now examine and interpret the posterior distributions for some of the parameters of the Spike-Poisson model combined with the `linear4` and `linear8` level-0 specifications, following the methodology of Section 3.2.2.

Figure 5.5 shows the marginal posterior distribution for each level in the population (up to level 3), for each of the `linear4`, `linear8`, and uniform specifications. We noticed two effects. First, the posterior distribution of level-0 and level-2 agents was essentially identical under the uniform and `linear4` specifications, with medians of 0.37. We found this surprising, since the level-0 agents behave very differently under the two specifications. In contrast, the posterior distribution

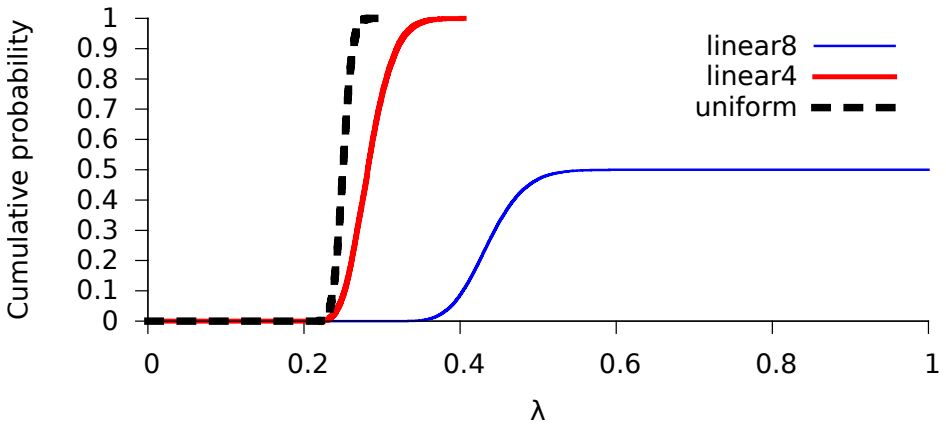


**Figure 5.6:** Marginal cumulative posterior distributions of the precision parameter ($\lambda$) in the ALL10 dataset, for Spike-Poisson QCH with `linear8`, `linear4`, and uniform specifications.

of level-0 agents under the `linear8` specification had a median of 0.65, nearly twice as large. Second, there was a large shift of mass (approximately 0.2) from agents higher than level-3 to level-1 agents under the `linear4` specification, and from all nonzero-level agents to level-0 under the `linear8` specification. This may indicate that models with a uniform level-0 specification were using higher-level agents to simulate a more accurate level-0 specification, in much the same way that QLk seemed to be using a low precision-beliefs parameter ($\lambda_{1(2)}$) to simulate a cognitive hierarchy model in Chapter 3.

The precision parameter was very similar between the uniform and `linear4` specifications. It was less sharply identified under the `linear4` specification, with mass shifting toward slightly higher precisions. Quantal response serves two purposes: it accounts for the errors of reasoning that people actually make and it also provides the error structure for the model itself. The higher precision may simply reflect the `linear4` specification's improved accuracy. Under the `linear8` specification, the precision parameter is not well identified, but has a much higher value with high probability. This lack of identification may arise from the small role played by nonzero agents in this specification.

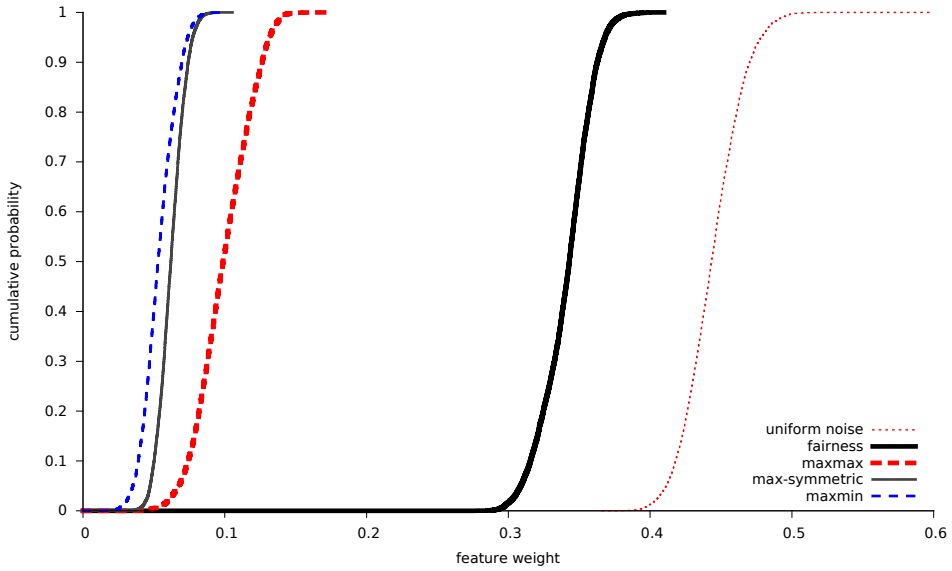


**Figure 5.7:** Marginal cumulative posterior distributions over weight parameters of the `linear4` specification, for Spike-Poisson QCH on the ALL10 dataset. The uniform noise weight is defined implicitly by the other four weights.

Figures 5.7 and 5.8 show the marginal posterior distribution for the weights of the `linear4` and `linear8` models respectively on the ALL10 dataset. As with the distribution over levels, the posterior distributions on the weight parameters had modes with very narrow supports, indicating that the data argued consistently for specific ranges of parameter values; the real-valued fairness feature of the `linear8` specification was the exception to this rule. The binary features had broadly similar weights in both the `linear4` and `linear8` specifications: the fairness feature had by far the highest median posterior weight, the maxmax feature had the second-highest weight, and the max-symmetric and maxmin features both had small and essentially identical weights, with very overlapping posterior distributions. Interestingly, even though the fairness feature was the highest weighted, it was not selected first by the forward selection procedure (max symmetric was selected first). This likely indicates that fairness is more predictive

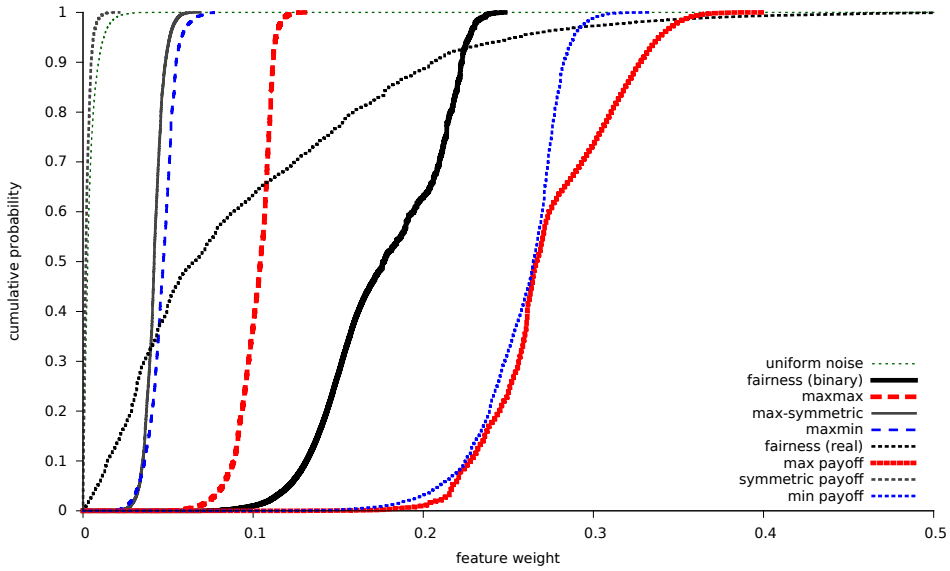


**Figure 5.8:** Marginal cumulative posterior distributions over weight parameters of the `linear8` specification, for Spike-Poisson QCH on the ALL10 dataset. The uniform noise weight is defined implicitly by the other eight weights.

than other features when it is present, but that it is predictive in fewer games than max symmetric.

The max-payoff and min-payoff real-valued features had very similar posterior weights in the `linear8` specification, with overlapping posterior supports. These were the highest-weighted features in the `linear8` specification. The real-valued fairness feature was not well identified. The symmetric-payoff feature was well-identified and had a very small weight; evidently, the action with the highest symmetric payoff is somewhat salient, but the actual value of the payoff is not salient in itself.

The weight allocated to uniform randomization between the `linear8` and `linear4` specifications is very different; the `linear4` specification allocates nearly half of its weight to uniform randomization, whereas for the `linear8` specification uniform randomization plays almost no part. This, combined with

the strong similarity in the weighting of binary features between the two specifications, suggests that the real-valued features (especially the max and min payoff features) are playing a genuine role in reducing uncertainty.

## 5.4 Related Work

Almost every study that employs iterative reasoning models of either the level-$k$ or cognitive hierarchy types assumes a uniform distribution of play for level-$0$ agents. However, there are a few exceptions. Crawford and Iriberri [2007b] specified truth-telling as the single salient action in first-price auctions. Crawford and Iriberri [2007a] manually designated certain actions as "salient" (based on visual features such as "leftmost") in a hide-and-seek game. They then estimated an iterative model with a level-$0$ specification in which level-$0$ agents play salient actions, with the strengths of each action's salience estimated under the assumption that no agent truly plays a level-$0$ distribution. Arad and Rubinstein [2009] specified a single action of reinforcing all battlefields equally in a Colonel Blotto game as the sole level-$0$ action. Arad and Rubinstein [2012] specified the highest possible request as the level-$0$ action in a money-request game where players receive the amount of money they request, but also receive a relatively large bonus for requesting exactly 1 shekel less than the other player. Arad [2012] manually specified two "anchor" strategies for a Colonel Blotto-like game in which players simultaneously assign four representatives to four separate contests in order of the representatives' ability.

In spite of the crucial dependence of iterative models upon the specification of the level-$0$ distribution, few studies have empirically investigated level-$0$ play. Agranov et al. [2010] incentivized subjects to choose an action quickly (and then to revise it after thinking) by imposing a randomized time limit when playing the beauty-contest game of Nagel [1995]. They hypothesized that early actions represent level-$0$ choices and that later actions represent higher-level choices. Based on this assumption, they found that level-$0$ behavior did not differ significantly from a uniform distribution. In contrast, Burchardi and Penczynski [2012] incentivized

players to reveal their reasoning by allowing a one-time simultaneous exchange of messages between teammates playing a beauty-contest game. Teams of two simultaneously sent each other a single message containing arguments in favor of a given action, and then simultaneously chose an action, with the team's action being chosen randomly from the two choices. Burchardi and Penczynski then classified each argument according to level of reasoning, and extracted the level-$0$ behavior from both level-$0$ and higher-level arguments. They found that level-$0$ play was significantly different from uniform. Intriguingly, they also found that the level-$0$ behavior hypothesized by higher-level players was very similar to the level-$0$ behavior that was actually proposed.

Hargreaves Heap et al. [2014] evaluate the transferability of level-$0$ specifications between three games in which all of the actions are strategically equivalent: a coordination game, a discoordination game, and a hide and seek game. They argue that any level-$0$ specification based only on the strategic structure of the game must produce an identical level-$0$ behavior for each type of game, since in each game each action is strategically equivalent to every other action. Based on experimental data, they reject a joint hypothesis that includes an identical distribution of levels for each game and an identical level-$0$ action in each game.[5] This may indicate that framing effects, in addition to strategic considerations, play a role in determining level-$0$ behavior. It may also indicate that the population distribution of levels varies between games; we are studying this latter possibility in ongoing work.

## 5.5 Conclusions

This chapter's main contribution is two specifications of level-$0$ behavior that dramatically improve the performance of each of the iterative models we evaluated—level-$k$, cognitive hierarchy, and quantal cognitive hierarchy. These specifications depend only upon the payoffs of the game, and are thus generally applicable to any

[5]They initially assume that no level-$0$ agents exist as part of their joint hypothesis. However, their results are robust to the existence of level-$0$ agents.

domain, even ones in which human intuition gives little guidance about the level-$0$ specification. A linear weighting of four nonstrategic binary features—maxmax payoff, maxmin payoff, minmin unfairness, and max symmetric—improved all three models' performances, with the weaker models (level-$k$ and cognitive hierarchy) improving the most. Including either or both of the remaining two binary features caused degradations in prediction performance due to overfitting. Fairness was the feature with by far the greatest weight. Including real-valued versions of the binary features further improved prediction performance for all three models at the expense of nearly doubling the dimensionality of the level-$0$ specification.

Conventional wisdom in the economics literature says that level-$0$ agents exist only in the minds of higher level agents; that is, that a level-$0$ specification acts solely as a starting point for higher level agents' strategies. Our results argue against this point of view: the best performing model estimated that more than a third of the agents were level-$0$. These results are strong evidence that nonstrategic behavior is an important aspect of human behavior, even in strategic settings. Further refining our understanding of nonstrategic behavior is an important direction for future work, both for the factors that are considered by nonstrategic agents, and for the details of incorporating these factors into predictive behavioral models.

# Chapter 6

# Deep Learning for Human Strategic Modeling

## 6.1 Introduction

In Chapter 5, we improved our models' performance by learning a model to determine which actions would be most attractive to nonstrategic players. Although we explored the use of Bayesian optimization to select the properties of actions that people might find salient, and which might thus be favored by nonstrategic agents, we nevertheless devised the candidate properties in the first place primarily by asking ourselves "How might I reason about playing this specific game?" Rather than relying solely on introspection and domain knowledge, one might hope to derive such properties directly from data.

The recent success of deep learning has demonstrated that predictive accuracy can often be enhanced, and expert feature engineering dispensed with, by fitting highly flexible models that are capable of learning novel representations. A key feature in successful deep models is the use of careful design choices to encode "*basic* domain knowledge of the input, in particular its topological structure... to *learn* better features" [Bengio et al., 2013, emphasis original]. For example, feed-forward neural nets can, in principle, represent the same functions as convolution

networks, but the latter tend to be more effective in vision applications, because they encode the prior that low-level features should be derived from the pixels within a small neighborhood and that predictions should be invariant to small input translations. Analogously, Clark and Storkey [2015] encoded the fact that a Go board is invariant to rotations. These modeling choices constrain more general architectures to part of the solution space that is likely to contain good solutions. Our work seeks to do the same for the behavioral game theory setting, identifying novel prior assumptions that extend deep learning to predicting behavior in strategic scenarios encoded as two player, normal-form games.

A key property required of such a model is invariance to game size: a model must be able to take as input an $m \times n$ bimatrix game (i.e., two $m \times n$ matrices encoding the payoffs of players 1 and 2 respectively) and output an $m$-dimensional probability distribution over player 1's actions, for arbitrary values of $n$ and $m$, including values that did not appear in training data. In contrast, existing deep models typically assume either a fixed-dimensional input or an arbitrary-length sequence of fixed-dimensional inputs, in both cases with a fixed-dimensional output. We also have the prior belief that permuting rows and columns in the input (i.e., changing the order in which actions are presented to the players) does not change the output beyond a corresponding permutation. In Section 6.3, we present an architecture that operates on matrices using scalar weights to capture invariance to changes in the size of the input matrices and to permutations of its rows and columns. In Section 6.5 we evaluate our model's ability to predict distributions of play given normal form descriptions of games on a dataset of experimental data from a variety of experiments, and find that our feature-free deep learning model significantly exceeds the performance of the current state-of-the-art model of Chapter 5, which has access to hand-tuned features based on expert knowledge.

## 6.2 Related Work

Deep learning has demonstrated much recent success in solving supervised learning problems in vision, speech and natural language processing [see, e.g., LeCun

et al., 2015; Schmidhuber, 2015]. By contrast, there have been relatively few applications of deep learning to multiagent settings. Notable exceptions are Clark and Storkey [2015] and the policy network used in Silver et al. [2016]'s work in predicting the actions of human players in Go. Their approach is similar in spirit to ours: they map from a description of the Go board at every move to the choices made by human players, while we perform the same mapping from a normal form game. The setting differs in that Go is a single, sequential, zero-sum game with a far larger, but fixed, action space, which requires an architecture tailored for pattern recognition on the Go board. In contrast, we focus on constructing an architecture that generalizes across general-sum, normal form games.

In its architectural design, our model is most mathematically similar to Lin et al. [2013]'s Network in Network model, though we derived our architecture independently using game theoretic invariances. We discuss the relationships between the two models at the end of Section 6.3.1.

## 6.3 Modeling Human Strategic Behavior with Deep Networks

A natural starting point in applying deep networks to a new domain is to test the performance of a regular feed-forward neural network. To apply such a model to a normal form game, we need to flatten the utility values into a single vector of length $mn + nm$ and learn a function that maps to the $m$-simplex output, possibly via multiple hidden layers. This approach is restricted to predicting the choices of players on games of some fixed size ($m \times n$), but could nevertheless be applied in that restricted domain. In practice, however, this approach often performs poorly as the network overfits the training data. One way of combating overfitting is to encourage invariance through data augmentation: for example, one may augment a dataset of images by rotating, shifting and scaling the images slightly. In games, it is natural to assume that players are indifferent to the order in which actions are

presented, implying invariance to permutations of the payoff matrix.[1] Incorporating this assumption by randomly permuting rows or columns of the payoff matrix at every epoch of training dramatically improves the generalization performance of a feed forward network (See Section 6.6 for experimental evidence from $3 \times 3$ games), but the network is still limited to games of the size that it was trained on.

Our approach is to enforce this invariance in the model architecture rather than through data augmentation. We then add further flexibility using novel "pooling units" and by incorporating iterative response ideas inspired by behavioral game theory models. The result is a model that is flexible enough to represent all the models and features of Chapter 5—and a huge space of novel models as well—and which can be identified automatically. The model is also invariant to the size of the input payoff matrix, differentiable end-to-end and trainable using standard gradient-based optimization.

The model has two parts: *feature layers* and *action response layers*; see Figure 6.1 for a graphical overview. The feature layers take the row and column player's normalized utility matrices $\mathbf{U}^{(r)}$ and $\mathbf{U}^{(c)} \in \mathbb{R}^{m \times n}$ as input, where the row player has $m$ actions and the column player has $n$ actions. The feature layers consist of multiple levels of *hidden matrix units*, $\mathbf{H}_{i,j}^{(r)} \in \mathbb{R}^{m \times n}$, each of which calculates a weighted sum of the units below and applies a non-linear activation function. Each layer of hidden units may be followed by *pooling units*, which output aggregated versions of the hidden matrices to be used by the following layer. After multiple layers, the matrices are aggregated to vectors and normalized to a distribution over actions, $\mathbf{f}_i^{(r)} \in \Delta^m$ in *softmax* units. We refer to these distributions as *features* because they encode higher-level representations of the input matrices that may be combined to construct the output distribution.

As we have seen in previous chapters, iterative strategic reasoning is an important phenomenon in human decision making; we thus want to allow our models the option of incorporating such reasoning. To do so, we compute features for the column player in the same manner by applying the feature layers to the transpose

[1] We thus ignore salience effects that could arise from action ordering.

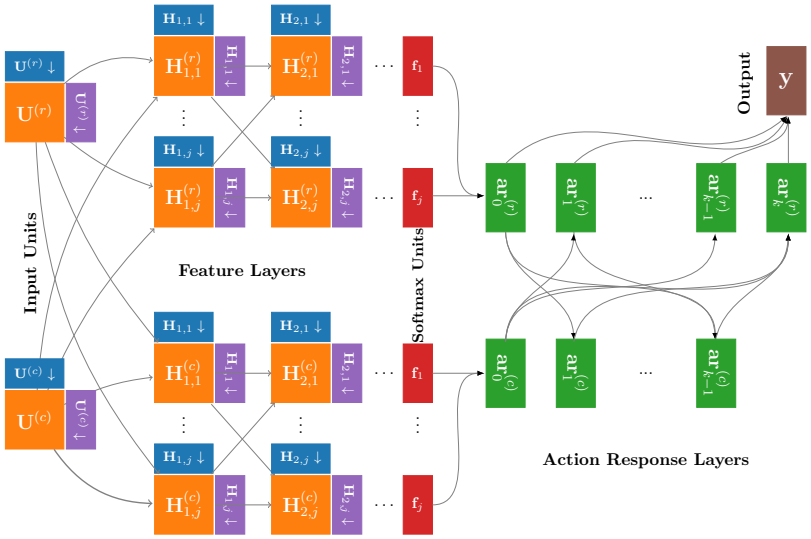


**Figure 6.1:** A schematic representation of our architecture. The feature layers consist of *hidden matrix units* (orange), each of which use *pooling units* to output row- and column-preserving aggregates (blue and purple) before being reduced to distributions over actions in the *softmax units* (red). Iterative response is modelled using the *action response layers* (green) and the final output, $y$, is a weighted sum of the row player's action response layers.

of the input matrices, which outputs $\mathbf{f}_i^{(c)} \in \Delta^n$. Each action response layer for a given player then takes the opposite player's preceding action response layers as input and uses them to construct distributions over the respective players' outputs. The final output $\mathbf{y} \in \Delta^m$ is a weighted sum of all action response layers' outputs.

### 6.3.1 Feature Layers

*Invariance Preserving Hidden Units.* We build a model that ties the parameters of our network by encoding the assumption that players reason about each action identically. Consider a normal form game represented by the utility matrices $\mathbf{U}^{(r)}$ and $\mathbf{U}^{(c)}$. Our claim that the row player evaluates each action in the same way implies that she applies the same function to each row in the utility matrices.

Thus the weights associated with each row of $\mathbf{U}^{(r)}$ and $\mathbf{U}^{(c)}$ must be the same. Similarly, the corresponding assumption about the column player implies that the weights associated with each column of $\mathbf{U}^{(r)}$ and $\mathbf{U}^{(c)}$ must also be the same. We can satisfy both assumptions by applying a single scalar weight to each of the utility matrices, computing $w_r\mathbf{U}^{(r)} + w_c\mathbf{U}^{(c)}$. This idea can be generalized as in a standard feed-forward network to allow us to fit more complex functions. A hidden matrix unit taking all the preceding hidden matrix units as input can be calculated as

$$\mathbf{H}_{l,i} = \phi\left(\sum_j w_{l,i,j}\,\mathbf{H}_{l-1,j} + b_{l,i}\right) \quad \mathbf{H}_{l,i} \in \mathbb{R}^{m\times n},$$

where $\mathbf{H}_{l,i}$ is the $i^{th}$ hidden unit matrix for layer $l$, $w_{l,i,j}$ is the $j^{th}$ scalar weight, $b_{l,i}$ is a scalar bias variable, and $\phi$ is a non-linear activation function applied element-wise. Notice that, as in a traditional feed-forward neural network, the output of each hidden unit is simply a nonlinear transformation of the weighted sum of the preceding layer's hidden units. Our architecture differs by maintaining a matrix at each hidden unit instead of a scalar. So while in a traditional feed-forward network each hidden unit maps the previous layer's vector of outputs into a scalar output, in our architecture each hidden unit maps a tensor of outputs from the previous layer into a matrix output.

Tying weights in this way reduces the number of parameters in our network by a factor of $nm$, offering two benefits. First, it reduces the degree to which the network is able to overfit; second and more importantly, it makes the model invariant to the size of the input matrices. To see this, notice that each hidden unit maps from a tensor containing the $k$ output matrices of the preceding layer in $\mathbb{R}^{k\times m\times n}$ to a matrix in $\mathbb{R}^{m\times n}$ using $k$ weights. Thus our number of parameters in each layer depends on the number of hidden units in the preceding layer, but not on the size of the input and output matrices. This allows the model to generalize to input sizes that do not appear in training data.

*Pooling units.* A limitation of the weight-tying used in our hidden matrix units is that it forces independence between the elements of their matrices, preventing the network from learning functions that compare the values of related elements (see Figure 6.2 (left)). Recall that each element of the matrices in our model corresponds to an outcome in a normal form game. A natural game theoretic notion of the "related elements" with which we'd like our model to be able to compare is the set of payoffs associated with each of the players' actions that led to that outcome. This corresponds to the row and column of each matrix associated with the particular element.

This observation motivates our pooling units, which allow information sharing by outputting aggregated versions of their input matrix that may be used by later layers in the network to learn to compare the values of a particular cell in a matrix and its row or column-wise aggregates.

$$\mathbf{H} \to \{\mathbf{H}_c, \mathbf{H}_r\} = \left\{ \begin{pmatrix} \max_i h_{i,1} & \max_i h_{i,2} & \dots \\ \max_i h_{i,1} & \max_i h_{i,2} & \dots \\ \vdots & \vdots & \\ \max_i h_{i,1} & \max_i h_{i,2} & \end{pmatrix}, \begin{pmatrix} \max_j h_{1,j} & \max_j h_{1,j} & \dots \\ \max_j h_{2,j} & \max_j h_{2,j} & \dots \\ \vdots & \vdots & \\ \max_j h_{m,j} & \max_j h_{m,j} & \dots \end{pmatrix} \right\} \tag{6.1}$$

A pooling unit takes a matrix as input and outputs two matrices constructed from row and column-preserving pooling operations respectively. We explain the pooling layer using the *max* function as our pooling operation, but we also allowed the *mean* and *sum* functions in our experiments. In Equation (6.1) we use the *max* function as the pooling operation for some arbitrary matrix $\mathbf{H}$. The first of the two outputs, $\mathbf{H}_c$, is column-preserving in that it selects the maximum value in each column of $\mathbf{H}$ and then stacks the resulting vector $n$-dimensional vector $m$ times such that the dimensionality of $\mathbf{H}$ and $\mathbf{H}_c$ are the same. Similarly, the row-preserving output constructs a vector of the max elements in each column and stacks the resulting $m$-dimensional vector $n$ times such that $\mathbf{H}_r$ and $\mathbf{H}$ have the same dimensionality. We stack the vectors that result from the pooling operation in this fashion so that at the hidden units from the next layer in the network may

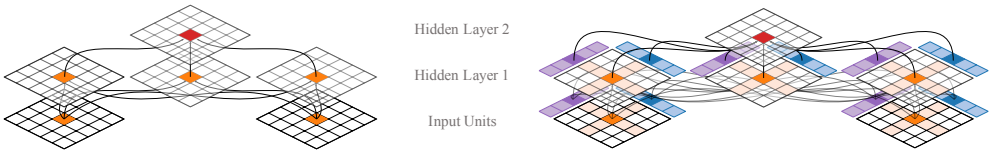


**Figure 6.2:** *Left:* Without pooling units, each element of every hidden matrix unit depends only on the corresponding elements in the units from the layer below; e.g., the middle element highlighted in red depends only on the value of the elements of the matrices highlighted in orange. *Right:* With pooling units at each layer in the network, each element of every hidden matrix unit depends both on the corresponding elements in the units below *and* the pooled quantity from each row and column. E.g., the light blue and purple blocks represent the row and column-wise aggregates corresponding to their adjacent matrices. The dark blue and purple blocks show which of these values the red element depends on. Thus, the red element depends on both the dark- and light-shaded orange cells.

take $\mathbf{H}, \mathbf{H}_c$ and $\mathbf{H}_r$ as input. This allows these later hidden units to learn functions where each element of their output is a function both of the corresponding element from the matrices below as well as their row and column-preserving maximums (see Figure 6.2 (right)).

*Softmax output.* Our model predicts a distribution over the row player's actions. In order to do this, we need to map from the hidden matrices in the final layer, $\mathbf{H}_{L,i} \in R^{m \times n}$, of the network onto a point on the $m$-simplex, $\Delta^m$. We achieve this mapping by applying a row-preserving sum to each of the final layer hidden matrices $\mathbf{H}_{L,i}$ (i.e., we sum uniformly over the columns of the matrix as described above) and then applying a softmax function to convert each of the resulting vectors $\mathbf{h}_i$ into normalized distributions. This produces $k$ features $\mathbf{f}_i$, each of which

is a distribution over the row player's $m$ actions:

$$\mathbf{f}_i = \text{softmax}\left(\mathbf{h}^{(i)}\right),$$
$$\text{where } \mathbf{h}_j^{(i)} = \sum_{k=1}^{n} h_{j,k}^{(i)} \text{ for all } j \in \{1, ..., m\},\ h_{j,k}^{(i)} \in \mathbf{H}^{(i)} \quad i \in \{1, ..., k\}.$$

We can then produce the output of our features, $\mathbf{ar}_0$, using a weighted sum of the individual features, $\mathbf{ar}_0 = \sum_{i=1}^{k} w_i \mathbf{f}_i$, where we optimize $w_i$ under simplex constraints, $w_i \geq 0$, $\sum_i w_i = 1$. Because each $\mathbf{f}_i$ is a distribution and our weights $w_i$ are points on the simplex, the output of the feature layers is a mixture of distributions.

*Action Response Layers.* The feature layers described above are sufficient to meet our objective of mapping from the input payoff matrices to a distribution over the row player's actions. However, this architecture is not capable of explicitly representing iterative strategic reasoning, which we have argued is an important modeling ingredient. We incorporate this ingredient using action response layers: the first player can respond to the second's beliefs, the second can respond to this response by the first player, and so on to some finite depth. The proportion of players in the population who iterate at each depth is a parameter of the model; thus, our architecture is also able to learn not to perform iterative reasoning.

More formally, we begin by denoting the output of the feature layers as $\mathbf{ar}_0^{(r)} = \sum_{i=1}^{k} w_{0i}^{(r)} \mathbf{f}_i^{(r)}$, where we now include an index $(r)$ to refer to the output of *row* player's action response layer $\mathbf{ar}_0^{(r)} \in \Delta^m$. Similarly, by applying the feature layers to a transposed version of the input matrices, the model also outputs a corresponding $\mathbf{ar}_0^{(c)} \in \Delta^n$ for the column player which expresses the row player's beliefs about which actions the column player will choose. Each action response layer composes its output by calculating the expected value of an internal representation of utility with respect to its belief distribution over the opposition actions. For this internal representation of utility we chose simply a weighted sum of the final layer of the hidden layers, $\sum_i w_i \mathbf{H}_{L,i}$, because each $\mathbf{H}_{L,i}$ is already

some non-linear transformation of the original payoff matrix, and so this allows the model to express utility as a transformation of the original payoffs. Given the matrix that results from this sum, we can compute expected utility with respect to the vector of beliefs about the opposition's choice of actions, $\mathbf{ar}_j^{(c)}$, by simply taking the dot product of the weighted sum and beliefs. When we iterate this process of responding to beliefs about one's opposition more than once, higher level players will respond to beliefs, $\mathbf{ar}_i$, for all $i$ less than their level and then output a weighted combination of these responses using some weights, $v_{l,i}$. Putting this together, the $l^{th}$ action response layer for the *row* player $(r)$ is defined as

$$\mathbf{ar}_l^{(r)} = \text{softmax}\left(\lambda_l\left(\sum_{j=0}^{l-1} v_{l,j}^{(r)}\left(\sum_{i=1}^{k} w_{l,i}^{(r)}\mathbf{H}_{L,i}^{(r)}\right)\cdot\mathbf{ar}_j^{(c)}\right)\right), \quad \mathbf{ar}_l^{(r)} \in \Delta^m, l \in \{1, ..., K\},$$

where $l$ indexes the action response layer, $\lambda_l$ is a scalar sharpness parameter that allows us to sharpen the resulting distribution, $w_{l,i}^{(r)}$ and $v_{l,j}^{(r)}$ are scalar weights, $\mathbf{H}_{L,i}$ are the *row* player's $k$ hidden units from the final hidden layer $L$, $\mathbf{ar}_j^{(c)}$ is the output of the *column* player's $j^{th}$ action response layer and $K$ is the total number of action response layers. We constrain $w_{li}^{(r)}$ and $v_{lj}^{(r)}$ to the simplex and use $\lambda_l$ to sharpen the output distribution so that we can optimize the sharpness of the distribution and relative weighting of its terms independently. We build up the column player's action response layer, $\mathbf{ar}_l^{(c)}$, similarly, using the column player's internal utility representation, $\mathbf{H}_{L,i}^{(c)}$, responding to the row player's action response layers, $\mathbf{ar}_l^{(r)}$. These layers are not used in the final output directly but are relied upon by subsequent action response layers of the row player.

*Output.* Our model's final output is a weighted sum of the outputs of the action response layers. This output needs to be a valid distribution over actions, and because each of the action response layers also outputs a distribution over actions, we can achieve this requirement by constraining these weights to the simplex, thereby ensuring that the output is just a mixture of distributions. The model's output is thus $\mathbf{y} = \sum_{j=1}^{K} w_j \mathbf{ar}_j^{(r)}$, where $\mathbf{y}$ and $\mathbf{ar}_j^{(r)} \in \Delta^m$, and $w_j \in \Delta^K$.

*Relation to existing deep models.* We designed this model to exploit game theoretic invariances, but the resulting functional form has connections with existing deep model architectures. We discuss two of these connections here. First, our invariance-preserving hidden layers can be encoded as *MLP Convolution Layers* [Lin et al., 2013] with the two-channel $1 \times 1$ input $x_{i,j}$ corresponding to the two players' respective payoffs when actions $i$ and $j$ are played. Using patches larger than $1 \times 1$ would require assuming that local structure is important, which is inappropriate in our domain; thus, we do not need multiple *mlpconv* layers. Second, our pooling units are superficially similar to the pooling units used in convolution networks. However, ours are designed for a different purpose: we use pooling as a way of sharing information between cells in the matrices that are processed through our network, while in computer vision, max-pooling units are used to produce invariance to small translations of the input image.

## 6.4 Experimental Setup

We used the ALL10 dataset introduced in Section 2.4.1 for our performance experiments in this chapter. We evaluated prediction performance using the gamewise cross-validation procedure described in Section 2.3.2.

We trained our models using a combination of stochastic gradient descent and RMSProp. All the softmax functions in our network initially output very flat distributions because the input matrices are normalized to lie within $[-1, 1]$ and the network weights are initialized to small random values. We noticed that the optimization converged to better solutions if we combat this effect by scaling the softmax inputs by a large constant (typically = 100) in order to sharpen the distributions output by the network.

Our architecture imposes simplex constraints on weight parameters. Fortunately, simplex constraints fall within the class of *simple constraints* that can be efficiently optimized using the projected gradient algorithm first proposed in [Goldstein, 1964] which projects the relevant parameters onto the constraint set at the end of every epoch of the standard stochastic gradient decent algorithm.

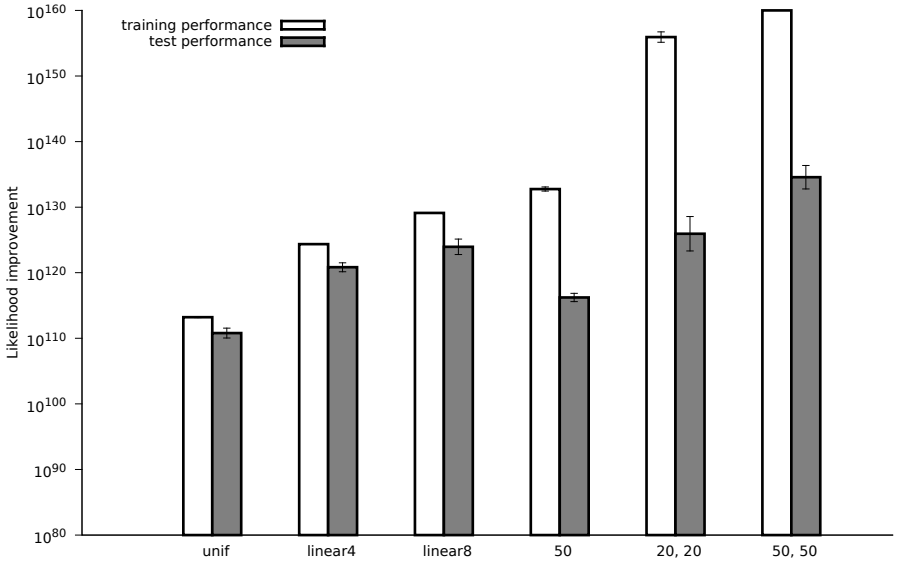


**Figure 6.3:** Prediction performance with 95% confidence intervals on the ALL10 dataset, for varying numbers of hidden units and layers, with a single action response layer. Spike-Poisson QCH with uniform, `linear4`, and `linear8` level-0 models are included for comparison.

## 6.5 Experimental Results

Figures 6.3, 6.4, and 6.5 show performance comparisons between models built using our deep learning architecture (details below) and the previous state of the art, Quantal Cognitive Hierarchy with hand-crafted features; for reference we also include the best feature-free model, QCH with a uniform model of level-0 behavior. Our model significantly improved on both alternatives and thus represents a new state of the art. Notably, the magnitude of the improvement was larger than that of adding hand-crafted features to the original QCH model.

Figure 6.3 considers the effect of varying the number of hidden units and layers on performance using a single action response layer. Perhaps unsurprisingly, we found that the network performed poorly on both training and test data with only a single hidden layer of 50 units and with two layers of 20 units per layer, but improved significantly with two layers of 50 hidden units. To test the effect

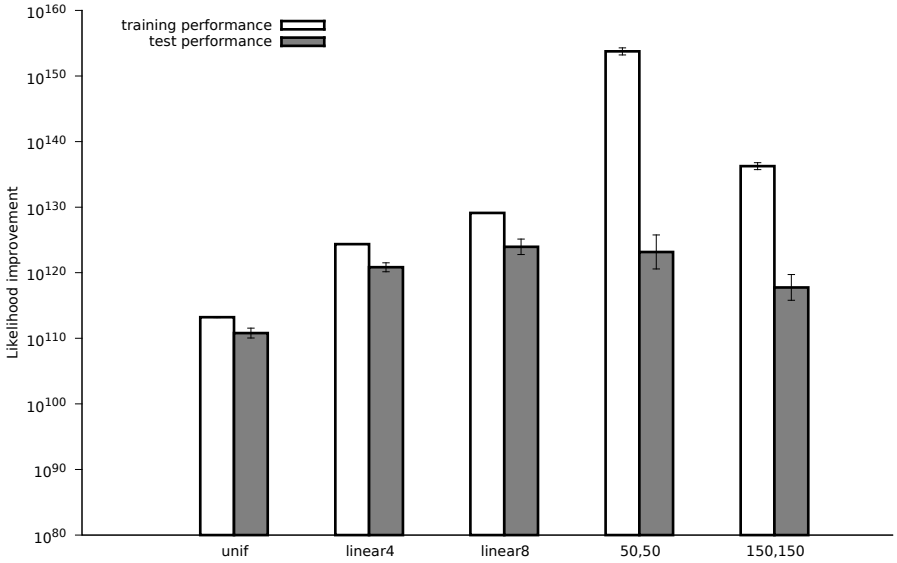


**Figure 6.4:** Prediction performance with 95% confidence intervals on the ALL10 dataset, for varying numbers of hidden units and layers, without pooling units. Spike-Poisson QCH with uniform, `linear4`, and `linear8` level-0 models are included for comparison.

of pooling units on performance, in Figure 6.4 we remove the pooling units from the network of 2 layers of 50 hidden units and also consider a much larger network with 3 layers of 100 hidden units and no pooling units. The results clearly demonstrate the value of pooling layers. The network performed poorly on both the training and test sets in the $50 \times 50$ network with no pooling layers. We were not able to improve test performance by using a larger number of hidden nodes; in fact, even training performance failed to improve, indicating the delicate balance that needs to be struck between overfitting and underfitting (where an extremely flexible model is too difficult to train effectively). Thus, our final network contained two layers of 50 hidden units and pooling units.

Our next set of experiments committed to this configuration for feature layers and investigated configurations of action response layers, varying their number between one and three (i.e., from no iterative reasoning up to two levels of itera-

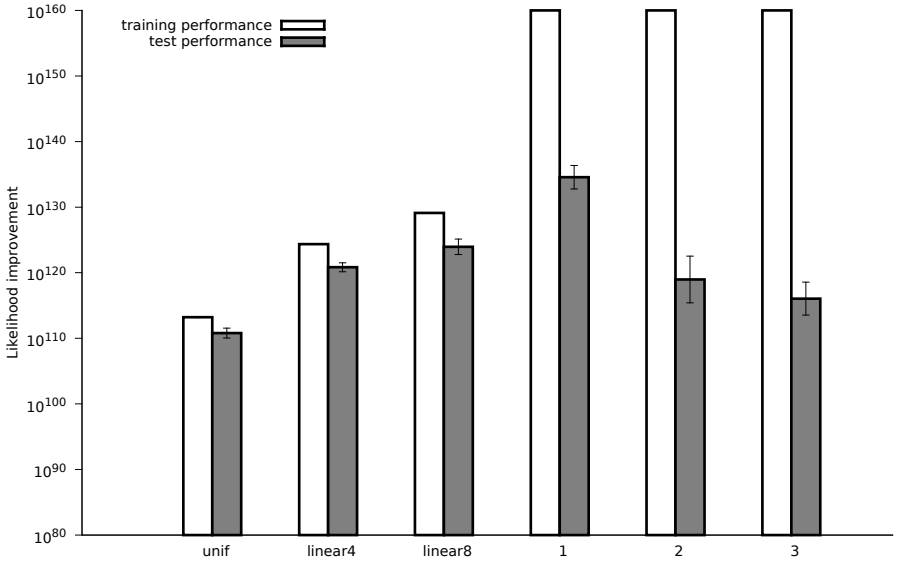


**Figure 6.5:** Prediction performance with 95% confidence intervals on the ALL10 dataset, for GameNet architecture with two layers of 50 hidden units, with pooling units, with varying numbers of action response layers. Spike-Poisson QCH with uniform, `linear4`, and `linear8` level-0 models are included for comparison.

tive reasoning; see Figure 6.5). All networks with more than one action response layer showed clear signs of overfitting: performance on the training set improved significantly but test set performance suffered. Thus, our final network used only one action response layer. We nevertheless remain committed to an architecture that can capture iterative strategic reasoning; we intend to investigate more effective methods of regularizing the parameters of action response layers in future work.

## 6.6 Regular Neural Network Performance

Figure 6.6 compares the performance of our architecture with that of a regular feed forward neural network, with and without data augmentation and the previous state-of-the-art model on this dataset. Since a regular feed-forward network can

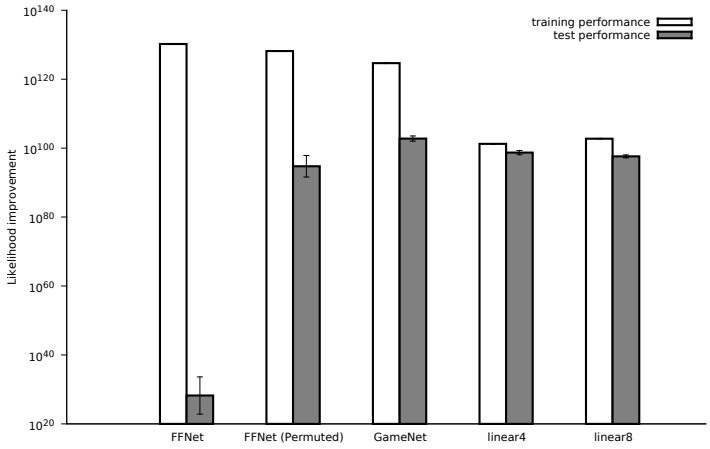


**Figure 6.6:** Performance comparison on $3\times3$ games of a feed forward neural network (FFNet), a feed forward neural network with data augmentation at every epoch (FFNet (Permuted)), our architecture fit with the same hyper parameters as used for our best performing model in the main results (GameNet), and Quantal Cognitive Hierarchy with four and eight hand-crafted features (QCH Linear4 and QCH Linear8 respectively).

only operate on games of a single dimensionality, we show results for the subset of ALL10 that contains only $3 \times 3$ games. The feed forward network dramatically overfits the data without data augmentation. Data augmentation improves test set performance, but it is still unable to match the state of the art performance. Our architecture improves upon the state of the art even in this relatively restricted subset of the full dataset.

## 6.7 Conclusions

We present an architecture that we call GameNet for learning such models that both significantly improves upon state-of-the-art performance without needing hand-tuned features developed by domain experts, and is powerful enough to generalize across inputs of different sizes.

Although GameNet can represent many game theoretic and behavioral con-

cepts, it is not perfectly expressive. We plan to explore the effect of including additional operators that are able to encode game theoretic concepts that we believe our model cannot currently represent, such as dominance and framing effects.

Overall, we believe that deep learning is an extremely promising direction for constructing models of human strategic behavior. Our architecture demonstrates that it is possible to learn a highly performant behavioral model automatically from data, with minimal need for fine-tuning by domain experts.

Although GameNet outperforms the structural models of earlier chapters, it provides no insight into how or why people choose certain actions. The deep model's superior prediction performance demonstrates that there is room to improve the structural models. Ideally, we would like to combine the interpretability of a structural model with the superior prediction performance of a deep model.

# Part II

# Application Domain: Peer Grading

# Chapter 7

# Incentivizing Evaluation via Limited Access to Ground Truth

In this part, we shift our attention from modeling human behavior in generic strategic environments to the specific strategic setting of peer grading. We begin by theoretically analyzing the incentivized elicitation setting, which includes peer grading as a special case. In the next chapter, we will present a case study of a real-life peer grading system.

This chapter does not directly build upon the behavioral foundation of Part I. Instead, we analyze the properties of the Nash equilibria of elicitation mechanisms. We took this approach for two reasons. First, this chapter's results can be understood as showing that peer-prediction mechanisms do not succeed even by their own standards; without any relaxation of rationality assumptions, an only slightly more realistic setting is enough to invalidate their equilibrium guarantees. Second, we show that the mechanism that most efficiently uses ground truth produces a dominant strategy for the agents. Many of the strategic considerations of Part I's framework (reasoning about nonstrategic agents, limits to strategic reasoning) do not apply in the context of a dominant-strategy mechanism, where an agent need not reason strategically about the other agents at all.[1]

[1]Other behavioral considerations, such as quantal response, *do* still apply in such a setting, but

## 7.1 Introduction

There are many practical settings in which it is important to collect accurate evaluations about objects of interest from dispersed individuals. For example, many millions of users rely on feedback from Rotten Tomatoes, Yelp, and TripAdvisor to choose among competing movies, restaurants, and travel destinations. Crowdsourcing platforms provide another example, enabling the collection of semantic labels of images and online content for use in training machine learning algorithms.

We are particularly motivated by the peer grading problem, which we will use as a running example. Students benefit from open-ended assignments such as essays or proofs. However, such assignments are used relatively sparingly, particularly in large classes, because they require considerable time and effort to grade properly. An efficient and scalable alternative is having students grade each other (and, in the process, learn from each other's work). Many peer grading systems have been proposed and evaluated in the education literature [Hamer et al., 2005; Cho and Schunn, 2007a; Paré and Joordens, 2008; de Alfaro and Shavlovsky, 2014a; Kulkarni et al., 2014; and Chapter 8 of this dissertation], albeit with a focus on evaluating the accuracy of grades collected under the assumption of full cooperation by students.

However, no experienced teacher would expect all students to behave non-strategically when asked to invest effort in a time-consuming task. An effective peer grading system must therefore provide motivation for students to formulate evaluations carefully and to report them honestly. Many approaches have been developed to provide such motivation. One notable category is peer-prediction methods [Prelec, 2004; Miller et al., 2005; Jurca and Faltings, 2009; Faltings et al., 2012; Witkowski and Parkes, 2012; Witkowski et al., 2013; Dasgupta and Ghosh, 2013; Witkowski and Parkes, 2013; Radanovic and Faltings, 2013, 2014; Riley, 2014; Zhang and Chen, 2014; Waggoner and Chen, 2014; Kamble et al., 2015; Kong et al., 2016; Shnayder et al., 2016]. In order to motivate each agent to re-

we do not explore them in this chapter.

veal his private, informative signal, peer-prediction methods offer a reward based on how each agent's reports compare with those of his peers. Such rewards are designed to induce truth telling in equilibrium—that is, they create a situation in which each agent has an interest in investing effort and revealing his private and informative signal truthfully, as long as he believes that all other agents will do the same.

Even if they do offer a truthful equilibrium, peer-prediction methods also always induce other uninformative equilibria, the existence of which is inevitable [Jurca and Faltings, 2009; Waggoner and Chen, 2014]. Intuitively, if no other agent follows a strategy that depends on her private information, there is no reason for a given agent to deviate in a way that does so either: agents can only be rewarded for coordination, not for accuracy. When private information is costly to obtain, uninformative equilibria are typically *less* demanding for agents to play. This raises significant doubt about whether peer-prediction methods can motivate truthful reporting in practice. Experimental evaluations of peer-prediction methods have mixed results. Some studies showed that agents reported truthfully [Shaw et al., 2011; John et al., 2012; Faltings et al., 2014]; another study found that agents colluded on uninformative equilibria [Gao et al., 2014].

Recent progress on peer-prediction mechanisms has focused on making the truthful equilibrium Pareto dominant, i.e., (weakly) more rewarding to every agent than any other equilibrium [Dasgupta and Ghosh, 2013; Witkowski and Parkes, 2013; Kamble et al., 2015; Radanovic and Faltings, 2015; Shnayder et al., 2016]. This can be achieved by rewarding agents based on the distributions of their reports for multiple objects. However, we show in this chapter that such arguments rely critically on the assumption that every agent has access to only one private signal per object. This is often untrue in practice; e.g., in peer grading, by taking a quick glance at an essay a student can observe characteristics such as length, formatting and the prevalence of grammatical errors. These characteristics require hardly any effort to observe, can be arbitrarily uninformative about true quality, and are of no interest to the mechanism. Yet their existence provides a means for

the agents to coordinate. We build on this intuition to prove that no mechanism can guarantee that an equilibrium in which all agents truthfully report their informative signals is always Pareto dominant. Furthermore, we show that for any mechanism, the truthful equilibrium is *always* Pareto dominated in some settings.

Motivated by these negative results, we move on to consider a setting in which the operator of the mechanism has access to trusted evaluators (e.g., teaching assistants) who can reliably provide noisy but informative signals of the object's true quality. This allows for a hybrid mechanism that blends peer-prediction with comparison to trusted reports. With a fixed probability, the mechanism obtains a trusted report and rewards the agent based on the agreement between the agent's report and the trusted report [Jurca and Faltings, 2005]. Otherwise, the mechanism rewards the agent using a peer-prediction mechanism. Such hybrid mechanisms can yield stronger incentive guarantees than other peer-prediction mechanisms, such as achieving truthful reporting of informative signals in Pareto-dominant equilibrium [e.g., Jurca and Faltings, 2005; Dasgupta and Ghosh, 2013]. Intuitively, if an agent seeks to be consistently close to a trusted report, then his best strategy is to reveal his informative signal truthfully.

In fact, the availability of trusted reports is so powerful that it gives us the option of dispensing with peer-prediction altogether. Specifically, we can reward students based on agreement with the trusted report when the latter is available, but simply pay students a constant reward otherwise. Indeed, in Chapter 8 we describe such a peer grading system and show that it worked effectively in practice, based on a study across three years of a large class. This mechanism has even stronger incentive properties than the hybrid mechanism—because it induces a single-agent game, it can give rise to dominant-strategy truthfulness.

This chapter's main focus is on comparing these two approaches in terms of the number of trusted reports that they require. One might expect that the peer-prediction approach would have the edge, both because it relies on a weaker solution concept and because it leverages a second source of information reported by other agents. Surprisingly, we prove that this intuition is backwards. We identify a

simple sufficient condition, which, if satisfied, guarantees that the peer-insensitive mechanism offers the dominant strategy of truthful reporting of informative signals while querying trusted reports with a lower probability than is required for a peer-prediction mechanism to motivate truthful reporting in Pareto-dominant equilibrium. We then show that all applicable peer-prediction mechanisms of which we are aware satisfy this sufficient condition.

## 7.2 Peer-Prediction Mechanisms

We begin by formally defining the game theoretic setting in which we will study the elicitation problem. A mechanism designer wishes to elicit information about a set $O$ of objects from $n$ risk-neutral agents. Each object $j$ has a latent quality $q_j \in Q$, where $Q$ is a finite set. For each object $j$, agent $i$ has access to two pieces of private information:[2] a *high-quality signal* $s_{ij}^h \in Q$, which is drawn from a distribution conditional on the actual quality $q_j$; and a *low-quality signal* $s_j^l$, which may be uncorrelated with $q_j$. The joint distributions of both the high-quality and low-quality signals are common knowledge among the agents. In particular, an agent $i$ can form a belief about the high-quality signal of another agent $i'$ by conditioning on his own high-quality signal.

We assume that agents can observe the low-quality signal $s_j^l$ without effort, but that observing the high-quality signal $s_{ij}^h$ requires a constant effort $c^E > 0$. Agents may strategize over both whether to incur the cost of effort to observe the high-quality signal and over what to report. The mechanism designer's goal is to incentivize each agent to both observe the high-quality signal, and to truthfully report it. We say that a mechanism has a *truthful equilibrium* when it is an equilibrium for agents to observe the high-quality signal and truthfully report it (and, for some mechanisms, their posterior belief about other agents' high-quality signals).

[2]We depart from a standard assumption in the literature: that agents receive only a single signal about any object. We argue that agents also inevitably have access to simple metadata about any object they are asked to evaluate, such as its name, location, color, the length of its description, etc. We model all of this metadata as a second, costless signal that is perfectly correlated across agents.

In the context of peer grading, the agents are the students assigned to grade each others' assignments, and the objects are the assignments. The high-quality signal is obtained by carefully reading and grading an assignment. The low-quality signal is obtained by observing trivial characteristics of the assignment such as its length, title, formatting, etc.

The mechanism designer's aim is to incentivize each agent $i \in \{1, \ldots, n\}$ to gather and truthfully report information about every object $j \in J_i \subseteq O$; in this chapter we denote by $J_i$ the set of objects assigned to an agent for evaluation, rather than the number of actions in a game as in Chapter 2. Let $r_{ij}$ and $b_{ij}$ denote agent $i$'s signal and belief reports for object $j$ respectively. A mechanism is defined by a reward function, which maps a profile of agent reports to a reward for each agent. We say that a mechanism is *universal* if it can be applied without prior knowledge of the distribution from which signals are elicited, and for any number of agents greater than or equal to 3.

**Definition 12** (Universal peer-prediction mechanism)**.** A peer-prediction mechanism is *universal* if it can be operated without knowledge of the joint distribution of the high-quality signals $s_{ij}^h$ (i.e., it is "detail free" [Wilson, 1987]) and well defined for any number of agents $n \geq 3$.

We focus on universal mechanisms for two reasons. First, in practice, it is extremely unrealistic to assume that a mechanism designer will have detailed knowledge of the joint signal distribution, so this allows us to focus on mechanisms that are more likely to be used in practice. Second, it is relatively unrestrictive, as nearly all of the peer-prediction mechanisms in the literature satisfy universality.

Existing, universal peer-prediction mechanisms can be divided into three categories: output agreement mechanisms, multi-object mechanisms, and belief based mechanisms.

*Output Agreement Mechanisms.* Output agreement mechanisms only collect signal reports from agents and reward an agent $i$ for evaluating object $j$ based on agents' signal reports for the object [Faltings et al., 2012; Witkowski et al., 2013;

Waggoner and Chen, 2014]. Waggoner and Chen [2014] and Witkowski et al. [2013] studied the standard output agreement mechanism, where agent $i$ is only rewarded when his signal report matches that of another randomly chosen agent $j$. Agent $i$'s reward is $z_i(r) = \mathbb{1}_{r_{ij}=r_{i'j}}$. The Faltings et al. [2012] mechanism also rewards agents for agreement, scaled by the empirical frequency of the report agreed upon. Agent $i$'s reward is $z_i(r) = \alpha + \beta\frac{\mathbb{1}_{r_{ij}=r_{i'j}}}{F(r_{i'j})}$, where $\alpha > 0$ and $\beta > 0$ are constants and $F(r_j)$ is the empirical frequency of $r_j$.

*Multi-Object Mechanisms.* Multi-object mechanisms reward each agent based on his reports for multiple objects [Dasgupta and Ghosh, 2013; Radanovic and Faltings, 2015; Kamble et al., 2015; Shnayder et al., 2016]. (The Shnayder et al. [2016] mechanism generalizes the Dasgupta and Ghosh [2013] mechanism to the multi-signal setting. Thus, we only refer to the Shnayder et al. [2016] mechanism below.)

The Shnayder et al. [2016] and Kamble et al. [2015] mechanisms also reward agents for agreement, as in output agreement mechanisms. They extend output agreement mechanisms by adding additional scaling terms to the reward. These scaling terms are intended to exploit correlations between multiple tasks to make the truthful equilibrium dominate (a particular kind of) uninformative equilibria, by reducing the reward to agents who agree to an amount that is "unsurprising" given their reports on other objects.

The Shnayder et al. [2016] mechanism adds an additive scaling term to the reward for agreement. To compute the scaling term, consider two sets of non-overlapping tasks $S_i$ and $S_{i'}$ such that agent $i$ has evaluated all objects in $S_i$ but none in $S_{i'}$ and agent $i'$ has evaluated all objects in $S_{i'}$ but none in $S_i$. Let $F_i(s)$ and $F_{i'}(s)$ denote the frequency of signal $s \in Q$ in sets $S_i$ and $S_{i'}$ respectively. Agent $i$ is rewarded according to $z_i(r) = \mathbb{1}_{r_{ij}=r_{i'j}} - \sum_{s\in Q} F_i(s)F_{i'}(s)$.

In contrast, the Kamble et al. [2015] mechanism adds a multiplicative scaling term to the reward for agreement. To compute the scaling term, choose 2 agents $k$ and $k'$ uniformly at random. For each signal $s \in Q$, let $f^j(s) = \mathbb{1}_{r_{kj}=s}\mathbb{1}_{r_{k'j}=s}$.

Define $\hat{f}(s) = \sqrt{\frac{1}{N}\sum_{j \in O} f^j(s)}$. If $\hat{f}(s) \in \{0, 1\}$, then agent $i$'s reward is 0. Otherwise, agent $i$'s reward is $\mathbb{1}_{r_{ij}=r_{i'j}} \cdot \frac{K}{\hat{f}(s)}$ for some constant $K > 0$.

The Radanovic and Faltings [2015] mechanism rewards the agents for report agreement using a reward function inspired by the quadratic scoring rule. To reward agent $i$ for evaluating object $j$, first choose another random agent $i'$ who also evaluated object $j$. Then construct a sample $\Sigma_i$ of reports which contains one report for every object that is not evaluated by agent $i$. The sample $\Sigma_i$ is double-mixed if it contains all possible signal realizations at least twice. If $\Sigma_i$ is not double-mixed, agent $i$'s reward is 0. Otherwise, if $\Sigma_i$ is double-mixed, the mechanism chooses two objects $j'$ and $j''$ ($j' \neq j$, $j'' \neq j$ and $j' \neq j''$) such that the reports of $j'$ and $j''$ in the sample are the same as agent $i$'s report for $j$, i.e. $\Sigma_i(j') = \Sigma_i(j'') = r_{ij}$. For each of $j'$ and $j''$, randomly select two reports $r_{i''j'}$ and $r_{i'''j''}$. Agent $i$'s is rewarded according to $z_i(r) = \frac{1}{2} + \mathbb{1}_{r_{i''j'}=r_{i'j}} - \frac{1}{2}\sum_{s \in Q} \mathbb{1}_{r_{i''j'}=s}\mathbb{1}_{r_{i'''j''}=s}$.

*Belief Based Mechanisms.* Finally, some peer-prediction mechanisms collect both signals and belief reports from agents and reward each agent based on all agents' signal and belief reports for each object [Witkowski and Parkes, 2012, 2013; Radanovic and Faltings, 2013, 2014; Riley, 2014]. Below, let $R$ denote a proper scoring rule.

The robust Bayesian Truth Serum (BTS) [Witkowski and Parkes, 2012, 2013] rewards agent $i$ for how well his belief report $b_i$ and shadowed belief report $b'_i$ predict the signal reports of another randomly chosen agent $k$. Agent $i$'s reward is $z_i(r, b) = R(b'_i, r_k) + R(b_i, r_k)$. Agent $i$'s shadowed belief report is calculated based on his signal report and another random agent $j$'s belief report: $b'_i = b_j + \delta$ if $r_i = 1$ and $b'_i = b_j - \delta$ if $r_i = 0$ where $\delta = \min(b_j, 1 - b_j)$.

The multi-valued robust BTS [Radanovic and Faltings, 2013] rewards agent $i$ if his signal report matches that of another random agent $j$ and his belief report accurately predicts agent $j$'s signal report. Agent $i$'s reward is $z_i(r, b) = \frac{1}{b_j(r_i)}\mathbb{1}_{r_i=r_j} + R(b_i, r_j)$.

The divergence-based BTS [Radanovic and Faltings, 2014] rewards agent $i$ if his belief report accurately predicts another random agent $j$'s signal report. In addition, it penalizes agent $i$ if his signal report matches that of agent $j$ but his belief report is sufficiently different from that of agent $j$. Agent $i$'s reward is $-\mathbb{1}_{r_i=r_j||D(b_i,b_j)>\theta} + R(b_i, r_j)$ where $D(||)$ is the divergence associated to the strictly proper scoring rule $R$, and $\theta$ is a parameter of the mechanism.

The Riley [2014] mechanism rewards agent $i$ for how well his belief report predicts other agents' signal reports. Moreover, agent $i$'s reward is bounded above by the score for the average belief report of other agents reporting the same signal. Formally, let $\delta_i = \min_{s \in Q} |\{r_j = s | j \neq i\}|$ be the minimum number of other agents who have reported any given signal. If $\delta_i = 0$, agent $i$'s reward is $R(b_i, r_{-i})$. Otherwise, if $\delta_i \geq 1$, compute the proxy prediction $q_i(r_i)$ to be the average belief report for all other agents who made the same signal report as agent $i$. Agent $i$'s reward is $\min\{R(b_i, r_{-i}), R(q_i(r_i), r_{-i})\}$.

*Non-Universal Mechanisms.* We are aware of several additional peer-prediction mechanisms that we do not consider further in this chapter because they are not universal in the sense of Definition 12. The Miller et al. [2005]; Zhang and Chen [2014] and Kong et al. [2016] mechanisms all derive the agents' posterior beliefs based on their signal reports (hence requiring knowledge of the distribution from which signals are drawn); they all then reward the agents based on how well the derived posterior belief predicts other agents' signal reports using proper scoring rules. The Jurca and Faltings [2009] mechanism requires knowledge of the prior distribution over signals to construct rewards that either penalize or eliminate symmetric, uninformative equilibria. The Bayesian Truth Serum (BTS) mechanism [Prelec, 2004] requires an infinite number of agents to guarantee the existence of the truthful equilibrium. While we do not consider this mechanism, we note that Prelec [2004] pioneered the idea of eliciting both signal and belief reports from each agent. This key idea was leveraged in much subsequent work to sustain the truthful equilibrium while not requiring the knowledge of the prior dis-

tributions of the signals to operate the mechanism [Witkowski and Parkes, 2012, 2013; Radanovic and Faltings, 2013, 2014; Riley, 2014].

## 7.3 Impossibility of Pareto-Dominant, Truthful Elicitation

In this section, we show that when agents have access to multiple signals about an object, Pareto-dominant truthful elicitation is impossible for any universal elicitation mechanism that computes agent rewards solely based on a profile of strategic agent reports (i.e., without any access to ground truth). The intuition is that without knowledge of the distributions from which the signals are drawn, the mechanism cannot distinguish the signal that it hopes to elicit from other, irrelevant signals. Thus, it cannot guarantee that the truthful equilibrium always yields the highest rewards to all agents.

We focus on universal elicitation mechanisms that compute agent rewards solely based on a profile of agent reports. Let $M$ denote such a mechanism. Let a *signal structure* be a collection of signals $\{s_i\}_{i=1}^n$ drawn from a joint distribution $F$, where each agent $i$ observes $s_i$. We say that a signal structure is $M$*-elicitable* if there exists an equilibrium of $M$ where every agent $i$ truthfully reports $s_i$. Let $\pi_i^F$ be agent $i$'s ex-ante expected reward in this equilibrium. A *multi-signal environment* is an environment in which the agents have access to at least two $M$-elicitable signal structures. We refer to the signal structure that the mechanism seeks to elicit as the *high-quality signal*, and all the others as *low-quality signals*.

**Theorem 1.** *For any universal elicitation mechanism, there exists a multi-signal environment in which the truthful equilibrium is not Pareto dominant.*

*Proof.* Let $F, F'$ be $M$-elicitable signal structures such that $\pi_i^F \geq \pi_i^{F'}$ for all $i$, with $\pi_i^F > \pi_i^{F'}$ for some $i$. If no such pair of signal structures exists, then the result follows directly, since the truthful equilibrium does not Pareto dominate an equilibrium where agents report a low-quality signal. Otherwise, consider a multi-signal environment where the high-quality signal is distributed according to $F'$,

and a low-quality signal is distributed according to $F$. The equilibrium in which agents reveal this low-quality signal Pareto dominates the truthful equilibrium in this environment. □

Now suppose that observing the high-quality signal is more costly to the agents than observing a low-quality signal. Concretely, assume that observing the high-quality signal has an additive cost of $c_i > 0$ for each agent $i$, and observing a low-quality signal has zero cost. Call this a *costly-observation multi-signal environment*. In this realistic environment, an even stronger result holds.

**Theorem 2.** *For any universal elicitation mechanism, there exists a costly-observation multi-signal environment in which the truthful equilibrium either does not exist or is Pareto dominated.*

*Proof.* Let $F, F'$ be $M$-elicitable signal structures such that $\pi_i^F \geq \pi_i^{F'}$ for all $i$. At least one such pair must exist, since every distribution has this relationship to itself. Fix a costly-observation multi-signal environment where the high-quality signal structure is jointly distributed according to $F'$, and a low-quality signal structure is jointly distributed according to $F$. If the mechanism has no truthful equilibrium in this environment, then we are done. Otherwise, each agent's expected utility in the truthful equilibrium is $\pi_i^{F'} - c_i < \pi_i^F$. Hence every agent prefers the equilibrium in which agents reveal this low-quality signal, and the truthful equilibrium is Pareto dominated. □

The essential insight of these results is that, in the presence of multiple elicitable signals, there is no way for a universal elicitation mechanism to be sure which signal it is eliciting. In particular, the truthful equilibrium is only Pareto dominant if the high-quality signal *happens* to be drawn from a distribution yielding higher reward than *every other* signal available to the agents. In costly-observation environments, the element of luck is even stronger. The truthful equilibrium is Pareto dominant only if the high-quality signal structure happens to yield sufficiently high reward to compensate for the cost of observing the signals.

One way for the mechanism designer to ensure that agents report the high-quality signal is to stochastically compare agents' reports to reports known to be correlated with that signal. In the next section, we introduce a class of mechanisms that takes this approach.

## 7.4 Combining Elicitation with Limited Access to Ground Truth

Elicitation mechanisms are designed for situations where it is infeasible for the mechanism designer to evaluate each object herself. However, in practice, it is virtually always possible, albeit costly, to obtain *trusted reports*, i.e. unbiased evaluations of a subset of the objects. In the peer grading setting, the instructor and teaching assistants can always mark some of the assignments. Similarly, review sites could in principle hire an expert to evaluate restaurants or hotels that its users have reviewed; and so on.

In this section, we define a class of mechanisms that take advantage of this limited access to ground truth to circumvent the result from Section 7.3.

**Definition 13** (spot-checking mechanism)**.** A *spot-checking mechanism* is a tuple $M = (p, y, z)$, where $p$ is the *spot check probability*; $y$ is a vector of functions $y_{ij}(r_{ij}, s^t_j)$ called the *spot check mechanism*; and $z$ is a vector of functions $z_{ij}(b, r)$ called the *unchecked mechanism*.

Let $\Delta(Q)$ be the set of all distributions over the elements of $Q$. Each agent $i$ makes a *signal report* $r_{ij} \in Q$, and a *belief report* $b_{ij} \in \Delta(Q)$ for each object $j \in J_i$. The signal report is the signal that $i$ claims to have observed, and the belief report represents $i$'s posterior belief over the signal reports of the other agents.

Agents may strategically choose whether or not to incur the cost of observing the high-quality signal, and having chosen which signal to observe, may report any function of either signal. Formally, let $G^h_i = \{g : Q \to Q\}$ be the set of all full-effort pure strategies, where an agent observes the high-quality signal—incurring observation cost $c^E$—and then reports a function $g(s^h_{ij})$ of the observed

value. Let $D_l$ be the domain of $s^l_{ij}$. Let $G^l_i = \{g : D^l \to Q\}$ be the set of all no-effort pure strategies, where an agent observes the low-quality signal—incurring no observation cost—and then reports a function $g^l(s^l_{ij})$ of the observed value. The set of pure strategies available to an agent is thus $G^h_i \cup G^l_i$. We assume that agents apply the same strategy to every object that they evaluate; however, we allow agents to play a mixed strategy by choosing the mapping stochastically.

With probability $p$, the mechanism will *spot check* an agent $i$'s report for a given object $j$. In this case, the mechanism obtains a *trusted report*—that is, a sample from the signal $s^t_j$. The agent is then rewarded according to the spot check mechanism, applied to the profile of signal reports and spot checked objects. With probability $1 - p$, the object is not spot checked, and the agent is rewarded according to the unchecked mechanism.

Thus, given a profile of signal reports $r \in \prod_{i \in N} Q^{J_i}$ and belief reports $b \in \prod_{i \in N} \Delta(Q)^{J_i}$, an agent $i$ receives a reward of $\pi_i = \sum_{j \in J_i} \pi_{ij}$, where

$$\pi_{ij} = \begin{cases} y_{ij}(r_i, s^t) & \text{if agent } i\text{'s report on object } j \text{ is spot checked,} \\ z_{ij}(b, r) & \text{otherwise.} \end{cases} \tag{7.1}$$

We assume that the mechanism designer has no value for the reward given to the agents. Instead, we seek only to minimize the probability of spot-checking required to make the truthful equilibrium either unique or Pareto dominant, since access to trusted reports is assumed to be costly.[3] This models situations where agents are rewarded by grades (as in peer grading), virtual points or badges (as in online reviews), or other artificial currencies. In this work we compare two approaches to using limited access to ground truth for elicitation. The first approach is to augment existing peer-prediction mechanisms with spot-checking:

**Definition 14** (spot-checking peer-prediction mechanism)**.** Let $z$ be a peer-prediction mechanism. Then any spot-checking mechanism that uses $z$ as its unchecked

[3] If access to trusted reports were not costly, then querying strategic agents rather than trusted reports on all the objects would be pointless.

mechanism is a *spot-checking peer-prediction mechanism*.

The second approach is to rely exclusively on ground truth access to incentivize truthful reporting:

**Definition 15** (peer-insensitive mechanism)**.** A *peer-insensitive mechanism* is a spot-checking mechanism in which the unchecked mechanism is a constant function. That is, $z_{ij}(b, r) = W$ for some constant $W > 0$.

## 7.5 When Does Peer-Prediction Help?

We compare the peer-insensitive mechanism with all universal spot-checking peer-prediction mechanisms. Theorem 3 states that, if a simple sufficient condition is satisfied, then compared to all universal spot-checking peer-prediction mechanisms, the peer-insensitive mechanism can achieve stronger incentive properties (dominant-strategy truthfulness versus Pareto dominance of truthful equilibrium) while requiring a smaller spot check probability.

We first define the $g^l$ strategy to be an agent's best no-effort strategy when a spot check is performed. What is special about this strategy is that, if an agent chooses to invest no effort, then this is his best strategy for any spot check probability $p \in [0, 1]$. Thus, the $g^l$ equilibrium is stable and the best equilibrium for all agents conditional on not investing effort.

**Definition 16.** Let $g^l = \arg\max_{g \in G} \mathbb{E}[y(g^l(s^l), s^t)]$ be an agent's best strategy when a spot check is performed and the agent invests no effort. Let the $g^l$ equilibrium be the equilibrium where every agent uses the $g^l$ strategy.

In Lemma 1, we analyze the peer-insensitive mechanism and derive an expression for the minimum spot check probability $p_{\text{ds}}$ at which the truthful strategy is a dominant strategy for the peer-insensitive mechanism. When the spot check probability is $p_{\text{ds}}$, any agent is indifferent between playing the $g^l$ strategy and investing effort and reporting truthfully.

**Lemma 1.** *The minimum spot check probability $p_{\text{ds}}$ at which the truthful strategy is dominant for the peer-insensitive mechanism satisfies the following equation.*

$$p_{\text{ds}}\,\mathbb{E}[y(s^h, s^t)] - c^E = p_{\text{ds}}\,\mathbb{E}[y(g^l(s^l), s^t)]. \tag{7.2}$$

*Proof.* See Section 7.7.1. □

Next, we consider any spot-checking peer-prediction mechanism. Our goal is to derive a lower bound for $p_{\text{Pareto}}$, the minimum spot check probability at which the truthful equilibrium is Pareto dominant.

For the truthful equilibrium to be Pareto dominant, it is necessary that the truthful equilibrium Pareto dominates the $g^l$ equilibrium. This can be achieved in two ways. If we can increase the spot check probability until $p_{\text{el}}$ at which the $g^l$ equilibrium is eliminated, then the truthful equilibrium trivially Pareto dominates the $g^l$ equilibrium. Otherwise, we can increase the spot check probability until $p_{\text{ex}}$ at which the truthful equilibrium Pareto dominates the $g^l$ equilibrium assuming that the $g^l$ equilibrium exists when $p = p_{\text{ex}}$. Thus, $\min(p_{\text{el}}, p_{\text{ex}})$ is the minimum spot check probability at which the truthful equilibrium Pareto dominates the $g^l$ equilibrium, and it is also a lower bound for $p_{\text{Pareto}}$.

In Lemma 2, we derive an expression for $p_{\text{el}}$ and show that it is greater than or equal to $p_{\text{ds}}$ under certain assumptions. Intuitively, in order to eliminate the $g^l$ equilibrium, we need to increase the spot check probability enough such that an agent is persuaded to play his best strategy with full effort rather than playing the $g^l$ strategy. On one hand, the agent incurs a cost deviating from the $g^l$ equilibrium when all other agents follow it. On the other hand, the agent's best strategy with full effort gives him no greater spot check reward than the truthful strategy. The combined effect means that it is more costly to persuade an agent to deviate from the $g^l$ equilibrium than to motivate a single agent to report truthfully.

The sufficient conditions characterized in Lemmas 2 and 3, and Theorem 3 are required to hold when $c^E = 0$. However, if this condition is satisfied when $c^E = 0$, then the consequents of these lemmas and theorems hold in any setting

with a positive cost of effort $c^E \geq 0$ as well. Moreover, we will show that these sufficient conditions are satisfied by all universal peer-prediction mechanisms of which we are aware in the literature.

**Lemma 2.** *For any spot-checking peer-prediction mechanism, if the $g^l$ equilibrium exists for $c^E = 0$ and $p = 0$, then $p_{\text{el}} \geq p_{\text{ds}}$ for all settings with positive effort cost $c^E \geq 0$.*

*Proof.* See Section 7.7.2. □

In Lemma 3, we show that $p_{\text{ex}}$ is greater than or equal to $p_{\text{ds}}$ under certain assumptions. The intuition is that, when no spot check is performed, the $g^l$ equilibrium Pareto dominates the truthful equilibrium. Thus, assuming that the $g^l$ equilibrium exists, it is more costly (in terms of increasing spot check probability) to make the truthful equilibrium Pareto dominate the $g^l$ equilibrium than to motivate a single agent to report truthfully.

**Lemma 3.** *For any spot-checking peer-prediction mechanism, if the $g^l$ equilibrium exists and Pareto dominates the truthful equilibrium for $c^E = 0$ and $p = 0$, then $p_{\text{ex}} \geq p_{\text{ds}}$ for all settings with positive effort cost $c^E \geq 0$.*

*Proof.* See Section 7.7.3. □

If the conditions in Lemmas 2 and 3 are satisfied, it is clear that $p_{\text{Pareto}} \geq p_{\text{ds}}$ because $\min(p_{\text{el}}, p_{\text{ex}})$, which lower bounds $p_{\text{Pareto}}$, is already greater than or equal to $p_{\text{ds}}$. Thus, a sufficient condition for $p_{\text{Pareto}} \geq p_{\text{ds}}$ is simply all conditions in the two lemmas, as shown in Theorem 3.

**Theorem 3** (Sufficient condition for Pareto comparison)**.** *For any spot-checking peer-prediction mechanism, if the $g^l$ equilibrium exists and Pareto dominates the truthful equilibrium for $c^E = 0$ and $p = 0$, then $p_{\text{Pareto}} \geq p_{\text{ds}}$ for all settings with positive effort cost $c^E \geq 0$.*

*Proof.* See Section 7.7.4. □

We now show that, under very natural conditions, *every* universal peer-prediction mechanism of which we are aware in the literature satisfies the conditions of Theorem 3; hence, in this setting, the peer-insensitive spot-checking mechanism requires less ground truth access than any spot-checking peer-prediction mechanism.

For mathematical convenience, we assume that the low-quality signal $s^l$ is drawn from a uniform distribution over $Q$. This is essentially without loss of generality, since in any setting where the agents see a description of the object as well as their evaluation, a distribution of this form can be obtained by, e.g., hashing the description. More realistically, objects may have names that are approximately uniformly distributed.

We fix the spot check mechanism as in Equation (7.3), using a form inspired by Dasgupta and Ghosh [2013]. Let $J^t$ be the set of objects that was spot-checked. Let $i$ be an agent whose report $r_{ij}$ on object $j \in J_i$ has been spot checked. Let $j' \in J_i$ be an object that $j$ evaluated, chosen uniformly at random, and let $j'' \in J^t \backslash J_i$ be a spot-checked object, also chosen uniformly at random.[4] Then agent $i$'s reward for object $j$ is

$$y_{ij}(r_i, s^t) = \mathbb{1}_{r_{ij}=s^t_j} - \mathbb{1}_{r_{ij'}=s^t_{j''}}. \tag{7.3}$$

**Lemma 4.** *For the spot check reward function in Equation* (7.3)*, an agent's best strategy conditional on not investing effort is always to report the low-quality signal* $s^l$.

*Proof.* See Section 7.7.5. □

**Corollary 1.** *For spot-checking peer-prediction mechanisms based on Faltings et al. [2012]; Witkowski et al. [2013]; Dasgupta and Ghosh [2013]; Waggoner and Chen [2014]; Kamble et al. [2015]; Radanovic and Faltings [2015] and*

[4]Note that in Dasgupta and Ghosh [2013], it is important for strategic reasons that object $j'$ has not been evaluated by the opposing agent; this is not important in our setting, since the trusted reports are nonstrategic.

*Shnayder et al. [2016], the minimum spot check probability $p_{\text{Pareto}}$ that guarantees Pareto dominance of the truthful equilibrium is greater than or equal to the minimum spot check probability $p_{\text{ds}}$ at which the truthful strategy is a dominant strategy for the peer-insensitive mechanism.*

*Proof.* See Section 7.7.6. □

**Corollary 2.** *For spot-checking peer-prediction mechanisms based on Witkowski and Parkes [2012, 2013]; Radanovic and Faltings [2013, 2014] and Riley [2014], if the peer-prediction mechanism uses a symmetric proper scoring rule, then the minimum spot check probability $p_{\text{Pareto}}$ that guarantees Pareto dominance of the truthful equilibrium is greater than or equal to the minimum spot check probability $p_{\text{ds}}$ at which the truthful strategy is a dominant strategy for the peer-insensitive mechanism.*

*Proof.* See Section 7.7.7. □

## 7.6 Conclusions

We consider the problem of using limited access to noisy but unbiased ground truth to incentivize agents to invest costly effort in evaluating and truthfully reporting the quality of some object of interest. Absent such spot-checking, peer-prediction mechanisms already guarantee the existence of a truthful equilibrium that induces both effort and honesty from the agents. However, this truthful equilibrium may be less attractive to the agents than other, uninformative equilibria.

Some mechanisms in the literature have been carefully designed to ensure that the truthful equilibrium is the most attractive equilibrium to the agents (i.e., Pareto dominates all other equilibria). However, these mechanisms rely crucially on the unrealistic assumption that agents' only means of correlating are via the signals that the mechanism aims to elicit. We show that under the more realistic assumption that agents have access to more than one signal, no universal peer-prediction mechanism has a Pareto dominant truthful equilibrium in all elicitable settings.

In contrast, we present a simpler peer-insensitive mechanism that provides incentives for effort and honesty only by checking the agents' reports against ground truth. While one might have expected that peer-prediction would require less frequent access to ground truth to achieve stronger incentive properties than the peer-insensitive mechanism, we proved the opposite for all universal spot-checking peer-prediction mechanisms.

This surprising finding is intuitive in retrospect. Peer-prediction mechanisms can only motivate agents to behave in a certain way as a group. An agent has a strong incentive to be truthful if all other agents are truthful; conversely, when all other agents coordinate on investing no effort, the agent again has a strong incentive to coordinate with the group. Peer-prediction mechanisms thus need to provide a strong enough incentive for agents to deviate from the most attractive uninformative equilibrium in the worst case, whereas the peer-insensitive mechanism only needs to motivate effort and honesty in an effectively single-agent setting.

Many exciting future directions remain to be explored. For example, we assumed that the principal does not care about the total amount of the artificial currency rewarded to the agents. One possible direction would consider a setting in which the principal seeks to minimize both spot checks and the agents' rewards. Also, in our analysis, we assumed that the spot check probability does not depend on the agents' reports. Conditioning the spot check probability on the agents' reports might allow the mechanism to more efficiently detect and punish uninformative equilibria.

## 7.7 Proofs

In this section we present proofs that were deferred from earlier in the chapter. Readers who are less interested in the details of the proofs can safely skip to the next chapter.

### 7.7.1 Proof of Lemma 1

**Lemma 1.** *The minimum spot check probability $p_{\text{ds}}$ at which the truthful strategy is dominant for the peer-insensitive mechanism satisfies the following equation.*

$$p_{\text{ds}}\,\mathbb{E}[y(s^h, s^t)] - c^E = p_{\text{ds}}\,\mathbb{E}[y(g^l(s^l), s^t)]. \tag{7.4}$$

*Proof.* Consider the peer insensitive mechanism with a fixed spot check probability $p \geq 0$. When an agent uses the truthful strategy, his expected utility is

$$p\,\mathbb{E}[y(s^h, s^t)] + (1-p)\,W - c^E. \tag{7.5}$$

When an agent invests no effort, his best strategy is $g^l$. His expected utility from playing the $g^l$ strategy is

$$p\,\mathbb{E}[y(g^l(s^l), s^t)] + (1-p)\,W. \tag{7.6}$$

When $p = p_{\text{ds}}$, it must be that an agent's expected utilities in the above two expressions (7.5) and (7.6) are the same.

$$\begin{gathered} p_{\text{ds}}\,\mathbb{E}[y(s^h, s^t)] + (1-p_{\text{ds}})\,W - c^E = p_{\text{ds}}\,\mathbb{E}[y(g^l(s^l), s^t)] + (1-p_{\text{ds}})\,W \\ p_{\text{ds}}\,\mathbb{E}[y(s^h, s^t)] - c^E = p_{\text{ds}}\,\mathbb{E}[y(g^l(s^l), s^t)]. \end{gathered}$$

□

### 7.7.2 Proof of Lemma 2

**Lemma 2.** *For any spot-checking peer-prediction mechanism, if the $g^l$ equilibrium exists for $c^E = 0$ and $p = 0$, then $p_{\text{el}} \geq p_{\text{ds}}$ for all settings with positive effort cost $c^E \geq 0$.*

*Proof.* Recall that $p_{\text{el}}$ is the minimum spot check probability at which the $g^l$ equilibrium is eliminated. We first derive an expression for $p_{\text{el}}$.

We consider a spot checking peer prediction mechanism. By our assumption, the $g^l$ equilibrium exists when $c^E = 0$ and the spot check probability is $0$.

Assume that all other agents play the $g^l$ strategy and analyze agent $i$'s best response. First, we note that, if agent $i$ invests no effort, then agent $i$'s best strategy is the $g^l$ strategy for any spot check probability. (To maximize his spot check reward $y$, he should play the $g^l$ strategy by the definition of the $g^l$ strategy. To maximize his non spot check reward, his best strategy is also the $g^l$ strategy because the $g^l$ equilibrium exists at $p = 0$. ) Thus, to eliminate the $g^l$ equilibrium, we need to increase the spot check probability until agent $i$ prefers to play his best strategy conditional on investing full effort.

Consider a fixed spot check probability $p$ and suppose that the $g^l$ equilibrium exists at this spot check probability. Suppose that all other agents play the $g^l$ strategy.

If agent $i$ does not invest effort, his best response is to also play the $g^l$ strategy and his expected utility is

$$p\,\mathbb{E}[y(g^l(s^l), s^t)] + (1-p)\,\mathbb{E}[z(g^l(s^l), g^l(s^l))]. \tag{7.7}$$

If agent $i$ invests full effort, let $g^{\text{br}}$ denote agent $i$'s best response and his expected utility by playing this best response is

$$p\,\mathbb{E}[y(g^{\text{br}}(s^h), s^t)] + (1-p)\,\mathbb{E}[z(g^{\text{br}}(s^h), g^l(s^l))] - c^E. \tag{7.8}$$

By definition of $p_{\text{el}}$, when $p = p_{\text{el}}$, an agent's expected utility in the above two

expressions (7.7) and (7.8) are the same. Thus $p_{\text{el}}$ must satisfy

$$\begin{aligned}
&p_{\text{el}}\,\mathbb{E}[y(g^{\text{br}}(s^h), s^t)] + (1-p_{\text{el}})\,\mathbb{E}[z(g^{\text{br}}(s^h), g^l(s^l))] - c^E \\
&\qquad = p_{\text{el}}\,\mathbb{E}[y(g^l(s^l), s^t)] + (1-p_{\text{el}})\,\mathbb{E}[z(g^l(s^l), g^l(s^l))] \\
&p_{\text{el}}\,\mathbb{E}[y(g^{\text{br}}(s^h), s^t)] + (1-p_{\text{el}})\,(\mathbb{E}[z(g^{\text{br}}(s^h), g^l(s^l))] - \mathbb{E}[z(g^l(s^l), g^l(s^l))]) - c^E \\
&\qquad = p_{\text{el}}\,\mathbb{E}[y(g^l(s^l), s^t)]. \qquad (7.9)
\end{aligned}$$

Next, we would like to show that $p_{\text{el}} \geq p_{\text{ds}}$.

Since the $g^l$ equilibrium exists when $c^E = 0$ and $p = 0$, it follows from the definition of equilibrium that

$$\mathbb{E}[z(g^{\text{br}}(s^h), g^l(s^l))] \leq \mathbb{E}[z(g^l(s^l), g^l(s^l))]. \qquad (7.10)$$

Taking $p_{\text{el}}$ and substituting into the LHS of (7.2) (definition of $p_{\text{ds}}$), in a setting with arbitrary positive $c^E \geq 0$, we have

$$\begin{aligned}
&p_{\text{el}}\,\mathbb{E}[y(s^h, s^t)] - c^E \\
&\geq p_{\text{el}}\,\mathbb{E}[y(s^h, s^t)] + (1-p_{\text{el}})\,(\mathbb{E}[z(g^{\text{br}}(s^h), g^l(s^l))] - \mathbb{E}[z(g^l(s^l), g^l(s^l))]) - c^E \qquad (7.11) \\
&> p_{\text{el}}\,\mathbb{E}[y(g^{\text{br}}(s^h), s^t)] + (1-p_{\text{el}})\,(\mathbb{E}[z(g^{\text{br}}(s^h), g^l(s^l))] - \mathbb{E}[z(g^l(s^l), g^l(s^l))]) - c^E \qquad (7.12) \\
&= p_{\text{el}}\,\mathbb{E}[y(g^l(s^l), s^t)]. \qquad (7.13)
\end{aligned}$$

Inequality (7.11) holds due to Equation (7.10). Inequality (7.12) holds due to the truthfulness of spot checks: reporting the high-quality signal maximizes the spot check reward. Equation (7.13) follows from Equation (7.9).

Thus, if we substitute $p_{\text{el}}$ into Equation (7.2), then the resulting LHS is greater than the RHS. By definition of $p_{\text{ds}}$, it is the minimum spot check probability for which the LHS of (7.2) is greater than its RHS. Thus, it must be that $p_{\text{el}} \geq p_{\text{ds}}$. □

### 7.7.3 Proof of Lemma 3

**Lemma 3.** *For any spot-checking peer-prediction mechanism, if the $g^l$ equilibrium exists and Pareto dominates the truthful equilibrium for $c^E = 0$ and $p = 0$, then $p_{\text{ex}} \geq p_{\text{ds}}$ for all settings with positive effort cost $c^E \geq 0$.*

*Proof.* Recall that $p_{\text{ex}}$ is the minimum spot check probability at which the $g^l$ equilibrium Pareto dominates the truthful equilibrium while the $g^l$ equilibrium exists at $p = p_{\text{ex}}$. We first derive an expression for $p_{\text{ex}}$.

We consider a spot checking peer prediction mechanism. By our assumption, the $g^l$ equilibrium exists and Pareto dominates the truthful equilibrium when $c^E = 0$ and $p = 0$.

Consider a fixed spot check probability $p \geq 0$. Assume that the $g^l$ equilibrium exists at this spot check probability. At the truthful equilibrium, an agent's expected utility is

$$p\,\mathbb{E}[y(s^h, s^t)] + (1-p)\,\mathbb{E}[z(s^h, s^h)] - c^E. \tag{7.14}$$

At the $g^l$ equilibrium, an agent's expected utility is

$$p\,\mathbb{E}[y(g^l(s^l), s^t)] + (1-p)\,\mathbb{E}[z(g^l(s^l), g^l(s^l))]. \tag{7.15}$$

When $p = p_{\text{ex}}$, it must be that an agent's expected utility in the above two expressions (7.14) and (7.15) are the same. Thus $p_{\text{ex}}$ must satisfy

$$\begin{aligned}
&p_{\text{ex}}\,\mathbb{E}[y(s^h, s^t)] + (1-p_{\text{ex}})\,\mathbb{E}[z(s^h, s^h)] - c^E \\
&\qquad = p_{\text{ex}}\,\mathbb{E}[y(g^l(s^l), s^t)] + (1-p_{\text{ex}})\,\mathbb{E}[z(g^l(s^l), g^l(s^l))] \\
&p_{\text{ex}}\,\mathbb{E}[y(s^h, s^t)] + (1-p_{\text{ex}})\,\Big(\mathbb{E}[z(s^h, s^h)] - \mathbb{E}[z(g^l(s^l), g^l(s^l))]\Big) - c^E \\
&\qquad = p_{\text{ex}}\,\mathbb{E}[y(g^l(s^l), s^t)].
\end{aligned} \tag{7.16}$$

Next, we would like to show that $p_{\text{ex}} \geq p_{\text{ds}}$.

Since the $g^l$ equilibrium exists and Pareto dominates the truthful equilibrium

for $c^E = 0$ and $p = 0$, it follows from the definition of Pareto dominance that

$$\mathbb{E}[z(s^h, s^h)] \leq \mathbb{E}[z(g^l(s^l), g^l(s^l))]. \tag{7.17}$$

Taking $p_{\text{ex}}$ and substituting it into the LHS of Equation (7.2) (definition of $p_{\text{ds}}$), in a setting with arbitrary positive $c^E \geq 0$, we have

$$\begin{aligned} & p_{\text{ex}}\, \mathbb{E}[y(s^h, s^t)] - c^E \\ & \geq p_{\text{ex}}\, \mathbb{E}[y(s^h, s^t)] + (1 - p_{\text{ex}}) \left(\mathbb{E}[z(s^h, s^h)] - \mathbb{E}[z(g^l(s^l), g^l(s^l))]\right) - c^E \end{aligned} \tag{7.18}$$

$$= p_{\text{ex}}\, \mathbb{E}[y(g^l(s^l), s^t)] \tag{7.19}$$

Equation (7.18) follows from Equation (7.17). Equation (7.19) follows from Equation (7.16).

Thus, if we substitute $p_{\text{ex}}$ into Equation (7.2), then the resulting LHS is weakly greater than the RHS. By definition of $p_{\text{ds}}$, it is the minimum spot check probability for which the LHS of (7.2) is greater than its RHS. Thus, it must be that $p_{\text{ex}} \geq p_{\text{ds}}$. □

### 7.7.4 Proof of Theorem 3

**Theorem 3** (Sufficient condition for Pareto comparison)**.** *For any spot-checking peer-prediction mechanism, if the $g^l$ equilibrium exists and Pareto dominates the truthful equilibrium for $c^E = 0$ and $p = 0$, then $p_{\text{Pareto}} \geq p_{\text{ds}}$ for all settings with positive effort cost $c^E \geq 0$.*

*Proof.* Consider any spot checking peer prediction mechanism.

For the truthful equilibrium to be Pareto dominant, it is necessary that either the $g^l$ equilibrium is eliminated or the truthful equilibrium Pareto dominates the $g^l$ equilibrium while the $g^l$ equilibrium exists. $p_{\text{el}}$ is the minimum spot check probability at which the $g^l$ equilibrium is eliminated. $p_{\text{ex}}$ is the minimum spot check probability at which the truthful equilibrium Pareto dominates the $g^l$ equilibrium

while the $g^l$ equilibrium exists at $p = p_{\text{ex}}$. Thus, the minimum of $p_{\text{el}}$ and $p_{\text{ex}}$ is a lower bound of $p_{\text{pareto}}$. Formally

$$p_{\text{pareto}} \geq \min(p_{\text{el}}, p_{\text{ex}}). \tag{7.20}$$

By assumption, the $g^l$ equilibrium exists when $p = 0$. By Lemma 2, we have

$$p_{\text{el}} \geq p_{\text{ds}}. \tag{7.21}$$

By assumption, the $g^l$ equilibrium exists and Pareto dominates the truthful equilibrium when $p = 0$. By Lemma 3, we have

$$p_{\text{ex}} \geq p_{\text{ds}}. \tag{7.22}$$

By Equations (7.20), (7.21) and (7.22), we have

$$\begin{aligned} p_{\text{pareto}} &\geq \min(p_{\text{el}}, p_{\text{ex}}) \\ &\geq \min(p_{\text{ds}}, p_{\text{ex}}) \\ &\geq \min(p_{\text{ds}}, p_{\text{ds}}) \\ &= p_{\text{ds}}. \end{aligned}$$

□

### 7.7.5 Proof of Lemma 4

**Lemma 4.** *For the spot check reward function in Equation* (7.3)*, an agent's best strategy conditional on not investing effort is always to report the low-quality signal* $s^l$.

*Proof.* Consider the spot check reward mechanism in Equation (7.3).

If an agent invests no effort, his expected spot check reward is:

$$\sum_{s \in Q} \Pr(r = s) \left( \Pr(s^t = s | r = s) - \sum_{s' \in Q} \Pr(s^t = s') \Pr(r = s') \right)$$
$$= \sum_{s \in Q} \Pr(s^t = s, r = s) - \sum_{s' \in Q} \Pr(s^t = s') \Pr(r = s')$$

If the agent always makes a fixed report $r$, then the TA's signal $s^t$ and the agent's report $r$ are independent random variables, i.e.

$$\Pr(s^t = s, r = s) = \Pr(s^t = s) \Pr(r = s),$$

for any $s \in Q$. Thus the agent's expected reward must be zero.

$$\sum_{s \in Q} \Pr(s^t = s, r = s) - \sum_{s' \in Q} \Pr(s^t = s') \Pr(r = s')$$
$$= \sum_{s \in Q} \Pr(s^t = s) \Pr(r = s) - \sum_{s' \in Q} \Pr(s^t = s') \Pr(r = s')$$
$$= 0$$

If the agent truthfully reports the low-quality signal $s^l$, then the agent's expected reward is:

$$\sum_{s \in Q} \Pr(r = s) \left( \Pr(s^t = s | r = s) - \sum_{s' \in Q} \Pr(s^t = s') \Pr(r = s') \right)$$
$$= \sum_{s \in Q} \Pr(r = s) \left( \Pr(s^t = s | r = s) - \Pr(s^t = s') \right)$$
$$\geq 0$$

Thus the agent's expected spot check reward is maximized when he reports the low-quality signal $s^l$. □

### 7.7.6 Proof of Corollary 1

**Corollary 1.** *For spot-checking peer-prediction mechanisms based on Faltings et al. [2012]; Witkowski et al. [2013]; Dasgupta and Ghosh [2013]; Waggoner and Chen [2014]; Kamble et al. [2015]; Radanovic and Faltings [2015] and Shnayder et al. [2016], the minimum spot check probability $p_{\text{Pareto}}$ that guarantees Pareto dominance of the truthful equilibrium is greater than or equal to the minimum spot check probability $p_{\text{ds}}$ at which the truthful strategy is a dominant strategy for the peer-insensitive mechanism.*

*Proof.* By Lemma 4, for any spot checking peer prediction mechanism, the $g^l$ strategy is to always report the low-quality signal $s^l$.

To verify that the conditions of Theorem 3 are satisfied, it suffices to verify that when $p = 0$, the $s^l$ equilibrium of the peer prediction mechanism exists and Pareto dominates the truthful equilibrium. We verify these two conditions for all of the listed peer prediction mechanisms below.

We first consider output agreement peer prediction mechanisms.

*The Standard Output Agreement Mechanism [Witkowski et al., 2013; Waggoner and Chen, 2014]* When $c^E = 0$ and $p = 0$, the $s^l$ equilibrium exists. (If all other agents except $i$ report $s^l$, then agent $i$'s best response is to also report $s^l$ in order to perfectly agree with other reports.)

When $c^E = 0$ and $p = 0$, at the $s^l$ equilibrium, every agent's expected utility is $1$ because their reports always perfectly agree.

When $c^E = 0$ and $p = 0$, at the truthful equilibrium, an agent's expected utility is

$$\sum_{s^h \in Q} \Pr(s^h)\Pr(s^h|s^h) < \sum_{s^h \in Q} \Pr(s^h) = 1,$$

where the inequality is due to the fact that the high-quality signals are noisy. That is, for every realization $s^h$ of the high-quality signal, $\Pr(s^h|s^h) \leq 1$ and there exists one realization $s^h$ of the high-quality signal such that $\Pr(s^h|s^h) < 1$. Thus, the $s^l$ equilibrium Pareto dominates the truthful equilibrium when $c^E = 0$ and

$p = 0$. The conditions of Theorem 3 are therefore satisfied, and hence $p_{\text{Pareto}} \geq p_{\text{ds}}$ for all settings with positive effort cost $c^E \geq 0$.

*Peer Truth Serum [Faltings et al., 2012]* When $c^E = 0$ and $p = 0$, the $s^l$ equilibrium exists. (If all other agents except $i$ report $s^l$, then agent $i$'s best response is to also report $s^l$.)

When $c^E = 0$ and $p = 0$, at the $s^l$ equilibrium, everyone reports $s^l$ and the empirical frequency of $s^l$ reports is 1 ($F(s^l) = 1$). Thus, every agent's expected utility is

$$\alpha + \beta \frac{1}{F(s^l)} = \alpha + \beta.$$

When $c^E = 0$ and $p = 0$, at the truthful equilibrium, if agent receives the high-quality signal $s^h$ for an object, then he expects the empirical frequency of this signal to be $\Pr(s^h|s^h)$. Thus, at this equilibrium, an agent's expected utility is

$$\alpha + \beta \sum_{s^h \in Q} \Pr(s^h) \Pr(s^h|s^h) \frac{1}{\Pr(s^h|s^h)} = \alpha + \beta.$$

Thus, the $s^l$ equilibrium (weakly) Pareto dominates the truthful equilibrium when $c^E = 0$ and $p = 0$. The conditions of Theorem 3 are therefore satisfied, and hence $p_{\text{Pareto}} \geq p_{\text{ds}}$ for all settings with positive effort cost $c^E \geq 0$.

Next, we consider multi-object peer prediction mechanisms.

*Dasgupta and Ghosh [2013]; Shnayder et al. [2016]* When $c^E = 0$ and $p = 0$, the $s^l$ equilibrium exists. (If all other agents always report the low-quality signal $s^l$ for every object, then agent $i$'s best response is also to report $s^l$ in order to maximize the probability of his report agreeing with other agents' reports for the same object.)

When $p = 0$, at the $s^l$ equilibrium, an agent's expected utility is

$$\sum_{s^l \in Q} \Pr(s^l)\Pr(s^l|s^l) - \sum_{s^l \in Q} \Pr(s^l)\Pr(s^l) = \sum_{s^l \in Q} \Pr(s^l) - \sum_{s^l \in Q} \Pr(s^l)\Pr(s^l)$$
$$= 1 - \sum_{s^l \in Q} \frac{1}{|Q|^2} = 1 - \frac{1}{|Q|},$$

where the first equality was due to the fact that the low-quality signal $s^l$ is noiseless ($\Pr(s^l|s^l) = 1$) and the second equality was due to the fact that $s^l$ is drawn from a uniform distribution ($\Pr(s^l) = \frac{1}{|Q|}$).

When $c^E = 0$ and $p = 0$, at the truthful equilibrium, an agent's expected utility is

$$\sum_{s^h \in Q} \Pr(s^h)\Pr(s^h|s^h) - \sum_{s^h \in Q} \Pr(s^h)\Pr(s^h) < \sum_{s^h \in Q} \Pr(s^h) - \sum_{s^h \in Q} \Pr(s^h)^2$$
$$= 1 - \sum_{s^h \in Q} \Pr(s^h)^2 \leq 1 - \frac{1}{|Q|},$$

where the first inequality was due to the fact that the high-quality signal is noisy. That is, for every realization $s^h$ of the high-quality signal, $\Pr(s^h|s^h) \leq 1$ and there exists one realization $s^h$ of the high-quality signal such that $\Pr(s^h|s^h) < 1$. Thus, the $s^l$ equilibrium Pareto dominates the truthful equilibrium when $c^E = 0$ and $p = 0$. The conditions of Theorem 3 are therefore satisfied, and hence $p_{\text{Pareto}} \geq p_{\text{ds}}$ for all settings with positive effort cost $c^E \geq 0$.

*Kamble et al. [2015]* When $c^E = 0$ and $p = 0$, the $s^l$ equilibrium exists. (If all other agents always report $s^l$, an agent's best response is also to report $s^l$ because doing so maximizes the probability of his report agreeing with other agents' reports for the same object.)

When $c^E = 0$ and $p = 0$, at the $s^l$ equilibrium, an agent's expected utility is

$$\sum_{s^l \in Q} \Pr(s^l)\Pr(s^l|s^l) \lim_{N\to\infty} r(s^l) = \sum_{s^l \in Q} \Pr(s^l) \frac{K}{\sqrt{\Pr(s^l, s^l)}} = K \sum_{s^l \in Q} \frac{\Pr(s^l)}{\sqrt{\Pr(s^l)}}$$
$$= K \sum_{s^l \in Q} \sqrt{\Pr(s^l)} = K \sum_{s_l \in Q} \sqrt{\frac{1}{|Q|}},$$

where the first two equalities were due to the fact that the low-quality signal $s^l$ is noiseless ($\Pr(s^l|s^l) = \Pr(s^l)$), and the final equality was due to the fact that the low-quality signal $s^l$ is drawn from a uniform distribution.

When $c^E = 0$ and $p = 0$, at the truthful equilibrium, an agent's expected utility is

$$\sum_{s^h \in Q} \Pr(s^h)\Pr(s^h|s^h) \lim_{N\to\infty} r(s^h) = \sum_{s^h \in Q} \Pr(s^h, s^h) \frac{K}{\sqrt{\Pr(s^h, s^h)}}$$
$$= K \sum_{s^h \in Q} \sqrt{\Pr(s^h, s^h)} < K \sum_{s^h \in Q} \sqrt{\Pr(s^h)} \leq K \sum_{s^h \in Q} \sqrt{\frac{1}{|Q|}},$$

where the first inequality was due to the fact that the high-quality signal $s^h$ is noisy. That is, for every realization $s^h$ of the high-quality signal, $\Pr(s^h|s^h) \leq 1$ and there exists one realization $s^h$ of the high-quality signal such that $\Pr(s^h|s^h) < 1$. Thus, the $s^l$ equilibrium Pareto dominates the truthful equilibrium when $c^E = 0$ and $p = 0$. The conditions of Theorem 3 are therefore satisfied, and hence $p_{\text{Pareto}} \geq p_{\text{ds}}$ for all settings with positive effort cost $c^E \geq 0$.

*Radanovic and Faltings [2015]* When $c^E = 0$ and $p = 0$, the $s^l$ equilibrium exists. (If all other agents always report $s^l$ for every object, then any sample taken will not be "double mixed".[5] Thus, an agent's expected utility is zero regardless of his strategy. In particular also reporting $s^l$ for every object is a best response.)

[5] A sample is double mixed if every possible value appears at least twice. This mechanism behaves differently depending on whether or not it collects a double mixed sample of reports from the agents.

When $c^E = 0$ and $p = 0$, at the $s^l$ equilibrium, it must be that $r_{i''j'} = r_{i'j}$ and $r_{i''j'} = r_{i'''j''} = r_{ij}$. An agent's expected utility at the $s^l$ equilibrium is:

$$\frac{1}{2} + \mathbb{1}_{r_{i''j'}=r_{i'j}} - \frac{1}{2}\sum_{s\in Q} \mathbb{1}_{r_{i''j'}=s}\mathbb{1}_{r_{i'''j''}=s} = \frac{1}{2} + 1 - \frac{1}{2} * 1 = 1.$$

Let $\pi(\Sigma)$ be the probability that the sample $\Sigma$ is double mixed. When $c^E = 0$ and $p = 0$, at the truthful equilibrium, an agent's expected utility is:

$$\pi(\Sigma)\left(\frac{1}{2} + \Pr(r_{i''j'}|r_{ij}) - \frac{1}{2}\sum_{s\in Q}\Pr(s|r_{ij})^2\right) \leq \frac{1}{2} + \Pr(r_{i''j'}|r_{ij}) - \frac{1}{2}\sum_{s\in Q}\Pr(s|r_{ij})^2$$
$$\leq \frac{1}{2} + 1 - \frac{1}{2} * 1 = 1,$$

where the first inequality is due to the fact that $\pi(\Sigma) \leq 1$ and the second inequality was due to the fact that the agent's expected utility is maximized when $\Pr(r_{i''j'}|r_{ij}) = 1$. Thus, the $s^l$ equilibrium Pareto dominates the truthful equilibrium when $c^E = 0$ and $p = 0$. The conditions of Theorem 3 are therefore satisfied, and hence $p_{\text{Pareto}} \geq p_{\text{ds}}$ for all settings with positive effort cost $c^E \geq 0$.

□

### 7.7.7 Proof of Corollary 2

**Corollary 2.** *For spot-checking peer-prediction mechanisms based on Witkowski and Parkes [2012, 2013]; Radanovic and Faltings [2013, 2014] and Riley [2014], if the peer-prediction mechanism uses a symmetric proper scoring rule, then the minimum spot check probability $p_{\text{Pareto}}$ that guarantees Pareto dominance of the truthful equilibrium is greater than or equal to the minimum spot check probability $p_{\text{ds}}$ at which the truthful strategy is a dominant strategy for the peer-insensitive mechanism.*

*Proof.* By Lemma 4, for any spot checking peer prediction mechanism, the $g^l$ strategy is to always report the low-quality signal $s^l$.

To verify that the conditions of Theorem 3 are satisfied, it suffices to verify that when $p = 0$, the $s^l$ equilibrium of the peer prediction mechanism exists and Pareto dominates the truthful equilibrium. We verify these two conditions for all of the listed peer prediction mechanisms below.

Let $b_s$ denote a belief report which predicts that signal $s$ is observed with probability 1, i.e. $\Pr(s) = 1$ and $\Pr(s') = 0, \forall s' \in Q, s' \neq s$. Let the $s^l$ equilibrium denote the equilibrium where every agent's signal report is $s^l$ and belief report is $b_{s^l}$.

For mathematical convenience, we assume that the scoring rule is *symmetric* [Gneiting and Raftery, 2007]. That it, the reward for reporting a signal that is predicted with probability 1 is the same regardless of the signal's identity:

$$R(b_s, s) = R(b_{s'}, s'), \forall s \neq s'.$$

This is a very mild condition that is satisfied by all standard scoring rules that compute rewards based purely on the predicted probabilities and the outcome, including the quadratic scoring rule and the log scoring rule.

For symmetric scoring rules, when $p = 0$, an agent's expected score is maximized by predicting $b_s$ when $s$ is observed for any signal $s \in Q$.

*Binary Robust BTS [Witkowski and Parkes, 2012, 2013]* When $c^E = 0$ and $p = 0$, the $s^l$ equilibrium exists. (If all other agents report $s^l$ and $b_{s^l}$, then the best belief report for agent $i$ is $b_{s^l}$. Moreover the best signal report for agent $i$ is $s^l$ which leads to a shadowed belief report of $b_{s^l}$.)

When $c^E = 0$ and $p = 0$, at the $s^l$ equilibrium, an agent's expected utility is $R(b_{s^l}, s^l) + R(s_{s^l}, s^l)$. This is the maximum possible expected utility that an agent can achieve because the proper scoring rule $R$ is symmetric. Therefore, it must be greater than or equal to the agent's expected utility at the truthful equilibrium when $c^E = 0$ and $p = 0$.

*Multi-valued Robust BTS [Radanovic and Faltings, 2013]* When $c^E = 0$ and $p = 0$, the $s^l$ equilibrium exists. (If all other agents report $s^l$ and $b_{s^l}$, then the best belief report for agent $i$ is $b_{s^l}$. Moreover, the best signal report for agent $i$ is $s^l$ which maximizes the probability of his signal report agreeing with other agents' signal reports.)

When $c^E = 0$ and $p = 0$, at the $s^l$ equilibrium, an agent's expected utility is

$$\sum_{s^l} \Pr(s^l)\Pr(s^l|s^l) + R(b_{s^l}, s^l) = \sum_{s^l} \Pr(s^l) + R(b_{s^l}, s^l) = 1 + R(b_{s^l}, s^l),$$

where the first equality was due to the fact that the low-quality signal $s^l$ is noiseless ($\Pr(s^l|s^l) = 1$).

When $c^E = 0$ and $p = 0$, at the truthful equilibrium, an agent's expected utility is

$$\begin{aligned}&\sum_{s^h \in Q} \Pr(s^h)\Pr(s^h|s^h)\frac{1}{\Pr(s^h|s^h)} + \mathbb{E}[R(\Pr(r_j|s^h), r_j)] \\ &= \sum_{s^h \in Q} \Pr(s^h) + \mathbb{E}[R(\Pr(r_j|s^h), r_j)] = 1 + \mathbb{E}[R(\Pr(r_j|s^h), r_j)] \leq 1 + R(b_{s^l}, s^l),\end{aligned}$$

where the inequality was due to the fact that the proper scoring rule $R$ is symmetric. Thus, the $s^l$ equilibrium Pareto dominates the truthful equilibrium when $c^E = 0$ and $p = 0$. The conditions of Theorem 3 are therefore satisfied, and hence $p_{\text{Pareto}} \geq p_{\text{ds}}$ for all settings with positive effort cost $c^E \geq 0$.

*Divergence-Based BTS [Radanovic and Faltings, 2014]* When $c^E = 0$ and $p = 0$, the $s^l$ equilibrium exists. (If all other agents report $s^l$ and $b_{s^l}$, then the best belief report for agent $i$ is $b_{s^l}$. Moreover, the best signal report for agent $i$ is $s^l$, which means that the penalty is $0$ because the agent's signal reports agree and their belief reports also agree.)

When $c^E = 0$ and $p = 0$, at the $s^l$ equilibrium, an agent's expected utility is

$$-\mathbb{1}_{s^l = s^l || D(b_{s^l}, b_{s^l}) > \theta} + R(b_{s^l}, s^l) = R(b_{s^l}, s^l).$$

At the truthful equilibrium, an agent's expected utility is

$$-\mathbb{1}_{s^h_{i'j} = s^h_{i'j} || D(\Pr(r|s^h_{ij}), \Pr(r|s^h_{i'j})) > \theta} + R(\Pr(r|s^h), s^h) < R(\Pr(r|s^h), s^h) < R(b_{s^l}, s^l),$$

where the first inequality was due to the fact that the high-quality signal $s^l$ is noisy. That is, for every realization $s^h$ of the high-quality signal, $\Pr(s^h|s^h) \leq 1$ and there exists one realization $s^h$ of the high-quality signal such that $\Pr(s^h|s^h) < 1$. The second inequality was due to the fact that the proper scoring rule $R$ is symmetric. Thus, the $s^l$ equilibrium Pareto dominates the truthful equilibrium when $c^E = 0$ and $p = 0$. The conditions of Theorem 3 are therefore satisfied, and hence $p_{\text{Pareto}} \geq p_{\text{ds}}$ for all settings with positive effort cost $c^E \geq 0$.

*Riley [2014]* When $c^E = 0$ and $p = 0$, the $s^l$ equilibrium exists. (When all other agents always report $s^l$, for agent $i$, $\delta_i = 0$ because for any signal other than $s^l$, the number of other agents who reported the signal is 0. Thus, agent $i$'s reward is $R(b_i, s^l)$. Since agent $i$'s signal report does not affect his reward, reporting $s^l$ is as good as reporting any other value. Moreover, since all other agents report $s^l$, the best belief report for agent $i$ is to report $b_{s^l}$.)

When $c^E = 0$ and $p = 0$, at the $s^l$ equilibrium, $\delta_i = 0$ because for any signal other than $s^l$, the number of other agents who reported the signal is 0. Thus, an agent's expected utility is $R(b_{s^l}, s^l)$. By the definition of the mechanism, an agent's reward is at most $R(b_i, r_{-i})$, which is less than or equal to $R(b_{s^l}, s^l)$ because $R$ is a symmetric proper scoring rule. Therefore, an agent achieves the maximum expected utility at the $s^l$ equilibrium, which is greater than or equal to the agent's expected utility at the truthful equilibrium when $c^E = 0$ and $p = 0$. □

# Chapter 8

# Application: Mechanical TA

In the previous chapter, we started from a setting where the mechanism designer had no access to ground truth whatsoever, and analyzed how one could improve its properties by adding minimal, costly access to ground truth. In this chapter, we present a case study that starts from a setting in which the mechanism designer typically observes ground truth for every single object (i.e., by marking every assignment), and consider the symmetric question of how much costly access to ground truth we can remove without damaging the mechanism's goal of accurate evaluation.

Whereas the previous chapter took a theoretical approach, in this chapter we present a case study of a real-life peer grading scenario. In the previous chapter, we focused exclusively on the incentives problem, assuming that agents all had a reliable high-quality signal. In practice, however, students have widely differing abilities. Thus, a major focus of this chapter is on validating that students (the agents) are competent graders (i.e., have access to a reliable signal).

## 8.1 Introduction

This chapter describes our experience with software-supported, anonymous peer grading in a fourth year undergraduate course ("Computers and Society"). The

course focuses on reasoning critically about the importance and social implications of computational advances. In earlier offerings of the course, students had to write three essays: on the midterm, final, and for a term project. However, shorter and more frequent essay writing assignments are both a more effective way to teach writing skills [Seabrook et al., 2005], and provide more opportunities to evaluate and improve writing skills and critical reasoning skills. Over the course of three offerings (2011, 2012, and 2013), we thus shifted to assigning students a total of 14 essays of about 300 words (11 weekly assignments, plus one essay on the midterm exam and two essays on the final exam). Manually marking essays is very expensive in terms of teaching assistant (TA) time. Furthermore, it can be difficult for students to learn to write such essays well. Peer grading offers a solution to both problems.

Peer grading is far from a new idea. However, students are often concerned that the quality and fairness of the evaluation that they receive from peer grading is lower than it would be from TAs [Robinson, 2001; Paré and Joordens, 2008; Walvoord et al., 2008]. Most systems (surveyed at the end of this article) attempt to address these concerns by evaluating the quality of the peer reviews in an automated way, whether by reweighting reviews based on some criterion [Chapman, 2001; Hamer et al., 2005], by "review the reviewer" schemes in which students rate the feedback they have received [Paré and Joordens, 2008; de Alfaro and Shavlovsky, 2014b; Gehringer, 2001; Cho and Schunn, 2007b], by evaluating how close a review is to the combined "consensus" grade for an assignment [de Alfaro and Shavlovsky, 2014b; Hamer et al., 2005], or by some combination of these ideas.

We wanted to use peer grading to make more efficient use of TAs, not to replace them entirely. We thus designed a new system, dubbed "Mechanical TA."[1] Our system leverages (human) TAs in three ways. First, students start out in a *supervised* state, in which all of their reviews are marked by a TA. They are only promoted to an *independent* state when they demonstrate that they understand the

[1]Mechanical TA is freely available at http://www.cs.ubc.ca/~jrwright/mta/.

grading rubric and are able to apply it competently. Second, students may use the system to appeal any peer grade that they consider unfair. (We reduce abuse of this feature by requiring a 100 word explanation of why a student believed that a review was unfair.) Finally, every independent review is eligible to be randomly spot checked by a TA, who can retroactively mark a reviewer's past reviews if they uncover a poor review. We found that students had surprisingly few concerns about fairness in Mechanical TA, and believe that the visible involvement of human TAs in marking assignments—especially in the early part of the class, when most students were supervised—was a major reason why.

Our system of random spot checks and appeals allows students to be persistently promoted. That is, once a student has been promoted, they can remain independent for the remainder of the class (i.e., if they are not demoted again due to a spot check or an appeal). This contrasts with systems such as Calibrated Peer Review (CPR) [Chapman, 2001], in which students' review skills are retested at the beginning of each assignment. The time required to complete such calibration was a source of complaints in one study of CPR [Walvoord et al., 2008].

Our implementation of calibration has a strong element of automated practice rather than just evaluation.[2] To our knowledge, this is a unique feature of our system of calibration. Students receive immediate feedback about their performance on calibration essays, and may optionally choose to perform many more than the required number of calibrations. In Section 8.4.2, we present our finding that calibration practice significantly improved students' review performance. In Section 8.3.3 we present evidence that it also improved students' writing performance, as measured by exam scores.

We begin by describing our particular peer review model in detail in Section 8.2. We survey our experience with this model over three years (2011, 2012, 2013) in Section 8.3, and compare some outcomes between different offerings of the course, paying particular attention to the differences between the 2013

[2]Indeed, our evaluation suggests that this aspect was the main benefit offered by calibration in the 2013 offering.

offering—which included automated calibration—and the previous two offerings. In Section 8.4 we analyze the data from the 2013 offering. In addition to showing that calibration practice improved students' review skills, we also demonstrate that the persistent division of students into independent reviewers and supervised reviewers was an effective strategy. After reviewing some related work in Section 8.5, we conclude in Section 8.6.

## 8.2 Peer Evaluation Model

In brief, our peer review system works as follows. Students submit their essays as free-form text in the Mechanical TA system.[3] After the essay submission deadline, each student is assigned three essays for double-blind peer review. After the deadline for submitting reviews, each essay is assigned the median peer-review mark. Students can register a request for a TA to regrade their essay if they believe that they received an unfair grade. The use of medians to compute grades means that an appeal is only worthwhile if the student believes they received two unfair reviews.

A review consists of a configurable set of text fields and multiple-choice questions. In our "Computers and Society" class, students were asked to rate each essay on a scale of 0–5 along four dimensions—following a detailed rubric that described what an essay would look like to justify each score in each dimension—and provide a textual justification of their scores. The grade assigned to an essay by a review is the sum of the scores in each dimension. The full text of the rubric we used is provided in Appendix A.

Mechanical TA implements a variant of the spot-checking scheme of Chapter 7. The mechanism automatically assigns a selection of essays to the TAs for *spot checking*, in which the TA reads the essay and evaluates its reviews. These essays are randomly selected. Unlike Chapter 7, where every review is selected with equal probability, Mechanical TA chooses some essays with higher probability than others. For example, reviews that assign a high grade are more likely

[3] This makes it easy for us to check all essays for plagiarism using TurnItIn, which we do.

to be selected than reviews that assign a low grade. This addresses an incentive asymmetry caused by the appeal system, since overly generous reviews are much less likely to be appealed than overly harsh reviews.

### 8.2.1 Supervised and Independent Reviewers

As mentioned above, we classify students as either *supervised* or *independent* reviewers. Every student begins as a supervised reviewer. Every essay that they review is also reviewed by a TA, and peer reviews are disregarded in this case for the purpose of grading: supervised essays are assigned grades from TA reviews. Furthermore, supervised students' reviews are also marked by TAs. Students are promoted from supervised to independent when their average review marks crosses a configurable threshold. Once promoted to independent status, a student automatically receives $100\%$ on each of their reviews unless it is subsequently checked by a TA as described earlier, in which case it is graded.

Supervised reviewers are assigned only the essays of other supervised reviewers; similarly, independent students are only matched with each other. This is important in terms of TA workload: indeed, it minimizes the number of essays that must be read by the TAs who evaluate the supervised reviewers' reviews. If independent and supervised students could review each others' essays, then potentially *every* submitted essay would have at least one supervised reviewer and would hence need to be read. Conversely (and for the same reason), our scheme maximizes the number of essays that are fully peer graded.

### 8.2.2 Calibration

In addition to giving them the opportunity to learn by reviewing the work of their peers, Mechanical TA also allows students to practice reviewing via *calibration essays*. A calibration essay is an essay from a past offering of the course[4] which was carefully evaluated by multiple TAs to establish a gold standard review. At

[4]Mechanical TA allows students to flag whether or not submissions may be used anonymously; we chose essays whose authors had permitted anonymous reuse.

any time during the course, a student can request a calibration essay from Mechanical TA. The student then enters a review in the usual way. However, immediately after the review is submitted, Mechanical TA shows the student the gold standard review, and highlights the dimensions in which the student's review differed from the gold standard. If the student's review is within a configurable distance of the gold standard review, the student is given a "review point". After the student has collected enough review points over a configurable (potentially decaying) time window, they are promoted to independent status. This makes it possible for students to become independent before a TA has evaluated any of their reviews.

## 8.3 Evolution of our Design

Our design of the Mechanical TA system evolved over time. Analyzing data from three consecutive offerings of Computers and Society allows us to argue that our current design helps to achieve better student outcomes. We have described the peer review process used in 2013 in Section 8.2. In the initial 2011 offering, each essay was reviewed by only two students; its mark was the average of the two reviews. In 2012, we switched to using the median of three reviews. In the 2013 offering, we added the calibration process.

One of the major differences that calibration required was an extensively reworked rubric for reviewers. In the 2011 and 2012 offerings, reviewers were asked to rate each essay along 4 dimensions (Argument, Subject, Evidence, English) on a scale from 0 to 2. We offered minimal guidance about what separated 2/2 on a given dimension from 1/2. We found that students were extremely reluctant to give 1/2 grades in this scheme, and received many comments that students did not want to deduct half the possible marks for a dimension. In the 2013 offering, we reworked the rubric in two ways. First, we expanded each dimension's scale to run from 0 to 5. Second, students were given explicit descriptions about what sort of essay deserved each score for each dimension.

In the remainder of this section, we first describe the process of setting up to offer calibration essays for the first time. Offering calibration reviews made

a substantial impact, both on students' achievement, and on the workload for the TAs. In the final two subsections we compare 2013 to the earlier offerings by when students were promoted to independent reviewer, and by exam performance.

### 8.3.1 Calibration Setup

Constructing a library of calibration essays was a time-intensive process. We started by considering every essay from the previous offering that students had flagged as available for anonymous reuse. We then hand-selected 27 candidate essays. Each of these essays was reviewed by the same four TAs. The review marks were reconciled during in-person meetings, and every essay where the TAs reached consensus was selected as a calibration essay, whereas the other essays were discarded.

One extremely valuable (and unintended) benefit of the process of creating calibration essays was calibrating the TAs themselves. With the exception of the lead TA, our course is run by a new contingent of TAs every year, most or all of whom have no particular past experience in evaluating essays. The meetings and discussions to determine marks for the calibration essays constituted an opportunity to give the TAs extensive extra training.

A one-time benefit of the initial process of creating calibration essays was that it pointed out opportunities to improve our rubric. The rubric went through multiple iterations during the process of calibrating the TAs, as they discovered various ambiguities.

### 8.3.2 Independent Reviewers

One bottleneck in our original Mechanical TA design was that all students begin in the supervised pool, requiring extensive TA work at the beginning of term. One of our main motivations for introducing an automated calibration process to reduce this TA workload by encouraging students to be promoted to the independent pool before the first assignment was marked. We were unsuccessful in achieving this goal in our 2013 course offering: no students were promoted to

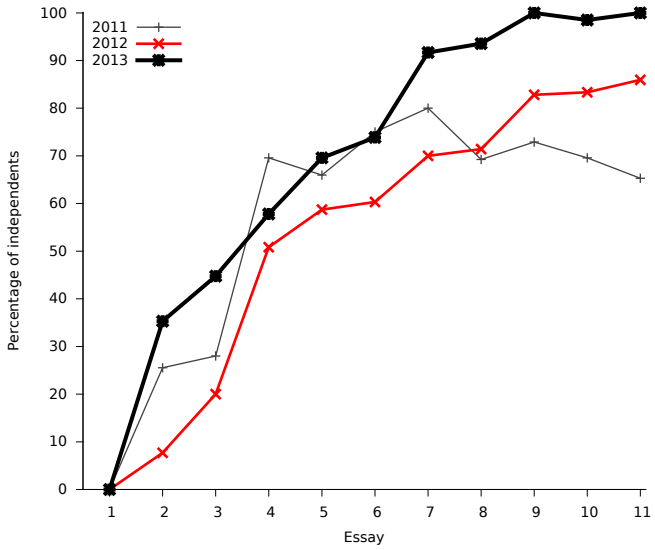


**Figure 8.1:** Proportion of independent reviewers at the beginning of each assignment.

independent before the first assignment, and hence TAs needed to mark every student's essay.[5] However, the opportunity to practice reviewing that the calibration essays provided appears to have had a large effect on students' review skills. More students were promoted to independent early in the 2013 offering than in either of the earlier offerings, and a larger overall proportion of the class (100%!) became independent during the term.

Figure 8.1 shows how many students reviewed independently over the course of the term in each of our past three offerings. The criteria for becoming independent in 2011 were much more lenient than in 2012, leading to a large number of students becoming independent fairly quickly. However, this leniency seems to have resulted in the promotion of many unreliable reviewers, and so many of these students were later moved back to the supervised pool as a result of spot checks and appeals. In contrast, all but one of the many students who became in-

[5]We've since tweaked our calibration threshold based on data from the 2013 offering, and the number of calibration essays required of students, with very positive effects. It is now routine for the majority of the class to join the supervised pool via calibrations before the first assignment.

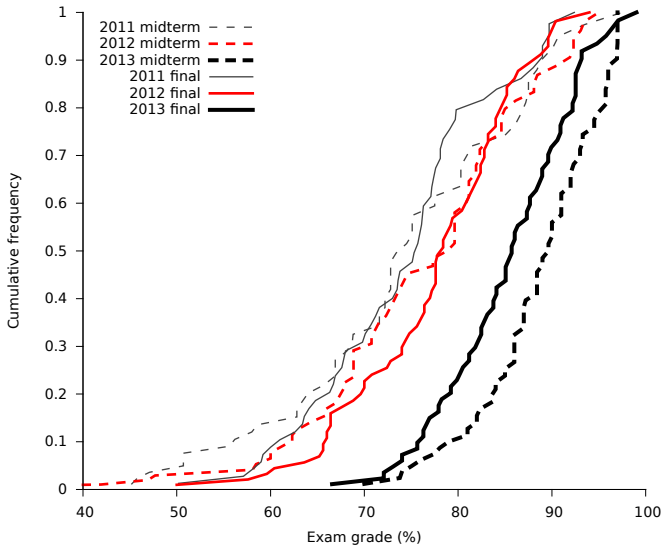


**Figure 8.2:** Cumulative distributions of final exam and midterm exam marks.

dependent in 2013 stayed in the independent pool throughout the class. Nearly a third of the students became independent after just one assignment; by the end of the course, *every* student was reviewing independently. The criteria for becoming independent based on review quality were identical in 2012 and 2013; the only differences between the two years were the introduction of our calibration system and the improvements we made to the review rubric to support calibration.

### 8.3.3 Exam Performance

It would be nice to compare assignment marks between years; however, this is difficult because we made dramatic changes to the rubric. In 2011 and 2012, we marked essays out of 8, and an "acceptable" essay received 8/8. In 2013, we marked essays out of 20, and gave an "acceptable" essay 16/20. Thus, we do not present an analysis of how assignment marks varied from one year to the next.

In contrast to assignments, we marked essays on the midterm and final exams in a very similar way across all three offerings, and indeed offered very similar

exams. This makes exams a more suitable target for analysis. Figure 8.2 gives the cumulative distributions of marks on the midterm and final exams across the three years. We observe that the mark distributions for both exams were strikingly higher in 2013 than in the prior years.[6] We thus conclude that one or both of the improvements associated with our calibration system had a positive impact on student performance.

## 8.4 Analysis of our Current Design

We now turn to a deeper analysis of data from the latest offering of Computers and Society. We first confirm that the division of supervised and independent reviewers meaningfully reflects differences in review quality. We then consider the effect of reviewing practice on calibration performance.

### 8.4.1 Review Quality

Mechanical TA is designed on the premise that independent reviewers can be trusted to reliably review peer work without oversight, whereas supervised reviewers cannot. An important question is therefore whether the two pools really do differ in terms of review quality. To answer this question, we followed the basic strategy of estimating the average quality of supervised reviews and the average quality of independent reviews, and checking whether these averages differed significantly. The quality of supervised reviews is easy to estimate, since all of them get marked by a TA. For independent reviews, we had access to TA marks of reviews that were randomly spot checked or appealed, unfortunately without a label indicating which criterion had led to their selection. The spot check selection criterion adds a complication, however: all essays that receive a grade of $80\%$ or higher get spot checked automatically; all other essays are spot checked at random. If the quality of an essay is independent of the quality of its reviews,

[6] A Mann-Whitney rank test confirms this. Both the midterm and final exam distributions for 2013 are significantly higher than the corresponding distribution for both 2011 and 2012 ($p < 0.001$).

then this does no harm. However, if high-quality essays are easier to grade, then this selection criterion could add an upward bias to the estimate of the independent reviews' quality, since our sample of independent reviews would contain disproportionately many easily graded essays. We address this by subdividing the independent and supervised reviews into those that were associated with essays that got a mark over $80\%$ and those that did not, giving a total of four groups of observations. This allows us to detect the situation where the supervised reviews have significantly different quality from the independent reviews of high-mark essays, but not significantly different quality from the independent reviews as a whole. Another possible source of bias is appeals, as low-quality reviews may be appealed more frequently than average. We do not attempt to correct for this bias, for two reasons. First, the bias is downward for independent reviews; if we found a statistical difference between the two pools in the presence of this bias, correcting for it would not change our finding. Second, we cannot distinguish retroactive spot checks that were triggered by random spot checks from those that were triggered by appeals.

For each of our four groups of reviews, we estimated a Bayesian joint posterior distribution over the following model:

$$\begin{aligned} \mu_g &\sim \text{Uniform}[0, 10] \\ \sigma_g &\sim \text{Uniform}[0.0001, 10] \\ q_{g,r} &\sim \text{N}(\mu_g, \sigma_g) \quad \text{truncated to } [0, 1], \end{aligned}$$

where $q_{g,r}$ is the quality of review $r$ in group $g$. We normalized all marks to lie within $[0, 1]$. The quality of each review in group $g$ is assumed to be drawn from the same Normal distribution, truncated at $0$ and $1$. We estimated the posterior distributions over the parameters $\mu_g, \sigma_g$ for each group using a Metropolis-Hastings sampler [Robert and Casella, 2004] to simulate $12,000$ samples after a burn-in period of $4000$ samples. We used the PyMC sampler to implement the sampler [Patil et al., 2010].

Figure 8.3 gives the cumulative posterior distribution over the average review quality for each group.[7] The 95% central credible interval for each distribution is shown as a bar on the $x$-axis.[8]

We observe that there is no overlap between the central credible intervals of either independent group with either supervised group. This answers our question: we have strong evidence that independent reviewers perform higher quality reviews.

We are also able to examine whether high-quality essays are easier to review well. In both the independent and supervised groups, the quality of the reviews of essays that received grades of at least 80% did indeed appear to be higher, although not substantially (nor significantly; the credible intervals for the above- and below-80% groups intersect). The effect was more pronounced in the supervised group than in the independent group, although again not statistically significant.

### 8.4.2 Calibration Performance

We have described two benefits offered by calibration: assessing students' review quality without TA intervention, and providing an opportunity for students to practice reviewing with immediate feedback. In this section, we evaluate whether students benefit from such practice by asking whether students' calibration marks improved as they completed more calibration reviews.

We begin by plotting the performance of each calibration that was completed. We index calibrations by the time of promotion to the independent pool; that is, the last calibration review performed before a student was promoted is calibration number 0, the calibration review completed just before that is calibration number -1, etc. We then perform a Bayesian linear regression by estimating the joint

[7] Due to the truncation of the Gaussian distributions to the interval $[0, 1]$, this is not identical to the posterior distribution of the $\mu_g$ parameter.

[8] A central credible interval is a Bayesian counterpart to a confidence interval. The true value of a parameter lies within its 95% central credible interval with probability $0.95$.

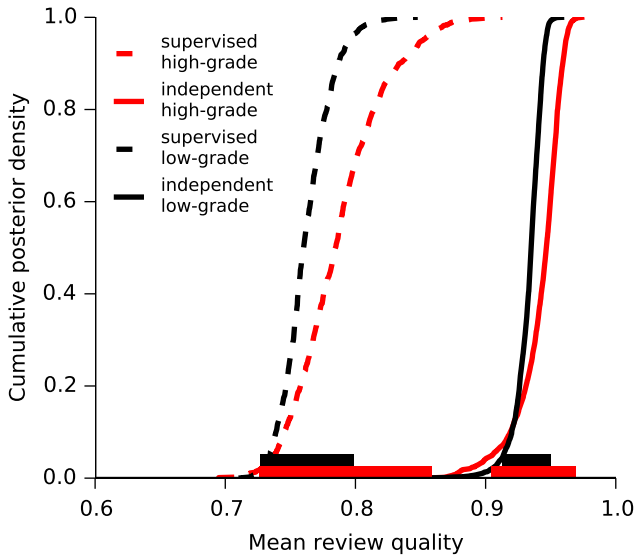


**Figure 8.3:** Cumulative posterior distributions for the mean review quality of supervised reviews of low-marked essays, supervised reviews of high-marked essays, independent reviews of low-marked essays, and independent reviews of high-marked essays. $95\%$ central credible intervals for each of the distributions are shown as bars on the $x$-axis.

posterior distribution of the following model:

$$b \sim \mathrm{N}(0, 5) \qquad \sigma \sim \mathrm{Uniform}[0, 10]$$
$$m \sim \mathrm{N}(0, 2) \qquad y_i \sim \mathrm{N}(mx_i + b, \sigma),$$

where $m$ and $b$ are the slope of the regression line, $x_i$ and $y_i$ are the number and performance for each calibration review, and each datapoint $(x_i, y_i)$ has zero-mean Gaussian noise with variance $\sigma$. Performance is measured as sum of absolute differences (i.e., the $L_1$ distance) from the instructor review; smaller performance values thus represent better performance. We again used Metropolis-Hastings sampling to estimate the posterior.

Figure 8.4 shows a plot of the number and performance for each calibration (with a small amount of jitter for readability). The maximum a posteriori regres-

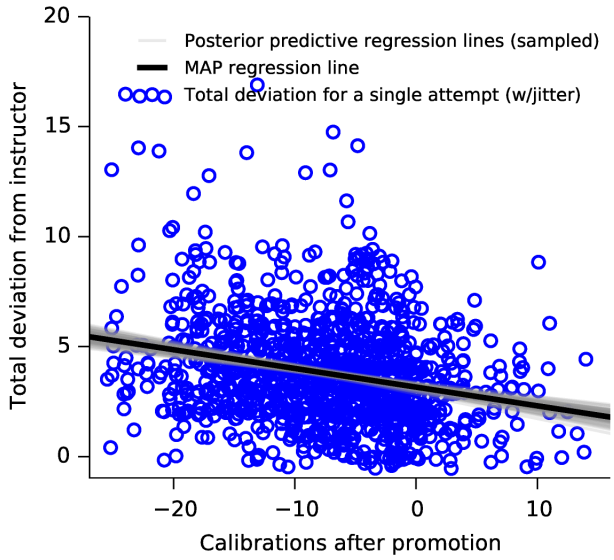


**Figure 8.4:** Total deviations from gold standard review on calibration reviews, versus number of calibrations after promotion. The bold line is the maximum a posteriori linear fit with Normal error. The gray lines are samples from the posterior predictive distribution of linear fits.

sion line is plotted as a bold line; this is the line whose slope and offset have the highest posterior probability. To illustrate the range of possible fits, we also plot the lines corresponding to 100 samples from the posterior distribution.

The MAP estimate of the slope is $-0.085$, with a $95\%$ central credible interval of $[-0.104, -0.066]$. The credible interval does not contain 0, so we conclude that students showed a significant improvement in their calibration performance as they practiced. Our rubric grades essays out of $20$, so a slope of $-0.085$ represents an average improvement of approximately $4\%$ with each calibration.

## 8.5 Related Work

Now that we have described Mechanical TA in detail, we give a more thorough survey of related work and describe how our own system differs. By far the most widely used online peer review system is Calibrated Peer Review (CPR) [Chap-

man, 2001; Robinson, 2001]. After submitting their own essays, students evaluate three instructor-provided calibration essays of varying known quality. They then anonymously review the essays of other students. Each review is weighted according to the reviewer's performance on the calibration task. Reviewers who do not pass the calibration task on the first two tries "flunk out" of the assignment and are not permitted to review at all. Review quality is further assessed by students' reviewing other students' reviews. The initial calibration essays are entirely for the purpose of evaluating students' reviewing skill, and form a portion of the students' grade. This contrasts with our calibration essays, which do not directly impact a student's grade, and which allow students to practice reviewing in addition to demonstrating reviewing competence.

Kulkarni et al. [2014] combine algorithmic assessment of written answers with peer review in a large online course. A learning algorithm first estimates both the assessment and its confidence in the assessment. These estimates are used to determine how many peer reviews are required for a given item. Other students then assess the peer reviews' accuracy.

Mechanical TA focuses on evaluating the final version of an assignment. SWoRD [Cho and Schunn, 2007b], PRAZE [Mulder and Pearce, 2007], and CaptainTeach [Politz et al., 2014] allow students to incorporate feedback from peer reviews during the course of an assignment.

CrowdGrader [de Alfaro and Shavlovsky, 2014b] dynamically assigns reviews to reviewers in an online fashion, in an attempt to provide an approximately equal number of reviews to each submission. Similarly to CPR, the quality of each review is assessed by comparing it to the "consensus" (trimmed average) review of the assignment; reviews that are further from the consensus are penalized. The Aropa system [Hamer et al., 2005] combines consistency scoring and weighting by reweighting reviews until the weights of the reviews are consistent with the weighted average. Both systems thus assess review quality "automatically," whereas Mechanical TA assesses review quality directly via TAs. Many other systems use a "review the reviewer" system to evaluate review quality, in which

students rate the quality of the reviews they have received [Gehringer, 2001; Cho and Schunn, 2007b; Mulder and Pearce, 2007; Paré and Joordens, 2008].

## 8.6 Conclusions

Mechanical TA is a system designed to support a novel model of high-stakes peer grading, in which marks from trusted *independent* reviewers are binding (but can be appealed), but marks from untrusted *supervised* reviewers are replaced by grades from a TA. Students are promoted to independent status based on the quality of their reviews, and after promotion they typically remain independent for the duration of the term. We have successfully used this system to set weekly essay assignments in a class of approximately 70 students. This would not be possible if every assignment had to be graded by a TA, as essays are very time consuming to grade. We have focused here on grading essays, but our system is easily applicable to other domains such as coding assignments or code review.

The initial version of Mechanical TA employed a peer-insensitive spot-checking mechanism (see Definition 15 in Chapter 7). A key driver of the theoretical investigation in Chapter 7 was an attempt to make more efficient use of spot-checking by incorporating peer-prediction into the spot checking mechanism. However, one of the main findings of that work was that peer-insensitive spot checking mechanisms are actually *more* efficient at incentivizing truthful reporting that peer-prediction spot checking mechanisms. In this chapter, we instead focused upon practical mechanisms for validating students' competence through the supervised/independent distinction.

A major bottleneck in our peer review approach is that the first assignment *does* require that TAs mark every submission along with all of the peer reviews. While we have found that TAs are willing to work hard at the beginning of term given assurances that they will subsequently have a much-reduced workload, this bottleneck nevertheless limits the scalability of our system. We thus introduced calibration reviews in the 2013 offering, in which students review carefully chosen assignments with known correct gold standard reviews constructed by the instruc-

tor and TAs. Each student receives automated feedback comparing their review to the gold standard review, and if they match the gold standard closely enough on enough repetitions, they are automatically promoted to independent status. This calibration mechanism has multiple goals. First, it aims to allow students to become independent before the first assignment, without TA intervention, thereby reducing TA workload on the first assignment. Second, it allows students to practice the reviewing process, with immediate feedback about how well they did. We did not achieve the first goal in the 2013 course offering. Nevertheless, offering students practice reviewing had a striking effect. Students in the 2013 offering were promoted sooner and received higher grades on roughly comparable exams than those in the 2011 and 2012 offerings. Students' average review performance improved by approximately $4\%$ per attempted calibration essay.

One additional benefit of a calibration system is that it allows the systematic training of TAs in how to mark according to subjective rubrics. (We described how our TAs benefited from constructing calibration questions; we've asked our 2014 TAs to do the existing calibration exercises before the class starts.) We believe that this leads to higher quality marking by TAs and more consistency between TAs.

Calibrating reviewers before the first assignment is a key requirement for increasing the scalability of Mechanical TA's peer review model. In the 2014 offering subsequent to the years analyzed in this chapter, we modified the calibration promotion threshold based on data from the 2013 offering. This yielded a vast improvement in how soon and how many students were moved to the supervised pool. In the 2014 offering, over half of the students were admitted to the independent pool before the first assignment.

# Bibliography


Agranov, M., Caplin, A., and Tergiman, C. (2010). The process of choice in guessing games. Social science working paper 1334r, California Institute of Technology. Available at http://www.wordsmatter.caltech.edu/SSPapers/sswp1334R.pdf, accessed Feb. 9, 2014. → pages 75

Altman, A., Bercovici-Boden, A., and Tennenholtz, M. (2006). Learning in one-shot strategic form games. In *ECML 2006, 17th European Conference on Machine Learning*, pages 6–17. → pages 37

Arad, A. (2012). The tennis coach problem: A game-theoretic and experimental study. *The BE Journal of Theoretical Economics*, 12(1). → pages 75

Arad, A. and Rubinstein, A. (2009). Colonel Blotto's top secret files. Working paper. Available at http://www.dklevine.com/archive/refs4814577000000000432.pdf, accessed Feb. 9, 2014. → pages 75

Arad, A. and Rubinstein, A. (2012). The 11–20 money request game: A level-$k$ reasoning study. *The American Economic Review*, 102(7):3561–3573. → pages 30, 75

Becker, T., Carter, M., and Naeve, J. (2005). Experts playing the traveler's dilemma. Working paper, University of Hohenheim. → pages 9

Bengio, Y. (2009). Learning deep architectures for AI. *Foundations and Trends in Machine Learning*, 2(1):1–127. → pages 5

Bengio, Y., Courville, A., and Vincent, P. (2013). Representation learning: A review and new perspectives. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 35(8):1798–1828. → pages 78

Bishop, C. (2006). *Pattern recognition and machine learning*. Springer. → pages 11

Breitmoser, Y. (2012). Strategic reasoning in p-beauty contests. *Games and Economic Behavior*, 75(2):555–569. → pages 36, 38

Burchardi, K. and Penczynski, S. (2011). Out of your mind: Eliciting individual reasoning in one shot games. Working paper, London School of Economics. → pages 29, 30

Burchardi, K. and Penczynski, S. (2012). Out of your mind: Eliciting individual reasoning in one shot games. Working paper, London School of Economics. Available at http://people.su.se/~kburc/research/BurchardiPenczynski2012.pdf, accessed Feb. 9, 2014. → pages 58, 75, 76

Cabrera, S., Capra, C., and Gmez, R. (2007). Behavior in one-shot travelers dilemma games: model and experiments with advice. *Spanish Economic Review*, 9(2):129–152. → pages 12

Camerer, C., Ho, T., and Chong, J. (2001). Behavioral game theory: Thinking, learning, and teaching. Nobel Symposium on Behavioral and Experimental Economics. → pages 36, 38

Camerer, C., Ho, T., and Chong, J. (2004). A cognitive hierarchy model of games. *Quarterly Journal of Economics*, 119(3):861–898. → pages 10, 15, 35, 36, 38, 40, 44, 45, 58, 70

Camerer, C. and Hua Ho, T. (1999). Experience-weighted attraction learning in normal form games. *Econometrica*, 67(4):827–874. → pages 12

Camerer, C., Nunnari, S., and Palfrey, T. R. (2011). Quantal response and nonequilibrium beliefs explain overbidding in maximum-value auctions. Working paper, California Institute of Technology. → pages 36, 38, 52

Camerer, C. and Thaler, R. H. (1995). Anomalies: Ultimatums, dictators and manners. *The Journal of Economic Perspectives*, 9(2):209–219. → pages 62

Camerer, C. F. (2003). *Behavioral Game Theory: Experiments in Strategic Interaction*. Princeton University Press. → pages 3, 9

Capraro, V. (2013). A model of human cooperation in social dilemmas. *PloS one*, 8(8):e72427. → pages 12

Carvalho, D. and Santos-Pinto, L. (2010). A cognitive hierarchy model of behavior in endogenous timing games. Working paper, Université de Lausanne, Faculté des HEC, DEEP. → pages 44

Chapman, O. L. (2001). Calibrated Peer Review™. → pages 130, 131, 142

Chawla, S., Hartline, J., and Nekipelov, D. (2014). Mechanism design for data science. In *Proceedings of the fifteenth ACM conference on Economics and computation*, pages 711–712. → pages 2

Cho, K. and Schunn, C. D. (2007a). Scaffolded writing and rewriting in the discipline: A web-based reciprocal peer review system. *Computers & Education*, 48(3):409–426. → pages 96

Cho, K. and Schunn, C. D. (2007b). Scaffolded writing and rewriting in the discipline: A web-based reciprocal peer review system. *Computers & Education*, 48(3):409–426. → pages 130, 143, 144

Choi, S. (2012). A cognitive hierarchy model of learning in networks. *Review of Economic Design*, 16(2-3):215–250. → pages 44

Chong, J., Camerer, C., and Ho, T. (2005). Cognitive hierarchy: A limited thinking theory in games. *Experimental Business Research, Vol. III: Marketing, accounting and cognitive perspectives*, pages 203–228. → pages 29, 36, 38

Clark, C. and Storkey, A. J. (2015). Training deep convolutional neural networks to play go. In *Proceedings of the 32nd International Conference on Machine Learning, ICML 2015, Lille, France, 6-11 July 2015*, pages 1766–1774. → pages 79, 80

Cooper, D. and Van Huyck, J. (2003). Evidence on the equivalence of the strategic and extensive form representation of games. *Journal of Economic Theory*, 110(2):290–308. → pages 22, 23, 24, 25, 32

Costa-Gomes, M. and Crawford, V. (2006). Cognition and behavior in two-person guessing games: An experimental study. *American Economic Review*, 96(5):1737–1768. → pages 22, 38

Costa-Gomes, M., Crawford, V., and Broseta, B. (1998). Cognition and behavior in normal-form games: an experimental study. Discussion paper 98-22, University of California, San Diego. → pages 22, 23, 25, 38

Costa-Gomes, M., Crawford, V., and Broseta, B. (2001). Cognition and behavior in normal-form games: An experimental study. *Econometrica*, 69(5):1193–1235. → pages 10, 14, 35, 38, 58, 60, 70

Costa-Gomes, M., Crawford, V., and Iriberri, N. (2009). Comparing models of strategic thinking in Van Huyck, Battalio, and Beil's coordination games. *Journal of the European Economic Association*, 7(2-3):365–376. → pages 11, 36, 38

Costa-Gomes, M. A. and Weizsäcker, G. (2008). Stated beliefs and play in normal-form games. *The Review of Economic Studies*, 75(3):729–762. → pages 22, 23, 25, 36, 38

Crawford, V. and Iriberri, N. (2007a). Fatal attraction: Salience, naivete, and sophistication in experimental "hide-and-seek" games. *American Economic Review*, 97(5):1731–1750. → pages 29, 30, 36, 38, 75

Crawford, V. and Iriberri, N. (2007b). Level-$k$ auctions: Can a nonequilibrium model of strategic thinking explain the winner's curse and overbidding in private-value auctions? *Econometrica*, 75(6):1721–1770. → pages 30, 75

Dasgupta, A. and Ghosh, A. (2013). Crowdsourced judgement elicitation with endogenous proficiency. In *Proceedings of the 22nd International Conference on the World Wide Web*, pages 319–330. → pages 96, 97, 98, 101, 111, 121, 122

de Alfaro, L. and Shavlovsky, M. (2014a). Crowdgrader: A tool for crowdsourcing the evaluation of homework assignments. In *Proceedings of the 45th ACM Technical Symposium on Computer Science Education*, pages 415–420. → pages 96

de Alfaro, L. and Shavlovsky, M. (2014b). CrowdGrader: A tool for crowdsourcing the evaluation of homework assignments. In *Proceedings of the 45th ACM Technical Symposium on Computer Science Education*, SIGCSE '14, pages 415–420, New York, NY, USA. ACM. → pages 130, 143

Faltings, B., Jurca, R., Pu, P., and Tran, B. D. (2014). Incentives to counter bias in human computation. In *Second AAAI Conference on Human Computation and Crowdsourcing*. → pages 97

Faltings, B., Li, J. J., and Jurca, R. (2012). Eliciting truthful measurements from a community of sensors. In *3rd International Conference on the Internet of Things*, pages 47–54. → pages 96, 100, 101, 111, 121, 122

Frey, S. and Goldstone, R. (2011). Going with the group in a competitive game of iterated reasoning. In *2011 Proceedings of the Cognitive Science Society*, pages 1912–1917. → pages 44

Gao, X. A., Mao, A., Chen, Y., and Adams, R. P. (2014). Trick or treat: putting peer prediction to the test. In *Proceedings of the Fifteenth ACM Conference on Economics and Computation*, pages 507–524. → pages 97

Gehringer, E. F. (2001). Electronic peer review and peer grading in computer-science courses. *ACM SIGCSE Bulletin*, 33(1):139–143. → pages 130, 144

Georganas, S., Healy, P. J., and Weber, R. (2010). On the persistence of strategic sophistication. Working paper, University of Bonn. → pages 38

Gilboa, I. and Schmeidler, D. (1989). Maxmin expected utility with non-unique prior. *Journal of Mathematical Economics*, 18(2):141–153. → pages 2

Gill, J. (2002). *Bayesian methods: A social and behavioral sciences approach*. CRC press. → pages 42

Gneiting, T. and Raftery, A. E. (2007). Strictly proper scoring rules, prediction, and estimation. *Journal of the American Statistical Association*, 102(477):359–378. → pages 126

Goeree, J. K. and Holt, C. A. (2001). Ten little treasures of game theory and ten intuitive contradictions. *American Economic Review*, 91(5):1402–1422. → pages 3, 9, 22, 23, 25, 30, 32

Goeree, J. K. and Holt, C. A. (2004). A model of noisy introspection. *Games and Economic Behavior*, 46(2):365–382. → pages 10, 18, 35, 36, 38

Goldstein, A. A. (1964). Convex programming in Hilbert space. *Bull. Amer. Math. Soc.*, 70(5):709–710. → pages 88

Goodie, A. S., Doshi, P., and Young, D. L. (2012). Levels of theory-of-mind reasoning in competitive games. *Journal of Behavioral Decision Making*, 25(1):95–108. → pages 44

Hahn, P. R., Lum, K., and Mela, C. (2010). A semiparametric model for assessing cognitive hierarchy theories of beauty contest games. Working paper, Duke University. → pages 36, 38

Hamer, J., Ma, K. T. K., and Kwong, H. H. F. (2005). A method of automatic grade calibration in peer assessment. In *Proceedings of the 7th Australasian Conference on Computing Education - Volume 42*, ACE '05, pages 67–72, Darlinghurst, Australia, Australia. Australian Computer Society, Inc. → pages 96, 130, 143

Hansen, N. and Ostermeier, A. (2001). Completely derandomized self-adaptation in evolution strategies. *Evolutionary Computation*, 9(2):159–195. → pages 26

Hargreaves Heap, S., Rojo Arjona, D., and Sugden, R. (2014). How portable is level-0 behavior? A test of level-$k$ theory in games with non-neutral frames. *Econometrica*, 82(3):1133–1151. → pages 76

Haruvy, E. and Stahl, D. (2007). Equilibrium selection and bounded rationality in symmetric normal-form games. *Journal of Economic Behavior and Organization*, 62(1):98–119. → pages 22, 24, 25

Haruvy, E., Stahl, D., and Wilson, P. (1999). Evidence for optimistic and pessimistic behavior in normal-form games. *Economics Letters*, 63(3):255–259. → pages 38

Haruvy, E., Stahl, D., and Wilson, P. (2001). Modeling and testing for heterogeneity in observed strategic behavior. *Review of Economics and Statistics*, 83(1):146–157. → pages 22, 23, 25, 29, 38

Ho, T., Camerer, C., and Weigelt, K. (1998). Iterated dominance and iterated best response in experimental "$p$-beauty contests". *American Economic Review*, 88(4):947–969. → pages 10, 11

Hutter, F., Hoos, H. H., and Leyton-Brown, K. (2010). Sequential model-based optimization for general algorithm configuration (extended version). Technical Report TR-2010-10, University of British Columbia, Department of Computer

Science. Available online:
http://www.cs.ubc.ca/~hutter/papers/10-TR-SMAC.pdf. → pages 67

Hutter, F., Hoos, H. H., and Leyton-Brown, K. (2011). Sequential model-based optimization for general algorithm configuration. In *Proc. of LION-5*, page 507523. → pages 67

Hutter, F., Hoos, H. H., and Leyton-Brown, K. (2012). Parallel algorithm configuration. In *Proc. of LION-6*, pages 55–70. → pages 67

Jiang, A. X., Leyton-Brown, K., and Bhat, N. A. (2011). Action-graph games. *Games and Economic Behavior*, 71(1):141–173. → pages 22

John, L. K., Loewenstein, G., and Prelec, D. (2012). Measuring the prevalence of questionable research practices with incentives for truth telling. *Psychological Science*, 23(5):524–532. → pages 97

Jurca, R. and Faltings, B. (2005). Enforcing truthful strategies in incentive compatible reputation mechanisms. *Internet and Network Economics*, pages 268–277. → pages 98

Jurca, R. and Faltings, B. (2009). Mechanisms for making crowds truthful. *Journal of Artificial Intelligence Research*, 34(1):209. → pages 96, 97, 103

Kamble, V., Shah, N., Marn, D., Parekh, A., and Ramachandran, K. (2015). Truth serums for massively crowdsourced evaluation tasks. *arXiv preprint arXiv:1507.07045*. → pages 96, 97, 101, 111, 121, 123

Kearns, M., Littman, M. L., and Singh, S. (2001). Graphical models for game theory. In *Proceedings of the Seventeenth Conference on Uncertainty in Artificial Intelligence*, pages 253–260. → pages 22

Koller, D. and Milch, B. (2001). Multi-agent influence diagrams for representing and solving games. In *IJCAI*, pages 1027–1036. → pages 22

Kong, Y., Schoenebeck, G., and Ligett, K. (2016). Putting peer prediction under the micro (economic) scope and making truth-telling focal. *arXiv preprint arXiv:1603.07319*. → pages 96, 103

Kulkarni, C. E., Socher, R., Bernstein, M. S., and Klemmer, S. R. (2014). Scaling short-answer grading by combining peer assessment with algorithmic

scoring. In *Proceedings of the First ACM Conference on Learning @ Scale*, pages 99–108. → pages 96, 143

LeCun, Y., Bengio, Y., and Hinton, G. (2015). Deep learning. *Nature*, 521(7553):436–444. → pages 79

LeCun, Y., Bottou, L., Bengio, Y., and Haffner, P. (1998). Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2324. → pages 6

Lin, M., Chen, Q., and Yan, S. (2013). Network in network. *CoRR*, abs/1312.4400. → pages 80, 88

McKelvey, R., McLennan, A., and Turocy, T. (2007). Gambit: Software tools for game theory, version 0.2007. 01.30. → pages 26

McKelvey, R. and Palfrey, T. (1995). Quantal response equilibria for normal form games. *Games and Economic Behavior*, 10(1):6–38. → pages 10, 13, 35, 38

Miller, N., Resnick, P., and Zeckhauser, R. (2005). Eliciting informative feedback: The peer-prediction method. *Management Science*, 51(9):1359–1373. → pages 96, 103

Morgan, J. and Sefton, M. (2002). An experimental investigation of unprofitable games. *Games and Economic Behavior*, 40(1):123–146. → pages 36, 38

Mulder, R. A. and Pearce, J. M. (2007). PRAZE: Innovating teaching through online peer review. In *Proceedings of the 24th Annual Conference of the Australasian Society for Computers in Learning in Tertiary Education*. → pages 143, 144

Nagel, R. (1995). Unraveling in guessing games: An experimental study. *American Economic Review*, 85(5):1313–1326. → pages 10, 35, 58, 75

Neal, R. M. (2001). Annealed importance sampling. *Statistics and Computing*, 11(2):125–139. → pages 42

Osborne, M. J. and Rubinstein, A. (1998). Games with procedurally rational players. *American Economic Review*, 88(4):834–847. → pages 12

Paré, D. E. and Joordens, S. (2008). Peering into large lectures: examining peer and expert mark agreement using peerScholar, an online peer assessment tool. *Journal of Computer Assisted Learning*, 24(6):526–540. → pages 96, 130, 144

Patil, A., Huard, D., and Fonnesbeck, C. (2010). PyMC: Bayesian stochastic modelling in Python. *Journal of Statistical Software*, 35(1). → pages 43, 139

Politz, J. G., Patterson, D., Krishnamurthi, S., and Fisler, K. (2014). CaptainTeach: Multi-stage, in-flow peer review for programming assignments. In *ACM SIGCSE Conference on Innovation and Technology in Computer Science Education*. → pages 143

Prelec, D. (2004). A Bayesian truth serum for subjective data. *science*, 306(5695):462–466. → pages 96, 103

Rabin, M. (2000). Risk aversion and expected-utility theory: A calibration theorem. *Econometrica*, pages 1281–1292. → pages 3

Radanovic, G. and Faltings, B. (2013). A robust Bayesian truth serum for non-binary signals. In *Proceedings of the 27th AAAI Conference on Artificial Intelligence*, pages 833–839. → pages 96, 102, 104, 112, 125, 127

Radanovic, G. and Faltings, B. (2014). Incentives for truthful information elicitation of continuous signals. In *Twenty-Eighth AAAI Conference on Artificial Intelligence*. → pages 96, 102, 103, 104, 112, 125, 127

Radanovic, G. and Faltings, B. (2015). Incentives for subjective evaluations with private beliefs. In *Twenty-Ninth AAAI Conference on Artificial Intelligence*. → pages 97, 101, 102, 111, 121, 124

Rey-Biel, P. (2009). Equilibrium play and best response to (stated) beliefs in normal form games. *Games and Economic Behavior*, 65(2):572–585. → pages 38

Riley, B. (2014). Minimum truth serums with optional predictions. In *Proceedings of the 4th Workshop on Social Computing and User Generated Content*. → pages 96, 102, 103, 104, 112, 125, 128

Robert, C. P. and Casella, G. (2004). *Monte Carlo statistical methods*. Springer Verlag. → pages 42, 139

Robinson, R. (2001). Calibrated Peer Review™ an application to increase student reading & writing skills. *The American Biology Teacher*, 63(7):pp. 474–476+478–480. → pages 130, 143

Rogers, B. W., Palfrey, T. R., and Camerer, C. F. (2009). Heterogeneous quantal response equilibrium and cognitive hierarchies. *Journal of Economic Theory*, 144(4):1440–1467. → pages 10, 12, 16, 22, 24, 25, 29, 32, 36, 37, 38

Savage, L. (1951). The theory of statistical decision. *Journal of the American Statistical Association*, 46(253):55–67. → pages 61

Schmidhuber, J. (2015). Deep learning in neural networks: An overview. *Neural Networks*, 61:85–117. → pages 80

Seabrook, R., Brown, G. D., and Solity, J. (2005). Distributed and massed practice: from laboratory to classroom. *Applied Cognitive Psychology*, 19(1):107–122. → pages 130

Selten, R. and Buchta, J. (1994). Experimental sealed bid first price auctions with directly observed bid functions. Discussion paper B-270, University of Bonn. → pages 12

Selten, R. and Chmura, T. (2008). Stationary concepts for experimental $2 \times 2$-games. *American Economic Review*, 98(3):938–966. → pages 12

Shaw, A. D., Horton, J. J., and Chen, D. L. (2011). Designing incentives for inexpert human raters. In *Proceedings of the ACM 2011 Conference on Computer Supported Cooperative Work*, pages 275–284. → pages 97

Shnayder, V., Agarwal, A., Frongillo, R., and Parkes, D. C. (2016). Informed truthfulness in multi-task peer prediction. *arXiv preprint arXiv:1603.03151*. → pages 96, 97, 101, 112, 121, 122

Shoham, Y. and Leyton-Brown, K. (2008). *Multiagent Systems: Algorithmic, Game-theoretic, and Logical Foundations*. Cambridge University Press. → pages 26

Silver, D., Huang, A., Maddison, C. J., Guez, A., Sifre, L., van den Driessche, G., Schrittwieser, J., Antonoglou, I., Panneershelvam, V., Lanctot, M., Dieleman, S., Grewe, D., Nham, J., Kalchbrenner, N., Sutskever, I., Lillicrap, T., Leach, M., Kavukcuoglu, K., Graepel, T., and Hassabis, D. (2016).

Mastering the game of Go with deep neural networks and tree search. *Nature*, 529:484–503. → pages 80

Stahl, D. and Haruvy, E. (2008). Level-$n$ bounded rationality and dominated strategies in normal-form games. *Journal of Economic Behavior and Organization*, 66(2):226–232. → pages 22, 24, 25, 38

Stahl, D. and Wilson, P. (1994). Experimental evidence on players' models of other players. *Journal of Economic Behavior and Organization*, 25(3):309–327. → pages 10, 11, 16, 17, 22, 23, 24, 25, 29, 35, 38, 44, 45, 58

Stahl, D. and Wilson, P. (1995). On players' models of other players: Theory and experimental evidence. *Games and Economic Behavior*, 10(1):218–254. → pages 10, 22, 23, 25, 29, 36, 38

Thaler, R. H. (1988). Anomalies: The Ultimatum Game. *The Journal of Economic Perspectives*, 2(4):195–206. → pages 62

Turocy, T. (2005). A dynamic homotopy interpretation of the logistic quantal response equilibrium correspondence. *Games and Economic Behavior*, 51(2):243–263. → pages 13

Tversky, A. and Kahneman, D. (1992). Advances in prospect theory: Cumulative representation of uncertainty. *Journal of Risk and Uncertainty*, 5(4):297–323. → pages 2

Von Neumann, J. and Morgenstern, O. (1944). *Theory of Games and Economic Behavior*. Princeton University Press. → pages 14

Waggoner, B. and Chen, Y. (2014). Output agreement mechanisms and common knowledge. In *Proceedings of the 2nd AAAI Conference on Human Computation and Crowdsourcing*. → pages 96, 97, 101, 111, 121

Walvoord, M. E., Hoefnagels, M. H., Gaffin, D. D., Chumchal, M. M., and Long, D. A. (2008). An analysis of Calibrated Peer Review (CPR) in a science lecture classroom. *Journal of College Science Teaching*, 37(4):66. → pages 130, 131

Weizsäcker, G. (2003). Ignoring the rationality of others: Evidence from experimental normal-form games. *Games and Economic Behavior*, 44(1):145–171. → pages 10, 12, 38

Wilson, R. (1987). Game-theoretic approaches to trading processes. In *Advances in Economic Theory: Fifth World Congress*, pages 33–77. → pages 100

Witkowski, J., Bachrach, Y., Key, P., and Parkes, D. C. (2013). Dwelling on the negative: Incentivizing effort in peer prediction. In *Proceedings of the 1st AAAI Conference on Human Computation and Crowdsourcing*. → pages 96, 100, 101, 111, 121

Witkowski, J. and Parkes, D. C. (2012). A robust Bayesian truth serum for small populations. In *Proceedings of the 26th AAAI Conference on Artificial Intelligence*. → pages 96, 102, 104, 112, 125, 126

Witkowski, J. and Parkes, D. C. (2013). Learning the prior in minimal peer prediction. In *Proceedings of the 3rd Workshop on Social Computing and User Generated Content at the ACM Conference on Electronic Commerce*, page 14. → pages 96, 97, 102, 104, 112, 125, 126

Witten, I. H. and Frank, E. (2000). *Data Mining: Practical Machine Learning Tools and Techniques with Java Implementations*. Morgan Kaufmann. → pages 22

Wright, J. R. and Leyton-Brown, K. (2010). Beyond equilibrium: Predicting human behavior in normal-form games. In *Proceedings of the Twenty-Fourth AAAI Conference on Artificial Intelligence*, pages 901–907. → pages iv

Wright, J. R. and Leyton-Brown, K. (2012). Behavioral game-theoretic models: A Bayesian framework for parameter analysis. In *Proceedings of the 11th International Conference on Autonomous Agents and Multiagent Systems*, volume 2, pages 921–928. → pages iv

Wright, J. R. and Leyton-Brown, K. (2014). Level-0 meta-models for predicting human behavior in games. In *Proceedings of the Fifteenth ACM Conference on Economics and Computation*, pages 857–874. → pages iv, 66

Wright, J. R., Thornton, C., and Leyton-Brown, K. (2015). Mechanical TA: Partially automated high-stakes peer grading. In *Proceedings of the 46th ACM Technical Symposium on Computer Science Education*, pages 96–101. → pages v

Zhang, P. and Chen, Y. (2014). Elicitability and knowledge-free elicitation with peer prediction. In *Proceedings of the 2014 International Conference on Autonomous Agents and Multiagent Systems*, pages 245–252. → pages 96, 103

# Appendices

# Appendix A

# CPSC 430 2014 grading rubric

For each of the following four dimensions, choose the option that best describes the essay:

## English

Was the essay presented CLEARLY AND IN CORRECT ENGLISH?

0. Completely indecipherable.
1. Very difficult to understand.
2. Weak presentation; errors that impede understanding.
3. Mostly correct, fairly clear writing.
4. Clear and correct writing.
5. Very clear and correct writing.

## Argument structure

Was the essay WELL STRUCTURED, stating a thesis, supporting it with argument(s) that are clearly related to this point and (if relevant) distinct from one another, and linking these arguments in a logical way?

0. It is unclear what this essay is arguing.
1. It is apparent what is being argued, but much of the reasoning is unsound, unclear, or unrelated.
2. The thesis is clearly stated, and some claims support the thesis, but others are irrelevant and/or redundant.
3. All claims lend support to a clearly stated thesis, but they are insufficiently distinct and/or poorly linked together.
4. All claims lend support to a clearly stated thesis, which in turn relates appropriately to the question asked. The claims are distinct from one another and build well on each other in a logical progression.
5. Very well structured: the thesis is clear and well related to the question asked; the logical structure of arguments does an excellent job of supporting this thesis.

## Case

Did the essay do a GOOD JOB OF MAKING ITS CASE, choosing relevant arguments, backing them up with evidence and examples at an appropriate level of detail, and responding to contrary views as appropriate?

0. Claims are asserted with no further support, or not asserted at all.

1. The essay stated many facts about the topic in question, but there is not a clear separation between argument and evidence.

2. The essay makes recognizable arguments and backs them up with evidence, but relevance and/or level of detail are very inappropriate and/or extremely relevant contrary views are disregarded.

3. Arguments are clearly stated and generally support the thesis; these arguments are backed up with generally relevant evidence at a broadly appropriate level of detail. No extremely relevant contrary view undermines these arguments, though such arguments may or may not be explicitly addressed in the essay.

4. All claims are grounded in relevant and specific arguments at an appropriate level of detail; some attempt is made to respond to alternate points of view.

5. Whether or not I personally agree with the essay's thesis, it makes a compelling argument for its point of view. Arguments are very relevant, backed up with evidence at an appropriate level of detail, and (within space available) responses are offered to obvious objections.

## Subject matter

Did the essay demonstrate a good UNDERSTANDING OF THE COURSE'S SUBJECT MATTER, including both the topic and the wider context?

0. Profound and fundamental misunderstanding of the subject matter.
1. Poor understanding of the subject matter; major errors.
2. Factual errors that substantially undermined the essay's main point.
3. Generally correct understanding, but minor errors and/or errors of omission (failure to introduce important facts).
4. Correct understanding, generally balanced presentation at an appropriate level of detail.
5. Insightful understanding, creative and balanced use of the course's subject matter.