Simulation-Based Approaches for Evaluating Information Elicitation Mechanisms and Information Aggregation Algorithms

by

Noah H. Burrell

A dissertation submitted in partial fulfillment
of the requirements for the degree of
Doctor of Philosophy
(Computer Science and Engineering)
in the University of Michigan
2023

Doctoral Committee:

Professor Christopher Peikert, Co-Chair
Associate Professor Grant Schoenebeck, Co-Chair
Associate Professor Danai Koutra
Professor Paul Resnick

Noah H. Burrell

burrelln@umich.edu

ORCID iD: 0000-0003-3448-080X

ACKNOWLEDGEMENTS

First and foremost, thanks to my research advisor, Grant Schoenebeck, for his patience, encouragement, and guidance. His advice has been the foundation of my success as a graduate student. Thanks also to my academic advisor, Chris Peikert, for his enthusiastic support. Thanks to Danai Koutra and Paul Resnick for volunteering their valuable time to serve on my dissertation committee.

To my fellow Schoenebeck advisees: Thanks to you all for sharing ideas and teaching me. Thanks especially to Fang-Yi Yu and Biaoshuai Tao for setting an example for me to follow and to Yichi Zhang for being my most persistent and dependable collaborator.

In terms of helping make the projects that form this dissertation possible: Thanks to Professor Jason Hartline at Northwestern University for sharing peer grading data and to Michalis Mamakos for helping us use it effectively. Thanks to Ryan Sanchez and Hedayat Zarkoob for sharing their code and diligently answering my questions about it.

Personally: Thanks to my partner, Kaity, and to my parents, Andrea and Dave, for being there for me every step of the way. Thanks also to Sheena Thurston, my math teacher for three years in high school, for inspiring a love of math that set me on this path.

This work was supported by the National Science Foundation, which supported this work under award #2007256.

TABLE OF CONTENTS

CHAPTER

LIST OF TABLES

TABLE

LIST OF FIGURES

FIGURE

LIST OF APPENDICES

LIST OF ACRONYMS

1PL One-Parameter Logistic Model

2PL Two-Parameter Logistic Model

3PL Three-Parameter Logistic Model

10FL Ten-Fold Cross Validation

ABM Agent-Based Model

AUC Area Under the Curve

AUCC Area Under the Curve Correlation

BIC Bayesian Information Criterion

C1PL Category-dependent One-Parameter Logistic Model

C2PL Category-dependent Two-Parameter Logistic Model

CDF Cumulative Distribution Function

CIRT Category-dependent Item Response Theory

DMI Determinant-based Mutual Information

DS Dawid-Skene

FOSD First-Order Stochastic Dominance

IRT Item Response Theory

KDE Kernel Density Estimation

LLH Log-Likelihood

MOOC Massive Open Online Course

MLE Maximum Likelihood Estimate

MSE Mean Squared Error

OA Output Agreement

PTS Peer Truth Serum

SE Squared Error

SOSD Second-Order Stochastic Dominance

SQUARE Statistical QUality Assurance Robustness Evaluation

ABSTRACT

The mathematical study of information elicitation has led to elegant theories about the behavior of economic agents asked to share their private information. Similarly, the study of information aggregation has illuminated the possibility of combining independent sources of imperfect information so that the combined information is more valuable than that from any single source. However, despite a flourishing academic literature in both areas, some of their key insights have yet to be embraced in many of their purported applications. In this dissertation, we revisit prior work in the applications of crowdsourcing and peer assessment to address overlooked obstacles to more widespread adoption of their key contributions.

We apply simulation-based methods to the evaluation of information elicitation mechanisms and information aggregation algorithms. First, we use real crowdsourcing data to explore common assumptions about the way that crowd workers make mistakes in labeling. We find that a common assumption from theoretical work, that workers identify correct labels with a constant probability, is implausible in the data sets we consider. We further find different forms of heterogeneity among both tasks and workers, which have different implications for the design and evaluation of label aggregation algorithms.

Then, we turn to peer assessment. Despite many potential benefits from peer grading, the traditional paradigm, where one instructor grades each submission, predominates. One persistent impediment to adopting a new grading paradigm is doubt that it will assign grades that are at least as good as those that would have been assigned under the existing paradigm. We address this impediment by using tools from economics to define a practical framework for determining when peer grades clearly exceed the standard set by the instructor baseline. We simulate realistic grading data using a model from prior work, and find that peer grading is unlikely to be clearly preferable to instructor grading for participants unless we either significantly increase the workload of students or make stronger assumptions about participants' utility functions. We also study the effectiveness of various interventions to improve the quality of peer grading and the optimal allocations of peer and instructor resources under a fixed grading budget.

Lastly, we propose *measurement integrity*, which quantifies a peer prediction mechanism's ability to assign rewards that reliably measure agents according to the quality of their reports, as a novel desideratum in many applications. We perform computational experiments with mechanisms that elicit information without verification in the setting of peer assessment to empirically evaluate mechanisms according to both measurement integrity and robustness against strategic reporting. We find an apparent trade-off between these properties: The best-performing mechanisms in terms of measurement integrity are highly susceptible to strategic reporting. But we also find that supplementing mechanisms with realistic parametric statistical models results in mechanisms that strike the best balance between them.

CHAPTER 1

Introduction

The mathematical study of information elicitation (Section 1.1.1) has led to elegant theories about the behavior of economic agents asked to share their private information. Similarly, the study of information aggregation (Section 1.1.2) has illuminated the possibility of combining independent sources of imperfect information so that the combined information is more valuable than that from any single source, thereby harnessing the "wisdom of crowds" [100]. However, despite a flourishing, active academic literature in both areas, some of their key insights have yet to be embraced in many of their purported applications. In this dissertation, we revisit prior work in both areas with applications in mind, in order to address important overlooked obstacles impeding more widespread adoption of their key contributions. A key component of this endeavor involves looking beyond the question of "What is the next research question within the dominant model?" and interrogating the implicit and explicit assumptions of that model or considering a new model or methodology altogether.

We find that simulation-based computational experiments are a powerful tool for conducting these lines of inquiry. Computational experiments have distinct strengths that complement the strengths and supplement the weaknesses of theoretical work. Computational experiments are more naturally outcome-oriented. Simulated outcomes can be generated cheaply, frequently, and reliably under a wide range of parameter specifications. In contrast, making general theoretical statements about *ex post* outcomes is difficult in many cases. While theorems have the advantage of potentially applying to a larger range of settings, such theorems are may fail to give tight bounds, be hard to interpret, or both. In contrast, computational experiments readily provide interpretable results, albeit on a chosen set of inputs.

Computational experiments can also uniquely complement human-subjects experiments. Computational experiments allow for access to latent variables that are not readily observable in human-subjects experiments. Also, in human-subjects experiments, outcomes may depend heavily on hard-to-quantify factors such as the subject population and their training. In particular, it can be difficult to explain the mathematical inner workings of a mechanism in a way that makes properties like incentive-compatibility salient and it can be costly to conduct an experiment with enough repeated interactions between subjects and a mechanism that such properties can be discovered organically by the participants. In computational experiments, we can decouple exploring the properties of a mechanism from learning how to effectively explain the mechanism to users. Simulated agents can be instructed to precisely follow a behavior against which we would like to evaluate a mechanism's robustness. Or, simulated agents can be allowed to repeatedly interact with mechanisms for many iterations and learn strategies to maximize their rewards. In both cases, the behaviors of agents can be grounded in empirical results from human-subjects research. Further, simulation code can be shared publicly, so that experiments can be easily replicated or modified for subsequent inquiries as new behavioral insights are discovered.

On the other hand, computational experiments have their own limitations. Importantly, simulation often requires the complete specification of a model of the setting–instead of making a few comparatively mild assumptions about a generic underlying model, as in theoretical work. This potentially limits the generalizability of simulation results beyond the specific model that is adopted. We will see, however, that this limitation can be a useful one. In particular, limiting ourselves to a particular model (e.g., of peer assessment in Chapter 4), or family of models, allows us to uncover features of an application setting that appear to be a significant driver of many of our results. Such effects are more difficult to uncover when the particularities of an application are abstracted away.

Ultimately, the strongest evidence for the utility of the methodologies that we develop based on computational experiments are in the clear, unambiguous results that we obtain from applying them.

1.1 Preliminaries

1.1.1 Information Elicitation

The study of information elicitation is concerned with understanding the properties of (economic) mechanisms that are designed to collect *reports* about the private *signals* observed by agents in some population on behalf of a *principal*. As is common in economics, a *mechanism* is a function that collects reports from the agents—and, in some cases, its own public or private information—as inputs and then outputs *payments* as compensation for the reports. Payments are typically thought of as being monetary, but this is not required. When private information is costly for agents to acquire or to report, then incentives become a critical concern and a number of questions are raised: How can agents be incentivized to submit reports to a mechanism? Which mechanisms incentivize agents to expend the cost to collect useful information and report that information truthfully? Etc. The literature on information elicitation in economics has, accordingly, largely focused on questions of this type. However, it is important to emphasize that incentives are not the only relevant concern in many applications.

Information elicitation is typically modeled as a *game* in the sense invoked by "game theory": an interaction between multiple strategic actors—in this case the principal, who commits to a mechanism, and the agent(s), who may report strategically—where the outcome depends on the choices of all relevant parties. The primary solution concepts in game theory, and thus in information elicitation, are equilibria. Further, the assumptions made about the nature of the agents interacting with a mechanism are the typical assumptions made in game theory. The agents are assumed to be "rational" in the sense that they have a well-defined utility function that encodes their preferences over the possible outcomes, they assume a prior distribution over outcomes and update their beliefs according to Bayes' rule whenever new information is revealed, and they synthesize their beliefs and preferences in order to select actions that maximize their expected utility. It is worth keeping in mind that these assumptions are sometimes unrealistic, especially when real people, not economic agents, are the ones who will interact with a given mechanism.

Perhaps the most prominent paradigm in information elicitation is *forecasting*, in which the reports submitted are probabilistic forecasts of an event for which the outcome will eventually be publicly observable. Historically, in the application of weather forecasting, it was observed that verification schemes for assessing the quality of forecasts could sometimes lead a forecaster to want to report something other than their true private belief in order to try to "game" the verification scheme. Eventually, Glenn Brier proposed a solution to this problem—a new verification scheme in which it was always in the best interest of a forecaster (from their perspective) to report their true private beliefs [7]. Brier's solution was later discovered to be a special case of a class of mechanisms called *proper scoring rules* that take a probabilistic forecast and the outcome of the forecasted event as input and then output a score such that the forecaster will always (weakly) maximize their *ex ante* expected score by reporting their true belief. Subsequently, an elegant mathematical theory of proper scoring rules has been developed [28] and that theory has been applied in various settings, including in the development of *prediction markets*, which are discussed in Section 1.1.3.

In general, forecasting is a special case of the information elicitation *with verification* paradigm, in which it is assumed that the mechanism is able to collect its own reliable *ground truth* signals for (a subset of) the objects about which it will collect or has collected possibly unreliable reports from the agents. As in forecasting, the ground truth signals are usually used by the mechanism in order to verify agents' reports and determine their reliability. Unlike in forecasting, though, a mechanism in this paradigm does not generally collect ground truth signals for every *task* about which it collects agent reports. This is because in many settings, the ground truth is costly to collect, as opposed to forecasting where the ground truth is typically a publicly observable outcome. As a result, in many cases, the goal of using an information elicitation mechanism is to collect honest reports to use as inputs in an *information aggregation* scheme (Section 1.1.2) that can determine the ground truth (with high probability) for a large number of tasks at a much lower cost than would be necessary to compensate a single reliable expert to submit reports for the same set of tasks.

1.1.1.1 Peer Prediction

The paradigm for information elicitation *without* verification is called *peer prediction* [60]. In the peer prediction paradigm, mechanisms do not rely on access to ground truth signals in assigning payments. In many settings, this paradigm is necessitated by some practical limitation—ground truth may be prohibitively expensive to collect or, as in the case of opinion surveys, may not exist at all. However, it is also possible and may sometimes be beneficial to apply techniques from peer prediction to supplement mechanisms from information elicitation with verification.

Without ground truth, peer prediction mechanisms rely on collecting reports from multiple agents on each task and leveraging an assumption that agents' private signals are correlated in some way, so that it is feasible to verify an agent's report using the reports of their peers. Many state-of-the-art peer prediction mechanisms can be understood through an insightful information-theoretic framework described by Kong and Schoenebeck [47].

Generally, peer prediction mechanisms are characterized by two properties:

1. An *equilibrium concept* related to truthful reporting.
2. An *assumption* on the joint prior distribution of signals that is sufficient to guarantee inducement of the equilibrium concept.

Given that the sufficient assumption holds, an equilibrium concept describes circumstances for which, if an agent were to know the strategies of their peers, truthful reporting would (weakly) maximize their expected utility. The sufficient assumption describes the weakest known correlation structure for the joint prior distribution of signals under which there is a known mathematical proof that the equilibrium concept applies. We will revisit this characterization of peer prediction mechanisms in Chapter 4.

1.1.2 Information Aggregation

Information aggregation is often the impetus for information elicitation. The principal wants to harness the “wisdom of crowds,” but recognizes that aggregating reports that have been strategically manipulated is likely to interfere with that objective.

Similarly to information elicitation, information aggregation can be conceptualized as encompassing two distinct paradigms, which differ based on whether the notion of ground truth is relevant for a particular application. The most prominent paradigm for information aggregation is the paradigm in which the goal is to discern some ground truth. This paradigm is closely linked with the long-standing idea of the wisdom of crowds, the mathematical study of which dates back at least to the Condorcet Jury Theorem, proved in 1785 [17]. Hong and Page [35] describe the result in the following manner:

> The theorem applies to a group of voters who must identify or predict the correct answer from among two alternatives. In the canonical version, each voter independently knows the answer with the same probability. If that probability exceeds one-half, four results follow: the majority identifies correctly with a higher probability than each individual, collective accuracy increases in individual accuracy and in group size, and large groups approach but never achieve perfect accuracy. (Hong and Page 2015)

Modern research on the wisdom of crowds has expanded beyond the limited setting considered by Condorcet to include a wide array of settings, including empirical studies about the practical performance of crowds compared to individual experts [100; 106] and theoretical work on the performance of crowds with correlated signals [35]. In both theoretical and empirical work, the fundamental principle is that it is often more reliable to aggregate the opinions (or predictions, responses, etc.) of multiple people, even if those people are relatively non-expert, instead of relying on the opinion of a single expert. Watts [106] goes so far as to contend that this principle alone is sufficient for harnessing the wisdom of crowds, suggesting that empirical evidence indicates that aggregation itself is what matters and that (within reason) the particular method of aggregation is far less important.

Nevertheless, much of the literature on information aggregation is concerned with methods, looking for ways to improve over naive (but powerful) baselines like majority voting (for discrete responses from a small set of possibilities) or averaging (for continuous responses). For example, Prelec et al. [76] propose a method that selects the “surprisingly popular” response as the most likely true answer and show that, under certain assumptions, their method can correctly recover the ground truth even when the majority is wrong.

Information aggregation without ground truth is studied in *social choice theory* [5; 6]. In social choice theory, the information to be aggregated is generally some representation of the *preferences* of the individual agents participating in a mechanism over a set of possible outcomes. For example, the agents may want to collectively select a single winner from a set of candidates in an election. The goal is to design mechanisms that incentivize agents to report their preferences honestly and that select outcomes that achieve some notion of fairness with respect to those preferences. Voting rules are an important component of social choice theory and, as demonstrated by the Condorcet Jury Theorem, voting rules that arise in the context of aggregating preferences can also be applied as methods for recovering ground truth.

Lastly, note that, while it is useful as an abstraction to consider the study of information aggregation as bifurcated by the relevance of ground truth, there are cases that do not fit neatly into this account. In particular, there are important settings where it is possible to conceptualize the existence of ground truth in the abstract, but where, in reality, the "truth" is not objective or uncontested. For example, consider the task of labeling content from social media according to its level of "toxicity." Different groups within the population may each have a shared within-group understanding of what content would be deemed toxic, but their standards may conflict. That is, the "truth" may vary according to the culture of each group, as is modeled in *cultural consensus theory* (e.g., Batchelder et al. [4]). Consequently, as argued by Gordon et al. [30], aggregating information without properly accounting for inherent disagreement can create the illusion of consensus. This illusion of consensus can lead to problems in downstream applications, e.g., training a machine learning classifier to label content according to toxicity.

1.1.3 Social Computing and Human Computation

In applications, information elicitation and information aggregation are frequently coupled together as parts of a larger system. Such systems are often either *social computing systems* or systems that leverage *human computation* [70]. In social computing systems, the focus is on human interactions that are mediated by a computational system. For example, review sites and social networking sites largely derive their value from social interactions that they facilitate. In human computation systems, "computational" tasks that have proven difficult to solve with electronic computers are outsourced to humans to complete. The quintessential human computation system is Amazon's Mechanical Turk (MTurk), for which the conceit behind both the name and concept of the system is the appearance that the system is completely automated, despite the fact that there is actually an essential, hidden human component that makes the system work.

Theoretical work on information elicitation and aggregation has been instrumental in the design of some notable social computing and human computation systems. In the vision paper “Mathematical Foundations of Social Computing,” Chen et al. [13] discuss three prominent success stories of mathematical research applied to social computing: participatory budgeting, prediction markets, and fair division. Each of these stories is, at least in part, about information elicitation and aggregation. Participatory budgeting and fair division are both applications of social choice theory. Fair division fits less neatly into the notion of information aggregation discussed above, because the nature of the item being divided plays an important role, but it still involves aggregating the preferences of the agents using the system. Participatory budgeting, in which agents interact with a mechanism to “vote” on how to allocate a limited budget, is a more pure application of the idea of aggregating preferences. Prediction markets, moreover, are a triumph of the synthesis of information elicitation and aggregation and are deeply rooted in the mathematical theory of those disciplines.

Prediction markets allow agents to buy and sell securities that disburse a fixed payout depending on the outcome of a future event that is being forecasted. For example, in an election with two candidates, a market could involve securities that pay out $1 in the event that a particular candidate wins the election. The price of a security in the market can be interpreted as an aggregation of the individual forecasts of the agents participating in the market, i.e., a collective forecast. For example, if the price falls below the expected value that an agent assigns to the payout of the security—which is equal to the probability that the agent assigns to the outcome under which the security pays out—it is in their best interest economically to purchase securities until the price becomes equal to their expected value. Thus, a principal interested in predicting the outcome of some future event can run a prediction market in order to harness the wisdom of crowds and produce a reliable forecast. Prediction markets also apply ideas from information elicitation to correctly manage agents' incentives. The price in the market is set by an automated *market maker* whose behavior is derived from an application of the theory of proper scoring rules. As a result, the price is set correctly to elicit honest forecasts. Further, the market maker has bounded risk (it cannot lose an unbounded amount of money) and is not susceptible to arbitrage [13].

However, despite these remarkable success stories, the mathematical literature on information elicitation and aggregation—and in particular on peer prediction—has so far had a limited impact in two common purported applications: crowdsourcing and peer assessment.

1.1.3.1 Crowdsourcing

Crowdsourcing encompasses a variety of practices that leverage the power of a “crowd” of people to complete a task or series of tasks. Geiger et al. [26] use a systems-theoretic approach to abstract crowdsourcing tasks into four categories based on the nature of the interactions between the individuals participating in the system (Are contributions processed in isolation or are contributions valuable due to their interaction with those of other participants?) and whether the inputs are considered homogeneous (Are the inputs qualitatively the same, e.g., labels selected from a specified set, or different, e.g., distinct solutions proposed for a complex problem?). The contemporary techniques in information elicitation and aggregation domains are best-suited for two of the resulting categories: *crowd processing*, in which inputs are homogeneous and contributions are processed in isolation, and *crowd rating*, in which inputs are homogeneous and contributions are processed collectively. A related term from outside of the systems-theoretic paradigm that also describes these two categories is *microtask crowdsourcing*, which is used to distinguish simple, relatively straightforward tasks like labeling from more complex, intellectually-intensive tasks like submitting a solution to a machine-learning competition and open-ended, creative tasks like writing.

In academic research, MTurk is the most popular crowdsourcing platform. It is often used for conducting experiments, including studies of the role of incentives in the quality and quantity of work completed that help to motivate the need for incentive-compatible information elicitation mechanisms for social computing and human computation tasks [34; 58].

1.1.3.2 Peer Assessment

Peer assessment is a pedagogical technique in which students are asked to give feedback on the work of their peers and, consequently, receive feedback from their peers about their own work. First and foremost, peer assessment is associated with myriad benefits for students [56; 97]. Additionally, it can benefit a course's instructional staff, particularly in a Massive Open Online Course (MOOC). Due to large class sizes in MOOCs, it is often intractable for the instructors alone to provide timely and productive feedback. Peer assessment can be done for its own sake, but it can also take the form of *peer grading*, a special case of peer assessment in which student feedback is used directly to compute grades for student submissions. It is reasonable to expect that in many cases, students will have some intrinsic utility for giving helpful feedback to their peers, even if it is somewhat costly. However, if students are assigned grades for the quality of their feedback in a peer assessment system, then extrinsic incentives—the kind primarily studied in information elicitation (and game theory more broadly)—become more salient.

1.2 Overview

This dissertation features an ensemble of three works applying simulation-based methods to the evaluation of information elicitation mechanisms and information aggregation algorithms. We design and implement these methods to develop empirical predictors of their performance in two important applications—crowdsourcing and peer grading. In doing so, we frequently introduce new dimensions of analysis along which to predict performance and argue that these new dimensions are important complements to the existing ones.

1.2.1 Part 1: Crowdsourcing

In Chapter 2, we explore the following question: Do common assumptions about the way that crowd workers make mistakes in microtask (labeling) applications manifest in real crowdsourcing data? Prior work only addresses this question indirectly. Instead, it primarily focuses on designing new label aggregation algorithms, seeming to imply that better performance justifies any additional assumptions. However, empirical evidence in past instances has raised significant challenges to common assumptions. We continue this line of work, using crowdsourcing data itself as directly as possible—using non-parametric and parametric simulation techniques for statistical hypothesis testing—to interrogate several basic assumptions about workers and tasks. We find strong evidence that the assumption that workers respond correctly to each task with a constant probability, which is common in theoretical work, is implausible in real data. We also illustrate how heterogeneity among tasks and workers can take different forms, which have different implications for the design and evaluation of label aggregation algorithms. As a whole, the insights that we uncover about the plausibility (or implausibility) of various assumptions establishes a firmer foundation for future work both in the theoretical and the empirical study of label aggregation.

The work that is presented in this chapter was jointly authored with Grant Schoenebeck and appeared in the proceedings of the 39th Conference on Uncertainty in Artificial Intelligence (UAI 2023) [10].

1.2.2 Part 2: Peer Assessment

In Chapter 3, we turn our attention to peer grading. Despite the many potential benefits of peer grading, the traditional grading paradigm, where one instructor (or their assistant) grades each submission, is still predominantly used. One persistent impediment to adopting a new grading paradigm is doubt that the new paradigm will assign grades that are at least as good as the grades that would have been assigned under the existing paradigm.

We address this impediment by using tools from economics to define a practical framework for determining when peer grades clearly exceed the standard set by the instructor grading baseline. We demonstrate the efficacy of our framework in simulated experiments using a realistic Bayesian model of peer grading from prior work. We find that, according to that model, instructors set a high standard for grading quality that is difficult for student graders to definitely outperform. However, we also find that the concrete objective functions defined within our framework prove useful for (1) evaluating interventions in the peer grading system to improve grading quality and (2) unearthing trade-offs in the problem of allocating grading work between instructor and student graders when there is a limited grading budget. Although our framework only addresses one aspect of the larger problem of designing, implementing, and promoting the adoption of reliable peer grading systems—i.e., assigning high-quality numerical scores to submissions—it frames that aspect in a unique way that we believe will prove useful in other areas like evaluating the quality of written feedback provided by peer graders.

The work that is presented in this chapter was jointly authored with Yichi Zhang and Grant Schoenebeck.

In Chapter 4, we propose *measurement integrity*, which quantifies a mechanism's ability to assign rewards that reliably measure agents according to the quality of their reports, as a novel desideratum in many natural applications. Like *robustness against strategic reporting*, the property that has been the primary focus of the peer prediction literature, measurement integrity is an important consideration for understanding the practical performance of peer prediction mechanisms.

We perform computational experiments, both with an agent-based model and with real data, to empirically evaluate peer prediction mechanisms according to both of these important properties. Our evaluations simulate the application of peer prediction mechanisms to peer assessment—a setting in which *ex post* fairness concerns are particularly salient. We find that peer prediction mechanisms, as proposed in the literature, largely fail to demonstrate significant measurement integrity in our experiments. We also find that theoretical properties concerning robustness against strategic reporting are somewhat noisy predictors of empirical robustness. Further, there is an apparent trade-off between our two dimensions of analysis. The best-performing mechanisms in terms of measurement integrity are highly susceptible to strategic reporting. Ultimately, however, we show that supplementing mechanisms with realistic parametric statistical models can, in some cases, improve performance along both dimensions of our analysis and result in mechanisms that strike the best balance between them. Altogether, our approach and the lessons that we derive from it constitute a significant step toward transforming peer prediction mechanisms from interesting theoretical objects to practical tools for information elicitation in real applications.

The work that is presented in this chapter was jointly authored with Grant Schoenebeck and appeared in the proceedings of the 24th ACM Conference on Economics and Computation (EC'23) [9].

Part I

Crowdsourcing

CHAPTER 2

Testing Conventional Wisdom (of the Crowd)

2.1 Introduction

As a whole, crowds can be surprisingly wise [100], but individuals within the crowd are unsurprisingly prone to making mistakes. This is true for complex applications like prediction markets—where individuals collaborate to create a probabilistic forecast for some event by trading securities linked to particular outcomes of that event. And it is equally true for simpler applications like microtasks—where individuals collaborate to complete a simple task, like labeling an image, by performing the task individually and submitting their responses to be algorithmically aggregated in a way that will discern the correct label from the various responses with high probability. Given this fallibility, then, it is important to understand how workers make mistakes.

Clearly, how workers make mistakes on individual tasks has important implications for the design of aggregation algorithms. These algorithms frequently leverage insights that flow from assumptions about how errors are made. For example, some algorithms, like those proposed by Burnap et al. [8] and Welinder and Perona [109] rely on the assumption that there are expert workers, who label more accurately than a typical worker. Under this assumption, a clear aggregation strategy emerges: identify the experts, then take the majority answer from the expert labels.

But how workers make mistakes on individual tasks also has important implications for the *evaluation* of aggregation algorithms. When an algorithm is tested on a group of data sets, the degree to which the results of those tests will be indicative of performance on future data depends on the degree to which the test data sets are representative of the future data. The (approximate) validity of basic assumptions about how workers make errors are important dimensions along which test data may or may not be representative of future data, because those assumptions are a key factor in the design of aggregation algorithms.

Further, it is important to understand the nature of individual mistakes in order to quantify uncertainty about labels. Uncertainty quantification can be used, e.g., as in active learning, to train classifiers that achieve a given level of accuracy at a lower cost than otherwise [11; 71; 88]. More specifically, the estimated parameters of an error model can also be used to improve the utility of a labeled data set as a training set for a machine learning algorithm, as shown by Lalor et al. [49], and to improve the reliability of crowdsourced experiments, as shown by Katsuno et al. [43].

The importance of understanding errors, thus, leads to a natural question: What evidence is there in real crowdsourcing data for the various assumptions that underlie the common error models? Surprisingly, work towards answering this question is largely absent from the literature. The most direct evidence for the utility of an error model is typically just that an algorithm based on it outperforms previous algorithms in aggregating labels to recover the ground truth. However, there are many factors that confound the relationship between an underlying model and the performance of an algorithm that is designed using that model, including the choice of test data sets. As a result, assessing the validity of modeling assumptions solely through algorithmic performance offers a limited view. It is insufficient for supporting any theoretical claims that follow from those assumptions. Moreover, it is insufficient for understanding how representative a group of test data sets is of future data and for assessing the utility of the various error models for uncertainty quantification.

We address these limitations by directly exploring the degree to which there exists evidence in real data sets to justify several common assumptions about worker errors. In doing so, we uncover some regularities that hold across a diverse collection of data sets alongside much variation in other fundamental characteristics. Specifically, we find that:

- Errors depend on the category of each task's correct response. In particular, workers do not have a constant probability of correctness for all tasks (Section 2.4).
- Whether errors appear to depend on factors beyond the correct response category varies (Sections 2.5.1 and 2.5.2).
- Worker proficiency distributions are well-characterized by (generally reliable) modal workers (Section 2.5.2).
- Exceptionally reliable "expert" workers do not appear to play a significant role (Section 2.5.2.2).

2.1.1 Related Work

Challenging Assumptions. In this work, we test common assumptions about crowdsourcing workers and tasks using publicly-available data to help guide future research and practice. Prior work includes three particularly notable examples that do exactly that. Yin et al. [118] uncover a robust network of communication among crowdsourcing workers. A key lesson of their analysis is that the group of workers that complete a task on Amazon Mechanical Turk (a popular crowdsourcing platform) is *not* a random sample of all active workers, because workers talk to their peers about tasks that are enjoyable, lucrative, etc. More directly in the domain of label aggregation, Li et al. [51] demonstrate that the common assumption that the average number of labels per task is small (≤ 3) often does not hold. Rather, in the publicly-available data sets they identify, the average number of labels per task is commonly at least 5. Further they show that, in this label-dense setting, many state-of-the-art aggregation algorithms perform no better than a simple majority vote, despite being much more computationally expensive. Lastly, Wei et al. [108] argue that certain noise models from the image classification literature fail to adequately describe real-world noise in two new benchmark data sets that they introduce for image classification tasks.

In testing the assumptions that we consider, we rely crucially on an additional assumption that is typical in crowdsourcing—that the underlying tasks have an objective ground truth category that can be recovered by aggregating labels. This assumption makes sense for the tasks that we consider, which are relatively simple to complete and in many cases have objectively correct responses. However, it is not an appropriate assumption for all crowdsourcing tasks. Recent work, e.g., by Basile et al. [3],Gordon et al. [30], and Plank [75], has explored alternative approaches to working with crowdsourced data in those settings where it does not make sense to assume that tasks have an objective ground truth.

Aggregating Labels. The assumptions that we focus on in this work are generally implicit in the error models that are employed in the design of label aggregation algorithms. Just two such families of error models are nearly ubiquitous in the literature, and they imply different assumptions about the heterogeneity of tasks and workers:

1. Dawid-Skene models (Section 2.2.1) assume that a worker's errors on a task depend primarily on the correct response for the task and their own proficiency [15; 42; 53; 54; 78; 109; 126].

2. Item response theory models (Section 2.2.2) generally assume that tasks, independently of the correct response, follow a particular pattern of heterogeneity that affects a given worker's probability of responding correctly in specific ways [2; 44; 111; 113].[1]

The most common approach in designing a label aggregation algorithm is to adopt a

[1] Otani et al. [67] also use a model based on item response theory in the related setting of pairwise comparisons.

model from one of these families. But there are a few prominent exceptions to this trend. For example, Zhou et al. [127, 128] adopt a very flexible error model under which each task-worker pair is associated with its own distribution over the possible responses. This model makes relatively few assumptions about the nature of workers and tasks, but as a result, also has limited utility for extrapolating to unseen examples. Another unique approach by Jung and Lease [41] is to apply probabilistic matrix factorization, a collaborative filtering technique, to predict labels from each worker on all tasks before aggregating the predicted labels via majority vote or some other algorithm.

Bayesian Annotation Models. In this work, we seek to validate or invalidate assumptions using data itself, and therefore to be agnostic to specific aggregation algorithms or models, as much as possible. However, certain outcomes are difficult to distinguish without an underlying model. For example, is a certain group of workers with high accuracy a group of experts, or were they just assigned easy tasks? As a result, in Section 2.5.2.1 we fit models from the Dawid-Skene and item response theory families and use the parameters of the best-fitting models to further explore the data. In that section, our work resembles that of Paun et al. [72] and Lakkaraju et al. [48], who each consider sets of Bayesian annotation models and evaluate their utility for various estimation and prediction tasks, including label aggregation.

Although Paun et al. and Lakkaraju et al. draw from essentially the same families of models that we consider, our work has significant methodological differences. And, ultimately, we apply models toward a different end—to answer specific questions about the data themselves and how they relate to common assumptions from the label aggregation literature, rather than to answer questions about the relative utility of the models for solving problems where the data is an input or qualitative questions about interpreting the parameters of more complex models.

2.2 Modeling

The fundamental elements of an error model are *tasks* and *workers*. In our setting, a task is an object that is associated with a collection of *categories* or labels with a fixed, finite size k. Among these categories, exactly one applies to the object. That category is called the *ground truth*. For example, a task might be an image of a duck, with the categories "Duck" and "No Duck." In that case, "Duck" is the ground truth. A worker's job is to select the ground truth category that applies to the object for each task assigned to them. Each instance of a worker selecting a category for a task is called a *response*.

2.2.1 Dawid-Skene

The most popular models in the label aggregation literature are Dawid-Skene (DS) models, which were proposed as a way to understand and mitigate individual errors in clinical diagnoses [15]. In a DS model, the interactions between workers and tasks are parameterized by a collection of *confusion matrices*. A confusion matrix M is a $k \times k$ stochastic matrix. Entry m_{ij} denotes the probability of a worker reporting category j on a task for which the ground truth is i. Typically, each worker is associated with their own confusion matrix, but variants of that basic model include models where a single confusion matrix is shared among a cluster of workers or among the entire population.

Intuitively, DS models suppose that the probability of a particular worker making an error on a particular task can depend on that task's ground truth category, but only on that. In our running example, that means that duck images and non-duck images may have different patterns of errors, but every image of a duck (and every image that is not of a duck) is more or less equally recognizable as such. We decompose this into two distinct assumptions: The first is that the pattern of errors is *category-dependent*. The second is that tasks with the same ground truth are *homogeneous*.

2.2.2 Item Response Theory

Item Response Theory (IRT) [19; 79] was developed in psychometrics for the purpose of designing tests (e.g., academic assessments) and interpreting their results. In contrast to DS models, IRT models parameterize both workers *and* tasks. Each worker i is characterized by an *ability* parameter θ_i, which may be a scalar or a vector. The value of θ_i represents i's adeptness at the underlying skill being "measured" by a particular test, i.e., set of tasks. Each task j is characterized by up to three scalar parameters: a discrimination parameter a_j, a difficulty parameter b_j, and a "guessing" parameter $c_j \in [0, 1]$. The worker and task parameters interact in the following way to determine the probability of a correct response on task j from worker i:

$$\Pr[\text{correct}] = c_j + (1 - c_j)\,\text{expit}\left(a_j\left(\theta_i - b_j\right)\right), \tag{2.1}$$

where $\text{expit}(x) = \frac{\exp(x)}{1+\exp(x)}$ is the standard inverse-logit (i.e., logistic) function. In this function, *ceteris paribus*, the discrimination parameter controls the rate of change in the probability of correctness as the worker's ability varies. The guessing parameter captures the intuition that, with a finite number of categories, it is possible to produce the correct response without identifying it by responding randomly.

Equation (2.1) captures the three basic IRT models. The difference between these models is that the simpler models impose stronger constraints on the task parameters. In the One-Parameter Logistic Model (1PL), $a_j = a$, a constant, and $c_j = 0$ for all tasks j. In the Two-Parameter Logistic Model (2PL), $c_j = 0$ for all tasks j. In the Three-Parameter Logistic Model (3PL), all of the task parameters are allowed to vary.

On the whole, IRT models suggest that the probability of a particular worker making an error on a particular task comes down to an interaction between the characteristics of the worker and the characteristics of the task. However, the characteristics of tasks that are assumed to be relevant are different under IRT than under DS. In particular, the ground truth category is generally not taken into consideration.

2.3 Data

Given a worker and a task, the DS model predicts a complete response distribution over the possible categories. IRT predicts a probability that the response will be correct, but does not predict which category will be chosen if the response is not correct. To remove this asymmetry, we limit our analysis to data sets with binary categories, so that specifying a probability of correctness is equivalent to specifying a complete distribution over the categories. In addition to binary categories, we also require the data sets to have ground truth labels. This introduces an important assumption we make throughout: that ground truth labels are sufficiently reliable.

The Statistical QUality Assurance Robustness Evaluation (SQUARE) ([98]) from Sheshadri and Lease [89] project, a benchmarking resource for label aggregation research, provides six data sets that meet our criteria[2] and are meant to be used to evaluate label aggregation algorithms:

- **BM** involves labeling the sentiment of tweets as either positive or negative [62; 63].
- **HCB** involves determining whether a particular Web page is relevant to a given search query [98].
- **RTE** involves *textual entailment*, i.e., deciding whether a given statement implies a subsequent one [96].
- **TEMP** involves deciding two events' temporal order [96].

[2] It also lists a seventh data set, SpamCF, which appears to meet our criteria, but upon closer inspection only contains ground truth categories for tasks where the workers were in unanimous agreement.

	Workers	Tasks	Responses	
			gt = 0	gt = 1
BM	83	1000	2545	2455
HCB	722	3267	8767	10932
RTE	164	800	4000	4000
TEMP	76	462	2590	2030
WB	39	108	2340	1872
WVSCM	17	159	1219	731
SP	143	500	4900	5100

Table 2.1: Summary of each crowdsourced data set.

	MAD	95% CI	Med TS	Max TS	p
BM	0.382	(0.318, 0.476)	0.100	0.167	0.001
HCB	0.364	(0.333, 0.471)	0.166	0.197	0.001
RTE	0.138	(0.131, 0.200)	0.088	0.111	0.001
TEMP	0.085	(0.050, 0.167)	0.061	0.119	0.020
WB	0.408	(0.319, 0.550)	0.058	0.179	0.001
WVSCM	0.238	(0.148, 0.436)	0.079	0.179	0.001
SP	0.065	(0.053, 0.091)	0.049	0.071	0.006

Table 2.2: Summary of randomization inference results: Testing null hypothesis of category independence.

- **WB** involves labeling images by whether they contain a certain kind of bird [109].
- **WVSCM** involves labeling whether images of smiles are of "Duchenne" smiles [113].

Several of these data sets are no longer available through the SQUARE project site, so we provide alternative links [61; 95; 110].

We also consider the following additional data set, which was released subsequently to the SQUARE project:

- **SP** involves labeling the sentiment of a sentence extracted from a movie review as either positive or negative [101].

2.4 Category-Dependent Errors

We begin with a model-agnostic, non-parametric approach. We apply randomization inference—also known as a permutation test—to the hypothesis that the errors in the data from our various sources are *not* category-dependent. Specifically, in each of 999 (unique) permutations, we randomly assign the ground truth category for each task (while preserving the size of each category) and then compute each worker's frequency of correctness conditioned on the (assigned) ground truth category. The test statistic is the median absolute difference between these frequencies. To obtain an exact p-value for this hypothesis test, we compute the number out of all 1000 computed medians[3] that are at least as extreme as the median observed under real categories.

Using this test, we find very strong evidence to *reject* the null hypothesis that errors in each data set are not category-dependent. For nearly every data set, the median observed under the real categories is the most extreme, corresponding to a p-value of 0.001 for our test. In the remaining datasets, TEMP and SP, the observed median is still quite extreme, corresponding to p-values of 0.020 and 0.006, respectively.[4] As a result, it is apparent that, even for this diverse collection of tasks with binary categories, category matters a great deal in determining the pattern of errors in crowdsourcing data.

In addition to being statistically significant, the dependence on categories is also *practically* significant. In Table 2.2, we show the median absolute difference (MAD) between frequencies of correctness conditioned on the ground truth categories for each data set and estimate a 95% confidence interval for these values via bootstrap resampling. We emphasize that these values are *absolute* differences. Large values do not necessarily imply that one category is substantially easier than the other; workers can differ in the category for which their responses are more accurate. Then, we compare the true median absolute differences—our test statistic (TS)—to the median and maximum values of the test statistics observed in the permuted data during our randomization inference. In this comparison, the true value of the test statistic, and even the lower bound of the confidence interval, is often much greater than the maximum value of the test statistic observed in any permutation.

2.5 Task & Worker Heterogeneity

In this section, we explore the degree to which tasks and workers exhibit heterogeneity with a variety of approaches. For tasks, we say they are heterogeneous if the probability of a correct response from a worker tends to vary with the underlying task (and homogeneous otherwise). For workers, we say they are heterogeneous if the probability of a correct response on a task tends to vary with the worker who is providing the response (and homogeneous otherwise).

[3] 999 under permutations of the ground truth categories and 1 under the real ground truth categories from the data.

[4] Further, if the mean absolute difference in frequency of correctness is used in place of the median as the test statistic, then the observed mean is the most extreme value (and, thus, $p = 0.001$) for every data set.

2.5.1 Model-Agnostic Analysis

Heterogeneity in Tasks. Previously, we found that there is strong evidence that tasks are heterogeneous based on their category. The next question we consider is whether tasks are homogeneous—i.e., whether workers have a constant probability of correctness—within each category. We once again employ randomization inference to test the null hypotheses that tasks within each ground truth category are homogeneous. Consequently, we perform two hypothesis tests in each data set—one per category. For these tests, our test statistic is the difference in the mean (DiM) frequency of correct responses between apparently easy tasks and apparently difficult tasks in the given category. The apparently easy and apparently difficult tasks are the upper and lower half, respectively, of the set of all tasks in that category when sorted according to their fraction of correct responses. We perform the randomization inference by (uniquely) permuting the identifiers of the tasks in the list of responses within the given category (999 times). This preserves the number of times each task appears in the set of all responses, but changes which workers are associated with which tasks. We obtain exact p-values as above. The results of these tests are displayed in Table 2.3.

For most categories, in most data sets, this test suggests rejecting the null hypothesis of homogeneity. However, in contrast to our randomization inference for categories, there is some reason to be skeptical of the practical significance of some of the results, even when they appear statistically significant. The values of the test statistics for the permutations are surprisingly consistent, even to the point of being nearly invariant for category 0 in the BM data set. As a result, there are certain values for which the difference between the true value of the test statistic in the real data and the values observed in the permutations (as summarized by the median and maximum values in Table 2.3) are quite small, even though the value in the real data is the most extreme value (and thus the associated p-value is small). For example, the true value in category 1 for both the HCB and RTE data sets is less than 0.06 more than the median of the values from the permutations. This suggests that, although we may reject the null hypothesis of homogeneity, the actual difference between homogeneity and the particular kind of heterogeneity that appears to be present in those data sets may not be very meaningful. We will return to this point in Section 2.5.2.1, when we test the fit of models with different assumptions about task heterogeneity. In particular, we find that the results of this test are fairly predictive of the results of model fitting.

Heterogeneity in Workers The assumption that is perhaps the most ubiquitous in label aggregation—which is rarely even explicitly acknowledged as an assumption—is that workers vary in their proficiency, e.g., by having different probabilities of correctness than other workers (when completing a task in a given category). This assumption, however, is not

	gt	DiM	Med TS	Max TS	p
BM	0	0.235	0.235	0.235	0.998
	1	0.603	0.346	0.386	0.001
HCB	0	0.437	0.354	0.376	0.001
	1	0.338	0.292	0.309	0.001
RTE	0	0.242	0.239	0.265	0.412
	1	0.244	0.193	0.217	0.001
TEMP	0	0.142	0.227	0.262	1.000
	1	0.186	0.194	0.228	0.723
WB	0	0.176	0.111	0.150	0.001
	1	0.266	0.125	0.174	0.001
WVSCM	0	0.351	0.228	0.281	0.001
	1	0.419	0.208	0.281	0.001
SP	0	0.184	0.102	0.119	0.001
	1	0.210	0.109	0.125	0.001

Table 2.3: Summary of randomization inference results: Testing null hypothesis of task homogeneity.

universal. For example, the image classification error models that are discussed by Wei et al. [108] assume that workers are homogeneous.

To explore this assumption, we once again employ randomization inference. This time, we test the null hypothesis that workers are homogeneous when completing tasks in the same category. The test is very similar to our randomization inference concerning heterogeneity of tasks in Section 2.5.1. We perform two hypothesis tests in each data set—one for each category. For these tests, our test statistic is the difference in the average frequency of correct responses between apparently more proficient workers and apparently less proficient workers in the given category. The apparently more proficient workers and less proficient workers are the upper and lower half, respectively, of the set of all workers when sorted in order of each worker's fraction of correct responses in the given category. To perform the randomization inference, we (uniquely) permute the identifiers of the workers within the given category, thereby preserving the number of times each worker appears in the set of all responses, but changing which tasks are associated with which workers (999 times). Lastly, to obtain an exact p-value, as in Section 2.5.1, we calculate the number of test statistics out of 1000 that are at least as extreme as the true test statistic from the original data. The results of these tests are displayed in Table 2.4.

We find that for each data set, in at least one category, there is strong evidence to reject the null hypothesis that workers are homogeneous. Unexpectedly, there are some data sets (BM, HCB, and TEMP) for which this does not hold for both categories. However, there

	gt	DiM	Med TS	Max TS	p
BM	0	0.226	0.188	0.316	0.103
	1	0.469	0.384	0.456	0.001
HCB	0	0.649	0.534	0.564	0.001
	1	0.391	0.430	0.468	0.999
RTE	0	0.273	0.215	0.261	0.001
	1	0.229	0.183	0.220	0.001
TEMP	0	0.254	0.237	0.307	0.197
	1	0.359	0.236	0.304	0.001
WB	0	0.381	0.089	0.130	0.001
	1	0.462	0.113	0.164	0.001
WVSCM	0	0.398	0.175	0.282	0.001
	1	0.318	0.183	0.297	0.001
SP	0	0.178	0.134	0.204	0.009
	1	0.188	0.138	0.201	0.002

Table 2.4: Summary of randomization inference results: Testing null hypothesis of worker homogeneity.

are reasons to interpret this result cautiously. As always, lack of evidence against the null hypothesis does not necessarily constitute evidence for it; in this case, we believe the null hypothesis is *a priori* unlikely. A possible explanation for these results that would not necessarily support the null hypothesis would be that for some categories, in some data sets, workers did not complete enough tasks for it to be clear that they have heterogeneous proficiency.

Another way in which the results from this test are somewhat weaker than those from our randomization inference about task heterogeneity is that, as we will see in Section 2.5.2, they are not obviously corroborated by our model-informed analysis. In the plot of logit-probabilities of correctness (Figure 2.1), the BM data set does appear to have the least dispersion among the distributions of logit-probability of correctness, but it is also dense in a region where probability of correctness changes quickly with changes in logit-probability of correctness. The HCB and TEMP data sets, on the other hand, have relatively high dispersion. This does not necessarily contradict the results of our tests—logit-probability of correctness incorporates worker proficiency in both categories, whereas the randomization inference indicates a lack of support for heterogeneity in just one category in each data set. However, it does not clearly corroborate the results either.

2.5.2 Model-Informed Analysis

To further investigate task and worker heterogeneity, we need to move beyond our model agnosticism. Without a model, it is not possible to distinguish between, for example, a group of expert workers who completed a standard set of tasks and a group of average workers who completed a set of particularly easy tasks. In contrast to prior modeling work, though, we employ models as a means to an end—to answer further questions about the data itself. As a result, we employ the standard version of each model. This allows us to perform exact inferences and to make minimal assumptions, while still capturing the essential features of the model.

2.5.2.1 Finding the Best Fit

We seek to identify the standard DS or IRT model that provides the best fit to each data set. Then, we can use the estimated parameters of those models to answer deeper questions about the data. In light of our results from Section 2.4, though, IRT models have an obvious shortcoming—they generally do not incorporate category-dependent errors, except in the special case where the same category is more difficult to label than the other for every worker. To address this shortcoming, we also consider an extension of IRT: Category-dependent Item Response Theory (CIRT), which is similar to the model proposed by Khattak et al. [44]. For CIRT, we split each data set into two parts by conditioning on the ground truth and fit the standard IRT models to each part independently.

Estimating DS Parameters. Because all of our data sets contain ground truth categories, fitting the DS model is quite straightforward. We use the Maximum Likelihood Estimate (MLE) given in the original paper by Dawid and Skene [15]. For each worker, their confusion matrix is completely determined given the diagonal entries, which represent the conditional probabilities of answering correctly given each ground truth category. The estimate for each of these entries is simply that worker's empirical frequency of correctness on the tasks they completed in that category. For practical purposes, we must augment these estimates in two ways. First, if a worker did not complete any tasks in a particular category, we use the population-level frequency of correctness for that category as the estimate. Second, to avoid undefined quantities in our model comparison techniques, we hedge extreme estimates $\hat{p} = 0$ or $\hat{p} = 1$ in the following manner:

$$\hat{p}_h = \frac{1}{2n} + \frac{(n-1)}{n}\hat{p},$$

where n is the number of tasks (and h stands for *hedged*).

Estimating IRT Parameters. Our model fitting techniques for IRT models similarly take advantage of the ground truth labels. Unlike in label aggregation generally, this is the standard setting for IRT—when you are grading a test, you generally need to know the answers. The standard algorithm for fitting an IRT model is to use a *marginal maximum likelihood* approach [19; 84]. In this algorithm, the item parameters are estimated first by computing an MLE while marginalizing over a population-level distribution of ability parameters that is estimated from the data using a quadrature method. Then, the ability parameters are estimated using MLE given the item parameter estimates.

A major assumption underlying IRT model-fitting procedures is that the correct dimension for the ability parameters is specified. We assume these parameters are unidimensional. Although tests to indicate whether ability parameters in a given data set are plausibly multidimensional have been proposed, those methods are designed for settings where nearly all participants respond to nearly all items. They do not readily generalize to crowdsourcing settings where each worker tends to only complete a small subset of the tasks.

We also limit ourselves to the 1PL and 2PL models for IRT and CIRT. Fitting the 3PL model is too computationally expensive in our data sets, which are large compared to typical IRT data. Further, a limitation of our model fitting software [84] is that it is not possible to specify or learn a constant value (i.e., 0.5) for the guessing parameter c when using the model fitting methods for the 1PL and 2PL. Thus, c is fixed at the default value of 0 for our experiments.

Lastly, for our largest data set, HCB, fitting the 2PL and Category-dependent Two-Parameter Logistic Model (C2PL) models is too computationally expensive for 10-fold cross validation (see below). Thus, for that comparison, we limit ourselves to the 1PL models for IRT and CIRT. However, we do use the 2PL models for our other model comparison, the Bayesian information criterion (see below).

Comparing Models. There is no perfect method to compare fit among models, particularly those belonging to different model families. Thus, we apply two different procedures: Ten-Fold Cross Validation (10FL) and the Bayesian Information Criterion (BIC) [27, Ch. 7].

10FL involves splitting the tasks into 10 equal-sized parts. For each part i, we fit each of the models on the 9 other parts and use the estimated parameters to make predictions about the probability of correctness for each worker-task pair in part i. These individual predictions are scored using the *quadratic scoring rule.* If there is a particular worker-task pair for which there is no data from the other parts on which to estimate parameters for the worker, that pair is skipped. Finally, models are evaluated using the sum total of their scores for all individual predictions in all 10 parts.

BIC is an adjusted Log-Likelihood (LLH) measure that penalizes the inclusion of additional parameters:

$$\mathrm{BIC} = k \log(n) - 2\,\mathrm{LLH},$$

where k is the number of parameters and n is the size of the data set, i.e., the total number of responses.

The results of these comparisons are summarized in Table 2.5. We find that our two comparison procedures tend to agree, giving us more confidence that we are selecting the best model. We put slightly more weight on cross-validation than BIC, so for the one data set (BM) where there is disagreement, we select the Category-dependent One-Parameter Logistic Model (C1PL) as the best-fitting model. Our results indicate that the data sets differ in terms of how useful it is to model characteristics of tasks beyond the ground truth category. The DS model providing the best fit indicates that tasks are more or less homogeneous, whereas the C1PL model indicates that there is heterogeneity. Given this understanding, it is noteworthy that the results of our randomization inference from Section 2.5.1 do a fairly good job of predicting the results of our model fitting. The data sets that we find are best fit by the DS model include TEMP, for which our randomization inference suggested we should not reject the hypothesis of homogeneity. Further, RTE and HCB are also best fit by the DS model. In those data sets, we found evidence of heterogeneity, but also found reason to question the practical significance of that apparent heterogeneity. Lastly, in both tests, the results for the BM data set are somewhat mixed.

Employing models, though, allows us to do more than just corroborate the results of our previous test. It allows us to gain additional perspective on task heterogeneity beyond what was possible with our model-agnostic analysis. In particular, we find that, even when there is evidence that tasks are heterogeneous, the complexity of that heterogeneity appears limited—the additional discrimination parameters in the 2PL and C2PL models do not improve model fit.

2.5.2.2 Examining Experts

We can also use our parameter estimates from the best-fitting models to investigate heterogeneity among workers. A convenient one-dimensional summary of a worker's proficiency is their *logit-probability of correctness*, which can be estimated from the parameters of the best-fitting model. For the DS model, probability of correctness is estimated as the sum over all categories of the product of the estimated probability of correctness for that category and its empirical frequency in the set of tasks. For the C1PL model, probability of correctness

	10FL	BIC
BM	C1PL	DS
HCB	DS	DS
RTE	DS	DS
TEMP	DS	DS
WB	C1PL	C1PL
WVSCM	C1PL	C1PL
SP	C1PL	C1PL

Table 2.5: Summary of model fitting results: Best-fitting model for each data set.

is estimated using a Monte Carlo method. First, for each ground truth category of tasks, we compute a Kernel Density Estimation (KDE) for the distribution of difficulties. Then, 500 total samples are drawn from these distributions[5], in proportion to the empirical frequency of the ground truth categories. For each sample, we use the IRT eq. (2.1) to estimate a probability of correctness. Then, we average the probability of correctness over all 500 samples. Lastly, applying the logit function to the estimated probabilities of correctness is a convenient transformation, because it extends the range of the values from $[0, 1]$ to $(-\infty, \infty)$.

Kernel density estimates of the distributions of logit-probability of correctness, where bandwidths are selected using Silverman's rule [86; 93], are displayed in Figure 2.1. For the data sets best fit by the DS model—HCB, RTE, TEMP—we remove outliers at the extreme values. The extreme values for the most part represent workers who responded correctly to every task they completed.[6] We are comfortable removing these workers as outliers, because their extreme estimated logit-probabilities of correctness are very likely to be an illusion of chance and sparse data. We can substantiate this intuition with the following resampling procedure: Fit a normal distribution to the logit-probabilities excluding extreme values (i.e., the max values in all three data sets and the min value in HCB). For each worker in the data set, draw a logit-probability of correctness from the fitted normal distribution. Then, sample a number of correct responses from a binomial distribution with the corresponding probability of correctness where the number of trials is equal to the number of tasks that worker completed in the data. Using this procedure, it is common to observe that both the number of extreme values and the average number of correct responses from the corresponding workers is greater than in the real data.

These estimated distributions offer insights into the validity of a key assumption of many crowdsourcing works—that there are *expert* workers. Who should count as an expert is

[5] We reuse the same 500 samples for each worker.

[6] For HCB, there are also workers who responded incorrectly on each of their tasks, whom we remove for analogous reasons.

	Dip Test	BW Test
BM	0.698	0.110
HCB	**<0.001**	**0.037**
RTE	**<0.001**	0.324
TEMP	**0.028**	0.379
WB	**0.011**	0.224
WVSCM	0.970	0.192
SP	0.259	0.616

Table 2.6: Summary of modality test results: Testing null hypothesis of unimodality.

a somewhat ill-defined concept in the literature. Sometimes, experts are a distinct group of participants apart from crowdsourcing workers, who are thought or known to be more reliable. We are more interested in experts within the crowd. But there are still questions about who, if anyone, should be counted as an expert. Is it any worker of above-average proficiency? Or is there something more distinct about an expert?

The first thing to note is that nearly all of the distributions in Figure 2.1 appear to be somewhat left-skewed. In such distributions, considering an above-average worker to be an expert seems inappropriate—the modal worker is above average. Further, the BM, SP, and WVSCM data sets each appear to have one prominent mode, after which the densities drop off relatively steeply. Thus, even if we were to set some threshold to the right of the mode and to consider any workers beyond the threshold to be experts, the density is small enough that expertise would appear to be relatively insignificant in these data sets. WVSCM is a possible exception. Its shape is very similar to SP, but it is centered in a region where the inverse logit function changes much more quickly. Thus, the relative difference in probability of correctness between a modal worker and a worker in the right tail in WVSCM is greater than that in SP, which may justify considering expertise as more significant in the WVSCM data.

The remaining data sets are all at least plausibly multimodal. Statistical hypothesis tests for unimodality of the empirical distributions of logit-probabilities (not of the KDEs for those distributions shown in Figure 2.1)—calibrated versions of Hartigan's dip test and Silverman's bandwidth test [38; 39]—corroborate this visual intuition. The results of these statistical tests are presented in Table 2.6. Specifically, plausible multimodality (i.e., the rejection of the null hypothesis of unimodality) under these tests indicates that the smaller apparent modes would be unlikely to result from random chance if the true underlying distributions were unimodal.

Like the unimodal distributions, the plausibly multimodal distributions are mostly left-skewed, with the right-most apparent mode being the largest. The distributions drop off less

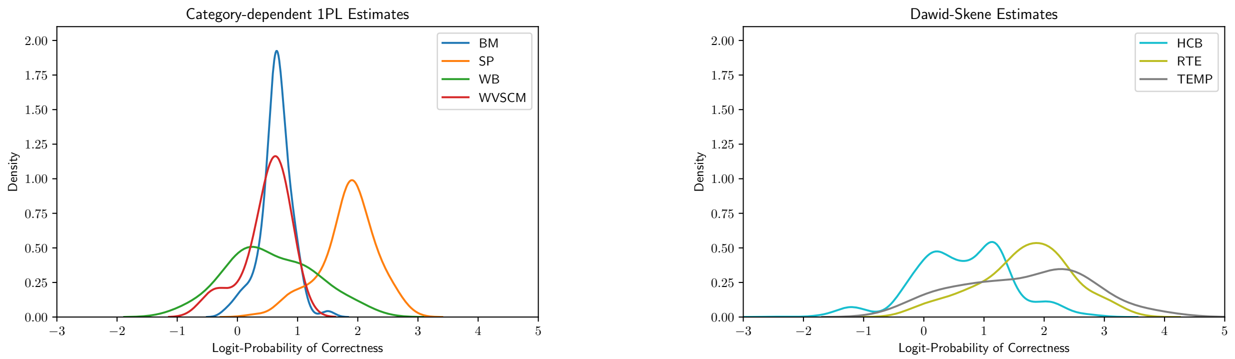


Figure 2.1: Kernel density estimates of the distributions of logit-probability of correctness in each data set.

steeply—they are more dispersed than the unimodal distributions. However, these larger tails are mostly in regions where the inverse logit function changes less quickly, so the change in probability of correctness that occurs in the larger tails is less significant. The WB distribution is the exception to these trends. There, the left-most apparent mode is the largest. Moreover, the distribution is centered in a region where the inverse logit function changes quickly. Thus, the right-most apparent mode can be considered a significant cluster of expert workers, distinct from the cluster of workers near the larger mode. We call this phenomenon *strong expertise* to distinguish it from the weaker notion of experts who are in the upper tail of the largest (apparent) mode.

2.5.2.3 Worker Heterogeneity Beyond Multi-Modality

Multi-modality (or plausible multi-modality) in the distribution of logit-probability of correctness suggests that workers are heterogeneous, i.e., they have different probabilities of correctness. In testing the null hypothesis of unimodality for distributions of logit-probability of correctness, however, there were three data sets (BM, WVSCM, and SP) for which the evidence did not suggest that we should reject the null hypothesis of unimodality. Those three data sets were all fit best by the C1PL model. So, for those distributions, we use the C1PL model to construct a model-informed test of the null hypothesis of heterogeneity that does not involve modality.

The test is a model-informed resampling procedure. First, we estimate the parameters of the C1PL model using marginal maximum likelihood estimation (as in Section 2.5.2.1). Then, we resample each worker's responses to each task that they responded to in the real data set according to the estimated C1PL model (999 times). The parameters for each task

	Observed Variance	Med TS	Max TS	p
BM	0.065	0.035	0.069	0.001
WB	0.481	0.030	0.057	0.001
WVSCM	0.135	0.037	0.128	0.001
SP	0.220	0.095	0.154	0.001

Table 2.7: Summary of model-informed resampling results: Testing null hypothesis of worker homogeneity.

in that model are assumed to be those that were estimated from the data. The ability parameters in each category for each worker in that model are assumed to be equal to the *average* of the ability parameters in that category that were estimated from the data. Thus, workers are assumed to be homogeneous.

Using the simulated data from each round of resampling, we estimate the empirical distribution of logit-probability of correctness as in Section 2.5.2.2. For our test statistic, we use the variance of the distribution of logit-probability of correctness. Thus, we compare the variance of the simulated distributions to the value for the variance that we observe in the real data.

Results are presented in Table 2.7. In all three data sets, the observed variance is more extreme than the variance of any distribution resulting from simulation under the null hypothesis of homogeneity ($p = 0.001$). Thus, the results of this test provide evidence against that null hypothesis. (Additionally, the result is the same for the WB data set, which was also fit best by the C1PL model, but was found to be plausibly multimodal above.)

2.5.2.4 Diabolical Tasks

In a setting with strong expertise, a natural question arises. How much are experts worth relative to a regular worker? In many cases, if it is costly to recruit or identify experts, then doing so might not be worth it. Aggregating the responses from a few non-expert workers may be cheaper and just as, if not more, accurate. However, it is not difficult to imagine cases where experts provide additional value. For example, they may have domain-specific knowledge that non-experts do not possess that leads them to produce correct responses even when the majority of non-experts fails to do so. That is, there may be cases where the aggregation of non-experts will fail to identify the correct category, but an expert will succeed. More generally, we refer to the kind of task where non-experts tend to respond incorrectly, but experts tend to respond correctly, as a *diabolical task*.

We search for possible diabolical tasks in the WB data set. First, we fit a Gaussian Mixture Model [94] to the logit-probabilities of correctness that we computed in Section 2.5.2.2

	Category-Dependent	Task Heterogeneity	Worker Heterogeneity		Expertise
	Errors	(Intra-Category)	Model-Agnostic	Model-Informed	
BM	Very Strong	Moderate	Moderate	Moderate	Weak
HCB	Very Strong	Moderate	Moderate	Strong	Moderate
RTE	Very Strong	Weak	Strong	Moderate	Weak
TEMP	Strong	Weak	Moderate	Moderate	Weak
WB	Very Strong	Strong	Strong	Moderate	Strong
WVSCM	Very Strong	Strong	Strong	Moderate	Weak
SP	Strong	Strong	Strong	Moderate	Weak

Table 2.8: Characterizing data sets based on strength of evidence for assumptions in experimental results.

in order to classify workers as either experts or non-experts. Then, we look for tasks that meet the following criteria:

1. At least two experts and non-experts completed the task.

2. A majority of non-experts produced an incorrect response.

3. A majority of experts produced a correct response.

There are 27 tasks that meet these criteria—25% of all tasks. This is a substantial number, but there are a few unusual features of the WB data set that may somewhat temper its significance. Most importantly, the relative frequency of experts is quite high. As a result, it is not uncommon for the majority of all workers to respond correctly even when the majority of non-experts responds incorrectly. This occurs for 17 out of the 27 apparently diabolical tasks. Also, relative to modal workers in the other data sets, the non-expert workers in WB perform fairly poorly. These mitigating factors suggest that the significance of diabolical tasks for label aggregation in this particular data set is likely narrow. More generally, diabolical tasks may be a bigger factor in settings where experts are a population distinct from crowd workers and, thus, may be more likely to differ from crowd workers in systematic ways.

2.6 Discussion

In Table 2.8, we summarize our results intuitively in terms of the strength of evidence for various assumptions we find in each data set.[7] Below, we discuss key implications of those pieces of evidence and their significance for future work:

[7] Table 2.8 gives an intuitive summary of our results; the precise meanings of the terms we use in it are discussed in the Appendix A.2.

Workers make errors that are category-dependent. In notable theoretical work [42; 54; 66], it is assumed that workers have a constant probability of correctness. Our results present a challenge to extend theoretical results beyond this simple setting. If provable guarantees are important for designing better algorithms, then those guarantees should be proven under more realistic assumptions. Our results also indicate that, when invoking the IRT model family, it is wise to adopt a CIRT-style model that allows for category-dependent errors, e.g., as is done by Khattak et al. [44].

Tasks with the same ground truth category may or may not be heterogeneous. When they are heterogeneous, that heterogeneity appears to have limited complexity. Some data sets were best fit by a CIRT model, others by DS. But when CIRT provided the best fit, it was always the least complex model—C1PL. This suggests that when there is heterogeneity within categories of tasks—i.e., when workers do not have a constant probability of correctness per category—the differences within categories can be represented simply.

Workers appear heterogeneous, with distributions of proficiency that are generally characterized well by the modal workers. Exceptionally reliable "expert" workers do not appear to play a significant role. In the supplementary material, our model-agnostic analysis finds evidence of worker heterogeneity in one (moderate evidence of heterogeneity) or both (strong evidence) categories of each data set. In our model-informed analysis, where we consider the distributions of logit-probability of correctness, workers in each data set exhibit clear heterogeneity. However, many of the distributions have densities that drop off relatively quickly from the largest mode, suggesting that even the most reliable workers do not report correctly with much higher probability than a relatively typical worker.

Relying on the existence of experts who can be reliable even when the majority is unreliable may be misguided. Overall, we find that it is often the case that the most reliable workers are not much more reliable than a relatively typical (modal) worker. Further, it can be argued in some cases that the improvement in probability of correctness for an "expert" worker does not fully compensate for their decreased frequency in the population. For example, consider a single expert worker, who is more proficient than a modal worker, and whose logit-probability of correctness corresponds to a density that is about one third of the density at the largest (approximate) mode according to the KDE for the distribution of logit-probability of correctness. If such an expert is less likely to produce a correct response than a majority of 3 workers, each with the (approximate) modal logit-probability of correctness, then the additional value provided by the expert worker may not be worth the additional cost of identifying them. Moreover, the modal workers are often both reliable and plentiful, meaning that their responses can be aggregated into very reliable labels. This corroborates the work of Li et al. [51], who find that majority vote is a powerful aggregation algorithm on real crowdsourcing data.

No set of assumptions universally characterizes the data sets that we consider. As a result, hierarchical (Bayesian) models like those of Lakkaraju et al. [48] and Paun et al. [72], which have hyperparameters to capture the degree of diversity across tasks and workers, are likely to be useful. These models can learn whether workers or tasks are completely homogeneous, completely heterogeneous, or something in between. For example, partitioning workers or tasks into a small set of homogeneous clusters may effectively capture the diversity among them. Hierarchical models infer these kinds of relationships directly from the data when estimating model parameters. Further, we note that our results offer some guidance in applying the hierarchical approach. For example, our results suggest that it would be reasonable to adopt a Gaussian prior for a logit-probability of correctness parameter (or, equivalently a logit-normal prior for a probability-of-correctness parameter), as long as category-dependent errors are properly incorporated.

Moreover, the diversity among data sets that we uncover suggests that the degree to which tasks and workers are heterogeneous is something that should be tested rather than assumed when working in a new domain. Understanding the amount (and form) of heterogeneity has important implications for designing or selecting an aggregation algorithm—since candidate algorithms should be tested on a group of representative data sets—and for subsequently quantifying uncertainty in aggregated labels. A concrete next step for future work is to test state-of-the-art label aggregation algorithms on groups of test data sets—including, but not necessarily limited to, those that we consider—that have apparently similar characteristics according to the evidence we summarize in Table 2.8 and document the extent to which relative algorithmic performance varies among the groups.

Part II

Peer Assessment

CHAPTER 3

Understanding When Peer Grades (Definitely) Outperform Instructor Grades

3.1 Introduction

Peer grading is a useful tool for furthering many worthy goals. At the pedagogical level, it can help students achieve a deeper understanding of the subject matter and increase the amount of feedback that students receive. At the administrative level, it can reduce the demand on instructor time, reducing costs, and thereby increasing access to education. However, there is an important constraint at the heart of peer grading that can be a bottleneck to accruing the benefits associated with it. As Zarkoob et al. [124] state:

> In order for peer grading systems to be both useful to instructors and acceptable to students, they must produce grades that are sufficiently similar to those that an instructor would have given. (Zarkoob et al. 2023)

Our perspective on this constraint is slightly different: We believe that peer grading systems need not produce grades that are similar to those than an instructor would have given, so long as they produce grades that are *better* than those that an instructor would have given. Importantly, this perspective shift allows us to maintain a baseline standard for the quality of peer grades, while also acknowledging the possibility that the instructor grades themselves may leave room for improvement.[1]

According to Bachelet et al. [1], there are mixed results in the literature concerning comparison between the quality of (individual) peer grades and instructor grades. However, even when peer grades are high-quality, there is an additional complication—the *perception* of the quality of peer grades. As Johnston [40] concludes, in the context of MOOCs:

[1] Although, in the setting of our experiments, we find that instructor grades set a high standard for grading quality, we think it is useful to include this as a possibility when framing the challenges faced by a prospective peer grading system.

> MOOCs serve a wide cognitive diversity of students in MOOCs [*sic*], which leads many students to not respect their peers as "qualified" to evaluate their work. (Johnston 2015)

In accordance with this new perspective, and the challenges inherent in both the reality and the perception of high-quality peer grading, we design a framework for evaluating the quality of an alternative grading system (e.g., peer grading) relative to the current standard (e.g., instructor-only grading) based on ideas from economics. We frame the question of whether peer grades are better than (or at least as good as) instructor grades in terms of forecasting: Are the predictions that result from observing the outcome of a peer grading system at least as good as the predictions that result from observing one instructor grade for each submission? On one hand instructors tend to be more reliable graders, but on the other hand with peer grading we can obtain multiple grades for each submission, so it is not clear in advance which system should be preferable. We use tools from the literature on evaluating forecasts to answer our question. In particular, we use the idea of stochastic dominance applied to forecast evaluation scores to define a family of objectives that, when they are achieved, would imply that peer grades are definitely better than instructor grades for agents with a wide array of utilities (Section 3.3).

Thinking in terms of forecasting is a natural fit for this problem, because much of the prior quantitative work on peer grading has focused on using Bayesian inference. At a high level, Bayesian inference combines observed data with a probabilistic model of how the data were generated to produce a forecast of the true values of each unobserved parameter in the model. These parameters may, for example, capture how "reliable" or "effortful" each grader is. By facilitating inference about these parameters, the model allows for a deeper understanding of the grading process, including, in many cases, an improved forecast of the quality of the submissions.

We apply our framework in the setting of the model proposed by Zarkoob et al. [124] (Section 3.4.2). We find that, for realistic simulated data,[2] it is rarely the case that peer grades are definitely better than instructor grades without imposing a significant workload burden on students. However, having concrete objectives for evaluating grading systems is useful, even when those objectives are difficult to achieve, because we can evaluate various interventions in the basic peer grading system based on the relative improvement towards achieving the objectives that we propose (Section 3.5) and identify trade-offs that arise in simultaneously optimizing multiple objective functions (Section 3.6).

[2] i.e., data that are simulated using a model and associated hyperparameters that were found to fit observed data well in terms of held-out likelihood [121; 124]

3.1.1 Our Contributions

1. We propose that being "at least as good as grading by one instructor" is a useful standard for confirming (and convincing stakeholders of) the quality of peer grading.
2. We provide a practical definition of being "at least as good as grading by one instructor" in terms of the quality of forecasts for the true grades of submissions that we can compute based on reported grades from each system. We show that different assumptions about utility for forecast scores correspond to different criteria under which peer grading is preferable.
3. We use our forecasting framework to conduct computational experiments using a realistic model of peer grading to establish that, in our setting:
    (a) Various peer-focused interventions, e.g., assigning graders strategically to avoid assigning only low-quality graders to some submissions, do not improve the performance of peer grading systems in our setting, but supplementing peer graders with a limited budget of instructor grades can improve performance significantly.
    (b) The optimal way of allocating a fixed budget of peer and instructor grades depends on the criteria—either comparing the average quality of grades assigned by the system or comparing the quality of the worst grades—under which peer grading is preferable to instructor grading.

3.1.2 Related Work

Evaluating Prediction. Our primary contribution is using instructor grades to design a benchmark for evaluating a peer grading system. In terms of benchmark design, Resnick et al. [81] propose the idea of *rater equivalence* (originally, *survey equivalence*) which uses the true label aggregated from multiple human labels as a benchmark to evaluate the accuracy of a classifier. In particular, the survey equivalence of a classifier is the minimum number of human raters needed to produce the same error as that of the classifier. Our main idea is to evaluate peer grading systems by checking whether (an analogue of) rater equivalence is at least one. A similar idea from Erev et al. [21] has been used to evaluate the prediction made by a theoretical model (e.g., the equilibrium outcome of a game). The model is compared with the minimum required number of prior observations of human participants playing the game to make as accurate a prediction as the model. In line with these ideas, we are interested in developing a framework for quantifying the number of peer grades per submission that is sufficient to be reliably at least as good as one instructor grade.

Bayesian Models for Peer Grading. A key component of our work involves using Bayesian inference to compute probabilistic forecasts for the true grades of submissions given a set of reported peer grades. A large family of related Bayesian models of peer grading have been proposed for this task. The initial models in this family—$\mathbf{PG}_1$, $\mathbf{PG}_2$, and $\mathbf{PG}_3$—were introduced by Piech et al. [74]. Model $\mathbf{PG}_1$ assumes that each submission has an underlying true grade and that each grader has a bias and a reliability parameter that interact with a submission's true grade to produce their reported grade for that submission. The other models introduce additional complexity: $\mathbf{PG}_2$ allows the graders' biases to vary over time and $\mathbf{PG}_3$ allows the graders' reliability to be correlated with their own submissions' true grades. Subsequently, Mi and Yeung [59] introduced models $\mathbf{PG}_4$ and $\mathbf{PG}_5$, which permit non-linear correlations between the graders' reliability and their true grades. Han et al. [32] propose models $\mathbf{PG}_6$, $\mathbf{PG}_7$, and $\mathbf{PG}_8$ for peer grading in small private online courses (SPOCs). They propose using knowledge tracing techniques to estimate an individualized value for a hyperparameter related to grading ability for each student prior to grading. Independently, Wang et al. [103] introduced their own distinct models $\mathbf{PG}_6$ and $\mathbf{PG}_7$, which incorporate the relative grades provided by the same grader for different assignments during inference. Most recently, Zarkoob et al. [124] conducted a comprehensive exploration of the key concepts that a peer grading model should capture. They find that the following two previously unexplored features are important for providing more accurate grades: (1) incentivizing grader effort by explicitly modeling it and (2) incorporating the uncertainty that arises from reported grades being censored data (because grading rubrics are essentially always finite and discrete). We adopt the resulting model, which is shown to outperform other variants in the PG family in terms of held-out likelihood on real classroom grading data, for our experiments. Zarkoob et al. do not give this model a name, so we will refer to it as model $\mathbf{PG}_Z$.

3.2 Model $\mathbf{PG}_Z$

Let $\mathcal{V}$ be the set of graders, and $\mathcal{U}$ be the set of submissions. Submissions are graded on one component, which is scored with an integer in the set $Z = \{0, 1, \dots, M\}$. The function $\mathcal{M} : \mathcal{U} \to \mathcal{V}$ maps each submission u to its corresponding set of graders.

Given these preliminaries, an instance of the model is specified by assigning values for each of the hyperparameters: $\big(\mu_s, \tau_s, \alpha_\tau, \beta_\tau, \tau_b, \alpha_e, \beta_e, \tau^\ell, \epsilon\big)$. The model itself, in which we follow the convention of writing random variables as uppercase letters and realizations of those variables as the corresponding lowercase letters, is as follows:[3]

$$\begin{aligned}
\text{(True grades)} \quad & S_u \sim \mathcal{N}\left(\mu_s, 1/\tau_s\right); \\
\text{(Reliabilities)} \quad & T^v \sim \mathcal{G}\left(\alpha_\tau, \beta_\tau\right); \\
\text{(Biases)} \quad & B^v \sim \mathcal{N}\left(0, 1/\tau_b\right); \\
\text{(Effort probabilities)} \quad & E^v \sim \text{Beta}\left(\alpha_e, \beta_e\right); \\
\text{(Efforts)} \quad & Z_u^v \sim \text{Ber}\left(e^v\right); \\
\text{(Peer grades)} \quad & G_u^v \sim \begin{cases} \mathcal{N}\left(s_u + b^v, 1/\tau^v\right), & z_u^v = 1; \\ D_\ell, & z_u^v = 0; \end{cases} \\
\text{(Reported grades)} \quad & r_u^v = n_Z\left(g_u^v\right).
\end{aligned}$$

In these definitions, n_G is a function that rounds a real number to the nearest integer in the set Z (rounding up). D_ℓ is a low-effort distribution:

$$D_\ell = \begin{cases} \mathcal{N}\left(\mu_s, 1/\tau^\ell\right), & \text{with probability } 1 - \epsilon; \\ \text{Uniform}\left(0, M\right), & \text{with probability } \epsilon. \end{cases}$$

Lastly, we need to specify the model for instructor grades. Instructors have reliability and bias parameters just like peer graders, but the hyperparameters of the prior distributions for those parameters (α_τ^I, β_τ^I, and τ_b^I) may be different. Also unlike peer graders, instructors' effort probabilities are fixed to 1. Given these parameters, the distributions of instructor grades (analogous to peer grades) and of reported grades for instructors follow the same form as that for peer graders.

3.2.1 Hyperparameters

We set the hyperparameters of model $\mathbf{PG}_Z$ to the following values:

- True grades: $\mu_s = 4$, $\tau_s = 1.5625$;
- Reliabilities: $\alpha_\tau = 2$, $\beta_\tau = 2$; $\alpha_\tau^I = 6$, $\beta_\tau^I = 2$;
- Biases: $\tau_b = 4$; $\tau_b^I = 100$;
- Effort probabilities: $\alpha_e = 8$, $\beta_e = 2$;
- Low-effort distribution: $\tau^\ell = 4$; $\epsilon = 0.05$.

[3] Note that for reliabilities, the parameter is tau, so the corresponding lowercase letter is τ.

Zarkoob et al. [124] selected these hyperparameter settings heuristically based on experiments optimizing for the model's the held-out likelihood on each of four individual courses' worth of peer grading data [121].

Using these hyperparameters, we simulate grading data according to model $\mathbf{PG}_Z$ for a single assignment that is graded on one component. Reported grades are restricted to be from the set $\{0, 1, 2, 3, 4, 5\}$. There are 120 submissions (one from each of 120 students). The number of peer grades reported by each student grader (which is equal to the number of peer grades per submission) varies in the experiments.

3.2.2 Inference

We use the same Gibbs sampling procedure as Zarkoob et al. [124] to perform inference on model $\mathbf{PG}_Z$. In contrast to Zarkoob et al., though, we perform inference on submissions and peer grades from a single assignment, rather than an entire semester's worth of assignments. After inference, we approximate the posterior distribution (i.e., the forecast) for the true grades of the submissions given reported peer grades using a multivariate Gaussian distribution fit using maximum likelihood estimation and use that approximation when evaluating the forecasts.

3.3 Scoring Forecasts

Recall our motivating question from Section 3.1: Are the predictions that result from observing the outcome of a peer grading system at least as good as the predictions that result from observing one instructor grade for each submission? In order to answer this question, we must first be able to evaluate individual forecasts.

A commonly-used measure for the performance of a (point) estimator that is computed based on a probabilistic forecast is the *squared error* of the resulting estimate.

Definition 3.3.1 (Squared Error (SE))**.** Suppose that we are given a probabilistic forecast (i.e., a probability density function) $p \in \Delta(\mathbb{R})$ for the outcome of a real-valued random variable X. Further, suppose that $f : \Delta(\mathbb{R}) \to \mathbb{R}$ is an estimator of X.

Given the outcome $X = x$, the *squared error* of the estimate $\hat{x} = f(p)$ is

$$\mathrm{SE}(\hat{x}, x) = (\hat{x} - x)^2.$$

When translating a forecast p into a point estimate, the mean of the forecast is optimal for minimizing the expected squared error (see, e.g., Jaynes [37, p. 172]). Thus, when we subsequently refer to evaluating forecasts with squared error, we will assume that the point estimate used for computing the squared error is the forecast mean, unless otherwise specified.

We can also evaluate a forecast directly, without depending on a particular estimator, by applying a *proper scoring rule*. Proper scoring rules are used to evaluate the quality of probabilistic forecasts (and to elicit probabilistic forecasts in an incentive-compatible way) [23, Chapter 2], [55, Section 2]. In principle, any proper scoring rule can be used to evaluate a forecast. In the case of our experiments, though, the log scoring rule in particular is well-motivated and computationally convenient.

Definition 3.3.2 (Log Scoring Rule)**.** Suppose that $p \in \Delta(\mathbb{R})$ is a probabilistic forecast for the outcome of the real-valued random variable X. Given the outcome $X = x$, the *log scoring rule* $S : \mathbb{R} \times \Delta(\mathbb{R}) \to \mathbb{R}$ is defined as:

$$S(x, p) = \log\left(p(x)\right).$$

These two different approaches to evaluating forecasts turn out to be closely related in the context of our experiments, because the forecasts that we are interested in can be approximated by Gaussian distributions. In the case that a forecast follows a Gaussian distribution (with a constant standard deviation), there exists a bijection between the two forecast evaluations that we have introduced:

Proposition 3.3.3. *Suppose X is a real-valued random variable and $p = \mathcal{N}(\mu, \sigma^2)$ is a forecast for X. Given the outcome $X = x$,*

$$S(x, p) = -\log\left(\sqrt{2\pi}\sigma\right) - \frac{1}{2\sigma^2}\, SE(\hat{x}, x),$$

where $\hat{x} = \mu$ is the expected value of X under the forecast p.

Proof.

$$\begin{aligned} S(x, p) &= \log\left(p(x)\right) \\ &= \log \frac{1}{\sqrt{2\pi}\sigma} \exp\left(-\frac{1}{2}\left(\frac{x-\mu}{\sigma}\right)^2\right) \\ &= -\log\left(\sqrt{2\pi}\sigma\right) - \frac{1}{2}\left(\frac{x-\mu}{\sigma}\right)^2 \\ &= -\log\left(\sqrt{2\pi}\sigma\right) - \frac{1}{2\sigma^2}\,\mathrm{SE}(\hat{x}, x). \end{aligned}$$

□

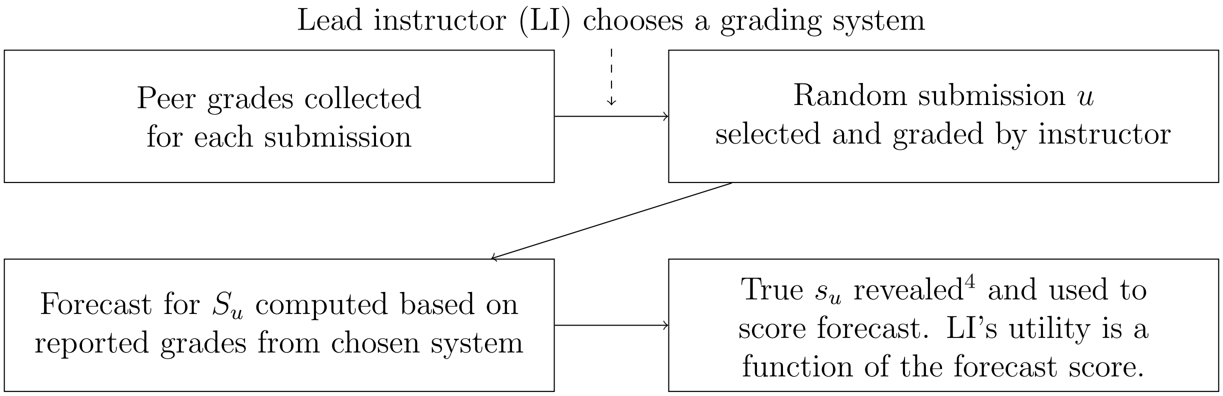

Lead instructor (LI) chooses a grading system
Peer grades collected for each submission
Random submission u selected and graded by instructor
Forecast for S_u computed based on reported grades from chosen system
True s_u revealed[4] and used to score forecast. LI's utility is a function of the forecast score.

Figure 3.1: A decision problem for the lead instructor of the course, wherein their utility is determined by a choice between forecasts based on reported peer grades or reported instructor grades. In particular, it is equal to the score for forecast computed using reported grades from their chosen system. We consider a peer grading system to be as good as a single-instructor grading system whenever the lead instructor prefers to choose peer grading based on their utility for forecast scores in the decision problem.

This result reveals an important insight: The squared error of a Gaussian forecast is closely related to its log score. But, there is a crucial distinction—the squared error depends on the mean, μ, but not the variance, σ^2, of the forecast $p = \mathcal{N}(\mu, \sigma^2)$, whereas the log scoring rule depends on both parameters.

3.4 Comparing Forecasts from Peer and Instructor Graders

We can use our tools for evaluating individual forecasts—squared error and proper scoring rules—to define a concrete setup for comparing peer and instructor grades. We construct our framework for comparison as an (economic) decision problem faced by the lead instructor (e.g., a professor) of a course where their utility is determined by a choice between peer grading and instructor grading. The problem is described in Figure 3.1.

Given the setup of the decision problem, we consider a peer grading system to be as good as a single-instructor grading system whenever the lead instructor prefers to choose peer grading based on their utility for forecast scores in the decision problem.

[4] Note that we use access to the true grade, which is available in simulations, but not in the real world, to evaluate the quality of forecasts. To apply this methodology with real world data, a reliable proxy for the true grade would need to be used instead. For example, if it is possible to obtain reports from two instructors independently, the reports from the second instructor may be used as a proxy. Alternatively, as in this work, real data can be used to select realistic hyperparameters, e.g., as described by Zarkoob et al. [124], for experiments with simulated data.

3.4.1 Forecast Dominance

When the lead instructor chooses a grading system in the problem described in Figure 3.1, uncertainty about (1) the submission that will be selected and (2) the grade that will be reported by the instructor for the selected submission means that the lead instructor is effectively choosing a lottery (i.e., a distribution of forecast scores) from which their utility will be drawn. Specifying assumptions about the lead instructor's utility function for forecast scores, then, allows us to characterize the conditions under which the lead instructor will prefer to choose peer grading.

Moreover, weaker assumptions about the lead instructor's utility for forecast scores correspond to stronger evidence that peer grades are at least as good as instructor grades when the conditions under which the lead instructor prefers to choose peer grading in the context of their decision problem hold. In particular, they allow for the possibility that the lead instructor may, for the sake of fairness, care especially about the relative quality of grading in the worst case under each system, not just on average. Thus, we begin with the weakest possible assumptions on the lead instructor's utility in order to delineate the conditions that correspond to the strongest possible evidence for the quality of peer grades. We will show that such conditions can be defined using the concept of stochastic dominance.

Let S_A (respectively, S_B) be a real-valued random variable with the Cumulative Distribution Function (CDF) F_A (resp., F_B).

Definition 3.4.1 (First-Order Stochastic Dominance (FOSD))**.** We say that S_A *first-order stochastically dominates* S_B if

$$F_A(x) \leq F_B(x),$$

for all $x \in \mathbb{R}$ and there exists some x for which the inequality is strict.

Definition 3.4.2 (Second-Order Stochastic Dominance (SOSD))**.** Suppose that the expectations of the absolute values of S_A and S_B are finite. We say that S_A *second-order stochastically dominates* S_B if

$$\int_{-\infty}^{a} F_A(x)\,dx \leq \int_{-\infty}^{a} F_B(x)\,dx,$$

for all $a \in \mathbb{R}$ and there exists some a for which the inequality is strict.

The usefulness of the concept of stochastic dominance is illustrated by the following: Suppose the lead instructor's utility is a function of some score, denoted by $u(s)$ and, without loss of generality, suppose higher scores are preferred. Then, the following is a well-known result:

Proposition 3.4.3 (Mas-Colell et al. [57]). *Let S_A and S_B be the random variables representing the scores for forecaster A and B respectively.*

1. *The lead instructor prefers forecaster A over B for any monotone increasing utility function u if and only if S_A first-order stochastically dominates S_B.*

2. *The lead instructor prefers forecaster A over B for any concave increasing utility function u if and only if S_A second-order stochastically dominates S_B.*

We extend the concept of stochastic dominance to the context of the lead instructor's decision problem as follows. We say that peer grading first-order (resp., second-order) forecast-dominates the single-instructor grading system if the lead instructor's utility for choosing peer grading (which recall, when the choice is made, is a random variable drawn from a distribution of forecast scores) first-order (second-order) stochastically dominates their utility for choosing instructor grading.

3.4.2 Determining Forecast Dominance in Model $\mathbf{PG}_Z$

To understand the circumstances under which peer grading forecast-dominates one-instructor grading in model $\mathbf{PG}_Z$, in what follows, we simulate the decision problem described in Figure 3.1 under a variety of different conditions, according to the following procedure:

1. Generate a set of n submissions and corresponding reported peer grades.

2. Compute a forecast for the true grade of each submission based on the complete set of reported peer grades.

3. Compute a forecast for the true grade of any submission given each possible value of a single reported instructor grade for that submission. (These forecasts, which are derived in Appendix B.1, are the same for each submission, since they are *a priori* identical.)

4. Then, repeat the following for k iterations:[5]

   (a) For each submission, sample an instructor report based on its true score and the instructor hyperparameters given in the model. (This report is a counterfactual value that would be observed if that submission were selected as the random submission and graded by an instructor.) Then, retrieve the forecast computed in Step 3 corresponding to the sampled report.

[5] For our experiments, $k = 1000$.

(b) For each submission, compute the score (e.g., log score or squared error) for the peer forecasts and the instructor forecasts respectively using that submission's true score.

(c) Compute empirical distributions of scores (over all n submissions) for the peer forecasts and the instructor forecasts, and determine whether the former distribution stochastically dominates the latter.

5. Estimate the probability that the peer grading scheme forecast-dominates the instructor scheme using the empirical frequency of stochastic dominance in the k repetitions of Step 4.

This procedure returns probabilities of first- and second-order peer forecast dominance corresponding to each set of generated submissions and corresponding peer grades. Thus, when we run multiple simulations, as in Section 3.4.3, we obtain two distributions of probabilities. Rather than aggregate these distributions into the average probabilities of first- and second-order peer forecast dominance, we preserve the entire distributions themselves. This is motivated by the fact that we think that this will be the most useful approach in practice. Lead instructors will want to verify the quality of realized peer grades on actual submissions, not just the prospective quality of future peer grades on future submissions. It is also useful for determining whether the average probabilities are useful summaries of the distributions.

Another feature of our approach to note is that our simulations operate as if a different instructor graded each problem. In fact, there may be just one instructor tasked with grading all of the submissions, or grading work may divided among several instructors. The effect of simulating as if a different instructor graded each problem is two-fold. First, our forecast scores based on instructor grades are likely to be a little lower than they might be otherwise. This is because, when a single instructor grades multiple submissions, the system has more information available to learn about that instructor's grading parameters, which may lead to slightly better true grade forecasts. Second, more extreme possibilities for instructor grading become less likely. If every submission were graded by the same instructor, the quality of the forecasts based on that instructor's reported grades would be correlated. An above-average instructor (in terms of grading quality) would tend to produce better forecasts; a below-average instructor would tend to produce worse forecasts. On the other hand, when a different instructor grades each submission, it is unlikely that all of the instructors would share the same grading tendencies.

Our simulation operates this way for two reasons. First, it is not clear that there is a single ideal approach. There are multiple ways to generate counterfactual instructor reports in a way that accords with our description of the lead instructor's decision problem, including our approach, and each has its own benefits and drawbacks. As discussed above, one alternative would be simulating entire sets of reported grades from single instructors. But this would allow, for example, some sets of reported instructor grades to come entirely from low-quality instructors. Second, there are computational limits on what approaches are feasible. Another alternative (and perhaps the most natural) approach would be to define the decision problem so as to compare forecasts based on an entire set of observed peer grades to forecasts based on an entire set of instructor grades (rather than just one observed instructor grade). In that case, the reported instructor grades would be correlated based on the instructor's grading parameters and an inference method like Gibbs sampling would need to be used to compute forecasts of the true grades. Then, to account for uncertainty about the instructor's parameters in the results of our experiments, this would need to be repeated many times. This is not computationally tractable for the experiments we conduct in this work.[6]

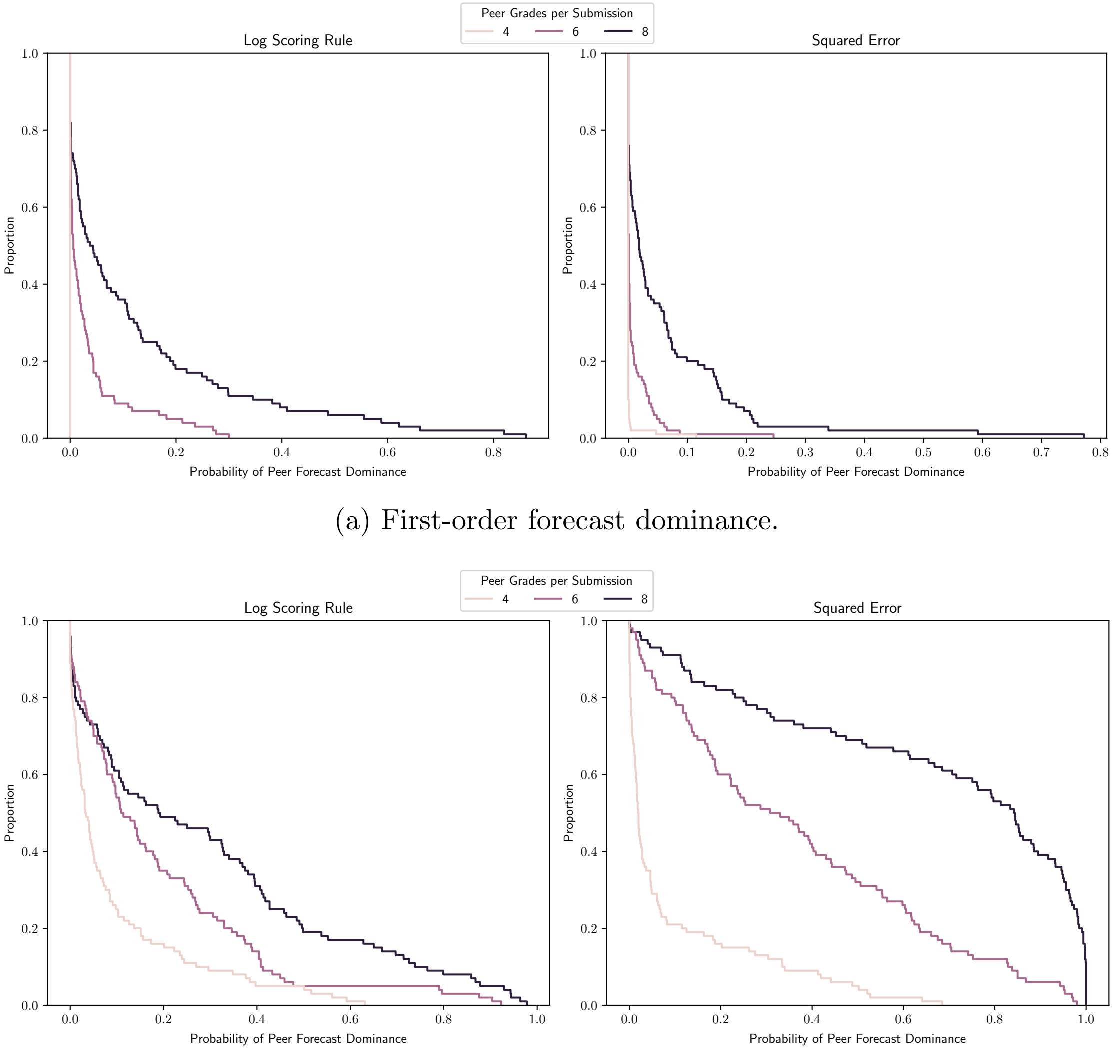
3.4.3 Applying Our Methodology in Simulated Experiments

We apply the methodology above to explore the following question: In a realistic model, what is the relationship between the number of peer grades per submission and the probability that the peer grading system with that number of grades per submission forecast dominates the single-instructor grading system?

To do so, we simulate data for a single assignment (with $n = 120$ submissions) according to model $\mathbf{PG}_Z$ using hyperparameters that Zarkoob et al. found produced plausible, realistic grading data [121] compared with the real data that they "gathered between September 2018 and December 2021 from four offerings of an undergraduate-level computer science course on the ethical and societal impacts of computing" [124] (Section 3.2). Using each simulated data set, we estimate the probability that the observed peer grading would forecast dominate instructor grading using the methodology described in Section 3.4.2. The results are shown in Figure 3.2. For each number of peer graders per submission on, the figure summarizes the 100 estimated probabilities—one for each simulated assignment—of first-order forecast dominance (Figure 3.2a) and second-order forecast dominance (Figure 3.2b).

Surprisingly, we find that the common choice of using four peer grades per submission (see, e.g., [1], [103], and [124]) results in forecasts that are generally unlikely to be clearly

[6] For students, on the other hand, the forecasts that result from peer grading do not depend too much on the particularities of any single student. In each simulation, there are enough students and submissions that the populations of each should be relatively similar to those in other simulations. As a result, to obtain reported peer grades, we simply draw parameters for each student in the population and generate an entire set of simulated student reported grades that we use as the input to Bayesian inference.



(a) First-order forecast dominance.

(b) Second-order forecast dominance.

Figure 3.2: Empirical survival functions (the complements of empirical cumulative distribution functions) that summarize the probabilities of forecast dominance for each of 100 assignments simulated according to model $\mathbf{PG}_Z$ as the number of peer graders per submission varies.

preferable to instructor grades. In terms of first-order forecast dominance, the results for six or eight peer graders per submission are similar—first-order peer forecast dominance appears to be a very high standard for both the log score and squared error. Indeed, as is shown in Figure 3.2, the probability of first-order peer forecast dominance rarely exceeds one-half (and only ever exceeds one-half when there are eight peer grades per submission). Second-order peer forecast dominance, on the other hand, is frequently probable with six or eight peer graders per submission, especially under squared error. Under squared error, for both six and eight peer grades per submission, the upper quartile (and for the latter, the median) for the probability of second-order forecast dominance is well above one-half. Alongside Theorem 3.3.3, this suggests that peer graders are in most cases relatively better at producing accurate point estimates than for making confident (i.e., low variance) forecasts.

3.4.4 Why Peer Grading Fails to Forecast-Dominate

Given the relatively low probabilities of (especially first-order) peer forecast dominance in Section 3.4.3, the natural question that arises is *why* peer forecast dominance is difficult to achieve. Recall that a necessary, but not sufficient, condition to achieve second-order peer forecast dominance is that the expected score on a random submission is higher for the peer graders than for one instructor. As a result, the fact that second-order peer forecast dominance is frequently probable with six or eight peer graders per submission suggests that it may be the case that peer graders outperform one instructor grader on average, but fail to outperform one instructor grader in the lower tail of the distribution of scores. We corroborate this intuition with the following procedure:

For each of the 100 simulated assignments from Section 3.4.3 (each of which involves 120 submissions), we compute the empirical CDFs of the scores achieved by the peer graders and of the expected scores that would be achieved by instructor grading. Then, as a summary, we compute an "average" empirical CDF, comprised of the average lowest score, the average second lowest score, the average third lowest score, etc. for both the peer graders and one instructor as described in Figure 3.3. We compare the mean and the quartiles of these average distributions in Figure 3.4.[7]

We find that scoring grading using both the log score and squared error leads to results that accord with our intuition, above.

1. For each number of peer grades per submission, the median and upper quantile of the average score distribution for peer graders is at least as high as that for one instructor.

[7] Note that for the instructor distribution, we combine the data from the three experiments, since the number of peer grades per submission does not affect the instructor outcomes.

$$\begin{bmatrix} \cdots & a_{1,j} & \cdots & a_{1,j'} & \cdots \\ & \vdots & & & \\ a_{i,1} & \cdots & a_{i,k} & \cdots & a_{i,n} \\ & \vdots & & & \\ \cdots & \cdots & a_{m,\ell} & a_{m,\ell+1} & \cdots \end{bmatrix} \longrightarrow \begin{bmatrix} a_{1,1} & \cdots & a_{1,j} & \cdots & a_{1,n} \\ & \vdots & & & \\ a_{i,1} & \cdots & a_{i,j} & \cdots & a_{i,n} \\ & \vdots & & & \\ a_{m,1} & \cdots & a_{m,j} & \cdots & a_{m,n} \end{bmatrix}$$

(a) First, sort the forecast scores from each simulation (each row) in ascending order.

$$\begin{bmatrix} a_{1,1} & \cdots & a_{1,j} & \cdots & a_{1,n} \\ & \vdots & & & \\ a_{i,1} & \cdots & a_{i,j} & \cdots & a_{i,n} \\ & \vdots & & & \\ a_{m,1} & \cdots & a_{m,j} & \cdots & a_{m,n} \end{bmatrix} \longrightarrow \begin{bmatrix} \overset{\downarrow}{\underset{\uparrow}{\overline{a_1}}} & \cdots & \overset{\downarrow}{\underset{\uparrow}{\overline{a_j}}} & \cdots & \overset{\downarrow}{\underset{\uparrow}{\overline{a_n}}} \end{bmatrix}$$

(b) Second, average over each column in the sorted array.

$$\begin{bmatrix} \overline{a_1} & \cdots & \overline{a_j} & \cdots & \overline{a_n} \end{bmatrix}$$

(c) The empirical CDF is a step function that increases by $\frac{1}{n}$ at each point in the final array.

Figure 3.3: Computing the "average" empirical CDF of forecast scores based on m simulations. Rows of the arrays correspond to a simulation $i \in \{1, \ldots, m\}$. Columns in the initial array correspond to a submission $j \in \{1, \ldots, n\}$. Each individual entry a_{ij} in the initial array is the forecast score achieved on submission j in simulation i. The color of an entry indicates its (relative) magnitude in its row, with darker colors corresponding to lower values.

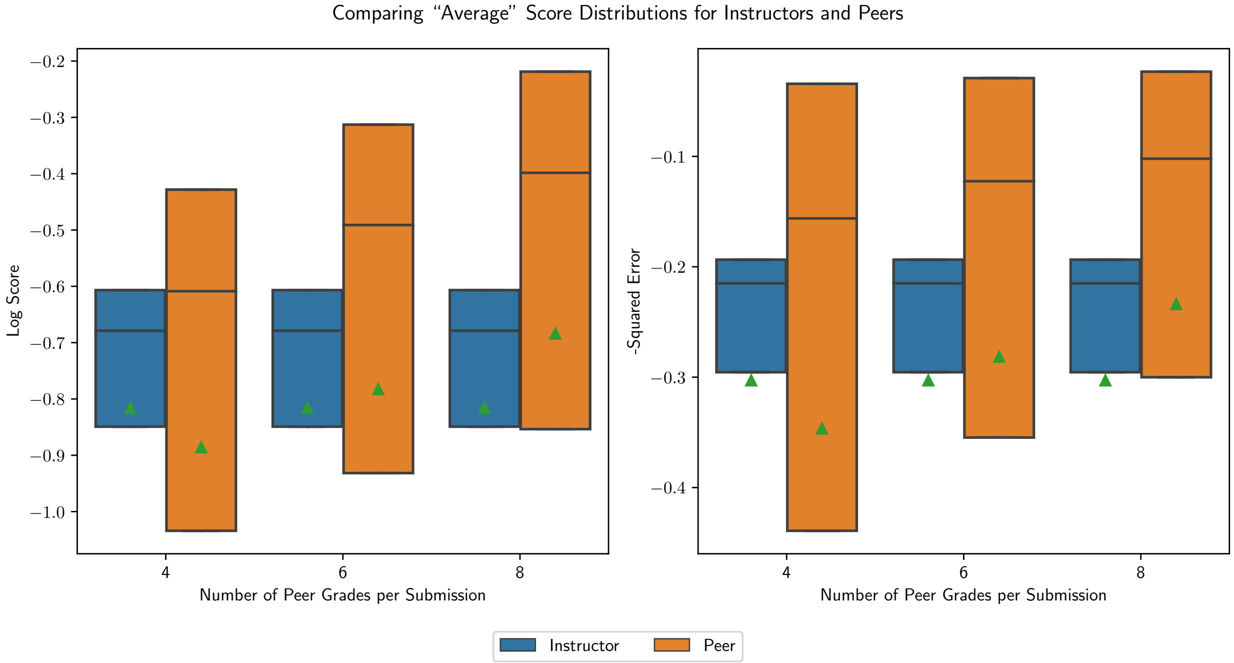


Figure 3.4: Summary of the “average” distribution of scores achieved by peer graders compared to that achieved, in expectation, by one instructor across the 100 simulated assignments from Section 3.4.3. The box plots illustrate the quartiles of the respective distributions; the green triangles illustrate the means.

This implies that when peer graders do well, they tend to exceed the standard set by instructor graders. That is, peer grading tends to have a higher upside.

2. However, the first quartile for peer graders is always lower than that for one instructor, even when there are eight peer grades per submission. This suggests a tendency for some "bad luck" submissions, where peer graders do poorly, to exist. This could result from solely unreliable graders being assigned to a submission, from some reliable graders nonetheless submitting poor reports, or both. In any case, the effect of this is a persistent deficit between the quality of the worst peer forecasts and the worst instructor forecasts that make it difficult for peer grading to first-order forecast dominate.

3. Lastly, the mean is the only one of the four values for which the values of the peer and instructor distributions cross—the mean is lower for peer graders than for one instructor when there are four peer grades per submission and higher for six and eight peer grades. This helps to explain our previous result about second-order peer forecast dominance, which was frequently probable for six and eight grades per submission, but not four. A higher mean is a necessary condition for second-order dominance.

3.4.5 A More Attainable Objective

We have seen that, while forecast dominance is difficult to achieve in our setting (i.e., in model $\mathbf{PG}_Z$, with the hyperparameters contained in Section 3.2.1), the mean of the distribution of forecast scores from peer grading is below the mean of the corresponding distribution from single-instructor grading for four peer grades per submission, but above for six and eight. Thus, in what follows we further consider having a greater mean as a weaker criterion (i.e., a criterion corresponding to a stronger assumption about the lead instructor's utility) under which peer grading can be considered at least as good as instructor grading.

To compare this criterion with forecast dominance: Let S_A and S_B be the random variables representing the scores for forecaster A and B respectively. The lead instructor prefers forecaster A over B for any linear increasing utility function u if and only if S_A has a higher mean than S_B.

3.5 Testing Interventions to Improve Peer Grading

The apparent trends in Figure 3.4 suggest that, although there are diminishing returns to increasing the number of peer grades per submission, the heavy workload of eight peer grades per submission may still be far from the region of the parameter space where the improvement from adding additional grades per submission is insignificant (especially for the first quartile of the distribution of scores). Thus, if it were feasible to assign heavy grading workloads to the students, it may be possible to achieve even first-order forecast dominance with high probability. However, since we believe that the returns in terms of pedagogical value for students to grade additional submissions from the peers are likely to diminish much more quickly than the returns in terms of shifting the distribution of scores, we now turn to the question of whether there are interventions for the standard setting of our experiments that can improve the performance of peer graders without increasing their workload.

3.5.1 Peer-Focused Interventions

Our first set of interventions involve only the peer graders themselves:

Drop Bad Graders. The main idea of this intervention is to ignore the reports of noisy graders. To do this, we sort graders according to one of three criteria: (1) their reliability, (2) their effort probability, or (3) a mixture of 0.75 times their standardized reliability plus 0.25 times their standardized effort. Then, according to the criterion being used, we ignore the reports of graders in the lowest 20% of the population.

Snake Draft Assignment. The main idea of this intervention is to assign peer graders to submissions so as to reduce the variance in the average quality of the graders assigned to each submission. To do this, we sort graders according to a mixture of 0.75 times their standardized reliability plus 0.25 times their standardized effort. Then, we order the submissions randomly. Then, we iterate over the submissions, first in order, then in reverse order, then in order again, and so on assigning one additional grader to each submission in each iteration until all submissions have the required number of graders. The grader that is assigned to a particular submission in a given iteration is the highest-quality grader (according to the sorted order) that has not yet been assigned to any submission in the current iteration and has not been assigned to that particular submission in any previous iteration.

It is worth noting that these interventions are idealized—they each require access to parameters of the model that cannot be observed directly. In practice, they would need to estimated somehow. For example, grading parameters could be estimated by having students grade some of a small number of calibration assignments with known true scores. However, in the case of our experiments, the ideal-ness of the tested interventions strengthens, rather than weakens, our results, because we find that even in a simulated world where the relevant parameters can be exactly known, these interventions do little (if anything) to improve the performance of the peer graders relative to one instructor.

3.5.2 Instructor-Focused Interventions

Our second set of interventions involve using an instructor to double-check the work of peer graders in certain instances. Submissions are selected to be regraded in order of the degree of uncertainty about their true grades, as measured by the variance of the Gaussian approximation to the samples of the marginal posterior distribution of each submission's true grade obtained via Gibbs sampling.

Regrade Submissions with High Uncertainty. The main idea of this intervention is to use instructors to decrease uncertainty about the true grades of a small fraction of the submissions. To do this, we collect the peer grades and perform inference via Gibbs sampling as in the previous experiments. Then, we select the 20% of submissions with the greatest posterior uncertainty about their true grade and generate a reported instructor grade for each of those submissions. Lastly, we perform inference again via Gibbs sampling based on all of the reported grades, both from peer graders and the instructor.

Adaptively Regrade Submissions with High Uncertainty. The main idea of this intervention is to use the information obtained in the first half of the regraded submissions to make more informed selections for submissions to be included in the second half of regrading. To do this, we collect the peer grades and perform inference via Gibbs sampling as in the previous experiments. Then, we do the following steps twice:

1. Select the 10% of submissions with the greatest posterior uncertainty about their true grade and generate a reported instructor grade for each of those submissions.

2. Perform inference again via Gibbs sampling based on all of the reported grades, both from peer graders and the instructor.

We note here that computational limits play a role in our implementation of these interventions. First, it is too computationally expensive to incorporate uncertainty about the parameters of the instructor who performs the regrades into the experiment. This is due to the fact that we would need to repeat the experiment (for each number of peer grades per submission) many times to fully cover the distribution of possible instructor parameters. As a result, we treat the instructor regrading submissions as an "average instructor," whose

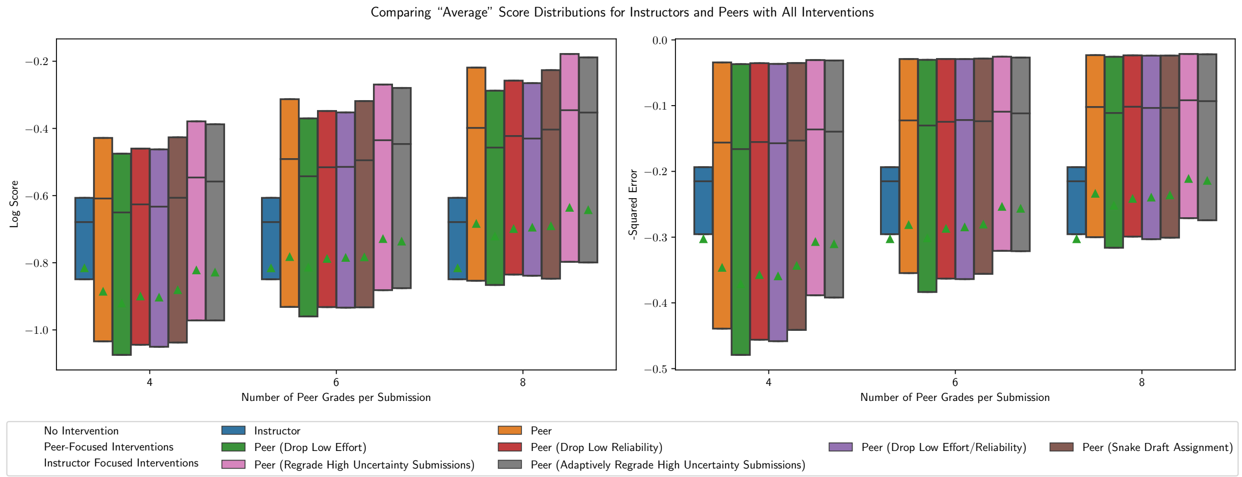


Figure 3.5: Summary of the "average" distribution of scores achieved by peer graders under various interventions compared to that achieved, in expectation, by one instructor across the 100 simulated assignments from Section 3.5.1.

parameters are fixed to the expected values of their respective distributions, and obtain reported grades from them on the submissions to be regraded just as if they were an additional peer grader.

3.5.3 Results

To explore the effect of these interventions, we repeat our experiment from Section 3.4.2 (including using the same random seeds, so that the populations of peer graders and submissions are the same for each intervention), modifying the experiment as detailed above in the descriptions of each intervention. Then, we create analogous plots to those in Figure 3.4 according to the procedure described in Section 3.4.4.[8] The plots for each intervention are shown in Figure 3.5.

For the peer-focused interventions, we find that those that drop bad graders appear to lead to worse performance, because they reduce the amount of information available in the system. That is, the weaker graders in our setting (whether defined as low effort graders, low reliability graders, or a combination of the two) appear to still be informative enough that it is worthwhile to collect their reports. The failure of the snake draft intervention to significantly improve performance, especially with respect to the first quartile, suggests that "bad luck" submissions are more likely due to noisy reported grades than due to the event that only unreliable graders are assigned to a submission. The snake draft intervention

[8] Note that for the one-instructor distribution, we simply re-use the distribution from Section 3.4.4, so that random noise does not introduce small variations in the baseline against which we compare.

mitigates the latter possibility, but does not address the former.

On the other hand, for instructor-focused interventions, we find that regrading is effective at shifting the "average" distribution of scores to higher values. For non-adaptive regrading, for example, the mean, median, and first quartile are always higher than for peer grading with no intervention. Also, these interventions are notably not idealized in the same way as the peer-focused interventions—they do not rely on knowing unobservable parameter values.

Uncertainty as a Proxy for Grading Quality. Above, we selected submissions to be regraded by an instructor when their (approximate) marginal posterior distribution had high variance. This idea is motivated by the fact that submissions with high-variance posteriors are more likely to have true values further from the posterior mean. However, we find that uncertainty (as measured by variance) is a relatively weak proxy for poorly graded submissions. On average, of the 24 submissions that were selected for one round of regrading, seven were among the worst 25% of submissions in terms of peer forecast scores, measured either by the log score or squared error. Choosing randomly, we would expect that, on average, six would be among the worst 25%, so choosing based on posterior uncertainty is only a modest improvement over guessing.

This detail also helps explain why selecting submissions to be regraded adaptively in two rounds does not improve over a single round of regrading. With adaptive regrading, it is still the case that, on average, seven of the 24 selected submissions were among the worst 25% of submissions in terms of peer forecast scores.

To conclude this discussion, we note that in the real world, it may be possible to select poorly-graded submissions for regrading more consistently. In particular, in the real world—unlike in the simulated world of our experiments—there is information beyond what is explicitly included in our model. Some of that information may be useful for identifying submissions that are more likely to have been graded poorly than is indicated by posterior distribution alone.

3.6 Allocating Graders with a Fixed Budget

In the previous sections, we established that, in the region of the parameter space where peer graders are only asked to perform a seemingly reasonable amount of work—an amount of work that is commensurate with the pedagogical benefits of that grading work—the resulting system is most readily improved by obtaining more information about the submissions. This information can come from assigning additional peer graders to each submission (as in Figure 3.4), from supplementing peer graders with instructors (as in the two right-most

	Peer Graded	Regraded by Instructor	Instructor-only Graded	Peers per Sub. if Peer Graded
TbV	100%	50%	0%	4
	80%	30%	20%	5
	67%	17% (20)	33%	6
	58% (69)	8% (9)	42% (51)	7
DaC	50%	0%	50%	8

Table 3.1: Allocations of a budget of 480 peer grades[11] and 60 instructor grades for grading 120 submissions. For values where the (rounded) percentages do not give an exact integer when multiplied by the number of submissions, the integer we use is given in parentheses.

box plots of Figure 3.5), or from some combination of the two. However, in practice, there are constraints on our ability to collect additional information. Peer graders should not be assigned grading work beyond what can reasonably be expected to provide pedagogical benefits. Instructors are severely limited in number. And both kinds of grader have natural constraints on the time that they can devote to grading.

In this section, we explore the consequences of these constraints. When the budget of grades that can come from peers and from instructors is fixed, how should we combine the efforts of both kinds of graders to achieve the best grading outcomes? To answer this, we assume that we have the following budget to assign a grade to each of 120 submissions:

- Each student can grade four submissions from their peers (480 peer grades).
- One instructor can grade (or regrade) 60 submissions (60 instructor grades).[9]

Then, we allocate this budget of grades by:

1. Selecting a percentage of submissions that should be graded by peers, which determines a number of peer grades that each peer-graded submission should receive.
2. Assigning an instructor to grade any remaining submissions.
3. Assigning an instructor to regrade peer-graded submissions, as in Section 3.5.2[10], if there are any remaining instructor grades in the budget.

The allocations that result from these rules are described in Table 3.1.

[9] This is a generous budget of instructor grades, but it still allows a significant saving of instructor resources over the baseline one-instructor system and is numerically convenient.

[10] Note that the computational limits discussed in Section 3.5.2 still apply in this experiment. As a result, the instructor used in regrading is once again the "average" instructor, whose parameters are fixed to the expected values of their respective prior distributions.

We compare the performance of these allocations by looking at percentiles and the mean of "average" score distributions, as we do to compare other grading systems and interventions. However, the computation of percentiles in this case is slightly more complicated, because they are for a distribution that is a mixture of the two kinds of distributions we have computed previously:

1. A peer-plus-regrade distribution (resulting from grades from the shaded columns in Table 3.1).
2. A single-instructor distribution that allows for uncertainty about the instructor parameters (resulting from grades from the third column in Table 3.1).

As a result, we compute percentiles using the CDF that results from a mixture of the average peer-plus-regrade distribution computed via the usual simulation procedure and the average single-instructor distribution computed in Section 3.4.4 (and used further in Section 3.5), where the mixture probabilities are those implied by the first and third columns of Table 3.1).

3.6.1 Results

To highlight the main results that are implied by the experiments described above, we focus on two quantities: the fifth percentile and the mean percentile of the average score distributions, which are plotted for each allocation in Figure 3.6. There is a clear trade-off in optimizing the objectives that correspond to these two quantities:

Divide-and-conquer. If the goal is to achieve first- or second-order forecast dominance over the one-instructor system, which as we have seen previously is difficult because of larger tails in the low end of the distribution of scores, then the best approach (with respect to the log score) is a **divide-and-conquer** (DaC) grading scheme, in which there is no overlap between peer and instructor grading, but as many peer grades per submission as is possible are obtained.

With respect to squared error, the divide-and-conquer grading scheme is only second best, but this result is somewhat sensitive to the way that the fifth percentile is estimated (and recall that the 58% peer graded allocation involves a slightly higher budget of peer grades). In any case, for both scoring methods, there is a general upward trend in the fifth percentile as the overlap between peer grading and instructor grading lessens.

[11] When the allocation involves seven peer grades per submission (the 58% peer graded allocation), we use 483 peer grades, since the number of peer-graded submissions must be an integer.

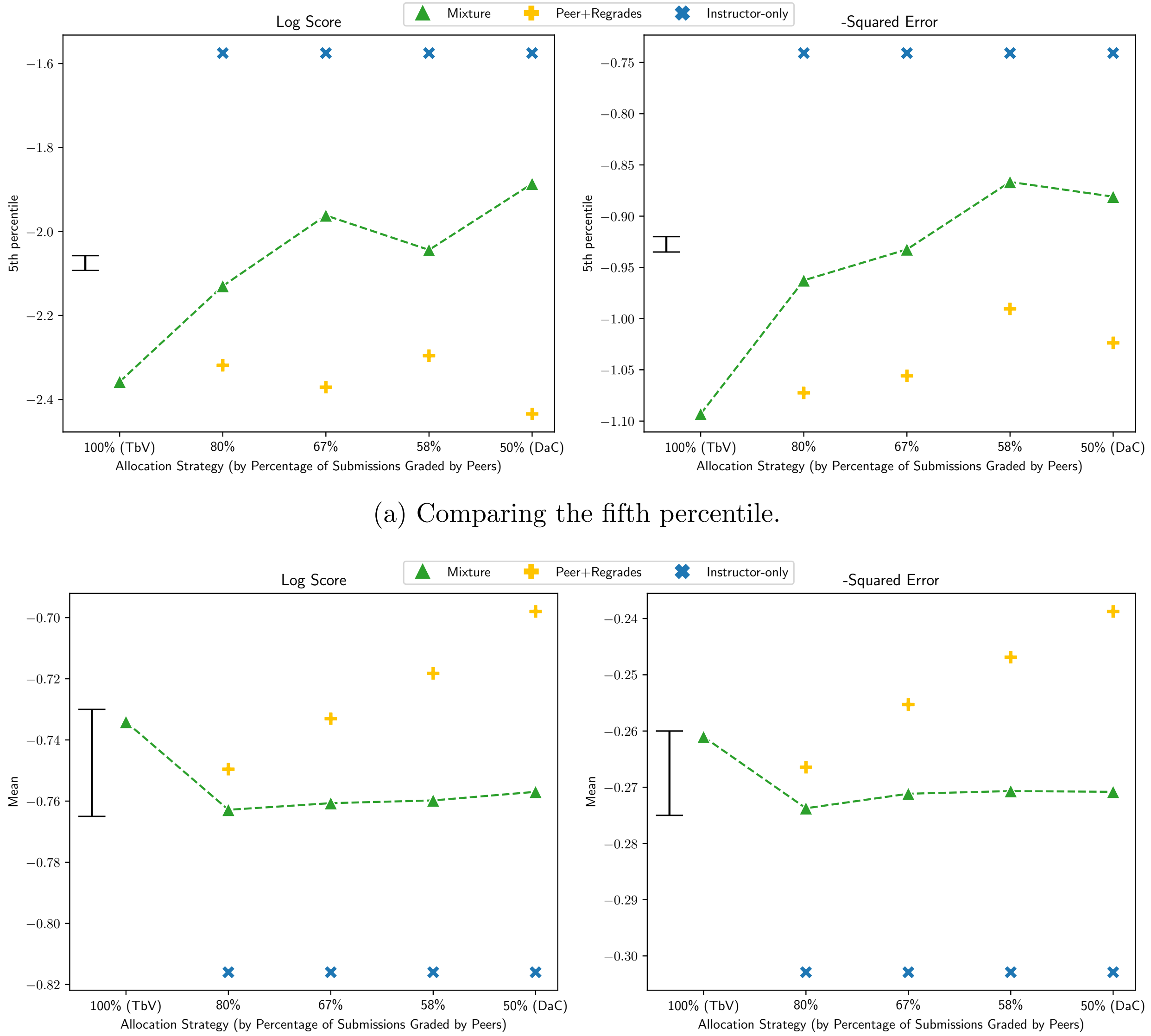
Trust-but-verify. If the goal is to simply achieve the highest average score possible, then the best approach is a **trust-but-verify** (TbV) grading scheme, where peer grading is used for all submissions, but supplemented using the full budget of instructor grades.

However, the magnitude of the difference in the average score among the allocations is much smaller than the difference in the fifth percentile, as indicated by the black bar to the left of each plot in Figure 3.6. Thus, in practice, it may make sense to optimize solely for the fifth percentile, despite the trade-off, since the average score is not affected too much.

These results can be explained by considering the individual components of the mixture distribution, for which the relevant quantities are also plotted in Figure 3.6. For the first component—the peer-plus-regrade score distribution (as described in the shaded columns of Table 3.1)—the trend for the fifth percentile is noisy (and sensitive to the method used for estimating the fifth percentile). Thus, the primary factor in determining the upward trend as the overlap between peer grading and instructor grading lessens is the increasing influence of the instructor-only component, which has a higher fifth percentile. For the mean, the trend in the peer-plus-regrade distribution is that the mean generally increases as the percentage of peer graded submissions decreases, after a decline between the first two allocations. However, the increased weight on the instructor-only component that accompanies the decrease in the percentage of peer graded submissions in the mixture distribution counteracts this effect, so the trust-but-verify scheme is optimal. The decline between the first two allocations suggests that the marginal benefit from one additional peer grade per submission is outweighed by the marginal cost of losing instructor regrades on 20% of the submissions. However, the marginal cost of one additional peer grade per submission in terms of instructor regrades decreases as the percentage of peer graded submissions in the allocation decreases, so this trend does not continue.

3.7 Predicting Performance from Observed Data

So far, our results have been derived under the same fundamental setup—the simulation of data from model $\mathbf{PG}_Z$ with the hyperparameters that were suggested for realistic data generation by Zarkoob [121]. We believe that many of the resulting lessons—e.g., the difficulty of achieving first-order peer forecast dominance—will generalize to other settings where there is a similar disparity between the average quality of a student grader and an instructor. However, the replication of our experiments with a different model of the data-generating process or with different hyperparameters for model $\mathbf{PG}_Z$ would involve a variety of computationally-intensive simulations, as we have conducted in this work. A natural



(a) Comparing the fifth percentile.

(b) Comparing the mean.

Figure 3.6: Comparing different allocations of a fixed budget of peer and instructor grades, in terms of the mean and fifth percentile of the "average" score distribution of the resulting grading schemes. Alongside this, to help explain the key trends, we include the analogous quantities from the component distributions that comprise the mixture distribution corresponding to each scheme.

Note the difference in magnitude of the scales of the y-axes for the different quantities, as indicated by the black bar to the left of each plot, which has the same magnitude within each column (i.e., for each scoring method).

question that arises, then, is the degree to which this process can be simplified, at least in certain circumstances.

In this section, we show that, in the region of the parameter space near our experimental setting, the results of our fundamental experiments from Section 3.4.2 with six peer graders per submission can be predicted accurately as two key hyperparameters are varied.

In particular, we find that the probabilities of first-order and second-order peer forecast dominance and the probability that the mean of the score distribution for peer grading is higher than that for the single-instructor system can be effectively modeled using a Bayesian linear regression of the shape parameter of the student reliability distribution (α_τ) and the mean of the distribution of effort probabilities ($\mu_e = \alpha_e/(\alpha_e + \beta_e)$), when the other hyperparameters, including the variance of the distribution of effort probabilities, are fixed. In fact, using α_τ alone is fairly reliable predictor, especially of the probability that the mean of the peer score distribution is higher. We choose these particular parameters to vary, and the values to which we vary them, based on model mis-specification experiments conducted by Zarkoob et al. [124].

Ultimately, we simulate data for nine different combinations of values for μ_e and α_τ: all combinations for $\mu_e \in \{0.7, 0.8, 0.9\}$ and $\alpha_\tau \in \{1.2, 2.0, 2.8\}$. For each combination, we run Bayesian linear regressions—specifically, Bayesian automatic relevance determination (ARD) regressions [65]—to predict the average probability of first-order peer forecast dominance, the average probability of second-order peer forecast dominance, and the average probability that the mean peer grading score is higher than that of the one-instructor system based on those parameter values. The results of these regressions are summarized in Table 3.2. To quantify the goodness-of-fit of these regressions, we use R^2—the proportion of the variance in the regressand (e.g., probability of second-order forecast dominance) that is explained by the regression model. We also calculate the gain in R^2 from including μ_e as a predictor. When this value is low and R^2 is high, as in many of our regressions, it indicates that α_τ alone is a strong predictor of the regressand. Lastly, for reference, we provide heat maps that summarize the regression results along with estimates of the uncertainty around each point. To illustrate, we include the heat map for predicting the probability that the score distribution for peer grading has a higher mean in Figure 3.7. The figures for first- and second-order forecast dominance are given in Appendix B.2.

These results are significant, because they have the potential to allow circumventing computationally expensive simulations in the event that the set of hyperparameters estimated from some real set of grading data were generally close to those in our experiments, but differing in the reliability or effort probabilities exhibited by the students. They also raise an interesting question to explore in future real-world experiments: To what degree do vari-

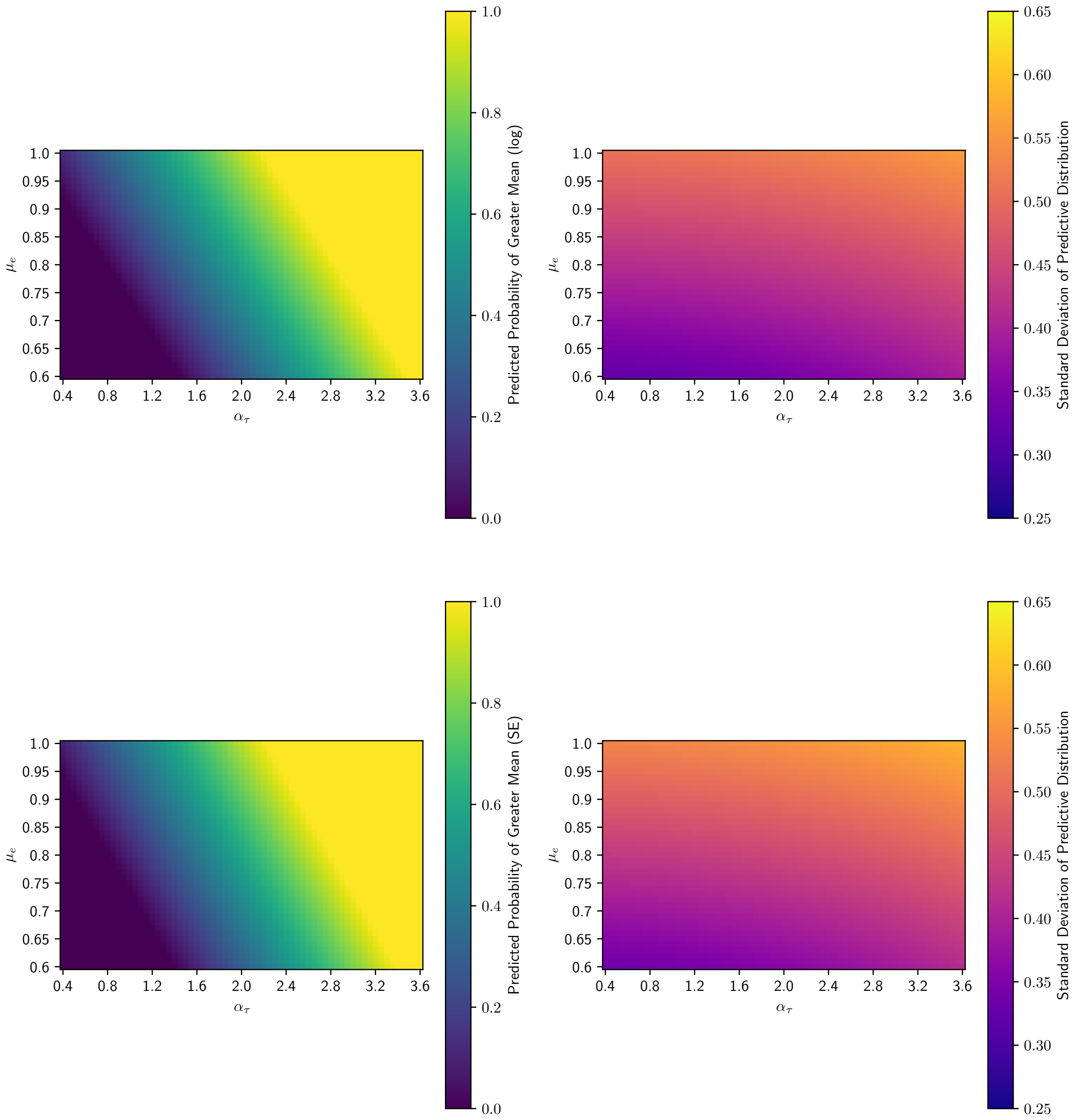


Figure 3.7: Predicted probabilities (left) and standard deviations of predictive distributions (right) of the mean of the peer score distribution being greater based on the values of α_τ and μ_e, under log score (top) and squared error (bottom), when there are six peer grades per submission.

Regressand	Score	R^2	R^2 Gain from μ_e
Probability of 1st-order Forecast Dominance	log score squared error	0.887 0.752	0.143 0.268
Probability of 2nd-order Forecast Dominance	log score squared error	0.882 0.923	0.142 0.248
Probability of Greater Mean	log score squared error	0.908 0.902	0.146 0.126

Table 3.2: Regression model fit for predicting various regressands when there are six peer grades per submission, given the values of μ_e and α_τ.

ous interventions shift the distributions of student reliabilities or effort probabilities toward higher values? If such interventions are costly, we can use our regression results to weigh the costs associated with them against the benefits that we would expect to accrue in regards to the regressands that we consider.

3.8 Discussion

We have proposed a framework for evaluating the quality of the grades produced by a new grading system—namely, peer grading—compared with those produced by the existing standard—namely, grading by one instructor. To evaluate our framework, we conducted simulated experiments using a model that was informed by real data obtained from a series of Computer Science courses at the University of British Columbia. However, student graders (and instructors) may exhibit strong heterogeneity across different courses, disciplines, and universities. Therefore, we believe that gathering additional data from a variety of sources will enhance our understanding of whether peer grading, in general, tends to outperform instructor grading, as assessed using our framework.

We found that, in the setting of our experiments—i.e., in model $\mathbf{PG}_Z$, with the hyperparameters contained in Section 3.2.1—peer grades are unlikely to be definitely better than instructor grades. However, our framework still proved useful by providing objectives with which to evaluate various interventions in the peer grading system. Ultimately, we found that the most useful intervention was simply to, in some form, collect more information about the submissions. We also showed how different objectives within our framework led to different strategies for collecting additional information with a fixed budget of peer and instructor grades. To maximize the average forecast score, it was best to supplement peer graders with instructors on the same submissions according to a trust-but-verify strategy. To maximize the lowest forecast scores, it was better to use a divide-and-conquer strategy that used instructor-only grading where possible, and allocated as many peer graders per submission on the remaining submissions given the budget constraint.

The focus of our work has been on improving the assignment of discrete numerical scores to submissions, so that the grades that follow from those scores reflect the quality of the submissions at least as well as instructor grades (and are perceived to do so by the participants in the grading system). However, we note that while this is a necessary component of making peer grading effective, it is not sufficient. A major challenge for effective peer grading systems is encouraging students to provide helpful feedback, primarily in the form of written comments or questions, in addition to numerical scores [80; 105; 114]. We believe that an interesting avenue for future work is to extend our central idea—developing a quantitative framework for establishing when peer grades definitely outperform instructor grades—to the challenge of encouraging peer feedback that is, if possible, at least as good as instructor feedback. An initial idea to bridge the gap between numerical scores and written feedback may be to have students review the feedback that they receive from peers and from instructors with numerical scores. An initial step towards developing this idea would be to extend models peer grading to incorporate comments of uncertain quality.

Given such a model, we may begin to ask—as we have asked in this work—under what circumstances are the scores for peer graders definitely preferable to the scores for instructors?

CHAPTER 4

Measurement Integrity in Peer Prediction: A Peer Assessment Case Study

4.1 Introduction

Peer prediction [60], or information elicitation without verification, is a paradigm for designing mechanisms that elicit reports from a population of agents about questions or tasks in settings where ground truth (and therefore the possibility of spot-checking) need not exist. One important dimension of evaluation for a peer prediction mechanism is the degree to which it rewards agents for their reports in a way that incentivizes truthfulness. This dimension, which we refer to as *robustness against strategic reporting*, has been the overwhelming focus of the theoretical peer prediction literature. We broaden this focus by introducing a new dimension of evaluation, *measurement integrity*, which quantifies a mechanism's ability to assign rewards that reliably measure agents according to the quality of their reports.

In the peer prediction paradigm, we assume that agents receive a signal (perhaps at some cost) about each task, drawn from some joint prior distribution. In the current literature, mechanisms are typically characterized by two properties that attest to their robustness against strategic reporting:

1. An equilibrium concept related to truthfulness that the mechanism induces under certain assumptions.

2. The assumptions, which typically constrain the form of the joint prior distribution of signals for every agent, that are sufficient to ensure inducement of the equilibrium concept.

Appendix C.1 details these two properties for a representative selection of fundamental mechanisms from the peer prediction literature. However, these properties alone are insufficient for evaluating peer prediction mechanisms' suitability for a given application. Firstly, this characterization omits other important desiderata. In many applications, for example, it is just as important for rewards to be fair as to be incentive-compatible. Secondly, even for a particular setting where incentive compatibility is a primary desiderata for peer prediction mechanisms, this characterization fails to determine the best mechanism to use.

It is possible for a mechanism to induce a stronger equilibrium concept than another mechanism, but only under a stronger assumption. It is also possible for a mechanism to "approximately" induce a stronger equilibrium concept under a given assumption. In both cases, there is no clear answer to the question of which mechanism is more robust against strategic reporting. Considering secondary desiderata also discussed in the theoretical peer prediction literature does not help. Such properties, for example, that mechanisms require little or no prior knowledge of the distribution of signals or that mechanisms only require simple reports from the agents, often fail to meaningfully differentiate the state-of-the-art mechanisms.

4.1.1 Our Contributions

- To address the insufficient characterization of peer prediction mechanisms, we introduce a new dimension of analysis. We call this new dimension *measurement integrity* and provide a formal definition alongside the motivating intuition.

- To address the issue of determining the best peer prediction mechanism for a given application, we perform extensive computational experiments, using both synthetic data and real data, to evaluate mechanisms' empirical properties in the context of an important purported application—peer assessment. First, we investigate the measurement integrity of state-of-the-art peer prediction mechanisms and find that they largely fail to demonstrate significant measurement integrity compared with simple baselines. Then, we broaden our experiments to develop a new, complementary perspective on robustness against strategic reporting, for which the state-of-the-art mechanisms are generally more effective. Together, the results of these experiments meaningfully differentiate the state-of-the-art mechanisms (Figure 4.1). These experiments also serve as a guide for comparing peer prediction mechanisms in other settings.

4.1.2 Measurement Integrity

Fundamentally, measurement integrity quantifies the strength of a mathematical relationship between the quality of an agent's reports and the reward they are allocated by a mechanism. Below, we define measurement integrity to be a concrete *empirical* estimand for which we can develop practical estimation strategies. The motivation for this definition, though, arises from its ability to represent a more abstract *theoretical* estimand: *ex post* fairness, where *ex post*, from the perspective of an agent, relates to all randomness from the mechanism's choices subsequent to their making a decision (e.g., choosing their reporting strategy) or related to actions of the other agents.

In peer assessment, for example, *ex post* fairness requires acknowledging that for a student, receiving an A with 80% probability and an F with 20% probability is not the same as certainly receiving a B; students should receive the grade they earn. Similarly, rewards should faithfully reflect the quality of the work submitted. We will see that mechanisms with high measurement integrity produce rewards that reliably reflect the quality of participants' reports. In general, when participants reflect on their experience with a mechanism and assess the fairness of that interaction, we believe they will tend to ask fundamentally *ex post* questions like "Was my reward fair compensation for my effort?" rather than *ex ante* ones. Measurement integrity speaks directly to these kinds of questions. As a result, taking measurement integrity seriously is an important step in transforming peer prediction mechanisms from intellectual curiosities into practical tools for eliciting information in the real world.

Moreover, the relationship between agent qualities and rewards at the heart of measurement integrity, and *ex post* fairness more generally, is useful for other goals: The rewards of a mechanism with high measurement integrity identify agents with high-quality reports—an important component of many strategies to aggregate information elicited from a population of agents with different levels of proficiency at the given task. On the other hand, a mechanism with low measurement integrity will have a noisy relationship between agent quality and rewards, which in tournament settings has been shown to increase the optimal payment required to elicit a certain effort level [18].

4.1.2.1 Defining Measurement Integrity

Suppose that P is a data-generating process for a given application (our model is described formally in Section 4.2), which, along with a (deterministic) *quality function* Q and (stochastic) mechanism M, produces (1) a vector of agent report *qualities* $\mathbf{q}$ and (2) a vector of agent *rewards* $\mathbf{r}^M$. These vectors have dimension n, the number of agents from P.

Given this setup, the last component needed for the definition of measurement integrity is a *correlation function* $\mathbf{corr} : \mathbb{R}^n \times \mathbb{R}^n \to [-1, 1]$. Correlation functions, also called correlation coefficients, describe the strength of a mathematical relationship between two quantities. For us, these two quantities are an agent's reward and an agent's quality. The absolute value of the correlation function increases with the strength of the relationship from 0, indicating no relationship, to 1, indicating a perfect correspondence. The sign of a non-zero correlation indicates whether changes in one variable are associated with the same kind of changes in the other variable—a positive relationship—or with opposite kinds of changes—a negative relationship.

Definition 4.1.1 (Measurement Integrity)**.** The *measurement integrity* of a peer prediction mechanism M with respect to a data-generating process P, a quality function Q, and a correlation function **corr** is

$$\underset{P,\,Q,\,\mathbf{corr}}{\mathrm{MI}}(M) = \mathbb{E}_{P,M}\left[\mathbf{corr}\left(\mathbf{q},\, \mathbf{r}^{M}\right)\right].$$

4.1.2.2 Unpacking the Definition

One crucial component of the definition of measurement integrity is the correlation function. Correlation functions are not an arbitrary class; they have several key features that make them uniquely appropriate. First, they require reference points for perfect (1), perfectly opposite (-1), and non-existent (0) relationships. This lends the values of correlation functions a relative interpretability that is absent from functions whose values have more absolute meaning, e.g., generic loss functions, which only require a reference point for perfect performance (0).

Intuitively, correlation functions allow us to quantify the extent to which rewards reflect the precise features of the qualities that are most important for a particular application. This is important, because practitioners in different contexts may care about different kinds of such relationships. For example, a practitioner might want (1) rewards to be proportional to some notion of quality (e.g., the inverse of absolute error); (2) the order of the rewards to represent the order of the reports' quality; or (3) the top 50% of agents recognized as such. Different (classes of) correlation functions, which measure the strength of different kinds of relationships, allow us to capture each of these different contexts, because correlation functions can be chosen so that their values are invariant under exactly the set of transformations (applied to the arguments) that preserve the relationship of interest. For the above examples, (1) Pearson correlation; (2) Kendall rank correlation; and (3) a transformation of area under the ROC curve are appropriate choices; we will return to these examples in our experiments.

Moreover, many mechanisms use a peer prediction score as an intermediate step in assigning rewards. Peer prediction mechanisms output scores that typically must be transformed into rewards in a manner that takes into account the particular context (cost, effort, etc.) and different mechanisms use different transformations. For example, scores may be linearly scaled or used to reward agents based on their rank or quantile in the agent population. In such cases, the rewards are unchanged by certain transformations of the scores (e.g., linear or monotone). Thus, to determine whether a peer prediction mechanism outputs useful scores in a given context, it is best to use an evaluation metric that is similarly invariant. Correlation functions, which exemplify such evaluation metrics, are thus more suitable than generic loss functions, which require the raw scores to be directly comparable.

The idea of invariance to a set of transformations also establishes the connection between correlation functions and measurement scales that inspires the term *measurement integrity*. Stevens [99], in a seminal work on measurement, characterizes four scales in terms of the set of transformations under which the information communicated by a particular measurement is invariant.[1]

Another crucial component of the definition of measurement integrity is that it goes beyond single evaluations of correlation functions. It considers the value of the correlation function across all of the potential outcomes of the data-generating process and the given mechanism. This is important, because it establishes that, for a mechanism with high measurement integrity, the relationship between rewards and qualities must be *consistently* strong.[2] The need for this consistency is at the heart of why measurement integrity is important in many peer prediction applications.

Lastly, the dependence of measurement integrity on a particular data-generating process is crucial. In this way, measurement integrity is similar to many more familiar quantities like precision, recall, accuracy, and other evaluation metrics from machine learning. Such metrics similarly have values that depend on the particular data-generating process under which they are evaluated, but contextualize that information in a consistent way. In the measurement theory literature, these are called *relative scales* and allow the creation of measurements on an absolute scale to enable us to "capture an underlying order in a complex problem" even though "such scales cannot exist objectively for all time and all objects, but only for a certain time and a given set of objects" [82]. This provides another contrast with generic loss functions, which, while absolute, are more difficult to compare across different contexts (e.g., when the number of agents is different).

[1] To highlight this connection, we reference specific scales, e.g., in the term ordinal measurement integrity, when the value of the correlation function we are considering is invariant under the set of permissible transformations for that scale, e.g., an ordinal scale, applied to $\mathbf{r}^M$ or $\mathbf{q}$. This allows us to explicitly specify the kind of information about agent qualities that we are evaluating the ability of the mechanisms to convey via their rewards.

[2] It is worth noting that expectations in general do not have this property—they can be driven up by extremely high values that occur with relatively low probability. However, for the expectation in the definition of measurement integrity, this is not the case, because correlation functions are bounded above by 1.

4.1.3 Our Approach

Alongside the relatively abstract definition, it is useful to have an explicit strategy for computing useful estimates of realistic measurement integrity values for a given application. Our work that follows is largely devoted to demonstrating such a strategy. Then, in Section 4.6, we apply a similar strategy for robustness against strategic reporting. The heart of our approach is to focus on empirical quantities, rather than analytically-derived ones. We estimate measurement integrity in computational experiments, first with a realistic Agent-Based Model (ABM) and then again with real data. Computational experiments offer important advantages over analytical or theoretical approaches for analyzing measurement integrity. Computational experiments are more naturally outcome-oriented. Simulated outcomes can be generated cheaply, frequently, and reliably under a wide range of parameter specifications. In contrast, making general theoretical statements quantifying *ex post* payments from a mechanism in this setting is cumbersome and difficult. While theorems have the advantage of potentially applying to a larger range of settings, we believe that, in this context, theorems are unlikely to give tight bounds and likely to be hard to interpret. In contrast, our computational experiments readily provide interpretable results, albeit on a chosen set of inputs. Further, our definition of measurement integrity requires fixing a specific data-generating process. This inherently limits the generality of any particular estimate of measurement integrity and, consequently, limits the potential benefits from the generality of a theoretical approach. We will see firsthand why this is essential, as the peculiarities of the setting we choose to explore—peer assessment—appear to be an important driver of our results in this work. We will also see, however, that this is not too limiting. Our results qualitatively accord across the related peer assessment data-generating processes—simulated and real—that we consider.

4.1.3.1 Peer Assessment

As emphasized above, our definition of measurement integrity is always instantiated with respect to a particular data-generating process. Thus, the usefulness of our estimates of measurement integrity for different mechanisms depends on the fidelity of the data-generating process or processes under which those estimates are derived to the real application of interest. With this in mind, we believe that peer assessment—where students provide feedback on the work of their peers—is a good candidate application for which to apply our techniques.

First, realistic models for data-generating processes (Section 4.8.3) in peer assessment exist and real peer assessment data (Section 4.2.2) are available for us to use in our computational experiments.

Second, peer assessment involves a common purported application of peer prediction (Section 4.8.2): peer *meta-grading.* Meta-grading is the task of "grading the graders": assessing the quality of peer assessments. In peer grading, the primary challenge is to correctly aggregate peer reports into grades that reflect the quality of a submission for an assignment. In peer meta-grading, incentive concerns are more salient. A student's grade for an assignment depends on other students' feedback (and the quality of their own submission). A student's meta-grade depends directly on their own feedback for other students. As a result, mechanisms for which the incentives are poorly designed will discourage rather than encourage high-quality feedback.

Lastly, meta-grading presents some unique challenges for the peer prediction paradigm that make it a particularly interesting case study:

Fairness Constraints. Grades are intended to reflect the quality of students' work, so the meta-grades assigned by any peer assessment mechanism must reflect the actual quality—not just the expected quality—of each student's performance in their assigned peer assessment tasks. To be suitable for assigning meta-grades, then, peer assessment mechanisms, should demonstrate significant measurement integrity.

Heterogeneous Quality. Agents may have different skills, exert different effort levels, and may not be fully calibrated. Mechanisms may vary in their ability to handle such differences.

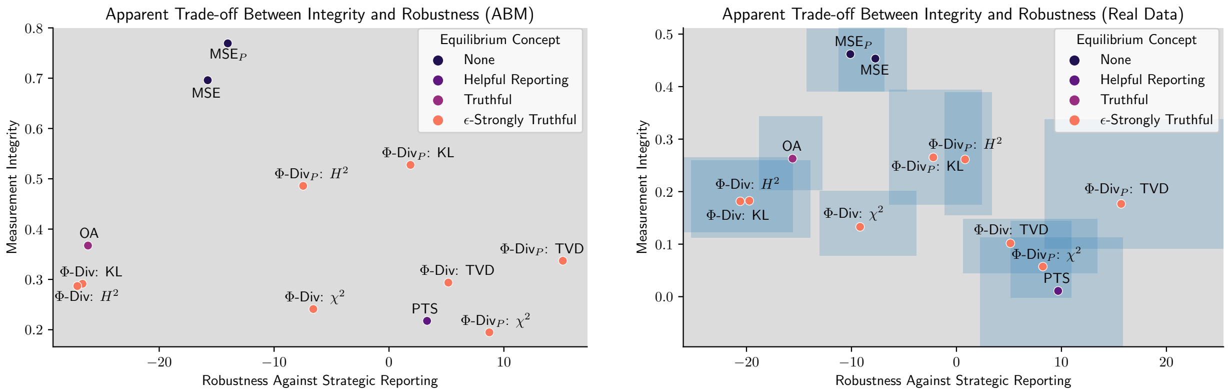
Scarcity of Data. Which mechanism performs best may be sensitive to the amount of data available. In peer assessment, the amount of data is severely constrained. There are limits on the number of assignments students can be expected to complete in a course (typically on the order of 10) and the number of submissions students can be expected to grade per assignment (typically between 3 and 6).

4.1.4 Our Results

Our experimental results, summarized in Figure 4.1, robustly differentiate existing peer prediction mechanisms according to their empirical performance. Moreover, they indicate an apparent trade-off inherent in seeking to simultaneously optimize measurement integrity (with respect to the notions of report quality that we consider) and empirical or theoretical robustness against strategic reporting. They also reveal the following lessons:

- Generic peer prediction mechanisms from the literature largely fail to demonstrate significant measurement integrity compared with simple baselines.

- Quantifying measurement integrity and empirical robustness against strategic reporting facilitates a more fine-grained comparison between mechanisms than solely consider-



(a) Experiments with ABM.

(b) Experiments with Real Data.

Figure 4.1: Direct, two-dimensional comparisons of peer prediction mechanisms that broadly summarize our experimental results.[3] In the experiments with real data, the final point values are computed by taking the average results over all four semesters in the dataset (Section 4.2.2), whereas uncertainty is estimated using the maximum and minimum values of the relevant quantities across the individual semesters.

Mechanisms are colored according to their theoretical robustness against strategic reporting, as described by the relevant equilibrium concept (Appendix C.1.1). Comparing the color scale to the x-axis, it is clear that theoretical robustness against strategic reporting is a somewhat noisy predictor of empirical robustness in our experiments.

ing theoretical notions of robustness against strategic reporting, which are often not directly comparable. For example, we find that implementation choices for some mechanisms affect their empirical performance in our experiments. In contrast, theoretical properties are typically defined to be agnostic about implementation choices.

- Certain peer prediction mechanisms can be augmented with parametric statistical models to improve their measurement integrity and empirical robustness against strategic reporting. As a consequence, parametric mechanisms should receive more attention in the peer prediction literature.

4.2 Data-Generating Processes

[3] Ordinal *measurement integrity* is computed with respect to the corresponding data-generating processes in Section 4.4 if the number of assignments/assignment blocks were uniformly random, mean squared error from ground truth, and the Kendall rank correlation coefficient (τ_B). Empirical *robustness against strategic reporting* is computed with respect to the analogous data-generating processes in Section 4.6 if the number of strategic graders and their strategy were uniformly random among those we consider and mean rank gain. Unfamiliar terms in these specifications are defined in their respective sections.

To complete our formal definition of measurement integrity, we first specify the necessary components of a data-generating process. At a high level, $P \to (I, J, G)$ generates a set of agents I, a set of tasks J, and an assignment graph G of agents to tasks. Each of these components may be characterized by a set of parameters described by P. More specifically, P should, at minimum: (1) generate exactly one *ground truth* response for each task j and (2) generate a *signal* $s(i, j)$ that agent i perceives about the ground truth for task j for each edge $(i, j) \in G$. Further, P may also describe how each agent i computes a *report* to submit to a mechanism for each edge $(i, j) \in G$ as a function of their signal $s(i, j)$. Without this specification, we assume that an agent's report is always *truthful*, i.e., equal to their signal.

Now, suppose a population of agents I and of tasks J has been drawn, the tasks have been assigned to the agents, and corresponding reports have been generated, all according to P. The reports are then submitted to a peer prediction mechanism M that computes a vector of *rewards* $\mathbf{r}^M = M(I, J, G)$. For each agent $i \in I$, their individual reward is r_i^M, the i-th component of $\mathbf{r}^M$. Corresponding to the vector of rewards is a vector of *qualities* $\mathbf{q} = Q(I, J, G)$, where Q is referred to as a *quality function.* Unlike mechanisms, we note that quality functions must be deterministic given a complete instance generated by P. For each agent $i \in I$, the i-th component of $\mathbf{q}$, q_i, quantifies the quality of i's reports. For example, q_i may be the mean squared error of i's reports with respect to known ground truth values.[4] We now turn to the specific peer assessment data-generating processes that we consider in this work.

4.2.1 Peer Assessment Agent-Based Model

Our ABM simulates a class of students enrolled in a semester-long course for which there is at least one graded assignment. For each assignment, each student turns in one submission and is randomly assigned submissions from four other students to grade.

Submissions. For each assignment j, each student i turns in a submission $s_{i,j}$. Each submission $s_{i,j}$ has a true integer score $g^*_{i,j} \in [0, 10]$, drawn (independently at random) from the binomial distribution $B\left(10, \frac{7}{10}\right)$.

The process of grading is modeled by an agent receiving a *signal* about the true score of a submission. The signal is a function of some number of draws from a *latent distribution* that depends on the true score of the submission being graded and the *bias* of the agent.

Bias. In practice, agents may have some bias in grading assignments. That is, the latent distribution from which draws are used to construct their signal for each assignment could

[4] This is the measure of quality that we adopt in this work. We explore alternative choices for the quality of an agent in Section 4.5.

have a mean that is slightly higher or lower than that assignment's ground truth score. An agent k's bias b_k is sampled uniformly at random from the normal distribution $\mathcal{N}(0,1)$. Their signal for submission $s_{i,j}$ is a function of draws from the latent distribution $B\left(10, \frac{g^*_{i,j}+b_k}{10}\right)$. If $g^*_{i,j} + b_k$ is less than 0 or greater than 10, then the value is truncated to be 0 or 10, respectively.

The number of draws from an agent's latent distribution that are used to create their signal—which determines the variance of the distribution of their signals—is a function of their *effort*; greater effort corresponds to lower variance.

When the signal is created using a single draw, defining the signal is trivial—the signal is equal to the outcome. When the signal is created using more than one draw from the latent distribution, the signal is defined as the simple average of the outcomes of the draws rounded to the nearest integer. This convention ensures that the space of signals is equal to the space of reports, so the notion of a "truthful report" is straightforward and well-defined. Our model of effort is as follows:

Continuous Effort. Effort is parameterized by a continuous value $\lambda \in (0, 2]$ drawn uniformly at random. The number of draws from the latent distribution used to create an agent's signal is equal to $1 + X$, where $X \sim \text{Pois}(\lambda)$ is drawn according to the Poisson distribution.

Lastly, we need to introduce an appropriate notion of the quality of agent's reports in order to reason about measurement integrity:

Report Quality. Here, we use a simple, intuitive notion of report quality—the squared distance between the report value and the true grade of the corresponding submission. In Section 4.5, we consider alternative conceptions of report quality and discuss the relative merits of the different approaches in detail.

4.2.2 Peer Assessment Data

Our real peer grading data set—which was collected for other projects [120] and graciously shared with us for this work—contains grading information from an undergraduate-level course on the design and analysis of algorithms taught at Northwestern University in both the Spring and Fall semesters of 2017 and 2019. For each student enrolled in the course, the data set contains information about the submissions they turned in during the course of the semester and the grades that they provided for submissions from other students. For each submission, the data set identifies the assignment that the submission corresponds to, specifies the grade that was ultimately awarded for that submission—which we treat as its true grade—and some number of peer grades. These awarded grades are a mix of grades assigned by instructors and grades assigned by the (non-parametric) `vancouver` algorithm [16], which takes into account instructor grades, peer grades, and the accuracy of peer graders. Because this combination of methods was deemed sufficient for fairly assigning grades to real students in the courses from which the data were collected, we are comfortable treating the assigned grades as unbiased estimates of the ground truth. For each peer grade that a student provided, the data set includes an identifier for the corresponding assignment, a numerical score, and written comments.

Many of the peer prediction mechanisms that we consider impose restrictions on the form of the data set. Certain mechanisms require that students grade at least two submissions for each assignment in which they will be evaluated by the mechanism. Other mechanisms require that at least two students grade each submission. Accordingly, we (iteratively) remove peer grades from students for assignments for which they graded fewer than 2 submissions and submissions with fewer than 2 graders until the modified data set meets these specifications. After this pre-processing, we simplify the grading and report space. The numerical scores—true grades and peer grades—from 2017 are out of 100 and from 2019 are out of 30. We coarsen these raw grades into the integer range $[0, 10]$. This simplifies the implementation of the peer prediction mechanisms and our experiments by keeping the space of possible reports the same for our ABM and all 4 semesters in the data set and helps to make the empirical distribution of reports less sparse in the space of all possible reports. It also lends our analysis some robustness to the method of assigning true grades, since small changes to the value of a true grade will tend not to change the value to which it is mapped during the pre-processing procedure.

Lastly, our parametric mechanisms (Section 4.3.3) and certain reporting strategies (Section 4.6.1) employ information about a (continuous) prior distribution for the true grades. Thus, we use maximum likelihood estimation to fit normal distributions to the empirical distribution of true grades for each semester.

The relevant information about each semester after all of the pre-processing is given in Table 4.1.

4.3 Peer Prediction Mechanisms

In this work, we consider a representative selection of fundamental mechanisms from the peer prediction literature. In what follows, we describe the intuition behind the mechanisms that we evaluate using our agent-based modeling framework. We also discuss the challenges that the particularities of the peer assessment setting and our peer assessment

	Students	Assignments	Submissions	Peer Grades	Retained	μ	σ^2
Spring 2017	94	16	758	4080	99.9%	8.71	3.8025
Fall 2017	86	16	577	3102	99.9%	7.57	4.9729
Spring 2019	49	13	313	1586	91.9%	7.68	3.6864
Fall 2019	65	14	389	2089	98.1%	8.25	2.8561

Table 4.1: Summary of the pre-processed peer grading data, including the total number of students, assignments, submissions, and peer grades for each semester. The percentage of grades in the raw data that are retained after pre-processing is also shown. For example, in Spring 2019, the 1586 grades left after pre-processing represent 91.9% of the total grades in the raw data. Lastly, the parameters μ and σ^2 are the mean and variance, respectively, of the normal prior distribution fit to the empirical distribution of true grades. These values are used in the implementations of parametric peer prediction mechanisms.

model pose to the implementation of the mechanisms. For a more specific discussion of the actual implementation of the various mechanisms and how we overcome these challenges, see Appendix C.2.

4.3.1 Baseline

In theoretical work, simple baselines would typically be excluded, due to the existence of trivial, non-truthful reporting equilibria. Despite this concern, however, such simple mechanisms are used in practice. Bachelet et al. [1], for example, recommend using the following mechanism to assign grades (when the reports have been appropriately pre-processed):

Mean Squared Error (MSE) Mechanism. On each submission that they grade, agents are paid according to the mean squared error of their reports from the *consensus grade* of each submission. The consensus grade of a submission—a basic estimate of its unobservable true grade—is defined to be the simple average of the reports of all 4 agents that graded it. To maintain the convention that the higher rewards correspond to higher quality agents, the payments are equal to the negative of the mean squared error.

4.3.2 Non-Parametric Mechanisms

Our first category of peer prediction mechanisms reflects the options that a novice mechanism designer would find in an initial search for peer prediction mechanisms to deploy in some application. In keeping with this, we implement these mechanisms as faithfully as possible to the descriptions given in the works in which they were proposed. We make changes only when necessary to ensure basic functionality within the setting of our model.

Note that for all mechanisms that involve pairing an agent with another agent in order to compute their scores on a grading task (i.e., generating a report for one submission), we take the expectation over all of the possible pairings to reduce the variance of the scores.

Output Agreement (OA) Mechanism. The simplest type of peer prediction mechanism, common in the literature [22], is an *output agreement* mechanism, which serves as another simple baseline with which to compare state-of-the-art peer prediction mechanisms. To compute payments for a task in the OA mechanism, agents are paired and their reports are compared. Agents are paid 1 if their reports match and 0 otherwise.

Peer Truth Serum (PTS) Mechanism. Developed by Faltings et al. [22], the PTS mechanism pays agents if their report for a task agrees with the report of a randomly selected peer on the same task. The magnitude of the payment is proportional to the inverse of the frequency of their report according to a distribution R over the report space.

Φ-Divergence Pairing (Φ-Div) Mechanism. This mechanism was proposed by Schoenebeck and Yu [85] and is based on the application of an information-theoretic framework for designing peer prediction mechanisms described by Kong and Schoenebeck [47]. Like the OA mechanism, this mechanism pairs agents with peers. The pairs are rewarded for submitting correlated reports on a *bonus task* and penalized for submitting correlated reports on a pair (one for each agent) of *penalty tasks* that are distinct from each other and from the bonus tasks.

The magnitudes of the respective reward and penalty depend on a convex function Φ chosen by the mechanism designer and on $\mathrm{JP}(x, y)$, the joint-to-marginal-product ratio of random variables X and Y drawn, respectively, from each agent's distribution of reports:

$$\mathrm{JP}(x, y) = \frac{P_{X,Y}(x, y)}{P_X(x)P_Y(y)},$$

where $P_{X,Y}(x, y)$ is the probability of observing reports x and y as answers to the same question under the joint distribution of reports X and Y, and $P_X(x)P_Y(y)$ is that probability according to the product of the marginal report distributions. Their ratio can be understood as measuring how much more likely a pair of reports x and y are to occur on the same question versus different questions. Note that each quantity is a function of the random variables X and Y, which depend both on the agents' strategies and the joint prior. In general, JP is unknown to the mechanism and will need to be estimated.

For a given pair of agents, a bonus task b, and a pair of *penalty tasks*, $p \neq q$, the payment is :

$$\partial\Phi(\mathrm{JP}(x_b, y_b)) - \Phi^*(\partial\Phi(\mathrm{JP}(x_p, y_q))),$$

where $\partial\Phi$ is the *subgradient* of Φ, Φ^* denotes the *convex conjugate* of Φ, and x_i and y_j denote

Φ(x)	Φ-divergence	Notation
$\frac{1}{2}\lvert x-1\rvert$	Total Variation Distance	TVD
$x\log x$	Kullback-Leibler divergence	KL
x^2-1	χ^2-divergence	χ^2
$(1-\sqrt{x})^2$	Squared Hellinger distance	H^2

Table 4.2: Common choices for Φ and their associated Φ-divergences.

the first agent's report on task i and the second agent's report on task j, respectively. The intuition is that the first term rewards an agent based on the likelihood of their report for the bonus question b given the other agent's report on b. The second term penalizes agents for reporting generically likely answers by considering the likelihood of an agent's report for the penalty question p given the other agent's report on the distinct penalty question q. See Schoenebeck and Yu [85] for a complete discussion of each component of this mechanism, including definitions for all of the relevant terms above.

Ideally, we would want to estimate the joint-to-marginal-product ratio of the reports for each pair of agents, but given the limited availability of data in this setting, the best we can do is treat the agents anonymously and compute one estimate, $\hat{\text{JP}}$, that applies to the entire agent population. See Section C.2.1.1 for the details of this estimation procedure.

In our experiments, we consider 4 common Φ-divergences, which are described in Table 4.2. One important note with respect to the choice of Φ-divergence is that when the total variation distance (TVD) is chosen, the Φ-Div mechanism emulates the Correlated Agreement (CA) mechanism from Shnayder et al. [91] (which in turn generalizes the ideas underlying the mechanism proposed by Dasgupta and Ghosh [14]).

One significant omission from our selection of state-of-the-art mechanisms is Kong's Determinant-based Mutual Information (DMI) mechanism [46], which has impressive theoretical properties. In our computational experiments, the mechanism requires significant modifications to the setting or else it will simply assign every agent a reward of 0. With the necessary modifications, the mechanism does not perform particularly well with respect to measurement integrity and the results for robustness against strategic reporting are not a fair comparison to the other mechanisms. As a result, we omit DMI from consideration in the body of this work. However, we discuss our implementation and the necessary modifications in Appendix C.2 and include DMI, when possible, in the additional experimental results presented in Appendix C.

4.3.3 Parametric Mechanisms

Anticipating the challenges that generic mechanisms might encounter when deployed in a specific setting, we also explore how certain peer prediction mechanisms can be supplemented with domain-specific, parametric statistical models. To implement these, we adopt the perspective of a real-life mechanism designer. In the real world the "true" distributions and parameter values that "govern" the behavior of students participating in peer assessment are inaccessible. Instead, a mechanism designer can examine the peer assessment literature to find a model of peer assessment inspired by and validated on real data for which the hyperparameters of the model can be tuned to fit their particular application. Here, model **PG**$_1$ from Piech et al. [74] meets both criteria. It constitutes a reasonable continuous approximation to our primarily discrete underlying model (in which "reliability" serves as a proxy for effort). The model, with hyperparameters that are appropriate for our setting, is described below:

$$\begin{aligned}
\text{True Score}: &\quad g^*_{i,j} \sim \mathcal{N}\left(7, 2.1\right) \text{ for each submission } s_{i,j}, \\
\text{Reliability}: &\quad \tau_i \sim \mathcal{G}\left(10/1.05, 10\right) \text{ for each agent } i^5, \\
\text{Bias}: &\quad b_i \sim \mathcal{N}\left(0, 1\right) \text{ for each agent } i \\
\text{Signal}: &\quad z^k_{i,j} \sim \mathcal{N}\left(g^*_{i,j} + b_k, \tau_k^{-1}\right) \text{ for a grader } k\text{, who is grading submission } s_{i,j},.
\end{aligned}$$

To reiterate, the simulated data in our experiments is always generated according to the model described previously in Section 4.2.1. But instead of estimating parameters of that underlying model, we estimate the parameters of model **PG**$_1$. We then use those estimates in deploying the parametric peer prediction mechanisms described below. This simulates the situation faced by a mechanism designer in a real deployment. They would be unable to know the "true" underlying model, but would be able to tune a reasonable statistical model for their application using past data.

Using model **PG**$_1$ is also useful because existing work from the peer assessment literature shows how to estimate its parameters. Chakraborty et al. [12] propose a method for estimating the parameters of model **PG**$_1$ (and computing meta-grades) using limited access to ground truth. In the absence of ground truth, their estimation method (though not their meta-grading method) can be adapted to estimate the parameters of the model using an expectation-maximization-style algorithm with Bayesian priors for the bias (when applicable) and reliability of each agent. The details of our estimation procedure are available in Section C.2.1.2.

[5] $\mathcal{G}$ denotes a Gamma distribution. The hyperparameters $\alpha_0 = 10/1.05$ and $\beta_0 = 10$ for $\mathcal{G}$ were chosen by inspection, subject to having the correct expected value for a continuous effort agent.

Parametric MSE ($\mathbf{MSE}_P$) Mechanism. Under this mechanism, each agent is awarded according to the mean squared error of their reports (corrected for estimated biases, when appropriate) from the estimated true scores. As with the baseline, the payments are equal to the negative of the mean squared error.

Parametric Φ-Divergence Pairing ($\mathbf{\Phi}$-$\mathbf{Div}_P$) Mechanism. Instead of using empirical estimates of the joint-to-marginal-product ratio of reports, we can pre-compute the joint-to-marginal-product ratio $\mathrm{JP}(x, y)$ analytically under model $\mathbf{PG}_1$ and score the tasks after estimating the parameters of model $\mathbf{PG}_1$ using the estimation procedure described above. See Appendix C.3 for the calculation. This allows us to individualize the joint-to-marginal-product ratio for each pair of agents, which as we noted earlier is desirable but intractable for the non-parametric version of this mechanism, given the scarcity of data.

For each task, agents are paired and scored according to the same procedure used for the non-parametric Φ-Div mechanism, but using the closed-form expression we derived for $\mathrm{JP}(x, y)$ instead of an empirical estimate. As with the MSE_P mechanism, reports submitted to this mechanism are corrected for estimated bias prior to scoring when appropriate.

4.4 Quantifying Measurement Integrity

In our experiments, we seek to empirically evaluate the above mechanisms according to their measurement integrity and robustness against strategic reporting. Evaluating mechanisms for both properties simultaneously would make it difficult to isolate which features were beneficial for which property. As a result, we first quantify measurement integrity in isolation, assuming that agents report their signals honestly. In these experiments, when using parametric mechanisms, we do not correct agents' reports based on their estimated biases, since we adopt a notion of quality that depends on the raw report values directly—the squared distance between the reports and the true scores.

4.4.1 Computational Experiments with ABM

One key advantage of the agent-based modeling approach is access to latent quantities, e.g., a submission's true score, that are generally not observable, without noise, in the real world. Another is the ability to readily repeat experiments over a range of parameter specifications. We leverage these advantages in order to analyze the relationship between agents' payments assigned under the various mechanisms and the squared error of their reports to the ground truth scores, considering various intuitive notions of measurement integrity.

4.4.1.1 Methods

For each mechanism, we perform the same procedure of simulating "semesters," which consist of 500 students submitting and grading some number of simulated assignments. The number of assignments is varied from $i = 1, 2, \ldots, 15$. For every value of i, we simulate 50 semesters, which each proceed as follows:

1. For each assignment:

   i. All students turn in a submission whose true grade is drawn from the true grade distribution.

   ii. A random 4-regular graph of agents is constructed.

   iii. Students grade the submissions of their neighbors in the graph according to our peer assessment model. The squared errors of their grades to the true grades are recorded.

   iv. The grades are reported to the mechanism, which assigns a reward to each student for their performance in peer assessment for that assignment.

2. Students' total rewards for the semester—the sums of their rewards for each of the i individual assignments—and their cumulative squared error to the true scores are used to evaluate the correlation functions.

4.4.1.2 Correlation Functions

Choosing an appropriate correlation function is crucial to operationalizing measurement integrity as a useful quantity for a peer prediction application. There are already many useful correlation coefficients that correspond well to the various measurement scales. In addition to these, measurement integrity can also accommodate more customized correlation functions. This is useful, because the best correlation function for a given setting can depend on a mechanism designer's tolerance for making different kinds of errors in that application. In peer grading, for example, a practitioner may consider it worse to fail a borderline student that should have passed than to pass a borderline student that should have failed. Their choice of correlation function, then, may take these preferences into account.

To illustrate these principles, in our experiments, we consider increasingly fine-grained measurements. For the first of these, we illustrate how we can modify a popular machine learning metric that is well-suited to evaluate that kind of measurement into a new correlation function. For the remaining scales, we adopt well-known correlation functions that are similarly well-suited for their corresponding kinds of measurement. In doing this, we note that we depart somewhat from the approach that we would expect others to take in applying our techniques. Generally, we would expect that a mechanism designer will have a utility function in mind that will help them determine a particular correlation function. We take a more exploratory approach—considering multiple correlation functions without a particular utility function in mind—in order to find out which kinds of measurement integrity, if any, are high for the peer prediction mechanisms we consider in our case-study application.

Coarse Ordinal Measurement. Nominal measurement is akin to standard classification, but in our setting, it is more instructive to consider classification with ordered classes, which, when the number of classes is small, is a coarse kind of ordinal measurement. In particular, we consider binary classification of agents as being above or below the median in terms of squared error to ground truth scores. For binary classification tasks, the Area Under the Curve (AUC) of the receiver operating characteristic (ROC) curve is an evaluation metric that is widely used in machine learning. This metric is particularly well-suited to evaluating mechanism performance at our binary classification task: AUC summarizes how useful the rewards assigned by the mechanism are for being translated into a classifier via the selection of a threshold such that all students with rewards above the threshold are classified as above-median and all students with rewards below the threshold are classified as below-median. At first glance, AUC does not quite meet our definition of a correlation function, because it varies between 0 and 1, not -1 and 1. However, it does have values that impart analogous meanings to those required by a correlation function: An AUC score of 1 indicates a perfect classifier, an AUC score of 0.5 is the expected value of a random classifier, and an AUC score of 0 indicates a perfectly opposite classifier. Thus, we can define a new correlation function—Area Under the Curve Correlation (AUCC)—by simply transforming AUC so that the significant values have the required numeric value:

$$\text{AUCC} = 2 \cdot \text{AUC} - 1.$$

Fine Ordinal Measurement. The most fine-grained ordinal measurement is ranking. For rankings, useful correlation functions already exist. Here, we adopt the Kendall rank correlation coefficient (τ_B). The value of τ_B is related to the number of pairs that appear in the same order (concordant pairs) and in the opposite order (discordant pairs) in the two rankings being compared. In the case that neither ranking has ties, τ_B is equal to the proportion of pairs that are concordant minus the proportion of pairs that are discordant.

Interval Measurement. It is conceivable that the rewards from some mechanisms might contain even more fine-grained information about agents' squared error from the ground truth than just the ordinal notion that higher payments correspond to lower squared error. That is, the *magnitude* of the difference in payments between agents may contain information about the magnitude of their difference in squared error. At the very least, this should be the case for the baseline MSE and MSE$_P$ mechanisms, if they are computing good estimates of the unobserved (by the mechanisms) ground truth scores. In particular, we would expect that those mechanisms would contain good *linear* information about agents' squared error. That is, a given magnitude of the difference in payment should more or less correspond to the same magnitude of difference in quality regardless of the particular values of the payments. This property is the defining characteristic of interval measurement, and can be evaluated with the most familiar correlation coefficient: the Pearson correlation coefficient (ρ), which measures the strength of the linear relationship between two variables.

4.4.1.3 Results

Estimates of measurement integrity with respect to the various correlation functions, while varying the number of assignments per semester, are plotted in Figure 4.2. These estimates are computed by taking an average of the value of the correlation function over the 50 semesters simulated for each value of the number of assignments in a semester.

The first result that stands out is that it is feasible to achieve high levels of measurement integrity, even under strict measurement scales. As the number of assignments in a semester increases, thereby increasing the amount of information available to the mechanisms, the baseline MSE and MSE$_P$ mechanisms score consistently highly according to each of the correlation functions, including near-perfect Pearson correlation. This indicates that, unsurprisingly, it is possible to estimate true scores that are not observed by the mechanism highly reliably in our model when agents report truthfully. However, despite this possibility, peer prediction mechanisms generally do not appear to take advantage of this. As a result, they largely perform relatively poorly according to each correlation function compared to simple baseline mechanisms. The exceptions are two parametric mechanisms, Φ-Div$_P$: KL and Φ-Div$_P$: H^2, which mostly outperform the OA baseline.

Interestingly, the pattern established in the plots of the first two correlation functions, which are qualitatively nearly identical, does not hold in the plot of the Pearson correlation (ρ). In particular, the Φ-Div$_P$: H^2 mechanism performs much less well according to Pearson correlation than the other correlation functions, indicating high *ordinal* measurement integrity, but relatively low *interval* measurement integrity. Importantly, this shows that the measurement scale that is relevant to a particular application—the notion of measurement that is desirable—can matter significantly with respect to how a mechanism performs. In this case, inspection reveals that the difference in performance is due to a tendency of the

Φ-Div$_P$: H^2 mechanism to occasionally assign very negative outlier payments. These outliers interfere with the linear relationship between the payments and agents' squared error, but not the ordinal relationship. The payments from the PTS mechanism are also notably less useful with respect to linear than ordinal information. However, the ordinal information conveyed via the PTS mechanism was already poor relative to the other mechanisms.

4.4.2 Computational Experiments with Real Data

4.4.2.1 Methods

We replicate the experiments with simulated data described in Section 4.4.1.1, substituting our four semesters worth of real data for the 50 semesters worth of data that we simulated via our peer grading ABM. For each semester, as in our experiments with our ABM, we assign rewards according to each mechanism for each assignment, one at a time. However, due to limitations in the data and the necessary pre-processing (Section 4.2.2), not every student is associated with peer grades for submissions on every assignment. In fact, different students are associated with different numbers of peer grades, which complicates our analysis.

We address this complication in two steps. First, we divide each student's squared error of reports from true scores and their payments by the number of peer grades with which they are associated. Thus, we consider the mean squared error and average payment of each student when evaluating the correlation functions. Second, to control for the fact that the mean squared error and payment for students associated with few peer grades may be much noisier than those of students associated with many peer grades, we focus on students associated with many peer grades when computing the values of the correlation functions. To accomplish this, for each semester, we split the assignments into four blocks of roughly equal size and consider only students who are associated with at least one peer grade in each assignment block when evaluating the correlation functions.[6] Under this rule, the number of students considered when evaluating the correlation functions is 84, 49, 42, and 54 for the Spring 2017, Fall 2017, Spring 2019, and Fall 2019 semesters, respectively. Overall, the correlation functions are evaluated 4 times for each semester—once after each assignment block has been processed. Note that information accumulates as each block is processed—the evaluation of the correlation function uses all the information obtained in the current block of assignments and its predecessors. Thus, a nice consequence of this procedure is that new information for every student is incorporated every time the correlation functions are

[6] In general, this rule need not exclude students associated with few peer grades. In practice, however, it strikes a good balance between being as inclusive as possible while excluding students associated with very few grades. In particular, it seems to do better than using a threshold based on a percentage of the maximum number of peer grades for the given semester.

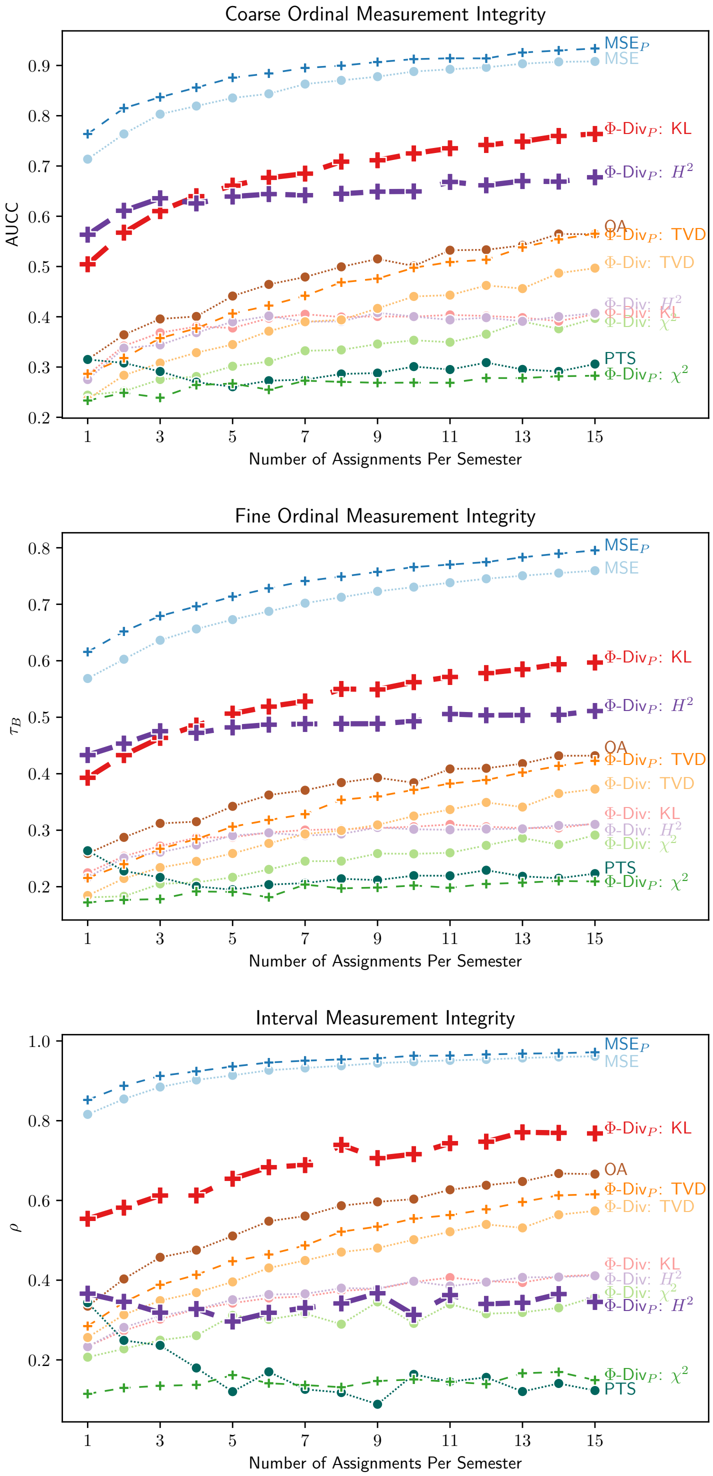


Figure 4.2: Quantifying Measurement Integrity with ABM. Average values of correlation functions corresponding to different measurement scales as the number of assignments in a simulated semester grows. The average for each number of assignments is taken over 50 simulated semesters.

re-evaluated.

Although there is more uncertainty in our estimates of measurement integrity in this setting, due to our inability to re-draw samples from the underlying data-generating process, we are still able to take the randomness of the mechanisms into account: We record the average value of the correlation functions over 50 iterations of the procedure described above.

4.4.2.2 Results

We begin with the results for fine ordinal measurement—shown in Figure 4.3—for which the correlation function is the Kendall rank correlation coefficient (τ_B), since, as in our simulated experiments, those results are qualitatively similar to those for coarse ordinal measurement and since τ_B connects with our experiments for quantifying robustness against strategic reporting, which involve rankings.

Unsurprisingly, the results for the real data are much noisier than those for the simulated data. However, there are some patterns that emerge, which corroborate observations from our experiments with ABM. In particular, there is a general consensus about the best-performing mechanisms. The MSE and MSE_P baselines are among the best-performing mechanisms at nearly every point. The best-performing peer prediction mechanisms from the simulated experiments, Φ-Div_P: KL and Φ-Div_P: H^2, are also often among the best mechanisms, albeit less consistently. Indeed, if we average the value of τ_B for each mechanism across all numbers of assignment blocks and all semesters, we find that, the five best-performing mechanisms on average are MSE_P, MSE, Φ-Div_P: KL, OA, and Φ-Div_P: H^2, in that order. These are exactly the same five mechanisms that results from a similar analysis of the simulated data, albeit in a slightly different order.[7] Notably, the two Φ-Div_P mechanisms perform especially well relative to the other mechanisms in the first block of assignments, when the least amount of information is available to a mechanism. In many courses, grades are assigned after each assignment and using only information from that particular assignment. In those cases, the amount of information available to a grading mechanism would be most similar to the amount of information available to the mechanisms in the first block of assignments in our experiments. This result mirrors the results from the experiments with ABM, in which the relative advantage of these two mechanisms over certain other peer prediction mechanisms (e.g., OA) tends to decrease as the number of assignments per semester increases (Figure 4.2).

The results for the other correlation functions are given in Figure 4.4. Repeating the analysis we did with τ_B, above—i.e., averaging the value of each correlation function for each

[7] In the experiments with real data, the differences between the values for Φ-Div_P: KL, OA, and Φ-Div_P: H^2 are not significant enough to make reliable statements about their relative ordering.

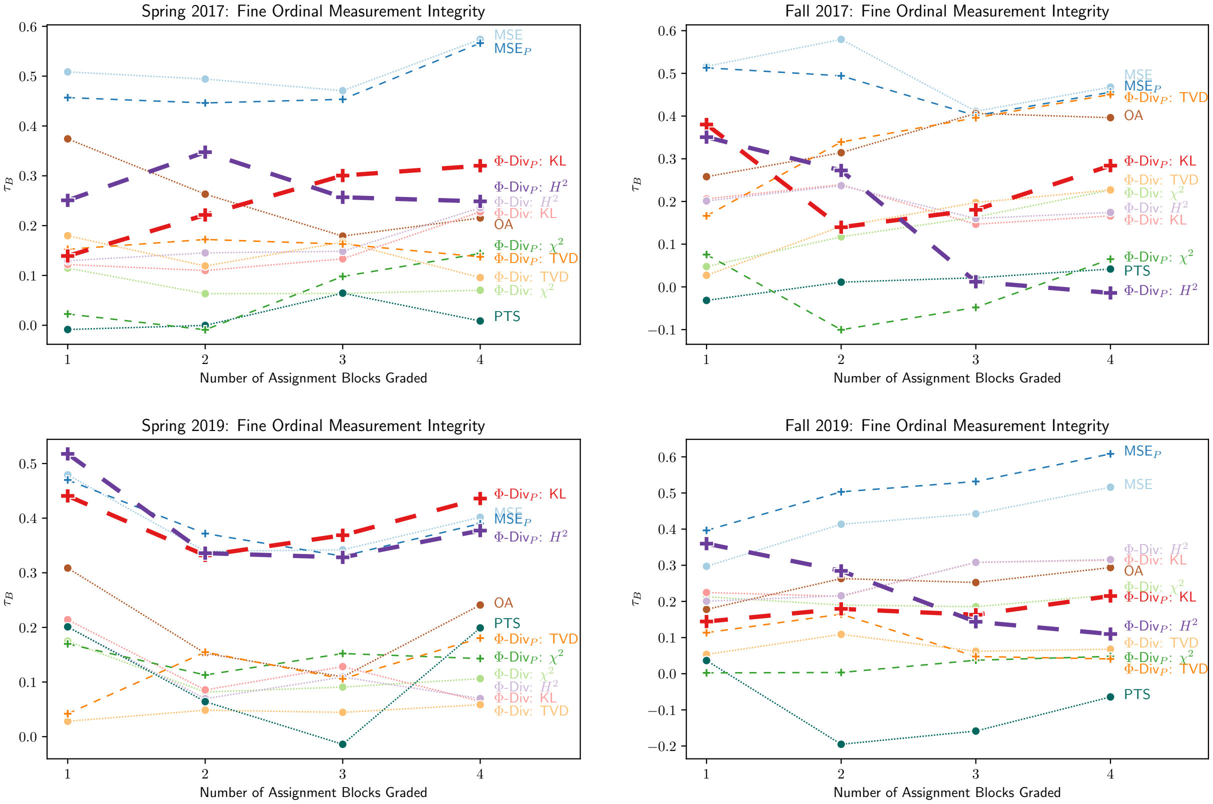


Figure 4.3: Quantifying Measurement Integrity with Real Data. For each semester, the average Kendall rank correlation coefficient (τ_B) between students' rewards and the average squared error of their reports from the true scores over 50 iterations of each mechanism.

mechanism across all numbers of assignment blocks and all semesters—yields the following top 5 mechanisms:

- AUCC: MSE_P, MSE, Φ-Div_P: KL, OA, Φ-Div: H^2.
- ρ: MSE_P, MSE, OA, Φ-Div_P: KL, Φ-Div: H^2.

Note, though, that as was the case with τ_B, it is not always the case that the differences in the average values between mechanisms are very meaningful. With AUCC, the two MSE-based mechanisms have very similar values and the last three mechanisms have very similar values. With ρ, the last two mechanisms have very similar values.

4.5 Quantifying Measurement Integrity Under an Alternative Conception of Report Quality

In Section 4.2.1, we adopt a simple notion of report quality: the squared distance between a report value and the true grade of the corresponding submission. Thus, the quality of a report is determined with respect to the report viewed as a fixed quantity. However, this fully *ex post* notion of report quality means that random chance plays a role in determining the quality of an agent's report. An agent who chooses a score uniformly at random may, by chance, submit a report that is close to or even exactly equal to a submission's true grade. In certain cases, it may be desirable to remove the role of chance in our definition, although it may result in a notion of report quality that is less intuitive. One way to accomplish this is to define report quality with respect to the report viewed as a random variable, where the randomness comes from uncertainty in the observation of a signal (and possible randomness used in the generation of a report after the observation of a signal). Then, a report can be considered high-quality if it is informative—i.e., likely to generate an outcome that is close to the underlying true grade.

In our model, assuming that agents report truthfully, *effort* determines the informativeness of an agent's report viewed as a random variable. As a result, another interesting notion of measurement integrity to consider is one where the quality of each agent's report is considered to be equivalent to their effort. Under this alternative regime, then, measurement integrity helps to quantify the degree to which mechanisms incentivize effort under the assumed data-generating process. Note that biased agents present a complication to this notion of quality, so in what follows we consider settings with and without agent bias. Ultimately, consistent with the peer assessment literature [74], we find that it is useful to

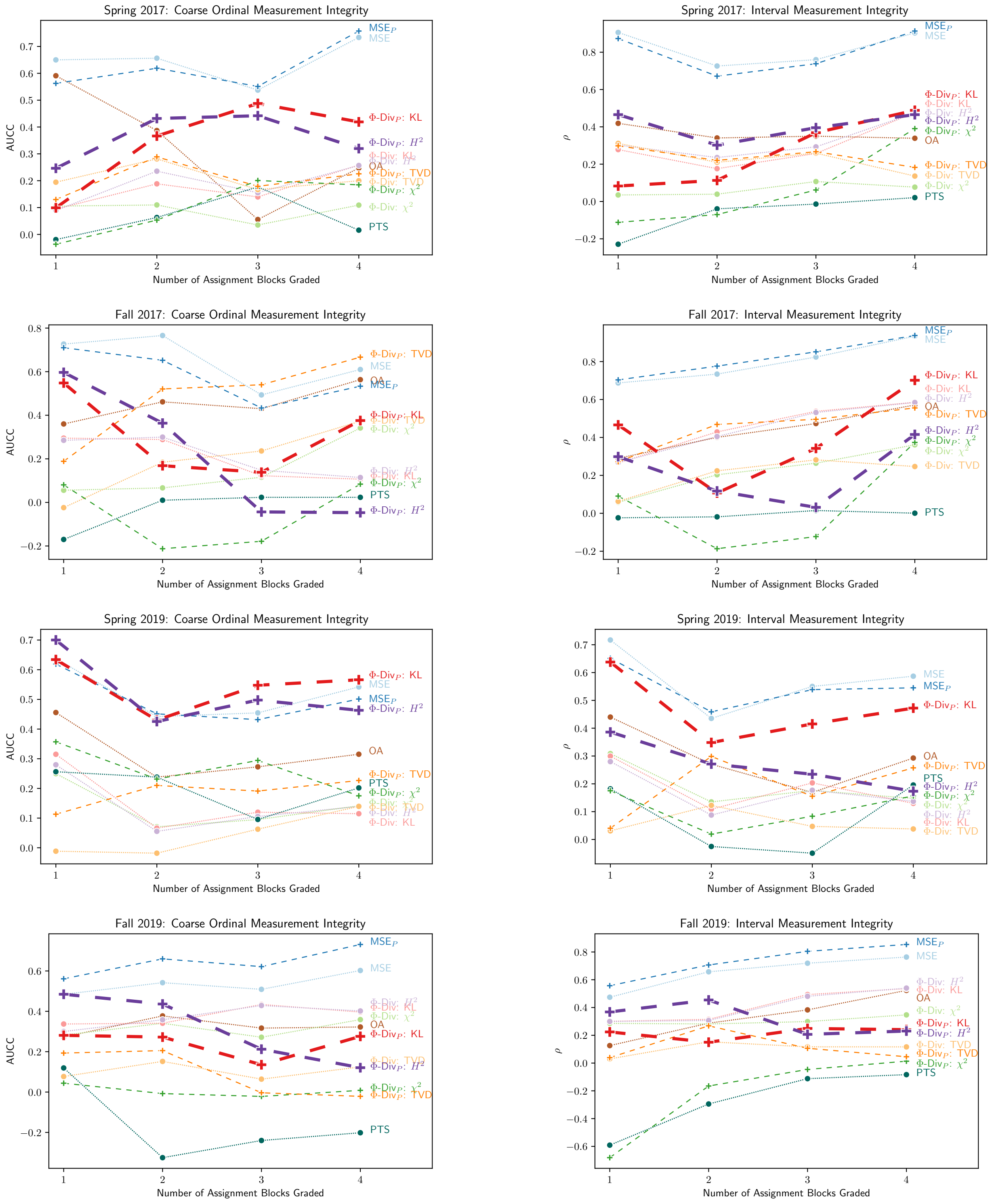


Figure 4.4: Quantifying Measurement Integrity with Real Data. For each semester, the average values of correlation functions (AUCC and ρ) over 50 iterations of each mechanism.

model bias and effort separately and correct for bias whenever possible (i.e., when using parametric mechanisms, which estimate agent bias).

Unbiased Agents. For an unbiased agent, the expectation of the latent distribution is equal to the ground truth score. Their signal for an assignment $s_{i,j}$ is a function of draws from the latent distribution $B\left(10, \frac{g^*_{i,j}}{10}\right)$.

We also find it is useful to consider a model with binary effort (in contrast to the model with continuous effort introduced in Section 4.2.1) to build intuition about the performance of peer prediction mechanisms according to this alternative notion of measurement integrity:

Binary Effort. Here, there are two types of agents: *active graders* exert high effort, *passive graders* exert low effort. Active graders receive signals created using three draws from their latent distribution. Passive graders receive signals created using a single draw from their latent distribution.

4.5.1 Computational Experiments with ABM

Under this alternative notion of report quality, we conduct additional computational experiments using our peer assessment ABM. As in Section 4.4, in order to explore measurement integrity in isolation, we require agents to report their signals truthfully. This is perhaps even more important in this context. In the experiments in Section 4.4, requiring agents to report truthfully affects the distribution of reports, but does not affect the notion of quality—whether or not a report was strategically manipulated, it is a high-quality report if it is close to the ground truth score. Here, strategic reporting would confound the relationship between effort and report quality that primarily interests us in this section, so it is especially important to quantify measurement integrity with respect to a data-generating process where agents report truthfully.

4.5.1.1 Methods

For each mechanism under consideration, the experiment consists of a number of simulated semesters, each of which proceeds as follows:

1. A population of 100 students is initialized.

2. For each of 10 assignments:

    i. Each student turns in a submission with a true grade drawn from the true grade distribution.

    ii. A random 4-regular graph is constructed with a vertex for each agent.

   iii. Each agent grades the submissions of their neighbors in the graph according to our peer assessment model.

   iv. The reported grades are collected by the mechanism, which assigns a reward to each student for their performance in peer assessment for that assignment.

3. The total reward accrued by each student, which is the sum of their rewards for each of the 10 individual assignments, is used to calculate the value of the relevant correlation function.

4. At the end of all simulated semesters, the mean, median, and variance of the values of the correlation functions are calculated.

In the binary effort case—where the performance of the mechanisms as the number of active graders varies is also of interest—we iterate the above procedure as we vary the number of active graders from 10 to 90 in increments of 10, simulating 100 semesters each time. In the continuous effort case, analogously to our experiments in Section 4.4.1, we vary the number of assignments per semester (from 1 to 15). In this section, though the number of students is fewer (100) and the number of simulated semesters is higher (100). Lastly, note that, in these experiments, when there are biased agents, we allow parametric mechanisms to correct for the estimated agent biases.

4.5.1.2 Correlation Functions

In this setting, our analysis is more focused than in Section 4.4. Here, the way that we model agent effort suggests natural notions of measurement and corresponding correlation functions to adopt. In settings where effort is binary—i.e., when there are only two types of agents—the natural goal is to be able to accurately classify each agent according to their type. Further, the classes are ordered—active graders are better than passive graders—so coarse ordinal measurement with AUC correlation (AUCC) is well-suited to this task.

In the settings where effort is continuous, the constraints of our model similarly establish a clear notion of measurement. In particular, the absolute magnitude of an agent's effort parameter λ, as well as the differences between two agents' effort parameters, does not have a straightforward interpretation in our model. Can we expect an agent with $\lambda = 1$ to be "twice as good" as an agent with $\lambda = 0.5$ in some clearly appreciable way? Or similarly, can we expect an agent with $\lambda = 1$ to be the same amount "better" than an agent with $\lambda = 0.5$ as an agent with $\lambda = 1.5$ would be "better" than an agent with $\lambda = 1.0$ in some clearly appreciable way? It seems unlikely that either question has an affirmative answer. Rather, it is most natural in our model to assign no meaning to the magnitude

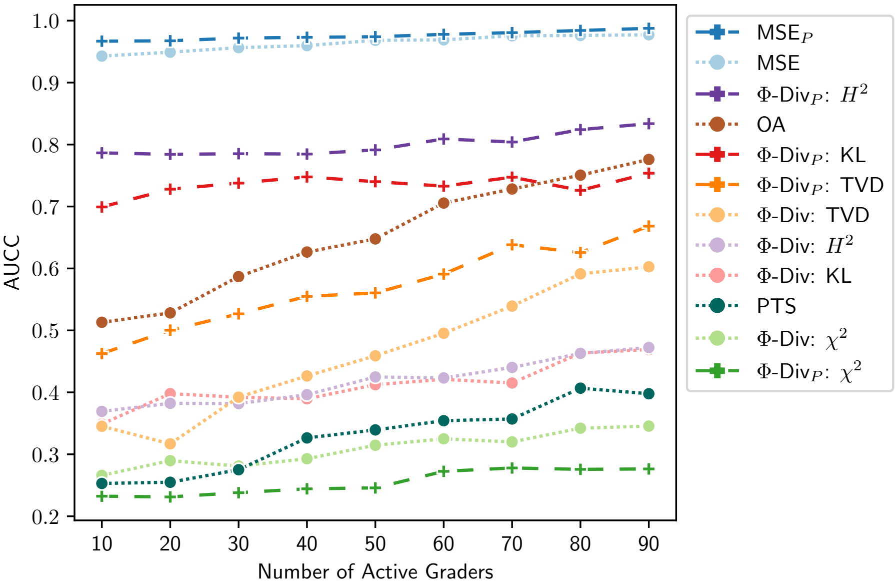


Figure 4.5: Binary Effort, Unbiased Agents. Averages values of AUCC as the number of active (i.e., high-effort) graders varies. The average for each number of active graders is taken over 100 simulated semesters.

of the effort parameter beyond that higher values for the effort parameter should tend to correspond to more accurate reports. This interpretation implies that the appropriate notion of measurement for the continuous effort settings is a *fine* ordinal scale, i.e., a ranking. Thus, we can use the Kendall rank correlation coefficient (τ_B) as our correlation function.

4.5.1.3 Results

Binary Effort, Unbiased Agents. The results are shown in Figure 4.5.

Many of the notable features of these results are similar to those from Section 4.4. It is largely the baseline mechanisms, not mechanisms from the peer prediction literature, that demonstrate high measurement integrity most reliably. One interesting difference from our other results is that the best-performing $\Phi\text{-Div}_P$ mechanism is different (H^2 instead of KL) for this notion of measurement integrity than for the notion we consider in Section 4.4.

Binary Effort, Biased Agents. The results for this setting are shown in Figure 4.6. Unsurprisingly, we find that in the presence of biased agents, the performance of the mechanisms that do not attempt to correct for agent bias, degrades significantly compared to settings with unbiased agents. This includes the MSE baseline and the best-performing non-parametric peer prediction mechanisms. The amount of degradation varies depending

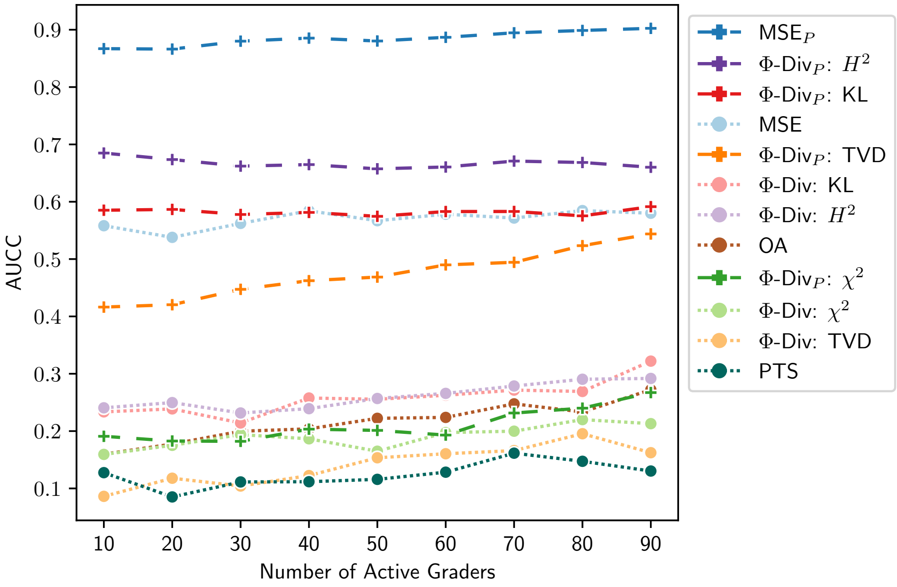


Figure 4.6: Binary Effort, Biased Agents. Average values of AUCC as the number of active (i.e., high-effort) graders varies. The average for each number of active graders is taken over 100 simulated semesters.

on the mechanism.

Continuous Effort, Biased Agents. The results for this setting, which uses the same underlying ABM that we describe in Section 4.2.1, are shown in Figure 4.7. Generally, the relative performances of the mechanisms is consistent with our other results. One notable difference is the larger magnitude of the increase in measurement integrity as the number of assignments per semester increases (at least among the best-performing mechanisms) under the random-variable notion of report quality compared to that in Section 4.4. This is not surprising: the relationship between observed reports and report quality in this setting is confounded by the noise in the data-generating process. However, there are still a few important implications for the mechanisms' ability to incentivize effort. First, these results suggest that the mechanisms may struggle to provide compelling incentives for effort in settings when data is relatively scarce. Second, measurement integrity for any number of assignments per semester in this setting is still relatively low even for the best mechanisms. This further suggests that, even with more data, the mechanisms may struggle to provide compelling incentives for effort at fine granularities. That is, the mechanisms may incentivize students to be in the top half or quarter of the graders (see above results), but may not do much, for example, to incentivize students to improve from being the tenth-best grader to the ninth-best or eighth-best grader.

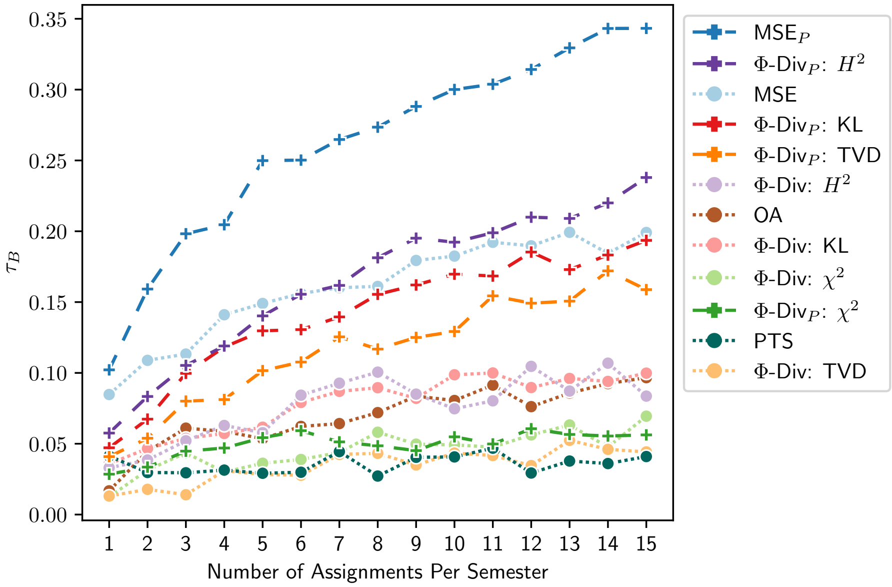


Figure 4.7: Continuous Effort, Biased Agents. Average values of τ_B as the number of assignments per semester varies. The average for each number of assignments per semester is taken over 100 simulated semesters

4.5.1.4 Conclusions

By setting aside incentive concerns and investigating the ability of the mechanisms to measure agents according to the latent underlying quality of their reports (i.e., effort), we are able to demonstrate that out-of-the-box peer prediction mechanisms from the literature generally fail to exhibit high levels of measurement integrity. This is significant, because mechanisms that fail to exhibit measurement integrity in a stylized agent-based model setting are unlikely to exhibit it in a real-world deployment. As a result, they will also be unlikely to provide compelling incentives for agents to exert effort in collecting high-quality information. These findings essentially corroborate our findings in Section 4.4, where we consider a simpler notion of report quality. There are some differences, however—e.g., the best-performing choice of Φ-divergence—which highlights the fact that the context-dependent choices that are made in selecting an appropriate notion of measurement integrity for a specific application matter for evaluating mechanisms.

4.6 Quantifying Robustness Against Strategic Reporting

We now turn to the second dimension of our analysis—robustness against strategic reporting—which has traditionally been the focus of the peer prediction literature. The key question we seek to answer is: to what extent can an individual agent improve their outcome, according to the rewards assigned by a given mechanism, by strategically manipulating their reports?

To explore this question in an analogous way to our exploration of measurement integrity, we first formally define the empirical estimand—*empirical robustness against strategic reporting*—that we adopt for the theoretical estimand of robustness against strategic reporting.

As in our definition of measurement integrity, suppose that P is a data-generating process for a given application. Recall that such a P may (and in this case, should) also describe how agents compute a *report* to submit to a mechanism as a function of their signal on a given task.[8] Now, suppose that an instance of P has been generated: a population of agents I, a set of tasks J with ground truth responses, an assignment graph G, and signals and corresponding reports for each edge $(i, j) \in G$. Let $\mathbf{u} = U(I, J, G, \mathbf{r}^M)$ be a vector of utilities. Suppose that agent i reports truthfully in this instance and receives utility u_i^T (the i-th component of $\mathbf{u}$). We are interested in quantifying how a random truthful agent's utility might change if they were to instead report strategically (according to some process specified by P). Thus, we consider an alternative instance of P where agent i reports strategically according to some process specified by P. In this alternative instance, i receives utility u_i^S—the i-th component of a utility vector $\mathbf{u}' = U(I, J, G', \mathbf{r}'^M)$, where G' may differ from G only in agent i's reports. In our case, we assume that each agent's utility is equal to the negative of their rank—i.e., the negative of the number of rewards greater than or equal to their own—among all students.

Definition 4.6.1 (Empirical Robustness Against Strategic Reporting)**.** The *empirical robustness against strategic reporting* of a peer prediction mechanism M with respect to a data-generating process P and a utility function u is

$$\underset{P,\,u}{\text{Robustness}}\,(M) = \mathbb{E}_{P,M,i}\left[u_i^S - u_i^T\right],$$

where i is chosen uniformly at random from among the agents who report truthfully under P.

[8] As in Section 4.1.2.1, we treat the signal and report of agent i on task j as properties of the edge $(i, j) \in G$.

Although this definition is quite flexible, in the setting of peer assessment, we expect that it is unlikely for students to expend the effort required to compute optimal deviations or play particularly complex strategies. Further, in educational settings, there are other ways to motivate honest, effortful grading, e.g., providing instruction and practice in grading accurately, that can complement the incentive properties of a peer assessment mechanism. Thus, in this line of inquiry, we seek to depart from the typical theoretical approach of considering all possible strategies. Instead, we focus on the relative performance of a few intuitive, easy-to-compute strategies.

4.6.1 Strategies

In addition to truthful reporting, we consider the following types of strategies:

First, there are *uninformed strategies*—strategies that do not depend on an agent's signal—which we consider primarily as robustness checks:

Report All 10s. Agents following this strategy constantly report 10, the highest possible score.

Revert to the Prior. Agents following this strategy constantly report μ, the expectation of the prior distribution of true scores (rounded to the nearest integer, $\lfloor \mu \rceil$, when applicable).

The more interesting strategies are *informed strategies*, which define a procedure by which agents manipulate their signals to generate their reports. We consider simple strategies for which there is some intuition as to why they might be present in or perform well in a peer assessment application:

Hedge. A more realistic strategy for incorporating prior beliefs than fully reverting to the prior is to hedge reports back toward the prior mean, μ. Piech et al. [74] find some evidence for this tendency in their peer grading data, which was collected from MOOCs. Agents following this strategy apply Bayesian reasoning by adopting a prior $\text{Beta}(\mu, 10-\mu)$ on p, the value of the ground truth score divided by 10. After receiving their signal, they update their prior and report 10 times the mean of their posterior distribution for p, which is given by $\frac{\mu+\text{signal}}{2}$, rounded to the nearest integer.

Fix Bias. Real students may have some indication of the direction of their bias—whether they tend to assign grades that are too high or too low—and attempt to correct for that bias. To model this, agents following this strategy are given the sign of their bias. At the beginning of a semester, they each draw a constant "bias correction" term β from the half-normal distribution that models the magnitude of biases drawn according to the bias

Signal	0	$1,2,3$	$4,5,6$	$7,8,9$	10
Report	0	3	6	$\lfloor\mu\rceil$	10

Table 4.3: Summary of the mapping of signals to reports for agents following the *Merge Signals* strategy. We write $\lfloor\mu\rceil$ to denote the mean of the prior distribution, μ, rounded to the nearest integer.

distribution $\mathcal{N}(0,1)$. For each submission that they grade, they report their signal plus or minus β—depending on the sign of their bias—rounded to the nearest integer in the report space.

Add Noise. On the other hand, students who do not have some indication of the direction of their bias, or who think the direction of their bias varies from submission to submission, might still try to perform some correction. Similarly, students might try to guess (without any outside information) whether their signal is above or below the average or their peers and try to adjust their report accordingly. The result of either of these actions would look a lot like adding noise to their signal to generate their report. To model this, agents following this strategy draw a value $\nu \sim \mathcal{N}(0,1)$ for each submission that they grade and report the sum of their signal and ν, rounded to the nearest integer in the report space.

Merge Signals. There is some evidence that when the report space is sufficiently large, students tend to under-utilize certain report values (particularly values that are low, but non-zero) [90]. To model this, agents following this strategy map the signal space to a lower-dimensional report space and report the outcome of applying that map to their signal. The map from signals to reports is detailed in Table 4.3.

4.6.1.1 Correcting for Bias

Recall that in Section 4.4, we did not correct for estimated biases in agents' reports when using parametric mechanisms. In this section, we will initially continue with this approach, so that our evaluations in each of the two dimensions of our analysis are conducted with respect to the same version of each (parametric) mechanism. Then, we will explore the consequences of this choice in Section 4.6.4.

4.6.2 Computational Experiments with ABM

In these experiments, we focus on the best-performing mechanisms from the measurement integrity experiments, including the baselines, with the goal of identifying mechanisms that perform well according to both dimensions of our analysis. However, the degree to which the remaining mechanisms create incentives for deviating from truthful reporting is still of interest, so we record the results of our experiments in this section with those mechanisms as well.

4.6.2.1 Methods

We explore how the incentives for deviating from truthful reporting change as the number of agents adopting some non-truthful strategy grows. We perform the following:

1. The number of strategic agents is varied from 10 to 90, in steps of size 10. At each step, we perform 100 iterations, each consisting of one simulated semester with 10 assignments:

    i. A population of 100 agents is initialized and a semester's worth of submissions and reports for grading them are generated, as in Section 4.4.

    ii. Rewards are assigned twice according to the given mechanism with a fixed random seed. In the first assignment, the number of truthful and strategic agents is as given by the current step. In the second assignment, one agent that reported truthfully in the first assignment is randomly selected. That agent modifies their reports according to the prescribed strategy. Due to the fixed random seed, every other factor is consistent with the first reward assignment.

    iii. The gain in rank achieved by that single agent, i.e difference in the ranks according to the two reward assignments computed by the mechanism, is recorded.[9]

2. The mean gain in rank over the 100 iterations is computed for each step.

A student's rank is calculated by counting the number of payments in the population of students that is greater than or equal to their own payment.

Note, in these experiments, we consider strategy profiles that are unlikely to arise organically from agents learning via repeated interactions with a mechanism and thus unlikely to be observed in real-world or laboratory data. Some of these, e.g., where nearly every student uses the *Report All 10s* strategy, strain the incentive properties of many mechanisms, even those that perform well against many of the other strategy profiles that we consider. This more comprehensive exploration of the space of strategy profiles that is possible in our experiments—though not as exhaustive as theoretical results, which often consider the entire space—is a useful advantage of applying ABMs.

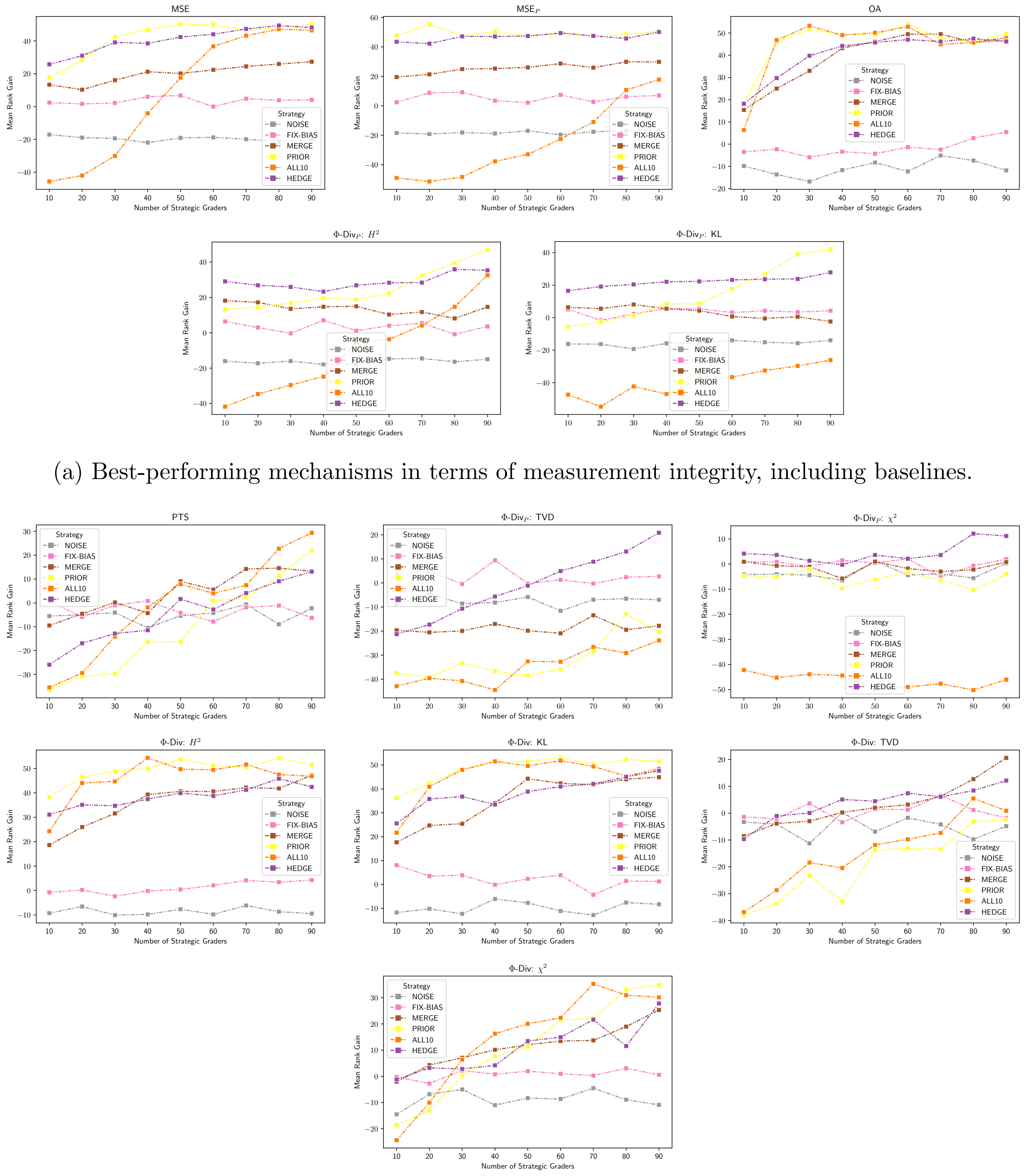
[9] For example, if the agent was given the 5th highest payment when they reported truthfully and the 10th highest payment when they reported strategically, a gain of -5, the difference in the ranks, would be recorded.

4.6.2.2 Results

As shown in Figure 4.8, we find that the individual incentives for deviating from truthful reporting are strong under the baseline mechanisms (including OA). They are also fairly strong for Φ-Div$_P$: H^2. Φ-Div$_P$: KL, by contrast, is more robust against strategic reporting. For the majority of strategies and strategy profiles that we consider, the rank gain achieved by deviating to strategic reporting is close to or less than zero under the Φ-Div$_P$: KL mechanism.

Predictably, not all strategies are equally effective against the mechanisms. There are some noteworthy strategies that stand out: *Hedge*, in particular, exhibits both high average rank gain (that does not depend too much on the number of other strategic agents) and relatively low variance (see Figure C.7 in Appendix C.5), which makes it a very attractive strategy for students deviating from truthful reporting. The uninformative strategies are also very effective, particularly when the number of strategic agents is high (this is especially true for *Report All 10s*). On the other hand, *Add Noise* is universally ineffective and *Fix Bias* is approximately neutral in all cases.

Results for the remaining mechanisms are shown in Figure 4.8b. We also record the *variance* of the rank gain for each mechanism (Figure C.7). One particularity of our approach in these experiments is that we consider only homogeneous strategy profiles (all non-truthful agents adopt the same strategy), which could potentially limit the generalizability of our results. This concern is somewhat mitigated by the fact that, for many mechanisms, robustness in our experiments is more or less binary. The degree to which mechanisms reward or punish strategic behavior varies, as shown in Figure 4.1, but mechanisms generally tend to either be susceptible to strategic behavior—rewarding it to some degree in nearly all cases that we consider in our experiments and having negative x-coordinates in Figure 4.1 and its analogues—or are robust against it—punishing it to some degree in most or nearly all cases except for a few extreme ones and having positive x-coordinates in Figure 4.1 and its analogues. It is primarily the Φ-Div$_P$: KL and Φ-Div$_P$: H^2 mechanisms, when not allowed to correct for bias, that do not fit neatly into this dichotomy. This consideration of strategy profiles is also not a concern in our experiments with the real data (Section C.5.2.2), since using real grades allows us to experiment with actual strategy profiles (whatever they may have been) followed by real students using peer grading in a course.



(a) Best-performing mechanisms in terms of measurement integrity, including baselines.

(b) Remaining mechanisms.

Figure 4.8: Quantifying Robustness with ABM. For each mechanism and each strategy, the mean gain in rank achieved, *ceteris paribus*, by a single agent changing their reports from truthful to strategic. The mean is taken over the outcomes of 100 simulated semesters as the number of other strategic agents varies in steps of size 10.

4.6.3 Computational Experiments with Real Data

4.6.3.1 Methods

As in our experiments with ABM, we continue to focus on the best-performing subset of mechanisms, while still recording the results of our experiments with the remaining mechanisms. Unlike in our experiments with ABM, however, we do not need to impose a strategy profile on the population of agents in this setting—the data were already generated according to some actual strategy profile adopted by the real students. As a result of this key difference, we modify the experiment described in Section 4.6.2.1 in the following manner.

For each semester in the real data, for each strategy, and for each mechanism:

1. The students, submissions, and reports for that semester are loaded from the data.

2. For each student s:

   i. Rewards are assigned twice according to the mechanism with a fixed random seed. In the first assignment, assignment of payments occurs without any modifications to the data. In the second assignment reports from student s are modified according to the prescribed strategy. Due to the fixed random seed, every other factor is consistent with the first reward assignment.

   ii. The gain in rank achieved by student s, i.e difference in the ranks according to the two reward assignments computed by the mechanism, is recorded.

3. The mean and variance of the gain in rank over all students is computed for each mechanism, for each semester.

Since we don't have access to a student's latent bias in the real data (and the true scores are noisy as a reference point) we do not consider the *Fix Bias* strategy in these experiments.

4.6.3.2 Results

The results of this experiment involve the mean gain (Figure 4.9) and the variance of the gain (Figure C.9) over the population of students achieved by each student deviating (one at a time) to each strategy in each semester.

As in our experiments with measurement integrity, we find that, at a high level, this experiment with the real data largely corroborates our analogous experiments with ABM. We find that the baseline mechanisms (including OA) are relatively susceptible to strategic behavior. Also, we again find that *Hedge*, and to a lesser extent *Revert to the Prior*, is an effective strategy. We also find that, aside from the parametric mechanisms, the PTS mechanism, and to some extent the non-parametric Φ-Div: TVD mechanism, are robust against strategic reporting for many or all semesters and strategies, whereas the remaining mechanisms are consistently susceptible to various kinds of strategic behavior. One interesting contrast with the results of our computational experiments with ABM is that, in our experiments with real data, it is the $\Phi\text{-Div}_P$: H^2 mechanism (not KL) that is more robust against strategic reporting.

4.6.4 Improving Robustness with Bias Correction

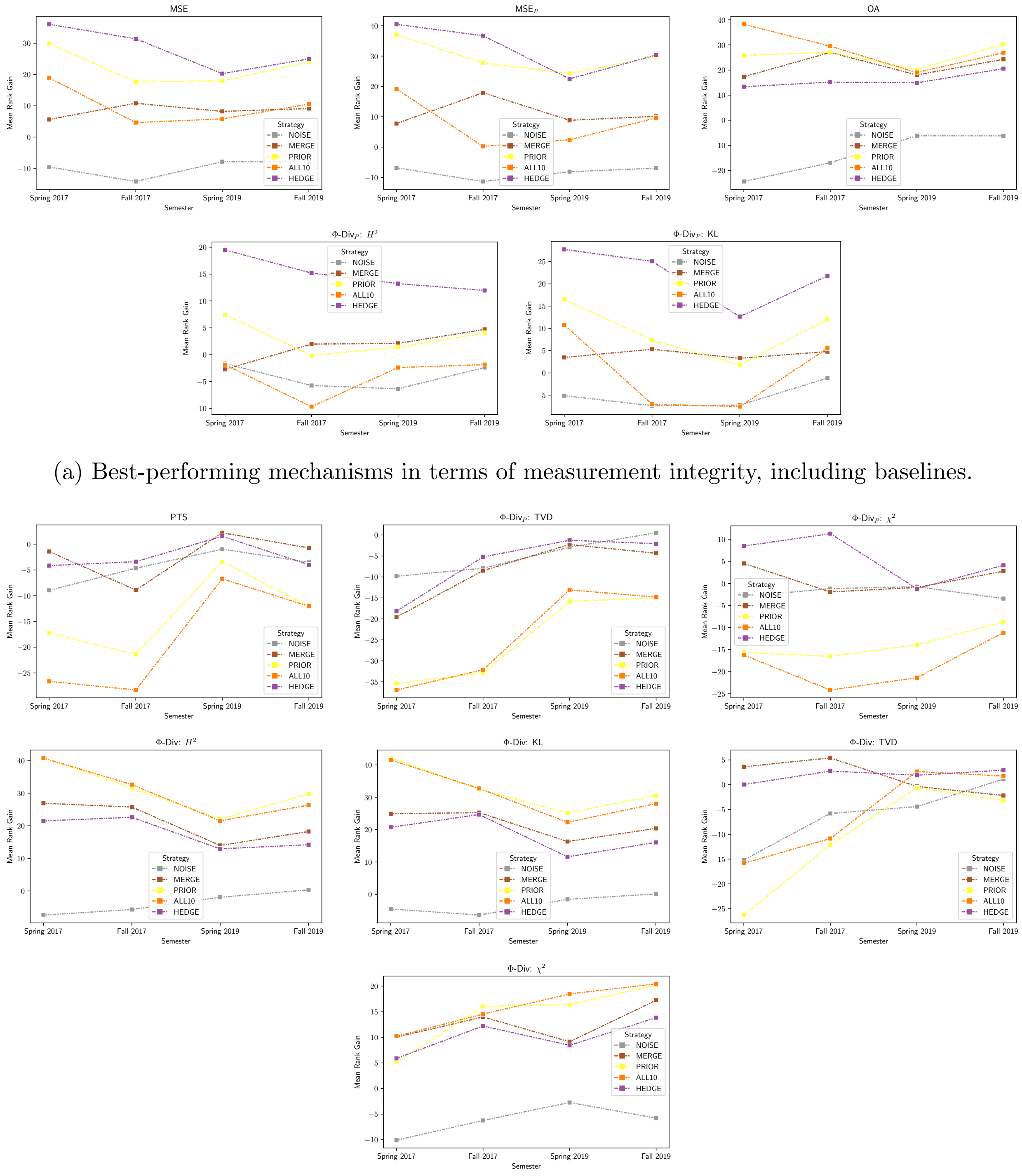
Previously, we have not employed bias correction for the parametric mechanisms in our experiments. In this section, we repeat our experiments for robustness against strategic behavior with the two $\Phi\text{-Div}_P$ mechanisms that we have highlighted and allow them to use the estimates of the bias in computing rewards.

Figure 4.10 shows (compared to Figures 4.8 and 4.9) that there is a drastic improvement in the robustness against strategic behavior of both mechanisms. The effect is particularly clear in the experiments with ABM, but is also quite noticeable for $\Phi\text{-Div}_P$: KL in the experiment with real data. This is quite a significant result—the mechanisms are robust against strategic behavior for strategy profiles that are far away from the truth-telling equilibrium strategy profiles that are embodied in the theoretical characterizations of peer prediction mechanisms. Moreover, as we show in Appendix C.5, this result holds for the other $\Phi\text{-Div}_P$ mechanisms.

However, this improved performance comes at a cost, which further highlights the trade-off between measurement integrity and robustness against strategic behavior that we identified in Figure 4.1. When we repeat the experiments from Section 4.4 with parametric mechanisms that correct for bias, we see that their measurement integrity decreases significantly. This is not a surprise—it is, as we anticipated previously, a consequence of defining quality in terms of raw report values. However, many intuitive quality functions have this property, and the trade-off inherent in choosing whether to correct for biases is an important consideration in those settings. In other settings, as we show in Section 4.5 when we explore a different perspective on report quality (which does not depend on raw report values), correcting for bias can be useful for improving performance in both dimensions of our analysis.

4.7 Measurement Integrity in the Presence of Strategic Agents

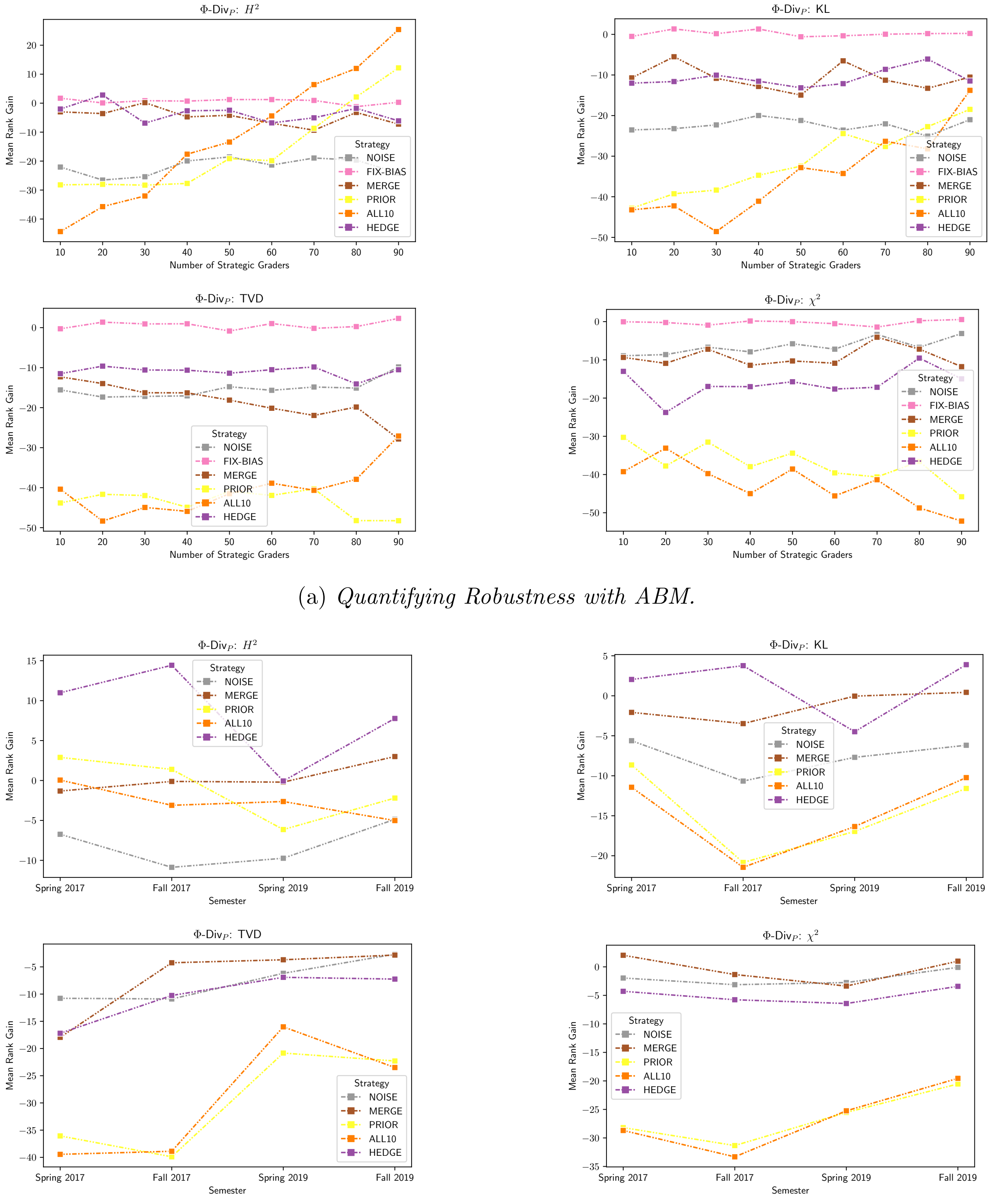
Strategic reporting confounds the relationship between the effort expended in observing a signal and the informative-ness of the resulting report. As a result, when we are interested



(a) Best-performing mechanisms in terms of measurement integrity, including baselines.

(b) Remaining mechanisms.

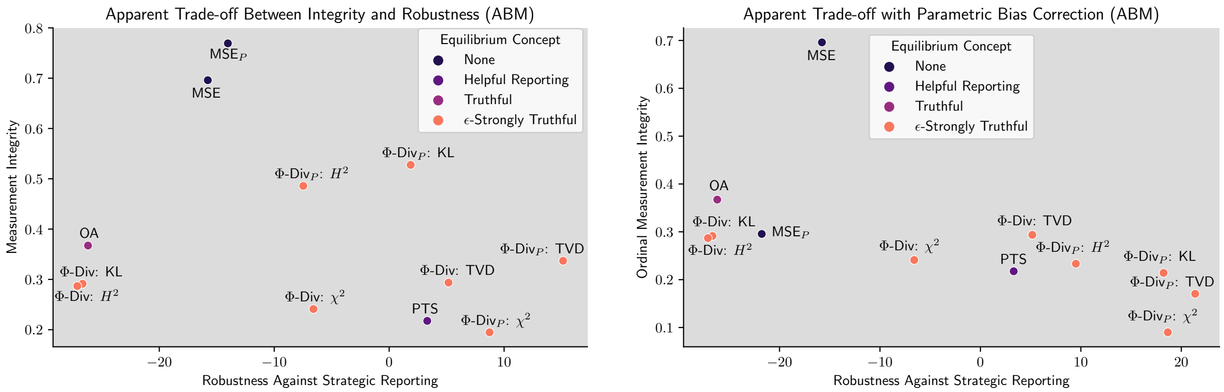
Figure 4.9: Quantifying Robustness with Real Data. For each mechanism, each strategy, and each semester (excluding *Fix Bias*), the mean gain in rank achieved, *ceteris paribus*, by a single agent changing their reports from truthful to strategic. The mean is taken over the population of students in the given semester.



(a) *Quantifying Robustness with ABM.*

(b) *Quantifying Robustness with Real Data.*

Figure 4.10: The Φ-Div$_P$ mechanisms are significantly more robust against strategic reporting when they are able to correct for estimated agent biases.



(a) *Without Bias Correction.*

(b) *With Bias Correction.*

Figure 4.11: The trade-off between measurement integrity and robustness against strategic reporting is illustrated quite starkly by the effect of bias correction in the Φ-Div$_P$ mechanisms.

in incentivizing effort, it is important to understand how strategic reporting might interfere with our ability to identify (and subsequently reward) high-effort agents. Our goal in this experiment is to quantify the effect of this interference. How is measurement integrity, in terms of the quality of the rankings of agents according to their continuous effort parameters (i.e., in the setting of Section 4.5), affected when agents are allowed to report strategically? To explore this question, we focus on the 3 best-performing mechanisms from our experiments in Section 4.5.1 and on the *informed strategies* from Section 4.6.1.

4.7.1 Methods

For each strategy, we replicate the measurement integrity experiment in the *Continuous Effort, Biased Agents* setting from Section 4.5.1 while varying the number of strategic agents from 0 to 100 in steps of 10. We simulate 100 semesters at each step.

4.7.2 Results

The results for this experiment are shown in Figure 4.12.

Although generally in peer prediction, the focus is on deterring strategic manipulation of reports, not all strategic manipulations are necessarily equally damaging to a given mechanism. Uninformed strategies are clearly detrimental, since the mechanism receives no information about the signals of agents following those strategies. However, for some informed strategies, it is possible that the mechanism might still be able to demonstrate relatively high measurement integrity even if those strategies become common among the agents. For

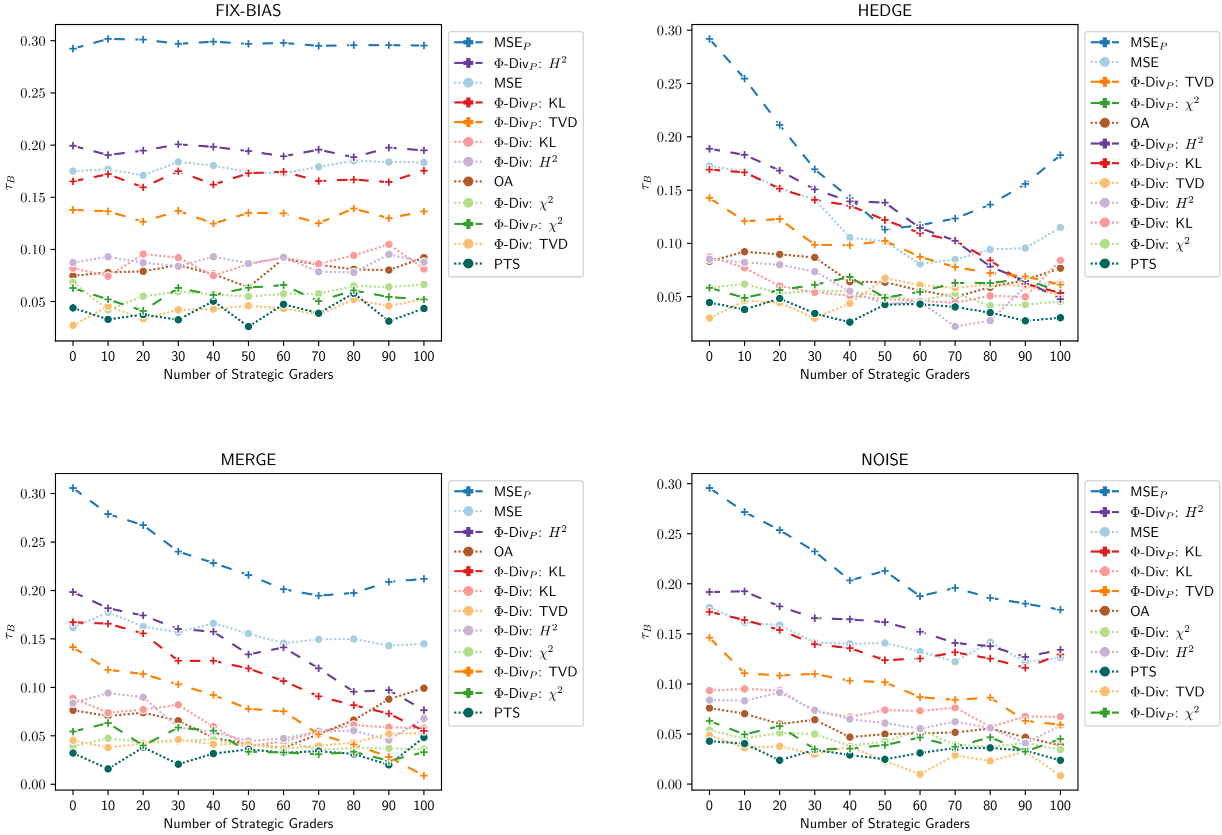


Figure 4.12: Measurement Integrity in the Presence of Strategic Agents. Average values of τ_B for each informative strategy as the number of strategic agents varies. The average for each number of strategic graders is taken over 100 simulated semesters.

example, it is reasonable to expect the *Fix Bias* strategy to only minimally affect the performance of parametric mechanisms, which take bias into account. Indeed, we find that this is the case in Section 4.7.2, which shows that the performance of each mechanism is fairly constant as the number of agents adopting the *Fix Bias* strategy grows.

On the other hand, though, some strategies do significantly affect the performance of the mechanisms that have relatively high measurement integrity when there is no strategic reporting. In Section 4.6, we singled out *Hedge* as a particularly attractive strategy for agents seeking to improve their rank under the MSE and MSE$_P$ mechanisms. In Section 4.7.2, we see that additionally, the number of agents adopting the *Hedge* strategy has a clear effect on the measurement integrity of those mechanisms (although the effect is not monotonic). When 50 agents adopt the *Hedge* strategy, for example, Φ-Div$_P$: H^2 supplants MSE$_P$ as the mechanism with the highest ordinal measurement integrity.

4.8 Related Work

4.8.1 Peer Prediction and Information Elicitation

For simplicity, we have so far used the generic term "peer prediction" primarily as a shorthand for a more specific setting—the elicitation of categorical signals without verification on multiple tasks, which was first explored (independently) by Dasgupta and Ghosh [14] and Witkowski and Parkes [115]. However, in the information elicitation literature, peer prediction mechanisms have been proposed in a variety of settings [23]. We choose to focus on one particular setting, because particularities of different settings necessitate different strategies for designing effective mechanisms. As a result, mechanisms for different settings will plausibly exhibit different empirical behaviors for properties like measurement integrity and robustness against strategic reporting. We leave the extension of our core ideas to other settings for future work.

The existing peer prediction literature, both in our setting and more broadly, is focused primarily on theoretical properties related to robustness against strategic behavior. However, there has been some work that explores incentive compatibility from other perspectives. Gao et al. [24] take an experimental approach and find evidence that agents are willing and able to exploit peer prediction mechanisms by coordinating on uninformative reports instead of truthful reports. Kim [45] draws on machine learning techniques to propose three "empirical" peer prediction mechanisms and demonstrates their robustness against three uninformed strategies using simulated data in a highly stylized setting that facilitates accurate learning of the quantities required by the mechanisms. Shnayder et al. [92] use replicator dynamics—a simulation-based approach that is quite different from our approach with ABM and real data—to quantify desirable incentive properties of peer prediction mechanisms.

Some recent work has studied properties of peer prediction mechanisms beyond incentive properties. The result from this line of work most closely related to measurement integrity is by Kong [46]. Kong establishes a theoretical "information evaluation" property for the DMI mechanism, which involves a theorem showing that (under certain assumptions) an agent's expected payment under the DMI mechanism, given their signal, is proportional to a DMI-based measure of the quality of their report. However, this property still allows for the possibility that, due to noise, the relationship between rewards and qualities is not very strong at the population level. The DMI mechanism is not well-suited to the data-generating processes that we consider. However, under a data-generating process where data is more abundant, it would be relatively straightforward to adapt the components of the theoretical result to define a version of the information evaluation property as an instance of measurement integrity and explore the population-level relationship between rewards and qualities empirically.

Properties beyond incentives have also recently garnered attention in other information elicitation settings, many of which are distinct from the broader peer prediction paradigm because they assume that it is possible to access to ground truth information. In the setting of general crowdsourcing with limited access to ground truth, Goel and Faltings [29] advance a novel notion of fairness—that the expected payment of each agent be directly proportional to the accuracy of their reports and independent from the strategy and accuracy of the other agents. The philosophical point embedded in this definition—that fair mechanisms must reward agents independently from the reports of other agents—would imply that any peer prediction mechanism is necessarily unfair. We do not accept this premise; our results demonstrate that, at least in some circumstances, certain peer prediction mechanisms have the ability to reliably reward agents fairly, even though they rely on the reports of other agents. In the forecasting setting, Li et al. [50] and Neyman et al. [64] both consider optimizing for properties related to incentivizing effort when selecting a *proper scoring rule* with which to score forecasts.

Another important work related to proper scoring rules is Liu et al. [55], which considers the elicitation of forecasts (as opposed to the elicitation of categorical reports) without access to ground truth information. They propose a family of mechanisms, surrogate scoring rules (SSRs), which extend useful properties of proper scoring rules to the setting without verification. These properties include robustness against strategic behavior and "quantifying the value of information," which is similar to Kong's "information evaluation" property for the DMI mechanism, above. In addition to theoretical exploration of this latter property, they conduct experiments that quantify the extent to which the scores assigned by various mechanisms—including SSRs and certain peer prediction mechanisms from our setting (adapted to elicit forecasts instead of categorical reports)—correlate empirically with various metrics of forecast quality in each of a single draw from a variety of data-generating processes. Their experiments do not substantially differentiate the (adapted) peer prediction mechanisms they consider, nor do they suggest strategies by which to improve their performance. Lastly, they do not consider the interaction between the properties of robustness against strategic behavior and quantifying the value of information in practice.

4.8.2 Peer Prediction and Peer Assessment

There has also been work that explores the specific application of peer prediction to peer assessment. Shnayder and Parkes [90] empirically analyze a limited set of peer prediction mechanisms using real MOOC data. Importantly, they show that some of the underlying assumptions made about data-generating processes in the theoretical literature—the self-dominating and self-predicting assumptions (see full version)—are likely to be violated in real peer assessment settings, especially when the report space is large. They also discuss the importance of considering factors beyond expected rewards (e.g., the variance of rewards under a mechanism) in settings where fairness concerns are salient.

Radanovic et al. [77] also apply peer prediction to peer assessment. In particular, they propose a particular mechanism, the Peer Truth Serum for Crowdsourcing (PTSC) mechanism[10], and conduct an experiment where that mechanism and some baselines are used to reward peer graders with extra credit when grading a set of quizzes in a course on artificial intelligence at EPFL. They find that students who are rewarded using the PTSC mechanism grade more accurately than those rewarded using the baselines. However, they do not consider the relationship between the rewards and some measure of grading accuracy, nor whether the improvement in grading accuracy was the result of decreased strategic behavior or increased effort in grading.

More broadly, peer assessment is often touted as a natural application for the peer prediction paradigm. However, the typical approach in the literature has been to design mechanisms that are as generic as possible. The resulting mechanisms can be ill-suited to the challenges of a specific application, like those we identify for peer assessment in Section 4.1.3.1. Indeed, existing mechanisms often rely on collecting lots of data and compensating agents with rewards that exhibit high variance. Both these characteristics help explain why out-of-the-box peer prediction mechanisms tend to have low measurement integrity in our experiments.

On the other hand, certain works have been skeptical of the application of peer grading to peer assessment. Gao et al. [25] and Zarkoob et al. [122] explore how limited spot-checking (in the form of "ground-truth" grading by teaching assistants on certain assignments) can (theoretically) incentivize truthful reporting in grading in a simple model. Surprisingly, Gao et al. [25] find that, compared with spot checking alone, supplementing spot-checking with peer prediction increases the number of spot-checks required to obtain the desired theoretical incentive properties. However, they do not consider how the objective of minimizing the number of spot checks may be in tension with rewarding students fairly. In more practical explorations of the utility of spot-checking, Wright et al. [116] and Zarkoob et al. [123] propose and refine, respectively, a peer grading system that is centered around the deployment of teaching assistants to improve the quality of feedback. We leave the application of our ideas to mechanisms that incorporate spot checking to future work.

[10] We note that the PTSC mechanism is essentially a prototype of the PTS mechanism we consider; the latter is more suited to our setting.

4.8.3 Modeling Peer Assessment

The construction of our agent-based model is most influenced by the analysis of MOOC data on the platform Coursera conducted by Piech et al. [74]. Piech et al. (and subsequent work, e.g., Zarkoob et al. [124]) propose a sequence of increasingly complex parametric statistical models of peer assessment. Piech et al. show that estimating the parameters of each of their models is useful for estimating the true grades of student submissions that have been evaluated by peers. Grade estimates computed using their models are found to outperform grade estimates computed by the algorithm used by Coursera at the time. The inclusion of grader biases in each of their models is found to be the most significant single factor underlying this result.

Our model (Section 4.2.1) is not one proposed by Piech et al. (or their successors), but it is structurally similar to their model $\mathbf{PG}_1$, which strikes a good balance between simplicity and performance in their analysis. The decision to propose a new model stems from a few important points:

1. Their models are continuous. In practice, though, essentially all assignment scores, rubrics, etc. are discrete. Thus, the "true grade" of a submission, and each grader's signal, in a peer assessment model should be discrete.

2. Nearly all peer prediction mechanisms require a discrete report space.

3. It allows us to use model $\mathbf{PG}_1$ in the implementation of our parametric peer prediction mechanisms (Section 4.3.3) without giving the mechanisms unrealistically accurate information about the underlying process by which true grades and reports are generated.

4.9 Discussion

We introduced measurement integrity as a novel property to consider in the design and analysis of peer prediction mechanisms. Alongside the more well-studied property of robustness against strategic reporting, measurement integrity plays an important role in understanding their practical performance. We focused on quantifying these properties empirically, using computational experiments with both an ABM and with real data. As a result, we were able to meaningfully differentiate mechanisms from the peer prediction literature in a way that has not been possible with theoretical analysis alone. Ultimately, we identified an apparent trade-off between our two dimensions of analysis (Figure 4.1) and found that parametric peer prediction mechanisms were best able to balance the two properties that characterize those dimensions.

Our unambiguous results suggest that our methodology itself—performing computational experiments to quantify mechanisms' empirical properties—is useful for investigating desiderata, including, but not necessarily limited to, measurement integrity and robustness against strategic reporting. In particular, that our approach facilitates more direct comparisons between mechanisms and uncovers consequences of implementation choices that are often abstracted away in theoretical analysis may be useful to practitioners looking select a peer prediction mechanism to deploy in a particular application.

Our results are driven by estimates of empirical quantities that are specific to the particular peer assessment data-generating processes that we consider. However, just as we find that they accord across these related data-generating processes, we expect many of those results to be relevant in other similar contexts. Our ABM is structurally similar to a model from the peer assessment literature that was validated using a large peer grading dataset from Massive Open Online Courses (MOOCs) by Piech et al. [74]. As a result, it is reasonable to expect that the qualitative results of our experiments with that ABM should, at the very least, extend to that important class of peer assessment settings. The corroboration of these results by similar experiments with real peer grading data (not from a MOOC) suggests even broader applicability. Additionally, the relative scarcity of data for any particular grading task or agent in the peer assessment setting (Section 4.1.3.1) appears to be a significant driver of many of our results. This suggests that much of the intuition that we gain from our case study in peer assessment is likely to generalize to other settings where data is similarly sparse. Future theoretical work should explore the design of mechanisms—including parametric mechanisms, which seem well-suited for this purpose—that are less reliant on an abundance of data than the current mechanisms from the peer prediction literature.

We also expect that the trade-off between measurement integrity and robustness against strategic behavior will remain in other settings. In Appendix C.4, we discuss experiments with additional mechanisms that have not yet been studied in the peer prediction literature. We find the trade-off to be persistent—none of the novel mechanisms were able to significantly extend the Pareto frontier established by the mechanisms from the current peer prediction literature. On the other hand, particular challenges of peer assessment, like the scarcity of data, are not a concern in some settings of interest. This indicates that a mechanism's performance in our experiments will not necessarily predict its performance universally.

CHAPTER 5

Conclusion

In this dissertation, we have explored the application of information elicitation and aggregation techniques to crowdsourcing and peer assessment in three works that employ simulation-based approaches. In each work, we have proposed specific directions for future research that follow from particular ideas or results developed in that work. To conclude, then, we take a broader view and discuss the impact of simulation-based approaches and suggest directions for future work at a high level. Epstein [20] poses and subsequently provides several answers to the question: “Why Model?” Since complex models are often explored via simulation, and simulation often is driven by a model, Epstein’s answers are readily applicable to the narrower question “Why Simulate?” and serve as a useful guide for our discussion.

In this dissertation, we have exemplified the following reasons to simulate:

Expose prevailing wisdom as incompatible with available data. In Chapter 2, we saw that simulation-based statistical hypothesis tests suggested that certain assumptions from the literature on designing label aggregation algorithms were implausible (as in the case of category-independent noise) or rarely plausible (as in the case of workers who exhibit significant expertise) in real crowdsourcing data. We also found many cases where an assumption was plausible in some data sets and implausible in others, implying that it might be fruitful to design and evaluate label aggregation algorithms in a way that is specific to a given application.

Bound (bracket) outcomes to plausible ranges. In Chapter 3, we used simulated peer grading data to demonstrate that, in the context of the real data which our simulations were calibrated to resemble, peer grading is unlikely to definitely outperform the traditional instructor grading paradigm. More positively, we showed that multiple grading interventions whose effect is to introduce additional information into the peer grading system can be expected to produce significant gains in the quality of the grades allocated by that system.

Demonstrate tradeoffs / suggest efficiencies. In Chapter 4, our main result was the persistent trade-off that we found between measurement integrity and robustness against strategic behavior. Another significant result was the efficiency gain that resulted from supplementing peer prediction mechanisms with parametric statistical models.

We also **challenge[d] the robustness of prevailing theory through perturbations** and found that many peer prediction mechanisms were indeed robust against strategic reporting even for strategy profiles that differed significantly from the truth-telling equilibria at the heart of theoretical robustness properties.

The following reasons to simulate provide a road map to future work that we believe will be fruitful:

Illuminate core dynamics. Two clear directions for further application of simulation-based techniques, particularly in the context of peer prediction, are:

1. Quantifying the extent to which peer prediction mechanisms create incentives for effort in obtaining a signal, in addition to incentives for reporting the observed signal truthfully. This idea was introduced briefly in Chapter 4. Contemporaneously, Zhang and Schoenebeck [125] and Xu et al. [117] have made progress in this area. Zhang and Schoenebeck [125] proposed a property for mechanisms called *sensitivity*, which relates to a mechanism's ability to incentivize effort, but can be difficult to quantify. Xu et al. [117] demonstrated that measurement integrity can be used to quantify incentives for effort by establishing a connection between measurement integrity and sensitivity.

2. Quantifying the extent to which peer prediction mechanisms facilitate agents who learn from repeated interactions with a mechanism to learn to report truthfully. In our experiments quantifying robustness against strategic behavior, we tested robustness against a limited set of particular strategies. A more robust approach (in terms of which strategies are tested in the experiments) would be to allow agents to learn a reporting strategy based on a small number of repeated interactions with the mechanism. A crucial next step in this direction is to obtain a realistic model of learning behavior for people interacting with peer prediction mechanisms in the context of particular applications.

Train practitioners. In the context of crowdsourcing, our simulation-based techniques led to a specific suggestion for practitioners looking for a label aggregation algorithm—try to understand the features of their data (including the features concerning worker errors that we identify) and evaluate candidate aggregation algorithms with test data sets that are similar according to those features.

In the context of peer assessment, we use simulation to set up frameworks for predicting the performance of different peer grading systems. These predictions can be a useful guide to implementing a grading system. However, it is not obvious how to adapt the prediction methods into methods for evaluating the observed performance of grading systems. In Chapter 4, we showed that our simulations could be adapted to perform computational experiments with real data, but the results were noisy and would be hard to interpret with the complementary results with ABM. In Chapter 3, we rely on accessing the true scores of submissions, which are unobservable in the real world, to evaluate forecast quality. It is unclear if there is an observable quantity in real data that would be a useful proxy for the true score in applying our framework to observed data.

In many cases, though, it is evaluation methods, not predictors, that would be the most directly useful to practitioners. For example, to persuade students about the quality of peer grades in a course, it would be even more useful to have a benchmark for determining that the peers (probably) *have outperformed* the instructor benchmark in grading the previous assignment than to have the benchmark we introduced, which determines whether peers (probably) *will outperform* the instructor benchmark in grading a future assignment.

Guide data collection. In Chapter 3, we found that a concrete question for future empirical work is to explore the degree to which various interventions can, in the context of model $\mathbf{PG}_Z$ (Section 3.2), improve either the distribution of effort probabilities or of reliabilities in a population of students. More generally, though, the results of simulation-based experiments can be more widely applicable when there is data from a variety of contexts with which to validate models of the data-generating process, select hyperparameters, etc.

Educate the general public. Ultimately, the goal of the lines of research followed in this dissertation and proposed for future work is to encourage the adoption and application of insights from the literature on information elicitation and information aggregation to settings where they will be useful in the real world. As we have seen, simulation can be a powerful tool to advance that goal.

APPENDIX A

Supplementary Material for Chapter 2

A.1 A Note on the Dimensionality of IRT Ability Parameters

As is discussed in Section 2.5.2.1, a major assumption underlying IRT model-fitting procedures is that the correct dimension for the ability parameters is specified. The IRT literature includes a few procedures that are designed to indicate whether ability parameters in a given data set are plausibly multidimensional, but those methods are designed for settings where nearly every participant responds to nearly every item. They do not readily generalize to crowdsourcing settings where each worker tends to only complete a small subset of the tasks. We attempted to adapt one such procedure—DIMTEST (See [79, Ch. 7])—to our setting, but the resulting procedure failed to reliably distinguish between synthetic data generated using unidimensional and multidimensional IRT models.

A.2 Discussing Terms Used in Table 2.8

Category-Dependent Errors. **Strong** means that the p-value for using randomization inference to test the null hypothesis of category-independent errors was below 0.05. **Very Strong** means that the observed test statistic was more extreme than every test statistic generated under the randomization inference procedure.

Task Heterogeneity (Intra-Category). **Weak** means that the p-value for using randomization inference to test the null hypothesis of task homogeneity was above 0.05 in at least one category and the data were best fit by the DS model according to both fit comparisons (10FL and BIC). **Moderate** means that either the p-value for using randomization inference to test the null hypothesis of task homogeneity was below 0.05 in both categories, despite the DS model providing the best fit for the data (as in the case of HCB) or that the

p-value for using randomization inference to test the null hypothesis of task homogeneity was below 0.05 in at least one category and the data were best fit by a CIRT model according to at least one fit comparison[1] (as in the case of BM). **Strong** means that the observed test statistic was more extreme than every test statistic generated under the randomization inference procedure and that the data were best fit by a CIRT model according to both comparisons.

Worker Heterogeneity, Model-Agnostic. **Moderate** means that the p-value for using randomization inference to test the null hypothesis of worker homogeneity was below 0.05 in at least one category. **Strong** means that the p-value for using randomization inference to test the null hypothesis of worker homogeneity was below 0.05 in both categories.

Worker Heterogeneity, Model-Informed. **Moderate** means either that the p-value for testing the null hypothesis of unimodality of the estimated distribution of logit-probabilities of correctness was below 0.05 for one of the modality tests (as in the case of RTE, TEMP, and WB) or that the p-value for testing the null hypothesis of worker homogeneity using model-informed resampling was below 0.05 (as in the case of BM, WB, WVSCM, and SP). **Strong** means that the p-value for testing the null hypothesis of unimodality of the estimated distribution of logit-probabilities of correctness was below 0.05 for both of the modality tests.

Expertise. **Weak** means that the estimated distributions of logit-probability of correctness were either apparently unimodal or plausibly multimodal (according to one modality test, but not both) with density that drops off relatively quickly from the largest mode, which is also the right-most apparent mode. **Moderate** means that the estimated distributions of logit-probability of correctness were plausibly multimodal according to both modality tests (as in the case of HCB). **Strong** means that the estimated distributions of logit-probability of correctness were plausibly multimodal (according to at least one modality test) with the largest mode not being the right-most apparent mode (as in the case of WB).

[1] Particularly if the method of comparison for which a CIRT model provided the best fit were 10FL, to which we give slightly more weight than BIC.

APPENDIX B

Supplementary Material for Chapter 3

B.1 Derivation of Instructor Forecasts

Recall the model for instructor reported grades:

$$\begin{aligned}
\text{(True grades)} \quad & S_u \sim \mathcal{N}\left(\mu_s, 1/\tau_s\right); \\
\text{(Reliabilities)} \quad & T^I \sim \mathcal{G}\left(\alpha_\tau^I, \beta_\tau^I\right); \\
\text{(Biases)} \quad & B^I \sim \mathcal{N}\left(0, 1/\tau_b^I\right); \\
\text{(Instructor grades)} \quad & G_u^I \sim \mathcal{N}\left(s_u + b^I, 1/\tau^I\right); \\
\text{(Reported grades)} \quad & r_u^I = n_Z\left(g_u^I\right),
\end{aligned}$$

where, n_Z is a function that rounds a real number to the nearest integer in the set Z (rounding up).

For a particular submission u, we can decompose the instructor grade G_u^I into its expectation s_u and a noise term $N \sim \mathcal{N}\left(b^I, 1/\tau^I\right)$.

If b^I were known, then $(N_u^I, T^I) \sim$ Normal-Gamma $\left(b^I, 1, \alpha_\tau^I, \beta_\tau^I\right)$ and the marginal distribution of the noise term would follow a location-scale t-distribution (also known as a non-standardized Student's t-distribution): $N_u^I \sim lst\left(b,\, \beta_\tau^I/\alpha_\tau^I,\, 2\alpha_\tau^I\right)$.

However, since b^I is unknown, we can obtain the marginal density of N_u^I by marginalizing the density of the location-scale t distribution over the possible values of b^I. Thus, the density of the noise term is given by

$$P_N(n) = \int_{-\infty}^{\infty} t\left(n \,|\, b,\, \beta_\tau^I/\alpha_\tau^I,\, 2\alpha_\tau^I\right) \cdot \varphi\left(b \,|\, 0, 1/\tau_b^I\right)\, db,$$

where $t\left(n \,|\, \mu,\, \tau^2,\, \nu\right)$ is the density of $lst\left(\mu,\, \tau^2,\, \nu\right)$ and $\varphi(\cdot \,|\, \mu, \sigma^2)$ is the density of $\mathcal{N}\left(\mu, \sigma^2\right)$.

Now, we can turn our attention to computing a posterior over a submission's true grade given one instructor report for that submission. Recall, the reported grade is $r_u^I = n_Z(g_u^I) =$

$n_Z(s_u + n_u^I)$.

Applying Bayes' rule:

$$\begin{aligned} P_{S|R^I}(s|R^I = r) &= \frac{P_{R^I|S}(r|S = s) \cdot P_S(s)}{P_{R^I}(r)} \\ &= \frac{\Pr_N[n_Z(s + n) = r] \cdot P_S(s)}{P_{R^I}(r)} \\ &= \frac{\Pr_N\left[n \in [\underline{g}(r) - s,\, \overline{g}(r) - s)\right] \cdot P_S(s)}{P_{R^I}(r)} \\ &= \frac{\Pr_N\left[n \in [\underline{g}(r) - s,\, \overline{g}(r) - s)\right] \cdot \varphi(s \,|\, \mu_s, 1/\tau_s)}{P_{R^I}(r)}, \end{aligned}$$

where $\overline{g}(r)$ and $\underline{g}(r)$ are the minimum and maximum values, respectively, of g^I such that $n_Z(g^I) = r$.

The term $\Pr_N\left[n \in [\underline{g}(r) - s,\, \overline{g}(r) - s)\right]$ in the numerator of the final expression above can be evaluated by replacing the probability density function t with the corresponding cumulative density function in the expression for $p_N(n)$. The denominator can be evaluated by integrating the numerator over all possible values of $s \in (-\infty, \infty)$:

$$P_{R^I}(r) = \int_{-\infty}^{\infty} P_{R^I|S}(r|S = s) \cdot P_S(s)\, ds$$

In practice, the value of each of these integrals can be estimated using a grid approximation. In our experiments, we use the following grids:

- $P_N(n)$: $\left[\frac{-5}{\sqrt{\tau_b^I}}, \frac{5}{\sqrt{\tau_b^I}}\right]$, in increments of $\frac{1}{10\sqrt{\tau_b^I}}$.
- $P_{R^I|S}(r|S = s)$: $\left[\frac{-5}{\sqrt{\tau_b^I}}, \frac{5}{\sqrt{\tau_b^I}}\right]$, in increments of $\frac{1}{10\sqrt{\tau_b^I}}$.
- $P_{R^I}(r)$: $[-5, 10]$, in increments of 0.01.

B.2 Additional Results: Predicting Performance from Observed Data

In this section, we provide the heat maps summarizing the regression results from Section 3.7 for first-order and second-order forecast dominance.

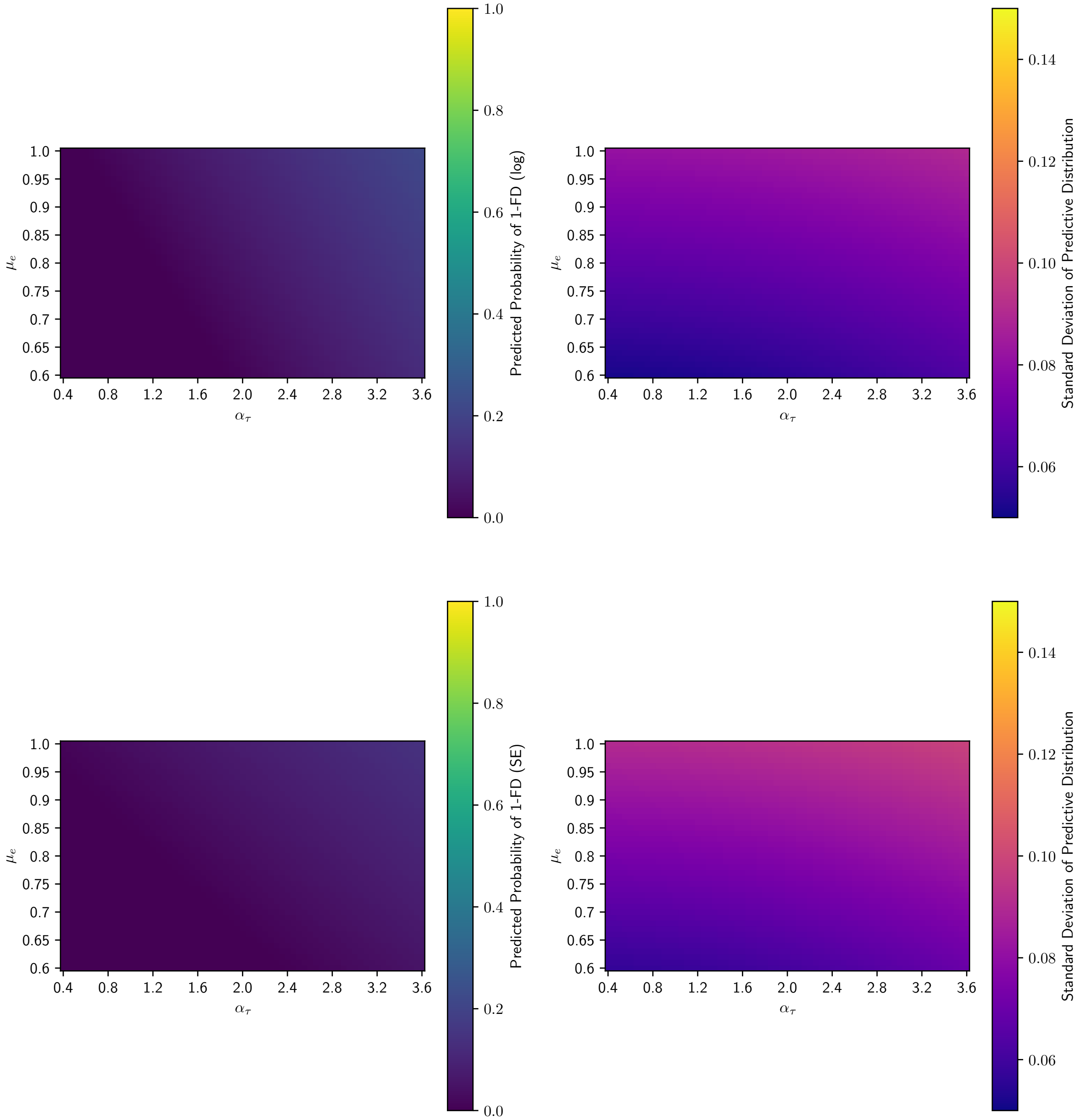


Figure B.1: Predicted probabilities (left) and standard deviations of predictive distributions (right) of peer first-order forecast dominance based on the values of α_τ and μ_e, under log score (top) and squared error (bottom), when there are six peer grades per submission.

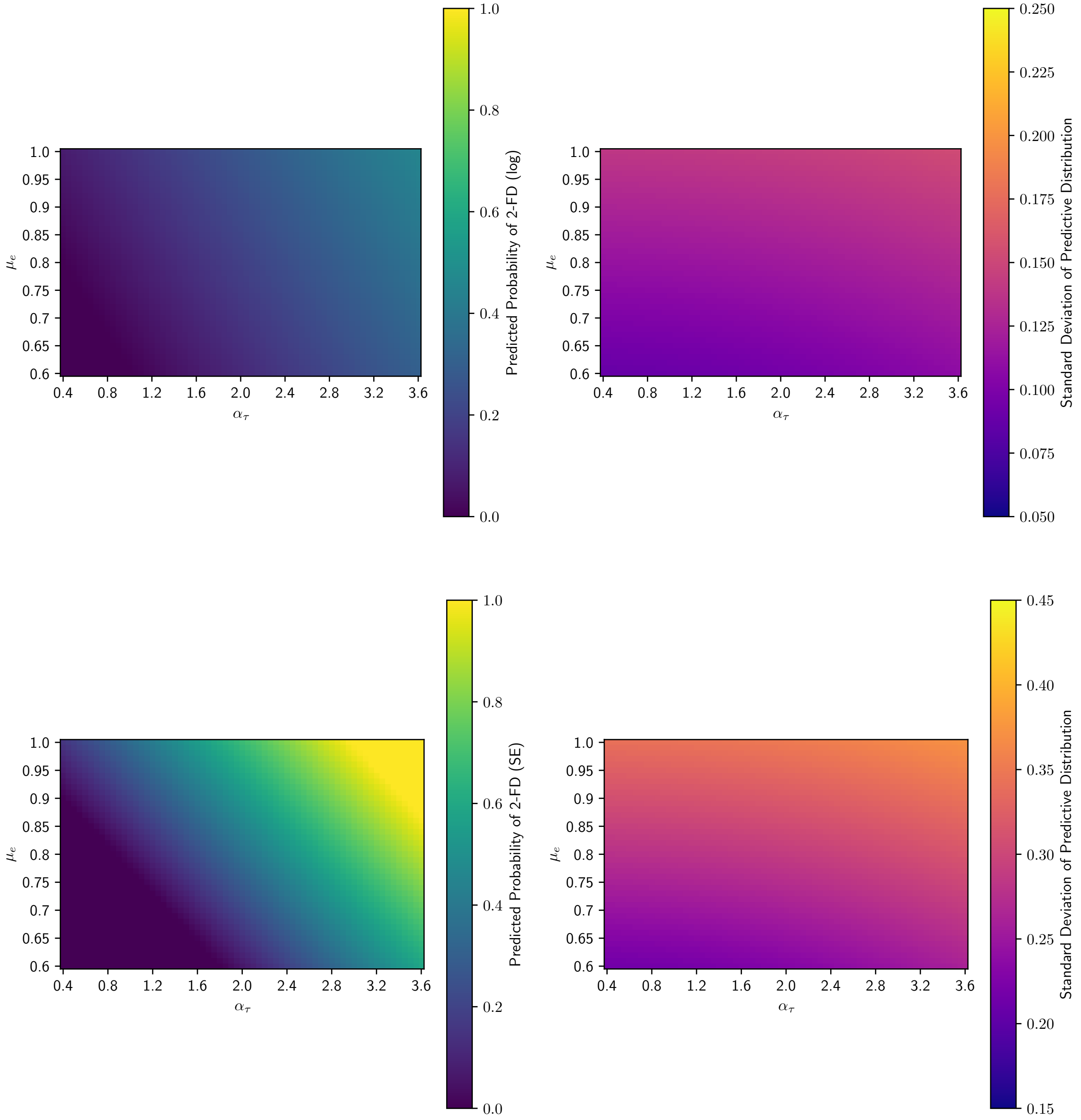


Figure B.2: Predicted probabilities (left) and standard deviations of predictive distributions (right) of peer second-order forecast dominance based on the values of α_τ and μ_e, under log score (top) and squared error (bottom), when there are six peer grades per submission.

APPENDIX C

Supplementary Material for Chapter 4

C.1 Equilibrium Concepts and Sufficient Assumptions

Mechanism	Equilibrium Concept	Sufficient Assumption
DMI	Dominantly Truthful	Strictly Correlated
MSE	None	-
MSE$_P$	None	-
OA	Truthful	Self-Dominating
PTS	Helpful Reporting	Self-Predicting
Φ-Div	ϵ-Strongly Truthful	Stochastically Relevant
Φ-Div$_P$	ϵ-Strongly Truthful	Stochastically Relevant

Table C.1: The equilibrium concepts related to truthful reporting that are induced by each mechanism and the weakest known assumption on the joint prior distribution of signals that is sufficient to guarantee that inducement.

C.1.1 Equilibrium Concepts

Helpful Reporting [22]. An agent with prior belief $\Pr[\cdot]$ is said to follow a *helpful reporting* strategy with respect to a publicly-known distribution R if both:

1. The agent reports truthfully if R is "close enough" to $\Pr[\cdot]$.

2. When the agent is not truthful, their report is never "over-represented" in R. That is, if $R[x] \geq \Pr[x]$, given that their signal is $x' \neq x$, the agent does not report x.

A strategy profile σ is an *ex post subjective equilibrium* if no agent can improve their expected payoff by deviating from σ, given that all other agents' posterior beliefs given their signals (and their observations of the publicly-known distribution of reports R) respect any assumption made (e.g., the self-predicting assumption) that constrains the form of the joint prior distribution of signals.

A *helpful reporting equilibrium* is an *ex post subjective equilibrium* in which each agent adopts a helpful reporting strategy. A *helpful reporting* mechanism admits a helpful reporting equilibrium.

Truthful [22]. A *truthful equilibrium* is an *ex post subjective equilibrium* in which each agent adopts a truthful reporting strategy, i.e reports their observed signal s. A *truthful* mechanism admits a (strict) truthful equilibrium.

Note that while this definition of *truthful* corresponds to its usage above in Table C.1, it takes on a larger range of meanings in the peer prediction literature as a whole.

Strongly Truthful [85]. A *strongly truthful* mechanism admits a Bayes-Nash equilibrium in which agents report truthfully and in which the following properties hold:

1. The expected payment to each agent is maximized over the set of payments to that agent in any Bayes-Nash equilibrium.

2. The expected payment to each agent is *strictly* higher than their payment in any Bayes-Nash equilibrium induced by a strategy profile that is not a *permutation strategy profile.*

A *permutation strategy* is a strategy in which an agent fixes a permutation of the signal space and then, for each task, reports the image of their signal for that task under the permutation. A *permutation strategy profile* is a strategy profile in which each agent adopts a permutation strategy.

ϵ-**Strongly Truthful** [85]. An ϵ-*strongly truthful* mechanism is approximately strongly truthful, in the sense that there exists a strongly truthful payment scheme such that 1) the expected payment to each agent in the truthful Bayes-Nash equilibrium is at most ϵ away from this strongly truthful payment scheme; and 2) the expected payment to each agent in any strategy profile is bounded above by the corresponding payment in this strongly truthful payment scheme. For the Φ-Div and $\Phi\text{-Div}_P$ mechanisms, the optimal value of ϵ for which ϵ-strong truthful-ness is achieved depends on how closely the estimated joint-to-marginal-product ratio $\hat{\text{JP}}$ (see Section 4.3) approximates the true joint-to-marginal-product ratio JP.

Dominantly Truthful [46]. A *dominantly truthful* mechanism admits a *dominant strategy* equilibrium in which agents report truthfully. That is, it admits an equilibrium in which both:

1. For every agent, truthful reporting maximizes their expected payment no matter what strategies other agents play (i.e., truthful reporting is a *dominant strategy.*)

2. For every agent, if they believe that at least one of their informative peers[1] will tell the truth, then reporting truthfully pays *strictly* higher, in expectation, than any non-permutation strategy (see above).

C.1.2 Sufficient Assumptions

Self-Dominating [22]. The joint prior distribution of signals is *self-dominating* if and only if, for each agent, their observed signal s is the most-probable outcome under the posterior distribution conditioned on having observed s (for each possible signal s):

$$\Pr[s|s] > \Pr[x|s],\ \forall x \neq s.$$

The posterior distribution mentioned in the latter expression of the definition (after the "iff") is a distribution for the signal of a peer whose signal is independent (conditioned on the ground truth) from the agent's own signal

Self-Predicting [22]. The joint prior distribution of signals is *self-predicting* if and only if, for each agent, the relative increase in probability for their observed signal s from the agent's prior distribution to their posterior distribution is greater than for any other possible outcome:

$$\frac{\Pr[s|s]}{\Pr[s]} > \frac{\Pr[x|s]}{\Pr[x]},\ \forall x \neq s.$$

As above, the prior and posterior distributions in the latter expression of this definition (after the "iff") are distributions for the signal of a peer whose signal is independent (conditioned on the ground truth) from the agent's own signal.

Strictly Correlated [46; 85]. A pair of agents (a_1, a_2) have *strictly correlated* signals (represented by random variables S_1 and S_2, respectively) if the determinant of the agents' joint probability distribution of signals (written as a matrix $\mathbf{P}$ with entries $\mathbf{P}_{ij} = \Pr[S_1 = s_i, S_2 = s_j]$ for each pair of possible signals (s_i, s_j)) is non-zero. The *strictly correlated* assumption for the DMI mechanism is that each agent has at least one *informative peer*—a peer with whom the agent forms a pair with *strictly correlated* signals.

Stochastically Relevant [85]. The joint prior distribution of signals is *stochastically relevant* if each agent's posterior distribution given their own signal s is unique for each possible signal. That is,

$$\Pr[S|s] \neq \Pr[S|s'] \text{ for each pair } s, s' \text{ such that } s' \neq s,$$

[1] Peers with whom the agent forms a pair with *strictly correlated* signals.

where $\Pr[S|\cdot]$ denotes the entire distribution over the possible signals of a peer whose signal is independent (conditioned on the ground truth) from the agent's own signal. This is in contrast to the expression $\Pr[s|\cdot]$, used above, which denotes the specific probability $\Pr[S = s|\cdot]$ for a possible signal s under that distribution.

C.2 Implementation Details

C.2.1 Implementing Peer Prediction Mechanisms

Note that in mechanisms that involve pairing an agent with another agent in order to compute their scores on a grading task (i.e., generating a report for one submission), we take the expectation over all of the possible pairings to reduce the variance of the scores.

C.2.1.1 Non-Parametric Mechanisms

Output Agreement (OA) Mechanism. The implementation is trivial.

Peer Truth Serum (PTS) Mechanism. To score a grading task for a given submission, a pair of agents that completed that task is selected and their reports are compared. If their reports are equal, then they are awarded $\frac{1}{R[\text{report}]}$, where $R[\text{report}]$ is the probability of the given report under the distribution R.

R is repeatedly updated over the course of a simulated semester via a histogram H of report values. After the payments for an assignment are computed by the mechanism, the report values submitted for that assignment are added to H. Then, R is recomputed by normalizing H with Laplace (add-one) smoothing. In particular, this means that R is initialized to the uniform distribution.

Φ-Divergence Pairing (Φ-Div) Mechanism. The mechanism randomly divides the tasks in half and uses each half to compute $\hat{\text{JP}}$, an estimate of JP, the joint-to-marginal-product ratio of the reports[2] for the other half by counting the frequency of pairs of report values given by pairs of agents grading the same submission (to estimate the joint distribution of reports) and counting the overall frequency of report values (to estimate the marginal distribution of reports).

Using these estimates, scores are computed for each grading task, as follows: For each task b, referred to as the *bonus task*, agents are paired with another agent who completed

[2] The value of the joint distribution of reports evaluated at the given pair of reports divided by the value of the product of the marginal distributions of reports evaluated at the given pair of report.

b. Then, a pair of *penalty tasks* p and q, distinct from each other and from the bonus task, are randomly chosen (one for each agent). The pair of agents is awarded the quantity $\partial\Phi(\hat{\text{JP}}(x_b, y_b)) - \Phi^*(\partial\Phi(\hat{\text{JP}}(x_p, y_q)))$, where Φ^* denotes the *convex conjugate* of Φ, x_i and y_j denote the first agent's reports on task i and the second agent's report on task j, respectively, and $\hat{\text{JP}}$ is the empirical estimate of JP computed using the other half of the tasks.

Recall that for mechanisms that compute payments in pairs, we assign payments according to the average payment over all possible pairings. When working with the real peer grading data, it is occasionally the case that a pair of agents does not have a valid pair of penalty tasks to use. That is, there are some pairs of agents who graded only the same pair of submissions for a given assignment, so distinct penalty tasks cannot be chosen. In those cases, we simply skip over that pairing when computing the average payments over all possible pairings. If that is the only possible pairing (i.e., only those two agents graded that submission), then we ignore those peer grades altogether when computing payments according to this mechanism.

Determinant-based Mutual Information (DMI) Mechanism. The DMI mechanism was developed by Kong [46]. It pays each agent according to a sum of unbiased estimates of the square of the Determinant-based Mutual Information between a random variable drawn from the distribution of their reports and random variables drawn from the distributions of the reports of each other agent who completed the same tasks.

The estimate of the square of the Determinant-based Mutual Information between two random variables depends on the product of the determinants of two matrices, each encoding the frequency of pairs of reports on one part of a bifurcation of the tasks being considered for scoring. As a result, it is easy to see that in our ABM, where there are 11 possible report values and agents complete only 4 grading tasks, the DMI mechanism will always pay every agent 0. Further, the DMI mechanism benefits from having the sets of tasks that agents complete overlap as much as possible. This stands in contrast to the other mechanisms we consider, where the number of tasks mutually completed by a pair of agents—as long as it is at least one—is not very significant. To make the mechanism as functional as possible while still remaining faithful to the original description of the mechanism, we make the following adjustments in our implementation:

1. We project the report space down to a space of 2 possible reports, so that each report is either 0—indicating that a submission has below average quality (< 7)—or 1—indicating that its quality is at least average (≥ 7).

2. We partition the agents into clusters of 4 agents so that each cluster grades the same 4 submissions (namely, the submissions from another cluster).

Note that these modifications have a significant impact on the effect of strategic behavior

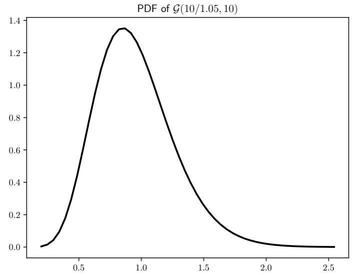


Figure C.1: PDF of $\mathcal{G}\,(10/1.05, 10)$ on values for which its support is non-negligible.

in our experiments. Since the report space is simplified to be binary, oftentimes, a strategy applied to a signal will produce a report equal to the signal (in the binary report space), so in many cases, there is no difference between strategic and truthful reporting. Also, these modifications are not possible in experiments with the real data, so the DMI mechanism is excluded completely from those experiments.

C.2.1.2 Parametric Mechanisms

Recall model $\mathbf{PG}_1$ from Piech et al. [74], which, with the appropriate parameters, provides a reasonable continuous approximation to our mostly discrete model from Section 4.2.1 in which reliability is a proxy for effort:

$$\begin{aligned}
\text{(True Score)} \quad & g^*_{i,j} \sim \mathcal{N}\,(7, 2.1) \text{ for each submission } s_{i,j}, \\
\text{(Reliability)} \quad & \tau_i \quad \sim \mathcal{G}\,(10/1.05, 10) \text{ for each agent } i, \\
\text{(Bias)} \quad & b_i \quad \sim \mathcal{N}\,(0, 1) \text{ for each biased agent } i \text{ (and 0 for each unbiased agent)}, \\
\text{(Observed Score)} \quad & z^k_{i,j} \sim \mathcal{N}\left(g^*_{i,j} + b_k, \tau_k^{-1}\right) \text{ for each submission } s_{i,j} \text{ and assigned grader } k,
\end{aligned}$$

where $\mathcal{G}$ is a Gamma distribution. Note that the hyperparameters $\alpha_0 = 10/1.05$ and $\beta_0 = 10$ for the Gamma distribution used to model reliability were chosen by inspection subject to having the correct expected value for a continuous effort agent (and for a uniformly random agent chosen from a population of binary effort agents with an equal number of active and passive graders). The PDF of the Gamma function with those hyperparameters is plotted in Figure C.1.

Our procedure for estimating the parameters of model $\mathbf{PG}_1$, inspired by Chakraborty et al. [12], is as follows:

Initialize: Set the bias and reliability of each agent and the score for each submission equal to their expectation (0, 1/1.05, and 10 respectively).

Update: For each submission $s_{i,j}$, update the true score estimate $\hat{g}_{i,j}$ as in equation (1) from Chakraborty et al. [12]:

$$\hat{g}_{i,j} = \frac{7 \cdot \sqrt{(2.1)^{-1}} + \sum_k \sqrt{\hat{\tau}_k}(r_{i,j}^k - \hat{b}_k)}{\sqrt{(2.1)^{-1}} + \sum_k \sqrt{\hat{\tau}_k}},$$

where the k in both sums varies over the graders of submission $s_{i,j}$, $\hat{\tau}_k$ and $\hat{b}_k$ are the estimated reliability and estimated bias, respectively, of grader k, and $r_{i,j}^k$ is grader k's report for submission $s_{i,j}$.

For model settings with unbiased agents, we skip the step of estimating the bias and fix all the biases as 0. Otherwise, for each agent k, update the bias estimate $\hat{b}_k$ as the mean of the posterior distribution of a Bayesian update from the conjugate prior $\mathcal{N}(0,1)$ given the agent's reports and the estimated true scores:

$$\hat{b}_k = \frac{\hat{\tau}_k \cdot \sum_{s_{i,j}}(r_{i,j}^k - \hat{g}_{i,j})}{1 + n \cdot \hat{\tau}_k},$$

where the sum in the numerator varies over the submissions graded by agent k for the fixed assignment j and n is the number terms in that sum.

For each agent k, update the reliability estimate $\hat{\tau}_k$ as the mean of the posterior distribution of a Bayesian update from the conjugate prior $\mathcal{G}(10/1.05, 10)$ given the agent's reports, the agent's estimated bias, and the estimated true scores:

$$\hat{\tau}_k = \frac{\frac{10}{1.05} + \frac{n}{2}}{10 + \frac{1}{2} \cdot \sum_{s_{i,j}}(r_{i,j}^k - (\hat{g}_{i,j} + \hat{b}_k))^2},$$

where the sum in the denominator varies over the submissions graded by agent k for the fixed assignment j and n is the number of terms in that sum.

Terminate: When the ℓ_2 norm of the difference between the vector of estimated true scores after the previous round and the vector of estimated true scores after the current round is less than or equal to 0.0001, terminate.

In our implementation, we also imposed a maximum of 1000 iterations of the update procedure, after which the procedure would terminate even if the ℓ_2 norm condition were not met, but after adding the priors to the bias and reliability estimates, this extra termination condition was never applicable.

Parametric MSE (MSE$_P$) Mechanism. Given the estimation procedure outlined above and the descriptions from Section 4.3.3, the implementation of the this mechanism is trivial.

Parametric Φ-Divergence Pairing (Φ-Div$_P$) Mechanism. Given the estimation procedure outlined above and the descriptions from Section 4.3.3 (and the derivation from Appendix C.3), the implementation of the parametric mechanisms is straightforward. However, we make some adjustments to improve its performance. In simulations, we find that this mechanism performs best when the reliability estimates are heavily regularized and the simplest way to achieve close-to-optimal (if not optimal) performance is to set all the reliability estimates equal to $\frac{1}{0.7}$, the approximate reliability of an "active grader" in our setting with binary effort (see Section 4.5). This is akin to replacing model **PG**$_1$ with model **PG**$_1$**-bias** [74]. Consequently, only the bias estimates given by the estimation procedure are used (and only in settings with biased agents).

As with the non-parametric version of this mechanism, in the real data, it is occasionally the case that a pair of agents does not have a valid pair of penalty tasks to use. We handle those cases in the same way for the parametric and non-parametric versions of the mechanism (see Section C.2.1.1).

C.3 Computing the Joint-to-Marginal-Product Ratio under model PG$_1$

Recall that in model **PG**$_1$, the true score for any submission is $g^* \sim \mathcal{N}\left(\mu, \sigma^2\right)$, and each grader i has an underlying bias score b_i and reliability score τ_i. Then, i's observed grade is $s_i \sim \mathcal{N}\left(g_i^* + b_i, \tau_i^{-1}\right)$.[3]

Now, consider two agents i and j receiving signals $x = s_i$ and $y = s_j$. For the purposes of the Φ-Div$_P$ mechanism, we need to compute the joint-to-marginal-product ratio of x and y:

$$\mathrm{JP}(x, y) = \frac{P_{X,Y}(x, y)}{P_X(x)P_Y(y)}.$$

[3] Here we use g_i^* instead of g^*, because the submissions graded by i and j are sometimes different, e.g., when considering the penalty tasks.

Considering random variables X and Y[4] such that

$$X \sim \mathcal{N}\left(\mu + b_i, \sigma^2 + \tau_i^{-1}\right) \text{ and } Y \sim \mathcal{N}\left(\mu + b_j, \sigma^2 + \tau_j^{-1}\right)$$

and setting $\mu_i = \mu + b_i$ and $\mu_j = \mu + b_j$, we have

$$P_X(x) = \frac{\exp\left(-\frac{1}{2}\left(\frac{x-\mu_i}{\sqrt{\sigma^2+\tau_i^{-1}}}\right)^2\right)}{\sqrt{\sigma^2 + \tau_i^{-1}}\sqrt{2\pi}} \text{ and } P_Y(y) = \frac{\exp\left(-\frac{1}{2}\left(\frac{y-\mu_j}{\sqrt{\sigma^2+\tau_j^{-1}}}\right)^2\right)}{\sqrt{\sigma^2 + \tau_j^{-1}}\sqrt{2\pi}},$$

and so we can write the product of the marginal distributions of X and Y as

$$P_X(x)P_Y(y) = \frac{\exp\left(-\frac{1}{2}\left[x-\mu_i, y-\mu_j\right] \cdot \begin{bmatrix} \frac{1}{\sigma^2+\tau_i^{-1}} & 0 \\ 0 & \frac{1}{\sigma^2+\tau_j^{-1}} \end{bmatrix} \cdot \begin{bmatrix} x-\mu_i \\ y-\mu_j \end{bmatrix}\right)}{2\pi\sqrt{\sigma^2+\tau_i^{-1}}\sqrt{\sigma^2+\tau_j^{-1}}}.$$

Then, we can compute that the joint distribution of X and Y is the following multivariate Gaussian:

$$\begin{bmatrix} X \\ Y \end{bmatrix} \sim \mathcal{N}\left(\begin{bmatrix} \mu_i \\ \mu_j \end{bmatrix}, \begin{bmatrix} \sigma^2+\tau_i^{-1} & \sigma^2 \\ \sigma^2 & \sigma^2+\tau_j^{-1} \end{bmatrix}\right).$$

Let

$$\Sigma = \begin{bmatrix} \sigma^2+\tau_i^{-1} & \sigma^2 \\ \sigma^2 & \sigma^2+\tau_j^{-1} \end{bmatrix},$$

and as a result

$$|\Sigma| = (\sigma^2+\tau_i^{-1})(\sigma^2+\tau_j^{-1}) - (\sigma^2)^2 = \frac{\sigma^2\tau_i + \sigma^2\tau_j + 1}{\tau_i\tau_j}, \quad \Sigma^{-1} = \frac{1}{|\Sigma|}\begin{bmatrix} \sigma^2+\tau_j^{-1} & -\sigma^2 \\ -\sigma^2 & \sigma^2+\tau_i^{-1} \end{bmatrix}.$$

We can write the joint distribution of X and Y as

$$P_{X,Y}(x,y) = \frac{\exp\left(-\frac{1}{2}\left[x-\mu_i, y-\mu_j\right] \cdot \Sigma^{-1} \cdot \begin{bmatrix} x-\mu_i \\ y-\mu_j \end{bmatrix}\right)}{\sqrt{(2\pi)^2|\Sigma|}}.$$

[4] The distributions of X and Y follow from writing X and Y as the sum of two normally-distributed random variables $G \sim \mathcal{N}\left(\mu, \sigma^2\right)$ and $B_X \sim \mathcal{N}\left(b_i, \tau_i^{-1}\right)$ for X or $B_Y \sim \mathcal{N}\left(b_j, \tau_j^{-1}\right)$ for Y.

Then, we can write out the joint-to-marginal-product ratio explicitly and simplify:

$$\mathrm{JP}(x,y) = \frac{P_{X,Y}(x,y)}{P_X(x)P_Y(y)} = \frac{\dfrac{\exp\left(-\frac{1}{2}[x-\mu_i,y-\mu_j]\cdot\Sigma^{-1}\cdot\begin{bmatrix}x-\mu_i\\ y-\mu_j\end{bmatrix}\right)}{\sqrt{(2\pi)^2|\Sigma|}}}{\dfrac{\exp\left(-\frac{1}{2}[x-\mu_i,y-\mu_j]\cdot\begin{bmatrix}\frac{1}{\sigma^2+\tau_i^{-1}} & 0\\ 0 & \frac{1}{\sigma^2+\tau_j^{-1}}\end{bmatrix}\cdot\begin{bmatrix}x-\mu_i\\ y-\mu_j\end{bmatrix}\right)}{2\pi\sqrt{\sigma^2+\tau_i^{-1}}\sqrt{\sigma^2+\tau_j^{-1}}}}$$

$$= \frac{\exp\left(-\frac{1}{2}\left[x-\mu_i,y-\mu_j\right]\Sigma^{-1}\begin{bmatrix}x-\mu_i\\ y-\mu_j\end{bmatrix}\right)}{2\pi\sqrt{|\Sigma|}}$$

$$\cdot\frac{2\pi\sqrt{\sigma^2+\tau_i^{-1}}\sqrt{\sigma^2+\tau_j^{-1}}}{\exp\left(-\frac{1}{2}\left[x-\mu_i,y-\mu_j\right]\begin{bmatrix}\frac{1}{\sigma^2+\tau_i^{-1}} & 0\\ 0 & \frac{1}{\sigma^2+\tau_j^{-1}}\end{bmatrix}\begin{bmatrix}x-\mu_i\\ y-\mu_j\end{bmatrix}\right)}$$

$$= \sqrt{\frac{(\sigma^2+\tau_i^{-1})(\sigma^2+\tau_j^{-1})}{|\Sigma|}}$$

$$\cdot\exp\left(-\frac{1}{2}\left[x-\mu_i,y-\mu_j\right]\left(\Sigma^{-1}-\begin{bmatrix}\frac{1}{\sigma^2+\tau_i^{-1}} & 0\\ 0 & \frac{1}{\sigma^2+\tau_j^{-1}}\end{bmatrix}\right)\begin{bmatrix}x-\mu_i\\ y-\mu_j\end{bmatrix}\right)$$

$$\propto \exp\Bigg(-\frac{1}{2}\frac{\sigma^2}{|\Sigma|(\sigma^2+\tau_i^{-1})(\sigma^2+\tau_j^{-1})}$$

$$\left[x-\mu_i,y-\mu_j\right]\begin{bmatrix}\sigma^2(\sigma^2+\tau_j^{-1}) & -(\sigma^2+\tau_i^{-1})(\sigma^2+\tau_j^{-1})\\ -(\sigma^2+\tau_i^{-1})(\sigma^2+\tau_j^{-1}) & \sigma^2(\sigma^2+\tau_i^{-1})\end{bmatrix}\begin{bmatrix}x-\mu_i\\ y-\mu_j\end{bmatrix}\Bigg).$$

Finally, setting

$$G(x,y) = \left[x-\mu_i,y-\mu_j\right]\begin{bmatrix}\sigma^2(\sigma^2+\tau_j^{-1}) & -(\sigma^2+\tau_i^{-1})(\sigma^2+\tau_j^{-1})\\ -(\sigma^2+\tau_i^{-1})(\sigma^2+\tau_j^{-1}) & \sigma^2(\sigma^2+\tau_i^{-1})\end{bmatrix}\begin{bmatrix}x-\mu_i\\ y-\mu_j\end{bmatrix},$$

and simplifying, we can write

$$\mathrm{JP}(x,y) = \sqrt{\frac{(\sigma^2+\tau_i^{-1})(\sigma^2+\tau_j^{-1})}{(\sigma^2+\tau_i^{-1})(\sigma^2+\tau_j^{-1})-\sigma^4}}$$

$$\cdot\exp\left(-\frac{1}{2}\frac{\sigma^2\tau_i\tau_j}{(\sigma^2\tau_i+\sigma^2\tau_j+1)(\sigma^2+\tau_i^{-1})(\sigma^2+\tau_j^{-1})}G(x,y)\right).$$

For our experiments with ABM, this gives:

$$\mathrm{JP}(x, y) = \sqrt{\frac{(2.1 + \tau_i^{-1})(2.1 + \tau_j^{-1})}{(2.1 + \tau_i^{-1})(2.1 + \tau_j^{-1}) - 4.41}} \cdot \exp\left(-\frac{1.05\tau_i\tau_j}{(2.1\tau_i + 2.1\tau_j + 1)(2.1 + \tau_i^{-1})(2.1 + \tau_j^{-1})} G(x, y)\right),$$

when the substitutions $\mu = 7$ (implicitly in the definitions of μ_i and μ_j) and $\sigma^2 = 2.1$ are also made in the definition of $G(x, y)$. For each semester in the real data, we just substitute the corresponding estimates of μ and σ^2 given in Table 4.1.

C.4 Considering Novel Mechanisms

In this section, we give a brief description of some novel (to our knowledge) mechanisms that we used in an effort to push the Pareto frontier delineated by the established mechanisms from the peer prediction literature. For these experiments, we adopt the alternative notion of report quality described in Section 4.5.

Although these novel mechanisms do not significantly expand the Pareto frontier in our setting, we expect that many of them may be useful in other settings where data is more readily available. In particular, as with some of the established parametric mechanisms described in Section 4.3.3, the robustness against strategic reporting of certain mechanisms may be compromised by the fact that agents' reports, through the estimation procedure, have an effect on the ground truth estimates that are later used in scoring their reports. In settings with more data, it may be possible to get good ground truth score estimates that are independent of the reports of the particular agent being scored by the mechanism.

Lastly, note that each of our novel mechanisms is parametric. Thus, each mechanism begins by estimating the parameters of model $\mathbf{PG}_1$ according to our estimation procedure.

Coefficient of Determination (R^2) Mechanism. This mechanism pays each agent the *coefficient of determination* (denoted R^2) between the set of their bias-corrected reports (which constitute the "predicted values" in the definition of R^2) and the set of true score estimates on those same tasks (which constitute the "observed data" in the definition of R^2).

The intuition behind this mechanism is that R^2 can be interpreted as the proportion of the variance in the observed data that can be explained from the predicted values. Thus, agents who report accurately ought to do well—their reports explain a high fraction of the variance in the ground truth, because most of the variance in the reports of a reliable grader comes from variation in the ground truth scores (as opposed to coming from noise in the process of generating their signals of the ground truth scores.) Further, conditioned on their signal, any strategy that an agent applies cannot depend on the ground truth, since the signal contains all of an agent's private information about the ground truth. As a result, the coefficient of determination between an agent's signals and the ground truth scores cannot be increased by applying a strategy to the signals to generate non-truthful reports. (Note, however, that this statement is a simplification in our setting, since it ignores the role that reports play in the estimation procedure.)

Correlation (CORR) Mechanism. This mechanism pays agents the (sample) Pearson correlation coefficient, r, between the set of their bias-corrected reports for the submissions that they graded and the set of true score estimates computed for those submissions.

The intuition here is similar to that of the previous mechanism. Accurate reports ought to be more highly correlated with the ground truth. Further, as described above, the correlation between an agent's signals and the ground truth scores cannot be increased by applying a strategy to the signals to generate non-truthful reports because—given their signal—any strategy that an agent applies cannot depend on additional private information about the ground truth score. (As above, though, this statement is a simplification in our setting, since it ignores the role that reports play in the estimation procedure.)

Leave-One-Out (LOO) Mechanism. The idea of this mechanism is to capture the value of an agent's report by determining how much the quality of the true score estimate deteriorates (for each submission that they grade) when they are omitted from the population of agents. The mechanism estimates the parameters of model $\mathbf{PG}_1$ using our estimation procedure once with the entire population of n agents, then n more times with a population of $n-1$ agents, leaving out a different agent each time. Note that, as a result, this mechanism takes significantly longer to run than the other mechanisms.

In implementing this mechanisms, especially in settings with more data, reliable ground truth estimates for each submission should not be difficult to compute, even when leaving one agent at a time out of the estimation procedure. In our setting, however, we found that reliable ground truth estimates would not be sufficient to make this mechanism worthwhile to run (particularly in light of the significantly increased computational resources it requires compared to the other mechanisms).

To gauge the potential of this mechanism, we gave it access to the underlying ground truth scores. Each agent was paid according to the reduction in squared error that resulted from including them in the agent population. That is, for each submission that they graded, each agent was paid the squared error of the true score estimate with them left out (with respect to the ground truth) minus the squared error of the true score estimate with them included (also with respect to the ground truth).

Even with access to the ground truth scores, this mechanism did not demonstrate significant measurement integrity compared to the best-performing mechanisms. Moreover—because of the access to the true scores—it would not be fair to compare it to the other mechanisms (especially with respect to robustness against strategic reporting). Consequently, this mechanism is omitted from Figure C.2.

Maximum Correlation Coefficient (MCC) Mechanism. This mechanism, along with the intuition behind it, was suggested by Fang-Yi Yu. Given a pair of random variables $(X, Y) \sim P_{X,Y}$, the *maximum correlation coefficient* between X, Y is

$$\rho^*(X, Y) = \max_{f,g} \left\{ \mathrm{E}[f(X)g(Y)] : \mathrm{E}[f(X)] = \mathrm{E}[g(Y)] = 0, \mathrm{E}[f(X)^2] = \mathrm{E}[g(Y)^2] = 1 \right\}.$$

For a bivariate normal distribution, it is known that $\rho^*(X, Y) = |\rho(X, Y)|$, where ρ is the typical correlation coefficient [119]. Since a pair of signals in model $\mathbf{PG}_1$ follows a bivariate normal distribution, we can apply that principle to a mechanism and pay each pair of agents that completed the same task according to the maximum correlation coefficient between their reports.

According to model $\mathbf{PG}_1$, for a pair of agents i and j who receive signals given by the random variables X and Y (respectively):

$$\rho^*(X, Y) = \left| \frac{2.1}{\sqrt{2.1 + \tau_i^{-1}}\sqrt{2.1 + \tau_j^{-1}}} \right| \tag{C.1}$$

In general, the intuition for the incentive compatibility of this mechanism follows from the revelation principle. The mechanism, in maximizing the correlation coefficient, applies the "optimal strategy" for the agents once it receives their reports. Therefore, agents need not perform their own strategic manipulations.

In our setting specifically, there is an even more straightforward argument: Each agent's reports do not play a direct role in determining their payments. That is, the reports do not appear in eq. (C.1), above. Note, however, that the reports do have an indirect effect, since they are used for computing estimates of τ_i and τ_j according to our estimation procedure.

Parametric Φ-Divergence* Pairing (Φ-Div$_P^*$) Mechanism. This mechanism, as implied by the name, is quite similar to the Φ-Div$_P$ mechanism. The only difference is, in the Φ-Div$_P^*$ mechanism, agents are "paired" with the ground truth instead of with other agents. For the bonus task, the second report is an estimated true score that is computed (using the formula and parameters in the estimation procedure) with only the other three agents' reports on that task. Thus, the agent who is being scored by the mechanism is not taken into consideration when computing the estimated true score with which they are "paired." For the penalty task, the second report is an estimated true score (given by the estimation procedure, i.e., computed using all 4 agents that submitted a report) for a task that was not completed by the agent who is being scored.

Parametric Adjusted Mean Squared Error (AMSE$_P$) Mechanism. To discourage agents from hedging, this mechanism introduces a penalty for being too close to the mean of the prior distribution (i.e., 7) into the scores computed in the established MSE$_P$ mechanism. For each submission $s_{i,j}$, each agent k who graded that submission is assigned the following reward:

$$-((r^k_{i,j} - \hat{b}_k) - \hat{g}_{i,j})^2 + 0.1 \cdot ((r^k_{i,j} - \hat{b}_k) - 7)^2,$$

where $\hat{b}_k$ is the estimated bias of grader k, and $r^k_{i,j}$ is grader k's report for submission $s_{i,j}$, and $\hat{g}_{i,j}$ is the estimated true score for submission $s_{i,j}$.

C.4.1 Revisiting the Pareto Frontier

Recall that in these experiments, we consider the alternative notion of report quality described in Section 4.5 and we use coarse ordinal measurement integrity (with respect to AUCC) rather than fine ordinal measurement integrity (with respect to τ_B), so the Pareto frontier in Figure C.2 is slightly different than that of Figure 4.1.

C.5 Additional Experimental Results

In this section, we plot the results of additional experiments, including results for the experiments from Sections 4.4 to 4.6 with the DMI mechanism.

C.5.1 Measurement Integrity

C.5.1.1 Measurement Integrity Under an Alternative Conception of Report Quality

In Figures C.3 to C.5, we show the results of our experiments from Section 4.5 including the results for the DMI mechanism. Recall that according to the methodology described in Section 4.5.1.1, agents are assigned to submissions according to a random 4-regular graph with a vertex for each agent. The DMI mechanism is not functional using this procedure, so

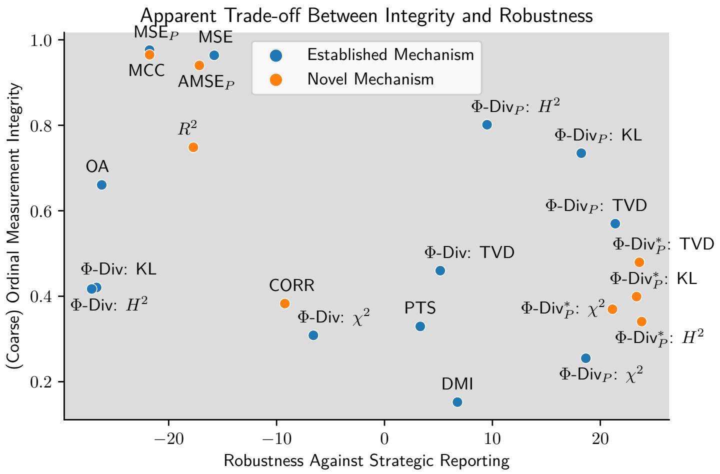


Figure C.2: The novel mechanisms that we consider do not significantly expand the Pareto frontier delineated by the established mechanisms from the peer prediction literature. The most notable change is that some of the Φ-Div$^*_P$ mechanisms fill in the space in the lower right section of the figure, narrowly supplanting their counterpart Φ-Div$_P$ mechanisms as the mechanisms that are most robust against strategic reporting in our experiments, but at the cost of lower measurement integrity.

for that mechanism the agents are instead randomly partitioned into disjoint 4-cliques, and each agent in a clique grades all 4 submissions from another clique.

The results from these experiments highlight a disconnect between theoretical properties and empirical performance for peer prediction mechanisms: for the former the DMI mechanism is perhaps the most exemplary and the OA mechanism perhaps the least; for the latter, the roles are starkly reversed.

C.5.2 Robustness Against Strategic Reporting

C.5.2.1 Computational Experiments with ABM

See Figure C.6 for the results of the experiment from Section 4.6.2 for the DMI mechanism. Note that for the DMI mechanism, the effect of the strategies is different, because of the mapping down to only 2 report options. This explains why several of the strategies are completely neutral under DMI; they do not affect the value of the report after the mapping.

In our simulated experiments (described in Section 4.6.2.1), during the first reward assignment of each iteration, we also record the AUC resulting from a consideration of each mechanism's rewards as scores with which to classify the agents as either truthful or strategic. This gives a more population-level view of the incentives for reporting truthfully or strategically. In this case, AUC can be interpreted as the probability that a uniformly random

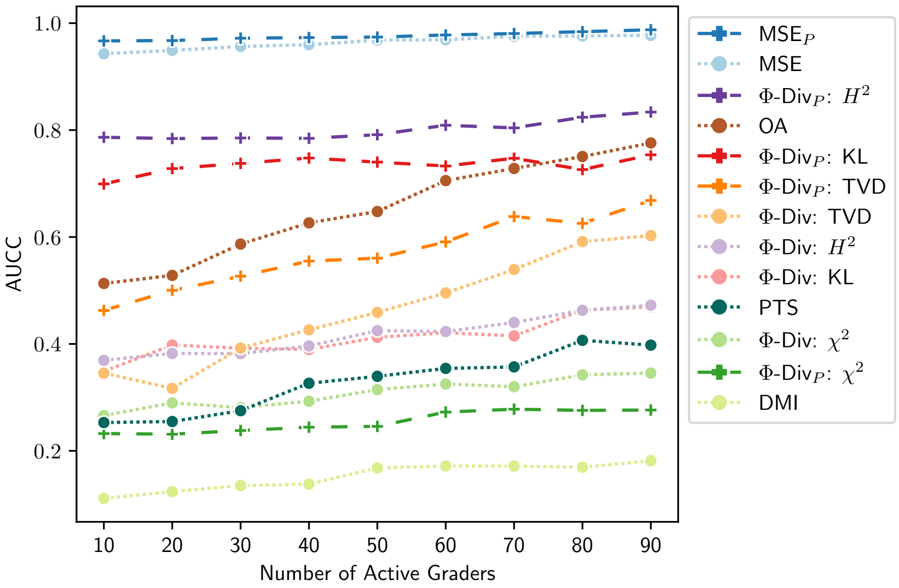


Figure C.3: Binary Effort, Unbiased Agents with DMI. Averages values of AUCC as the number of active (i.e., high-effort) graders varies. The average for each number of active graders is taken over 100 simulated semesters.

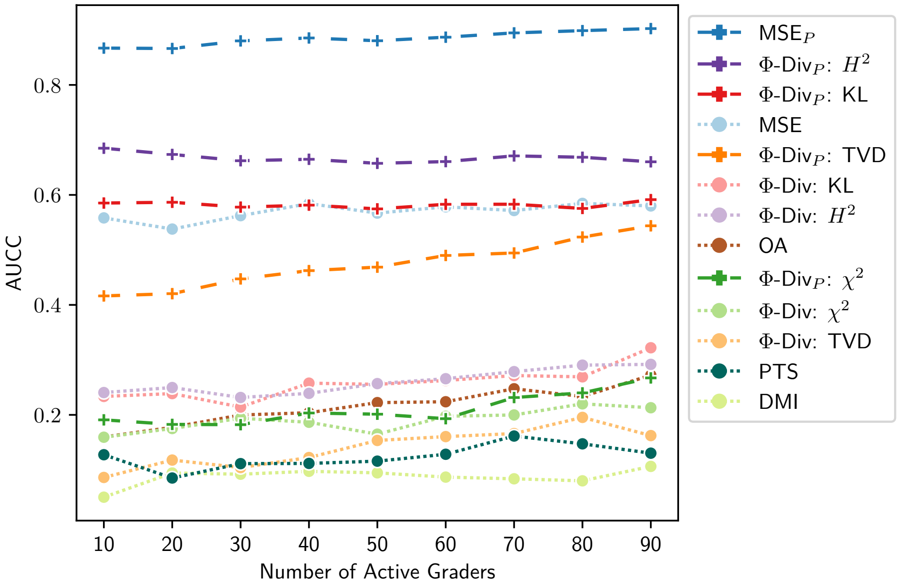


Figure C.4: Binary Effort, Biased Agents with DMI. Average values of AUCC as the number of active (i.e., high-effort) graders varies. The average for each number of active graders is taken over 100 simulated semesters.

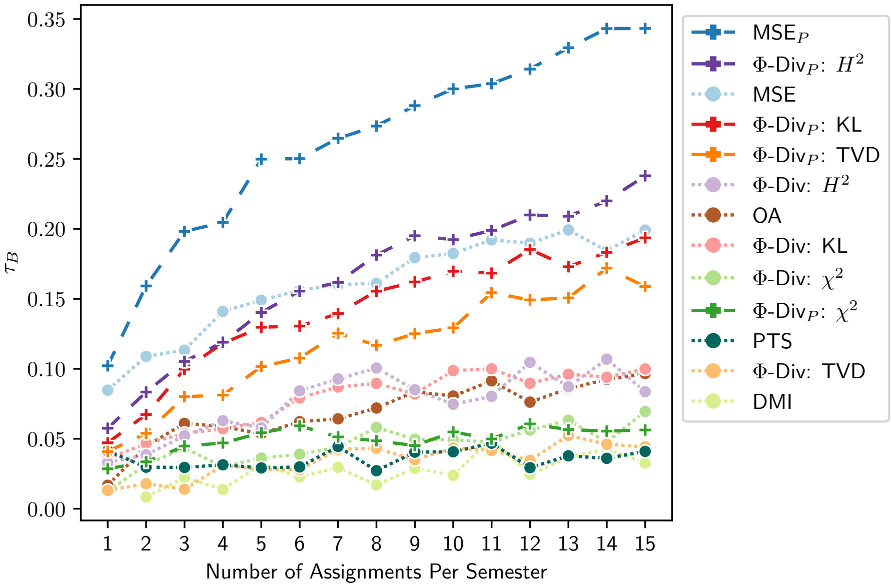


Figure C.5: Continuous Effort, Biased Agents with DMI. Average values of τ_B as the number of assignments per semester varies. The average for each number of assignments per semester is taken over 100 simulated semesters.

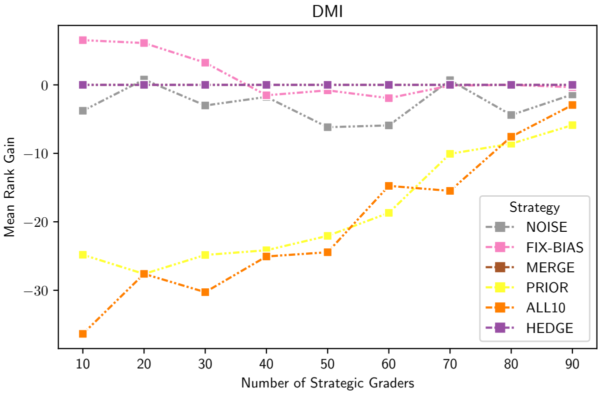


Figure C.6: Quantifying Robustness with ABM. Average rank gain achieved by a single student deviating from truthful to strategic reporting.

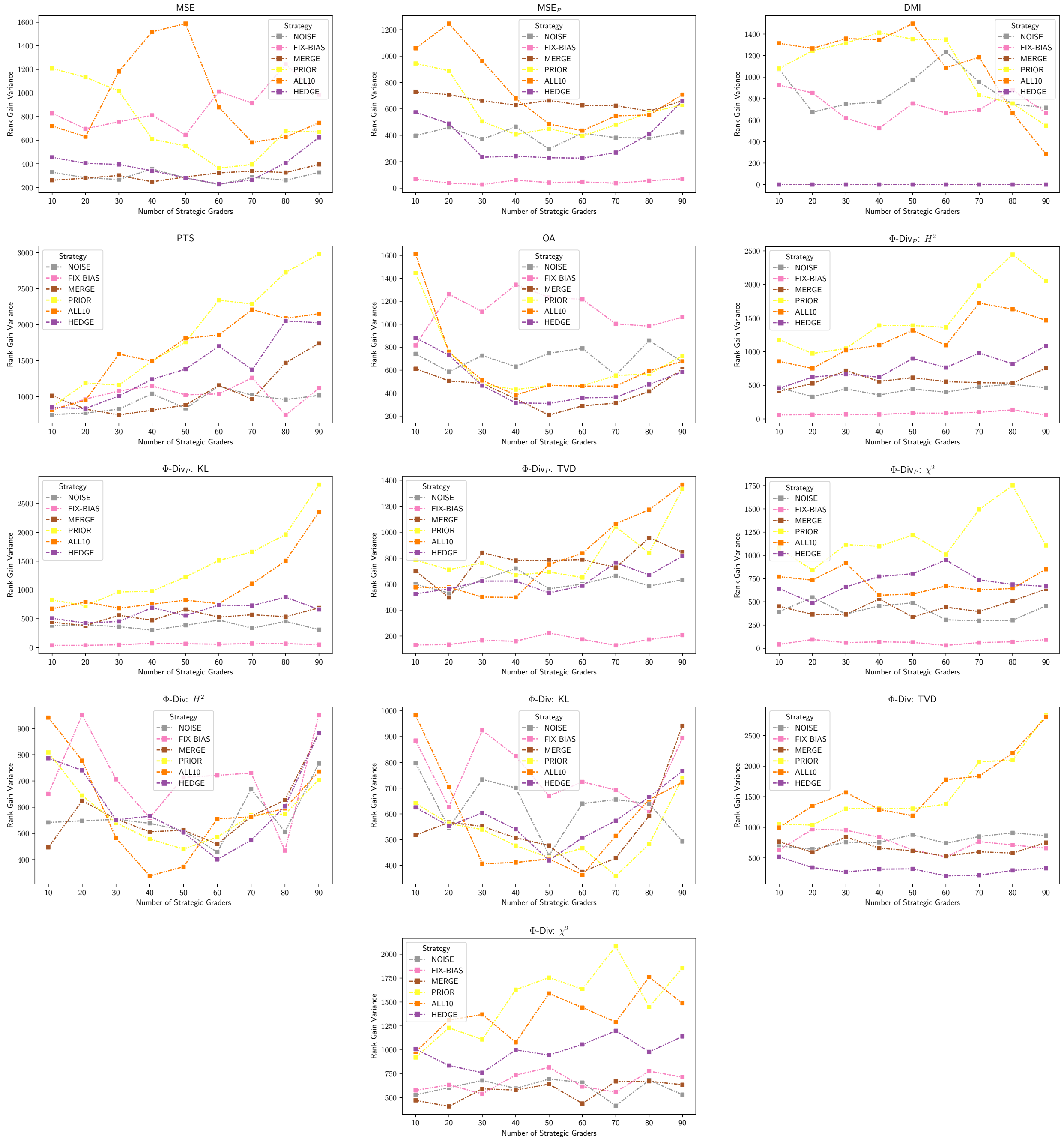


Figure C.7: Quantifying Robustness with ABM. Variance of the rank gain achieved by a single student deviating from truthful to strategic reporting.

truthful agent is ranked higher than a uniformly random strategic agent. The results are shown in Figure C.8.[5] Note that for these plots, we allow bias correction from the parametric mechanisms. The population-level view tells the same story as the results of our individual-level analysis (as depicted in Figure 4.8 for non-parametric mechanisms and Figure 4.10 for parametric mechanisms).

C.5.2.2 Computational Experiments with Real Data

See Figure C.9 for the corresponding results concerning the variance of the gain.

C.5.2.3 Improving Robustness with Bias Correction

See Figure C.10 has the corresponding variances in rank gain for the bias-correcting the Φ-Div$_P$ mechanisms.

C.5.3 Measurement Integrity in the Presence of Strategic Agents

See Figure C.11 for the results of the experiment from Section 4.7 including the results for the DMI mechanism.

C.5.4 Estimating Ground Truth Scores

Here, we provide evidence for the utility of our parameter estimation procedure that is described in Section C.2.1.2.

In the *Continuous Effort, Unbiased Agents* setting with truthfully reporting agents, we simulate the grading of 1000 assignments with 100 submissions each and record the mean squared error of the estimation of the ground truth scores for each of the following two methods:

1. *Consensus Grade.* Estimates the true score of a submission as the mean of the graders' reports.

2. *Parameter Estimation Procedure, No Bias (Procedure-NB).* Estimates the true score of a submission using the parameter estimation procedure from Section C.2.1.2, but without estimating agent biases. All agent biases are assumed to be 0 and the **Update** step in which biases are estimated is skipped in each iteration of the procedure.

We do the same in the *Continuous Effort, Biased Agents* setting for each of the following 3 methods:

[5] For clarity, the plots are shown in steps of size 20 instead of 10.

1. *Consensus Grade.* Estimates the true score of a submission as the mean of the graders' reports.

2. *Parameter Estimation Procedure, No Bias (Procedure-NB).* Estimates the true score of a submission using the parameter estimation procedure from Section C.2.1.2, but without estimating agent biases. All agent biases are assumed to be 0 and the **Update** step in which biases are estimated is skipped in each iteration of the procedure.

3. *Parameter Estimation Procedure (Procedure).* Estimates the true score of a submission using the parameter estimation procedure from Section C.2.1.2 (including estimating agent biases).

The results of both experiments are plotted in Figure C.12. We find that our estimation procedure improves over the consensus grade in both cases. We also, once again, see the value of modeling the bias of agents in Appendix C.5.4.

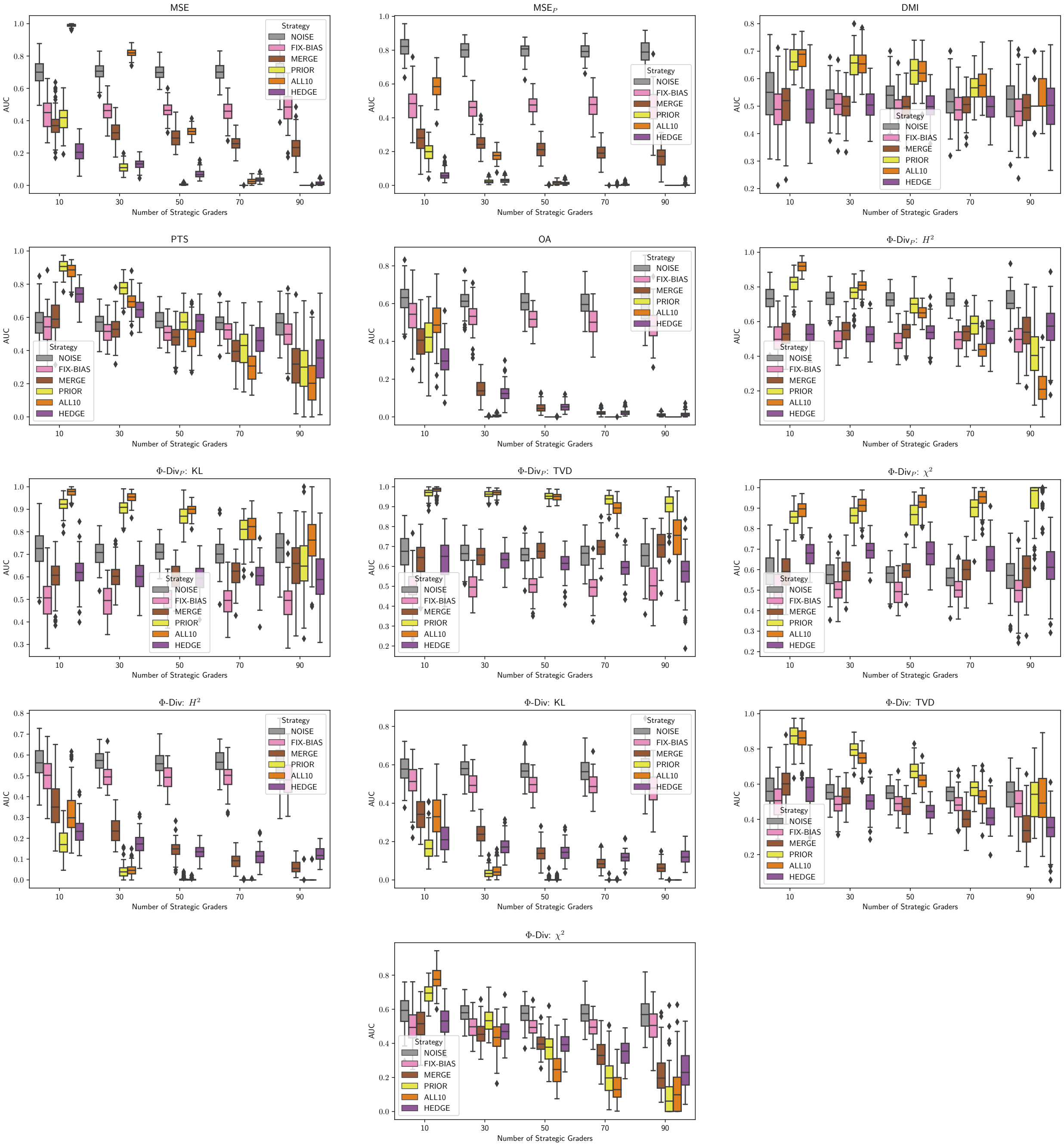


Figure C.8: Quantifying Robustness with ABM. Comparing the rewards between strategic and truthful agents using AUC.

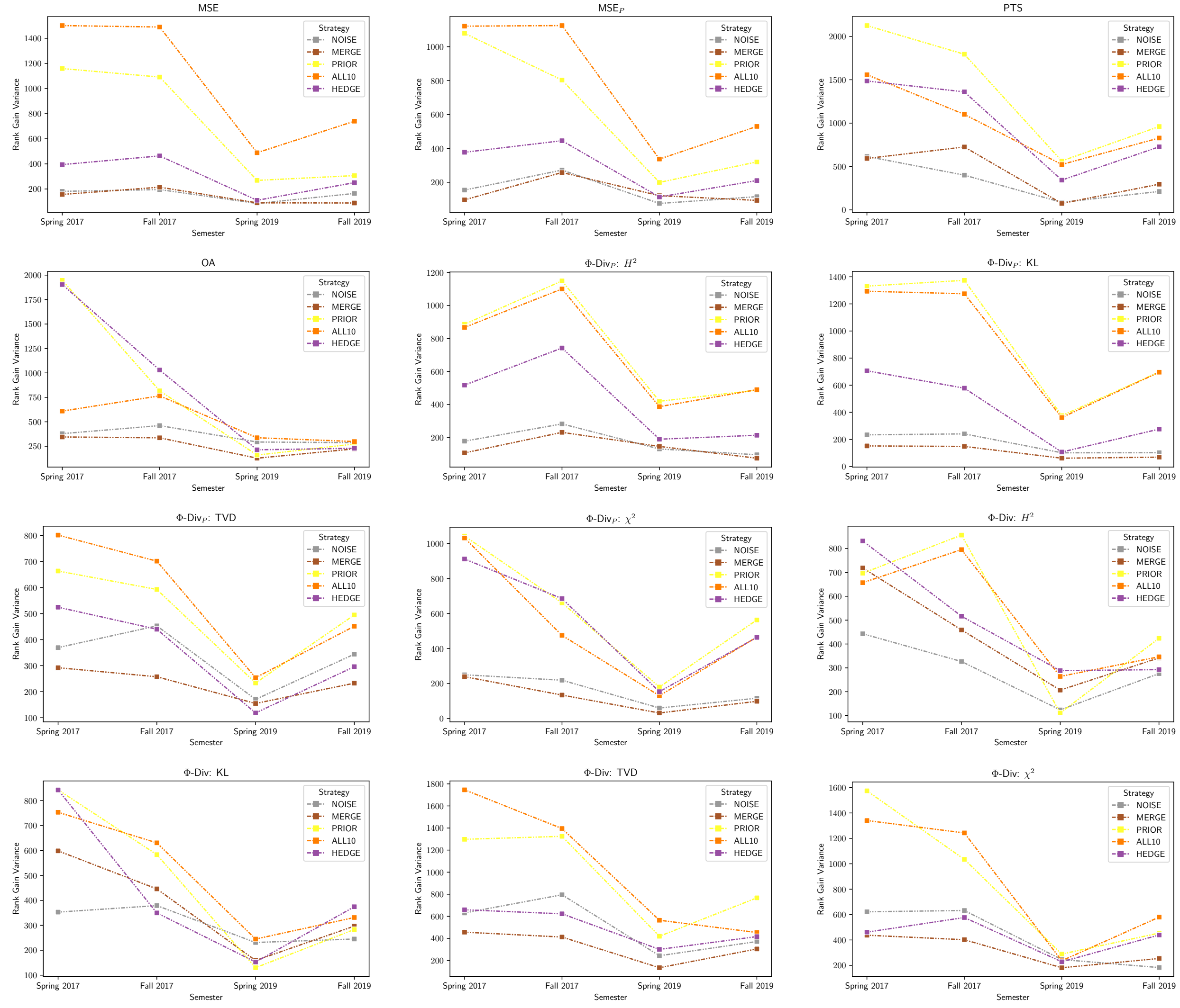
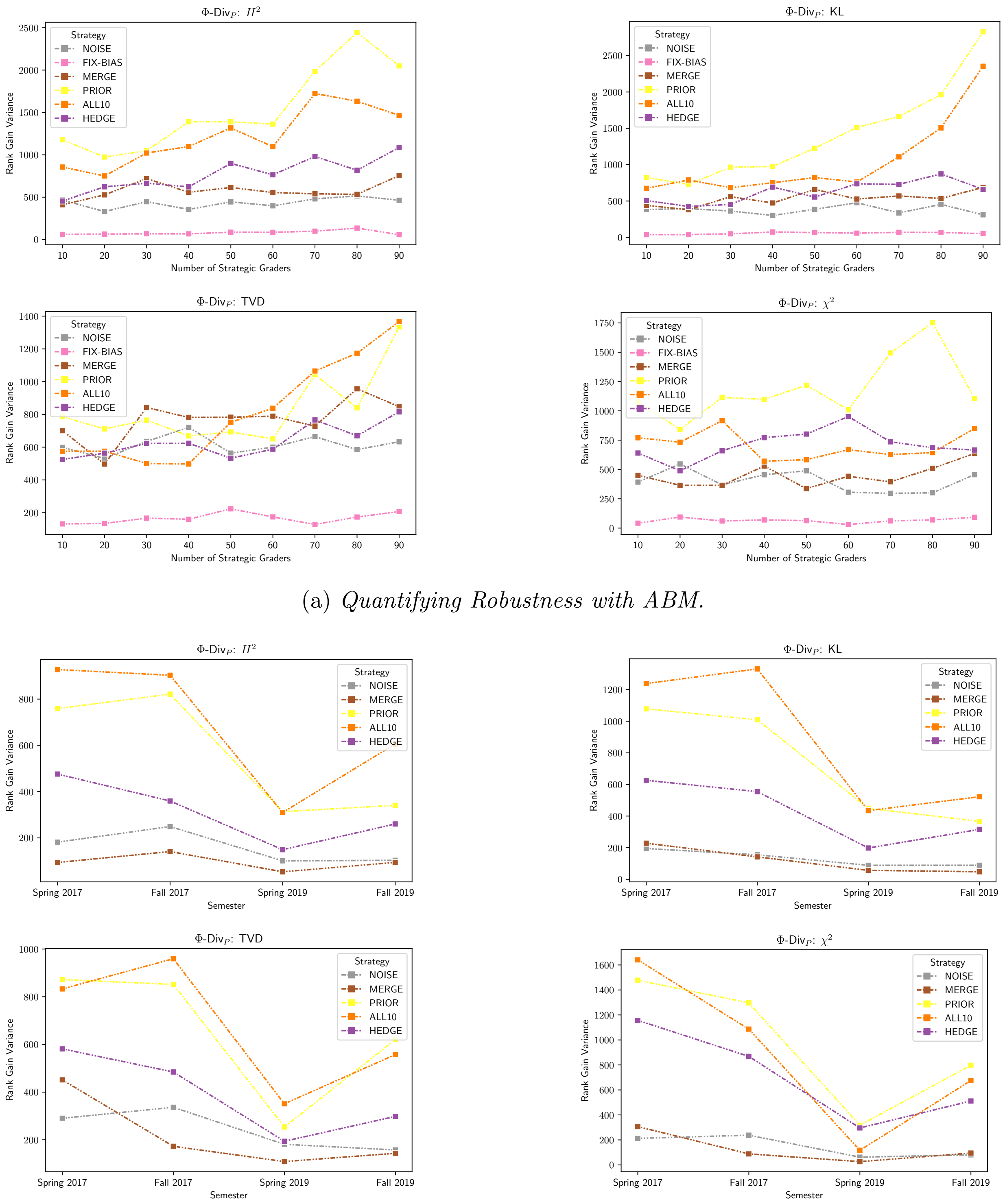


Figure C.9: Quantifying Robustness with Real Data. Variance of the rank gain achieved by a single student deviating from truthful to strategic reporting.



(a) *Quantifying Robustness with ABM.*

(b) *Quantifying Robustness with Real Data.*

Figure C.10: Improving Robustness with Bias Correction. Variance of the rank gain achieved by a single student deviating from truthful to strategic reporting for the the Φ-Div$_P$ mechanisms, when they correct for estimated agent biases.

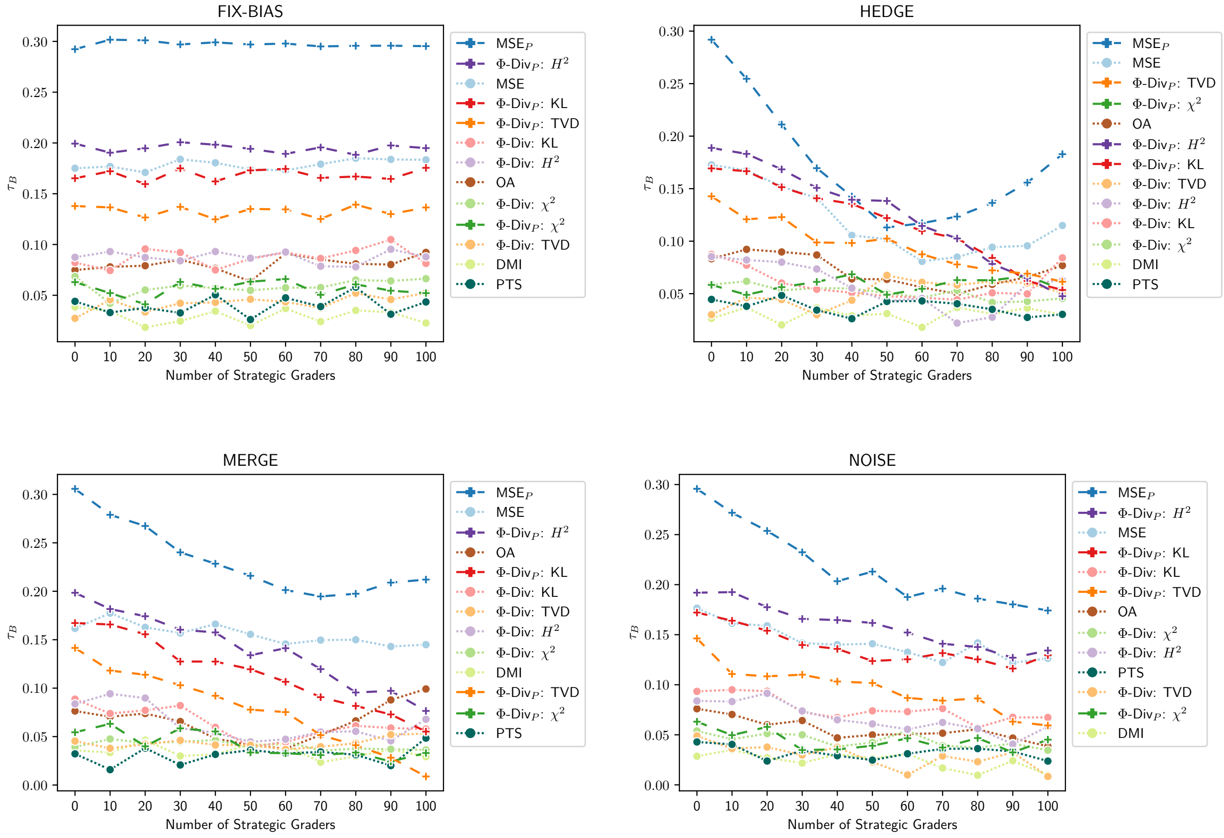


Figure C.11: Measurement Integrity in the Presence of Strategic Agents with DMI. Average values of τ_B for each informative strategy as the number of strategic agents varies. The average for each number of strategic graders is taken over 100 simulated semesters.

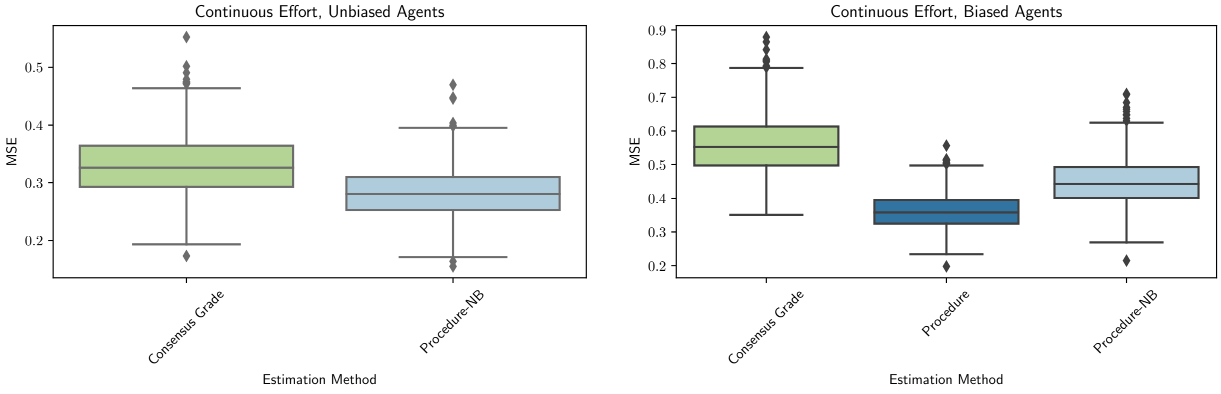


Figure C.12: Estimating Ground Truth Scores. Mean squared errors for the estimation of ground truth scores on 1000 assignments with 100 submissions each.

APPENDIX D

Software

Testing Conventional Wisdom (of the Crowd). The code for our analysis, available at https://github.com/burrelln/Testing-Conventional-Wisdom, is implemented in Python 3. To fit IRT models using the standard marginal maximum likelihood (MML) technique, we use the G. Item Response Theory (`girth`) package [84]. To perform calibrated statistical hypothesis tests for the unimodality of empirical distributions, we use the `modality` package [39]. In order for this package to work in Python 3, we had to modify the source code. In particular, it was necessary to change the `print` statements from the Python 2 syntax to the Python 3 syntax.

The rest of our tests and procedures were implemented by us. They rely on the following well-known Python packages: `numpy` [33], `pandas` [68; 112], `scikit-learn` [73], and `scipy` [102]. The logit-probabilities of correctness in the Figure 2.1 were plotted using the `seaborn` package [104].

Understanding When Peer Grades (Definitely) Outperform Instructor Grades. Our simulation code is implemented Python 3 using the NumPy [33], using the pandas [69; 112], SciPy [102], Scikit-learn [73], and statsmodels [87] packages. Results of the experiments are plotted using the pandas, Matplotlib [36], and seaborn [104] packages.

In order to make our experiments tractable, we ran simulations in parallel using the Lithops multi-cloud serverless computing framework [52; 83] to interface with IBM Code Engine and IBM Cloud Object Storage.

Our code for generating simulated data from and conducting inference for model $\mathbf{PG}_Z$ is based heavily on code that was graciously shared with us by Zarkoob et al. [124]. The software for computing quantiles of mixture distributions is based on code from [107].

Measurement Integrity in Peer Prediction. The code for our experiments is available at https://github.com/burrelln/Measurement-Integrity-and-Peer-Assessment.

Our ABM and all experiments are implemented Python 3 and use the NetworkX [31], NumPy [33], SciPy [102], and Scikit-learn [73] packages. Results of the experiments are plotted using the pandas [68; 112], Matplotlib [36], and seaborn [104] packages.

BIBLIOGRAPHY

[1] Rémi Bachelet, Drissa Zongo, and Aline Bourelle. Does peer grading work? How to implement and improve it? Comparing instructor and peer assessment in MOOC GdP. In *European MOOCs Stakeholders Summit 2015*, Proceedings of the Research Track, Mons, Belgium, May 2015. HAL:halshs-01146710v2.

[2] Yoram Bachrach, Thore Graepel, Tom Minka, and John Guiver. How to grade a test without knowing the answers — a bayesian graphical model for adaptive crowdsourcing and aptitude testing. In *29th International Conference on Machine Learning*, ICML 2012, 2012. arXiv:1206.6386.

[3] Valerio Basile, Federico Cabitza, Andrea Campagner, and Michael Fell. Toward a perspectivist turn in ground truthing for predictive computing, 2021. arXiv:2109.04270.

[4] William H. Batchelder, Royce Anders, and Zita Oravecz. *Cultural Consensus Theory*, pages 1–64. John Wiley & Sons, Ltd, 2018. ISBN 9781119170174. doi:10.1002/9781119170174.epcn506.

[5] Christoph Börgers. *Mathematics of Social Choice: Voting, Compensation, and Division*. Society for Industrial and Applied Mathematics (SIAM), Philadelphia, Pa., 2010. ISBN 9780898716955. doi:10.1137/1.9780898717624.

[6] Felix Brandt, Vincent Conitzer, Ulle Endriss, Jérôme Lang, and Ariel D Procaccia. *Handbook of Computational Social Choice*. Cambridge University Press, New York, NY, 2016. ISBN 9781107446984. doi:10.1017/CBO9781107446984.

[7] Glenn W. Brier. Verification of forecasts expressed in terms of probability. *Monthly Weather Review*, 78(1):1–3, 1950. doi:10.1175/1520-0493(1950)078<0001:VOFEIT>2.0.CO;2.

[8] Alex Burnap, Yi Ren, Richard Gerth, Giannis Papazoglou, Richard Gonzalez, and Panos Y. Papalambros. When crowdsourcing fails: A study of expertise on crowdsourced design evaluation. *Journal of Mechanical Design*, 137(3), 03 2015. ISSN 1050-0472. doi:10.1115/1.4029065.

[9] Noah Burrell and Grant Schoenebeck. Measurement integrity in peer prediction: A peer assessment case study. In *Proceedings of the 24th ACM Conference on Economics and Computation*, EC '23, New York, NY, USA, July 2023. Association for Computing Machinery. ISBN 979840070104. doi:10.1145/3580507.3597744.

[10] Noah Burrell and Grant Schoenebeck. Testing conventional wisdom (of the crowd). In Robin J. Evans and Ilya Shpitser, editors, *Proceedings of the Thirty-Ninth Conference on Uncertainty in Artificial Intelligence*, volume 216 of *Proceedings of Machine Learning Research*, pages 237–248. PMLR, 2023. URL `https://proceedings.mlr.press/v216/burrell23a.html`.

[11] Jesus Cerquides, Mehmet Oğuz Mülâyim, Jerónimo Hernández-González, Amudha Ravi Shankar, and Jose Luis Fernandez-Marquez. A conceptual probabilistic framework for annotation aggregation of citizen science data. *Mathematics*, 9(8), 2021. ISSN 2227-7390. doi:10.3390/math9080875.

[12] Anujit Chakraborty, Jatin Jindal, and Swaprava Nath. Removing bias and incentivizing precision in peer-grading. arXiv:1807.11657 (Working Paper), 2021.

[13] Yiling Chen, Arpita Ghosh, Michael Kearns, Tim Roughgarden, and Jennifer Wortman Vaughan. Mathematical foundations for social computing. *Commun. ACM*, 59(12): 102–108, December 2016. ISSN 0001-0782. doi:10.1145/2960403 arXiv:2007.03661.

[14] Anirban Dasgupta and Arpita Ghosh. Crowdsourced judgement elicitation with endogenous proficiency. In *Proceedings of the 22nd International Conference on World Wide Web*, WWW '13, pages 319–330, New York, NY, USA, 2013. Association for Computing Machinery. ISBN 9781450320351. doi:10.1145/2488388.2488417 arXiv:1303.0799.

[15] A. P. Dawid and A. M. Skene. Maximum likelihood estimation of observer error-rates using the em algorithm. *Journal of the Royal Statistical Society. Series C (Applied Statistics)*, 28(1):20–28, 1979. ISSN 00359254, 14679876.

[16] Luca de Alfaro and Michael Shavlovsky. 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, 2014. Association for Computing Machinery. ISBN 9781450326056. doi:10.1145/2538862.2538900 arXiv:1308.5273.

[17] Nicolas De Condorcet. *Essai sur l'application de l'analyse à la probabilité des décisions rendues à la pluralité des voix.* Cambridge University Press, 2014. Original work published 1785.

[18] Mikhail Drugov and Dmitry Ryvkin. How noise affects effort in tournaments. *Journal of Economic Theory*, 188:105065, 2020. ISSN 0022-0531. doi:10.1016/j.jet.2020.105065.

[19] Susan E. Embretson and Steven P. Reise. *Item Response Theory for Psychologists.* Multivariate Applications Book Series. Psychology Press, 2000. ISBN 9780805828184.

[20] Joshua M. Epstein. Why model? *Journal of Artificial Societies and Social Simulation*, 11(4):12, 2008. ISSN 1460-7425. URL `https://www.jasss.org/11/4/12.html`.

[21] Ido Erev, Alvin E Roth, Robert L Slonim, and Greg Barron. Learning and equilibrium as useful approximations: Accuracy of prediction on randomly selected constant sum games. *Economic Theory*, 33(1):29–51, 2007. doi:10.1007/s00199-007-0214-y.

[22] Boi Faltings, Radu Jurca, and Goran Radanovic. Peer truth serum: Incentives for crowdsourcing measurements and opinions, April 2017. arXiv:1704.05269.

[23] Boi Faltings, Goran Radanovic, and Ronald Brachman. *Game Theory for Data Science: Eliciting Truthful Information*. Morgan & Claypool Publishers, 2017. ISBN 1627057293. doi:10.1007/978-3-031-01577-9.

[24] Xi Alice Gao, Andrew Mao, Yiling Chen, and Ryan Prescott Adams. Trick or treat: Putting peer prediction to the test. In *15th ACM Conference on Economics and Computation (EC 2014)*, pages 507–524, June 2014. doi:10.1145/2600057.2602865.

[25] Xi Alice Gao, James R. Wright, and Kevin Leyton-Brown. Incentivizing evaluation via limited access to ground truth: Peer-prediction makes things worse. Workshop on Algorithmic Game Theory and Data Science at the 17th ACM Conference on Electronic Commerce, June 2016. arXiv:1606.07042.

[26] David Geiger, Michael Rosemann, and Erwin Fielt. Crowdsourcing information systems - a systems theory perspective. In D. Bunker, L. Dawson, M. Indulska, and P. Seltsikas, editors, *Proceedings of the 22nd Australasian Conference on Information Systems (ACIS) 2011 - Identifying the Information Systems Discipline*, pages 1–12. AIS Electronic Library (AISeL), 2011. URL `https://aisel.aisnet.org/acis2011/33/`.

[27] Andrew Gelman, John B. Carlin, Hal S. Stern, David B. Dunson, Aki Vehtari, and Donald B. Rubin. *Bayesian Data Analysis*. Chapman & Hall/CRC Texts in Statistical Science. Chapman and Hall/CRC, an imprint of Taylor and Francis, Boca Raton, FL, 3rd edition, 2013. ISBN 9780429113079. doi:10.1201/b16018.

[28] Tilmann Gneiting and Adrian E Raftery. Strictly proper scoring rules, prediction, and estimation. *Journal of the American Statistical Association*, 102(477):359–378, 2007. doi:10.1198/016214506000001437.

[29] Naman Goel and Boi Faltings. Deep bayesian trust: A dominant and fair incentive mechanism for crowd. *Proceedings of the AAAI Conference on Artificial Intelligence*, 33(01):1996–2003, Jul. 2019. doi:10.1609/aaai.v33i01.33011996.

[30] Mitchell L. Gordon, Kaitlyn Zhou, Kayur Patel, Tatsunori Hashimoto, and Michael S. Bernstein. The disagreement deconvolution: Bringing machine learning performance metrics in line with reality. In *Proceedings of the 2021 CHI Conference on Human Factors in Computing Systems*, CHI '21, New York, NY, USA, 2021. Association for Computing Machinery. ISBN 9781450380966. doi:10.1145/3411764.3445423.

[31] Aric A. Hagberg, Daniel A. Schult, and Pieter J. Swart. Exploring network structure, dynamics, and function using networkx. In Gaël Varoquaux, Travis Vaught, and Jarrod Millman, editors, *Proceedings of the 7th Python in Science Conference*, pages 11–15, Pasadena, CA USA, 2008.

[32] Yong Han, Wenjun Wu, and Xuan Zhou. Improving models of peer grading in spoc. https://www.researchgate.net/publication/361889950_Improving_Models_of_Peer_Grading_in_SPOC, 06 2017.

[33] Charles R. Harris, K. Jarrod Millman, St'efan J. van der Walt, Ralf Gommers, Pauli Virtanen, David Cournapeau, Eric Wieser, Julian Taylor, Sebastian Berg, Nathaniel J. Smith, Robert Kern, Matti Picus, Stephan Hoyer, Marten H. van Kerkwijk, Matthew Brett, Allan Haldane, Jaime Fern'andez del R'ıo, Mark Wiebe, Pearu Peterson, Pierre G'erard-Marchant, Kevin Sheppard, Tyler Reddy, Warren Weckesser, Hameer Abbasi, Christoph Gohlke, and Travis E. Oliphant. Array programming with NumPy. *Nature*, 585(7825):357–362, September 2020. doi:10.1038/s41586-020-2649-2.

[34] Chien-Ju Ho, Aleksandrs Slivkins, Siddharth Suri, and Jennifer Wortman Vaughan. Incentivizing high quality crowdwork. In *Proceedings of the 24th International Conference on World Wide Web*, WWW '15, pages 419–429, Republic and Canton of Geneva, CHE, 2015. International World Wide Web Conferences Steering Committee. ISBN 9781450334693. doi:10.1145/2736277.2741102 arXiv:1503.05897.

[35] Lu Hong and Scott E Page. Re-interpreting the condorcet jury theorem (draft). https://www.law.nyu.edu/sites/default/files/upload_documents/Re-Interpreting%20the%20Condorcet%20Jury%20Theorem.pdf, 2015.

[36] J. D. Hunter. Matplotlib: A 2d graphics environment. *Computing in Science & Engineering*, 9(3):90–95, 2007. doi:10.1109/MCSE.2007.55.

[37] E. T. Jaynes. Elementary parameter estimation. In G. Larry Bretthorst, editor, *Probability Theory: The Logic of Science*, pages 149–197. Cambridge University Press, 2003. doi:10.1017/CBO9780511790423.

[38] Kerstin Johnsson, Magnus Linderoth, and Magnus Fontes. What is a "unimodal" cell population? using statistical tests as criteria for unimodality in automated gating and quality control. *Cytometry Part A*, 91(9):908–916, 2017. doi:10.1002/cyto.a.23173.

[39] Kerstin Johnsson, Russell Jarvis, Avinash Varna, and Tom Pollard. modality, 2018. URL https://github.com/kjohnsson/modality. Version 1.1.

[40] Timothy C. Johnston. Lessons from MOOCs: Video lectures and peer assessment. *Academy of Educational Leadership Journal*, 19(2):91–97, 2015. ResearchGate link.

[41] Hyun Joon Jung and Matthew Lease. Improving quality of crowdsourced labels via probabilistic matrix factorization. In *Workshops at the Twenty-Sixth AAAI Conference on Artificial Intelligence*, 2012. URL https://www.aaai.org/ocs/index.php/WS/AAAIW12/paper/viewFile/5258/5609.

[42] David Karger, Sewoong Oh, and Devavrat Shah. Iterative learning for reliable crowdsourcing systems. *Advances in neural information processing systems*, 24, 2011. NeurIPS'11:4396.

[43] Kohta Katsuno, Masaki Matsubara, Chiemi Watanabe, and Atsuyuki Morishima. Improving reproducibility of crowdsourcing experiments. (Presented in the Work in Progress and Demo track, HCOMP 2019), 2019. URL https://www.humancomputation.com/2019/assets/papers/119.pdf.

[44] Faiza Khattak, Ansaf Salleb, and Anita Raja. Accurate crowd-labeling using item response theory, 03 2016. URL https://www.researchgate.net/publication/299389507_Accurate_Crowd-labeling_using_Item_Response_Theory.

[45] Richard Kim. Empirical methods in peer prediction. Master's thesis, Harvard Extension School, 2016. URL http://nrs.harvard.edu/urn-3:HUL.InstRepos:33797348.

[46] Yuqing Kong. Dominantly truthful multi-task peer prediction with a constant number of tasks. In *Proceedings of the 2020 ACM-SIAM Symposium on Discrete Algorithms (SODA20)*, 2020. doi:10.1137/1.9781611975994.147 arXiv:1911.00272.

[47] Yuqing Kong and Grant Schoenebeck. An information theoretic framework for designing information elicitation mechanisms that reward truth-telling. *ACM Transactions on Economics and Computation*, 7(1), January 2019. doi:10.1145/3296670 arXiv:1605.01021.

[48] Himabindu Lakkaraju, Jure Leskovec, Jon Kleinberg, and Sendhil Mullainathan. A bayesian framework for modeling human evaluations. In *Proceedings of the 2015 SIAM International Conference on Data Mining*, pages 181–189. SIAM, 2015. doi:10.1137/1.9781611974010.21.

[49] John P. Lalor, Hao Wu, and Hong Yu. Learning latent parameters without human response patterns: Item response theory with artificial crowds. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing*, EMNLP-IJCNLP '19, pages 4249–4259, Hong Kong, China, November 2019. Association for Computational Linguistics. doi: 10.18653/v1/D19-1434. arXiv:1908.11421.

[50] Yingkai Li, Jason D. Hartline, Liren Shan, and Yifan Wu. Optimization of scoring rules. In *Proceedings of the 23rd ACM Conference on Economics and Computation*, EC '22, pages 988–989, New York, NY, USA, 2022. Association for Computing Machinery. ISBN 9781450391504. doi:10.1145/3490486.3538338 arXiv:2007.02905.

[51] Yuan Li, Benjamin I. P. Rubinstein, and Trevor Cohn. Truth inference at scale: A bayesian model for adjudicating highly redundant crowd annotations. In *The World Wide Web Conference*, WWW '19, page 1028–1038, New York, NY, USA, 2019. Association for Computing Machinery. ISBN 9781450366748. doi:10.1145/3308558.3313459.

[52] The lithops development team. lithops-cloud/lithops, Accessed May 2023. URL https://lithops-cloud.github.io/docs/.

[53] Chao Liu and Yi-Min Wang. Truelabel + confusions: A spectrum of probabilistic models in analyzing multiple ratings. In *Proceedings of the 29th International Coference*

on International Conference on Machine Learning, ICML'12, pages 17–24, Madison, WI, USA, 2012. Omnipress. ISBN 9781450312851. arXiv:1206.4606.

[54] Qiang Liu, Jian Peng, and Alexander Ihler. Variational inference for crowdsourcing. In *Proceedings of the 25th International Conference on Neural Information Processing Systems - Volume 1*, NeurIPS'12, pages 692–700, Red Hook, NY, USA, 2012. Curran Associates Inc. NeurIPS'12:4627.

[55] Yang Liu, Juntao Wang, and Yiling Chen. Surrogate scoring rules. In *Proceedings of the 21st ACM Conference on Economics and Computation*, EC '20, pages 853–871, New York, NY, USA, 2020. Association for Computing Machinery. ISBN 9781450379755. doi:10.1145/3391403.3399488 arXiv:1802.09158.

[56] Yanxin Lu, Joe Warren, Christopher Jermaine, Swarat Chaudhuri, and Scott Rixner. Grading the graders: Motivating peer graders in a mooc. In *Proceedings of the 24th International Conference on World Wide Web*, page 680–690, Republic and Canton of Geneva, CHE, 2015. International World Wide Web Conferences Steering Committee. ISBN 9781450334693. doi:10.1145/2736277.2741649.

[57] Andreu Mas-Colell, Michael Dennis Whinston, and Jerry R. Green. *Microeconomic theory*. Oxford University Press, 1995. ISBN 9780195073409.

[58] Winter Mason and Duncan J. Watts. Financial incentives and the "performance of crowds". In *Proceedings of the ACM SIGKDD Workshop on Human Computation*, HCOMP '09, pages 77–85, New York, NY, USA, 2009. Association for Computing Machinery. ISBN 9781605586724. doi:10.1145/1600150.1600175.

[59] Fei Mi and Dit-Yan Yeung. Probabilistic graphical models for boosting cardinal and ordinal peer grading in moocs. In *Proceedings of the Twenty-Ninth AAAI Conference on Artificial Intelligence*, AAAI'15, page 454–460. AAAI Press, 2015. ISBN 0262511290.

[60] Nolan Miller, Paul Resnick, and Richard Zeckhauser. Eliciting informative feedback: The peer-prediction method. *Management Science*, 51(9):1359–1373, 2005. doi:10.1287/mnsc.1050.0379.

[61] Barzan Mozafari, Purnamrita Sarkar, Michael J. Franklin, Michael I. Jordan, and Samuel Madden. BM data set. URL `https://github.com/ipeirotis/Get-Another-Label/tree/master/data/BarzanMozafari`.

[62] Barzan Mozafari, Purnamrita Sarkar, Michael J. Franklin, Michael I. Jordan, and Samuel Madden. Active learning for crowd-sourced databases, 2012. arXiv:1209.3686.

[63] Barzan Mozafari, Purnamrita Sarkar, Michael J. Franklin, Michael I. Jordan, and Samuel Madden. Scaling up crowd-sourcing to very large datasets: A case for active learning. *PVLDB*, 8(2):125–136, 2014.

[64] Eric Neyman, Georgy Noarov, and S. Matthew Weinberg. Binary scoring rules that incentivize precision. In *Proceedings of the 22nd ACM Conference on Economics*

and Computation, EC '21, pages 718–733, New York, NY, USA, 2021. Association for Computing Machinery. ISBN 9781450385541. doi:10.1145/3465456.3467639 arXiv:2002.10669.

[65] Documentation: `sklearn.linear_model.ARDRegression`, Accessed: Jun. 2023. URL `https://scikit-learn.org/stable/modules/generated/sklearn.linear_model.ARDRegression.html`.

[66] Jungseul Ok, Sewoong Oh, Jinwoo Shin, and Yung Yi. Optimality of belief propagation for crowdsourced classification. In *International Conference on Machine Learning*, pages 535–544. PMLR, 2016. arXiv:1602.03619.

[67] Naoki Otani, Toshiaki Nakazawa, Daisuke Kawahara, and Sadao Kurohashi. IRT-based aggregation model of crowdsourced pairwise comparison for evaluating machine translations. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 511–520, Austin, Texas, November 2016. Association for Computational Linguistics. doi:10.18653/v1/D16-1049.

[68] The pandas development team. pandas-dev/pandas: Pandas, February 2020. doi:10.5281/zenodo.3509134.

[69] The pandas development team. pandas-dev/pandas: Pandas, 2020. doi:10.5281/zenodo.3509134.

[70] David Parkes and Sven Seuken. Social computing and human computation. In *Economics and Computation*, chapter 13, pages 329–360. Book in Preparation, 2016. URL `http://economicsandcomputation.org/`.

[71] Rebecca J. Passonneau and Bob Carpenter. The benefits of a model of annotation. *Transactions of the Association for Computational Linguistics*, 2:311–326, 10 2014. ISSN 2307-387X. doi:`10.1162/tacl_a_00185`.

[72] Silviu Paun, Bob Carpenter, Jon Chamberlain, Dirk Hovy, Udo Kruschwitz, and Massimo Poesio. Comparing Bayesian Models of Annotation. *Transactions of the Association for Computational Linguistics*, 6:571–585, 12 2018. ISSN 2307-387X. doi:`10.1162/tacl_a_00040`.

[73] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830, 2011.

[74] Chris Piech, Jonathan Huang, Zhenghao Chen, Chuong Do, Andrew Ng, and Daphne Koller. Tuned models of peer assessment in moocs. In *6th International Conference on Educational Data Mining*, Memphis, TN, 2013. arXiv:1307.2579.

[75] Barbara Plank. The "problem" of human label variation: On ground truth in data, modeling and evaluation. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 10671–10682, Abu Dhabi, United

Arab Emirates, December 2022. Association for Computational Linguistics. URL https://aclanthology.org/2022.emnlp-main.731.

[76] Dražen Prelec, H. Sebastian Seung, and John McCoy. A solution to the single-question crowd wisdom problem. *Nature*, 541(7638):532–535, Jan 2017. ISSN 1476-4687. doi:10.1038/nature21054.

[77] Goran Radanovic, Boi Faltings, and Radu Jurca. Incentives for effort in crowdsourcing using the peer truth serum. *ACM Trans. Intell. Syst. Technol.*, 7(4), mar 2016. ISSN 2157-6904. doi:10.1145/2856102.

[78] Vikas C. Raykar, Shipeng Yu, Linda H. Zhao, Gerardo Hermosillo Valadez, Charles Florin, Luca Bogoni, and Linda Moy. Learning from crowds. *J. Mach. Learn. Res.*, 11:1297–1322, aug 2010. ISSN 1532-4435. URL http://jmlr.org/papers/v11/raykar10a.html.

[79] Mark D. Reckase. *Multidimensional Item Response Theory*. Statistics for Social and Behavioral Sciences. Springer, New York, NY, 1 edition, 2009. ISBN 978-0-387-89975-6. doi:10.1007/978-0-387-89976-3.

[80] Jonathan Rees. Peer grading can't work. *Inside Higher Ed*, 04 March 2013. Available at: https://www.insidehighered.com/views/2013/03/05/essays-flaws-peer-grading-moocs (Accessed: July 1st, 2023).

[81] Paul Resnick, Yuqing Kong, Grant Schoenebeck, and Tim Weninger. Survey equivalence: A procedure for measuring classifier accuracy against human labels, 2021. arXiv:2106.01254.

[82] Thomas L Saaty. Scales from measurement–not measurement from scales. In *Proceedings of the 17th International Conference on Multiple Criteria Decision Making, Whistler, BC Canada*, pages 6–11, 2004.

[83] Josep Sampé, Marc Sánchez-Artigas, Gil Vernik, Ido Yehekzel, and Pedro García-López. Outsourcing data processing jobs with lithops. *IEEE Transactions on Cloud Computing*, 11(1):1026–1037, 2023. doi:10.1109/TCC.2021.3129000.

[84] Ryan Sanchez. Girth: G. item response theory, November 2021. URL https://github.com/eribean/girth. Version 0.8.0.

[85] Grant Schoenebeck and Fang-Yi Yu. Learning and strongly truthful multi-task peer prediction: A variational approach. In *12th Innovations in Theoretical Computer Science Conference (ITCS 2021)*, January 2021. arXiv:2009.14730.

[86] Documentation: scipy.stats.gaussian_kde, Accessed: Oct. 2022. URL https://docs.scipy.org/doc/scipy/reference/generated/scipy.stats.gaussian_kde.html.

[87] Skipper Seabold and Josef Perktold. statsmodels: Econometric and statistical modeling with python. In *9th Python in Science Conference*, 2010.

[88] Victor S. Sheng, Foster Provost, and Panagiotis G. Ipeirotis. Get another label? improving data quality and data mining using multiple, noisy labelers. In *Proceedings of the 14th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '08, pages 614–622, New York, NY, USA, 2008. Association for Computing Machinery. ISBN 9781605581934. doi:10.1145/1401890.1401965.

[89] Aashish Sheshadri and Matthew Lease. Square: A benchmark for research on computing crowd consensus. *Proceedings of the AAAI Conference on Human Computation and Crowdsourcing*, 1(1):156–164, Nov. 2013. URL https://ojs.aaai.org/index.php/HCOMP/article/view/13088.

[90] Victor Shnayder and David C. Parkes. Practical peer prediction for peer assessment. In *Fourth AAAI Conference on Human Computation and Crowdsourcing*, Austin, TX, October 2016. URL http://nrs.harvard.edu/urn-3:HUL.InstRepos:34732142.

[91] Victor Shnayder, Arpit Agarwal, Rafael Frongillo, and David C. Parkes. Informed truthfulness in multi-task peer prediction. In *17th ACM Conference on Economics and Computation (EC 2016)*, pages 179–196, July 2016. doi:10.1145/2940716.2940790 arXiv:1603.03151.

[92] Victor Shnayder, Rafael M. Frongillo, and David C. Parkes. Measuring performance of peer prediction mechanisms using replicator dynamics. In *25th International Joint Conference on Artificial Intelligence (IJCAI 2016)*, New York, NY, July 2016. URL http://nrs.harvard.edu/urn-3:HUL.InstRepos:32220916.

[93] Bernard W Silverman. *Density Estimation for Statistics and Data Analysis*, volume 26 of *Monographs on Statistics and Applied Probability*. Chapman and Hall, London, 1986.

[94] Documentation: sklearn.mixture.GaussianMixture, Accessed: Oct. 2022. URL https://scikit-learn.org/stable/modules/generated/sklearn.mixture.GaussianMixture.html.

[95] Rion Snow, Brendan O'Connor, Daniel Jurafsky, and Andrew Y. Ng. RTE and TEMP data sets. URL http://sites.google.com/site/nlpannotations/. Original source can no longer be accessed except through the Wayback Machine from the Internet Archive: archived site (Mar. 2023); archived download link (Oct. 2020).

[96] Rion Snow, Brendan O'Connor, Daniel Jurafsky, and Andrew Y. Ng. Cheap and fast—but is it good?: Evaluating non-expert annotations for natural language tasks. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, EMNLP '08, pages 254–263, Stroudsburg, PA, USA, 2008. Association for Computational Linguistics.

[97] Harald Søndergaard. Learning from and with peers: The different roles of student peer reviewing. *SIGCSE Bull.*, 41(3):31–35, July 2009. ISSN 0097-8418. doi:10.1145/1595496.1562893.

[98] SQUARE. Links to data sets, Accessed: Oct. 2022. URL https://ir.ischool.utexas.edu/square/data.html.

[99] S. S. Stevens. On the theory of scales of measurement. *Science*, 103(2684):677–680, 1946. ISSN 00368075, 10959203. URL http://www.jstor.org/stable/1671815.

[100] James Surowiecki. *The Wisdom of Crowds: Why the Many Are Smarter Than the Few and How Collective Wisdom Shapes Business, Economies, Societies and Nations.* Doubleday, 1st ed. edition, 2004.

[101] Matteo Venanzi, William Teacy, Alexander Rogers, and Nicholas Jennings. Sentiment popularity data set, 2015. doi: 10.5258/SOTON/376544.

[102] Pauli Virtanen, Ralf Gommers, Travis E. Oliphant, Matt Haberland, Tyler Reddy, David Cournapeau, Evgeni Burovski, Pearu Peterson, Warren Weckesser, Jonathan Bright, Stéfan J. van der Walt, Matthew Brett, Joshua Wilson, K. Jarrod Millman, Nikolay Mayorov, Andrew R. J. Nelson, Eric Jones, Robert Kern, Eric Larson, C J Carey, İlhan Polat, Yu Feng, Eric W. Moore, Jake VanderPlas, Denis Laxalde, Josef Perktold, Robert Cimrman, Ian Henriksen, E. A. Quintero, Charles R. Harris, Anne M. Archibald, Antônio H. Ribeiro, Fabian Pedregosa, Paul van Mulbregt, and SciPy 1.0 Contributors. SciPy 1.0: Fundamental Algorithms for Scientific Computing in Python. *Nature Methods*, 17:261–272, 2020. doi:10.1038/s41592-019-0686-2.

[103] Tianqi Wang, Xia Jing, Qi Li, Jing Gao, and Jie Tang. Improving peer assessment accuracy by incorporating relative peer grades. In *International Conference on Educational Data Mining 2019*, EDM2019, Montréal, QC, 2019. URL https://eric.ed.gov/?id=ED599252.

[104] Michael Waskom and the seaborn development team. mwaskom/seaborn, September 2020. doi:10.5281/zenodo.592845.

[105] Audrey Watters. Confessions of a mooc-her. *Campus Technology Magazine*, 26(4):6–7, 2012. ISSN 15537544.

[106] Duncan J. Watts. *Everything Is Obvious: Once You Know the Answer.* Crown Business, New York, 2011. ISBN 9780385531696.

[107] Andrew M. Webb. Quantiles of mixture distributions, 2017. URL https://www.awebb.info/probability/2017/05/12/quantiles-of-mixture-distributions.html.

[108] Jiaheng Wei, Zhaowei Zhu, Hao Cheng, Tongliang Liu, Gang Niu, and Yang Liu. Learning with noisy labels revisited: A study using real-world human annotations. In *International Conference on Learning Representations*, 2022. URL https://openreview.net/forum?id=TBWA6PLJZQm.

[109] Peter Welinder and Pietro Perona. Online crowdsourcing: Rating annotators and obtaining cost-effective labels. In *2010 IEEE Computer Society Conference on*

Computer Vision and Pattern Recognition - Workshops, pages 25–32, June 2010. doi:10.1109/CVPRW.2010.5543189.

[110] Peter Welinder, Steve Branson, Serge Belongie, and Pietro Perona. WB data set. URL https://github.com/welinder/cubam/tree/public/demo/bluebirds.

[111] Peter Welinder, Steve Branson, Pietro Perona, and Serge Belongie. The multidimensional wisdom of crowds. In J. Lafferty, C. Williams, J. Shawe-Taylor, R. Zemel, and A. Culotta, editors, *Advances in Neural Information Processing Systems*, volume 23, pages 2424–2432. Curran Associates, Inc., 2010. URL https://proceedings.neurips.cc/paper/2010/file/0f9cafd014db7a619ddb4276af0d692c-Paper.pdf.

[112] Wes McKinney. Data Structures for Statistical Computing in Python. In Stéfan van der Walt and Jarrod Millman, editors, *Proceedings of the 9th Python in Science Conference*, pages 56–61, 2010. doi:10.25080/Majora-92bf1922-00a.

[113] Jacob Whitehill, Paul Ruvolo, Tingfan Wu, Jacob Bergsma, and Javier Movellan. Whose vote should count more: Optimal integration of labels from labelers of unknown expertise. In *Proceedings of the 22nd International Conference on Neural Information Processing Systems*, NeurIPS'09, page 2035–2043, Red Hook, NY, USA, 2009. Curran Associates Inc. ISBN 9781615679119. URL https://proceedings.nips.cc/paper/2009/file/f899139df5e1059396431415e770c6dd-Paper.pdf.

[114] Benjamin Wiggins. Can peer grading actually work? *More or Less Bunk*, 17 February 2014. Available at: https://moreorlessbunk.wordpress.com/2014/02/17/can-peer-grading-actually-work/ (Accessed: July 1st, 2023).

[115] Jens Witkowski and David C. Parkes. Learning the prior in minimal peer prediction. In *3rd Workshop on Social Computing and User Generated Content*, SC 2013, pages 39:1–39:12, Philadelphia, PA, June 2013. URL http://nrs.harvard.edu/urn-3:HUL.InstRepos:34222829.

[116] James R. Wright, Chris Thornton, and Kevin Leyton-Brown. Mechanical ta: Partially automated high-stakes peer grading. In *46th ACM Technical Symposium on Computer Science Education*, pages 96–101, February 2015. doi:10.1145/2676723.2677278.

[117] Shengwei Xu, Grant Schoenebeck, Yichi Zhang, and Paul Resnick. Spot check equivalence: a metric of information elicitation mechanisms' motivational proficiency. Work in progress., 2023.

[118] Ming Yin, Mary L. Gray, Siddharth Suri, and Jennifer Wortman Vaughan. The communication network within the crowd. In *Proceedings of the 25th International Conference on World Wide Web*, WWW '16, pages 1293–1303, Republic and Canton of Geneva, CHE, 2016. International World Wide Web Conferences Steering Committee. ISBN 9781450341431. doi:10.1145/2872427.2883036.

[119] Yaming Yu. On the maximal correlation coefficient. *Statistics & Probability Letters*, 78(9):1072–1075, 2008. doi:10.1016/j.spl.2007.10.006.

[120] Zheng Yuan and Doug Downey. *Practical Methods for Semi-Automated Peer Grading in a Classroom Setting*, pages 363–367. Association for Computing Machinery, New York, NY, USA, 2020. ISBN 9781450368612. doi:10.1145/3340631.3394878.

[121] Hedayat Zarkoob. Personal correspondence, April 2023.

[122] Hedayat Zarkoob, Hu Fu, and Kevin Leyton-Brown. Report-sensitive spot-checking in peer-grading systems. In *Proceedings of the 19th International Conference on Autonomous Agents and MultiAgent Systems*, AAMAS '20, pages 1593–1601, Richland, SC, 2020. International Foundation for Autonomous Agents and Multiagent Systems. ISBN 9781450375184. arXiv:1906.05884.

[123] Hedayat Zarkoob, Farzad Abdolhosseini, and Kevin Leyton-Brown. Mechanical ta 2: A system for peer grading with ta support, 2021. arXiv:2101.10078.

[124] Hedayat Zarkoob, Greg d'Eon, Lena Podina, and Kevin Leyton-Brown. Better peer grading through bayesian inference. In *37th AAAI Conference on Artificial Intelligence*, AAAI-23, pages 6137–6144, June 2023. doi:10.1609/aaai.v37i5.25757 arXiv:2209.01242.

[125] Yichi Zhang and Grant Schoenebeck. High-effort crowds: Limited liability via tournaments. In *Proceedings of the ACM Web Conference 2023*, WWW '23, pages 3467–3477, New York, NY, USA, 2023. Association for Computing Machinery. ISBN 9781450394161. doi:10.1145/3543507.3583334.

[126] Yuchen Zhang, Xi Chen, Dengyong Zhou, and Michael I Jordan. Spectral methods meet em: A provably optimal algorithm for crowdsourcing. In Z. Ghahramani, M. Welling, C. Cortes, N. Lawrence, and K.Q. Weinberger, editors, *Advances in Neural Information Processing Systems*, volume 27. Curran Associates, Inc., 2014. arXiv:1406.3824.

[127] Dengyong Zhou, Sumit Basu, Yi Mao, and John Platt. Learning from the wisdom of crowds by minimax entropy. In F. Pereira, C.J. Burges, L. Bottou, and K.Q. Weinberger, editors, *Advances in Neural Information Processing Systems*, volume 25. Curran Associates, Inc., 2012. URL `https://proceedings.neurips.cc/paper/2012/file/46489c17893dfdcf028883202cefd6d1-Paper.pdf`.

[128] Dengyong Zhou, Qiang Liu, John C. Platt, Christopher Meek, and Nihar B. Shah. Regularized minimax conditional entropy for crowdsourcing, 2015. arXiv:1503.07240.