Machine-Learning Aided Peer Prediction

YANG LIU, Harvard University
YILING CHEN, Harvard University

Information Elicitation without Verification (IEWV) is a classic problem where a principal wants to truthfully elicit high-quality answers of some tasks from strategic agents despite that she cannot evaluate the quality of agents' contributions. The established solution to this problem is a class of peer prediction mechanisms, where each agent is rewarded based on how his answers compare with those of his peer agents. These peer prediction mechanisms are designed by exploring the stochastic correlation of agents' answers. The prior distribution of agents' true answers is often assumed to be known to the principal or at least to the agents.

In this paper, we consider the problem of IEWV for heterogeneous binary signal tasks, where the answer distributions for different tasks are different and unknown a priori. A concrete setting is eliciting labels for training data. Here, data points are represented by their feature vectors $\mathbf{x}$'s and the principal wants to obtain corresponding binary labels y's from strategic agents. We design peer prediction mechanisms that leverage not only the stochastic correlation of agents' labels for the same feature vector $\mathbf{x}$ but also the (learned) correlation between feature vectors $\mathbf{x}$'s and the ground-truth labels y's. In our mechanism, each agent is rewarded by how his answer compares with a reference answer generated by a classification algorithm specialized for dealing with noisy data. Every agent truthfully reporting and exerting high effort form a Bayesian Nash Equilibrium. Some benefits of this approach include: (1) we do not need to always re-assign each task to multiple workers to obtain redundant answers. (2) A class of surrogate loss functions for binary classification can help us design new reward functions for peer prediction. (3) Symmetric uninformative reporting strategy (pure or mixed) is not an equilibrium strategy. (4) The principal does not need to know the joint distribution of workers' information a priori. We hope this work can point to a new and promising direction of information elicitation via more intelligent algorithms.

CCS Concepts: •**Information systems** → **Incentive schemes;** •**Theory of computation** → **Algorithmic mechanism design;** •**Computing methodologies** → *Supervised learning by classification;*

1 INTRODUCTION

Information Elicitation without Verification (IEWV) [25] is a classic problem where a principal wants to truthfully elicit high-quality answers of some tasks from strategic agents despite that she cannot evaluate the quality of agents' contributions. The lack of verification is either due to the subjective nature of the answers or because verifying answers is too costly to be practical. For example, a principal may be interested in knowing whether people find each of a set of restaurants as a desirable place for Valentine's Day dinner. Or the principal may want to find out whether each of a set of websites contains adult content.

A class of mechanisms, collectively called *peer prediction* [4, 5, 13, 15, 17, 22], has been proposed for the IEWV problem. The high-level idea of these peer prediction mechanisms is to take advantage

This work is supported by the National Science Foundation under grant CCF-1301976. Authors' emails: Yang Liu yangl@seas.harvard.edu, Yiling Chen yiling@seas.harvard.edu.

EC'17, June 26–30, 2017, Cambridge, MA, USA. ACM ISBN 978-1-4503-4527-9/17/06...$15.00.

DOI: http://dx.doi.org/10.1145/3033274.3085126

of the stochastic correlation of agents' answers for the same (or sometimes different) tasks and reward an agent based on how his answers compare with those of some peer agents. The reward functions of peer prediction mechanisms are designed in a way such that all agents truthfully reporting their answers (or exerting effort to obtain high-quality answers and then truthfully reporting them) forms a Bayesian Nash equilibrium (BNE). This is quite marvelous to achieve given the lack of direct verification.

One major benefit of being able to elicit high-quality information from strategic agents without direct verification is that it can serve as a way to collect training samples for machine learning. In fact, getting training data from crowd workers (e.g., to label images and to collect personal data for social studies) has become a popular practice for machine learning [12, 21]. For instance, the above-mentioned principal may want to train a classifier on elicited data to help her predict whether a new restaurant is Valentine's Day friendly or whether a new website contains adult content. Because people are error-prone and/or strategic, techniques on learning from crowd-generated training samples have been developed to innovatively handle the noise of the training data [2, 12, 16, 20].

The above brief overview of the current status quo of solutions to the IEWV problem and how crowd-generated data are used to train machine learning algorithms reveals one interesting observation. We can view the tasks (e.g. restaurants and websites) that a principal wants to elicit information about as feature vectors $\mathbf{x}$'s and the desirable information about these tasks as their corresponding labels y's. Then, the reward functions of existing peer prediction mechanisms are designed by leveraging the correlations of y's, because such correlation gives a way to accurately predict the true label for a particular feature vector and the prediction can be used as a benchmark to evaluate the elicited label for the feature vector, while machine learning algorithms trained on the elicited data aims to learn the correlation between $\mathbf{x}$'s and y's. Since the learned structural relationship between $\mathbf{x}$'s and y's can also offer a good prediction on the true label of any given feature vector, this suggests that we may take advantage of the correlation between $\mathbf{x}$'s and y's, in addition to the correlation of y's, to design peer prediction mechanisms that are possibly more efficient at eliciting information from strategic agents for heterogeneous tasks for IEWV and may have better incentive properties. This is the approach that we take in this paper.

More specifically, we design peer prediction mechanisms for eliciting binary labels from strategic agents for a set of heterogeneous tasks. Agents observe labels of tasks with errors (i.e. agents' observed labels sometimes are different from the true labels) and they can strategically decide whether to truthfully reveal their observed labels or not. The exact error rates are unknown to the principal. Our mechanism trains a classifier using the collected data, this classifier can generate a prediction on the label of a given feature vector and this prediction is then used as a machine-generated reference report in a scoring function to evaluate a reported label for this feature vector. We find that if the structural relationship between $\mathbf{x}$'s and the corresponding true y's is "learnable" in the sense that there exists a concept class $\mathcal{F}$ with bounded VC-dimension such that the optimal classifier $f^* \in \mathcal{F}$ has performance that is better than random guessing, which is a rather weak requirement for classification problems, then we can use the above described approach to design a peer prediction mechanism such that all agents truthfully reporting their observed labels is a BNE. In an effort sensitive model where agents can incur a costly effort to reduce the error of their observed labels, this result generalizes to induce a BNE where all agents exert the effort and then truthfully report their observed labels.

The advantages of this machine-learning inspired approach and our designed mechanism include: (1) Our mechanism is more "efficient" in the sense that we do not need to assign every task to multiple agents to obtain redundant labels, while almost all existing peer prediction mechanisms depend on such redundant assignments. (2) Our mechanisms have better incentive properties.

Everyone always reporting the same label or the same distribution of labels independent of the observed label or everyone always reporting a fixed permutation of observed labels is not an equilibrium in our mechanism, while these are equilibria in existing peer prediction mechanisms. (3) Our mechanism allows the principal and the agents to have less knowledge about agents' observations. The principal and the agents do not need to know the error rates of the agent observations a priori and hence do not know the prior distribution of agents' labels. Many peer prediction mechanisms require that the principal knows the joint distribution of agents' labels to operate the mechanisms [9, 10, 15]. Most peer prediction mechanisms require agents know this joint distribution for the equilibrium results to hold [4, 9, 10, 15, 22]. (4) Our mechanism is especially suitable for heterogeneous tasks, while existing multi-task peer prediction mechanisms [4, 13, 22] have been designed for homogeneous tasks, with the exception of [14]. (5) A class of surrogate loss functions for binary classification can help us design reward functions for peer prediction.

At the technical level, our approach is built upon a set of machine learning techniques, often referred as learning with "noisy data" [16]. This set of techniques helps to learn a classifier, using training data with *known* error, that converges to the optimal classifier as if it is trained on data without error. It offers a way to counter the noise of the data to generate a good prediction on the true label of a feature vector. Our setting has additional challenges in that the error of agents' observations is unknown and the reported labels are a result of agents' strategic decisions.

The rest of the work is organized as follows. We survey related work in the rest of this section. Our problem is formulated in Section 2. Section 3 introduces our approach, and reviews preliminaries on machine learning techniques that we build our results on. Our main results are discussed in Section 4. Then we discuss how we overcome the challenge of not knowing the errors of agents' observations in Section 5. In Section 6, we show how our mechanism can be adapted so that uninformative strategies do not form an equilibrium. A simple output agreement type of mechanism is then presented in Section 7. We show the extension of our results to effort-sensitive workers in Section 8. Section 9 concludes the paper. Full version with appendix of this paper can be found on arXiv and the authors' websites.

1.1 Related work

Our work is an addition to an already large literature on *peer prediction* [4, 9, 10, 15, 17, 18, 27, 28]. The term *peer prediction* was coined up by Miller et al. [2005] who showed that any *strictly proper scoring rule* [7] for truthfully eliciting information about events with (future observable) ground truth can be turned into a scoring function for truthfully eliciting information for the IEWV problem if the principal knows the joint distribution of private signals and this distribution is common knowledge to all agents. The mechanism of Miller et al. [2005] has truthful reporting as a BNE. But it also has uninformative BNEs where all agents play a strategy that is independent of their observed signal. Moreover, agents receive higher payoff at the uninformative BNEs than at the truthful BNE. One year prior to the publication of Miller et al. [2005], Prelec [2004] proposed *Bayesian Truth Serum* (BTS), an elegant mechanism for IEVW that achieves truthful reporting as a BNE for a large enough population of agents by asking each agent to report not only his observed signal but also a prediction on the distribution of others' reports. While BTS still requires agents to know the joint distribution of private signals for the truthful BNE to hold, the principal in BTS doesn't need to know this distribution to operate the mechanism. BTS also admits uninformative equilibria. A sequence of follow up work has been done to relax the assumptions made by these mechanisms [18, 27, 28] and to make the truthful reporting equilibrium more focal [10]. More recently, Witkowski et al. [2013] and Dasgupta and Ghosh [2013] formally studied an effort sensitive model for eliciting binary signals for IEWV. In particular, Dasgupta and Ghosh [2013] proposed a multi-task peer prediction mechanism that ensures a desirable BNE where every agent first exerts maximum effort to obtain a high-quality signal and then truthfully reports it. Moreover, among all BNEs, this equilibrium gives the highest payoff to all agents. Shnayder et al. [2016] and Kong and Schoenebeck [2016] then extended the results to non-binary signal elicitation. However, in all existing mechanisms, there exists BNEs where agents play an uninformative reporting strategy, although such equilibrium leads to lower agent payoff than the truthful equilibrium in the recent mechanisms of Dasgupta and Ghosh [2013], Shnayder et al. [2016] and Kong and Schoenebeck [2016]. Furthermore, agents reporting a fixed permutation of signals is also a BNE in existing mechanisms and this equilibrium offers the same payoff to agents as the truthful equilibrium. Our mechanism in this paper removes both the uninformative equilibria and the permutation equilibrium.

Our mechanism uses a "machine" prediction as a reference answer. This has some resemblance to the notion of creating a reference answer from a model's output. Several studies focused on using models that aggregate answers from different agents [6, 11, 19]. These studies didn't consider a machine learning setting like us, and didn't establish equilibrium behavior.

From a different angle, our work fits into the notion of *learning with strategic data sources*, which concerns machine learning systems when either their training data or incoming test data or both come from strategic agents. For example, several recent results have focused on optimally training regression models via incentivizing high-quality training data, when they are collected from strategic agents who are either effort sensitive [2] or privacy sensitive [3, 8]. Our work focuses on classification problems.

In machine learning, how to learn appropriately with non-strategic, yet biased data has received quite a bit of attention in recently [16, 20]. We demonstrate yet another novel application of these methods for information elicitation. In some sense, our work completes this line of research by demonstrating that a method for learning from noisy training data can also be made Bayesian incentive compatible for eliciting the training data.

2 PROBLEM FORMULATION

A principal has a set of data $\{\mathbf{x}_i\}_{i=1}^N$, where each $x_i \in \mathbb{R}^M$ can be viewed as an M-dimensional feature vector. The space of x_i's is $\mathcal{X}$. This set of data has corresponding (binary) labels, $\{\mathbf{y}_i\}_{i=1}^N$ where $y_i \in \{-1, +1\}$, that are unknown to the principal. $\{(\mathbf{x}_i, y_i)\}_{i=1}^N$ are drawn i.i.d. according to an unknown joint distribution $\mathcal{D}$. The prior probabilities for the $+1$ and the -1 labels are $\mathcal{P}_+$ and $\mathcal{P}_-$ respectively. The principal knows $\mathcal{P}_+$ and $\mathcal{P}_-$. She can ask a set of agents (e.g. crowd workers) to label the data. Without knowing the true labels, she cannot directly verify the elicited labels. The principal would like to design reward mechanisms so that she can obtain high-quality labels from the agents.

2.1 Agent model

The principal recruits $T \geq N$ agents (workers). Each worker i is assigned exactly one data point, denoted $\mathbf{x}_i$ (with a bit abuse of notation), and is asked to report a label $\hat{y}_i$. Each data point is assigned to one worker and sometimes more.[1] The re-assigned tasks are randomly selected, and if selected the task will be re-assigned once (so assigned twice in all, for details please refer to Section 5). Denote $\mathcal{U}$ as the set that contains each independent task exactly once, so $|\mathcal{U}| = N$. We consider two worker observation models in this paper: an *effort insensitive* model where each worker exogenous observes a label with noise and an *effort sensitive* model where a worker can choose to exert costly effort to improve the accuracy of his observation. We introduce the effort

[1] Note with redundant assignments, we may have $\mathbf{x}_i = \mathbf{x}_j$, $i \neq j$, $i, j = 1, \ldots, T$.

insensitive model below. Our analysis in the next few sections will focus on this model. Then, we will introduce the effort sensitive model and extend our main results to it in Section 8.

Effort insensitive worker observation model: Each worker observes a signal (which is the worker's observed label) once a task is assigned. Denote worker i's observation for data x_i as $\tilde{y}_i$. Worker observations are *conditionally independent* (on y_i) and follow a two-coin flipping-error model: $\Pr(\tilde{y}_i = -1 | y_i = +1) = p_+$ and $\Pr(\tilde{y}_i = +1 | y_i = -1) = p_-$, where p_+ and p_- satisfy:

(i) $p_+ + p_- < 1$, that is, the sum of the error rates for the two classes is less than 1.

This assumption requires that a worker's observation is informative with respect to the true label in the sense that the probability for the worker to observe label s is higher when the true label is s than when the true label is $-s$, i.e. $\Pr(\tilde{y}_i = s | y_i = s) > \Pr(\tilde{y}_i = s | y_i = -s), \forall s \in \{-1, +1\}$. It is a necessary and sufficient condition so that Bayesian updating increases one's belief that the observed label is the true label. This is formally stated in Lemma 2.1.

Lemma 2.1. $\Pr(y_i = s | \tilde{y}_i = s) > \mathcal{P}_s, \forall s \in \{-1, +1\}$, *if and only if* $p_+ + p_- < 1$.

Similar assumptions are made for agent belief models in the peer prediction literature [4, 22].

(ii) $p_-, p_+ \geq \kappa > 0$ where $0 < \kappa < 1/2$. That is, workers do not have perfect information.

We assume that the error models are common knowledge to both the workers and the principal. The prior on true labels, $\mathcal{P}_+$ and $\mathcal{P}_-$, are common knowledge too. But the principal and the workers do not necessarily know the values of p_+ and p_-. Most of our discussion in this paper focuses on homogeneous workers that have the same p_+ and p_-, but we discuss the possibility of extending to the case with heterogeneous workers in the Appendix.

After observing the signals, each worker i decides on which label to report. A pure reporting strategy of an agent is a mapping from the agent's observation to a report: $r_i(\tilde{y}_i) : \{-1, +1\} \to \{-1, +1\}$. In the case of mix strategies, the mapping is to a distribution on the two possible reports $\{-1, +1\}$. For example, a truthful reporting strategy corresponds to the case that an agent truthfully reports his observation, and an uninformative strategy, pure or mixed, has the same report distribution for all signals. We then define $\hat{p}_{i,+} := \Pr(\hat{y}_i = -1 | y_i = +1, r_i(\tilde{y}_i)) = \sum_{s \in \{-1,+1\}} \Pr(\tilde{y}_i = s | y_i = +1) \Pr(r_i(\tilde{y}_i) = -1 | \tilde{y}_i = s)$ and $\hat{p}_{i,-} := \Pr(\hat{y}_i = +1 | y_i = -1, r_i(\tilde{y}_i)) = \sum_{s \in \{-1,+1\}} \Pr(\tilde{y}_i = s | y_i = -1) \Pr(r_i(\tilde{y}_i) = +1 | \tilde{y}_i = s)$ as the flipping error rates of agent i's report when he plays strategy $r_i(\tilde{y}_i)$. Workers can be incentivized via payment, and they would like to maximize their expected payment.

2.2 The principal's design objective

The principal wants to design a reward mechanism $\mathcal{M}$ to incentivize agents to contribute high-quality labels. Under the effort insensitive model, this means that the principal would like the agents to truthfully report their observed labels. Under the effort sensitive model, the principal wants agents to exert effort to obtain more accurate labels and then truthfully report them. Since the principal cannot directly verify the contributed labels, the mechanism can only reward agents based on the data and the reported labels collected from the T agents, i.e. $\{\mathbf{x}_i, \hat{y}_i\}_{i=1}^{T}$. The principal would like to achieve the above high-quality elicitation at a Bayesian Nash equilibrium (BNE) and ideally remove other BNE where the elicited labels are of lower quality. In this paper, we focus on symmetric BNEs where agents' strategies $r_i(\tilde{y}_i)$ are the same.

3 OUR APPROACH

In this section we first describe our approach toward designing the principal's mechanism. Then, we briefly introduce a set of machine learning techniques, learning with "noisy data" (when the rates of flipping errors are known), that our approach is built upon.

3.1 Design of scoring function and reference answer using Machine Learning

In designing a mechanism $\mathcal{M}$, the principal needs to choose a scoring function $S((\mathbf{x}_i, \hat{y}_i), \{(\mathbf{x}_j, \hat{y}_j)\}_{j \neq i})$ to determine the reward for agent i. Let $\mathcal{K}_{-i}$ denotes $\{(\mathbf{x}_j, \hat{y}_j)\}_{j \neq i}$. In this paper, we consider a smaller design space where the mechanism's scoring function for each agent i is of the form:

$$S(\hat{y}_i, f^{\mathcal{A}}_{\mathcal{K}_{-i}}(\mathbf{x}_i)) : \mathbb{R}^2 \to \mathbb{R}. \tag{1}$$

where $f^{\mathcal{A}}_{\mathcal{K}_{-i}}(\cdot) : \mathbb{R}^M \to \mathbb{R}$ is a function obtained by training algorithm $\mathcal{A}$ on dataset $\mathcal{K}_{-i}$, that is, $f^{\mathcal{A}}_{\mathcal{K}_{-i}}(\cdot) = \mathcal{A}(\mathcal{K}_{-i})$. $\mathcal{F}$ is the concept class that algorithm $\mathcal{A}$ maps into and hence $f^{\mathcal{A}}_{\mathcal{K}_{-i}}(\cdot) \in \mathcal{F}$. In other words, we consider mechanisms that reward a worker based on how his reported label compares to that of a prediction of a learning algorithm, where the algorithm is trained on the data and the reported labels of other agents. Hence, a mechanism $\mathcal{M}$ in this paper is a tuple $(S, \mathcal{A})$.

As mentioned earlier, the principal would like to obtain high-quality labels at a BNE. Denote $\mathcal{P}(\{(\mathbf{x}_j, \tilde{y}_j)\}_{j=1}^T | (\mathbf{x}_i, \tilde{y}_i))$ as agent i's belief about the realized labels of all agents after agent i has observed his own label. Then a mechanism $\mathcal{M} = (S, \mathcal{A})$ induces a (strictly) truthful BNE if

$$\mathbb{E}_{\mathcal{P}(\{(\mathbf{x}_j, \tilde{y}_j)\}_{j=1}^T | (\mathbf{x}_i, \tilde{y}_i))}[S(\tilde{y}_i, f^{\mathcal{A}}_{\mathcal{K}_{-i}}(\mathbf{x}_i))] > \mathbb{E}_{\mathcal{P}(\{(\mathbf{x}_j, \tilde{y}_j)\}_{j=1}^T | (\mathbf{x}_i, \tilde{y}_i))}[S(\hat{y}_i, f^{\mathcal{A}}_{\mathcal{K}_{-i}}(\mathbf{x}_i))] \tag{2}$$

for all i, $\tilde{y}_i$ and $\hat{y}_i \neq \tilde{y}_i$.

In the rest of the paper, we'll write $S(\hat{y}_i, f^{\mathcal{A}}_{\mathcal{K}_{-i}}(\mathbf{x}_i))$ as $S(\hat{y}_i, f(\mathbf{x}_i))$ for notation simplicity, but a reader should understand the dependency of f on algorithm $\mathcal{A}$ and training dataset $\mathcal{K}_{-i}$.

The scoring function (1) resembles those designed in the peer prediction literature for the IEWV problem in the sense that it scores an agent's report against a benchmark that is based on other agents' reports. But it has some fundamental differences. Existing peer prediction mechanisms use scoring functions that are completely defined over agents' reports: $S(\hat{y}_i, \{\hat{y}_j\}_{j \neq i}) : \mathbb{R}^T \to \mathbb{R}$. The second argument of the scoring function for many peer prediction mechanisms is simply the report from a random peer agent for the *same* task. In some more recent mechanisms, this second argument includes agents' reports on other tasks but still needs to contain reports from peer agent for the same tasks [4, 22]. In contrast, the scoring function in (1) incorporates the feature vectors $\mathbf{x}_i$'s. It doesn't require that we have to have redundant reports for the same task (except for a small number of selected re-assigned tasks). Instead, we seek to learn a machine predictor f from a heterogeneous set of data to generate the second argument in the scoring function S.

Our approach toward designing a mechanism to induce a truth-telling BNE takes two-steps. First, assuming agents reports truthfully, we focus on designing an algorithm $\mathcal{A}$ such that, although it is trained on data with flipping errors, it outputs a function f that converges to the optimal classifier $f^* \in \mathcal{F}$ as if it is trained on data without flipping errors. The algorithm $\mathcal{A}$ "counters" the noise in the reported data. This guarantees that if everyone else report truthfully, the reference report $f(\mathbf{x}_i)$ for agent i is an accurate prediction of the true label y_i when the number of data is large. Second, we design the scoring function S such that truth-telling is the best action for agent i if the agent knows that his reference report is an accurate prediction of the true label y_i. Now, the agent's observed label has noise as a result of the flipping errors, but the reference report does not in expectation. When designing the scoring function, we engage a similar "counter-noise" technique as the machine learning algorithm does in the previous step. These two steps together ensure that truth-telling is a BNE in the resultant mechanism $(S, \mathcal{A})$.

At the technical level, our approach is built on a set of machine learning techniques, often referred as learning with "noisy data" [16]. If flipping errors $\hat{p}_+$ and $\hat{p}_-$ of training data are *known* to the principal, this set of techniques can learn an f that converges to the optimal classifier f^* in classification risk from the noisy training data. In our setting, however, the principal does not know the flipping-error rates in the reported data due to lack of knowledge of both p_+ and p_- and agents' strategies. This poses additional challenge of finding such an f. We will defer the details on learning the flipping-error rates to Section 5.

3.2 Learning with noisy labels

We now introduce the techniques for learning with noisy lables, which our mechanisms will build upon. Though there are clear differences between different methods for learning with noisy labels, the basic idea tends to build on finding surrogate loss functions that can compensate the bias in training data, when evaluating a function $f \in \mathcal{F}$. We briefly survey a particular algorithm and its results from Natarajan et al. [2013], in hope of delivering the idea on how learning can be done in face of biased data.

Preliminaries: Denote the 0-1 loss of f as $R_{\mathcal{D}}(f) = \mathbb{E}_{(\mathbf{x},y)\sim\mathcal{D}}[\mathbb{1}(f(\mathbf{x}) \neq y)]$, and a general l loss as $R_{l,\mathcal{D}}(f) = \mathbb{E}_{(\mathbf{x},y)\sim\mathcal{D}}[l(f(\mathbf{x}), y)]$. Denote the minimum risk over the concept class as $R^* := \min_{f\in\mathcal{F}} R_{\mathcal{D}}(f)$ and $R^*_l := \min_{f\in\mathcal{F}} R_{l,\mathcal{D}}(f)$ respectively. We further denote $f^* := \text{argmin}_{f\in\mathcal{F}} R_{\mathcal{D}}(f)$ and $f^*_l := \text{argmin}_{f\in\mathcal{F}} R_{l,\mathcal{D}}(f)$.

The setup of Natarajan et al. [2013] assumes that the flipping-errors of the reports are the same across all agents, that is $\hat{p}_+ := \hat{p}_{1,+} = \ldots = \hat{p}_{N,+}$ and $\hat{p}_- := \hat{p}_{1,-} = \ldots = \hat{p}_{N,-}$, and $\hat{p}_+ + \hat{p}_- < 1$. Furthermore, the principal *knows* the values of $\hat{p}_+$ and $\hat{p}_-$. (This can be viewed as a special case of our setting where the principal knows p_+ and p_- and every agent truthfully reports his observed signal.) Suppose the principal has collected N i.i.d. training samples $\{(\mathbf{x}_j, \hat{y}_j)\}_{j=1}^N$, indexed from 1 to N. The naive approach of directly minimizing empirical 0-1 loss over reported data not only is technically difficult but also would give a biased classifier that may not converge to the optimal classifier. Natarajan et al. [2013] tackle this problem by first finding a convex and *classification-calibrated* (CC) loss function $l(t, y)$ (with prediction t and label y as the inputs), where CC is defined as follows:

Definition 3.1 (Classification-Calibrated (CC)). l is CC if $\exists$ a convex, invertible, nondecreasing transformation ϕ_l with $\phi_l(0) = 0$ s.t. $\phi_l(R_{\mathcal{D}}(\tilde{f}) - R^*) \leq R_{l,\mathcal{D}}(\tilde{f}) - \min_f R_{l,\mathcal{D}}(f)$.

The introduce of a convex loss function l helps to remove the computational challenge in minimizing 0-1 loss directly. Classification-calibration helps us control the 0-1 loss via controlling the l-loss, i.e., if we find a classifier that converges to the optimal one in l-loss, its 0-1 loss also converges. It is generally possible to assume the existence of such an l [1, 16]. Particularly it is shown in by Bartlett et al. [2006] that, if the loss function can be written as $l(t, y) = \psi(yt)$ where ψ is convex and differentiable at 0 with $\psi'(0) < 0$, then l is classification-calibrated. Following above assumptions, we assume such an l exists for now.

Natarajan et al. [2013] then further define an *"un-biased" surrogate loss functions* over l to help "remove" noise, when $\hat{p}_+ + \hat{p}_- < 1$:

$$\varphi(t, y) := \frac{(1 - \hat{p}_{\text{sgn}(-y)})l(t, y) - \hat{p}_{\text{sgn}(y)} l(t, -y)}{1 - \hat{p}_+ - \hat{p}_-}, \tag{3}$$

where sgn$(\cdot)$ is a sign function that $\text{sgn}(+1) = +, \text{sgn}(-1) = -$. The surrogate loss φ is defined such that when a prediction is evaluated against a noisy label using this surrogate loss function, the prediction is as if evaluated against the ground-truth label using l in expectation. Hence the loss of the prediction is "unbiased" in expectation. This is formally stated in Lemma 3.2 below.

Lemma 3.2 (Lemma 1 [16]). *$\forall$ prediction t, $\mathbb{E}_{\tilde{y}}[\varphi(t,\tilde{y})] = l(t,y)$, where $\tilde{y}$ is the report from non-strategic workers with known flipping errors, and y is the ground-truth label.*

Natarajan et al. [2013] proceeds to find a classifier via minimizing the empirical risk w.r.t. $\varphi(\cdot)$:

$$\tilde{f}^*_\varphi = \text{argmin}_{f \in \mathcal{F}} \hat{R}_\varphi(f) := \frac{1}{N} \sum_{j=1}^{N} \varphi(f(\mathbf{x}_j), \hat{y}_j). \tag{4}$$

The performance guarantee for $\tilde{f}^*_\varphi$ is given by the following lemma.

Lemma 3.3 (Theorem 3 [16]). *With probability at least $1-\delta$,*

$$R_{l,\mathcal{D}}(\tilde{f}^*_\varphi) \le \min_{f \in \mathcal{F}} R_{l,\mathcal{D}}(f) + 4L_p \mathfrak{R}(\mathcal{F}) + \sqrt{\frac{\log 1/\delta}{2N}}, \tag{5}$$

where $L_p \le 2L/(1-\hat{p}_+ - \hat{p}_-)$. $\mathfrak{R}(\mathcal{F})$ is the Rademacher complexity of function class $\mathcal{F}$.

Sine the VC dimension of the concept class $\mathcal{F}$ is finite [23], we have a converging $\mathfrak{R}(\mathcal{F}) = O\left(\sqrt{\log \mathtt{VCDim}(\mathcal{F})/N}\right)$. Also as l is classification calibrated, this lemma guarantees that the 0-1 risk of $\tilde{f}^*_\varphi$ converges to that of R^*. It also implies that the smaller $\hat{p}_+, \hat{p}_-$ are, the faster the error convergence is.

Other successful examples in defining surrogate loss functions for dealing with biased training data include *label dependent cost surrogate loss* [16, 20]. The idea is to scale up the loss function differently for different labels; e.g., the following one was also studied by Natarajan et al. [2013]:

$$\varphi(t,y) := (1-\alpha)\mathbb{1}(y=+1) \cdot l(t,y) + \alpha \mathbb{1}(y=-1) \cdot l(t,y), \ \alpha = \frac{1-\hat{p}_+ + \hat{p}_-}{2}. \tag{6}$$

4 MECHANISM AND RESULTS FOR ELICITING TRUTHFUL REPORTS

In this section, we will establish that a class of surrogate loss functions can both serve to train a classifier to provide an accurate reference answer and be used as peer prediction scoring functions for eliciting truthful reports. Then we show that our mechanism no longer admit the undesirable permutation equilibrium.

To better deliver the intuition of our mechanism, we will present our mechanism and results assuming everything is "learnable" in this section. Particularly we will assume that the principal can estimate the flipping error rates of the reported data and the optimal classifier to arbitrary accuracy. While this is obviously too strong to assume without investigation, this assumption allows us to focus on establishing the equilibrium results of our mechanism, given that we have access to such accurate estimates. Later in Section 5, we will show how to obtain such accurate estimates from the reported data if agents are playing some symmetric strategies.

4.1 When a distribution is elicitable via Machine Learning?

While we seek to use machine learning (ML) to help data elicitation, asking ML methods to help eliciting arbitrarily distributed data is likely over demanding. After all, we are hoping that the learned structural relationship between $\mathbf{x}$ and y can provide a good prediction of y for any given $\mathbf{x}$ and this prediction is then used as a reference answer for scoring a reported label for this $\mathbf{x}$. If the structural relationship cannot be learned, we shouldn't hope that this approach can help us to achieve better elicitation. We formalize this idea as *ML elicitability*:

Definition 4.1 (ML elicitability). $(\mathbf{x},y) \sim \mathcal{D}$ is *ML elicitable*, if there exists a mechanism $\mathcal{M} = (S,\mathcal{A})$ that satisfies Eqn. (2), that is, $\mathcal{M}$ induces a (strictly) truthful BNE for workers who have been assigned tasks drawn from $\mathcal{D}$.

ML elicitability intends to capture when the structural relationship between $\mathbf{x}$ and y is "learnable" so that a machine prediction can be used in peer prediction scoring to induce the truthful reporting BNE. Classic peer prediction has an analogous concept: most existing peer prediction mechanisms admit the truthful BNE only when signals of agents are *stochastic relevant* [15, 29], a condition that essentially captures when the correlation of agents' signals are strong enough so that peer reports can be used in scoring to induce the truthful reporting BNE.

We now introduce a set of *ML elicitability conditions*. We will design mechanisms in Section 4.2 that induce the truthful reporting BNE when these conditions are satisfied.

ML elicitability conditions:

(1) There exists a concept class $\mathcal{F}$, with each decision function $f \in \mathcal{F}$ mapping a feature vector $\mathbf{x} \in \mathcal{X}$ to a prediction on label y (i.e. $f : \mathcal{X} \rightarrow \{-1, +1\}$), that has a bounded Vapnik-Chervonenkis (VC) dimension [23], $\texttt{VCDim}(\mathcal{F}) < \infty$.
(2) Denote $f^* := \min_{f \in \mathcal{F}} \mathbb{E}_{(\mathbf{x},y)\sim\mathcal{D}}[\mathbb{1}(f(\mathbf{x}) \neq y)]$. Then $\mathbb{E}_{(\mathbf{x},y)\sim\mathcal{D}}[\mathbb{1}(f^*(\mathbf{x}) \neq y)] < 0.5$.
(3) $\mathbb{E}_{(\mathbf{x},y)\sim\mathcal{D}|y}[\mathbb{1}(f^*(\mathbf{x}) \neq y)] \leq 0.5$ for $y \in \{-1, +1\}$.

From a learning perspective, these ML elicitability conditions are rather weak learnability conditions. All they require is that there exists a f^* that comes from a concept class with bounded VC dimension (learnable in finite samples), and it can separate the data from one class to another with its 0-1 prediction performance being strictly better than random guess. Condition (3) states that the optimal classifier f^* does not perform worse than random guess on both classes' conditional distributions. Throughout this paper, we assume for the data distribution of interest, the principal is aware of such an $\mathcal{F}$ (but not f^*). Denote its VC dimension as $d := \texttt{VCDim}(\mathcal{F})$.

4.2 Main Mechanism

With the above preparation, we now construct a mechanism that induces the truthful BNE when the ML elicitability conditions are satisfied.

```
MECHANISM 1: ML Prediction (MLP)
For each worker $i$:
  1. Assign tasks, and estimate flipping errors $\tilde{p}_{-i,+}, \tilde{p}_{-i,-}$ based on reported data from workers $j \neq i$
     (Mechanism 2).
       – When returned solution satisfies $\tilde{p}_{-i,+} + \tilde{p}_{-i,-} \neq 1$, continue.
       – Otherwise trigger exception handler (Mechanism 4), and stop.
  2. Define $\tilde{\varphi}(\cdot)$ using $(\tilde{p}_{-i,+}, \tilde{p}_{-i,-})$. Train a classifier $\tilde{f}^*_{\tilde{\varphi},-i}$:
     $\tilde{f}^*_{\tilde{\varphi},-i} = \text{argmin}_{f \in \mathcal{F}} \frac{1}{N-1} \sum_{j \in \mathcal{U}\setminus\{i\}} \tilde{\varphi}(f(\mathbf{x}_j), \tilde{y}_j)$.
  3. If $\tilde{p}_{-i,+} + \tilde{p}_{-i,-} > 1$, flip them: $\tilde{p}_{-i,+} := 1 - \tilde{p}_{-i,+}, \tilde{p}_{-i,-} := 1 - \tilde{p}_{-i,-}$. Define $\tilde{\varphi}_{\text{unbias}}(\cdot)$ using
     $(\tilde{p}_{-i,+}, \tilde{p}_{-i,-})$.
  4. Score each worker $i$ using $S(\hat{y}_i, \tilde{f}^*_{\tilde{\varphi},-i}) := -\tilde{\varphi}_{\text{unbias}}(\tilde{f}^*_{\tilde{\varphi},-i}, \hat{y}_i)$ (or any affine transformation of it).
```

The basic recipes of the mechanism are as follows. First, for each agent i, assuming that other agents play a symmetric strategy, the principal estimates the flipping-error rates of other agents' reports (denote the estimates as $\tilde{p}_{-i,+}$ and $\tilde{p}_{-i,-}$). Then, an estimated surrogate loss function $\tilde{\varphi}(\cdot)$ (any surrogate loss function that can learn from noisy data, e.g., the ones in Eqn.(3) and (6)) defined using $\tilde{p}_{-i,+}$ and $\tilde{p}_{-i,-}$ allows the principal to train a classifier $\tilde{f}^*_{\tilde{\varphi},-i}$, with its prediction performance converging to the optimal one. Then we use this classifier to make prediction on worker i's assigned data, to serve as a "machine prediction" instead of "peer prediction". Finally, we construct a reward function for each agent i using surrogate loss function $\tilde{\varphi}_{\text{unbias}}$:

$$\tilde{\varphi}_{\text{unbias}}(t,y) := \frac{(1-\tilde{p}_{-i,\text{sgn}(-y)})l(t,y) - \tilde{p}_{-i,\text{sgn}(y)}l(t,-y)}{1-\tilde{p}_{-i,+}-\tilde{p}_{-i,-}},$$

as defined in Eqn. (3), with $(\hat{y}_i, \tilde{f}^*_{\tilde{\varphi}}(\mathbf{x}_i))$ being its inputs.

When $\tilde{p}_{-i,+} + \tilde{p}_{-i,-} = 1$, the assumptions made in existing learning with noisy data methods will be violated, in that both the surrogate loss function and its training steps are not well defined. We handle this exception case in Section 6.

We'd like to focus on the equilibrium analysis of our proposed mechanism in this section, thus for now, we assume that the principal can accurately learn the error rates in the reported data if the reported data are generated from either a symmetric truthful reporting strategy or a symmetric permutation strategy, and hence the principal can approximate the surrogate loss functions defined using these error rates and the optimal classifier well.

Assumption 1 (Informal). *When either all agents $j \neq i$ play the truthful reporting strategy or they all play a permutation strategy, the principal can obtain arbitrarily accurate estimates $\tilde{p}_{-i,+}$,$\tilde{p}_{-i,-}$, $\tilde{\varphi}$, and $\tilde{f}^*_{\tilde{\varphi},-i}$. That is, there exist positive constants $\epsilon, \epsilon_1, \delta_1, \epsilon_2$, and δ_2 s.t.*

1. $|\tilde{p}_{-i,+} - \hat{p}_{-i,+}| \le \epsilon$, $|\tilde{p}_{-i,-} - \hat{p}_{-i,-}| \le \epsilon$.
2. *For the surrogate loss function defined using $\tilde{p}_{-i,+}$,$\tilde{p}_{-i,-1}$, with probability at least $1-\delta_1$, $|\tilde{\varphi}(t,y) - \varphi(t,y)| \le \epsilon_1$, $\forall t, y$.*
3. *With probability at least $1-\delta_2$, $R_{\mathcal{D}}(\tilde{f}^*_{\tilde{\varphi},-i}) - R^* \le \epsilon_2$.*

All terms $\epsilon, \epsilon_1, \delta_1, \epsilon_2$, and δ_2 can be made arbitrarily small with increasing number of samples N.

This assumption is made entirely for the sake of presentation. Particularly, Assumption 1 will be used in place of Steps 1 and 2 of Mechanism 1. Later in Section 5, we will show how to estimate the error rates of the reported data to satisfy this assumption.

Assuming workers have perfect knowledge of the mechanism, and know that ML elicitability conditions are satisfied. Our next theorem shows that $S(\cdot) := -\tilde{\varphi}_{\text{unbias}}(\cdot)$, along with the Assumption 1 induces strictly truthful BNE.

Theorem 4.2. *Suppose that ML elicitability conditions are satisfied. Under Assumption 1, when the error terms in Assumption 1 approach 0 (small enough, decreasing as functions of N), (*`MLP`*) induces strictly truthful BNE for all workers.*

Intuition: Proving above theorem hinges on the following facts: first when agents truthfully report, we know their score is going to converge to $-\mathbb{E}[l(f^*(\mathbf{x}), y)]$, negative of the minimum l-loss, due to the fact that the surrogate loss function calibrates the noise in agent's report. Then we show that deviating to other strategies will introduce $l(f^*(\mathbf{x}), -y)$ term into the expected score. This leads to higher surrogate loss, due to the fact that $l(f^*(\mathbf{x}), -y) = l(-f^*(\mathbf{x}), y)$ and informativeness of f^* w.r.t. the ground-truth label y. For easiness of presentation we will omit the unbias subscript in φ, and when we use φ we meant the un-bias surrogate loss function defined in Eqn. (3). Also we will short hand $f^*(\mathbf{x})$ as f^*.

Proof. (Sketch) Consider agent i. Denote the reporting strategy as follows: $\hat{y}_i = \tilde{y}_i \cdot (-1)^{r(\tilde{y}_i)}, r(\tilde{y}_i) \in \{0,1\}$, i.e., based on the observation $\tilde{y}_i$, the agent can decide whether to truthfully report ($r(\tilde{y}_i) = 0$), or revert the answer ($r(\tilde{y}_i) = 1$). All together the agent has four possible pure strategies: $\{-1,+1\} \times \{0,1\}$. It is straightforward to argue that any other strategy can be written as a linear combination of the four, i.e., these four strategies form the basis of the strategy space. Therefore we only need to check that truth telling is strictly proper among the four strategies. [2] First note $\mathbb{E}[S(\tilde{y}_i(-1)^{\tilde{y}_i}, \tilde{f}_{\tilde{\varphi},-i}(\mathbf{x}_i))] \to -\mathbb{E}[p_{\mathrm{sgn}(y_i)}\varphi(f^*(\mathbf{x}_i), y_i(-1)^{r(y_i)}) + (1-p_{\mathrm{sgn}(y_i)})\varphi(f^*(\mathbf{x}_i), y_i(-1)^{r(-y_i)}].$

The above convergence is due to Assumption 1, as well as the fact that we consider the case agents $j \neq i$ are truthfully reporting, so that $\hat{p}_{-i,+} = p_+, \hat{p}_{-i,-} = p_-$. We will then reason with this clean case with (φ, f^*), as we can always bring down the approximation error via increasing N.

By truthfully reporting $r(\tilde{y}_i) \equiv 0$, agent i's expected utility is:

$$\mathbb{E}_{(\mathbf{x},y)\sim\mathcal{D}}[S(\tilde{y}_i, \tilde{f}_{\tilde{\varphi},-i}(\mathbf{x}_i))] \to -\mathbb{E}[l(f^*(\mathbf{x}), y)].$$

Then we first prove the following fact: $R_{l,\mathcal{D}}(f^*) - R_{l,\mathcal{D}}(-f^*) < 0$, which is essentially saying f^* performs strictly better than $-f^*$ in l-loss. This will help us prove non-profitability for the case that $r(\tilde{y}_i) \equiv 1$, that is the agent always reverts the answer. In this case we have (replacing $(\mathbf{x}_i, y_i)$ pair with $(\mathbf{x}, y)$ in notation)

$$\mathbb{E}[\varphi(f^*, -\tilde{y}_i)] = \mathbb{E}[p_{\mathrm{sgn}(y)}\varphi(f^*, y) + (1-p_{\mathrm{sgn}(y)})\varphi(f^*, y)], \tag{7}$$

and we prove that conditional on each label $y \in \{-1, +\}$

$$\mathbb{E}_{\cdot|y=+1}[p_{\mathrm{sgn}(y)}\varphi(f^*, y) + (1-p_{\mathrm{sgn}(y)})\varphi(f^*, -y)] \geq \mathbb{E}_{\cdot|y=+1} l(f^*, +1), \tag{8}$$
$$\mathbb{E}_{\cdot|y=-1}[p_{\mathrm{sgn}(y)}\varphi(f^*, y) + (1-p_{\mathrm{sgn}(y)})\varphi(f^*, -y)] \geq \mathbb{E}_{\cdot|y=-1} l(f^*, -1), \tag{9}$$

and thus $\mathbb{E}[\varphi(f^*, -\tilde{y})] \geq \mathbb{E}[l(f^*, y)]$. Yet the two inequalities in Eqn. (8) and (9) cannot hold simultaneously, as otherwise we will have $\mathbb{E}[l(f^*, y)] = \mathbb{E}[l(-f^*, y)]$, contradicting the fact $R_{l,\mathcal{D}}(f^*) - R_{l,\mathcal{D}}(-f^*) < 0$. Thus we have proved that reverting is strictly dominated by truthfully reporting.

Consider the case that agents only revert one observation: denote $\alpha_+ := \frac{1-p_-}{1-p_+-p_-}, \alpha_- := \frac{1-p_+}{1-p_+-p_-}$.

- When $r(+1) = 1, r(-1) = 0$: $\mathbb{E}[\varphi(f^*, \tilde{y}_i \cdot (-1)^{r(\tilde{y}_i)})] = \alpha_-\mathbb{E}[l(f^*, -1)] + (1-\alpha_-)\mathbb{E}[l(f^*, +1)]$
- When $r(+1) = 0, r(-1) = 1$: $\mathbb{E}[\varphi(f^*, \tilde{y}_i \cdot (-1)^{r(\tilde{y}_i)})] = \alpha_+\mathbb{E}[l(f^*, +1)] + (1-\alpha_+)\mathbb{E}[l(f^*, -1)]$

We will then prove that

$$\begin{aligned}\mathbb{E}[\varphi(f^*, -\tilde{y}_i)] > \max\{&\alpha_-\mathbb{E}[l(f^*, -1)] + (1-\alpha_-)\mathbb{E}[l(f^*, +1)],\\ &\alpha_+\mathbb{E}[l(f^*, +1)] + (1-\alpha_+)\mathbb{E}[l(f^*, -1)]\}.\end{aligned} \tag{10}$$

Since the sum of the expected surrogate loss for cases of reporting $\tilde{y}, -\tilde{y}$ is the same as the sum for the other two reporting strategies that only revert "half" of the observations, we have

$$\begin{aligned}&\mathbb{E}[\varphi(f^*, -\tilde{y}_i)] + \mathbb{E}[\varphi(f^*, \tilde{y}_i)] = (\alpha_+\mathbb{E}[l(f^*, +1)]\\ &+ (1-\alpha_+)\mathbb{E}[l(f^*, -1)]) + (\alpha_-\mathbb{E}[l(f^*, -1)] + (1-\alpha_-)\mathbb{E}[l(f^*, +1)]).\end{aligned} \tag{11}$$

Since $\mathbb{E}[\varphi(f^*, -\tilde{y}_i)] > \mathbb{E}[\varphi(f^*, \tilde{y}_i)]$ and the fact that Eqn.(10) holds, we must have both terms on the RHS of Eqn. (11) being larger than $\mathbb{E}[\varphi(f^*, \tilde{y}_i)]$ – so their negatives (reward) are smaller than truthful reporting. □

We can also similarly show that, (MLP) doesn't admit a permutation equilibrium.

Theorem 4.3. *Suppose that ML elicitability conditions are satisfied. Under Assumption 1, when the error terms in Assumption 1 approach 0 (small enough, decreasing as functions of N), everyone playing a fixed permutation strategy is not a BNE in (*MLP*).*

[2] Similar argument can be found in [4]. For completeness we give details in the full version.

Almost all existing peer prediction mechanisms[4, 13, 22, 26] have a permutation BNE. Intuitively, when workers reach an agreement on permuting their reports according to a fixed order, it's difficult for the the principal to tell the day from night. While Assumption 1 assumes this difficulty away, in Section 5, we'll show that we can still correctly estimate the error rates of the reported data (hence can tell the day from night) even if agents all permute their reports. It may be noticed that in this case $\tilde{p}_{-i,+} + \tilde{p}_{-i,-} > 1$, which violates the assumption for surrogate loss function based learning we introduced in Section 3.2. However we will show that plugging this pair of true error rates into the surrogate loss functions will help revert the surrogate loss function back to the one as if workers are truthfully reporting (Eqn. 16). So in training, it is as if we are reverting the label back. Thus the learning part will still go through.

In practice, how to select the number of samples so to make the error terms in Assumption 1 small enough depends on how faster the trained classifier converges, which can be quantified using our analysis in Section 5. Often the large enough quantity is a function of the following parameters: $(d, 1/2 - R^*, \delta_p, \Delta)$, which control the difficulties of learning.

It may be noticed that in order to run Mechanism 1, we need to train a classifier for each worker to provide a reference answer, which raises computational concerns. Practically we only need to train two classifiers. Consider separating the group of workers into two groups with equal sizes, completely randomly and uniformly; let's call these two groups as G_A and G_B. Each of the group has size $|G_A|, |G_B| \geq \lfloor N/2 \rfloor$. Then train classifiers $\tilde{f}^*_{\tilde{\varphi},-G_A}, \tilde{f}^*_{\tilde{\varphi},-G_B}$ using data from group G_B and G_A respectively. For worker i in group G_A, reward him using $S(\hat{y}_i, \tilde{f}^*_{\tilde{\varphi},-G_A}(\mathbf{x}_i))$. Similarly we will reward a worker from group G_B using $\tilde{f}^*_{\tilde{\varphi},-G_B}$. This significantly reduces the effort for re-training.

5 LEARNABILITY OF ERROR RATES, AND SURROGATE LOSS FUNCTIONS

In our setting, the principle doesn't know the flipping-error rates in agents' reported data. In this section we discuss the challenges for the learning due to unknown flipping errors and how we address them. We note that many, if not all, of the surrogate loss function based methods that we are aware of for solving the learning with noisy labels problem, e.g. the ones defined in Eqn.(3) and (6), require the knowledge of the error rates of the noisy input data. In practice, however, this is hardly the case. In this section, we discuss the concept of learnability of $\hat{p}_{-i,+}$,$\hat{p}_{-i,-}$ for each agent i, and further the surrogate loss. We'll show that our learning process satisfies Assumption 1 when all other agents play either the truthful or a permutation strategy.

We assume that the principal knows the lower bounds of the following quantities: $\Delta := 1/2 - R^*$ and $\delta_p := 1 - (p_+ + p_-)$. The principal would need such knowledge to decide the number of training samples N to be elicited such that the learning of the error rates converges.

First we demonstrate how to estimate $\hat{p}_{-i,+}$,$\hat{p}_{-i,-}$, and prove a couple of consistency results under noisy labels. Assuming agents play a symmetric strategy, our method starts with estimating the following two quantities that can be computed from the collected/reported data directly:

(i) Matching probability: this is the probability when a task is assigned to two different workers, the chance of the outputs match each other. Denote this quantity as q. In order to evaluate the matching on the same task, we need to re-assign the tasks. First we separate the tasks into two sets: we will randomly select K (to ensure enough samples for an accurate estimation) tasks to re-assign to one more worker [3]. Denote the set of re-assigned tasks as $\mathcal{U}^r$. Suppose $T = K + N$. Estimate for

[3] When N is large enough, taking $K = \mathrm{O}\big(\log N\big)$ often suffices. Also note, if the principal knows p_+ and p_-, we don't even need to obtain redundant labels.

worker $j \neq i$:

$$\tilde{q}_{-i} := \frac{1}{|\mathcal{U}^r \backslash \{i\}|} \sum_{n \in \mathcal{U}^r \backslash \{i\}} \mathbb{1}(\text{there is a match on task } n). \tag{12}$$

(ii) Fraction of +1/-1 labels that we observe from workers' contribution P_+, P_-; denote $\tilde{P}_+^{-i}, \tilde{P}_-^{-i}$ as the corresponding estimations for workers $j \neq i$:

$$\tilde{P}_-^{-i} := \frac{1}{|\mathcal{U} \backslash \{i\}|} \sum_{n \in \mathcal{U} \backslash \{i\}} \mathbb{1}(\text{task } n \text{ has label -1}). \tag{13}$$

MECHANISM 2: Estimation of $\hat{p}_+, \hat{p}_-$

1. Assign data; randomly select K of the tasks to re-assign once; estimate $\tilde{q}_{-i}, \tilde{P}_-^{-i}$ as in Eqn. (12) and (13).
2. Solve the following set of equations.

$$\text{(I): } \mathcal{P}_+[\tilde{p}_{-i,+}^2 + (1 - \tilde{p}_{-i,+})^2] + \mathcal{P}_-[\tilde{p}_{-i,-}^2 + (1 - \tilde{p}_{-i,-})^2] = \tilde{q}_{-i},$$

$$\text{(II): } \mathcal{P}_+ \tilde{p}_{-i,+} + \mathcal{P}_-(1 - \tilde{p}_{-i,-}) = \tilde{P}_-^{-i}.$$

Denote the solution(s) as $\tilde{p}_{-i,+}, \tilde{p}_{-i,-}$.

3. When there are more than one solution, call Mechanism 3.
4. Return the selected root, or no solution to Mechanism 1.

We propose Mechanism 2 for learning $\tilde{p}_{-i,+}, \tilde{p}_{-i,-}$. The set of equations set up in Mechanism 2 aims to estimate the error rates in agents' reported data when agents play a symmetric strategy. When workers are arbitrarily reporting, the solution for the system of equations in Mechanism 2 leads to meaningless numbers, or simply there does not exist a solution. When workers are reporting according to symmetric strategies[4], the first equation characterizes the probability of observing a matching signal, while the second one characterizes the fraction of observed negative samples. Note that the second equation is also equivalent with $\mathcal{P}_+(1 - \tilde{p}_{-i,-}) + \mathcal{P}_- \tilde{p}_{-i,-} = \tilde{P}_+^{-i}$, i.e., the equation for probability of observing a positive signal. Readers may also notice that there may exist more than one pair of solutions from the system of equations we have formed ((I) & (II)), due to its quadratic form. Nevertheless the following result holds, which will help remove this ambiguity.

Lemma 5.1. *When there are two pairs of solutions from Mechanism 2, only one pair of them satisfies that* $\tilde{p}_{-i,+} + \tilde{p}_{-i,-} < 1$.

When agents are truthfully reporting, the above estimation algorithm is ready to give us the right flipping error rate if we choose the pair s.t. $\tilde{p}_{-i,+} + \tilde{p}_{-i,-} < 1$ after step 2 and terminate without going into step 3. But this won't give us the right solution if agents play a permutation strategy. Mechanism 3 makes the estimation robust to permutation strategies.

MECHANISM 3: Root selection

When $P_+ - P_- > 0$, return the solution $\tilde{p}_{-i,+}, \tilde{p}_{-i,-}$ s.t.,

$$\{\tilde{p}_{-i,+}, \tilde{p}_{-i,-}\} := \text{argmax}_{\{\tilde{p}_{-i,+}, \tilde{p}_{-i,-}\}} \tilde{p}_{-i,-} / \tilde{p}_{-i,+} \ , \tag{14}$$

When $P_+ - P_- < 0$, return the solution $\tilde{p}_{-i,+}, \tilde{p}_{-i,-}$ s.t.,

$$\{\tilde{p}_{-i,+}, \tilde{p}_{-i,-}\} := \text{argmax}_{\{\tilde{p}_{-i,+}, \tilde{p}_{-i,-}\}} \tilde{p}_{-i,+} / \tilde{p}_{-i,-} \ . \tag{15}$$

The basic implication of Mechanism 3 is that when there is ambiguity in root selection, we select the solution with $\tilde{p}_{-i,-}, \tilde{p}_{-i,+}$ being further away from each other in ratio. Note (1) our solution does not cover the case $P_+ = P_-$. (2) In the above selection, when agents truthfully report, the solution

[4] The estimation approach in the current form works only for symmetric strategies.

satisfying $\tilde{p}_{-i,+} + \tilde{p}_{-i,-} < 1$ will be selected (see proof in our full version). (3) When agents report a permutation of their signals, the solution captures the actual flipping error rates in the reports, i.e. $\tilde{p}_{-i,+}$ approaches to $1 - p_+$ and $\tilde{p}_{-i,-}$ approaches to $1 - p_-$.

We thus have the following consistency results in estimating the error rates in the reported data using Mechanism 2.

Lemma 5.2. *When agents $j \neq i$ are either all truthfully reporting or all reporting permuted signals, and when K, N are large enough, w.p. $\geq 1 - \delta_1(K,N), 1 - \delta_2(K,N)$ respectively: $|\tilde{p}_{-i,+} - \hat{p}_{-i,+}| \leq \epsilon_1(K,N)$, $|\tilde{p}_{-i,-} - \hat{p}_{-i,-}| \leq \epsilon_2(K,N)$. $\delta_1(\cdot), \delta_2(\cdot), \epsilon_1(\cdot), \epsilon_2(\cdot)$ are diminishing in K, N uniformly, s.t. $\forall \gamma > 0$, there exists K, N such that $0 \leq \delta_1(K,N), \delta_2(K,N), \epsilon_1(K,N), \epsilon_2(K,N) \leq \gamma$.*

This can be established based on our estimation of $\tilde{q}_{-i}, \tilde{P}_{-}^{-i}$, followed by perturbation analysis on solving the quadratic equations in Mechanism 2.

Denote by $\tilde{\varphi}(,)$ the noisy surrogate loss function defined using estimated $(\tilde{p}_{-i,+}, \tilde{p}_{-i,-})$s. With above consistency results, we define "learnable surrogate loss function":

Definition 5.3 (Learnable surrogate loss function). A surrogate loss function $\varphi(,)$ is learnable in $\hat{p}_{-i,+}, \hat{p}_{-i,-}$, if for any $\delta, \epsilon > 0$, there exists a (K, N) pair such that with probability at least $1 - \delta$, we have $|\tilde{\varphi}(t,y) - \varphi(t,y)| \leq \epsilon, \forall t, y$.

For example, consider the un-biased surrogate loss function proposed in [16]. Recall in this case we will have

$$\tilde{\varphi}(t,y) := \frac{(1 - \tilde{p}_{-i,\mathrm{sgn}(-y)})l(t,y) - \tilde{p}_{-i,\mathrm{sgn}(y)}l(t,-y)}{1 - \tilde{p}_{-i,+} - \tilde{p}_{-i,-}}.$$

Using Lemma 5.2, we prove the following (following notations in Lemma 5.2)

Lemma 5.4. *When agents $j \neq i$ are either all truthfully reporting or all reporting permuted signals, and when K, N are large enough s.t. $\epsilon_1(K,N) + \epsilon_2(K,N) \leq \delta_p/2.$, with probability at least $1 - \delta_1(K,N) - \delta_2(K,N)$, $|\tilde{\varphi}(t,y) - \varphi(t,y)| \leq \epsilon^{est}(K,N)$, where $\epsilon^{est}(K,N) := 2l^*(4 + \delta_p)\delta_p^{-2}(\epsilon_1(K,N) + \epsilon_2(K,N))$.*

So the φ is indeed learnable for this case. This claim is generally not uncommon for the others, but the analysis is rather ad-hoc. In this paper we focus on surrogate loss functions that are learnable.

With the same set of training data $\{\mathbf{x}_i, \hat{y}_i\}_{i=1}^N$ as we assumed in Section 3.2, denote $\tilde{f}^*_{\tilde{\varphi}} = \mathrm{argmin}_{f \in \mathcal{F}} \hat{R}_{\tilde{\varphi}}(f)$ as the optimal classifier trained using the learned surrogate loss function. Denote the distribution over $(\mathbf{x}, \tilde{y})$ (workers' observed labels) as $\mathcal{D}_\rho$.

When agents truthfully report such that $\hat{p}_+ + \hat{p}_- < 1$, using sample complexity bound, we can prove that with probability at least $1 - \delta$, $\max_{f \in \mathcal{F}} |R_{\varphi, \mathcal{D}_\rho}(f) - \hat{R}_\varphi(f)| \leq \epsilon(\mathfrak{R}(\mathcal{F}), N, \delta, p_+, p_-)$ (proved in [16]), where $\epsilon(\cdot) \to 0$ as N increases. Suppose $R_{l,\mathcal{D}}(f) = c_A R_{\varphi,\mathcal{D}_\rho}(f) + c_B$ for some constants (c_A, c_B): using the surrogate loss function defined in Eqn.(3) we have $c_A = 1$, while for Eqn. (6) we have $c_A = \frac{1 - \hat{p}_+ - \hat{p}_-}{2}$. We assert that this is also true when agents' permute their signal s.t. $\hat{p}_+ = 1 - p_+$, $\hat{p}_- = 1 - p_-$ and $\hat{p}_+ + \hat{p}_- > 1$. Consider the following fact

$$\frac{(1 - \hat{p}_{\mathrm{sgn}(-y)})l(t,y) - \hat{p}_{\mathrm{sgn}(y)}l(t,-y)}{1 - \hat{p}_+ - \hat{p}_-} = \frac{p_{\mathrm{sgn}(-y)}l(t,y) - (1 - p_{\mathrm{sgn}(y)})l(t,-y)}{p_+ + p_- - 1} = \varphi(t,-y), \tag{16}$$

where $\varphi(t,-y)$ denotes the surrogate loss defined over the error rates p_+, p_- when workers truthfully report. So in training, it is as if we are reverting the label back. Therefore the trained classifier will be converging to the optimal one, instead of its opposite.

The following lemma shows that the estimation error in surrogate loss function wouldn't affect the learning results by much (δ, ϵ as in Definition 5.3).

Lemma 5.5. *W.p.* $\geq 1 - 2\delta$ $R_{l,\mathcal{D}}(\tilde{f}^*_{\tilde{\varphi}}) \leq \min_{f \in \mathcal{F}} R_{l,\mathcal{D}}(f) + 2c_A^{-1} \cdot (\epsilon(\mathfrak{R}(\mathcal{F}), N, \delta, p_+, p_-) + \epsilon)$.

Above lemma, along with the classification calibration property of l, let us know that training with $\tilde{\varphi}(\cdot)$, despite the noise, will let us converge to the optimal classifier f^*:

Lemma 5.6. *For any* $(\epsilon, \delta) > 0$ *pair, there exists large enough* K, N *such that with probability at least* $1 - \delta$, $R_{\mathcal{D}}(\tilde{f}^*_{\tilde{\varphi}}) - R^* \leq \epsilon$.

6 UNINFORMATIVE STRATEGIES

We show the above mechanism can help eliminate undesirable uninformative equilibria that exist in peer prediction literature. Though we are not really comparing two agents' outputs (may it be simple output agreement, or more sophisticated ones) for sending payment, it is possible that workers reporting the same labels leads to trivial reference answers that favor collusion. We show that with the call to Mechanism 4, agents reporting a symmetric uninformative strategy is not a BNE in our mechanism.

Theorem 6.1. *In Mechanism 1, (1) reporting symmetric uninformative pure signal is not an equilibrium. (2) Reporting symmetric uninformative mixed signal with randomization probability* $p \neq 1/2$ *is also not an equilibrium.*

We first observe that when agents report according to uninformative strategies, we will have $\tilde{p}_{-i,+} + \tilde{p}_{-i,-} = 1$. Thus Mechanism 1 will trigger a call to Mechanism 4. When the de-bias technique cannot be applied, we will randomly select another worker's report as a reference answer, and plug it into the scoring function, which creates a penalty for agreement. When there doesn't exist solution for $\tilde{p}_{-i,+}, \tilde{p}_{-i,-}$, we will score agent a sufficiently small value to ensure that truth-telling returns higher payment. Knowing only bounds on p_+, p_- would allow us the estimate a lower bound for $\min -\varphi_{\text{unbias}}(\cdot)$. In practice, after we shift the score in Mechanism 1 to be positive (use affine transformation, for individual rationality), the payment for this case can be set to 0.

MECHANISM 4: Exception handler

```
When $\tilde{p}_{-i,+} + \tilde{p}_{-i,-} = 1$:
    1. Set $\tilde{p}_{-i,+} := \min\{1, \tilde{p}_{-i,+} + \kappa\}$, $\tilde{p}_{-i,-} := \min\{1, \tilde{p}_{-i,-} + \kappa\}$.
    2. Randomly select another agent's report $\hat{y}_j$, $j \neq i$.
    3. Score each worker $i$: $-\tilde{\varphi}_{\text{unbias}}(\hat{y}_i, \hat{y}_j)$.
When there doesn't exist solution for $\tilde{p}_{-i,+}, \tilde{p}_{-i,-}$: Score agent a sufficiently small value that is less than
$\min -\varphi_{\text{unbias}}(\cdot)$ (the minimum value of $-\varphi_{\text{unbias}}(\cdot)$).
```

Proof. W.l.o.g., we consider the case workers contribute uninformative signals by always reporting label -1. Consider worker i. According to our estimation Eqn.(12) and (13), we have $\tilde{q}_{-i} = 1$ (always match), $\tilde{P}^{-i}_{-} = 1$. Plug back in Mechanism 2:

$$\text{(I): } \mathcal{P}_+(\tilde{p}^2_{-i,+} + (1 - \tilde{p}_{-i,+})^2] + \mathcal{P}_-(\tilde{p}^2_{-i,-} + (1 - \tilde{p}_{-i,-})^2] = 1, \text{ (II): } \mathcal{P}_+\tilde{p}_{-i,+} + \mathcal{P}_-(1 - \tilde{p}_{-i,-}) = 1 \,.$$

Since $\mathcal{P}_+\tilde{p}_{-i,+} \leq \mathcal{P}_+$, $\mathcal{P}_-(1 - \tilde{p}_{-i,-}) \leq \mathcal{P}_-$, and $\mathcal{P}_+ + \mathcal{P}_- = 1$, we know the only possible case that equation (II) holds is when both of the equalities hold. So we are led to the solution that $\tilde{p}_{-i,+} = 1, \tilde{p}_{-i,-} = 0$. Not hard to validate the above solution also satisfies Equation (1). According to Mechanism 4, we reset $\tilde{p}_{-i,-} := \kappa > 0$. Then the scoring function for agent i becomes:[5]

$$-\tilde{\varphi}(\hat{y}_i, y = -1) := -\frac{(1 - \tilde{p}_{-i,+})l(\hat{y}_i, -1) - \tilde{p}_{-i,-}l(\hat{y}_i, +1)}{1 - \tilde{p}_{-i,+} - \tilde{p}_{-i,-}} = -l(\hat{y}_i, +1)$$

[5] For simplicity of presentation, we drop the subscript and use φ to denote φ_{unbias}.

So to maximize the score it is profitable for agent i to report $\hat{y}_i = +1 \Rightarrow$ workers contributing the same uninformative signal is not an equilibrium. □

The case with mixed uninformative strategy can be argued in a similar flavor (see full version).

7 A SIMPLE MACHINE OUTPUT AGREEMENT MECHANISM

We have established the fact that a class of surrogate loss functions, combined with the classifier trained with it, can serve as a peer prediction scoring function that induces strictly truthful BNE. Likely there exist many other scoring functions that can also incorporate the machine prediction in peer prediction scoring. Probably the most intuitive mechanism under a peer prediction setting is the Output Agreement (OA) [24] mechanism. However, (OA) does not induce a truthful BNE if some agents know that they have received a minority signal. Yet in our mechanism, each agent's signal will be scored by an "informative" prediction given by the trained classifier, thus a truthful BNE will be induced. In this section we demonstrate the existence of a very simple Machine OA (MOA) mechanism that induces truthful BNE when ML elicitability conditions are satisfied.

MECHANISM 5: Machine Output Agreement (MOA)

For each worker i:

1. Train a classifier $\tilde{f}^*_{\tilde{\varphi},-i}$ as similarly done in Mechanism 1.
2. Pay worker i: $C_p \cdot \mathbb{1}(\tilde{f}^*_{\tilde{\varphi},-i}(\mathbf{x}_i) = \hat{y}_i)$, for some $C_p > 0$.

The idea is simply to check whether agent's report matches the classifier prediction. Denote by $\delta_R := 1 - R_{\mathcal{D}|y=+1}(f^*) - R_{\mathcal{D}|y=+1}(f^*)$ (which we will prove to be positive later) and assume workers have perfect knowledge of the mechanism. We have the following results (where we use *Const(·)* to denote a constant that depends only on its inputs):

Theorem 7.1. *With (MOA), when $K \geq Const1(\delta_p, \delta_R, \Delta, N)$, $N \geq Const2(\delta_p, \delta_R, \Delta)$, and C_p is set appropriately, every worker truthfully reporting is a BNE.*

For concise presentation, we only provide some intuitive reasoning on why the mechanism works and how we obtain the results: first we show that $R_{\mathcal{D}|y=+1}(f^*) + R_{\mathcal{D}|y=+1}(f^*) < 1$. The implication of this result is not unlike the $p_+ + p_- < 1$ one, in that the optimal classifier's prediction is informative under Bayesian updates. Particularly similar to Lemma 2.1, we can show this condition is equivalent with $\Pr(y_i = s | f^*(\mathbf{x}_i) = s|) > \mathcal{P}_s$, $\forall s \in \{+1, -1\}$. So in short, f^* is "informative" in a posterior way. When the number of training samples is large enough, the trained classifier $\tilde{f}^*_{\tilde{\varphi},-i}$ is also going to be informative. As $\hat{y}_i$ is compared to an informative answer, worker i is better off truth telling, due to Bayesian informativeness of his own observation proved in Lemma 2.1.

8 EFFORT SENSITIVE WORKERS

We extend our model and results to the case when workers are effort sensitive: once given a task, each worker i can choose to exert effort $e_i = 1$ to improve the quality of his label, or he can shirk from doing so ($e_i = 0$). Exerting effort incurs cost $c > 0$, and this is common knowledge to all workers and the principal. After making decision on effort exertion, each worker observes a signal (label) $\tilde{y}_i$. We assume that different effort levels lead to different flipping-error rates:

$$\Pr(\tilde{y}_i = -1 | y_i = +1, e_i) = p_+(e_i), \ \Pr(\tilde{y}_i = +1 | y_i = -1, e_i) = p_-(e_i).$$

Not exerting effort leads to an uniform random observation, while exerting effort returns an informative observation with less error: that is, we assume $p_+(0) = p_-(0) = 1/2$ and $p_+(1) + p_-(1) < 1$. Workers would like to maximize their net payment (i.e. payment minus cost of effort). A worker's strategy now has two components: an effort exertion decision and a reporting decision.

The principal's goal is to design a mechanism $\mathcal{M} = (S, \mathcal{A})$ to induce a strict BNE where every worker exerts effort and truthfully reports his observed signal. Denote $\mathcal{P}(\{(\mathbf{x}_j, \tilde{y}_j)\}_{j=1}^T | \{e_j\}_{j=1}^T)$ as an agent's belief about the realized labels of all agents when effort levels e_j's are selected. Then a mechanism $\mathcal{M}$ induces such a BNE if for $\mathcal{K}_{-i} = \{(\mathbf{x}_j, \tilde{y}_j)\}_{j \neq i}$,

$$\mathbb{E}_{\mathcal{P}(\{(\mathbf{x}_j,\tilde{y}_j)\}_{j=1}^T | \{e_j=1\}_{j=1}^T)}[S(\tilde{y}_i, f^{\mathcal{A}}_{\mathcal{K}_{-i}}(\mathbf{x}_i))] > \mathbb{E}_{\mathcal{P}(\{(\mathbf{x}_j,\tilde{y}_j)\}_{j=1}^T | \{e_i, \{e_j=1\}_{j\neq i}\})}[S(\hat{y}_i, f^{\mathcal{A}}_{\mathcal{K}_{-i}}(\mathbf{x}_i))],$$

for all i, either that $e_i \neq 1$ or $\hat{y}_i \neq \tilde{y}_i$, or both.

We show that we can again design mechanisms using surrogate loss functions to incentivize both effort and truth-telling. The truth telling part can be similarly established as for the effort insensitive case, but a slightly different argument is needed for eliciting effort.

Theorem 8.1. *When K, N are large enough, $S(\hat{y}_i, \tilde{f}_{\tilde{\varphi},-i}(\mathbf{x}_i)) = -a \cdot \tilde{\varphi}_{unbias}(\tilde{f}_{\tilde{\varphi},-i}(\mathbf{x}_i), \tilde{y}_i)$, along with the algorithm detailed in* (MLP)*, induce a strict BNE for workers to exert effort and report truthfully, with an appropriately selected scaling factor $a > 0$.*

Proof. (Sketch) Again for easiness of presentation we will omit the unbias subscript in φ. According to Theorem 4.2, we know if all other workers exert effort and report truthfully, if worker i decides to exert effort, he will report truthfully. The only case we need to consider is when agent i chooses not to exert effort. When agent i sets $e_i = 0$, the negative of his expected utility is [6] $\mathbb{E}[\varphi(f^*, \hat{y}_i) | e_i = 0] = \frac{1}{2}(\mathbb{E}[\varphi(f^*, y_i)] + \mathbb{E}[\varphi(f^*, y_i)])$, regardless of his reporting strategy. We show the difference between above and the case with exerting effort and truthfully reporting is negative, which can be proved via using the fact that $-\mathbb{E}[\varphi(f^*, \tilde{y}_i) | e_i = 1] > -\mathbb{E}[\varphi(f^*, -\tilde{y}_i) | e_i = 1]$ (truth telling is better than reverting):

$$\mathbb{E}[\varphi(f^*, \tilde{y}_i) | e_i = 1] - \mathbb{E}[\varphi(f^*, \hat{y}_i) | e_i = 0] = \mathbb{E}[(p_{\text{sgn}(y_i)} - 1/2)\varphi(f^*, -y_i) + (1/2 - p_{\text{sgn}(y_i)})\varphi(f^*, y_i)] > 0.$$

Scaling up $(-a \cdot \varphi(f^*, \hat{y}_i))$ will cover cost c so to make it incentive compatible to exert effort. □

9 DISCUSSIONS AND CONCLUSION

We have introduced a new approach that connects information elicitation without verification with machine learning. We now summarize the practical advantages of this approach. First our mechanism does not heavily rely on obtaining redundant labels by assigning a task to multiple workers. Hence, we can save learning budget to more efficiently train a classifier as the performance of classifiers ties closely to the number of unique training data. Second, we remove the requirement of knowing the joint distribution of workers' signals; rather such statistics is learned through workers' reports (via learning workers' error rates). Third, our mechanism can easily handle (and are specialized for) heterogeneous tasks. Finally, our mechanism no long have the undesirable uninformative and permutation equilibria that most other peer prediction mechanisms have.

Our mechanism is also robust when a small number of agents deviate from the equilibrium: this can be established by using the robustness of the training procedure. When a sublinear (in N) $N^\theta, 0 < \theta < 1$ number of agents deviate, this will create a $O(N^\theta / N) = O(N^{\theta - 1})$ bias in the learned surrogate loss function, as well as in the empirical loss defined over training data. According to Lemma 5.5, the resultant classifier will still converge to the optimal one, with this additional converging error (in N). Also agent's report will be evaluated against an accurate scoring function.

REFERENCES

[1] Peter L Bartlett, Michael I Jordan, and Jon D McAuliffe. 2006. Convexity, Classification, and Risk Bounds. *J. Amer. Statist. Assoc.* 101, 473 (2006), 138–156.

[6] Again for simplicity we will evaluate the noise free scoring function, along with f^*, due to convergence.

[2] Yang Cai, Constantinos Daskalakis, and Christos H Papadimitriou. 2015. Optimum Statistical Estimation with Strategic Data Sources. In *COLT*.

[3] Rachel Cummings, Stratis Ioannidis, and Katrina Ligett. 2015. Truthful Linear Regression. In *Proceedings of The 28th Conference on Learning Theory, COLT 2015, Paris, France, July 3-6, 2015*. 448–483.

[4] Anirban Dasgupta and Arpita Ghosh. 2013. Crowdsourced Judgement Elicitation with Endogenous Proficiency. In *Proceedings of the 22nd international conference on World Wide Web*. 319–330.

[5] Boi Faltings, Radu Jurca, Pearl Pu, and Bao Duy Tran. 2014. Incentives to Counter Bias in Human Computation. In *Second AAAI Conference on Human Computation and Crowdsourcing*.

[6] Boi Faltings, Jason Jingshi Li, and Radu Jurca. 2014. Incentive mechanisms for community sensing. *IEEE Trans. Comput.* 63, 1 (2014), 115–128.

[7] Tilmann Gneiting and Adrian E. Raftery. 2007. Strictly Proper Scoring Rules, Prediction, and Estimation. *J. Amer. Statist. Assoc.* 102, 477 (2007), 359–378.

[8] Stratis Ioannidis and Patrick Loiseau. 2013. Linear Regression as a Non-cooperative Game. In *International Conference on Web and Internet Economics*. Springer, 277–290.

[9] Radu Jurca and Boi Faltings. 2006. Minimum Payments that Reward Honest Reputation Feedback. In *Proceedings of the 7th ACM conference on Electronic commerce (EC '06)*. ACM, 190–199.

[10] Radu Jurca, Boi Faltings, and others. 2009. Mechanisms for Making Crowds Truthful. *Journal of Artificial Intelligence Research* 34, 1 (2009), 209.

[11] Ece Kamar and Eric Horvitz. 2012. Incentives for Truthful Reporting in Crowdsourcing. In *Proceedings of the 11th international conference on autonomous agents and multiagent systems-volume 3*. International Foundation for Autonomous Agents and Multiagent Systems, 1329–1330.

[12] David R Karger, Sewoong Oh, and Devavrat Shah. 2011. Iterative Learning for Reliable Crowdsourcing Systems. In *Advances in neural information processing systems*. 1953–1961.

[13] Yuqing Kong and Grant Schoenebeck. 2016. A Framework For Designing Information Elicitation Mechanisms That Reward Truth-telling. *arXiv preprint arXiv:1605.01021* (2016).

[14] Debmalya Mandal, Matthew Leifer, David C. Parkes, Galen Pickard, and Victor Shnayder. 2016. Peer Prediction with Heterogeneous Tasks. In *Proc. of the NIPS Workshop on Crowdsourcing and Machine Learning*. https://arxiv.org/abs/1612.00928

[15] Nolan Miller, Paul Resnick, and Richard Zeckhauser. 2005. Eliciting Informative Feedback: The Peer-Prediction Method. *Management Science* 51, 9 (2005), 1359 –1373.

[16] Nagarajan Natarajan, Inderjit S Dhillon, Pradeep K Ravikumar, and Ambuj Tewari. 2013. Learning with Noisy Labels. In *Advances in neural information processing systems*. 1196–1204.

[17] Dražen Prelec. 2004. A Bayesian Truth Serum for Subjective Data. *Science* 306, 5695 (2004), 462–466.

[18] Goran Radanovic and B. Faltings. 2013. A Robust Bayesian Truth Serum for Non-Binary Signals. In *Proceedings of the 27th AAAI Conference on Artificial Intelligence (AAAI '13)*.

[19] Goran Radanovic, Boi Faltings, and Radu Jurca. 2016. Incentives for Effort in Crowdsourcing using the Peer Truth Serum. *ACM Transactions on Intelligent Systems and Technology (TIST)* 7, 4 (2016), 48.

[20] Clayton Scott. 2015. A Rate of Convergence for Mixture Proportion Estimation, with Application to Learning from Noisy Labels.. In *AISTATS*.

[21] Victor S Sheng, Foster Provost, and Panagiotis G Ipeirotis. 2008. 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*. ACM, 614–622.

[22] Victor Shnayder, Arpit Agarwal, Rafael Frongillo, and David C. Parkes. 2016. Informed Truthfulness in Multi-Task Peer Prediction. *ACM EC* (March 2016). arXiv:cs.GT/1603.03151

[23] Vladimir Vapnik. 2013. *The Nature of Statistical Learning Theory*. Springer Science & Business Media.

[24] Luis von Ahn and Laura Dabbish. 2004. Labeling images with a computer game. In *Proceedings of the SIGCHI conference on human factors in computing systems (CHI '04)*. ACM, 319–326.

[25] Bo Waggoner and Yiling Chen. 2014. Output Agreement Mechanisms and Common Knowledge. In *Proceedings of the 2nd AAAI Conference on Human Computation and Crowdsourcing (HCOMP'14)*.

[26] Jens Witkowski, Yoram Bachrach, Peter Key, and David C. Parkes. 2013. Dwelling on the Negative: Incentivizing Effort in Peer Prediction. In *Proceedings of the 1st AAAI Conference on Human Computation and Crowdsourcing (HCOMP'13)*.

[27] Jens Witkowski and David Parkes. 2012. A Robust Bayesian Truth Serum for Small Populations. In *Proceedings of the 26th AAAI Conference on Artificial Intelligence (AAAI '12)*.

[28] Jens Witkowski and David C. Parkes. 2012. Peer Prediction without a Common Prior. In *Proceedings of the 13th ACM Conference on Electronic Commerce (EC '12)*. ACM, 964–981.

[29] Peter Zhang and Yiling Chen. 2014. Elicitability and Knowledge-free Elicitation with Peer Prediction. In *Proceedings of the 2014 International Conference on Autonomous Agents and Multi-agent Systems (AAMAS '14)*. 245–252.