Unifying Framework for Crowd-sourcing via Graphon Estimation

Christina E. Lee
Massachusetts Institute of Technology
Cambridge, MA 02139
celee@mit.edu

Devavrat Shah
Massachusetts Institute of Technology
Cambridge, MA 02139
devavrat@mit.edu

ABSTRACT

We consider the question of inferring true answers associated with tasks based on potentially noisy answers obtained through a micro-task crowd-sourcing platform such as Amazon Mechanical Turk. We propose a generic, non-parametric model for this setting: for a given task i, $1 \le i \le T$, the response of worker j, $1 \le j \le W$ for this task is correct with probability F_{ij}, where matrix $F = [F_{ij}]_{i \le T, j \le W}$ may satisfy one of a collection of regularity conditions including low rank, which can express the popular Dawid-Skene model; piecewise constant, which occurs when there is finitely many worker and task types; monotonic under permutation, when there is some ordering of worker skills and task difficulties; or Lipschitz with respect to an associated latent non-parametric function. This model, contains most, if not all, of the previously proposed models to the best of our knowledge.

We show that the question of estimating the true answers to tasks can be reduced to solving the *Graphon estimation* problem, for which there has been much recent progress. By leveraging these techniques, for a large class of regularity conditions under which there exists performance bounds for Graphon estimation, we can equivalently provide bounds on the fraction of incorrectly estimated tasks of the resulting crowdsourcing algorithm. Subsequently, we have a solution for inferring the true answers for tasks using noisy answers collected from crowd-sourcing platform under a significantly larger class of models. Concretely, we establish that if the (i, j)th element of F, F_{ij}, is equal to a Lipschitz continuous function over latent features associated with the task i and worker j for all i, j, then all task answers can be inferred correctly with high probability by soliciting $\tilde{O}(\ln(T)^{3/2})$ responses per task even *without* any knowledge of the Lipschitz function, task and worker features, or the matrix F.

CCS CONCEPTS

•Theory of computation → Machine learning theory; •Computing methodologies → Machine learning; •Human-centered computing → *Collaborative and social computing theory, concepts and paradigms;*

KEYWORDS

Crowdsourcing, Graphon estimation, Matrix estimation, Nonparametric, Dawid-Skene model, Univeral singular value thresholding

1 INTRODUCTION

1.1 Background

In the recent years, crowd-sourcing has become a complementary computing system to scale tasks that are difficult for algorithms to solve, but easy and trivial for humans. For example, this includes tasks such as image recognition ("is this picture culturally acceptable"), content censorship ("is this webpage suitable for children") or social opinion ("is this a good coffee shop for writing a paper"). It may be computationally challenging to train an algorithm to determine if an image is culturally offensive, or if a webpage contains explicit content; and it would be impossible for an algorithm to provide a human opinion on the suitability of a coffee shop for writing a paper, or on the pros and cons of legalizing marijuana. On the other hand, a human could relatively easily and quickly provide answers to these aforementioned tasks.

As a result, crowd-sourcing platforms such as Amazon Mechanical Turk have emerged, on which requesters can post tasks that they would like to be solved along with a monetary reward for completion, and human workers can browse the posted tasks and earn money for providing responses to these tasks. However, for a variety of reasons, the responses provided by human workers may not be consistent amongst themselves, and may not correspond to the true answer or solution for the task. For example, consider a language translation task which asks to translate a phrase between two languages. Different human workers may have different levels of language proficiency, leading to noisy responses. Alternatively, even if a worker is capable of solving the task, s/he may be lazy and may respond arbitrarily to save the effort. In the context of collecting social opinion, lack of consensus in the population responding to the task or question is expected. Therefore, the challenge is, given a set of responses provided by human workers for a set of tasks that are noisy, unreliable and potentially contradicting each other, can we infer the true answer or solution for the task?

The history of this problem pre-dates crowd-sourcing, as it involves the meta question of how to infer information given noisy data. For example, consider a survey or census, in which we may gather self-reported data from a subset of the population, and we would like to infer some property of the population at large. Again the data could be unreliable due to people misreporting information. Alternatively consider the setting of patient care in a medical hospital where a patient may receive many tests and diagnoses from different doctors, medical residents, and nurses. There could be noise or variability amongst the experts, and we would like to infer the true diagnosis. The classical approach taken to model

this problem is the Dawid-Skene model, which assumes that every worker is associated with a reliability parameter, which determines the probability that the worker provides a correct response for any assigned task [9]. This basic model assumes that all tasks are essentially homogeneous. Subsequently, a few different generalized models have been proposed. [17] introduces a task difficulty parameter, which is the probability that a task is perceived incorrectly. [12, 13] assumes that there are finite k types of tasks, and a worker has a separate reliability parameter for each task type. [21] assumes that there is an ordering of worker skill levels and task difficulty levels under which the probability of a correct response is monotonic. [22] proposes a Gaussian graphical model for modeling responses based on task and worker parameters. Each of these models impose different assumptions which are exploited in the associated algorithm and analysis.

1.2 Contributions

The key contribution of this paper is to show that in fact there does exist a simple unifying framework which contains all of these models. We propose a fully general nonparametric model for crowd-sourcing, which associates the probability of a correct response to an arbitrary parameter specific to the worker and task pair. Under basic regularity conditions within which all of the above models can be expressed, we present a simple algorithm and corresponding error bounds. We essentially show that the crowd-sourcing inference problem can be reduced to solving a Graphon estimation problem, which is also equivalent in some conditions to matrix estimation and latent variable model estimation.

We can translate performance guarantees for Graphon estimation into performance guarantees for the proposed crowd-sourcing inference algorithm. This results in concrete performance bounds for our proposed crowd-sourcing algorithm by leveraging existing results in the literature for Graphon estimation. Our proposed algorithm assumes access to a subroutine, as a black-box. The input to the sub-routine is a data matrix and it outputs another matrix of the same dimensions. If the data matrix is sampled from a distribution over matrices, then the output of the sub-routine is a reasonable estimate of the average data matrix under this distribution.

The stochastic block model, graphon estimation, and matrix estimation literature provide various options for such a subroutine under the assumptions that the expected data matrix is monotonic under permutation, piecewise constant, Lipschitz, or low rank. Each of these assumptions leads to a slightly different method and corresponding performance guarantees.

Our algorithm applies such a sub-routine to the matrix of noisy answers, where the (i, j)-th element of the matrix represents the answer provided by worker j for task i if worker j was assigned to task i, and remains empty otherwise. A simple aggregation rule is then applied to the resulting output matrix of this subroutine, resulting in the desired estimate for the underlying task answers. For example, under the assumption that each of the entries of the underlying expected data matrix is equal to the output of a Lipschitz function over associated row (or task) and column (or worker) latent features, then we can prove that the answers to all tasks can be recovered with high probability (i.e. probability going to 1 quickly enough as number of tasks grow) by soliciting $\tilde{O}(\delta_0^{-6}\ln(T)^{3/2})$ responses per task; here δ_0 represent the effective "signal" in the model (see Corollary 5.3 in Section 5 for precise details).

1.3 Organization

The remainder of this paper is organized as follows. In Section 2, we describe the formal setup. Section 3 describes various related works including prior work in the crowd-sourcing literature. Section 4 describes the inference algorithm for tasks based on noisy answers. Section 5 states the main result about the performance of the algorithm. Section 6 provides detailed proofs. Finally, Section 7 provides discussion about this work as well as directions for future research.

1.4 Notation

We shall use $\mathbb{R}$ to denote all real values, $\mathbb{R}_+$ to denote strictly positive real values, $\mathbb{Z}$ to represent all integers, and $\mathbb{Z}_+$ to represent strictly positive integers. For any $A \in \mathbb{Z}_+$, $[A]$ represents the set $\{1, \ldots, A\}$. For an $a \times b$ real-valued matrix $Q = [Q_{ij}]$, its Frobenius norm, denoted by $\|Q\|_F$, is given by $\|Q\|_F = \left(\sum_{i=1}^{a}\sum_{j=1}^{b} Q_{ij}^2\right)^{\frac{1}{2}}$. The nuclear norm of Q, denoted by $\|Q\|_*$, is defined as $\|Q\|_* = \sum_{i=1}^{\min(a,b)} s_i$, where $s_i,\ 1 \le i \le \min(a, b)$ are the singular values of Q.

Given an $a \times b$ matrix Q, let $\hat{Q}$ be a random matrix that is an estimator of Q. Then the average mean squared error of this estimator, denoted as $\mathrm{MSE}(\hat{Q})$, is defined as

$$\mathrm{MSE}(\hat{Q}) = \frac{1}{ab}\mathbb{E}\left[\|Q - \hat{Q}\|_F^2\right]. \tag{1}$$

The root mean squared error, denoted as $\mathrm{RMSE}(\hat{Q})$, is simply defined as the square-root of $\mathrm{MSE}(\hat{Q})$, that is,

$$\mathrm{RMSE}(\hat{Q}) = \sqrt{\mathrm{MSE}(\hat{Q})}. \tag{2}$$

The indicator function is denoted by $\mathbb{I}$ and defined as

$$\mathbb{I}(x) = \begin{cases} 1 & \text{if } x = true \\ 0 & \text{otherwise.} \end{cases} \tag{3}$$

2 SETUP

In this section, we shall introduce the model and problem statement, the regularity conditions and assumptions, and the performance metric utilized to measure the algorithmic performance.

2.1 Model and Problem Statement

We shall use $T \in \mathbb{Z}_+$ to denote the number of tasks and $W \in \mathbb{Z}_+$ to denote the number of workers. Let each task $i \in [T]$ be associated with a true answer $a_i \in \{-1, 1\}$. We shall also use $t_i,\ i \in [T]$ to denote the i-th task and $w_j,\ j \in [W]$ to denote the j-th worker. A system designer can assign a given task $i \in [T]$ to any of the workers $j \in [W]$. Let M_{ij} be the answer provided to task $i \in [T]$ by worker $j \in [W]$. We assume that $M_{ij} \in \{-1, 0, 1\}$ such that if task i is assigned to worker j, then

$$M_{ij} = \begin{cases} a_i & \text{with probability } F_{ij} \\ -a_i & \text{with probability } 1 - F_{ij}, \end{cases}$$

and if task i is not assigned to worker j then $M_{ij} = 0$. The entries M_{ij} are independent across $i \in [T]$, $j \in [W]$. We introduce the matrix $F = [F_{ij}]_{i\in[T],j\in[W]}$ as the associated parameter matrix. In Section 2.3, we shall discuss various regularity conditions on the structure of F induced by different model assumptions. Given the noisy answer matrix $M = [M_{ij}]_{i\in[T],j\in[W]}$, the goal is to infer the true answers a_i, for all $i \in [T]$.

To summarize, the two operational questions of interest are: (1) how should we assign tasks to workers, and (2) how can we infer the true answers for all tasks based on the noisy answers obtained from the task-worker assignments. The cost incurred (or budget spent by task requester) is proportional to the total number of answers solicited from workers (in our setting, this is equal to $\|M\|_F^2$). Therefore, we would like to minimize the number of solicited answers. On the other hand, the accuracy of the inferred true answers from the noisy answers is likely to increase as our system collects more answers, or responses from workers. In a nutshell, the primary goal of designing a crowd-sourcing system is to achieve the best possible trade-off between the cost and accuracy.

2.2 Performance Metric

As mentioned, there are two key performance metrics: cost and accuracy. We shall formally define them in this section.

Cost. The total number of answers solicited is $\sum_{i\in[T],j\in[W]} \mathbb{I}(M_{ij} \neq 0)$ which happens to be equal to $\|M\|_F^2$ since $M_{ij} \in \{-1, 0, 1\}$ for all $i \in [T], j \in [W]$. For each task, there is a fixed cost (or wage) associated to collecting the responses used to estimate the final true answer. That cost is proportional to the total number of questions. Therefore, without loss of generality, we shall utilize the total number of questions, $\|M\|_F^2$, as a proxy for the total cost. We will be interested in the cost per task (equivalently, number of questions per task) which is equal to $\|M\|_F/T$.

Accuracy. Let $\hat{a}_i$ denote the estimate of the answer for task i. A common loss function used is the fraction of tasks that are incorrectly estimated,

$$\mathcal{L}_1 = \frac{1}{T}\sum_{i\in[T]} \mathbb{I}(\hat{a}_i \neq a_i).$$

As [21] pointed out, we may also be interested in a weighted loss function which penalizes incorrect estimates for "easy" tasks more than "difficult" tasks. For example, if $F_{ij} = \frac{1}{2}$ for all w_j, then the collected responses for task i will be distributionally equivalent to a set of coin flips that have no correlation with the true answer a_i. Since there is no information gained, the algorithm cannot expect to estimate the answer better than a random coin flip, and it may be reasonable not to penalize the method in the loss function for such a task. Therefore, we define a general weighted loss function

$$\mathcal{L}_\theta = \frac{1}{T}\sum_{i\in[T]} \theta(i)\mathbb{I}(\hat{a}_i \neq a_i), \tag{4}$$

where $\theta : \mathbb{Z}_+ \to \mathbb{R}_+$ is some weight function. In [21], they choose the weight function to be

$$\theta(i) = \frac{1}{W}\Big(\sum_{j\in[W]} (2F_{ij} - 1)^2\Big) = \hat{\mathbb{E}}_j\left[(2F_{ij} - 1)^2\right]. \tag{5}$$

Remarks. In this paper, we will not specifically address the issue of hard constraints on the maximum number of tasks assigned to a worker, or maximum number of workers assigned to a given task, but the methods we propose have natural modifications to a budget-constrained setting while preserving the theoretical guarantees. We also limit ourselves to a static method in this paper, where the tasks are first assigned, next the responses are collected, and finally the estimator is computed. We can use similar techniques as the recent adaptive method of [17], to modify our method to suit an adaptive setting, in which the workers and tasks arrive in sequence, and sequential decisions are made to assign batches of tasks and workers, informed by previous estimates computed.

2.3 Operating Assumptions

Random Task Assignment. We shall assume that task assignments are done randomly. We formalize it as follows.

Condition 2.1. *Task i is assigned to worker j with probability $p_{obs} \in [0, 1]$ for all $i \in [T], j \in [W]$ independently. Given this,*

$$M_{ij} = \begin{cases} a_i & \text{with probability } p_{obs}F_{ij} \\ -a_i & \text{with probability } p_{obs}(1 - F_{ij}) \\ 0 & \text{with probability } (1 - p_{obs}). \end{cases}$$

Define the expected response matrix as $G = [G_{ij}]_{i\in[T],j\in[W]}$, where $G_{ij} = \mathbb{E}[M_{ij}] = p_{obs}a_i(2F_{ij} - 1)$.

Regularity Conditions on F. As explained in Section 4, our algorithm crucially utilizes a 'sub-routine' that can estimate the expected response matrix G from the sampled response matrix M. In the existing literature of Matrix completion, Graphon estimation, and Latent variable models, there are various such subroutines proposed. Each of the proposed methods provide different performance guarantees under different regularity conditions on the expected response matrix. In our setting, these translate into conditions on the parameter matrix F. We list a few of such regularity conditions:

Condition 2.2. *Let the parameter matrix F satisfy one of the following regularity conditions:*

1. *F is a low rank matrix with rank r.*
2. *$F_{ij} = f(t_i, w_j)$ for all $i \in [T], j \in [W]$, where t_i, w_j are the* latent *features associated with the row (or task) i, and the column (or worker) j. They are sampled from distributions over* latent feature spaces *Ω_T and Ω_W respectively. The sampling is done independently and identically distributed across all rows $i \in [T]$, and across all columns $j \in [W]$. The* latent function *takes value between 0 and 1, i.e. $f : \Omega_T \times \Omega_W \to [0, 1]$. Additionally, one of the following properties holds:*
   (a) *The latent spaces Ω_T and Ω_W are compact subsets of $\mathbb{R}^{d_1}$ and $\mathbb{R}^{d_2}$ respectively for some $d_1, d_2 \in \mathbb{Z}_+$. The latent function f is assumed to be piece-wise Lipschitz.*
   (b) *The latent spaces are the unit inverval $\Omega_T = \Omega_W = [0, 1]$, and the features are sampled uniformly from the space. The latent function f is such that the expected "degrees", $\int_0^1 f(t, w)dw$, is strictly monotone in t, and $\int_0^1 f(t, w)dt$ is strictly monotone in w.*

(3) *There are at most k_1 distinct valued rows and k_2 distinct valued columns in F. This is equivalent to enforcing that there are finitely many types of rows and columns, and the value F_{ij} is only a function of the type of row and type of column.*

2.4 An Essential Assumption

The current problem as stated is still impossible because we do not know the parameter matrix F. Under Condition 2.2, every satisfying matrix F has a complement F' such that $F'_{ij} = 1 - F_{ij}$, and F' also satisfies the same regularity condition. Therefore, the distribution over collected responses under parameter matrix F for a task answer vector $\mathbf{a} \in \{-1, 1\}^T$ is exactly equivalent to the distribution over collected responses under the complement parameter matrix F' for a completely opposite task answer vector $\mathbf{a}' = -\mathbf{a}$. And hence we cannot hope to distinguish between the two parameter settings, which have entirely opposite solutions. Therefore, we must impose appropriate conditions on F (or provide one bit of additional information that can help distinguish between F and F' above). There have been a couple different sufficient assumptions that have been previously proposed in the literature.

One such condition is to require that for all $i \in [T], j \in [W]$, $F_{ij} \in \left(\frac{1}{2}, 1\right]$ (or $\left[0, \frac{1}{2}\right)$). This rules out the existence of the pair of confounding matrices (F, F') as constructed above, as long as the algorithm has knowledge of this condition. This assumption in fact implies that $a_i = \text{sign}(G_{ij})$ (or $-\text{sign}(G_{ij})$) for all $j \in [W]$ for any given $i \in [T]$.

A less stringent condition is to require such a condition only in aggregate: for any $i \in [T]$, the average of the i-th row of F is at least $1/2$ (or at most $1/2$), i.e.

$$\frac{1}{W} \sum_{j \in [W]} F_{ij} > \frac{1}{2} \quad \Big(\text{or } < \frac{1}{2}\Big). \tag{6}$$

In this case, the solution is given by the relationship

$$a_i = \text{sign}\Big(\frac{1}{W} \sum_{j \in [W]} G_{ij}\Big). \tag{7}$$

In other words, the solution is equal to the population majority vote. This assumption is a lot more flexible, as it does not impose assumptions on all individual worker task pairs, but only on the population average. We shall assume this model for establishing our results in the remainder of this paper. Given this assumption, the number of workers required to effectively determine the true answer to a task naturally depends on the distance of the average away from zero, which is expressed as

$$\begin{aligned} \Big|\frac{1}{W} \sum_{j \in [W]} G_{ij}\Big| &= |a_i| p_{obs} \Big|\frac{1}{W} \sum_{j \in [W]} (2F_{ij} - 1)\Big)\Big| \\ &= p_{obs} \Big|\frac{1}{W} \sum_{j \in [W]} (2F_{ij} - 1)\Big|. \end{aligned} \tag{8}$$

This leads to the definition of the effective *gap*,

$$\delta_i \equiv \Big|\frac{1}{W} \sum_{j \in [W]} (2F_{ij} - 1)\Big|. \tag{9}$$

It has been argued that the efficacy of an inference algorithm to infer G accurately from M might be affected by a variant of such *gap*, which is also called *collective intelligence* in the literature and defined as,

$$\sigma_i^2 \equiv \frac{1}{W}\Big(\sum_{j \in [W]} (2F_{ij} - 1)^2\Big). \tag{10}$$

By Jensen's inequality, $\sigma_i^2 \geq \delta_i^2$ for all $i \in [T]$.

3 RELATED WORKS

Many inference algorithms have been proposed and analyzed for the Dawid and Skene model, using techniques such as the EM algorithm [9, 10, 25], belief propagation and iterative methods [14–16, 18–20], and spectral methods [3, 8, 11]. However, as mentioned above, this model does not allow for task heterogeneity. Many generalizations to the model either lack computationally efficient methods or provable performance guarantees. There has been recent progress towards this end, most notably in [12, 13, 17, 21].

The work of [12, 13] assumes a fairly general model in which a task i is associated to one of k finite categories, denoted by $t_i \in [k]$. Each worker j has an arbitrary associated vector $(f(1, j), f(2, j) \ldots f(k, j)) \in [0, 1]^k$, where $F_{ij} = f(t_i, j)$ denotes the probability that worker j answers task of type t_i correctly. However, their algorithm assumes that the ground truth is known for some tasks, which is used to bootstrap the estimates for a primal-dual approximation method. In general we may not know the ground truth for any of the tasks a priori. We note that even *without assuming ground-truth for any task*, if there were also only finitely many worker types, this model satisfies Condition 2.2(3), for which our solution works.

The work of [17] assumes that every worker j is associated with a parameter $w_j \in [0, 1]$ such that $\frac{1}{W}\Big(\sum_{j \in [W]} w_j\Big) > \frac{1}{2}$, every task i is associated with a parameter $t_i \in \left[\frac{1}{2}, 1\right]$, and the probability of worker j answering task i correctly is given by the function

$$F_{ij} = f(t_i, w_j) = t_i w_j + (1 - t_i)(1 - w_j). \tag{11}$$

Indeed, this becomes a special case of our Condition 2.2(2a) since it can be verified that f is a piecewise Lipschitz. They show that in order to achieve accuracy (fraction of correct answers) at least $1 - \alpha$, a nonadaptive algorithm requires the number of responses per task to be scaling as

$$\frac{\log(1/\alpha)}{\Big(\min_{i \in T}(2t_i - 1)^2\Big)\Big(\frac{1}{W}\sum_{j \in W}(2w_j - 1)^2\Big)}. \tag{12}$$

The work of [21] assumes that there exists some permutation of the workers and tasks such that the probability of a correct response is monotonic. Specifically, this means that each task i is associated with an order t_i in the task permutation, and a worker j is associated with an order w_j in the worker permutation, such that the probability of a correct response, denoted as $F_{ij} = f(t_i, w_j)$, is monotonically increasing in t_i and w_j. This satisfies regularity Condition 2.2(2b). Although this model allows for different task types, the property of monotonicity with respect to an order does not allow for specialization, e.g. one set of workers is better at one set of tasks, while another set of worker is better on a different subset of tasks.

[22] and [16] propose a Gaussian graphical model describing the probability of a worker answering a task correctly. In the basic setting, consider that each worker i is associated with parameters α_j and β_j, and that each task i is associated with the true answer a_i and a parameter $r_i > 0$. The worker's response to a task is modeled as the sign of a Gaussian random variable Z_{ij} with mean $\alpha_j a_i + \beta_j$ and variance r_i. If Φ denotes the Cumulative Density Function (CDF) of a standard normal random variable, then

$$\mathbb{P}(M_{ij} = a_i) = \Phi\left(\frac{\alpha_j a_i + \beta_j}{\sqrt{r_i}}\right).$$

To formulate this within our framework, let the worker type be $w_j = (\alpha_j, \beta_j)$, and let the task type $t_i = (a_i, r_i)$. We can compute the probability of a correct answer to be

$$F_{ij} = \mathbb{P}(M_{ij} = a_i) = \Phi\left(\frac{\alpha_j + \beta_j a_i}{\sqrt{r_i}}\right) =: f(t_i, w_j).$$

We can in fact verify that this function f as defined is piecewise Lipschitz, satisfying Condition 2.2(2a), as long as r_i is bounded away from zero, and α_j and β_j are bounded in magnitude. [22] proposes the generalized version when the parameters are multi-dimensional. They provide empirical studies to illustrate the performance, but lack theoretical guarantees.

[23] proposes a model in which workers and tasks are associated to parameters w_j, t_i respectively, and the probability of a correct answer is modeled as

$$F_{ij} = f(t_i, w_j) = \frac{1}{1 + \exp(-t_i w_j)}.$$

As long as the parameters t_i, w_j are bounded in magnitude, then we can verify that this function f is Lipschitz, satisfying Condition 2.2(2a). We note that [23] provides a primarily empirical study. In a sense, our result as a special case, provides the missing theoretical guarantees for [23].

4 ALGORITHM

Assume that we are given an algorithm ESTIMATOR which takes as input a noisy data matrix $Q \in \{-1, 0, 1\}^{L \times L}$ and outputs an estimated matrix $\hat{Q} \in [-1, 1]^{L \times L}$ which approximates the expected response matrix $\mathbb{E}[Q]$; here it is assumed that Q is generated by sampling according to a distribution over $\{-1, 0, 1\}$ valued $L \times L$ matrices with a well defined average, $\mathbb{E}[Q] \in [-1, 1]^{L \times L}$. Provided such a method, we propose an algorithm for estimating the true answers for tasks based on noisy response data matrix from the workers; our algorithm uses ESTIMATOR as a black-box subroutine. We shall use an extremely simple task assignment algorithm – a batch procedure. Below, we describe our overall algorithm that describes how the task assignment is done as well as how answers are inferred.

Task Assignment. The task assignment is done through a very simple randomized batch-assignment policy. The precise details are below. The algorithm utilizes a batch-size parameter $L \in \mathbb{Z}_+$ which governs the accuracy of the algorithm.

For simplicity, we shall assume that the number of tasks T and number of workers W are multiples of L. Of course, if they are not, we can make them multiples of L by removing (or adding) up to $L - 1$ tasks chosen at random from the T tasks; and potentially getting rid of (or additionally recruiting) up to $L - 1$ workers from the W workers at random. We shall be interested in the scenario where $L \ll T, L \ll W$ and $T, W \to \infty$, such that a minor *adjustment* of worker or task numbers does not affect the results of this work. Thus, assuming that T, W are multiples of L for our results is without loss of generality.

(1) Arbitrarily partition the tasks into batches of size $L \in \mathbb{Z}_+$, denoted $\mathcal{T}_1, \ldots, \mathcal{T}_{T/L}$. Arbitrarily partition the workers into batches of size L, denoted $\mathcal{W}_1, \ldots, \mathcal{W}_{W/L}$.
(2) For each $a \in [T/L]$ and $b \in [W/L]$,
    (a) For each task $i \in \mathcal{T}_a$ and worker $j \in \mathcal{W}_b$, assign task i to worker j with probability $p_{obs} \in (0, 1]$.

Inferring True Answers. Given the randomized batch task assignment, we provide an algorithm for inferring the true answers based on the answers provided by workers for their assigned tasks. We shall use ESTIMATOR as a subroutine.

(1) For each $a \in [T/L]$ and $b \in [W/L]$,
    (a) Let M^{ab} denote the $L \times L$ submatrix of M, containing answers provided by workers belonging to batch $\mathcal{W}_b$ for tasks assigned to batch $\mathcal{T}_a$.
    (b) Compute $\hat{G}^{ab} = \text{ESTIMATOR}(M^{ab})$, the output of the ESTIMATOR procedure when provided M^{ab} as an input.
(2) Let $\hat{G}$ denote the $T \times W$ matrix which combines together $\hat{G}^{ab}$ for all $(a, b) \in [T/L] \times [W/L]$.
(3) Compute the final estimate for each $i \in [T]$,

$$\hat{a}_i = \text{sign}\left(\sum_{j \in [W]} \hat{G}_{ij}\right).$$

Remarks. Our algorithm as currently stated assigns all batches of tasks to all batches of workers. However, in certain scenarios, additional constraints may be enforced, e.g. no worker can receive more than certain number of tasks. Of course, such a constraint will lead to trade-offs between the limit on the number of tasks and the accuracy or performance of the estimate. In the algorithm described above, each worker batch is assigned to each task batch; and between batches, each worker is assigned to each task with probability $p_{obs} \in (0, 1]$. Therefore, in principle, there is a nonzero probability that a worker could be assigned to all T tasks; and similarly for each task, there is a nonzero probability it could be assigned to all W workers. As a reader will notice from our main result, the accuracy of the algorithm depends on the batch size and the number of worker batches assigned to a given task batch. For a given accuracy α, let the corresponding desired batch size be $L(\alpha)$, and let the number of assigned worker batches be $B(\alpha)$. In the algorithm described above, the number of involved workers is $W(\alpha) = L(\alpha)B(\alpha)$, since we assign all worker batches to all task batches. However, if the total number of workers in the system is much larger than $W(\alpha)T/L(\alpha) = TB(\alpha)$, then for each task batch, we can select a different subset of $W(\alpha)$ workers from the population, form $B(\alpha)$ batches of workers each of size $L(\alpha)$ from this subset, and assign these worker batches *only* to the current selected task batch. Then the number of tasks assigned to a worker is guaranteed to be no more than $L = L(\alpha)$. This will make sure

that both the number of tasks assigned to a worker and number of workers assigned to a task is bounded, as desired.

4.1 Choice of Subroutine ESTIMATOR

In the existing literature of matrix completion, graphon estimation, and latent variable models, there are many proposed options for the subroutine ESTIMATOR , each providing different convergence rates on the error under different regularity assumptions on F. If F is piecewise constant such that it can be modeled with finitely many worker types and finitely many task types, then results in the stochastic block model literature (cf. [1, 2, 7]) would provide suitable options for ESTIMATOR . If F is associated to some function whose expected row and column sums are strictly monotonic with respect to the task and worker types, then results in graphon estimation (cf. [4, 5, 24]) would provide options for ESTIMATOR . If F is associated to some Lipschitz function, then results in graphon estimation and matrix completion (cf. [6, 26]) would provide good options for ESTIMATOR . Although we will present our results in the general setting for any choice of ESTIMATOR , we would like to specifically highlight the work of [6] which uses a spectral method, and [26] which uses a neighborhood smoothing estimator.

We summarize the desired property of the ESTIMATOR algorithm that will be satisfied under the Condition 2.2 by different algorithms known in literature as discussed above. Note that many of the sparse matrix or graphon results in the literature present the error for estimating the matrix G/p_{obs}, while we have presented the error bound in terms of the estimation of the matrix G which has been scaled for the sparse sampling p_{obs}. The results are equivalent and a simple constant scaling of one equals the other. We will make brief remarks on known results for different choices of ESTIMATOR that satisfy either Property 4.1 or 4.2 under conditions (1), (2a), (2b) and (3) of Condition 2.2.

Property 4.1. *Let the data model for M satisfy Condition 2.1 and one of the regularity conditions in Condition 2.2. Then the output of* ESTIMATOR(M) *satisfies*

$$\mathbb{E}[\text{MSE}] = \mathbb{E}\left[\frac{1}{L^2}\sum_{(i,j)\in[L]\times[L]}(\hat{G}_{ij} - G_{ij})^2\right] \le p_{obs}^2\zeta(L;p_{obs}),$$

for some function $\zeta(L;p_{obs}) = o_L(1)$, i.e. $\limsup_{L\to\infty}\zeta(L;p_{obs}) = 0$, and for some appropriate constant choice of p_{obs}.

We provide justification for the existence of such an ESTIMATOR under various regularity conditions in Condition 2.2 that lead to Property 4.1 by recalling appropriate results from the literature. To start with, under Condition 2.2(2a), which assumes a Lipschitz function, Chatterjee's USVT estimator achieves an expected MSE which decays as $\zeta(L;p_{obs}) = CL^{-1/3}p_{obs}^{-1/2}$, for a sampling probability of $p_{obs} = \omega(L^{-2/3})$ [6]. While choosing a smaller p_{obs} may lead to fewer samples collected per batch of task and workers, due to the noisier estimate and looser bound on the MSE, we may need to use more batches of workers to increase precision.

The Condition 2.2(1) corresponds to a low-rank assumption. Let r be the rank of the expected matrix. By using the USVT estimator with $p_{obs} = L^{-1+\epsilon}$ for some $\epsilon > 0$, the Property 4.1 is satisfied with $\zeta(L;p_{obs}) = Cr^{1/2}L^{-1/2}p_{obs}^{-1/2}$ for some constant C [6]. This immediately applies to the scenario where Condition 2.2(3) is satisfied. This is because Condition 2.2(3) states that there are finitely many distinct rows or columns of the expected matrix. Clearly, for such a matrix the rank is no larger than the number of distinct rows or columns. Therefore, the matrix is low rank in this case, where the rank is bounded by the minimum of the number of distinct rows (k_1) and columns (k_2).

Under Condition 2.2(2b), which enforces monotonicity of the expected row or column sum according to some underlying function f, we could use methods from graphon estimation to estimate G . For example, the methods proposed in [4, 5] essentially implement a sort-and-smooth paradigm, in which they first sort the rows and columns by the sum of the associated entries in the row or column, and then compute estimates for some entry (i, j) by averaging over observed entries (i', j') for which the row sums of i and i' are similar, and the column sums of j and j' are similar. In order to bound the error of the smoothing step, additional local smoothness or regularity conditions for f must hold. In essence, the monotonicity condition guarantees that there is a unique representation for rows and columns (i.e. the ordering), which can be directly estimated from the data via sorting the row and column sums. The result as presented is framed in the symmetric matrix case, however, it would be straightforward to show that the results should extend to an asymmetric matrix estimation setting in which the monotonicity condition holds for each dimension respectively. Under the dense sampling setting of $p_{obs} = 1$ and additionally assuming that f is Lipschitz, [5] provides a MSE bound which decays as $O(\log n/n)$. In the general sparse sampling setting, [4] proves consistency for any measureable function f, where the rate of convergence of the MSE bound depends on local regularity properties of f.

A bound on the expected mean squared error does not guarantee that the error across entries is uniform. The analysis for some estimators leads to a mildly stronger statement which bounds the expected mean absolute error (MAE) of each row, which is the form of the bound that our analysis naturally depends on.

Property 4.2. *Let the data model for M satisfy Condition 2.1 and one of the regularity conditions in Condition 2.2. Then, for each $i \in [L]$, the output of* ESTIMATOR(M) *satisfies*

$$\mathbb{E}\left[\left|\frac{1}{L}\sum_{j\in[L]}(\hat{G}_{ij} - G_{ij})\right|\right] \le p_{obs}\xi(L;p_{obs}) \text{ for all } i \in [L],$$

for some function $\xi(L;p_{obs}) = o_L(1)$, i.e. $\limsup_{L\to\infty}\xi(L;p_{obs}) = 0$, and for some appropriate constant choice of p_{obs}.

Under regularity condition 2.2(2a), which assumes a Lipschitz function, the neighborhood smoothing estimator of [26] achieves an MSE bound of $\zeta(L;1) = C(\log n/n)^{1/2}$ for some constant C, assuming the dense sampling regime of $p_{obs} = 1$. Although their theorem is presented in terms of a bound on the MSE, delving further into their lemmas and proof reveals that their analysis also implies that the expected row mean absolute error (MAE) decays as $\xi(L;1) = C(\log L/L)^{1/4}$. Their analysis relies on showing that for every row i' which the algorithm determines to be a "neighbor" of row i, the mean squared difference of the entries in the two rows

is bounded with high probability, i.e. $\frac{1}{W}\sum_{j\in[W]}(G_{ij} - G_{i'j})^2 = O((\log n/n)^{1/2})$. Therefore, a bound on the mean absolute error of all rows follows from this intermediate lemma, since the final estimate is an average over entries gathered from rows which were determined to be "neighbors". Although their result is stated for the symmetric matrix setting, one can verify that their method and analysis also hold in the asymmetric matrix setting as long as the number of columns and rows are linearly proportional to each other. A simple way to see this is to transform the estimation problem from an asymmetric to symmetric by defining a new $TW \times TW$ symmetric data matrix $M' = \begin{bmatrix} 0 & M \\ M^T & 0 \end{bmatrix}$, and associated expected response matrix $G' = \begin{bmatrix} 0 & G \\ G^T & 0 \end{bmatrix}$.

5 MAIN RESULTS

As our algorithm hinges upon the choice of the subroutine ESTIMATOR , our analysis will similarly depend on the particular theoretical guarantees which are associated with the choice of ESTIMATOR . In order to estimate a task correctly, we will need the estimated response $\mathbb{E}[\hat{G}_{ij}]$ produced by ESTIMATOR to be close to the true value G_{ij}. Therefore, we will define the set of "bad" tasks to be the tasks for which the bias in the estimated response is larger than $p_{obs}\delta_i/2$,

$$\mathcal{T}_0 := \left\{ i \in [T] \; s.t. \; \left| \frac{1}{W} \sum_{j\in[W]} \left(\mathbb{E}[\hat{G}_{ij}] - G_{ij} \right) \right| \geq \frac{p_{obs}\delta_i}{2} \right\}.$$

As we recall from (7)-(9), the true answer for task i corresponds to the sign of the average over entries in the i-th row of G, whose magnitude is equal to $p_{obs}\delta_i$. Therefore, tasks with smaller δ_i require more precise estimates in order to distinguish the sign correctly, which motivates why the error threshold to determine each task's membership in $\mathcal{T}_0$ is chosen proportional to its gap δ_i. We will first present the general error bound as a function of this set $\mathcal{T}_0$, and then we will follow by showing that having an upper bound on either the MSE or row MAE of ESTIMATOR will each lead to different conditions on this set of "bad" tasks $\mathcal{T}_0$.

The key theorem provides an upper bound on the probability that a task i is estimated incorrectly, for tasks which are in the complement of $\mathcal{T}_0$.

Theorem 5.1. *For any task $i \notin \mathcal{T}_0$ such that $\left(\frac{1}{W}\sum_j F_{ij}\right) \in \left(\frac{1}{2}, 1\right]$, the estimate produced by our algorithm achieves*

$$\mathbb{P}(\hat{a}_i \neq a_i) \leq 2\exp\left(-\frac{W\delta_i^2}{8L}\right).$$

We then use this basic result to provide bounds on the loss function in expectation and with high probability, which follow from direct applications of Theorem 5.1.

Corollary 5.2. *Assume that for all $i \in [T]$, $\left(\frac{1}{W}\sum_j F_{ij}\right) \in \left(\frac{1}{2}, 1\right]$. The expected fraction of incorrect estimates produced by our algorithm is bounded above by*

$$\mathbb{E}[\mathcal{L}_1] \leq \frac{|\mathcal{T}_0|}{T} + \frac{1}{T}\sum_{i\notin\mathcal{T}_0} 2\exp\left(-\frac{W\delta_i^2}{8L}\right),$$

and the expected weighted loss is bounded by

$$\mathbb{E}[\mathcal{L}_\theta] \leq \frac{1}{T}\sum_{i\in\mathcal{T}_0}\theta(i) + \frac{1}{T}\sum_{i\notin\mathcal{T}_0} 2\theta(i)\exp\left(-\frac{W\delta_i^2}{8L}\right).$$

Our bounds are a function of δ_i and $\theta(i)$. The analysis provided in [16], for example, requires that for all tasks i, δ_i is bounded away from zero by some positive constant. Our results show that in fact the expected $\mathcal{L}_1$ loss is a function of the moment generating function of the the distribution over δ_i; hence it is okay for a minuscule fraction of tasks to actually have $\delta_i = 0$!

If we wanted a bound on the loss with high probability, we could choose W to be growing with T logarithmically, leading to the following result.

Corollary 5.3. *Assume that $\left(\frac{1}{W}\sum_j F_{ij}\right) \in \left(\frac{1}{2}, 1\right]$ for all tasks $i \in [T]$, and assume*

$$W \geq \frac{8L\ln(T)^{3/2}}{\min_{i\notin\mathcal{T}_0}\delta_i^2}.$$

With probability at least $1 - 2\exp(-0.5\ln(T)^{3/2})$, all tasks in the complement of $\mathcal{T}_0$ are estimated correctly, such that

$$\mathcal{L}_1 \leq \frac{|\mathcal{T}_0|}{T} \text{ and } \mathcal{L}_\theta \leq \frac{1}{T}\sum_{i\in\mathcal{T}_0}\theta(i).$$

In the remainder of the section, we shall state how Properties 4.1 and 4.2 lead to bounds on $|\mathcal{T}_0|$ and hence bound the overall performance implied by the above stated results.

5.1 Results when bounding $|\mathcal{T}_0|$ via Property 4.2

The Property 4.2 provides the following bound on $|\mathcal{T}_0|$.

Lemma 5.4. *If* ESTIMATOR *satisfies Property 4.2,*

$$\mathcal{T}_0 \subseteq \{i \in [L] : \delta_i \leq 2\xi(L; p_{obs})\}.$$

As a function of the distribution over the gaps δ_i over all tasks, we can choose L as large as we need to reduce the set of bad tasks $\mathcal{T}_0$ as small as we want. The results of Lemma 5.4 imply that $\min_{i\notin\mathcal{T}_0}\delta_i \geq 2\xi(L; p_{obs})$ and $|\mathcal{T}_0| \leq T\hat{\mathbb{P}}(\delta_i \leq 2\xi(L; p_{obs}))$ where $\hat{\mathbb{P}}$ denotes the probability under the empirical distribution of tasks. By plugging these into Corollaries 5.2 and 5.3, we obtain the following result.

Theorem 5.5. *Assume that for all $i \in [T]$, $\left(\frac{1}{W}\sum_j F_{ij}\right) \in \left(\frac{1}{2}, 1\right]$, and* ESTIMATOR *satisfies Property 4.2. The expected fraction of incorrect estimates produced by our algorithm is bounded above by*

$$\mathbb{E}[\mathcal{L}_1] \leq \hat{\mathbb{P}}(\delta_i \leq 2\xi(L; p_{obs})) + 2\exp\left(-\frac{W\xi(L; p_{obs})^2}{2L}\right).$$

For some δ_0 which satisfies $\hat{\mathbb{P}}(\delta_i \leq \delta_0) \leq \alpha$, if we choose $L = \xi^{-1}(\delta_0/2; p_{obs})$ and $W = 8L\ln(T)^{3/2}/\delta_0^2$, we can guarantee that $\mathcal{L}_1 \leq \alpha$ with probability at least $1 - 2\exp(-0.5\ln(T)^{3/2})$ with an expected number of solicited responses per task of

$$p_{obs}W = \frac{8p_{obs}}{\delta_0^2}\xi^{-1}\left(\frac{\delta_0}{2}; p_{obs}\right)\ln(T)^{3/2}.$$

This expression depends on the function ξ, which is the error bound for the subroutine *ESTIMATOR*, and will depend on the sample sparsity p_{obs} as well.

For example, under the Lipschitz regularity condition 2.2(2a), we can specifically plug in the results from choosing ESTIMATOR to be the neighborhood smoothing estimator of [26], with $p_{obs} = 1$ and $\xi(L;1) = C\,(\ln(L)/L)^{1/4} \le C' L^{-1/(4+\epsilon)}$ for constants C, C' and any $\epsilon > 0$ to obtain the following.

Corollary 5.6. *Assume that for all $i \in [T]$, $\left(\frac{1}{W}\sum_j F_{ij}\right) \in \left(\frac{1}{2}, 1\right]$. Assume that the data model satisfies Condition 2.1 and regularity Condition 2.2(2a) with $\Omega_T = \Omega_W = [0,1]$, and $p_{obs} = 1$. Choose* ESTIMATOR *to be the neighborhood smoothing estimator of [26]. For any $\epsilon > 0$, and for some δ_0 which satisfies $\hat{\mathbb{P}}(\delta_i \le \delta_0) \le \alpha/2$, we can guarantee that $\mathbb{E}[\mathcal{L}_1] \le \alpha$ using an expected number of solicited responses per task of*

$$W = O\left(\delta_0^{-6-\epsilon} \ln\left(\frac{4}{\alpha}\right)\right).$$

We can guarantee that $\mathcal{L}_1 \le \alpha/2$ with probability at least $1 - 2\exp(-0.5\ln(T)^{3/2})$ using an expected number of solicited responses per task of

$$W = O\left(\delta_0^{-6-\epsilon} \ln(T)^{3/2}\right).$$

If the gap of all tasks were bounded away from zero, i.e. $\delta_i \ge \delta_* > 0$ for all i, then we can in fact guarantee with high probability that all tasks are estimated correctly and the loss is zero. Similar results hold for the general weighted loss function by incorporating the distribution over $\theta(i)$.

Corollary 5.7. *Assume that for all $i \in [T]$, $\left(\frac{1}{W}\sum_j F_{ij}\right) \in \left(\frac{1}{2}, 1\right]$. Assume that the data model satisfies Condition 2.1 and regularity Condition 2.2(2a) with $\Omega_T = \Omega_W = [0,1]$, $p_{obs} = 1$. Choose* ESTIMATOR *to be the neighborhood smoothing estimator of [26]. For any $\epsilon > 0$, and for some δ_θ which satisfies*

$$\frac{1}{T}\sum_{i\in[T]} \theta(i)\mathbb{I}(\delta_i \le \delta_\theta) \le \alpha,$$

we can guarantee that $\mathcal{L}_\theta \le \alpha$ with probability at least $1 - 2\exp(-0.5\ln(T)^{3/2})$ using an expected number of solicited responses per task of

$$W = O\left(\delta_\theta^{-6-\epsilon} \ln(T)^{3/2}\right).$$

If the penalty $\theta(i)$ is smaller for tasks which have smaller gap δ_i, and if the penalties are normalized such that $\frac{1}{T}\left(\sum_i \theta(i)\right) = 1$, then we can show that $\delta_\theta \ge \delta_0$, such that it would take more responses per task to reduce $\mathcal{L}_1$ below some threshold α as opposed to the weighted loss $\mathcal{L}_\theta$.

We could alternatively choose a sparser matrix estimation method, for which $p_{obs} = o_L(1)$ such that the solicited responses per batch would be fewer, but $\xi(L, p_{obs})$ may decrease at a slower rate, thus requiring a larger number of batches to reduce the variance of the estimate.

5.2 Results when bounding $|\mathcal{T}_0|$ via Property 4.1

Naturally, the Property 4.2 is stronger as it implies Property 4.1. However, we can still get useful bounds using Property 4.2.

Lemma 5.8. *If* ESTIMATOR *satisfies Property 4.1, then $\mathcal{T}_0$ must satisfy*

$$\sum_{i\in\mathcal{T}_0} \delta_i \le 2T\zeta(L)^{1/2}.$$

This bound is less restrictive in the sense that it would allow for a set $\mathcal{T}_0$ which includes a variety of tasks with large or small gap δ_i as long as the sum satisfies the constraint. Obtaining a concrete bound on the expected loss as presented in Corollary 5.2 corresponds to solving an instance of the 0-1 Knapsack problem, in which we would compute the set $\mathcal{T}_0$ which maximizes the bound while satisfying the constraint in Lemma 5.8. This reduction follows from the fact that the upper bounds on the loss can be reduced to a sum of values associated to tasks in $\mathcal{T}_0$. The value of each task is equal to its contribution to the bound on the loss function, and the weight of each task is δ_i. We would like to maximize the value of set $\mathcal{T}_0$ while constraining the total weight of set $\mathcal{T}_0$. We can use a greedy method to obtain a factor 2 approximation for this bound by sorting the tasks in decreasing order of the ratio of the value to weight of each task. When the values are nonincreasing in the weights δ_i, the greedy solution is in fact optimal. We will present the results for the setting in which the gaps of all tasks are bounded away from zero, i.e. $\delta_i \ge \delta_* > 0$ for all i.

Theorem 5.9. *Assume that for all $i \in [T]$, $\left(\frac{1}{W}\sum_j F_{ij}\right) \in \left(\frac{1}{2}, 1\right]$, $\delta_i \ge \delta_* > 0$, and* ESTIMATOR *satisfies Property 4.1. Then*

$$\mathbb{E}[\mathcal{L}_1] \le \frac{2\zeta(L;p_{obs})^{1/2}}{\delta_*} + 2\exp\left(-\frac{W\delta_*^2}{8L}\right).$$

Therefore, if $L = \zeta^{-1}\left(\frac{\alpha^2\delta_*^2}{4}; p_{obs}\right)$, and $W = \frac{8L\ln(T)^{3/2}}{\delta_*^2}$, we can guarantee that $\mathcal{L}_1 \le \alpha$ with probability at least $1 - 2\exp(-0.5\ln(T)^{3/2})$, with an expected number of solicited responses per task of

$$p_{obs}W = \frac{8p_{obs}}{\delta_*^2}\zeta^{-1}\left(\frac{\alpha^2\delta_*^2}{4}; p_{obs}\right)\ln(T)^{3/2}.$$

In the case, regularity Condition 2.2(2a) is satisfied, i.e. F is consistent with some underlying Lipschitz function f, we can choose ESTIMATOR to be the USVT estimator of [6]. For some $p_{obs} = \omega(L^{-2/3})$, the provided analysis upper bounds the MSE by $\zeta(L;p_{obs}) = CL^{-1/3}p_{obs}^{-1/2}$ for some constant C. We shall choose $p_{obs} = 1$, such that $\zeta(L;1) = CL^{-1/3}$. The following corollary results from plugging in these expressions into Theorem 5.9.

Corollary 5.10. *Assume that for all $i \in [T]$, $\left(\frac{1}{W}\sum_j F_{ij}\right) \in \left(\frac{1}{2}, 1\right]$, $\delta_i \ge \delta_* > 0$, the data model satisfies Condition 2.1 and regularity Condition 2.2(2a) with $\Omega_T = \Omega_W = [0,1]$, and* ESTIMATOR *is the USVT estimator of [6] with $p_{obs} = 1$. We can guarantee that $\mathbb{E}[\mathcal{L}_1] \le \alpha$ using an expected number of solicited responses per task of*

$$W = O\left(\alpha^{-6}\delta_*^{-8} \ln\left(\frac{4}{\alpha}\right)\right).$$

We can guarantee that $\mathcal{L}_1 \le \alpha/2$ with probability at least $1 - 2\exp(-0.5\ln(T)^{3/2})$ using an expected number of solicited responses per task of

$$W = O\left(\alpha^{-6}\delta_*^{-8} \ln(T)^{3/2}\right).$$

In the case when our model satisfies regularity condition 2.2(1), i.e. F is low rank with rank r, then we can choose ESTIMATOR to be the USVT estimator of [6]. For $p_{obs} = L^{-1+\epsilon}$ for some $\epsilon > 0$, the provided analysis upper bounds the MSE by $\zeta(L; p_{obs}) = Cr^{1/2}L^{-1/2}p_{obs}^{-1/2} = Cr^{1/2}L^{-\epsilon/2}$ for some constant C. The following corollary results from plugging in these expressions into Theorem 5.9.

Corollary 5.11. *Assume that for all* $i \in [T]$, $\left(\frac{1}{W}\sum_j F_{ij}\right) \in \left(\frac{1}{2}, 1\right]$, $\delta_i \geq \delta_* > 0$, *the data model satisfies Condition 2.1 and regularity Condition 2.2(1) with rank* r, *and* ESTIMATOR *is the USVT estimator of [6] with* $p_{obs} = L^{-1+\epsilon}$. *We can guarantee that* $\mathbb{E}[\mathcal{L}_1] \leq \alpha$ *using an expected number of solicited responses per task of*

$$p_{obs}W = O\left(r\alpha^{-4}\delta_*^{-6}\ln\left(\frac{4}{\alpha}\right)\right).$$

We can guarantee that $\mathcal{L}_1 \leq \alpha/2$ *with probability at least* $1 - 2\exp(-0.5\ln(T)^{3/2})$ *using an expected number of solicited responses per task of*

$$p_{obs}W = O\left(r\alpha^{-4}\delta_*^{-6}\ln(T)^{3/2}\right).$$

For the more general weighted loss function, computing an upper bound involves maximizing $\frac{1}{T}\sum_{i\in\mathcal{T}_0}\theta(i)$ subject to $\sum_{i\in\mathcal{T}_0}\delta_i \leq 2T\zeta(L; p_{obs})^{1/2}$, which does not have a simple closed form expression in general without enforcing further assumptions on the weight function $\theta(\cdot)$. We can use the factor 2 approximation in order to compute similar bounds on the number of solicited responses sufficient to reduce the $\mathcal{L}_\theta$ loss below some value α, which would follow essentially the same steps as the corollaries presented above.

As Corollaries 5.6 to 5.11 illustrate, given different regularity conditions on the data model, we can choose appropriate ESTIMATOR methods from the literature, and plug in the error bounds to directly produce bounds on the loss achieved by our algorithm.

6 PROOFS

Recall that we have defined the set of "bad" tasks $\mathcal{T}_0$ according to

$$\mathcal{T}_0 := \left\{ i \in [T] \; s.t. \; \left| \frac{1}{W}\sum_{j\in[W]} \left(\mathbb{E}[\hat{G}_{ij}] - G_{ij}\right) \right| \geq \frac{p_{obs}\delta_i}{2} \right\}.$$

Proof of Theorem 5.1. By assumption, $\left(\frac{1}{W}\sum_j F_{ij}\right) \in \left(\frac{1}{2}, 1\right]$, which implies that $\text{sign}(\frac{1}{W}\sum_j G_{ij}) = a_i$. Recall from the algorithm that

$$\hat{a}_i = \text{sign}\left(\frac{1}{W}\sum_{j\in[W]}\hat{G}_{ij}\right),$$

such that $\hat{a}_i = a_i$ if and only if $\frac{1}{W}\sum_{j\in[W]}\hat{G}_{ij}$ has the same sign as $\frac{1}{W}\sum_{j\in[W]}G_{ij}$. Therefore the event

$$\left|\frac{1}{W}\sum_{j\in[W]}\hat{G}_{ij} - \frac{1}{W}\sum_{j\in[W]}G_{ij}\right| < \left|\frac{1}{W}\sum_{j\in[W]}G_{ij}\right| = p_{obs}\delta_i,$$

implies that $\hat{a}_i = a_i$. Therefore, the probability of an incorrect response is bounded by

$$\mathbb{P}(\hat{a}_i \neq a_i) \leq \mathbb{P}\left(\left|\frac{1}{W}\sum_{j\in[W]}(\hat{G}_{ij} - G_{ij})\right| \geq p_{obs}\delta_i\right).$$

We can rewrite this expression as

$$\left|\frac{1}{W}\sum_{j\in[W]}\left(\hat{G}_{ij} - G_{ij}\right)\right| \leq \left|\frac{1}{W}\sum_{j\in[W]}\left(\hat{G}_{ij} - \mathbb{E}[\hat{G}_{ij}]\right)\right| + \left|\frac{1}{W}\sum_{j\in[W]}\left(\mathbb{E}[\hat{G}_{ij}] - G_{ij}\right)\right|.$$

By assuming that $i \notin \mathcal{T}_0$, the second term is bounded by $p_{obs}\delta_i/2$, such that

$$\mathbb{P}(\hat{a}_i \neq a_i) \leq \mathbb{P}\left(\left|\frac{1}{W}\sum_{j\in[W]}\left(\hat{G}_{ij} - \mathbb{E}[\hat{G}_{ij}]\right)\right| \geq \frac{p_{obs}\delta_i}{2}\right).$$

Recall that $\hat{G}_{ij}$ is the output of ESTIMATOR given the response matrix M. Therefore, $\hat{G}_{ij}$ may not be independent across entries, since the estimates for different entries could have been derived from overlapping sets of observed responses. However, because our algorithm applied ESTIMATOR separately across different batches of tasks and workers, $\hat{G}_{ij}$ and $\hat{G}_{i'j'}$ are independent as long as either i and i' are in different task batches, or j and j' are in different worker batches. Therefore, we can write $\sum_{j\in[W]}\hat{G}_{ij}$ as a sum of independent random variables by summing over each of the worker batches.

$$\frac{1}{W}\sum_{j\in[W]}\hat{G}_{ij} = \frac{L}{W}\sum_{b\in[W/L]}\left(\frac{1}{L}\sum_{j\in\mathcal{W}_b}\hat{G}_{ij}\right).$$

Since G_{ij} by definition is bounded between $[-p_{obs}, p_{obs}]$, we can assume without loss of generality that $\frac{1}{L}\sum_{j\in\mathcal{W}_b}\hat{G}_{ij} \in [-p_{obs}, p_{obs}]$. Therefore, by Hoeffding's inequality for bounded random variables,

$$\mathbb{P}\left(\left|\frac{1}{W}\sum_{j\in[W]}\left(\hat{G}_{ij} - \mathbb{E}[\hat{G}_{ij}]\right)\right| \geq \frac{p_{obs}\delta_i}{2}\right) \leq 2\exp\left(-\frac{W\delta_i^2}{8L}\right),$$

which completes the proof. □

Proof of Corollary 5.2. This result follows from a direct application of Theorem 5.1. The expected $\mathcal{L}_1$ loss is bounded by

$$\mathbb{E}\left[\frac{1}{T}\sum_{i\in\mathcal{T}}\mathbb{I}(\hat{a}_i \neq a_i)\right] = \frac{1}{T}\sum_{i\in\mathcal{T}}\mathbb{P}(\hat{a}_i \neq a_i) \leq \frac{|\mathcal{T}_0|}{T} + \frac{1}{T}\sum_{i\notin\mathcal{T}_0}2\exp\left(-\frac{W\delta_i^2}{8L}\right).$$

Similarly, the expected $\mathcal{L}_\theta$ loss is bounded by

$$\mathbb{E}\left[\frac{1}{T}\sum_{i\in\mathcal{T}}\theta(i)\mathbb{I}(\hat{a}_i \neq a_i)\right] = \frac{1}{T}\sum_{i\in\mathcal{T}}\theta(i)\mathbb{P}(\hat{a}_i \neq a_i) \leq \frac{1}{T}\sum_{i\in\mathcal{T}_0}\theta(i) + \frac{1}{T}\sum_{i\notin\mathcal{T}_0}2\theta(i)\exp\left(-\frac{W\delta_i^2}{8L}\right).$$

□

Proof of Corollary 5.3. This result follows from a direct application of Theorem 5.1. The probability that any task in the complement of $\mathcal{T}_0$ is estimated incorrectly is upper bounded by

$$\mathbb{P}\left(\cup_{i\notin\mathcal{T}_0}\{\hat{a}_i \neq a_i\}\right) \leq 2\sum_{i\notin\mathcal{T}_0}\exp\left(-\frac{W\delta_i^2}{8L}\right) \leq 2\exp\left(-\frac{W\left(\min_{i\notin\mathcal{T}_0}\delta_i^2\right)}{8L}+\ln(T)\right).$$

Therefore, if we choose W large enough such that

$$W \geq \frac{8L\ln(T)^{3/2}}{\min_{i\notin\mathcal{T}_0}\delta_i^2},$$

then with probability at least

$$1-2\exp\left(-\ln(T)^{3/2}\left(1-\ln(T)^{-1/2}\right)\right),$$

all tasks in the complement of $\mathcal{T}_0$ are estimated correctly, which implies that

$$\mathcal{L}_1 \leq \frac{|\mathcal{T}_0|}{T} \text{ and } \mathcal{L}_\theta \leq \frac{1}{T}\sum_{i\in\mathcal{T}_0}\theta(i).$$

Without loss of generality, we can assume that $T \geq 55$, such that $\left(1-\ln(T)^{-1/2}\right) \geq 0.5$. □

6.1 Proofs when bounding $|\mathcal{T}_0|$ via Property 4.2

Proof of Lemma 5.4. If ESTIMATOR satisfies Property 4.2, then for all $i \in [L]$, then

$$\left|\frac{1}{W}\sum_{j\in[W]}\left(\mathbb{E}[\hat{G}_{ij}]-G_{ij}\right)\right| \leq \frac{L}{W}\sum_{b\in[W/L]}\left|\mathbb{E}\left[\frac{1}{L}\sum_{j\in\mathcal{W}_b}\left(\hat{G}_{ij}-G_{ij}\right)\right]\right| \leq \frac{L}{W}\sum_{b\in[W/L]}\mathbb{E}\left[\left|\frac{1}{L}\sum_{j\in\mathcal{W}_b}\left(\hat{G}_{ij}-G_{ij}\right)\right|\right] \leq p_{obs}\xi(L;p_{obs}).$$

Therefore, it follows that $\mathcal{T}_0 \subseteq \left\{i\in[L]:\delta_i \leq 2\xi(L;p_{obs})\right\}$. □

Proof of Theorem 5.5. The results of Lemma 5.4 imply that $\min_{i\notin\mathcal{T}_0}\delta_i \geq 2\xi(L;p_{obs})$ and $|\mathcal{T}_0| \leq T\hat{\mathbb{P}}(\delta_i \leq 2\xi(L;p_{obs}))$, where $\hat{\mathbb{P}}$ denotes the probability under the empirical distribution of tasks. Therefore, by plugging into Corollary 5.2, it follows that

$$\mathbb{E}[\mathcal{L}_1] \leq \hat{\mathbb{P}}(\delta_i \leq 2\xi(L;p_{obs})) + 2\exp\left(-\frac{W\xi(L;p_{obs})^2}{2L}\right).$$

For some δ_0 which satisfies $\hat{\mathbb{P}}(\delta_i \leq \delta_0) \leq \alpha$, if we choose $L = \xi^{-1}(\delta_0/2;p_{obs})$ and $W = 8L\ln(T)^{3/2}/\delta_0^2$, then we can plug into Corollary 5.3 to show that with probability at least $1-2\exp(-0.5\ln(T)^{3/2})$, the $\mathcal{L}_1$ loss is upper bounded by α. This choice of parameters results in an expected number of solicited responses per task of

$$p_{obs}W = \frac{8p_{obs}}{\delta_0^2}\xi^{-1}\left(\frac{\delta_0}{2};p_{obs}\right)\ln(T)^{3/2}.$$

□

Proof of Corollary 5.6. If we choose ESTIMATOR to be the neighborhood smoothing estimator of [26], then $p_{obs} = 1$ and $\xi(L;1) = C\left(\frac{\ln(L)}{L}\right)^{1/4} \leq C'L^{-1/(4+\epsilon)}$ for constants C, C' and any $\epsilon > 0$. For some δ_0 which satisfies $\hat{\mathbb{P}}(\delta_i \leq \delta_0) \leq \alpha/2$, we can choose $L = \xi^{-1}(\delta_0/2;1) = (2C'/\delta_0)^{4+\epsilon}$ to ensure that $|\mathcal{T}|/T \leq \alpha/2$. Then the final result follows from applying these expressions to Theorem 5.5. □

Proof of Corollary 5.7. We have assumed that $p_{obs} = 1$, and that ESTIMATOR is chosen to be the neighborhood smoothing estimator of [26], such that $\xi(L;1) = C\left(\frac{\ln(L)}{L}\right)^{1/4} \leq C'L^{-1/(4+\epsilon)}$ for constants C, C' and any $\epsilon > 0$. The results of Lemma 5.4 imply that $\sum_{i\in\mathcal{T}_0}\theta(i) \leq \sum_{i\in[T]}\theta(i)\mathbb{I}(\delta_i \leq 2\xi(L;1))$. Therefore, for some δ_θ which satisfies

$$\frac{1}{T}\sum_{i\in[T]}\theta(i)\mathbb{I}(\delta_i \leq \delta_\theta) \leq \alpha,$$

if we choose $L = \xi^{-1}(\delta_\theta/2;1) = (2C'/\delta_\theta)^{4+\epsilon}$ and $W = 8L\ln(T)^{3/2}/\delta_\theta^2$, it follows from Corollary 5.3 that with probability at least $1-2\exp(-0.5\ln(T)^{3/2})$ the weighted loss $\mathcal{L}_\theta$ is upper bounded by α. This choice of parameters results in an expected number of solicited responses per task of

$$W = O\left(\delta_\theta^{-6-\epsilon}\ln(T)^{3/2}\right).$$

□

6.2 Proofs when bounding $|\mathcal{T}_0|$ via Property 4.1

We can also get more specific bounds when the guarantees on ESTIMATOR are provided in the form of a MSE bound, however the result will be looser.

Proof of Lemma 5.8. If ESTIMATOR satisfies Property 4.1, then

$$\begin{aligned}
&\frac{1}{T}\sum_{i\in[T]}\left|\frac{1}{W}\sum_{j\in[W]}\left(\mathbb{E}[\hat{G}_{ij}]-G_{ij}\right)\right| \\
&\leq \frac{L}{T}\sum_{a\in[T/L]}\frac{1}{L}\sum_{i\in\mathcal{T}_a}\left|\frac{L}{W}\sum_{b\in[W/L]}\mathbb{E}\left[\frac{1}{L}\sum_{j\in\mathcal{W}_b}\left(\hat{G}_{ij}-G_{ij}\right)\right]\right| \\
&\leq \frac{L}{T}\sum_{a\in[T/L]}\frac{L}{W}\sum_{b\in[W/L]}\mathbb{E}\left[\frac{1}{L^2}\sum_{i\in\mathcal{T}_a}\sum_{j\in\mathcal{W}_b}\left|\hat{G}_{ij}-G_{ij}\right|\right] \\
&\leq \frac{L}{T}\sum_{a\in[T/L]}\frac{L}{W}\sum_{b\in[W/L]}\mathbb{E}\left[\frac{1}{L^2}\sqrt{L^2}\left(\sum_{i\in\mathcal{T}_a}\sum_{j\in\mathcal{W}_b}\left(\hat{G}_{ij}-G_{ij}\right)^2\right)^{1/2}\right] \\
&\leq \frac{L}{T}\sum_{a\in[T/L]}\frac{L}{W}\sum_{b\in[W/L]}\mathbb{E}\left[\frac{1}{L^2}\sum_{i\in\mathcal{T}_a}\sum_{j\in\mathcal{W}_b}\left(\hat{G}_{ij}-G_{ij}\right)^2\right]^{1/2} \\
&\leq p_{obs}\zeta(L;p_{obs})^{1/2}.
\end{aligned}$$

Therefore, $\mathcal{T}_0$ must satisfy $\frac{1}{T}\sum_{i\in\mathcal{T}_0}\frac{\delta_i}{2} \leq \zeta(L;p_{obs})^{1/2}$. □

The results of Lemma 5.8 combined with Corollary 5.2 imply that

$$\mathbb{E}[\mathcal{L}_1] \le \max_{\mathcal{T}_0 \subset [T]} \quad \left(\frac{|\mathcal{T}_0|}{T} + \frac{1}{T} \sum_{i \notin \mathcal{T}_0} 2 \exp\left(-\frac{W\delta_i^2}{8L} \right) \right)$$
$$\text{s.t.} \quad \sum_{i \in \mathcal{T}_0} \delta_i \le 2T\zeta(L; p_{obs})^{1/2},$$

and

$$\mathbb{E}[\mathcal{L}_\theta] \le \max_{\mathcal{T}_0 \subset [T]} \quad \left(\frac{1}{T} \sum_{i \in \mathcal{T}_0} \theta(i) + \frac{1}{T} \sum_{i \notin \mathcal{T}_0} 2\theta(i) \exp\left(-\frac{W\delta_i^2}{8L} \right) \right)$$
$$\text{s.t.} \quad \sum_{i \in \mathcal{T}_0} \delta_i \le 2T\zeta(L; p_{obs})^{1/2}.$$

Both of these bounds can be formulated as the solution to an instance of the 0-1 Knapsack problem.

Proof of Theorem 5.9. By Lemma 5.8,

$$\sum_{i \in \mathcal{T}_0} \delta_i \le 2T\zeta(L; p_{obs})^{1/2}.$$

Since $\min_i \delta_i \ge \delta_*$, it follows that $\sum_{i \in \mathcal{T}_0} \delta_i \ge |\mathcal{T}_0|\delta_*$. Therefore, it must follow that $|\mathcal{T}_0| \le 2T\zeta(L; p_{obs})^{1/2}/\delta_*$. Therefore, by plugging into Corollary 5.2, it follows that

$$\mathbb{E}[\mathcal{L}_1] \le \frac{2\zeta(L; p_{obs})^{1/2}}{\delta_*} + 2\exp\left(-\frac{W\delta_*^2}{8L} \right).$$

If $L = \zeta^{-1}\left(\alpha^2\delta_*^2/4; p_{obs}\right)$, then $|\mathcal{T}_0|/T \le \alpha$. By Corollary 5.3, if $W = 8L\ln(T)^{3/2}/\delta_*^2$, then $\mathcal{L}_1 \le \alpha$ with probability at least $1 - 2\exp(-0.5\ln(T)^{3/2})$, with an expected number of solicited responses per task of

$$p_{obs}W = \frac{8p_{obs}}{\delta_*^2} \zeta^{-1}\left(\frac{\alpha^2\delta_*^2}{4}; p_{obs} \right) \ln(T)^{3/2}.$$

□

Proof of Corollary 5.10. In the case, regularity Condition 2.2(2a) is satisfied, i.e. F is consistent with some underlying Lipschitz function f, we can choose ESTIMATOR to be the USVT estimator of [6]. For some $p_{obs} = \omega(L^{-2/3})$, the provided analysis upper bounds the MSE by $\zeta(L; p_{obs}) = CL^{-1/3}p_{obs}^{-1/2}$ for some constant C. We shall choose $p_{obs} = 1$, such that $\zeta(L; 1) = CL^{-1/3}$. Choose the batch size L to be

$$L = \zeta^{-1}\left(\frac{\alpha^2\delta_*^2}{16}; p_{obs} \right) = \frac{2^{12}C^3}{\alpha^6\delta_*^6},$$

and $W = 8L\ln(4/\alpha)/\delta_*^2$. Then, by Theorem 5.9, it follows that $\mathbb{E}[\mathcal{L}_1] \le \alpha$. This choice of parameters leads to an expected number of solicited responses per task of

$$W = O\left(\delta_*^{-8}\alpha^{-6} \ln\left(\frac{4}{\alpha} \right) \right).$$

Furthermore, we can guarantee that $\mathcal{L}_1 \le \alpha/2$ with probability at least $1 - 2\exp(-0.5\ln(T)^{3/2})$ by choosing $W = 8L\ln(T)^{3/2}/\delta_*^2$, as per Corollary 5.3, such that

$$W = O\left(\alpha^{-6}\delta_*^{-8} \ln(T)^{3/2} \right).$$

□

Proof of Corollary 5.11. In the case when our model satisfies regularity condition 2.2(1), i.e. F is low rank with rank r, then we can also choose ESTIMATOR to be the USVT estimator of [6]. For $p_{obs} = L^{-1+\epsilon}$ for some $\epsilon > 0$, the provided analysis upper bounds the MSE by $\zeta(L; p_{obs}) = Cr^{1/2}L^{-1/2}p_{obs}^{-1/2} = Cr^{1/2}L^{-\epsilon/2}$ for some constant C. We choose the batch size L to be

$$L = \zeta^{-1}\left(\frac{\alpha^2\delta_*^2}{16}; L^{-1+\epsilon} \right) = \left(\frac{256C^2r}{\alpha^4\delta_*^4} \right)^{1/\epsilon},$$

and $W = \frac{8L}{\delta_*^2}\ln\left(\frac{4}{\alpha}\right)$, such that

$$p_{obs}W = O\left(L^\epsilon\delta_*^{-2} \ln\left(\frac{4}{\alpha} \right) \right) = O\left(\alpha^{-4}\delta_*^{-6} r \ln\left(\frac{4}{\alpha} \right) \right).$$

Then by Theorem 5.9, it follows that $\mathbb{E}[\mathcal{L}_1] \le \alpha$. Furthermore, we can guarantee that $\mathcal{L}_1 \le \alpha/2$ with probability at least $1 - 2\exp(-0.5\ln(T)^{3/2})$ by choosing $W = 8L\ln(T)^{3/2}/\delta_*^2$, according to Corollary 5.3. This leads to an expected number of solicited responses of

$$p_{obs}W = O\left(r\alpha^{-4}\delta_*^{-6} \ln(T)^{3/2} \right).$$

□

For the more general weighted loss function, computing an upper bound involves maximizing

$$\frac{1}{T} \sum_{i \in \mathcal{T}_0} \theta(i) \qquad \text{subject to} \qquad p_{obs} \sum_{i \in \mathcal{T}_0} \delta_i \le 2T\zeta(L; p_{obs})^{1/2}.$$

This is the classical knapsack problem. We could obtain a 2-approximation by greedily choosing tasks in decreasing order of $\theta(i)/\delta_i$ until the constraint is violated, and comparing the value of the violating task with the value of the sum of previously chosen tasks. Let $\pi : [T] \to [T]$ be a permutation such that $k \le k'$ if and only if $\frac{\theta(\pi(k))}{\delta_{\pi(k)}} \ge \frac{\theta(\pi(k'))}{\delta_{\pi(k')}}$. Then let

$$k_* = \max\left\{ k' : \sum_{k=1}^{k'} \delta_{\pi(k)} \le 2T\zeta(L; p_{obs})^{1/2} \right\}.$$

It follows that

$$\mathbb{E}[\mathcal{L}_\theta] \le \frac{2}{T} \max\left(\theta(\pi(k_* + 1)), \sum_{k=1}^{k_*} \theta(\pi(k)) \right) + \frac{2}{T} \exp\left(-\frac{W\delta_*^2}{8L} \right) \sum_{i \in [T]} \theta(i).$$

Depending on the distribution of θ_i and it's relationship to δ_i, we can similarly compute the necessary L and W to guarantee a desired upper bound on the weighted loss $\mathcal{L}_\theta$.

7 DISCUSSION

In this work, we present a unifying general framework for crowdsourcing, in which the probability that a worker j answers a task i correctly is a generic value F_{ij}, satisfying basic regularity conditions. This is able to encompass different types of model assumptions that have been studied in the literature. Furthermore, it is more expressive, allowing for non-linearities or non-monotonicities in the function. We show that the inference problem can be reduced to finding a subroutine which performs Graphon estimation.

Therefore, we are able to leverage existing work in literature to show that our algorithm and analysis cover settings in which F is either piecewise constant, Lipschitz, monotonic or low rank. We characterize the difficulty of estimating a task by the parameter $\delta_i = \left|\frac{1}{W}\sum_j (2F_{ij} - 1)\right|$, and we present our bounds as a function of the general distribution of δ_i rather than enforcing that δ_i must be bounded away from zero. Our algorithm could naturally be extended to an adaptive setting, as the responses from workers are collected and processed in batches. For example, our results may be improved by considering similar techniques to [17], where the number of worker batches assigned to each task could be selected adaptively to assign more workers only to tasks that are more difficult.

Our key theorem proves that for tasks for which the subroutine ESTIMATOR has a small bias, the probability of an incorrect estimate is bounded above by

$$\mathbb{P}(\hat{a}_i \neq a_i) \leq 2\exp\left(-\frac{W\delta_i^2}{8L}\right).$$

Therefore, it is clear from our bounds that an adaptive scheme may want to choose the number of worker batches, W/L inversely proportional to δ_i^2. As δ_i is not initially known, we could bootstrap and iteratively refine the estimate of δ_i and adaptively decide whether to continue to solicit responses or not.

In this paper, we consider a simple aggregation technique which takes the sum of the estimated expected responses in matrix $\hat{G}$, and relies upon the condition that $\frac{1}{W}\sum_j (2F_{ij} - 1) > \frac{1}{2}$. However, there may be more complex aggregation techniques which could improve the efficiency of the estimator and relax the required assumption on F. For example, consider a worker task pair for which $F_{ij} = 1$. If we somehow could estimate that this worker has perfect accuracy for this task, then we could confidently base the estimate fully on the response of this worker rather than aggregating across other potentially noisy responses. Given an estimate of the expected response matrix G, every vector of task answers $\mathbf{a} \in \{-1, 1\}^T$ directly leads to an estimate of matrix F according to

$$\hat{F} = p_{obs}^{-1}\mathrm{Diag}(\mathbf{a})\hat{G},$$

where $\mathrm{Diag}(\mathbf{a})$ denotes a diagonal matrix whose diagonal entries are the values in $\mathbf{a}$. Then we could choose the answer vector $\mathbf{a}$ which maximizes some notion of regularity over F. This would be similar to the minimax entropy method of [28] and [27], which does not yet have rigorous analysis or provable guarantees. Our framework may provide a natural way to think about the theoretical properties of this and other general aggregation techniques.

REFERENCES

[1] Emmanuel Abbe and Colin Sandon. 2015. Community detection in general stochastic block models: fundamental limits and efficient recovery algorithms. *FOCS* (2015).

[2] Emmanuel Abbe and Colin Sandon. 2015. Recovering communities in the general stochastic block model without knowing the parameters. In *Advances in Neural Information Processing Systems*. 676–684.

[3] Thomas Bonald and Richard Combes. 2016. Crowdsourcing: Low complexity, Minimax Optimal Algorithms. *arXiv preprint arXiv:1606.00226* (2016).

[4] Christian Borgs, Jennifer T Chayes, Henry Cohn, and Shirshendu Ganguly. 2015. Consistent nonparametric estimation for heavy-tailed sparse graphs. *arXiv preprint arXiv:1508.06675* (2015).

[5] Stanley H Chan and Edoardo Airoldi. 2014. A Consistent Histogram Estimator for Exchangeable Graph Models.. In *ICML*. 208–216.

[6] Sourav Chatterjee. 2015. Matrix estimation by universal singular value thresholding. *The Annals of Statistics* 43, 1 (2015), 177–214.

[7] Peter Chin, Anup Rao, and Van Vu. 2015. Stochastic block model and community detection in the sparse graphs: A spectral algorithm with optimal rate of recovery. *arXiv preprint arXiv:1501.05021* 2, 4 (2015).

[8] Nilesh Dalvi, Anirban Dasgupta, Ravi Kumar, and Vibhor Rastogi. 2013. Aggregating crowdsourced binary ratings. In *Proceedings of the 22nd international conference on World Wide Web*. ACM, 285–294.

[9] Alexander Philip Dawid and Allan M Skene. 1979. Maximum likelihood estimation of observer error-rates using the EM algorithm. *Applied statistics* (1979), 20–28.

[10] Chao Gao and Dengyong Zhou. 2013. Minimax optimal convergence rates for estimating ground truth from crowdsourced labels. *arXiv preprint arXiv:1310.5764* (2013).

[11] Arpita Ghosh, Satyen Kale, and Preston McAfee. 2011. Who moderates the moderators?: crowdsourcing abuse detection in user-generated content. In *Proceedings of the 12th ACM conference on Electronic commerce*. ACM, 167–176.

[12] Chien-Ju Ho, Shahin Jabbari, and Jennifer Wortman Vaughan. 2013. Adaptive task assignment for crowdsourced classification. (2013).

[13] Chien-Ju Ho and Jennifer Wortman Vaughan. 2012. Online Task Assignment in Crowdsourcing Markets.. In *Proceedings of the 26th Conference on Artificial Intelligence*, Vol. 12. 45–51.

[14] David R Karger, Sewoong Oh, and Devavrat Shah. 2011. Iterative learning for reliable crowdsourcing systems. In *Advances in neural information processing systems*. 1953–1961.

[15] David R Karger, Sewoong Oh, and Devavrat Shah. 2013. Efficient crowdsourcing for multi-class labeling. *ACM SIGMETRICS Performance Evaluation Review* 41, 1 (2013), 81–92.

[16] David R Karger, Sewoong Oh, and Devavrat Shah. 2014. Budget-optimal task allocation for reliable crowdsourcing systems. *Operations Research* 62, 1 (2014), 1–24.

[17] Ashish Khetan and Sewoong Oh. 2016. Achieving budget-optimality with adaptive schemes in crowdsourcing. In *Advances in Neural Information Processing Systems*. 4844–4852.

[18] Hongwei Li and Bin Yu. 2014. Error rate bounds and iterative weighted majority voting for crowdsourcing. *arXiv preprint arXiv:1411.4086* (2014).

[19] Qiang Liu, Jian Peng, and Alexander T Ihler. 2012. Variational inference for crowdsourcing. In *Advances in Neural Information Processing Systems*. 692–700.

[20] Jungseul Ok, Sewoong Oh, Jinwoo Shin, and Yung Yi. 2016. Optimality of Belief Propagation for Crowdsourced Classification. *arXiv preprint arXiv:1602.03619* (2016).

[21] Nihar B Shah, Sivaraman Balakrishnan, and Martin J Wainwright. 2016. A permutation-based model for crowd labeling: Optimal estimation and robustness. *arXiv preprint arXiv:1606.09632* (2016).

[22] Peter Welinder, Steve Branson, Serge J Belongie, and Pietro Perona. 2010. The multidimensional wisdom of crowds. In *Advances in neural information processing systems*. 2424–2432.

[23] Jacob Whitehill, Paul L. Ruvolo, Ting fan Wu, Jacob Bergsma, and Javier R. Movellan. 2009. Whose Vote Should Count More: Optimal Integration of Labels from Labelers of Unknown Expertise. In *Advances in Neural Information Processing Systems 22*. 2035–2043.

[24] Justin Yang, Christina Han, and Edoardo Airoldi. 2014. Nonparametric estimation and testing of exchangeable graph models.. In *AISTATS*. 1060–1067.

[25] Yuchen Zhang, Xi Chen, Denny Zhou, and Michael I Jordan. 2014. Spectral methods meet EM: A provably optimal algorithm for crowdsourcing. In *Advances in neural information processing systems*. 1260–1268.

[26] Yuan Zhang, Elizaveta Levina, and Ji Zhu. 2015. Estimating network edge probabilities by neighborhood smoothing. *arXiv preprint arXiv:1509.08588* (2015).

[27] Denny Zhou, Sumit Basu, Yi Mao, and John C Platt. 2012. Learning from the wisdom of crowds by minimax entropy. In *Advances in Neural Information Processing Systems*. 2195–2203.

[28] Dengyong Zhou, Qiang Liu, John C Platt, Christopher Meek, and Nihar B Shah. 2015. Regularized minimax conditional entropy for crowdsourcing. *arXiv preprint arXiv:1503.07240* (2015).