Reducing Crowdsourcing to Graphon Estimation, Statistically

Devavrat Shah
Massachusetts Institute of Technology

Christina Lee Yu
Microsoft Research New England

Abstract

Inferring the correct answers to binary tasks based on multiple noisy answers in an unsupervised manner has emerged as the canonical question for micro-task crowdsourcing or more generally aggregating opinions. In graphon estimation, one is interested in estimating edge intensities or probabilities between nodes using a single snapshot of a graph realization. In the recent literature, there has been exciting development within both of these topics. In the context of crowdsourcing, the key intellectual challenge is to understand whether a given task can be more accurately denoised by aggregating answers collected from other different tasks. In the context of graphon estimation, precise information limits and estimation algorithms remain of interest.

In this paper, we utilize a statistical reduction from crowdsourcing to graphon estimation to advance the state-of-art for both of these challenges. We use concepts from graphon estimation to design an algorithm that achieves better performance than the *majority voting* scheme for a setup that goes beyond the *rank one* models considered in the literature. We use known lower bounds for crowdsourcing to derive lower bounds for graphon estimation.

1 Introduction

Crowdsourcing: background. In recent years, crowd-sourcing platforms have become complementary computing systems 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. However, a human could relatively easily and quickly provide answers to these tasks.

As a result, crowd-sourcing platforms such as Amazon Mechanical Turk have emerged, on which requesters post tasks that they would like to be solved along with a monetary reward for completion, and human workers browse the posted tasks and earn money for providing responses to these tasks. 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, workers may have different levels of language proficiency, leading to noisy responses for language translation tasks. Alternatively, even if a worker is capable of solving the task, s/he may be lazy and may respond arbitrarily to save 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?

Crowdsourcing: formally. Let there be $T \geq 1$ binary tasks with $a_i \in \{-1, 1\}$ for $i \in [T]$ representing the *true* answer for the ith task[1]. Let there be $W \geq 1$ workers. If worker j is asked to provide answer to task i, then it will be denoted M_{ij}, which has distribution

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

Proceedings of the 21^{st} International Conference on Artificial Intelligence and Statistics (AISTATS) 2018, Lanzarote, Spain. PMLR: Volume 84. Copyright 2018 by the author(s).

[0] Author names appear in alphabetical order of their last names.

[1] We shall use notation $[N] \equiv \{1, \ldots, N\}$.

where $F_{ij} \in [0,1]$ represents the "skill" of worker j for task i. Therefore, $\mathbb{E}[M_{ij}] = a_i(2F_{ij} - 1)$. We shall assume that M_{ij} are independent across all i, j. We shall use notation $M = [M_{ij}] \in \{-1,1\}^{T \times W}$, $\mathbf{a} = [a_i] \in \{-1,1\}^T$ and $F = [F_{ij}] \in [0,1]^{T \times W}$.

As a crowdsourcing system operator, our decision is two fold: (a) determine a minimal assignment of tasks to workers denoted by the subset $\mathcal{E} \subset [T] \times [W]$ where M_{ij} is then queried for all $(i,j) \in \mathcal{E}$, and (b) determine an algorithm to infer $a_i,\ i \in [T]$ so that the fraction of answers we infer incorrectly is minimized. Naturally, the design for (a) and (b) will collectively determine the trade-off of between the accuracy of inference and number of queries per task. The eventual goal is to achieve the *pareto* boundary of this trade-off.

Crowdsourcing: prior work. We describe the prior work that have attempted to understand the above trade-off between accuracy and the number of worker queries per task by imposing structure on F (and hence $\mathbb{E}[M]$). The first formal model (an instance of (1)) was proposed by Dawid and Skene [13] where $F_{ij} = w_j$ for all $i \in [T]$, with $w_j \in [0,1]$ representing the "skill" of worker $j \in [W]$. Various inference algorithms were proposed including EM algorithm [13, 14, 34], belief propagation and iterative methods [18, 19, 20, 25, 24, 26], and spectral methods [15, 11, 4]. The minimax optimal trade-off curve for this model states that to achieve accuracy α, the number of worker queries per task requires a scaling of $(\frac{1}{W}\sum_j (2w_j - 1)^2)^{-1} \ln(\frac{1}{\alpha}))$ [18, 19, 20], which does not change even when one has the option to query adaptively.

The simple *majority voting* algorithm where the we infer the answer to a given task as simply the majority of the answers provided to it, provides a trade-off such that to achieve accuracy α, the number of worker queries per task requires a scaling of $(\frac{1}{W}\sum_j (2w_j - 1))^{-2} \ln(\frac{1}{\alpha}))$. Therefore, if the worker skills are not identical, then majority vote strictly requires more queries compared to the optimal algorithm. Put it other way, there is a *strict benefit* in utilizing answers of workers for different tasks to better estimate answers for a given task for the Dawid-Skene model. Effectively, this has led to the following intellectual quest:

Under what conditions on F (or $\mathbb{E}[M]$) can combining answers of workers for all tasks help, i.e. when is it possible to beat majority voting?

Surprisingly, this has turned out to be a hard question. The notable progress towards this quest have been in recent works [21, 27] where authors allow the F_{ij} to depend on the task difficulty in addition to worker skill as in Dawid-Skene model. They show that it is indeed possible to beat majority voting in specific settings; however, all of the models for which an algorithm has been formally established to beat majority vote, result in assumptions that limit $\mathbb{E}[M]$ to a *rank one* matrix. We ask the question: *is it possible to beat majority voting when $\mathbb{E}[M]$ has rank $d > 1$?*

Crowdsourcing: our contribution. We answer the above question in the affirmative. We show that for a natural model where the workers and tasks are of d types, with $F_{ij} = p$ if i and j are same type, and $F_{ij} = \frac{1}{2}$ otherwise, then a simple algorithm achieves accuracy α with the number of worker queries per task scaling as $(\frac{d}{(2p-1)^2} \ln(\frac{d}{\alpha}))$; in contrast, majority voting requires $(\frac{d^2}{(2p-1)^2} \ln(\frac{1}{\alpha}))$ queries per task.

Graphon: formally, prior work. In graphon or 1-bit matrix estimation, we partially observe a symmetric matrix $Y = [Y_{ij}] \in \{0,1\}^{n \times n}$. Specifically, we observe entries of Y for $\mathcal{E} \subset [n] \times [n]$, i.e. for all $(i,j) \in \mathcal{E}$, we observe Y_{ij}. The matrix Y is generated so that Y_{ij} are independent. The goal is to recover the expected matrix $\mathbb{E}[Y] \in [0,1]^{n \times n}$ by observing as few entries $(= |\mathcal{E}|)$ of Y as possible. In the setting of graphon estimation, one assumes that each $i \in [n]$ is associated with a *latent* parameter $\theta_i \sim U(0,1)$ sampled uniformly on the unit interval independently across all i. Then $\mathbb{E}[Y_{ij}] = f(\theta_i, \theta_j)$ for some measurable *latent* function $f : [0,1] \times [0,1] \to [0,1]$ for all $i, j \in [n]$. The fraction of observations, or samples from the $\binom{n}{2}$ unique points in matrix Y, that is required to recover P accurately depends on the structure of f.

There has been lots of recent progress in answering this question for different structures of f. For the stochastic block model where $\mathbb{E}[Y]$ has finitely many distinct rows (and columns), very precise understanding has been obtained, cf. [1, 2, 10]. For P with strictly monotonic row and column sums, [7, 32, 6] provide estimators that are consistent with minimal observations required. If f is Lipschitz, there are consistent estimators for both graphon estimation and matrix completion (cf. [33, 8, 23, 5]). Results in matrix completion and mixed membership stochastic block model estimation provide estimators when $\mathbb{E}[Y]$ is low rank (cf. [8, 9, 12, 3]). For the most general setting in which the latent function f can be any measurable function, there do not yet exist polynomial time estimators, although [22] has studied the rate achieved by a least squares estimator, and provided accompanying minimax convergence rates.

Despite this remarkable progress, lower bounds for graphon estimation are still being developed. In the recent work of [29], for the mixed membership model, an explicit lower bound has been conjectured for the detection of communities. Establishing such lower bounds unconditionally can lead to resolution of statistical and computational trade-offs. [31] and [22] have shown that the minimax optimal rate for graphon estimation with α-Holder smoothness is 1 for $np = O(1)$, $\log(np)/np$ for $\log(np) \leq \alpha \log n + (\alpha+1) \log\log n$, and $(n^2 p)^{-\alpha/(\alpha+1)}$ for $\log(np) \geq \alpha \log n + (\alpha + 1) \log\log n$.

Graphon: our contribution. By providing an explicit statistical reduction of crowdsourcing to graphon estimation, we transfer a lower bound utilized in [18, 19, 20] to obtain lower bounds for graphon estimation. In [18, 19, 20], this lower bound was crucial for establishing the minimax optimality for the Dawid-Skene model. The implied lower bound suggests an invariant that the *number of observations multiplied by the square of the minimal eigenvalue of f with respect to its spectral decomposition must be larger than a universal constant.* This lower bound provides a analytic form for an "incoherence" like assumption that has been made popular in the matrix estimation literature. In comparison to the previous lower bound, our bound is looser as a function of n and p, but it shows the dependence on spectral properties such as the smallest eigenvalue and eigenfunction amplitude, which is omitted in the previous bounds.

We present our contribution in the context of graphon parameter estimation because it is a recently active area of particular interest which emphasizes the binary data model. However our results relate to the broader setting of asymmetric matrix estimation as well.

2 Metrics, Models and Baselines

2.1 Metrics

Crowdsourcing. We shall measure the performance of an inference algorithm in terms of the fraction of answers obtained correctly. Precisely, given true answers $\mathbf{a} = [a_i] \in \{-1, 1\}^T$, the error of inferred answers $\hat{\mathbf{a}} = [\hat{a}_i] \in \{-1, 1\}^T$ is

$$\text{error}(\hat{\mathbf{a}}, \mathbf{a}) = \tfrac{1}{T}\Big(\textstyle\sum_{i=1}^{T} \mathbb{I}(\hat{a}_i \neq a_i)\Big), \tag{2}$$

where $\mathbb{I}(\cdot)$ is the indicator function. The number of queries per task is simply $|\mathcal{E}|/T$. The goal is to determine what is the minimal number of queries per task needed to obtain $\mathbb{E}[\text{error}(\hat{\mathbf{a}}, \mathbf{a})] \leq \alpha$ for any $\alpha \in (0, 1)$.

Graphon. The accuracy of an estimate $\hat{Y}$ for the matrix $\mathbb{E}[Y]$ is measured in terms of the Mean-Squared-Error (MSE) defined as

$$\text{MSE} = \mathbb{E}\Big[\tfrac{1}{n^2}\textstyle\sum_{i,j}(\hat{Y}_{ij} - \mathbb{E}[Y_{ij}])^2\Big]. \tag{3}$$

We would like to produce an estimate $\hat{Y}$ given observed entries $\{Y_{ij}\}_{(i,j)\in\mathcal{E}}$, indexed by $\mathcal{E} \subset [n] \times [n]$, which denotes the observed locations of matrix $Y = [Y_{ij}]$. We assume the entries are uniformly sampled, i.e. $\mathcal{E}$ includes each of the possible entries with probability p, independently, such that on average $|\mathcal{E}| = n^2 p$. The question is, what is the minimal sample probability p that is required to achieve MSE $\leq \alpha$.

2.2 Models

Prior Models Are Rank 1. There are three prior works that provide theoretical analyses establishing that the performance of the *majority estimation algorithm can be beaten* in the context of crowdsourcing. We describe those three models below. As we shall see, for each of these three models, the expected response matrix $\mathbb{E}[M]$ turns out to be rank 1.

Model 2.1 (Dawid-Skene)**.** This model assumes every task is homogeneous; and worker j is associated with a reliability parameter $w_j \in [0, 1]$ for $j \in [W]$, which is the probability a worker solves the task correctly [13]. The probability that worker j answers task i correctly is given by $F_{ij} = w_j$. Therefore, $\mathbb{E}[M_{ij}] = a_i(2w_j - 1)$, such that $\mathbb{E}[M]$ is a rank 1 matrix.

Model 2.2. In [21], every task i is associated with a difficulty parameter $t_i \in \left[\frac{1}{2}, 1\right]$, i.e. the probability a task is perceived incorrectly, and every worker j is associated with a reliability parameter $w_j \in [0, 1]$, i.e. the probability a worker responds consistently with perceived answer. The probability worker j answers task i correctly is given by $F_{ij} = t_i w_j + (1-t_i)(1-w_j)$. Therefore, $\mathbb{E}[M]$ is a rank 1 matrix,

$$\mathbb{E}[M_{ij}] = a_i(2F_{ij} - 1) \; = \; (a_i(2t_i - 1)) \times (2w_j - 1).$$

Model 2.3. In [27], it is assumed that there is an ordering (or permutation) of worker skill levels and task difficulty levels under which the probability of a correct response is monotonic. Specifically, their theoretical guarantees hold for the model $F_{ij} = w_j(1 - t_i) + \frac{1}{2}t_i$, which also implies $\mathbb{E}[M]$ is a rank 1 matrix,

$$\mathbb{E}[M_{ij}] = a_i(2F_{ij} - 1) \; = \; (a_i(1 - t_i)) \times (2w_j - 1).$$

There are other models considered in the literature such as [17, 16] assuming that each task (and hence worker) are of d different types. It does not result in $\mathbb{E}[M]$ being rank 1, but their theoretical analysis assumes the knowledge of the correct answers for a subset of tasks in order to bootstrap the answer. In [30], the F_{ij} is derived from a Gaussian distribution. In effect, $F_{ij} = f(t_i, w_j)$ where f is a Lipschitz function. They do not provide a theoretical analysis for how their approach compares to majority voting. [28] considers a model in which there are three types of collected responses: ratings provided by a large fraction of reliable workers, ratings from bounded fraction of adversarial workers, and stochastic ratings obtained “ourselves”. Their model allows for adversarial responses which is beyond a purely stochastic model. Their reliable workers provide answers which are monotonically increasing with the quality, similar to the monotonic model of [27]. However their model still does not allow for specialization of different types of workers for tasks, where workers may be good or bad depending on the type of task.

Rank d Model. In this work, we shall argue that it is possible to beat majority voting even when $\mathbb{E}[M]$ has rank d with $d > 1$. We use a simple model that leads to a rank d structure of $\mathbb{E}[M]$. It is similar to the classical stochastic block model with d communities.

Model 2.4 (d-Type Specialization Model)**.** Each task $i \in [T]$ is associated with a task type $t_i \in [d]$, each worker $j \in [W]$ is associated with a worker type $w_j \in [d]$. Assume the probability that worker j correctly answers task i is denoted F_{ij}, which takes value p if $t_i = w_j$, and $\frac{1}{2}$ otherwise. It follows that $\mathbb{E}[M_{ij}] = 2p - \frac{1}{2}$ if $t_i = w_j$ and $\mathbb{E}[M_{ij}] = 0$ otherwise. $\mathbb{E}[M]$ is equivalent to a rank d block matrix with d disjoint blocks. We assume worker types are sampled uniformly amongst the $[d]$ types and task types are uniformly drawn from the d types. Let $\{\mathcal{W}_1, \dots \mathcal{W}_d\}$ partition the workers into their associated types such that $w_j = z$ for all $j \in \mathcal{W}_z$. We assume $T \gg W$.

2.3 Baselines for Performance

We discuss the majority voting estimation algorithm and its performance. This serves as the baseline for any crowd-sourcing algorithm since majority voting only utilizes the answers associated with a given task to make inference. We also discuss the improvement that could be gained by a maximum likelihood oracle with extra knowledge of the parameter matrix F.

Majority voting. Let $\mathcal{N}(i) = \{j : (i,j) \in \mathcal{E}\}$ denote the set of workers which are assigned to task $i \in [T]$. Define the majority vote estimate for a_i as

$$\hat{a}_i^{MV} = \operatorname{sign}\left(\textstyle\sum_{j\in\mathcal{N}(i)} M_{ij}\right). \tag{4}$$

Under either of the assumptions that $F_{ij} > \frac{1}{2}$ for all (i,j) or simply that $\sum_{j\in\mathcal{N}(i)} F_{ij} > \frac{1}{2}$ for all i, a simple application of Chernoff's bound implies

$$\mathbb{P}(\hat{a}_i^{MV} \neq a_i) \leq \exp\Big(- \tfrac{|\mathcal{N}(i)|}{2} \Big(\frac{\sum_{j\in\mathcal{N}(i)}(2F_{ij} - 1)}{|\mathcal{N}(i)|}\Big)^2\Big).$$

If we choose $\mathcal{N}(i) \subset [W]$ at random, effectively

$$\mathbb{P}(\hat{a}_i^{MV} \neq a_i) \leq \exp\Big(- \tfrac{|\mathcal{N}(i)|}{2}(2\mathbb{E}[F_{i\star}] - 1)^2\Big),$$

where $\mathbb{E}[F_{i\star}]$ denotes the expected value of F_{ij} assuming that worker j is chosen uniformly at random amongst $[W]$. To achieve $\mathbb{P}(\hat{a}_i^{MV} \neq a_i) \leq \alpha$, we assign $|\mathcal{N}(i)| \approx \frac{2}{(2\mathbb{E}[F_{i\star}]-1)^2} \ln(\frac{1}{\alpha})$ workers per task. For the d-Type Specialization Model, this requires

$$|\mathcal{N}(i)| = \Theta\Big(\tfrac{d^2}{(2p-1)^2} \ln\big(\tfrac{1}{\alpha}\big)\Big). \tag{5}$$

Maximum Likelihood Oracle. Suppose we had an oracle that told us the parameter matrix F. How much can we improve the estimator? We can simply compute the maximum likelihood solution,

$$\hat{a}_i^{ML} = \operatorname{sign}\left(\textstyle\sum_{j\in\mathcal{N}(i)} \ln\left(\tfrac{F_{ij}}{1-F_{ij}}\right) M_{ij}\right). \tag{6}$$

Let us denote $\omega_{ij} = \ln(\frac{F_{ij}}{1-F_{ij}})$. We can verify that $\operatorname{sign}(\mathbb{E}[\omega_{ij} M_{ij}]) = a_i$, and the magnitude increases as F_{ij} is more polarized towards 0 or 1. If $F_{ij} = 1$, then $\omega_{ij} = \infty$, and if $F_{ij} = 0$, then $\omega_{ij} = -\infty$, such that a single datapoint would dominate the estimate. In either of these two cases, the probability of error is zero, since at least one worker is fully reliable. Otherwise, the probability of error is bounded by

$$\mathbb{P}\Big(\Big|\textstyle\sum_{j\in\mathcal{N}(i)} \omega_{ij}(M_{ij} - \mathbb{E}[M_{ij}])\Big| \geq \sum_{j\in\mathcal{N}(i)} \omega_{ij}\mathbb{E}[M_{ij}]\Big).$$

Clearly, even if we had $F_{ij} = 1$ for even one worker j for a given task i, then maximum likelihood could outperform majority voting drastically. But issue is that we *do not* know F. So the question remains: *when can we beat majority vote?*

Directly Using Graphon Estimation. We can approximate the maximum likelihood oracle by estimating the matrix $\mathbb{E}[M]$ directly from data using methods from graphon estimation, and then using it to estimate $\mathbf{a}$. Let $\hat{M}$ be the estimate of $\mathbb{E}[M]$ obtained from the observed data matrix M. Under the assumption that $\sum_{j\in[W]} F_{ij} > \frac{1}{2}$, this leads to estimates of

$$\hat{a}_i = \operatorname{sign}(\textstyle\sum_{j\in[W]} \hat{M}_{ij}) \quad i \in [T]. \tag{7}$$

Hence, $\mathbb{P}(\hat{a}_i \neq a_i)$ is bounded above by

$$\mathbb{P}\Big(\Big|\textstyle\sum_{j\in[W]}(\hat{M}_{ij} - \mathbb{E}[M_{ij}])\Big| \geq \sum_{j\in[W]} \mathbb{E}[M_{ij}]\Big).$$

Observe that we can even include inferred values $\hat{M}_{ij}$ for $(i,j) \notin \mathcal{E}$, for which the response M_{ij} was not even observed. Evaluating this error bound expression will depend on the spectral properties of the model and the matrix estimation algorithm.

Under the assumption that $F_{ij} > \frac{1}{2}$, an estimation algorithm could be

$$\hat{a}_i = \operatorname{sign}(\hat{M}_{ij^*}) \text{ for } j^* \in \operatorname{argmax}_j |\hat{M}_{ij}|. \tag{8}$$

If $|\hat{M}_{ij} - \mathbb{E}[M_{ij}]| \le \frac{1}{2}(\max_j |\mathbb{E}[M_{ij}]| + \min_j |\mathbb{E}[M_{ij}]|)$ for all (i,j), then $\hat{a}_i = a_i$. Therefore $\mathbb{P}(\hat{a}_i \neq a_i)$ is bounded by

$$\mathbb{P}\Big(\max_{j\in[W]} |\hat{M}_{ij} - \mathbb{E}[M_{ij}]| \ge \tfrac{1}{2}\Big(\max_j |\mathbb{E}[M_{ij}]| + \min_j |\mathbb{E}[M_{ij}]|\Big)\Big).$$

This evaluation depends on the graphon estimation algorithm (and $\mathbb{E}[M]$), which unfortunately is more difficult. Effectively, crowdsourcing requires estimating only the *sign* of a column sum of $\mathbb{E}[M]$ while graphon estimation requires estimating the entire matrix. As a result, the naive use of a graphon estimator is unlikely to beat majority vote. This is precisely the challenge we overcome in this paper.

3 Result I: Beating Majority

We describe how we can beat majority voting for the d-Type Specialization Model using a simple graphon estimation algorithm coupled with a clever crowdsourcing design (assignment of workers to tasks). We state the theorem below, followed by a description of the algorithm that achieves the theorem and the corresponding proof.

Theorem 3.1. *Assuming the d-Type Specialization model, for any given $\alpha \in (0,1)$ and $T \gg W$, there exists a design and estimation algorithm for crowdsourcing such that by soliciting no more than*

$$\frac{16d}{(2p-1)^2} \ln\left(\frac{6d}{\alpha}\right)$$

responses per task, the estimation $\hat{\mathbf{a}}$ of $\mathbf{a}$ is such that

$$\mathbb{E}[\text{error}(\hat{\mathbf{a}}, \mathbf{a})] \le \alpha.$$

Two Stage Algorithm. We describe the algorithm that leads to Theorem 3.1 for the d-Type Specialization model. The algorithm takes parameter R, L and ξ that will be determined later.

- Stage 1: Clustering Workers by Types.
  - Let $\mathcal{S} \subset [T]$ represent randomly chosen R tasks from T tasks, i.e. $|\mathcal{S}| = R$.
  - Assign *all* W workers to solve tasks in $\mathcal{S}$.
  - For $j \in [W]$, assign them to (disjoint) clusters of workers sequentially as follows:
    - If there exists a cluster of workers $\mathcal{Q} \subset [j-1]$ such that for each $j' \in \mathcal{Q}$

      $$\tfrac{1}{R}\textstyle\sum_{i\in\mathcal{S}} \mathbb{I}(M_{ij} = M_{ij'}) > \xi,$$

      then assign j to (any) such cluster $\mathcal{Q}$.
    - If no such cluster $\mathcal{Q}$ exists, create a new cluster containing j, i.e. $\{j\}$.
  - Let C be the number of disjoint clusters constructed in the previous step, and let $\{\mathcal{V}_1 \dots \mathcal{V}_C\}$ denote the constructed clusters, which forms a disjoint partitioning of $[W]$.
  - Given the above estimated worker clusters, for each task $i \in [T] \setminus S$ and for each cluster $z \in [C]$, assign task i to L workers sampled uniformly at random amongst the set of workers $\mathcal{V}_z$. Therefore, each task i is assigned to a total of LC workers.
- Stage 2: Estimating the Task Answers
  - Let $\mathcal{N}(i) \subset [W]$ denote the workers assigned to task i. Let $z^*(i) = \text{argmax}_{z\in[C]} |\sum_{j\in\mathcal{N}(i)\cap\mathcal{V}_z} M_{ij}|$.
  - Then $\hat{a}_i = \text{sign}\left(\sum_{j\in\mathcal{N}(i)\cap\mathcal{V}_{z^*(i)}} M_{ij}\right)$.

Proof of Theorem 3.1. We choose the following parameters for the proof. Let $\xi = \frac{1}{2} + \frac{(2p-1)^2}{4d}, R = \frac{8d^2}{(2p-1)^4} \ln\left(\frac{3W(W-1)}{2\alpha}\right), L = \frac{8}{(2p-1)^2}\ln\left(\frac{6d}{\alpha}\right), W \ge \frac{16d}{(2p-1)^2}\ln\left(\frac{6d}{\alpha}\right)$, and $T \ge C'W^3$, where C' is a large enough constant so that with the above choices of parameters (esp. W), we have that

$$\tfrac{8Wd^2}{T(2p-1)^4} \ln\left(\tfrac{3W(W-1)}{2\alpha}\right) \le \tfrac{8d}{(2p-1)^2}\ln\left(\tfrac{6d}{\alpha}\right).$$

First we analyze the result of partitioning the workers into clusters. For a pair of workers (a,b), the expected fraction of matching responses will be larger if the workers have the same type as opposed to different types. Consider a pair of workers (a,b) whose type is equal. For each assigned task $i \in S$, with probability $1/d$, the task type is equal to the worker type, such that the probability that $M_{ia} = M_{ib}$ is equal to $p^2 + (1-p)^2$. With probability $1 - 1/d$, the task type is different from the worker type, such that the probability that $M_{ia} = M_{ib}$ is equal to $\frac{1}{2}$. Therefore, the indicator random variable $\mathbb{I}(M_{ia} = M_{ib})$ is a Bernoulli random variable with parameter

$$\tfrac{p^2+(1-p)^2}{d} + \tfrac{(d-1)}{2d} = \tfrac{1}{2} + \tfrac{(2p-1)^2}{2d}.$$

If the pair of workers (a,b) have different types, then no matter what the task type is, at least one worker's response is equal to a random coin flip. Therefore the probability that $M_{ia} = M_{ib}$ is equal to $\frac{1}{2}$ for any task i, such that $\mathbb{I}(M_{ia} = M_{ib})$ is a Bernoulli random variable with parameter $\frac{1}{2}$. For a threshold of ξ, the probability that the estimated clusters exactly recovers the d worker types is at least as large as the probability that for all worker pairs (a,b) of the same type, $\frac{1}{R}\sum_{i\in S}\mathbb{I}(M_{ia} = M_{ib}) > \xi$, and for all workers pairs (a,b) of different types, $\frac{1}{R}\sum_{i\in S}\mathbb{I}(M_{ia} = M_{ib}) \le \xi$. This can be loosely bounded by using Chernoff's bound for concentration of the average of Bernoulli random variables in addition to the union bound over all worker pairs.

$$\begin{aligned}
1 &- \textstyle\sum_{a\neq b} \mathbb{P}\Big(\frac{1}{R}\sum_{i\in\mathcal{S}} \mathbb{I}(M_{ia}=M_{ib}) \leq \xi\Big)\mathbb{I}(w_a=w_b) \\
&\quad - \textstyle\sum_{a\neq b} \mathbb{P}\Big(\frac{1}{R}\sum_{i\in\mathcal{S}} \mathbb{I}(M_{ia}=M_{ib}) > \xi\Big)\mathbb{I}(w_a\neq w_b) \\
&= 1-\exp\Big(-2(\tfrac{1}{2}+\tfrac{(2p-1)^2}{2d}-\xi)^2R\Big)\textstyle\sum_{a\neq b}\mathbb{I}(w_a=w_b) \\
&\quad - \exp\Big(-2(\tfrac{1}{2}-\xi)^2R\Big)\textstyle\sum_{a\neq b}\mathbb{I}(w_a\neq w_b).
\end{aligned}$$

If we choose $\xi = \frac{1}{2}+\frac{(2p-1)^2}{4d}$, then the probability of exactly recovering the partition of worker types is bounded below by

$$1-\tbinom{W}{2}\exp\left(-\tfrac{R(2p-1)^4}{8d^2}\right).$$

We also need to guarantee that the number of workers per type is at least equal to L so that it is possible to find at least L workers per type. The number of workers of type z is given by $|\mathcal{W}_z| = \sum_{j\in[W]}\mathbb{I}(w_j = z)$, which is a Binomial$(W, \frac{1}{d})$ random variable. By Chernoff's bound and union bound,

$$\begin{aligned}
\mathbb{P}(\cup_{z\in[d]}\{|\mathcal{W}_z|<L\}) &\leq \sum_{z\in[d]}\mathbb{P}(|\mathcal{W}_z|\geq L) \\
&\leq d\exp(-2(\tfrac{1}{d}-\tfrac{L}{W})^2W).
\end{aligned}$$

In order to analyze the estimates of the task solution, we show that the distribution of responses collected within each worker type for each task concentrates to its expectation. Let S_{iz} be defined as

$$\begin{aligned}
S_{iz} &:= \textstyle\sum_{j\in\mathcal{N}(i)\cap\mathcal{W}_z}\mathbb{I}(M_{ij}=+1) \\
&= \textstyle\sum_{j\in\mathcal{N}(i)\cap\mathcal{W}_z}\frac{1+M_{ij}}{2} \\
&= L\left(\tfrac{1}{2}+\tfrac{1}{L}\textstyle\sum_{j\in\mathcal{N}(i)\cap\mathcal{W}_z}M_{ij}\right),
\end{aligned}$$

such that $\sum_{j\in\mathcal{N}(i)\cap\mathcal{W}_z}M_{ij} = L(\frac{S_{iz}}{L}-\frac{1}{2})$. By model assumptions, the distribution of S_{iz} will be Binomial$(|\mathcal{N}(i)\cap\mathcal{W}_z|, \frac{1}{2})$ if $t_i \neq z$, i.e. the worker type is different than the task type. If $t_i = z$, i.e. the worker type is equal to the task type, then the distribution of S_{iz} will either be Binomial$(|\mathcal{N}(i)\cap\mathcal{W}_z|, p)$ if the task solution a_i is equal to $+1$ and Binomial$(L, 1-p)$ if the task solution a_i is equal to -1. We bound the probability that S_{iz} concentrates around its expectation within an additive error of $\frac{1}{2}|p-\frac{1}{2}||\mathcal{N}(i)\cap\mathcal{W}_z|$ using Chernoff's bound,

$$\begin{aligned}
&\mathbb{P}\left(|S_{iz}-\mathbb{E}[S_{iz}]| \geq \tfrac{1}{2}|p-\tfrac{1}{2}||\mathcal{N}(i)\cap\mathcal{W}_z|\right) \\
&\leq 2\exp\Big(-2\Big(\tfrac{1}{2}|p-\tfrac{1}{2}|\Big)^2|\mathcal{N}(i)\cap\mathcal{W}_z|\Big) \\
&= 2\exp\left(-\tfrac{(2p-1)^2|\mathcal{N}(i)\cap\mathcal{W}_z|}{8}\right).
\end{aligned}$$

Conditioned on the events that the worker cluster partitioning is exactly correct, i.e. $C = d$ and $\mathcal{V}_z = \mathcal{W}_z$ for all $z \in [d]$, and that S_{iz} concentrates within an additive error of $\frac{1}{2}|p-\frac{1}{2}|$ around its expectation for all $z \in [d]$, it follows that $z^*(i) = t_i$ and $a_i = \text{sign}\left(\sum_{j\in\mathcal{N}(i)\cap\mathcal{W}_{t_i}}M_{ij}\right)$. Recall that by construction $|\mathcal{N}(i)\cap\mathcal{W}_z| = L$ for all $i\in[T]\setminus\mathcal{S}$ and $z\in[d]$, and $|\mathcal{N}(i)\cap\mathcal{W}_z| = |\mathcal{W}_z|$ for $i\in\mathcal{S}$. Conditioned on the event that $|\mathcal{W}_z| \geq L$ for all $z\in[d]$, this implies that $|\mathcal{N}(i)\cap\mathcal{W}_z| \geq L$ for all $i\in[T]\setminus\mathcal{S}$ and $z\in[d]$. The expected fraction of incorrect responses is bounded by

$$\begin{aligned}
&\mathbb{E}\left[\tfrac{1}{T}\textstyle\sum_{i\in[T]}\mathbb{I}(\hat{a}_i\neq a_i)\right] \\
&\leq \tbinom{W}{2}\exp\left(-\tfrac{R(2p-1)^4}{8d^2}\right)+d\exp(-2(\tfrac{1}{d}-\tfrac{L}{W})^2W) \\
&\quad + 2d\exp\left(-\tfrac{(2p-1)^2L}{8}\right).
\end{aligned}$$

To limit the fraction of errors to α, we can choose $R = \frac{8d^2}{(2p-1)^4}\ln\left(\frac{3W(W-1)}{2\alpha}\right), L = \frac{8}{(2p-1)^2}\ln\left(\frac{6d}{\alpha}\right)$, and $W > \frac{16d}{(2p-1)^2}\ln\left(\frac{6d}{\alpha}\right)$. The average number of solicited responses per task average out to

$$\begin{aligned}
&\tfrac{1}{T}(WR+Ld(T-R)) \leq Ld+\tfrac{WR}{T} \\
&= \tfrac{8d}{(2p-1)^2}\ln\left(\tfrac{6d}{\alpha}\right)+\tfrac{8Wd^2}{T(2p-1)^4}\ln\left(\tfrac{3W(W-1)}{2\alpha}\right).
\end{aligned}$$

With a sufficiently large constant C' such that $T \geq C'W^3$, the second term is dominated by the first term in the above equations, showing the desired result.

4 Result II: Graphon Lower Bound

In this section, we illustrate that we can use the reduction between crowdsourcing and graphon estimation in order to produce lower bounds for graphon estimation, since there are known lower bounds in the crowdsourcing literature. Let us consider the "spammer-hammer" model, i.e. $F(i,j) = w_j \in \{\frac{1}{2}, 1\}$ for all i, j, and let $\sigma^2 := \frac{1}{W}\sum_{j\in[W]}(2w_j-1)^2 = \frac{1}{W}\sum_{j\in[W]}\mathbb{I}(w_j = 1)$. This model assumes that each collected response is either completely useless (equivalent to a coin toss), or it is 100 percent reliable and true. The challenge now becomes to filter out and distinguish the high quality responses from the noise. A minimax lower bound for nonadaptive algorithms for the Dawid-Skene model has been shown using the spammer-hammer model, since a task can only be answered correctly if there was at least one response collected from a "hammer" [20]. We recap the argument presented in their paper. Since the data is collected without knowledge of who is a spammer vs. hammer, the probability of collecting responses from all spammers is at least $(1-\sigma^2)^{pW}$, where pW is the number of responses collected. Thus for any estimator (assuming $\sigma^2 \le \frac{2}{3}$),

$$\tfrac{1}{T}\textstyle\sum_{i\in[T]}\mathbb{P}(\hat{a}_i \neq a_i) \ge \tfrac{1}{2}(1-\sigma^2)^{pW} \ge \tfrac{1}{2}e^{-(\sigma^2+\sigma^4)pW}$$

Specifically, for a task i assigned to L workers,

$$\mathbb{P}(\hat{a}_i \neq a_i) \ge \tfrac{1}{2}(1-\sigma^2)^{L} \ge \tfrac{1}{2}e^{-(\sigma^2+\sigma^4)L}.$$

If we use the estimator described (8), which estimates $\hat{M} \approx \mathbb{E}[M]$, and chooses $\hat{a}_i$ based on the response of the worker which maximizes $|\hat{M}_{ij}|$, then

$$\mathbb{P}\left(\|\hat{M}-M\|_\infty \ge \tfrac{1}{2}\right) \ge \mathbb{P}(\hat{a}_i \neq a_i) \ge \tfrac{1}{2}e^{-(\sigma^2+\sigma^4)pW}.$$

This also implies a lower bound on the MSE,

$$\begin{aligned} \text{MSE} &= \tfrac{1}{T}\textstyle\sum_{i\in[T]}\mathbb{E}\left[\tfrac{1}{W}\textstyle\sum_{j\in[W]}(\hat{M}_{ij}-M_{ij})^2\right] \\ &\ge \tfrac{1}{T}\textstyle\sum_{i\in[T]}\mathbb{P}\left(\bigcup_{j\in[W]}\left\{|\hat{M}_{ij}-M_{ij}| \ge \tfrac{1}{2}\right\}\right)\tfrac{1}{4W} \\ &\ge \tfrac{1}{8W}\exp\left(-(\sigma^2+\sigma^4)pW\right). \qquad (9) \end{aligned}$$

Construct an Equivalent Graphon Estimation Task. Given an instance of the crowdsourcing task under the spammer-hammer model, we construct a graphon estimation task such that the graphon data matrix is statistically equivalent to the collected crowdsourced response data matrix. If there existed an algorithm for graphon estimation that could achieve a given precision, it would directly imply that one could estimate the corresponding expected crowdsourced response matrix within the same precision given a uniformly random assignment scheme. The lower bound of (9) would then imply a lower bound for the constructed graphon estimation task.

Consider a spammer-hammer model as described previously, where $F_{ij} = w_j \in \{0,1\}$, and we assume a worker j is a spammer ($w_j = \frac{1}{2}$) with probability $1-\sigma^2$ and a hammer ($w_j = 1$) with probability σ^2. Let α be a prior on the task answer, such that $a_i = +1$ with probability α and $a_i = -1$ with probability $1-\alpha$. Assume that the number of workers is approximately proportional to the number of tasks, such that $T \sim \text{Binomial}(n,\beta)$ and $W = n - T$ for some n. Note that $T + W = n$ and $\mathbb{E}[T/(T+W)] = \beta$. As $n \to \infty$, the proportion of workers to tasks converges, i.e. $T \to \beta n$ and $W \to (1-\beta)n$.

Since the collected crowdsourced response matrix is asymmetric, we consider a larger symmetric $(T+W)\times(T+W)$ matrix where the upper right $T \times W$ block of the matrix is M and the lower right $W \times T$ block of the matrix is M^T. We consider the graphon estimation task with centered binary data matrices which take values $\{-1,+1\}$ rather than $\{0,1\}$, which simplifies the notation in translating between the crowdsourcing model and the matrix estimation model. The (centered) graphon model assumes that each $i \in [T+W]$ is associated to a latent variable θ_i sampled uniformly on the unit interval. For uniformly sampled entries indexed by $\mathcal{E} \in [(T+W)\times(T+W)]$, the observed data distribution when centered is

$$Y_{ij} = \begin{cases} +1 & \text{with probability } \frac{1}{2}(1+f(\theta_i,\theta_j)) \\ -1 & \text{with probability } \frac{1}{2}(1-f(\theta_i,\theta_j)). \end{cases}$$

such that $\mathbb{E}[Y_{ij}] = f(\theta_i,\theta_j)$.

Recall for the spammer hammer model that $\mathbb{E}[M_{ij}] = a_i(2w_j - 1)$ takes value 0 if $w_j = \frac{1}{2}$, and otherwise takes value a_i. We represent our worker or task types with the latent parameter $\theta_i \in [0,1]$. Split $[0,1]$ into four intervals, specified by $\mathcal{I}_1 = [0,\alpha\beta], \mathcal{I}_2 = [\alpha\beta,\beta], \mathcal{I}_3 = [\beta, 1-\sigma^2(1-\beta)]$, and $\mathcal{I}_4 = [1-\sigma^2(1-\beta),1]$. $\mathcal{I}_1$ corresponds to a task with $+1$ answer, $\mathcal{I}_2$ corresponds to a task with -1 answer, $\mathcal{I}_3$ corresponds to a spammer worker, and $\mathcal{I}_4$ corresponds to a hammer worker. Construct the latent function f according to

$$f(\theta_i,\theta_j) = \begin{cases} \text{sign}(\alpha\beta - \theta_i) & \text{if } \theta_i \in \mathcal{I}_1 \cup \mathcal{I}_2, \theta_j \in \mathcal{I}_4 \\ \text{sign}(\alpha\beta - \theta_j) & \text{if } \theta_i \in \mathcal{I}_4, \theta_j \in \mathcal{I}_1 \cup \mathcal{I}_2 \\ 0 & \text{otherwise .} \end{cases}$$

We can verify that the data matrix Y randomly generated from latent variables θ_i and constructed latent function f is statistically equivalent (under permutation) to the block matrix $\left[\begin{smallmatrix} Q & M \\ M^T & Q' \end{smallmatrix}\right]$, where M is the collected response matrix from the corresponding spammer-hammer crowdsourced model, and Q, Q' are $T \times T$ and $W \times W$ binary symmetric matrices respectively with observed entries taking value +1 or -1 with equal probability. Thus an estimate of $\mathbb{E}[Y]$ directly translates to an estimate of $\mathbb{E}[M]$, such that a graphon estimation algorithm that achieves some precision for this latent function f implies that there exists an algorithm for estimating the expected crowdsourced response matrix $\mathbb{E}[M]$ for the corresponding spammer-hammer model with parameter σ^2.

For the symmetric function $f : [0,1]^2 \to \mathbb{R}$, one can consider the corresponding HilbertSchmidt integral operator $F : L^2([0,1],\mathbb{R}) \to L^2([0,1],\mathbb{R})$ which operates over L^2 functions over the unit interval. Since the operator is compact and self-adjoint, the operator F has a discrete spectrum such that

$$f(\theta_i,\theta_j) = \textstyle\sum_{k\in\mathbb{Z}_+}\lambda_k q_k(\theta_i) q_k(\theta_j),$$

for orthonormal eigenfunctions q_k and eigenvalues λ_k, i.e. $\langle q_k, q_{k'}\rangle = 0$ for $k \neq k'$ and $\langle q_k, q_k\rangle = 1$ for all k, where

$$\langle q_k, q_{k'}\rangle = \int_0^1 q_k(\theta) q_{k'}(\theta) d\theta.$$

In the next two sections, we provide minimax lower bounds for graphon estimation with respect to related spectral properties of f.

Lower bounds with respect to maximum eigenfunction magnitude. Let $\mathcal{F}_B$ indicate the class of graphon models, represented by the associated set of latent functions f, for which $\sup_{k\in\mathbb{Z}_+}\sup_{\theta\in[0,1]}|q_k(\theta)| = B$, where $\{q_k\}_{k\in\mathbb{Z}_+}$ are the eigenfunctions of the integral operator associated to f. We compute the minimax bounds on the mean squared error and the probability that the max error is larger than one half. For a fixed p (density of observations), n, and $B \in [1,\infty)$, we minimize the error over all choices of estimators $\hat{Y}$, which is a function that takes in a data matrix Y and produces an estimated matrix $\hat{Y} \approx \mathbb{E}[Y]$, and we maximize the error over all models in $\mathcal{F}_B$, where the models are specified by $f(\theta_i,\theta_j)$. The minimax lower bounds comes from constructing an instance of the spammer-hammer crowdsourcing model such that $\sup_{\theta\in[0,1]}|q_k(\theta)| = B$, and then applying the above lower bounds from the crowdsourcing setting.

Theorem 4.1. *Let $\mathcal{F}_B$ be the class of latent variable models where the supremum of the eigenfunction amplitudes is $B \in [1,\infty)$. Let Y be the $n\times n$ data matrix generated according to the centered graphon latent variable model, and let p be the density of entries observed in the matrix uniformly at random. Let $\hat{Y}$ be an estimator which takes in the partially observed data matrix Y and produces an estimated matrix $\hat{Y}$.*

$$\min_{\hat{Y}}\max_{\mathcal{F}_B}\mathbb{P}\left(\|\hat{Y}-\mathbb{E}[Y]\|_\infty \le \tfrac{1}{2}\right) \ge \tfrac{1}{2}\exp\left(-\tfrac{pn}{2B^2-1}\right),$$

$$\min_{\hat{Y}}\max_{\mathcal{F}_B}\mathbb{E}\left[\tfrac{1}{n^2}\textstyle\sum_{ij}(\hat{Y}_{ij}-\mathbb{E}[Y])^2\right] \ge \frac{B^2 e^{-\frac{pn}{2B^2-1}}}{4(2B^2-1)n}.$$

These lower bounds follow from choosing $\beta = \frac{1}{2B^2}$ and $\sigma^2 = \frac{1}{2B^2-1}$ for the spammer-hammer model. Previous lower bounds have looked specifically at the scaling of n or pn with respect to the rank of the matrix, but this lower bound focuses on showing the scaling with respect to B, or the maximum amplitude of the eigenfunctions. This result implies that pn must scale as B^2. The property B is related to the incoherence property as well.

Lower bounds with respect to minimum eigenvalue. We can also show a lower bound in relation to the eigenvalues much in the same way. Let $\mathcal{F}_\lambda$ indicate the set of latent variable models for which $\sup_{k\in\mathbb{Z}_+}|\lambda_k| = \lambda$. We compute the minimax bounds on the mean squared error and the probability that the max error is larger than one half. For a fixed p, n, and $\lambda \in [0,\frac{1}{2}]$, we minimize the error over all choices of estimators $\hat{Y}$, which is a function that takes in a data matrix Y and produces an estimated matrix $\hat{Y}$, and we maximize the error over all models in $\mathcal{F}_\lambda$, where the models are specified by $f(\theta_i,\theta_j)$. The minimax lower bounds comes from constructing an instance of the spammer-hammer crowdsourcing model such that $\sup_{k\in\mathbb{Z}_+}|\lambda_k| = \lambda$, and then applying the above lower bounds from the crowdsourcing setting.

Theorem 4.2. *Let $\mathcal{F}_\lambda$ be the class of latent variable models where the minimum magnitude nonzero eigenvalue has magnitude $\lambda \in [0,\frac{1}{2}]$. Let Y be the $n\times n$ data matrix generated according to the latent variable model, and where p is the density of entries observed in the matrix. Let $\hat{Y}$ be an estimator which takes in the partially observed data matrix Y and produces and estimated matrix $\hat{Y}$.*

$$\min_{\hat{Y}}\max_{\mathcal{F}_\lambda}\mathbb{P}\left(\|\hat{Y}-\mathbb{E}[Y]\|_\infty \le \tfrac{1}{2}\right) \ge \tfrac{1}{2}e^{-2\lambda^2(4\lambda^2+1)pn},$$

$$\min_{\hat{Y}}\max_{\mathcal{F}_\lambda}\mathbb{E}\left[\tfrac{1}{n^2}\textstyle\sum_{ij}(\hat{Y}_{ij}-\mathbb{E}[Y])^2\right] \ge \tfrac{1}{4n}e^{-2\lambda^2(4\lambda^2+1)pn}.$$

These lower bounds follow from choosing $\sigma^2 = 4\lambda^2$ and $\beta = \frac{1}{2}$ for the spammer-hammer model. This result implies that pn must scale as $\lambda_{\min}^{-2}$.

5 Conclusion

In this paper, we showed that the crowdsourcing inference problem can be statistically reduced to graphon or 1-bit matrix estimation problem. This helps in multiple ways. First, it helps us show that it is indeed feasible to beat the majority voting baseline performance for setup where the underlying model leads to rank d setup for $d > 1$. This is in contrast to all known theoretical results that have established such property as all known results have been about rank 1 model. Second, it helps us establish refined, explicit lower bounds for graphon estimation problem by invoking lower bound for crowdsourcing problem. This allows us to obtain a useful invariant: the number of samples multiplied by the square of the minimum eigenvalue associated with the spectral decomposition of the graphon model is always lower bounded by a universal constant. Going forward, we believe that this relationship will help develop better estimation algorithms for the crowdsourcing problem. Further it will advance our understanding of graphon estimation as the crowdsourcing problem requires only row-sum estimation, not the entire matrix.

Acknowledgments

This work is supported in parts by a Microsoft Research postdoctoral fellowship and NSF projects CMMI 1462158, CMMI 1634259, CNS 1523546 and ARO MURI W911NF-16-1-0551.

References

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

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

[3] Animashree Anandkumar, Rong Ge, Daniel Hsu, and Sham Kakade. A tensor spectral approach to learning mixed membership community models. In *Conference on Learning Theory*, pages 867–881, 2013.

[4] Thomas Bonald and Richard Combes. Crowdsourcing: Low complexity, minimax optimal algorithms. *arXiv preprint arXiv:1606.00226*, 2016.

[5] Christian Borgs, Jennifer Chayes, Christina E. Lee, and Devavrat Shah. Thy friend is my friend: Iterative collaborative filtering for sparse matrix estimation. *Proceedings of Neural Information Processing Systems*, 2017.

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

[7] Stanley H Chan and Edoardo Airoldi. A consistent histogram estimator for exchangeable graph models. In *ICML*, pages 208–216, 2014.

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

[9] Yudong Chen and Martin J Wainwright. Fast low-rank estimation by projected gradient descent: General statistical and algorithmic guarantees. *arXiv preprint arXiv:1509.03025*, 2015.

[10] Peter Chin, Anup Rao, and Van Vu. 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.

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

[12] Mark A Davenport, Yaniv Plan, Ewout van den Berg, and Mary Wootters. 1-bit matrix completion. *Information and Inference*, 3(3):189–223, 2014.

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

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

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

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

[17] Chien-Ju Ho and Jennifer Wortman Vaughan. Online task assignment in crowdsourcing markets. In *Proceedings of the 26th Conference on Artificial Intelligence*, volume 12, pages 45–51, 2012.

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

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

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

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

[22] Olga Klopp, Alexandre B Tsybakov, and Nicolas Verzelen. Oracle inequalities for network models and sparse graphon estimation. *To appear in Annals of Statistics*, 2015.

[23] Christina E Lee, Yihua Li, Devavrat Shah, and Dogyoon Song. Blind regression: Nonparametric regression for latent variable models via collaborative filtering. In *Advances in Neural Information Processing Systems*, pages 2155–2163, 2016.

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

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

[26] Jungseul Ok, Sewoong Oh, Jinwoo Shin, and Yung Yi. Optimality of belief propagation for crowdsourced classification. *arXiv preprint arXiv:1602.03619*, 2016.

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

[28] Jacob Steinhardt, Gregory Valiant, and Moses Charikar. Avoiding imposters and delinquents: Adversarial crowdsourcing and peer prediction. In *Advances in Neural Information Processing Systems*, pages 4439–4447, 2016.

[29] David Steurer and Sam Hopkins. Efficient estimation from few samples: Stochastic block models and related problems. 2017.

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

[31] Jiaming Xu. Rates of Convergence of Spectral Methods for Graphon Estimation. *ArXiv e-prints*, September 2017.

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

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

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