SOCIAL COMPUTATION: FUNDAMENTAL LIMITS AND EFFICIENT ALGORITHMS

BY

ASHISH KUMAR KHETAN

DISSERTATION

Submitted in partial fulfillment of the requirements
for the degree of Doctor of Philosophy in Industrial Engineering
in the Graduate College of the
University of Illinois at Urbana-Champaign, 2018

Urbana, Illinois

Doctoral Committee:

Assistant Professor Sewoong Oh, Chair
Professor Bruce Hajek
Assistant Professor Ruoyu Sun
Professor Pramod Viswanath

ABSTRACT

Social computing systems bring enormous value to society by harnessing the data generated by the members of a community. Though each individual reveals a little information through his online traces, collectively this information gives significant insights on the societal preferences that can be used in designing better systems for the society. Challenging societal problems can be solved using the collective power of a crowd wherein each individual offers only a limited knowledge on a specifically designed online platform. There exists general approaches to design such online platforms, to aggregate the collected data, and to use them for the downstream tasks, but are typically sub-optimal and inefficient. In this work, we investigate several social computing problems and provide efficient algorithms for solving them.

This work studies several topics: (a) designing efficient algorithms for aggregating preferences from partially observed traces of online activities, and characterizing the fundamental trade-off between the computational complexity and statistical efficiency; (b) characterizing the fundamental trade-off between the budget and accuracy in aggregated answers in crowdsourcing systems, and designing efficient algorithms for training supervised learning models using the crowdsourced answers; (c) designing efficient algorithms for estimating fundamental spectral properties of a partially observed data such as a movie rating data matrix in recommendation systems, and connections in a large network.

To my lovely sisters, for their unconditional love and support.

ACKNOWLEDGMENTS

I was extremely fortunate to have Sewoong Oh as my PhD advisor. During the entire PhD journey, I've immensely benefitted from numerous meetings with Sewoong. I could always walk to his office whenever I needed help. He was always ready to help me with the details of the proofs and teach me the advanced maths. I am truly thankful to Sewoong for his patience and support. Further, I learnt the joy and enthusiasm of doing research from Sewoong. The lessons I learnt from him will have a lasting influence on my career and my life. I am proud and honored to be his student.

I also enjoyed the guidance and advice from my committee members, Bruce Hajek, Ruoyu Sun, and Pramod Viswanath.

I was fortunate to have a number of good friends here at UIUC. Some of these friends are Prasanna, Vivek, Reena, Bhargava, Puru, Aditya, Shaileshh and Kiran. Prasanna deserves a special mention for being my all time friend at the campus. Also, I would like to thank Vivek with whom I had numerous discussions on various research problems.

Most importantly, I would like to thank my beloved parents. This thesis would not have been complete without their support and patience.

TABLE OF CONTENTS

CHAPTER 1

INTRODUCTION

This work considers three related problems in social computing. First, we study the problem of rank aggregation from the partially observed preferences. If the collected partial preferences are heterogenous, the existing approaches are either computationally intractable or are statistically inefficient. We characterize the fundamental trade-off between the computational complexity and statistical efficiency, and provide efficient rank-breaking algorithms. Second, we consider a canonical crowdsourcing model and compare the fundamental trade-off between adaptive and non-adaptive task assignment schemes. Further, we also study efficient algorithms for learning supervised models when the annotations are collected through crowdsourcing platforms. Third, we study the problem of estimating spectrum of a partially observed data matrix, below the threshold of its completion.

1.1 Rank Aggregation

In several applications such as electing officials, choosing policies, or making recommendations, we are given partial preferences from individuals over a set of alternatives, with the goal of producing a global ranking that represents the collective preference of the population or the society. This process is referred to as *rank aggregation*. One popular approach is *learning to rank*. Economists have modeled each individual as a rational being maximizing his/her perceived utility. Parametric probabilistic models, known collectively as Random Utility Models (RUMs), have been proposed to model such individual choices and preferences [149]. This allows one to infer the global ranking by learning the inherent utility from individuals' revealed preferences, which are noisy manifestations of the underlying true utility of the alternatives.

Traditionally, learning to rank has been studied under the following data collection scenarios: pairwise comparisons, best-out-of-k comparisons, and k-way comparisons. *Pairwise comparisons* are commonly studied in the classical context of sports matches as well as more recent applications in crowdsourcing, where each worker is presented with a pair of choices and asked to choose the more favorable one. *Best-out-of-k comparisons* data sets are commonly available from purchase history of customers. Typically, a set of k alternatives are offered among which one is chosen or purchased by each customer. This has been widely studied in operations research in the context of modeling customer choices for revenue management and assortment optimization. The *k-way comparisons* are assumed in traditional rank aggregation scenarios, where each person reveals his/her preference as a ranked list over a set of k items. In some real-world elections, voters provide ranked preferences over the whole set of candidates [140]. We refer to these three types of ordinal data collection scenarios as 'traditional' throughout this work.

For such traditional data sets, there are several computationally efficient inference algorithms for finding the Maximum Likelihood (ML) estimates that provably achieve the minimax optimal performance [156, 182, 84]. However, modern data sets can be unstructured. Individual's revealed ordinal preferences can be implicit, such as movie ratings, time spent on the news articles, and whether the user finished watching the movie or not. In crowdsourcing, it has also been observed that humans are more efficient at performing batch comparisons [79], as opposed to providing the full ranking or choosing the top item. This calls for more flexible approaches for rank aggregation that can take such diverse forms of ordinal data into account. For such non-traditional data sets, finding the ML estimate can become significantly more challenging, requiring run-time exponential in the problem parameters.

To avoid such a computational bottleneck, a common heuristic is to resort to *rank-breaking*. The collected ordinal data is first transformed into a bag of pairwise comparisons, ignoring the dependencies that were present in the original data. This is then processed via existing inference algorithms tailored for *independent* pairwise comparisons, hoping that the dependency present in the input data does not lead to inconsistency in estimation. This idea is one of the main motivations for numerous approaches specializing in learning to rank from pairwise comparisons, e.g., [72, 157, 12]. However, such a heuristic of full rank-breaking, where all pairwise comparisons are weighted and treated equally ignoring their dependencies, has been recently shown to introduce inconsistency [13].

The key idea to produce accurate and consistent estimates is to treat the pairwise comparisons unequally, depending on the topology of the collected data. A fundamental question of interest to practitioners is how to choose the weight of each pairwise comparison in order to achieve not only consistency but also the best accuracy, among those consistent estimators using rank-breaking. In Chapter 2, we study how the accuracy of resulting estimate depends on the topology of the data and the weights on the pairwise comparisons. This provides a guideline for the optimal choice of the weights, driven by the topology of the data, that leads to accurate estimates.

However, this computational gain of pairwise rank-breaking comes at the cost of statistical efficiency. [13] showed that if we include all paired comparisons, then the resulting estimate can be statistically inconsistent due to the ignored correlations among the paired orderings, even with infinite samples. In order to get a consistent estimate, [13] provides a rule for choosing which pairs to include, and we provided an estimator that optimizes how to weigh each of those chosen pairs to get the best finite sample complexity bound. However, such a consistent pairwise rank-breaking results in throwing away many of the ordered relations, resulting in significant loss in accuracy. For example, there exist partial rankings such that including any paired relations from them results in a biased estimator. None of the pairwise orderings can be used from such a partial ranking, without making the estimator inconsistent as shown in [12]. Whether we include all paired comparisons or only a subset of consistent ones, there is a significant loss in accuracy. For the precise condition for consistent rank-breaking one can refer to [12, 13, 114].

For general partial orderings, the state-of-the-art approaches operate on either one of the two extreme points on the computational and statistical trade-off. The MLE requires exponential summations to just evaluate the objective function, in the worst case. On the other hand, the pairwise rank-breaking requires only quadratic number of summations, but suffers from significant loss in the sample complexity. Ideally, we would like to give the analyst the flexibility to choose a target computational complexity she is willing to tolerate, and provide an algorithm that achieves the optimal trade-off at the chosen operating point.

In Chapter 3, we introduce a novel *generalized rank-breaking* that bridges the gap between MLE and pairwise rank-breaking. Our approach allows the user the freedom to choose the level of computational resources to be used, and provides an estimator tailored for the desired complexity. We prove that the proposed estimator is tractable and consistent, and provide an upper bound and a lower bound on the error rate in the finite sample regime. The analysis explicitly characterizes the dependence on the topology of the data. This in turn provides a guideline for designing surveys and experiments in practice, in order to maximize the sample efficiency. The proposed generalized rank-breaking mechanism involves set-wise comparisons as opposed to traditional pairwise comparisons. In order to compute the rank-breaking estimate, we generalize the celebrated minorization maximization algorithm for computing maximum likelihood estimate of pairwise comparisons [92] to more general set-wise comparisons and give guarantees on its convergence.

1.2 Crowdsourcing

Crowdsourcing platforms provide labor markets in which pieces of micro-tasks are electronically distributed to any workers who are willing to complete them for a small fee. In typical crowdsourcing scenarios, such as those on Amazon's Mechanical Turk, a requester first posts a collection of tasks, for example a set of images to be labelled. Then, from a pool of workers, whoever is willing can pick up a subset of those tasks and provide her labels for a small amount of payment. Typically, a fixed amount of payment per task is predetermined and agreed upon between the requester and the workers, and hence the worker is paid the amount proportional to the number of tasks she answers. Further, as the verification of the correctness of the answers is difficult, and also as the requesters are afraid of losing reputation among the crowd, requesters typically choose to pay for every label she gets regardless of the correctness of the provided labels. Hence, the budget of the total payments the requester makes to the workers is proportional to the total number of labels she collects.

One of the major issues in such crowdsourcing platforms is label quality assurance. Some workers are spammers trying to make easy money, and even those who are willing to work frequently make mistakes as the reward is small and tasks are tedious. To correct for these errors, a common approach is to introduce redundancy by collecting answers from multiple workers on the same task and aggregating these responses using some schemes such as majority voting. A fundamental problem of interest in such a system is how to maximize the accuracy of thus aggregated answers, while minimizing the cost. Collecting multiple labels per task can improve the accuracy of our estimates, but increases the budget proportionally. Given a fixed number of tasks to be labelled, a requester hopes to achieve the best trade-off between the accuracy, i.e. the average probability of error in aggregated responses with respect to the ground truth labels, and the budget, i.e. the total number of responses the requester collects on the crowdsourcing platform. There are two design choices the requester has in achieving this goal: *task assignment* and *inference algorithm*.

In typical crowdsourcing platforms, tasks are assigned as follows. Since the workers are fleeting, the requester has no control over who will be the next arriving worker. Workers arrive in an online fashion, complete the tasks that they are given, and leave. Each arriving worker is completely new and you may never get her back. Nevertheless, it might be possible to improve accuracy under the same budget, by designing better task assignments. The requester has the following control over the *task assignment*. At each point in time, we have the control over which tasks to assign to the next arriving worker. The requester is free to use all the information collected thus far, including all the task assignments to previous workers and the answers collected on those assigned tasks. By adaptively identifying tasks that are more difficult and assigning more (future) workers on those tasks, one hopes to be more efficient in the budget-accuracy trade-off. In this work, we make this intuition precise, by studying a canonical crowdsourcing model and comparing the fundamental trade-offs between adaptive schemes and non-adaptive schemes. Unlike adaptive schemes, a non-adaptive scheme fixes all the task assignments before any labels are collected and does not allow future assignments to adapt to the labels collected thus far for each arriving worker.

Adaptive schemes, where tasks are assigned adaptively based on the data collected thus far, are widely used in practical crowdsourcing systems to efficiently use a given fixed budget. However, existing theoretical analyses of crowdsourcing systems suggest that the gain of adaptive task assignments is minimal. To bridge this gap, in Chapter 4, we investigate this question under a strictly more general probabilistic model, which has been recently introduced to model practical crowdsourced annotations. Under this generalized Dawid-Skene model, we characterize the fundamental trade-off between budget and accuracy. We introduce a novel adaptive scheme that matches this fundamental limit. We further quantify the fundamental gap between adaptive and non-adaptive schemes, by comparing the trade-off with the one for non-adaptive schemes. Our analyses confirm that the gap is significant.

The downstream goal of many crowdsourcing projects is to train supervised learning models. Supervised learning requires large annotated datasets which due to economic reasons cannot be collected alone from the experts. Since crowdsourcing platforms such as Amazon Mechanical Turk (AMT), provide access to low-skilled workers who can perform simple tasks, such as classifying images, at low cost, most practitioners turn to these platforms for collecting annotations for training supervised learning models.

Compared to experts, crowd-workers provide noisier annotations, possibly owing to high variation in worker skill; and a per-answer compensation structure that encourages rapid answers, even at the expense of accuracy. To address variation in worker skill, practitioners typically collect multiple independent labels for each training example from different workers. In practice, these labels are often aggregated by applying a simple majority vote. Academics have proposed many efficient algorithms for estimating the ground truth from noisy annotations. Research addressing the crowd-sourcing problem goes back to the early 1970s. [47] proposed a probabilistic model to jointly estimate worker skills and ground truth labels and used expectation maximization (EM) to estimate the parameters. [208, 205, 222] proposed generalizations of the Dawid-Skene model, e.g. by estimating the difficulty of each example.

However, crowdsourcing research seldom accounts for the downstream utility of the produced annotations as training data in machine learning (ML) algorithms. And ML research seldom exploits the noisy labels collected from multiple human workers. A few recent papers use the original noisy labels and the corresponding worker identities together with the predictions of a supervised learning model trained on those same labels, to estimate the ground truth [28, 83, 205]. However, these papers do not realize the full potential of combining modeling and crowd-sourcing. In particular, they are unable to estimate worker qualities when there is only one label per training example.

In Chapter 5, we present a new supervised learning algorithm that alternately models the labels and worker quality. The EM algorithm bootstraps itself in the following way: Given a trained model, the algorithm estimates worker qualities using the disagreement between workers and the current predictions of the learning algorithm. Given estimated worker qualities, our algorithm optimizes a suitably modified loss function. We show that accurate estimates of worker quality can be obtained even when only collecting one label per example provided that each worker labels sufficiently many examples. An accurate estimate of the worker qualities leads to learning a better model. This addresses a shortcoming of the prior work and overcomes a significant hurdle to achieving practical crowdsourcing without redundancy.

We give theoretical guarantees on the performance of our algorithm. We analyze the two alternating steps: (a) estimating worker qualities from disagreement with the model, (b) learning a model by optimizing the modified loss function. We obtain a bound on the accuracy of the estimated worker qualities and the generalization error of the model. Through the generalization error bound, we establish that it is better to label many examples once than to label less examples multiply when worker quality is above a threshold. Empirically, we verify our approach on several multi-class classification datasets: ImageNet and CIFAR10 (with simulated noisy workers), and MS-COCO (using the real noisy annotator labels). Our experiments validate that when the cost of obtaining unlabeled examples is negligible and the total annotation budget is fixed, it is best to collect a single label per training example for as many examples as possible. Although this work applies our approach to classification problems, the main ideas of the algorithm can be extended to other tasks in supervised learning.

1.3 Spectrum Estimation

Computing and analyzing the set of singular values of a data in a matrix form, which is called the spectrum, provide insights into the geometry and topology of the data. Such a spectral analysis is routinely a first step in general data analysis with the goal of checking if there exists a lower dimensional subspace explaining the important aspects of the data, which itself might be high dimensional. Concretely, it is a first step in dimensionality reduction methods such as principal component analysis or canonical correlation analysis.

However, spectral analysis becomes challenging in practical scenarios where the data is only partially observed. We commonly observe pairwise relations of randomly chosen pairs: each user only rates a few movies in recommendation systems, each player/team only plays against a few opponents in sports leagues, each word appears in the same sentence with a small number of other words in word count matrices, and each worker answers a few questions in crowdsourcing. In other applications, we have more structured samples. For example, in a network analysis we might be interested in the spectrum of a large network, but only get to see the connections within a small subset of nodes corresponding to sampling a sub-matrix of the adjacency matrix. Whatever the sampling pattern is, typical number of paired relations we observe is significantly smaller than the dimension of the data matrix.

In Chapter 6, we study all such variations in sampling patterns for partially observed data matrices, and ask the following fundamental question: *can we estimate spectral properties of a data matrix from partial observations?* We propose a novel approach that allows us to estimate the spectrum, i.e. the singular values. A crucial building block in our approach is that spectral properties can be accurately approximated from the first few moments of the spectrum known as the Schatten k-norms defined as

$$\|M\|_k \;=\; \Big(\sum_{i=1}^{d} \sigma_i(M)^k\Big)^{1/k}\;, \tag{1.1}$$

where $\sigma_1(M) \geq \sigma_2(M) \geq \cdots \geq \sigma_d(M) \geq 0$ are the singular values of the data matrix $M \in \mathbb{R}^{d\times d}$. Once we obtain accurate estimates of Schatten k-norms, these estimates, as well as corresponding performance guarantees, can readily be translated into accurate estimates of the spectrum of the matrix. Further, if we are interested in estimating a class of functions known as spectral sum functions, our estimates of the Schatten norms can be used to estimate any spectral sum function using Chebyshev expansions. Our theoretical analysis shows that Schatten norms can be recovered accurately from strictly smaller number of samples compared to what is needed to recover the underlying low-rank matrix. Numerical experiments suggest that we significantly improve upon a competing approach of using matrix completion methods, below the matrix completion threshold, above which matrix completion algorithms recover the underlying low-rank matrix exactly.

We want to estimate the Schatten k-norm of a positive semidefinite matrix $M \in \mathbb{R}^{d\times d}$ from a subset of its entries. The restriction to positive semidefinite matrices is primarily for notational convenience, and our analyses, the estimator, and the efficient algorithms naturally generalize to any non-square matrices. Namely, we can extend our framework to bipartite graphs and estimate Schatten k-norm of any matrix for any even k. Let Ω denote the set of indices of samples we are given and let $\mathcal{P}_\Omega(M) = \{(i, j, M_{ij})\}_{(i,j)\in\Omega}$ denote the samples. With a slight abuse of notation, we used $\mathcal{P}_\Omega(M)$ to also denote the $d \times d$ sampled matrix:

$$\mathcal{P}_\Omega(M)_{ij} = \begin{cases} M_{ij} & \text{if } (i,j) \in \Omega \ , \\ 0 & \text{otherwise} \ , \end{cases}$$

and it should be clear from the context which one we refer to. Although we propose a framework that generally applies to any probabilistic sampling, it is necessary to propose specific sampling scenarios to provide tight analyses on the performance. Hence, we focus on *Erdös-Rényi sampling*.

There is an extensive line of research in low-rank matrix completion problems [31, 110], which addresses a fundamental question of how many samples are required to *complete* a matrix (i.e. estimate all the missing entries) from a small subset of sampled entries. It is typically assumed that each entry of the matrix is sampled independently with a probability $p \in (0, 1]$. We refer to this scenario as *Erdös-Rényi sampling*, as the resulting pattern of the samples encoded as a graph is distributed as an Erdös-Rényi random graph. The spectral properties of such an sampled matrix have been well studied in the literature [75, 1, 69, 110, 123]. In particular, it is known that the original matrix is close in spectral norm to the sampled one where the missing entries are filled in with zeros and properly rescaled under certain incoherence assumptions. This suggests using the singular values of the sampled and rescaled matrix $(d^2/|\Omega|)\mathcal{P}(M)$ directly for estimating the Schatten norms. However, in the sub-linear regime in which the number of samples $|\Omega| = d^2p$ is comparable to or significantly smaller than the degrees of freedom in representing a symmetric rank-r matrix, which is $dr - r^2$, the spectrum of the sampled matrix is significantly different from the spectrum of the original matrix. In this work, we design novel estimators that are more sample efficient in the sub-linear regime where $d^2p \ll dr$.

Further, in Chapter 7 we give a novel application of the Schatten norm estimation techniques. We give a new estimator for estimating the number of connected components in a graph by sampling a subgraph. It is a challenging problem with no existing estimator that works for general graphs. The connection between the observed subgraph and the number of connected components has remained a mystery. In order to make this connection transparent, we propose a highly redundant and large-dimensional representation of the subgraph, which at first glance seems counter-intuitive. A subgraph is represented by the counts of patterns, known as network motifs. This representation is crucial in introducing a novel estimator for the number of connected components for general graphs. The connection is made precise via the Schatten k-norms of the graph Laplacian and the spectral representation of the number of connected components. We provide a guarantee on the resulting mean squared error that characterizes the bias variance tradeoff. Experiments on special structured graphs suggest that we improve upon competing algorithms tailored for those structures for a broad range of parameters.

CHAPTER 2

DATA-DRIVEN RANK BREAKING FOR EFFICIENT RANK AGGREGATION

Initially motivated by elections and voting, rank aggregation has been a topic of mathematical interest dating back to Condorcet and Borda [49, 48]. Using probabilistic models to infer preferences has been popularized in operations research community for applications such as assortment optimization and revenue management. The PL model studied in this paper is a special case of MultiNomial Logit (MNL) models commonly used in discrete choice modeling, which has a long history in operations research [149]. Efficient inference algorithms has been proposed to either find the MLE efficiently or approximately, such as the iterative approaches in [72, 59], minorization-maximization approach in [92], and Markov chain approaches in [156, 146]. These approaches are shown to achieve minimax optimal error rate in the traditional comparisons scenarios. Under the pairwise comparisons scenario, Negahban et al. [156] provided Rank Centrality that provably achieves minimax optimal error rate for randomly chosen pairs, which was later generalized to arbitrary pairwise comparisons in [157]. The analysis shows the explicit dependence on the topology of data shows that the spectral gap of comparisons graph similar to the one presented in this paper. This analysis was generalized to k-way comparisons in [84] and generalized to best-out-of-k comparisons with sharper bounds in [182]. In an effort to give a guarantee for exact recovery of the top-ℓ items in the ranking, Chen et al. in [39] proposed a new algorithm based on Rank Centrality that provides a tighter error bound for L_∞ norm, as opposed to the existing L_2 error bounds. Another interesting direction in learning to rank is non-parametric learning from paired comparisons, initiated in several recent papers such as [57, 171, 183, 186].

More recently, a more general problem of learning *personal* preferences from ordinal data has been studied [213, 136, 55]. The MNL model provides a natural generalization of the PL model to this problem. When users are classified into a small number of groups with same preferences, mixed MNL model can be learned from data as studied in [7, 160, 212]. A more general scenario is when each user has his/her individual preferences, but inherently represented by a lower dimensional feature. This problem was first posed as an inference problem in [138] where convex relaxation of nuclear norm minimization was proposed with provably optimal guarantees. This was later generalized to k-way comparisons in [161]. A similar approach was studied with a different guarantees and assumptions in [164]. Our algorithm and ideas of rank-breaking can be directly applied to this collaborative ranking under MNL, with the same guarantees for consistency in the asymptotic regime where sample size grows to infinity. However, the analysis techniques for MNL rely on stronger assumptions on how the data is collected, and especially on the independence of the samples. It is not immediate how the analysis techniques developed in this paper can be applied to learn MNL.

In an orthogonal direction, new discrete choice models with sparse structures has been proposed recently in [67] and optimization algorithms for revenue management has been proposed [68]. In a similar direction, new discrete choice models based on Markov chains has been introduced in [21], and corresponding revenue management algorithms has been studied in [70]. However, typically these models are analyzed in the asymptotic regime with infinite samples, with the exception of [8]. A non-parametric choice models for pairwise comparisons also have been studied in [171, 183]. This provides an interesting opportunities to studying learning to rank for these new choice models.

We consider a fixed design setting, where inference is separate from data collection. There is a parallel line of research which focuses on adaptive ranking, mainly based on pairwise comparisons. When performing sorting from noisy pairwise comparisons, Braverman et al. in [30] proposed efficient approaches and provided performance guarantees. Following this work, there has been recent advances in adaptive ranking [3, 98, 147].

2.1 Problem formulation.

The premise in the current race to collect more data on user activities is that, a hidden true preference manifests in the user's activities and choices. Such data can be explicit, as in ratings, ranked lists, pairwise comparisons, and like/dislike buttons. Others are more implicit, such as purchase history and viewing times. While more data in general allows for a more accurate inference, the heterogeneity of user activities makes it difficult to infer the underlying preferences directly. Further, each user reveals her preference on only a few contents.

Traditional collaborative filtering fails to capture the diversity of modern data sets. The sparsity and heterogeneity of the data renders typical similarity measures ineffective in the nearest-neighbor methods. Consequently, simple measures of similarity prevail in practice, as in Amazon's "people who bought ... also bought ..." scheme. Score-based methods require translating heterogeneous data into numeric scores, which is a priori a difficult task. Even if explicit ratings are observed, those are often unreliable and the scale of such ratings vary from user to user.

We propose aggregating ordinal data based on users' revealed preferences that are expressed in the form of *partial orderings* (notice that our use of the term is slightly different from its original use in revealed preference theory). We interpret user activities as manifestation of the hidden preferences according to discrete choice models (in particular the Plackett-Luce model defined in (2.1)). This provides a more reliable, scale-free, and widely applicable representation of the heterogeneous data as partial orderings, as well as a probabilistic interpretation of how preferences manifest. In full generality, the data collected from each individual can be represented by a *partially ordered set (poset)*. Assuming consistency in a user's revealed preferences, any ordered relations can be seamlessly translated into a poset, represented as a Hasse diagram by a directed acyclic graph (DAG). The DAG below represents ordered relations $a > \{b, d\}$, $b > c$, $\{c, d\} > e$, and $e > f$. For example, this could have been translated from two sources: a five star rating on a and a three star ratings on b, c, d, a two star rating on e, and a one star rating on f; and the item b being purchased after reviewing c as well.

There are n users or agents, and each agent j provides his/her ordinal evaluation on a subset S_j of d items or alternatives. We refer to $S_j \subset \{1, 2, \ldots, d\}$ as *offerings* provided to j, and use $\kappa_j = |S_j|$ to denote the size of the offerings. We assume that the partial ordering over the offerings is a manifestation of her preferences as per a popular choice model known as Plackett-Luce (PL) model. As we explain in detail below, the PL model produces total orderings (rather than partial ones). The data collector queries each user for a partial

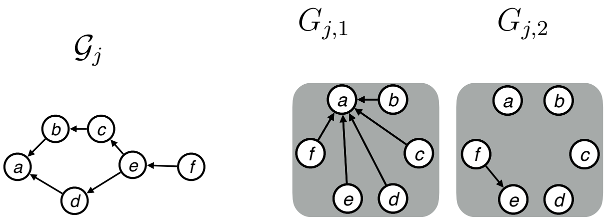


Figure 2.1: A DAG representation of consistent partial ordering of a user j, also called a Hasse diagram (left). A set of rank-breaking graphs extracted from the Hasse diagram for the separator item a and e, respectively (right).

ranking in the form of a poset over S_j. For example, the data collector can ask for the top item, unordered subset of three next preferred items, the fifth item, and the least preferred item. In this case, an example of such poset could be $a < \{b, c, d\} < e < f$, which could have been generated from a total ordering produced by the PL model and taking the corresponding partial ordering from the total ordering. Notice that we fix the topology of the DAG first and ask the user to fill in the node identities corresponding to her total ordering as (randomly) generated by the PL model. Hence, the structure of the poset is considered deterministic, and only the identity of the nodes in the poset is considered random. Alternatively, one could consider a different scenario where the topology of the poset is also random and depends on the outcome of the preference, which is out-side the scope of this paper and provides an interesting future research direction.

The PL model is a special case of *random utility models*, defined as follows [201, 14]. Each item i has a real-valued latent utility θ_i. When presented with a set of items, a user's reveled preference is a partial ordering according to noisy manifestation of the utilities, i.e. i.i.d. noise added to the true utility θ_i's. The PL model is a special case where the noise follows the standard Gumbel distribution, and is one of the most popular model in social choice theory [148, 150]. PL has several important properties, making this model realistic in various domains, including marketing [82], transportation [149, 15], biology [188], and natural language processing [153]. Precisely, each user j, when presented with a set S_j of items, draws a noisy utility of each item i according to

$$u_i \;=\; \theta_i + Z_i \;,$$

where Z_i's follow the independent standard Gumbel distribution. Then we observe the ranking resulting from sorting the items as per noisy observed utilities u_j's. Alternatively, the PL model is also equivalent to the following random process. For a set of alternatives S_j, a ranking $\sigma_j : [|S|] \to S$ is generated in two steps: (1) independently assign each item $i \in S_j$ an unobserved value X_i, exponentially distributed with mean $e^{-\theta_i}$; (2) select a ranking σ_j so that $X_{\sigma_j(1)} \leq X_{\sigma_j(2)} \leq \cdots \leq X_{\sigma_j(|S_j|)}$.

The PL model (i) satisfies Luce's 'independence of irrelevant alternatives' in social choice theory [173], and has a simple characterization as sequential (random) choices as explained below; and (ii) has a maximum likelihood estimator (MLE) which is a convex program in θ in the traditional scenarios of pairwise, best-out-of-k and k-way comparisons. Let $\mathbb{P}(a > \{b, c, d\})$ denote the probability a was chosen as the best alternative among the set $\{a, b, c, d\}$. Then, the probability that a user reveals a linear order $(a > b > c > d)$ is equivalent as making sequential choice from the top to bottom:

$$\begin{aligned} \mathbb{P}(a > b > c > d) &= \mathbb{P}(a > \{b, c, d\})\ \mathbb{P}(b > \{c, d\})\ \mathbb{P}(c > d) \\ &= \frac{e^{\theta_a}}{(e^{\theta_a} + e^{\theta_b} + e^{\theta_c} + e^{\theta_d})} \frac{e^{\theta_b}}{(e^{\theta_b} + e^{\theta_c} + e^{\theta_d})} \frac{e^{\theta_c}}{(e^{\theta_c} + e^{\theta_d})} \ . \end{aligned}$$

We use the notation $(a > b)$ to denote the event that a is preferred over b. In general, for user j presented with offerings S_j, the probability that the revealed preference is a total ordering σ_j is

$$\mathbb{P}(\sigma_j) = \prod_{i \in \{1, \ldots, \kappa_j - 1\}} (e^{\theta_{\sigma^{-1}(i)}})/(\sum_{i'=i}^{\kappa_j} e^{\theta_{\sigma^{-1}(i')}}).$$

We consider the true utility $\theta^* \in \Omega_b$, where we define Ω_b as

$$\Omega_b \ \equiv \ \Big\{ \theta \in \mathbb{R}^d \,\Big|\, \sum_{i \in [d]} \theta_i = 0\,,\ |\theta_i| \leq b \text{ for all } i \in [d] \ \Big\}\,.$$

Note that by definition, the PL model is invariant under shifting the utility θ_i's. Hence, the centering ensures uniqueness of the parameters for each PL model. The bound b on the dynamic range is not a restriction, but is written explicitly to capture the dependence of the accuracy in our main results.

We have n users each providing a partial ordering of a set of offerings S_j according to the PL model. Let $\mathcal{G}_j$ denote both the DAG representing the partial ordering from user j's preferences. With a slight abuse of notations, we also let $\mathcal{G}_j$ denote the set of rankings that are consistent with this DAG. For general partial orderings, the probability of observing $\mathcal{G}_j$ is the sum of all total orderings that is consistent with the observation, i.e. $\mathbb{P}(\mathcal{G}_j) = \sum_{\sigma \in \mathcal{G}_j} \mathbb{P}(\sigma)$. The goal is to efficiently learn the true utility $\theta^* \in \Omega_b$, from the n sampled partial orderings. One popular approach is to compute the maximum likelihood estimate (MLE) by solving the following optimization:

$$\underset{\theta \in \Omega_b}{\text{maximize}} \qquad \sum_{j=1}^{n} \log \mathbb{P}(\mathcal{G}_j) \; .$$

This optimization is a simple convex optimization, in particular a logit regression, when the structure of the data $\{\mathcal{G}_j\}_{j \in [n]}$ is traditional. This is one of the reasons the PL model is attractive. However, for general posets, this can be computationally challenging. Consider an example of position-p ranking, where each user provides which item is at p-th position in his/her ranking. Each term in the log-likelihood for this data involves summation over $O((p-1)!)$ rankings, which takes $O(n\,(p-1)!)$ operations to evaluate the objective function. Since p can be as large as d, such a computational blow-up renders MLE approach impractical. A common remedy is to resort to rank-breaking, which might result in inconsistent estimates.

Rank-breaking. Rank-breaking refers to the idea of extracting a set of pairwise comparisons from the observed partial orderings and applying estimators tailored for paired comparisons treating each piece of comparisons as independent. Both the choice of which paired comparisons to extract and the choice of parameters in the estimator, which we call *weights*, turns out to be crucial as we will show. Inappropriate selection of the paired comparisons can lead to inconsistent estimators as proved in [13], and the standard choice of the parameters can lead to a significantly suboptimal performance.

A naive rank-breaking that is widely used in practice is to apply rank-breaking to all possible pairwise relations that one can read from the partial ordering and weighing them equally. We refer to this practice as *full rank-breaking*. In the example in Figure 2.1, full rank-breaking first extracts the bag of comparisons $\mathcal{C} = \{(a > b), (a > c), (a > d), (a > e), (a > f), \ldots, (e > f)\}$ with 13 paired comparison outcomes, and apply the maximum likelihood estimator treating each paired outcome as independent. Precisely, the *full* *rank-breaking estimator* solves the convex optimization of

$$\widehat{\theta} \;\in\; \arg\max_{\theta \in \Omega_b} \sum_{(i > i') \in \mathcal{C}} \Big(\theta_i - \log \Big(e^{\theta_i} + e^{\theta_{i'}} \Big) \Big) \,. \tag{2.1}$$

There are several efficient implementation tailored for this problem [72, 92, 156, 146], and under the traditional scenarios, these approaches provably achieve the minimax optimal rate [84, 182]. For general non-traditional data sets, there is a significant gain in computational complexity. In the case of position-p ranking, where each of the n users report his/her p-th ranking item among κ items, the computational complexity reduces from $O(n\,(p-1)!)$ for the MLE in (2.1) to $O(n\,p\,(\kappa - p))$ for the full rank-breaking estimator in (2.1). However, this gain comes at the cost of accuracy. It is known that the full-rank breaking estimator is inconsistent [13]; the error is strictly bounded away from zero even with infinite samples.

Perhaps surprisingly, Azari Soufiani et al. [13] recently characterized the entire set of consistent rank-breaking estimators. Instead of using the bag of paired comparisons, the sufficient information for consistent rank-breaking is a set of rank-breaking graphs defined as follows.

Recall that a user j provides his/her preference as a poset represented by a DAG $\mathcal{G}_j$. Consistent rank-breaking first identifies all *separators* in the DAG. A node in the DAG is a separator if one can partition the rest of the nodes into two parts. A partition $A_{\rm top}$ which is the set of items that are preferred over the separator item, and a partition $A_{\rm bottom}$ which is the set of items that are less preferred than the separator item. One caveat is that we allow $A_{\rm top}$ to be empty, but $A_{\rm bottom}$ must have at least one item. In the example in Figure 2.1, there are two separators: the item a and the item e. Using these separators, one can extract the following partial ordering from the original poset: $(a > \{b, c, d\} > e > f)$. The items a and e separate the set of offerings into partitions, hence the name separator. We use ℓ_j to denote the number of separators in the poset $\mathcal{G}_j$ from user j. We let $p_{j,a}$ denote the ranked position of the a-th separator in the poset $\mathcal{G}_j$, and we sort the positions such that $p_{j,1} < p_{j,2} < \ldots < p_{j,\ell_j}$. The set of separators is denoted by $\mathcal{P}_j = \{p_{j,1}, p_{j,2}, \cdots, p_{j,\ell_j}\}$. For example, since the separator a is ranked at position 1 and e is at the 5-th position, $\ell_j = 2$, $p_{j,1} = 1$, and $p_{j,2} = 5$. Note that f is not a separator (whereas a is) since corresponding $A_{\rm bottom}$ is empty.

Conveniently, we represent this extracted partial ordering using a set of DAGs, which are called *rank-breaking graphs*. We generate one rank-breaking graph per separator. A rank breaking graph $G_{j,a} = (S_j, E_{j,a})$ for user j and the a-th separator is defined as a directed graph over the set of offerings S_j, where we add an edge from a node that is less preferred than the a-th separator to the separator, i.e. $E_{j,a} = \{(i, i') \,|\, i' \text{ is the } a\text{-th separator, and } \sigma_j^{-1}(i) > p_{j,a}\}$. Note that by the definition of the separator, $E_{j,a}$ is a non-empty set. An example of rank-breaking graphs are shown in Figure 2.1.

This rank-breaking graphs were introduced in [12], where it was shown that the pairwise ordinal relations that is represented by edges in the rank-breaking graphs are sufficient information for using any estimation based on the idea of rank-breaking. Precisely, on the converse side, it was proved in [13] that any pairwise outcomes that is not present in the rank-breaking graphs $G_{j,a}$'s lead to inconsistency for a general θ^*. On the achievability side, it was proved that all pairwise outcomes that are present in the rank-breaking graphs give a consistent estimator, as long as all the paired comparisons in each $G_{j,a}$ are weighted equally.

It should be noted that rank-breaking graphs are defined slightly differently in [12]. Specifically, [12] introduced a different notion of rank-breaking graph, where the vertices represent positions in total ordering. An edge between two vertices i_1 and i_2 denotes that the pairwise comparison between items ranked at position i_1 and i_2 is included in the estimator. Given such observation from the PL model, [12] and [13] prove that a rank-breaking graph is consistent if and only if it satisfies the following property. If a vertex i_1 is connected to any vertex i_2, where $i_2 > i_1$, then i_1 must be connected to all the vertices i_3 such that $i_3 > i_1$. Although the specific definitions of rank-breaking graphs are different from our setting, the mathematical analysis of [12] still holds when interpreted appropriately. Specifically, we consider only those rank-breaking that are consistent under the conditions given in [12]. In our rank-breaking graph $G_{j,a}$, a separator node is connected to all the other item nodes that are ranked below it (numerically higher positions).

In the algorithm described in (2.33), we satisfy this sufficient condition for consistency by restricting to a class of convex optimizations that use the same weight $\lambda_{j,a}$ for all $(\kappa - p_{j,a})$ paired comparisons in the objective function, as opposed to allowing more general weights that defer from a pair to another pair in a rank-breaking graph $G_{j,a}$.

Algorithm. Consistent rank-breaking first identifies separators in the collected posets $\{\mathcal{G}_j\}_{j\in[n]}$ and transform them into rank-breaking graphs $\{G_{j,a}\}_{j\in[n],a\in[\ell_j]}$ as explained above. These rank-breaking graphs are input to the MLE for paired comparisons, assuming all directed edges in the rank-breaking graphs are independent outcome of pairwise comparisons. Precisely, the *consistent rank-breaking estimator* solves the convex optimization of maximizing the paired log likelihoods

$$\mathcal{L}_{\rm RB}(\theta) \;=\; \sum_{j=1}^{n}\sum_{a=1}^{\ell_j} \lambda_{j,a} \Big\{ \sum_{(i,i')\in E_{j,a}} \Big(\theta_{i'} - \log\Big(e^{\theta_i} + e^{\theta_{i'}}\Big)\Big) \Big\} \;, \qquad (2.2)$$

where $E_{j,a}$'s are defined as above via separators and different choices of the non-negative weights $\lambda_{j,a}$'s are possible and the performance depends on such choices. Each weight $\lambda_{j,a}$ determine how much we want to weigh the contribution of a corresponding rank-breaking graph $G_{j,a}$. We define the *consistent rank-breaking estimate* $\widehat{\theta}$ as the optimal solution of the convex program:

$$\widehat{\theta} \;\in\; \arg\max_{\theta\in\Omega_b} \; \mathcal{L}_{\rm RB}(\theta)\;. \qquad (2.3)$$

By changing how we weigh each rank-breaking graph (by choosing the $\lambda_{j,a}$'s), the convex program (2.3) spans the entire set of consistent rank-breaking estimators, as characterized in [13]. However, only asymptotic consistency was known, which holds independent of the choice of the weights $\lambda_{j,a}$'s. Naturally, a uniform choice of $\lambda_{j,a} = \lambda$ was proposed in [13].

Note that this can be efficiently solved, since this is a simple convex optimization, in particular a logit regression, with only $O(\sum_{j=1}^{n} \ell_j\,\kappa_j)$ terms. For a special case of position-p breaking, the $O(n\,(p-1)!)$ complexity of evaluating the objective function for the MLE is now significantly reduced to $O(n\,(\kappa-p))$ by rank-breaking. Given this potential exponential gain in efficiency, a natural question of interest is "what is the price we pay in the accuracy?". We provide a sharp analysis of the performance of rank-breaking estimators in the finite sample regime, that quantifies the price of rank-breaking. Similarly, for a practitioner, a core problem of interest is how to choose the weights in the optimization in order to achieve the best accuracy. Our analysis provides a data-driven guideline for choosing the optimal weights.

Contributions. In this paper, we provide an upper bound on the error achieved by the rank-breaking estimator of (2.3) for any choice of the weights in Theorem 7.3. This explicitly shows how the error depends on the choice of the weights, and provides a guideline for choosing the optimal weights $\lambda_{j,a}$'s in a data-driven manner. We provide the explicit formula for the optimal choice of the weights and provide the the error bound in Theorem 2.2. The analysis shows the explicit dependence of the error in the problem dimension d and the number of users n that matches the numerical experiments.

If we are designing surveys and can choose which subset of items to offer to each user and also can decide which type of ordinal data we can collect, then we want to design such surveys in a way to maximize the accuracy for a given number of questions asked. Our analysis provides how the accuracy depends on the topology of the collected data, and provides a guidance when we do have some control over which questions to ask and which data to collect. One should maximize the spectral gap of corresponding comparison graph. Further, for some canonical scenarios, we quantify the price of rank-breaking by comparing the error bound of the proposed data-driven rank-breaking with the lower bound on the MLE, which can have a significantly larger computational cost (Theorem 2.4).

Notations. Following is a summary of all the notations defined above. We use d to denote the total number of items and index each item by $i \in \{1, 2, \dots, d\}$. $\theta \in \Omega_b$ denotes vector of utilities associated with each item. θ^* represents true utility and $\widehat{\theta}$ denotes the estimated utility. We use n to denote the number of users/agents and index each user by $j \in \{1, 2, \dots, n\}$. $S_j \subseteq \{1, \dots, d\}$ refer to the offerings provided to the j-th user and we use $\kappa_j = |S_j|$ to denote the size of the offerings. $\mathcal{G}_j$ denote the DAG (Hasse diagram) representing the partial ordering from user j's preferences. $\mathcal{P}_j = \{p_{j,1}, p_{j,2}, \cdots, p_{j,\ell_j}\}$ denotes the set of separators in the DAG $\mathcal{G}_j$, where $p_{j,1}, \cdots, p_{j,\ell_j}$ are the positions of the separators, and ℓ_j is the number of separators. $G_{j,a} = (S_j, E_{j,a})$ denote the rank-breaking graph for the a-th separator extracted from the partial ordering $\mathcal{G}_j$ of user j.

For any positive integer N, let $[N] = \{1, \cdots, N\}$. For a ranking σ over S, i.e., σ is a mapping from $[|S|]$ to S, let σ^{-1} denote the inverse mapping.For a vector x, let $\|x\|_2$ denote the standard l_2 norm. Let $\mathbf{1}$ denote the all-ones vector and $\mathbf{0}$ denote the all-zeros vector with the appropriate dimension. Let $\mathcal{S}^d$ denote the set of $d \times d$ symmetric matrices with real-valued entries. For $X \in \mathcal{S}^d$, let $\lambda_1(X) \le \lambda_2(X) \le \cdots \le \lambda_d(X)$ denote its eigenvalues sorted in increasing order. Let $\mathrm{Tr}(X) = \sum_{i=1}^d \lambda_i(X)$ denote its trace and $\|X\| = \max\{|\lambda_1(X)|, |\lambda_d(X)|\}$ denote its spectral norm. For two matrices $X, Y \in \mathcal{S}^d$, we write $X \succeq Y$ if $X - Y$ is positive semi-definite, i.e., $\lambda_1(X - Y) \ge 0$. Let e_i denote a unit vector in $\mathbb{R}^d$ along the i-th direction.

2.2 Comparisons Graph and the Graph Laplacian

In the analysis of the convex program (2.3), we show that, with high probability, the objective function is strictly concave with $\lambda_2(H(\theta)) \le -C_b\, \gamma\, \lambda_2(L) < 0$ (Lemma 2.11) for all $\theta \in \Omega_b$ and the gradient is bounded by $\|\nabla \mathcal{L}_{\mathrm{RB}}(\theta^*)\|_2 \le C'_b \sqrt{\log d \sum_{j \in [n]} \ell_j}$ (Lemma 2.10). Shortly, we will define γ and $\lambda_2(L)$, which captures the dependence on the topology of the data, and C'_b and C_b are constants that only depend on b. Putting these together, we will show that there exists a $\theta \in \Omega_b$ such that

$$\|\widehat{\theta} - \theta^*\|_2 \;\le\; \frac{2\|\nabla \mathcal{L}_{\mathrm{RB}}(\theta^*)\|_2}{-\lambda_2(H(\theta))} \;\le\; C''_b \frac{\sqrt{\log d \sum_{j \in [n]} \ell_j}}{\gamma\, \lambda_2(L)} \,.$$

Here $\lambda_2(H(\theta))$ denotes the second largest eigenvalue of a negative semi-definite Hessian matrix $H(\theta)$ of the objective function. The reason the second largest eigenvalue shows up is because the top eigenvector is always the all-ones vector which by the definition of Ω_b is infeasible. The accuracy depends on the topology of the collected data via the comparison graph of given data.

Definition 2.1. *(Comparison graph $\mathcal{H}$). We define a graph $\mathcal{H}([d], E)$ where each alternative corresponds to a node, and we put an edge (i, i') if there exists an agent j whose offerings is a set S_j such that $i, i' \in S_j$. Each edge $(i, i') \in E$ has a weight $A_{ii'}$ defined as*

$$A_{ii'} \;=\; \sum_{j \in [n] : i, i' \in S_j} \frac{\ell_j}{\kappa_j(\kappa_j - 1)} \,,$$

where $\kappa_j = |S_j|$ is the size of each sampled set and ℓ_j is the number of separators in S_j defined by rank-breaking.

Define a diagonal matrix $D = \text{diag}(A\mathbf{1})$, and the corresponding graph Laplacian $L = D - A$, such that

$$L \;=\; \sum_{j=1}^{n} \frac{\ell_j}{\kappa_j(\kappa_j - 1)} \sum_{i<i' \in S_j} (e_i - e_{i'})(e_i - e_{i'})^\top . \tag{2.4}$$

Let $0 = \lambda_1(L) \leq \lambda_2(L) \leq \cdots \leq \lambda_d(L)$ denote the (sorted) eigenvalues of L. Of special interest is $\lambda_2(L)$, also called the spectral gap, which measured how well-connected the graph is. Intuitively, one can expect better accuracy when the spectral gap is larger, as evidenced in previous learning to rank results in simpler settings [157, 182, 84]. This is made precise in (2.4), and in the main result of Theorem 2.2, we appropriately rescale the spectral gap and use $\alpha \in [0, 1]$ defined as

$$\alpha \;\equiv\; \frac{\lambda_2(L)(d-1)}{\text{Tr}(L)} \;=\; \frac{\lambda_2(L)(d-1)}{\sum_{j=1}^{n} \ell_j} \,. \tag{2.5}$$

The accuracy also depends on the topology via the maximum weighted degree defined as $D_{\max} \equiv \max_{i\in[d]} D_{ii} = \max_{i\in[d]}\{\sum_{j:i\in S_j} \ell_j/\kappa_j\}$. Note that the average weighted degree is $\sum_i D_{ii}/d = \text{Tr}(L)/d$, and we rescale it by $D_{\max}$ such that

$$\beta \;\equiv\; \frac{\text{Tr}(L)}{dD_{\max}} \;=\; \frac{\sum_{j=1}^{n} \ell_j}{dD_{\max}} \,. \tag{2.6}$$

We will show that the performance of rank breaking estimator depends on the topology of the graph through these two parameters. The larger the spectral gap α the smaller error we get with the same effective sample size. The degree imbalance $\beta \in [0, 1]$ determines how many samples are required for the analysis to hold. We need smaller number of samples if the weighted degrees are balanced, which happens if β is large (close to one).

The following quantity also determines the convexity of the objective function.

$$\gamma \;\equiv\; \min_{j\in[n]} \left\{ \left(1 - \frac{p_{j,\ell_j}}{\kappa_j}\right)^{\lceil 2e^{2b} \rceil - 2} \right\} . \tag{2.7}$$

Note that γ is between zero and one, and a larger value is desired as the objective function becomes more concave and a better accuracy follows. When we are collecting data where the size of the offerings κ_j's are increasing with d but the position of the separators are close to the top, such that $\kappa_j = \omega(d)$ and $p_{j,\ell_j} = O(1)$, then for $b = O(1)$ the above quantity γ can be made arbitrarily close to one, for large enough problem size d. On the other hand, when p_{j,ℓ_j} is close to κ_j, the accuracy can degrade significantly as stronger alternatives might have small chance of showing up in the rank breaking. The value of γ is quite sensitive to b. The reason we have such a inferior dependence on b is because we wanted to give a universal bound on the Hessian that is simple. It is not difficult to get a tighter bound with a larger value of γ, but will inevitably depend on the structure of the data in a complicated fashion.

To ensure that the (second) largest eigenvalue of the Hessian is small enough, we need enough samples. This is captured by η defined as

$$\eta \;\equiv\; \max_{j\in[n]}\{\eta_j\}\;, \qquad \text{where} \qquad \eta_j \;=\; \frac{\kappa_j}{\max\{\ell_j, \kappa_j - p_{j,\ell_j}\}}\;. \tag{2.8}$$

Note that $1 < \eta_j \leq \kappa_j/\ell_j$. A smaller value of η is desired as we require smaller number of samples, as shown in Theorem 2.2. This happens, for instance, when all separators are at the top, such that $p_{j,\ell_j} = \ell_j$ and $\eta_j = \kappa_j/(\kappa_j - \ell_j)$, which is close to one for large κ_j. On the other hand, when all separators are at the bottom of the list, then η can be as large as κ_j.

We discuss the role of the topology of data captures by these parameters in Section 2.4.

2.3 Main Results

We present the main theoretical results accompanied by corresponding numerical simulations in this section.

2.3.1 Upper Bound on the Achievable Error

We present the main result that provides an upper bound on the resulting error and explicitly shows the dependence on the topology of the data. As explained above, we assume that each user provides a partial ranking according to his/her position of the separators. Precisely, we assume the set of offerings S_j, the number of separators ℓ_j, and their respective positions $\mathcal{P}_j = \{p_{j,1}, \dots, p_{j,\ell_j}\}$ are predetermined. Each user draws the ranking of items from the PL model, and provides the partial ranking according to the separators of the form of $\{a > \{b, c, d\} > e > f\}$ in the example in the Figure 2.1.

Theorem 2.2. *Suppose there are n users, d items parametrized by $\theta^* \in \Omega_b$, each user j is presented with a set of offerings $S_j \subseteq [d]$, and provides a partial ordering under the PL model. When the effective sample size $\sum_{j=1}^n \ell_j$ is large enough such that*

$$\sum_{j=1}^n \ell_j \;\; \geq \;\; \frac{2^{11} e^{18b} \eta \log(\ell_{\max}+2)^2}{\alpha^2 \gamma^2 \beta} d \log d \; , \tag{2.9}$$

where $b \equiv \max_i |\theta_i^*|$ is the dynamic range, $\ell_{\max} \equiv \max_{j\in[n]} \ell_j$, α is the (rescaled) spectral gap defined in (2.5)*, β is the (rescaled) maximum degree defined in* (2.6)*, γ and η are defined in Eqs.* (2.7) *and* (2.8)*, then the* rank-breaking estimator *in* (2.3) *with the choice of*

$$\lambda_{j,a} \;\; = \;\; \frac{1}{\kappa_j - p_{j,a}} \; , \tag{2.10}$$

for all $a \in [\ell_j]$ and $j \in [n]$ achieves

$$\frac{1}{\sqrt{d}} \big\| \widehat{\theta} - \theta^* \big\|_2 \;\; \leq \;\; \frac{4\sqrt{2} e^{4b}(1+e^{2b})^2}{\alpha\gamma} \sqrt{\frac{d \log d}{\sum_{j=1}^n \ell_j}} \; , \tag{2.11}$$

with probability at least $1 - 3e^3 d^{-3}$.

Consider an ideal case where the spectral gap is large such that α is a strictly positive constant and the dynamic range b is finite and $\max_{j\in[n]} p_{j,\ell_j}/\kappa_j = C$ for some constant $C < 1$ such that γ is also a constant independent of the problem size d. Then the upper bound in (2.11) implies that we need the effective sample size to scale as $O(d \log d)$, which is only a logarithmic factor larger than the number of parameters to be estimated. Such a logarithmic gap is also unavoidable and due to the fact that we require high probability bounds, where we want the tail probability to decrease at least polynomially in d. We discuss the role of the topology of the data in Section 2.4.

The upper bound follows from an analysis of the convex program similar to those in [156, 84, 182]. However, unlike the traditional data collection scenarios, the main technical challenge is in analyzing the probability that a particular pair of items appear in the rank-breaking. We provide a proof in Section 2.7.1.

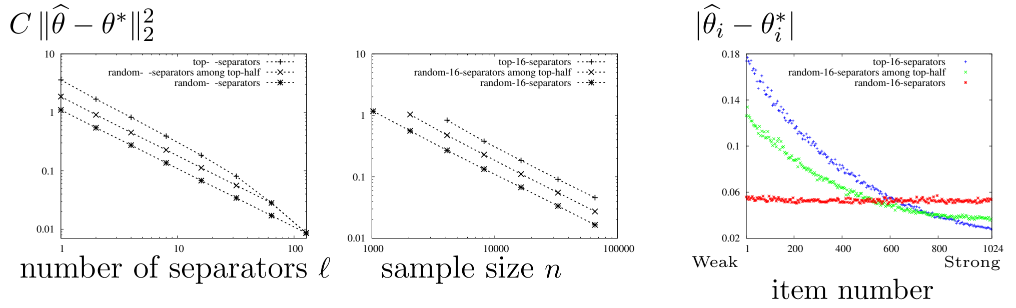


Figure 2.2: Simulation confirms $\|\theta^* - \widehat{\theta}\|_2^2 \propto 1/(\ell\, n)$, and smaller error is achieved for separators that are well spread out.

In Figure 2.2 , we verify the scaling of the resulting error via numerical simulations. We fix $d = 1024$ and $\kappa_j = \kappa = 128$, and vary the number of separators $\ell_j = \ell$ for fixed $n = 128000$ (left), and vary the number of samples n for fixed $\ell_j = \ell = 16$ (middle). Each point is average over 100 instances. The plot confirms that the mean squared error scales as $1/(\ell\, n)$. Each sample is a partial ranking from a set of κ alternatives chosen uniformly at random, where the partial ranking is from a PL model with weights θ^* chosen i.i.d. uniformly over $[-b, b]$ with $b = 2$. To investigate the role of the position of the separators, we compare three scenarios. The *top-ℓ-separators* choose the top ℓ positions for separators, the *random-ℓ-separators among top-half* choose ℓ positions uniformly random from the top half, and the *random-ℓ-separators* choose the positions uniformly at random. We observe that when the positions of the separators are well spread out among the κ offerings, which happens for *random-ℓ-separators*, we get better accuracy.

The figure on the right provides an insight into this trend for $\ell = 16$ and $n = 16000$. The absolute error $|\theta^*_i - \widehat{\theta}_i|$ is roughly same for each item $i \in [d]$ when breaking positions are chosen uniformly at random between 1 to $\kappa - 1$ whereas it is significantly higher for weak preference score items when breaking positions are restricted between 1 to $\kappa/2$ or are top-ℓ. This is due to the fact that the probability of each item being ranked at different positions is different, and in particular probability of the low preference score items being ranked in top-ℓ is very small. The third figure is averaged over 1000 instances. Normalization constant C is n/d^2 and $10^3\ell/d^2$ for the first and second figures respectively. For the first figure n is chosen relatively large such that $n\ell$ is large enough even for $\ell = 1$.

2.3.2 The Price of Rank Breaking for the Special Case of Position-p Ranking

Rank-breaking achieves computational efficiency at the cost of estimation accuracy. In this section, we quantify this tradeoff for a canonical example of position-p ranking, where each sample provides the following information: an unordered set of $p-1$ items that are ranked high, one item that is ranked at the p-th position, and the rest of $\kappa_j - p$ items that are ranked on the bottom. An example of a sample with position-4 ranking six items $\{a, b, c, d, e, f\}$ might be a partial ranking of $(\{a, b, d\} > \{e\} > \{c, f\})$. Since each sample has only one separator for $2 < p$, Theorem 2.2 simplifies to the following Corollary.

Corollary 2.3. *Under the hypotheses of Theorem 2.2, there exist positive constants C and c that only depend on b such that if $n \geq C(\eta d \log d)/(\alpha^2\gamma^2\beta)$ then*

$$\frac{1}{\sqrt{d}}\big\|\widehat{\theta} - \theta^*\big\|_2 \;\; \leq \;\; \frac{c}{\alpha\gamma}\sqrt{\frac{d\,\log d}{n}}\;. \tag{2.12}$$

Note that the error only depends on the position p through γ and η, and is not sensitive. To quantify the price of rank-breaking, we compare this result to a fundamental lower bound on the minimax rate in Theorem 2.4. We can compute a sharp lower bound on the minimax rate, using the Cramér-Rao bound, and a proof is provided in Section 2.7.3.

Theorem 2.4. *Let $\mathcal{U}$ denote the set of all unbiased estimators of θ^* and suppose $b > 0$, then*

$$\inf_{\widehat{\theta}\in\mathcal{U}}\sup_{\theta^*\in\Omega_b} \mathbb{E}[\|\widehat{\theta} - \theta^*\|^2] \;\; \geq \;\; \frac{1}{2p\log(\kappa_{\max})^2}\sum_{i=2}^{d}\frac{1}{\lambda_i(L)} \;\; \geq \;\; \frac{1}{2p\log(\kappa_{\max})^2}\frac{(d-1)^2}{n}\;,$$

where $\kappa_{\max} = \max_{j\in[n]} |S_j|$ and the second inequality follows from the Jensen's inequality.

Note that the second inequality is tight up to a constant factor, when the graph is an expander with a large spectral gap. For expanders, α in the bound (3.23) is also a strictly positive constant. This suggests that rank-breaking gains in computational efficiency by a super-exponential factor of $(p-1)!$, at the price of increased error by a factor of p, ignoring poly-logarithmic factors.

2.3.3 Tighter Analysis for the Special Case of Top-ℓ Separators Scenario

The main result in Theorem 2.2 is general in the sense that it applies to any partial ranking data that is represented by positions of the separators. However, the bound can be quite loose, especially when γ is small, i.e. p_{j,ℓ_j} is close to κ_j. For some special cases, we can tighten the analysis to get a sharper bound. One caveat is that we use a slightly sub-optimal choice of parameters $\lambda_{j,a} = 1/\kappa_j$ instead of $1/(\kappa_j - a)$, to simplify the analysis and still get the order optimal error bound we want. Concretely, we consider a special case of top-ℓ separators scenario, where each agent gives a ranked list of her most preferred ℓ_j alternatives among κ_j offered set of items. Precisely, the locations of the separators are $(p_{j,1}, p_{j,2}, \ldots, p_{j,\ell_j}) = (1, 2, \ldots, \ell_j)$.

Theorem 2.5. *Under the PL model, n partial orderings are sampled over d items parametrized by $\theta^* \in \Omega_b$, where the j-th sample is a ranked list of the top-ℓ_j items among the κ_j items offered to the agent. If*

$$\sum_{j=1}^{n} \ell_j \;\;\geq\;\; \frac{2^{12} e^{6b}}{\beta\alpha^2} d \log d\,, \tag{2.13}$$

where $b \equiv \max_{i,i'} |\theta^*_i - \theta^*_{i'}|$ and α, β are defined in (2.5) *and* (2.6)*, then the* rank-breaking estimator *in* (2.3) *with the choice of $\lambda_{j,a} = 1/\kappa_j$ for all $a \in [\ell_j]$ and $j \in [n]$ achieves*

$$\frac{1}{\sqrt{d}} \big\| \widehat{\theta} - \theta^* \big\|_2 \;\;\leq\;\; \frac{16(1+e^{2b})^2}{\alpha} \sqrt{\frac{d \,\log d}{\sum_{j=1}^{n} \ell_j}}\;, \tag{2.14}$$

with probability at least $1 - 3e^3 d^{-3}$.

A proof is provided in Section 2.7.4. In comparison to the general bound in Theorem 2.2, this is tighter since there is no dependence in γ or η. This gain is significant when, for example, p_{j,ℓ_j} is close to κ_j. As an extreme example, if all agents are offered the entire set of alternatives and are asked to rank all of them, such that $\kappa_j = d$ and $\ell_j = d-1$ for all $j \in [n]$, then the generic bound in (2.11) is loose by a factor of $(e^{4b}/2\sqrt{2})d^{\lceil 2e^{2b}\rceil - 2}$, compared to the above bound.

In the top-ℓ separators scenario, the data set consists of the ranking among top-ℓ_j items of the set S_j, i.e., $[\sigma_j(1), \sigma_j(2), \cdots, \sigma_j(\ell_j)]$. The corresponding log-likelihood of the PL model is

$$\mathcal{L}(\theta) \\ = \sum_{j=1}^{n}\sum_{m=1}^{\ell_j}\Big[\theta_{\sigma_j(m)} - \log\Big(\exp(\theta_{\sigma_j(m)}) + \exp(\theta_{\sigma_j(m+1)}) + \cdots + \exp(\theta_{\sigma_j(\kappa_j)})\Big)\Big]\,, \tag{2.15}$$

where $\sigma_j(a)$ is the alternative ranked at the a-th position by agent j. The Maximum Likelihood Estimator (MLE) for this *traditional* data set is efficient. Hence, there is no computational gain in rank-breaking. Consequently, there is no loss in accuracy either, when we use the optimal weights proposed in the above theorem. Figure 2.3 illustrates that the MLE and the data-driven rank-breaking estimator achieve performance that is identical, and improve over naive rank-breaking that uses uniform weights. We also compare performance of Generalized Method-of-Moments (GMM) proposed by [12] with our algorithm. In addition, we show that performance of GMM can be improved by optimally weighing pairwise comparisons with $\lambda_{j,a}$. MSE of GMM in both the cases, uniform weights and optimal weights, is larger than our rank-breaking estimator. However, GMM is on average about four times faster than our algorithm. We choose $\lambda_{j,a} = 1/(\kappa_j - a)$ in the simulations, as opposed to the $1/\kappa_j$ assumed in the above theorem. This settles the question raised in [84] on whether it is possible to achieve optimal accuracy using rank-breaking under the top-ℓ separators scenario. Analytically, it was proved in [84] that under the top-ℓ separators scenario, naive rank-breaking with uniform weights achieves the same error bound as the MLE, up to a constant factor. However, we show that this constant factor gap is not a weakness of the analyses, but the choice of the weights. Theorem

2.5 provides a guideline for choosing the optimal weights, and the numerical simulation results in Figure 2.3 show that there is in fact no gap in practice, if we use the optimal weights. We use the same settings as that of the first figure of Figure 2.2 for the figure below.

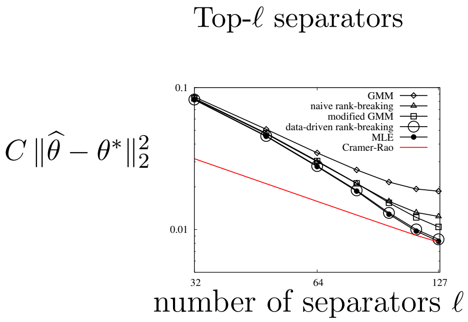


Figure 2.3: The proposed data-driven rank-breaking achieves performance identical to the MLE, and improves over naive rank-breaking with uniform weights.

To prove the order-optimality of the rank-breaking approach up to a constant factor, we can compare the upper bound to a Cramér-Rao lower bound on any unbiased estimators, in the following theorem. A proof is provided in Section 2.7.5.

Theorem 2.6. *Consider ranking $\{\sigma_j(i)\}_{i\in[\ell_j]}$ revealed for the set of items S_j, for $j \in [n]$. Let $\mathcal{U}$ denote the set of all unbiased estimators of $\theta^* \in \Omega_b$. If $b > 0$, then*

$$\begin{aligned}\inf_{\widehat{\theta}\in\mathcal{U}}\sup_{\theta^*\in\Omega_b}\mathbb{E}[\|\widehat{\theta}-\theta^*\|^2] \quad &\geq \quad \left(1-\frac{1}{\ell_{\max}}\sum_{i=1}^{\ell_{\max}}\frac{1}{\kappa_{\max}-i+1}\right)^{-1}\sum_{i=2}^{d}\frac{1}{\lambda_i(L)} \\ &\geq \quad \frac{(d-1)^2}{\sum_{j=1}^{n}\ell_j}\,, \end{aligned} \tag{2.16}$$

where $\ell_{\max} = \max_{j\in[n]}\ell_j$ and $\kappa_{\max} = \max_{j\in[n]}\kappa_j$. The second inequality follows from the Jensen's inequality.

Consider a case when the comparison graph is an expander such that α is a strictly positive constant, and $b = O(1)$ is also finite. Then, the Cramér-Rao lower bound show that the upper bound in (2.14) is optimal up to a logarithmic factor.

2.3.4 Optimality of the Choice of the Weights

We propose the optimal choice of the weights $\lambda_{j,a}$'s in Theorem 2.2. In this section, we show numerical simulations results comparing the proposed approach to other naive choices of the weights under various scenarios. We fix $d = 1024$ items and the underlying preference vector θ^* is uniformly distributed over $[-b, b]$ for $b = 2$. We generate n rankings over sets S_j of size κ for $j \in [n]$ according to the PL model with parameter θ^*. The comparison sets S_j's are chosen independently and uniformly at random from $[d]$.

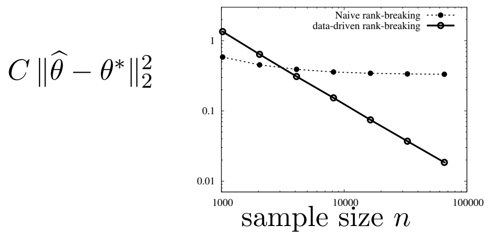


Figure 2.4: Data-driven rank-breaking is consistent, while a random rank-breaking results in inconsistency.

Figure 2.4 illustrates that a naive choice of rank-breakings can result in inconsistency. We create partial orderings data set by fixing $\kappa = 128$ and select $\ell = 8$ random positions in $\{1, \ldots, 127\}$. Each data set consists of partial orderings with separators at those 8 random positions, over 128 randomly chosen subset of items. We vary the sample size n and plot the resulting mean squared error for the two approaches. The data-driven rank-breaking, which uses the optimal choice of the weights, achieves error scaling as $1/n$ as predicted by Theorem 2.2, which implies consistency. For fair comparisons, we feed the same number of pairwise orderings to a naive rank-breaking estimator. This estimator uses randomly chosen pairwise orderings with uniform weights, and is generally inconsistent. However, when sample size is small, inconsistent estimators can achieve smaller variance leading to smaller error. Normalization constant C is $10^3\ell/d^2$, and each point is averaged over 100 trials. We use the minorization-maximization algorithm from [92] for computing the estimates from the rank-breakings.

Even if we use the consistent rank-breakings first proposed in [13], there is ambiguity in the choice of the weights. We next study how much we gain by using the proposed optimal choice of the weights. The optimal choice,

$\lambda_{j,a} = 1/(\kappa_j - p_{j,a})$, depends on two parameters: the size of the offerings κ_j and the position of the separators $p_{j,a}$. To distinguish the effect of these two parameters, we first experiment with fixed $\kappa_j = \kappa$ and illustrate the gain of the optimal choice of $\lambda_{j,a}$'s.

Top-1 and bottom-$(\ell - 1)$ separators

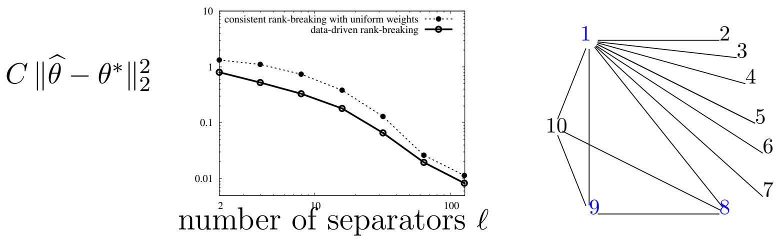


Figure 2.5: There is a constant factor gain of choosing optimal $\lambda_{j,a}$'s when the size of offerings are fixed, i.e. $\kappa_j = \kappa$ (left). We choose a particular set of separators where one separators is at position one and the rest are at the bottom. An example for $\ell = 3$ and $\kappa = 10$ is shown, where the separators are indicated by blue (right).

Figure 2.5 illustrates that the optimal choice of the weights improves over consistent rank-breaking with uniform weights by a constant factor. We fix $\kappa = 128$ and $n = 128000$. As illustrated by a figure on the right, the position of the separators are chosen such that there is one separator at position one, and the rest of $\ell - 1$ separators are at the bottom. Precisely, $(p_{j,1}, p_{j,2}, p_{j,3}, \ldots, p_{j,\ell}) = (1, 128-\ell+1, 128-\ell+2, \ldots, 127)$. We consider this scenario to emphasize the gain of optimal weights. Observe that the MSE does not decrease at a rate of $1/\ell$ in this case. The parameter γ which appears in the bound of Theorem 2.2 is very small when the breaking positions $p_{j,a}$ are of the order κ_j as is the case here, when ℓ is small. Normalization constant C is n/d^2.

The gain of optimal weights is significant when the size of S_j's are highly heterogeneous. Figure 2.6 compares performance of the proposed algorithm, for the optimal choice and uniform choice of weights $\lambda_{j,a}$ when the comparison sets S_j's are of different sizes. We consider the case when n_1 agents provide their top-ℓ_1 choices over the sets of size κ_1, and n_2 agents provide their top-1 choice over the sets of size κ_2. We take $n_1 = 1024$, $\ell_1 = 8$, and $n_2 = 10 n_1 \ell_1$. Figure 2.6 shows MSE for the two choice of weights, when we fix $\kappa_1 = 128$, and vary κ_2 from 2 to 128. As predicted from our bounds, when optimal

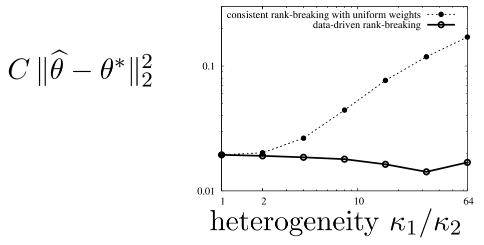


Figure 2.6: The gain of choosing optimal $\lambda_{j,a}$'s is significant when κ_j's are highly heterogeneous.

choice of $\lambda_{j,a}$ is used MSE is not sensitive to sample set sizes κ_2. The error decays at the rate proportional to the inverse of the effective sample size, which is $n_1\ell_1 + n_2\ell_2 = 11n_1\ell_1$. However, with $\lambda_{j,a} = 1$ when $\kappa_2 = 2$, the MSE is roughly 10 times worse. Which reflects that the effective sample size is approximately $n_1\ell_1$, i.e. pairwise comparisons coming from small set size do not contribute without proper normalization. This gap in MSE corroborates bounds of Theorem 7.3. Normalization constant C is $10^3/d^2$.

2.4 The Role of the Topology of the Data

We study the role of topology of the data that provides a guideline for designing the collection of data when we do have some control, as in recommendation systems, designing surveys, and crowdsourcing. The core optimization problem of interest to the designer of such a system is to achieve the best accuracy while minimizing the number of questions.

2.4.1 The Role of the Graph Laplacian

Using the same number of samples, comparison graphs with larger spectral gap achieve better accuracy, compared to those with smaller spectral gaps. To illustrate how graph topology effects the accuracy, we reproduce known spectral properties of canonical graphs, and numerically compare the performance of data-driven rank-breaking for several graph topologies. We follow the examples and experimental setup from [182] for a similar result with pairwise comparisons. Spectral properties of graphs have been a topic of wide interest for decades. We consider a scenario where we fix the size of offerings as $\kappa_j = \kappa = O(1)$ and each agent provides partial ranking with ℓ separators, positions of which are chosen uniformly at random. The resulting spectral gap α of different choices of the set S_j's are provided below. The total number edges in the comparisons graph (counting hyper-edges as multiple edges) is defined as $|E| \equiv \binom{\kappa}{2} n$.

- Complete graph: when $|E|$ is larger than $\binom{d}{2}$, we can design the comparison graph to be a complete graph over d nodes. The weight $A_{ii'}$ on each edge is $n\,\ell/(d(d-1))$, which is the effective number of samples divided by twice the number of edges. Resulting spectral gap is one, which is the maximum possible value. Hence, complete graph is optimal for rank aggregation.

- Sparse random graph: when we have limited resources we might not be able to afford a dense graph. When $|E|$ is of order $o(d^2)$, we have a sparse graph. Consider a scenario where each set S_j is chosen uniformly at random. To ensure connectivity, we need $n = \Omega(\log d)$. Following standard spectral analysis of random graphs, we have $\alpha = \Theta(1)$. Hence, sparse random graphs are near-optimal for rank-aggregation.

- Chain graph: we consider a chain of sets of size κ overlapping only by one item. For example, $S_1 = \{1, \dots, \kappa\}$ and $S_2 = \{\kappa, \kappa+1, \dots, 2\kappa-1\}$, etc. We choose n to be a multiple of $\tau \equiv (d-1)/(\kappa-1)$ and offer each set n/τ times. The resulting graph is a chain of size κ cliques, and standard spectral analysis shows that $\alpha = \Theta(1/d^2)$. Hence, a chain graph is strictly sub-optimal for rank aggregation.

- Star-like graph: We choose one item to be the center, and every offer set consists of this center node and a set of $\kappa - 1$ other nodes chosen uniformly at random without replacement. For example, center node $= \{1\}$, $S_1 = \{1, 2, \dots, \kappa\}$ and $S_2 = \{1, \kappa+1, \kappa+2, \dots, 2\kappa-1\}$, etc. n is chosen in the way similar to that of the Chain graph. Standard spectral analysis shows that $\alpha = \Theta(1)$ and star-like graphs are near-optimal for rank-aggregation.

- Barbell-like graph: We select an offering $S = \{S', i, j\}$, $|S'| = \kappa - 2$ uniformly at random and divide rest of the items into two groups V_1 and V_2. We offer set S $n\kappa/d$ times. For each offering of set S, we offer

$d/\kappa - 1$ sets chosen uniformly at random from the two groups $\{V_1, i\}$ and $\{V_2, j\}$. The resulting graph is a barbell-like graph, and standard spectral analysis shows that $\alpha = \Theta(1/d^2)$. Hence, a chain graph is strictly sub-optimal for rank aggregation.

Figure 2.7 illustrates how graph topology effects the accuracy. When θ^* is chosen uniformly at random, the accuracy does not change with d (left), and the accuracy is better for those graphs with larger spectral gap. However, for a certain worst-case θ^*, the error increases with d for the chain graph and the barbell-like graph, as predicted by the above analysis of the spectral gap. We use $\ell = 4$, $\kappa = 17$ and vary d from 129 to 2049. κ is kept small to make the resulting graphs more like the above discussed graphs. Figure on left shows accuracy when θ^* is chosen i.i.d. uniformly over $[-b, b]$ with $b = 2$. Error in this case is roughly same for each of the graph topologies with chain graph being the worst. However, when θ^* is chosen carefully error for chain graph and barbell-like graph increases with d as shown in the figure right. We chose θ^* such that all the items of a set have same weight, either $\theta_i = 0$ or $\theta_i = b$ for chain graph and barbell-like graph. We divide all the sets equally between the two types for chain graph. For barbell-like graph, we keep the two types of sets on the two different sides of the connector set and equally divide items of the connector set into two types. Number of samples n is $100(d-1)/(\kappa-1)$ and each point is averaged over 100 instances. Normalization constant C is $n\ell/d^2$.

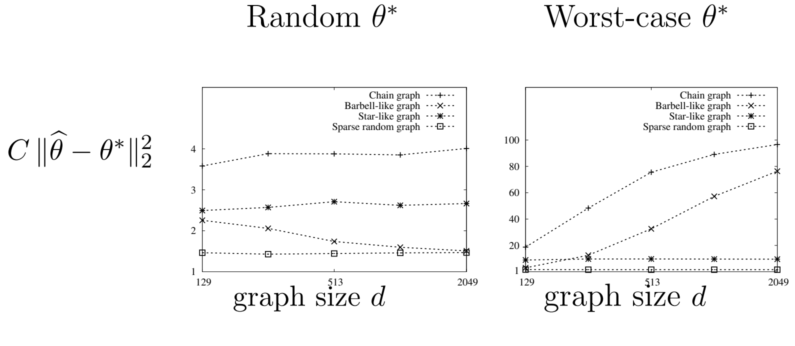


Figure 2.7: For randomly chosen θ^* the error does not change with d (left). However, for particular worst-case θ^* the error increases with d for the Chain graph and the Barbell-like graph as predicted by the analysis of the spectral gap (right).

2.4.2 The Role of the Position of the Separators

As predicted by theorem 2.2, rank-breaking fails when γ is small, i.e. the position of the separators are very close to the bottom. An extreme example is the bottom-ℓ separators scenario, where each person is offered κ randomly chosen alternatives, and is asked to give a ranked list of bottom ℓ alternatives. In other words, the ℓ separators are placed at $(p_{j,1},\ldots,p_{j,\ell}) = (\kappa_j-\ell,\ldots,\kappa-1)$. In this case, $\gamma \simeq 0$ and the error bound is large. This is not a weakness of the analysis. In fact we observe large errors under this scenario. The reason is that many alternatives that have large weights θ_i's will rarely be even compared once, making any reasonable estimation infeasible.

Figure 2.8 illustrates this scenario. We choose $\ell = 8$, $\kappa = 128$, and $d = 1024$. The other settings are same as that of the first figure of Figure 2.2. The left figure plots the magnitude of the estimation error for each item. For about 200 strong items among 1024, we do not even get a single comparison, hence we omit any estimation error. It clearly shows the trend: we get good estimates for about 400 items in the bottom, and we get large errors for the rest. Consequently, even if we only take those items that have at least one comparison into account, we still get large errors. This is shown in the figure right. The error barely decays with the sample size. However, if we focus on the error for the bottom 400 items, we get good error rate decaying inversely with the sample size. Normalization constant C in the second figure is $10^2\, x\, d/\ell$ and $10^2(400)d/\ell$ for the first and second lines respectively, where x is the number of items that appeared in rank-breaking at least once. We solve convex program (2.3) for θ restricted to the items that appear in rank-breaking at least once. The second figure of Figure 2.8 is averaged over 1000 instances.

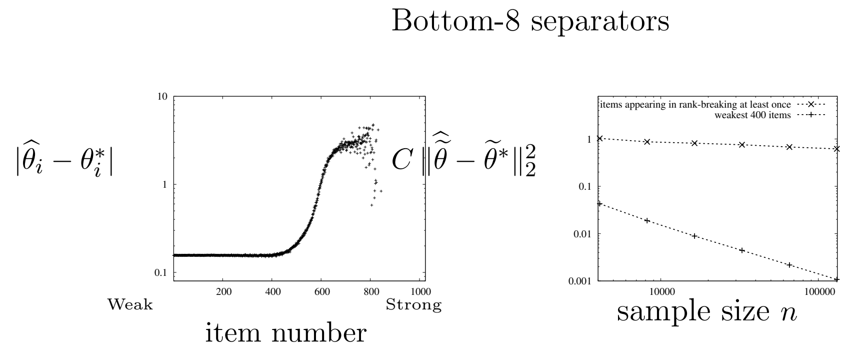

Bottom-8 separators

Figure 2.8: Under the bottom-ℓ separators scenario, accuracy is good only for the bottom 400 items (left). As predicted by Theorem 2.7, the mean squared error on the bottom 400 items scale as $1/n$, where as the overall mean squared error does not decay (right).

We make this observation precise in the following theorem. Applying rank-breaking to only to those weakest $\widetilde{d}$ items, we prove an upper bound on the achieved error rate that depends on the choice of the $\widetilde{d}$. Without loss of generality, we suppose the items are sorted such that $\theta^*_1 \leq \theta^*_2 \leq \cdots \leq \theta^*_d$. For a choice of $\widetilde{d} = \ell d/(2\kappa)$, we denote the weakest $\widetilde{d}$ items by $\widetilde{\theta}^* \in \mathbb{R}^{\widetilde{d}}$ such that $\widetilde{\theta}^*_i = \theta^*_i - (1/\widetilde{d})\sum_{i'=1}^{\widetilde{d}} \theta^*_{i'}$, for $i \in [\widetilde{d}]$. Since $\theta^* \in \Omega_b$, $\widetilde{\theta}^* \in [-2b, 2b]^{\widetilde{d}}$. The space of all possible preference vectors for $[\widetilde{d}]$ items is given by $\widetilde{\Omega} = \{\widetilde{\theta} \in \mathbb{R}^{\widetilde{d}} : \sum_{i=1}^{\widetilde{d}} \widetilde{\theta}_i = 0\}$ and $\widetilde{\Omega}_{2b} = \widetilde{\Omega} \cap [-2b, 2b]^{\widetilde{d}}$.

Although the analysis can be easily generalized, to simplify notations, we fix $\kappa_j = \kappa$ and $\ell_j = \ell$ and assume that the comparison sets S_j, $|S_j| = \kappa$, are chosen uniformly at random from the set of d items for all $j \in [n]$. The rank-breaking log likelihood function $\mathcal{L}_{\rm RB}(\widetilde{\theta})$ for the set of items $[\widetilde{d}]$ is given by

$$\mathcal{L}_{\rm RB}(\widetilde{\theta}) \;=\; \sum_{j=1}^{n}\sum_{a=1}^{\ell_j} \lambda_{j,a} \Big\{ \sum_{(i,i') \in E_{j,a}} \mathbb{I}_{\big\{i,i' \in [\widetilde{d}]\big\}} \Big(\theta_{i'} - \log\Big(e^{\theta_i} + e^{\theta_{i'}}\Big)\Big) \Big\} \tag{2.17}$$

We analyze the rank-breaking estimator

$$\widehat{\widetilde{\theta}} \;\equiv\; \max_{\widetilde{\theta} \in \widetilde{\Omega}_{2b}} \mathcal{L}_{\rm RB}(\widetilde{\theta}) \,. \tag{2.18}$$

We further simplify notations by fixing $\lambda_{j,a} = 1$, since from Equation (2.24), we know that the error increases by at most a factor of 4 due to this sub-optimal choice of the weights, under the special scenario studied in this theorem.

Theorem 2.7. *Under the bottom-ℓ separators scenario and the PL model, S_j's are chosen uniformly at random of size κ and n partial orderings are sampled over d items parametrized by $\theta^* \in \Omega_b$. For $\widetilde{d} = \ell d/(2\kappa)$ and any $\ell \geq 4$, if the effective sample size is large enough such that*

$$n\ell \;\geq\; \left(\frac{2^{14}e^{8b}}{\chi^2}\frac{\kappa^3}{\ell^3}\right) d\log d\;, \tag{2.19}$$

where

$$\chi \;\equiv\; \frac{1}{4}\left(1-\exp\left(-\frac{2}{9(\kappa-2)}\right)\right), \tag{2.20}$$

then the rank-breaking *estimator in* (2.18) *achieves*

$$\frac{1}{\sqrt{\widetilde{d}}}\big\|\widehat{\widetilde{\theta}}-\widetilde{\theta}^*\big\|_2 \;\leq\; \frac{128(1+e^{4b})^2}{\chi}\frac{\kappa^{3/2}}{\ell^{3/2}}\sqrt{\frac{d\log d}{n\ell}}\;, \tag{2.21}$$

with probability at least $1-3e^3d^{-3}$.

Consider a scenario where $\kappa = O(1)$ and $\ell = \Theta(\kappa)$. Then, χ is a strictly positive constant, and also κ/ℓ is s finite constant. It follows that rank-breaking requires the effective sample size $n\ell = O(d\log d/\varepsilon^2)$ in order to achieve arbitrarily small error of $\varepsilon > 0$, on the weakest $\widetilde{d} = \ell\, d/(2\kappa)$ items.

2.5 Real-World Data Sets

On real-world data sets on sushi preferences [104], we show that the data-driven rank-breaking improves over Generalized Method-of-Moments (GMM) proposed by [12]. This is a widely used data set for rank aggregation, for instance in [12, 14, 147, 124, 137, 136]. The data set consists of complete rankings over 10 types of sushi from $n = 5000$ individuals. Below, we follow the experimental scenarios of the GMM approach in [12] for fair comparisons.

To validate our approach, we first take the estimated PL weights of the 10 types of sushi, using [92] implementation of the ML estimator, over the entire input data of 5000 complete rankings. We take thus created output as the ground truth θ^*. To create partial rankings and compare the performance of the data-driven rank-breaking to the state-of-the-art GMM approach in Figure 2.9, we first fix $\ell = 6$ and vary n to simulate top-ℓ-separators scenario by removing the known ordering among bottom $10 - \ell$ alternatives for each sample in the data set (left). We next fix $n = 1000$ and vary ℓ and simulate top-ℓ-separators scenarios (right). Each point is averaged over 1000 instances. The mean squared error is plotted for both algorithms.

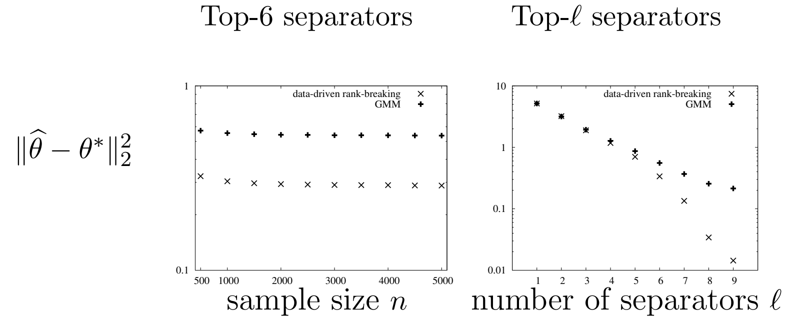


Figure 2.9: The data-driven rank-breaking achieves smaller error compared to the state-of-the-art GMM approach.

Figure 2.10 illustrates the Kendall rank correlation of the rankings estimated by the two algorithms and the ground truth. Larger value indicates that the estimate is closer to the ground truth, and the data-driven rank-breaking outperforms the state-of-the-art GMM approach.

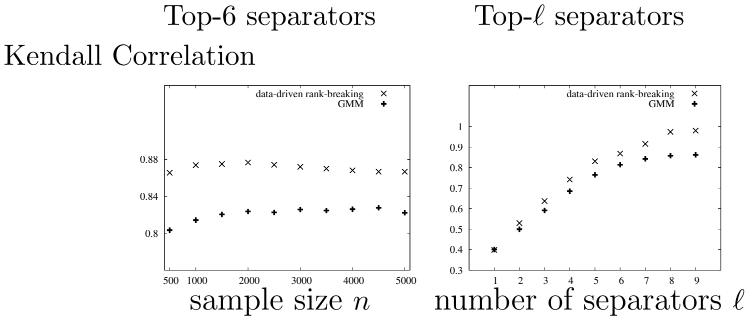


Figure 2.10: The data-driven rank-breaking achieves larger Kendall rank correlation compared to the state-of-the-art GMM approach.

To validate whether PL model is the right model to explain the sushi data set, we compare the data-driven rank-breaking, MLE for the PL model, GMM for the PL model, Borda count and Spearman's footrule optimal aggregation. We measure the Kendall rank correlation between the estimates and the samples and show the result in Table 2.1. In particular, if

$\sigma_1, \sigma_2, \cdots, \sigma_n$ denote sample rankings and $\widehat{\sigma}$ denote the aggregated ranking then the correlation value is $(1/n)\sum_{i=1}^{n}\big(1 - \frac{4\mathcal{K}(\widehat{\sigma},\sigma_i)}{\kappa(\kappa-1)}\big)$, where $\mathcal{K}(\sigma_1, \sigma_2) = \sum_{i<j\in[\kappa]} \mathbb{I}_{\{(\sigma_1^{-1}(i)-\sigma_1^{-1}(j))(\sigma_2^{-1}(i)-\sigma_2^{-1}(j))<0\}}$. The results are reported for different number of samples n and different values of ℓ under the top-ℓ separators scenarios. When $\ell = 9$, we are using all the complete rankings, and all algorithms are efficient. When $\ell < 9$, we have partial orderings, and Spearman's footrule optimal aggregation is NP-hard. We instead use scaled footrule aggregation (SFO) given in [58]. Most approaches achieve similar performance, except for the Spearman's footrule. The proposed data-driven rank-breaking achieves a slightly worse correlation compared to other approaches. However, note that none of the algorithms are necessarily maximizing the Kendall correlation, and are not expected to be particularly good in this metric.

	MLE under PL	data-driven RB	GMM	Borda count	Spear-man's footrule
$n = 500$, $\ell = 9$	0.306	0.291	0.315	0.315	0.159
$n = 5000$, $\ell = 9$	0.309	0.309	0.315	0.315	0.079
$n = 5000$, $\ell = 2$	0.199	0.199	0.201	0.200	0.113
$n = 5000$, $\ell = 5$	0.217	0.200	0.217	0.295	0.152

Table 2.1: Kendall rank correlation on sushi data set.

We compare our algorithm with the GMM algorithm on two other real-world data-sets as well. We use jester data set [78] that consists of over 4.1 million continuous ratings between -10 to $+10$ of 100 jokes from 48,483 users. The average number of jokes rated by an user is 72.6 with minimum and maximum being 36 and 100 respectively. We convert continuous ratings into ordinal rankings. This data-set has been used by [154, 166, 40, 125] for rank aggregation and collaborative filtering.

Similar to the settings of sushi data experiments, we take the estimated PL weights of the 100 jokes over all the rankings as ground truth. Figure 2.11 shows comparative performance of the data-driven rank-breaking and the GMM for the two scenarios. We first fix $\ell = 10$ and vary n to simulate random-10 separators scenario (left). We next take all the rankings $n = 73421$ and vary ℓ to simulate random-ℓ separators scenario (rights). Since sets have different sizes, while varying ℓ we use full breaking if the setsize is smaller than ℓ. Each point is averaged over 100 instances. The mean squared error is plotted for both algorithms.

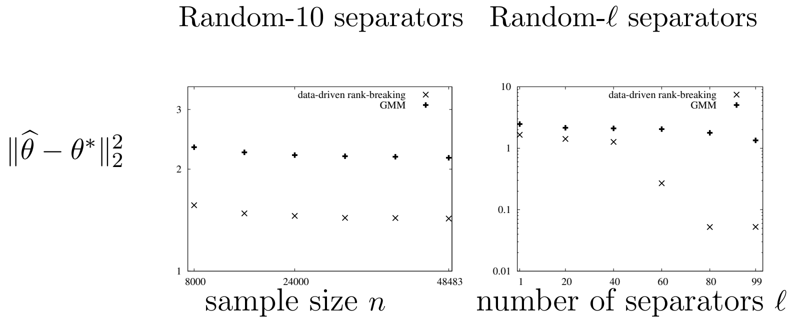


Figure 2.11: jester data set: The data-driven rank-breaking achieves smaller error compared to the state-of-the-art GMM approach.

We perform similar experiments on American Psychological Association (APA) data-set [54]. The APA elects a president each year by asking each member to rank order a slate of five candidates. The data-set represents full rankings given by 5738 members of the association in 1980's election. The mean squared error is plotted for both algorithms under the settings similar to that of jester data-set.

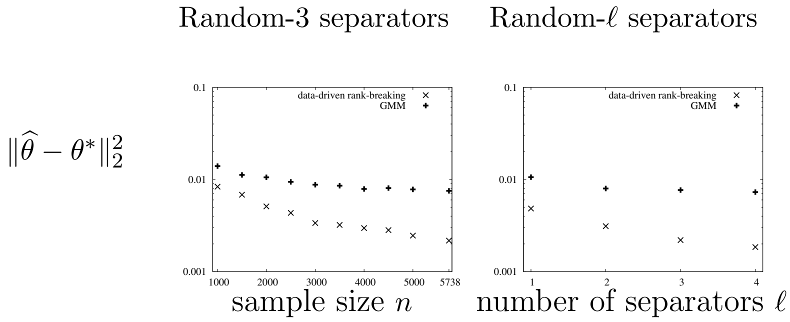


Figure 2.12: APA data set: The data-driven rank-breaking achieves smaller error compared to the state-of-the-art GMM approach.

2.6 Discussion

We study the problem of learning the PL model from ordinal data. Under the traditional data collection scenarios, several efficient algorithms find the maximum likelihood estimates and at the same time provably achieve minimax optimal performance. However, for some non-traditional scenarios, computational complexity of finding the maximum likelihood estimate can scale super-exponentially in the problem size. We provide the first finite-sample analysis of computationally efficient estimators known as rank-breaking estimators. This provides guidelines for choosing the weights in the estimator to achieve optimal performance, and also explicitly shows how the accuracy depends on the topology of the data.

This paper provides the first analytical result in the sample complexity of rank-breaking estimators, and quantifies the price we pay in accuracy for the computational gain. In general, more complex higher-order rank-breaking can also be considered, where instead of breaking a partial ordering into a collection of paired comparisons, we break it into a collection of higher-order comparisons. The resulting higher-order rank-breakings will enable us to traverse the whole spectrum of computational complexity between the pairwise rank-breaking and the MLE. We believe this paper opens an interesting new direction towards understanding the whole spectrum of such approaches. However, analyzing the Hessian of the corresponding objective function is significantly more involved and requires new technical innovations.

2.7 Proofs

2.7.1 Proof of Theorem 2.2

We prove a more general result for an arbitrary choice of the parameter $\lambda_{j,a} > 0$ for all $j \in [n]$ and $a \in [\ell_j]$. The following theorem proves the (near)-optimality of the choice of $\lambda_{j,a}$'s proposed in (6.27), and implies the corresponding error bound as a corollary.

Theorem 2.8. *Under the hypotheses of Theorem 2.2 and any $\lambda_{j,a}$'s, the* *rank-breaking estimator achieves*

$$\frac{1}{\sqrt{d}}\big\|\,\widehat{\theta}-\theta^*\,\big\|_2 \;\leq\; \frac{4\sqrt{2}e^{4b}(1+e^{2b})^2\sqrt{d\log d}}{\alpha\,\gamma}\frac{\sqrt{\sum_{j=1}^{n}\sum_{a=1}^{\ell_j}\big(\lambda_{j,a}\big)^2\big(\kappa_j-p_{j,a}\big)\big(\kappa_j-p_{j,a}+1\big)}}{\sum_{j=1}^{n}\sum_{a=1}^{\ell_j}\lambda_{j,a}(\kappa_j-p_{j,a})}\,, \tag{2.22}$$

with probability at least $1-3e^3d^{-3}$*, if*

$$\sum_{j=1}^{n}\sum_{a=1}^{\ell_j}\lambda_{j,a}(\kappa_j-p_{j,a}) \;\geq\; 2^6e^{18b}\frac{\eta\delta}{\alpha^2\beta\gamma^2\tau}d\log d\,, \tag{2.23}$$

where γ*,* η*,* τ*,* δ*,* α*,* β*, are now functions of* $\lambda_{j,a}$*'s and defined in* (2.7)*,* (2.8)*,* (2.25)*,* (2.27) *and* (2.30)*.*

We first claim that $\lambda_{j,a}=1/(\kappa_j-p_{j,a}+1)$ is the optimal choice for minimizing the above upper bound on the error. From Cauchy-Schwartz inequality and the fact that all terms are non-negative, we have that

$$\frac{\sqrt{\sum_{j=1}^{n}\sum_{a=1}^{\ell_j}\big(\lambda_{j,a}\big)^2(\kappa_j-p_{j,a})(\kappa_j-p_{j,a}+1)}}{\sum_{j=1}^{n}\sum_{a=1}^{\ell_j}\lambda_{j,a}(\kappa_j-p_{j,a})} \;\geq\; \frac{1}{\sqrt{\sum_{j=1}^{n}\sum_{a=1}^{\ell_j}\frac{(\kappa_j-p_{j,a})}{(\kappa_j-p_{j,a}+1)}}}\,, \tag{2.24}$$

where $\lambda_{j,a}=1/(\kappa_j-p_{j,a}+1)$ achieves the universal lower bound on the right-hand side with an equality. Since $\sum_{j=1}^{n}\sum_{a=1}^{\ell_j}\frac{(\kappa_j-p_{j,a})}{(\kappa_j-p_{j,a}+1)}\geq\sum_{j=1}^{n}\ell_j$, substituting this into (7.24) gives the desired error bound in (2.11). Although we have identified the optimal choice of $\lambda_{j,a}$'s, we choose a slightly different value of $\lambda=1/(\kappa_j-p_{j,a})$ for the analysis. This achieves the same desired error bound in (2.11), and significantly simplifies the notations of the sufficient conditions.

We first define all the parameters in the above theorem for general $\lambda_{j,a}$. With a slight abuse of notations, we use the same notations for $\mathcal{H}$, L, α and β for both the general $\lambda_{j,a}$'s and also the specific choice of $\lambda_{j,a}=1/(\kappa_j-p_{j,a})$.

It should be clear from the context what we mean in each case. Define

$$\tau \;\equiv\; \min_{j\in[n]} \tau_j \;, \qquad \text{where } \;\tau_j \equiv \frac{\sum_{a=1}^{\ell_j} \lambda_{j,a}(\kappa_j - p_{j,a})}{\ell_j} \tag{2.25}$$

$$\delta_{j,1} \;\equiv\; \left\{ \max_{a\in[\ell_j]} \Big\{ \lambda_{j,a}(\kappa_j - p_{j,a}) \Big\} + \sum_{a=1}^{\ell_j} \lambda_{j,a} \right\} \;, \text{ and } \qquad \delta_{j,2} \equiv \sum_{a=1}^{\ell_j} \lambda_{j,a} \tag{2.26}$$

$$\delta \;\equiv\; \max_{j\in[n]} \left\{ 4\delta_{j,1}^2 + \frac{2\big(\delta_{j,1}\delta_{j,2} + \delta_{j,2}^2\big)\kappa_j}{\eta_j \ell_j} \right\} \;. \tag{2.27}$$

Note that $\delta \geq \delta_{j,1}^2 \geq \max_a \lambda_{j,a}^2(\kappa_j - p_{j,a})^2 \geq \tau^2$, and for the choice of $\lambda_{j,a} = 1/(\kappa_j - p_{j,a})$ it simplifies as $\tau = \tau_j = 1$. We next define a comparison graph $\mathcal{H}$ for general $\lambda_{j,a}$, which recovers the proposed comparison graph for the optimal choice of $\lambda_{j,a}$'s

Definition 2.9. *(Comparison graph $\mathcal{H}$). Each item $i \in [d]$ corresponds to a vertex i. For any pair of vertices i, i', there is a weighted edge between them if there exists a set S_j such that $i, i' \in S_j$; the weight equals $\sum_{j:i,i'\in S_j} \frac{\tau_j \ell_j}{\kappa_j(\kappa_j - 1)}$.*

Let A denote the weighted adjacency matrix, and let $D = \mathrm{diag}(A\mathbf{1})$. Define,

$$D_{\max} \;\equiv\; \max_{i\in[d]} D_{ii} \;=\; \max_{i\in[d]} \left\{ \sum_{j:i\in S_j} \frac{\tau_j \ell_j}{\kappa_j} \right\} \;\geq\; \tau_{\min} \max_{i\in[d]} \left\{ \sum_{j:i\in S_j} \frac{\ell_j}{\kappa_j} \right\}. \tag{2.28}$$

Define graph Laplacian L as $L = D - A$, i.e.,

$$L \;=\; \sum_{j=1}^{n} \frac{\tau_j \ell_j}{\kappa_j(\kappa_j - 1)} \sum_{i<i'\in S_j} (e_i - e_{i'})(e_i - e_{i'})^\top. \tag{2.29}$$

Let $0 = \lambda_1(L) \leq \lambda_2(L) \leq \cdots \leq \lambda_d(L)$ denote the sorted eigenvalues of L. Note that $\mathrm{Tr}(L) = \sum_{i=1}^{d} \sum_{j:i\in S_j} \tau_j \ell_j/\kappa_j = \sum_{j=1}^{n} \tau_j \ell_j$. Define α and β such that

$$\alpha \equiv \frac{\lambda_2(L)(d-1)}{\mathrm{Tr}(L)} = \frac{\lambda_2(L)(d-1)}{\sum_{j=1}^{n} \tau_j \ell_j} \;\text{ and }\; \beta \equiv \frac{\mathrm{Tr}(L)}{dD_{\max}} = \frac{\sum_{j=1}^{n} \tau_j \ell_j}{dD_{\max}} \;. \tag{2.30}$$

For the proposed choice of $\lambda_{j,a} = 1/(\kappa_j - p_{j,a})$, we have $\tau_j = 1$ and the definitions of $\mathcal{H}$, L, α, and β reduce to those defined in Definition 2.1. We are left to prove an upper bound, $\delta \leq 32(\log(\ell_{\max}+2))^2$, which implies the sufficient condition in (2.9) and finishes the proof of Theorem 2.2. We have,

$$\begin{aligned}
\delta_{j,1} = \max_{a\in[\ell_j]}\Big\{\lambda_{j,a}(\kappa_j - p_{j,a})\Big\} + \sum_{a=1}^{\ell_j}\lambda_{j,a} &= 1 + \sum_{a=1}^{\ell_j}\frac{1}{\kappa_j - p_{j,a}} \\
&\leq 1 + \sum_{a=1}^{\ell_j}\frac{1}{a} \\
&\leq 2\log(\ell_j+2)\,, \qquad (2.31)
\end{aligned}$$

where in the first inequality follows from taking the worst case for the positions, i.e. $p_{j,a} = \kappa_j - \ell_j + a - 1$ Using the fact that for any integer x, $\sum_{a=0}^{\ell-1} 1/(x+a) \leq \log((x+\ell-1)/(x-1))$, we also have

$$\begin{aligned}
\frac{\delta_{j,2}\kappa_j}{\eta_j\ell_j} &\leq \sum_{a=1}^{\ell_j}\frac{1}{\kappa_j - p_{j,a}}\frac{\max\{\ell_j, \kappa_j - p_{j,\ell_j}\}}{\ell_j} \\
&\leq \min\Big\{\log(\ell_j+2)\,,\, \log\Big(\frac{\kappa_j - p_{j,\ell_j} + \ell_j - 1}{\kappa_j - p_{j,\ell_j} - 1}\Big)\Big\}\frac{\max\{\ell_j, \kappa_j - p_{j,\ell_j}\}}{\ell_j} \\
&\leq \frac{\log(\ell_j+2)\ell_j}{\max\{\ell_j, \kappa_j - p_{j,\ell_j} - 1\}}\frac{\max\{\ell_j, \kappa_j - p_{j,\ell_j}\}}{\ell_j} \\
&\leq 2\log(\ell_j+2)\,, \qquad (2.32)
\end{aligned}$$

where the first inequality follows from the definition of η_j, Equation (2.8). From (2.31), (2.32), and the fact that $\delta_{j,2} \leq \log(\ell_j+2)$, we have

$$\delta = \max_{j\in[n]}\left\{4\delta_{j,1}^2 + \frac{2\big(\delta_{j,1}\delta_{j,2} + \delta_{j,2}^2\big)\kappa_j}{\eta_j\ell_j}\right\} \;\leq\; 28(\log(\ell_{\max}+2))^2\,.$$

2.7.2 Proof of Theorem 7.3

We first introduce two key technical lemmas. In the following lemma we show that $\mathbb{E}_{\theta^*}[\nabla\mathcal{L}_{\rm RB}(\theta^*)] = 0$ and provide a bound on the deviation of $\nabla\mathcal{L}_{\rm RB}(\theta^*)$ from its mean. The expectation $\mathbb{E}_{\theta^*}[\cdot]$ is with respect to the randomness in the samples drawn according to θ^*. The log likelihood Equation (2.2) can be rewritten as

$$\mathcal{L}_{\mathrm{RB}}(\theta) = \sum_{j=1}^{n}\sum_{a=1}^{\ell_j}\sum_{i<i'\in S_j} \mathbb{I}_{\left\{(i,i')\in G_{j,a}\right\}}\lambda_{j,a}\Big(\theta_i \mathbb{I}_{\left\{\sigma_j^{-1}(i)<\sigma_j^{-1}(i')\right\}} + \theta_{i'}\mathbb{I}_{\left\{\sigma_j^{-1}(i)>\sigma_j^{-1}(i')\right\}} - \log\left(e^{\theta_i}+e^{\theta_{i'}}\right)\Big)\,. \qquad (2.33)$$

We use $(i,i')\in G_{j,a}$ to mean either (i,i') or (i',i) belong to $E_{j,a}$. Taking the first-order partial derivative of $\mathcal{L}_{\mathrm{RB}}(\theta)$, we get

$$\nabla_i\mathcal{L}_{\mathrm{RB}}(\theta^*) = \sum_{j:i\in S_j}\sum_{a=1}^{\ell_j}\sum_{\substack{i'\in S_j\\ i'\neq i}} \lambda_{j,a}\,\mathbb{I}_{\left\{(i,i')\in G_{j,a}\right\}}\left(\mathbb{I}_{\left\{\sigma_j^{-1}(i)<\sigma_j^{-1}(i')\right\}} - \frac{\exp(\theta_i^*)}{\exp(\theta_i^*)+\exp(\theta_{i'}^*)}\right)\,. \qquad (2.34)$$

Lemma 2.10. *Under the hypotheses of Theorem 2.2, with probability at least* $1-2e^3d^{-3}$,

$$\left\|\nabla\mathcal{L}_{\mathrm{RB}}(\theta^*)\right\|_2 \;\leq\; \sqrt{6\log d\sum_{j=1}^{n}\sum_{a=1}^{\ell_j}\big(\lambda_{j,a}\big)^2\big(\kappa_j-p_{j,a}\big)\big(\kappa_j-p_{j,a}+1\big)}\,.$$

The Hessian matrix $H(\theta)\in\mathcal{S}^d$ with $H_{ii'}(\theta)=\frac{\partial^2\mathcal{L}_{\mathrm{RB}}(\theta)}{\partial\theta_i\partial\theta_{i'}}$ is given by

$$H(\theta) = -\sum_{j=1}^{n}\sum_{a=1}^{\ell_j}\sum_{i<i'\in S_j}\mathbb{I}_{\left\{(i,i')\in G_{j,a}\right\}}\lambda_{j,a}\left((e_i-e_{i'})(e_i-e_{i'})^\top\frac{\exp(\theta_i+\theta_{i'})}{[\exp(\theta_i)+\exp(\theta_{i'})]^2}\right). \qquad (2.35)$$

It follows from the definition that $-H(\theta)$ is positive semi-definite for any $\theta\in\mathbb{R}^d$. The smallest eigenvalue of $-H(\theta)$ is equal to zero and the corresponding eigenvector is all-ones vector. The following lemma lower bounds its second smallest eigenvalue $\lambda_2(-H(\theta))$.

Lemma 2.11. *Under the hypotheses of Theorem 2.2, if*

$$\sum_{j=1}^{n}\sum_{a=1}^{\ell_j}\lambda_{j,a}(\kappa_j - p_{j,a}) \geq 2^6 e^{18b}\frac{\eta\delta}{\alpha^2\beta\gamma^2\tau} d\log d \tag{2.36}$$

then with probability at least $1-d^{-3}$*, the following holds for any* $\theta \in \Omega_b$*:*

$$\lambda_2(-H(\theta)) \;\geq\; \frac{e^{-4b}}{(1+e^{2b})^2}\frac{\alpha\gamma}{d-1}\sum_{j=1}^{n}\sum_{a=1}^{\ell_j}\lambda_{j,a}(\kappa_j - p_{j,a})\,. \tag{2.37}$$

Define $\Delta = \widehat{\theta} - \theta^*$. It follows from the definition that Δ is orthogonal to the all-ones vector. By the definition of $\hat{\theta}$ as the optimal solution of the optimization (2.3), we know that $\mathcal{L}_{\text{RB}}(\widehat{\theta}) \geq \mathcal{L}_{\text{RB}}(\theta^*)$ and thus

$$\begin{aligned}\mathcal{L}_{\text{RB}}(\widehat{\theta}) - \mathcal{L}_{\text{RB}}(\theta^*) - \langle \nabla\mathcal{L}_{\text{RB}}(\theta^*), \Delta\rangle &\geq -\langle \nabla\mathcal{L}_{\text{RB}}(\theta^*), \Delta\rangle \\ &\geq -\|\nabla\mathcal{L}_{\text{RB}}(\theta^*)\|_2\|\Delta\|_2, \quad (2.38)\end{aligned}$$

where the last inequality follows from the Cauchy-Schwartz inequality. By the mean value theorem, there exists a $\theta = a\widehat{\theta} + (1-a)\theta^*$ for some $a \in [0,1]$ such that $\theta \in \Omega_b$ and

$$\begin{aligned}\mathcal{L}_{\text{RB}}(\widehat{\theta}) - \mathcal{L}_{\text{RB}}(\theta^*) - \langle \nabla\mathcal{L}_{\text{RB}}(\theta^*), \Delta\rangle &= \frac{1}{2}\Delta^\top H(\theta)\Delta \\ &\leq -\frac{1}{2}\lambda_2(-H(\theta))\|\Delta\|_2^2, \quad (2.39)\end{aligned}$$

where the last inequality holds because the Hessian matrix $-H(\theta)$ is positive semi-definite with $H(\theta)\mathbf{1} = \mathbf{0}$ and $\Delta^\top\mathbf{1} = 0$. Combining (2.38) and (2.39),

$$\|\Delta\|_2 \;\leq\; \frac{2\|\nabla\mathcal{L}_{\text{RB}}(\theta^*)\|_2}{\lambda_2(-H(\theta))}. \tag{2.40}$$

Note that $\theta \in \Omega_b$ by definition. Theorem 7.3 follows by combining Equation (2.40) with Lemma 2.10 and Lemma 2.11.

Proof of Lemma 2.10

The idea of the proof is to view $\nabla\mathcal{L}_{\text{RB}}(\theta^*)$ as the final value of a discrete time vector-valued martingale with values in $\mathbb{R}^d$. Define $\nabla\mathcal{L}_{G_{j,a}}(\theta^*)$ as the gradient vector arising out of each rank-breaking graph $\{G_{j,a}\}_{j\in[n],a\in[\ell_j]}$ that is

$$\nabla_i \mathcal{L}_{G_{j,a}}(\theta^*) \equiv \sum_{\substack{i'\in S_j\\ i'\neq i}} \lambda_{j,a}\, \mathbb{I}_{\left\{(i,i')\in G_{j,a}\right\}} \left(\mathbb{I}_{\left\{\sigma_j^{-1}(i)<\sigma_j^{-1}(i')\right\}} - \frac{\exp(\theta_i^*)}{\exp(\theta_i^*)+\exp(\theta_{i'}^*)} \right) . \tag{2.41}$$

Consider $\nabla \mathcal{L}_{G_{j,a}}(\theta^*)$ as the incremental random vector in a martingale of $\sum_{j=1} \ell_j$ time steps. Lemma 2.12 shows that the expectation of each incremental vector is zero. Observe that the conditioning event $\{i'' \in S : \sigma^{-1}(i'') < p_{j,a}\}$ given in (2.43) is equivalent to conditioning on the history $\{G_{j,a'}\}_{a'<a}$. Therefore, using the assumption that the rankings $\{\sigma_j\}_{j\in[n]}$ are mutually independent, we have that the conditional expectation of $\nabla \mathcal{L}_{G_{j,a}}(\theta^*)$ conditioned on $\{G_{j',a''}\}_{j'<j,a''\in[\ell_{j'}]}$ is zero. Further, the conditional expectation of $\nabla \mathcal{L}_{G_{j,a}}(\theta^*)$ is zero even when conditioned on the rank breaking due to previous separators $\{G_{j,a'}\}_{a'<a}$ that are ranked higher (i.e. $a' < a$), which follows from the next lemma.

Lemma 2.12. *For a position-p rank breaking graph G_p, defined over a set of items S, where $p \in [|S|-1]$,*

$$\mathbb{P}\Big[\sigma^{-1}(i) < \sigma^{-1}(i') \;\Big|\; (i,i') \in G_p\Big] \;=\; \frac{\exp(\theta_i^*)}{\exp(\theta_i^*)+\exp(\theta_{i'}^*)} \,, \tag{2.42}$$

for all $i, i' \in S$ and also

$$\begin{aligned} &\mathbb{P}\Big[\sigma^{-1}(i) < \sigma^{-1}(i') \;\Big|\; (i,i') \in G_p \text{ and } \{i'' \in S \,:\, \sigma^{-1}(i'') < p\}\Big] \\ &\quad= \frac{\exp(\theta_i^*)}{\exp(\theta_i^*)+\exp(\theta_{i'}^*)} \,. \end{aligned} \tag{2.43}$$

This is one of the key technical lemmas since it implies that the proposed rank-breaking is consistent, i.e. $\mathbb{E}_{\theta^*}[\nabla \mathcal{L}_{\rm RB}(\theta^*)] = 0$. Throughout the proof of Theorem 2.2, this is the only place where the assumption on the proposed (consistent) rank-breaking is used. According to a companion theorem in [13, Theorem 2], it also follows that any rank-breaking that is not union of position-p rank-breakings results in inconsistency, i.e. $\mathbb{E}_{\theta^*}[\nabla \mathcal{L}_{\rm RB}(\theta^*)] \neq 0$. We claim that for each rank-breaking graph $G_{j,a}$, $\|\nabla \mathcal{L}_{G_{j,a}}(\theta^*)\|_2^2 \leq (\lambda_{j,a})^2(\kappa_j - p_{j,a})(\kappa_j - p_{j,a} + 1)$. By Lemma 2.13 which is a generalization of the vector version of the Azuma-Hoeffding inequality found in [89, Theorem 1.8], we have

$$\begin{aligned} & \mathbb{P}\big[\big\|\nabla\mathcal{L}_{\mathrm{RB}}(\theta^*)\big\|_2 \geq \delta\big] \\ \leq \quad & 2e^3 \exp\left(\frac{-\delta^2}{2\sum_{j=1}^{n}\sum_{a=1}^{\ell_j}\big(\lambda_{j,a}\big)^2\big(\kappa_j - p_{j,a}\big)\big(\kappa_j - p_{j,a}+1\big)}\right) , \end{aligned}$$

which implies the result.

Lemma 2.13. *Let $(X_1, X_2, \cdots, X_n)$ be real-valued martingale taking values in $\mathbb{R}^d$ such that $X_0 = 0$ and for every $1 \leq i \leq n$, $\|X_i - X_{i-1}\|_2 \leq c_i$, for some non-negative constant c_i. Then for every $\delta > 0$,*

$$\mathbb{P}[\|X_n\|_2 \geq \delta] \;\; \leq \;\; 2e^3 e^{-\frac{\delta^2}{2\sum_{i=1}^{n} c_i^2}} \, . \tag{2.44}$$

It follows from the upper bound on $\|\nabla\mathcal{L}_{G_{j,a}}(\theta^*)\|_2^2 \leq c_i^2$ with $c_i^2 = \lambda^2\big((k_j - p_{j,a})^2 + (k_j - p_{j,a})\big)$. In the expression (2.41), $\nabla\mathcal{L}_{G_{j,a}}(\theta^*)$ has one entry at $p_{j,a}$-th position that is compared to $(k_j - p_{j,a})$ other items and $(k_j - p_{j,a})$ entries that is compared only once, giving the bound

$$\|\nabla\mathcal{L}_{G_{j,a}}(\theta^*)\|_2^2 \;\; \leq \;\; \lambda_{j,a}^2(k_j - p_{j,a})^2 + \lambda_{j,a}^2(k_j - p_{j,a}) \, .$$

Proof of Lemma 2.12

Define event $E \equiv \{(i, i') \in G_p\}$. Observe that

$$E = \Big\{ \Big(\mathbb{I}_{\{(\sigma^{-1}(i)=p\}} + \mathbb{I}_{\{\sigma^{-1}(i'))=p\}} = 1 \Big) \wedge \Big(\sigma^{-1}(i), \sigma^{-1}(i') \geq p \Big) \Big\} \, .$$

Consider any set $\Omega \subset S \setminus \{i, i'\}$ such that $|\Omega| = p - 1$. Let M denote an event that items of the set Ω are ranked in top-$(p-1)$ positions in a particular order. It is easy to verify the following:

$$\begin{aligned}\mathbb{P}\Big[\sigma^{-1}(i) < \sigma^{-1}(i')\Big|E, M\Big] &= \frac{\mathbb{P}\Big[\big(\sigma^{-1}(i) < \sigma^{-1}(i')\big), E, M\Big]}{\mathbb{P}\Big[E, M\Big]} \\ &= \frac{\mathbb{P}\Big[\big(\sigma^{-1}(i) = p\big), M\Big]}{\mathbb{P}\Big[\big(\sigma^{-1}(i) = p\big), M\Big] + \mathbb{P}\Big[\big(\sigma^{-1}(i') = p\big), M\Big]} \\ &= \frac{\exp(\theta^*_i)}{\exp(\theta^*_i) + \exp(\theta^*_{i'})} = \mathbb{P}\Big[\sigma^{-1}(i) < \sigma^{-1}(i')\Big] \ .\end{aligned}$$

Since M is any particular ordering of the set Ω and Ω is any subset of $S\backslash\{i, i'\}$ such that $|\Omega| = p - 1$, conditioned on event E probabilities of all the possible events M over all the possible choices of set Ω sum to 1.

Proof of Lemma 2.13

It follows exactly along the lines of proof of Theorem 1.8 in [89].

Proof of Lemma 2.11

The Hessian $H(\theta)$ is given in (2.35). For all $j \in [n]$, define $M^{(j)} \in \mathcal{S}^d$ as

$$M^{(j)} \equiv \sum_{a=1}^{\ell_j} \lambda_{j,a} \sum_{i<i' \in S_j} \mathbb{I}_{\big\{(i,i') \in G_{j,a}\big\}} (e_i - e_{i'})(e_i - e_{i'})^\top, \qquad (2.45)$$

and let $M \equiv \sum_{j=1}^n M^{(j)}$. Observe that M is positive semi-definite and the smallest eigenvalue of M is zero with the corresponding eigenvector given by the all-ones vector. If $|\theta_i| \leq b$, for all $i \in [d]$, $\frac{\exp(\theta_i+\theta_{i'})}{[\exp(\theta_i)+\exp(\theta_{i'})]^2} \geq \frac{e^{2b}}{(1+e^{2b})^2}$. Recall the definition of $H(\theta)$ from Equation (2.35). It follows that $-H(\theta) \succeq \frac{e^{2b}}{(1+e^{2b})^2} M$ for $\theta \in \Omega_b$. Since, $-H(\theta)$ and M are symmetric matrices, from Weyl's inequality we have, $\lambda_2(-H(\theta)) \geq \frac{e^{2b}}{(1+e^{2b})^2}\lambda_2(M)$. Again from Weyl's inequality, it follows that

$$\lambda_2(M) \geq \lambda_2(\mathbb{E}[M]) - \|M - \mathbb{E}[M]\| \ , \qquad (2.46)$$

where $\|\cdot\|$ denotes the spectral norm. We will show in (2.51) that $\lambda_2(\mathbb{E}[M]) \geq 2\gamma e^{-6b}(\alpha/(d-1))\sum_{j=1}^{n}\tau_j\ell_j$, and in (2.63) that

$$\|M - \mathbb{E}[M]\| \leq 8e^{3b}\sqrt{\frac{\eta\delta\log d}{\beta\tau d}\sum_{j=1}^{n}\tau_j\ell_j}.$$

$$\lambda_2(M) \;\; \geq \;\; \frac{2e^{-6b}\alpha\gamma}{d-1}\sum_{j=1}^{n}\tau_j\ell_j - 8e^{3b}\sqrt{\frac{\eta\delta\log d}{\beta\tau d}\sum_{j=1}^{n}\tau_j\ell_j} \;\; \geq \;\; \frac{e^{-6b}\alpha\gamma}{d-1}\sum_{j=1}^{n}\tau_j\ell_j \;, \tag{2.47}$$

where the last inequality follows from the assumption that $\sum_{j=1}^{n}\tau_j\ell_j \geq 2^6 e^{18b}\frac{\eta\delta}{\alpha^2\beta\gamma^2\tau}d\log d$. This proves the desired claim.

To prove the lower bound on $\lambda_2(\mathbb{E}[M])$, notice that

$$\mathbb{E}[M] \;\; = \;\; \sum_{j=1}^{n}\sum_{a=1}^{\ell_j}\lambda_{j,a}\sum_{i<i'\in S_j}\mathbb{P}\Big[(i,i')\in G_{j,a}\Big|(i,i'\in S_j)\Big](e_i-e_{i'})(e_i-e_{i'})^\top \;. \tag{2.48}$$

The following lemma provides a lower bound on $\mathbb{P}[(i,i')\in G_{j,a}|(i,i'\in S_j)]$.

Lemma 2.14. *Consider a ranking σ over a set $S\subseteq[d]$ such that $|S|=\kappa$. For any two items $i,i'\in S$, $\theta\in\Omega_b$, and $1\leq\ell\leq\kappa-1$,*

$$\mathbb{P}_\theta\Big[\sigma^{-1}(i)=\ell,\sigma^{-1}(i')>\ell\Big] \;\; \geq \;\; \frac{e^{-6b}(\kappa-\ell)}{\kappa(\kappa-1)}\left(1-\frac{\ell}{\kappa}\right)^{\alpha_{i,i',\ell,\theta}-2} \;, \tag{2.49}$$

where the probability $\mathbb{P}_\theta$ is with respect to the sampled ranking resulting from PL weights $\theta\in\Omega_b$, and $\alpha_{i,i',\ell,\theta}$ is defined as $1\leq\alpha_{i,i',\ell,\theta}=\lceil\widetilde{\alpha}_{i,i',\ell,\theta}\rceil$, and $\widetilde{\alpha}_{i,i',\ell,\theta}$ is,

$$\widetilde{\alpha}_{i,i',\ell,\theta} \;\; \equiv \;\; \max_{\ell'\in[\ell]}\max_{\substack{\Omega\subseteq S\setminus\{i,i'\}\\ :|\Omega|=\kappa-\ell'}}\left\{\frac{\exp(\theta_i)+\exp(\theta_{i'})}{\left(\sum_{j\in\Omega}\exp(\theta_j)\right)/|\Omega|}\right\}. \tag{2.50}$$

Note that we do not need $\max_{\ell'\in[\ell]}$ in the above equation as the expression achieves its maxima at $\ell'=\ell$, but we keep the definition to avoid any confusion. In the worst case, $2e^{-2b}\leq\widetilde{\alpha}_{i,i',\ell,\theta}\leq 2e^{2b}$. Therefore, using definition of rank breaking graph $G_{j,a}$, and Equations (2.48) and (3.30) we have,

$$\begin{aligned}
\mathbb{E}[M] &\succeq \gamma e^{-6b} \sum_{j=1}^{n} \sum_{a=1}^{\ell_j} \lambda_{j,a} \frac{2(\kappa_j - p_{j,a})}{\kappa_j(\kappa_j-1)} \sum_{i<i' \in S_j} (e_i - e_{i'})(e_i - e_{i'})^\top \\
&\succeq 2\gamma e^{-6b} \sum_{j=1}^{n} \frac{1}{\kappa_j(\kappa_j-1)} \sum_{a=1}^{\ell_j} \lambda_{j,a}(\kappa_j - p_{j,a}) \sum_{i<i' \in S_j} (e_i - e_{i'})(e_i - e_{i'})^\top \\
&= 2\gamma e^{-6b} L, \qquad (2.51)
\end{aligned}$$

where we used $\gamma \le (1 - p_{j,\ell_j}/\kappa_j)^{\alpha_1 - 2}$ which follows for the definition in (2.7). (2.51) follows from the definition of Laplacian L, defined for the comparison graph $\mathcal{H}$ in Definition 2.9. Using $\lambda_2(L) = (\alpha/(d-1)) \sum_{j=1}^{n} \tau_j \ell_j$ from (2.30), we get the desired bound $\lambda_2(\mathbb{E}[M]) \ge 2\gamma e^{-6b}(\alpha/(d-1)) \sum_{j=1}^{n} \tau_j \ell_j$.

Next we need to upper bound $\| \sum_{j=1}^{n} \mathbb{E}[(M^j)^2] \|$ to bound the deviation of M from its expectation. To this end, we prove an upper bound on $\mathbb{P}[\sigma_j^{-1}(i) = p_{j,a} \mid i \in S_j]$ in the following lemma.

Lemma 2.15. *Under the hypotheses of Lemma 2.14,*

$$\mathbb{P}_\theta\Big[\sigma^{-1}(i) = \ell\Big] \;\le\; \frac{e^{6b}}{\kappa}\left(1 - \frac{\ell}{\kappa + \alpha_{i,\ell,\theta}}\right)^{\alpha_{i,\ell,\theta} - 1} \;\le\; \frac{e^{6b}}{\kappa - \ell}\,, \qquad (2.52)$$

where $0 \le \alpha_{i,\ell,\theta} = \lfloor \widetilde{\alpha}_{i,\ell,\theta} \rfloor$*, and* $\widetilde{\alpha}_{i,\ell,\theta}$ *is,*

$$\widetilde{\alpha}_{i,\ell,\theta} \;\equiv\; \min_{\ell' \in [\ell]} \; \min_{\substack{\Omega \in S \setminus \{i\} \\ : |\Omega| = \kappa - \ell' + 1}} \left\{ \frac{\exp(\theta_i)}{\big(\sum_{j \in \Omega} \exp(\theta_j)\big)/|\Omega|} \right\}. \qquad (2.53)$$

In the worst case, $e^{-2b} \le \widetilde{\alpha}_{i,\ell,\theta} \le e^{2b}$*. Note that* $\alpha_{i,\ell,\theta} = 0$ *gives the worst upper bound.*

Therefore using Equation (2.52), for all $i \in [d]$, we have,

$$\mathbb{P}\Big[\sigma_j^{-1}(i) \in \mathcal{P}_j\Big] \le \min\left\{1, \frac{e^{6b}\ell_j}{\kappa_j - p_{j,\ell_j}}\right\} \;\le\; \frac{e^{6b}\ell_j}{\max\{\ell_j, \kappa_j - p_{j,\ell_j}\}} \le \frac{e^{6b}\eta\ell_j}{\kappa_j}\,, \qquad (2.54)$$

where we used η defined in Equation (2.8). Define a diagonal matrix $D^{(j)} \in$

$\mathcal{S}^d$ and a matrix $A^{(j)} \in \mathcal{S}^d$,

$$A^{(j)}_{ii'} \;\equiv\; \mathbb{I}_{\big\{i,i' \in S_j\big\}} \sum_{a=1}^{\ell_j} \lambda_{j,a} \mathbb{I}_{\big\{(i,i') \in G_{j,a}\big\}}\,, \text{ for all } \; i, i' \in [d]\,, \tag{2.55}$$

and $D^{(j)}_{ii} = \sum_{i' \neq i} A^{(j)}_{ii'}$. Observe that $M^{(j)} = D^{(j)} - A^{(j)}$. For all $i \in [d]$, we have,

$$\begin{aligned} D^{(j)}_{ii} \;&=\; \mathbb{I}_{\big\{i \in S_j\big\}} \sum_{i'=1}^{\kappa_j} \mathbb{I}_{\big\{\sigma_j^{-1}(i) = i'\big\}} \sum_{a=1}^{\ell_j} \lambda_{j,a} \mathrm{deg}_{G_{j,a}}(\sigma_j^{-1}(i')) \\ &\leq\; \mathbb{I}_{\big\{i \in S_j\big\}} \Bigg\{ \mathbb{I}_{\big\{\sigma_j^{-1}(i) \in \mathcal{P}_j\big\}} \Bigg(\max_{a \in [\ell_j]} \Big\{ \lambda_{j,a}(\kappa_j - p_{j,a}) \Big\} + \sum_{a=1}^{\ell_j} \lambda_{j,a} \Bigg) \\ &\qquad\qquad + \mathbb{I}_{\big\{\sigma_j^{-1}(i) \notin \mathcal{P}_j\big\}} \Bigg(\sum_{a=1}^{\ell_j} \lambda_{j,a} \Bigg) \Bigg\} \\ &=\; \mathbb{I}_{\big\{i \in S_j\big\}} \Bigg\{ \mathbb{I}_{\big\{\sigma_j^{-1}(i) \in \mathcal{P}_j\big\}} \delta_{j,1} \;+\; \mathbb{I}_{\big\{\sigma_j^{-1}(i) \notin \mathcal{P}_j\big\}} \delta_{j,2} \Bigg\}, \end{aligned} \tag{2.56}$$

where the last equality follows from the definition of $\delta_{j,1}$ and $\delta_{j,2}$ in Equation (2.26). Note that $\max_{i \in [d]}\{D_{ii}\} = \delta_{j,1}$. Using (2.54) and (2.56), we have,

$$\mathbb{E}\Big[D^{(j)}_{ii}\Big] \;\leq\; \mathbb{I}_{\big\{i \in S_j\big\}} \Bigg\{ \frac{e^{6b} \eta \ell_j}{\kappa_j} \Bigg(\delta_{j,1} + \frac{\delta_{j,2} \kappa_j}{\eta \ell_j} \Bigg) \Bigg\}. \tag{2.57}$$

Similarly we have,

$$\mathbb{E}\Big[\big(D^{(j)}_{ii}\big)^2\Big] \;\leq\; \mathbb{I}_{\big\{i \in S_j\big\}} \Bigg\{ \frac{e^{6b} \eta \ell_j}{\kappa_j} \Bigg(\delta_{j,1}^2 + \frac{\delta_{j,2}^2 \kappa_j}{\eta \ell_j} \Bigg) \Bigg\} \tag{2.58}$$

For all $i \in [d]$, we have,

$$\begin{aligned} \mathbb{E}\Bigg[\sum_{i'=1}^{d} \big(\big(A^{(j)}\big)^2\big)_{ii'} \Bigg] \;&\leq\; \mathbb{E}\Bigg[\Bigg(\sum_{i'=1}^{d} A^{(j)}_{ii'} \Bigg) \max_{i \in [d]} \Bigg\{ \sum_{i'=1}^{d} A^{(j)}_{ii'} \Bigg\} \Bigg] \\ &\leq\; \mathbb{E}\Big[D^{(j)}_{ii} \delta_{j,1} \Big] \\ &\leq\; \mathbb{I}_{\big\{i \in S_j\big\}} \Bigg\{ \frac{e^{6b} \eta \ell_j}{\kappa_j} \Bigg(\delta_{j,1}^2 + \frac{\delta_{j,1} \delta_{j,2} \kappa_j}{\eta \ell_j} \Bigg) \Bigg\}. \end{aligned} \tag{2.59}$$

Using (2.58) and (2.59), we have, for all $i \in [d]$,

$$\begin{aligned}
& \sum_{i'=1}^{d} \Big| \mathbb{E}\Big[\big((M^{(j)})^2 \big)_{ii'} \Big] \Big| \\
= & \sum_{i'=1}^{d} \Bigg| \mathbb{E}\Big[\big((D^{(j)})^2 \big)_{ii'} \Big] - \mathbb{E}\Big[\big(D^{(j)} A^{(j)} \big)_{ii'} \Big] \\
& \qquad - \mathbb{E}\Big[\big(A^{(j)} D^{(j)} \big)_{ii'} \Big] + \mathbb{E}\Big[\big((A^{(j)})^2 \big)_{ii'} \Big] \Bigg| \\
\leq & \; 2\mathbb{E}\Big[\big(D^{(j)}_{ii} \big)^2 \Big] + \sum_{i'=1}^{d} \left(\mathbb{E}\Big[\delta_{j,1} \big(A^{(j)} \big)_{ii'} \Big] + \mathbb{E}\Big[\big((A^{(j)})^2 \big)_{ii'} \Big] \right) \\
\leq & \; \mathbb{I}_{\{i \in S_j\}} \left\{ \frac{e^{6b} \eta \ell_j}{\kappa_j} \left(4\delta_{j,1}^2 + \frac{2\big(\delta_{j,1}\delta_{j,2} + \delta_{j,2}^2\big)\kappa_j}{\eta \ell_j} \right) \right\} \\
= & \; \mathbb{I}_{\{i \in S_j\}} \left\{ \frac{e^{6b} \delta \eta \ell_j}{\kappa_j} \right\}, \qquad (2.60)
\end{aligned}$$

where the last equality follows from the definition of δ, Equation (2.27).

To bound $\| \sum_{j=1}^{n} \mathbb{E}[(M^{(j)})^2] \|$, we use the fact that for $J \in \mathbb{R}^{d \times d}$, $\|J\| \leq \max_{i \in [d]} \sum_{i'=1}^{d} |J_{ii'}|$. Therefore, we have

$$\begin{aligned}
\left\| \sum_{j=1}^{n} \mathbb{E}\Big[(M^{(j)})^2 \Big] \right\| \; & \leq \; e^{6b} \delta \eta \max_{i \in [d]} \left\{ \sum_{j : i \in S_j} \frac{\ell_j}{\kappa_j} \right\} \\
& = \; \frac{e^{6b} \eta \delta}{\tau} D_{\max} \qquad (2.61) \\
& = \; \frac{e^{6b} \eta \delta}{\beta \tau d} \sum_{j=1}^{n} \tau_j \ell_j \;, \qquad (2.62)
\end{aligned}$$

where (2.61) follows from the definition of $D_{\max}$ in Equation(2.28) and (2.62) follows from the definition of β in (2.30). Observe that from Equation (2.56), $\|M^{(j)}\| \leq 2\delta_{j,1} \leq 2\sqrt{\delta}$. Applying matrix Bernstein inequality, we have,

$$\mathbb{P}\Big[\big\| M - \mathbb{E}[M] \big\| \geq t \Big] \leq d \, \exp\left(\frac{-t^2/2}{\frac{e^{6b}\eta\delta}{\beta\tau d} \sum_{j=1}^{n} \tau_j \ell_j + 4\sqrt{\delta} t/3} \right).$$

Therefore, with probability at least $1 - d^{-3}$, we have,

$$\big\|M - \mathbb{E}[M]\big\| \le 4e^{3b}\sqrt{\frac{\eta\delta\log d}{\beta\tau d}\sum_{j=1}^{n}\tau_j\ell_j} + \frac{64\sqrt{\delta}\log d}{3} \le 8e^{3b}\sqrt{\frac{\eta\delta\log d}{\beta\tau d}\sum_{j=1}^{n}\tau_j\ell_j}\ , \tag{2.63}$$

where the second inequality uses $\sum_{j=1}^{n}\tau_j\ell_j \ge 2^6(\beta\tau/\eta)d\log d$ which follows from the assumption that $\sum_{j=1}^{n}\tau_j\ell_j \ge 2^6e^{18b}\frac{\eta\delta}{\tau\gamma^2\alpha^2\beta}d\log d$ and the fact that $\alpha,\beta \le 1$, $\gamma \le 1$, $\eta \ge 1$, and $\delta > \tau^2$.

Proof of Lemma 2.14

Since providing a lower bound on $\mathbb{P}_\theta\big[\sigma^{-1}(i) = \ell, \sigma^{-1}(i') > \ell\big]$ for arbitrary θ is challenging, we construct a new set of parameters $\{\widetilde{\theta}_j\}_{j\in[d]}$ from the original θ. These new parameters are constructed such that it is both easy to compute the probability and also provides a lower bound on the original distribution. We denote the sum of the weights by $W \equiv \sum_{j\in S}\exp(\theta_j)$. We define a new set of parameters $\{\widetilde{\theta}_j\}_{j\in S}$:

$$\widetilde{\theta}_j \;=\; \begin{cases} \log(\widetilde{\alpha}_{i,i',\ell,\theta}/2) & \text{for } j = i \text{ or } i'\ , \\ 0 & \text{otherwise}\ . \end{cases} \tag{2.64}$$

Similarly define $\widetilde{W} \equiv \sum_{j\in S}\exp(\widetilde{\theta}_j) = \kappa - 2 + \widetilde{\alpha}_{i,i',\ell,\theta}$. We have,

$$\begin{aligned}
& \mathbb{P}_\theta\Big[\sigma^{-1}(i)=\ell, \sigma^{-1}(i')>\ell\Big] \\
= \; & \sum_{\substack{j_1\in S\\ j_1\neq i,i'}} \Bigg(\frac{\exp(\theta_{j_1})}{W} \sum_{\substack{j_2\in S\\ j_2\neq i,i',j_1}} \Bigg(\frac{\exp(\theta_{j_2})}{W-\exp(\theta_{j_1})}\cdots \\
& \Bigg(\sum_{\substack{j_{\ell-1}\in S\\ j_{\ell-1}\neq i,i',\\ j_1,\cdots,j_{\ell-2}}} \frac{\exp(\theta_{j_{\ell-1}})}{W-\sum_{k=j_1}^{j_{\ell-2}}\exp(\theta_k)} \frac{\exp(\theta_i)}{W-\sum_{k=j_1}^{j_{\ell-1}}\exp(\theta_k)}\Bigg)\cdots\Bigg)\Bigg) \\
= \; & \frac{\exp(\theta_i)}{W} \sum_{\substack{j_1\in S\\ j_1\neq i,i'}} \Bigg(\frac{\exp(\theta_{j_1})}{W-\exp(\theta_{j_1})} \sum_{\substack{j_2\in S\\ j_2\neq i,i',j_1}} \Bigg(\frac{\exp(\theta_{j_2})}{W-\exp(\theta_{j_1})-\exp(\theta_{j_2})}\cdots \\
& \sum_{\substack{j_{\ell-1}\in S\\ j_{\ell-1}\neq i,i',\\ j_1,\cdots,j_{\ell-2}}} \Bigg(\frac{\exp(\theta_{j_{\ell-1}})}{W-\sum_{k=j_1}^{j_{\ell-1}}\exp(\theta_k)}\Bigg)\cdots\Bigg)\Bigg)
\end{aligned} \tag{2.65}$$

Consider the last summation term in the above equation and let $\Omega_\ell = S \setminus \{i, i', j_1, \ldots, j_{\ell-2}\}$. Observe that, $|\Omega_\ell| = \kappa - \ell$ and from equation (3.39),

$\frac{\exp(\theta_i)+\exp(\theta_{i'})}{\sum_{j\in\Omega_\ell}\exp(\theta_j)} \leq \frac{\widetilde{\alpha}_{i,i',\ell,\theta}}{\kappa-\ell}$. We have,

$$\begin{aligned}
& \sum_{j_{\ell-1}\in\Omega_\ell} \frac{\exp(\theta_{j_{\ell-1}})}{W-\sum_{k=j_1}^{j_{\ell-1}}\exp(\theta_k)} \\
= & \sum_{j_{\ell-1}\in\Omega_\ell} \frac{\exp(\theta_{j_{\ell-1}})}{W-\sum_{k=j_1}^{j_{\ell-2}}\exp(\theta_k)-\exp(\theta_{j_{\ell-1}})} \\
\geq & \frac{\sum_{j_{\ell-1}\in\Omega_\ell}\exp(\theta_{j_{\ell-1}})}{W-\sum_{k=j_1}^{j_{\ell-2}}\exp(\theta_k)-\big(\sum_{j_{\ell-1}\in\Omega_\ell}\exp(\theta_{j_{\ell-1}})\big)/|\Omega_\ell|} \qquad (2.66) \\
= & \frac{\sum_{j_{\ell-1}\in\Omega_\ell}\exp(\theta_{j_{\ell-1}})}{\exp(\theta_i)+\exp(\theta_{i'})+\sum_{j_{\ell-1}\in\Omega_\ell}\exp(\theta_{j_{\ell-1}})-\big(\sum_{j_{\ell-1}\in\Omega_\ell}\exp(\theta_{j_{\ell-1}})\big)/|\Omega_\ell|} \\
= & \left(\frac{\exp(\theta_i)+\exp(\theta_{i'})}{\sum_{j_{\ell-1}\in\Omega_\ell}\exp(\theta_{j_{\ell-1}})}+1-\frac{1}{\kappa-\ell}\right)^{-1} \\
\geq & \left(\frac{\widetilde{\alpha}_1}{\kappa-\ell}+1-\frac{1}{\kappa-\ell}\right)^{-1} \qquad (2.67) \\
= & \frac{\kappa-\ell}{\widetilde{\alpha}_1+\kappa-\ell-1} \\
= & \sum_{j_{\ell-1}\in\Omega_\ell} \frac{\exp(\widetilde{\theta}_{j_{\ell-1}})}{\widetilde{W}-\sum_{k=j_1}^{j_{\ell-2}}\exp(\widetilde{\theta}_k)-\exp(\widetilde{\theta}_{j_{\ell-1}})}\ , \qquad (2.68)
\end{aligned}$$

where (3.41) follows from the Jensen's inequality and the fact that for any $c > 0$, $0 < x < c$, $\frac{x}{c-x}$ is convex in x. Equation (3.42) follows from the definition of $\widetilde{\alpha}_{i,i',\ell,\theta}$, (3.39), and the fact that $|\Omega_\ell| = \kappa-\ell$. Equation (3.43) uses the definition of $\{\widetilde{\theta}_j\}_{j\in S}$.

Consider $\{\Omega_{\widetilde{\ell}}\}_{2\leq\widetilde{\ell}\leq\ell-1}$, $|\Omega_{\widetilde{\ell}}| = \kappa-\widetilde{\ell}$, corresponding to the subsequent summation terms in (3.40). Observe that $\frac{\exp(\theta_i)+\exp(\theta_{i'})}{\sum_{j\in\Omega_{\widetilde{\ell}}}\exp(\theta_j)} \leq \widetilde{\alpha}_{i,i',\ell,\theta}/|\Omega_{\widetilde{\ell}}|$. Therefore, each summation term in equation (3.40) can be lower bounded by the corresponding term where $\{\theta_j\}_{j\in S}$ is replaced by $\{\widetilde{\theta}_j\}_{j\in S}$. Hence, we have

$$\begin{aligned}
&\mathbb{P}_\theta\Big[\sigma^{-1}(i)=\ell,\sigma^{-1}(i')>\ell\Big]\\
\geq\;& \frac{\exp(\theta_i)}{W}\sum_{\substack{j_1\in S\\ j_1\neq i,i'}}\Bigg(\frac{\exp(\widetilde{\theta}_{j_1})}{\widetilde{W}-\exp(\widetilde{\theta}_{j_1})}\sum_{\substack{j_2\in S\\ j_2\neq i,i',j_1}}\Bigg(\frac{\exp(\widetilde{\theta}_{j_2})}{\widetilde{W}-\exp(\widetilde{\theta}_{j_1})-\exp(\widetilde{\theta}_{j_2})}\cdots\\
&\sum_{\substack{j_{\ell-1}\in S\\ j_{\ell-1}\neq i,i',\\ j_1,\cdots,j_{\ell-2}}}\Bigg(\frac{\exp(\widetilde{\theta}_{j_{\ell-1}})}{\widetilde{W}-\sum_{k=j_1}^{j_{\ell-1}}\exp(\widetilde{\theta}_k)}\Bigg)\Bigg)\Bigg)\\
\geq\;& \frac{e^{-4b}\exp(\widetilde{\theta}_i)}{\widetilde{W}}\sum_{\substack{j_1\in S\\ j_1\neq i,i'}}\Bigg(\frac{\exp(\widetilde{\theta}_{j_1})}{\widetilde{W}-\exp(\widetilde{\theta}_{j_1})}\sum_{\substack{j_2\in S\\ j_2\neq i,i',j_1}}\Bigg(\frac{\exp(\widetilde{\theta}_{j_2})}{\widetilde{W}-\exp(\widetilde{\theta}_{j_1})-\exp(\widetilde{\theta}_{j_2})}\cdots\\
&\sum_{\substack{j_{\ell-1}\in S\\ j_{\ell-1}\neq i,i',\\ j_1,\cdots,j_{\ell-2}}}\Bigg(\frac{\exp(\widetilde{\theta}_{j_{\ell-1}})}{\widetilde{W}-\sum_{k=j_1}^{j_{\ell-1}}\exp(\widetilde{\theta}_k)}\Bigg)\Bigg)\Bigg)\\
=\;& \big(e^{-4b}\big)\mathbb{P}_{\widetilde{\theta}}\Big[\sigma^{-1}(i)=\ell,\sigma^{-1}(i')>\ell\Big]\;. \qquad (2.69)
\end{aligned}$$

The second inequality uses $\frac{\exp(\theta_i)}{W}\geq e^{-2b}/\kappa$ and $\frac{\exp(\widetilde{\theta}_i)}{\widetilde{W}}\leq e^{2b}/\kappa$. Observe that $\exp(\widetilde{\theta}_j)=1$ for all $j\neq i,i'$ and $\exp(\widetilde{\theta}_i)+\exp(\widetilde{\theta}_{i'})=\widetilde{\alpha}_{i,i',\ell,\theta}\leq\lceil\widetilde{\alpha}_{i,i',\ell,\theta}\rceil=\alpha_{i,i',\ell,\theta}\geq 1$. Therefore, we have

$$\begin{aligned}
&\mathbb{P}_{\widetilde{\theta}}\Big[\sigma^{-1}(i)=\ell,\sigma^{-1}(i')>\ell\Big]\\
&=\binom{\kappa-2}{\ell-1}\frac{(\widetilde{\alpha}_{i,i',\ell,\theta}/2)(\ell-1)!}{(\kappa-2+\widetilde{\alpha}_{i,i',\ell,\theta})(\kappa-2+\widetilde{\alpha}_{i,i',\ell,\theta}-1)\cdots(\kappa-2+\widetilde{\alpha}_{i,i',\ell,\theta}-(\ell-1))}\\
&\geq\frac{(\kappa-2)!}{(\kappa-\ell-1)!}\frac{e^{-2b}}{(\kappa+\alpha_{i,i',\ell,\theta}-2)(\kappa+\alpha_{i,i',\ell,\theta}-3)\cdots(\kappa+\alpha_{i,i',\ell,\theta}-(\ell+1))}\\
&\qquad (2.70)\\
&=\frac{e^{-2b}(\kappa-\ell+\alpha_{i,i',\ell,\theta}-2)(\kappa-\ell+\alpha_{i,i',\ell,\theta}-3)\cdots(\kappa-\ell)}{(\kappa+\alpha_{i,i',\ell,\theta}-2)(\kappa+\alpha_{i,i',\ell,\theta}-3)\cdots(\kappa-1)}\\
&=\frac{e^{-2b}}{(\kappa-1)}\frac{(\kappa-\ell+\alpha_{i,i',\ell,\theta}-2)(\kappa-\ell+\alpha_{i,i',\ell,\theta}-3)\cdots(\kappa-\ell)}{(\kappa+\alpha_{i,i',\ell,\theta}-2)(\kappa+\alpha_{i,i',\ell,\theta}-3)\cdots(\kappa)}\\
&\geq\frac{e^{-2b}}{(\kappa-1)}\left(1-\frac{\ell}{\kappa}\right)^{\alpha_{i,i',\ell,\theta}-1}\\
&=\frac{e^{-2b}(\kappa-\ell)}{\kappa(\kappa-1)}\left(1-\frac{\ell}{\kappa}\right)^{\alpha_{i,i',\ell,\theta}-2}, \qquad (2.71)
\end{aligned}$$

where (2.70) follows from the fact that $\widetilde{\alpha}_{i,i',\ell,\theta} \geq 2e^{-2b}$. Claim (3.30) follows by combining Equations (3.44) and (3.45).

Proof of Lemma 2.15

Analogous to the proof of Lemma 2.14, we construct a new set of parameters $\{\widetilde{\theta}_j\}_{j\in[d]}$ from the original θ. We denote the sum of the weights by $W \equiv \sum_{j\in S}\exp(\theta_j)$. We define a new set of parameters $\{\widetilde{\theta}_j\}_{j\in S}$:

$$\widetilde{\theta}_j \quad = \quad \begin{cases} \log(\widetilde{\alpha}_{i,\ell,\theta}) & \text{for } j = i \;, \\ 0 & \text{otherwise} \;. \end{cases} \tag{2.72}$$

Similarly define $\widetilde{W} \equiv \sum_{j\in S}\exp(\widetilde{\theta}_j) = \kappa - 1 + \widetilde{\alpha}_{i,\ell,\theta}$. We have,

$$\begin{aligned}
&\mathbb{P}_\theta\Big[\sigma^{-1}(i) = \ell\Big] \\
&= \sum_{\substack{j_1\in S\\ j_1\neq i}}\left(\frac{\exp(\theta_{j_1})}{W}\sum_{\substack{j_2\in S\\ j_2\neq i,j_1}}\left(\frac{\exp(\theta_{j_2})}{W-\exp(\theta_{j_1})}\cdots\right.\right. \\
&\left.\left.\left(\sum_{\substack{j_{\ell-1}\in S\\ j_{\ell-1}\neq i,\\ j_1,\cdots,j_{\ell-2}}}\frac{\exp(\theta_{j_{\ell-1}})}{W-\sum_{k=j_1}^{j_{\ell-2}}\exp(\theta_k)}\frac{\exp(\theta_i)}{W-\sum_{k=j_1}^{j_{\ell-1}}\exp(\theta_k)}\right)\right)\right) \\
&\leq \sum_{\substack{j_1\in S\\ j_1\neq i}}\left(\frac{\exp(\theta_{j_1})}{W}\sum_{\substack{j_2\in S\\ j_2\neq i,j_1}}\left(\frac{\exp(\theta_{j_2})}{W-\exp(\theta_{j_1})}\cdots\right.\right. \\
&\left.\left.\left(\sum_{\substack{j_{\ell-1}\in S\\ j_{\ell-1}\neq i,\\ j_1,\cdots,j_{\ell-2}}}\frac{\exp(\theta_{j_{\ell-1}})}{W-\sum_{k=j_1}^{j_{\ell-2}}\exp(\theta_k)}\right)\right)\right)\frac{e^{2b}}{\kappa-\ell+1}
\end{aligned} \tag{2.73}$$

Consider the last summation term in the equation (3.47), and let $\Omega_\ell = S \setminus \{i, j_1, \ldots, j_{\ell-2}\}$, such that $|\Omega_\ell| = \kappa - \ell + 1$. Observe that from equation

(2.53), $\frac{\exp(\theta_i)}{\sum_{j\in\Omega_\ell}\exp(\theta_j)} \geq \frac{\widetilde{\alpha}_{i,\ell,\theta}}{\kappa-\ell+1}$. We have,

$$\begin{aligned}
\sum_{j_{\ell-1}\in\Omega_\ell} \frac{\exp(\theta_{j_{\ell-1}})}{W-\sum_{k=j_1}^{j_{\ell-2}}\exp(\theta_k)} &= \frac{\sum_{j_{\ell-1}\in\Omega_\ell}\exp(\theta_{j_{\ell-1}})}{\exp(\theta_i)+\sum_{j_{\ell-1}\in\Omega_\ell}\exp(\theta_{j_{\ell-1}})} \\
&\leq \left(\frac{\widetilde{\alpha}_{i,\ell,\theta}}{\kappa-\ell+1}+1\right)^{-1} \\
&= \frac{\kappa-\ell+1}{\widetilde{\alpha}_{i,\ell,\theta}+\kappa-\ell+1} \\
&= \sum_{j_{\ell-1}\in\Omega_\ell} \frac{\exp(\widetilde{\theta}_{j_{\ell-1}})}{\widetilde{W}-\sum_{k=j_1}^{j_{\ell-2}}\exp(\widetilde{\theta}_k)}, \qquad (2.74)
\end{aligned}$$

where (2.74) follows from the definition of $\{\widetilde{\theta}\}_{j\in S}$.

Consider $\{\Omega_{\widetilde{\ell}}\}_{2\leq\widetilde{\ell}\leq\ell-1}$, $|\Omega_{\widetilde{\ell}}| = \kappa-\widetilde{\ell}+1$, corresponding to the subsequent summation terms in (3.47). Observe that $\frac{\exp(\theta_i)}{\sum_{j\in\Omega_{\widetilde{\ell}}}\exp(\theta_j)} \geq \widetilde{\alpha}_{i,\ell,\theta}/|\Omega_{\widetilde{\ell}}|$. Therefore, each summation term in equation (3.40) can be lower bounded by the corresponding term where $\{\theta_j\}_{j\in S}$ is replaced by $\{\widetilde{\theta}_j\}_{j\in S}$. Hence, we have

$$\begin{aligned}
&\mathbb{P}_\theta\Big[\sigma^{-1}(i)=\ell\Big] \\
\leq\ & \sum_{\substack{j_1\in S\\ j_1\neq i}}\left(\frac{\exp(\widetilde{\theta}_{j_1})}{\widetilde{W}}\sum_{\substack{j_2\in S\\ j_2\neq i,j_1}}\left(\frac{\exp(\widetilde{\theta}_{j_2})}{\widetilde{W}-\exp(\widetilde{\theta}_{j_1})}\cdots\right.\right. \\
&\quad \left.\left.\left(\sum_{\substack{j_{\ell-1}\in S\\ j_{\ell-1}\neq i,\\ j_1,\cdots,j_{\ell-2}}}\frac{\exp(\widetilde{\theta}_{j_{\ell-1}})}{\widetilde{W}-\sum_{k=j_1}^{j_{\ell-2}}\exp(\widetilde{\theta}_k)}\right)\right)\right)\frac{e^{2b}}{\kappa-\ell+1} \\
\leq\ & e^{4b}\sum_{\substack{j_1\in S\\ j_1\neq i}}\left(\frac{\exp(\widetilde{\theta}_{j_1})}{\widetilde{W}}\sum_{\substack{j_2\in S\\ j_2\neq i,j_1}}\left(\frac{\exp(\widetilde{\theta}_{j_2})}{\widetilde{W}-\exp(\widetilde{\theta}_{j_1})}\cdots\right.\right. \\
&\quad \left.\left.\left(\sum_{\substack{j_{\ell-1}\in S\\ j_{\ell-1}\neq i,\\ j_1,\cdots,j_{\ell-2}}}\frac{\exp(\widetilde{\theta}_{j_{\ell-1}})}{\widetilde{W}-\sum_{k=j_1}^{j_{\ell-2}}\exp(\widetilde{\theta}_k)}\frac{\exp(\widetilde{\theta}_i)}{\widetilde{W}-\sum_{k=j_1}^{j_{\ell-1}}\exp(\widetilde{\theta}_k)}\right)\right)\right) \\
\leq\ & e^{4b}\mathbb{P}_{\widetilde{\theta}}\Big[\sigma^{-1}(i)=\ell\Big] \qquad (2.75)
\end{aligned}$$

The second inequality uses $\widetilde{\alpha}_2/(\kappa-\ell+\widetilde{\alpha}_{i,\ell,\theta}) \geq e^{-2b}/(\kappa-\ell+1)$. Observe that $\exp(\widetilde{\theta}_j) = 1$ for all $j\neq i$ and $\exp(\widetilde{\theta}_i) = \widetilde{\alpha}_{i,\ell,\theta} \geq \lfloor\widetilde{\alpha}_{i,\ell,\theta}\rfloor = \alpha_{i,\ell,\theta} \geq 0$.

Therefore, we have

$$\begin{aligned}\mathbb{P}_{\widetilde{\theta}}\Big[\sigma^{-1}(i)=\ell\Big] &= \binom{\kappa-1}{\ell-1}\frac{\widetilde{\alpha}_{i,\ell,\theta}(\ell-1)!}{(\kappa-1+\widetilde{\alpha}_{i,\ell,\theta})(\kappa-2+\widetilde{\alpha}_{i,\ell,\theta})\cdots(\kappa-\ell+\widetilde{\alpha}_{i,\ell,\theta})}\\ &\le \frac{(\kappa-1)!}{(\kappa-\ell)!}\frac{e^{2b}}{(\kappa-1+\alpha_{i,\ell,\theta})(\kappa-2+\alpha_{i,\ell,\theta})\cdots(\kappa-\ell+\alpha_{i,\ell,\theta})}\\ &\le \frac{e^{2b}}{\kappa}\bigg(1-\frac{\ell}{\kappa+\alpha_{i,\ell,\theta}}\bigg)^{\alpha_{i,\ell,\theta}-1}, \qquad (2.76)\end{aligned}$$

Note that equation (3.48) holds for all values of $\alpha_{i,\ell,\theta}\ge 0$. Claim 2.52 follows by combining Equations (2.75) and (3.48).

2.7.3 Proof of Theorem 2.4

Let $H(\theta)\in\mathcal{S}^d$ be Hessian matrix such that $H_{ii'}(\theta)=\frac{\partial^2\mathcal{L}(\theta)}{\partial\theta_i\partial\theta_{i'}}$. The Fisher information matrix is defined as $I(\theta)=-\mathbb{E}_\theta[H(\theta)]$. Fix any unbiased estimator $\widehat{\theta}$ of $\theta\in\Omega_b$. Since, $\widehat{\theta}\in\mathcal{U}$, $\widehat{\theta}-\theta$ is orthogonal to $\mathbf{1}$. The Cramér-Rao lower bound then implies that $\mathbb{E}[\|\widehat{\theta}-\theta^*\|^2]\ge\sum_{i=2}^d\frac{1}{\lambda_i(I(\theta))}$. Taking the supremum over both sides gives

$$\sup_\theta\mathbb{E}[\|\widehat{\theta}-\theta\|^2] \;\ge\; \sup_\theta\sum_{i=2}^d\frac{1}{\lambda_i(I(\theta))}\ge\sum_{i=2}^d\frac{1}{\lambda_i(I(\mathbf{0}))}\,.$$

The following lemma provides a lower bound on $\mathbb{E}_\theta[H(\mathbf{0})]$, where $\mathbf{0}$ indicates the all-zeros vector.

Lemma 2.16. *Under the hypotheses of Theorem 2.4,*

$$\mathbb{E}_\theta[H(\mathbf{0})] \;\succeq\; -\sum_{j=1}^n\frac{2p\log(\kappa_j)^2}{\kappa_j(\kappa_j-1)}\sum_{i'<i\in S_j}(e_i-e_{i'})(e_i-e_{i'})^\top\,. \qquad (2.77)$$

Observe that $I(\mathbf{0})$ is positive semi-definite. Moreover, $\lambda_1(I(\mathbf{0}))$ is zero and the corresponding eigenvector is the all-ones vector. It follows that

$$\begin{aligned}I(0) &\preceq \sum_{j=1}^n\frac{2p\log(\kappa_j)^2}{\kappa_j(\kappa_j-1)}\sum_{i'<i\in S_j}(e_i-e_{i'})(e_i-e_{i'})^\top\\ &\preceq 2p\log(\kappa_{\max})^2\underbrace{\sum_{j=1}^n\frac{1}{\kappa_j(\kappa_j-1)}\sum_{i'<i\in S_j}(e_i-e_{i'})(e_i-e_{i'})^\top}_{=L},\end{aligned}$$

where L is the Laplacian defined for the comparison graph $\mathcal{H}$, Definition 2.1, as $\ell_j = 1$ for all $j \in [n]$ in this setting. By Jensen's inequality, we have

$$\sum_{i=2}^{d} \frac{1}{\lambda_i(L)} \geq \frac{(d-1)^2}{\sum_{i=2}^{d} \lambda_i(L)} = \frac{(d-1)^2}{\mathrm{Tr}(L)} = \frac{(d-1)^2}{n}.$$

Proof of Lemma 2.16

Define $\mathcal{L}_j(\theta)$ for $j \in [n]$ such that $\mathcal{L}(\theta) = \sum_{j=1}^{n} \mathcal{L}_j(\theta)$. Let $H^{(j)}(\theta) \in \mathcal{S}^d$ be the Hessian matrix such that $H^{(j)}_{ii'}(\theta) = \frac{\partial^2 \mathcal{L}_j(\theta)}{\partial\theta_i \partial\theta_{i'}}$ for $i, i' \in S_j$. We prove that for all $j \in [n]$,

$$\mathbb{E}_\theta[H^{(j)}(\mathbf{0})] \;\succeq\; -\frac{2p\log(\kappa_j)^2}{\kappa_j(\kappa_j - 1)} \sum_{i' < i \in S_j} (e_i - e_{i'})(e_i - e_{i'})^\top . \tag{2.78}$$

In the following, we omit superscript/subscript j for brevity. With a slight abuse of notation, we use $\mathbb{I}_{\{\Omega^{-1}(i)=a\}} = 1$ if item i is ranked at the a-th position in all the orderings $\sigma \in \Omega$. Let $\mathbb{P}[\theta]$ be the likelihood of observing $\Omega^{-1}(p) = i^{(p)}$ and the set Λ (the set of the items that are ranked before the p-th position). We have,

$$\mathbb{P}(\theta) = \sum_{\sigma \in \Omega} \left(\frac{\exp\left(\sum_{m=1}^{p} \theta_{\sigma(m)}\right)}{\prod_{a=1}^{p} \left(\sum_{m'=a}^{\kappa} \exp\left(\theta_{\sigma(m')}\right)\right)} \right). \tag{2.79}$$

For $i, i' \in S_j$, we have

$$H_{ii'}(\theta) = \frac{1}{\mathbb{P}(\theta)} \frac{\partial^2 \mathbb{P}(\theta)}{\partial\theta_i \partial\theta_{i'}} - \frac{\nabla_i \mathbb{P}(\theta) \nabla_{i'} \mathbb{P}(\theta)}{\left(\mathbb{P}(\theta)\right)^2} \tag{2.80}$$

We claim that at $\theta = \mathbf{0}$,

$$-H_{ii'}(\mathbf{0}) = \begin{cases} C_1 & \text{if } i = i',\ \left\{\Omega^{-1}(i) \geq p\right\} \\ C_2 + A_3^2 - C_3 & \text{if } i = i',\ \left\{\Omega^{-1}(i) < p\right\} \\ -B_1 & \text{if } i \neq i',\ \left\{\Omega^{-1}(i) \geq p,\ \Omega^{-1}(i') \geq p\right\} \\ -B_2 & \text{if } i \neq i',\ \left\{\Omega^{-1}(i) \geq p,\ \Omega^{-1}(i') < p\right\} \\ -B_2 & \text{if } i \neq i',\ \left\{\Omega^{-1}(i) < p,\ \Omega^{-1}(i') \geq p\right\} \\ -(B_3 + B_4 - A_3^2) & \text{if } i \neq i',\ \left\{\Omega^{-1}(i) < p,\ \Omega^{-1}(i') < p\right\} . \end{cases} \tag{2.81}$$

where constants $A_3, B_1, B_2, B_3, B_4, C_1, C_2$ and C_3 are defined in Equations (2.88), (2.90), (2.91), (2.92), (2.93), (2.95), (2.96) and (2.97) respectively. From this computation of the Hessian, note that we have

$$H(\mathbf{0}) = \sum_{i'<i\in S} (e_i - e_{i'})(e_i - e_{i'})^\top \Big(H_{ii'}(\mathbf{0}) \Big) \,. \tag{2.82}$$

which follows directly from the fact that the diagonal entries are summations of the off-diagonals, i.e. $C_1 = B_1(\kappa - p) + B_2(p-1)$ and $C_2 + A_3^2 - C_3 = B_2(\kappa - p + 1) + (B_3 + B_4 - A_3^2)(p-2)$. The second equality follows from the fact that $C_2 = B_2(\kappa - p + 1) + B_3(p-2)$ and $A_3^2(p-1) = B_4(p-2) + C_3$. Note that since $\theta = \mathbf{0}$, all items are exchangeable. Hence, $\mathbb{E}[H_{ii'}(\mathbf{0})] = \mathbb{E}[H_{ii}(\mathbf{0})]/(\kappa - 1)$, and substituting this into (2.82) and using Equations (2.81), we get

$$\begin{aligned}
&\mathbb{E}\Big[H(\mathbf{0})\Big] \\
&= -\frac{1}{\kappa - 1}\Big(\mathbb{P}\big[\Omega^{-1}(i) \geq p\big] C_1 + \mathbb{P}\big[\Omega^{-1}(i) < p\big](C_2 + A_3^2 - C_3) \Big) \\
&\quad \sum_{i'<i\in S} (e_i - e_{i'})(e_i - e_{i'})^\top \\
&\succeq -\frac{1}{\kappa(\kappa-1)} \sum_{i'<i\in S} (e_i - e_{i'})(e_i - e_{i'})^\top \\
&\Bigg((\kappa - p + 1)\log\left(\frac{\kappa}{\kappa - p}\right) + (p-1)\Bigg(\log\left(\frac{\kappa}{\kappa - p + 1}\right) + \log\left(\frac{\kappa}{\kappa - p + 1}\right)^2 \Bigg)\Bigg)
\end{aligned} \tag{2.83}$$

$$\succeq -\frac{2p\log(\kappa)^2}{\kappa(\kappa-1)} \sum_{i'<i\in S} (e_i - e_{i'})(e_i - e_{i'})^\top \,, \tag{2.84}$$

where (2.83) uses $\sum_{a=1}^{p} \frac{1}{\kappa - a + 1} \leq \log\left(\frac{\kappa}{\kappa - p}\right)$ and $C_3 \geq 0$. Equation (2.84) follows from the fact that for any $x > 0$, $\log(1+x) \leq x$. To prove (2.81), we have the first order partial derivative of $\mathbb{P}(\theta)$ given by

$$\begin{aligned}
&\nabla_i \mathbb{P}(\theta) \\
&= \mathbb{I}_{\{\Omega^{-1}(i) \leq p\}} \mathbb{P}(\theta) \\
&- \sum_{\sigma\in\Omega} \left(\frac{\exp\left(\sum_{m=1}^{p} \theta_{\sigma(m)}\right)}{\prod_{a=1}^{p}\left(\sum_{m'=a}^{\kappa} \exp\left(\theta_{\sigma(m')}\right)\right)} \left(\sum_{a=1}^{p} \frac{\mathbb{I}_{\{\sigma^{-1}(i) \geq a\}} \exp(\theta_i)}{\sum_{m'=a}^{\kappa} \exp\left(\theta_{\sigma(m')}\right)} \right)\right) .
\end{aligned} \tag{2.85}$$

Define constants A_1, A_2 and A_3 such that

$$A_1 \equiv \mathbb{P}(\theta)\big|_{\{\theta=\mathbf{0}\}} = \frac{(p-1)!}{\kappa(\kappa-1)\cdots(\kappa-p+1)}, \tag{2.86}$$

$$A_2 \equiv \left(\sum_{a=1}^{p} \frac{\exp(\theta_i)}{\sum_{m'=a}^{\kappa} \exp\left(\theta_{\sigma(m')}\right)}\right)\Bigg|_{\{\theta=\mathbf{0}\}} = \left(\frac{1}{\kappa} + \frac{1}{\kappa-1} + \cdots + \frac{1}{\kappa-p+1}\right), \tag{2.87}$$

$$A_3 \equiv \left(\frac{(p-1)(p-2)!}{(p-1)!(\kappa)} + \frac{(p-2)(p-2)!}{(p-1)!(\kappa-1)} + \cdots + \frac{(p-2)!}{(p-1)!(\kappa-p+2)}\right). \tag{2.88}$$

Observe that, for all $i \in [d]$,

$$\nabla_i \mathbb{P}(\theta)\big|_{\{\theta=\mathbf{0}\}} = A_1\Big(\mathbb{I}_{\{\Omega_j^{-1}(i)=p\}}(1-A_2) + \mathbb{I}_{\{\Omega_j^{-1}(i)<p\}}(1-A_3) - \mathbb{I}_{\{\Omega_j^{-1}(i)>p\}}A_2\Big) \quad . \tag{2.89}$$

Further define constants B_1, B_2, B_3 and B_4 such that

$$B_1 \equiv \left(\frac{1}{\kappa^2} + \frac{1}{(\kappa-1)^2} + \cdots + \frac{1}{(\kappa-p+1)^2}\right), \tag{2.90}$$

$$B_2 \equiv \left(\frac{p-1}{(p-1)\kappa^2} + \frac{p-2}{(p-1)(\kappa-1)^2} + \cdots + \frac{1}{(p-1)(\kappa-p+2)^2}\right), \tag{2.91}$$

$$B_3 \equiv \left(\frac{(p-1)(p-2)(p-3)!}{(p-1)!\kappa^2} + \frac{(p-2)(p-3)(p-3)!}{(p-1)!(\kappa-1)^2} + \cdots + \frac{2(p-3)!}{(p-1)!(\kappa-p+3)^2}\right), \tag{2.92}$$

$$B_4 \equiv \frac{(p-3)!}{(p-1)!}\left(\sum_{a,b\in[p-1],b\neq a} \left(\frac{1}{\kappa} + \frac{1}{\kappa-1} + \cdots + \frac{1}{\kappa-a+1}\right)\left(\frac{1}{\kappa} + \frac{1}{\kappa-1} + \cdots + \frac{1}{\kappa-b+1}\right)\right). \tag{2.93}$$

Observe that,

$$
\begin{aligned}
& \left.\frac{\partial^2 \mathbb{P}(\theta)}{\partial\theta_i \partial\theta_{i'}}\right|_{\theta=\mathbf{0}} \\
= \; & \mathbb{I}_{\left\{\Omega^{-1}(i),\Omega^{-1}(i')>p\right\}} A_1\Big((-A_2)(-A_2)+B_1\Big) \\
& + \Big(\mathbb{I}_{\left\{\Omega^{-1}(i)>p,\Omega^{-1}(i')=p\right\}} + \mathbb{I}_{\left\{\Omega^{-1}(i)=p,\Omega^{-1}(i')>p\right\}}\Big) A_1\Big((-A_2)(1-A_2)+B_1\Big) \\
& + \Big(\mathbb{I}_{\left\{\Omega^{-1}(i)=p,\Omega^{-1}(i')<p\right\}} + \mathbb{I}_{\left\{\Omega^{-1}(i)<p,\Omega^{-1}(i')=p\right\}}\Big) \\
& A_1\Big((1-A_3)+(-A_2)(1-A_3)+B_2\Big) \\
& + \Big(\mathbb{I}_{\left\{\Omega^{-1}(i)>p,\Omega^{-1}(i')<p\right\}} + \mathbb{I}_{\left\{\Omega^{-1}(i)<p,\Omega^{-1}(i')>p\right\}}\Big) A_1\Big((-A_2)(1-A_3)+B_2\Big) \\
& + \mathbb{I}_{\left\{\Omega^{-1}(i)<p,\Omega^{-1}(i')<p\right\}} A_1\Big((1-A_3)+(-A_3)+B_4+B_3\Big)\,. \qquad (2.94)
\end{aligned}
$$

The claims in (2.81) are easy to verify by combining Equations (2.89) and (2.94) with (2.80). Also, define constants C_1, C_2 and C_3 such that,

$$
\begin{aligned}
C_1 &\equiv \left(\frac{\kappa-1}{(\kappa)^2} + \frac{\kappa-2}{(\kappa-1)^2} + \cdots + \frac{\kappa-p}{(\kappa-p+1)^2}\right), \qquad (2.95) \\
C_2 &\equiv \left(\frac{(p-1)(p-2)!(\kappa-1)}{(p-1)!(\kappa)^2} + \frac{(p-2)(p-2)!(\kappa-2)}{(p-1)!(\kappa-1)^2} + \cdots \right. \\
& \quad \left. + \frac{(p-2)!(\kappa-p+1)}{(p-1)!(\kappa-p+2)^2}\right), \qquad (2.96) \\
C_3 &\equiv \frac{(p-2)!}{(p-1)!}\left(\sum_{a,b\in[p-1],b=a}\left(\frac{1}{\kappa}+\frac{1}{\kappa-1}+\cdots+\frac{1}{\kappa-a+1}\right)\right. \\
& \quad \left.\left(\frac{1}{\kappa}+\frac{1}{\kappa-1}+\cdots+\frac{1}{\kappa-b+1}\right)\right), \qquad (2.97)
\end{aligned}
$$

such that,

$$
\begin{aligned}
\left.\frac{\partial^2 \mathbb{P}(\theta)}{\partial\theta_i^2}\right|_{\theta=\mathbf{0}} &= \mathbb{I}_{\{\Omega^{-1}(i)>p\}} A_1\Big((-A_2)(-A_2)-C_1\Big) \\
& \quad + \mathbb{I}_{\{\Omega^{-1}(i)=p\}} A_1\Big((1-A_2)-A_2(1-A_2)-C_1\Big) \\
& \quad + \mathbb{I}_{\{\Omega^{-1}(i)<p\}} A_1\Big((1-A_3)-A_3-C_2+C_3\Big)\,. \qquad (2.98)
\end{aligned}
$$

The claims (2.81) is easy to verify by combining Equations (2.89) and (2.98) with (2.80).

2.7.4 Proof of Theorem 2.5

The proof is analogous to the proof of Theorem 7.3. It differs primarily in the lower bound that is achieved for the second smallest eigenvalue of the Hessian matrix $H(\theta)$, (2.35).

Lemma 2.17. *Under the hypotheses of Theorem 2.5, if*

$$\sum_{j=1}^{n} \ell_j \geq (2^{12} e^{6b}/\beta\alpha^2) d \log d$$

then with probability at least $1 - d^{-3}$,

$$\lambda_2(-H(\theta)) \;\geq\; \frac{\alpha}{2(1+e^{2b})^2} \frac{1}{d-1} \sum_{j=1}^{n} \ell_j \,. \tag{2.99}$$

Using Lemma 2.10 that is derived for the general value of $\lambda_{j,a}$ and $p_{j,a}$, and by substituting $\lambda_{j,a} = 1/(\kappa_j - 1)$ and $p_{j,a} = a$ for each $j \in [n]$, we get that with probability at least $1 - 2e^3 d^{-3}$,

$$\|\nabla \mathcal{L}_{\mathrm{RB}}(\theta^*)\|_2 \;\leq\; \sqrt{16 \log d \sum_{j=1}^{n} \ell_j} \,. \tag{2.100}$$

Theorem 2.5 follows from Equations (2.100), (2.99) and (2.40).

Proof of Lemma 2.17

Define $M^{(j)} \in \mathcal{S}^d$ as

$$M^{(j)} \;=\; \frac{1}{\kappa_j - 1} \sum_{i<i' \in S_j} \sum_{a=1}^{\ell_j} \mathbb{I}_{\{(i,i') \,\in\, G_{j,a}\}} (e_i - e_{i'})(e_i - e_{i'})^\top, \tag{2.101}$$

and let $M = \sum_{j=1}^{n} M^{(j)}$. Similar to the analysis carried out in the proof of Lemma 2.11, we have $\lambda_2(-H(\theta)) \geq \frac{e^{2b}}{(1+e^{2b})^2} \lambda_2(M)$, when $\lambda_{j,a} = 1/(\kappa_j - 1)$ is substituted in the Hessian matrix $H(\theta)$, Equation (2.35). From Weyl's inequality we have that

$$\lambda_2(M) \quad \geq \lambda_2(\mathbb{E}[M]) - \|M - \mathbb{E}[M]\| \,. \tag{2.102}$$

We will show in (2.107) that $\lambda_2(\mathbb{E}[M]) \geq e^{-2b}(\alpha/(d-1))\sum_{j=1}^{n}\ell_j$ and in (2.112) that $\|M - \mathbb{E}[M]\| \leq 32e^b\sqrt{\frac{\log d}{\beta d}\sum_{j=1}^{n}\ell_j}$.

$$\lambda_2(M) \;\geq\; \frac{\alpha e^{-2b}}{d-1}\sum_{j=1}^{n}\ell_j \;-\; 32e^b\sqrt{\frac{\log d}{\beta d}\sum_{j=1}^{n}\ell_j} \;\geq\; \frac{\alpha e^{-2b}}{2(d-1)}\sum_{j=1}^{n}\ell_j \;, \tag{2.103}$$

where the last inequality follows from the assumption that

$$\sum_{j=1}^{n}\ell_j \geq (2^{12}e^{6b}/\beta\alpha^2)d\log d \,.$$

This proves the desired claim.

To prove the lower bound on $\lambda_2(\mathbb{E}[M])$, notice that

$$\begin{aligned}&\mathbb{E}[M]\\ =\;& \sum_{j=1}^{n}\frac{1}{\kappa_j-1}\sum_{i<i'\in S_j}\mathbb{E}\left[\sum_{a=1}^{\ell_j}\mathbb{I}_{\{(i,i')\in G_{j,a}\}}\middle|(i,i'\in S_j)\right](e_i-e_{i'})(e_i-e_{i'})^\top \,.\end{aligned} \tag{2.104}$$

Using the fact that $p_{j,a} = a$ for each $j \in [n]$, and the definition of rank-breaking graph $G_{j,a}$, we have that

$$\begin{aligned}&\mathbb{E}\left[\sum_{a=1}^{\ell_j}\mathbb{I}_{\{(i,i')\in G_{j,a}\}}\middle|(i,i'\in S_j)\right]\\ =\;& \mathbb{P}\Big[\mathbb{I}_{\{\sigma_j^{-1}(i)\leq\ell_j\}} + \mathbb{I}_{\{\sigma_j^{-1}(i')\leq\ell_j\}} \geq 1\Big|(i,i'\in S_j)\Big]\\ \geq\;& \mathbb{P}\Big[(\sigma^{-1}(i)\leq\ell_j\Big|(i,i'\in S_j)\Big] \,.\end{aligned} \tag{2.105}$$

The following lemma provides a lower bound on $\mathbb{P}[(\sigma^{-1}(i)\leq\ell_j|(i,i'\in S_j)]$.

Lemma 2.18. *Consider a ranking σ over a set of items S of size κ. For any* *item $i \in S$,*

$$\mathbb{P}[(\sigma^{-1}(i) \leq \ell] \geq e^{-2b}\frac{\ell}{\kappa} \; . \tag{2.106}$$

Therefore, using the fact that $(e_i - e_{i'})(e_i - e_{i'})^\top$ is positive semi-definite, and Equations (2.104), (2.105) and (2.106) we have

$$\mathbb{E}[M] \;\; \succeq \;\; e^{-2b}\sum_{j=1}^{n}\frac{\ell_j}{\kappa_j(\kappa_j-1)}\sum_{i<i'\in S_j}(e_i - e_{i'})(e_i - e_{i'})^\top = e^{-2b}L, \tag{2.107}$$

where L is the Laplacian defined for the comparison graph $\mathcal{H}$, Definition 2.1. Using $\lambda_2(L) = (\alpha/(d-1))\sum_{j=1}^{n}\ell_j$ from (2.5), we get the desired bound $\lambda_2(\mathbb{E}[M]) \geq e^{-2b}(\alpha/(d-1))\sum_{j=1}^{n}\ell_j$.

For top-ℓ_j rank breaking, $M^{(j)}$ is also given by

$$\begin{aligned} & M^{(j)} \\ & = \frac{1}{\kappa_j - 1}\Big((\kappa_j - \ell_j)\mathrm{diag}(e_{\{I_j\}}) + \ell_j\mathrm{diag}(e_{\{S_j\}}) \\ & - e_{\{I_j\}}e_{\{S_j\}}^\top - e_{\{S_j\}}e_{\{I_j\}}^\top + e_{\{I_j\}}e_{\{I_j\}}^\top\Big), \end{aligned} \tag{2.108}$$

where $e_{\{S_j\}}, e_{\{I_j\}} \in \mathbb{R}^d$ are zero-one vectors, $e_{\{S_j\}}$ has support corresponding to the set of items S_j and $e_{\{I_j\}}$ has support corresponding to the random top-ℓ_j items in the ranking σ_j. $I_j = \{\sigma_j(1), \sigma_j(2), \cdots, \sigma_j(\ell_j)\}$ for $j \in [n]$. $(M^{(j)})^2$ is given by

$$\begin{aligned} (M^{(j)})^2 \;\; = \;\; & \frac{1}{(\kappa_j - 1)^2}\Big((\kappa_j^2 - \ell_j^2)\mathrm{diag}(e_{\{I_j\}}) + {\ell_j}^2\mathrm{diag}(e_{\{S_j\}}) - \\ & (\kappa_j + \ell_j)(e_{\{I_j\}}e_{\{S_j\}}^\top + e_{\{S_j\}}e_{\{I_j\}}^\top - e_{\{I_j\}}e_{\{I_j\}}^\top) + \ell_j e_{\{S_j\}}e_{\{S_j\}}^\top\Big). \end{aligned}$$

Note that $\mathbb{P}[i \in I_j | i \in S_j] \leq \ell_j e^{2b}/\kappa_j$ for all $i \in S_j$. Its proof is similar to the proof of Lemma 2.18. Therefore, we have $\mathbb{E}[\mathrm{diag}(e_{\{I_j\}})] \preceq \ell_j e^{2b}/\kappa_j \mathrm{diag}(e_{\{\mathbf{1}\}})$. To bound $\|\sum_{j=1}^{n}\mathbb{E}[(M^{(j)})^2]\|$, we use the fact that for $J \in \mathbb{R}^{d\times d}, \|J\| \leq \max_{i\in[d]}\sum_{i'=1}^{d}|J_{ii'}|$. Maximum of row sums of $\mathbb{E}[e_{\{I_j\}}e_{\{I_j\}}^\top]$ is upper bounded by $\max_{i\in[d]}\big\{\ell_j\mathbb{P}[i \in I_j | i \in S_j]\big\} \leq {\ell_j}^2 e^{2b}/\kappa_j$. Therefore using triangle inequality, we have,

$$\begin{aligned}
& \Bigg\|\sum_{j=1}^{n}\mathbb{E}\big[(M^{(j)})^2\big]\Bigg\| \\
\leq\ & \max_{i\in[d]}\Bigg\{\sum_{j:i\in S_j}\frac{1}{(\kappa_j-1)^2}\Bigg(\frac{(\kappa_j^2-{\ell_j}^2)\ell_j e^{2b}}{\kappa_j}+{\ell_j}^2 \\
& +e^{2b}(\kappa_j+\ell_j)(2\ell_j+{\ell_j}^2/\kappa_j)+\ell_j\kappa_j\Bigg)\Bigg\} \\
\leq\ & \max_{i\in[d]}\Bigg\{\sum_{j:i\in S_j}\frac{\ell_j e^{2b}}{\kappa_j}\Bigg(\frac{(\kappa_j^2-{\ell_j}^2)}{(\kappa_j-1)^2}+\frac{\ell_j\kappa_j}{(\kappa_j-1)^2}+\frac{2(\kappa_j+\ell_j)\kappa_j}{(\kappa_j-1)^2} \\
& +\frac{(\kappa_j+\ell_j)\ell_j}{(\kappa_j-1)^2}+\frac{\kappa_j^2}{(\kappa_j-1)^2}\Bigg)\Bigg\} \\
\leq\ & \max_{i\in[d]}\Bigg\{\sum_{j:i\in S_j}\frac{\ell_j e^{2b}}{\kappa_j}\Bigg(\frac{(\kappa_j^2-1)}{(\kappa_j-1)^2}+\frac{\kappa_j(\kappa_j-1)}{(\kappa_j-1)^2}+\frac{4\kappa_j^2}{(\kappa_j-1)^2} \\
& +\frac{2\kappa_j(\kappa_j-1)}{(\kappa_j-1)^2}+\frac{\kappa_j^2}{(\kappa_j-1)^2}\Bigg)\Bigg\} \\
\leq\ & \max_{i\in[d]}\Bigg\{\sum_{j:i\in S_j}\frac{\ell_j e^{2b}}{\kappa_j}\Bigg(3+2+16+4+4\Bigg)\Bigg\} \qquad (2.109) \\
\leq\ & 29e^{2b}\max_{i\in[d]}\Bigg\{\sum_{j:i\in S_j}\frac{\ell_j}{\kappa_j}\Bigg\} \\
=\ & 29e^{2b}D_{\max} \qquad (2.110) \\
=\ & \frac{29e^{2b}}{\beta d}\sum_{j=1}^{n}\ell_j\ , \qquad (2.111)
\end{aligned}$$

where (2.109) uses the fact that $\kappa_j \geq 2$ and $1 \leq \ell_j \leq \kappa_j - 1$ for all $j \in [n]$. (2.110) follows from the definition of $D_{\max}$, Definition 2.1 and (2.111) follows from the Equation (2.6). Also, note that $\|M_j\| \leq 2$ for all $j \in [n]$. Applying matrix Bernstien inequality, we have,

$$\mathbb{P}\Big[\|M-\mathbb{E}[M]\|\geq t\Big]\leq d\ \exp\left(\frac{-t^2/2}{\frac{29e^{2b}}{\beta d}\sum_{j=1}^{n}\ell_j+4t/3}\right).$$

Therefore, with probability at least $1 - d^{-3}$, we have,

$$\|M - \mathbb{E}[M]\| \leq 22e^b \sqrt{\frac{\log d}{\beta d} \sum_{j=1}^{n} \ell_j + \frac{64 \log d}{3}} \leq 32e^b \sqrt{\frac{\log d}{\beta d} \sum_{j=1}^{n} \ell_j}\;, \quad (2.112)$$

where the second inequality follows from the assumption that $\sum_{j=1}^{n} \ell_j \geq 2^{12} d \log d$ and $\beta \leq 1$.

Proof of Lemma 2.18

Define $i_{\min} \equiv \arg\min_{i \in S} \theta_i$. We claim the following. For all $i \in S$ and any $1 \leq \ell \leq |S| - 1$,

$$\mathbb{P}[\sigma^{-1}(i) > \ell] \;\leq\; \mathbb{P}[\sigma^{-1}(i_{\min}) > \ell] \;\text{ and }\; \mathbb{P}[\sigma^{-1}(i_{\min}) = \ell] \;\geq\; \mathbb{P}[\sigma^{-1}(i_{\min}) = 1]\,. \quad (2.113)$$

Therefore $\mathbb{P}[\sigma^{-1}(i) \leq \ell] \;\geq\; \mathbb{P}[\sigma^{-1}(i_{\min}) \leq \ell]$. Using $\mathbb{P}[\sigma^{-1}(i_{\min}) = 1] > e^{-2b}/\kappa$, we get the desired bound $\mathbb{P}[\sigma^{-1}(i) \leq \ell] > e^{-2b}\ell/\kappa$.

To prove the claim (2.113), let $\widehat{\sigma}_1^{\ell}$ denote a ranking of top-ℓ items of the set S and $\mathbb{P}[\widehat{\sigma}_1^{\ell}]$ be the probability of observing $\widehat{\sigma}_1^{\ell}$. Let $i \in (\widehat{\sigma}_1^{\ell})^{-1}$ denote that $i = (\widehat{\sigma}_1^{\ell})^{-1}(j)$ for some $1 \leq j \leq \ell$. Let

$$\Omega_1 = \Big\{\widehat{\sigma}_1^{\ell} : i \notin (\widehat{\sigma}_1^{\ell})^{-1}, i_{\min} \in (\widehat{\sigma}_1^{\ell})^{-1}\Big\}\text{and}\Omega_2 = \Big\{\widehat{\sigma}_1^{\ell} : i \in (\widehat{\sigma}_1^{\ell})^{-1}, i_{\min} \notin (\widehat{\sigma}_1^{\ell})^{-1}\Big\}.$$

We have $\mathbb{P}[\sigma^{-1}(i) > \ell] - \mathbb{P}[\sigma^{-1}(i_{\min}) > \ell] = \sum_{\widehat{\sigma}_1^{\ell} \in \Omega_1} \mathbb{P}[\widehat{\sigma}_1^{\ell}] - \sum_{\widehat{\sigma}_1^{\ell} \in \Omega_2} \mathbb{P}[\widehat{\sigma}_1^{\ell}]$. Now, take any ranking $\widehat{\sigma}_1^{\ell} \in \Omega_1$ and construct another ranking $\widetilde{\sigma}_1^{\ell}$ from $\widehat{\sigma}_1^{\ell}$ by replacing $i_{\min}$ with i-th item. Observe that $\mathbb{P}[\widehat{\sigma}_1^{\ell}] \leq \mathbb{P}[\widetilde{\sigma}_1^{\ell}]$ and $\widetilde{\sigma}_1^{\ell} \in \Omega_2$. Moreover, such a construction gives a bijective mapping between Ω_1 and Ω_2. Hence, the first claim is proved. For the second claim, let

$$\widehat{\Omega}_1 = \Big\{\widehat{\sigma}_1^{\ell} : (\widehat{\sigma}_1^{\ell})^{-1}(i_{\min}) = 1\Big\} \;\text{ and }\; \widehat{\Omega}_2 = \Big\{\widehat{\sigma}_1^{\ell} : (\widehat{\sigma}_1^{\ell})^{-1}(i_{\min}) = \ell\Big\}.$$

We have $\mathbb{P}[\sigma^{-1}(i_{\min}) = 1] - \mathbb{P}[\sigma^{-1}(i_{\min}) = \ell] = \sum_{\widehat{\sigma}_1^{\ell} \in \widehat{\Omega}_1} \mathbb{P}[\widehat{\sigma}_1^{\ell}] - \sum_{\widehat{\sigma}_1^{\ell} \in \widehat{\Omega}_2} \mathbb{P}[\widehat{\sigma}_1^{\ell}]$. Now, take any ranking $\widehat{\sigma}_1^{\ell} \in \widehat{\Omega}_1$ and construct another ranking $\widetilde{\sigma}_1^{\ell}$ from $\widehat{\sigma}_1^{\ell}$ by swapping items at 1st position and ℓ-th position. Observe that $\mathbb{P}[\widehat{\sigma}_1^{\ell}] \leq \mathbb{P}[\widetilde{\sigma}_1^{\ell}]$ and $\widetilde{\sigma}_1^{\ell} \in \widehat{\Omega}_2$. Moreover, such a construction gives a bijective mapping between $\widehat{\Omega}_1$ and $\widehat{\Omega}_2$. Hence, the claim is proved.

2.7.5 Proof of Theorem 2.6

The first order partial derivative of $\mathcal{L}(\theta)$, Equation (2.15), is given by

$$\begin{aligned} &\nabla_i \mathcal{L}(\theta) \\ = \; &\sum_{j : i \in S_j} \sum_{m=1}^{\ell_j} \mathbb{I}_{\{\sigma_j^{-1}(i) \geq m\}} \Big[\mathbb{I}_{\{\sigma_j(m) = i\}} \\ &- \frac{\exp(\theta_i)}{\exp(\theta_{\sigma_j(m)}) + \exp(\theta_{\sigma_j(m+1)}) + \cdots + \exp(\theta_{\sigma_j(\kappa_j)})} \Big], \; \forall i \in [d] \end{aligned}$$

and the Hessian matrix $H(\theta) \in \mathcal{S}^d$ with $H_{ii'}(\theta) = \frac{\partial^2 \mathcal{L}(\theta)}{\partial \theta_i \partial \theta_{i'}}$ is given by

$$\begin{aligned} &H(\theta) = \\ &-\sum_{j=1}^{n} \sum_{i < i' \in S_j} (e_i - e_{i'})(e_i - e_{i'})^\top \\ &\quad \sum_{m=1}^{\ell_j} \frac{\exp(\theta_i + \theta_{i'}) \mathbb{I}_{\{\sigma_j^{-1}(i), \sigma_j^{-1}(i') \geq m\}}}{[\exp(\theta_{\sigma_j(m)}) + \exp(\theta_{\sigma_j(m+1)}) + \cdots + \exp(\theta_{\sigma_j(\kappa_j)})]^2} . \end{aligned} \tag{2.114}$$

It follows from the definition that $-H(\theta)$ is positive semi-definite for any $\theta \in \mathbb{R}^n$.

The Fisher information matrix is defined as $I(\theta) = -\mathbb{E}_\theta[H(\theta)]$ and given by

$$\begin{aligned} &I(\theta) = \\ &\sum_{j=1}^{n} \sum_{i < i' \in S_j} (e_i - e_{i'})(e_i - e_{i'})^\top \\ &\quad \sum_{m=1}^{\ell_j} \mathbb{E}\left[\frac{\mathbb{I}_{\{\sigma_j^{-1}(i), \sigma_j^{-1}(i') \geq m\}}}{[\exp(\theta_{\sigma_j(m)}) + \cdots + \exp(\theta_{\sigma_j(\kappa_j)})]^2} \right] \exp(\theta_i + \theta_{i'}). \end{aligned}$$

Since $-H(\theta)$ is positive semi-definite, it follows that $I(\theta)$ is positive semi-definite. Moreover, $\lambda_1(I(\theta))$ is zero and the corresponding eigenvector is the all-ones vector. Fix any unbiased estimator $\widehat{\theta}$ of $\theta \in \Omega_b$. Since, $\widehat{\theta} \in \mathcal{U}$, $\widehat{\theta} - \theta$ is orthogonal to $\mathbf{1}$. The Cramér-Rao lower bound then implies that

$\mathbb{E}[\|\widehat{\theta}-\theta^*\|^2] \geq \sum_{i=2}^{d} \frac{1}{\lambda_i(I(\theta))}$. Taking the supremum over both sides gives

$$\sup_{\theta} \mathbb{E}[\|\widehat{\theta}-\theta\|^2] \geq \sup_{\theta} \sum_{i=2}^{d} \frac{1}{\lambda_i(I(\theta))} \geq \sum_{i=2}^{d} \frac{1}{\lambda_i(I(\mathbf{0}))} \, .$$

If θ equals the all-zero vector, then

$$\mathbb{P}_\theta[\sigma_j^{-1}(i), \sigma_j^{-1}(i') \geq m] = \frac{\binom{\kappa_j - m+1}{2}}{\binom{\kappa_j}{2}} = \frac{(\kappa_j - m+1)(\kappa_j - m)}{\kappa_j(\kappa_j-1)}.$$

It follows from the definition that

$$\begin{aligned} I(0) &= \sum_{j=1}^{n} \sum_{i<i' \in S_j} (e_i - e_{i'})(e_i - e_{i'})^\top \sum_{m=1}^{\ell_j} \frac{(\kappa_j - m)}{\kappa_j(\kappa_j-1)(\kappa_j-m+1)} \\ &\preceq \ell\Big(1 - \frac{1}{\ell_j}\sum_{m=1}^{\ell_j} \frac{1}{\kappa_{\max} - m+1}\Big) \underbrace{\sum_{j=1}^{n} \frac{1}{\kappa_j(\kappa_j-1)} \sum_{i<i' \in S_j} (e_i - e_{i'})(e_i - e_{i'})^\top}_{=L} \, , \end{aligned}$$

where L is the Laplacian defined for the comparison graph $\mathcal{H}$, Definition 2.1. By Jensen's inequality, we have

$$\sum_{i=2}^{d} \frac{1}{\lambda_i(L)} \geq \frac{(d-1)^2}{\sum_{i=2}^{d} \lambda_i(L)} = \frac{(d-1)^2}{\mathrm{Tr}(L)} = \frac{(d-1)^2}{n}.$$

2.7.6 Proof of Theorem 2.7

We prove a slightly more general result that implies the desired theorem. For $\ell \geq 4$, we can choose $\beta_1 = 1/2$. Then, the condition that $\gamma_{\beta_1} \leq 1$ implies $\widetilde{d} \leq (\ell/2+1)(d-2)/(\kappa-2)$, which implies $\widetilde{d} \leq \ell d/(2\kappa)$. With the choice of $\widetilde{d} = \ell d/(2\kappa)$, this implies Theorem 2.7.

Theorem 2.19. *Under the bottom-ℓ separators scenario and the PL model, n partial orderings are sampled over d items parametrized by $\theta^* \in \Omega_b$. For any β_1 with $0 \leq \beta_1 \leq \frac{\ell-2}{\ell}$, define*

$$\gamma_{\beta_1} \equiv \frac{\widetilde{d}(\kappa-2)}{(\lfloor \ell\beta_1 \rfloor + 1)(d-2)}, \tag{2.115}$$

and for $\gamma_{\beta_1} \leq 1$,

$$\chi_{\beta_1} \;\equiv\; \big(1 - \lfloor \ell\beta_1 \rfloor / \ell\big)^2 \left(1 - \exp\left(- \frac{(\lfloor \ell\beta_1 \rfloor + 1)^2 (1-\gamma_{\beta_1})^2}{2(\kappa-2)} \right)\right) \tag{2.116}$$

If

$$n\ell \;\geq\; \left(\frac{2^{12} e^{8b}}{\chi_{\beta_1}^2} \frac{d^2}{\widetilde{d}^2} \frac{\kappa}{\ell} \right) d \log d \;, \tag{2.117}$$

then the rank-breaking *estimator in* (2.18) *achieves*

$$\frac{1}{\sqrt{\widetilde{d}}} \big\| \widehat{\widetilde{\theta}} - \widetilde{\theta}^* \big\|_2 \;\leq\; \frac{32\sqrt{2}(1+e^{4b})^2}{\chi_{\beta_1}} \frac{d^{3/2}}{\widetilde{d}^{3/2}} \sqrt{\frac{d \log d}{n\ell}} \;, \tag{2.118}$$

with probability at least $1 - 3e^3 d^{-3}$.

Proof is very similar to the proof of Theorem 7.3. It mainly differs in the lower bound that is achieved for the second smallest eigenvalue of the Hessian matrix $H(\widetilde{\theta})$ of $\mathcal{L}_{\rm RB}(\widetilde{\theta})$, Equation (2.17). Equation (2.17) can be rewritten as

$$\begin{aligned} &\mathcal{L}_{\rm RB}(\widetilde{\theta}) = \\ &\sum_{j=1}^{n} \sum_{a=1}^{\ell} \sum_{\substack{i<i' \in S_j \\ : i,i' \in [\widetilde{d}]}} \mathbb{I}_{\big\{(i,i') \in G_{j,a}\big\}} \lambda_{j,a} \Big(\widetilde{\theta}_i \mathbb{I}_{\big\{\sigma_j^{-1}(i) < \sigma_j^{-1}(i')\big\}} + \widetilde{\theta}_{i'} \mathbb{I}_{\big\{\sigma_j^{-1}(i) > \sigma_j^{-1}(i')\big\}} \\ &- \log\Big(e^{\widetilde{\theta}_i} + e^{\widetilde{\theta}_{i'}} \Big) \Big) \;, \end{aligned} \tag{2.119}$$

where $(i, i') \in G_{j,a}$ implies either (i, i') or (i', i) belong to $E_{j,a}$. The Hessian matrix $H(\widetilde{\theta}) \in \mathcal{S}^{\widetilde{d}}$ with $H_{ii'}(\widetilde{\theta}) = \frac{\partial^2 \mathcal{L}_{\rm RB}(\widetilde{\theta})}{\partial \widetilde{\theta}_i \partial \widetilde{\theta}_{i'}}$ is given by

$$\begin{aligned} &H(\widetilde{\theta}) = \\ &-\sum_{j=1}^{n} \sum_{a=1}^{\ell} \sum_{\substack{i<i' \in S_j: \\ i,i' \in [\widetilde{d}]}} \mathbb{I}_{\big\{(i,i') \in G_{j,a}\big\}} \left((\widetilde{e}_i - \widetilde{e}_{i'})(\widetilde{e}_i - \widetilde{e}_{i'})^\top \frac{\exp(\widetilde{\theta}_i + \widetilde{\theta}_{i'})}{[\exp(\widetilde{\theta}_i) + \exp(\widetilde{\theta}_{i'})]^2} \right). \end{aligned} \tag{2.120}$$

The following lemma gives a lower bound for $\lambda_2(-H(\widetilde{\theta}))$.

Lemma 2.20. *Under the hypothesis of Theorem 2.19, with probability at least* $1 - d^{-3}$,

$$\lambda_2(-H(\widetilde{\theta})) \geq \frac{\chi_{\beta_1}}{8(1+e^{4b})^2} \frac{n\widetilde{d}\ell^2}{d^2} \;. \tag{2.121}$$

Observe that although $\widetilde{\theta}^* \in \mathbb{R}^{\widetilde{d}}$, Lemma 2.10 can be directly applied to upper bound $\|\nabla \mathcal{L}_{\rm RB}(\widetilde{\theta}^*)\|_2$. It might be possible to tighten the upper bound, given that $\widetilde{d} \leq d$. However, for $\ell \ll \kappa$, for the smallest preference score item, $i_{\min} \equiv \arg\min_{i\in[d]} \widetilde{\theta}^*_i$, the upper bound $\mathbb{P}[\sigma^{-1}(i_{\min}) > \kappa - \ell] \leq 1$ is tight upto constant factor (Lemma 2.15). Substituting $\lambda_{j,a} = 1$ and $p_{j,a} = \kappa - \ell + a$ for each $j \in [n]$, $a \in [\ell]$, in Lemma 2.10, we have that with probability at least $1 - 2e^3 d^{-3}$,

$$\|\nabla \mathcal{L}_{\rm RB}(\widetilde{\theta}^*)\|_2 \;\; \leq \;\; (\ell - 1)\sqrt{8n\ell \log d}. \tag{2.122}$$

Theorem 2.19 follows from Equations (2.40), (2.121) and (2.122).

Proof of Lemma 2.20

Define $\widetilde{M}^{(j)} \in \mathcal{S}^{\widetilde{d}}$,

$$\widetilde{M}^{(j)} = \sum_{i<i' \in S_j : i,i' \in [\widetilde{d}]} \sum_{a=1}^{\ell} \mathbb{I}_{\{(i,i') \in G_{j,a}\}} (\widetilde{e}_i - \widetilde{e}_{i'})(\widetilde{e}_i - \widetilde{e}_{i'})^\top, \tag{2.123}$$

and let $\widetilde{M} = \sum_{j=1}^n \widetilde{M}^{(j)}$. Similar to the analysis in Lemma 2.11, we have $\lambda_2(-H(\widetilde{\theta})) \geq \frac{e^{4b}}{(1+e^{4b})^2} \lambda_2(\widetilde{M})$. Note that we have e^{4b} instead of e^{2b} as $\widetilde{\theta} \in \widetilde{\Omega}_{2b}$. We will show a lower bound on $\lambda_2(\mathbb{E}[\widetilde{M}])$ in (2.129) and an upper bound on $\|\widetilde{M} - \mathbb{E}[\widetilde{M}]\|$ in (2.133). Therefore using $\lambda_2(\widetilde{M}) \geq \lambda_2(\mathbb{E}[\widetilde{M}]) - \|\widetilde{M} - \mathbb{E}[\widetilde{M}]\|$,

$$\lambda_2(\widetilde{M}) \;\geq\; \frac{e^{-4b}}{4} \underbrace{(1-\beta_1)^2 \left(1 - \exp\left(-\frac{(\lfloor \ell\beta_1 \rfloor + 1)^2 (1-\gamma_{\beta_1})^2}{2(\kappa - 2)}\right)\right)}_{\equiv \chi_{\beta_1}} \frac{n\widetilde{d}\ell^2}{d^2}$$

$$- \, 8\ell \sqrt{\frac{n\kappa \log d}{d}} \;. \tag{2.124}$$

The desired claim follows from the assumption that $n\ell \geq \big(\frac{2^{12}e^{8b}}{\chi_{\beta_1}^2}\frac{d^2}{\widetilde{d}^2}\frac{\kappa}{\ell}\big)d\log d$, where χ_{β_1} is defined in (2.117). To prove the lower bound on $\lambda_2(\mathbb{E}[\widetilde{M}])$, notice that

$$\begin{aligned}
&\mathbb{E}\big[\widetilde{M}\big] \\
&= \sum_{j=1}^{n}\sum_{i<i'\in[\widetilde{d}]}\mathbb{E}\Bigg[\sum_{a=1}^{\ell}\mathbb{I}_{\big\{(i,i')\in G_{j,a}\big\}}\Big|(i,i'\in S_j)\Bigg]\mathbb{P}\Big[i,i'\in S_j\Big](\widetilde{e}_i-\widetilde{e}_{i'})(\widetilde{e}_i-\widetilde{e}_{i'})^\top .
\end{aligned} \tag{2.125}$$

Since the sets S_j are chosen uniformly at random, $\mathbb{P}[i,i'\in S_j] = \kappa(\kappa-1)/d(d-1)$. Using the fact that $p_{j,a} = \kappa-\ell+a$ for each $j\in[n]$, and the definition of rank breaking graph $G_{j,a}$, we have that

$$\mathbb{E}\Bigg[\sum_{a=1}^{\ell}\mathbb{I}_{\big\{(i,i')\in G_{j,a}\big\}}\Big|(i,i'\in S_j)\Bigg] = \mathbb{P}\Big[\big(\sigma_j^{-1}(i),\sigma_j^{-1}(i') > \kappa-\ell\big)\Big|(i,i'\in S_j)\Big] . \tag{2.126}$$

The following lemma provides a lower bound on $\mathbb{P}[(\sigma_j^{-1}(i),\sigma_j^{-1}(i')) > \kappa-\ell|(i,i'\in S_j)]$.

Lemma 2.21. *Under the hypotheses of Theorem 2.19, for any two items $i,i'\in[\widetilde{d}]$, the following holds:*

$$\begin{aligned}
&\mathbb{P}\Big[\sigma^{-1}(i),\sigma^{-1}(i') > \kappa-\ell \,\Big|\, i,i'\in S\Big] \\
&\geq \frac{e^{-4b}(1-\beta_1)^2(1-\exp(-\eta_{\beta_1}(1-\gamma_{\beta_1})^2))}{2}\frac{\ell^2}{\kappa^2} ,
\end{aligned} \tag{2.127}$$

where $\gamma_{\beta_1} \equiv \widetilde{d}(\kappa-2)/(\lfloor\ell\beta_1\rfloor+1)(d-2)$ and $\eta_{\beta_1} \equiv (\lfloor\ell\beta_1\rfloor+1)^2/2(\kappa-2)$.

Therefore, using Equations (2.125), (2.126) and (2.127) we have,

$$\begin{aligned}
\mathbb{E}\big[\widetilde{M}\big] &\succeq \frac{e^{-4b}(1-\beta_1)^2(1-\exp(-\eta_{\beta_1}(1-\gamma_{\beta_1})^2))}{2}\frac{\ell^2}{\kappa^2}\frac{\kappa(\kappa-1)}{d(d-1)} \\
&\sum_{j=1}^{n}\sum_{i<i'\in[\widetilde{d}]}(\widetilde{e}_i-\widetilde{e}_{i'})(\widetilde{e}_i-\widetilde{e}_{i'})^\top .
\end{aligned} \tag{2.128}$$

Define $\widetilde{L} = \sum_{j=1}^{n}\sum_{i<i'\in[\widetilde{d}]}(\widetilde{e}_i-\widetilde{e}_{i'})(\widetilde{e}_i-\widetilde{e}_{i'})^\top$. We have, $\lambda_1(\widetilde{L}) = 0$ and $\lambda_2(\widetilde{L}) = \lambda_3(\widetilde{L}) = \cdots = \lambda_{\widetilde{d}}(\widetilde{L})$. Therefore, using $\lambda_2(\widetilde{L}) = \mathrm{Tr}(\widetilde{L})/(\widetilde{d}-1) = n\widetilde{d}$.

Using the fact that $\mathbb{E}[\widetilde{M}]$ and $\widetilde{L}$ are symmetric matrices, we have,

$$\lambda_2(\mathbb{E}\big[\widetilde{M}\big]) \geq \frac{e^{-4b}(1-\beta_1)^2(1-\exp(-\eta_{\beta_1}(1-\gamma_{\beta_1})^2))}{4}\frac{n\widetilde{d}\ell^2}{d^2}. \tag{2.129}$$

To get an upper bound on $\|\widetilde{M}-\mathbb{E}[\widetilde{M}]\|$, notice that $\widetilde{M}^{(j)}$ is also given by,

$$\widetilde{M}^{(j)} \;\; = \;\; \ell\,\mathrm{diag}(\widetilde{e}_{\{I_j\}}) - \widetilde{e}_{\{I_j\}}\widetilde{e}_{\{I_j\}}^\top \;, \tag{2.130}$$

where $\widetilde{e}_{\{I_j\}} \in \mathbb{R}^{\widetilde{d}}$ is a zero-one vector, with support corresponding to the bottom-ℓ subset of items in the ranking σ_j. $I_j = \{\sigma_j(\kappa-\ell+1), \cdots, \sigma_j(\kappa)\}$ for $j \in [n]$. $(\widetilde{M}^{(j)})^2$ is given by

$$(\widetilde{M}^{(j)})^2 \;\; = \;\; \ell^2\,\mathrm{diag}(\widetilde{e}_{\{I_j\}}) - \ell\,\widetilde{e}_{\{I_j\}}\widetilde{e}_{\{I_j\}}^\top \,. \tag{2.131}$$

Using the fact that sets $\{S_j\}_{j\in[n]}$ are chosen uniformly at random and $\mathbb{P}[i \in \mathbb{I}_j | i \in S_j] \leq 1$, we have $\mathbb{E}[\mathrm{diag}(\widetilde{e}_{\{I_j\}})] \preceq (\kappa/d)\mathrm{diag}(\widetilde{e}_{\{\mathbf{1}\}})$. Maximum of row sums of $\mathbb{E}\big[\widetilde{e}_{\{I_j\}}\widetilde{e}_{\{I_j\}}^\top\big]$ is upper bounded by $\ell\kappa/d$. Therefore, from triangle inequality we have $\|\sum_{j=1}^n \mathbb{E}[(\widetilde{M}^{(j)})^2]\| \leq 2n\ell^2\kappa/d$. Also, note that $\|\widetilde{M}^{(j)}\| \leq 2\ell$ for all $j \in [n]$. Applying matrix Bernstien inequality, we have that

$$\mathbb{P}\Big[\|\widetilde{M}-\mathbb{E}[\widetilde{M}]\| \geq t\Big] \leq d\,\exp\Big(\frac{-t^2/2}{2n\ell^2\kappa/d + 4\ell t/3}\Big). \tag{2.132}$$

Therefore, with probability at least $1-d^{-3}$, we have,

$$\|\widetilde{M}-\mathbb{E}[\widetilde{M}]\| \leq 4\ell\sqrt{\frac{2n\kappa\log d}{d}} + \frac{64\ell\log d}{3} \leq 8\ell\sqrt{\frac{n\kappa\log d}{d}}\;, \tag{2.133}$$

where the second inequality follows from the assumption that $n\ell \geq 2^{12} d\log d$.

Proof of Lemma 2.21

Without loss of generality, assume that $i' < i$, i.e., $\widetilde{\theta}_{i'}^* \leq \widetilde{\theta}_i^*$. Define Ω such that $\Omega = \{j : j \in S, j \neq i, i'\}$. For any $\beta_1 \in [0, (\ell-2)/\ell]$, define event E_{β_1} that occurs if in the randomly chosen set S there are at most $\lfloor \ell\beta_1 \rfloor$ items that have preference scores less than $\widetilde{\theta}_i^*$, i.e.,

$$E_{\beta_1} \;\equiv\; \Big\{ \textstyle\sum_{j\in\Omega} \mathbb{I}_{\{\widetilde{\theta}_i^* > \widetilde{\theta}_j^*\}} \leq \lfloor \ell\beta_1 \rfloor \Big\}\,. \tag{2.134}$$

We have,

$$\begin{aligned}
&\mathbb{P}\Big[\sigma^{-1}(i),\sigma^{-1}(i') \;>\; \kappa-\ell \;\Big|\; i,i'\in S\Big] \\
>\;& \mathbb{P}\Big[\sigma^{-1}(i),\sigma^{-1}(i') > \kappa-\ell \;\Big|\; i,i'\in S; E_{\beta_1}\Big]\mathbb{P}\Big[E_{\beta_1} \;\Big|\; i,i'\in S\Big] \qquad (2.135)
\end{aligned}$$

The following lemma provides a lower bound on $\mathbb{P}[\sigma^{-1}(i),\sigma^{-1}(i') > \kappa - \ell \mid i,i'\in S; E_{\beta_1}]$.

Lemma 2.22. *Under the hypotheses of Lemma 2.21,*

$$\mathbb{P}\Big[\sigma^{-1}(i),\sigma^{-1}(i') \;>\; \kappa-\ell \;\Big|\; i,i'\in S; E_{\beta_1}\Big] \;\geq\; \frac{e^{-4b}(1-\lfloor \ell\beta_1\rfloor/\ell)^2}{2}\frac{\ell^2}{\kappa^2}\;. \qquad (2.136)$$

Next, we provide a lower bound on $\mathbb{P}[E_{\beta_1} \mid i,i'\in S]$. Fix i,i' such that $i,i'\in S$. Selecting a set uniformly at random is probabilistically equivalent to selecting items one at a time uniformly at random without replacement. Without loss of generality, assume that i,i' are the 1st and 2nd pick. Define Bernoulli random variables $Y_{j'}$ for $3\leq j'\leq\kappa$ corresponding to the outcome of the j'-th random pick from the set of $(d-j'-1)$ items to generate the set Ω such that $Y_{j'}=1$ if and only if $\widetilde{\theta}^*_i > \widetilde{\theta}^*_{j'}$.

Recall that $\gamma_{\beta_1}\equiv \widetilde{d}(\kappa-2)/(\lfloor\ell\beta_1\rfloor+1)(d-2)$ and $\eta_{\beta_1}\equiv(\lfloor\ell\beta_1\rfloor+1)^2/2(\kappa-2)$. Construct Doob's martingale $(Z_2,\cdots,Z_\kappa)$ from $\{Y_{k'}\}_{3\leq k'\leq\kappa}$ such that $Z_{j'}=\mathbb{E}[\sum_{k'=3}^{\kappa}Y_{k'}\mid Y_3,\cdots,Y_{j'}]$, for $2\leq j'\leq\kappa$. Observe that, $Z_2=\mathbb{E}[\sum_{k'=3}^{\kappa}Y_{k'}]\leq\frac{(i-2)(\kappa-2)}{d-2}\leq\gamma_{\beta_1}(\lfloor\ell\beta_1\rfloor+1)$, where the last inequality follows from the assumption that $i\leq\widetilde{d}$. Also, $|Z_{j'}-Z_{j'-1}|\leq 1$ for each j'. Therefore, we have

$$\begin{aligned}
\mathbb{P}\Big[\textstyle\sum_{j\in\Omega}\mathbb{I}_{\{\widetilde{\theta}^*_i>\widetilde{\theta}^*_j\}}\leq\lfloor\ell\beta_1\rfloor\Big] &= \mathbb{P}\Big[\textstyle\sum_{j'=3}^{\kappa}Y_{j'}\leq\lfloor\ell\beta_1\rfloor\Big] \\
&= 1-\mathbb{P}\Big[\textstyle\sum_{j'=3}^{\kappa}Y_{j'}\geq\lfloor\ell\beta_1\rfloor+1\Big] \\
&\geq 1-\mathbb{P}\Big[Z_{\kappa-2}-Z_2\geq(\ell\beta_1+1)-\gamma(\lfloor\ell\beta_1\rfloor+1)\Big] \\
&\geq 1-\exp\Big(-\frac{(\lfloor\ell\beta_1\rfloor+1)^2(1-\gamma_1)^2}{2(\kappa-2)}\Big) \\
&= 1-\exp\Big(-\eta_{\beta_1}(1-\gamma_{\beta_1})^2\Big), \qquad (2.137)
\end{aligned}$$

where the inequality follows from the Azuma-Hoeffding bound. Since, the above inequality is true for any fixed $i,i'\in S$, for random indices i,i' we have $\mathbb{P}[E_{\beta_1} \mid i, i' \in S] \geq 1 - \exp(-\eta_{\beta_1}(1-\gamma_{\beta_1})^2)$. Claim (2.127) follows by combining Equations (2.135), (2.136) and (2.137).

Proof of Lemma 2.22

Without loss of generality, assume that $i' < i$, i.e., $\widetilde{\theta}^*_{i'} \leq \widetilde{\theta}^*_i$. Define $\Omega = \{j : j \in S, j \neq i, i'\}$, and event $E_{\beta_1} = \{i, i' \in S; \sum_{j\in\Omega} \mathbb{I}_{\{\widetilde{\theta}^*_i > \widetilde{\theta}^*_j\}} \leq \lfloor \ell\beta_1 \rfloor\}$. Since set S is chosen randomly, i, i' and $j \in \Omega$ are random. Throughout this section, we condition on the random indices i, i' and the set Ω such that event E_{β_1} holds. To get a lower bound on $\mathbb{P}[\sigma^{-1}(i), \sigma^{-1}(i') > \kappa - \ell]$, define independent exponential random variables $X_j \sim \exp(e^{\widetilde{\theta}^*_j})$ for $j \in S$. Observe that given event E_{β_1} holds, there exists a set $\Omega_1 \subseteq \Omega$ such that

$$\Omega_1 = \left\{ j \in S : \widetilde{\theta}^*_i \leq \widetilde{\theta}^*_j \right\}, \tag{2.138}$$

and $|\Omega_1| = \kappa - \lfloor \ell\beta_1 \rfloor - 2$. In fact there can be many such sets, and for the purpose of the proof we can choose one such set arbitrarily. Note that $\lfloor \ell\beta_1 \rfloor + 2 \leq \ell$ by assumption on β_1, so $|\Omega_1| \geq \kappa - \ell$. From the Random Utility Model (RUM) interpretation of the PL model, we know that the PL model is equivalent to ordering the items as per *random cost* of each item drawn from exponential random variable with mean $e^{\tilde{\theta}^*_i}$. That is, we rank items according to X_j's such that the lower cost items are ranked higher. From this interpretation, we have that

$$\begin{aligned} \mathbb{P}\Big[\sigma^{-1}(i), \sigma^{-1}(i') > \kappa - \ell\Big] &= \mathbb{P}\Big[\sum_{j\in\Omega} \mathbb{I}_{\Big\{\min\{X_i, X_{i'}\} > X_j\Big\}} \geq \kappa - \ell\Big] \\ &> \mathbb{P}\Big[\sum_{j'\in\Omega_1} \mathbb{I}_{\Big\{\min\{X_i, X_{i'}\} > X_{j'}\Big\}} \geq \kappa - \ell\Big] \end{aligned} \tag{2.139}$$

The above inequality follows from the fact that $\Omega_1 \subseteq \Omega$ and $|\Omega_1| \geq \kappa - \ell$. It excludes some of the rankings over the items of the set S that constitute the event $\{\sigma^{-1}(i), \sigma^{-1}(i') > \kappa - \ell\}$. Define $\Omega_2 = \{\Omega_1, i, i'\}$. Observe that items i, i' have the least preference scores among all the items in the set Ω_2. Therefore, the term in Equation (2.139) is the probability of the least two preference score items in the set Ω_2, that is of size $(\kappa - \lfloor \ell\beta_1 \rfloor)$, being ranked in bottom $(\ell - \lfloor \ell\beta_1 \rfloor)$ positions.

The following lemma shows that the probability of the least two preference score items in a set being ranked at any two positions is lower bounded by their probability of being ranked at 1st and 2nd position.

Lemma 2.23. *Consider a set of items S and a ranking σ over it. Define $i_{\min_1} \equiv \arg\min_{i \in S} \theta_i$, $i_{\min_2} \equiv \arg\min_{i \in S \setminus i_{min_1}} \theta_i$. For all $1 \le i_1, i_2 \le |S|$, $i_1 \neq i_2$, following holds:*

$$\mathbb{P}\Big[\sigma^{-1}(i_{\min_1}) = i_1, \sigma^{-1}(i_{\min_2}) = i_2\Big] \ge \mathbb{P}\Big[\sigma^{-1}(i_{\min_1}) = 1, \sigma^{-1}(i_{\min_2}) = 2\Big]. \tag{2.140}$$

Using the fact that $i' = \arg\min_{j \in \Omega_2} \widetilde{\theta}^*_j$, $i = \arg\min_{j \in \Omega_2 \setminus i'} \widetilde{\theta}^*_j$, for all $1 \le i_1, i_2 \le \kappa - \lfloor \ell\beta_1 \rfloor$, $i_1 \neq i_2$, we have that

$$\mathbb{P}\Big[\sigma^{-1}(i') = i_1, \sigma^{-1}(i) = i_2\Big] \ge \mathbb{P}\Big[\sigma^{-1}(i') = 1, \sigma^{-1}(i) = 2\Big] \ge e^{-4b}\frac{1}{\kappa^2}\ , \tag{2.141}$$

where the second inequality follows from the definition of the PL model and the fact that $\widetilde{\theta}^* \in \widetilde{\Omega}_{2b}$. Together with Equation (2.141) and the fact that there are a total of $(\ell - \lfloor \ell\beta \rfloor)(\ell - \lfloor \ell\beta \rfloor - 1) \ge (\ell - \lfloor \ell\beta \rfloor)^2/2$ pair of positions that i, i' can occupy in order to being ranked in bottom $(\ell - \lfloor \ell\beta \rfloor)$, we have,

$$\mathbb{P}\Big[\sigma^{-1}(i), \sigma^{-1}(i') > \kappa - \ell\Big] \ge \frac{e^{-4b}(1 - \lfloor \ell\beta_1 \rfloor / \ell)^2}{2}\frac{\ell^2}{\kappa^2}. \tag{2.142}$$

Since, the above inequality is true for any fixed i, i' and $j \in \Omega$ such that event E holds, it is true for random indices i, i' and $j \in \Omega$ such that event E holds, hence the claim is proved.

Proof of Lemma 2.23

Let $\widehat{\sigma}$ denote a ranking over the items of the set S and $\mathbb{P}[\widehat{\sigma}]$ be the probability of observing $\widehat{\sigma}$. Let

$$\widehat{\Omega}_1 = \Big\{\widehat{\sigma} : \widehat{\sigma}^{-1}(i_{\min_1}) = i_1, \widehat{\sigma}^{-1}(i_{\min_2}) = i_2\Big\} \text{ and}$$
$$\widehat{\Omega}_2 = \Big\{\widehat{\sigma} : \sigma^{-1}(i_{\min_1}) = 1, \sigma^{-1}(i_{\min_2}) = 2\Big\}. \tag{2.143}$$

Now, take any ranking $\widehat{\sigma} \in \widehat{\Omega}_1$ and construct another ranking $\widetilde{\sigma}$ from $\widehat{\sigma}$ as following. If $i_1 = 2, i_2 = 1$, then swap the items at i_1-th and i_2-th position in ranking $\widehat{\sigma}$ to get $\widetilde{\sigma}$. Else, if $i_1 < i_2$, then first: swap items at i_1-th position and 1st position, and second: swap items at i_2-th position and 2nd position, to get $\widetilde{\sigma}$; if $i_2 < i_1$, then first: swap items at i_2-th position and 2nd position, and second: swap items at i_1-th position and 1st position, to get $\widetilde{\sigma}$.

Observe that $\mathbb{P}[\widetilde{\sigma}] \leq \mathbb{P}[\widehat{\sigma}]$ and $\widetilde{\sigma}_1^\ell \in \widehat{\Omega}_2$. Moreover, such a construction gives a bijective mapping between $\widehat{\Omega}_1$ and $\widehat{\Omega}_2$. Hence, the claim is proved.

CHAPTER 3

COMPUTATIONAL AND STATISTICAL TRADEOFFS IN RANK AGGREGATION

In classical statistical inference, we are typically interested in characterizing how more data points improve the accuracy, with little restrictions or considerations on computational aspects of solving the inference problem. However, with massive growths of the amount of data available and also the complexity and heterogeneity of the collected data, computational resources, such as time and memory, are major bottlenecks in many modern applications. As a solution, recent advances in learning theory introduce hierarchies of algorithmic solutions, ordered by the respective computational complexity, for several fundamental machine learning applications in [25, 187, 34, 2, 139]. Guided by sharp analyses on the sample complexity, these approaches provide theoretically sound guidelines that allow the analyst the flexibility to fall back to simpler algorithms to enjoy the full merit of the improved run-time.

Inspired by these advances, we study the time-data tradeoff in rank aggregation. In many applications such as election, policy making, polling, and recommendation systems, we want to aggregate individual preferences to produce a global ranking that best represents the collective social preference. We assume that the data comes from a parametric family of choice models, and learns the parameters that determine the global ranking. Traditionally, each revealed preference is assumed to have one of the following three structures. *Pairwise comparison*, where one item is preferred over another, is common in sports and chess matches. *Best-out-of-κ comparison*, where one is chosen among a set of κ alternatives, is common in historical purchase data. *κ-way comparison*, where we observe a linear ordering of a set of κ candidates, is used in some elections and surveys. We will refer to such structures as *traditional* in comparisons to modern datasets with non-traditional structures whose behavior change drastically. For such traditional preferences, efficient schemes for rank aggregation have been proposed, such as [72, 92, 84, 39], which we explain in detail in Section 3.2. However, modern datasets are unstructured and heterogeneous. As [114] show, this can lead to significant increase in the computational complexity, requiring exponential run-time in the size of the problem in the worst case.

To alleviate this computational challenge, we propose a hierarchy of estimators which we call *generalized rank-breaking*, ordered in increasing computational complexity and achieving increasing accuracy. The key idea is to break down the heterogeneous revealed preferences into simpler pieces of ordinal relations, and apply an estimator tailored for those simple structures treating each piece as independent. Several aspects of rank-breaking makes this problem interesting and challenging. A priori, it is not clear which choices of the simple ordinal relations are rich enough to be statistically efficient and yet lead to tractable estimators. Even if we identify which ordinal relations to extract, the ignored correlations among those pieces can lead to an inconsistent estimate, unless we choose carefully which pieces to include and which to omit in the estimation. We further want sharp analysis on the sample complexity, which reveals how computational and statistical efficiencies trade off. We would like to address all these challenges in providing generalized rank-breaking methods.

3.1 Problem formulation.

We study the problem of aggregating ordinal data based on users' preferences that are expressed in the form of *partially ordered sets (poset)*. A poset is a collection of ordinal relations among items. For example, consider a poset $\{(i_6 \prec \{i_5, i_4\}), (i_5 \prec i_3), (\{i_3, i_4\} \prec \{i_1, i_2\})\}$ over items $\{i_1, \dots, i_6\}$, where $(i_6 \prec \{i_5, i_4\})$ indicates that item i_5 and i_4 are both preferred over item i_6. Such a relation is extracted from, for example, the user giving a 2-star rating to i_5 and i_4 and a 1-star to i_6.

We assume there are n users and d items. We denote the set of n users by $[n] = \{1, \dots, n\}$ and the set of d items by $[d]$. We assume that each user $j \in [n]$ is presented with a subset of items $S_j \subseteq [d]$, and independently provides her ordinal preference in the form of a poset, where the ordering is drawn from the Plackett-Luce (PL) model. Since, an ordering drawn from the PL model is consistent, a poset can be represented as a directed acyclic graph (DAG). Let $\mathcal{G}_j$ denote the DAG representation of the poset provided by the user j over $S_j \subseteq [d]$ according to the PL model with weights θ^*. The task is to learn $\widehat{\theta}$, an estimate of the true weights θ^*. Below is an example of a DAG $\mathcal{G}_j$. We use index i to denote items and j to denote users.

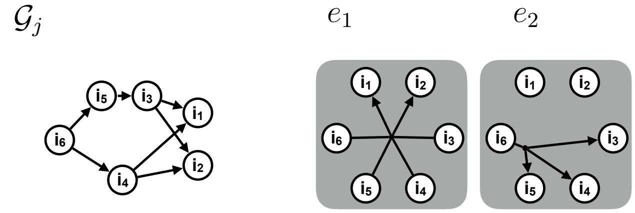


Figure 3.1: An example of $\mathcal{G}_j$ for user j's consistent poset, and two rank-breaking hyper edges extracted from it: $e_1 = (\{i_6, i_5, i_4, i_3\} \prec \{i_2, i_1\})$ and $e_2 = (\{i_6\} \prec \{i_5, i_4, i_3\})$.

Plackett-Luce model. The PL model is a popular choice model from operations research and psychology, used to model how people make choices under uncertainty. It is a special case of *random utility models*, where each item i is parametrized by a latent true utility $\theta_i \in \mathbb{R}$. When offered with S_j, the user samples the perceived utility U_i for each item independently according to $U_i = \theta_i + Z_i$, where Z_i's are i.i.d. noise. In particular, the PL model assumes Z_i's follow the standard Gumbel distribution. The observed poset is a partial observation of the ordering according to this perceived utilities. We discuss possible extensions to general class of random utility models in Section 3.2.

The particular choice of the Gumbel distribution has several merits, largely stemming from the fact that the Gumbel distribution has a log-concave pdf and is inherently memoryless. In our analyses, we use the log-concavity to show that our proposed algorithm is a concave maximization (Remark 3.1) and the memoryless property forms the basis of our rank-breaking idea. Precisely, the PL model is statistically equivalent to the following procedure. Consider a ranking as a mapping from a position in the rank to an item, i.e. $\sigma_j : [|S_j|] \to S_j$. It can be shown that the PL model is generated by first independently assigning each item $i \in S_j$ an unobserved value Y_i, exponentially distributed with mean $e^{-\theta_i}$, and the resulting ranking σ_j is inversely ordered in Y_i's so that $Y_{\sigma_j(1)} \leq Y_{\sigma_j(2)} \leq \cdots \leq Y_{\sigma_j(|S_j|)}$.

This inherits the memoryless property of exponential variables, such that $\mathbb{P}(Y_1 < Y_2 < Y_3) = \mathbb{P}(Y_1 < \{Y_2, Y_3\})\mathbb{P}(Y_2 < Y_3)$, leading to a simple interpretation of the PL model as sequential choices:

$$\begin{aligned}\mathbb{P}_\theta(i_3 \prec i_2 \prec i_1) &= \mathbb{P}_\theta(\{i_3, i_2\} \prec i_1)\mathbb{P}_\theta(i_3 \prec i_2) \\ &= \frac{e^{\theta_{i_1}}}{e^{\theta_{i_1}} + e^{\theta_{i_2}} + e^{\theta_{i_3}}} \times \frac{e^{\theta_{i_2}}}{e^{\theta_{i_2}} + e^{\theta_{i_3}}} \; . \end{aligned} \tag{3.1}$$

In general, for true utility θ^*, we have

$$\mathbb{P}_{\theta^*}[\sigma_j] \;=\; \prod_{i=1}^{|S_j|-1} \frac{e^{\theta^*_{\sigma_j(i)}}}{\sum_{i'=i}^{|S_j|} e^{\theta^*_{\sigma_j(i')}}} \, .$$

We assume that the true utility $\theta^* \in \Omega_b$ where

$$\Omega_b \;=\; \left\{ \theta \in \mathbb{R}^d \,\middle|\, \sum_{i\in[d]} \theta_i = 0, |\theta_i| \le b \text{ for all } i \in [d] \right\} . \tag{3.2}$$

Notice that centering of θ ensures its uniqueness as PL model is invariant under shifting of θ. The bound b on θ_i is written explicitly to capture the dependence in our main results. We interchangeably refer θ as utilities and weights.

Maximum Likelihood Estimate of DAG. Probability of observing a DAG $\mathcal{G}_j$ is the sum of probabilities of all possible rankings that are consistent with it. Precisely, under the PL model, for a DAG $\mathcal{G}_j$, we have,

$$\mathbb{P}_\theta[\mathcal{G}_j] \;=\; \sum_{\sigma\in\mathcal{G}_j} \mathbb{P}_\theta[\sigma] \, ,$$

where we slightly abuse the notation $\mathcal{G}_j$ to denote the set of all rankings σ that are consistent with the observation. For example, if $\mathcal{G}_j$ consists of only one hyper edge $e_1 = (\{i_3\} \prec \{i_2, i_1\})$ then $\mathbb{P}[\mathcal{G}_j] = \mathbb{P}(i_3 \prec i_2 \prec i_1) + \mathbb{P}(i_3 \prec i_1 \prec i_2)$. The maximum likelihood estimate (MLE) maximizes log-likelihood of observing $\mathcal{G}_j$ for each j:

$$\widehat{\theta} \;\in\; \arg\max_{\theta\in\Omega_b} \left\{ \sum_{j=1}^{n} \log \mathbb{P}_\theta[\mathcal{G}_j] \right\} . \tag{3.3}$$

When $\mathcal{G}_j$ has a *traditional* structure as explained earlier in this section, then the optimization is a simple multinomial logit regression, that can be solved efficiently with off-the-shelf convex optimization tools. [84] provides full analysis of the statistical complexity of this MLE under traditional structures. For general posets, it can be shown that the above optimization is a concave maximization, using similar techniques as Remark 3.1. However, the summation over rankings in $\mathcal{G}_j$ can involve number of terms super exponential in the size $|S_j|$, in the worst case. This renders MLE intractable and impractical.

Pairwise rank-breaking. A common remedy to this computational blow-up is to use rank-breaking. Rank-breaking traditionally refers to *pairwise rank-breaking*, where a bag of all the pairwise comparisons is extracted from observations $\{\mathcal{G}_j\}_{j\in[n]}$ and is applied to estimators that are tailored for pairwise comparisons, treating each paired outcome as independent. This is one of the motivations behind the algorithmic advances in the popular topic of aggregation from pairwise comparisons in [72, 92, 157, 182, 146].

It is computationally efficient to apply maximum likelihood estimator assuming independent pairwise comparisons, which takes $O(d^2)$ operations to evaluate. However, this computational gain comes at the cost of statistical efficiency. [13] showed that if we include all paired comparisons, then the resulting estimate can be statistically inconsistent due to the ignored correlations among the paired orderings, even with infinite samples. In the example from Figure 3.1, there are 12 paired relations implied by the DAG: $(i_6 \prec i_5), (i_6 \prec i_4), (i_6 \prec i_3), \ldots, (i_3 \prec i_1), (i_4 \prec i_1)$. In order to get a consistent estimate, [13] provide a rule for choosing which pairs to include, and [114] provide an estimator that optimizes how to weigh each of those chosen pairs to get the best finite sample complexity bound. However, such a consistent pairwise rank-breaking results in throwing away many of the ordered relations, resulting in significant loss in accuracy. For example, including any paired relation from $\mathcal{G}_j$ in the example results in a biased estimator. None of the pairwise orderings can be used from $\mathcal{G}_j$, without making the estimator inconsistent as shown in [12]. Whether we include all paired comparisons or only a subset of consistent ones, there is a significant loss in accuracy as illustrated in Figure 4.1. For the precise condition for consistent rank-breaking we refer to [12, 13, 114].

The state-of-the-art approaches operate on either one of the two extreme points on the computational and statistical trade-off. The MLE in (3.3) requires $O(\sum_{j\in[n]} |S_j|!)$ summations to just evaluate the objective function, in the worst case. On the other hand, the pairwise rank-breaking requires only

$O(d^2)$ summations, but suffers from significant loss in the sample complexity. Ideally, we would like to give the analyst the flexibility to choose a target computational complexity she is willing to tolerate, and provide an algorithm that achieves the optimal trade-off at the chosen operating point.

Contribution. We introduce a novel *generalized rank-breaking* that bridges the gap between MLE and pairwise rank-breaking. Our approach allows the user the freedom to choose the level of computational resources to be used, and provides an estimator tailored for the desired complexity. We prove that the proposed estimator is tractable and consistent, and provide an upper bound and a lower bound on the error rate in the finite sample regime. The analysis explicitly characterizes the dependence on the topology of the data. This in turn provides a guideline for designing surveys and experiments in practice, in order to maximize the sample efficiency. The proposed generalized rank-breaking mechanism involves set-wise comparisons as opposed to traditional pairwise comparisons. In order to compute the rank-breaking estimate, we generalize the celebrated minorization maximization algorithm for computing maximum likelihood estimate of pairwise comparisons [92] to more general set-wise comparisons and give guarantees on its convergence.

3.2 Related work

In classical statistics, one is interested in the tradeoff between the sample size and the accuracy, with little considerations to the computational complexity or time. As more computations are typically required with increasing availability of data, the computational resources are often the bottleneck. Recently, a novel idea known as “algorithmic weakening” has been investigated to overcome such a bottleneck, in which a hierarchy of algorithms is proposed to allow for faster algorithms at the expense of decreased accuracy. When guided by sound theoretical analyses, this idea allows the statistician to achieve the same level of accuracy and *save* time when more data is available. This is radically different from classical setting where processing more data typically requires more computational time.

Depending on the application, several algorithmic weakenings have been studied. In the application of supervised learning, [25] proposed the idea that weaker approximate optimization algorithms are sufficient for learning when more data is available. Various gradient based algorithms are analyzed that show the time-accuracy-sample tradeoff. In a similar context, [187] analyze a particular implementation of support vector machine and show that the target accuracy can be achieved faster when more data is available, by running the iterative algorithm for shorter amount of time. In the application of denoising, [34] provide a hierarchy of convex relaxations where constraints are defined by convex geometry with increasing complexity. For unsupervised learning, [139] introduce a hierarchy of data representations that provide more representative elements when more data is available at no additional computation. Standard clustering algorithms can be applied to thus generated summary of the data, requiring less computational complexity.

In the application of rank aggregation, we follow the principle of algorithmic weakening and propose a novel rank-breaking to allow the practitioner to navigate gracefully the time-sample trade off as shown in the Figure 4.2. We propose a hierarchy of estimators indexed by $M \in \mathbb{Z}^+$ indicating how complex the estimator is (defined formally in Section 3.3). Figure 4.2 shows the result of a experiment on synthetic datasets on how much time (in seconds) and how many samples are required to achieve a target accuracy. If we are given more samples, then it is possible to achieve the target accuracy, which in this example is MSE$\leq 0.3d^2 \times 10^{-6}$, with fewer operations by using a simpler estimator with smaller M. The details of the experiment is explained in Figure 4.1.

Rank aggregation under the PL model has been studied extensively under the *traditional* scenario dating back to [214] who first introduced the PL model for pairwise comparisons. Various approaches for estimating the PL weights from traditional samples have been proposed. The problem can be formulated as a convex optimization that can be solved efficiently using the off-the-shelf solvers. However, tailored algorithms for finding the optimal solution have been proposed in [72] and [92], which iteratively finds the fixed point of the KKT condition. [157] introduce Rank Centrality, a novel spectral ranking algorithm which formulates a random walk from the given data, and show that the stationary distribution provides accurate estimates of the PL weights. [146] provide a connection between those previous approaches, and give a unified random walk approach that finds the fixed point of the KKT conditions.

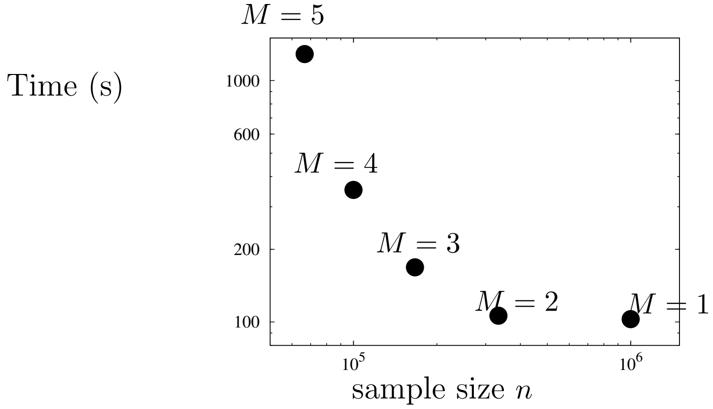


Figure 3.2: Depending on how much computational resources are available, the various choices of M achieve different operating points on the time-data trade-off to achieve some fixed target accuracy $\varepsilon > 0$. If more samples are available, one can resort to faster methods with smaller M while achieving the same level of accuracy.

On the theoretical side, when samples consist of pairwise comparisons, [192] first established consistency and asymptotic normality of the maximum likelihood estimate when all teams play against each other. For a broader class of scenarios where we allow for sparse observations, where the number of total comparisons grow linearly in the number of teams, [157] show that Rank Centrality achieves optimal sample complexity by comparing it to a lower bound on the minimax rate. For a more general class of traditional observations, including pairwise comparisons, [84] provide similar optimal guarantee for the maximum likelihood estimator. [39] introduced Spectral MLE that applies Rank Centrality followed by MLE, and showed that the resulting estimate is optimal in L_∞ error as well as the previously analyzed L_2 error. [182] study a new measure of the error induced by the Laplacian of the comparisons graph and prove a sharper upper and lower bounds that match up to a constant factor.

However, in modern applications, the computational complexity of the existing approaches blow-up due to the heterogeneity of modern datasets. Although, statistical and computational tradeoffs have been investigated under other popular choice models such as the Mallows models by [19] or stochastically transitive models by [184], the algorithmic solutions do not apply to random utility models and the analysis techniques do not extend. We provide a novel rank-breaking algorithms and provide finite sample complexity analysis under the PL model. This approach readily generalizes to some RUMs such as the flipped Gumbel distribution. However, it is also known from [13], that for general RUMs there is no consistent rank-breaking, and the proposed approach does not generalize.

3.3 Generalized rank-breaking

Given $\mathcal{G}_j$'s representing the users' preferences, *generalized rank-breaking* extracts a set of ordered relations and applies an estimator treating each ordered relation as independent. Concretely, for each $\mathcal{G}_j$, we first extract a maximal ordered partition $\mathcal{P}_j$ of S_j that is consistent with $\mathcal{G}_j$. An ordered partition is a partition with a linear ordering among the subsets, e.g. $\mathcal{P}_j = (\{i_6\} \prec \{i_5, i_4, i_3\} \prec \{i_2, i_1\})$ for $\mathcal{G}_j$ from Figure 3.1. This is maximal, since we cannot further partition any of the subsets without creating artificial ordered relations that are not present in the original $\mathcal{G}_j$.

To precisely define maximal ordered partition $\mathcal{P}_j$, first, let's define an ordered partition $\widetilde{\mathcal{P}}_j$ of S_j that is consistent with $\mathcal{G}_j$. Consider disjoint subsets $\mathcal{C}_1, \cdots, \mathcal{C}_{\ell_j} \subseteq S_j$ such that their union is S_j that is $\cup_{a=1}^{\ell_j} C_a = S_j$. The subsets $\mathcal{C}_1, \cdots, \mathcal{C}_{\ell_j}$ define an ordered partition

$$\widetilde{\mathcal{P}}_j = \mathcal{C}_1 \prec \mathcal{C}_2 \prec \cdots \prec \mathcal{C}_{\ell_j} ,$$

if each ordered relation that can be read from this linear ordering of subsets is present in the DAG $\mathcal{G}_j$. Let $|\mathcal{P}_j|$ denote the size of the partition, that is $|\mathcal{P}_j| = \ell_j$. A maximal ordered partition $\mathcal{P}_j$ is the one which has the largest size.

$$\mathcal{P}_j = \arg\max_{\widetilde{\mathcal{P}}_j} \left\{ |\widetilde{\mathcal{P}}_j| \right\} .$$

It can be checked that for a given $\mathcal{G}_j$ there is a unique maximal ordered partition $\mathcal{P}_j$ of S_j that is consistent with $\mathcal{G}_j$.

Finding Maximal Ordered Partition. Given a DAG $\mathcal{G}_j$, a corresponding maximal ordered partition $\mathcal{P}_j$ can be extracted by recursively finding common ancestors of the sink-nodes of the vertex induced sub-graph starting with the DAG $\mathcal{G}_j$. Algorithm 1 gives a pseudocode to find $\mathcal{P}_j$'s. Common ancestors of all the sink nodes of a DAG can be found in time $O(d^{2.6})$ using fast algorithms given in [42, 16]. Therefore, computational complexity of the Algorithm 1 is $O(d^{3.6})$. In line 2, Algorithm 1, $V(\mathcal{G})$ denotes the set of vertices of DAG $\mathcal{G}$. In line 5, $\mathcal{G}(S)$ denote the vertex induced subgraph of graph $\mathcal{G}$ corresponding to vertex set S. Note that Algorithm 1 returns a unique maximal ordered partition $\mathcal{P}_j$ for a given DAG $\mathcal{G}_j$.

Algorithm 1 Finding Maximal Ordered Partition

```
Require: DAG $\mathcal{G}_j$
Ensure: maximal ordered partition $\mathcal{P}_j$
1: $\mathcal{G} \leftarrow \mathcal{G}_j$, $\mathcal{P}_j = \{\}$
2: while $|V(\mathcal{G})| > 0$ do
3:     $S \leftarrow$ Common ancestors of all sink-nodes of DAG $\mathcal{G}$ [42]
4:     $\mathcal{P}_j \leftarrow \mathcal{P}_j \succ \{V(\mathcal{G}) \setminus S\}$
5:     $\mathcal{G} \leftarrow \mathcal{G}(S)$
6: end while
```

In general there is no one-to-one mapping from a DAG $\mathcal{G}_j$ to its maximal ordered partition $\mathcal{P}_j$. There may be many ordered relations present in $\mathcal{G}_j$ that are not represented in the ordered partition $\mathcal{P}_j$. This gives a many-to-one mapping from $\mathcal{G}_j$ to $\mathcal{P}_j$. In our generalized rank-breaking framework, we can only use those ordered relations that can be represented in an ordered partition. This is required for the estimator to be consistent. This is the cost we pay to reduce computational complexity from $O(|S_j|!)$, complexity of MLE of DAG (3.3), to $O(M!)$ for a suitably desired $M \in \mathbb{Z}^+$ as explained below. However, if the DAG $\mathcal{G}_j$ represents a full ranking or a traditional structure then its maximal ordered partition $\mathcal{P}_j$ will represent all the ordered relations present in $\mathcal{G}_j$ and our rank-breaking will reduce to MLE of the DAG $\mathcal{G}_j$. In such a case, all the subsets of $\mathcal{P}_j$ will have cardinality one except the least preferred set which can have more than one item in case of best-out-of κ comparison.

Rank-Breaking Graph. The extracted maximal ordered partition $\mathcal{P}_j$ is represented by a directed hypergraph $G_j(S_j, E_j)$, which we call a *rank-breaking graph.* Each edge $e = (B(e), T(e)) \in E_j$ is a directed hyper edge from a subset of nodes $B(e) \subseteq S_j$ to another subset $T(e) \subseteq S_j$. The number of edges in E_j is $|\mathcal{P}_j| - 1$. For each subset in $\mathcal{P}_j$ except for the least preferred subset, there is a corresponding edge whose *top-set* $T(e)$ is the subset, and the *bottom-set* $B(e)$ is the set of all items less preferred than $T(e)$. For the example in Figure 3.1, we have $E_j = \{e_1, e_2\}$ where $e_1 = (B(e_1), T(e_1)) = (\{i_6, i_5, i_4, i_3\}, \{i_2, i_1\})$ and $e_2 = (B(e_2), T(e_2) = (\{i_6\}, \{i_5, i_4, i_3\})$ extracted from $\mathcal{G}_j$. Algorithm 2 gives the precise method to construct a rank-breaking graph.

Algorithm 2 Constructing Rank-Breaking Graph

```
Require: maximal ordered partition $\mathcal{P}_j = \mathcal{C}_1 \prec \mathcal{C}_2 \prec \cdots \prec \mathcal{C}_{\ell_j}$ of set $S_j$
Ensure: directed hypergraph $G_j(S_j, E_j)$
construct directed hypergraph $G_j(S_j, E_j = \{\})$
for $a = 2$ to $\ell_j$ do
    construct hyper edge $e$ between top-set $T(e) = \mathcal{C}_a$ and bottom-set $B(e) = \cup_{a'=1}^{a-1} \mathcal{C}_{a'}$
    $E_j \leftarrow E_j \cup e$
end for
Return $G_j(S_j, E_j)$
```

Denote the probability that $T(e)$ is preferred over $B(e)$ when $T(e) \cup B(e)$ is offered as

$$\begin{aligned} \mathbb{P}_\theta(e) &= \mathbb{P}_\theta\big(B(e) \prec T(e)\big) \\ &= \sum_{\sigma \in \Lambda_{T(e)}} \frac{\exp\left(\sum_{c=1}^{|T(e)|} \theta_{\sigma(c)}\right)}{\prod_{u=1}^{|T(e)|}\left(\sum_{c'=u}^{|T(e)|} \exp\left(\theta_{\sigma(c')}\right) + \sum_{i \in B(e)} \exp\left(\theta_i\right)\right)}\,, \end{aligned} \quad (3.4)$$

which follows from the definition of the PL model, where $\Lambda_{T(e)}$ is the set of all rankings over $T(e)$. The computational complexity of evaluating this probability is determined by the size of the *top-set* $|T(e)|$, as it involves $(|T(e)|!)$ summations.

In the subsequent results, we show that by maximizing likelihood of the hyper edges assuming they are independent we get a consistent estimator. Therefore, our approach provides flexibility to choose which hyper edge to include in the likelihood maximization function. We let the analyst choose the order $M \in \mathbb{Z}^+$ depending on how much computational resource is available, and include only those edges with $|T(e)| \leq M$ in the likelihood objective function.

If in a given G_j there are no hyper edges with top sets of size less than M, then the analyst does not get any ordered relations from that rank-breaking graph under her computational constraint reflected in the particular choice of M. Artificially reducing the size of the top-sets so as to get the hyper edges with top sets of size less than M implies we need to add new ordered relations that are not present in the DAG provided by the user. Such an estimator could result in a non-zero bias. A concrete example of such cases has been studied in Azari Soufiani et al. 2013, where the authors showed that for $M = 1$, applying rank-breaking to those comparisons with top-set larger than one results in non-zero bias.

We emphasize that M is chosen by the analyst and our estimator works for any choice of $M \in \mathbb{Z}^+$. Given unlimited computational resources, an analyst would chose $M = d$, and all the hyper edges would be included in the likelihood objective function.

Sampling. We assume that for each $j \in [n]$, the topology of DAG $\mathcal{G}_j$ which represent the partial preference order provided by the j-th user is fixed apriori. Also, the set of hyper edges $e \in E_j$ of each rank breaking graph $G_j(S_j, E_j)$ that are included in the likelihood objective function are fixed apriori. The randomness that we observe is in the position of the S_j items in the DAG $\mathcal{G}_j$. For an hyper edge $e \in E_j$, the randomness is in which items of the set S_j appear in the bottom $|B(e)|$ positions and the bottom $|T(e)| + |B(e)|$ positions in the preference order of the user j. Note that this precisely captures the randomness due to the PL model in the observed DAG $\mathcal{G}_j$. We do not impose any restrictions on the topology of the DAG $\mathcal{G}_j$'s and each of them can be different. Further, our analysis captures effect of their topologies on the statistical efficiency of the estimation.

Pseudo-MLE of Rank-Breaking Graph. We apply the MLE for comparisons over paired subsets, assuming all hyper edges in the rank-breaking graph G_j are independently drawn. Precisely, for any choice of $M \in \mathbb{Z}^+$, we propose *order-M rank-breaking estimate*, which is the solution that maximizes the log-likelihood under the independence assumption:

$$\begin{aligned} \widehat{\theta} &\in \arg\max_{\theta \in \Omega_b} \mathcal{L}_{\mathrm{RB}}(\theta)\,, \text{ where} \\ \mathcal{L}_{\mathrm{RB}}(\theta) &= \sum_{j \in [n]} \sum_{e \in E_j : |T(e)| \le M} \ln \mathbb{P}_\theta(e)\,. \end{aligned} \tag{3.5}$$

Due to independence assumption, we refer to it as pseudo-MLE. In a special case when $M = 1$, this can be transformed into the traditional pairwise rank-breaking, where (*i*) this is a concave maximization; (*ii*) the estimate is (asymptotically) unbiased and consistent as shown in [12, 13]; and (*iii*) and the finite sample complexity have been analyzed in [114]. Although, this order-1 rank-breaking provides a significant gain in computational efficiency, the information contained in higher-order edges are unused, resulting in a significant loss in accuracy.

We provide the analyst the freedom to choose the computational complexity he/she is willing to tolerate. However, for general M, it has not been known if the optimization in (7.12) is tractable and/or if the solution is consistent. Since $\mathbb{P}_\theta(B(e) \prec T(e))$ as explicitly written in (3.4) is a sum of log-concave functions, it is not clear if the sum is also log-concave. Due to the ignored dependency in the formulation (7.12), it is not clear if the resulting estimate is consistent.

We first establish that it is a concave maximization in Section 3.3.1. Though one can use any off-the-shelf convex maximization tool to compute $\widehat{\theta}$, we provide an efficient minorization-maximization (MM) algorithm for estimating $\widehat{\theta}$, Section 3.3.2. In Section 3.3.3, we show that the MM algorithm converges to the unique global optimal solution $\widehat{\theta}$ under the standard assumption given by [72] for pairwise comparisons. Under the same assumption, we show that the estimate $\widehat{\theta}$ is consistent, Section 3.3.4. In Section 3.3.5, we give the complete algorithm to compute $\widehat{\theta}$ using the proposed MM algorithm, given $\mathcal{G}_j$'s representing users' preferences. In Section 7.4 and Section 3.5, we provide a sharp analysis of the performance in the finite sample regime, characterizing the trade-off between computation and sample size, and verify the results from the numerical experiments.

3.3.1 Concavity of likelihood of rank-breaking graph

In the following, we show that likelihood of a hyper edge is log-concave for a family of Random Utility Models including the PL model.

Remark 3.1. *$\mathcal{L}_{\mathrm{RB}}(\theta)$ is concave in $\theta \in \mathbb{R}^d$.*

Proof. Recall that $\mathbb{P}_\theta(B(e) \prec T(e))$ is the probability that an agent ranks the collection of items $T(e)$ above $B(e)$ when offered $S = B(e) \cup T(e)$. We want to show that $\mathbb{P}_\theta(B(e) \prec T(e))$ is log-concave under the PL model. We prove a slightly general result which works for a family of RUMs in the location family. RUM are defined as a probabilistic model where there is a real-valued utility parameter θ_i associated with each items $i \in S$, and an agent independently samples random utilities $\{U_i\}_{i \in S}$ for each item i with conditional distribution $\mu_i(\cdot|\theta_i)$. Then the ranking is obtained by sorting the items in decreasing order as per the observed random utilities U_i's. *Location family* is a subset of RUMs where the shapes of μ_i's are fixed and the only parameters are the means of the distributions. For location family, the noisy utilities can be written as $U_i = \theta_i + Z_i$ for i.i.d. random variable Z_i's. In particular, it is PL model when Z_i's follow the independent standard Gumbel distribution. We will show that for the location family if the probability density function for each Z_i's is log-concave then $\log \mathbb{P}_\theta(B(e) \prec T(e))$ is concave. The desired claim follows as the pdf of standard Gumbel distribution is log-concave. We use the following Theorem from [168]. A similar technique was used to prove concavity when $|T(e)| = 1$ in [14].

Lemma 3.2 (Extension of Theorem 9 in [168])**.** *Suppose $g_1(\theta, Y), \cdots, g_r(\theta, Y)$ are concave functions in $\mathbb{R}^{2q}$, where $\theta, Y \in \mathbb{R}^q$, and Z is a $q-$component random vector whose probability distribution is logarithmic concave in $\mathbb{R}^q$, then the function*

$$h(\theta) = \mathbb{P}[g_1(\theta, Z) \geq 0, \cdots, g_r(\theta, Z) \geq 0], \qquad \textit{for } \theta \in \mathbb{R}^q$$

is logarithmic concave on $\mathbb{R}^q$. Moreover, concavity is strict if the probability density function of Z is strictly logarithmic concave and $\theta \neq \tilde{\theta}$ implies $H(\theta) \neq H(\tilde{\theta})$. Where $H(\theta)$ is

$$H(\theta) \;\; \equiv \;\; \big\{ Y \;\big|\; g_i(\theta, Y) \geq 0, \quad i = 1, \cdots, r \big\} \; .$$

Proof. Theorem 9 in [168] proves concavity. The strict concavity follows from the fact that for a strictly logarithmic concave measure the following inequality is strict if $H(\theta) \neq H(\tilde{\theta})$.

$$\mathbb{P}[Z \in \lambda H(\theta) + (1-\lambda) H(\tilde{\theta})] \;\; \geq \;\; \mathbb{P}[Z \in \lambda H(\theta)]^\lambda \mathbb{P}[Z \in (1-\lambda) H(\tilde{\theta})]^{1-\lambda} \, ,$$

where $\lambda \in (0, 1)$. For a detailed proof, we refer the reader to the proof of Theorem 9 in [168]. □

To apply the above lemma to get concavity, let $q = |S|$, $r = 1$, $g_1(\theta, Y) = \min_{i \in T(e)}\{\theta_i + Y_i\} - \max_{i' \in B(e)}\{\theta_{i'} + Y_{i'}\}$. Observe that $g_1(\theta, Y)$ is concave in $\mathbb{R}^{2q}$, and $\mathbb{P}_\theta(B(e) \prec T(e)) = \mathbb{P}(g_1(\theta, Z) \geq 0)$. We use strict concavity part of the lemma in the subsequent section. □

3.3.2 Minorization-maximization algorithm for pseudo-MLE of rank-breaking graph

We give a minorization-maximization algorithm for computing $\widehat{\theta}$ defined in (3.5). It is inspired from the MM algorithm given by [92] for the case of pairwise comparisons and full-ranking. For any fixed parameter $\theta^{(t)} \in \mathbb{R}^d$, and a hyper edge e in a rank breaking graph G, define $Q(e, \theta; \theta^{(t)})$ as

$$Q(e,\theta;\theta^{(t)}) \equiv$$

$$\sum_{\sigma \in \Lambda_{T(e)}} \left(\frac{\mathbb{P}_{\theta^{(t)}}(e,\sigma)}{\mathbb{P}_{\theta^{(t)}}(e)} \sum_{u=1}^{|T(e)|} \left(\theta_{\sigma(u)} - \frac{\sum_{c'=u}^{|T(e)|} \exp\left(\theta_{\sigma(c')}\right) + \sum_{i \in B(e)} \exp\left(\theta_i\right)}{\sum_{c'=u}^{|T(e)|} \exp\left(\theta^{(t)}_{\sigma(c')}\right) + \sum_{i \in B(e)} \exp\left(\theta^{(t)}_i\right)} \right) \right)$$

where $\mathbb{P}_\theta(e,\sigma)$ is defined such that $\mathbb{P}_\theta(e) = \sum_{\sigma \in \Lambda_{T(e)}} \mathbb{P}_\theta(e,\sigma)$. Recall from Equation (3.4) that $\Lambda_{T(e)}$ is the set of all rankings over $T(e)$.

$$\mathbb{P}_\theta(e,\sigma) \;\; \equiv \;\; \frac{\exp\left(\sum_{c=1}^{|T(e)|} \theta_{\sigma(c)}\right)}{\prod_{u=1}^{|T(e)|} \left(\sum_{c'=u}^{|T(e)|} \exp\left(\theta_{\sigma(c')}\right) + \sum_{i \in B(e)} \exp\left(\theta_i\right)\right)} .$$

We show that $Q(e, \theta; \theta^{(t)})$ minorizes $\ln(\mathbb{P}_\theta(e))$ at $\theta^{(t)}$. It is equal to $\ln(\mathbb{P}_\theta(e))$, up to a constant, if and only if $\theta^{(t)} = \theta$.

Lemma 3.3.

$$Q(e,\theta;\theta^{(t)}) + f(e,\theta^{(t)}) \;\; \leq \;\; \ln(\mathbb{P}_\theta(e)) \quad \textit{with equality if } \theta = \theta^{(t)} ,$$

where $f(e, \theta^{(t)})$ is a function of the hyper edge e and the parameter $\theta^{(t)}$, it does not depend upon θ.

We give a proof of the Lemma in Section 3.6.1. It follows that for any $Q(e, \theta; \theta^{(t)})$ satisfying minorizing condition in the above lemma,

$$Q(e,\theta;\theta^{(t)}) \;\geq\; Q(e,\theta^{(t)};\theta^{(t)}) \quad \text{implies} \;\; \ln(\mathbb{P}_\theta(e)) \;\geq\; \ln(\mathbb{P}_{\theta^{(t)}}(e)) . \qquad (3.6)$$

Property (3.6) suggests an iterative algorithm in which we let $\theta^{(t)}$ be the parameter vector before the t-th iteration and define $\theta^{(t+1)}$ to be the maximizer of the $Q(e, \theta; \theta^{(t)})$. Since this algorithm consists of alternately creating a minorizing function $Q(e, \theta; \theta^{(t)})$ and then maximizing it, it is called an MM algorithm [93]. To compute $\widehat{\theta}$ in (3.5), starting from an arbitrary initialization $\theta^{(1)}$, we estimate $\theta^{(t+1)}$ by maximizing

$$\theta^{(t+1)} \quad = \quad \arg\max_{\theta \in \mathbb{R}^d} \left\{ \sum_{j=1}^{n} \sum_{e \in E_j : |T(e)| \leq M} Q(e, \theta; \theta^{(t)}) \right\} .$$

Since the parameters $\{\theta_i\}_{i \in [d]}$ are separated in $Q(e, \theta; \theta^{(t)})$, its maximization can be explicitly accomplished as, for $i \in [d]$

$$\begin{aligned} & \frac{N_i}{e^{\theta_i^{(t+1)}}} = \\ & \sum_{j=1}^{n} \sum_{e \in E_j : |T(e)| \leq M} \sum_{\sigma \in \Lambda_{T(e)}} \frac{\mathbb{P}_{\theta^{(t)}}(e, \sigma)}{\mathbb{P}_{\theta^{(t)}}(e)} \sum_{u=1}^{|T(e)|} \delta_{i,e,\sigma,u} \\ & \left(\sum_{c'=u}^{|T(e)|} \exp\left(\theta_{\sigma(c')}^{(t)}\right) + \sum_{i \in B(e)} \exp\left(\theta_i^{(t)}\right) \right)^{-1} , \end{aligned} \tag{3.7}$$

where N_i is the total number of hyper edges in which the i-th item is in the top set.

$$N_i \quad = \quad \sum_{j=1}^{n} \sum_{e \in E_j : |T(e)| \leq M} \mathbb{I}[i \in T(e)] .$$

$\delta_{i,e,\sigma,u}$ is the indicator variable defined as

$$\delta_{i,e,\sigma,u} = \begin{cases} 1 , & \text{if } i \in \{T(e) \cup B(e)\} \text{ and } \sigma^{-1}(i) \geq u , \\ 0 , & \text{otherwise.} \end{cases}$$

3.3.3 Convergence properties of the MM algorithm

In the following we show that $\lim_{t\to\infty} \theta^{(t)}$, (3.7), converges to the global optimal solution of the pseudo-likelihood objective given in (3.5) under standard assumption on the observed comparisons.

For pairwise comparisons, [72] noted that if it is possible to partition the set of items into two subsets A and B such that there are never any inter-set comparisons, then there is no basis for rating any item in set A with respect to any item in set B. On the other hand, if in all the inter-set comparisons, items from set A are preferred over the items in set B, then if all the parameters θ_i belonging to set A are doubled and the resulting vector θ renormalized, the likelihood must increase; thus the likelihood has no maximizer. The following assumption [72] eliminates these possibilities.

Assumption 3.4. *In every possible partition of the items into two nonempty subsets, some item in the second set is preferred over some item in the first set at least once.*

Assumption 3.4 has a graph-theoretic interpretation: if the items are the nodes of a graph and the directed edge (i, j) denotes that there is at least one user who prefers i over j, then Assumption 3.4 is equivalent to the statement that there is a path from i to j for all nodes i and j. It implies that there exists a unique maximizer of the log-likelihood function of pairwise comparisons.

In our setting, Assumption 3.4 makes sense if we interpret i being preferred over j to mean that there exists a hyper edge e such that i is in its top set $T(e)$ and j is in its bottom set $B(e)$. In the following, we show that under this assumption, the MM algorithm, Equation (3.7), produces a sequence $\theta^{(1)}, \theta^{(2)}, \cdots$ guaranteed to converge to the unique estimate, (3.5).

In general, it is always not possible to prove that the sequence of parameters $\theta^{(t)}$ defined by an MM algorithm converges at all. Nonetheless, it is often possible to obtain convergence results in specific cases. For pairwise comparisons, using property of stationary point, [72] showed that the MM algorithm converges to the unique maximum likelihood estimate under Assumption 3.4. [92] established strict concavity of the likelihood function under Assumption 3.4 and proved the same result using the Liapounov's theorem. We follow the approach used by [92]. The following Liapounov's theorem guarantees that the MM algorithm converges to the stationary point of the pseudo log-likelihood objective (3.5). In Remark 3.6, we show that the likelihood function (3.5) has a unique stationary point, namely the global maximizer. Which concludes that the MM algorithm converges to the unique global optimal solution irrespective of its starting point.

Theorem 3.5 (Liapounov's theorem)**.** *Suppose $M : \Omega \to \Omega$ is continuous and $\mathcal{L}_{\mathrm{RB}} : \Omega \to \mathbb{R}$ is differentiable and for all $\theta \in \Omega$ we have $\mathcal{L}_{\mathrm{RB}}(M(\theta)) \geq \mathcal{L}_{\mathrm{RB}}(\theta)$, with equality only if θ is a stationary point of $\mathcal{L}_{\mathrm{RB}}(\cdot)$. Then, for arbitrary $\theta^{(1)} \in \Omega$, any limit point of the sequence $\{\theta^{(t+1)} = M(\theta^{(t)})\}_{t\geq 1}$ is a stationary point of $\mathcal{L}_{\mathrm{RB}}(\theta)$.*

Let $\Omega = \{\theta \in \mathbb{R}^d | \sum_{i\in[d]} \theta_i = 0\}$ be the parameter space Ω_b defined in (3.2) with $b = \infty$. Taking M to be the map implicitly defined by one iteration of the MM algorithm, we have $\mathcal{L}_{\mathrm{RB}}(M(\theta)) \geq \mathcal{L}_{\mathrm{RB}}(\theta)$ from (3.6). $\mathcal{L}_{\mathrm{RB}}(M(\theta)) = \mathcal{L}_{\mathrm{RB}}(\theta)$ implies that θ is a stationary point follows from the fact that the differentiable minorizing function Q is a tangent to the log-likelihood $\mathcal{L}_{\mathrm{RB}}(\theta)$ at the current iterate $\theta^{(t)}$. Therefore, $\lim_{t\to\infty} \theta^{(t)}$, defined by the MM algorithm in (3.7) converges to the stationary point of the pseudo-likelihood objective (3.5). It remains to prove that $\mathcal{L}_{\mathrm{RB}}(\theta)$ has a unique stationary point, the global maximizer.

Remark 3.6. *Under Assumption 3.4 $\mathcal{L}_{\mathrm{RB}}(\theta)$ has a unique stationary point.*

Proof. First, we show that $\mathcal{L}_{\mathrm{RB}}(\theta)$ is an upper compact function under the Assumption 3.4. $\mathcal{L}_{\mathrm{RB}}(\theta)$ is defined to be upper compact if, for any constant c, the set $\{\theta \in \Omega : \mathcal{L}_{\mathrm{RB}}(\theta) \geq c\}$ is a compact set of the parameter space Ω. Second, we show that $\mathcal{L}_{\mathrm{RB}}(\theta)$ is strictly concave. Since upper compactness implies the existence of at least one limit point and strict concavity implies the existence of at most one stationary point, we conclude that $\mathcal{L}_{\mathrm{RB}}(\theta)$ has a unique stationary point.

We prove upper compactness following the idea of [92]. Consider what happens to $\mathcal{L}_{\mathrm{RB}}(\theta)$ when θ approaches the boundary of Ω. If $\tilde{\theta}$ lies on the boundary of Ω, then $\tilde{\theta}_i = -\infty$ and $\tilde{\theta}_j = \infty$ for some items i and j. If items are nodes of a directed graph in which edge (i, i') represent that there is at least one user who prefers i over i', then Assumption 3.4 implies that a directed path exists from i to j. Therefore, there must be some item a with $\tilde{\theta}_a = -\infty$ which is preferred over item b with $\tilde{\theta}_b > C$, for some constant C.

That is there exists an hyper edge e with $a \in T(e)$ and $b \in B(e)$. Which means that for $\theta \in \Omega$, taking limits in

$$\begin{aligned}\mathcal{L}_{\mathrm{RB}}(\theta) &\leq \ln \mathbb{P}_\theta(e) \\ &= \ln \left(\sum_{\sigma \in \Lambda_{T(e)}} \frac{\exp\left(\sum_{c=1}^{|T(e)|} \theta_{\sigma(c)}\right)}{\prod_{u=1}^{|T(e)|}\left(\sum_{c'=u}^{|T(e)|} \exp\left(\theta_{\sigma(c')}\right) + \sum_{i \in B(e)} \exp\left(\theta_i\right)\right)} \right)\end{aligned}$$

gives $\lim_{\theta \to \tilde{\theta}} \mathcal{L}_{\mathrm{RB}}(\theta) = -\infty$. Thus, for any given constant c, the set $\{\theta \in \Omega : \mathcal{L}_{\mathrm{RB}}(\theta) \geq c\}$ is a closed and bounded set, and hence a compact set.

To prove strict concavity, we use Lemma 3.2. Define $\widetilde{\Omega} = \{\theta \in \mathbb{R}^d | \theta_1 = 0\}$, a reparameterization of the set $\Omega = \{\theta \in \mathbb{R}^d | \sum_{i \in [d]} \theta_i = 0\}$. To apply Lemma 3.2 to prove strict concavity of log-likelihood of an hyper edge e, take $g_{ij}(\theta, Y) = (\theta_i + Y_i) - (\theta_j + Y_j)$, for all $i \in T(e)$ and $j \in B(e)$. Consider $\theta, \tilde{\theta} \in \widetilde{\Omega}$. $H(\theta) = H(\tilde{\theta})$ implies that $\theta_i - \theta_j = \tilde{\theta}_i - \tilde{\theta}_j$, for all $i \in T(e)$ and $j \in B(e)$. This follows from the fact that for a fixed parameter θ, the hyper planes $\{g_{ij}(\theta, Y) \geq 0\}_{ij}$ are linearly independent. Thus, Assumption 3.4 combined with the fact that $\theta_1 = \tilde{\theta}_1$ means that $\theta = \tilde{\theta}$. Since the Gumbel distribution has strictly logarithmic concave density function, we conclude that $\mathcal{L}_{\mathrm{RB}}(\theta)$ is strictly concave. □

3.3.4 Consistency of pseudo-MLE of rank-breaking graph

In order to discuss consistency of the proposed approach, we need to specify how we sample the set of items to be offered S_j and also which partial ordering over S_j is to be observed. Here, we consider a simple but canonical scenario for sampling ordered relations, and show the proposed method is consistent for all non-degenerate cases. However, we study a more general sampling scenario, when we analyze the order-M estimator in the finite sample regime in Section 7.4.

Following is the canonical sampling scenario. There is a set of ℓ integers $(m_1, \ldots, m_\ell)$ whose sum is strictly less than d. A new arriving user is presented with all d items and is asked to provide her top m_1 items as an unordered set, and then the next m_2 items, and so on. This is sampling from the PL model and observing an ordered partition with $(\ell + 1)$ subsets of sizes m_a's, and the last subset includes all remaining items. We apply the generalized rank-breaking to get rank-breaking graphs $\{G_j\}$ with ℓ edges each, and order-M estimate is computed. We show that this is consistent, i.e. asymptotically unbiased in the limit of the number of users n.

Remark 3.7. *Under the* PL *model and the above sampling scenario, the order-M rank-breaking estimate $\widehat{\theta}$ in* (7.12) *is consistent for all choices of $M \geq \min_{a \in \ell} m_a$.*

Proof. It is sufficient to show that (a) the estimate $\widehat{\theta}$, (3.5) is unique under the above sampling scenario, and (b) expectation of the gradient of $\mathcal{L}_{\mathrm{RB}}(\theta^*)$ is zero, i.e, $\mathbb{E}_{\theta^*}[\nabla\mathcal{L}_{\mathrm{RB}}(\theta^*)] = 0$, [12]. For the above sampling scenario in the limit of the number of users n, Assumption 3.4 is satisfied. Therefore, from Remark 3.6, the estimate $\widehat{\theta}$, (3.5) is unique. In Lemma 3.10, we show that $\mathbb{E}_{\theta^*}[\nabla\mathcal{L}_{\mathrm{RB}}(\theta^*)] = 0$.

□

3.3.5 Algorithm to estimate $\widehat{\theta}$ given DAG $\mathcal{G}_j$'s

Summarizing the rank-breaking approach explained in the previous sections, we give Algorithm 3, an algorithm to compute $\widehat{\theta}$, (3.5), an estimate of θ^*. Algorithm 3 takes as input DAG $\mathcal{G}_j$'s generated under PL model with parameter θ^*, rank-breaking order $M \in \mathbb{Z}^+$, a desired error threshold ϵ, and returns $\widehat{\theta}$.

3.4 Main Results

We first summarize the notations defined so far and introduce some new notations that are used in our theoretical results. We define a *comparison graph* that captures the topology of the offer sets S_j. Our upper and lower bounds both depend on the spectral properties of the comparison graph. Then, we present main theoretical analyses and numerical simulations confirming the theoretical results.

Notations. Following is a summary of all the notations defined above. Also, we introduce some new notations that are used in our theoretical results. We use n to denote the number of users providing partial rankings, indexed

Algorithm 3 Estimate θ^* given DAG $\mathcal{G}_j$'s.

```
Require: DAG $\{\mathcal{G}_j\}_{1\leq j\leq n}$ generated under PL model with parameter $\theta^*$,
    rank-breaking order $M$, error threshold $\epsilon$
Ensure: $\widehat{\theta}$ - an estimate of $\theta^*$
find maximal ordered partitions $\{\mathcal{P}_j\}_{1\leq j\leq n}$ consistent with $\{\mathcal{G}_j\}_{1\leq j\leq n}$ [Algorithm 1]
construct rank breaking graph $\{G_j(S_j, E_j)\}_{1\leq j\leq n}$ from $\{\mathcal{P}_j\}_{1\leq j\leq n}$ [Algorithm 2]
$\widehat{\theta} \leftarrow \mathbf{0}_{d\times 1}$
repeat
    $\widetilde{\theta} \leftarrow \widehat{\theta}$
    for $i = 1$ to $d$ do
        $\widehat{\theta}_i \leftarrow$ from minorizing maximizing Equation (3.7) using $\widetilde{\theta}$,
            $\{G_j(S_j, E_j)\}_{1\leq j\leq n}$, $M$
    end for
until $\|\widehat{\theta} - \widetilde{\theta}\|_\infty \leq \epsilon$
return $\widehat{\theta}$
```

by $j \in [n]$ where $[n] = \{1, 2, \dots, n\}$. We use d to denote the number of items, indexed by $i \in [d]$. Given rank-breaking graphs $\{G_j(S_j, E_j)\}_{j\in[n]}$ extracted from the DAGs $\{\mathcal{G}_j\}$, we first define the order M rank-breaking graphs $\{G_j^{(M)}(S_j, E_j^{(M)})\}$, where $E_j^{(M)}$ is a subset of E_j that includes only those edges $e_j \in E_j$ with $|T(e_j)| \leq M$. This represents those edges that are included in the estimation for a choice of M. For finite sample analysis, the following quantities capture how the error depends on the topology of the data collected. Let $\kappa_j \equiv |S_j|$ and $\ell_j \equiv |E_j^{(M)}|$. We index each edge e_j in $E_j^{(M)}$ by $a \in [\ell_j]$ and define $m_{j,a} \equiv |T(e_{j,a})|$, size of top-set, for the a-th hyper edge of the j-th rank-breaking graph, and $r_{j,a} \equiv |T(e_{j,a})| + |B(e_{j,a})|$, sum of size of the top-set and the bottom-set. We let $p_j \equiv \sum_{a\in[\ell_j]} m_{j,a}$ denote the effective sample size for the observation $G_j^{(M)}$.

$$m_{j,a} \equiv |T(e_{j,a})| \,,$$

size of top-set for the $e_{j,a}$ hyper edge of rank-breaking graph $G_j^{(M)}$. (3.8)

$$r_{j,a} \equiv |T(e_{j,a})| + |B(e_{j,a})| \,,$$

sum of size of the top-set and the bottom-set for the $E_{j,a}$. (3.9)

$$p_j \equiv \sum_{a \in [\ell_j]} m_{j,a} \,,$$

sum of size of all top-sets of $G_j^{(M)}$ (which are smaller than M). (3.10)

Notice that although we do not explicitly write the dependence on M, all of the above quantities implicitly depend on the choice of M. For ease of notations, we remove the superscript M from $G_j^{(M)}$ in the following.

For a ranking σ over S, i.e., σ is a mapping from $[|S|]$ to S, let σ^{-1} denote the inverse mapping. For a vector x, let $\|x\|_2$ denote the standard l_2 norm. Let $\mathbf{1}$ denote the all-ones vector and $\mathbf{0}$ denote the all-zeros vector with the appropriate dimension. Let $\mathcal{S}^d$ denote the set of $d \times d$ symmetric matrices with real-valued entries. For $X \in \mathcal{S}^d$, let $\lambda_1(X) \leq \lambda_2(X) \leq \cdots \leq \lambda_d(X)$ denote its eigenvalues sorted in increasing order. Let $\mathrm{Tr}(X) = \sum_{i=1}^d \lambda_i(X)$ denote its trace and $\|X\| = \max\{|\lambda_1(X)|, |\lambda_d(X)|\}$ denote its spectral norm. For two matrices $X, Y \in \mathcal{S}^d$, we write $X \succeq Y$ if $X - Y$ is positive semi-definite, i.e., $\lambda_1(X - Y) \geq 0$. Let e_i denote a unit vector in $\mathbb{R}^d$ along the i-th direction.

3.4.1 Comparison graph

We define a comparison graph $\mathcal{H}([d], E)$ as a weighted undirected graph with weights

$$A_{ii'} = \sum_{j \in [n] : i, i' \in S_j} \frac{p_j}{\kappa_j(\kappa_j - 1)}.$$

That is we put an edge (i, i') if there exists a user j whose offerings is a set S_j such that $i, i' \in S_j$. Define a diagonal matrix $D = \mathrm{diag}(A\mathbf{1})$, and the corresponding graph Laplacian $L = D - A$ such that

$$L \;\equiv\; \sum_{j=1}^{n} \frac{p_j}{\kappa_j(\kappa_j - 1)} \sum_{i<i' \in S_j} (e_i - e_{i'})(e_i - e_{i'})^\top \,. \tag{3.11}$$

It is immediate that $\lambda_1(L) = 0$ with $\mathbf{1}$ as the eigenvector. There are remaining $d-1$ eigenvalues that sum to $\mathrm{Tr}(L) = \sum_j p_j$. The rescaled $\lambda_2(L)$ and $\lambda_d(L)$ capture the dependency on the topology:

$$\alpha \equiv \frac{\lambda_2(L)(d-1)}{\mathrm{Tr}(L)} \quad , \qquad \beta \equiv \frac{\mathrm{Tr}(L)}{\lambda_d(L)(d-1)} \,. \tag{3.12}$$

In an ideal case where the graph is well connected, then the spectral gap of the Laplacian is large. The chosen rescaling ensures that if all the non-zero eigenvalues are of the same order then there exists constants $0 \leq c_1, c_2 \leq 1$ such that $c_1 \leq \alpha \leq 1$ and $c_2 < \beta \leq 1$. If the graph is connected then c_1 is strictly greater than zero. If $\lambda_2(L) = \cdots = \lambda_d(L)$ then $\alpha = \beta = 1$. We will show that the performance of our estimator depends upon topology of the comparison graph through these two parameters. The larger the rescaled spectral gap α the smaller error we get with the same effective sample size. The rescaled largest eigenvalue β along with α determine how many samples are required for the analysis to hold. In general, α and β depend upon both the topology of the offer sets S_j and the topology of the rank-breaking graphs G_j, through the edge weights $A_{ii'}$. However, if topology of all the rank-breaking graphs G_j's is same then the comparison graph $\mathcal{H}$ and α, β depend only upon the topology of the offer sets S_j. For such a comparison graph, [114] provides a detailed discussion on the spectral gap for various canonical graphs following the setup given in [182].

The concavity of $\mathcal{L}_{\mathrm{RB}}(\theta)$ also depends on the following quantities.

$$\gamma_1 \;\equiv\; \min_{j,a} \left\{ \left(\frac{r_{j,a} - m_{j,a}}{\kappa_j} \right)^{2e^{2b}-2} \right\}, \;\; \gamma_2 \equiv \min_{j,a} \left\{ \left(\frac{r_{j,a} - m_{j,a}}{r_{j,a}} \right)^{2} \right\} \tag{3.13}$$

γ_1 incorporates asymmetry in probabilities of items being ranked at different positions depending upon their weight θ_i^*. Recall that b is the upper bound on $\|\theta^*\|_\infty$, Equation 3.2. γ_1 is 1 for $b = 0$ that is when all the items have the same weight θ_i^*, and it decreases exponentially with increase in b. The exponential decrease is tight and reflects the fact that under PL model probability of the highest weight item being ranked last is exponentially smaller than its probability of being ranked first.

When the rank-breaking graphs G_j's are determined such that size of the offered subsets κ_j's are increasing with d but all the top-set sizes are much smaller than the corresponding bottom set sizes, such that $\kappa_j = \omega(d)$, $m_{j,a} = o(r_{j,a})$ and $r_{j,a} = \Theta(\kappa_j)$, then for $b = O(1)$ γ_1 can be made arbitrarily close to one, for large enough problem size d. On the other hand, when either $r_{j,a}$ is much smaller than κ_j or if $r_{j,a} = \Theta(\kappa_j)$ but $m_{j,a} = O(r_{j,a})$ then accuracy can degrade significantly as stronger alternatives will have small chance of showing up in the bottom set. The value of γ_1 is quite sensitive to b.

γ_2 controls the range of the size of the top-set with respect to the size of the bottom-set for which the error decays with the rate of 1/(size of the top-set). If size of top sets $m_{j,a} = o(r_j, a)$ then γ_2 would be close to one. The dependence of accuracy on γ_1, γ_2 is demonstrated in simulations as well, Figure 3.5.

We define the following additional quantities that control our upper bound. The dependence in γ_3 and ν are due to weakness in the analysis, and ensures that the Hessian matrix is strictly negative definite.

$$\gamma_3 \;\equiv\; 1 - \max_{j,a} \left\{ \frac{4e^{16b}}{\gamma_1} \frac{m_{j,a}^2 r_{j,a}^2 \kappa_j^2}{(r_{j,a} - m_{j,a})^5} \right\} \,, \quad \nu \;\equiv\; \max_{j,a} \left\{ \frac{m_{j,a} \kappa_j^2}{(r_{j,a} - m_{j,a})^2} \right\} . \tag{3.14}$$

For our analysis to hold we need $\gamma_3 > 0$ which in addition to the conditions needed for γ_1 being close to one require that $m_{j,a} = O(\sqrt{r_{j,a}})$. We believe this is a limitation on our analysis and the results should hold for any values of $m_{j,a} = o(r_{j,a})$. For the special case when $m_{j,a} \leq 3$ for all j, a, we provide a tighter result that does not depend upon γ_3. However, in general getting rid of γ_3 is challenging. ν shows up in the number of samples required for our analysis to hold. Note that the quantities defined in this section implicitly depend on the choice of M, which controls the necessary computational power, via the definition of the rank-breaking graphs $\{G_j^{(M)}\}_{j \in [n]}$.

3.4.2 Upper bound on the achievable error

We provide an upper bound on the error for the order-M rank-breaking Algorithm 3, showing the explicit dependence on the topology of the offered sets $\{S_j\}_{j\in[n]}$. Recall from the sampling assumptions in Section 3.3 that we assume the topology of the observed DAG $\mathcal{G}_j$'s, and the rank-breaking order M is fixed apriori. The randomness that we observe is in the position of S_j items in the DAG $\mathcal{G}_j$. For an hyper edge $e \in E_j$, the randomness is in which items of the set S_j appear in the bottom $|B(e)|$ positions and the bottom $|T(e)| + |B(e)|$ positions in the preference order of the user j. This precisely captures the randomness due to the PL model in the observed DAG $\mathcal{G}_j$. The following theorem provides an upper bound on the achieved error, and a proof is provided in Section 4.6.

Theorem 3.8. *Suppose there are n users, d items parametrized by $\theta^* \in \Omega_b$, and each user $j \in [n]$ is presented with a set of offerings $S_j \subseteq [d]$ and the user provides a partial ordering under the PL model consistent with the topology of the apriori fixed DAG $\mathcal{G}_j$. For a choice of $M \in \mathbb{Z}^+$, if $\gamma_3 > 0$ and the effective sample size $\sum_{j=1}^n p_j$ is large enough such that*

$$\sum_{j=1}^n p_j \;\geq\; \frac{2^{14} e^{20b} \nu^2}{(\alpha\gamma_1\gamma_2\gamma_3)^2 \beta} \frac{p_{\max}}{\kappa_{\min}} d \log d \;, \tag{3.15}$$

where $b \equiv \max_i |\theta_i^*|$ is the dynamic range, $p_{\max} = \max_{j\in[n]} p_j$, $\kappa_{\min} = \min_{j\in[n]} \kappa_j$, α is the (rescaled) spectral gap, β is the (rescaled) spectral radius in (3.12), *and γ_1, γ_2, γ_3, and ν are defined in* (3.13) *and* (3.14), *then the generalized rank-breaking estimator in* (7.12) *achieves*

$$\frac{1}{\sqrt{d}} \|\widehat{\theta} - \theta^*\|_2 \;\leq\; \frac{40 e^{7b}}{\alpha\gamma_1\gamma_2^{3/2}\gamma_3} \sqrt{\frac{d \log d}{\sum_{j=1}^n p_j}} \;, \tag{3.16}$$

with probability at least $1 - 3e^3 d^{-3}$. Moreover, for $M \leq 3$ the above bound holds with γ_3 replaced by one, giving a tighter result.

Note that the dependence on the choice of M is not explicit in the bound, but rather is implicit in the construction of the comparison graph and the number of effective samples.

In an ideal case, $b = O(1)$ and $m_{j,a} = O((r_{j,a})^{1/3})$, $r_{j,a} = \Theta(\kappa_j)$, $\kappa_j = \omega(d)$, for all (j, a). There exist positive constants c_1, c_2 such that if $m_{j,a} < c_1 (r_{j,a})^{1/3}$

and $r_{j,a} > c_2 \kappa_j$ for all $\{j, a\}$, then the condition $\gamma_3 > 0$ is met, for large enough problem size d. Moreover, in this ideal case there exists constants $0 < c_3, c_4 \leq 1$ such that $c_3 \leq \gamma_1 < 1$, $c_4 \leq \gamma_2 < 1$. If the comparison graph $\mathcal{H}$ is well connected then there exists a constant $0 < c_5 \leq 1$ such that the rescaled spectral gap $c_5 \leq \alpha \leq 1$. Further, in this ideal case, the condition on the effective sample size is met with $\sum_j p_j = O(d \log d)$, (3.15). Therefore the effective sample size $\sum_{j=1}^{n} p_j = \Omega(d \log d)$ is sufficient to ensure $\|\widehat{\theta} - \theta^*\|_2 = o(\sqrt{d})$ which is only a logarithmic factor larger than the number of parameters. We need $m_{j,a} = O((r_{j,a})^{1/3})$ to satisfy $(\nu^2 p_{\max})/\kappa_{\min} = O(1)$, otherwise $m_{j,a} = O((r_{j,a})^{1/2})$ is sufficient to ensure $\gamma_3 > 0$. We believe that dependence in γ_3 is weakness of our analysis and there is no dependence as long as $m_{j,a} < r_{j,a}$. For, rank-breaking order $M \leq 3$, we are able to give tighter results where there is no dependence on γ_3.

As explained above, in the ideal case, for large enough problem size d, there exists a positive constant C such that $\|\widehat{\theta} - \theta^*\|_2^2 \leq C d^2 \log d / (\sum_{j=1}^{n} p_j)$. Recall from the construction of the likelihood objective function, $\mathcal{L}_{\mathrm{RB}}(\theta) = \sum_{j \in [n]} \sum_{e \in E_j : |T(e)| \leq M} \ln \mathbb{P}_\theta(e)$. If we fix all the problem parameters including topology of the DAG $\mathcal{G}_j$'s and increase M then $p_j = \sum_{a \in [\ell_j]} m_{j,a}$ increases. Therefore, by increasing M we can get the same number of effective samples $\sum_{j=1}^{n} p_j$ with smaller number of rankings n. However, increasing M increases computational complexity as $M!$. Therefore, to achieve a fixed target accuracy $\|\widehat{\theta} - \theta^*\|_2$, an analyst can trade-off the required number of rankings with the budgeted computational complexity.

If the DAG $\mathcal{G}_j$'s are complete graph that is each user provides a full ranking over the offered subset S_j, we get $m_{j,a} = 1$, $\ell_j = \kappa_j - 1$, and the total effective sample size $\sum_j p_j = \sum_{j \in [n]} (\kappa_j - 1)$. Therefore, from the above theorem, $\sum_{j \in [n]} (\kappa_j - 1) = \Omega(d \log d)$ is sufficient to ensure $\|\widehat{\theta} - \theta^*\|_2 = o(\sqrt{d})$. It matches with the results for full rankings given in [84, 114].

Unordered vs. ordered top-m ranking. In the ideal case, a perhaps surprising observation is that, for a ranking j, sizes of the top-sets $\{m_{j,a}\}_{a \in [\ell_j]}$ impacts estimation accuracy only via $p_j = \sum_{a \in [\ell_j]} m_{j,a}$, when $m_{j,a}$'s are sufficiently small in comparison to $r_{j,a}$'s, sum of the top-set size and the bottom-set size. In particular, for estimation accuracy it does not matter whether users reveal their top-m choices in the ordered way $\{i_1\} \succ \{i_2\} \succ \cdots \succ \{i_m\} \succ \{i_{m+1}, \cdots, i_k\}$ or the unorderd way $\{i_1, i_2, \cdots, i_m\} \succ \{i_{m+1}, \cdots i_k\}$, when m is sufficiently small in comparison to k. Numerical results in Figure 3.5 confirm this.

Proof idea. The analysis of the optimization in (7.12) shows that, with high probability, $\mathcal{L}_{\rm RB}(\theta)$ is strictly concave with $\lambda_2(H(\theta)) \leq -C_b\gamma_1\gamma_2\gamma_3\lambda_2(L) < 0$ for all $\theta \in \Omega_b$ (Lemma 3.12), and the gradient is also bounded with $\|\nabla\mathcal{L}_{\rm RB}(\theta^*)\| \leq C'_b\gamma_2^{-1/2}(\sum_j p_j \log d)^{1/2}$ (Lemma 3.11). This leads to Theorem 4.5:

$$\|\widehat{\theta}-\theta^*\|_2 \;\leq\; \frac{2\|\nabla\mathcal{L}_{\rm RB}(\theta^*)\|}{-\lambda_2(H(\theta))} \;\leq\; C''_b\frac{\sqrt{\sum_j p_j \log d}}{\gamma_1\gamma_2^{3/2}\gamma_3\lambda_2(L)}\;,$$

where C_b, C'_b, and C''_b are constants that only depend on b, and $\lambda_2(H(\theta))$ is the second largest eigenvalue of a negative semidefinite Hessian matrix $H(\theta)$ of $\mathcal{L}_{\rm RB}(\theta)$. Recall that $\theta^\top\mathbf{1}=0$ since we restrict our search in Ω_b. Hence, the error depends on $\lambda_2(H(\theta))$ instead of $\lambda_1(H(\theta))$ whose corresponding eigenvector is the all-ones vector.

3.4.3 Lower bound on computationally unbounded estimators

Suppose $M = d$. We prove a fundamental lower bound on the achievable error rate that holds for any *unbiased* estimator with no restrictions on the computational complexity. For each (j,a), define $\eta_{j,a}$ as

$$\begin{aligned}
\eta_{j,a} \;\equiv\;& \sum_{u=0}^{m_{j,a}-1}\Big(\frac{1}{r_{j,a}-u}+\frac{u(m_{j,a}-u)}{m_{j,a}(r_{j,a}-u)^2}\Big) \\
&+\sum_{u<u'\in[m_{j,a}-1]}\frac{2u}{m_{j,a}(r_{j,a}-u)}\frac{m_{j,a}-u'}{r_{j,a}-u'} && (3.17)\\
<\;& \sum_{u=0}^{m_{j,a}-1}\Big(\frac{1}{m_{j,a}-u}+\frac{u}{m_{j,a}(m_{j,a}-u)}\Big) \\
&+\sum_{u<u'\in[m_{j,a}-1]}\frac{2u}{m_{j,a}(m_{j,a}-u)} && (3.18)\\
=\;& \sum_{u=0}^{m_{j,a}-1}\Big(\frac{1}{m_{j,a}-u}+\frac{u}{m_{j,a}(m_{j,a}-u)}+\frac{2u(m_{j,a}-1-u)}{m_{j,a}(m_{j,a}-u)}\Big)=m_{j,a}\,,
\end{aligned}$$

where (3.18) follows from the fact that (3.17) is monotonically strictly decreasing in $r_{j,a}$ for $r_{j,a} \geq m_{j,a}$. Since by definition $r_{j,a} > m_{j,a}$, we substitute $r_{j,a} = m_{j,a}$ to get a strict upper bound.

Theorem 3.9. *Let $\mathcal{U}$ denote the set of all unbiased estimators of θ^* that are centered such that $\widehat{\theta}\mathbf{1} = 0$, and let $\mu = \max_{j\in[n],a\in[\ell_j]}\{m_{j,a} - \eta_{j,a}\}$. For all $b > 0$,*

$$\inf_{\widehat{\theta}\in\mathcal{U}} \sup_{\theta^*\in\Omega_b} \mathbb{E}[\|\widehat{\theta} - \theta^*\|^2] \;\geq\; \max\left\{\frac{(d-1)^2}{\sum_{j=1}^{n}\sum_{a=1}^{\ell_j}(m_{j,a} - \eta_{j,a})}\,,\,\frac{1}{\mu}\sum_{i=2}^{d}\frac{1}{\lambda_i(L)}\right\}. \tag{3.19}$$

The proof relies on the Cramer-Rao bound and is provided in Section 3.6.6. Since $0 < \eta_{j,a} < m_{j,a}$, the mean squared error is lower bounded by $(d-1)^2/(\sum_{j=1}^{n}\sum_{a=1}^{\ell_j} m_{j,a}) = (d-1)^2/(\sum_{j=1}^{n} p_j)$, where $\sum_{j=1}^{n} p_j$ is the effective sample size. Comparing it to the upper bound in (5.9), this is tight up to a logarithmic factor when (a) the topology of the data is well-behaved such that all the quantities $\gamma_1, \gamma_2, \gamma_3, \alpha, \beta$ are greater than a positive constant $c \leq 1$; and (b) there is no limit on the computational power and M can be made as large as we need. For full-rankings, this bound reduces to the one given in [84, 114]. For full rankings, $\sum_{a=1}^{\ell_j}(m_{j,a} - \eta_{j,a}) = (\kappa_j - 1)^2/\kappa_j$.

The bound in Eq. (3.19) further gives a tighter lower bound, capturing the dependency in $\eta_{j,a}$'s and $\lambda_i(L)$'s. The second term in (3.19) implies we get a tighter bound when $\lambda_2(L)$ is smaller. If the comparison graph $\mathcal{H}$ is disconnected that is $\lambda_2(L) = 0$, the bound shows that θ^* can not be estimated.

To understand the impact of $\eta_{j,a}$ on MSE, we plot $(m_{j,a} - \eta_{j,a})/r_{j,a}$ as a function of $m_{j,a}/r_{j,a}$ for different values of $r_{j,a}$ in Figure 3.3. Recall that $m_{j,a}$ is the size of the top-set, (3.8) and $r_{j,a}$ is the sum of size of the top-set and the bottom-set, (3.9). We vary $m_{j,a}$ from 1 to $r_{j,a} - 1$, for $r_{j,a}$ in $\{2, 4, 8, 16, 32, 256, 1024\}$. From the Theorem 3.9, contribution of an hyper edge $e_{j,a}$ to the effective samples is $(m_{j,a} - \eta_{j,a})$. Since $\eta_{j,a}$ increases with $m_{j,a}$, a natural question is what is the optimal value of $m_{j,a}$ that gives the smallest MSE, for a fixed $r_{j,a}$. Figure 3.3 shows that $(m_{j,a} - \eta_{j,a})/r_{j,a}$ achieves its maximum value at $m/r \approx 0.8$ when r is sufficiently large. It also shows that $(m_{j,a} - \eta_{j,a}) \geq c\, m_{j,a}$, for $m_{j,a}/r_{j,a} \leq c_1 (\approx 0.8)$, for positive constants $c, c_1 < 1$, when $r_{j,a}$ is large. That is the contribution of an hyper edge $e_{j,a}$ to the effective sample size is at least $c\,m_{j,a}$ for $m_{j,a}/r_{j,a} \leq c_1$. Comparing this with the lower bound for top-$m_{j,a}$ ranking given in [114], it can be concluded that the (unobserved) relative ordering among the items in the top-set of the hyper edge $e_{j,a}$ has limited impact on the MSE. [114] show in the their lower bound that the contribution of top-m ranking on the effective sample size is m.

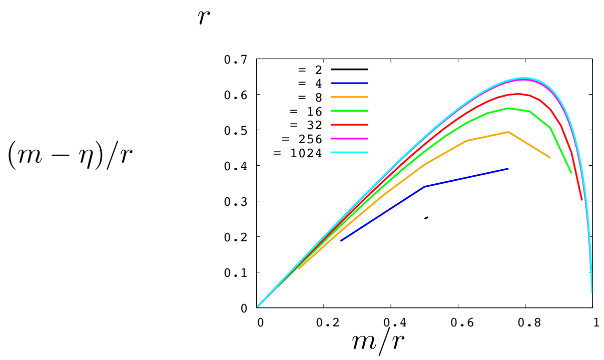


Figure 3.3: It shows how η varies as a function of m, size of the top-set, for a fixed value of r, sum of top-set and the bottom-set sizes, Equation (3.17).

Note that the lower bound is derived for the easiest case, $b = 0$, when all the items $i \in [d]$ have the same weight $\theta_i^* = \theta_0^*$. Therefore, the above conclusion that the relative ordering among the items in the top-set of the hyper edge $e_{j,a}$ has limited impact on the accuracy can be made only for this case when all the items have the same PL weight. However, the upper bound shows that this conclusion holds true in general. The 'unordered vs. ordered top-m ranking' paragraph in the previous section explains that for the ideal case when $m_{j,a}$ is sufficiently small in comparison to $r_{j,a}$, the relative ordering has limited impact.

Recall that in the upper bound, γ_1 and γ_2 capture the impact of $m_{j,a}/r_{j,a}$ on the effective number of samples. However, for $b = 0$, $\gamma_1 = 1$, and for $b > 0$ it captures asymmetry in the probability of the highest weight item appearing in bottom set. $\gamma_2 = \min_{j,a}\big\{\big(\frac{r_{j,a}-m_{j,a}}{r_{j,a}}\big)^2\big\}$ captures the role played by $\eta_{j,a}$ in the lower bound.

3.4.4 Numerical results

In the following, we give numerical results confirming our theoretical results. Our numerical experiments show that the dependence of MSE on $n, d, \kappa_j, r_{j,a}, m_{j,a}, \ell_j$ given in Theorem 4.5, Equation (5.9) hold true, even when the conditions for the theorem to hold are not met. For the theorem to hold, it is required that the number of items d, the set sizes κ_j and the hyper edge sizes $r_{j,a}$ are sufficiently large such that $\gamma_3 > 0$ and the number of effective samples satisfies (3.15). However, in all our experiments the number of items $d <= 512$ and $b = 2$, therefore from (3.14) $\gamma_3 < 0$, and the condition in (3.15) is not met.

Impact of the number of independent rankings n and the number of rank-breaking hyper edges ℓ_j on accuracy. Figure 4.1 (first panel) shows the accuracy-sample tradeoff for increasing computation M on the same data. As predicted by the anlaysis, generalized rank-breaking (GRB) is consistent (Remark 3.7) and the mean square error (MSE) decays at the rate $(1/n)$, and decreases with increase in M, order of rank-breaking (Theorem 4.5). For comparison, we also plot the MSE achieved by pairwise rank-breaking (PRB) approach where we include all paired relations derived from

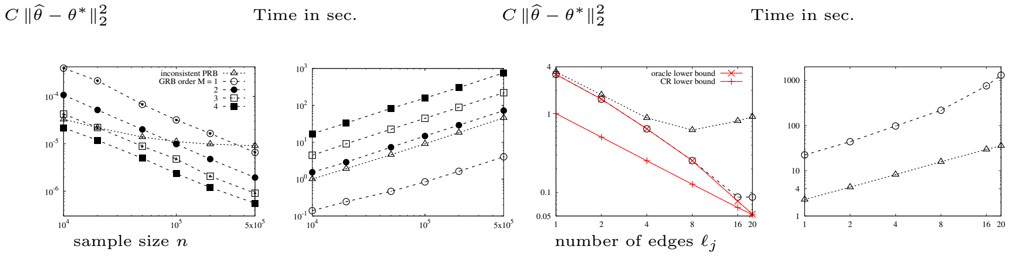


Figure 3.4: Smaller error is achieved when using more computational resources with larger M and using all paired comparisons results in an inconsistent Pairwise Rank-Breaking (PRB) whose error does not vanish with sample size (first panel). Generalized Rank-Breaking (GRB) utilizes all the observations achieving the oracle lower bound (third panel). The PL weights are chosen uniformly spaced over $[-2, 2]$. On the first panel, we fix $d = 256$, $\kappa = 32$, $\ell = 4$, $m_a = a$ for $a \in \{1, 2, 3, 4\}$, and sample posets from the canonical scenario explained in Section 3.3.4. On the third panel, we let $n = 10^5$, $d = 512$, $\kappa = 64$, $m_a = 3$ for all $a \in [\ell]$ and vary $\ell \in \{1, 2, 4, 8, 16\}$. The second and the fourth panel show the computation time for the first and the third panel respectively.

data, which we call *inconsistent PRB*. As predicted by [13], this results in an inconsistent estimate, whose MSE does not vanish as we increase the sample size. Notice that including all paired comparisons increases bias, but also decreases variance of the estimate. Hence, when sample size is limited and variance is dominating the bias, it is actually beneficial to include those biased paired relations to gain in variance at the cost of increased bias. Theoretical analysis of such a bias-variance tradeoff is outside the scope of this paper, but proposes an interesting research direction.

In the third panel, the GRB with $M = 3$ achieves decreasing MSE, whereas for PRB the increased bias dominates the MSE. For comparisons, we provide the error achieved by an oracle estimator who knows the exact ordering among those items belonging to the top-sets and runs MLE. For example, if $\ell = 2$, the GRB observes an ordering $(\{i_1, i_2, i_4, i_5, \ldots\} \prec \{i_{17}, i_3, i_6\} \prec \{i_9, i_2, i_{11}\})$ whereas the oracle estimator has extra information on the ordering among those top sets, i.e. $(\{i_1, i_2, i_4, i_5, \ldots\} \prec i_{17} \prec i_3 \prec i_6 \prec i_9 \prec i_2 \prec i_{11}\})$. Perhaps surprisingly, GRB is able to achieve a similar performance without this significant extra information, unless ℓ is large. The performance degradation in large ℓ regime stems from the fact that the ratio of m_a and r_a approaches 1 for a close to ℓ when ℓ is large. Therefore the parameters γ_1 and γ_2 become small, and the upper bound MSE increases consequentially. The normalization constant C is $1/d^2$ for the first panel and nm/d^2 for the third panel. All the numerical results in this paper are averaged over 10 instances. Standard error is very small in all the results, therefore we do not give error bars, except in the first panel in Figure 4.1.

Impact of the top-set size m and the set-size κ on accuracy. In Figure 3.5 first and third panel, we compare performance of our algorithm with pairwise breaking, Cramer Rao lower bound and oracle MLE lower bound. Oracle MLE knows relative ordering of items in the top-sets $T(e)$ and hence is strictly better than the GRB. For the settings chosen, Oracle MLE gets the ordered ranking of top-m items whereas GRB gets unordered top-m items. As predicted by our analysis, GRB matches with the oracle MLE which means relative ordering of top-m items among themselves is statistically insignificant when m is sufficiently small in comparison to $r = \kappa$. For $r = \kappa = 32$ in the first panel, MSE decays as m increases from 1 to 5. However, when $r = \kappa = 16$ in the third panel, for the same increase of m

from 1 to 5 MSE starts increasing when m grows beyond 4. The reason is that the quantities γ_1 and γ_2 get smaller as m increases, and the upper bound increases consequently. The normalization constant C is n/d^2 for these two panels.

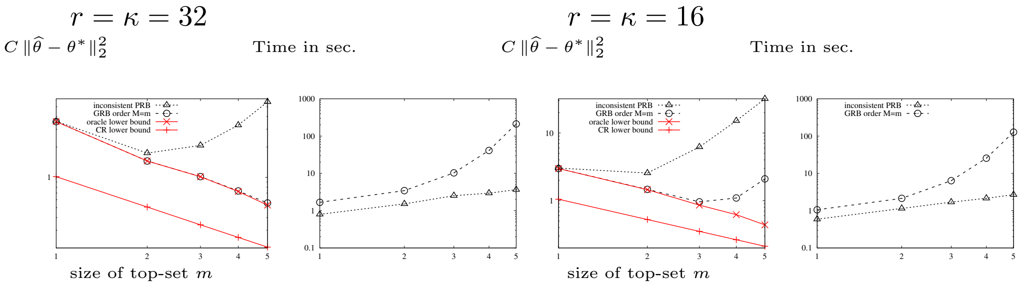


Figure 3.5: PRB: pairwise rank-breaking, GRB: generalized rank-breaking. MSE decreases as m increases when r, sum of the size of the top-set and the bottom-set is sufficiently large (first panel). When r is small, with increase in m MSE initially decreases but as m grows large MSE starts increasing (third panel). θ^* is chosen uniformly spaced over $[-2, 2]$ and $d = 512$, $n = 10^5$ and number of hyperedges $\ell = 1$. The second and the fourth panel show the computation time for the first and the third panel respectively.

Impact of the dynamic range b on accuracy. In Figure 3.6, we show the impact of b and $r = \kappa$ on the accuracy for fixed $m = 4$. When κ is small, γ_2 is small, and hence error is large; when b is large γ_1 is exponentially small, and hence error is significantly large. This is different from learning Mallows models in [4] where peaked distributions are easier to learn, and is related to the fact that we are not only interested in recovering the (ordinal) ranking but also the (cardinal) weight. The normalization constant C is nm/d^2.

3.4.5 Real-world datasets

On sushi preferences [104] and jester dataset [78], we improve over pairwise breaking and achieve same performance as the oracle MLE.

Sushi dataset. There are $d = 100$ types of sushi. Full rankings over subsets S_j of size $\kappa = 10$ are provided by $n = 5000$ individuals. The offering subsets S_j are chosen uniformly at random from the entire set d. We set the ground truth θ^* to be the MLE of the PL weights over the entire data. In the left panel, for each $m \in \{3, 4, 5, 6\}$, we remove the known ordering among

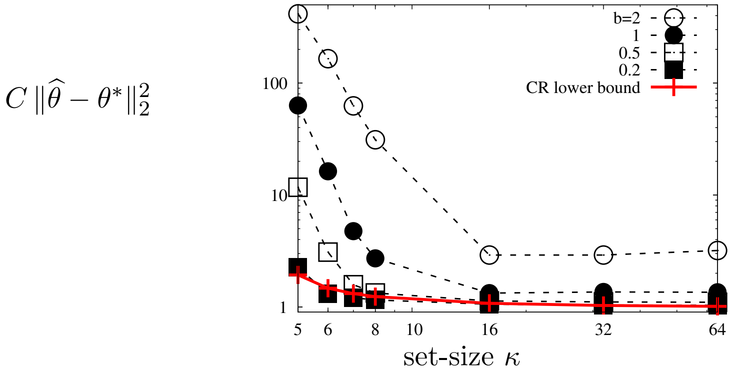


Figure 3.6: MSE increases as the dynamic range b gets large. $d = 512$, $n = 10^5$ and θ^* is chosen uniformly spaced over $[-2, 2]$. Number of hyper edges $\ell = 1$ with $r = \kappa$ and $m = 4$.

the top-m and bottom-$(10 - m)$ sushi in each set, and run our estimator with one rank-breaking hyper edge between top-m and bottom-$(10 - m)$ items. We compare our algorithm with inconsistent pairwise breaking (using optimal choice of parameters from [114]) and the oracle MLE. For $m \leq 6$, the proposed rank-breaking performs as good as the oracle who knows the relative ordering among the top m items. In other words, an individual providing a set of ordered top-6 sushi or a set of unordered top-6 sushi statistically reveals the same information, for the purpose of estimating the ground truth parameters. As predicted by our theory, error decreases with increase in top-set size m.

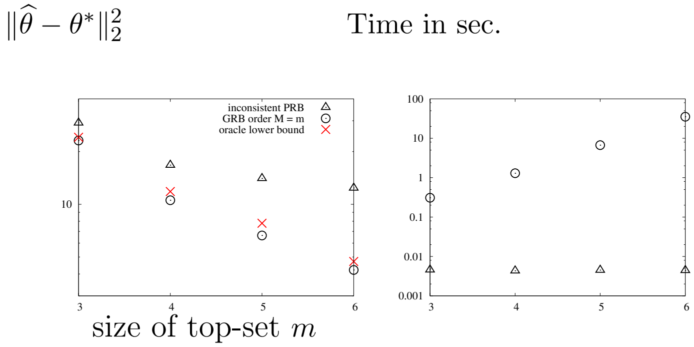


Figure 3.7: Generalized rank-breaking improves over pairwise RB and performs as good as oracle MLE on sushi dataset. The sushi dataset has $d = 100$, $n = 5000$, and $\kappa = 10$. The right panel shows computation time.

Jester dataset. It consists of continuous ratings between -10 to $+10$ of 100 jokes on sets of size κ, $36 \leq \kappa \leq 100$, by $24,983$ users. We convert cardinal ratings into ordinal full rankings. The ground truth θ^* is set to be the MLE of the PL weights over the entire data. For $m \in \{2,3,4,5\}$, we convert each full ranking into a poset that has $\ell = \lfloor \kappa/m \rfloor$ partitions of size m, by removing known relative ordering from each partition. The leads to total number of effective samples $\sum_j p_j = \sum_j \sum_{a \in [\ell_j]} m_{j,a} = \sum_j (\kappa_j - m)$, which is approximately equal for each $m \in \{2,3,4,5\}$. However, with increasing m, the quantities $\gamma_1, \gamma_2, \gamma_3$ become smaller and hence the error increases (third panel in Figure 3.8). Figure 3.8 compares the three algorithms for two different settings. In the first panel, we fix $m = 4$ and vary the number of samples n. Mean square error decreases with increase in the number of samples. In the third panel, we use $n = 5000$ samples, and vary $m \in \{2,3,4,5\}$.

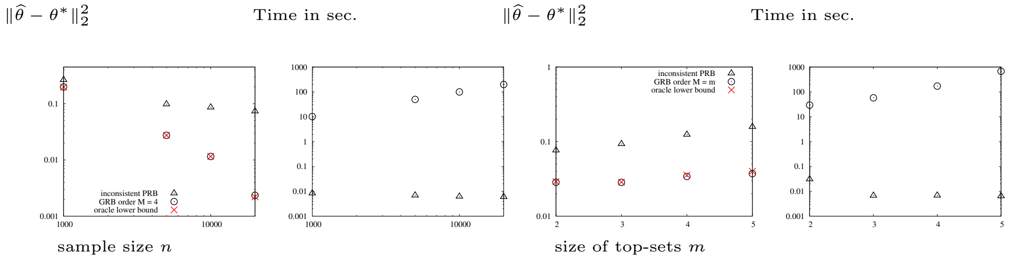


Figure 3.8: Generalized rank-breaking improves over pairwise RB and performs as good as oracle MLE on jester dataset. The jester dataset which has $d = 100$, $n = 24,983$, and $36 \leq \kappa_j \leq 100$. The second and the fourth panel show the computation time for the first and the third panel respectively.

3.5 Computational and statistical tradeoff

For estimators with limited computational power, however, the lower bound Theorem 3.9 fails to capture the dependency on the allowed computational power. Understanding such fundamental trade-offs is a challenging problem, which has been studied only in a few special cases, e.g. planted clique problem [52, 151]. This is outside the scope of this paper, and we instead investigate the trade-off achieved by the proposed rank-breaking approach.

When we are limited on computational power, Theorem 4.5 implicitly captures this dependence when order-M rank-breaking is used. The dependence is captured indirectly via the resulting rank-breaking $\{G_{j,a}\}_{j\in[n],a\in[\ell_j]}$ and the topology of it. We make this trade-off explicit by considering a simple but canonical example. Suppose $\theta^* \in \Omega_b$ with $b = O(1)$. Each user gives an i.i.d. partial ranking, where all items are offered and the partial ranking is based on an ordered partition with $\ell_j = \lfloor\sqrt{2c}d^{1/3}\rfloor$ subsets for a constant c. The top subset has size $m_{j,1} = 1$, and the a-th subset has size $m_{j,a} = a$, up to $a < \ell_j$. The choice of ℓ_j with a sufficiently small constant c ensures that all the conditions of the ideal case explained in the previous section for holding the Theorem 4.5 are satisfied.

Computation. For a choice of M such that $M \leq \ell_j - 1$, we consider the computational complexity in computing $\theta^{(t)}$, (3.7) in one iteration of the minorization-maximization algorithm, which scales as $T(M, n) = O(M! \times dn)$. A detailed analysis of the convergence rate of the MM algorithm is outside the scope of this paper.

Accuracy. Under the canonical setting, for $M \leq \ell_j - 1$, Laplacian matrix L of the comparison graph $\mathcal{H}$ is $L = nM(M+1)/(2d(d-1))\big(d\,\mathbb{I} - \mathbf{1}\mathbf{1}^\top\big)$. All the non-zero eigenvalues of this complete graph are equal, $\lambda_2(L) = \cdots = \lambda_d(L) = \mathrm{Tr}(L)/(d-1)$. Therefore, the resclaed spectral gap $\alpha = 1$, and the rescaled largest eigenvalue $\beta = 1$. Since the effective sample size is $\sum_{j,a} m_{j,a}\mathbb{I}\{m_{j,a} \leq M\} = nM(M+1)/2$, it follows from Theorem 4.5 that the (rescaled) root mean squared error is $O(\sqrt{(d\log d)/(nM^2)})$. In order to achieve a smaller target error rate of ε for a fixed problem size d, an analyst can increase the rank-breaking order M and/or increase n that is collect more i.i.d. rankings. Fixing the rank-breaking order M, we need to collect $n = \Omega((d\log d)/(\varepsilon^2 M^2))$ i.i.d. rankings. The resulting trade-off between run-time and root mean squared error ε is $T(\varepsilon) \propto (M!(d^2\log d)/(\varepsilon^2 M^2)$. The computational complexity is quadratic in the target error ϵ, when we can collect more rankings. On the other hand, fixing the number of rankings n, we need to choose $M = \Omega((1/\varepsilon)\sqrt{(d\log d)/n})$. The resulting trade-off between run-time and root mean squared error ε is $T(\epsilon) \propto (\lceil(1/\varepsilon)\sqrt{(d\log d)/n}\rceil)!dn$. The computational complexity is super exponential in the target error ϵ, for a fixed problem size d and the number of rankings n. Super exponential complexity is unavoidable as computing likelihood is super exponential in M. However, our approach provides flexibility to the analyst to choose between collecting more rankings n or increasing the rank-breaking order M to achieve the desired target error. We show numerical experiment under this canonical setting in Figure 4.1 (left) with $d = 256$ and $M \in \{1, 2, 3, 4, 5\}$, illustrating the trade-off in practice.

3.6 Proofs

We provide the proofs of the main results.

3.6.1 Proof of Lemma 3.3

In the following, we show that $Q(e, \theta; \theta^{(t)})$ minorizes $\ln(\mathbb{P}_\theta(e))$ at $\theta^{(t)}$. Using Jensen's inequality $\ln(\mathbb{E}[X]) \geq \mathbb{E}[\ln(X)]$, for any given parameter $\theta^{(t)} \in \mathbb{R}^d$, we have,

$$
\begin{aligned}
& \ln(\mathbb{P}_\theta(e)) \\
= \;& \ln\big(\mathbb{P}_\theta\big(B(e) \prec T(e)\big)\big) \\
= \;& \ln\left(\sum_{\sigma\in\Lambda_{T(e)}} \frac{\exp\left(\sum_{c=1}^{|T(e)|}\theta_{\sigma(c)}\right)}{\prod_{u=1}^{|T(e)|}\left(\sum_{c'=u}^{|T(e)|}\exp\left(\theta_{\sigma(c')}\right)+\sum_{i\in B(e)}\exp\left(\theta_i\right)\right)}\right) \\
\geq \;& \sum_{\sigma\in\Lambda_{T(e)}} \frac{\mathbb{P}_{\theta^{(t)}}(e,\sigma)}{\mathbb{P}_{\theta^{(t)}}(e)} \\
& \ln\left(\frac{\exp\left(\sum_{c=1}^{|T(e)|}\theta_{\sigma(c)}\right)}{\prod_{u=1}^{|T(e)|}\left(\sum_{c'=u}^{|T(e)|}\exp\left(\theta_{\sigma(c')}\right)+\sum_{i\in B(e)}\exp\left(\theta_i\right)\right)}\,\frac{\mathbb{P}_{\theta^{(t)}}(e)}{\mathbb{P}_{\theta^{(t)}}(e,\sigma)}\right) \qquad (3.20) \\
= \;& \sum_{\sigma\in\Lambda_{T(e)}} \frac{\mathbb{P}_{\theta^{(t)}}(e,\sigma)}{\mathbb{P}_{\theta^{(t)}}(e)} \\
& \sum_{u=1}^{|T(e)|}\left(\theta_{\sigma(u)} - \ln\left(\sum_{c'=u}^{|T(e)|}\exp\left(\theta_{\sigma(c')}\right)+\sum_{i\in B(e)}\exp\left(\theta_i\right)\right)\right) \\
& + \sum_{\sigma\in\Lambda_{T(e)}} \frac{\mathbb{P}_{\theta^{(t)}}(e,\sigma)}{\mathbb{P}_{\theta^{(t)}}(e)} \ln\left(\frac{\mathbb{P}_{\theta^{(t)}}(e)}{\mathbb{P}_{\theta^{(t)}}(e,\sigma)}\right) \\
\geq \;& \sum_{\sigma\in\Lambda_{T(e)}} \frac{\mathbb{P}_{\theta^{(t)}}(e,\sigma)}{\mathbb{P}_{\theta^{(t)}}(e)} \sum_{u=1}^{|T(e)|}\left(\theta_{\sigma(u)} - \frac{\sum_{c'=u}^{|T(e)|}\exp\left(\theta_{\sigma(c')}\right)+\sum_{i\in B(e)}\exp\left(\theta_i\right)}{\sum_{c'=u}^{|T(e)|}\exp\left(\theta^{(t)}_{\sigma(c')}\right)+\sum_{i\in B(e)}\exp\left(\theta^{(t)}_i\right)}\right) \\
& + f(e,\theta^{(t)}) \\
\equiv \;& Q(e,\theta;\theta^{(t)})\,.
\end{aligned}
$$

Note that inequality in (3.20) is tight if $\theta_t = \theta$. The last inequality follows from the fact that for any positive x and y, we have

$$-\ln x \geq 1 - \ln y - (x/y) \quad \text{with equality if and only if } x = y.$$

Therefore, $Q(e,\theta;\theta^{(t)})$ minorizes $\ln(\mathbb{P}_\theta(e))$ and is equal to $\ln(\mathbb{P}_\theta(e))$ if and only if $\theta^{(t)} = \theta$.

3.6.2 Proof of Theorem 4.5

We define few additional notations. $p \equiv (1/n)\sum_{j=1}^{n} p_j$. $V(e_{j,a}) \equiv T(e_{j,a}) \cup B(e_{j,a})$ for all $j \in [n]$ and $a \in [\ell_j]$. Note that by definition of rank-breaking edge $e_{j,a}$, $V(e_{j,a})$ is a random set of items that are ranked in bottom $r_{j,a}$ positions in a set of S_j items by the user j.

The proof sketch is inspired from [114]. The main difference and technical challenge is in showing the strict concavity of $\mathcal{L}_{\rm RB}(\theta)$ when restricted to Ω_b. We want to prove an upper bound on $\Delta = \widehat{\theta} - \theta^*$, where $\widehat{\theta}$ is the sample dependent solution of the optimization (7.12) and θ^* is the true utility parameter from which the samples are drawn. Since $\widehat{\theta}, \theta^* \in \Omega_b$, it follows that $\Delta\mathbf{1} = 0$. Since $\widehat{\theta}$ is the maximizer of $\mathcal{L}_{\rm RB}(\theta)$, we have the following inequality,

$$\begin{aligned} \mathcal{L}_{\rm RB}(\widehat{\theta}) - \mathcal{L}_{\rm RB}(\theta^*) - \langle \nabla \mathcal{L}_{\rm RB}(\theta^*), \Delta \rangle &\geq -\langle \nabla \mathcal{L}_{\rm RB}(\theta^*), \Delta \rangle \\ &\geq -\|\nabla \mathcal{L}_{\rm RB}(\theta^*)\|_2 \|\Delta\|_2, \quad (3.21) \end{aligned}$$

where the last inequality uses the Cauchy-Schwartz inequality. By the mean value theorem, there exists a $\theta = c\widehat{\theta} + (1-c)\theta^*$ for some $c \in [0,1]$ such that $\theta \in \Omega_b$ and

$$\begin{aligned} \mathcal{L}_{\rm RB}(\widehat{\theta}) - \mathcal{L}_{\rm RB}(\theta^*) - \langle \nabla \mathcal{L}_{\rm RB}(\theta^*), \Delta \rangle &= \frac{1}{2}\Delta^\top H(\theta)\Delta \\ &\leq -\frac{1}{2}\lambda_2(-H(\theta))\|\Delta\|_2^2, \end{aligned} \quad (3.22)$$

where $\lambda_2(-H(\theta))$ is the second smallest eigen value of $-H(\theta)$. We will show in Lemma 3.12 that $-H(\theta)$ is positive semi definite with one eigenvalue at zero with a corresponding eigen vector $\mathbf{1} = [1, \ldots, 1]^\top$. The last inequality follows since $\Delta^\top \mathbf{1} = 0$. Combining Equations (7.24) and (3.22),

$$\|\Delta\|_2 \leq \frac{2\|\nabla \mathcal{L}_{\rm RB}(\theta^*)\|_2}{\lambda_2(-H(\theta))},$$

where we used the fact that $\lambda_2(-H(\theta)) > 0$ from Lemma 3.12. The following technical lemmas prove that the norm of the gradient is upper bounded by $\gamma_2^{-1/2} e^b \sqrt{6np\log d}$ with high probability and the second smallest eigen value of negative of the Hessian is lower bounded by $(1/8)\, e^{-6b}\alpha\gamma_1\gamma_2\gamma_3(np/(d-1))$. This finishes the proof of Theorem 4.5.

The (random) gradient of the log likelihood in (7.12) can be written as the following, where the randomness is in which items ended up in the top set $T(e_{j,a})$ and the bottom set $B(e_{j,a})$:

$$\nabla_i \mathcal{L}_{\rm RB}(\theta) \;\; = \;\; \sum_{j=1}^{n} \sum_{a=1}^{\ell_j} \sum_{\substack{\mathcal{C} \subseteq S_j, \\ |\mathcal{C}| = r_{j,a}-1}} \mathbb{I}\big\{ \, V(e_{j,a}) = \{\mathcal{C}, i\} \, \big\} \frac{\partial \log \mathbb{P}_\theta(e_{j,a})}{\partial \theta_i} \, .$$

Note that we are intentionally decomposing each summand as a summation over all $\mathcal{C}$ of size $r_{j,a}-1$, such that we can separate the analysis of the expectation in the following lemma. The random variable $\mathbb{I}\{\{\mathcal{C}, i\} = V(e_{j,a})\}$ indicates that we only include one term for any given instance of the sample. Note that the event $\mathbb{I}\{\{\mathcal{C}, i\} = V(e_{j,a})\}$ is equivalent to the event that the $\{\mathcal{C}, i\}$ items are ranked in bottom $r_{j,a}$ positions in the set S_j, that is $V(e_{j,a})$ items are ranked in bottom $r_{j,a}$ positions in the set S_j.

Lemma 3.10. *If the j-th poset is drawn from the PL model with weights θ^* then for any given $\mathcal{C}' \subseteq S_j$ with $|\mathcal{C}'| = r_{j,a}$,*

$$\mathbb{E}\left[\mathbb{I}\big\{ \mathcal{C}' = V(e_{j,a}) \big\} \frac{\partial \log \mathbb{P}_{\theta^*}(e_{j,a})}{\partial \theta_i^*} \middle| \{e_{j,a'}\}_{a'<a} \right] \;\; = \;\; 0 \, .$$

First, this lemma implies that $\mathbb{E}\big[\mathbb{I}\big\{ \, \mathcal{C}' = V(e_{j,a}) \, \big\} \frac{\partial \log \mathbb{P}_{\theta^*}(e_{j,a})}{\partial \theta_i^*}\big] = 0$. Secondly, the above lemma allows us to construct a vector-valued martingale and apply a generalization of Azuma-Hoeffding's tail bound on the norm to prove the following concentration of measure. This proves the desired bound on the gradient.

Lemma 3.11. *If n posets are independently drawn over d items from the PL model with weights θ^* then with probability at least $1 - 2e^3 d^{-3}$,*

$$\|\nabla \mathcal{L}_{\rm RB}(\theta^*)\| \le \gamma_2^{-1/2} e^b \sqrt{6np \log d} \, ,$$

where γ_2 depend on the choice of the rank-breaking and are defined in Section 3.4.1.

We will prove in (3.26) that the Hessian matrix $H(\theta) \in \mathcal{S}^d$ with $H_{ii'}(\theta) =$

$\frac{\partial^2 \mathcal{L}_{\mathrm{RB}}(\theta)}{\partial\theta_i \partial\theta_{i'}}$ can be expressed as

$$\begin{aligned} & -H(\theta) \\ = \; & \sum_{j=1}^{n} \sum_{a=1}^{\ell_j} \sum_{i<i' \in S_j} \mathbb{I}\{(i,i') \subseteq V(e_{j,a})\} \left(\frac{\partial^2 \log \mathbb{P}_\theta(e_{j,a})}{\partial\theta_i \partial\theta_{i'}} (e_i - e_{i'})(e_i - e_{i'})^\top \right) . \end{aligned}$$

It is easy to see that $H(\theta)\mathbf{1} = 0$. The following lemma proves a lower bound on the second smallest eigenvalue $\lambda_2(-H(\theta))$ in terms of re-scaled spectral gap α of the comparison graph $\mathcal{H}$ defined in Section 3.4.1.

Lemma 3.12. *Under the hypothesis of Theorem 4.5, if the assumptions in Equation* (3.15) *are satisfied then with probability at least* $1 - d^{-3}$*, the following holds for any* $\theta \in \Omega_b$*:*

$$\lambda_2(-H(\theta)) \;\; \geq \;\; \frac{e^{-6b}\alpha\gamma_1\gamma_2\gamma_3}{8} \frac{np}{(d-1)} ,$$

and $\lambda_1(-H(\theta)) = 0$ *with corresponding eigenvector* $\mathbf{1}$.

This finishes the proof of the desired claim.

3.6.3 Proof of Lemma 3.10

Recall that $e_{j,a}$ is a random event where randomness is in which items ended up in the top-set $T(e_{j,a})$ and the bottom-set $B(e_{j,a})$, and $\mathbb{P}_{\theta^*}(e_{j,a}) = \mathbb{P}_{\theta^*}[B(e_{j,a}) \prec T(e_{j,a})]$ that is the probability of observing $B(e_{j,a}) \prec T(e_{j,a})$ when the offer set is $B(e_{j,a}) \cup T(e_{j,a})$ as defined in (3.4). Define,

$$\mathbb{P}_{\theta^*, S_j}[e_{j,a} | V(e_{j,a}) = \mathcal{C}']$$

to be the conditional probability of observing $B(e_{j,a}) \prec T(e_{j,a})$, when the offer set is S_j, conditioned on the event that $V(e_{j,a}) = \mathcal{C}'$. Note that we have put subscript S_j in $\mathbb{P}_{\theta^*}$ to specify that the offer set is S_j. Observe that for any set $\mathcal{C}' \subseteq S_j$, the event $\{\mathcal{C}' = V(e_{j,a})\}$ is equivalent to $\mathcal{C}'$ items being ranked in bottom $r_{j,a}$ positions when the offer set is S_j. In other words, it is conditioned on the event that the subset $V(e_{j,a})$ items are ranked in bottom $r_{j,a}$ positions when the offer set is S_j. In Equation (3.23), we show that under

PL model

$$\mathbb{P}_{\theta^*,S_j}[e_{j,a}|V(e_{j,a})=\mathcal{C}'] = \mathbb{P}_{\theta^*}[e_{j,a}]\,.$$

Also, by conditioning on any outcome of $\{e_{j,a'}\}_{a'<a}$ it can be checked that

$$\mathbb{P}_{\theta^*,S_j}[e_{j,a}|V(e_{j,a})=\mathcal{C}',\{e_{j,a'}\}_{a'<a}] = \mathbb{P}_{\theta^*,S_j}[e_{j,a}|V(e_{j,a})=\mathcal{C}'].$$

Therefore, we have

$$\begin{aligned}
&\mathbb{E}\bigg[\frac{\partial \log \mathbb{P}_{\theta^*}\big[e_{j,a}\big]}{\partial\theta_i^*}\bigg|V(e_{j,a})=\mathcal{C}',\{e_{j,a'}\}_{a'<a}\bigg] \\
&= \mathbb{E}\bigg[\frac{\partial \log \mathbb{P}_{\theta^*,S_j}\big[e_{j,a}|V(e_{j,a})=\mathcal{C}',\{e_{j,a'}\}_{a'<a}\big]}{\partial\theta_i^*}\bigg|V(e_{j,a})=\mathcal{C}',\{e_{j,a'}\}_{a'<a}\bigg] \\
&= \sum_{\substack{e_{j,a}:V(e_{j,a})=\mathcal{C}' \\ \{e_{j,a'}\}_{a'<a}}} \mathbb{P}_{\theta^*,S_j}\Big[e_{j,a}\Big|V(e_{j,a})=\mathcal{C}',\{e_{j,a'}\}_{a'<a}\Big] \\
&\frac{\partial}{\partial\theta_i^*}\log \mathbb{P}_{\theta^*,S_j}\Big[e_{j,a}\Big|V(e_{j,a})=\mathcal{C}',\{e_{j,a'}\}_{a'<a}\Big] \\
&= \frac{\partial}{\partial\theta_i^*}\sum_{e_{j,a}:V(e_{j,a})=\mathcal{C}'} \mathbb{P}_{\theta^*,S_j}\Big[e_{j,a}\Big|V(e_{j,a})=\mathcal{C}'\Big] = \frac{\partial}{\partial\theta_i^*}1 = 0\,,
\end{aligned}$$

where we used $\{e_{j,a}:V(e_{j,a})=\mathcal{C}'\} = \{e_{j,a}:V(e_{j,a})=\mathcal{C}',\{e_{j,a'}\}_{a'<a}\}$ which follows from the definition of rank-breaking edges $e_{j,a}$. This proves the desired claim. It remains to show that

$$\mathbb{P}_{\theta^*,S_j}[e_{j,a}|V(e_{j,a})=\mathcal{C}'] = \mathbb{P}_{\theta^*}[e_{j,a}]\,.$$

This follows from the fact that under PL model for any disjoint set of items $\{\mathcal{C}_i\}_{i\in[\ell]}$ such that $\cup_{i=1}^{\ell}\mathcal{C}_i = S$,

$$\begin{aligned}
&\mathbb{P}\big(\mathcal{C}_\ell \prec \mathcal{C}_{\ell-1} \prec \cdots \prec \mathcal{C}_1\big) \\
&= \mathbb{P}\big(\mathcal{C}_\ell \prec \mathcal{C}_{\ell-1}\big)\mathbb{P}\big(\{\mathcal{C}_\ell,\mathcal{C}_{\ell-1}\} \prec \mathcal{C}_{\ell-2}\big)\cdots\mathbb{P}\big(\{\mathcal{C}_\ell,\mathcal{C}_{\ell-1},\cdots,\mathcal{C}_2\} \prec \mathcal{C}_1\big)\,, \quad (3.23)
\end{aligned}$$

where $\mathbb{P}(\mathcal{C}_{i_1} \prec \mathcal{C}_{i_2})$ is the probability that $\mathcal{C}_{i_2}$ items are ranked higher than $\mathcal{C}_{i_1}$ items when the offer set is $\{\mathcal{C}_{i_1} \cup \mathcal{C}_{i_2}\}$.

3.6.4 Proof of Lemma 3.11

We view $\nabla\mathcal{L}_{\rm RB}(\theta^*)$ as the final value of a discrete time vector-valued martingale with values in $\mathbb{R}^d$. Define $\nabla\mathcal{L}_{\rm RB}^{(e_{j,a})} \in \mathbb{R}^d$ as the gradient vector arising out of each rank-breaking edge $\{e_{j,a}\}_{j\in[n],a\in[\ell_j]}$ as

$$\nabla_i\mathcal{L}_{\rm RB}^{(e_{j,a})}(\theta^*) \;\; \equiv \;\; \sum_{\mathcal{C}\subseteq S_j} \mathbb{I}\big\{V(e_{j,a}) = \{\mathcal{C}, i\}\big\}\nabla_i \log \mathbb{P}_{\theta^*}(e_{j,a})\,,$$

such that $\nabla\mathcal{L}_{\rm RB}(\theta^*) = \sum_{j\in[n]}\sum_{a\in[\ell_j]}\nabla\mathcal{L}_{\rm RB}^{(e_{j,a})}$. We take $\nabla\mathcal{L}_{\rm RB}^{(e_{j,a})}$ as the incremental random vector in a martingale of $\sum_{j=1}^n \ell_j$ time steps. Let $H_{j,a}$ denote (the sigma algebra of) the history up to $e_{j,a}$ and define a sequence of random vectors in $\mathbb{R}^d$:

$$Z_{j,a} \;\; \equiv \;\; \mathbb{E}[\nabla\mathcal{L}_{\rm RB}^{(e_{j,a})}(\theta^*)|H_{j,a}]\,,$$

with the convention that $Z_{1,1} = \mathbb{E}[\nabla\mathcal{L}_{\rm RB}^{(e_{j,a})}(\theta^*)] = 0$ as proved in Lemma 3.10. It also follows from Lemma 3.10 that $\mathbb{E}[Z_{j,a+1}|Z_{j,a}] = Z_{j,a}$ for $a < \ell_j$. Also, from the independence of samples, it follows that $\mathbb{E}[Z_{j+1,1}|Z_{j,\ell_j}] = Z_{j,\ell_j}$. Applying a generalized version of the vector Azuma-Hoeffding inequality which readily follows from [Theorem 1.8, [90]], we have

$$\mathbb{P}\big[\,\|\nabla\mathcal{L}_{\rm RB}(\theta^*)\| \geq \delta\,\big] \;\; \leq \;\; 2e^3\exp\left(-\frac{\delta^2}{\sum_{j=1}^n\sum_{a=1}^{\ell_j} m_{j,a}2\gamma_2^{-1}e^{2b}}\right),$$

where we used $\|\nabla\mathcal{L}_{\rm RB}^{(e_{j,a})}\|^2 \leq m_{j,a}2\gamma_2^{-1}e^{2b}$. Choosing $\delta = \gamma_2^{-1}e^b\sqrt{6np\log d}$ gives the desired bound.

Now we are left to show that $\|\nabla\mathcal{L}_{\rm RB}^{(e_{j,a})}\|^2 \leq 2m_{j,a}\gamma_2^{-1}e^{2b}$ for any $\theta \in \Omega_b$. Recall that $\sigma \in \Lambda_{T(e_{j,a})}$ is the set of all full rankings over $T(e_{j,a})$ items. In rest of the proof, with a slight abuse of notations, we extend each of these ranking σ over $T(e_{j,a}) \cup B(e_{j,a})$ items in the following way. Consider any full ranking $\tilde{\sigma}$ over $B(e_{j,a})$ items. Then for each $\sigma \in \Lambda_{T(e_{j,a})}$, the extension is such that $\sigma(|T(e_{j,a})| + c) = \tilde{\sigma}(c)$ for $1 \leq c \leq |B(e_{j,a})|$. The choice of ranking $\tilde{\sigma}$ will have no impact on any of the following mathematical expressions. From the definition of $\mathbb{P}_\theta(e_{j,a})$ (3.4), we have, for any $i \in V(e_{j,a})$,

$$\begin{aligned}\frac{\partial \mathbb{P}_\theta(e_{j,a})}{\partial \theta_i} &= \mathbb{I}\{i \in T(e_{j,a})\}\mathbb{P}_\theta(e_{j,a}) \\ &- \underbrace{\sum_{\sigma\in\Lambda_{T(e_{j,a})}} \underbrace{\frac{\exp\left(\sum_{c=1}^{m_{j,a}}\theta_{\sigma(c)}\right)}{\prod_{u=1}^{m_{j,a}}\left(\sum_{c'=u}^{r_{j,a}}\exp\left(\theta_{\sigma(c')}\right)\right)}}_{\equiv A_\sigma} \underbrace{\left(\sum_{u'=1}^{m_{j,a}} \frac{\mathbb{I}\{\sigma^{-1}(i)\ge u'\}\exp(\theta_i)}{\sum_{c'=u'}^{r_{j,a}}\exp\left(\theta_{\sigma(c')}\right)}\right)}_{\equiv B_{\sigma,i}}}_{\equiv E_i}.\end{aligned} \tag{3.24}$$

Note that $A_\sigma, B_{\sigma,i}$ and E_i depend on $e_{j,a}$. Observe that for any $1 \le u' \le m_{j,a}$ and any $\sigma \in \Lambda_{T(e_{j,a})}$,

$$\sum_{i\in V(e_{j,a})} \mathbb{I}\{\sigma^{-1}(i)\ge u'\}\exp(\theta_i) = \sum_{c'=u'}^{r_{j,a}} \exp\left(\theta_{\sigma(c')}\right).$$

Therefore, $\sum_{i\in V(e_{j,a})} B_{\sigma,i} = m_{j,a}$. It follows that

$$\begin{aligned}\sum_{i\in V(e_{j,a})} E_i &= \sum_{\sigma\in\Lambda_{T(e_{j,a})}} A_\sigma \Bigg(\sum_{i\in V(e_{j,a})} B_{\sigma,i}\Bigg) \\ &= m_{j,a}\sum_{\sigma\in\Lambda_{T(e_{j,a})}} A_\sigma \;=\; m_{j,a}\mathbb{P}_\theta(e_{j,a}),\end{aligned} \tag{3.25}$$

where the last equality follows from the definition of $\mathbb{P}_\theta(e_{j,a})$ (7.12). Also, since for any i, i', $e^{(\theta_i-\theta_{i'})} \le e^{2b}$; for any i, $B_{\sigma,i} \le e^{2b}\sum_{k=r_{j,a}-m_{j,a}+1}^{r_{j,a}}(1/k) \le e^{2b}(1+\log(r_{j,a}/(r_{j,a}-m_{j,a}+1))) \le \gamma_2^{-1}e^{2b}$, where the last inequality follows from the definition of γ_2 (3.13) and the fact that $x \le \sqrt{1+\log x}$ for all $x \ge 1$. Therefore, $E_i \le \gamma_2^{-1}e^{2b}\sum_{\sigma\in\Lambda_{T(e_{j,a})}} A_\sigma = \gamma_2^{-1}e^{2b}\mathbb{P}_\theta(e_{j,a})$. We have $\partial \log \mathbb{P}_\theta(e_{j,a})/\partial\theta_i = (1/\mathbb{P}_\theta(e_{j,a}))\partial\mathbb{P}_\theta(e_{j,a})/\partial\theta_i = \mathbb{I}\{i \in T(e_{j,a})\} - E_i/\mathbb{P}_\theta(e_{j,a})$. Since $|T(e_{j,a})| = m_{j,a}$,

$$\|\nabla\mathcal{L}_{\text{RB}}^{(e_{j,a})}\|^2 \le m_{j,a} + \sum_{i\in V(e_{j,a})} (E_i/\mathbb{P}_\theta(e_{j,a}))^2 \le 2m_{j,a}\gamma_2^{-1}e^{2b},$$

where we used (3.25) and the fact that $\gamma_2^{-1} \ge 1$.

Proof of Lemma 3.12

First, we prove (3.23). For brevity, remove $\{j,a\}$ from $\mathbb{P}_\theta(e_{j,a})$. From Equations (3.24) and (3.25), and $|T(e_{j,a})| = m_{j,a}$, we have $\sum_{i\in V(e_{j,a})} \frac{\partial}{\partial\theta_i}\mathbb{P}_\theta(e) = m_{j,a}\mathbb{P}_\theta(e) - m_{j,a}\mathbb{P}_\theta(e) = 0$. It follows that

$$\sum_{i\in V(e_{j,a})} \left(\frac{\partial^2 \log \mathbb{P}_\theta(e)}{\partial\theta_{i'}\partial\theta_i}\right) =$$
$$\frac{1}{\mathbb{P}_\theta(e)}\frac{\partial}{\partial\theta_{i'}}\left(\sum_{i\in V(e_{j,a})}\left(\frac{\partial\mathbb{P}_\theta(e)}{\partial\theta_i}\right)\right) - \frac{1}{(\mathbb{P}_\theta(e))^2}\frac{\partial\mathbb{P}_\theta(e)}{\partial\theta_{i'}}\left(\sum_{i\in V(e_{j,a})}\left(\frac{\partial\mathbb{P}_\theta(e)}{\partial\theta_i}\right)\right) = 0\,. \tag{3.26}$$

Since by definition $\mathcal{L}_{\text{RB}}(\theta) = \sum_{j=1}^{n}\sum_{a=1}^{\ell_j}\log\mathbb{P}_\theta(e_{j,a})$, and $H_{ii'}(\theta) = \frac{\partial^2\mathcal{L}_{\text{RB}}(\theta)}{\partial\theta_i\partial\theta_{i'}}$ which is a symmetric matrix, Equation (3.26) implies that it can be expressed as given in Equation (3.23). It follows that all-ones is an eigenvector of $H(-\theta)$ with the corresponding eigenvalue being zero.

To get a lower bound on $\lambda_2(-H(\theta))$, we apply Weyl's inequality

$$\lambda_2(-H(\theta)) \geq \lambda_2(\mathbb{E}[-H(\theta)]) - \|H(\theta) - \mathbb{E}[H(\theta)]\|\,.$$

We will show in (3.27) that $\lambda_2(\mathbb{E}[-H(\theta)]) \geq e^{-6b}\alpha\gamma_1\gamma_2\gamma_3(np/(4(d-1)))$ and in (3.38) that $\|H(\theta) - \mathbb{E}[H(\theta)]\| \leq 16e^{4b}\nu\sqrt{\frac{p_{\max}}{\kappa_{\min}}\frac{np}{\beta(d-1)}\log d}$. Putting these together,

$$\begin{aligned}\lambda_2(-H(\theta)) &\geq e^{-6b}\alpha\gamma_1\gamma_2\gamma_3\frac{np}{4(d-1)} - 16e^{4b}\nu\sqrt{\frac{p_{\max}}{\kappa_{\min}}\frac{np}{\beta(d-1)}\log d}\\ &\geq \frac{e^{-6b}\alpha\gamma_1\gamma_2\gamma_3}{8}\frac{np}{(d-1)}\,,\end{aligned}$$

where the last inequality follows from the assumption on $n\kappa_{\min}$ given in (3.15).

To prove a lower bound on $\lambda_2(\mathbb{E}[-H(\theta)])$, we claim that for $\theta\in\Omega_b$,

$$\begin{aligned}\mathbb{E}\big[-H(\theta)\big] &\succeq e^{-6b}\gamma_1\gamma_2\gamma_3\sum_{j=1}^{n}\frac{p_j}{4\kappa_j(\kappa_j-1)}\sum_{i<i'\in S_j}(e_i-e_{i'})(e_i-e_{i'})^\top \qquad (3.27)\\ &= \frac{e^{-6b}\gamma_1\gamma_2\gamma_3}{4}L\,,\end{aligned}$$

where $L \in \mathcal{S}^d$ is defined in (3.11). Using $\lambda_2(L) = np\alpha/(d-1)$ from (3.12), we have $\lambda_2(-H(\theta)) \geq e^{-6b}\alpha\gamma_1\gamma_2\gamma_3(np/(4(d-1)))$. To prove (3.27), notice that

$$\mathbb{E}[-H(\theta)_{ii'}] \;=\; \mathbb{E}\Big[\sum_{j\in[n]}\sum_{a\in[\ell_j]} \mathbb{I}\big\{(i,i') \subseteq V(e_{j,a})\big\}\,\frac{\partial^2 \log \mathbb{P}_\theta(e_{j,a})}{\partial\theta_i \partial\theta_{i'}}\Big]\;, \tag{3.28}$$

when $i \neq i'$. We will show that for any $i \neq i' \in V(e_{j,a})$,

$$\frac{\partial^2 \log \mathbb{P}_\theta(e_{j,a})}{\partial\theta_i \partial\theta_{i'}} \;\geq\; \begin{cases} \frac{e^{-2b} m_{j,a}}{r_{j,a}^2} & \text{if } i, i' \in B(e_{j,a}) \\ -\frac{e^{4b} m_{j,a}^2}{(r_{j,a}-m_{j,a}+1)^2} & \text{otherwise}\,. \end{cases} \tag{3.29}$$

We need to bound the probability of two items appearing in the bottom-set $B(e_{j,a})$ and in the top-set $T(e_{j,a})$.

Lemma 3.13. *Consider a ranking σ over a set $S \subseteq [d]$ such that $|S| = \kappa$. For any two items $i, i' \in S$, $\theta \in \Omega_b$, and $1 \leq \ell, \ell_1, \ell_2 \leq \kappa - 1$,*

$$\mathbb{P}_\theta\big[\sigma^{-1}(i), \sigma^{-1}(i') > \ell\big] \;\geq\; \frac{e^{-4b}(\kappa-\ell)(\kappa-\ell-1)}{\kappa(\kappa-1)}\left(1-\frac{\ell}{\kappa}\right)^{2e^{2b}-2} \tag{3.30}$$

$$\mathbb{P}_\theta\big[\sigma^{-1}(i) = \ell\big] \;\leq\; \frac{e^{6b}}{\kappa-\ell}\;, \tag{3.31}$$

$$\mathbb{P}_\theta\big[\sigma^{-1}(i) = \ell_1, \sigma^{-1}(i') = \ell_2\big] \;\leq\; \frac{e^{10b}}{(\kappa-\ell_1-1)(\kappa-\ell_2)}\,. \tag{3.32}$$

where the probability $\mathbb{P}_\theta$ is with respect to the sampled ranking resulting from PL weights $\theta \in \Omega_b$.

Substituting $\ell = \kappa_j - r_{j,a} + m_{j,a}$ in (3.30), and $\ell, \ell_1, \ell_2 \leq \kappa_j - r_{j,a} + m_{j,a}$ in

(3.31) and (3.32), we have,

$$\begin{aligned}
&\mathbb{P}_\theta\big[(i,i') \subseteq B(e_{j,a})\big] \\
\geq\;& \frac{e^{-4b}(r_{j,a}-m_{j,a})^2}{4\kappa_j(\kappa_j-1)}\Big(\frac{r_{j,a}-m_{j,a}}{\kappa_j}\Big)^{2e^{2b}-2}, && (3.33)\\
&\mathbb{P}_\theta\big[i \in T(e_{j,a}), i' \in B(e_{j,a})\big] \\
\leq\;& m_{j,a} \max_{\ell \in [\kappa_j - r_{j,a}+m_{j,a}]} \mathbb{P}(\sigma^{-1}(i)=\ell) \\
\leq\;& \frac{e^{6b}m_{j,a}}{r_{j,a}-m_{j,a}}, && (3.34)\\
&\mathbb{P}_\theta\big[(i,i') \subseteq T(e_{j,a})\big] \\
\leq\;& m_{j,a}^2 \max_{\ell_1,\ell_2 \in [\kappa_j - r_{j,a}+m_{j,a}]} \mathbb{P}(\sigma^{-1}(i)=\ell_1, \sigma^{-1}(i')=\ell_2) \\
\leq\;& \frac{e^{10b}m_{j,a}^2}{2\,(r_{j,a}-m_{j,a}-1)(r_{j,a}-m_{j,a})}, && (3.35)
\end{aligned}$$

where (3.33) uses $r_{j,a}-m_{j,a}-1 \geq (r_{j,a}-m_{j,a})/4$, (3.34) uses $\mathbb{P}_\theta[i \in T(e_{j,a}), i' \in B(e_{j,a})] \leq \mathbb{P}_\theta[i \in T(e_{j,a})]$, and (3.34)-(3.35) uses counting on the possible choices. The bound in (3.35) is smaller than the one in (3.34) as per our assumption that $\gamma_3 > 0$.

Using Equations (3.28)-(3.29) and (3.33)-(3.35), and the definitions of $\gamma_1, \gamma_2, \gamma_3$ from Section 3.4.1, we get

$$\begin{aligned}
&\mathbb{E}[-H(\theta)_{ii'}] \geq \\
&\sum_{j\in[n]}\sum_{a\in[\ell_j]}\Big\{\underbrace{\Big(\frac{r_{j,a}-m_{j,a}}{\kappa_j}\Big)^{2e^{2b}-2}}_{\geq\gamma_1}\underbrace{\Big(\frac{r_{j,a}-m_{j,a}}{r_{j,a}}\Big)^2}_{\geq\gamma_2} \\
&\frac{e^{-6b}m_{j,a}}{4\kappa_j(\kappa_j-1)} \;-\; \frac{e^{6b}m_{j,a}}{r_{j,a}-m_{j,a}}\,\frac{e^{4b}m_{j,a}^2}{(r_{j,a}-m_{j,a}+1)^2}\Big\} \\
&\geq \sum_{j,a}\frac{\gamma_1\gamma_2 e^{-6b}m_{j,a}}{4\kappa_j(\kappa_j-1)}\;\underbrace{\Big(1 \;-\; \frac{4e^{16b}}{\gamma_1}\,\frac{m_{j,a}^2 r_{j,a}^2 \kappa_j^2}{(r_{j,a}-m_{j,a})^5}\Big)}_{\geq\gamma_3}\;.
\end{aligned}$$

This combined with (3.23) proves the desired claim (3.27). Further, in Appendix 3.6.7, we show that if $m_{j,a} \leq 3$ for all $\{j,a\}$ then $\frac{\partial^2 \log \mathbb{P}_\theta(e_{j,a})}{\partial\theta_i \partial\theta_{i'}}$ is non-negative even for $i \neq i' \in T(e_{j,a})$, and $i \in T(e_{j,a}), i' \in B(e_{j,a})$ as opposed to a negative lower-bound given in (3.29). Therefore, bound on $\mathbb{E}[-H(\theta)]$ in (3.27) can be tightened by a factor of γ_3.

To prove claim (3.29), define the following for $\sigma \in \Lambda_{T(e_{j,a})}$,

$$\begin{aligned}
&A_\sigma \equiv \frac{\exp\left(\sum_{c=1}^{m_{j,a}} \theta_{\sigma(c)}\right)}{\prod_{u=1}^{m_{j,a}} \left(\sum_{c'=u}^{r_{j,a}} \exp\left(\theta_{\sigma(c')}\right)\right)}\,, B_\sigma \equiv \sum_{u'=1}^{m_{j,a}} \frac{1}{\sum_{c'=u'}^{r_{j,a}} \exp\left(\theta_{\sigma(c')}\right)}\,,\\
&B_{\sigma,i} \equiv \sum_{u'=1}^{m_{j,a}} \frac{\mathbb{I}\{\sigma^{-1}(i) \geq u'\}}{\sum_{c'=u'}^{r_{j,a}} \exp\left(\theta_{\sigma(c')}\right)}\,, C_\sigma \equiv \sum_{u'=1}^{m_{j,a}} \frac{1}{\left(\sum_{c'=u'}^{r_{j,a}} \exp\left(\theta_{\sigma(c')}\right)\right)^2}\,,\\
&C_{\sigma,i} \equiv \sum_{u'=1}^{m_{j,a}} \frac{\mathbb{I}\{\sigma^{-1}(i) \geq u'\}}{\left(\sum_{c'=u'}^{r_{j,a}} \exp\left(\theta_{\sigma(c')}\right)\right)^2}\,, C_{\sigma,i,i'} \equiv \sum_{u'=1}^{m_{j,a}} \frac{\mathbb{I}\{\sigma^{-1}(i), \sigma^{-1}(i') \geq u'\}}{\left(\sum_{c'=u'}^{r_{j,a}} \exp\left(\theta_{\sigma(c')}\right)\right)^2}\,.
\end{aligned} \tag{3.36}$$

First, a few observations about the expression of A_σ. For any $\sigma \in \Lambda_{T(e_{j,a})}$ and any $i \in V(e_{j,a})$, θ_i is in the numerator if and only if $i \in T(e_{j,a})$, since in all the rankings that are consistent with the observation $e_{j,a}$, $T(e_{j,a})$ items are ranked in top $m_{j,a}$ positions. For any $\sigma \in \Lambda_{T(e_{j,a})}$ and any $i \in B(e_{j,a})$, θ_i is in all the product terms $\prod_{u=1}^{m_{j,a}}(\cdot)$ of the denominator, since in all the consistent rankings these items are ranked below $m_{j,a}$ position. For any $i \in T(e_{j,a})$, θ_i appears in product term corresponding to index u if and only if item i is ranked at position u or lower than u in the ranking $\sigma \in \Lambda_{T(e_{j,a})}$. Now, observe that B_σ is defined such that the partial derivative of A_σ with respect to any $i \in B(e_{j,a})$ is $-A_\sigma B_\sigma e^{\theta_i}$, and $B_{\sigma,i}$ is defined such that the partial derivative of A_σ with respect to any $i \in T(e_{j,a})$ is $A_\sigma - A_\sigma B_\sigma e^{\theta_i}$. Further, observe that $-C_\sigma e^{\theta_i}$ is the partial derivative of B_σ with respect to $i \in B(e_{j,a})$, $-C_{\sigma,i}e^{\theta_i}$ is the partial derivative of $B_{\sigma,i}$ with respect to $i \in T(e_{j,a})$, and $-C_{\sigma,i}e^{\theta_{i'}}$ is the partial derivative of $B_{\sigma,i}$ with respect to $i' \in B(e_{j,a})$. $-C_{\sigma,i,i'}e^{\theta_{i'}}$ is the partial derivative of $B_{\sigma,i}$ with respect to $i' \neq i \in T(e_{j,a})$.

For ease of notation, we omit subscript (j,a) whenever it is clear from the context. Also, we use $\sum_\sigma$ to denote $\sum_{\sigma \in \Lambda_{T(e_{j,a})}}$. With the above defined notations, from (7.12), we have, $\mathbb{P}_\theta(e) = \sum_\sigma A_\sigma$. With the above given observations for the notations in (3.36), first partial derivative of $\mathbb{P}_\theta(e)$ can be expressed as following:

$$\frac{\partial \mathbb{P}_\theta(e)}{\partial \theta_i} = \begin{cases} \sum_\sigma \left(A_\sigma - A_\sigma B_{\sigma,i} e^{\theta_i}\right) & \text{if } \ i \in T(e_{j,a}) \\ \sum_\sigma \left(-A_\sigma B_\sigma e^{\theta_i}\right) & \text{if } \ i \in B(e_{j,a})\,. \end{cases} \tag{3.37}$$

It follows that for $i \neq i' \in V(e_{j,a})$,

$$\frac{\partial^2 \mathbb{P}_\theta(e)}{\partial\theta_i\partial\theta_{i'}}$$

$$= \begin{cases} \sum_\sigma \big((A_\sigma(B_\sigma)^2 + A_\sigma C_\sigma)e^{(\theta_i+\theta_{i'})}\big) \\ \text{if } i,i' \in B(e_{j,a}) \\ \sum_\sigma \big(A_\sigma - A_\sigma B_{\sigma,i'}e^{\theta_{i'}} + (A_\sigma B_{\sigma,i}B_{\sigma,i'} + A_\sigma C_{\sigma,i,i'})e^{(\theta_i+\theta_{i'})} - A_\sigma B_{\sigma,i}e^{\theta_i}\big) \\ \text{if } i,i' \in T(e_{j,a}) \\ \sum_\sigma \big((A_\sigma B_\sigma B_{\sigma,i} + A_\sigma C_{\sigma,i})e^{(\theta_i+\theta_{i'})} - A_\sigma B_\sigma e^{\theta_{i'}}\big) \\ \text{otherwise}\,. \end{cases}$$

Using $\frac{\partial^2 \log \mathbb{P}_\theta(e)}{\partial\theta_i\partial\theta_{i'}} = \frac{1}{\mathbb{P}_\theta(e)}\frac{\partial^2\mathbb{P}_\theta(e)}{\partial\theta_i\partial\theta_{i'}} - \frac{1}{(\mathbb{P}_\theta(e))^2}\frac{\partial\mathbb{P}_\theta(e)}{\partial\theta_i}\frac{\partial\mathbb{P}_\theta(e)}{\partial\theta_{i'}}$, with above derived first and second derivatives, and after following some algebra, we have

$$\frac{(\mathbb{P}_\theta(e))^2}{e^{(\theta_i+\theta_{i'})}}\frac{\partial^2 \log \mathbb{P}_\theta(e)}{\partial\theta_i\partial\theta_{i'}}$$

$$= \begin{cases} (\sum_\sigma A_\sigma)(\sum_\sigma A_\sigma(B_\sigma)^2) - (\sum_\sigma A_\sigma B_\sigma)^2 + (\sum_\sigma A_\sigma)(\sum_\sigma A_\sigma C_\sigma) \\ \text{if } \; i,i' \in B(e_{j,a}) \\ (\sum_\sigma A_\sigma)(\sum_\sigma A_\sigma B_{\sigma,i}B_{\sigma,i'} + A_\sigma C_{\sigma,i,i'}) - (\sum_\sigma A_\sigma B_{\sigma,i})(\sum_\sigma A_\sigma B_{\sigma,i'}) \\ \text{if } \; i,i' \in T(e_{j,a}) \\ (\sum_\sigma A_\sigma)(\sum_\sigma A_\sigma B_\sigma B_{\sigma,i} + A_\sigma C_{\sigma,i}) - (\sum_\sigma A_\sigma B_\sigma)(\sum_\sigma A_\sigma B_{\sigma,i}) \\ \text{otherwise} \end{cases}$$

Observe that from Cauchy-Schwartz inequality

$$(\sum_\sigma A_\sigma)(\sum_\sigma A_\sigma(B_\sigma)^2) - (\sum_\sigma A_\sigma B_\sigma)^2 \geq 0\,.$$

Also, we have $e^{(\theta_i+\theta_{i'})}C_\sigma \geq e^{-2b}(m/r^2)$ and $e^{\theta_i}B_{\sigma,i} \leq e^{\theta_i}B_\sigma \leq e^{2b}(m/(r-m+1))$ for any $i \in V(e_{j,a})$. This proves the desired claim (3.29).

Next we need to upper bound deviation of $-H(\theta)$ from its expectation. From above equations, we have, $\left|\frac{\partial^2 \log \mathbb{P}_\theta(e_{j,a})}{\partial\theta_i\partial\theta_{i'}}\right| \leq 3e^{4b}m_{j,a}^2/(r_{j,a}-m_{j,a}+1)^2 \leq 3e^{4b}\nu m_{j,a}/(\kappa_j(\kappa_j-1))$, where the last inequality follows from the definition of ν (3.14). Therefore,

$$\begin{aligned}
&-H(\theta) \\
\preceq\;& 3e^{4b}\nu \sum_{j=1}^{n} \sum_{a=1}^{\ell_j} \sum_{i<i' \in S_j} \mathbb{I}\{(i,i') \subseteq V(e_{j,a})\} \frac{m_{j,a}}{\kappa_j(\kappa_j - 1)} (e_i - e_{i'})(e_i - e_{i'})^\top \\
\preceq\;& 3e^{4b}\nu \sum_{j=1}^{n} \sum_{i<i' \in S_j} \frac{\sum_{a=1}^{\ell_j} m_{j,a}}{\kappa_j(\kappa_j - 1)} (e_i - e_{i'})(e_i - e_{i'})^\top \equiv \sum_{j=1}^{n} y_j L_j \,,
\end{aligned}$$

where $y_j = (3e^{4b}\nu p_j)/(\kappa_j(\kappa_j - 1))$ and $L_j = \sum_{i<i' \in S_j} (e_i - e_{i'})(e_i - e_{i'})^\top = \kappa_j \mathrm{diag}(e_{S_j}) - e_{S_j} e_{S_j}^\top$ for $e_{S_j} = \sum_{i \in S_j} e_i$. Observe that $\|y_j L_j\| \le (3e^{4b}\nu p_{\max})/\kappa_{\min}$. Moreover, $L_j^2 \preceq \kappa_j L_j$, and it follows that

$$\sum_{j=1}^{n} y_j^2 L_j^2 \;\preceq\; 9e^{8b}\nu^2 \sum_{j=1}^{n} \frac{p_j^2}{\kappa_j^2(\kappa_j - 1)^2} \kappa_j L_j \preceq \frac{9e^{8b}\nu^2 p_{\max}}{\kappa_{\min}} L \,,$$

where we used the fact that $L = (p_j/(\kappa_j(\kappa_j - 1))) \sum_{j=1}^{n} L_j$, for L defined in (3.11). Using $\lambda_d(L) = np/(\beta(d-1))$ from (3.12), it follows that

$$\| \sum_{j=1}^{n} \mathbb{E}_\theta[y_j^2 Y_j^2] \| \le \frac{9e^{8b}\nu^2 p_{\max}}{\kappa_{\min}} \frac{np}{\beta(d-1)} \,.$$

By the matrix Bernstien inequality, with probability at least $1 - d^{-3}$,

$$\begin{aligned}
\|H(\theta) - \mathbb{E}[H(\theta)]\| \;&\le\; 12e^{4b}\nu \sqrt{\frac{p_{\max}}{\kappa_{\min}} \frac{np}{\beta(d-1)} \log d} + \frac{8e^{4b}\nu p_{\max} \log d}{\kappa_{\min}} \\
&\le\; 16e^{4b}\nu \sqrt{\frac{p_{\max}}{\kappa_{\min}} \frac{np}{\beta(d-1)} \log d} \,, \qquad (3.38)
\end{aligned}$$

where the last inequality follows from the assumption on $n\kappa_{\min}$ given in (3.15).

3.6.5 Proof of Lemma 3.13

Claim (3.30)**:** Since providing a lower bound on $\mathbb{P}_\theta\big[\sigma^{-1}(i), \sigma^{-1}(i') > \ell\big]$ for arbitrary θ is challenging, we construct a new set of parameters $\{\widetilde{\theta}_j\}_{j \in [d]}$ from the original θ. These new parameters are constructed such that it is both easy to compute the probability and also provides a lower bound on the original distribution. Define $\widetilde{\alpha}_{i,i',\ell,\theta}$ as

$$\widetilde{\alpha}_{i,i',\ell,\theta} \;\; \equiv \;\; \max_{\ell' \in [\ell]} \max_{\substack{\Omega \subseteq S \setminus \{i,i'\} \\ :|\Omega| = \kappa - \ell'}} \left\{ \frac{\exp(\theta_i) + \exp(\theta_{i'})}{\big(\sum_{j \in \Omega} \exp(\theta_j)\big)/|\Omega|} \right\} , \tag{3.39}$$

and $\alpha_{i,i',\ell,\theta} = \lceil \widetilde{\alpha}_{i,i',\ell,\theta} \rceil$. For ease of notation we remove the subscript from α and $\widetilde{\alpha}$. We denote the sum of the weights by $W \equiv \sum_{j \in S} \exp(\theta_j)$. We define a new set of parameters $\{\widetilde{\theta}_j\}_{j \in S}$:

$$\widetilde{\theta}_j \;\; = \;\; \begin{cases} \log(\widetilde{\alpha}/2) & \text{for } j = i \text{ or } i' \, , \\ 0 & \text{otherwise} \, . \end{cases}$$

Similarly define $\widetilde{W} \equiv \sum_{j \in S} \exp(\widetilde{\theta}_j) = \kappa - 2 + \widetilde{\alpha}$. We have,

$$\begin{aligned} & \mathbb{P}_\theta \Big[\sigma^{-1}(i), \sigma^{-1}(i') > \ell \Big] \\ = \; & \sum_{\substack{j_1 \in S \\ j_1 \neq i,i'}} \left(\frac{\exp(\theta_{j_1})}{W} \sum_{\substack{j_2 \in S \\ j_2 \neq i,i',j_1}} \left(\frac{\exp(\theta_{j_2})}{W - \exp(\theta_{j_1})} \cdots \right.\right. \\ & \left.\left. \left(\sum_{\substack{j_\ell \in S \\ j_\ell \neq i,i', \\ j_1,\cdots,j_{\ell-1}}} \frac{\exp(\theta_{j_\ell})}{W - \sum_{k=j_1}^{j_{\ell-1}} \exp(\theta_k)} \right) \cdots \right) \right) \\ = \; & \sum_{\substack{j_1 \in S \\ j_1 \neq i,i'}} \left(\frac{\exp(\theta_{j_1})}{W - \exp(\theta_{j_1})} \cdots \right. \\ & \left. \sum_{\substack{j_{\ell-1} \in S \\ j_{\ell-1} \neq i,i', \\ j_1,\cdots,j_{\ell-2}}} \left(\frac{\exp(\theta_{j_{\ell-1}})}{W - \sum_{k=j_1}^{j_{\ell-1}} \exp(\theta_k)} \sum_{\substack{j_\ell \in S \\ j_\ell \neq i,i', \\ j_1,\cdots,j_{\ell-1}}} \left(\frac{\exp(\theta_{j_\ell})}{W} \right) \cdots \right) \right) \end{aligned} \tag{3.40}$$

Consider the second-last summation term in the above equation and let $\Omega_\ell = S \setminus \{i, i', j_1, \ldots, j_{\ell-2}\}$. Observe that, $|\Omega_\ell| = \kappa - \ell$ and from equation (3.39),

$\frac{\exp(\theta_i)+\exp(\theta_{i'})}{\sum_{j\in\Omega_\ell}\exp(\theta_j)} \le \frac{\widetilde{\alpha}}{\kappa-\ell}$. We have,

$$
\begin{aligned}
& \sum_{j_{\ell-1}\in\Omega_\ell} \frac{\exp(\theta_{j_{\ell-1}})}{W-\sum_{k=j_1}^{j_{\ell-1}}\exp(\theta_k)} \\
= & \sum_{j_{\ell-1}\in\Omega_\ell} \frac{\exp(\theta_{j_{\ell-1}})}{W-\sum_{k=j_1}^{j_{\ell-2}}\exp(\theta_k)-\exp(\theta_{j_{\ell-1}})} \\
\ge & \ \frac{\sum_{j_{\ell-1}\in\Omega_\ell}\exp(\theta_{j_{\ell-1}})}{W-\sum_{k=j_1}^{j_{\ell-2}}\exp(\theta_k)-\big(\sum_{j_{\ell-1}\in\Omega_\ell}\exp(\theta_{j_{\ell-1}})\big)/|\Omega_\ell|} \qquad (3.41)\\
= & \ \frac{\sum_{j_{\ell-1}\in\Omega_\ell}\exp(\theta_{j_{\ell-1}})}{\exp(\theta_i)+\exp(\theta_{i'})+\sum_{j_{\ell-1}\in\Omega_\ell}\exp(\theta_{j_{\ell-1}})-\big(\sum_{j_{\ell-1}\in\Omega_\ell}\exp(\theta_{j_{\ell-1}})\big)/|\Omega_\ell|} \\
= & \left(\frac{\exp(\theta_i)+\exp(\theta_{i'})}{\sum_{j_{\ell-1}\in\Omega_\ell}\exp(\theta_{j_{\ell-1}})}+1-\frac{1}{\kappa-\ell}\right)^{-1} \\
\ge & \left(\frac{\widetilde{\alpha}}{\kappa-\ell}+1-\frac{1}{\kappa-\ell}\right)^{-1} \qquad (3.42)\\
= & \ \frac{\kappa-\ell}{\widetilde{\alpha}+\kappa-\ell-1} = \sum_{j_{\ell-1}\in\Omega_\ell} \frac{\exp(\widetilde{\theta}_{j_{\ell-1}})}{\widetilde{W}-\sum_{k=j_1}^{j_{\ell-1}}\exp(\widetilde{\theta}_k)}\ , \qquad (3.43)
\end{aligned}
$$

where (3.41) follows from the Jensen's inequality and the fact that for any $c > 0$, $0 < x < c$, $\frac{x}{c-x}$ is convex in x. Equation (3.42) follows from the definition of $\widetilde{\alpha}_{i,i',\ell,\theta}$, (3.39), and the fact that $|\Omega_\ell| = \kappa - \ell$. Equation (3.43) uses the definition of $\{\widetilde{\theta}_j\}_{j\in S}$.

Consider $\{\Omega_{\widetilde{\ell}}\}_{2\le\widetilde{\ell}\le\ell-1}$, $|\Omega_{\widetilde{\ell}}| = \kappa - \widetilde{\ell}$, corresponding to the subsequent summation terms in (3.40). Observe that $\frac{\exp(\theta_i)+\exp(\theta_{i'})}{\sum_{j\in\Omega_{\widetilde{\ell}}}\exp(\theta_j)} \le \alpha/|\Omega_{\widetilde{\ell}}|$. Therefore, each summation term in equation (3.40) can be lower bounded by the corresponding term where $\{\theta_j\}_{j\in S}$ is replaced by $\{\widetilde{\theta}_j\}_{j\in S}$. Hence, we have

$$\begin{aligned}
&\mathbb{P}_\theta\Big[\sigma^{-1}(i),\sigma^{-1}(i') > \ell\Big] \\
\geq\;& \sum_{\substack{j_1\in S\\ j_1\neq i,i'}}\left(\frac{\exp(\widetilde{\theta}_{j_1})}{\widetilde{W}-\exp(\widetilde{\theta}_{j_1})}\cdots\right. \\
&\sum_{\substack{j_{\ell-1}\in S\\ j_{\ell-1}\neq i,i',\\ j_1,\cdots,j_{\ell-2}}}\left(\frac{\exp(\widetilde{\theta}_{j_{\ell-1}})}{\widetilde{W}-\sum_{k=j_1}^{j_{\ell-1}}\exp(\widetilde{\theta}_k)}\sum_{\substack{j_\ell\in S\\ j_\ell\neq i,i',\\ j_1,\cdots,j_{\ell-1}}}\left.\left.\left(\frac{\exp(\theta_{j_\ell})}{W}\right)\cdots\right)\right) \\
\geq\;& e^{-4b}\sum_{\substack{j_1\in S\\ j_1\neq i,i'}}\left(\frac{\exp(\widetilde{\theta}_{j_1})}{\widetilde{W}-\exp(\widetilde{\theta}_{j_1})}\cdots\right. \\
&\sum_{\substack{j_{\ell-1}\in S\\ j_{\ell-1}\neq i,i',\\ j_1,\cdots,j_{\ell-2}}}\left(\frac{\exp(\widetilde{\theta}_{j_{\ell-1}})}{\widetilde{W}-\sum_{k=j_1}^{j_{\ell-1}}\exp(\widetilde{\theta}_k)}\sum_{\substack{j_\ell\in S\\ j_\ell\neq i,i',\\ j_1,\cdots,j_{\ell-1}}}\left.\left.\left(\frac{\exp(\widetilde{\theta}_{j_\ell})}{\widetilde{W}}\right)\cdots\right)\right) \\
=\;& \big(e^{-4b}\big)\mathbb{P}_{\widetilde{\theta}}\Big[\sigma^{-1}(i),\sigma^{-1}(i') > \ell\Big]\,. \qquad (3.44)
\end{aligned}$$

The second inequality uses $\frac{\exp(\theta_i)}{W} \geq e^{-2b}/\kappa$ and $\frac{\exp(\widetilde{\theta}_i)}{\widetilde{W}} \leq e^{2b}/\kappa$. Observe that $\exp(\widetilde{\theta}_j) = 1$ for all $j \neq i, i'$ and $\exp(\widetilde{\theta}_i) + \exp(\widetilde{\theta}_{i'}) = \widetilde{\alpha} \leq \lceil\widetilde{\alpha}\rceil = \alpha \geq 1$. Therefore, we have

$$\begin{aligned}
&\mathbb{P}_{\widetilde{\theta}}\Big[\sigma^{-1}(i),\sigma^{-1}(i') > \ell\Big] \\
=\;& \binom{\kappa-2}{\ell}\frac{\ell\,!}{(\kappa-2+\widetilde{\alpha})(\kappa-2+\widetilde{\alpha}-1)\cdots(\kappa-2+\widetilde{\alpha}-(\ell-1))} \\
\geq\;& \frac{(\kappa-2)!}{(\kappa-\ell-2)!}\frac{1}{(\kappa+\alpha-2)(\kappa+\alpha-3)\cdots(\kappa+\alpha-(\ell+1))} \\
\geq\;& \frac{(\kappa-\ell+\alpha-2)(\kappa-\ell+\alpha-3)\cdots(\kappa-\ell-1)}{(\kappa+\alpha-2)(\kappa+\alpha-3)\cdots(\kappa-1)} \\
\geq\;& \frac{(\kappa-\ell)(\kappa-\ell-1)}{\kappa(\kappa-1)}\left(1-\frac{\ell}{\kappa+1}\right)^{\alpha-2}. \qquad (3.45)
\end{aligned}$$

Claim (3.30) follows by combining Equations (3.44) and (3.45) and using the fact that $\alpha \leq 2e^{2b}$.

Claim (3.31)**:** Define,

$$\widetilde{\alpha}_{\ell,\theta} \;\; \equiv \;\; \min_{i\in S} \min_{\ell'\in[\ell]} \min_{\substack{\Omega\in S\setminus\{i\} \\ :|\Omega|=\kappa-\ell'+1}} \left\{ \frac{\exp(\theta_i)}{\big(\sum_{j\in\Omega}\exp(\theta_j)\big)/|\Omega|} \right\} . \qquad (3.46)$$

Also, define $\alpha_{\ell,\theta} \equiv \lfloor \widetilde{\alpha}_{\ell,\theta} \rfloor$. Note that $\alpha_{\ell,\theta} \geq 0$ and $\widetilde{\alpha}_{\ell,\theta} \leq e^{2b}$. We denote the sum of the weights by $W \equiv \sum_{j\in S}\exp(\theta_j)$. Analogous to the proof of claim (3.30), we define the new set of parameters $\{\widetilde{\theta}_j\}_{j\in S}$:

$$\widetilde{\theta}_j \;\; = \;\; \begin{cases} \log(\widetilde{\alpha}_{\ell,\theta}) & \text{for } j = i \;, \\ \hfill 0 & \text{otherwise} \;. \end{cases}$$

Similarly define $\widetilde{W} \equiv \sum_{j\in S}\exp(\widetilde{\theta}_j) \;=\; \kappa - 1 + \widetilde{\alpha}_{\ell,\theta}$. Using the techniques similar to the ones used in proof of claim (3.30), we have,

$$\mathbb{P}_\theta\Big[\sigma^{-1}(i) = \ell\Big] \;\; \leq \;\; e^{4b}\mathbb{P}_{\widetilde{\theta}}\Big[\sigma^{-1}(i) = \ell\Big] . \qquad (3.47)$$

Observe that $\exp(\widetilde{\theta}_j) = 1$ for all $j \neq i$ and $\exp(\widetilde{\theta}_i) = \widetilde{\alpha}_{\ell,\theta} \geq \lfloor\widetilde{\alpha}_{\ell,\theta}\rfloor = \alpha_{\ell,\theta} \geq 0$. Therefore, we have

$$\begin{aligned} \mathbb{P}_{\widetilde{\theta}}\Big[\sigma^{-1}(i) = \ell\Big] \;\; &= \;\; \binom{\kappa-1}{\ell-1} \frac{\widetilde{\alpha}_{\ell,\theta}(\ell-1)!}{(\kappa-1+\widetilde{\alpha}_{\ell,\theta})(\kappa-2+\widetilde{\alpha}_{\ell,\theta})\cdots(\kappa-\ell+\widetilde{\alpha}_{\ell,\theta})} \\ &\leq \;\; \frac{(\kappa-1)!}{(\kappa-\ell)!} \frac{e^{2b}}{(\kappa-1+\alpha_{\ell,\theta})(\kappa-2+\alpha_{\ell,\theta})\cdots(\kappa-\ell+\alpha_{\ell,\theta})} \\ &\leq \;\; \frac{e^{2b}}{\kappa}\left(1 - \frac{\ell}{\kappa+\alpha_{\ell,\theta}}\right)^{\alpha_{\ell,\theta}-1} \leq \frac{e^{2b}}{\kappa-\ell} . \qquad (3.48) \end{aligned}$$

Claim 3.31 follows by combining Equations (3.47) and (3.48).
Claim (3.32)**:** Again, we construct a new set of parameters $\{\widetilde{\theta}_j\}_{j\in[d]}$ from the original θ using $\widetilde{\alpha}_{\ell,\theta}$ defined in (3.46):

$$\widetilde{\theta}_j \;\; = \;\; \begin{cases} \log(\widetilde{\alpha}_{\ell,\theta}) & \text{for } j \in \{i, i'\} \;, \\ \hfill 0 & \text{otherwise} \;. \end{cases}$$

Similarly define $\widetilde{W} \equiv \sum_{j\in S}\exp(\widetilde{\theta}_j) \;=\; \kappa - 2 + 2\widetilde{\alpha}_{\ell,\theta}$. Using the techniques similar to the ones used in proof of claim (3.30), we have,

$$\mathbb{P}_\theta\Big[\sigma^{-1}(i) = \ell_1, \sigma^{-1}(i') = \ell_2\Big] \;\; \leq \;\; e^{8b}\mathbb{P}_{\widetilde{\theta}}\Big[\sigma^{-1}(i) = \ell_1, \sigma^{-1}(i') = \ell_2\Big] (3.49)$$

Observe that $\exp(\widetilde{\theta}_j) = 1$ for all $j \neq i, i'$ and $\exp(\widetilde{\theta}_i) = \exp(\widetilde{\theta}_{i'}) = \widetilde{\alpha}_{\ell,\theta} \geq \lfloor \widetilde{\alpha} \rfloor_{\ell,\theta} = \alpha_{\ell,\theta} \geq 0$. Therefore, we have

$$
\begin{aligned}
&= \mathbb{P}_{\widetilde{\theta}}\Big[\sigma^{-1}(i) = \ell_1, \sigma^{-1}(i') = \ell_2\Big] \\
&= \Bigg(\frac{\binom{\kappa-2}{\ell_2-2}\widetilde{\alpha}_{\ell,\theta}^2(\ell_2-2)!}{(\kappa-2+2\widetilde{\alpha}_{\ell,\theta})(\kappa-1+2\widetilde{\alpha}_{\ell,\theta})\cdots(\kappa-2+2\widetilde{\alpha}_{\ell,\theta}-(\ell_1-1))} \\
&\quad \frac{1}{(\kappa-2+\widetilde{\alpha}_{\ell,\theta}-(\ell_1-1))\cdots(\kappa-2+\widetilde{\alpha}_{\ell,\theta}-(\ell_2-2))} \Bigg) \\
&\leq \frac{(\kappa-2)!}{(\kappa-\ell_2)!}\frac{e^{4b}}{(\kappa-2)(\kappa-1)\cdots(\kappa-\ell_1-1)(\kappa-\ell_1-1)\cdots(\kappa-\ell_2)} \\
&\leq \frac{e^{4b}}{(\kappa-\ell_1-1)(\kappa-\ell_2)}\,. \qquad (3.50)
\end{aligned}
$$

Claim 3.32 follows by combining Equations (3.49) and (3.50).

3.6.6 Proof of Theorem 3.9

Let $H(\theta) \in \mathcal{S}^d$ be Hessian matrix such that $H_{ii'}(\theta) = \frac{\partial^2 \mathcal{L}_{\rm RB}(\theta)}{\partial\theta_i \partial\theta_{i'}}$. The Fisher information matrix is defined as $I(\theta) = -\mathbb{E}_\theta[H(\theta)]$. From lemma 3.1, $\mathcal{L}_{\rm RB}(\theta)$ is concave. This implies that $I(\theta)$ is positive-semidefinite and from (3.23) its smallest eigenvalue is zero with all-ones being the corresponding eigenvector. Fix any unbiased estimator $\widehat{\theta}$ of $\theta \in \Omega_b$. Since, $\widehat{\theta} \in \mathcal{U}$, $\widehat{\theta} - \theta$ is orthogonal to $\mathbf{1}$. The Cramer-Rao lower bound then implies that $\mathbb{E}[\|\widehat{\theta} - \theta^*\|^2] \geq \sum_{i=2}^d \frac{1}{\lambda_i(I(\theta))}$. Taking supremum over both sides gives

$$
\sup_\theta \mathbb{E}[\|\widehat{\theta} - \theta^*\|^2] \;\; \geq \;\; \sup_\theta \sum_{i=2}^d \frac{1}{\lambda_i(I(\theta))} \geq \sum_{i=2}^d \frac{1}{\lambda_i(I(\mathbf{0}))}\,.
$$

In the following, we will show that

$$
\begin{aligned}
I(\mathbf{0}) \;=\; -\mathbb{E}_\theta[H(\mathbf{0})] \;&\preceq\; \sum_{j=1}^n \sum_{a=1}^{\ell_j} \frac{m_{j,a} - \eta_{j,a}}{\kappa_j(\kappa_j - 1)} \sum_{i<i' \in S_j} (e_i - e_{i'})(e_i - e_{i'})^\top \\
&\preceq\; \max_{j,a}\big\{m_{j,a} - \eta_{j,a}\big\}\, L\,. \qquad (3.51)
\end{aligned}
$$

Using Jensen's inequality, we have $\sum_{i=2}^d \frac{1}{\lambda_i(I(\mathbf{0}))} \geq \frac{(d-1)^2}{\sum_{i=2}^d \lambda_i(I(\mathbf{0}))} = \frac{(d-1)^2}{\mathrm{Tr}(I(\mathbf{0}))}$. From (6.28), we have $\mathrm{Tr}(I(\mathbf{0})) \leq \sum_{j,a}(m_{j,a} - \eta_{j,a})$. From (6.54), we have

$\sum_{i=2}^{d} 1/\lambda_i(I(\mathbf{0})) \;\geq\; (1/\max\{m_{j,a}-\eta_{j,a}\})\sum_{i=1}^{d} 1/\lambda_i(L)$. This proves the desired claim.

Now we are left to show claim (6.28). Consider a rank-breaking edge $e_{j,a}$. Using notations defined in lemma 3.12, in particular Equation (3.36), and omitting subscript $\{j,a\}$ whenever it is clear from the context, we have, for any $i \in V(e_{j,a})$,

$$\frac{\partial^2 \mathbb{P}_\theta(e_{j,a})}{\partial^2 \theta_i} = \begin{cases} \sum_\sigma \big(- A_\sigma B_\sigma e^{\theta_i} + A_\sigma (B_\sigma)^2 e^{2\theta_i} + A_\sigma C_\sigma e^{\theta_i} \big) \\ \text{if } \; i \in B(e_{j,a}) \\ \sum_\sigma \big(A_\sigma - 3A_\sigma B_{\sigma,i} e^{\theta_i} + A_\sigma C_{\sigma,i}) e^{2\theta_i} + A_\sigma (B_{\sigma,i})^2 e^{2\theta_i} \big) \\ \text{if } \; i \in T(e_{j,a})\,, \end{cases}$$

and using (3.37), we have

$$\frac{\partial^2 \log \mathbb{P}_\theta(e_{j,a})}{\partial^2 \theta_i}\Big|_{\theta=\mathbf{0}} = \begin{cases} \big((C_\sigma - B_\sigma)\big)_{\theta=\mathbf{0}} \\ \text{if } \; i \in B(e_{j,a}) \\ \big(\frac{1}{m_{j,a}!} \sum_\sigma \big(C_{\sigma,i} - B_{\sigma,i} + (B_{\sigma,i})^2\big) - \big(\sum_\sigma \frac{B_{\sigma,i}}{m_{j,a}!}\big)^2\big)_{\theta=\mathbf{0}} \\ \text{if } \; i \in T(e_{j,a})\,, \end{cases}$$

where $\sigma \in \Lambda_{T(e_{j,a})}$ and the subscript $\theta = 0$ indicates the the respective quantities are evaluated at $\theta = 0$. From the definitions given in (3.36), for $\theta = \mathbf{0}$, we have $B_\sigma - C_\sigma = \sum_{u=0}^{m-1} \frac{(r-u-1)}{(r-u)^2}$ and, $\sum_\sigma (B_{\sigma,i} - C_{\sigma,i})/(m!) = \frac{1}{m}\sum_{u=0}^{m-1} \frac{(m-u)(r-u-1)}{(r-u)^2}$. Also, $\sum_\sigma B_{\sigma,i}/(m!) = \frac{1}{m}\sum_{u=0}^{m-1}\frac{m-u}{r-u}$ and $\sum_\sigma (B_{\sigma,i})^2/(m!) = \frac{1}{m}\sum_{u=0}^{m-1}\big(\sum_{u'=0}^{u}\frac{1}{r-u'}\big)^2$. Combining all these and, using $\mathbb{P}_{\theta=\mathbf{0}}[i \in T(e_{j,a})] = m/\kappa$ and $\mathbb{P}_{\theta=\mathbf{0}}[i \in B(e_{j,a})] = (r-m)/\kappa$, and after following some algebra, we have for any $i \in S_j$,

$$\begin{aligned} & -\mathbb{E}\Big[\frac{\partial^2 \log \mathbb{P}_\theta(e_{j,a})}{\partial^2 \theta_i}\Big|_{\theta=\mathbf{0}}\Big] \\ = \;& \frac{1}{\kappa}\Bigg(m - \sum_{u=0}^{m-1}\frac{1}{r-u} - \frac{1}{m}\sum_{u=0}^{m-1}\frac{u(m-u)}{(r-u)^2} - \frac{1}{m}\sum_{u=0}^{m-2}\frac{2u}{r-u}\Bigg(\sum_{u'>u}^{m-1}\frac{m-u'}{r-u'}\Bigg)\Bigg) \\ = \;& \frac{m_{j,a}-\eta_{j,a}}{\kappa_j}\,, \end{aligned} \tag{3.52}$$

where $\eta_{j,a}$ is defined in (3.17). Since row-sums of $H(\theta)$ are zeroes, (3.23), and for $\theta = \mathbf{0}$, all the items are exchangeable, we have for any $i \neq i' \in S_j$,

$$\mathbb{E}\left[\frac{\partial^2 \log \mathbb{P}_\theta(e_{j,a})}{\partial\theta_i \partial\theta_{i'}}\Big|_{\theta=\mathbf{0}}\right] = \frac{m_{j,a} - \eta_{j,a}}{\kappa_j(\kappa_j - 1)},$$

The claim (6.28) follows from the expression of $H(\theta)$, Equation (3.23).

To verify (6.53), observe that $(r - m)(B_\sigma - C_\sigma) + m(\sum_\sigma B_{\sigma,i}/(m!)) = m - \sum_{u=0}^{m-1} \frac{1}{r-u}$. And,

$$\begin{aligned}
& \frac{1}{m}\Big(\sum_{u=0}^{m-1} \frac{m-u}{r-u}\Big)^2 - \sum_{u=0}^{m-1}\Big(\sum_{u'=0}^{u} \frac{1}{r-u'}\Big)^2 \\
= & \sum_{u=0}^{m-1}\Big(\frac{(m-u)^2}{m(r-u)^2} - \frac{m-u}{(r-u)^2}\Big) \\
& + \sum_{0 \le u < u' \le m-1}\Big(\frac{2(m-u)(m-u')}{m(r-u)(r-u')} - \frac{2(m-u')}{(r-u)(r-u')}\Big) \\
= & \sum_{u=0}^{m-1} \frac{-u(m-u)}{m(r-u)^2} + \sum_{0 \le u < u' \le m-1} \frac{-2u(m-u')}{m(r-u)(r-u')}.
\end{aligned}$$

3.6.7 Tightening of Lemma 3.12

Recall that $\mathbb{P}_\theta(e_{j,a})$ is same as probability of $\mathbb{P}_\theta[T(e_{j,a}) \succ B(e_{j,a})]$ that is the probability that an agent ranks $T(e_{j,a})$ items above $B(e_{j,a})$ items when provided with a set comprising $V(e_{j,a})$ items. As earlier, for brevity of notations, we omit subscript $\{j, a\}$ whenever it is clear from the context. For $m = 1$ or 2, it is easy to check that all off-diagonal elements in hessian matrix of $\log \mathbb{P}_\theta(e)$ are non-negative. However, since number of terms in summation in $\mathbb{P}_\theta(e)$ grows as $m!$, for $m \geq 3$ the straight-forward approach becomes too complex. Below, we derive expressions for cross-derivatives in hessian, for general m, using alternate definition (sorting of independent exponential r.v.'s in increasing order) of PL model, where the number of terms grow only as 2^m. However, we are unable to analytically prove that the cross-derivatives are non-negative for $m > 2$. Feeding these expressions in MATLAB and using symbolic computation, for $m = 3$, we can simplify these expressions and it turns out that they are sum of only positive numbers. For $m = 4$, with limited computational power it becomes intractable. We believe that it should hold for any value of $m < r$. Using (3.29), we need to check only for cross-derivatives for the case when $i \neq i' \in T(e_{j,a})$ or $i \in T(e_{j,a}), i' \in B(e_{j,a})$. Since, minimum of exponential random variables is an exponential random variable, we can assume that $|B(e_{j,a})| = 1$ that is $r = m+1$. Define $\lambda_i \equiv e^{\theta_i}$. Without loss of generality, assume $T(e_{j,a}) = \{2, \cdots, m+1\}$ and $B(e_{j,a}) = \{1\}$. Define $C_x = \prod_{i=3}^{m+1}(1 - e^{-\lambda_i x})$. Then, using the alternate definition of the PL model, we have, $\mathbb{P}_\theta(e) = \int_0^\infty C_x(1 - e^{-\lambda_2 x})\lambda_1 e^{-\lambda_1 x} dx$. Following some algebra, $\frac{\partial^2 \log \mathbb{P}_\theta(e)}{\partial\theta_1 \partial\theta_2} \geq 0$ is equivalent to $A_1 \geq 0$, where $A_1 \equiv$

$$\Big(\int C_x \big(x e^{-\lambda_1 x} - x e^{-\lambda x} \big) dx \Big) \Big(\int C_x x e^{-\lambda x} dx \Big)$$
$$- \Big(\int C_x (e^{\lambda_1 x} - e^{-\lambda x}) dx \Big) \Big(\int C_x x^2 e^{-\lambda x} dx \Big) ,$$

where all integrals are from 0 to ∞ and, $\lambda \equiv \lambda_1 + \lambda_2$. Consider A_1 as a function of λ_1. Since $A_1(\lambda_1) = 0$ for $\lambda_1 = \lambda$, showing $\partial A_1 / \partial \lambda_1 \leq 0$ for $0 \leq \lambda_1 \leq \lambda$ would suffice. Following some algebra, and using $\lambda_1 \leq \lambda$, $\partial A_1 / \partial \lambda_1 \leq 0$ is equivalent to $A_2(\lambda_1) \equiv \big(\int_0^\infty C_x x e^{-\lambda_1 x} \big) / \big(\int_0^\infty C_x x^2 e^{-\lambda_1 x} \big)$ being monotonically non-decreasing in λ_1. To further simplify the condition, define $f^{(0)}(y) = 1/y^2$, $g^{(0)}(y) = 1/y^3$ and, $f^{(1)}(y) = f^{(0)}(y) - f^{(0)}(y + \lambda_3)$, and recursively $f^{(m-1)}(y) = f^{(m-2)}(y) - f^{(m-2)}(y + \lambda_{m+1})$. Similarly define $g^{(0)}, \cdots, g^{(m-1)}$. Using these recursively defined functions,

$$\begin{aligned} 2A_2(\lambda_1) &= \frac{f^{(m-1)}(\lambda_1)}{g^{(m-1)}(\lambda_1)} , \\ \text{for } m = 3, \;\; 2A_2(\lambda_1) &= \frac{\lambda_1^{-2} - (\lambda_1 + \lambda_3)^{-2} - (\lambda_1 + \lambda_4)^{-2} + (\lambda_1 + \lambda_3 + \lambda_4)^{-2}}{\lambda_1^{-3} - (\lambda_1 + \lambda_3)^{-3} - (\lambda_1 + \lambda_4)^{-3} + (\lambda_1 + \lambda_3 + \lambda_4)^{-3}} . \end{aligned}$$

Therefore, we need to show that $A_2(\lambda_1)$ is monotonically non-decreasing in $\lambda_1 \geq 0$ for any non-negative $\lambda_3, \cdots, \lambda_m$, and that would suffice to prove that the cross-derivatives arising from $i \in T(e_{j,a}), i' \in B(e_{j,a})$ are non-negative.

For cross-derivatives arising from $i \neq i' \in T(e_{j,a})$, define $B_x = \prod_{i=4}^{m+1}(1 - e^{\lambda_i x})e^{-\lambda_1 x}$. $\frac{\partial^2 \log \mathbb{P}_\theta(e)}{\partial\theta_2 \partial\theta_3} \geq 0$ is equivalent to $A_3 \geq 0$, where $A_3 \equiv$

$$\Big(\int B_x (1 - e^{-\lambda_2 x})(1 - e^{-\lambda_3 x}) dx \Big) \Big(\int B_x x^2 e^{-(\lambda_2 + \lambda_3)x} dx \Big)$$
$$- \Big(\int B_x (1 - e^{-\lambda_2 x}) x e^{-\lambda_3 x} dx \Big) \Big(\int B_x (1 - e^{-\lambda_3 x}) x e^{-\lambda_2 x} dx \Big) ,$$

where all integrals are from 0 to ∞. For $m = 3$, using MATLAB one can check that the above stated conditions hold true. Therefore both types of cross-derivatives are non-negative.

CHAPTER 4

ACHIEVING BUDGET-OPTIMALITY WITH ADAPTIVE SCHEMES IN CROWDSOURCING

The generalized Dawid-Skene model studied in this paper allows the tasks to be heterogeneous (having different difficulties) and the workers to be heterogeneous (having different reliabilities). The original Dawid-Skene (DS) model introduced in [46] and analyzed in [107] is a special case, when only workers are allowed to be heterogeneous. All tasks have the same difficulty with $\lambda_i = 0$ for all $i \in [m]$ and q_i can be either zero or one depending on the true label. Most of the existing work on the DS model assumes that tasks are randomly assigned and focuses only on the inference problem of finding the true labels. Several inference algorithms have been proposed [46, 193, 100, 190, 77, 105, 134, 221, 126, 215, 43, 106, 162, 22, 23, 141].

A most relevant work is by [107]. It is shown that in order to achieve a probability of error less than a small positive constant $\varepsilon > 0$, it is necessary to have an expected budget scaling as $\Gamma = O((m/\sigma^2)\log(1/\varepsilon))$, even for the best possible inference algorithm together with the best possible task assignment scheme, including all possible adaptive task assignment schemes. Further, a simple randomized non-adaptive task assignment is proven to achieve this optimal trade-off with a novel spectral inference algorithm. Namely, an efficient task assignment and an inference algorithm are proposed that together guarantees to achieve $p_{\text{error}} \leq \varepsilon$ with budget scaling as $\Gamma = O((m/\sigma^2)\log(1/\varepsilon))$. It is expected that this necessary and sufficient budget constraint scales linearly in m, the number of tasks to be labelled. The technical innovation of [107] is in (i) designing a new spectral algorithm that achieves a logarithmic dependence in the target error rate ε; and (ii) identifying σ^2 defined in (4.3) as the fundamental statistics of $\mathcal{P}$ that captures the collective quality of the crowd. The budget-accuracy trade-off mainly depends on the prior distribution of the crowd $\mathcal{P}$ via a single parameter σ^2. When we have a reliable crowd with many workers having p_j's close to one, the collective quality σ^2 is close to one and the required budget Γ is small. When we have an unreliable crowd with many workers having p_j's close to a half, then the collective quality is close to zero and the required budget is large. However, perhaps one of the most surprising result of [107] is that the optimal trade-off is matched by a non-adaptive task assignment scheme. In other words, there is only a marginal gain in using adaptive task assignment schemes.

This negative result relies crucially on the fact that, under the standard DS model, all tasks are inherently equally difficult. As all tasks have q_i's either zero or one, the individual difficulty of a task is $\lambda_i \equiv (2q_i - 1)^2 = 1$, and a worker's probability of making an error on one task is the same as any other tasks. Hence, adaptively assigning more workers to relatively more ambiguous tasks has only a marginal gain. However, simple adaptive schemes are widely used in practice, where significant gains are achieved. In real-world systems, tasks are widely heterogeneous. Some images are much more difficult to classify (and find the true label) compared to other images. To capture such varying difficulties in the tasks, generalizations of the DS model were proposed in [207, 204, 220, 185] and significant improvements have been reported on real datasets.

The generalized DS model serves as the missing piece in bridging the gap between practical gains of adaptivity and theoretical limitations of adaptivity (under the standard DS model). We investigate the fundamental question of "do adaptive task assignments improve accuracy?" under this generalized Dawid-Skene model of Eq. (7.51).

On the theoretical understanding of the original DS model, the dense regime has been studied first, where all workers are assigned all tasks. A spectral method for finding the true labels was first analyzed in [77] and an EM approach followed by spectral initial step is analyzed in [215] to achieve a near-optimal performance. The minimax error rate of this problem was identified in [76] by analyzing the MAP estimator, which is computationally intractable. In this paper, we are interested in the sparse regime where each task is assigned only a small number of workers of $O(\log m)$.

One of the main weaknesses of the DS model is that it does not capture how some tasks are more difficult than the others. To capture such heterogeneity in the tasks, several practical models have been proposed recently [100, 207, 204, 220, 91]. Although such models with more parameters can potentially better describe real-world datasets, there is no analysis on their performance under adaptive or non-adaptive task assignments. We do not have the analytical tools to understand the fundamental trade-offs involved in those models yet. In this work, we close this gap by providing a theoretical analysis of one of the generalizations of the DS model, namely the one proposed in [220]. It captures the heterogeneous difficulties in the tasks, while remaining simple enough for theoretical analyses.

4.1 Model and problem formulation

In this work, we assume that the requester has m binary classification tasks to be labelled by querying a crowdsourcing platform multiple times. For example, those might be image classification tasks, where the requester wants to classify m images as either suitable for children $(+1)$ or not (-1). The requester has a budget Γ on how many responses she can collect on the crowdsourcing platform, assuming one unit of payment is made for each response collected. We use Γ interchangeably to refer to both a target budget and also the budget used by a particular task assignment scheme (as defined in (4.1)), and it should be clear from the context which one we mean. We want to find the true label by querying noisy workers who are arriving in an online fashion, one at a time.

Task assignment and inference. Typical crowdsourcing systems are modeled as a discrete time systems where at each time we have a new arriving worker. At time j, the requester chooses an action $T_j \subseteq [m]$, which is a subset of tasks to be assigned to the j-th arriving worker. Then, the j-th arriving worker provides her answer $A_{ij} \in \{+1, -1\}$ for each task $i \in T_j$. We use the index j to denote both the j-th time step in this discrete time system as well as the j-th arriving worker. At this point (at the end of j-th time step), all previous responses are stored in a sparse matrix $A \in \{0, +1, -1\}^{m \times j}$, and this data matrix is increasing by one column at each time. We let $A_{ij} = 0$ if task i is not assigned to worker j, i.e. $i \notin T_j$, and otherwise we let $A_{ij} \in \{+1, -1\}$ be the previous worker j's response on task i. At the next time $j + 1$, the next task assignment T_{j+1} is chosen, and this process is repeated. At time j, the action (or the task assignment) can depend on all previously collected responses up to the current time step stored in a sparse (growing) matrix $A \in \{0, +1, -1\}^{m \times (j-1)}$. This process is repeated until the task assignment scheme decides to stop, typically when the total number of collected responses (the number of nonzero entires in A) meet a certain budget constraint or when a certain target accuracy is estimated to be met.

We consider both a *non-adaptive scenario* and an *adaptive scenario*. In a non-adaptive scenario, a fixed number n of workers to be recruited are pre-determined (and hence the termination time is set to be n) and also fixed task assignments T_j's for all $j \in [n]$ are pre-determined, before any response is collected. In an *adaptive scenario*, the requester chooses T_j's in an online fashion based on all the previous answers collected thus far. For both adaptive and non-adaptive scenarios, when we have determined that we have collected all the data we need, an inference algorithm is applied on the collected data $A \in \{0, +1, -1\}^{m \times n}$ to output an estimate $\hat{t}_i \in \{+1, -1\}$ for the ground truth label $t_i \in \{+1, -1\}$ for the i-th task for each $i \in [m]$. Note that we use n to denote the total number of workers recruited, which is a random variable under the adaptive scenario. Also, note that the estimated labels for all the tasks do not have to be simultaneously output in the end, and we can choose to output estimated labels on some of the tasks in the middle of the process before termination. The average accuracy of our estimates is measured by the average probability of error $P_{\text{error}} = (1/m) \sum_{i=1}^{m} \mathbb{P}[t_i \neq \hat{t}_i]$ under a probabilistic model to be defined later in this section in Eq. (7.51).

Budget. The total *budget* used in one instance of such a process is measured by the total number of responses collected, which is equal to the number of non-zero entries in A. This inherently assumes that there is a prefixed fee of one unit for each response that is agreed upon, and the requester pays this constant fee for every label that is collected. The expected budget used by a particular task assignment scheme will be denoted by

$$\Gamma \;\equiv\; \mathbb{E}\Big[\sum_{j=1}^{n} |T_j|\Big] \,, \tag{4.1}$$

where the expectation is over all the randomness in the model (the problem parameters representing the quality of the tasks and the quality of the workers, and the noisy responses from workers) and any randomness used in the task assignment. We are interested in designing task assignment schemes and inference algorithms that achieve the best accuracy within a target expected budget, under the following canonical model of how workers respond to tasks.

Worker responses. We assume that when a task is assigned to a worker, the response follows a probabilistic model introduced by [220], which is a recent generalization of the Dawid-Skene model originally introduced by [46]. Precisely, each new arriving worker is parametrized by a latent worker quality parameter $p_j \in [0, 1]$ (for the j-th arriving worker). Each task is parametrized by a latent task quality parameter $q_i \in [0, 1]$ (for the i-th task). When a worker j is assigned a task i, the *generalized Dawid-Skene model* assumes that the response $A_{ij} \in \{+1, -1\}$ is a random variable distributed as

$$A_{ij} \;=\; \begin{cases} +1, & \text{w.p. } \; q_i p_j + \bar{q}_i \bar{p}_j \;, \\ -1, & \text{w.p. } \; q_i \bar{p}_j + \bar{q}_i p_j \;, \end{cases} \quad , \tag{4.2}$$

conditioned on the parameters q_i and p_j, where $\bar{q}_i = 1 - q_i$ and $\bar{p}_j = 1 - p_j$. The classical Dawid and Skene model assumes that task quality parameter q_i is one for each task. The task parameter q_i represents the probability that a task is perceived as a positive task to a worker, and the worker parameter p_j represents the probability the worker makes a mistake in labelling the task. Concretely, when a task i is presented to any worker, the task is perceived as a positive task with a probability q_i or a negative task otherwise, independent of any other events. Let $\tilde{t}_{ij}$ denote this perceived label of task i as seen by worker j. Conditioned on this perceived label of the task, a worker j with parameter p_j makes a mistake with probability $1 - p_j$. She provides a 'correct' label $\tilde{t}_{ij}$ as she perceives it with probability p_j, or provides an 'incorrect' label $-\tilde{t}_{ij}$ with probability $\bar{p}_j$. Hence, the response A_{ij} follows the distribution in (7.51). The response A_{ij} is, for example, a positive label if the task is perceived as a positive task and the worker does not make a mistake (which happens with a probability $q_i p_j$), or if the task is perceived as a negative task and the worker does not make a mistake (which happens with a probability $\bar{q}_i \bar{p}_j$). Alternately, the task parameter q_i represents the probability that a task is labeled as a positive task by a perfect worker, a worker with parameter $p_j = 1$. That is q_i represents inherent ambiguity of the task being labeled positive. The strengths and weaknesses of this model are discussed in comparisons to related work in Section 3.2.

Prior distribution on worker reliability. We assume that worker parameters p_j's are i.i.d. according to some prior distribution $\mathcal{P}$. For example, each arriving worker might be sampled with replacement from a pool of workers, and $\mathcal{P}$ denotes the discrete distribution of the quality parameters of the pool. The individual reliabilities p_j's are hidden from us, and the prior distribution $\mathcal{P}$ is also unknown. We assume we only know some statistics of the prior distribution $\mathcal{P}$, namely

$$\mu \;\equiv\; \mathbb{E}_{\mathcal{P}}[2p_j - 1]\;, \text{ and} \qquad \sigma^2 \;\equiv\; \mathbb{E}_{\mathcal{P}}[(2p_j-1)^2]\;, \tag{4.3}$$

where p_j is a random variable distributed as $\mathcal{P}$, and $\mu \in [-1,1]$ is the (shifted and scaled) average reliability of the crowd and $\sigma^2 \in [0,1]$ is the key quantity of $\mathcal{P}$ capturing the collective quality of the crowd as a whole. Intuitively, when all workers are truthful and have p_j close to a one, then the collective reliability σ^2 will be close to its maximum value of one. On the other hand, if most of the workers are giving completely random answers with p_j's close to a half, then σ^2 will be close to its minimum value of a zero. The fundamental trade-off between the accuracy and the budget will primarily depend on the distribution of the crowd $\mathcal{P}$ via σ^2. We do not impose any conditions on the distribution $\mathcal{P}$.

Prior distribution on task quality. We assume that the task parameters q_i's are drawn i.i.d. according to some prior distribution $\mathcal{Q}$. The individual difficulty of a task with a quality parameter q_i is naturally captured by

$$\lambda_i \;\;\equiv\;\; (2q_i-1)^2\;, \tag{4.4}$$

as tasks with q_i close to a half are confusing and ambiguous tasks and hence difficult to correctly label (λ_i close to zero), whereas tasks with q_i close to zero or one are unambiguous tasks and easy to correctly label (λ_i close to one). Note that the larger the λ_i the easier is the task but with a slight abuse of notation we call it *task difficulty*. The average difficulty and the collective difficulty of tasks drawn from a prior distribution $\mathcal{Q}$ are captured by the quantities $\rho \in [0,1]$ and $\lambda \in [0,1]$, defined as

$$\rho \;\equiv\; \mathbb{E}_{\mathcal{Q}}\left[(2q_i-1)^2\right]\;, \quad \lambda \;\equiv\; \left(\mathbb{E}_{\mathcal{Q}}\left[\frac{1}{(2q_i-1)^2}\right]\right)^{-1}\;, \tag{4.5}$$

where q_i is distributed as $\mathcal{Q}$. The fundamental budget-accuracy trade-off depends on $\mathcal{Q}$ primarily via this λ. Another quantities that will show up in our main results is the worst-case difficulty in the given set of m tasks (conditioned on all the q_i's) defined as

$$\lambda_{\min} \;\equiv\; \min_{i\in[m]}(2q_i-1)^2\,, \qquad \text{and} \qquad \lambda_{\max} \;\equiv\; \max_{i\in[m]}(2q_i-1)^2\,. \tag{4.6}$$

When we refer to a similar quantities from the population distributed as $\mathcal{Q}$, we abuse the notation and denote $\lambda_{\min} = \min_{q_i\in\text{supp}(\mathcal{Q})}(2q_i-1)^2$ and $\lambda_{\max} = \max_{q_i\in\text{supp}(\mathcal{Q})}(2q_i-1)^2$. The individual task parameters q_i's are hidden from us. We do not have access to the prior distribution $\mathcal{Q}$ on the task qualities q_i's, but we assume we know the statistics ρ, λ, $\lambda_{\min}$, and $\lambda_{\max}$, and we assume we also know a quantized version of the prior distribution on the task difficulties λ_i's, which we explain below.

Quantized prior distribution on task difficulty. Given a distribution $\mathcal{Q}$ on q_i's, let $\widetilde{\mathcal{Q}}$ be the induced distribution on λ_i's. For example, if $\mathcal{Q}(q_i) = (1/10)\mathbb{I}_{(q_i=0.9)}+(3/10)\mathbb{I}_{(q_i=0.1)}+(1/10)\mathbb{I}_{(q_i=0.8)}+(3/10)\mathbb{I}_{(q_i=0.2)}+(2/10)\mathbb{I}_{(q_i=0.6)}$, then the induced distribution on λ_i is $\widetilde{\mathcal{Q}}(\lambda_i) = (4/10)\mathbb{I}_{(\lambda_i=0.64)}+(4/10)\mathbb{I}_{(\lambda_i=0.36)}+(2/10)\mathbb{I}_{(\lambda_i=0.04)}$. Our approach requires only the knowledge of a quantized version of the distribution $\widetilde{\mathcal{Q}}$, namely $\widehat{\mathcal{Q}}$. This quantized distribution has support at $\tilde{T}$ discrete values $\{\lambda_{\max}, \lambda_{\max}/2, \ldots, \lambda_{\max}/2^{(\tilde{T}-1)}\}$, where

$$\tilde{T} \;\equiv\; 1 + \left\lceil \log_2\left(\frac{\lambda_{\max}}{\lambda_{\min}}\right)\right\rceil\,, \tag{4.7}$$

such that $\lambda_{\max}2^{-(\tilde{T}-1)} \le \lambda_{\min} \le \lambda_{\max}2^{-(\tilde{T}-2)}$. We denote these values by $\{\tilde{\lambda}_a\}_{a\in[\tilde{T}]}$ such that $\tilde{\lambda}_a = \lambda_{\max}2^{-(a-1)}$ for each $a \in [\tilde{T}]$. Then the quantized distribution is $\sum_{a=1}^{\tilde{T}} \tilde{\delta}_a \mathbb{I}_{(\lambda_i=\tilde{\lambda}_a)}$, where the probability mass $\tilde{\delta}_a$ for the a-th partition is

$$\tilde{\delta}_a = \widetilde{\mathcal{Q}}(\,(\lambda_{\max}/2^a, \lambda_{\max}/2^{(a-1)}]\,)\,, \qquad \text{for } \; a\in[\tilde{T}]\,,$$

which is the fraction of tasks whose difficulty λ_i is in $(\tilde{\lambda}_{a+1}, \tilde{\lambda}_a]$. We use the closed interval $[(1/2)\tilde{\lambda}_{\tilde{T}}, \tilde{\lambda}_{\tilde{T}}]$ for the last partition. In the above example, we have $\tilde{T}=5$, $\{\tilde{\lambda}_a\}_{a\in\tilde{T}} = \{0.64, 0.32, 0.16, 0.08, 0.04\}$, and $\{\tilde{\delta}_a\}_{a\in\tilde{T}} = \{0.8, 0, 0, 0, 0.2\}$. For notational convenience, we eliminate those partitions with zero probability mass, and re-index the quantization $\{\tilde{\lambda}_a, \tilde{\delta}_a\}_{a\in[\tilde{T}]}$ to get $\{\lambda_a, \delta_a\}_{a\in[T]}$, for $T \le \tilde{T}$, such that $\delta_a \neq 0$ for all $a \in T$. We define $\widehat{\mathcal{Q}}$ to be the re-indexed quantized distribution $\{\lambda_a, \delta_a\}_{a\in[T]}$. In the above example, we finally have $\widehat{\mathcal{Q}}(\lambda_i) = 0.8\mathbb{I}_{(\lambda_i=0.64)} + 0.2\mathbb{I}_{(\lambda_i=0.04)}$.

We denote the maximum and minimum probability mass in $\widehat{\mathcal{Q}}$ as

$$\delta_{\max} \;\equiv\; \max_{a\in[T]} \delta_a \;, \quad \text{and} \qquad \delta_{\min} \;\equiv\; \min_{a\in[T]} \delta_a \;. \tag{4.8}$$

Similar to the collective quality λ defined for the distribution $\mathcal{Q}$ in (6.27), we define $\widehat{\lambda}$, collective quality for the quantized distribution $\widehat{\mathcal{Q}}$, which is used in our algorithm. $\widehat{\lambda} \equiv (\sum_{a\in[T]}(\delta_a/\lambda_a))^{-1}$.

Ground truth. The ground truth label t_i of a task is naturally defined as what the majority of the crowd would agree on if we ask all the workers to label that task, i.e. $t_i \;\equiv\; \text{sign}(\mathbb{E}[A_{ij}|q_i]) \;=\; \text{sign}(2q_i - 1)\text{sign}(\mu)$, where the expectation is with respect to the prior distribution of $p_j \sim \mathcal{P}$ and the randomness in the response as per the generalized Dawid-Skene model in (7.51). Without loss of generality, we assume that the average reliability of the worker is positive, i.e. $\text{sign}(\mu) = +1$ and take $\text{sign}(2q_i - 1)$ as the ground truth label t_i of task i conditioned on its difficulty parameter q_i:

$$t_i \;=\; \text{sign}(2q_i - 1)\,. \tag{4.9}$$

The latent parameters $\{q_i\}_{i\in[m]}$, $\{p_j\}_{j\in[n]}$, and $\{t_i\}_{i\in[m]}$ are unknown, and we want to infer the true labels t_i's from only A_{ij}'s.

Performance measure. The accuracy of the final estimate is measured by the average probability of error:

$$P_{\text{error}} \;=\; \frac{1}{m}\sum_{i=1}^{m} \mathbb{P}[t_i \neq \hat{t}_i]\;. \tag{4.10}$$

We investigate the fundamental trade-off between budget and error rate by identifying the sufficient and necessary conditions on the expected budget Γ for achieving a desired level of accuracy $P_{\text{error}} \le \varepsilon$. Note that we are interested in achieving the best trade-off, which in turn can give the best approach for both scenarios: when we have a fixed budget constraint and want to minimize the error rate, and when we have a target error rate and want to minimize the cost.

4.1.1 Contributions

To investigate the gain of adaptivity, we first characterize the fundamental lower bound on the budget required to achieve a target accuracy. To match this fundamental limit, we introduce a novel *adaptive* task assignment scheme. The proposed adaptive task assignment is simple to apply in practice, and numerical simulations confirm the superiority compared to state-of-the-art non-adaptive schemes. Under certain assumptions on the choice of parameters in the algorithm, which requires a moderate access to an oracle, we can prove that the performance of the proposed adaptive scheme matches that of the fundamental limit up to a constant factor. Finally, we quantify the gain of adaptivity by proving a strictly larger lower bound on the budget required for any non-adaptive schemes to achieve a desired error rate of ε for some small positive ε.

Precisely, we show that the minimax rate on the budget required to achieve a target average error rate of ε scales as $\Theta((m/\lambda\sigma^2)\log(1/\varepsilon))$. The dependence on the prior $\mathcal{P}$ and $\mathcal{Q}$ are solely captured in σ^2 (the quality of the crowd as a whole) and λ (the quality of the tasks as a whole). We show that the fundamental trade-off for *non-adaptive* schemes is $\Theta((m/\lambda_{\min}\sigma^2)\log(1/\varepsilon))$, requiring a factor of $\lambda/\lambda_{\min}$ larger budget for non-adaptive schemes. This factor of $\lambda/\lambda_{\min}$ is always at least one and quantifies precisely how much we gain by adaptivity.

4.1.2 Outline and notations

We present a list of notations and their definitions in Table 4.1. In Section 7.4, we present the fundamental lower bound on the necessary budget to achieve a target average error rate of ε. We present a novel adaptive approach which achieves the fundamental lower bound up to a constant. In comparison, we provide the fundamental lower bound on the necessary budget for non-adaptive approaches in Section 4.3, and we present a non-adaptive approach that achieves this fundamental limit. In Section 4.4, we give a spectral interpretation of our approach justifying the proposed inference algorithm, leading to a parameter estimation algorithm that serves as a building block in the main approach of Algorithm 4. As our proposed sub-routine using Algorithm 5 suffers when the budget is critically limited (known as spectral barrier in Section 4.4), we present another algorithm that can substitute Algorithm 5 in Section 4.5 and compare their performances. The proofs of the main results are provided in Section 4.6. We present a conclusion with future research directions in Section 6.5.

notation	data type	definition
m	$\mathbb{Z}_+$	the number of tasks
n	$\mathbb{Z}_+$	total number of workers recruited
$A = [A_{ij}]$	$\{0, +1, -1\}^{m \times n}$	labels collected from the workers
Γ	$\mathbb{R}_+$	budget used in collecting A is the number of nonzero entries in A
ℓ	$\mathbb{Z}_+$	average budget per task : Γ/m
Γ_ε	$\mathbb{R}_+$	the budget required to achieve error at most ε
$\mathscr{T}_\Gamma$		set of task assignment schemes using at most Γ queries in expectation
i	[m]	index for tasks
j	[n]	index for workers
W_i	subset of $[n]$	a set of workers assigned to task i
T_j	subset of $[m]$	a set of tasks assigned to worker j
q_i	$[0, 1]$	quality parameter of task i
t_i	$\{-1, +1\}$	ground truths label of task i
$\hat{t}_i$	$\{-1, +1\}$	estimated label of task i
p_j	$[0, 1]$	quality parameter of worker j
$\mathcal{P}$	$[0, 1] \to \mathbb{R}$	prior distribution of p_j
$\mathcal{Q}$	$[0, 1] \to \mathbb{R}$	prior distribution of q_i
$\widetilde{\mathcal{Q}}$	$[0, 1] \to \mathbb{R}$	prior distribution of λ_i induced from $\mathcal{Q}$
$\widehat{\mathcal{Q}}$	$[0, 1] \to \mathbb{R}$	quantized version of the distribution $\widetilde{\mathcal{Q}}$

Table 4.1: Notations

4.2 Main Results under the Adaptive Scenario

In this section, we present our main results under the adaptive task assignment scenario.

4.2.1 Fundamental limit under the adaptive scenario

With a slight abuse of notations, we let $\hat{t}(A)$ be a mapping from $A \in \{0, +1, -1\}^{m\times n}$ to $\hat{t}(A) \in \{+1, -1\}^m$ representing an inference algorithm outputting the estimates of the true labels. We drop A and write only $\hat{t}$ whenever it is clear from the context. We let $\mathcal{P}_{\sigma^2}$ be the set of all the prior distributions on p_j such that the collective worker quality is σ^2, i.e.

$$\mathcal{P}_{\sigma^2} \equiv \left\{\mathcal{P} \,|\, \mathbb{E}_{\mathcal{P}}[(2p_j - 1)^2] = \sigma^2\right\} . \tag{4.11}$$

We let $\mathcal{Q}_\lambda$ be the set of all the prior distributions on q_i such that the collective task difficulty is λ, i.e.

$$\mathcal{Q}_\lambda \equiv \left\{\mathcal{Q} \,\middle|\, \left(\mathbb{E}_{\mathcal{Q}}\left[\frac{1}{(2q_i - 1)^2}\right]\right)^{-1} = \lambda\right\} . \tag{4.12}$$

We consider all task assignment schemes in $\mathscr{T}_\Gamma$, the set of all task assignment schemes that make at most Γ queries to the crowd in expectation. We prove a lower bound on the standard minimax error rate: the error that is achieved by the best inference algorithm $\hat{t}$ using the best adaptive task assignment scheme $\tau \in \mathscr{T}_\Gamma$ under a worst-case worker parameter distribution $\mathcal{P} \in \mathcal{P}_{\sigma^2}$ and the worst-case task parameter distribution $\mathcal{Q} \in \mathcal{Q}_\lambda$. A proof of this theorem is provided in Section 4.6.1.

Theorem 4.1. *For $\sigma^2 < 1$, there exists a positive constant C' such that the average probability of error is lower bounded by black*

$$\min_{\tau\in\mathscr{T}_\Gamma,\hat{t}} \quad \max_{\mathcal{Q}\in\mathcal{Q}_\lambda,\mathcal{P}\in\mathcal{P}_{\sigma^2}} \quad \frac{1}{m}\sum_{i=1}^{m}\mathbb{P}[t_i \neq \hat{t}_i] \quad \geq \quad \frac{1}{2}\,e^{-C'\left(\frac{\Gamma\lambda\sigma^2}{m}+1\right)} \, , \tag{4.13}$$

where m is the number of tasks, Γ is the expected budget allowed in $\mathscr{T}_\Gamma$, λ is the collective difficulty of the tasks from a prior distribution $\mathcal{Q}$ defined in (6.27)*, and σ^2 is the collective reliability of the crowd from a prior distribution $\mathcal{P}$ defined in* (4.3)*.*

In the proof, we provide a proof of a slightly stronger statement in Lemma 4.7, where a similar lower bound holds for not only the worst-case $\mathcal{Q}$ but for all discrete $\mathcal{Q} \in \mathcal{Q}_\lambda$. One caveat is that there is now an extra additive term in the error exponent in the RHS of the lower bound that depends on $\mathcal{Q}$, which is reflected in the constant term (1/2) for the worst-case $\mathcal{Q}$ in the RHS of (4.13). To get the lower bound, we assume that the best task assignment scheme has access to an oracle that knows difficulty of each task q_i. It assigns the appropriate number of workers to each task depending on its difficulty such that the probability of error in each task is approximately same. It assigns more workers to difficult tasks and fewer workers to easy tasks. This motivates us to design an adaptive algorithm that aims to assign workers to tasks according to the task difficulties.

We are assigning Γ/m queries per task on average, and it is intuitive that the error decays exponentially in Γ/m. The novelty in the above analysis is that it characterizes how the error exponent depends on the $\mathcal{P}$, which determines the quality of the crowd you have in your crowdsourcing platform, and $\mathcal{Q}$, which determines the quality of the tasks you have in your hand. If we have easier tasks and reliable workers, the error rate should be smaller. Eq. (4.13) shows that this is captured by the error exponent scaling linearly in $\lambda\sigma^2$. This gives a lower bound (i.e. a necessary condition) on the budget required to achieve error at most ε; there exists a constant C'' such that if the total budget is

$$\Gamma_\varepsilon \;\; \leq \;\; C''\frac{m}{\lambda\sigma^2}\log\left(\frac{1}{\varepsilon}\right) \;, \tag{4.14}$$

then no task assignment scheme (adaptive or not) with any inference algorithm can achieve error less than ϵ. This recovers the known fundamental limit for standard DS model where all tasks have $\lambda_i = 1$ and hence $\lambda = 1$ in [107]. For this standard DS model, it is known that there exists a constant C''' such that if the total budget is less than

$$\Gamma_\varepsilon \;\; \leq \;\; C'''\frac{m}{\sigma^2}\log\left(\frac{1}{\epsilon}\right) \;,$$

then no task assignment with any inference algorithm can achieve error rate less than ε. For example, consider two types of prior distributions where in one we have the original DS tasks with $\mathcal{Q}(q_i = 0) = \mathcal{Q}(q_i = 1) = 1/2$ and in the other we have $\mathcal{Q}'(q_i = 0) = \mathcal{Q}'(q_i = 1) = \mathcal{Q}'(q_i = 3/4) = \mathcal{Q}'(q_i = 1/4) = 1/4$. We have $\lambda = 1$ under $\mathcal{Q}$ and $\lambda' = 2/5$ under $\mathcal{Q}'$. Our analysis, together with the matching upper bound in the following section, shows that one needs 5/2 times more budget to achieve the same accuracy under the tasks from $\mathcal{Q}'$.

4.2.2 Why do we need an adaptive algorithm?

Should we put the lower bound under the non-adaptive scheme here in this section before we explain our adaptive algorithm? It will show that no non-adaptive algorithm can achieve the above lower bound. Therefore, we present an adaptive algorithm.

4.2.3 Adaptive algorithm when workers are perfect

To explain the main idea of our adaptive algorithm, we first consider a simple scenario when all the workers are perfect, $p_j = 1$, for $j \in [n]$, $\sigma^2 = \mathbb{E}[(2p_j - 1)^2] = 1$. In this case, the optimal inference algorithm is the majority voting.

First get a simplified theorem and its proof!

Second build up the algorithm by explaining why one needs to design such an algorithm.

Theorem 4.2. *Suppose Algorithm 8 returns the exact value of*

$$\rho_{t,u}^2 = (1/|M|) \sum_{i \in M} \lambda_i \,.$$

With the choice of $C_\delta = (4 + \lceil \log(2\delta_{\max}/\delta_{\min}) \rceil)^{-1}$, for any given quantized prior distribution of task difficulty $\{\lambda_a, \delta_a\}_{a \in [T]}$ such that $\delta_{\max}/\delta_{\min} = O(1)$ and $\lambda_{\max}/\lambda_{\min} = O(1)$, and the budget $\Gamma = \Theta(m \log m)$, the expected number of queries made by Algorithm 4 is asymptotically bounded by

$$\lim_{m \to \infty} \sum_{t \in [T], u \in [s_t]} \ell_t \, \mathbb{E}[m_{t,u}] \ \ \leq \ \ \Gamma \,,$$

where $m_{t,u}$ is the number of tasks remaining unclassified in the (t,u) sub-round, and ℓ_t is the pre-determined number of workers assigned to each of these tasks in that round. Further, Algorithm 4 returns estimates $\{\hat{t}_i\}_{i \in [m]}$ that asymptotically achieve,

$$\lim_{m \to \infty} \frac{1}{m} \sum_{i=1}^{m} \mathbb{P}[t_i \neq \hat{t}_i] \ \ \leq \ \ C_1 e^{-(C_\delta/4)(\Gamma/m)\lambda\sigma^2} \,, \tag{4.15}$$

if $(\Gamma/m)\lambda\sigma^2 = \Theta(1)$, *where* $C_1 = \log_2(2\delta_{\max}/\delta_{\min})\log_2(2\lambda_{\max}/\lambda_{\min})$, *and*

$$\lim_{m\to\infty} \frac{1}{m}\sum_{i=1}^{m} \mathbb{P}[t_i \neq \hat{t}_i] \;\; = \;\; 0\,, \tag{4.16}$$

if $(\Gamma/m)\lambda\sigma^2 = \omega(1)$.

A proof of this theorem is provided in Section 4.2.4.

4.2.4 Proof of Theorem 4.2

First we show that the messages returned by Algorithm 5 are normally distributed and identify their conditional means and conditional variances in the following lemma. Assume in a sub-round (t,u), $t \in [T], u \in [s_t]$, the number of tasks remaining unclassified are $m_{t,u}$ and the task assignment is performed according to an (ℓ_t, r_t)-regular random graph. To simplify the notation, let $\hat{\ell}_t \equiv \ell_t - 1$, $\hat{r}_t \equiv r_t - 1$, and recall $\mu = \mathbb{E}[2p_j - 1]$, $\sigma^2 = \mathbb{E}_{\mathcal{P}}[(2p_j-1)^2]$. Note that μ, σ^2 remain same in each round. Let $\rho_{t,u}^2 = (1/|M|)\sum_{i\in[M]}\lambda_i$ be the exact value of average task difficulty of the tasks present in the (t,u) sub-round. When ℓ_t and r_t are increasing with the problem size, the messages converge to a Gaussian distribution due to the central limit theorem. We provide a proof of this lemma in Section 4.6.5.

Lemma 4.3. *Suppose for* $\ell_t = \Theta(\log m_{t,u})$ *and* $r_t = \Theta(\log m_{t,u})$, *tasks are assigned according to* (ℓ_t, r_t)*-regular random graphs. In the limit* $m_{t,u} \to \infty$, *if* $\mu > 0$, *then after* $k = \Theta(\sqrt{\log m_{t,u}})$ *number of iterations in Algorithm 5, the conditional mean* $\mu_q^{(k)}$ *and the conditional variance* $\big(\rho_q^{(k)}\big)^2$ *conditioned on the task difficulty* q *of the message* x_i *corresponding to the task* i *returned by the Algorithm 5 are*

$$\begin{aligned}
&\mu_q^{(k)} = (2q-1)\mu\ell_t(\hat{\ell}_t\hat{r}_t\rho_{t,u}^2\sigma^2)^{(k-1)}\;,\\
&\big(\rho_q^{(k)}\big)^2\\
&= \mu^2\ell_t(\hat{\ell}_t\hat{r}_t\rho_{t,u}^2\sigma^2)^{2(k-1)}\Bigg(\rho_{t,u}^2 - (2q-1)^2\\
&+\frac{\rho_{t,u}^2\hat{\ell}_t(1-\rho_{t,u}^2\sigma^2)(1+\hat{r}_t\rho_{t,u}^2\sigma^2)\big(1-(\hat{\ell}_t\hat{r}_t\rho_{t,u}^4\sigma^4)^{-(k-1)}\big)}{\hat{\ell}_t\hat{r}_t\rho_{t,u}^4\sigma^4 - 1}\Bigg)\\
&+\ell_t(2-\mu^2\rho_{t,u}^2)(\hat{\ell}_t\hat{r}_t)^{k-1}\,.
\end{aligned} \tag{4.17}$$

We will show in (4.56) that the probability of misclassification for any task in sub-round (t,u) in Algorithm 4 is upper bounded by $e^{-(C_\delta/4)(\Gamma/m)\lambda\sigma^2}$. Since, there are at most $C_1 = s_{\max}T \le \log_2(2\delta_{\max}/\delta_{\min})\log_2(2\lambda_{\max}/\lambda_{\min})$ rounds, using union bound we get the desired probability of error. In (4.60), we show that the expected total number of worker assignments across all rounds is at most Γ.

Let's consider any task $i \in [m]$ having difficulty λ_i. Without loss of generality assume that $t_i = 1$ that is $q_i > 1/2$. Let us assume that the task i gets classified in the (t,u) sub-round, $t \in [T], u \in [s_t]$. That is the number of workers assigned to the task i when it gets classified is $\ell_t = C_\delta(\Gamma/m)(\widehat{\lambda}/\lambda_t)$ and the threshold $\mathcal{X}_{t,u}$ set in that round for classification is $\mathcal{X}_{t,u} = \sqrt{\lambda_t}\mu\ell_t\big((\ell_t-1)(r_t-1)\rho_{t,u}^2\sigma^2\big)^{k_t-1}$. From Lemma 4.10 the message x_i returned by Algorithm 5 is Gaussian with conditional mean and conditional variance as given in (4.51). Therefore in the limit of m, the probability of error in task i is

$$\begin{aligned}
\lim_{m\to\infty}\mathbb{P}\big[\hat{t}_i \neq t_i | q_i\big] &= \lim_{m\to\infty}\mathbb{P}\big[x_i < -\mathcal{X}_{t,u}|q_i\big] \\
&= \lim_{m\to\infty} Q\Big(\frac{\mu_{q_i}^{(k)} + \mathcal{X}_{t,u}}{\rho_{q_i}^{(k)}}\Big) &(4.18)\\
&\le \lim_{m\to\infty}\exp\Big(\frac{-(\mu_{q_i}^{(k)} + \mathcal{X}_{t,u})^2}{2(\rho_{q_i}^{(k)})^2}\Big) &(4.19)\\
&= \exp\Big(\frac{-((2q_i-1)+\sqrt{\lambda_t})^2\ell_t\sigma^2}{2(1-(2q_i-1)^2\sigma^2)}\Big) &(4.20)\\
&\le \exp\Big(\frac{-\lambda_t\ell_t\sigma^2}{2}\Big)\\
&= \exp\Big(\frac{-C_\delta(\Gamma/m)\widehat{\lambda}\sigma^2}{2}\Big) &(4.21)\\
&\le \exp\Big(\frac{-C_\delta(\Gamma/m)\lambda\sigma^2}{4}\Big), &(4.22)
\end{aligned}$$

where $Q(\cdot)$ in (4.52) is the tail probability of a standard Gaussian distribution, and (4.53) uses the Chernoff bound. (4.54) follows from substituting conditional mean and conditional variance from Equation (4.51), and using $\ell_t = \Theta(\log m_{t,u})$, $k = \Theta(\sqrt{\log m_{t,u}})$ where m grows to infinity. (4.55) uses $\ell_t = C_\delta(\Gamma/m)(\widehat{\lambda}/\lambda_t)$, our choice of ℓ_t in Algorithm 4 line 4. (4.56) uses the fact that for the quantized distribution $\{\lambda_a,\delta_a\}_{a\in[T]}$, $\widehat{\lambda} = \big(\sum_{a\in[T]}(\delta_a/\lambda_a)\big)^{-1} \ge \lambda/2$. We have established that our approach guarantees the desired level of accuracy. We are left to show that we use at most Γ assignments in expectation.

We upper bound the expected total number of workers used for tasks of quantized difficulty level λ_a's for each $1 \leq a \leq T$. Recall that our adaptive algorithm runs in T rounds indexed by t, where each round t further runs s_t sub-rounds. The total expected number of workers assigned to δ_a fraction of tasks of quantized difficulty λ_a in $t = 1$ to $t = a-1$ rounds is upper bounded by $m\delta_a \sum_{t=1}^{a-1} s_t \ell_t$. The upper bound assumes the worst-case (in terms of the budget) that these tasks do not get classified in any of these rounds as the threshold $\mathcal{X}$ set in these rounds is more than absolute value of the conditional mean message x of these tasks.

Next, in $s_{t=a}$ sub-rounds the threshold $\mathcal{X}$ is set less than or equal to the absolute value of the conditional mean message x of these tasks, i.e. $\mathcal{X} \leq |\mu_{q_a}^{(k)}|$ for $(2q_a - 1)^2 = \lambda_a$. Therefore, in each of these s_a sub-rounds, probability of classification of these tasks is at least $1/2$. That is the expected total number of workers assigned to these tasks in s_a sub-rounds is upper bounded by $2m\delta_a\ell_a$. Further, s_a is chosen such that the fraction of these tasks remaining un-classified at the end of s_a sub-rounds is at most same as the fraction of the tasks having difficulty λ_{a+1}. That is to get the upper bound, we can assume that the fraction of λ_{a+1} difficulty tasks at the start of s_{a+1} sub-rounds is $2\delta_{a+1}$, and the fraction of λ_a difficulty tasks at the start of s_{a+1} sub-rounds is zero. Further, recall that we have set $s_T = 1$ as in this round our threshold $\mathcal{X}$ is equal to zero. Therefore, we have the following upper bound on the expected total number of worker assignments.

$$\begin{aligned}
\sum_{i=1}^{m} \mathbb{E}[|W_i|] &\leq 2m\delta_1\ell_1 + \sum_{a=2}^{T-1} 4m\delta_a\ell_a + 2m\delta_T\ell_T + \sum_{a=2}^{T}\Big(m\delta_a \sum_{b=1}^{a-1} s_b\ell_b\Big) \\
&\leq \sum_{a=1}^{T} 4m\delta_a\ell_a + s_{\max}\sum_{a=1}^{T} m\delta_a\ell_a \qquad (4.23)\\
&\leq (4 + \lceil \log(2\delta_{\max}/\delta_{\min})\rceil) \sum_{a=1}^{T} m\delta_a\ell_a \qquad (4.24)\\
&\leq (4 + \lceil \log(2\delta_{\max}/\delta_{\min})\rceil)\Gamma C_\delta \qquad (4.25)\\
&= \Gamma\,, \qquad (4.26)
\end{aligned}$$

Equation (4.57) uses the fact that $\ell_t = (C_\delta(\Gamma/m)(\widehat{\lambda}/\lambda_t)$ where λ_t's are separated apart by at least a ratio of 2 (recall the quantized distribution), therefore $\sum_{t=1}^{a-1} \ell_t \leq \ell_a$. Equation (4.58) follows from the choice of s_t's in the algorithm. Equation (4.59) follows from using $\ell_t = (C_\delta(\Gamma/m)(\widehat{\lambda}/\lambda_t)$ and $\lambda = (\sum_{a\in[T]}(\delta_a/\lambda_a))^{-1}$, and Equation (4.60) uses $C_\delta = (4 + \lceil \log(2\delta_{\max}/\delta_{\min})\rceil)^{-1}$.

4.2.5 Upper bound on the achievable error rate

We present an adaptive task assignment scheme and an iterative inference algorithm that asymptotically achieve an error rate of $C_1 e^{-(C_\delta/4)(\Gamma/m)\lambda\sigma^2}$, when the number of tasks m grows large and the expected budget is increasing as $\Gamma = \Theta(m \log m)$ where

$$C_1 = \log_2(2\delta_{\max}/\delta_{\min}) \log_2(2\lambda_{\max}/\lambda_{\min})$$

and C_δ is a constant that only depends on $\{\delta_a\}_{a\in[T]}$. This matches the lower bound in (4.13) when C_1 and C_δ are $O(1)$. Comparing it to a fundamental lower bound in Theorem 4.1 establishes the near-optimality of our approach, and the sufficient condition to achieve average error ε is for the average total budget to be larger than,

$$\Gamma_\varepsilon \quad \geq \quad \frac{4}{C_\delta} \frac{m}{\lambda\sigma^2} \log\left(\frac{C_1}{\varepsilon}\right) . \tag{4.27}$$

Our proposed adaptive approach in Algorithm 4 takes as input the number of tasks m, a target budget Γ, hyper parameter C_δ to be determined by our theoretical analyses in Theorem 4.4, the quantized prior distribution $\widehat{\mathcal{Q}}$, the statistics μ and σ^2 on the worker prior $\mathcal{P}$. The proposed scheme makes at most Γ queries in expectation to the crowd and outputs the estimated labels $\hat{t}_i$'s for all the tasks $i \in [m]$.

The proposed adaptive approach: overview.

At a high level, our approach works in T *rounds* indexed by $t \in [T]$, the support size of the quantized distribution $\widehat{\mathcal{Q}}$, and s_t *sub-rounds* at each round t, where s_t is chosen by the algorithm in line 5. In each sub-round, we perform both task assignment and inference, sequentially. Guided by the inference algorithm, we permanently label a subset of the tasks and carry over the

Algorithm 4 Adaptive Task Assignment and Inference Algorithm

```
Require: number of tasks $m$, allowed budget $\Gamma$, hyper parameter $C_\delta$, quantized prior distribution $\{\lambda_a, \delta_a\}_{a\in[T]}$, collective quality of the workers $\sigma^2$, average reliability $\mu$
Ensure: Estimated labels $\{\hat{t}_i\}_{i\in[m]}$
 1: $M \leftarrow \{1, 2, \cdots, m\}$
 2: $\widehat{\lambda} \leftarrow \Big(\sum_{a\in[T]}(\delta_a/\lambda_a)\Big)^{-1}$
 3: for all $t = 1, 2, \cdots, T$ do
 4:   $\ell_t \leftarrow (C_\delta\widehat{\lambda}\;\Gamma)/(m\,\lambda_t)$ , $r_t \leftarrow \ell_t$
 5:   $s_t \leftarrow \max\Big\{0, \Big\lceil\log\Big(\frac{2\delta_t}{\delta_{t+1}}\Big)\Big\rceil\Big\}\mathbb{I}\{t<T\} + 1\,\mathbb{I}\{t=T\}$
 6:   for all $u = 1, 2, \cdots, s_t$ do
 7:     if $M \neq \varnothing$ then
 8:       $n \leftarrow |M|$ ,  $k \leftarrow \sqrt{\log|M|}$
 9:       Draw $E \in \{0,1\}^{|M|\times n} \sim (\ell_t, r_t)$-regular random graph
10:       Collect answers $\{A_{i,j} \in \{1,-1\}\}_{(i,j)\in E}$
11:       $\{x_i\}_{i\in M} \leftarrow$ Algorithm 5 $\big[E, \{A_{i,j}\}_{(i,j)\in E}, k\big]$
12:       $\rho^2_{t,u} \leftarrow$ Algorithm 8 $[E, \{A_{i,j}\}_{(i,j)\in E}, \ell_t, r_t]$
13:       $\mathcal{X}_{t,u} \leftarrow \sqrt{\lambda_t}\mu\ell_t\big((\ell_t-1)(r_t-1)\rho^2_{t,u}\sigma^2\big)^{k-1}\mathbb{I}\{t<T\} + 0\,\mathbb{I}\{t=T\}$
14:       for $i \in M$ do
15:         if $x_i > \mathcal{X}_{t,u}$ then
16:           $\hat{t}_i \leftarrow +1$
17:         else if $x_i < -\mathcal{X}_{t,u}$ then
18:           $\hat{t}_i \leftarrow -1$
19:         end if
20:       end for
21:       $M \leftarrow \{i \in M : |x_i| \leq \mathcal{X}_{t,u}\}$
22:     end if
23:   end for
24: end for
```

remaining ones to subsequent sub-rounds. *Inference* is done in line 11 to get a confidence score x_i's on the tasks $i \in M$, where $M \subseteq [m]$ is the set of tasks that are remaining to be labelled at the current sub-round. The *adaptive task assignment* of our approach is entirely managed by the choice of this set M in line 21, as only those tasks in M will be assigned new workers in the next sub-round in lines 9 and 10.

At each round, we choose how many responses to collect for each task present in that round as prescribed by our theoretical analysis. Given this choice of ℓ_t, the number of responses collected for each task at round t, we repeat the key inner-loop in line 9-21 of Algorithm 4. In round t the sub-round is repeated s_t times to ensure that sufficient number of 'easy' tasks are classified. Given a set M of remaining tasks to be labelled, the sub-round collects ℓ_t response per task on those tasks in M and runs an inference algorithm (Algorithm 5) to give confidence scores x_i's to all $i \in M$. Our theoretical analysis prescribes a choice of a threshold $\mathcal{X}_{t,u}$ to be used in round $t \in [T]$ sub-round $u \in [s_t]$. All tasks in M with confidence score larger than $\mathcal{X}_{t,u}$ are permanently labelled as positive tasks, and those with confidence score less than $-\mathcal{X}_{t,u}$ are permanently labelled as negative tasks. Those permanently labelled tasks are referred to as 'classified' and removed from the set M. The remaining tasks with confidence scores between $\mathcal{X}_{t,u}$ and $-\mathcal{X}_{t,u}$ are carried over to the next sub-round. The confidence scores are designed such that the sign of x_i provides the estimated true label, and we are more confident about this estimated label if the absolute value of the score x_i is larger. The art is in choosing the appropriate number of responses to be collected for each task ℓ_t and the threshold $\mathcal{X}_{t,u}$, and our theoretical analyses, together with the provided statistics of the prior distribution $\mathcal{P}$, and the prior quantized distribution $\widehat{\mathcal{Q}}$ allow us to choose the ones that achieve a near optimal performance.

Note that we are mixing inference steps and task assignment steps. Within each sub-round, we are performing both task assignment and inference. Further, the inner-loop within itself uses a non-adaptive task assignment, and hence our approach is a series of non-adaptive task assignments with inference in each sub-round. However, Algorithm 4 is an adaptive scheme, where the adaptivity is fully controlled by the set of remaining unclassified tasks M. We are adaptively choosing which tasks to carry over in the set M based on all the responses we have collected thus far, and we are assigning more workers to only those tasks in M.

Since difficulty levels are varying across the tasks, it is intuitive to assign fewer workers to easy tasks and more workers to hard tasks. Supposing that we know the difficulty levels λ_i's, we could choose to assign the ideal number of workers to each task according to λ_i's. However, the difficulty levels are not known. The proposed approach starts with a smaller budget in the first round classifying easier tasks, and carries over the more difficult tasks to the later rounds where more budget per task will be assigned.

The proposed adaptive approach: precise.

More precisely, given a budget Γ and the statistics of $\mathcal{P}$, and the known quantized distribution $\widehat{\mathcal{Q}}$ we know what target probability of error to aim for, say ε, from Theorem 4.4. The main idea behind our approach is to allocate the given budget Γ over multiple rounds appropriately, and at each round get an estimate of the labels of the remaining tasks in M and also the confidence scores, such that with an appropriate choice of the threshold $\mathcal{X}_{t,u}$ those tasks we choose to classify in the current round achieve the desired target error rate of $\mathbb{P}[t_i \neq \hat{t}_i \,|\, |x_i| > \mathcal{X}_{t,u}] \leq \varepsilon$. As long as this guarantee holds at each round for all classified tasks, then the average error rate will also be bounded by $(1/m)\sum_{i=1}^{m} \mathbb{P}[t_i \neq \hat{t}_i] \leq \varepsilon$ when the process terminates eventually. The only remaining issue is how many queries are made in total when this process terminates. We guarantee that in expectation at most Γ queries are made under our proposed choices of ℓ_t's and $\mathcal{X}_{t,u}$'s in the algorithm.

At round zero, we initially put all the tasks in $M = [m]$. A fraction of tasks are permanently labelled in each round and the un-labelled ones are taken to the next round. At round $t \in \{1, \dots, T\}$, our goal is to classify a sufficient fraction of those tasks in the t-th difficulty group $\{i \in M \,|\, \lambda_i \in [(1/2)\lambda_t, \lambda_t]\}$ with the desired level of accuracy. The art is in choosing the right number of responses to be collected per task ℓ_t for that round and also the right threshold $\mathcal{X}_{t,u}$ on the confidence score, to be used in the inner-loop in line 9-21 of Algorithm 4. If ℓ_t is too low and/or threshold $\mathcal{X}_{t,u}$ too small, then misclassification rate will be too large. If ℓ_t is too large and/or $\mathcal{X}_{t,u}$ is too large, we are wasting our budget and achieving unnecessarily high accuracy on those tasks classified in the current round, and not enough tasks will be classified in that round. We choose ℓ_t and $\mathcal{X}_{t,u}$ appropriately to ensure that the misclassification probability is at most $C_1 e^{-(C_\delta/4)(\Gamma/m)\lambda\sigma^2}$ based on our analysis (see (4.56)) of the inner-loop. We run the identical sub-rounds $s_t = \max\{0, \lceil \log_2(2\delta_t/\delta_{t+1})\rceil\}$ times to ensure that enough fraction of tasks with difficulty $\lambda_i \in [(1/2)\lambda_t, \lambda_t]$ are classified. Precisely, the choice of s_t insures that the expected number of tasks with difficulty $\lambda_i \in [(1/2)\lambda_t, \lambda_t]$ remaining unclassified after t-th round is at most equal to the number of tasks in the next group, i.e., difficulty level $\lambda_i \in [(1/2)\lambda_{t+1}, \lambda_{t+1}]$.

Note that statistically, the fraction of the t-th group (i.e. tasks with difficulty $[\lambda_{t+1}, \lambda_t]$) that get classified before the t-th round is very small as the threshold set in these rounds is more than their absolute mean message. Most tasks in the t-th group will get classified in round t. Further, the proposed pre-processing step of binning the tasks ensures that $\ell_{t+1} \geq 2\ell_t$. This ensures that the total extraneous budget spent on the t-th group of tasks is not more than a constant times the allocated budget on those tasks.

The main algorithmic component is the inner-loop in line 9-21 of Algorithm 4. For a choice of the (per task) budget ℓ_t, we collect responses according to a $(\ell_t, r_t = \ell_t)$-regular random graph on $|M|$ tasks and $|M|$ workers. The leading eigen-vector of the non-backtracking operator on this bipartite graph, weighted by the ± 1 responses reveals a noisy observation of the true class and the difficulty levels of the tasks. The non-backtracking matrix B of a directed graph G is indexed by its directed edges such that for any $k \geq 1$, B_{ef}^k counts the number of non-backtracking walks of $k+1$ edges on G starting with the directed edge e and ending with f. Let $x \in \mathbb{R}^{|M|}$ denote this top left eigenvector, computed as per the message-passing algorithm of Algorithm 5. Then the i-th entry x_i asymptotically converges in the large number of tasks m limit to a Gaussian random variable with mean proportional to the difficulty level $(2q_i - 1)$, with mean and variance specified in Lemma 4.10. This non-backtracking operator approach to crowdsourcing was first introduced in [105] for the standard DS model. We generalize their analysis to this generalized DS model in Theorem 4.5 for finite sample regime, and further give a sharper characterization based on central limit theorem in the asymptotic regime (Lemma 4.10). For a detailed explanation of Algorithm 5 and its analyses, we refer to Section 4.3.

Justification of the choice of ℓ_t and $\mathcal{X}_{t,u}$.

The main idea behind our approach is to allocate a target budget to each i-th task according to its quantized difficulty λ_t where t is such that $\lambda_i \in [(1/2)\lambda_t, \lambda_t]$. Given a total budget Γ and the quantized distribution $\widehat{\mathcal{Q}}$ which gives the collective difficulty of tasks λ (line 2, Algorithm 4), we target to assign $(\widehat{\lambda}/\lambda_t)(\Gamma/m)$ workers to a task of quantized difficulty λ_t. This choice of the target budget is motivated from the proof of the lower bound Theorem 4.1. If we had identified the tasks with respect to their difficulty then the near-optimal choice of the budget that achieves the lower bound is given in (4.49). Our target budget is a simplified form of the near-optimal choice and

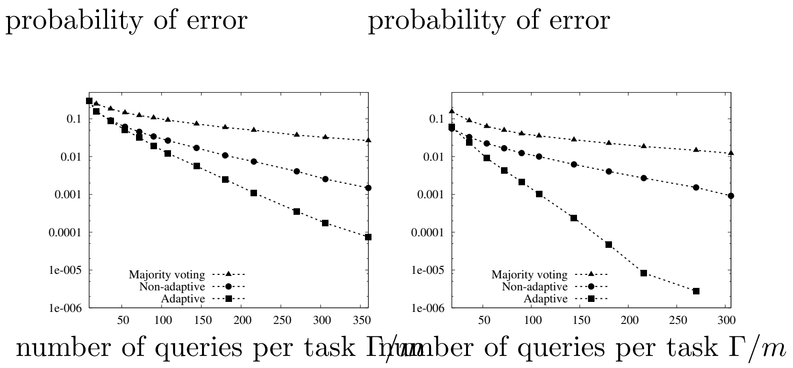


Figure 4.1: Algorithm 4 improves significantly over its non-adaptive version and majority voting with an adaptive task assignment for tasks with $\lambda = 1/7$ (left) and $\lambda = 4/13$ (right).

ignores the constant part that does not depend upon the total budget. This choice of the budget would give the equal probability of misclassification for the tasks of varying difficulties. We refer to this error rate as the desired probability of misclassification. As we do not know which tasks belong to which quantized difficulty group λ_t, a factor of $1/C_\delta$ is needed to compensate for the extra budget needed to infer those difficulty levels. This justifies our choice of budget in line 4 of the Algorithm 4.

From our theoretical analysis of the inner loop, we know the probability of misclassification for a task that belongs to difficulty group λ_t as a function of the classification threshold $\mathcal{X}$ and the budget that is assigned to it. Therefore, in each round we set the classification threshold $\mathcal{X}_{t,u}$ such that even the possibly most difficult task achieves the desired probability of misclassification. This choice of $\mathcal{X}_{t,u}$ is provided in line 13 of Algorithm 4.

Numerical experiments.

In Figure 4.1, we compare the performance of our algorithm with majority voting and also a non-adaptive version of our Algorithm 4, where we assign to each task $\ell = \Gamma/m$ number of workers in one round and set classification threshold $\mathcal{X}_{1,1} = 0$ so as to classify all the tasks (choosing $T = 1$ and $s_1 = 1$). Since this performs the non-adaptive inner-loop *once*, this is a non-adaptive algorithm, and has been introduced for the standard DS model in [107].

For numerical experiments, we make a slight modification to our proposed

Algorithm 4. In the final round, when the classification threshold is set to zero, we include all the responses collected thus far when running the message passing Algorithm 5, and not just the fresh samples collected in that round. This creates dependencies between rounds, which makes the analysis challenging. However, in practice we see improved performance and it allows us to use the given fixed budget efficiently.

We run synthetic experiments with $m = 1800$ and fix $n = 1800$ for the non-adaptive version. The crowds are generated from the spammer-hammer model where a worker is a hammer ($p_j = 1$) with probability 0.3 and a spammer ($p_j = 1/2$) otherwise. In the left panel, we take difficulty level λ_a to be uniformly distributed over $\{1, 1/4, 1/16\}$, that gives $\lambda = 1/7$. In the right panel, we take $\lambda_a = 1$ with probability $3/4$, otherwise we take it to be $1/4$ or $1/16$ with equal probability, that gives $\lambda = 4/13$. Our adaptive algorithm improves significantly over its non-adaptive version, and our main results in Theorems 4.4 and 4.5 predicts such gain of adaptivity. In particular, for the left panel, the non-adaptive algorithm's error scaling depends on smallest λ_{min} that is $1/16$ while for the adaptive algorithm it scales with $\lambda = 1/7$. In the left figure, it can be seen that the adaptive algorithm requires approximately a factor of $\lambda_{\text{min}}/\lambda = 7/16$ more queries to achieve the same error as achieved by the non-adaptive scheme. For example, non-adaptive version of Algorithm 4 requires $\Gamma/m = 360$ to achieve error rate 0.002, whereas the adaptive approach only requires $180 \simeq 360 \times 7/16 = 157.5$. Quantifying such a gap is one of our main results in Theorems 4.4 and 4.5. This gap widens in the right panel to approximately $\lambda_{\text{min}}/\lambda = 13/64$ as predicted. For a fair comparison with the non-adaptive version, we fix the total budget to be Γ and assign workers in each round until the budget is exhausted, such that we are strictly using budget at most Γ deterministically.

Performance Guarantee

Algorithm 4 is designed in such a way that we are not wasting any budget on any of the tasks; we are not getting unnecessarily high accuracy on easier tasks, which is the root cause of inefficiency for non-adaptive schemes. In order to achieve this goal, the internal parameter $\rho_{t,u}^2$ computed in line 12 of Algorithm 4 has to satisfy $\rho_{t,u}^2 = (1/|M|)\sum_{i\in[M]} \lambda_i$, which is the average difficulty of the remaining tasks. Such a choice is important in choosing the right threshold $\mathcal{X}_{t,u}$.

As the set M of remaining tasks is changing over the course of the algorithm, we need to estimate this value in each sub-routine. We provide an estimator of $\rho^2_{t,u}$ in Algorithm 8 that only uses the sampled responses that are already collected. All numerical results are based on this estimator. However, analyzing the sensitivity of the performance with respect to the estimation error in $\rho^2_{t,u}$ is quite challenging, and for a theoretical analysis, we assume we have access to an oracle that provides the exact value of $\rho^2_{t,u} = (1/|M|)\sum_{i\in[M]} \lambda_i$, replacing Algorithm 8.

Theorem 4.4. *Suppose Algorithm 8 returns the exact value of*

$$\rho^2_{t,u} = (1/|M|)\sum_{i\in M} \lambda_i \, .$$

With the choice of $C_\delta = (4 + \lceil\log(2\delta_{\max}/\delta_{\min})\rceil)^{-1}$, for any given quantized prior distribution of task difficulty $\{\lambda_a, \delta_a\}_{a\in[T]}$ such that $\delta_{\max}/\delta_{\min} = O(1)$ and $\lambda_{\max}/\lambda_{\min} = O(1)$, and the budget $\Gamma = \Theta(m \log m)$, the expected number of queries made by Algorithm 4 is asymptotically bounded by

$$\lim_{m\to\infty} \sum_{t\in[T],u\in[s_t]} \ell_t \, \mathbb{E}[m_{t,u}] \quad \le \quad \Gamma \, ,$$

where $m_{t,u}$ is the number of tasks remaining unclassified in the (t,u) sub-round, and ℓ_t is the pre-determined number of workers assigned to each of these tasks in that round. Further, Algorithm 4 returns estimates $\{\hat{t}_i\}_{i\in[m]}$ that asymptotically achieve,

$$\lim_{m\to\infty} \frac{1}{m}\sum_{i=1}^{m} \mathbb{P}[t_i \ne \hat{t}_i] \quad \le \quad C_1 e^{-(C_\delta/4)(\Gamma/m)\lambda\sigma^2} \, , \tag{4.28}$$

if $(\Gamma/m)\lambda\sigma^2 = \Theta(1)$, where $C_1 = \log_2(2\delta_{\max}/\delta_{\min})\log_2(2\lambda_{\max}/\lambda_{\min})$, and

$$\lim_{m\to\infty} \frac{1}{m}\sum_{i=1}^{m} \mathbb{P}[t_i \ne \hat{t}_i] \quad = \quad 0 \, , \tag{4.29}$$

if $(\Gamma/m)\lambda\sigma^2 = \omega(1)$.

A proof of this theorem is provided in Section 4.6.4. In this theoretical analysis, we are considering a family of problem parameters $(m, \mathcal{Q}, \mathcal{P}, \Gamma)$ in an increasing number of tasks m. All the problem parameters $\mathcal{Q}$, $\mathcal{P}$, and Γ can vary as functions of m. For example, consider a family of $\mathcal{Q}(q_i) = (1/2)\mathbb{I}(q_i = 0) + (1/2)\mathbb{I}(q_i = 1)$ independent of m and $\mathcal{P}(p_j) = (1 - 1/\sqrt{m})\mathbb{I}(p_j = 0.5) + (1/\sqrt{m})\mathbb{I}(p_j = 1)$. As m grows, most of the workers are spammers giving completely random answers. In this setting, we can ask how should the budget grow with m, in order to achieve a target accuracy of, say, e^{-5}? We have $\lambda = 1$ and $\sigma^2 = 1/\sqrt{m}$, indicating that the collective difficulty is constant but collective quality of the workers are decreasing in m. It is a simple calculation to show that $C_1 = 1$ and $C_\delta = 1/5$ in this case, and the above theorem proves that $\Gamma = 100m^{3/2}$ is sufficient to achieve the desired error rate. Further such dependence of the budget in m is also necessary, as follows from our lower bound in Theorem 4.1.

Consider now a scenario where we have tasks with increasing difficulties in m. For example, $\mathcal{Q}(q_i) = (1/4)\mathbb{I}(q_i = 1/2 + 1/\log m) + (1/4)\mathbb{I}(q_i = 1/2 - 1/\log m) + (1/4)\mathbb{I}(q_i = 1/2 + 2/\log m) + (1/4)\mathbb{I}(q_i = 1/2 - 2/\log m)$ and $\mathcal{P}(p_j) = \mathbb{I}(p_j = 3/4)$. We have $\lambda = 32/(5(\log m)^2)$ and $\sigma^2 = 1/4$. It follows from simple calculations that $C_1 = 2$ and $C_\delta = 1/5$. It follows that it is sufficient and necessary to have budget scaling in this case as $\Gamma = \Theta(m(\log m)^2)$.

For families of problem parameters for increasing m, we give asymptotic performance guarantees. Finite regime of m is challenging as our analysis relies on a version of central limit theorem and the resulting asymptotic distribution of the score value x_i's. However, the numerical simulations in Figure 4.1 suggests that the improvement of the proposed adaptive approach is significant for moderate values of m as well.

Our main result in Eq. (4.28) gives the sufficient condition of our approach in (4.27). Compared to the fundamental lower bound in Theorem 4.1, this proves the near-optimality of our adaptive approach. Under the regime considered in Theorem 4.4, it is necessary and sufficient to have budget scaling as $\Gamma = \Theta((m/(\lambda\sigma^2))\log(1/\varepsilon))$.

4.3 Analysis of the inner-loop and the minimax error rate under the non-adaptive scenario

In this section, we provide the analysis of the non-adaptive task assignment and inference algorithm in the sub-routine in line 9-21 of Algorithm 4. To simplify the notations, we consider the very first instance of the sub-round where we have a set $M = [m]$ of tasks to be labelled, and all the subsequent subroutines will follow similarly up to a change of notations. Perhaps surprisingly, we show that this inner-loop itself achieves near optimal performance for *non-adaptive schemes*. We show that $\Gamma = O((m/(\lambda_{\min}\sigma^2))\log(1/\varepsilon))$ is sufficient to achieve a target probability of error $\varepsilon > 0$ in Theorem 4.5. We show this is close to optimal by comparing it to a necessary condition that scales in the same way in Theorem 4.6. First, here is the detailed explanation of the inner-loop.

Task assignment (line 9 of Algorithm 4). Suppose we are given a budget of $\Gamma = m\ell$, so that each task can be assigned to ℓ workers on average. Further assume that each worker is assigned r tasks. We are analyzing a slightly more general setting than Algorithm 4 where $r = \ell$ for all instances. We follow the recipe of [107] and use a random regular graph for a non-adaptive task assignment. Namely, we know that we need to recruit $n = m\ell/r$ workers in total. Before any responses are collected, we make all the task assignments for all n workers in advance and store it in a bipartite graph $G([m], [n], E)$ where $[m]$ are the task nodes, $[n]$ are the worker nodes, and $E \subseteq [m] \times [n]$ is the collection of edges indicating that task i is assigned to worker j if $(i, j) \in E$. This graph E is drawn from a random regular graph with task degree ℓ and worker degree r. Such random graphs can be drawn efficiently, for example, using the configuration model [174].

Inference algorithm (line 11 of Algorithm 4). The message passing algorithm of Algorithm 5, is a state-of-the-art spectral method based on non-backtracking operators, first introduced for inference in [105]. A similar approach has been later applied to other inference problems, e.g. [122, 24]. This is a message passing algorithm that operates on two sets of messages: the task messages $\{x_{i\to j}\}_{(i,j)\in E}$ capturing how likely the task is to be a positive task and the worker messages $\{y_{j\to i}\}_{(i,j)\in E}$ capturing how reliable the worker is. Consider a data collected on m tasks and n workers such that $A \in$

$\{0,+1,-1\}^{m\times n}$ under the non-adaptive scenario with task assigned according to a random regular graph E of task degree ℓ and worker degree r. In each round, all messages are updated as

$$x_{i\to j} \;=\; \sum_{j'\in W_i\setminus j} A_{ij'}y_{j'\to i}\ ,\ \text{and} \tag{4.30}$$

$$y_{j\to i} \;=\; \sum_{i'\in T_j\setminus i} A_{i'j}x_{i'\to j}\ , \tag{4.31}$$

where $W_i \subseteq [n]$ is the set of workers assigned to task i, and $T_j \subseteq [m]$ is the set of workers assigned to worker j. The first is taking the weighted majority according to how reliable each worker is, and the second is updating the reliability according to how many times the worker agreed with what we believe. After a prefixed $k_{\max}$ iterations, we provide a confidence score by aggregating the messages at each task node $i \in [m]$:

$$x_i \;=\; \sum_{j'\in W_i} A_{ij'}y_{j'\to i}\ . \tag{4.32}$$

The precise description is given in Algorithm 5. Perhaps surprisingly, this algorithm together with the random regular task assignment achieve the minimax optimal error rate among all non-adaptive schemes. This will be made precise in the upper bound in Theorem 4.5 and a fundamental lower bound in Theorem 4.6. An intuitive explanation of why this algorithm works is provided in Section 4.4 via spectral interpretation of this approach.

4.3.1 Performance guarantee

For this non-adaptive scenario, we provide a sharper upper bound on the achieved error, that holds for all (non-asymptotic) regimes of m. Define σ_k^2 as

$$\sigma_k^2 \;\equiv\; \frac{2\sigma^2}{\mu^2\big(\hat{\ell}\hat{r}(\rho^2\sigma^2)^2\big)^{k-1}} + 3\Big(1+\frac{1}{\hat{r}\rho^2\sigma^2}\Big)\frac{1-1/\big(\hat{\ell}\hat{r}(\rho^2\sigma^2)^2\big)^{k-1}}{1-1/\big(\hat{\ell}\hat{r}(\rho^2\sigma^2)^2\big)}\ , \tag{4.33}$$

where $\hat{\ell}=\ell-1$, $\hat{r}=r-1$, $\mu=\mathbb{E}_{\mathcal{P}}[2p_j-1]$, $\sigma^2=\mathbb{E}_{\mathcal{P}}[(2p_j-1)^2]$, and $\rho^2=\mathbb{E}_{\mathcal{Q}}[(2q_i-1)^2]$. This captures the effective variance in the sub-Gaussian tail of the messages x_i's after k iterations of Algorithm 5, as shown in the

Algorithm 5 Message-Passing Algorithm

```
Require: $E \in \{0,1\}^{|M| \times n}$, $\{A_{ij} \in \{1,-1\}\}_{(i,j)\in E}$, $k_{\max}$
Ensure: $\{x_i \in \mathbb{R}\}_{i \in [|M|]}$
for all $(i,j) \in E$ do
    Initialize $y^{(0)}_{j\to i}$ with a Gaussian random variable $Z_{j\to i} \sim \mathcal{N}(1,1)$
end for
for all $k = 1, 2, \cdots, k_{\max}$ do
    for all $(i,j) \in E$ do
        $x^{(k)}_{i\to j} \leftarrow \sum_{j' \in W_i \setminus j} A_{ij'} y^{(k-1)}_{j'\to i}$
    end for
    for all $(i,j) \in E$ do
        $y^{(k)}_{j\to i} \leftarrow \sum_{i' \in T_j \setminus i} A_{i'j} x^{(k)}_{i'\to j}$
    end for
end for
for all $i \in [m]$ do
    $x^{(k_{\max})}_i \leftarrow \sum_{j \in W_i} A_{ij} y^{(k_{\max}-1)}_{j\to i}$
end for
```

proof of the following theorem in Section 4.6.6.

Theorem 4.5. *For any $\ell > 1$ and $r > 1$, suppose m tasks are assigned according to a random (ℓ, r)-regular graph drawn from the configuration model. If $\mu > 0$, $\hat{\ell}\hat{r}\rho^4\sigma^4 > 1$, and $\hat{r}\rho^2 > 1$, then for any $t \in \{\pm 1\}^m$, the estimate $\hat{t}^{(k)}_i = \text{sign}(x^{(k)}_i)$ after k iterations of Algorithm 5 achieves*

$$\mathbb{P}\big[t_i \neq \hat{t}^{(k)}_i \big| \lambda_i\big] \quad \leq \quad e^{-\ell\sigma^2\lambda_i/(2\sigma_k^2)} + \frac{3\ell r}{m}(\hat{\ell}\hat{r})^{2k-2}. \tag{4.34}$$

Therefore, the average error rate is bounded by

$$\frac{1}{m}\sum_{i=1}^{m}\mathbb{P}[t_i \neq \hat{t}^{(k)}_i] \quad \leq \quad \mathbb{E}_{\mathcal{Q}}\Big[e^{\frac{-\ell\sigma^2\lambda_i}{2\sigma_k^2}}\Big] + \frac{3\ell r}{m}(\hat{\ell}\hat{r})^{2k-2}. \tag{4.35}$$

The second term, which is the probability that the resulting (ℓ, r)-regular random graph is not locally tree-like, can be made small for large m as long as $k = O(\sqrt{\log m})$ (which is the choice we make in Algorithm 4). Hence, the dominant term in the error bound is the first term. Further, when we run our algorithm for large enough numbers of iterations, σ_k^2 converges linearly to a finite limit $\sigma_\infty^2 \equiv \lim_{k\to\infty} \sigma_k^2$ such that

$$\sigma_\infty^2 \;=\; 3\Big(1+\frac{1}{\hat{r}\rho^2\sigma^2}\Big)\frac{(\hat{\ell}\hat{r}\rho^2\sigma^2)^2}{(\hat{\ell}\hat{r}\rho^2\sigma^2)^2-1}\;, \tag{4.36}$$

which is upper bounded by a constant for large enough $\hat{r}\rho^2\sigma^2$ and $\hat{\ell}$, for example $\hat{r}\rho^2\sigma^2 \geq 1$ and $\hat{\ell} \geq 2$. Hence, for a wide range of parameters, the average error in (4.35) is dominated by $\mathbb{E}_{\mathcal{Q}}\big[e^{-\ell\sigma^2\lambda_i/2\sigma_k^2}\big]$. When the fraction of tasks with worst-case difficulty $\lambda_{\min}$ is strictly positive, the error is dominated by them as illustrated in Figure 4.2. Hence, it is sufficient to have budget

$$\Gamma_\varepsilon \;\geq\; \frac{C''m}{\lambda_{\min}\sigma^2}\log(1/\varepsilon)\;, \tag{4.37}$$

to achieve an average error of $\varepsilon > 0$. Such a scaling is also necessary as we show in the next section.

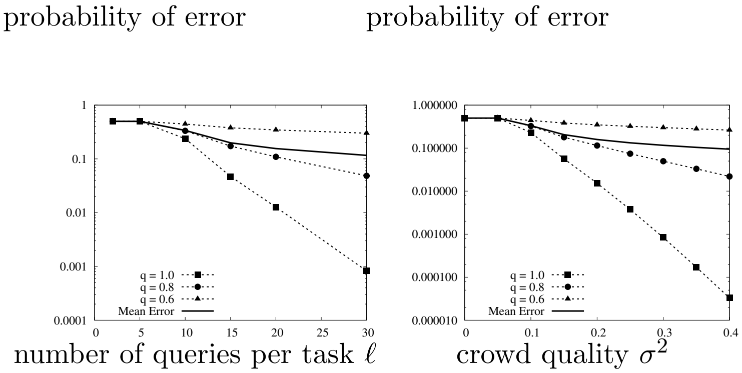


Figure 4.2: Non-adaptive schemes suffer as average error is dominated by difficult tasks. Dotted lines are error achieved by those tasks with the same quality q_i's, and the overall average error in solid line eventually has the same slope as the most difficult tasks with $q_i = 0.6$.

This is further illustrated in Figure 4.2. The error decays exponentially in ℓ and σ^2 as predicted, but the rate of decay crucially hinges on the individual difficulty level of the task being estimated. We run synthetic experiments with $m = n = 1000$ and the crowds are generated from the spammer-hammer model where $p_j = 1$ with probability σ^2 and $p_j = 1/2$ with probability $1-\sigma^2$, where the choice of this probability is chosen to match the collective difficulty $\sigma^2 = \mathbb{E}[(2p_j-1)^2]$. We fix $\sigma^2 = 0.3$ and vary ℓ in the left figure and fix $\ell = 30$ and vary σ^2 in the right figure. We let q_i's take values in $\{0.6, 0.8, 1\}$

with equal probability such that $\rho^2 = 1.4/3$. The error rate of each task grouped by their difficulty is plotted in the dashed lines, matching predicted $e^{-\Omega(\ell\sigma^2(2q_i-1)^2)}$. The average error rates in solid lines are dominated by those of the difficult tasks, which is a universal drawback for all non-adaptive schemes.

4.3.2 Fundamental limit under the non-adaptive scenario

Theorem 4.5 implies that it suffices to assign $\ell \geq (c/(\sigma^2\lambda_i))\log(1/\varepsilon)$ workers to achieve an error smaller than ε for a task i. We show in the following theorem that this scaling is also necessary when we consider all *non-adaptive* schemes. Even the best non-adaptive task assignment with the best inference algorithm still required budget scaling in the same way. Hence, applying *one round* of Algorithm 4 (which is a non-adaptive scheme) is near-optimal in the non-adaptive scenario compared to a minimax rate where the nature chooses the worst distribution of worker p_j's among the set of distributions with the same σ^2. We provide a proof of the theorem in Section 4.6.8.

Theorem 4.6. *There exists a positive constant C' and a distribution $\mathcal{P}$ of workers with average reliability $\mathbb{E}[(2p_j-1)^2] = \sigma^2$ s.t. when $\lambda_i < 1$, if the number of workers assigned to task i by any non-adaptive task assignment scheme is less than $(C'/(\sigma^2\lambda_i))\log(1/\epsilon)$, then no algorithm can achieve conditional probability of error on task i less than ϵ for any m and r.*

For formal comparisons with the upper bound, consider a case where the induced distribution on task difficulties λ_i's, $\widetilde{\mathcal{Q}}$, is same as its quantized version $\widehat{\mathcal{Q}}$ such that $\widetilde{\mathcal{Q}}(\lambda_i) = \sum_{a=1}^{T}\delta_a\mathbb{I}_{(\lambda_i=\lambda_a)}$. Since in this non-adaptive scheme, task assignments are done a priori, there are on average ℓ workers assigned to any task, regardless of their difficulty. In particular, if the total budget is less than

$$\Gamma_\varepsilon \quad \leq \quad C'\frac{m}{\lambda_{\min}\sigma^2}\log\frac{\delta_{\min}}{\varepsilon}, \tag{4.38}$$

then there will a a proportion of at least $\delta_{\min}$ tasks with error larger than $\varepsilon/\delta_{\min}$, resulting in overall average error to be larger than ε even if the rest of the tasks are error-free. Compared to the adaptive case in (4.14) (nearly achieved up to a constant factor in (4.27)), the gain of adaptivity is a factor of $\lambda/\lambda_{\min}$. When $\delta_{\min} < \varepsilon$, the above necessary condition is trivial as the RHS is negative. In such a case, the necessary condition can be tightened to $C'(m/\lambda_a\sigma^2)\log(\sum_{b=1}^{a}\delta_b/\varepsilon)$ where a is the smallest integer such that $\sum_{b=1}^{a}\delta_b > \varepsilon$.

4.4 Spectral interpretation of Algorithm 5 and parameter estimation

In this section, we give a spectral analysis of Algorithm 5, which leads to a spectral algorithm for estimating ρ^2 (Algorithm 8), to be used in the inner-loop of Algorithm 4. This spectral interpretation provides a natural explanation of how Algorithm 5 is extracting information and estimating the labels. Precisely, we are computing the top eigenvector of a matrix known as a weighted non-backtracking operator, via standard power method. Note that the above mapping is a *linear mapping* from the messages to the messages. This mapping, if formed into a $2|E| \times 2|E|$ dimensional matrix B is known as the non-backtracking operator. Precisely, for $(i,j),(i',j') \in E$,

$$B_{(i\to j),(j'\to i')} = \begin{cases} A_{i'j'} & \text{if } j = j' \text{ and } i \neq i' \ , \\ A_{i'j'} & \text{if } j \neq j' \text{ and } i = i' \ , \\ 0 & \text{otherwise} \ , \end{cases}$$

and the message update of Equations (4.30) and (4.31) are simply

$$\begin{bmatrix} x \\ y \end{bmatrix} \;=\; B \begin{bmatrix} x \\ y \end{bmatrix} \ ,$$

where x and y denote vectorizations of $x_{i\to j}$'s and $y_{i\to j}$'s. This is exactly the standard power method to compute the singular vector of the matrix $B \in \mathbb{R}^{2|E|\times 2|E|}$.

The spectrum, which is the set of eigenvalues of this square but non-symmetric matrix B illustrates when and why spectral method might work.

First consider decomposing the data matrix as

$$A \;=\; \underbrace{\mathbb{E}[A]}_{\text{true signal}} + \underbrace{(A - \mathbb{E}[A])}_{\text{random noise}} \;.$$

Simple analysis shows that $\mathbb{E}[A|q,p]$, where the expectation is taken with respect to the randomness in the graph and also in the responses, is a rank one matrix with spectral norm $\|\mathbb{E}[A|q,p]\| = \sqrt{\ell r \hat{\rho}^2 \hat{\sigma}^2}$, where

$$\hat{\rho}^2 \;\equiv\; \frac{1}{m}\sum_{i=1}^{m}(2q_i - 1)^2 \;, \quad \text{and} \;\; \hat{\sigma}^2 \;\equiv\; \frac{1}{n}\sum_{j=1}^{n}(2p_j - 1)^2 \;.$$

This is easy to see as $\mathbb{E}[A_{ij}|q,p] = (\ell/n)(2q_i - 1)(2p_j - 1)$. It follows that the expected matrix is $\mathbb{E}[A|q,p] = \sqrt{\ell\, r/(mn)}\,\sqrt{\hat{\rho}^2\hat{\sigma}^2 mn}\, uv^T$, where u and v are norm-one vectors with $u_i = (1/\sqrt{\sum_{i'\in[m]}(2q_{i'} - 1)^2})(2q_i - 1)$ and $v_j = (1/\sqrt{\sum_{j'\in[n]}(2p_{j'} - 1)^2})(2p_j - 1)$.

Also, typical random matrix analyses, such as those in [113, 106], show that the spectral norm (the largest singular value) of the noise matrix $(A - \mathbb{E}[A|q,p])$ is bounded by $C(\ell r)^{1/4}$ with some constant C. Hence, when the spectral norm of the signal is larger then that of the noise, i.e. $\|\mathbb{E}[A|q,p]\| > \|(A - \mathbb{E}[A|q,p])\|$, the top eigenvector of this matrix A corresponds to the true underlying signal, and we can hope to estimate the true labels from this top eigenvector. On the other hand, if $\|\mathbb{E}[A|q,p]\| < \|(A - \mathbb{E}[A|q,p])\|$, one cannot hope to recover any signal from the top eigenvector of A. This phenomenon is known as the spectral barrier.

This phenomenon is more prominent in the non-backtracking operator matrix B. Note that B is not symmetric and hence the eigen values are complex valued. Similar spectral analysis can be applied to show that when we are above the spectral barrier, the top eigenvalue is real-valued and concentrated around the mean $\Lambda_1(B) \simeq \sqrt{(\ell-1)(r-1)\hat{\rho}^2\hat{\sigma}^2}$ and the mode of the rest of the complex valued eigenvalues are bounded within a circle of radius: $|\Lambda_i(B)| \leq ((\ell-1)(r-1))^{1/4}$. Hence, the spectral barrier is exactly when $\Lambda_1(B) = |\Lambda_i(B)|$ which happens at $(\ell-1)(r-1)\hat{\rho}^4\hat{\sigma}^4 = 1$, and this plays a crucial role in the performance guarantee in Theorem 4.5. Note that because of the bipartite nature of the graph we are considering, we always have a pair of dominant eigenvalue as $\Lambda_1(B) = \sqrt{(\ell-1)(r-1)\hat{\rho}^2\hat{\sigma}^2}$ and

$\Lambda_2(B) = -\sqrt{(\ell-1)(r-1)\hat{\rho}^2\hat{\sigma}^2}$.

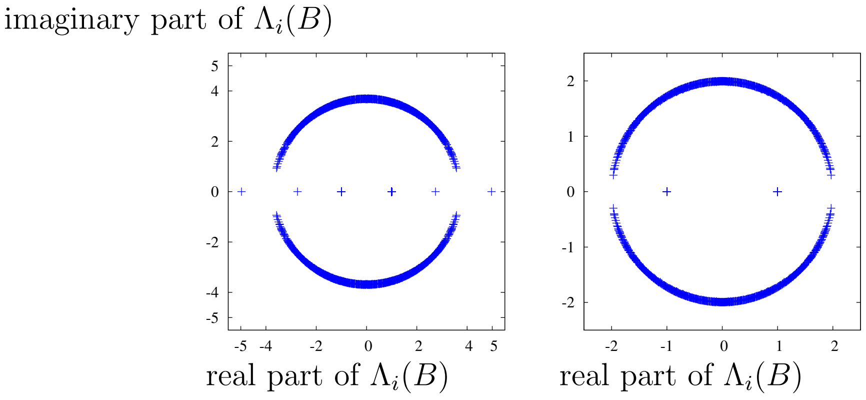


Figure 4.3: Scatter plot of the complex-valued eigenvalues of two realizations of non-backtracking matrix B of the model with $m = n = 300$, $\sigma^2 = 0.3, \rho^2 = 1.4/3$. On left for $\ell = 15$ and right for $\ell = 5$ which are above and below spectral barrier, respectively. We can clearly see the two top eigen values at (5,0) and (-5,0)

Figure 4.3 illustrates two sides of the spectral barrier. The one on the left shows the scatter plot of the complex valued eigen values of B. Notice a pair of top eigen values at $\sqrt{0.3 \times (1.4/3) \times 14 \times 14} \simeq 5.24$ and -5.24 as predicted by the analysis. They always appear in pairs, due to the bipartite nature of the graph involved. The rest of the spurious eigenvalues are constrained within a circle of radius $(14 \times 14)^{1/4} \simeq 3.74$ as predicted. The figure on the right is when we are below the spectral barrier, since the eigenvalue corresponding to the signal is $\sqrt{0.3 \times (1.4/3) \times 4 \times 4} \simeq 1.5$ which is smaller than $(4 \times 4)^{1/4} = 2$. The relevant eigenvalue is buried under other spurious eigenvalues and does not show.

Parameter estimation algorithm (line 12 of Algorithm 4). Among other things, this spectral interpretation gives an estimator for the problem parameter ρ^2, to be used in the inner-loop of Algorithm 4. Consider the data matrix $\widetilde{A}$ defined below. Again, a simple analysis shows that $\mathbb{E}[\widetilde{A}|q,p]$ is a rank one matrix with $\|\mathbb{E}[\widetilde{A}|q,p]\| = \sqrt{\ell r \hat{\rho}^2 \hat{\sigma}^2}$. Since the spectral norm of the noise matrix $\|\widetilde{A} - \mathbb{E}[\widetilde{A}|q,p]\|$ is upper bounded by $C(\ell r)^{1/4}$ for some constant C, we have $\|\widetilde{A}\|/\sqrt{\ell r \sigma^2} = \sqrt{\hat{\rho}^2\hat{\sigma}^2/\sigma^2} + O((\ell r \hat{\rho}^4 \hat{\sigma}^4)^{-1/4})$. We know ℓ and r, and assuming we know σ, this provides a natural estimator for $\hat{\rho}^2$. Note that $\hat{\sigma}^2 = \sigma^2 + O(\log(n)/\sqrt{n})$ with high probability. The performance of this estimator is empirically evaluated, as we use this in all our numerical simulations to implement our adaptive scheme in Algorithm 4.

Algorithm 6 Parameter Estimation Algorithm

```
Require: assignment graph adjacency matrix $E \in \{0,1\}^{|M|\times n}$, binary responses from the crowd $\{A_{ij}\}_{(i,j)\in E}$, task degree $\ell$, worker degree $r$, worker collective quality parameter $\sigma^2$
Ensure: estimate $\rho^2$ of $(1/|M|)\sum_{i\in M}\lambda_i$
1: Construct matrix $\widetilde{A} \in \{0,\pm 1\}^{|M|\times n}$ such that
       $$\widetilde{A}_{i,j} = \begin{cases} A_{i,j} & \text{, if } (i,j)\in E \\ 0 & \text{, otherwise} \end{cases}$$
   for all $i \in [|M|]$, $j \in [n]$.
2: Set $\sigma_1(\widetilde{A})$ to be the top singular value of matrix $\widetilde{A}$
3: $\rho^2 \leftarrow \big(\sigma_1(\widetilde{A})/\sqrt{\ell r \sigma^2}\big)^2$
```

4.5 Alternative inference algorithm for the generalized DS model

Our main contribution is a general framework for adaptive crowdsourcing: starting with a small-budget, classify tasks with high-confidence, and then gradually increase the budget per round, classifying remaining tasks. If we have other inference algorithms with which we can get reliable confidence levels in the estimated task labels, we can replace Algorithm 5. In this section, we propose such a potential candidate and discuss the computational challenges involved.

Under the original DS model, various standard methods such as Expectation Maximization (EM) and Belief Propagation (BP) provide efficient inference algorithms that also work well in practice [134]. However, under the generalized DS model, both approaches fail to give computationally tractable inference algorithms. The reason is that both tasks and workers are parametrized by continuous variables, making EM and BP computationally infeasible. In this section, we propose an alternative inference algorithm based on alternating minimization. This approach enjoys the benefits of EM and BP, such as seamlessly extending to k-ary alphabet labels, while remaining computationally manageable. Figure 4.4 illustrates how this alternating minimization performs at least as well as the iterative algorithm (Algorithm 5), and improves significantly when the budget is critically small, i.e. only a few workers are assigned to each task.

We propose to maximize the posterior distribution,

$$\mathbb{P}[q,p|A] \;\; \propto \;\; \prod_{i\in[m]} \mathbb{P}_{\mathcal{Q}}[q_i] \prod_{j\in[n]} \Bigg(\mathbb{P}_{\mathcal{P}}[p_j] \prod_{i'\in W_j} \mathbb{P}[A_{i'j}|p_{i'},q_j] \Bigg) \,. \tag{4.39}$$

Although this function is not concave, maximizing over q (or p) fixing p (or q) is simple due to the bipartite nature of the graph. Define a function $g : \{\pm 1\} \times [0,1] \times [0,1] \to [-\infty, 0]$ such that

$$g(A_{ij}, q_i, p_j) = \begin{cases} \log(q_i p_j + \bar{q}_i \bar{p}_j) & \text{if } A_{ij} = 1 \\ \log(\bar{q}_i p_j + q_j \bar{p}_i) & \text{if } A_{ij} = -1 \end{cases} \tag{4.40}$$

The logarithm of the joint posterior distribution (4.39) is

$$\mathcal{L}(q,p|A) \;\; = \;\; \sum_{i\in[m]} \sum_{j\in W_i} g(A_{ij}, q_i, p_j) + \sum_{i\in[m]} \log(\mathbb{P}_{\mathcal{Q}}[q_i]) + \sum_{j\in[n]} \log(\mathbb{P}_{\mathcal{P}}[p_j]) \,. \tag{4.41}$$

With properly chosen prior distributions $\mathcal{Q}$ and $\mathcal{P}$, in particular Beta priors, it is easy to see that the log likelihood is a concave function of p for fixed q. The same is true when fixing p and considering a function over q. Further, each coordinate p_j (and q_i) can be maximized separately. We start with $q_i = |W_i^+|/(|W_i^+| + |W_i^-|)$ and perform alternating minimization on (4.41) with respect to q and p iteratively until convergence, where $W_i^+ = \{j \in W_i : A_{ij} = 1\}$, $W_i^- = \{j \in W_i : A_{ij} = -1\}$, and W_i is the set of workers assigned to task i.

In Figure 4.4, we compare our algorithm with alternating minimization and majority voting on simulated data and real data. The first plot is generated under the same settings as the first plot of Figure 4.2 except that here we use $n = m = 300$ and $\sigma^2 = 0.2$. It shows that Algorithm 5 and alternating minimization performs almost the same after the spectral barrier, while the proposed Algorithm 5 fails below the spectral barrier as expected from the spectral analysis of Section 4.4. For the figure on the left, we choose $\sigma^2 = 0.2, \rho^2 = 1.4/3$. From the analyses in Section 4.4, we predict the spectral barrier to be at $\ell = 11$. In the second plot, we compare all the three algorithms on real data collected from Amazon Mechanical Turk in [107]. This dataset considers binary classification tasks for comparing closeness in human perception of colors; three colors are shown in each task and the worker is asked to indicate "whether the first color is more similar to the second color or the third color." This is asked on 50 of such color comparison tasks and 28 workers are recruited to complete all the tasks. We take the ground truth according to which color is closer to the first color in pairwise distances in the *Lab color* space. The second plot shows probability of error of the three algorithms when number of queries per task $\ell = \Gamma/m$ is varied. We generated responses for different values of Γ/m by uniformly sub-sampling. The alternating minimization and iterative algorithm perform similarly. However, for very small Γ, alternating minimization outperforms the iterative algorithm.

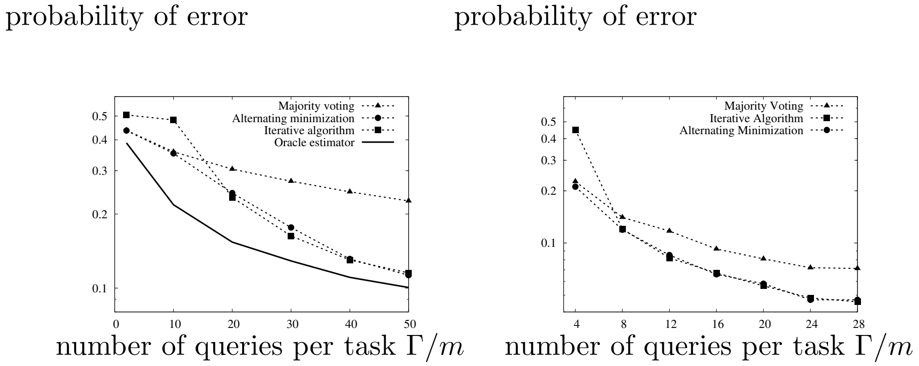


Figure 4.4: The iterative algorithm improves over majority voting and has similar performance as alternating minimization on both synthetic data (left) and real data from Amazon's Mechanical Turk (right).

4.6 Proofs

In this section, we provide the proofs of the main technical results.

4.6.1 Proof of Theorem 4.1

In this section, we first prove a slightly stronger result in Lemma 4.7 and prove Theorem 4.1 as a corollary. Lemma 4.7 is stronger as it is an *adaptive* lower bound that holds for *all* discrete prior distributions $\mathcal{Q}$. The lower bound in Equation (4.43) is adaptive in the sense that it automatically adjusts for any given $\mathcal{Q}$ as shown in its explicit dependence in $\delta_{\min}$ and λ. On the other hand, Theorem 4.1 only has to hold for *one* worst-case prior distribution $\mathcal{Q}$.

Let $\mathcal{Q}_{\lambda,\delta_{\min}}$ be the set of all discrete prior distributions on q_i such that the collective task difficulty is λ, and the minimum probability mass in it is $\delta_{\min}$, i.e.

$$\mathcal{Q}_{\lambda,\delta_{\min}} \equiv \left\{ \text{discrete } \mathcal{Q} \,\middle|\, \left(\mathbb{E}_{\mathcal{Q}} \left[\frac{1}{(2q_a-1)^2} \right] \right)^{-1} = \lambda \min_{\lambda_a \in \text{supp}(\widetilde{\mathcal{Q}})} \widetilde{\mathcal{Q}}(\lambda_a) = \delta_{\min} \right\}, \tag{4.42}$$

where $\widetilde{\mathcal{Q}}$ is the induced distribution on λ_a's. We let $\mathscr{T}_\Gamma$ be the set of all task assignment schemes that make at most Γ queries to the crowd in expectation. We prove a lower bound on the standard minimax error rate: the error that is achieved by the best inference algorithm $\hat{t}$ using the best adaptive task assignment scheme $\tau \in \mathscr{T}_\Gamma$ under a worst-case worker parameter distribution $\mathcal{P} \in \mathcal{P}_{\sigma^2}$ and any task parameter distribution $\mathcal{Q} \in \mathcal{Q}_{\lambda,\delta_{\min}}$. A proof of this lemma is provided in the following section.

Lemma 4.7. *For $\sigma^2 < 1$, for any discrete $\mathcal{Q} \in \mathcal{Q}_{\lambda,\delta_{\min}}$, there exists a positive constant C' such that the average probability of error is lower bounded by*

$$\min_{\tau \in \mathscr{T}_\Gamma, \hat{t}} \quad \max_{\mathcal{P} \in \mathcal{P}_{\sigma^2}} \quad \frac{1}{m} \sum_{i=1}^{m} \mathbb{P}[t_i \neq \hat{t}_i] \quad \geq \quad \delta_{\min} e^{-C'\left(\frac{\Gamma\lambda\sigma^2}{m}+1\right)}, \tag{4.43}$$

where m is the number of tasks, Γ is the expected budget allowed in $\mathscr{T}_\Gamma$, λ is the collective difficulty of the tasks from a prior distribution $\mathcal{Q} \in \mathcal{Q}_{\lambda,\delta_{\min}}$ defined in (4.42)*, σ^2 is the collective reliability of the crowd from a prior distribution $\mathcal{P}$ defined in* (4.3)*, and $\delta_{\min}$ is defined in* (4.42)*.*

Theorem 4.1 follows immediately from Lemma 4.7 as it considers the worst-case $\mathcal{Q} \in \mathcal{Q}_\lambda$ whereas the Lemma is proved for any discrete $\mathcal{Q} \in \mathcal{Q}_{\lambda,\delta_{\min}}$. For any given λ, there exists a discrete distribution $\mathcal{Q} \in \mathcal{Q}_{\lambda,\delta_{\min}}$, namely a distribution that is supported at two points $q = (1 \pm \sqrt{\lambda})/2$ with equal probability mass of $1/2$. Such a distribution has $\delta_{\min} = 1/2$ and therefore the Theorem 4.1 follows.

4.6.2 Proof of Lemma 4.7

Let $W_i \subseteq [n]$ denote the (random) set of workers assigned to task i in the end, when n (random) number of workers have provided their responses. For a task assignment scheme τ, we let

$$\ell_{i,q_i}^{(\tau)}(\mathcal{Q},\mathcal{P}) \;\equiv\; \mathbb{E}[\,|W_i||q_i\,]\;,$$

denote the conditional expectation of number of workers assigned to a task i conditioned on its quality q_i, where expectation is w.r.t. the randomness in the latent variables from $(\mathcal{Q},\mathcal{P})$ except q_i, the randomness in the task assignment scheme τ and the responses A. Let

$$\mathscr{T}_{\ell_{i,q_i}} \;\equiv\; \left\{\tau : \ell_{i,q_i}^{(\tau)} = \ell_{i,q_i}\right\}\;,$$

denote the set of all task assignment schemes that in expectation assign ℓ_{i,q_i} workers to the i-th task conditioned on its quality q_i. Further, let

$$\mathscr{T}_{\{\ell_{i,q_i}\}_{i=1}^m} \;\equiv\; \left\{\tau : \left(\{\ell_{i,q_i}\}_{i=1}^m\right)^{(\tau)} = \{\ell_{i,q_i}\}_{i=1}^m\right\}\;,$$

denote the set of all task assignment schemes that in expectation assign ℓ_{i,q_i} workers to each task $i \in [m]$ conditioned on its quality q_i. The fundamental lower bound crucially relies on the following technical lemma, whose proof is provided in the following section.

Lemma 4.8. *For any $\sigma^2 < 1$, there exists a positive constant C' and a prior distribution $\mathcal{P}^* \in \mathcal{P}_{\sigma^2}$ such that for each task $i \in [m]$ with task difficulty q_i*

$$\min_{\tau \in \mathscr{T}_{\ell_{i,q_i}},\hat{t}} \;\mathbb{P}[t_i \neq \hat{t}_i|q_i] \;\;\geq\;\; e^{-C'(\lambda_i\sigma^2\,\ell_{i,q_i}+1)}\,,$$

where $\lambda_i = (2q_i-1)^2$. Moreover, the prior distribution $\mathcal{P}^*$ does not depend upon q_i, therefore the bound holds simultaneously for all tasks $i \in [m]$ with varying q_i's.

This proves a lower bound on *per task* probability of error that decays exponentially with exponent scaling as $\lambda_i\sigma^2\ell_{i,q_i}$. The easier the task ($\lambda_i = (2q_i-1)^2$ large), the more reliable the workers are (σ^2 large), and the more workers assigned to that task (ℓ_{i,q_i} large), the smaller the achievable error.

To get a lower bound on the minimax *average* probability of error, where the error probability is over the randomness in the latent variables from $(\mathcal{Q}, \mathcal{P})$ and the randomness in the task assignment scheme τ and the responses A, we have,

$$
\begin{aligned}
&\min_{\tau\in\mathscr{T}_\Gamma,\hat{t}} \; \max_{\mathcal{P}\in\mathcal{P}_{\sigma^2}} \frac{1}{m}\sum_{i=1}^{m} \mathbb{P}_{(\mathcal{P},\mathcal{Q},\tau)}[t_i \neq \hat{t}_i] \\
&= \min_{\tau\in\mathscr{T}_\Gamma,\hat{t}} \; \max_{\mathcal{P}\in\mathcal{P}_{\sigma^2}} \frac{1}{m}\sum_{i=1}^{m} \mathbb{E}_{q_i\sim\mathcal{Q}}\big[\, \mathbb{P}_{(\mathcal{P},\mathcal{Q},\tau)}[t_i \neq \hat{t}_i | q_i] \,\big] \\
&= \min_{\ell_{i,q_i}:\sum_{i\in[m]}\mathbb{E}_{\mathcal{Q}}[\ell_{i,q_i}]\le\Gamma} \; \min_{\tau\in\mathscr{T}_{\{\ell_{i,q_i}\}_{i=1}^m},\hat{t}} \; \max_{\mathcal{P}\in\mathcal{P}_{\sigma^2}} \frac{1}{m}\sum_{i=1}^{m} \mathbb{E}_{q_i\sim\mathcal{Q}}\big[\, \mathbb{P}_{(\mathcal{P},\mathcal{Q},\tau)}[t_i \neq \hat{t}_i | q_i] \,\big] \\
&\ge \min_{\ell_{i,q_i}:\sum_{i\in[m]}\mathbb{E}_{\mathcal{Q}}[\ell_{i,q_i}]=\Gamma} \; \min_{\tau\in\mathscr{T}_{\{\ell_{i,q_i}\}_{i=1}^m},\hat{t}} \; \frac{1}{m}\sum_{i=1}^{m} \mathbb{E}_{q_i\sim\mathcal{Q}}\big[\, \mathbb{P}_{(\mathcal{P}^*,\mathcal{Q},\tau)}[t_i \neq \hat{t}_i | q_i] \,\big] && (4.44) \\
&\ge \min_{\ell_{i,q_i}:\sum_{i\in[m]}\mathbb{E}_{\mathcal{Q}}[\ell_{i,q_i}]=\Gamma} \; \frac{1}{m}\sum_{i=1}^{m} \left\{ \mathbb{E}_{q_i\sim\mathcal{Q}}\left[\min_{\tau\in\mathscr{T}_{\ell_{i,q_i}},\hat{t}} \; \mathbb{P}_{(\mathcal{P}^*,\mathcal{Q},\tau)}[t_i \neq \hat{t}_i | q_i] \right]\right\} && (4.45) \\
&\ge \min_{\ell_{i,q_i}:\sum_{i\in[m]}\mathbb{E}_{\mathcal{Q}}[\ell_{i,q_i}]=\Gamma} \; \frac{1}{m}\sum_{i=1}^{m} \mathbb{E}_{q_i\sim\mathcal{Q}}\left[\, e^{-C'} e^{-C'\lambda_i\sigma^2\ell_{i,q_i}} \,\right] && (4.46) \\
&= \min_{\ell_a:\sum_{a\in[T]}\delta_a\ell_a=\Gamma/m} \; \sum_{a=1}^{T} \delta_a e^{-C'} e^{-C'\lambda_a\sigma^2\ell_a} && (4.47) \\
&= e^{-C'} e^{-C'\frac{\Gamma\lambda\sigma^2}{m}} \left(\sum_{a=1}^{T} \delta_a e^{-\lambda\sum_{a'\neq a}(\delta_{a'}/\lambda_{a'})\log(\lambda_a/\lambda_{a'})} \right) && (4.48) \\
&\ge e^{-C'} \delta_{\min} e^{-C'\frac{\Gamma\lambda\sigma^2}{m}} \, .
\end{aligned}
$$

(4.44) follows from the fact that fixing a prior $\mathcal{P}^*$ provides a lower bound. (4.45) follows from the fact that exchanging min and sum (and also expectation which is essentially a weighted sum) provides a lower bound. (4.46) uses Lemma 4.8. In (4.47), we used the assumption that $\mathcal{Q}$ is a discrete distribution supported on $\{q_a\}_{a\in[T]}$ points with probability mass δ_a's. ℓ_a is expected number of workers assigned to each task with difficulty δ_a and $\lambda_a = (2q_a - 1)^2$.

To achieve (4.48), we assume that the task assignment scheme has access to an oracle that reveals difficulty of each task q_a. Therefore, solving the optimization problem in (4.47) we get (4.48). The optimal choice of ℓ_a is,

$$\ell_a \quad = \quad \frac{\lambda}{\lambda_a}\frac{\Gamma}{m} \;+\; \frac{\lambda}{\lambda_a C'\sigma^2}\Bigg(\sum_{a'\neq a}\frac{\delta_{a'}}{\lambda_{a'}}\log\Big(\frac{\lambda_a}{\lambda_{a'}}\Big)\Bigg)\,, \tag{4.49}$$

where $\lambda = \left(\mathbb{E}_{\mathcal{Q}}\left[\frac{1}{(2q_i-1)^2}\right]\right)^{-1} = \left(\sum_a \delta_a/\lambda_a\right)^{-1}$. The summand in (4.48) does not depend upon the budget Γ/m, and it is lower bounded by $\delta_{\min} > 0$. This follows from the fact that in the summand the term corresponding to a such that $\lambda_a = \lambda_{\min}$ is lower bounded by $\delta_{\min}$. Ignoring the second term in the above equation, which does not depend on Γ, in our adaptive algorithm, we aim to assign $\ell_a = \frac{\lambda}{\lambda_a}\frac{\Gamma}{m}$ workers to tasks of difficulty λ_a.

4.6.3 Proof of Lemma 4.8

We will show that there exists a family of worker reliability distributions $\mathcal{P}^* \in \mathcal{P}_{\sigma^2}$ such that for any adaptive task assignment scheme that assigns $\mathbb{E}[|W_i||q_i]$ workers in expectation to a task i conditioned on its difficulty q_i, the conditional probability of error of task i conditioned on q_i is lower bounded by $\exp\left(-C'\lambda_i\sigma^2\mathbb{E}[|W_i||q_i]\right)$. We define the following family of distributions according to the spammer-hammer model. Define

$$p_j = \begin{cases} 1/2, & \text{w.p. } \; 1-\sigma^2, \\ 1, & \text{w.p. } \; \sigma^2\,, \end{cases} \quad ,$$

such that $E[(2p_j-1)^2] = \sigma^2$. Let $\mathbb{E}[W_i|q_i]$ denote the expected number of workers conditioned on the task difficulty q_i, that the adaptive task assignment scheme assigns to the task i. We consider a labeling algorithm that has access to an oracle that knows reliability of every worker (all the p_j's). Focusing on a single task i, since we know who the spammers are and spammers give no information about the task, we only need the responses from the hammers in order to make an optimal estimate. In particular, the optimal estimate would be the majority vote of the hammers.

Let $\mathscr{E}_i$ denote the conditional error probability of the optimal estimate conditioned on the realizations of the answers $\{A_{ij}\}_{j\in W_i}$ and the worker reliability $\{p_j\}_{j\in W_i}$. We have $\mathbb{E}[\mathscr{E}_i|q_i] \equiv \mathbb{P}[t_i \neq \hat{t}_i|q_i]$. The following lemma gives a lower bound on the error that depends only on the number of hammer workers, which we denote by ℓ_i, a random variable.

Lemma 4.9. *For any $C < 1$, there exists a positive constant C' such that when $(2q_i - 1)^2 < C$, the conditional error achieved by majority vote of ℓ_i hammer workers is at least*

$$\mathbb{E}[\mathscr{E}_i|q_i, \ell_i] \;\; \geq \;\; e^{-C'\left(\ell_i(2q_i-1)^2+1\right)} .$$

The lemma follows immediately from Lemma 2.6 in [107] using the fact that the label provided by each hammer worker is an i.i.d. random variable that takes value 1 with probability q_i and -1 otherwise, conditioned on the task difficulty q_i. Therefore, wlog, assuming the ground truth label $t_i = \text{sign}(2q_i - 1) = 1$, we have, $\mathbb{E}[\mathscr{E}_i|q_i, \ell_i] = \mathbb{P}[x < 0]$ where $x \sim 2\,\text{Binom}(\ell_i, q_i) - \ell_i$. By convexity and Jensen's inequality, it follows that

$$\mathbb{E}[\mathscr{E}_i|q_i] \;\; \geq \;\; e^{-C'\left(\mathbb{E}[\ell_i|q_i](2q_i-1)^2+1\right)} .$$

When we recruit $|W_i|$ workers, using Doob's Optional-Stopping Theorem [210, 10.10], conditional expectation of number of hammer workers is

$$\mathbb{E}[\ell_i|q_i] = \sigma^2\mathbb{E}[|W_i|\,|q_i] . \tag{4.50}$$

Combining the above two equations, we get the desired result

$$\mathbb{P}[t_i \neq \hat{t}_i|q_i] \;\; = \;\; \mathbb{E}[\mathscr{E}_i|q_i] \;\;\; \geq \;\; e^{-C'\left(\sigma^2(2q_i-1)^2\mathbb{E}[|W_i|\,|q_i]+1\right)} .$$

To verify Equation 4.50: Define $X_{i,k}$ for $k \in [|W_i|]$ to be a Bernoulli random variable, for a fixed $i \in [m]$ and fixed task difficulty q_i. Let $X_{i,k}$ take value one when the k-th recruited worker for task i is reliable and zero otherwise. Observe that the number of reliable workers is $\ell_i = \sum_{k=1}^{|W_i|} X_{i,k}$. From the spammer-hammer model that we have considered, $\mathbb{E}[X_{i,k} - \sigma^2] = 0$. Define $Z_{i,k} \equiv \sum_{k'=1}^{k}(X_{i,k'} - \sigma^2)$ for $k \in [|W_i|]$. Since $\{(X_{i,k} - \sigma^2)\}_{k\in[|W_i|]}$ are mean zero i.i.d. random variables, $\{Z_{i,k}\}_{k\in[|W_i|]}$ is a martingale with respect to the filtration $\mathcal{F}_{i,k} = \sigma(X_{i,1}, X_{i,2}, \cdots, X_{i,k})$. Further, it is easy to check that the random variable $|W_i|$ for a fixed q_i is a stopping time with respect to the same filtration $\mathcal{F}_{i,k}$ and is almost surely bounded assuming the budget is finite. Therefore using Doobs Optional-Stopping Theorem [210, 10.10], we have $\mathbb{E}[Z_{i,|W_i|}] = \mathbb{E}[Z_{i,1}] = 0$. That is we have, $\mathbb{E}[X_{i,1} + X_{i,2} + \cdots + X_{i,|W_i|}] =$

$\sigma^2\mathbb{E}[|W_i|]$. Since this is true for any fixed task difficulty q_i, we get Equation (4.50).

4.6.4 Proof of Theorem 4.4

First we show that the messages returned by Algorithm 5 are normally distributed and identify their conditional means and conditional variances in the following lemma. Assume in a sub-round (t, u), $t \in [T], u \in [s_t]$, the number of tasks remaining unclassified are $m_{t,u}$ and the task assignment is performed according to an (ℓ_t, r_t)-regular random graph. To simplify the notation, let $\hat{\ell}_t \equiv \ell_t - 1$, $\hat{r_t} \equiv r_t - 1$, and recall $\mu = \mathbb{E}[2p_j - 1]$, $\sigma^2 = \mathbb{E}_{\mathcal{P}}[(2p_j - 1)^2]$. Note that μ, σ^2 remain same in each round. Let $\rho_{t,u}^2 = (1/|M|)\sum_{i\in[M]}\lambda_i$ be the exact value of average task difficulty of the tasks present in the (t, u) sub-round. When ℓ_t and r_t are increasing with the problem size, the messages converge to a Gaussian distribution due to the central limit theorem. We provide a proof of this lemma in Section 4.6.5.

Lemma 4.10. *Suppose for $\ell_t = \Theta(\log m_{t,u})$ and $r_t = \Theta(\log m_{t,u})$, tasks are assigned according to (ℓ_t, r_t)-regular random graphs. In the limit $m_{t,u} \to \infty$, if $\mu > 0$, then after $k = \Theta(\sqrt{\log m_{t,u}})$ number of iterations in Algorithm 5, the conditional mean $\mu_q^{(k)}$ and the conditional variance $\big(\rho_q^{(k)}\big)^2$ conditioned on the task difficulty q of the message x_i corresponding to the task i returned by the Algorithm 5 are*

$$\begin{aligned}
&\mu_q^{(k)} = (2q-1)\mu\ell_t(\hat{\ell}_t\hat{r}_t\rho_{t,u}^2\sigma^2)^{(k-1)}\ ,\\
&\big(\rho_q^{(k)}\big)^2\\
&= \mu^2\ell_t(\hat{\ell}_t\hat{r}_t\rho_{t,u}^2\sigma^2)^{2(k-1)}\Bigg(\rho_{t,u}^2 - (2q-1)^2\\
&+\frac{\rho_{t,u}^2\hat{\ell}_t(1-\rho_{t,u}^2\sigma^2)(1+\hat{r}_t\rho_{t,u}^2\sigma^2)\big(1-(\hat{\ell}_t\hat{r}_t\rho_{t,u}^4\sigma^4)^{-(k-1)}\big)}{\hat{\ell}_t\hat{r}_t\rho_{t,u}^4\sigma^4 - 1}\Bigg)\\
&+\ell_t(2-\mu^2\rho_{t,u}^2)(\hat{\ell}_t\hat{r}_t)^{k-1}\,.
\end{aligned}\tag{4.51}$$

We will show in (4.56) that the probability of misclassification for any task in sub-round (t, u) in Algorithm 4 is upper bounded by $e^{-(C_\delta/4)(\Gamma/m)\lambda\sigma^2}$. Since, there are at most $C_1 = s_{\max}T \le \log_2(2\delta_{\max}/\delta_{\min})\log_2(2\lambda_{\max}/\lambda_{\min})$ rounds, using union bound we get the desired probability of error. In (4.60), we show that the expected total number of worker assignments across all rounds is at most Γ.

Let's consider any task $i \in [m]$ having difficulty λ_i. Without loss of generality assume that $t_i = 1$ that is $q_i > 1/2$. Let us assume that the task i gets classified in the (t, u) sub-round, $t \in [T], u \in [s_t]$. That is the number of workers assigned to the task i when it gets classified is $\ell_t = C_\delta(\Gamma/m)(\widehat{\lambda}/\lambda_t)$ and the threshold $\mathcal{X}_{t,u}$ set in that round for classification is $\mathcal{X}_{t,u} = \sqrt{\lambda_t}\mu\ell_t\big((\ell_t - 1)(r_t - 1)\rho^2_{t,u}\sigma^2\big)^{k_t-1}$. From Lemma 4.10 the message x_i returned by Algorithm 5 is Gaussian with conditional mean and conditional variance as given in (4.51). Therefore in the limit of m, the probability of error in task i is

$$\begin{align}
\lim_{m\to\infty} \mathbb{P}\big[\hat{t}_i \neq t_i | q_i\big] &= \lim_{m\to\infty} \mathbb{P}\big[x_i < -\mathcal{X}_{t,u} | q_i\big] \notag \\
&= \lim_{m\to\infty} Q\Big(\frac{\mu^{(k)}_{q_i} + \mathcal{X}_{t,u}}{\rho^{(k)}_{q_i}}\Big) \tag{4.52} \\
&\leq \lim_{m\to\infty} \exp\Big(\frac{-(\mu^{(k)}_{q_i} + \mathcal{X}_{t,u})^2}{2(\rho^{(k)}_{q_i})^2}\Big) \tag{4.53} \\
&= \exp\Big(\frac{-((2q_i - 1) + \sqrt{\lambda_t})^2\ell_t\sigma^2}{2(1 - (2q_i - 1)^2\sigma^2)}\Big) \tag{4.54} \\
&\leq \exp\Big(\frac{-\lambda_t\ell_t\sigma^2}{2}\Big) \notag \\
&= \exp\Big(\frac{-C_\delta(\Gamma/m)\widehat{\lambda}\sigma^2}{2}\Big) \tag{4.55} \\
&\leq \exp\Big(\frac{-C_\delta(\Gamma/m)\lambda\sigma^2}{4}\Big), \tag{4.56}
\end{align}$$

where $Q(\cdot)$ in (4.52) is the tail probability of a standard Gaussian distribution, and (4.53) uses the Chernoff bound. (4.54) follows from substituting conditional mean and conditional variance from Equation (4.51), and using $\ell_t = \Theta(\log m_{t,u})$, $k = \Theta(\sqrt{\log m_{t,u}})$ where m grows to infinity. (4.55) uses $\ell_t = C_\delta(\Gamma/m)(\widehat{\lambda}/\lambda_t)$, our choice of ℓ_t in Algorithm 4 line 4. (4.56) uses the fact that for the quantized distribution $\{\lambda_a, \delta_a\}_{a\in[T]}$, $\widehat{\lambda} = \big(\sum_{a\in[T]}(\delta_a/\lambda_a)\big)^{-1} \geq \lambda/2$. We have established that our approach guarantees the desired level of accuracy. We are left to show that we use at most Γ assignments in expectation.

We upper bound the expected total number of workers used for tasks of quantized difficulty level λ_a's for each $1 \leq a \leq T$. Recall that our adaptive algorithm runs in T rounds indexed by t, where each round t further runs s_t sub-rounds. The total expected number of workers assigned to δ_a fraction of tasks of quantized difficulty λ_a in $t = 1$ to $t = a-1$ rounds is upper bounded by $m\delta_a \sum_{t=1}^{a-1} s_t\ell_t$. The upper bound assumes the worst-case (in terms of the budget) that these tasks do not get classified in any of these rounds as the threshold $\mathcal{X}$ set in these rounds is more than absolute value of the conditional mean message x of these tasks.

Next, in $s_{t=a}$ sub-rounds the threshold $\mathcal{X}$ is set less than or equal to the absolute value of the conditional mean message x of these tasks, i.e. $\mathcal{X} \le |\mu^{(k)}_{q_a}|$ for $(2q_a-1)^2 = \lambda_a$. Therefore, in each of these s_a sub-rounds, probability of classification of these tasks is at least $1/2$. That is the expected total number of workers assigned to these tasks in s_a sub-rounds is upper bounded by $2m\delta_a\ell_a$. Further, s_a is chosen such that the fraction of these tasks remaining un-classified at the end of s_a sub-rounds is at most same as the fraction of the tasks having difficulty λ_{a+1}. That is to get the upper bound, we can assume that the fraction of λ_{a+1} difficulty tasks at the start of s_{a+1} sub-rounds is $2\delta_{a+1}$, and the fraction of λ_a difficulty tasks at the start of s_{a+1} sub-rounds is zero. Further, recall that we have set $s_T = 1$ as in this round our threshold $\mathcal{X}$ is equal to zero. Therefore, we have the following upper bound on the expected total number of worker assignments.

$$\begin{aligned}
\sum_{i=1}^{m} \mathbb{E}[|W_i|] &\le 2m\delta_1\ell_1 + \sum_{a=2}^{T-1} 4m\delta_a\ell_a + 2m\delta_T\ell_T + \sum_{a=2}^{T}\Big(m\delta_a \sum_{b=1}^{a-1} s_b\ell_b\Big) \\
&\le \sum_{a=1}^{T} 4m\delta_a\ell_a + s_{\max}\sum_{a=1}^{T} m\delta_a\ell_a & (4.57)\\
&\le (4 + \lceil \log(2\delta_{\max}/\delta_{\min})\rceil)\sum_{a=1}^{T} m\delta_a\ell_a & (4.58)\\
&\le (4 + \lceil \log(2\delta_{\max}/\delta_{\min})\rceil)\Gamma C_\delta & (4.59)\\
&= \Gamma\,, & (4.60)
\end{aligned}$$

Equation (4.57) uses the fact that $\ell_t = (C_\delta(\Gamma/m)(\widehat{\lambda}/\lambda_t)$ where λ_t's are separated apart by at least a ratio of 2 (recall the quantized distribution), therefore $\sum_{t=1}^{a-1}\ell_t \le \ell_a$. Equation (4.58) follows from the choice of s_t's in the algorithm. Equation (4.59) follows from using $\ell_t = (C_\delta(\Gamma/m)(\widehat{\lambda}/\lambda_t)$ and $\lambda = (\sum_{a\in[T]}(\delta_a/\lambda_a))^{-1}$, and Equation (4.60) uses $C_\delta = (4 + \lceil\log(2\delta_{\max}/\delta_{\min})\rceil)^{-1}$.

4.6.5 Proof of Lemma 4.10

We omit subscripts t and (t, u) from all the quantities for simplicity of notations. Also, we use notation ℓ, average budget per task, $\ell = \Gamma/m$. We will prove it for a randomly chosen task $\mathbf{I}$, and all the analyses naturally holds for a specific i, when conditioned on q_i. Let n be the number of workers, that is $n = (mr)/\ell$. In our algorithm, we perform task assignment on a random bipartite graph $\mathbf{G}([m] \cup [n], E)$ constructed according to the configuration model. Let $\mathbf{G}_{i,k}$ denote a subgraph of $\mathbf{G}([m] \cup [n], E)$ that includes all the nodes that are within k distance from the the "root" i. If we run our inference algorithm for one run to estimate $\hat{t}_i$, we only use the responses provided by the workers who were assigned to task i. That is we are running inference algorithm only on the local neighborhood graph $\mathbf{G}_{i,1}$. Similarly, when we run our algorithm for k iterations to estimate $\hat{t}_i$, we perform inference only on the local subgraph $\mathbf{G}_{i,2k-1}$. Since we update both task and worker messages at each iteration, the local subgraph grows by distance two at each iteration. We use a result from [107] to show that the local neighborhood of a randomly chosen task node $\mathbf{I}$ is a tree with high probability. Therefore, assuming that the graph is locally tree like with high probability, we can apply a technique known as *density evolution* to estimate the conditional mean and conditional variance. The next lemma shows that the local subgraph converges to a tree in probability, in the limit $m \to \infty$ for the specified choice of ℓ, r and k.

Lemma 4.11 (Lemma 5 from [107]). *For a random (ℓ, r)-regular bipartite graph generated according to the configuration model,*

$$\mathbb{P}\big[\mathbf{G}_{\mathbf{I},2k-1} \text{ is not a tree}\big] \leq \big((\ell-1)(r-1)\big)^{2k-2}\frac{3\ell r}{m}. \tag{4.61}$$

Density Evolution. Let $\{x_{i\to j}^{(k)}\}_{(i,j)\in E}$ and $\{y_{j\to i}^{(k)}\}_{(i,j)\in E}$ denote the messages at the k-th iteration of the algorithm. For an edge (i, j) chosen uniformly at random, let $\mathbf{x}_q^{(k)}$ denote the random variable corresponding to the message $x_{i\to j}^{(k)}$ conditioned on the i-th task's difficulty being q. Similarly, let $\mathbf{y}_p^{(k)}$ denote the random variable corresponding to the message $y_{j\to i}^{(k)}$ conditioned on the j-th worker's quality being p.

At the first iteration, the task messages are updated according to $x_{i\to j}^{(1)} = \sum_{j'\in\partial i\setminus j} A_{ij'} y_{j'\to i}^{(0)}$. Since we initialize the worker messages $\{y_{j\to i}^{(0)}\}_{(i,j)\in E}$ with independent Gaussian random variables with mean and variance both one, if we know the distribution of $A_{ij'}$'s, then we have the distribution of $x^{(1)}_{i\to j}$. Since, we are assuming that the local subgraph is tree-like, all $x^{(1)}_{i\to j}$ for $i \in \mathbf{G}_{\mathbf{I},2k-1}$ for any randomly chosen node $\mathbf{I}$ are independent. Further, because of the symmetry in the construction of the random graph $\mathbf{G}$ all messages $x^{(1)}_{i\to j}$'s are identically distributed. Precisely, $x^{(1)}_{i\to j}$ are distributed according to $\mathbf{x}^{(1)}_q$ defined in Equation (7.5). In the following, we recursively define $\mathbf{x}^{(k)}_q$ and $\mathbf{y}^{(k)}_p$ in Equations (7.5) and (4.64).

For brevity, here and after, we drop the superscript k-iteration number whenever it is clear from the context. Let $\mathbf{x}_{q,a}$'s and $\mathbf{y}_{p,b}$'s be independent random variables distributed according to $\mathbf{x}_q$ and $\mathbf{y}_p$ respectively. We use a and b as indices for independent random variables with the same distribution. Also, let $\mathbf{z}_{p,q,a}$'s and $\mathbf{z}_{p,q,b}$'s be independent random variables distributed according to $\mathbf{z}_{p,q}$, where

$$\mathbf{z}_{p,q} = \begin{cases} +1 & \text{w.p.} \quad pq + (1-p)(1-q)\,, \\ -1 & \text{w.p.} \quad p(1-q) + (1-p)q\,. \end{cases} \tag{4.62}$$

This represents the response given by a worker conditioned on the task having difficulty q and the worker having ability p. Let $\mathcal{P}_1$ and $\mathcal{P}_2$ over $[0,1]$ be the distributions of the tasks' difficulty level and workers' quality respectively. Let $q \sim \mathcal{P}_1$ and $p \sim \mathcal{P}_2$. Then q_a's and p_b's are independent random variables distributed according to q and p respectively. Further, $\mathbf{z}_{p,q_a,a}$'s and $\mathbf{x}_{q_a,a}$'s are conditionally independent conditioned on q_a; and $\mathbf{z}_{p_b,q,b}$'s and $\mathbf{y}_{p_b,b}$'s are conditionally independent conditioned on p_b.

Let $\overset{d}{=}$ denote equality in distribution. Then for $k \in \{1, 2, \cdots\}$, the task messages (conditioned on the latent task difficulty level q) are distributed as the sum of $\ell - 1$ incoming messages that are i.i.d. according to $\mathbf{y}^{(k-1)}_p$ and weighted by i.i.d. responses:

$$\mathbf{x}^{(k)}_q \overset{d}{=} \sum_{b\in[\ell-1]} \mathbf{z}_{p_b,q,b}\mathbf{y}^{(k-1)}_{p_b,b}\,. \tag{4.63}$$

Similarly, the worker messages (conditioned on the latent worker quality p) are distributed as the sum of $r-1$ incoming messages that are i.i.d. according to $\mathbf{x}_q^{(k)}$ and weighted by the i.i.d. responses:

$$\mathbf{y}_p^{(k)} \overset{d}{=} \sum_{a \in [r-1]} \mathbf{z}_{p,q_a,a} \mathbf{x}_{q_a,a}^{(k)}. \tag{4.64}$$

For the decision variable $\mathbf{x}_{\mathbf{I}}^{(k)}$ on a task $\mathbf{I}$ chosen uniformly at random, we have

$$\hat{\mathbf{x}}_q^{(k)} \overset{d}{=} \sum_{a \in [\ell]} \mathbf{z}_{p_a,q,a} \mathbf{y}_{p_a,a}^{(k-1)}. \tag{4.65}$$

Mean and Variance Computation. Define $m_q^{(k)} \equiv \mathbb{E}[\mathbf{x}_q^{(k)}|q]$ and $\hat{m}_p^{(k)} \equiv \mathbb{E}[\mathbf{y}_p^{(k)}|p]$, $\nu_q^{(k)} \equiv \text{Var}(\mathbf{x}_q^{(k)}|q)$ and $\hat{\nu}_p^{(k)} \equiv \text{Var}(\mathbf{y}_p^{(k)}|p)$. Recall the notations $\mu \equiv \mathbb{E}[2p-1]$, $\rho^2 \equiv \mathbb{E}[(2q-1)^2]$, $\sigma^2 \equiv \mathbb{E}[(2p-1)^2]$, $\hat{\ell} = \ell - 1$, and $\hat{r} = r - 1$. Then from (7.5) and (4.64) and using $\mathbb{E}[\mathbf{z}_{p,q}] = (2p-1)(2q-1)$ we get the following:

$$m_q^{(k)} = \hat{\ell}(2q-1)\mathbb{E}_p\big[(2p-1)\hat{m}_p^{(k-1)}\big], \tag{4.66}$$

$$\hat{m}_p^{(k)} = \hat{r}(2p-1)\mathbb{E}_q\big[(2q-1)m_q^{(k)}\big], \tag{4.67}$$

$$\nu_q^{(k)} = \hat{\ell}\Big\{\mathbb{E}_p\big[\hat{\nu}_p^{(k-1)} + (\hat{m}_p^{(k-1)})^2\big] - (m_q^{(k)}/\hat{\ell})^2\Big\}, \tag{4.68}$$

$$\hat{\nu}_p^{(k)} = \hat{r}\Big\{\mathbb{E}_q\big[\nu_q^{(k)} + (m_q^{(k)})^2\big] - (\hat{m}_p^{(k)}/\hat{r})^2\Big\}. \tag{4.69}$$

Define $m^{(k)} \equiv \mathbb{E}_q[(2q-1)m_q^{(k)}]$ and $\nu^{(k)} \equiv \mathbb{E}_q[\nu_q^{(k)}]$. From (4.66) and (4.67), we have the following recursion on the first moment of the random variable $x_q^{(k)}$:

$$m_q^{(k)} = \hat{\ell}\hat{r}(2q-1)\sigma^2 m^{(k-1)}, m^{(k)} = \hat{\ell}\hat{r}\rho^2\sigma^2 m^{(k-1)}\,. \tag{4.70}$$

From (4.68) and (4.69), and using $\mathbb{E}_q[(m_q^{(k)})^2] = (m^{(k)})^2/\rho^2$ (from (4.70)), and $\mathbb{E}_p[(\hat{m}_p^{(k)})^2] = \hat{r}^2\sigma^2(m^{(k)})^2$ (from (4.67)) , we get the following recursion on the second moment:

$$\begin{aligned} \nu_q^{(k)} &= \hat{\ell}\hat{r}\nu^{(k-1)} + \hat{\ell}\hat{r}(m^{(k-1)})^2\big((1-\rho^2\sigma^2)(1+\hat{r}\rho^2\sigma^2) \\ &\quad +\hat{r}\rho^2(\sigma^2)^2(\rho^2-(2q-1)^2)\big)/\rho^2\,, & (4.71) \\ \nu^{(k)} &= \hat{\ell}\hat{r}\nu^{(k-1)} + \hat{\ell}\hat{r}(m^{(k-1)})^2(1-\rho^2\sigma^2)(1+\hat{r}\rho^2\sigma^2)/\rho^2. & (4.72) \end{aligned}$$

Since $\hat{m}_p^{(0)} = 1$ as per our assumption, we have $m_q^{(1)} = \hat{\ell}\mu(2q-1)$ and $m^{(1)} = \hat{\ell}\mu\rho^2$. Therefore from (4.70), we have $m^{(k)} = \hat{\ell}\mu\rho^2(\hat{\ell}\hat{r}\rho^2\sigma^2)^{k-1}$ and $m_q^{(k)} = \hat{\ell}\mu(2q-1)(\hat{\ell}\hat{r}\rho^2\sigma^2)^{k-1}$. Further, since $\hat{\nu}_p^{(0)} = 1$ as per our assumption, we have $\nu_q^{(1)} = \hat{\ell}(2-\mu^2(2q-1)^2)$ and $\nu^{(1)} = \hat{\ell}(2-\mu^2\rho^2)$. This implies that $\nu^{(k)} = a\nu^{(k-1)} + bc^{k-2}$, with $a = \hat{\ell}\hat{r}$, $b = \mu^2\rho^2\hat{\ell}^3\hat{r}(1-\rho^2\sigma^2)(1+\hat{r}\rho^2\sigma^2)$ and $c = (\hat{\ell}\hat{r}\rho^2\sigma^2)^2$. After some algebra, we have that $\nu^{(k)} = \nu^{(1)}a^{k-1} + bc^{k-2}\sum_{\ell=0}^{k-2}(a/c)^{\ell}$. For $\hat{\ell}\hat{r}(\rho^2\sigma^2)^2 > 1$, we have $a/c < 1$ and

$$\begin{aligned}\nu_q^{(k)} &= \hat{\ell}(2-\mu^2\rho^2)(\hat{\ell}\hat{r})^{k-1} + \mu^2\hat{\ell}(\hat{\ell}\hat{r}\rho^2\sigma^2)^{2k-2}(\rho^2-(2q-1)^2) \\ &\quad + \left(\frac{1-1/(\hat{\ell}\hat{r}(\rho^2\sigma^2)^2)^{k-1}}{\hat{\ell}\hat{r}\rho^4\sigma^4 - 1}\right)(1-\rho^2\sigma^2)(1+\hat{r}\rho^2\sigma^2)\mu^2\rho^2\hat{\ell}^2(\hat{\ell}\hat{r}\rho^2\sigma^2)^{2k-2}.\end{aligned} \tag{4.73}$$

By a similar analysis, mean and variance of the decision variable $\hat{\mathbf{x}}_q^{(k)}$ in (4.65) can also be computed. In particular, they are $\ell/\hat{\ell}$ times $m_q^{(k)}$ and $\nu_q^{(k)}$. Gaussianity of the messages follows due to Central limit theorem.

4.6.6 Proof of Theorem 4.5

The proof uses the results derived in the proof of Lemma 4.10.

Let $\hat{t}_i^{(k)}$ denote the resulting estimate of task i after running the iterative inference algorithm for k iterations. We want to compute the conditional probability of error of a task $\mathbf{I}$ selected uniformly at random in $[m]$, conditioned on its difficulty level, i.e.,

$$\mathbb{P}\big[t_{\mathbf{I}} \neq \hat{t}_{\mathbf{I}}^{(k)} \big| q_{\mathbf{I}}\big]\,.$$

In the following, we assume $q_{\mathbf{I}} \geq (1/2)$, i.e. the true label is $t_i = 1$. Analysis for $q_{\mathbf{I}} \leq (1/2)$ would be similar and result in the same bounds. Using the arguments given in Lemma 4.10, we have,

$$\mathbb{P}\big[t_{\mathbf{I}} \neq \hat{t}_{\mathbf{I}}^{(k)} \big| q_{\mathbf{I}}\big] \;\leq\; \mathbb{P}\big[t_{\mathbf{I}} \neq \hat{t}_{\mathbf{I}}^{(k)} \big| \mathbf{G}_{\mathbf{I},2k-1} \text{ is a tree, } q_{\mathbf{I}}\big] + \mathbb{P}\big[\mathbf{G}_{\mathbf{I},2k-1} \text{ is not a tree}\big]. \tag{4.74}$$

To provide an upper bound on the first term in (7.1), let $x_i^{(k)}$ denote the decision variable for task i after k iterations of the algorithm such that $\hat{t}_i^{(k)} =$

$\text{sign}(x_i^{(k)})$. Then as per our assumption that $t_i = 1$, we have,

$$\mathbb{P}\big[t_{\mathbf{I}} \neq \hat{t}_{\mathbf{I}}^{(k)} | \mathbf{G}_{\mathbf{I},2k-1} \text{is a tree}, q_{\mathbf{I}}\big] \;\; \leq \;\; \mathbb{P}\big[x_{\mathbf{I}}^{(k)} \leq 0 | \mathbf{G}_{\mathbf{I},2k-1} \text{is a tree}, q_{\mathbf{I}}\big]. \tag{4.75}$$

Next, we apply "density evolution" [152] and provide a sharp upper bound on the probability of the decision variable $x_{\mathbf{I}}^{(k)}$ being negative in a locally tree like graph given $q_{\mathbf{I}} \geq (1/2)$. The proof technique is similar to the one introduced in [107]. Precisely, we show,

$$\mathbb{P}\big[x_{\mathbf{I}}^{(k)} \leq 0 | \mathbf{G}_{\mathbf{I},2k-1} \text{ is a tree }, q_{\mathbf{I}}\big] = \mathbb{P}\big[\hat{\mathbf{x}}_q^{(k)} \leq 0\big]\,, \tag{4.76}$$

where $\hat{\mathbf{x}}_q^{(k)}$ is defined in Equations (7.5)-(4.65) using density evolution. We will prove in the following that when $\hat{\ell}\hat{r}(\rho^2\sigma^2)^2 > 1$ and $\hat{r}\rho^2 > 1$,

$$\mathbb{P}\big[\hat{\mathbf{x}}_q^{(k)} \leq 0\big] \;\; \leq \;\; e^{-\ell\sigma^2(2q_{\mathbf{I}}-1)^2/(2\tilde{\sigma}_k^2)}. \tag{4.77}$$

Theorem 4.5 follows by combining Equations (7.1),(7.2),(7.3) and (4.76).

we show that $\hat{\mathbf{x}}^{(k)}$ is sub-Gaussian with some appropriate parameter and then apply the Chernoff bound. A random variable $\mathbf{x}$ with mean μ is said to be *sub-Gaussian* with parameter σ if for all $\lambda \in \mathbb{R}$ the following bound holds for its moment generating function:

$$\mathbb{E}[e^{\lambda \mathbf{x}}] \;\; \leq \;\; e^{\mu\lambda + (1/2)\sigma^2\lambda^2}\,. \tag{4.78}$$

Define,

$$\tilde{\sigma}_k^2 \equiv 3\hat{\ell}^3\hat{r}\mu^2\rho^2(\hat{r}\rho^2\sigma^2+1)(\hat{\ell}\hat{r}\rho^2\sigma^2)^{2k-4}\big(\frac{1-1/(\hat{\ell}\hat{r}(\rho^2\sigma^2)^2)^{k-1}}{1-1/(\hat{\ell}\hat{r}\rho^2\sigma^2)}\big) + 2\hat{\ell}(\hat{\ell}\hat{r})^{k-1}\,, \tag{4.79}$$

$m_k \equiv \mu\hat{\ell}(\hat{\ell}\hat{r}\rho^2\sigma^2)^{k-1}$, and $m_{k,q} \equiv (2q-1)m_k$ for $k \in \mathbb{Z}$, where $q \sim \mathcal{P}_1$. We will show that, $\mathbf{x}_q^{(k)}$ is sub-Gaussian with mean $m_{k,q}$ and parameter $\tilde{\sigma}_k^2$ for $|\lambda| \leq 1/(2m_{k-1}\hat{r}\rho^2)$, i.e.,

$$\mathbb{E}[e^{\lambda \mathbf{x}_q^{(k)}} | q] \;\; \leq \;\; e^{m_{k,q}\lambda + (1/2)\tilde{\sigma}_k^2\lambda^2}\,. \tag{4.80}$$

Analyzing the Density. Notice that the parameter $\tilde{\sigma}_k^2$ does not depend upon the random variable q. By definition of $\hat{\mathbf{x}}_q^{(k)}$, (4.65), we have

$$\mathbb{E}[e^{\lambda\hat{\mathbf{x}}_q^{(k)}}|q] = \mathbb{E}[e^{\lambda\mathbf{x}_q^{(k)}}|q]^{(\ell/\hat{\ell})}\,.$$

Therefore, it follows that $\mathbb{E}[e^{\lambda\hat{\mathbf{x}}_q^{(k)}}|q] \leq e^{(\ell/\hat{\ell})m_{k,q}\lambda+(\ell/2\hat{\ell})\tilde{\sigma}_k^2\lambda^2}$. Using the Chernoff bound with $\lambda = -m_{k,q}/(\tilde{\sigma}_k^2)$, we have

$$\mathbb{P}[\hat{\mathbf{x}}_q^{(k)} \leq 0 \mid q] \;\leq\; \mathbb{E}[e^{\lambda\hat{\mathbf{x}}_q^{(k)}}|q] \;\leq\; e^{-\ell m_{k,q}^2/(2\hat{\ell}\tilde{\sigma}_k^2)}\,. \qquad (4.81)$$

Note that, with the assumption that $q \geq (1/2)$, $m_{k,q}$ is non-negative. Since

$$\frac{m_{k,q}m_{k-1,q}}{\tilde{\sigma}_k^2} \leq \frac{(2q-1)^2\mu^2\hat{\ell}^2(\hat{\ell}\hat{r}\rho^2\sigma^2)^{2k-3}}{3\mu^2\sigma^2(\rho^2)^2\hat{\ell}^3\hat{r}^2(\hat{\ell}\hat{r}\rho^2\sigma^2)^{2k-4}} = \frac{(2q-1)^2}{3\hat{r}\rho^2}\,,$$

it follows that $|\lambda| \leq 1/(2m_{k-1}\hat{r}\rho^2)$. The desired bound in (7.4) follows.

Now, we are left to prove Equation (4.80). From (7.5) and (4.64), we have the following recursive formula for the evolution of the moment generating functions of $\mathbf{x}_q$ and $\mathbf{y}_p$:

$$\mathbb{E}[e^{\lambda\mathbf{x}_q^{(k)}}|q] \;=\; \Big(\mathbb{E}_p\big[(pq+\bar{\mathbf{p}}\bar{\mathbf{q}})\mathbb{E}[e^{\lambda\mathbf{y}_p^{(k-1)}}|p] + (p\bar{\mathbf{q}}+\bar{\mathbf{p}}q)\mathbb{E}[e^{-\lambda\mathbf{y}_p^{(k-1)}}|p]\big]\Big)^{\hat{\ell}}\,, \qquad (4.82)$$

$$\mathbb{E}[e^{\lambda\mathbf{y}_p^{(k)}}|p] \;=\; \Big(\mathbb{E}_q\big[(pq+\bar{\mathbf{p}}\bar{\mathbf{q}})\mathbb{E}[e^{\lambda\mathbf{x}_q^{(k)}}|q] + (p\bar{\mathbf{q}}+\bar{\mathbf{p}}q)\mathbb{E}[e^{-\lambda\mathbf{x}_q^{(k)}}|q]\big]\Big)^{\hat{r}}\,, \qquad (4.83)$$

where $\bar{\mathbf{p}} = 1-p$ and $\bar{\mathbf{q}} = 1-q$. We apply induction to prove that the messages are sub-Gaussian. First, for $k=1$, we show that $\mathbf{x}_q^{(1)}$ is sub-Gaussian with mean $m_{1,q} = (2q-1)\mu\hat{\ell}$ and parameter $\tilde{\sigma}_1^2 = 2\hat{\ell}$. Since, $\mathbf{y}_p$ is initialized as a Gaussian random variable with mean and variance both one, we have $\mathbb{E}[e^{\lambda\mathbf{y}_p^{(0)}}] = e^{\lambda+(1/2)\lambda^2}$. Substituting this into Equation (4.82), we get for any λ,

$$\mathbb{E}[e^{\lambda\mathbf{x}_q^{(1)}}|q] \;=\; \Big(\big(\mathbb{E}[p]q+\mathbb{E}[\bar{\mathbf{p}}]\bar{\mathbf{q}}\big)e^{\lambda} + \big(\mathbb{E}[p]\bar{\mathbf{q}}+\mathbb{E}[\bar{\mathbf{p}}]q\big)e^{-\lambda}\Big)^{\hat{\ell}}e^{(1/2)\lambda^2\hat{\ell}} \qquad (4.84)$$

$$\leq\; e^{(2q-1)\mu\hat{\ell}\lambda+(1/2)(2\hat{\ell})\lambda^2}\,, \qquad (4.85)$$

where the inequality follows from the fact that $ae^z+(1-a)e^{-z} \leq e^{(2a-1)z+(1/2)z^2}$

for any $z \in \mathbb{R}$ and $a \in [0,1]$ (Lemma A.1.5 from [5]). Next, assuming

$$\mathbb{E}[e^{\lambda \mathbf{x}_q^{(k)}}|q] \leq e^{m_{k,q}\lambda + (1/2)\tilde{\sigma}_k^2\lambda^2}$$

for $|\lambda| \leq 1/(2m_{k-1}\hat{r}\rho^2)$, we show that

$$\mathbb{E}[e^{\lambda \mathbf{x}_q^{(k+1)}}|q] \leq e^{m_{k+1,q}\lambda + (1/2)\tilde{\sigma}_{k+1}^2\lambda^2}$$

for $|\lambda| \leq 1/(2m_k\hat{r}\rho^2)$, and compute appropriate $m_{k+1,q}$ and $\tilde{\sigma}_{k+1}^2$.

Substituting the bound $\mathbb{E}[e^{\lambda \mathbf{x}_q^{(k)}}|q] \leq e^{m_{k,q}\lambda + (1/2)\tilde{\sigma}_k^2\lambda^2}$ in (4.83), we have

$$\begin{aligned}
&\mathbb{E}[e^{\lambda \mathbf{y}_p^{(k)}}|p] \\
&\leq \Big(\mathbb{E}_q\big[(pq + \bar{\mathbf{p}}\bar{\mathbf{q}})e^{m_{k,q}\lambda} + (p\bar{\mathbf{q}} + \bar{\mathbf{p}}q)e^{-m_{k,q}\lambda}\big]\Big)^{\hat{r}} e^{(1/2)\tilde{\sigma}_k^2\lambda^2\hat{r}} \\
&\leq \Big(\mathbb{E}_q\big[e^{(2q-1)(2p-1)m_{k,q}\lambda + (1/2)(m_{k,q}\lambda)^2}\big]\Big)^{\hat{r}} e^{(1/2)\tilde{\sigma}_k^2\lambda^2\hat{r}} \qquad (4.86) \\
&= \Big(\mathbb{E}_q\big[e^{(2p-1)(2q-1)^2 m_k\lambda + (1/2)(2q-1)^2(m_k\lambda)^2}\big]\Big)^{\hat{r}} e^{0.5\tilde{\sigma}_k^2\lambda^2\hat{r}} \qquad (4.87)
\end{aligned}$$

where (4.86) uses the inequality $ae^z + (1-a)e^{-z} \leq e^{(2a-1)z+(1/2)z^2}$ and (4.87) follows from the definition of $m_{k,q} \equiv (2q-1)m_k$. To bound the term in (4.87), we use the following lemma.

Lemma 4.12. *For any random variable $s \in [0,1]$, $|z| \leq 1/2$ and $|t| < 1$, we have*

$$\mathbb{E}\big[e^{stz+(1/2)sz^2}\big] \leq \exp\big(\mathbb{E}[s]tz + (3/2)\mathbb{E}[s]z^2\big)\,. \qquad (4.88)$$

For $|\lambda| \leq 1/(2m_k\hat{r}\rho^2)$, using the assumption that $\hat{r}\rho^2 > 1$, we have $m_k\lambda \leq (1/2)$. Applying Lemma 4.12 on the term in (4.87), with $s = (2q-1)^2$, $z = m_k\lambda$ and $t = (2p-1)$, we get

$$\mathbb{E}[e^{\lambda \mathbf{y}_p^{(k)}}|p] \leq e^{\rho^2(2p-1)\hat{r}m_k\lambda + (1/2)\left(3\rho^2 m_k^2 + \tilde{\sigma}_k^2\right)\lambda^2\hat{r}}\,. \qquad (4.89)$$

Substituting the bound in (4.89) in Equation (4.82), we get

$$\begin{aligned}
&\mathbb{E}[e^{\lambda \mathbf{x}_q^{(k+1)}}|q] \\
\leq\;& \Big(\mathbb{E}_p\big[(pq + \bar{\mathbf{p}}\bar{\mathbf{q}})e^{\rho^2(2p-1)m_k\lambda\hat{r}} + (p\bar{\mathbf{q}} + \bar{\mathbf{p}}q)e^{-\rho^2(2p-1)m_k\lambda\hat{r}}\big]\Big)^{\hat{\ell}} e^{(1/2)(3\rho^2 m_k^2 + \tilde{\sigma}_k^2)\lambda^2\hat{\ell}\hat{r}} \\
\leq\;& \Big(\mathbb{E}_p\big[e^{(2q-1)(2p-1)^2\rho^2 m_k\lambda\hat{r} + (1/2)(2p-1)^2(\rho^2 m_k\lambda\hat{r})^2}\big]\Big)^{\hat{\ell}} e^{(1/2)(3\rho^2 m_k^2 + \tilde{\sigma}_k^2)\lambda^2\hat{\ell}\hat{r}} \qquad (4.90) \\
\leq\;& e^{\hat{\ell}\hat{r}\rho^2\sigma^2 m_{k,q}\lambda + (1/2)\hat{\ell}\hat{r}\left(\tilde{\sigma}_k^2 + 3\rho^2 m_k^2(1+\hat{r}\rho^2\sigma^2)\right)\lambda^2}\,, \qquad (4.91)
\end{aligned}$$

where (4.90) uses the inequality $ae^{z}+(1-a)e^{-z}\leq e^{(2a-1)z+(1/2)z^2}$. Equation (4.91) follows from the application of Lemma 4.12, with $s=(2p-1)^2$, $z=\rho^2 m_k\lambda\hat{r}$ and $t=(2q-1)$. For $|\lambda|\leq 1/(2m_k\hat{r}\rho^2)$, $|z|<(1/2)$.

In the regime where $\hat{\ell}\hat{r}(\rho^2\sigma^2)^2>1$, as per our assumption, m_k is non-decreasing in k. At iteration k, the above recursion holds for

$$|\lambda|\leq 1/(2\hat{r}\rho^2)\min\{1/m_1,\cdots,1/m_{k-1}\}=1/(2m_{k-1}\hat{r}\rho^2)\,.$$

Hence, we get the following recursion for $m_{k,q}$ and $\tilde{\sigma}_k^2$ such that (4.80) holds for $|\lambda|\leq 1/(2m_{k-1}\hat{r}\rho^2)$:

$$\begin{aligned} m_{k,q} &= \hat{\ell}\hat{r}\rho^2\sigma^2 m_{k-1,q}, \\ \tilde{\sigma}_k^2 &= \hat{\ell}\hat{r}\tilde{\sigma}_{k-1}^2+3\hat{\ell}\hat{r}(1+\hat{r}\rho^2\sigma^2)\rho^2 m_{k-1}^2\,. \end{aligned} \tag{4.92}$$

With the initialization $m_{1,q}=(2q-1)\mu\hat{\ell}$ and $\tilde{\sigma}_1^2=2\hat{\ell}$, we have $m_{k,q}=\mu(2q-1)\hat{\ell}(\rho^2\sigma^2\hat{\ell}\hat{r})^{k-1}$ for $k\in\{1,2,\cdots\}$ and $\tilde{\sigma}_k^2=a\tilde{\sigma}_{k-1}^2+bc^{k-2}$ for $k\in\{2,3\cdots\}$, with $a=\hat{\ell}\hat{r}$, $b=3\hat{\ell}^3\hat{r}\mu^2\rho^2(1+\rho^2\sigma^2\hat{r})$, and $c=(\rho^2\sigma^2\hat{\ell}\hat{r})^2$. After some algebra, we have $\tilde{\sigma}_k^2=\tilde{\sigma}_1^2 a^{k-1}+bc^{k-2}\sum_{\ell=0}^{k-2}(a/c)^{\ell}$. For $\hat{\ell}\hat{r}(\rho^2\sigma^2)^2\neq 1$, we have $a/c\neq 1$, whence $\tilde{\sigma}_k^2=\tilde{\sigma}_1^2 a^{k-1}+bc^{k-2}(1-(a/c)^{k-1})/(1-a/c)$. This finishes the proof of (4.80).

4.6.7 Proof of Lemma 4.12

Using the fact that $e^a\leq 1+a+0.63a^2$ for $|a|\leq 5/8$,

$$\begin{aligned} & \mathbb{E}\big[e^{stz+(1/2)sz^2}\big] \\ \leq\ & \mathbb{E}\big[1+stz+(1/2)sz^2+0.63\big(stz+(1/2)sz^2\big)^2\big] \\ \leq\ & \mathbb{E}\big[1+stz+(1/2)sz^2+0.63\big((5/4)z\sqrt{s}\big)^2\big] \\ \leq\ & 1+\mathbb{E}[s]tz+(3/2)\mathbb{E}[s]z^2 \\ \leq\ & \exp\big(\mathbb{E}[s]tz+(3/2)\mathbb{E}[s]z^2\big)\,. \end{aligned}$$

4.6.8 Proof of Theorem 4.6

Let $\mathcal{P}$ denote a distribution on the worker quality p_j such that $p_j \sim \mathcal{P}$. Let $\mathcal{P}_{\sigma^2}$ be a collection of all distributions $\mathcal{P}$ such that:

$$\mathcal{P}_{\sigma^2} = \left\{ \mathcal{P} \mid \mathbb{E}_{\mathcal{P}}[(2p_j - 1)^2] = \sigma^2 \right\}.$$

Define the minimax rate on the probability of error of a task i, conditioned on its difficulty level q_i, as

$$\min_{\tau \in \mathscr{T}_{\ell_i}, \hat{t}} \ \max_{t_i \in \{\pm\}, \mathcal{P} \in \mathcal{P}_{\sigma^2}} \mathbb{P}[t_i \neq \hat{t}_i \mid q_i], \tag{4.93}$$

where $\mathscr{T}_{\ell_i}$ is the set of all nonadaptive task assignment schemes that assign ℓ_i workers to task i, and $\hat{t}$ ranges over the set of all estimators of t_i. Since the minimax rate is the maximum over all the distributions $\mathcal{P} \in \mathcal{P}_{\sigma^2}$, we consider a particular worker quality distribution to get a lower bound on it. In particular, we assume the p_j's are drawn from a spammer-hammer model with perfect hammers:

$$p_j \;\; = \;\; \begin{cases} 1/2 & \text{with probability } 1 - \sigma^2, \\ 1 & \text{otherwise.} \end{cases}$$

Observe that the chosen spammer-hammer models belongs to $\mathcal{P}_{\sigma^2}$, i.e. $\mathbb{E}[(2p_j - 1)^2] = \sigma^2$. To get the optimal estimator, we consider an oracle estimator that knows all the p_j's and hence makes an optimal estimation. It estimates $\hat{t}_i$ using majority voting on hammers and ignores the answers of hammers. If there are no hammers then it flips a fair coin and estimates $\hat{t}_i$ correctly with half probability. It does the same in case of tie among the hammers. Concretely,

$$\hat{t}_i \;\; = \;\; \text{sign}\Bigg(\sum_{j \in W_i} \mathbb{I}\{j \in \mathbb{H}\}) A_{ij} \Bigg),$$

where W_i denotes the neighborhood of node i in the graph and $\mathbb{H}$ is the set of hammers. Note that this is the optimal estimation for the spammer-hammer model. We want to compute a lower bound on $\mathbb{P}[t_i \neq \hat{t}_i | q_i]$. Let $\tilde{\ell}_i$ be the number of hammers answering task i, i.e.,$\tilde{\ell}_i = |W_i \cap \mathbb{H}|$. Since p_j's are drawn from spammer-hammer model, $\tilde{\ell}_i$ is a binomial random variable

Binom(ℓ_i, σ^2). We first compute probability of error conditioned on $\tilde{\ell}_i$, i.e. $\mathbb{P}[t_i \neq \hat{t}_i | \tilde{\ell}_i, q_i]$. For this, we use the following lemma from [107].

Lemma 4.13 (Lemma 2 from [107]). *For any $C < 1$, there exists a positive constant C' such that when $(2q_i - 1) \leq C$, the error achieved by majority voting is at least*

$$\min_{\tau \in \mathscr{T}_{\tilde{\ell}}} \max_{t_i \in \{\pm\}} \mathbb{P}[t_i \neq \hat{t}_i | \tilde{\ell}_i, q_i] \;\; \geq \;\; e^{-C'(\tilde{\ell}_i (2q_i - 1)^2 + 1)} . \tag{4.94}$$

Taking expectation with respect to random variable $\tilde{\ell}_i$ and applying Jensen's inequality on the term in right side, we get a lower bound on the minimax probability of error in (4.93)

$$\min_{\tau \in \mathscr{T}_{\tilde{\ell}}, \hat{t}} \max_{\substack{\mathcal{P} \in \mathcal{P}_{\sigma^2} \\ t_i \in \{\pm\}}} \mathbb{P}[t_i \neq \hat{t}_i | q_i] \;\; \geq \;\; e^{-C'(\ell_i \sigma^2 (2q_i - 1)^2 + 1)} . \tag{4.95}$$

4.7 Discussion

Recent theoretical advances in crowdsourcing systems have not been able to explain the gain in *adaptive* task assignments, widely used in practice. This is mainly due to the fact that existing models of the worker responses failed to capture the heterogeneity of the tasks, while the gain in adaptivity is signified when tasks are widely heterogeneous. To bridge this gap, we propose studying the gain of adaptivity under a more general model recently introduced by [220], which we call the generalized Dawid-Skene model.

We identify that the minimax error rate decays as $e^{-C\lambda\sigma^2\Gamma/m}$, where the dependence on the heterogeneity in the task difficulties is captured by the error exponent λ defined as (6.27). This is proved by showing a fundamental limit in Theorem 4.1 analyzing the best possible adaptive task assignment scheme, together with the best possible inference algorithm, where the nature chooses the worst-case task difficulty parameters $q = (q_1, \ldots, q_m)$ and the worst-case worker reliability parameters $p = (p_1, \ldots, p_n)$. We propose an efficient adaptive task assignment scheme together with an efficient inference algorithm that matches the minimax error rate as shown in Theorem 4.4. To characterize the gain in adaptivity, we also identify the minimax error rate of non-adaptive schemes decaying as $e^{-C'\lambda_{\min}\sigma^2\ell}$, where $\lambda_{\min}$ is strictly smaller than λ. We show this fundamental limit in Theorem 4.6 and a matching efficient scheme in Theorem 4.5. Hence, the gain of adaptivity is captured in the budget required to achieve a target accuracy, which differ by a factor of $\lambda/\lambda_{\min}$.

Adaptive task assignment schemes for crowdsourced classifications have been first addressed in [91], where a similar setting was assumed. Tasks are binary classification tasks, with heterogeneous difficulties, and workers arrive in an online fashion. One difference is that, [91] studies a slightly more general model where tasks are partitioned into a finite number of types and the worker error probability only depends on the type (and the identity of the worker), i.e. $\mathbb{P}(A_{ij} = t_i) = f(T(i), j)$ where $T(i)$ is the type of the task i. This includes the generalized Dawid-Skene model, if we restrict the difficulty q_i's from a finite set. [91] provides an adaptive scheme based on a linear program relaxation, and show that the sufficient condition to achieve average error ε is for the average total budget to be larger than,

$$\Gamma_\varepsilon \;\; \geq \;\; C\frac{m}{\lambda_{\min}\lambda\sigma^2}\big(\log(1/\varepsilon)\big)^{3/2} \, .$$

Compared to the sufficient condition in (4.27), this is larger by a factor of $(1/\lambda_{\min})\sqrt{\log(1/\varepsilon)}$. In fact, this is larger than what can be achieved with a non-adaptive scheme in (4.38).

On the other hand, there are other types of expert systems, where a finite set of experts are maintained and a stream of incoming tasks are assigned. This clearly departs from typical crowdsourcing scenario, as the experts are identifiable and can be repeatedly assigned tasks. One can view this as a multi-armed bandit problem with noisy feedback [56, 219, 63, 145], and propose task assignment schemes with guarantees on the regret.

We have provided a precise characterization of the minimax rate under the generalized Dawid-Skene model. Such a complete characterization is only known only for a few simple cases: binary classification tasks with symmetric Dawid-Skene model in [107] and binary classification tasks with symmetric generalized Dawid-Skene model in this paper. Even for binary classification tasks, there are other models where such fundamental trade-offs are still unknown: e.g. permutation-based model in [185]. The analysis techniques developed in this paper does not directly generalize to such models, and it remains an interesting challenge.

Technically, our analysis could be improved in two directions: finite Γ/m regime and parameter estimation. First, our analysis is asymptotic in the size of the problem, and also in the average degree of the task $\ell \equiv \Gamma/m$ which increases as $\log m$. This is necessary for applying the central limit theorem. However, in practice, we observe the same error rate when ℓ does not necessarily increases with m. In order to generalize our analysis to finite ℓ regime, we need sharp bounds on the tail of a sub-Gaussian tail of the distribution of the messages. This is partially plausible, and we provide an upper bound on this tail in (4.91). However, the main challenge is that we also need a lower bound on this tail, which is generally difficult.

Secondly, we empirically observe that our parameter estimation algorithm in Algorithm 8 works well in practice. It is possible to precisely analyze the sample complexity of this estimator using spectral analysis. However, such an error in the value of $\rho^2_{t,u}$ used in the inner-loop can result in accumulated errors over iterations, and it is not clear how to analyze it. Currently, we do not have the tools to analyze such error propagation, which is a challenging research direction. Also, the parameter estimation algorithm can be significantly improved, by applying some recent advances in estimating such smaller dimensional spectral properties of such random matrices, for example [217, 119, 129, 115], which is an active topic for research.

notation	data type	definition
μ	$[-1,1]$	average reliability of the crowd as per $\mathcal{P}$: $\mathbb{E}_{\mathcal{P}}[2p_j - 1]$
σ^2	$[0,1]$	collective reliability of the crowd as per $\mathcal{P}$: $\mathbb{E}_{\mathcal{P}}[(2p_j - 1)^2]$
λ_i	$[0,1]$	individual difficulty level of task i: $(2q_i - 1)^2$
$\lambda_{\min}$	$[0,1]$	worst-case difficulty as per $\mathcal{Q}$: $\min_{q_i \in \mathrm{supp}(\mathcal{Q})}(2q_i - 1)^2$
$\lambda_{\max}$	$[0,1]$	best-case difficulty as per $\mathcal{Q}$: $\max_{q_i \in \mathrm{supp}(\mathcal{Q})}(2q_i - 1)^2$
λ	$[0,1]$	collective difficulty level of the tasks as per $\mathcal{Q}$: $\mathbb{E}_{\mathcal{Q}}[(2q_i - 1)^{-2}]^{-1}$
$\widehat{\lambda}$	$[0,1]$	collective difficulty level of the tasks as per $\widehat{\mathcal{Q}}$: $(\sum_{a \in [T]} \delta_a/\lambda_a)^{-1}$
ρ^2	$[0,1]$	average difficulty of tasks as per $\mathcal{Q} : \mathbb{E}_{\mathcal{Q}}[(2q_i - 1)^2]$
a	$[T]$	index for support points of quantized distribution $\widehat{\mathcal{Q}}$
λ_a	$[0,1]$	difficulty level of a-th support point of $\widehat{\mathcal{Q}}$
δ_a	$[0,1]$	probability mass at λ_a in $\widehat{\mathcal{Q}}$
$\delta_{\min}$	$[0,1]$	minimum probability mass in $\widehat{\mathcal{Q}}$: $\min_{a \in [T]} \delta_a$
$\delta_{\max}$	$[0,1]$	maximum probability mass in $\widehat{\mathcal{Q}}$: $\max_{a \in [T]} \delta_a$
T	$\mathbb{Z}_+$	number of rounds in Algorithm 4
t	$\mathbb{Z}_+$	index for a round in Algorithm 4
s_t	$\mathbb{Z}_+$	number of sub-rounds in round t of Algorithm 4
u	$\mathbb{Z}_+$	index for a sub-round of Algorithm 4

Table 4.2: Notations

CHAPTER 5

LEARNING FROM NOISY SINGLY-LABELED DATA

The traditional crowdsourcing problem addresses the challenge of aggregating multiple noisy labels. A naive approach is to aggregate the labels based on majority voting. More sophisticated agreement-based algorithms jointly model worker skills and ground truth labels, estimating both using EM or similar techniques [47, 101, 208, 205, 223, 135, 44, 135]. [218] shows that the EM algorithm with spectral initialization achieves minimax optimal performance under the Dawid-Skene model. [108] introduces a message-passing algorithm for estimating binary labels under the Dawid-Skene model, showing that it performs strictly better than majority voting when the number of labels per example exceeds some threshold. Similar observations are made by [27]. A primary criticism of EM-based approaches is that in practice, it's rare to collect more than 3 to 5 labels per example; and with so little redundancy, the small gains achieved by EM over majority voting are not compelling to practitioners. In contrast, our algorithm performs well in the low-redundancy setting. Even with just one label per example, we can accurately estimate worker quality.

Several prior crowdsourcing papers incorporate the predictions of a supervised learning model, together with the noisy labels, to estimate the ground truth labels. [205] consider binary classification and frames the problem as a generative Bayesian model on the features of the examples and the labels. [28] considers a generalization of the Dawid-Skene model and estimates its parameters using supervised learning in the loop. In particular, they consider a joint probability over observed image features, ground truth labels, and the worker labels and computes the maximum likelihood estimate of the true labels using alternating minimization. We also consider a joint probability model but it is significantly different from theirs as we assume that the optimal labeling function gives the ground truth labels. We maximize the joint likelihood using a variation of expectation maximization to learn the optimal labeling function and the true labels. Further, they train the supervised learning model using the intermediate predictions of the labels whereas we train the model by minimizing a weighted loss function where the weights are the intermediate posterior probability distribution of the labels. Moreover, with only one label per example, their algorithm fails and estimates all the workers to be equally good. They only consider binary classification, whereas we verify our algorithm on multi-class (ten classes) classification problem.

A rich body of work addresses human-in-loop annotation for computer vision tasks. However, these works assume that humans are experts, i.e., that they give noiseless annotations [29, 51, 200]. We assume workers are unreliable and have varying skills. A recent work by [172] also proposes to use predictions of a supervised learning model to estimate the ground truth. However, their algorithm is significantly different than ours as it does not use iterative estimation technique, and their approach of incorporating worker quality parameters in the supervised learning model is different. Their theoretical results are limited to the linear classifiers.

Another line of work employs active learning, iteratively filtering out examples for which aggregated labels have high confidence and collect additional labels for the remaining examples [208, 206, 116]. The underlying modeling assumption in these papers is that the questions have varying levels of difficulty. At each iteration, these approaches employ an EM-based algorithm to estimate the ground truth label of the remaining unclassified examples. For simplicity, our paper does not address example difficulties, but we could easily extend our model and algorithm to accommodate this complexity.

Several papers analyze whether repeated labeling is useful. [191] analyzed the effect of repeated labeling and showed that it depends upon the relative cost of getting an unlabeled example and the cost of labeling. [96] shows that if worker quality is below a threshold then repeated labeling is useful, otherwise not. [131, 130] argues that it also depends upon expressiveness of the classifier in addition to the factors considered by others. However, these works do not exploit predictions of the supervised learning algorithm to estimate the ground truth labels, and hence their findings do not extend to our methodology.

Another body of work that is relevant to our problem is learning with noisy labels where usual assumption is that all the labels are generated through the same noisy rate given their ground truth label. Recently [155] proposed a generic unbiased loss function for binary classification with noisy labels. They employed a modified loss function that can be expressed as a weighted sum of the original loss function, and gave theoretical bounds on the performance. However, their weights become unstably large when the noise rate is large, and hence the weights need to be tuned. [195, 102] learns noise rate as parameters of the model. A recent work by [83] trains an individual softmax layer for each expert and then predicts their weighted sum where weights are also learned by the model. It is not scalable to crowdsourcing scenario where there are thousands of workers. There are works that aim to create noise-robust models [103, 120], but they are not relevant to our work.

5.1 Problem Formulation

Let $\mathcal{D}$ be the underlying true distribution generating pairs $(X, Y) \in \mathcal{X} \times \mathcal{K}$ from which n i.i.d. samples $(X_1, Y_1), (X_2, Y_2), \cdots, (X_n, Y_n)$ are drawn, where $\mathcal{K}$ denotes the set of possible labels $\mathcal{K} \coloneqq \{1, 2, \cdots, K\}$, and $\mathcal{X} \subseteq \mathbb{R}^d$ denotes the set of euclidean features. We denote the marginal distribution of Y by $\{q_1, q_2, \cdots, q_K\}$, which is unknown to us. Consider a pool of m workers indexed by $1, 2, \cdots, m$. We use $[m]$ to denote the set $\{1, 2, \cdots, m\}$. For each i-th sample X_i, r workers $\{w_{ij}\}_{j \in [r]} \in [m]^r$ are selected randomly, independent of the sample X_i. Each selected worker provides a noisy label Z_{ij} for the sample X_i, where the distribution of Z_{ij} depends on the selected worker and the true label Y_i. We call r the *redundancy* and, for simplicity, assume it to be the same for each sample. However, our algorithm can also be applied when redundancy varies across the samples. We use $Z_i^{(r)}$ to denote $\{Z_{ij}\}_{j \in [r]}$, the set of r labels collected on the i-th example, and $w_i^{(r)}$ to denote $\{w_{ij}\}_{j \in [r]}$.

Following [47], we assume the probability that the a-th worker labels an item in class $k \in \mathcal{K}$ as class $s \in \mathcal{K}$ is independent of any particular chosen item, that is, it is a constant over $i \in [n]$. Let us denote this constant by π_{ks}; by definition, $\sum_{s \in \mathcal{K}} \pi_{ks} = 1$ for all $k \in \mathcal{K}$, and we call $\pi^{(a)} \in [0, 1]^{K \times K}$ the confusion matrix of the a-th worker. In particular, the distribution of Z is:

$$\mathbb{P}\left[Z_{ij} = s \mid Y_i = k, w_{ij} = a\right] = \pi_{ks}^{(a)} . \tag{5.1}$$

The diagonal entries of the confusion matrix correspond to the probabilities of correctly labeling an example. The off-diagonal entries represent the probability of mislabeling. We use π to denote the collection of confusion matrices $\{\pi^{(a)}\}_{a\in[m]}$.

We assume nr workers $w_{1,1}, w_{1,2}, \cdots, w_{n,r}$ are selected uniformly at random from a pool of m workers with replacement and a batch of r workers are assigned to each of the examples $X_1, X_2, \cdots, X_n$. The corrupted labels along with the worker information

$$(X_1, Z_1^{(r)}, w_1^{(r)}), \cdots, (X_n, Z_n^{(r)}, w_n^{(r)})$$

are what the learning algorithm sees.

Let $\mathcal{F}$ be the hypothesis class, and $f \in \mathcal{F}$, $f : \mathcal{X} \to \mathbb{R}^K$, denote a vector valued predictor function. Let $\ell(f(X), Y)$ denote a loss function. For a predictor f, its ℓ-risk under $\mathcal{D}$ is defined as

$$R_{\ell,\mathcal{D}}(f) \;\coloneqq\; \mathbb{E}_{(X,Y)\sim\mathcal{D}}\left[\ell(f(X), Y)\right] . \tag{5.2}$$

Given the observed samples $(X_1, Z_1^{(r)}, w_1^{(r)}), \cdots, (X_n, Z_n^{(r)}, w_n^{(r)})$, we want to learn a good predictor function $\widehat{f} \in \mathcal{F}$ such that its risk under the true distribution $\mathcal{D}$, $R_{\ell,\mathcal{D}}(\widehat{f})$ is minimal. Having access to only noisy labels $Z^{(r)}$ by workers $w^{(r)}$, we compute $\widehat{f}$ as the one which minimizes a suitably modified loss function $\ell_{\widehat{\pi},\widehat{q}}(f(X), Z^{(r)}, w^{(r)})$. Where $\widehat{\pi}$ denote an estimate of confusion matrix π, and $\widehat{q}$ an estimate of q, the prior distribution on Y. We define $\ell_{\widehat{\pi},\widehat{q}}$ in the following section.

5.2 Algorithm

Assume that there exists a function $f^* \in \mathcal{F}$ such that $f^*(X_i) = Y_i$ for all $i \in [n]$. Under the Dawid-Skene model (described in previous section), the joint likelihood of true labeling function $f^*(X_i)$ and observed labels $\{Z_{ij}\}_{i\in[n],j\in[r]}$ as a function of confusion matrices of workers π can be written as

$$\begin{aligned} &L\left(\pi; f^*, \{X_i\}_{i\in[n]}, \{Z_{ij}\}_{i\in[n],j\in[r]}\right) \coloneqq \\ &\prod_{i=1}^{n}\left(\sum_{k\in\mathcal{K}} q_k \mathbb{I}[f^*(X_i) = k]\left(\prod_{j=1}^{r}\left(\sum_{s\in\mathcal{K}} \mathbb{I}[Z_{ij} = s]\pi_{ks}^{(w_{ij})}\right)\right)\right) . \end{aligned} \tag{5.3}$$

q_k's are the marginal distribution of the true labels Y_i's. We estimate the worker confusion matrices π and the true labeling function f^* by maximizing the likelihood function $L(\pi; f^*(X), Z)$. Observe that the likelihood function $L(\pi; f^*(X), Z)$ is different than the standard likelihood function of Dawid-Skene model in that we replace each true hidden labels Y_i by $f^*(X_i)$. Like the EM algorithm introduced in [47], we propose 'Model Bootstrapped EM' (MBEM) to estimate confusion matrices π and the true labeling function f^*. EM converges to the true confusion matrices and the true labels given an appropriate spectral initialization of worker confusion matrices [218]. We show in Section 5.2.4 that MBEM converges under mild conditions when the worker quality is above a threshold and the number of training examples is sufficiently large. In the following two subsections, we motivate and explain our iterative algorithm to estimate the true labeling function f^* given a good estimate of worker confusion matrices π and vice-versa.

5.2.1 Learning with noisy labels

To begin, we ask, *what is the optimal approach to learn the predictor function $\widehat{f}$ when for each worker we have $\widehat{\pi}$, a good estimation of the true confusion matrix π, and $\widehat{q}$, an estimate of the prior*? A recent paper, [155] proposes minimizing an unbiased loss function specifically, a weighted sum of the original loss over each possible ground truth label. They provide weights for binary classification where each example is labeled by only one worker. Consider a worker with confusion matrix π, where $\pi_y > 1/2$ and $\pi_{-y} > 1/2$ represent her probability of correctly labeling the examples belonging to class y and $-y$ respectively. Then their weights are $\pi_{-y}/(\pi_y + \pi_{-y} - 1)$ for class y and $-(1 - \pi_y)/(\pi_y + \pi_{-y} - 1)$ for class $-y$. It is evident that their weights become unstably large when the probabilities of correct classification π_y and π_{-y} are close to 1/2, limiting the method's usefulness in practice. As explained below, for the same scenario, our weights would be $\pi_y/(1+\pi_y-\pi_{-y})$ for class y and $(1-\pi_{-y})/(1+\pi_y-\pi_{-y})$ for class $-y$. Inspired by their idea, we propose weighing the loss function according to the posterior distribution of the true label given the $Z^{(r)}$ observed labels and an estimate of the confusion matrices of the worker who provided those labels. In particular, we define

$\ell_{\widehat{\pi},\widehat{q}}$ to be

$$\ell_{\widehat{\pi},\widehat{q}}(f(X), Z^{(r)}, w^{(r)}) \;\; := \;\; \sum_{k\in\mathcal{K}} \mathbb{P}_{\widehat{\pi},\widehat{q}}[Y = k \mid Z^{(r)}; w^{(r)}]\; \ell(f(X), Y = k)\,. \tag{5.4}$$

If the observed label is uniformly random, then all weights are equal and the loss is identical for all predictor functions f. Absent noise, we recover the original loss function. Under the Dawid-Skene model, given the observed noisy labels $Z^{(r)}$, an estimate of confusion matrices $\widehat{\pi}$, and an estimate of prior $\widehat{q}$, the posterior distribution of the true labels can be computed as follows:

$$\mathbb{P}_{\widehat{\pi},\widehat{q}}[Y_i = k \mid Z_i^{(r)}; w_i^{(r)}] \;\; = \;\; \frac{\widehat{q}_k \prod_{j=1}^{r} \Big(\sum_{s\in\mathcal{K}} \mathbb{I}[Z_{ij} = s] \widehat{\pi}_{ks}^{(w_{ij})} \Big)}{\sum_{k'\in\mathcal{K}} \Big(\widehat{q}_{k'} \prod_{j=1}^{r} \Big(\sum_{s\in\mathcal{K}} \mathbb{I}[Z_{ij} = s] \widehat{\pi}_{k's}^{(w_{ij})} \Big) \Big)}\,, \tag{5.5}$$

where $\mathbb{I}[.]$ is the indicator function which takes value one if the identity inside it is true, otherwise zero. We give guarantees on the performance of the proposed loss function in Theorem 7.3. In practice, it is robust to noise level and significantly outperforms the unbiased loss function. Given $\ell_{\widehat{\pi},\widehat{q}}$, we learn the predictor function $\widehat{f}$ by minimizing the empirical risk

$$\widehat{f} \;\; \leftarrow \;\; \arg\min_{f\in\mathcal{F}} \; \frac{1}{n} \sum_{i=1}^{n} \ell_{\widehat{\pi},\widehat{q}}(f(X_i), Z_i^{(r)}, w_i^{(r)})\,. \tag{5.6}$$

5.2.2 Estimating annotator noise

The next question is: how do we get a good estimate $\widehat{\pi}$ of the true confusion matrix π for each worker. If redundancy r is sufficiently large, we can employ the EM algorithm. However, in practical applications, redundancy is typically three or five. With so little redundancy, the standard applications of EM are of limited use. In this paper we look to transcend this problem, posing the question: Can we estimate confusion matrices of workers even when there is only one label per example? While this isn't possible in the standard approach, we can overcome this obstacle by incorporating a supervised learning model into the process of assessing worker quality.

Under the Dawid-Skene model, the EM algorithm estimates the ground truth labels and the confusion matrices in the following way: It alternately fixes the ground truth labels and the confusion matrices by their estimates and and updates its estimate of the other by maximizing the likelihood of the observed labels. The alternating maximization begins by initializing the ground truth labels with a majority vote. With only 1 label per example, EM estimates that all the workers are perfect.

We propose using model predictions as estimates of the ground truth labels. Our model is initially trained on the majority vote of the labels. In particular, if the model prediction is $\{t_i\}_{i\in[n]}$, where $t_i \in \mathcal{K}$, then the maximum likelihood estimate of confusion matrices and the prior distribution is given below. For the a-th worker, $\widehat{\pi}^{(a)}_{ks}$ for $k, s \in \mathcal{K}$, and $\widehat{q}_k$ for $k \in \mathcal{K}$, we have,

$$\widehat{\pi}^{(a)}_{ks} = \frac{\sum_{i=1}^{n}\sum_{j=1}^{r}\mathbb{I}[w_{ij}=a]\mathbb{I}[t_i=k]\mathbb{I}[Z_{ij}=s]}{\sum_{i=1}^{n}\sum_{j=1}^{r}\mathbb{I}[w_{ij}=a]\mathbb{I}[t_i=k]}, \qquad \widehat{q}_k = (1/n)\sum_{i=1}^{n}\mathbb{I}[t_i=k] \tag{5.7}$$

The estimate is effective when the hypothesis class $\mathcal{F}$ is expressive enough and the learner is robust to noise. Thus the model should, in general, have small training error on correctly labeled examples and large training error on wrongly labeled examples. Consider the case when there is only one label per example. The model will be trained on the raw noisy labels given by the workers. For simplicity, assume that each worker is either a *hammer* (always correct) or a *spammer* (chooses labels uniformly random). By comparing model predictions with the training labels, we can identify which workers are hammers and which are spammers, as long as each worker labels sufficiently many examples. We expect a hammer to agree with the model more often than a spammer.

5.2.3 Iterative Algorithm

Building upon the previous two ideas, we present ‘Model Bootstrapped EM’, an iterative algorithm for efficient learning from noisy labels with small redundancy. MBEM takes data, noisy labels, and the corresponding worker IDs, and returns the best predictor function $\widehat{f}$ in the hypothesis class $\mathcal{F}$. In the first round, we compute the weights of the modified loss function $\ell_{\widehat{\pi},\widehat{q}}$ by using the weighted majority vote. Then we obtain an estimate of the worker confusion matrices $\widehat{\pi}$ using the maximum likelihood estimator by taking the model predictions as the ground truth labels. In the second round, weights of the loss function are computed as the posterior probability distribution of the ground truth labels conditioned on the noisy labels and the estimate of the confusion matrices obtained in the previous round. In our experiments, only two rounds are required to achieve substantial improvements over baselines.

```
**Algorithm 7** Model Bootstrapped EM (MBEM)
Input: $\{(X_i, Z_i^{(r)}, w_i^{(r)})\}_{i\in[n]}$, $T$ : number of iterations
Output: $\widehat{f}$ : predictor function
Initialize posterior distribution using weighted majority vote
  $\mathbb{P}_{\widehat{\pi},\widehat{q}}[Y_i = k \mid Z_i^{(r)}; w_i^{(r)}] \leftarrow (1/r)\sum_{j=1}^{r}\mathbb{I}[Z_{ij} = k]$ , for $k \in \mathcal{K}, i \in [n]$
Repeat $T$ times:
  learn predictor function $\widehat{f}$
  $\widehat{f} \leftarrow \arg\min_{f\in\mathcal{F}} \frac{1}{n}\sum_{i=1}^{n}\sum_{k\in\mathcal{K}}\mathbb{P}_{\widehat{\pi},\widehat{q}}[Y_i = k \mid Z_i^{(r)}; w_i^{(r)}]\ \ell(f(X_i), Y_i = k)$
  predict on training examples
  $t_i \leftarrow \arg\max_{k\in\mathcal{K}} \widehat{f}(X_i)_k$, for $i \in [n]$
  estimate confusion matrices $\widehat{\pi}$ and prior class distribution $\widehat{q}$ given $\{t_i\}_{i\in[n]}$
  $\widehat{\pi}^{(a)} \leftarrow$ Equation (5.7), for $a \in [m]$; $\widehat{q} \leftarrow$ Equation (5.7)
  estimate label posterior distribution given $\widehat{\pi}, \widehat{q}$
  $\mathbb{P}_{\widehat{\pi},\widehat{q}}[Y_i = k \mid Z_i^{(r)}; w_i^{(r)}], \leftarrow$ Equation (5.5), for $k \in \mathcal{K}, i \in [n]$
Return $\widehat{f}$
```

5.2.4 Performance Guarantees

The following result gives guarantee on the excess risk for the learned predictor function $\widehat{f}$ in terms of the VC dimension of the hypothesis class $\mathcal{F}$. Recall that risk of a function f w.r.t. loss function ℓ is defined to be $R_{\ell,\mathcal{D}}(f) := \mathbb{E}_{(X,Y)\sim\mathcal{D}}\left[\ell(f(X), Y)\right]$, Equation (5.2). We assume that the classification problem is binary, and the distribution q, prior on ground truth labels Y, is uniform and is known to us. We give guarantees on the excess risk of the predictor function $\widehat{f}$, and accuracy of $\widehat{\pi}$ estimated in the second round. For the purpose of analysis, we assume that fresh samples are used in each round for computing function $\widehat{f}$ and estimating $\widehat{\pi}$. In other words, we assume that $\widehat{f}$ and $\widehat{\pi}$ are each computed using $n/4$ fresh samples in the first two rounds. We define α and β_ϵ to capture the average worker quality. Here, we give their concise bound for a special case when all the workers are identical, and their confusion matrix is represented by a single parameter, $0 \leq \rho < 1/2$. Where $\pi_{kk} = 1 - \rho$, and $\pi_{ks} = \rho$ for $k \neq s$. Each worker makes a mistake with probability ρ. $\beta_\epsilon \leq (\rho + \epsilon)^r \sum_{u=0}^{r} \binom{r}{u} (\tau^u + \tau^{r-u})^{-1}$, where $\tau \coloneqq (\rho + \epsilon)/(1 - \rho - \epsilon)$. α for this special case is ρ. A general definition of α and β_ϵ for any confusion matrices π is provided in the Appendix.

Theorem 5.1. *Define $N \coloneqq nr$ to be the number of total annotations collected on n training examples with redundancy r. Suppose $\min_{f \in \mathcal{F}} R_{\ell,\mathcal{D}}(f) \leq 1/4$. For any hypothesis class $\mathcal{F}$ with a finite VC dimension V, and any $\delta < 1$, there exists a universal constant C such that if N is large enough and satisfies*

$$N \;\geq\; \max\left\{ Cr\big(\big(\sqrt{V} + \sqrt{\log(1/\delta)}\big)/(1 - 2\alpha)\big)^2, 2^{12} m \log(2^6 m/\delta) \right\}, \quad (5.8)$$

then for binary classification with 0-1 loss function ℓ, $\widehat{f}$ and $\widehat{\pi}$ returned by Algorithm 7 after $T = 2$ iterations satisfies

$$R_{\ell,\mathcal{D}}(\widehat{f}) - \min_{f \in \mathcal{F}} R_{\ell,\mathcal{D}}(f) \;\leq\; \frac{C\sqrt{r}}{1 - 2\beta_\epsilon} \left(\sqrt{\frac{V}{N}} + \sqrt{\frac{\log(1/\delta)}{N}} \right), \quad (5.9)$$

and $\|\widehat{\pi}^{(a)} - \pi^{(a)}\|_\infty \leq \epsilon_1$ for all $a \in [m]$, with probability at least $1 - \delta$. Where $\epsilon \coloneqq 2^4\gamma + 2^8\sqrt{m \log(2^6 m\delta)/N}$, and $\gamma \coloneqq \min_{f \in \mathcal{F}} R_{\ell,\mathcal{D}}(f) + C(\sqrt{V} + \sqrt{\log(1/\delta)})/((1 - 2\alpha)\sqrt{N/r})$. ϵ_1 is defined to be ϵ with α in it replaced by β_ϵ.

The price we pay in generalization error bound on $\widehat{f}$ is $(1 - 2\beta_\epsilon)$. Note that, when n is large, ϵ goes to zero, and $\beta_\epsilon \leq 2\rho(1 - \rho)$, for $r = 1$.

If $\min_{f \in \mathcal{F}} R_{\ell,\mathcal{D}}(f)$ is sufficiently small, VC dimension is finite, and ρ is bounded away from $1/2$ then for $n = O(m \log(m)/r)$, we get ϵ_1 to be sufficiently small. Therefore, for any redundancy r, error in confusion matrix estimation is small when the number of training examples is sufficiently large. Hence, for N large enough, using Equation (5.9) and the bound on β_ϵ, we get that for fixed total annotation budget, the optimal choice of redundancy r is 1 when the worker quality $(1 - \rho)$ is above a threshold. In particular, if $(1 - \rho) \geq 0.825$ then label once is the optimal strategy. However, in experiments we observe that with our algorithm the choice of $r = 1$ is optimal even for much smaller values of worker quality.

5.3 Experiments

We experimentally investigate our algorithm, MBEM, on multiple large datasets. On CIFAR-10 [121] and ImageNet [50], we draw noisy labels from synthetic worker models. We confirm our results on multiple worker models. On the MS-COCO dataset [132], we accessed the real raw data that was used to produce this annotation. We compare MBEM against the following baselines:

- **MV**: First aggregate labels by performing a majority vote, then train the model.
- **weighted-MV**: Model learned using weighted loss function with weights set by majority vote.
- **EM**: First aggregate labels using EM. Then train model in the standard fashion. [47]
- **weighted-EM**: Model learned using weighted loss function with weights set by standard EM.
- **oracle weighted EM**: This model is learned by minimizing ℓ_π, using the true confusion matrices.
- **oracle correctly labeled**: This baseline is trained using the standard loss function ℓ but only using those training examples for which at least one of the r workers has given the true label.

Note that oracle models cannot be deployed in practice. We show them to build understanding only. In the plots, the dashed lines correspond to MV and EM algorithm. The black dashed-dotted line shows generalization error if the model is trained using ground truth labels on all the training examples. For experiments with synthetic noisy workers, we consider two models of worker skill:

- **hammer-spammer:** Each worker is either a *hammer* (always correct) with probability γ or a *spammer* (chooses labels uniformly at random).
- **class-wise hammer-spammer:** Each worker can be a hammer for some subset of classes and a spammer for the others. The confusion matrix in

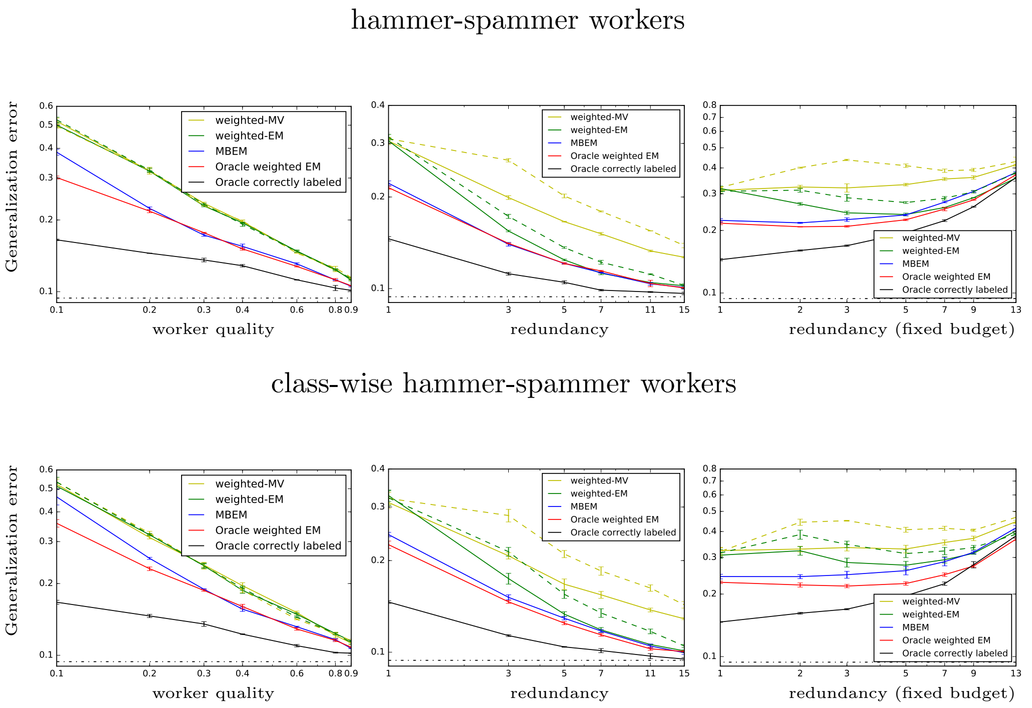
this case has two types of rows: (a) hammer class: row with all off-diagonal elements being 0. (b) spammer class: row with all elements being $1/|\mathcal{K}|$. A worker is a hammer for any class $k \in \mathcal{K}$ with probability γ.

We sample m confusion matrices $\{\pi^{(a)}\}_{a\in[m]}$ according to the given worker skill distribution for a given γ. We assign r workers uniformly at random to each example. Given the ground truth labels, we generate noisy labels according to the probabilities given in a worker's confusion matrix, using Equation (5.1). While our synthetic workers are sampled from these specific worker skill models, our algorithms do not use this information to estimate the confusion matrices. A Python implementation of the MBEM algorithm is available for download at https://github.com/khetan2/MBEM.

CIFAR-10 This dataset has a total of 60K images belonging to 10 different classes where each class is represented by an equal number of images. We use 50K images for training the model and 10K images for testing. We use the ground truth labels to generate noisy labels from synthetic workers. We choose $m = 100$, and for each worker, sample confusion matrix of size 10×10 according to the worker skill distribution. All our experiments are carried out with a 20-layer ResNet which achieves an accuracy of 91.5%. With the larger ResNet-200, we can obtain a higher accuracy of 93.5% but to save training time we restrict our attention to ResNet-20. We run MBEM 7 for $T = 2$ rounds. We assume that the prior distribution $\widehat{q}$ is uniform. We report mean accuracy of 5 runs and its standard error for all the experiments.

Figure 5.1 shows plots for CIFAR-10 dataset under various settings. The three plots in the first row correspond to "hammer-spammer" worker skill distribution and the plots in the second row correspond to "class-wise hammer-spammer" distribution. In the first plot, we fix redundancy $r = 1$, and plot generalization error of the model for varying hammer probability γ. MBEM significantly outperforms all baselines and closely matches the Oracle weighted EM. This implies MBEM recovers worker confusion matrices accurately even when we have only one label per example. When there is only one label per example, MV, weighted-MV, EM, and weighted-EM all reduce learning with the standard loss function ℓ.

In the second plot, we fix hammer probability $\gamma = 0.2$, and vary redundancy r. This plot shows that weighted-MV and weighted-EM perform signif-


hammer-spammer workers

class-wise hammer-spammer workers

Figure 5.1: Plots for CIFAR-10. Line colors- black: oracle correctly labeled, red: oracle weighted EM, blue: MBEM, green: weighted EM, yellow: weighted MV.

icantly better than MV and EM and confirms that our approach of weighing the loss function with posterior probability is effective. MBEM performs much better than weighted-EM at small redundancy, demonstrating the effect of our bootstrapping idea. However, when redundancy is large, EM works as good as MBEM.

In the third plot, we show that when the total annotation budget is fixed, it is optimal to collect one label per example for as many examples as possible. We fixed hammer probability $\gamma = 0.2$. Here, when redundancy is increased from 1 to 2, the number of of available training examples is reduced by 50%, and so on. Performance of weighted-EM improves when redundancy is increased from 1 to 5, showing that with the standard EM algorithm it might be better to collect redundant annotations for fewer example (as it leads to better estimation of worker qualities) than to singly annotate more examples. However, MBEM always performs better than the standard EM algorithm, achieving lowest generalization error with many singly annotated examples. Unlike standard EM, MBEM can estimate worker qualities even with singly annotated examples by comparing them with model predictions. This cor-

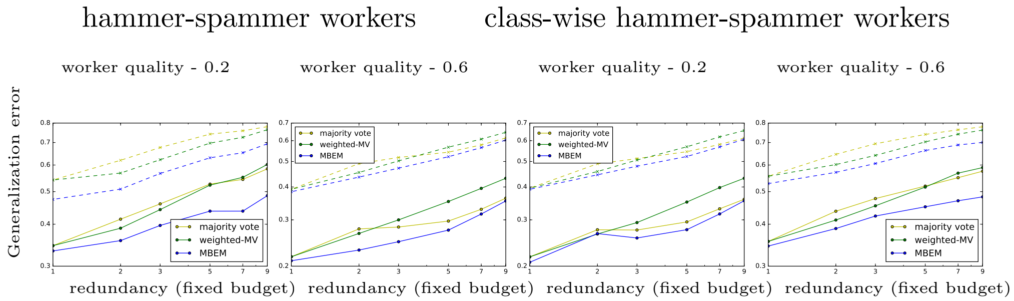


Figure 5.2: Plots for ImageNet. Solid lines represent top-5 error, dashed-lines represent top-1 error. Line colors- blue: MBEM, green: weighted majority vote, yellow: majority vote

roborates our theoretical result that label-once is the optimal strategy when worker quality is above a threshold. The plots corresponding to *class-wise hammer-spammer* workers follow the same trend. Estimation of confusion matrices in this setting is difficult and hence the gap between MBEM and the baselines is less pronounced.

ImageNet The ImageNet-1K dataset contains 1.2M training examples and 50K validation examples. We divide test set in two parts: 10K for validation and 40K for test. Each example belongs to one of the possible 1000 classes. We implement our algorithms using a ResNet-18 that achieves top-1 accuracy of 69.5% and top-5 accuracy of 89% on ground truth labels. We use $m = 1000$ simulated workers. Although in general, a worker can mislabel an example to one of the 1000 possible classes, our simulated workers mislabel an example to only one of the 10 possible classes. This captures the intuition that even with a larger number of classes, perhaps only a small number are easily confused for each other. Therefore, each workers' confusion matrix is of size 10×10. Note that without this assumption, there is little hope of estimating a 1000×1000 confusion matrix for each worker by collecting only approximately 1200 noisy labels from a worker. The rest of the settings are the same as in our CIFAR-10 experiments. In Figure 5.2, we fix total annotation budget to be 1.2M and vary redundancy from 1 to 9. When redundancy is 9, we have only (1.2/9)M training examples, each labeled by 9 workers. MBEM outperforms baselines in each of the plots, achieving the minimum generalization error with many singly annotated training examples.

Approach	F1 score
majority vote	0.433
EM	0.447
MBEM	0.451
ground truth labels	0.512

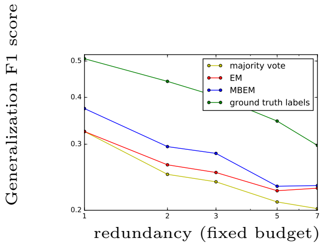


Figure 5.3: Results on raw MS-COCO annotations.

MS-COCO These experiments use the real raw annotations collected when MS-COCO was crowdsourced. Each image in the dataset has multiple objects (approximately 3 on average). For validation set images (out of 40K), labels were collected from 9 workers on average. Each worker marks which out of the 80 possible objects are present. However, on many examples workers disagree. These annotations were collected to label bounding boxes but we ask a different question: what is the best way to learn a model to perform multi-object classification, using these noisy annotations. We use 35K images for training the model and 1K for validation and 4K for testing. We use raw noisy annotations for training the model and the final MS-COCO annotations as the ground truth for the validation and test set. We use ResNet-98 deep learning model and train independent binary classifier for each of the 80 object classes. Table in Figure 5.3 shows generalization F1 score of four different algorithms: majority vote, EM, MBEM using all 9 noisy annotations on each of the training examples, and a model trained using the ground truth labels. MBEM performs significantly better than the standard majority vote and slightly improves over EM. In the plot, we fix the total annotation budget to 35K. We vary redundancy from 1 to 7, and accordingly reduce the number of training examples to keep the total number of annotations fixed. When redundancy is $r < 9$ we select uniformly at random r of the original 9 noisy annotations. Again, we find it best to singly annotate as many examples as possible when the total annotation budget is fixed. MBEM significantly outperforms majority voting and EM at small redundancy.

5.4 Conclusion

We introduced a new algorithm for learning from noisy crowd workers. We also presented a new theoretical and empirical demonstration of the insight that when examples are cheap and annotations expensive, it's better to label many examples once than to label few multiply when worker quality is above a threshold. Many avenues seem ripe for future work. We are especially keen to incorporate our approach into active query schemes, choosing not only which examples to annotate, but which annotator to route them to based on our models current knowledge of both the data and the worker confusion matrices.

5.5 Proofs

Assuming the prior on Y, distribution q, to be uniform, we change the notation for the modified loss function $\ell_{\widehat{\pi},\widehat{q}}$ to $\ell_{\widehat{\pi}}$. Observe that for binary classification, $Z^{(r)} \in \{\pm 1\}^r$. Let ρ denote the posterior distribution of Y, Equation (5.5), when q is uniform. Let τ denote the probability of observing an instance of $Z^{(r)}$ as a function of the latent true confusion matrices π, conditioned on the ground truth label $Y = y$.

$$\rho_{\widehat{\pi}}(y, Z^{(r)}, w^{(r)}) := \mathbb{P}_{\widehat{\pi}}[Y = y \mid Z^{(r)}; w^{(r)}]\,, \tag{5.10}$$

$$\tau_{\pi}(y, Z^{(r)}, w^{(r)}) := \mathbb{P}_{\pi}[Z^{(r)} \mid Y = y; w^{(r)}]\,. \tag{5.11}$$

Let W denote the uniform distribution over a pool of m workers, from which nr workers are selected i.i.d. with replacement, and a batch of r workers are assigned to each example X_i. We define the following quantities which play an important role in our analysis.

$$\beta_{\widehat{\pi}}(y) \;:=\; \mathbb{E}_{w\sim W}\Bigg[\sum_{Z^{(r)}\in\{\pm 1\}^r} \rho_{\widehat{\pi}}(-y, Z^{(r)}, w^{(r)})\tau_{\pi}(y, Z^{(r)}, w^{(r)})\Bigg]\,. \tag{5.12}$$

$$\beta_{\widehat{\pi}} \;:=\; \mathbb{E}_{w\sim W}\Bigg[\max_{y\in\{\pm 1\}}\Bigg\{\sum_{Z^{(r)}\in\{\pm 1\}^r} \rho_{\widehat{\pi}}(-y, Z^{(r)}, w^{(r)})\tau_{\pi}(y, Z^{(r)}, w^{(r)})\Bigg\}\Bigg]\,. \tag{5.13}$$

$$\alpha(y) \;:=\; \mathbb{E}_{w\sim W}\Big[\mathbb{P}_{\pi}[Z=-y \mid Y=y; w]\Big]\,. \tag{5.14}$$

$$\alpha \;:=\; \mathbb{E}_{w\sim W}\Big[\max_{y\in\{\pm 1\}} \mathbb{P}_{\pi}[Z=-y \mid Y=y; w]\Big]\,. \tag{5.15}$$

For any given $\widehat{\pi}$ with $|\widehat{\pi}^{(a)}_{ks} - \pi^{(a)}_{ks}| \le \epsilon$, for all $a \in [m]$, $k, s \in \mathcal{K}$, we can compute β_ϵ from the definition of $\beta_{\widehat{\pi}}$ such that $\beta_{\widehat{\pi}} \le \beta_\epsilon$. For the special case described in Section 5.2.4, we have the following bound on β_ϵ.

$$\begin{aligned} &\beta_\epsilon \\ \le\;& \sum_{u=0}^{r} \frac{(\rho+\epsilon)^{(r-u)}(1-\rho-\epsilon)^u}{(\rho+\epsilon)^u(1-\rho-\epsilon)^{(r-u)} + (\rho+\epsilon)^{(r-u)}(1-\rho-\epsilon)^u}\binom{r}{u}(1-\rho)^{r-u}\rho^u \\ =\;& (\rho+\epsilon)^r \sum_{u=0}^{r}\binom{r}{u}\left(\left(\frac{\rho+\epsilon}{1-\rho-\epsilon}\right)^u + \left(\frac{\rho+\epsilon}{1-\rho-\epsilon}\right)^{r-u}\right)^{-1}\,. \end{aligned} \tag{5.16}$$

It can easily be checked that $\beta_\epsilon \le (\rho+\epsilon)^r \sum_{u=0}^{\lceil r/2\rceil}\binom{r}{u}(1-\rho-\epsilon)^u(\rho+\epsilon)^{-u}$.

We present two lemma that analyze the two alternative steps of our algorithm. The following lemma gives a bound on the excess risk of function $\widehat{f}$ learnt by minimizing the modified loss function $\ell_{\widehat{\pi}}$.

Lemma 5.2. *Under the assumptions of Theorem 7.3, the excess risk of function $\widehat{f}$ in Equation* (5.6)*, computed with posterior distribution $\mathbb{P}_{\widehat{\pi}}$* (5.5) *using n training examples is bounded by*

$$R_{\ell,D}(\widehat{f}) - \min_{f\in\mathcal{F}} R_{\ell,\mathcal{D}}(f) \;\le\; \frac{C}{1-2\beta_{\widehat{\pi}}}\left(\sqrt{\frac{V}{n}} + \sqrt{\frac{\log(1/\delta_1)}{n}}\right)\,, \tag{5.17}$$

with probability at least $1-\delta_1$, where C is a universal constant. When $\mathbb{P}_{\widehat{\pi}}$ is computed using majority vote, while initializing the iterative Algorithm 7, the above bound holds with $\beta_{\widehat{\pi}}$ replaced by α.

The following lemma gives an ℓ_∞ norm bound on confusion matrices $\widehat{\pi}$ estimated using model prediction $\widehat{f}(X)$ as the ground truth labels. In the analysis, we assume fresh samples are used for estimating confusion matrices in step 3, Algorithm 7. Therefore the function $\widehat{f}$ is independent of the samples X_i's on which $\widehat{\pi}$ is estimated. Let $K = |\mathcal{K}|$.

Lemma 5.3. *Under the assumptions of Theorem 7.3, ℓ_∞ error in estimated confusion matrices $\widehat{\pi}$ as computed in Equation* (5.7)*, using n samples and a predictor function $\widehat{f}$ with risk $R_{\ell,\mathcal{D}} \leq \delta$, is bounded by*

$$\left|\widehat{\pi}^{(a)}_{ks} - \pi^{(a)}_{ks}\right| \;\leq\; \frac{2\delta + 16\sqrt{m\log(4mK^2\delta_1)/(nr)}}{1/K - \delta - 8\sqrt{m\log(4mK^2/\delta_1)/(nr)}}\,,\ \forall\, a \in [m],\ k,s \in \mathcal{K}\,, \tag{5.18}$$

with probability at least $1-\delta_1$.

First we apply Lemma 5.2 with $\mathbb{P}_{\widehat{\pi}}$ computed using majority vote. We get a bound on the risk of function $\widehat{f}$ computed in the first round. With this $\widehat{f}$, we apply Lemma 5.3. When n is sufficiently large such that Equation (5.8) holds, the denominator in Equation (5.18), $1/K - \delta - 8\sqrt{m\log(4mK^2/\delta_1)/(nr)} \geq 1/8$. Therefore, in the first round, the error in confusion matrix estimation is bounded by ϵ, which is defined in the Theorem.

For the second round: we apply Lemma 5.2 with $\mathbb{P}_{\widehat{\pi}}$ computed as the posterior distribution (5.5). Where ℓ_∞ error in $\widehat{\pi}$ is bounded by ϵ. This gives the desired bound in (5.9). With this $\widehat{f}$, we apply Lemma 5.3 and obtain ℓ_∞ error in $\widehat{\pi}$ bounded by ϵ_1, which is defined in the Theorem.

For the given probability of error δ in the Theorem, we chose δ_1 in both the lemma to be $\delta/4$ such that with union bound we get the desired probability of δ.

5.5.1 Proof of Lemma 5.2

Let $f^* := \arg\min_{f\in\mathcal{F}} R_{\ell,\mathcal{D}}(f)$. Let's denote the distribution of $(X, Z^{(r)}, w^{(r)})$ by $\mathcal{D}_{W,\pi,r}$. For ease of notation, we denote $\mathcal{D}_{W,\pi,r}$ by $\mathcal{D}_\pi$. Similar to $R_{\ell,\mathcal{D}}$, risk of decision function f with respect to the modified loss function $\ell_{\widehat{\pi}}$ is characterized by the following quantities:

1. $\ell_{\widehat{\pi}}$-risk under $\mathcal{D}_\pi$: $R_{\ell_{\widehat{\pi}},\mathcal{D}_\pi}(f) := \mathbb{E}_{(X,Z^{(r)},w^{(r)})\sim\mathcal{D}_\pi}\big[\ell_{\widehat{\pi}}(f(X), Z^{(r)}, w^{(r)})\big]$.

2. Empirical $\ell_{\widehat{\pi}}$-risk on samples: $\widehat{R}_{\ell_{\widehat{\pi}},\mathcal{D}_\pi}(f) \coloneqq \frac{1}{n}\sum_{i=1}^{n} \ell_{\widehat{\pi}}(f(X_i), Z_i^{(r)}, w_i^{(r)})$.

With the above definitions, we have the following,

$$\begin{aligned}
& R_{\ell,\mathcal{D}}(\widehat{f}) - R_{\ell,\mathcal{D}}(f^*) \\
= \;& R_{\ell_{\widehat{\pi}},\mathcal{D}_\pi}(\widehat{f}) - R_{\ell_{\widehat{\pi}},\mathcal{D}_\pi}(f^*) \\
& + \Big(R_{\ell,\mathcal{D}}(\widehat{f}) - R_{\ell_{\widehat{\pi}},\mathcal{D}_\pi}(\widehat{f})\Big) - (R_{\ell,\mathcal{D}}(f^*) - R_{\ell_{\widehat{\pi}},\mathcal{D}_\pi}(f^*)) \\
\leq \;& R_{\ell_{\widehat{\pi}},\mathcal{D}_\pi}(\widehat{f}) - R_{\ell_{\widehat{\pi}},\mathcal{D}_\pi}(f^*) + 2\beta_{\widehat{\pi}}\Big(R_{\ell,\mathcal{D}}(\widehat{f}) - R_{\ell,\mathcal{D}}(f^*)\Big) && (5.19) \\
= \;& \widehat{R}_{\ell_{\widehat{\pi}},\mathcal{D}_\pi}(\widehat{f}) - \widehat{R}_{\ell_{\widehat{\pi}},\mathcal{D}_\pi}(f^*) \\
& + \Big(R_{\ell_{\widehat{\pi}},\mathcal{D}_\pi}(\widehat{f}) - \widehat{R}_{\ell_{\widehat{\pi}},\mathcal{D}_\pi}(\widehat{f})\Big) + \Big(\widehat{R}_{\ell_{\widehat{\pi}},\mathcal{D}_\pi}(f^*) - R_{\ell_{\widehat{\pi}},\mathcal{D}_\pi}(f^*)\Big) \\
& + 2\beta_{\widehat{\pi}}\Big(R_{\ell,\mathcal{D}}(\widehat{f}) - R_{\ell,\mathcal{D}}(f^*)\Big) \\
\leq \;& 2\max_{f\in\mathcal{F}}\Big|\widehat{R}_{\ell_{\widehat{\pi}},\mathcal{D}_\pi}(f) - R_{\ell_{\widehat{\pi}},\mathcal{D}_\pi}(f)\Big| + 2\beta_{\widehat{\pi}}\Big(R_{\ell,\mathcal{D}}(\widehat{f}) - R_{\ell,\mathcal{D}}(f^*)\Big) && (5.20) \\
\leq \;& C\left(\sqrt{\frac{V}{n}} + \sqrt{\frac{\log(1/\delta)}{n}}\right) + 2\beta_{\widehat{\pi}}\Big(R_{\ell,\mathcal{D}}(\widehat{f}) - R_{\ell,\mathcal{D}}(f^*)\Big)\,, && (5.21)
\end{aligned}$$

where (7.31) follows from Equation (5.24). (5.20) follows from the fact that $\widehat{f}$ is the minimizer of $\widehat{R}_{\ell_{\widehat{\pi}},\mathcal{D}_\pi}$ as computed in (5.6). (5.21) follows from the basic excess-risk bound. V is the VC dimension of hypothesis class $\mathcal{F}$, and C is a universal constant.

Following shows the inequality used in Equation (7.31). For binary classification, we denote the two classes by $Y, -Y$.

$$\begin{aligned}
= \;& R_{\ell,\mathcal{D}}(\widehat{f}) - R_{\ell_{\widehat{\pi}},\mathcal{D}_\pi}(\widehat{f}) - (R_{\ell,\mathcal{D}}(f^*) - R_{\ell_{\widehat{\pi}},\mathcal{D}_\pi}(f^*)) \\
= \;& \mathbb{E}_{(X,Y)\sim\mathcal{D}}\Bigg[\beta_{\widehat{\pi}}(Y)\Bigg(\Big(\ell(\widehat{f}(X),Y) - \ell(f^*(X),Y)\Big) \\
& - \Big(\ell(\widehat{f}(X),-Y) - \ell(f^*(X),-Y)\Big)\Bigg)\Bigg] && (5.22) \\
= \;& 2\mathbb{E}_{(X,Y)\sim\mathcal{D}}\Big[\beta_{\widehat{\pi}}(Y)\Big(\ell(\widehat{f}(X),Y) - \ell(f^*(X),Y)\Big)\Big] && (5.23) \\
\leq \;& 2\beta_{\widehat{\pi}}\Big(R_{\ell,\mathcal{D}}(\widehat{f}) - R_{\ell,\mathcal{D}}(f^*)\Big)\,, && (5.24)
\end{aligned}$$

where (5.22) follows from Equation (7.34). (7.35) follows from the fact that for 0-1 loss function $\ell(f(X),Y) + \ell(f(X),-Y) = 1$. (5.24) follows from the definition of $\beta_{\widehat{\pi}}$ defined in Equation (5.13). When $\ell_{\widehat{\pi}}$ is computed using weighted majority vote of the workers then (5.24) holds with $\beta_{\widehat{\pi}}$ replaced by

α. α is defined in (5.15).

Following shows the equality used in Equation (5.22). Using the notations $\rho_{\widehat{\pi}}$ and τ_{π}, in the following, for any function $f \in \mathcal{F}$, we compute the excess risk due to the unbiasedness of the modified loss function $\ell_{\widehat{\pi}}$.

$$\begin{aligned}
& R_{\ell,\mathcal{D}}(f) - R_{\ell_{\widehat{\pi}},\mathcal{D}_\pi}(f) \\
= \; & \mathbb{E}_{(X,Y)\sim\mathcal{D}}\left[\ell(f(X),Y)\right] - \mathbb{E}_{(X,Z^{(r)},w^{(r)})\sim\mathcal{D}_\pi}\left[\ell_{\widehat{\pi}}(f(X),Z^{(r)},w^{(r)})\right] \\
= \; & \mathbb{E}_{(X,Y)\sim\mathcal{D}}\left[\ell(f(X),Y)\right] \\
& -\mathbb{E}_{(X,Y,w^{(r)})\sim\mathcal{D}_\pi}\Bigg[\sum_{Z^{(r)}\in\{\pm1\}^r} \Big((1-\rho_{\widehat{\pi}}(-Y,Z^{(r)},w^{(r)}))\ell(f(X),Y) \qquad (5.25) \\
& +\rho_{\widehat{\pi}}(-Y,Z^{(r)},w^{(r)})\ell(f(X),-Y)\Big)\tau_\pi(Y,Z^{(r)},w^{(r)})\Bigg] \\
= \; & \mathbb{E}_{(X,Y)\sim\mathcal{D}}\left[\beta_{\widehat{\pi}}(Y)\left(\ell(f(X),Y)-\ell(f(X),-Y)\right)\right], \qquad (5.26)
\end{aligned}$$

where $\beta_{\widehat{\pi}}(Y)$ is defined in (5.12). Where (5.25) follows from the definition of $\ell_{\widehat{\pi}}$ given in Equation (5.4). Observe that when $\ell_{\widehat{\pi}}$ is computed using weighted majority vote of the workers then Equation (7.34) holds with $\beta_{\widehat{\pi}}(Y)$ replaced by $\alpha(y)$. $\alpha(y)$ is defined in (5.14).

5.5.2 Proof of Lemma 5.3

Recall that we have

$$\widehat{\pi}_{ks}^{(a)} = \frac{\sum_{i=1}^n\sum_{j=1}^r \mathbb{I}[w_{ij}=a]\mathbb{I}[t_i=k]\mathbb{I}[Z_{ij}=s]}{\sum_{i=1}^n\sum_{j=1}^r \mathbb{I}[w_{ij}=a]\mathbb{I}[t_i=k]} \qquad (5.27)$$

Let t_i denote $\widehat{f}(X_i)$. By the definition of risk, for any $k \in \mathcal{K}$, we have

$$\mathbb{P}\Big[\big|\mathbb{I}[Y_i=k]-\mathbb{I}[t_i=k]\big|=1\Big]=\delta\,.$$

Let $|\mathcal{K}| = K$. Define, for fixed $a \in [m]$, and $k, s \in \mathcal{K}$,

$$A \;:=\; \sum_{i=1}^{n}\sum_{j=1}^{r} \mathbb{I}[w_{ij} = a]\mathbb{I}[t_i = k]\mathbb{I}[Z_{ij} = s]\,, \qquad \bar{A} := \frac{nr\pi_{ks}}{mK} \tag{5.28}$$

$$B \;:=\; \sum_{i=1}^{n}\sum_{j=1}^{r} \mathbb{I}[w_{ij} = a]\mathbb{I}[t_i = k]\,, \qquad \bar{B} := \frac{nr}{mK} \tag{5.29}$$

$$C \;:=\; \sum_{i=1}^{n}\sum_{j=1}^{r} \mathbb{I}[w_{ij} = a]\Big|\mathbb{I}[Y_i = k] - \mathbb{I}[t_i = k]\Big|\,, \qquad \bar{C} := \frac{nr\delta}{m}\,, \tag{5.30}$$

$$D \;:=\; \sum_{i=1}^{n}\sum_{j=1}^{r} \mathbb{I}[w_{ij} = a]\mathbb{I}[Y_i = k]\mathbb{I}[Z_{ij} = s]\,, \tag{5.31}$$

$$E \;:=\; \sum_{i=1}^{n}\sum_{j=1}^{r} \mathbb{I}[w_{ij} = a]\mathbb{I}[Y_i = k]\,. \tag{5.32}$$

Note that A, B, C, D, E depend upon $a \in [m]$, $k, s \in \mathcal{K}$. However, for ease of notations, we have not included the subscripts. We have,

$$\begin{aligned}\left|\widehat{\pi}^{(a)}_{ks} - \pi^{(a)}_{ks}\right| \;=\; \frac{A - B\pi_{ks}}{B} \;&=\; \frac{|(A-\bar{A}) - (B-\bar{B})\pi_{ks}|}{|\bar{B} + (B-\bar{B})|} \\ &\le\; \frac{|A-\bar{A}| + |(B-\bar{B})|\pi_{ks}}{|\bar{B}| - |B-\bar{B}|}\end{aligned} \tag{5.33}$$

Now, we have,

$$\begin{aligned}|A-\bar{A}| \;&\le\; |A-D| \;+\; |D-\bar{A}| \\ &\le\; C \;+\; |D-\bar{A}|\,.\end{aligned} \tag{5.34}$$

We have,

$$\begin{aligned}|B-\bar{B}| \;&\le\; |B-E| \;+\; |E-\bar{B}| \\ &\le\; C \;+\; |E-\bar{B}|\end{aligned} \tag{5.35}$$

Observe that C is a sum of nr i.i.d. Bernoulli random variables with mean δ/m. Using Chernoff bound we get that

$$C \;\le\; \frac{nr\delta}{m} + \sqrt{\frac{3nr\delta\log(2mK/\delta_1)}{m}}\,, \tag{5.36}$$

for all $a \in [m]$, and $k \in \mathcal{K}$ with probability at least $1-\delta_1$. Similarly, D is a sum of nr i.i.d. Bernoulli random variables with mean $\pi_{ks}/(mk)$. Again, using Chernoff bound we get that

$$\left|D - \bar{A}\right| \;\leq\; \sqrt{\frac{3nr\pi_{ks}\log(2mK^2/\delta_1)}{mK}}\,, \tag{5.37}$$

for all $a \in [m]$, $k, s \in \mathcal{K}$ with probability at least $1-\delta_1$. From the bound on $|D - \bar{A}|$, it follows that

$$|E - \bar{B}| \;\leq\; \sqrt{\frac{3nr\log(2mK^2/\delta_1)}{m}} \tag{5.38}$$

Collecting Equations (5.33)-(5.38), we have for all $a \in [m]$, $k, s \in \mathcal{K}$

$$\left|\widehat{\pi}^{(a)}_{ks} - \pi^{(a)}_{ks}\right| \;\leq\; \frac{2\delta + 16\sqrt{m\log(2mK^2\delta_1/(nr)}}{1/K - \delta - 8\sqrt{m\log(2mK^2/\delta_1)/(nr)}}\,, \tag{5.39}$$

with probability at least $1-2\delta_1$.

CHAPTER 6

SPECTRUM ESTIMATION FROM A FEW ENTRIES

We want to estimate the Schatten k-norm of a positive semidefinite matrix $M \in \mathbb{R}^{d\times d}$ from a subset of its entries. The restriction to positive semidefinite matrices is primarily for notational convenience, and our analyses, the estimator, and the efficient algorithms naturally generalize to any non-square matrices. Namely, we can extend our framework to bipartite graphs and estimate Schatten k-norm of any matrix for any even k. Let Ω denote the set of indices of samples we are given and let $\mathcal{P}_\Omega(M) = \{(i,j,M_{ij})\}_{(i,j)\in\Omega}$ denote the samples. With a slight abuse of notation, we used $\mathcal{P}_\Omega(M)$ to also denote the $d \times d$ sampled matrix:

$$\mathcal{P}_\Omega(M)_{ij} = \begin{cases} M_{ij} & \text{if } (i,j) \in \Omega \;, \\ 0 & \text{otherwise} \;, \end{cases}$$

and it should be clear from the context which one we refer to. Although we propose a framework that generally applies to any probabilistic sampling, it is necessary to propose specific sampling scenarios to provide tight analyses on the performance. Hence, we focus on *Erdös-Rényi sampling*.

There is an extensive line of research in low-rank matrix completion problems [31, 110], which addresses a fundamental question of how many samples are required to *complete* a matrix (i.e. estimate all the missing entries) from a small subset of sampled entries. It is typically assumed that each entry of the matrix is sampled independently with a probability $p \in (0,1]$. We refer to this scenario as *Erdös-Rényi sampling*, as the resulting pattern of the samples encoded as a graph is distributed as an Erdös-Rényi random graph. The spectral properties of such an sampled matrix have been well studied in the literature [75, 1, 69, 110, 123]. In particular, it is known that the original matrix is close in spectral norm to the sampled one where the missing entries are filled in with zeros and properly rescaled under certain incoherence assumptions. This suggests using the singular values of the sampled and rescaled matrix $(d^2/|\Omega|)\mathcal{P}(M)$ directly for estimating the Schatten norms. However, in the sub-linear regime in which the number of samples $|\Omega| = d^2p$ is comparable to or significantly smaller than the degrees of freedom in representing a symmetric rank-r matrix, which is $dr - r^2$, the spectrum of the sampled matrix is significantly different from the spectrum of the original matrix as shown in Figure 6.1. We need to design novel estimators that are more sample efficient in the sub-linear regime where $d^2p \ll dr$.

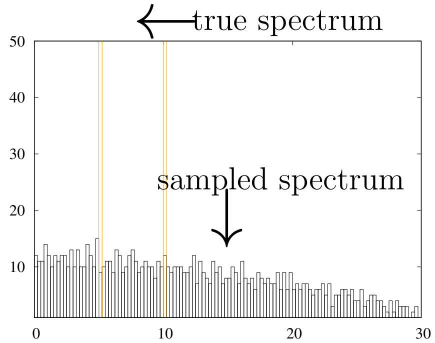


Figure 6.1: In yellow, we show the histogram of the singular values of a positive semi-definite matrix $M \in \mathbb{R}^{d\times d}$ of size $d = 1000$ with rank $r = 100$, with $\sigma_1 = \cdots = \sigma_{50} = 10$, $\sigma_{51} = \cdots = \sigma_{100} = 5$, and the rest at zero (we omit zero singular values in the plot for illustration). In comparison, we show in black the histogram of the singular values of the sampled matrix where each entry of M is sampled with probability $p = (1/d)r^{1-2/7}$ (properly rescaled by $1/p$ to best match the original spectrum).

6.1 Summary of the approach and preview of results

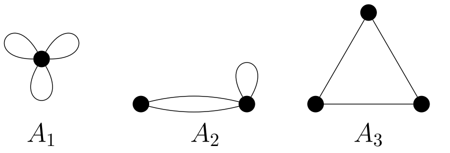
We propose first estimating one or a few Scahtten norms, which can be accurately estimated from samples, and using these estimated Schatten norms to approximate the spectral properties of interest: spectral sum functions and the spectrum. We use an alternative expression of the Schatten k-norm for positive semidefinite matrices as the trace of the k-th power of M, i.e. $(\|M\|_k)^k = \text{Tr}(M^k)$. This sum of the entries along the diagonal of M^k is the sum of total weights of all the closed walks of length k. Consider the entries of M as weights on a complete graph K_d over d nodes (with self-loops). A closed walk of length k is defined as a sequence of nodes $w = (w_1, w_2, \ldots, w_{k+1})$ with

$w_1 = w_{k+1}$, where we allow repeated nodes and repeated edges. The *weight* of a closed walk $w = (w_1, \dots, w_k, w_1)$ is defined as $\omega_M(w) \equiv \prod_{i=1}^{k} M_{w_i w_{i+1}}$, which is the product of the weights along the walk. It follows that

$$\|M\|_k^k \;=\; \sum_{w:\text{ all length } k \text{ closed walks}} \omega_M(w)\;. \tag{6.1}$$

Following the notations from enumeration of small simple cycles in a graph by [6], we partition this summation into those with the same pattern H that we call a *k-cyclic pseudograph*. Let $C_k = (V_k, E_k)$ denote the undirected simple cycle graph with k nodes, e.g. A_3 in Figure 7.1 is C_3. We expand the standard notion of simple k-cyclic graphs to include multiedges and loops, hence the name *pseudograph*.

Definition 6.1. We define an unlabelled and undirected pseudograph $H = (V_H, E_H)$ to be a *k-cyclic pseudograph* for $k \geq 3$ if there exists an onto node-mapping from $C_k = (V_k, E_k)$, i.e. $f : V_k \to V_H$, and a one-to-one edge-mapping $g : E_k \to E_H$ such that $g(e) = (f(u_e), f(v_e))$ for all $e = (u_e, v_e) \in E_k$. We use $\mathcal{H}_k$ to denote the set of all k-cyclic pseudographs. We use $c(H)$ to the number of different node mappings f from C_k to a k-cyclic pseudograph H.



$c(A_1) = 1 \quad c(A_2) = 3 \quad c(A_3) = 6$

Figure 6.2: The 3-cyclic pseudographs $\mathcal{H}_3 = \{A_1, A_2, A_3\}$.

In the above example, each member of $\mathcal{H}_3$ is a distinct pattern that can be mapped from C_3. For A_1, it is clear that there is only one mapping from C_3 to A_1 (i.e. $c(A_1) = 1$). For A_2, one can map any of the three nodes to the left-node of A_2, hence $c(A_2) = 3$. For A_3, any of the three nodes can be mapped to the bottom-left-node of A_3 and also one can map the rest of the nodes clockwise or counter-clockwise, resulting in $c(A_3) = 6$. For $k \leq 7$, all the k-cyclic pseudo graphs are given in the Section 6.9 (See Figures 7.2–6.17).

Each closed walk w of length k is associated with one of the graphs in $\mathcal{H}_k$, as there is a unique H that the walk is an Eulerian cycle of (under a one-to-one mapping of the nodes). We denote this graph by $H(w) \in \mathcal{H}_k$. Considering the weight of a walk $\omega_M(w)$, there are multiple distinct walks with the same weight. For example, a length-3 walk $w = (v_1, v_2, v_2, v_1)$ has $H(w) = A_2$ and there are 3 walks with the same weight $\omega(w) = (M_{v_1 v_2})^2 M_{v_2 v_2}$, i.e. (v_1, v_2, v_2, v_1), (v_2, v_2, v_1, v_2), and (v_2, v_1, v_2, v_2). This multiplicity of the weight depends only on the structure $H(w)$ of a walk, and it is exactly $c(H(w))$ the number of mappings from C_k to $H(w)$ in Definition 7.1. The total sum of the weights of closed walks of length k can be partitioned into their respective pattern, which will make computation of such terms more efficient (see Section 6.2) and also de-biasing straight forward (see Equation (7.12)):

$$\|M\|_k^k \;=\; \sum_{H \in \mathcal{H}_k} \omega_M(H)\, c(H)\,, \tag{6.2}$$

where with a slight abuse of a notation, we let $\omega_M(H)$ for $H \in \mathcal{H}_k$ be the sum of all *distinct* weights of walks w with $H(w) = H$, and $c(H)$ is the multiplicity of each of those distinct weights. This gives an alternative tool for computing the Schatten k-norm without explicitly computing the singular values.

Given only the access to a subset of sampled entries, one might be tempted to apply the above formula to the sampled matrix with an appropriate scaling, i.e.

$$\Big\| \frac{d^2}{|\Omega|} \mathcal{P}_\Omega(M) \Big\|_k^k \;=\; \frac{d^2}{|\Omega|} \sum_{H \in \mathcal{H}_k} \omega_{\mathcal{P}_\Omega(M)}(H)\, c(H)\,, \tag{6.3}$$

to estimate $\|M\|_k^k$. However, this is significantly biased. To eliminate the bias, we propose rescaling each term in (7.8) by the inverse of the probability of sampling that particular walk w (i.e. the probability that all edges in w are sampled). A crucial observation is that, for any sampling model that is invariant under a relabelling of the nodes, this probability only depends on the pattern $H(w)$. In particular, this is true for the Erdös-Rényi sampling. Based on this observation, we introduce a novel estimator that de-biases each group separately:

$$\widehat{\Theta}_k(\mathcal{P}_\Omega(M)) \;=\; \sum_{H \in \mathcal{H}_k} \frac{1}{p(H)}\, \omega_{\mathcal{P}_\Omega(M)}(H)\, c(H)\;. \tag{6.4}$$

where $p(H)$ is the probability a pattern H is sampled, i.e. all edges traversed in a walk w with $H(w) = H$ is sampled. It immediately follows that this estimator is unbiased, i.e. $\mathbb{E}_\Omega[\widehat{\Theta}_k(\mathcal{P}_\Omega(M))] = \|M\|_k^k$, where the randomness is in Ω. However, computing this estimate can be challenging. Naive enumeration over all closed walks of length k takes time scaling as $O(d\,\Delta^{k-1})$, where Δ is the maximum degree of the graph. Except for extremely sparse graphs, this is impractical. Inspired by the work of [6] in counting short cycles in a graph, we introduce a novel and efficient method for computing the proposed estimate for small values of k.

Proposition 6.2. *For a positive semidefinite matrix M and any sampling pattern Ω, the proposed estimate $\widehat{\Theta}_k(\mathcal{P}_\Omega(M))$ in* (7.12) *can be computed in time $O(d^\alpha)$ for $k \in \{3, 4, 5, 6, 7\}$, where $\alpha < 2.373$ is the exponent of matrix multiplication. For $k = 1$ or 2, $\widehat{\Theta}_k(\mathcal{P}_\Omega(M))$ can be computed in time $O(d)$ and $O(d^2)$, respectively.*

This bound holds regardless of the degree, and the complexity can be even smaller for sparse graphs as matrix multiplications are more efficient. We give a constructive proof by introducing a novel algorithm achieving this complexity in Section 6.2. For $k \geq 8$, our approach can potentially be extended, but the complexity of the problem fundamentally changes as it is at least as hard as counting K_4 in a graph, for which the best known run time is $O(d^{\alpha+1})$ for general graphs [117].

We make the following contributions in this paper:

- We give in (7.12) an unbiased estimator of the Schatten k-norm of a positive semidefinite matrix M, from a random sampling of its entries. In general, the complexity of computing the estimate scales as $O(d\Delta^{k-1})$ where Δ is the maximum degree (number of sampled entries in a column) in the sampled matrix. We propose a novel efficient algorithm for computing the estimate in (7.12) exactly for small $k \leq 7$, which involves only matrix operations. This algorithm is significantly more efficient and has run-time scaling as $O(d^\alpha)$ independent of the degree and for all $k \leq 7$ (see Proposition 6.2) .

- Under the typical Erdös-Rényi sampling, we show that the Schatten k-norm of an incoherent rank-r matrix can be approximated within any constant multiplicative error, with number of samples scaling as

$O(dr^{1-2/k})$ (see Theorem 6.3). In particular, this is strictly smaller than the number of samples necessary to complete the matrix, which scales as $O(dr \log d)$. Below this matrix completion threshold, numerical experiments confirm that the proposed estimator significantly outperforms simple heuristics of using singular values of the sampled matrices directly or applying state-of-the-art matrix completion methods (see Figure 6.4).

- Given estimation of first K Schatten norms, it is straight forward to approximate spectral sum functions of the form (6.5) using Chebyshev's expansion, and also estimate the spectrum itself using moment matching in Wasserstein distance. We apply our Schatten norm estimates to the applications of estimating the generalized rank studied in [217] and estimating the spectrum studied in [119]. We provide performance guarantees for both applications and provide experimental results suggesting we improve upon other competing methods.

- We propose a new sampling model, which we call *graph sampling*, that preserves the structural properties of the pattern of the samples. We identify a fundamental property of the structure of the pattern ($\lambda^*_{G,r}$ in Eq.(6.27)) that captures the difficulty of estimating the Schatten k-norm from such graph sampling (see Theorem 6.11). Under this graph sampling, we show that there are sampling patterns that are significantly more efficient, for estimating the spectral properties, than Erdös-Rényi sampling.

In the remainder of this section, we review existing work in Schatten norm approximation, and provide an efficient implementation of the estimator (7.12) for small k in Section 6.2. In Section 6.3, we provide a theoretical analysis of our estimator under the Erdös-Rényi sampling scenario. In Section 6.4, we provide a theoretical analysis under the graph sampling scenario. We conclude with a discussion on interesting observations and remaining challenges in Section 6.5.

6.1.1 Related work

We review existing methods in approximating the Schatten norms, counting small structures in graphs, and various applications of Schatten norms.

Estimating k-Schatten norms of a data matrix. The proposed Schatten norm estimator can be used as a black box in various applications where we want to test the property of a data matrix or a network but limited to observe only a small portion of the data. These include, for example, network forensics, matrix spectral property testing, and testing for graph isospectral properties. Relatively little is known under the matrix completion setting studied in this paper. However, Schatten norm estimation under different resource constrained scenarios have been studied. [94] propose a randomized algorithm for approximating the trace of any large matrix, where the constraint is on the computational complexity. The goal is to design a random rank-one linear mapping such that the trace is preserved in expectation and the variance is small [11, 175]. [127] propose an optimal bilinear sketching of a data matrix, where the constraint is on the memory, i.e. the size of the resulting sketch. The goal is to design a sketch of a data matrix M using minimal storage and a corresponding approximate reconstruction method for $\|M\|_k^k$. [128] propose an optimal streaming algorithm where only one-pass on the data is allowed in a data stream model and the constraint is on the space complexity of the algorithm. The goal is to design a streaming algorithm using minimal space to estimate $\|M\|_k^k$. [217] propose an estimator under a distributed setting where columns of the data are store in distributed storage and the constraint is on the communication complexity. The goal is to design a distributed protocol minimizing the communication to estimate $\|M\|_k^k$. Given a random vector X, [119] propose an optimal estimator for the Schatten k-norm of the covariance matrix, where the constraint is on the number of samples n. The goal is to design an estimator using minimum number of samples to estimate $\|\mathbb{E}[XX^T]\|_k^k$.

One of our contribution is that we propose an efficient algorithm for computing the weighted counts of small structures in Section 6.2, which can significantly improve upon less sample-efficient counterpart in, for example, [119]. Under the setting of [119] (and also [127]), the main idea of the estimator is that the weight of each length-k cycle in the observed empirical covariance matrix $(1/n)\sum_{i=1}^{n} X_i X_i^T$ provides an unbiased estimator of

$\|\mathbb{E}[XX^T]\|_k^k$. One prefers to sum over the weights of as many cycles as computationally allowed in order to reduce the variance. As counting all cycles is in general computationally hard, they propose counting only increasing cycles (which only accounts for only 1/k! fraction of all the cycles), which can be computed in time $O(d^\alpha)$. If one has an efficient method to count all the (weighted) cycles, then the variance of the estimator could potentially decrease by an order of $k!$. For $k \leq 7$, our proposed algorithm in Section 6.2 provides exactly such an estimator. We replace [119, Algorithm 1] with ours, and run the same experiment to showcase the improvement in Figure 6.3, for dimension $d = 2048$ and various values of number of samples n comparing the multiplicative error in estimating $\|\mathbb{E}[XX^T]\|_k^k$, for $k = 7$. With the same run-time, significant gain is achieved by simply substituting our proposed algorithm for counting small structures, in the sub-routine. In general, the efficient algorithm we propose might be of independent interest to various applications, and can directly substitute (and significantly improve upon) other popular but less efficient counterparts.

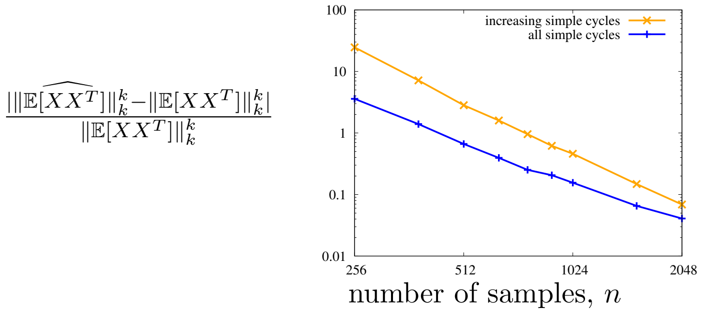


Figure 6.3: By replacing Algorithm 1 in [119] that only counts increasing cycles with our proposed algorithm that counts all cycles, significant gain is acheived in estimating $\|\mathbb{E}[XX^T]\|_k^k$, for $k = 7$.

One of the main challenges under the sampling scenario considered in this paper is that existing counting methods like that of [119] cannot be applied, regardless of how much computational power we have. Under the matrix completion scenario, we need to (a) sum over all small structures $H \in \mathcal{H}_k$ and not just C_k as in [119]; and (b) for each structure we need to sum over all subgraphs with the same structure and not just those walks whose labels form a monotonically increasing sequence as in [119].

Algorithms for counting structures. An important problem in graph theory is to count the number of small structures, also called network motifs, in a given graph. This has many practical applications in designing good LDPC codes [196], understanding the properties social networks [198], and explaining gene regulation networks [189]. Exact and approximate algorithms have been proposed in [6, 117, 133, 85, 109, 202]. The most relevant one is the work of [6] on counting the number of cycles C_k, where counts of various small structures called k-cyclic graphs are used as sub-routines and efficient approaches are proposed for $k \leq 7$. These are similar to k-cyclic pseudographs, but with multiedges condensed to a single edge. When counting cycles in a simple (unweighted) graph, k-cyclic graphs are sufficient as all the edges have weight one. Hence, one does not need to track how many times an edge has been traversed; the weight of that walk is one, regardless. In our setting, the weight of a walk depends on how many times an edge has been traversed, which we track using multiedges. It is therefore crucial to introduce the class of k-cyclic pseudographs in our estimator.

In a distributed environment, fast algorithms for counting small structures have been proposed by [61] and [62] for small values of $k \in \{3, 4\}$. However, the main strength of this approach is in distributed computing, and under the typical centralized setting we study, this approach can be slower by a factor exponential in k for, say $k \leq 7$.

From Schatten norms to spectral sum functions. A dominant application of Schatten norms is in approximating a family of functions of a matrix, which are called *spectral sum functions* [87] of the form

$$F(M; f) \;\equiv\; \sum_{i=1}^{d} f(\sigma_i(M)) \;\simeq\; \sum_{k=0}^{K} a_k \Big\{ \sum_{i=1}^{d} \sigma_i(M)^k \Big\} . \tag{6.5}$$

A typical approach is to compute the coefficients of a Chebyshev approximation of f, which immediately leads to an approximation of the spectral sum function of interest as the weighted sum of Schatten k-norms. This follows from the approximation of $f(x) \simeq \sum_{k=0}^{K} a_k x^k$. This approach has been widely used in fast methods for approximating the log-determinant [163, 216, 26, 10, 88], corresponding to $f(x) = \log x$. Practically, log-determinant computations are routinely (approximately) required in applications including Gaussian graphical models [176], minimum-volume ellipsoids [199], and metric learning [45]. Fast methods for approximating trace of matrix inverse has been studied in [211, 37], corresponding to $f(x) = x^{-1}$, motivated by applications in lattice quantum chromodynamics [194]. Fast methods for approximating the Estarada index has been studied in [87], corresponding to $f(x) = \exp(x)$. Practically, it is used in characterizing 3-dimensional molecular structure [64] and measuring graph centrality [65], the entropy of a graph [33], and the bipartivity of a graph [66]. Approximating the generalized rank under communication constraints has been studied in [217], corresponding to $f(x; c_1) = \mathbb{I}(x \leq c_1)$. The generalized rank approximates a necessary tuning parameter in a number of problems where low-rank solutions are sought including robust PCA [32, 159] and matrix completion [111, 110, 97], and also is required in sampling based methods in numerical analysis [142, 86]. Similarly, [177] studied the number of singular values in an interval, corresponding to $f(x; c_1, c_2) = \mathbb{I}(c_1 \leq x \leq c_2)$. In practice, a number of eigensolvers [167, 178, 179] require the number of eigenvalues in an given interval. For more comprehensive list of references and applications of this framework, we refer to the related work section in [87].

In a recent work, [119] provide a novel approach to tackle the challenging problem of estimating the singular values themselves. Considering the histogram of the singular values as a one-dimensional distribution and the Schatten k-norm as the k-th moment of this distribution, the authors provide an innovative algorithm to estimate the histogram that best matches the moments in Wasserstein distance.

Matrix completion. Low-rank matrix completion addresses the problem of recovering a low-rank matrix from its sampled entries. Tight lower and upper bounds on the sample complexity is well studied in both cases where you want exact recovery when samples are noiseless [31, 110, 20], and also when samples are noisy and where you want approximate recovery [111, 158]. In practical applications, one might not have enough samples to estimate all the missing entries with sufficient accuracy. However, one might still be able to infer important spectral properties of the data, such as the singular values or the rank. Such spectral properties can also assist in making decisions on how many more samples to collect in order to make accurate inference on the quantity of interest. In this paper, one of the fundamental question we ask and answer affirmatively is: Can we accurately recover the spectral properties of a low-rank matrix from sampling of its entries, below the matrix completion threshold?

6.2 Efficient Algorithm

In this section we give a constructive proof of Proposition 6.2, inspired by the seminal work of [6] and generalize their counting algorithm for k-cyclic graphs for counting (weighted) k-cyclic pseudographs. In computing the estimate in (7.12), $c(H)$ can be computed in time $O(k!)$ and suppose $p(H)$ has been computed (we will explain how to compute $p(H)$ for Erös-Rényi sampling and graph sampling in Sections 6.3 and 6.4). The bottleneck then is computing the weights $\omega_{\mathcal{P}_\Omega(M)}(H)$ for each $H \in \mathcal{H}_k$. Let $\gamma_M(H) \equiv \omega_M(H)c(H)$. We give matrix multiplication based equations to compute $\gamma_M(H)$ for every $H \in \mathcal{H}_k$ for $k \in \{3,4,5,6,7\}$. This establishes that $\gamma_M(H)$, and hence $\omega_M(H)$, can be computed in time $O(d^\alpha)$, proving Proposition 6.2.

For any matrix $A \in \mathbb{R}^{d\times d}$, let $\text{diag}(A)$ to be a diagonal matrix such that $(\text{diag}(A))_{ii} = A_{ii}$, for all $i \in [d]$ and $(\text{diag}(A))_{i,j} = 0$, for all $i \neq j \in [d]$. For a given matrix $M \in \mathbb{R}^{d\times d}$, define the following: O_M to be matrix of off-diagonal entries of M that is $O_M \equiv M - \text{diag}(M)$ and we let $D_M \equiv \text{diag}(M)$. Let $\text{tr}(A)$ denote trace of A, that is $\text{tr}(A) = \sum_{i\in[d]} A_{ii}$, and let $A{*}B$ denote the standard matrix multiplication of two matrices A and B to make it more explicit. Consider computing $\gamma_M(H)$ for $H \in \mathcal{H}_3$ as labeled in Figure 7.1:

$$\gamma_M(A_1) \quad = \quad \text{tr}(D_M{*}D_M{*}D_M) \tag{6.6}$$

$$\gamma_M(A_2) \quad = \quad 3\,\text{tr}(D_M{*}O_M{*}O_M) \tag{6.7}$$

$$\gamma_M(A_3) \quad = \quad \text{tr}(O_M{*}O_M{*}O_M) \tag{6.8}$$

The first weighted sum $\gamma_M(A_1)$ is sum of all weights of walks of length 3 that consists of three self-loops. One can show that $\gamma_M(A_1) = \sum_{i\in[d]} M_{ii}^3$, which in our matrix operation notations is (6.6). Similarly, $\gamma_M(A_3)$ is the sum of weights of length 3 walks with no self-loop, which leads to (6.8). $\gamma_M(A_2)$ is the sum of weights of length 3 walks with a single self-loop, which leads to (6.7). The factor 3 accounts for the fact that the self loop could have been placed at first, second, or third in the walk.

Similarly, for each k-cyclic pseudographs in $\mathcal{H}_k$ for $k \leq 7$, computing $\gamma_M(H)$ involves a few matrix operations with run-time $O(d^\alpha)$. We provide the complete set of explicit expressions in Section 6.10. A MATLAB implementation of the estimator (7.12), that includes as its sub-routines the computation of the weights of all k-cyclic pseudographs, is available for download at `https://github.com/khetan2/Schatten_norm_estimation`. The explicit formulae in Section 6.10 together with the implementation in the above url might be of interest to other problems involving counting small structures in graphs.

For $k = 1$, the estimator simplifies to $\widehat{\Theta}_k(\mathcal{P}_\Omega(M)) = (1/p)\sum_i \mathcal{P}_\Omega(M)_{ii}$, which can be computed in time $O(d)$. For $k = 2$, the estimator simplifies to $\widehat{\Theta}_k(\mathcal{P}_\Omega(M)) = (1/p)\sum_{i,j} \mathcal{P}_\Omega(M)_{ij}^2$, which can be computed in time $O(|\Omega|)$. However, for $k \geq 8$, there exists walks over K_4, a clique over 4 nodes, that cannot be decomposed into simple computations involving matrix operations. The best known algorithm for a simpler task of counting K_4 has run-time scaling as $O(d^{\alpha+1})$, which is fundamentally different. We refer to Section 6.5 for further discussions on the computational complexity beyond $k = 7$.

Algorithm 8 Schatten k-norm estimator

```
Input: $\mathcal{P}_\Omega(M)$, $k$, $\mathcal{H}_k$, $p(H)$ for all $H \in \mathcal{H}_k$
Output: $\widehat{\Theta}_k(\mathcal{P}_\Omega(M))$
if $k \leq 7$ then
    For each $H \in \mathcal{H}_k$, compute $\gamma_{\mathcal{P}_\Omega(M)}(H)$ using the formula from Eq. (6.6)–(6.8) for $k = 3$ and Eq. (6.90) – (??) for $k \in \{4, 5, 6, 7\}$
    $\widehat{\Theta}_k(\mathcal{P}_\Omega(M)) \leftarrow \sum_{H \in \mathcal{H}_k} \frac{1}{p(H)} \gamma_{\mathcal{P}_\Omega(M)}(H)$
else
    $\widehat{\Theta}_k(\mathcal{P}_\Omega(M)) \leftarrow$ Algorithm 11[$\mathcal{P}_\Omega(M)$, $k$, $\mathcal{H}_k$, $p(H)$ for all $H \in \mathcal{H}_k$] [Section 6.6]
end if
```

6.3 Erdös-Rényi sampling

Under the stylized but canonical Erdös-Rényi sampling, notice that the probability $p(H)$ that we observe all edges in a walk with pattern H is

$$p(H) \;=\; p^{m(H)}\,, \qquad (6.9)$$

where p is the probability an edge is sampled and $m(H)$ is the number of distinct edges in a k-cyclic pseudograph H. Plugging in this value of $p(H)$, which can be computed in time linear in k, into the estimator (7.12), we get an estimate customized for Erdös-Rényi sampling.

Given a rank-r matrix M, the difficulty of estimating properties of M from sampled entries is captured by the *incoherence* of the original matrix M, which we denote by $\mu(M) \in \mathbb{R}$ [31]. Formally, let $M \equiv U\Sigma U^\top$ be the singular value decomposition of a positive definite matrix where U is a $d \times r$ orthonormal matrix and $\Sigma \equiv \text{diag}(\sigma_1, \cdots, \sigma_r)$ with singular values $\sigma_1 \geq \sigma_2 \geq \cdots \geq \sigma_r > 0$. Let $U_{i,r}$ denote the i-th row and j-th column entry of matrix U. The incoherence $\mu(M)$ is defined as the smallest positive value μ such that the following holds:

A1. For all $i \in [d]$, we have $\sum_{a=1}^{r} U_{ia}^2(\sigma_a/\sigma_1) \leq \mu r/d$.

A2. For all $i \neq j \in [d]$, we have $|\sum_{a=1}^{r} U_{ia}U_{ja}(\sigma_a/\sigma_1)| \leq \mu\sqrt{r}/d$.

The incoherence measures how well spread out the matrix is and is a common measure of difficulty in completing a matrix from random samples [31, 110]. The lower the incoherence, the more spread out the entries are, and estimation is easier. On the other hand, if there a a few entries that are much larger than the rest, estimating a property of the matrix (such as the Schatten k-norm) from sampled entries can be extremely challenging.

6.3.1 Performance guarantee

For any $d \times d$ positive semidefinite matrix M of rank r with incoherence $\mu(M) = \mu$ and the effective condition number $\kappa = \sigma_{\max}(M)/\sigma_{\min}(M)$, we define

$$\rho^2 \quad \equiv \quad (\kappa\mu)^{2k} g(k) \max\left\{1, \frac{(dp)^{k-1}}{d}, \frac{r^k p^{k-1}}{d^{k-1}}\right\}, \tag{6.10}$$

such that the variance of our estimator is bounded by $\text{Var}(\widehat{\Theta}(\mathcal{P}_\Omega(M))/\|M\|_k^k) \leq \rho^2(r^{1-2/k}/dp)^k$ as we show in the proof of Theorem 6.3 in Section 6.8.1. Here, $g(k) = O(k!)$ is a function depending only on k.

Theorem 6.3 (Upper bound under the Erdös-Rényi sampling)**.** *For any integer $k \in [3, \infty)$, any $\delta > 0$, any rank-r positive semidefinite matrix*

$M \in \mathbb{R}^{d\times d}$, and given i.i.d. samples of the entries of M with probability p, the proposed estimate of (7.12) *achieves normalized error δ with probability bounded by*

$$\mathbb{P}\left(\frac{\left|\widehat{\Theta}_k(\mathcal{P}_\Omega(M)) - \|M\|_k^k\right|}{\|M\|_k^k} \;\geq\; \delta\right) \;\leq\; \frac{\rho^2}{\delta^2}\Big(\frac{r^{1-2/k}}{dp}\Big)^k \,. \qquad (6.11)$$

Consider a typical scenario where μ, κ, and k are finite with respect to d and r. Then the Chebyshev's bound in (6.11) implies that the sample $d^2p = O(dr^{1-2/k})$ is sufficient to recover $\|M\|_k^k$ up to arbitrarily small multiplicative error and arbitrarily small (but strictly positive) error probability. This is strictly less than the known minimax sample complexity for recovering the entire low-rank matrix, which scales is $\Theta(rd\log d)$. As we seek to estimate only a property of the matrix (i.e. the Schatten k-norm) and not the whole matrix itself, we can be more efficient on the sample complexity by a factor of $r^{2/k}$ in rank and a factor of $\log d$ in the dimension. We emphasize here that such a gain can only be established using the proposed estimator based on the structure of the k-cyclic pseudographs. We will show empirically that the standard matrix completion approaches fail in the critical regime of samples below the recovery threshold of $O(rd\log d)$.

Figure 6.4 is a scatter plot of the absolute relative error in estimated Schatten k-norm, $\big|\|M\|_k^k - \widehat{\|M\|_k^k}\big|/\|M\|_k^k$, for $k = 5$, for three approaches: the proposed estimator, Schatten norm of the scaled sampled matrix (after rank-r projection), and Schatten norm of the completed matrix, using state-of-the-art alternating minimization algorithm [97]. All the three estimators are evaluated 20 times for each value of p. M is a symmetric positive semi-definite matrix of size $d = 500$, and rank $r = 100$ (left panel) and $r = 500$ (right panel). Singular vectors U of $M = U\Sigma U^\top$, are generated by QR decomposition of $\mathcal{N}(0, \mathbb{I}_{d\times d})$ and $\Sigma_{i,i}$ is uniformly distributed over $[1, 2]$. For a low rank matrix on the left, there is a clear critical value of $p \simeq 0.45$, above which matrix completion is exact with high probability. However, this algorithm knows the underlying rank and crucially exploits the fact that the underlying matrix is exactly low-rank. In comparison, our approach is agnostic to the low-rank assumption but finds the accurate estimate that is adaptive to the actual rank in a data-driven manner. Using the first r sin-

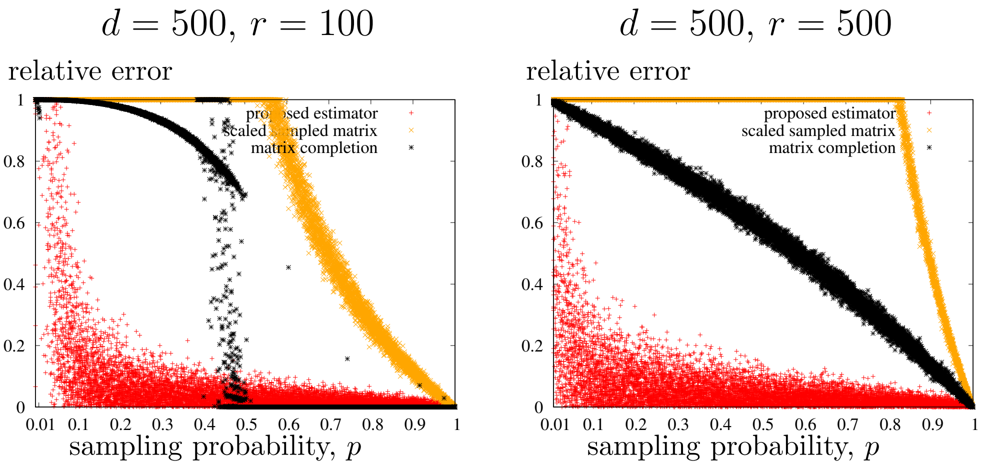


Figure 6.4: The proposed estimator outperforms both baseline approaches below the matrix completion threshold. For $k = 5$, comparison of the absolute relative error in estimated Schatten norm that is $\big|\|M\|_k^k - \widehat{\|M\|_k^k}\big|/\|M\|_k^k$ for the three algorithms: (1) the proposed estimator, $\widehat{\|M\|_k^k} = \widehat{\Theta}_k(\mathcal{P}_\Omega(M))$, (2) Schatten norm of the scaled sampled matrix, $\widehat{\|M\|_k^k} = \|(1/p)\mathcal{P}_r(\mathcal{P}_\Omega(M))\|_k^k$, (3) Schatten norm of the completed matrix, $\widetilde{M} = \mathrm{AltMin}(\mathcal{P}_\Omega(M))$ from [97], $\widehat{\|M\|_k^k} = \|\widetilde{M}\|_k^k$, where $\mathcal{P}_r(\cdot)$ is the standard best rank-r projection of a matrix. Ω is generated by Erdös-Rényi sampling of matrix M with probability p.

gular values of the (rescaled) sampled matrix fails miserably for all regimes (we truncate the error at one for illustration purposes). In this paper, we are interested in the regime where exact matrix completion is impossible as we do not have enough samples to exactly recover the underlying matrix: $p \leq 0.45$ in the left panel and all regimes in the right panel.

The sufficient condition of $d^2 p = O(dr^{1-2/k})$ in Theorem 6.3 holds for a broad range of parameters where the rank is sufficiently small $r = O(d^{k/((k-1)(k-2))})$ (to ensure that the first term in ρ^2 dominates). However, the following results in Figure 6.5 on numerical experiments suggest that our analysis holds more generally for all regimes of the rank r, even those close to d. M is generated using settings similar to that of Figure 6.4. Empirical probabilities are computed by averaging over 100 instances.

One might hope to tighten the Chebyshev bound by exploiting the fact that the correlation among the summands in our estimator (7.12) is weak. This can be made precise using recent result from [180], where a Bernstein-type bound was proved for sum of polynomials of independent random vari-

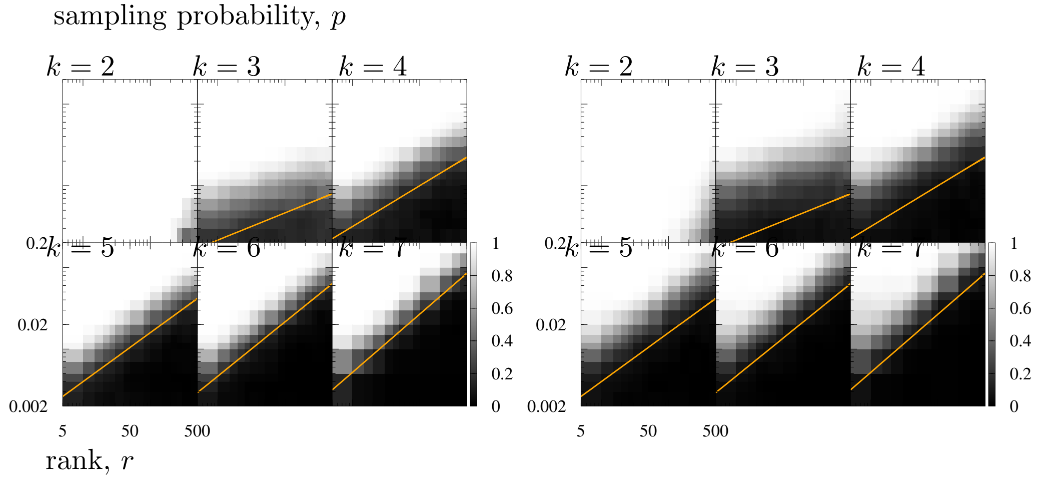


Figure 6.5: Each colormap in each block for $k \in \{2, 3, 4, 5, 6, 7\}$ show empirical probability of the event $\{\big|\|M\|_k^k - \widehat{\Theta}_k(\mathcal{P}_\Omega(M))\big|/\|M\|_k^k \le \delta\}$, for $\delta = 0.5$ (left panel) and $\delta = 0.2$ (right panel). Ω is generated by Erdös-Rényi sampling of matrix M with probability p (vertical axis). M is a symmetric positive semi-definite matrix of size $d = 1000$. The solid lines correspond to our theoretical prediction $p = (1/d)r^{1-2/k}$.

ables that are weakly correlated. The first term in the bound (6.12) is the natural Bernstein-type bound corresponding to the Chebyshev's bound in (6.11). However, under the regime where k is large or p is large, the correlation among the summands become stronger, and the second and third term in the bound (6.12) starts to dominate. In the typical regime of interest where μ, κ, k are finite, $d^2p = O(dr^{1-2/k})$, and sufficiently small rank $r = O(d^{k/((k-1)(k-2))})$, the error probability is dominated by the first term in the right-hand side of (6.12). Neither one of the two bounds in (6.11) and (6.12) dominates the other, and depending on the values of the problem parameters, we might want to apply the one that is tighter. We provide a proof in Section 6.8.2.

Theorem 6.4. *Under the hypotheses of Theorem 6.3, the error probability is upper bounded by*

$$\mathbb{P}\left(\frac{\big|\widehat{\Theta}_k(\mathcal{P}_\Omega(M)) - \|M\|_k^k\big|}{\|M\|_k^k} \ge \delta\right)$$

$$\le e^2 \max\left\{ e^{-\frac{\delta^2}{\rho^2}\left(\frac{dp}{r^{1-2/k}}\right)^k}, e^{-(dp)\left(\frac{\delta d}{\rho r^{k-1}}\right)^{1/k}}, e^{-(dp)\left(\frac{\delta d}{\rho r^{k-1}}\right)}, e^{-\frac{\delta dp}{\rho}} \right\}. \qquad (6.12)$$

These two results show that the sample size of $d^2p = O(dr^{1-2/k})$ is sufficient to estimate a Schatten k-norm accurately. In general, we do not expect to get a universal upper bound that is significantly tighter for all r, because for a special case of $r = d$, the following corollary of [127, Theorem 3.2] provides a lower bound; it is necessary to have sample size $d^2p = O(d^{2-4/k})$ when $r = d$. Hence, the gap is at most a factor of $r^{2/k}$ in the sample complexity.

Corollary 6.5. *Consider any linear observation $X \in \mathbb{R}^n$ of a matrix $M \in \mathbb{R}^{d\times d}$ and any estimate $\theta(X)$ satisfying $(1-\delta_k)\|M\|_k^k \le \theta(X) \le (1+\delta_k)\|M\|_k^k$ for any M with probability at least $3/4$, where $\delta_k = (1.2^k-1)/(1.2^k+1)$. Then, $n = \Omega(d^{2-4/k})$.*

For $k \in \{1, 2\}$, precise bounds can be obtained with simpler analyses. In particular, we have the following remarks, whose proof follows immediately by applying Chebyshev's inequality and Bernstien's inequality along with the incoherence assumptions.

Remark 6.6. *For $k = 1$, the probability of error in (6.11) is upper bounded by $\min\{\nu_1, \nu_2\}$, where*

$$\nu_1 \equiv \frac{1}{\delta^2}\frac{(\kappa\mu)^2}{dp} \ , \quad \text{and} \quad \nu_2 \equiv 2\exp\Big(\frac{-\delta^2}{2}\Big(\frac{(\kappa\mu)^2}{dp} + \delta\frac{(\kappa\mu)}{3dp}\Big)^{-1}\Big) \ .$$

Remark 6.7. *For $k = 2$, the probability of error in (6.11) is upper bounded by $\min\{\nu_1, \nu_2\}$, where*

$$\nu_1 \equiv \frac{1}{\delta^2}\frac{(\kappa\mu)^4}{d^2p}\big(2 + \frac{r^2}{d}\big) \ , \text{and}$$

$$\nu_2 \equiv 2\exp\Big(-\frac{\delta^2}{2}\Big(\frac{(\kappa\mu)^4}{d^2p}\big(2 + \frac{r^2}{d}\big) + \delta\frac{(\kappa\mu)^2 r}{3d^2p}\Big)^{-1}\Big) \ .$$

When $k = 2$, for rank small $r \le C\sqrt{d}$, only we only need $d^2p = O(1)$ samples for recovery up to any arbitrary small multiplicative error. When rank r is large, our estimator requires $d^2p = O(d)$ for both $k \in \{1, 2\}$.

6.3.2 From Schatten norms to spectrum and spectral sum functions

Schatten norms by themselves are rarely of practical interest in real applications, but they provide a popular means to approximate functions of singular values, which are often of great practical interest [53, 217, 119]. In this section, we consider two such applications using the first few Schatten norms explicitly: estimating the generalized rank in Section 6.3.2 and estimating the singular values in Section 6.3.2.

Estimating the generalized rank

For a matrix $M \in \mathbb{R}^{d \times d}$ and a given constant $c \geq 0$, its *generalized rank* of order c is given by

$$\text{rank}(M, c) = \sum_{i=1}^{d} \mathbb{I}\big[\sigma_i(M) > c\big] \,. \tag{6.13}$$

This recovers the standard rank as a special case when $c = 0$. Without loss of generality, we assume that $\sigma_{\max}(M) \leq 1$. For any given $0 \leq c_2 < c_1 \leq 1$, and $\delta \in [0, 1)$, our goal is to get an estimate $\widehat{r}(\mathcal{P}_\Omega(M))$ from sampled entries $\mathcal{P}_\Omega(M)$ such that

$$(1 - \delta)\,\text{rank}(M, c_1) \;\;\leq\;\; \widehat{r}(\mathcal{P}_\Omega(M)) \;\;\leq\;\; (1 + \delta)\,\text{rank}(M, c_2)\,. \tag{6.14}$$

The reason we take two different constants c_1, c_2 is to handle the ambiguous case when the matrix M has many eigenvalues smaller but very close to c_1. If we were to set $c_2 = c_1$, then any estimator $\widehat{r}(M)$ would be strictly prohibited from counting these eigenvalues. However, since these eigenvalues are so close to the threshold, distinguishing them from other eigenvalues just above the threshold is difficult. Setting $c_2 < c_1$ allows us to avoid this difficulty and focus on the more fundamental challenges of the problem.

Consider the function $H_{c_1,c_2} : \mathbb{R} \to [0, 1]$ given by

$$H_{c_1,c_2}(x) = \begin{cases} 1 & \text{if } x > c_1 \\ 0 & \text{if } x < c_2 \\ \frac{x - c_2}{c_1 - c_2} & \text{otherwise.} \end{cases} \tag{6.15}$$

It is a piecewise linear approximation of a step function and satisfies the following:

$$\mathrm{rank}(M,c_1) \;\; \leq \;\; \textstyle\sum_{i=1}^{d} H_{c_1,c_2}(\sigma_i(M)) \;\; \leq \;\; \mathrm{rank}(M,c_2)\,. \qquad (6.16)$$

We exploit this sandwich relation and estimate the generalized rank. Given a polynomial function $f:\mathbb{R}\to\mathbb{R}$ of finite degree m such that $f(x)\approx H_{c_1,c_2}(x)$ for all x, such that $f(x)=a_0+a_1x+\cdots+a_mx^m$, we immediately have the following relation, which extends to a function on the cone of PSD matrices in the standard way:

$$\sum_{i=1}^{d} f(\sigma_i(M)) \;\; = \;\; a_0 d + \sum_{k=1}^{m} a_k \|M\|_k^k\,. \qquad (6.17)$$

Using this equality, we propose the estimator:

$$\widehat{r}(\mathcal{P}_\Omega(M);c_1,c_2) \;\; \equiv \;\; a_0 d + \sum_{k=1}^{m} a_k \widehat{\Theta}_k(\mathcal{P}_\Omega(M))\,, \qquad (6.18)$$

where we use the first several $\widehat{\Theta}_k(\mathcal{P}_\Omega(M))$'s obtained by the estimator (7.12). Note that function f depends upon c_1, c_2. The remaining task is to obtain the coefficients of the polynomials in f that is a suitable approximation of the function H_{c_1,c_2}. In a similar context of estimating the generalized rank from approximate Schatten norms, [217] propose to use a composite function $f=q_s\circ q$ where q is a finite-degree Chebyshev polynomial of the first kind such that $\sup_{x\in[0,1]}|q(x)-H_{c_1,c_2}(x)|\leq 0.1$, and q_s is a polynomial of degree $2s+1$ given by

$$q_s(x) \;\; = \;\; \frac{1}{B(s+1,s+1)}\int_0^x t^s(1-t)^s dt\,, \qquad (6.19)$$

where $B(\cdot,\cdot)$ is the Beta function. Note that, since H_{c_1,c_2} is a continuous function with bounded variation, classical theory in [143, Theorem 5.7], guarantees existence of the Chebyshev polynomial q of a finite constant degree, say C_b, that depends upon c_1 and c_2. Concretely, for a given choice of thresholds $0\leq c_1<c_2\leq 1$ and degree of the beta approximation s, the estimator $\widehat{r}(\mathcal{P}_\Omega(M);c_1,c_2)$ in (6.18) can be computed as follows.

The approximation of H_{c_1,c_2} with $f = q_s\circ q$ and our upper bound on

Algorithm 9 Generalized rank estimator (a variation of [217])

```
Input: $\mathcal{P}_\Omega(M)$, $c_1$, $c_2$, $s$
Output: $\widehat{r}(\mathcal{P}_\Omega(M); c_1, c_2)$
For given $c_1$ and $c_2$, find a Chebyshev polynomial of the first kind $q(x)$ satisfying
    $$\sup_{x\in[0,1]} |q(x) - H_{c_1,c_2}(x)| < 0.1$$
                                                        [Algorithm 6.7]
Let $C_b$ denote the degree of $q(x)$
Find the degree $(2s+1)C_b$ polynomial expansion of $q_s \circ q(x) = \sum_{k=0}^{(2s+1)C_b} a_k x^k$
$\widehat{r}(\mathcal{P}_\Omega(M); c_1, c_2) \leftarrow a_0 d + \sum_{k=1}^{(2s+1)C_b} a_k \widehat{\Theta}_k(\mathcal{P}_\Omega(M))$     [Algorithm 8]
```

estimated Schatten norms $\widehat{\Theta}_k(\mathcal{P}_\Omega(M))$ translate into the following guarantee on generalized rank estimator $\widehat{r}(\mathcal{P}_\Omega(M); c_1, c_2)$ given in (6.18).

Corollary 6.8. *Suppose $\|M\|_2 \le 1$. Under the hypotheses of Theorem 6.3, for any given $1 \ge c_1 > c_2 \ge 0$, there exists a constant C_b, such that for any $s \ge 0$ and any $\gamma > 0$, the estimate in (6.18) with the choice of $f = q_s \circ q$ satisfies*

$$\begin{aligned}(1-\delta)(\operatorname{rank}(M, c_1) - 2^{-s}d) &\le \widehat{r}(\mathcal{P}_\Omega(M); c_1, c_2) \\ &\le (1+\delta)(\operatorname{rank}(M, c_2) + 2^{-s}d)\,, \quad (6.20)\end{aligned}$$

with probability at least $1 - \gamma C_b(2s+1)$, where $\delta \equiv \max_{1\le k\le C_b(2s+1)} \left\{ \sqrt{\frac{\rho^2}{\gamma}\left(\frac{\max\{1, r^{1-2/k}\}}{dp}\right)^k} \right\}$.

The proof follows immediately using Theorem 6.3 and the following lemma which gives a uniform bound on the approximation error between H_{c_1,c_2} and $f = q_s \circ q$. Lemma 6.9, together with Equations. (6.16) and (7.4), provides a (deterministic) functional approximation guarantee of

$$\operatorname{rank}(M, c_1) - d\,2^{-s} \;\le\; \sum_{i=1}^{d} f(\sigma_i(M)) \;\le\; \operatorname{rank}(M, c_1) + d\,2^{-s}\,, \quad (6.21)$$

for any $c_1 < c_2$ and any choice of s, as long as C_b is large enough to guarantee 0.1 uniform error bound on the Chebyshev polynomial approximation. Since we can achieve $1 \pm \delta$ approximation on each polynomial in $f(\sigma_i(x))$, Theorem 6.3 implies the desired Corollary 6.8. Note that using Remarks 6.6

and 6.7, the bounds in (6.12) hold for $k \in [1, \infty)$ with $r^{1-2/k}$ replaced by $\max\{1, r^{1-2/k}\}$.

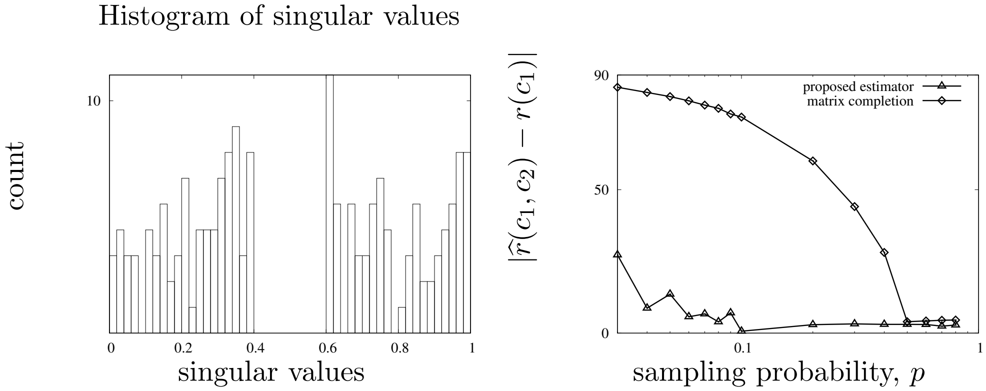


Figure 6.6: The left panel shows a histogram of singular values of M chosen for the experiment. The right panel compares absolute error in estimation $\widehat{r}(\mathcal{P}_\Omega(M); c_1 = 0.5, c_2 = 0.6)$ for two choices of the Schatten norm estimates $\widehat{\|M\|_k^k}$: first the proposed estimator $\widehat{\Theta}_k(\mathcal{P}_\Omega(M))$ in (7.12), and second the Schatten norm of the completed matrix, $\widetilde{M} = \text{AltMin}(\mathcal{P}_\Omega(M))$ from [97].

Lemma 6.9 ([217], Lemma 1)**.** *Consider the composite polynomial* $f(x) = q_s(q(x))$*. Then* $f(x) \in [0, 1]$ *for all* $x \in [0, 1]$*, and moreover*

$$|f(x) - H_{c_1,c_2}(x)| \leq 2^{-s}\,, \qquad \textit{for all } x \in [0, c_2] \cup [c_1, 1]\,. \qquad (6.22)$$

In Figure 6.6, we evaluate the performance of estimator (6.18) numerically. We construct a symmetric matrix M of size $d = 1000$ and rank $r = 200$. $\sigma_i \sim \text{Uni}(0, 0.4)$ for $1 \leq i \leq r/2$, and $\sigma_i \sim \text{Uni}(0.6, 1)$ for $r/2 + 1 \leq i \leq r$. We estimate $\widehat{r}(\mathcal{P}_\Omega(M); c_1, c_2)$ for Erdös-Rényi sampling Ω, and a choice of $c_2 = 0.5$ and $c_1 = 0.6$, which is motivated by the distribution of σ_i. We use Chebyshev polynomial of degree $C_b = 2$, and $s = 1$ for q_s. That is function f is of degree 6. Accuracy of the estimator can be improved by increasing C_b and s, however that would require estimating higher Schatten norms.

Estimating the spectrum

Given accurate estimates of first K Schatten norms of a matrix M, we can estimate singular values of M using a linear programming based algorithm given in [119]. In particular, we get the following guarantees on the estimated singular values, whose proof follows directly using the analysis techniques in the proof of [119, Theorem 2]. The main idea is that given the rank, the maximum support size of the true spectrum, and an estimate of its first K moments, one can find r singular values whose K first moments are close to the estimated Schatten norms.

Algorithm 10 Spectrum estimator (a variation of [119])

```
Input: $\mathcal{P}_\Omega(M)$, $K$, $\epsilon$, target rank $r$, lower bound $a$ and upper bound $b$ on
    the positive singular values
Output: estimated singular values $(\widehat{\sigma}_1, \widehat{\sigma}_2, \ldots, \widehat{\sigma}_r)$
  $L \in \mathbb{R}^K : L_k = \widehat{\Theta}_k(\mathcal{P}_\Omega(M))$ for $k \in [K]$        [Algorithm 8]
  $t = \lceil (b-a)/\epsilon \rceil + 1$, $x \in \mathbb{R}^t$: $x_i = a + \epsilon(i-1)$, for $i \in [t]$,
  $V \in \mathbb{R}^{K \times t} : V_{ij} = x_j^i$ for $i \in [K], j \in [t]$
  $p^* \equiv \{\min_{p \in \mathbb{R}^t} |Vp - L|_1 : \mathbb{1}_t^\top p = 1, p \geq 0\}$
  $\widehat{\sigma}_i = \min\{x_j : \sum_{\ell \leq j} p^*_\ell \geq \frac{i}{r+1}\}$, $i$th $(r+1)$st-quantile of distribution
    corresponding to $p^*$
```

Further, our upper bound on the first K moments can be translated into an upper bound on the Wasserstein distance between those two distributions, which in turn gives the following bound on the singular values. With small enough ϵ and large enough K and r, we need sample size $d^2 p > C_{r,K,\epsilon,\gamma} d r^{1-2/k}$ to achieve arbitrary small error.

Corollary 6.10. *Under the hypotheses of Theorem 6.3, given rank r, constants $0 \leq a < b$ such that $\sigma_{\min} \geq a$, $\sigma_{\max} \leq b$, and estimates of the first K Schatten norms of M, $\{\widehat{\Theta}_k(\mathcal{P}_\Omega(M))\}_{k \in [K]}$ obtained by the estimator (7.12), for any $0 < \epsilon \ll (b-a)$, and $\gamma > 0$, Algorithm 10 runs in time* $\text{poly}(r, K, (b-a)/\epsilon)$ *and returns $\{\widehat{\sigma}_i\}_{i \in [r]}$ an estimate of $\{\sigma_i(M)\}_{i \in [r]}$ such* *that*

$$\begin{aligned}
&\frac{1}{r}\sum_{i=1}^{r}|\widehat{\sigma}_i-\sigma_i| \\
&\leq \frac{C(b-a)}{K}+\frac{b-a}{r} \\
&+g(K)(b-a)\left(\epsilon K b^{K-1}+\sum_{k=1}^{K}\sigma_{\max}^{k}\sqrt{\frac{\rho^2}{\gamma}\left(\frac{\max\{1,r^{1-2/k}\}}{dp}\right)^{k}}\right),
\end{aligned} \tag{6.23}$$

with probability at least $1-\gamma K$*, where* C *is an absolute constant and* $g(K)$ *only depends on* K*.*

In Figure 6.7, we evaluate the performance of the proposed estimator (7.12), in recovering the true spectrum using Algorithm 10. We compare the results with the case when Schatten norms are estimated using matrix completion. We consider two distributions on singular values, one peak and two peaks. More general distributions of spectrum can be recovered accurately, however that would require estimating higher Schatten norms. For both cases, the proposed estimator outperforms matrix completion approaches, and achieves better accuracy as sample size increases with α. In each graph, the black solid line depicts the empirical Cumulative Distribution Function (CDF) of the ground truths $\{\sigma_i\}_{i\in[r]}$ for those r strictly positive singular values. In the first experiment (the top panel), there are r singular values at one peak $\sigma_i=1$, and in the second experiment (the bottom pannel) there are $r/2$ singular values at each of the two peaks at $\sigma_i=1$ and $\sigma_i=2$. Each cell shows the result of a choice of rank $r\in\{50,200,500\}$ and a parameter $\alpha\in\{3,5,8,10\}$, where Ω is generated using Erdös-Rényi sampling with probability $p=(\alpha/d)r^{1-2/7}$. M is a symmetric matrix of size $d=1000$ and rank r with singular values $\{\sigma_i\}_{i\in[d]}$. In each cell, there are one black line, three blue lines, and three orange lines. Each blue line corresponds to the empirical CDF of $\{\widehat{\sigma}_i\}_{i\in[d]}$ for each trial, over three independent trials. Each orange line corresponds to the empirical CDF of $\{\widetilde{\sigma}_i\}_{i\in[d]}$. Here, $\widehat{\sigma}_i$'s are estimated using $\{\widehat{\Theta}_k(\mathcal{P}_\Omega(M))\}_{k\in[K]}$ obtained by the estimator (7.12), and $\widetilde{\sigma}_i$'s are estimated using $\{\|\widetilde{M}\|_k^k\}_{k\in[K]}$ where $\widetilde{M}=\text{AltMin}(\mathcal{P}_\Omega(M))$, along with Algorithm 2 in [119], for $K=7$.

6.4 Graph sampling

Our framework for estimating the Schatten k-norms can be applied more generally to any random sampling, as long as the distribution is permutation invariant. In practice, we typically observe one instance of a sampled matrix and do not know how the samples were generated. Under a mild assumption that the probability of sampling an entry is independent of the value of that entry, the only information about the sampling model that we have is the *pattern*, i.e. an unlabelled graph $G = (V, E)$ capturing the pattern of sampled indices by the edges. This naturally suggests a novel sampling scenario that we call *graph sampling*.

The Erdös-Rényi sampling has been criticized as being too strict for explaining how real-world datasets are sampled. When working with natural data, we typically only get one instance of a sampled matrix without the knowledge of how those entries are sampled. In this section, we propose a new sampling model that we call *graph sampling* that makes minimal assumptions about how the data was sampled. We assume that the *pattern* has been determined a priori, which is represented by a deterministic graph $G = (V, E)$ with d nodes denoted by V and undirected edges denoted by E. The random sampling Ω is chosen uniformly at random over all relabeling of the nodes in G. Formally, for a given $G = (V, E)$, a permutation $\pi : [d] \to V$ is drawn uniformly at random and samples are drawn according to

$$\mathcal{P}_\Omega(M) \;=\; \{(i, j, M_{ij})\}_{(\pi(i),\pi(j))\in E} \;. \tag{6.24}$$

As the sampling pattern G is completely known to the statistician who only has one instance of a random sampling, we are only imposing that the samples are drawn uniformly at random from all instances that share the same pattern. Further, understanding this graph sampling model has a potential to reveal the subtle dependence of the estimation problem to the underlying pattern, which is known to be hard even for an established area of matrix completion.

In this section, we provide an estimator under graph sampling, and characterize the fundamental limit on the achievable error. This crucially depends on the original pattern G via a fundamental property $\lambda^*_{G,r}$, which is generally challenging to compute. However, we provide a bound on $\lambda^*_{G,r}$

for two extreme cases of varying difficulty: a clique sampling that requires only $O(r^{2-4/k})$ samples and a clique-star sampling that requires as many as $O(dr^{1-4/k})$ samples. This is made formal by showing a lower bound on the minimax sample complexity. Comparing the two necessary conditions on sample complexity, $O(r^{2-4/k})$ for clique sampling and $O(dr^{1-4/k})$ for clique-star sampling, it follows that depending on the pattern of the samples, the sample complexity can vary drastically, especially for low-rank matrices where $r \ll d$.

Under the *graph sampling*, the probability $p(H)$ that we observe all edges in a walk with pattern H is

$$p(H) \;=\; \frac{\omega_{\mathcal{P}_\Omega(\mathbb{1}_d\mathbb{1}_d^T)}(H)}{\omega_{\mathbb{1}_d\mathbb{1}_d^T}(H)}\;, \tag{6.25}$$

where $\mathbb{1}_d\mathbb{1}_d^T$ is the all ones matrix, and by permutation invariance, the probability is the ratio between total (unweighted) number of walks with $H(w) = H$ in the original pattern Ω and that of the complete graph K_d. Note that although Ω is a random quantity, $\omega_{\mathcal{P}_\Omega(\mathbb{1}\mathbb{1}^T)}(H)$ only depends on the structure and not the labelling of the nodes and hence is a deterministic quantity. Plugging in this value of $p(H)$, which can be computed in time $O(d^\alpha)$ for $k \le 7$ as shown in Proposition 6.2 (and in general only increases the computational complexity of the estimate by a factor of two), into the estimator (7.12), we get an estimate customized for graph sampling.

6.4.1 Performance Guarantees

Recall the graph sampling defined above, where we relabel the nodes of a pattern graph $G(V, E)$ according to a random uniform permutation, and sample the entries of the matrix M on the edges. We prove a fundamental lower bound on the sample complexity that crucially depends on the following property of the pattern G. Let $G_\pi(\widetilde{V}, \Omega)$ denote the graph after relabeling the nodes of $G = (V, E)$ with permutation $\pi : [d] \to [d]$. For independent Rademacher variables u_i for $i \in [r]$

$$f_{G,r}(\lambda) \;\equiv\; \max_{\pi}\left\{\mathbb{E}_u\left[\exp\left((5/d)^2\lambda^2 \sum_{(i,j)\in\mathcal{P}^{(r)}(G_\pi)} u_i u_j\right)\right]\right\}\,, \tag{6.26}$$

where $\mathcal{P}^{(r)}(G_\pi) \subseteq [r]\times[r]$ is a projection of the edges Ω over d nodes to a set of edges over r nodes by mapping a node $i \in [d]$ to a node $1 + (i - 1 \bmod r) \in [r]$. Precisely, $(i,j) \in \mathcal{P}^{(r)}(G_\pi)$ if there exists an edge $(i',j') \in \Omega$ such that $i = 1 + (i' - 1 \bmod r)$ and $j = 1 + (j' - 1 \bmod r)$. Observe that $f_{G,r}(\lambda)$ is a non-decreasing function of λ. It follows from the fact that for any positive λ and random variable x and any $\epsilon > 0$, we have $\mathbb{E}[e^{\lambda(1+\epsilon)x}] \geq \mathbb{E}[e^{\lambda x}](\mathbb{E}[e^{\lambda x}])^\epsilon \geq \mathbb{E}[e^{\lambda x}]e^{\epsilon\lambda\mathbb{E}[x]} \geq \mathbb{E}[e^{\lambda x}]$. The first and the second inequalities use Jensen's inequality and the third one holds when $\mathbb{E}[x] \geq 0$. Note that $\mathbb{E}_u[\sum_{(i,j)\in\mathcal{P}^{(r)}(G_\pi)} u_i u_j] \geq 0$, since u_i's are i.i.d. Rademacher variables.

This function measures the distance between a particular low-rank matrix with Gaussian entries and its rank one perturbation, which is used in our constructive lower bound (see Eq. (6.60)). Intuitively, smaller $f_{G,r}(\lambda)$ implies that two rank-r matrices with separated Schatten norms look similar after graph sampling w.r.t. G. Hence, when this function is small, say less than 26/25, then it is hard to distinguish which of the two (distributions of) matrices we are observing. This is captured by the largest value of λ that still maintains $f_{G,r}(\lambda)$ sufficiently small:

$$\lambda^*_{G,r} \;\; \equiv \max_{\{\lambda>0: f_{G,r}(\lambda)\leq 26/25\}} \lambda \; . \tag{6.27}$$

One can choose any number not necessarily 26/25 as long as it is strictly larger than one and strictly smaller than two, and this will only change the probability upper bound in (6.28). If we sample from a graph G with large $\lambda^*_{G,r}$, then we cannot distinguish two distributions even if they have a large Schatten norm separation. We do not have enough samples and/or our pattern is not sample efficient. The dependence of the fundamental lower bound on the graph G is captured by this property $\lambda^*_{G,r}$, which is made precise in the following theorem. We provide a lower bound that captures how sample complexity depends on the pattern G and also on the underlying matrix, by providing analysis customized for each family of matrices $\mathcal{M}_{r,\mu}$ parametrized by its rank and incoherence:

$$\mathcal{M}_{r,\mu} \equiv \big\{ M \in \mathbb{R}^{d\times d} : M = M^\top, \text{rank}(M) \leq r \, , \, \mu(M) \leq \mu \big\}.$$

Theorem 6.11 (General lower bound under graph sampling)**.** *For any finite $k \in [3,\infty)$ suppose we observe samples under the graph sampling defined* *above with respect to a pattern graph $G = (V, E)$. Then there exist universal constants $C > 0$, $C' > 0$ and $C'' > 0$ such that for any $r \geq e^{C''k}$ and $\mu \geq C'\sqrt{\log r}$, if $\lambda^*_{G,r} \geq C d r^{1/k-1/2}$ then*

$$\inf_{M \in \mathcal{M}_{r,\mu}} \sup_{\widetilde{\Theta}} \mathbb{P}\bigg(\frac{1}{2}\|M\|_k \leq \widetilde{\Theta}(\mathcal{P}_{\Omega(M)}) \leq 2\|M\|_k\bigg) \;\; \leq \;\; \frac{3}{4}\,, \qquad (6.28)$$

where the supremum is over any measurable function of $\mathcal{P}_{\Omega(M)}$ and the probability is with respect to the random sampling Ω.

A proof of Theorem 6.11 is given in Section 6.8.3. It is in general challenging to evaluate $\lambda^*_{G,r}$ for a given graph. For a special case of *clique sampling* where the pattern $G(V, E)$ is a clique over a subset of ℓ nodes among d, we provide a sharp upper bound on $\lambda^*_{G,r}$.

Lemma 6.12 (Lower bound for clique sampling)**.** *If the pattern graph $G(V, E)$ is a clique over a subset of ℓ nodes, then $\lambda^*_{G,r} \leq 2^{-4} d(\min\{\ell, r\})^{-1/2}$.*

Together with Theorem 6.11, this implies that if $\ell \leq 2^{-8}C^{-2}r^{1-2/k}$ (such that $\lambda^*_{G,r} \geq C d r^{1/k-1/2}$), then with probability at least 1/4 any estimator makes an multiplicative error larger than two. Hence, sample size of $\ell(\ell + 1)/2 = O(r^{2-4/k})$ is necessary to achieve multiplicative error of two with high probability. We show that our estimator is optimal, by providing a matching upper bound on the sample complexity when $k = 3$. For any positive semidefinite matrix $M \in \mathbb{R}^{d \times d}$ of rank r with incoherence $\mu(M) = \mu$, $\kappa = \sigma_{\max}(M)/\sigma_{\min}(M)$, and some function $g(k) = O(k!)$, we define

$$\tilde{\rho}^2 \;\; \equiv \;\; (\kappa\mu)^{2k} g(k) \max\left\{1\,,\, \frac{\ell^{k-1}}{r^{k-2}}\,,\, \frac{\ell}{r}\,,\, \frac{r^{1/2}\ell^k}{d}\right\},$$

such that the variance of our estimator is bounded by $\mathrm{Var}(\widehat{\Theta}(\mathcal{P}_\Omega(M))/\|M\|_k^k) \leq \rho^2(r^{1-2/k}/\ell)^k$ as we show for $k = 3$ in the proof of Theorem 6.13 in Section 6.8.6. Here, $g(k) = O(k!)$ is a function of k only.

Theorem 6.13 (Upper bound for clique sampling)**.** *For $k = 3$, any $\delta > 0$, and any rank-r matrix $M \succeq 0$, the proposed estimator* (7.12) *achieves a* *multiplicative error δ with probability of error bounded by*

$$\mathbb{P}\left(\frac{\big|\widehat{\Theta}_k(\mathcal{P}_\Omega(M)) - \|M\|_k^k\big|}{\|M\|_k^k} \;\geq\; \delta \right) \;\leq\; \frac{\tilde{\rho}^2}{\delta^2}\Big(\frac{r^{1-2/k}}{\ell}\Big)^k \,, \qquad (6.29)$$

under the graph sampling with the pattern graph G that is a clique over ℓ nodes.

For a typical scenario with finite μ and κ, this upper bound shows that sample size of $\ell(\ell+1)/2 = O(r^{2-4/k})$ is sufficient to achieve any arbitrarily small multiplicative error for $k = 3$ and sufficiently small rank $r \leq d^{2k/(3k-2)}$ and $\ell \leq r^{(k-2)/(k-1)}$, to ensure that the first term dominates in $\tilde{\rho}^2$. However, the numerical experiments suggest that our analysis holds more generally for all regimes of the rank r. This matches the previous lower bound, proving optimality of the proposed estimator. Although the current analysis holds only for $k = 3$, we are intentionally writing the guarantee in general form as we expect the bound to hold more generally. In particular, we believe that Lemma 6.19 holds for all $k \geq 3$, and thereby Theorem 6.13 holds for any fixed integer $k \in [3, \infty)$. In the numerical experiments in Figure 6.8, M is generated using settings similar to that of Figure 6.4. Empirical probabilities are computed by averaging over 100 instances.

Although our analysis does not give a tight lower bound for Erdös-Rényi sampling, there exists graph patterns such that sample complexity is large, i.e. scales linearly in d. Consider a *clique-star sampling* where the pattern graph $G(V, E)$ has a clique on a small subset of nodes V_1, $|V_1| = \ell$, and the remaining nodes $V \setminus V_1$ are disconnected among themselves and are fully connected with the clique in V_1. Precisely, $G = (V, E)$ with $(i, j) \in E$ if $i \in V_1$ or $j \in V_1$.

Lemma 6.14 (Lower bound for clique-star sampling)**.** *Under the clique-star sampling over a clique of size ℓ, there exists an absolute constant c such that $\lambda^*_{G,r} \leq cd(r(\min\{\ell, r\}))^{-1/4}$.*

Together with Theorem 6.11, this implies that if $\ell \leq c^4 C^{-4} r^{1-4/k}$, then with probability at least 1/4 any estimator makes an multiplicative error larger than two. This implies that the total number of edges in the pattern graph should be $O(dr^{1-4/k})$ for accurate estimation. Together with the upper bound on clique sampling in Theorem 6.13, this shows that the sample complexity can drastically change based on the pattern of your sampling model. Clique sampling requires only $O(r^{2-4/k})$ samples (for $k = 3$) whereas clique-star sampling requires at least $O(dr^{1-4/k})$. A proof of Lemma 6.12 and Lemma 6.14 is given in Section 6.8.4 and 6.8.5 respectively.

6.5 Discussion

We list some observations and future research directions.

Complexity of the estimator beyond $k = 7$. For $k \geq 8$, our approach of using matrix operations to count (the weights of) walks for each pattern $H \in \mathcal{H}_k$ can potentially be extended. However, the complexity of the problem fundamentally changes for $k \geq 8$. As our estimator is at least as hard as counting small structures in a simple (unweighted) graph, we can borrow known complexity results to get a lower bound. For instance, for $k \geq 8$, we need to count K_4 in a graph, which the best known run time is $O(d^{\alpha+1})$ for general graphs [117]. For general k, under standard hardness assumptions, [71] show that there is no algorithm with run time $O(f(k)d^c)$ for counting cycles of length k, for any function $f(k)$ and a constant c that does not depend on k. In comparison, finding *one cycle* of length k can be done in time $2^{O(k)}d^\alpha$ [6]. This implies that the complexity should scale as $O(d^{f(k)})$, and we believe $f(k)$ should be larger than $(\alpha\sqrt{2k}/3)$. The reason is that for $k \geq \binom{\ell}{2}$ for an odd ℓ, our estimator needs to count the number of cliques K_ℓ of size ℓ. Similarly, for $k \geq (1/2)\ell^2$ for an even ℓ, we require counting K_ℓ. The best known algorithm for counting K_ℓ takes time $O(\min\{d^{1+\alpha\lceil(\ell-1)/3\rceil}, d^{2+\alpha\lceil(\ell-2)/3\rceil}\})$ for general graphs [6, Theorem 6.4]. Putting these bounds together, we believe that the estimator take time at least $d^{\alpha\sqrt{2k}/3}$.

Graph sampling. Typical guarantees known for matrix completion assumes the Erdös-Rényi sampling. One exception is the deterministic sampling studied by [20], but such generalization in sampling comes at a price of requiring more strict assumptions on the matrix M. We propose graph sampling, which can potentially capture how estimation guarantees depends explicitly on the pattern G, and still remain analytically tractable. We give such examples for special graphs in Section 6.4, and graph sampling model can potentially be used to bridge the gap in sampling models between theory and practice.

(Standard) rank estimation. As several popular matrix completion approaches require the knowledge of the rank of the original matrix, it is of great practical interest to estimate the standard rank of a matrix from sampled entries. Our framework in Section 6.3.2 provides a way to estimate the standard rank from samples. However, there are a few parameters that needs to be tuned, such as the thresholds c_1 and c_2, and the degree of the polynomial approximation and the degree of the Schatten norm. For rank estimation, [112] give an estimator that is provably correct in the regime where matrix completion works, justifying the requirement that popular matrix completion algorithms [110, 97] need to know the underlying rank. However, in the regime of our interest, which is below the standard matrix completion threshold, the algorithm fails miserably and there are no guarantees. In a more recent work, [177] propose a novel rank estimator of counting the negative eigenvalues of Bethe Hessian matrix. It is an interesting future direction to build upon our framework to provide a guideline for choosing the parameters for standard rank estimation, and compare its performance to existing methods.

The effect of the effective rank. One property of the Schatten norm is that as k gets large and as the singular values have small effective rank (meaning that they decay fast), the summation is dominated by the largest few singular values. In such scenarios, in the estimation problem, any algorithm that tracks the first few singular values correctly would achieve small error. Hence, the gap get smaller as effective rank gets smaller, between the proposed estimator and the simple Schatten k-norm of the rescaled sampled matrix, as depicted in Figure 6.9. We are using the same setting as those in Figure 6.4 with a full rank matrix M with $r = d = 500$, but the effective rank is relatively small as the singular values are decaying as $\sigma_i = 1/i^2$. For the current choice of $k = 5$, notice that the contribution in $\|M\|_k^k$ of the 2nd singular value is a factor of 2^{10} smaller than the top singular value, making it effectively a rank one matrix.

Technical challenges. The technical challenge in proving bounds on the necessary number of samples needed to estimate Schatten k-norms lies in getting tight bounds on the variance of the estimator. Variance is a function of weighted counts of each pseudograph of $2k$-closed walks, in the complete matrix. As the weight of each walk can be positive or negative, significant cancellation occurs when we sum all the weights. However, this stochastic cancellation is hard to capture in the analysis and we assume the worst case when all the weights are positive, which cannot occur for incoherent and well-conditioned matrices. This weakness of the analysis leads to the requirement of rank being sufficiently small in the case of Erdös-Rényi sampling and k small in the case of clique sampling. We believe these bounds can be tightened and the same is reflected in the numerical simulations which show the same scaling holds for all small values of k and rank close to the dimension of the matrix.

6.6 Algorithm for estimating Schatten k-norm for $k \geq 8$

The collection of pseudographs $\mathcal{H}_k$ is partitioned into sets $\{\mathcal{H}_{k,i}^{\text{iso}}\}_{1\leq i\leq r}$, for some $r \leq k!$. The partitions $\mathcal{H}_{k,i}^{\text{iso}}$ are defined such that the pseudographs in one partition are isomorphic to each other when multi-edges are condensed into one. This is useful since all the pseudographs in one partition are observed together in $G([d], \Omega)$ for any fixed subgraph in G. The underlying simple graph (including self loops) for each partition $\mathcal{H}_{k,i}^{\text{iso}}$ is denoted by $F_{k,i}$.

The main idea is to enumerate a list $\mathcal{L}_\ell$ of all connected ℓ-vertex induced subgraphs (possibly with loops) of the graph $G([d], \Omega)$, for each $1 \leq \ell \leq k$. The unbiased weighted count of all pseudographs $\mathcal{H}_k$ for each of these vertex induced subgraphs $g \in \mathcal{L}_\ell$ is computed. This is achieved by further enumerating a list $\mathcal{S}_{g,\ell}$ of all ℓ-vertex subgraphs for each g. Then the unbiased weight of all pseudographs $H \in \mathcal{H}_k$ that exist in the subgraph h is computed and is summed over to get the estimate of the k-th Schatten norm. Recall the notation $\mathcal{P}_\Omega(M)$ which is used to denote the partially observed matrix corresponding to the index set Ω with the unobserved entries being replaced by zero. We abuse this notation and use $h(M)$ to represent the matrix M restricted to the subgraph h of the observed graph $G([d], \Omega)$.

Each connected induced subgraphs of size k in a graph can be enumerated in time polynomial in d and k [60]. The number of connected induced subgraphs of size k in a graph is upper bounded by $(e\Delta)^k/((\Delta-1)k)$ where Δ is the maximum degree of the graph [197]. Therefore, Algorithm 11 runs in time, super exponential in k, polynomial in d and the number of k connected induced subgraphs in the observed graph $G([d],\Omega)$.

Algorithm 11 Schatten k-norm estimator

```
Input: $\mathcal{P}_\Omega(M)$, $k$, $\mathcal{H}_k$, $p(H)$ for all $H \in \mathcal{H}_k$
Output: $\widehat{\Theta}_k(\mathcal{P}_\Omega(M))$
$\widehat{\Theta}_k(\mathcal{P}_\Omega(M)) \leftarrow 0$
for $1 \le \ell \le k$ do
    Enumerate a list, $\mathcal{L}_\ell$, of all connected $\ell$-vertex induced subgraphs (possibly with loops) of the graph $G([d],\Omega)$
    for all $g \in \mathcal{L}_\ell$ do
        Enumerate a list $\mathcal{S}_{g,\ell}$ of all connected $\ell$-vertex subgraphs of the graph $g$ by removing one or more edges
        for all $h \in \mathcal{S}_{g,\ell}$ do
            for $1 \le i \le r$ do
                if $h$ is isomorphic to $F_{k,i}$ then
                    $\widehat{\Theta}_k(\mathcal{P}_\Omega(M)) \leftarrow \widehat{\Theta}_k(\mathcal{P}_\Omega(M)) + \sum_{H \in \mathcal{H}^{\mathrm{iso}}_{k,i}} \frac{1}{p(H)} \omega_{h(M)}(H) c(H)$
                end if
            end for
        end for
    end for
end for
```

6.7 Algorithm for computing the Chebyshev polynomial

In the following, we algorithm to compute coefficients of the Chebyshev polynomial of first kind.

6.8 Proofs

We provide proofs for main results and technical lemmas.

6.8.1 Proof of Theorem 6.3

Consider $\widetilde{W}$ to be the collection of all length k closed walks on a complete graph of d vertices. Here we slightly overload the notion of complete graph to refer to an undirected graph with not only all the $d(d-1)/2$ simple edges

Algorithm 12 Chebyshev polynomial of the first kind approximating $H_{c_1,c_2}(x)$

```
Input: $H_{c_1,c_2}$, $c_1$, $c_2$, and target accuracy $\delta = 0.1$
Output: Chebyshev polynomial $q(x)$ of first kind
$g(x) \equiv \frac{x-c_2}{c_1-c_2}$
$T_0(x) \equiv 1$, $T_1(x) \equiv x$
$q(x) \leftarrow \frac{1}{\pi}\int_{c_2}^{c_1}(1-x^2)^{-1/2}g(x)T_0(x)dx + \frac{1}{\pi}\int_{c_1}^{1}(1-x^2)^{-1/2}T_0(x)dx$
$i = 1$
while $\sup_{x\in[0,1]} |q(x) - H_{c_1,c_2}(x)| \ge \delta$ do
    $q(x) \leftarrow q(x) + \frac{2T_i(x)}{\pi}\int_{c_2}^{c_1}(1-x^2)^{-1/2}g(x)T_i(x)dx + \frac{2T_i(x)}{\pi}\int_{c_1}^{1}(1-x^2)^{-1/2}T_i(x)dx$
    $i \leftarrow i + 1$
    $T_i(x) \equiv 2xT_{i-1}(x) - T_{i-2}(x)$
end while
```

but also with d self loops as well. Construct the largest possible collection W from $\widetilde{W}$ wherein each walk has distinct weights that is $\omega(w) \neq \omega(w')$ for all $w, w' \in W$. We partition W according to the pattern among k-cyclic pseudographs, which are further partitioned into four groups. The estimator (7.12) can be re-written as

$$\begin{aligned}
\widehat{\Theta}_k(\mathcal{P}_\Omega(M)) &= \sum_{w\in W} \frac{c(H(w))}{p(H(w))}\,\omega_{\mathcal{P}_\Omega(M)}(w) \\
&= \sum_{H\in\mathcal{H}_k} \left\{ \frac{c(H)}{p(H)} \sum_{w:H(w)=H} \omega_M(w)\,\mathbb{I}(w \subseteq \Omega) \right\} \qquad (6.30) \\
&= \sum_{i=1}^{4} \sum_{H\in\mathcal{H}_{k,i}} \left\{ \frac{c(H)}{p(H)} \sum_{w:H(w)=H} \omega_M(w)\,\mathbb{I}(w \subseteq \Omega) \right\}, \qquad (6.31)
\end{aligned}$$

where we write $w \subseteq \Omega$ to denote the event that all the edges in the walk w are sampled, and we define

- $\mathcal{H}_{k,1} \equiv \{C_k\}$ is just a (set of a) simple cycle of length k and there are total $|\{w \in W : H(w) \in \mathcal{H}_{k,1}\}| = \binom{d}{k}(k!/2k) \le (d^k/2k)$ corresponding walks to this set, and $c(C_k) = 2k$.

- $\mathcal{H}_{k,2} \equiv \{H(V_H, E_H) \in \mathcal{H}_k : |V_H| \le k-1$ and no self loops$\}$, and there are total $|\{w \in W : H(w) \in \mathcal{H}_{k,2}| \le d^{k-1}$ corresponding walks to this set.

- $\mathcal{H}_{k,3} \equiv \bigcup_{s=1}^{k-1} \mathcal{H}_{k,3,s}$ where $\mathcal{H}_{k,3,s} = \{H \in \mathcal{H}_k$ with s self loops$\}$, and there are total $|\{w \in W : H(w) \in \mathcal{H}_{k,3}\}| \leq d^{k-s}$ corresponding walks in this set.

- $\mathcal{H}_{k,4} \equiv \{H(V_H, E_H) \in \mathcal{H}_k : |V_H| = 1\}$ is a (set of a) graph with k self loops and there are total $|\{w \in W : H(w) \in \mathcal{H}_{k,4}\}| = d$ corresponding walks to this set.

Given this unbiased estimator, we provide an upper bound on the variance of each of the partitions to prove concentration with Chebyshev's inequality. For any walk $w \in W$, let $|w|$ denote the number of unique edges (including self loops) that the walk w traverses. Let $|w \cap w'|$ denote the number of unique overlapping edges (including self loops) of walks w and w'. We have,

$$\begin{aligned}
&\mathrm{Var}\Big(\widehat{\Theta}_k(\mathcal{P}_\Omega(M))\Big) \\
&= 2\sum_{\ell=1}^{k-1} \sum_{\substack{w \neq w' \in \widetilde{W} \\ |w \cap w'| = \ell}} \mathrm{Covar}\left(\frac{\mathbb{I}(w \subseteq \Omega)\omega_M(w)c(H(w))}{p(H(w))}, \frac{\mathbb{I}(w' \subseteq \Omega)\omega_M(w')c(H(w'))}{p(H(w'))} \right) \\
&+ \; \sum_{i=1}^{4} \sum_{H \in \mathcal{H}_{k,i}} \left\{ \frac{c(H)^2}{p(H)^2} \sum_{w : H(w) = H} \omega_M(w)^2 \mathrm{Var}\Big(\mathbb{I}(w \subseteq \Omega)\Big) \right\} \qquad (6.32) \\
&< 4\sum_{\ell=1}^{k-1} \sum_{\substack{w \neq w' \in W \\ |w \cap w'| = \ell}} \mathbb{E}\Big[\mathbb{I}(w \subseteq \Omega)\mathbb{I}(w' \subseteq \Omega)\Big] \left(\frac{\big|\omega_M(w)\,\omega_M(w')\big| c(H(w))c(H(w'))}{p(H(w))\,p(H(w'))} \right) \\
&+ \; \sum_{i=1}^{4} \sum_{H \in \mathcal{H}_{k,i}} \sum_{w : H(w) = H} \frac{c(H)^2 \omega_M(w)^2}{p(H)^2} \mathbb{E}\Big[\mathbb{I}(w \subseteq \Omega)\Big]. \qquad (6.33)
\end{aligned}$$

Recall from the definition of incoherence that $|M_{ii}| \leq \sigma_1(M)\mu r/d$ and $|M_{ij}| = \sigma_1(M)\mu r^{1/2}/d$, and let $\alpha = \sigma_1(M)\mu r^{1/2}/d$ denote the maximum off-diagonal entry, such that $|M_{ij}| \leq \alpha$ and $|M_{ii}| \leq \alpha\sqrt{r}$ for all $i, j \in [d]$. Let $A_{p,k,\alpha,d} = d^k \alpha^{2k}/p^k$ denote the target scaling of the variance, then

$$\sum_{H \in \mathcal{H}_{k,i}} \sum_{w : H(w) = H} \frac{c(H)^2\, \omega_M(w)^2}{p(H)^2} \mathbb{E}\Big[\mathbb{I}(w \subseteq \Omega)\Big] \;\leq$$

$$\begin{cases} \dfrac{d^k}{2k}\dfrac{(2k)^2\alpha^{2k}}{p^k} = 2kA_{p,k,\alpha,d}\,, & \text{for } i=1 \quad, \qquad (6.34) \\ d^{k-1}\dfrac{f(k)^2\alpha^{2k}}{p^k} = \dfrac{f(k)^2}{d}A_{p,k,\alpha,d}\,, & \text{for } \ i=2 \quad, \qquad (6.35) \\ d\dfrac{r^k\alpha^{2k}}{p} = \dfrac{r^kp^{k-1}}{d^{k-1}}A_{p,k,\alpha,d}\,, & \text{for } \ i=4\,, \qquad (6.36) \end{cases}$$

and for $i=3$ and for $1 \leq s \leq k-1$, we have

$$\sum_{H\in\mathcal{H}_{k,3,s}} \sum_{w:H(w)=H} \frac{c(H)^2\,\omega_M(w)^2}{p(H)^2}\mathbb{E}\Big[\mathbb{I}(w \subseteq \Omega)\Big]$$
$$\leq \ d^{k-s}\frac{f(k)^2\alpha^{2k}r^s}{p^k} = \frac{f(k)^2r^s}{d^s}A_{p,k,\alpha,d}\ , \qquad (6.37)$$

where $c(H)$ is defined as the multiplicity of walks with the same weight satisfying $c(H) \leq f(k)$. For $w \neq w'$ and $|w \cap w'| = \ell$, where the range of ℓ varies across equations depending upon the set to which w, w' belongs, we have the following:

$$\sum_{\substack{w\neq w'\in W \\ |w\cap w'|=\ell, H(w)\in\mathcal{H}_{k,i,s}, H(w')\in\mathcal{H}_{k,i',s'}}} \mathbb{E}\big[\mathbb{I}(w\in\Omega)\mathbb{I}(w'\in\Omega)\big]\cdot$$
$$\frac{\big|\omega_M(H(w))\omega_M(H(w'))\big|\,c(H(w))c(H(w'))}{p(H(w))p(H(w'))} \ \leq$$

$$
\begin{cases}
\frac{d^k d^{k-(\ell+1)}}{2k}\frac{\alpha^{2k}(2k)^2}{p^\ell} \; = \; \frac{(dp)^{k-\ell}}{d}\; 2kA_{p,k,\alpha,d}, \\
\qquad\qquad\qquad\qquad\qquad\qquad \text{for } \; i=i'=1 \quad (6.38) \\
\frac{f(k)^2 d^{k-1} d^{k-1-(\ell+1)}\alpha^{2k}}{p^\ell} \; \le \; \frac{f(k)^2(dp)^{k-\ell}}{d^3}A_{p,k,\alpha,d} \\
\qquad\qquad\qquad\qquad\qquad\qquad \text{for } i=i'=2 \quad (6.39) \\
\frac{f(k)^2 d^{k-s} d^{k-s'-\ell}\alpha^{2k-s-s'}(\alpha\sqrt{r})^{s+s'}}{p^\ell} \; \le \; \frac{f(k)^2(dp)^{k-\ell}}{(d/\sqrt{r})^{s+s'}}A_{p,k,\alpha,d}\,, \\
\qquad\qquad\qquad\qquad\qquad\qquad \text{for } \; i=i'=3 \quad (6.40) \\
\frac{f(k)^2 d^k d^{k-1-(\ell+1)}\alpha^{2k}}{p^\ell} \; \le \; \frac{f(k)^2(dp)^{k-\ell}}{d^2}A_{p,k,\alpha,d}. \\
\qquad\qquad\qquad\qquad\qquad\qquad \text{for } i=1, i'=2 (6.41) \\
\frac{f(k)^2 d^k d^{k-s-(\ell+1)}\alpha^{2k-s}(\alpha\sqrt{r})^{s}}{p^\ell} \; \le \; \frac{f(k)^2(dp)^{k-\ell}}{d(d/\sqrt{r})^{s}}A_{p,k,\alpha,d} \\
\qquad\qquad\qquad\qquad\qquad\qquad \text{for } i=1, i'=3 (6.42) \\
\frac{f(k)^2 d^{k-1} d^{k-s-(\ell+1)}\alpha^{2k-s}(\alpha\sqrt{r})^{s}}{p^\ell} \; \le \; \frac{f(k)^2(dp)^{k-\ell}}{d^2(d/\sqrt{r})^{s}}A_{p,k,\alpha,d} \\
\qquad\qquad\qquad\qquad\qquad\qquad \text{for } i=2, i'=3 (6.43) \\
\frac{f(k)^2 d d^{k-s-\ell}\alpha^{k-s}(\alpha\sqrt{r})^{k+s}}{p^\ell} \; \le \; \frac{f(k)^2(dp)^{k-\ell}}{d^{k-1}(d/\sqrt{r})^{k+s}}A_{p,k,\alpha,d} \\
\qquad\qquad\qquad\qquad\qquad\qquad \text{for } i=3, i'=4 (6.44)
\end{cases}
$$

where (6.44) is valid only for $\ell = 1$. Note that for any w with $H(w) \in \mathcal{H}_{k,1} \bigcup \mathcal{H}_{k,2}$, it has no overlap with w' such that $H(w') \in \mathcal{H}_{k,4}$.

Observe that $\mathrm{Var}\big(\widehat{\Theta}_k(\mathcal{P}_\Omega(M))\big)$ as bounded in (6.33) is upper bounded by the sum of quantities in (6.70)-(6.44), summating over all possible values of $1 \le \ell \le k-1$, and $1 \le s, s' \le k-1$. Let $h(k) \equiv f(k)^2 A_{p,k,\alpha,d}$. Observe that quantities in (6.70),(6.71), and (6.73) are upper bounded by $h(k)$. Quantities in (6.38)-(6.44) are upper bounded by $h_1(k) \equiv h(k)(dp)^{k-1}/d$. Quantity in (6.72) is upper bounded by $h_2(k) \equiv h(k) r^k p^{k-1}/d^{k-1}$.

Given $\|M\|_k^k \ge r(\sigma_{\min})^k$, recall a bound on off diagonals of matrix M by $|M_{ij}| \le \alpha = \mu\sigma_{\max}\sqrt{r}/d$ and $A_{p,k,\alpha,d} = d^k\alpha^{2k}/p^k$. This gives

$$
\frac{A_{p,k,\alpha,d}}{\|M\|_k^{2k}} \; \le \; \frac{\kappa^{2k}\mu^{2k}r^{k-2}}{d^k p^k}\;. \tag{6.45}
$$

Using Chebyshev's inequality and collecting all terms in the upper bound on the variance, we have for sufficiently large d, the following bound:

$$\begin{aligned}&\mathbb{P}\left(\frac{\left|\widehat{\Theta}_k(\mathcal{P}_\Omega(M)) - \|M\|_k^k\right|}{\|M\|_k^k} \;\geq\; \delta\right)\\ \leq\;& \frac{(\kappa\mu)^{2k} f(k)^2 r^{k-2}}{\delta^2 (dp)^k} \max\left\{1, \frac{(dp)^{k-1}}{d}, \frac{r^k p^{k-1}}{d^{k-1}}\right\}, \end{aligned} \tag{6.46}$$

where the second and the third term in the max expression follow by evaluating $h_1(k)$ and $h_2(k)$. If sampling probability p is small enough such that $dp \leq C d^{1/(k-1)}$ for some constant C, then the second and the third terms are smaller than the first term. Hence, the desired result in Theorem 6.3 follows.

6.8.2 Proof of Theorem 6.4

We can prove a Bernstien-type bound on accuracy of the estimator. The estimator (7.12) can be re-written as a multi-linear polynomial function of $d(d+1)/2$ i.i.d. Bernoulli(p) random variables.

$$\widehat{\Theta}_k(\mathcal{P}_\Omega(M)) \;=\; \sum_{w\in W}\left\{\frac{c(H(w))}{p(H(w))}\,\omega_M(w) \prod_{(i,j)\in\text{unique}(w)} \mathbb{I}((i,j)\in\Omega)\right\} \tag{6.47}$$

where $\mathbb{I}((i,j)\subseteq\Omega)$ is a random variable that takes value 1 if the $(i,j)_{\text{th}}$ entry of the matrix M is sampled, and unique(w) denotes the set of the unique edges (and self loops) that the walk w traverses. Let q denote the power of the polynomial function that is the maximum number of unique edges in the walk w, that is $q = k$.

We use the following Bernstien-type concentration results of [180] for the polynomials of independent random variables.

Lemma 6.15 ([180],Theorem 1.3)**.** *We are given $d(d+1)/2$ independent central moment bounded random variables $\{\mathbb{I}((i,j)\in\Omega)\}_{1\leq i\leq j\leq d}$ with same parameter L. We are given a multilinear polynomial $\widehat{\Theta}_k(\mathcal{P}_\Omega(M))$ of power q, then*

$$\begin{aligned}&\mathbb{P}\Big[\Big|\widehat{\Theta}_k(\mathcal{P}_\Omega(M)) - \mathbb{E}\big[\widehat{\Theta}_k(\mathcal{P}_\Omega(M))\big]\Big| \geq \lambda\Big]\\ \leq\;& e^2 \max\left\{e^{\frac{-\lambda^2}{\text{Var}[\widehat{\Theta}_k(\mathcal{P}_\Omega(M))]R^q}}, \max_{t\in[q]} e^{-(\frac{\lambda}{\mu_t L^t R^q})^{1/t}}\right\},\end{aligned} \tag{6.48}$$

where R is some absolute constant and μ_t is defined as follows:

$$\mu_t \quad \max_{\substack{S\subseteq\{(i,j):i,j\in[d]\}\\ |S|=t}} \left(\sum_{w\in W|w\supseteq S} \frac{c(H(w))}{p(H(w))} |\omega_M(w)| \prod_{(i,j)\in\text{unique}(w)\setminus S} \mathbb{E}[\mathbb{I}((i,j)\in\Omega)] \right), \tag{6.49}$$

where $w \supseteq S$ denotes that the walk w comprises edges(and self loops) contained in the set S. L is defined as follows: A random variable Z is called central moment bounded with real parameter $L > 0$, if for any integer $i \geq 1$ we have

$$E\big[|Z-\mathbb{E}[Z]|^i\big] \;\; \leq \;\; i\,L\,\mathbb{E}[|Z-\mathbb{E}[Z]|^{i-1}]\,. \tag{6.50}$$

For Bernoulli random variables $L \in [1/4, 1]$. In the following, we show that $\mu_t \leq (\mu\sigma_{\max})^k g(k) r^k/(d(dp)^t)$, for $t \in [k]$. Using Lemma 6.15, along with $\|M\|_k^k \geq r(\sigma_{\min})^k$, the bound in (6.12) follows immediately.

To compute μ_t, define a set of walks $W_{\ell,s,\hat{s}}$ such that $w \in W_{\ell,s,\hat{s}}$ has $0 \leq \ell \leq k$ unique edges and $0 \leq s \leq k$ unique self loops, and $\hat{s}$ total self loops with $\ell + \hat{s} \leq k$. For the set S as required in (6.49), let $S_{\tilde{\ell},\tilde{s}}$ be a set of $\tilde{\ell}$ unique edges and $\tilde{s}$ unique self loops, with $|S_{\tilde{\ell},\tilde{s}}| = \tilde{\ell} + \tilde{s}$ where $1 \leq \tilde{\ell} + \tilde{s} \leq k$.

Therefore, we have

$$
\begin{aligned}
&\mu_t \\
&= \max_{\substack{S_{\tilde{\ell},\tilde{s}} \\ :\tilde{\ell}+\tilde{s}=t}} \left(\sum_{\substack{0\leq s\leq \hat{s}\leq k \\ \ell\in[k]:\ell+\hat{s}\leq k}} \sum_{\substack{w\in W_{\ell,s,\hat{s}} \\ :w\supseteq S_{\tilde{\ell},\tilde{s}}}} \frac{c(H(w))}{p(H(w))}\, |\omega_M(w)| \prod_{(i,j)\in\text{unique}(w)\setminus S_{\tilde{\ell},\tilde{s}}} \mathbb{E}[\mathbb{I}((i,j)\subseteq\Omega)] \right) \\
&\leq \max_{\substack{S_{\tilde{\ell},\tilde{s}} \\ :\tilde{\ell}+\tilde{s}=t}} \left(\sum_{\substack{0\leq s\leq \hat{s}\leq k \\ \ell\in[k]:\ell+\hat{s}\leq k}} \sum_{\substack{w\in W_{\ell,s,\hat{s}} \\ :w\supseteq S_{\tilde{\ell},\tilde{s}}}} \frac{f(k)}{p^{\ell+s}}\, \alpha^k r^{\hat{s}/2} p^{\ell+s-(\tilde{\ell}+\tilde{s})} \right) \\
&\leq \max_{\substack{S_{\tilde{\ell},\tilde{s}} \\ :\tilde{\ell}+\tilde{s}=t}} \left(\sum_{\substack{0\leq s\leq \hat{s}\leq k \\ \ell\in[k]:\ell+\hat{s}\leq k,\tilde{s}\leq s}} \frac{d^{\ell-(1+\tilde{\ell})} f(k)}{p^{\ell+s}}\, \frac{(\mu\sigma_{\max})^k r^{(k+\hat{s})/2}}{d^k} p^{\ell+s-(\tilde{\ell}+\tilde{s})} \right) \\
&= \max_{\substack{S_{\tilde{\ell},\tilde{s}} \\ :\tilde{\ell}+\tilde{s}=t}} \left(\sum_{\substack{0\leq s\leq \hat{s}\leq k \\ \ell\in[k]:\ell+\hat{s}\leq k,\tilde{s}\leq s}} \frac{f(k)(\mu\sigma_{\max})^k r^{(k+\hat{s})/2}}{dd^{(k-\ell-\tilde{s})}(dp)^{(\tilde{\ell}+\tilde{s})}} \right) \\
&\leq \max_{\substack{S_{\tilde{\ell},\tilde{s}} \\ :\tilde{\ell}+\tilde{s}=t}} \left(\frac{k^3 f(k)(\mu\sigma_{\max})^k r^{(k+\hat{s})/2}}{dd^{(k-\ell-\tilde{s})}(dp)^{(\tilde{\ell}+\tilde{s})}} \right) \;\leq\; \frac{(\mu\sigma_{\max})^k g(k) r^k}{d(dp)^t}\,.
\end{aligned}
$$

6.8.3 Proof of Theorem 6.11

The proof technique is a generalization to a rank r symmetric matrix of the proof given by [127] for deriving lower bound on the size of a random bi-linear sketch needed for approximating Schatten norm of any matrix. It also draws on the techniques used in [9] for proving a lower bound on the size of the linear sketches of moments.

We prove Theorem 6.11 for an arbitrary fixed relabeling permutation π of the graph nodes. Indeed, by Yao's minimax principle, it suffices to give two distributions on matrix $M\in\mathcal{M}_r$ for which the $\|M\|_k$ values differ by a constant factor with high probability, but for any relabeling permutation π of the nodes of the pattern graph G, the induced distributions on the sampled entries $\mathcal{P}_\Omega(M)$ corresponding to the relabeled graph $G_\pi(\widetilde{V},\Omega)$, have low total variation distance.

For positive $C>0$ to be specified later, define $\lambda\equiv Cdr^{1/k-1/2}$. We construct distributions $\mathcal{D}_1$ and $\mathcal{D}_2$ for $M\in\mathcal{M}_{r,\mu}$ with $\mu=C'\sqrt{\log r}$, for some absolute constant C', such that the following holds:

1. $\|M\|_k \leq \lambda$ on the entire support of $\mathcal{D}_1$, and $\|M\|_k \geq 4\lambda$ on the entire support of $\mathcal{D}_2$.

2. Let $\mathcal{E}_1$ and $\mathcal{E}_2$ denote the distribution of the sampled matrix $\mathcal{P}_\Omega(M)$ when M is drawn from $\mathcal{D}_1$ and $\mathcal{D}_2$ respectively. Recall that Ω is the set of edges of the relabeled graph $G_\pi(\widetilde{V}, \Omega)$ as defined in Section 6.4.1. If $\lambda^*_{G,r} \geq \lambda$ then, the total variation distance between $\mathcal{E}_1$ and $\mathcal{E}_2$ is bounded by $\mathrm{TV}(\mathcal{E}_1, \mathcal{E}_2) \leq 1/2$.

The desired result (6.28) follows from the above claims and the following relationship between statistical tests and estimators:

$$\begin{aligned}
&\mathbb{P}_{M\sim\frac{1}{2}(\mathcal{D}_1+\mathcal{D}_2)}\left(\frac{1}{2}\|M\|_k \leq \widetilde{\Theta}(\mathcal{P}_{\Omega(M)}) \leq 2\|M\|_k\right) \\
&\leq \frac{1}{2}\mathbb{P}_{M\sim\mathcal{D}_2}\left(\widetilde{\Theta}(\mathcal{P}_{\Omega(M)}) \geq 2\lambda\right) + \frac{1}{2}\mathbb{P}_{M\sim\mathcal{D}_1}\left(\widetilde{\Theta}(\mathcal{P}_{\Omega(M)}) \leq 2\lambda\right) \qquad (6.51) \\
&\leq \frac{1}{2}\big(1 + \mathrm{TV}(\mathcal{E}_1, \mathcal{E}_2)\big) \;\; \leq \;\; \frac{3}{4}\,, \qquad (6.52)
\end{aligned}$$

where the last inequality follows from the following characterization of the total variation distance $\mathrm{TV}(\mathcal{E}_1, \mathcal{E}_2) \equiv \sup_A |\mathcal{E}_1(A) - \mathcal{E}_2(A)|$.

To prove the two claims, we construct one of the desired rank-r random matrix via tiling, i.e. covering the matrix with copies of a single $r \times r$ sub-matrix from the *Gaussian Wigner Ensemble*, where diagonals and off-diagonals(upper triangle) are both distributed as i.i.d. standard Gaussians. Another one is constructed by adding a rank one perturbation. Precisely, we define a random matrix drawn from $\mathcal{D}_1$ as follows.

A random $r \times r$ matrix Z chosen from Gaussian Wigner Ensemble, $\mathcal{G}(r, r)$, is a symmetric matrix whose entries $Z_{i,i}$ and $Z_{i,j}$ for $i < j$ are independent with $N(0, 1)$ distribution. Define $B \equiv \mathbb{1}_{\lceil d/r \rceil}\mathbb{1}^\top_{\lceil d/r \rceil}$ to be an all-ones matrix of size $\lceil d/r \rceil \times \lceil d/r \rceil$. Let $\bar{\mathcal{D}}_1$ denote the distribution of $M_1 = Y \otimes B$ where $Y \sim \mathcal{G}(r, r)$, and $\otimes$ denotes the standard Kronecker product of two matrices. Note that the matrix norm of M_1 and Y are related by $\|M_1\|_k = \lceil d/r \rceil \|Y\|_k$. Since the Schatten norm of $Y \sim \mathcal{G}(r, r)$ takes value on the entire $\mathbb{R}_+$, we need to truncate it. We set $\mathcal{D}_1$ to be $\bar{\mathcal{D}}_1$ conditioned on the event $S_1 = \{M_1 : \|M_1\|_k \leq \lambda, \mu(M_1) \leq C'\sqrt{\log r}\}$, i.e. $\mathcal{D}_1(A) = \bar{\mathcal{D}}_1(A \cap S_1)/\bar{\mathcal{D}}_1(S_1)$.

We define $\bar{\mathcal{D}}_2$ by adding a rank one perturbation. Precisely, let $M_2 = M_1 + (5/d)\lambda U$, where $M_1 \sim \bar{\mathcal{D}}_1$ and $U = uu^\top \otimes B$. Here a random vector

$u \in \{\pm 1\}^r$ is a vector of i.i.d. Rademacher random variables. Note that U is a rank one matrix and $\|U\|_k = \lceil d/r \rceil \|uu^\top\|_k = d$. We set $\mathcal{D}_2$ to be $\bar{\mathcal{D}}_2$ conditioned on the event $S_2 = \{M_2 : \|M_2\|_k \geq 4\lambda, \mu(M_2) \leq C'\sqrt{\log r}\}$. Observe that $M_1 \sim \bar{\mathcal{D}}_1$ and $M_2 \sim \bar{\mathcal{D}}_2$ belong to $\mathbb{R}^{d\times d}$, are symmetric and both are rank at most $r+1$.

Let $\bar{\mathcal{E}}_1$ and $\bar{\mathcal{E}}_2$ denote the distribution of $\mathcal{P}_\Omega(M)$ when M is drawn from $\bar{\mathcal{D}}_1$ and $\bar{\mathcal{D}}_2$ respectively. We first show that their total variation distance is not too large. Using the triangle inequality, we have

$$\begin{aligned}
\mathrm{TV}(\mathcal{E}_1, \mathcal{E}_2) &\leq \mathrm{TV}(\bar{\mathcal{E}}_1, \bar{\mathcal{E}}_2) + \mathrm{TV}(\bar{\mathcal{E}}_1, \mathcal{E}_1) + \mathrm{TV}(\bar{\mathcal{E}}_2, \mathcal{E}_2) \\
&\leq \mathrm{TV}(\bar{\mathcal{E}}_1, \bar{\mathcal{E}}_2) + \mathrm{TV}(\bar{\mathcal{D}}_1, \mathcal{D}_1) + \mathrm{TV}(\bar{\mathcal{D}}_2, \mathcal{D}_2) \qquad (6.53) \\
&= \mathrm{TV}(\bar{\mathcal{E}}_1, \bar{\mathcal{E}}_2) + \mathbb{P}_{M_1 \sim D_1}\big((\|M_1\|_k \geq \lambda) \cup (\mu(M_1) \geq C'\sqrt{\log r})\big) \\
&\quad + \mathbb{P}_{M_2 \sim \mathcal{D}_2}\big((\|M_2\|_k \leq 4\lambda) \cup (\mu(M_2) \geq C'\sqrt{\log r})\big)\,, \qquad (6.54)
\end{aligned}$$

where (6.53) follows from the data processing inequality and (6.54) follows from $\mathrm{TV}(\mathcal{E}_1, \mathcal{E}_2) \equiv \sup_A |\mathcal{E}_1(A) - \mathcal{E}_2(A)|$. We next show that the three terms in (6.54) are sufficiently small.

We first provide an upper bound on $\mathrm{TV}(\bar{\mathcal{E}}_1, \bar{\mathcal{E}}_2)$. As per our construction, only the upper triangular (including diagonals) of the upper-left submatrix of size $r \times r$ of $M_1 \sim \mathcal{D}_1$ and $M_2 \sim \mathcal{D}_2$ has unique entries and the rest are copies of these. Observe that the set of unique entries of M_1(or M_2) corresponding to any pattern graph $G(V,E)$ are precisely the following entries of the projection graph $\mathcal{P}^{(r)}(G)$ that is defined in Section 6.4.1:

$$E(\mathcal{P}^{(r)}(G)) \;\equiv\; \Big\{(i,j) : i \leq j \in [r], (i,j) \in \mathcal{P}^{(r)}(G(V,E))\Big\}\,. \qquad (6.55)$$

For the purpose of computing the total variation distance $\mathrm{TV}(\bar{\mathcal{E}}_1, \bar{\mathcal{E}}_2)$, it is sufficient to consider only $E(\mathcal{P}^{(r)}(G_\pi))$ entries of M_1 distributed as i.i.d. standard Gaussians $N(0, I_{\ell_1\times\ell_1})$, and the entries of M_2 distributed as $N(W, I_{\ell_1\times\ell_1}))$, where $\ell_1 = |E(\mathcal{P}^{(r)}(G_\pi))|$. The random vector W represents the rank one perturbation and is distributed as

$$W_{i,j} = (5/d)\lambda\, u_i u_j\,, \qquad (i,j) \in E(\mathcal{P}^{(r)}(G_\pi))\,. \qquad (6.56)$$

To bound total variation distance between $\bar{\mathcal{E}}_1$ and $\bar{\mathcal{E}}_2$, we use the following lemma and the fact that for any two distributions μ and ν, $\mathrm{TV}(\mu,\nu) \leq$

$\sqrt{\mathcal{X}^2(\mu \,\|\, \nu)}$. Let $\mu * \nu$ denote the convolution of the density (or equivalently addition of the two random variables).

Lemma 6.16 ([95], p97)**.** *It holds that* $\mathcal{X}^2(N(0, I_n) * \mu \,\|\, N(0, I_n)) \leq \mathbb{E}\exp(\langle z, z' \rangle) - 1$*, where* $z, z' \sim \mu$ *are independent.*

It follows that

$$\mathrm{TV}(\bar{\mathcal{E}}_1, \bar{\mathcal{E}}_2) \;\leq\; \sqrt{\mathbb{E}e^{\langle W, W' \rangle} - 1} \;\leq\; 1/5\,,$$

for $\lambda^*_G \geq \lambda$ where the expectation is taken over independent W and W' which are identically distributed. We show that if $\lambda^*_G \geq \lambda$ the last inequality holds, as following:

$$
\begin{aligned}
&\mathbb{E}_{W,W'} \exp\big(\langle W, W'\rangle\big) \\
&= \mathbb{E}_{u,u'} \exp\left((5/d)^2\lambda^2 \sum_{(i,j)\in E(\mathcal{P}^{(r)}(G_\pi))} u_i u_i' u_j u_j'\right) \\
&= \mathbb{E}_{u} \exp\left((5/d)^2\lambda^2 \sum_{(i,j)\in E(\mathcal{P}^{(r)}(G_\pi))} u_i u_j\right) && (6.57) \\
&= \mathbb{E}_{u}\Bigg[\exp\left((5/d)^2\lambda^2 \sum_{\substack{(i,j)\in E(\mathcal{P}^{(r)}(G_\pi))\\ :i\neq j}} u_i u_j\right)\Bigg] \\
&\qquad \exp\left((5/d)^2\lambda^2 \sum_{\substack{(i,j)\in E(\mathcal{P}^{(r)}(G_\pi))\\ :i=j}} u_i u_j\right) \\
&\leq \mathbb{E}_{u}\Bigg[\exp\left((5/d)^2\lambda^2 \sum_{\substack{(i,j)\in E(\mathcal{P}^{(r)}(G_\pi))\\ :i\neq j}} 2u_i u_j\right)\Bigg] \\
&\qquad \exp\left((5/d)^2\lambda^2 \sum_{\substack{(i,j)\in E(\mathcal{P}^{(r)}(G_\pi))\\ :i=j}} u_i u_j\right) && (6.58) \\
&= \mathbb{E}_{u}\Bigg[\exp\left((5/d)^2\lambda^2 \sum_{\substack{(i,j)\in \mathcal{P}^{(r)}(G_\pi)\\ :i\neq j}} u_i u_j\right)\Bigg] \\
&\qquad \exp\left((5/d)^2\lambda^2 \sum_{\substack{(i,j)\in \mathcal{P}^{(r)}(G_\pi)\\ :i=j}} u_i u_j\right) && (6.59) \\
&= \mathbb{E}_{u}\Bigg[\exp\left((5/d)^2\lambda^2 \sum_{(i,j)\in \mathcal{P}^{(r)}(G_\pi)} u_i u_j\right)\Bigg] \\
&\leq 1 + 1/25\,, && (6.60)
\end{aligned}
$$

where (6.57) follows from the fact that u, u' are i.i.d. Rademacher variables, (6.58) follows from the fact that $f_{G,r}(\lambda)$ defined in (6.26) is non-decreasing in λ, (6.59) follows from the definition of $E(\mathcal{P}^{(r)}(G_\pi))$ in (6.55),and (6.60) follows from the definition of λ^*_G in (6.27).

To bound the other two terms in (6.54), we use Wigner's semicircular law and its rate of convergence for Gaussian Wigner Ensemble, $\mathcal{G}(r, r)$ as defined above. Consider the empirical spectral distribution of $Z \in \mathbb{R}^{r\times r}$ as

$$F_Z(x) = \frac{1}{r}|\{i : \lambda_i(Z) \le x\}|. \tag{6.61}$$

Lemma 6.17 ([209])**.** *Define $Z = (1/\sqrt{r})Y$ for $Y \sim \mathcal{G}(r, r)$. Then as $r \to \infty$ the empirical distribution $F_Z(x)$ of Z converges weakly to the distribution $G(x)$ with density*

$$g(t) = \frac{\sqrt{4-t^2}}{2\pi} \;\; t \in [-2, 2]\,. \tag{6.62}$$

Lemma 6.18 ([81])**.** *For any positive constant $\alpha > 0$, let $\ell_{r,\alpha} = \log r(\log\log r)^\alpha$. There exists an absolute positive constant C and c such that for r large enough,*

$$\mathbb{P}\Big\{ \sup_x \big|F_Z(x) - G(x)\big| \ge r^{-1}\log r \ell_{r,\alpha}^6 \Big\} \;\le\; C\exp\big\{-c\ell_{r,\alpha}\big\}\,. \tag{6.63}$$

To bound the schatten norm of a matrix $Y \sim \mathcal{G}(r, r)$, along with Lemma 6.17 and Lemma 6.18 we use the following. If $F(x)$ and $G(x)$ are cumulative distribution functions of densities μ, ν then for any continuous and bounded function f, we have

$$\left| \int f d\mu - \int f d\nu \right| \;\le\; \|f\|_\infty \sup_x \big|F(x) - G(x)\big|\,. \tag{6.64}$$

Choosing $f(x) = x^k$ for $x \in [-2, 2]$, we can see that for $k = O(\log r)$ there exists a constant $C > 2$ such that with probability $1 - 1/80$ it holds that

$$\|(1/\sqrt{r})Y\|_k^k = \left(\int_{-2}^{2} x^k \frac{\sqrt{4-x^2}}{2\pi} dx + o(1) \right) r \;\le\; (2^k + o(1))r \;\le\; C^k r. \tag{6.65}$$

Hence $\|Y\|_k \le Cr^{(1/k+1/2)}$. By construction of distribution $\bar{\mathcal{D}}_1$, for $M_1 \sim \bar{\mathcal{D}}_1$, $\|M_1\|_k = (d/r)\|Y\|_k \le Cdr^{(1/k-1/2)} = \lambda$. Also, by construction $M_2 \sim \bar{\mathcal{D}}_2$ is $M_2 = M_1 + (5/d)\lambda U$ where $\|U\|_k = d$. Using triangle inequality, we have

$$\begin{aligned} \|M_2\|_k &\ge \|(5/d)\lambda U\|_k - \|M_1\|_k \\ &\ge 5\lambda - Cdr^{1/k-1/2} \;=\; 4\lambda\,, \end{aligned}$$

Recall that, incoherence parameter $\mu(M)$ is defined as $\mu(M) =$

$\max_{i\neq j\in[d]} M_{i,j}/(|\sigma_{\max}(M)|\sqrt{r}/d)$. From (6.65), there exists a constant $0 < C' < 1$ such that with probability $1-1/160$ it holds that $\|Y\|_2 \geq C'r$. The integral evaluates to 1 for $k=2$. Therefore, the largest singular value of M_1 is lower bounded: $|\sigma_{\max}(M_1)| \geq C'd/\sqrt{r}$. Using the fact that there exists a constant C'' such that $\max_{i,j\in[r]}\{Y_{i,j}\} \leq C''\sqrt{\log r}$ with probability at least $1-1/160$, we have, $\mu(M_1) \leq (C''/C')\sqrt{\log r}$. The same $\mu(M_1)$ satisfies the upper bound on diagonals as well. Therefore, using union bound, the second and the third term in (6.54) are upper bounded by 1/40.

6.8.4 Proof of Lemma 6.12

Observe that for any given permutation π, $\mathcal{P}^{(r)}(G_\pi)$ as defined in Section 6.4.1 is a clique over a subset of nodes $\widetilde{V}_\pi$, where $|\widetilde{V}_\pi| \leq \min\{\ell, r\}$. From the definition of $f_{G,r}(\lambda)$, (6.26), we have the following:

$$\begin{aligned}
f_{G,r}(\lambda) &= \max_\pi \left\{ \mathbb{E}_u \exp\left((5/d)^2\lambda^2 \sum_{(i,j)\in\mathcal{P}^{(r)}(G_\pi)} u_i u_j \right) \right\} \\
&= \max_\pi \left\{ \mathbb{E}_u \exp\left((5/d)^2\lambda^2 \big(\sum_{i\in\widetilde{V}_\pi} u_i\big)^2 \right) \right\} \\
&= \max_\pi \left\{ \sum_{t=0}^{\infty} \frac{(5/d)^{2t}\lambda^{2t}\mathbb{E}_u\big[\big(\sum_{i\in\widetilde{V}_\pi} u_i\big)^{2t}\big]}{t!} \right\} \\
&\leq \max_\pi \left\{ \left(1 + 2\sum_{t=1}^{\infty} \big((5/d)^2\lambda^2|\widetilde{V}_\pi| \big)^t \right) \right\},
\end{aligned}$$

where the inequality follows from the bound in (6.66). Therefore, from the definition of $\lambda^*_{G,r}$, we have that $\lambda^*_{G,r}$ is upper bounded by $2^{-4}d(\min\{\ell,r\})^{-1/2}$.

To bound $\mathbb{E}(\sum_{i\in\widetilde{V}_\pi} u_i)^{2t}$, for $t \in [1,\infty)$, using Hoeffding bound we have that

$$\begin{aligned}
\mathbb{E}\Big|\sum_{i\in\widetilde{V}_\pi} u_i\Big|^{2t} &= \int_0^{|\widetilde{V}_\pi|^{2t}} \mathbb{P}\Big(\Big|\sum_{i\in\widetilde{V}_\pi} u_i\Big|^{2t} \geq z \Big) dz \\
&\leq 2\int_0^{|\widetilde{V}_\pi|^{2t}} \exp\left(\frac{-z^{1/t}}{2|\widetilde{V}_\pi|} \right) dz \;\leq\; 2(2|\widetilde{V}_\pi|)^t t!\,, \qquad (6.66)
\end{aligned}$$

where the integral is evaluated by variable substitution.

6.8.5 Proof of Lemma 6.14

For the given pattern graph G and any given permutation π, let $\widetilde{A}_\pi \in \{0,1\}^{r\times r}$ be the adjacency matrix of the graph $\mathcal{P}^{(r)}(G_\pi)$ that is defined in Section 6.4.1. Observe that for a permutation π, ℓ_π rows of $\widetilde{A}_\pi$ are all-ones and the remaining are all-zeros, where $\ell_\pi \leq \min\{\ell, r\}$. Let A_π be a copy of $\widetilde{A}_\pi$ where all the diagonal entries are replaced with zero. Note that $\mathbb{E}_u(u^\top A_\pi u)^{2t+1} = 0$ for all $t \geq 0$, where u_i's are i.i.d. Rademacher random variables. Define $C_\pi \equiv \exp((5/d)^2\lambda^2\ell_\pi)$.

From the definition of $f_{G,r}(\lambda)$, (6.26), we have the following:

$$\begin{aligned} f_{G,r}(\lambda) &= \max_\pi \left\{ \mathbb{E}_u \exp\left((5/d)^2\lambda^2 \sum_{(i,j)\in\mathcal{P}^{(r)}(G_\pi)} u_i u_j \right) \right\} \\ &= \max_\pi \left\{ C_\pi \mathbb{E}_u \exp\left((5/d)^2\lambda^2 (u^\top A_\pi u) \right) \right\} \\ &= \max_\pi \left\{ C_\pi \sum_{t=0}^{\infty} \frac{(5/d)^{4t}\lambda^{4t}\mathbb{E}_u\big[(u^\top A_\pi u)^{2t}\big]}{(2t)!} \right\} \\ &\leq \max_\pi \left\{ C_\pi \left(1 + 4\sum_{t=1}^{\infty} \left(2c(5/d)^2\lambda^2\sqrt{\ell_\pi r}\right)^{2t} \right) \right\}, \end{aligned}$$

where the inequality follows from the bound in (6.67), and c is some absolute constant. Therefore, from the definition of $\lambda^*_{G,r}$, we have that $\lambda^*_{G,r}$ is upper bounded by $cd((\min\{\ell,r\})r)^{-1/4}$.

To bound $\mathbb{E}_u\big[(u^\top A_\pi u)^{2t}\big]$, for $t \in [1,\infty)$, we use Hanson-Wright Inequality. Observe that $\|A_\pi\|_2 \leq \sqrt{\ell_\pi r}$, and $\|A_\pi\|_{\mathrm{F}}^2 = (r-1)\ell_\pi < \ell_\pi r$.

$$\begin{aligned} &\mathbb{E}_u\big[(u^\top A_\pi u)^{2t}\big] \\ =\;& \int_0^{(2\sqrt{r\ell_\pi})^{2t}} \mathbb{P}\big((u^\top A_\pi u)^{2t} \geq z\big)dz \;+\; \int_{(2\sqrt{r\ell_\pi})^{2t}}^{(\ell_\pi r)^{2t}} \mathbb{P}\big((u^\top A_\pi u)^{2t} \geq z\big)dz \\ \leq\;& \int_0^{(2\sqrt{r\ell_\pi})^{2t}} \exp\left(\frac{-cz^{1/t}}{4\ell_\pi r}\right)dz + \int_{(2\sqrt{r\ell_\pi})^{2t}}^{(\ell_\pi r)^{2t}} \exp\left(\frac{-cz^{1/(2t)}}{2\sqrt{\ell_\pi r}}\right)dz \\ \leq\;& 2(4\ell_\pi r/c)^t t! + 2(2\sqrt{\ell_\pi r}/c)^{2t}(2t)! \;\leq\; 4(2\sqrt{\ell_\pi r}/c)^{2t}(2t)!\,, \end{aligned} \tag{6.67}$$

where the integral is evaluated by variable substitution.

6.8.6 Proof of Theorem 6.13

For a clique of size m selected uniformly at random, we derive an upper bound on variance of our estimator. Following the notations defined in the proof of Theorem 6.3, we have the following bound on the variance.

$$
\begin{aligned}
&\mathrm{Var}\big(\widehat{\Theta}_k(\mathcal{P}_\Omega(M))\big) \\
&= 2\sum_{\ell=0}^{k}\sum_{\substack{w\neq w'\in\widetilde{W}\\ |w\cap w'|=\ell}} \mathrm{Covar}\left(\frac{\mathbb{I}(w\subseteq\Omega)\omega_M(w)c(H(w))}{p(H(w))}, \frac{\mathbb{I}(w'\subseteq\Omega)\omega_M(w')c(H(w'))}{p(H(w'))}\right) \\
&+\ \sum_{i=1}^{4}\sum_{H\in\mathcal{H}_{k,i}}\left\{\frac{c(H)^2}{p(H)^2}\sum_{w:H(w)=H}\omega_M(w)^2\mathrm{Var}\Big(\mathbb{I}(w\subseteq\Omega)\Big)\right\} \qquad (6.68)\\
&< 2\sum_{\ell=0}^{k}\sum_{\substack{w\neq w'\in W\\ |w\cap w'|=\ell}} \mathbb{E}\Big[\mathbb{I}(w\subseteq\Omega)\mathbb{I}(w'\subseteq\Omega)\Big]\left(\frac{\omega_M(w)\,\omega_M(w')c(H(w))c(H(w'))}{p(H(w))\,p(H(w'))}\right) \\
&- 2\sum_{\ell=0}^{k}\sum_{\substack{w\neq w'\in W\\ |w\cap w'|=\ell}} \mathbb{E}\Big[\mathbb{I}(w\subseteq\Omega)\Big]\mathbb{E}\Big[\mathbb{I}(w'\subseteq\Omega)\Big]\left(\frac{\omega_M(w)\,\omega_M(w')c(H(w))c(H(w'))}{p(H(w))\,p(H(w'))}\right) \\
&+\ \sum_{i=1}^{4}\sum_{H\in\mathcal{H}_{k,i}}\sum_{w:H(w)=H}\frac{c(H)^2\omega_M(w)^2}{p(H)^2}\mathbb{E}\Big[\mathbb{I}(w\subseteq\Omega)\Big]\,. \qquad (6.69)
\end{aligned}
$$

where we abuse the earlier defined notation $|w\cap w'|$ to denote the number of overlapping nodes in the two walks $w, w'\in W$ instead of number of overlapping edges. Note that in pattern sampling, covariance term for two walks that do not have any overlapping node is not zero. As earlier, we provide bound on each of the terms in (6.69).

Probability of any walk w being sampled is $\mathbb{P}[w\in\Omega] = \binom{m}{\ell}/\binom{d}{\ell} \leq f(\ell)m^\ell/d^\ell$, where ℓ is the number of unique nodes that the walk traverses and $f(\ell)$ is an exponential function in ℓ. Recall that off diagonals of matrix M are bounded by $|M_{ij}|\leq\alpha=\mu\sigma_{\max}\sqrt{r}/d$ and the diagonals are bounded by $|M_{ii}|\leq\mu\sigma_{\max}r/d$. We have,

$$
\sum_{H\in\mathcal{H}_{k,i}}\sum_{w:H(w)=H}\frac{c(H)^2\,\omega_M(w)^2}{p(H)^2}\mathbb{E}\Big[\mathbb{I}(w\subseteq\Omega)\Big]\ \leq
$$

$$\begin{cases} \frac{d^k}{2k}\frac{f(k)^2\alpha^{2k}d^k}{m^k} \leq \frac{f(k)^2(\mu\sigma_{\max})^{2k}r^k}{m^k}\,, & \text{for } i = 1 \;\;, \quad (6.70) \\ \left(\frac{d^2}{m}\right)^{k-1} f(k)^2\alpha^{2k} = \frac{m}{d^2}\frac{f(k)^2(\mu\sigma_{\max})^{2k}r^k}{m^k}\,, & \text{for } \; i = 2 \;\;, \quad (6.71) \\ \frac{d^2}{m}r^k\alpha^{2k} = \frac{r^k m^{k-1}}{d^{2k-2}}\frac{f(k)^2(\mu\sigma_{\max})^{2k}r^k}{m^k}\,, & \text{for } \; i = 4\,, \quad (6.72) \end{cases}$$

and for $i = 3$ and for $1 \leq s \leq k-1$, we have

$$\sum_{H\in\mathcal{H}_{k,3,s}} \sum_{w:H(w)=H} \frac{c(H)^2\,\omega_M(w)^2}{p(H)^2}\mathbb{E}\Big[\mathbb{I}(w \subseteq \Omega)\Big]$$

$$\leq \quad \left(\frac{d^2}{m}\right)^{k-s} f(k)^2\alpha^{2k}r^s = \frac{m^s r^s}{d^{2s}}\frac{f(k)^2(\mu\sigma_{\max})^{2k}r^k}{m^k}\,, \qquad (6.73)$$

For any two walks w, w' with $\ell \geq 0$ overlapping nodes, $\mathbb{P}[w, w' \in \Omega]/(\mathbb{P}[w \in \Omega]\mathbb{P}[w' \in \Omega]) \leq f(k)d^\ell/m^\ell$. For $w \neq w'$ and $|w \cap w'| = \ell$, where the range of ℓ varies across equations depending upon the set to which w, w' belongs, we have the following:

$$\sum_{\substack{w\neq w'\in W \\ |w\cap w'|=\ell \\ H(w)\in\mathcal{H}_{k,i,s} \\ H(w')\in\mathcal{H}_{k,i',s'}}} \left(\mathbb{E}\Big[\mathbb{I}(w \subseteq \Omega)\mathbb{I}(w' \subseteq \Omega)\Big] - \mathbb{E}\Big[\mathbb{I}(w \subseteq \Omega)\Big]\mathbb{E}\Big[\mathbb{I}(w' \subseteq \Omega)\Big]\right)$$

$$\left(\frac{\omega_M(w)\,\omega_M(w')c(H(w))c(H(w'))}{p(H(w))\,p(H(w'))}\right) \;\leq$$

$$\left\{\begin{array}{lcl}
\frac{f(k)^2 d^\ell}{m^\ell}\frac{(\mu\sigma_{\max})^{2k} r^2}{d^\ell} & = & \frac{f(k)^2(\mu\sigma_{\max})^{2k}\max\{r^2, r^\ell\}}{m^\ell}, \\
 & & \text{for } \ i = i' = 1, \ell \geq 1 \quad (6.74) \\
\frac{m^{2k-1}}{d^{2k}}\frac{d^{2k} f(k)^2(\mu\sigma_{\max})^{2k} r^2}{m^{2k}} & = & \frac{f(k)^2(\mu\sigma_{\max})^{2k} r^2}{m}, \\
 & & \text{for } \ i = i' = 1, \ell = 0 \quad (6.75) \\
\frac{f(k)^2 d^\ell d^{2k-2-\ell}\alpha^{2k}}{m^\ell} & \leq & \frac{f(k)^2(\mu\sigma_{\max})^{2k} r^k}{m^\ell d^2}, \\
 & & \text{for } i = i' = 2 \quad (6.76) \\
\frac{f(k)^2 d^\ell d^{2k-s-s'-\ell}\alpha^{2k}(\sqrt{r})^{s+s'}}{m^\ell} & \leq & \frac{f(k)^2(\mu\sigma_{\max})^{2k} r^k}{m^\ell d}, \\
 & & \text{for } \ i = i' = 3 \quad (6.77) \\
f(k)^2 d^2 \alpha^{2k}(\sqrt{r})^{2k} & \leq & \frac{f(k)^2(\mu\sigma_{\max})^{2k} r^k}{d^{2k-2}/r^k}, \\
 & & \text{for } \ i = i' = 4 \quad (6.78) \\
\frac{f(k)^2 d^\ell d^{2k-1-\ell}\alpha^{2k}}{m^\ell} & \leq & \frac{f(k)^2(\mu\sigma_{\max})^{2k} r^k}{m^\ell d}, \\
 & & \text{for } i = 1, i' = 2 \quad (6.79) \\
\frac{f(k)^2 d^\ell d^{2k-s-\ell}\alpha^{2k}(\sqrt{r})^{s}}{m^\ell} & \leq & \frac{f(k)^2(\mu\sigma_{\max})^{2k} r^k}{m^\ell d/\sqrt{r}}, \\
 & & \text{for } i = 1, i' = 3 \quad (6.80) \\
\frac{f(k)^2 d^\ell d^{k+1-\ell}\alpha^{2k}(\sqrt{r})^{k}}{m^\ell} & \leq & \frac{f(k)^2(\mu\sigma_{\max})^{2k} r^k}{m^\ell d^{k-1}/(\sqrt{r})^k}, \\
 & & \text{for } i = 1, i' = 4\,, \quad (6.81) \\
\frac{f(k)^2 d^\ell d^{2k-1-s-\ell}\alpha^{2k}(\sqrt{r})^{s}}{m^\ell} & \leq & \frac{f(k)^2(\mu\sigma_{\max})^{2k} r^k}{m^\ell d^2/\sqrt{r}}, \\
 & & \text{for } i = 2, i' = 3 \quad (6.82) \\
\frac{f(k)^2 d^\ell d^{k-\ell}\alpha^{2k}(\sqrt{r})^{k}}{m^\ell} & \leq & \frac{f(k)^2(\mu\sigma_{\max})^{2k} r^k}{m^\ell d^{k}(\sqrt{r})^k}, \\
 & & \text{for } i = 2, i' = 4\,, \quad (6.83) \\
\frac{f(k)^2 d^\ell d^{k+1-s-\ell}\alpha^{2k}(\sqrt{r})^{s+k}}{m^\ell} & \leq & \frac{f(k)^2(\mu\sigma_{\max})^{2k} r^k}{m^\ell d^{k-1}/(\sqrt{r})^k}, \\
 & & \text{for } i = 3, i' = 4\,, \quad (6.84)
\end{array}\right.$$

Where (6.74) and (6.75) both use (6.86), and (6.75) also uses (6.85). Note that ℓ is zero in (6.78). Collecting all the terms, and using Chebyshev's inequality, along with $\|M\|_k^k \geq r(\sigma_{\min})^k$, we get the desired result.

For any two disjoint simple cycles $w \neq w' \in \mathcal{H}_{k,1}$ with $|w \cap w'| = 0$, we have the following

$$
\begin{aligned}
& \mathbb{P}\big[w \in \Omega\big] - \mathbb{P}\big[w \in \Omega \;\big|\; w' \in \Omega\big] \\
= \;\; & \frac{\binom{m}{k}}{\binom{d}{k}} - \frac{\binom{m-k}{k}}{\binom{d-k}{k}} \\
\leq \;\; & \frac{m^k}{(d-k+1)^k} - \frac{(m-2k+1)^k}{(d-k)^k} \leq \frac{f(k)m^{k-1}}{d^k}\,,
\end{aligned}
\tag{6.85}
$$

where the last inequality assumes that $k < d/2$.

Lemma 6.19. *For $k = 3$, and any $0 \leq \ell \leq k$*

$$
\sum_{w \neq w' \in \mathcal{H}_{k,1} : |w \cap w'| = \ell} \omega_M(w)\omega_M(w') \;\; \leq \;\; \frac{f(k)(\mu\sigma_{\max})^{2k}\max\{r^2, r^\ell\}}{d^\ell}\,. \tag{6.86}
$$

Although we give a proof for $k = 3$ only, we are intentionally writing the lemma for general k as we expect the lemma holds for all $k \geq 3$. The joint walk $w \neq w' \in \mathcal{H}_{k,1} : |w \cap w'| = \ell$ corresponds to $H(w) = D_{27}$, for $\ell = 1$; and $H(w) = D_{23}$, for $\ell = 2$ in Figure 6.13. Define $\tilde{M} \equiv M - \text{diag}(M)$, and let $\odot$ denote the Hadamard product of two matrices. We have,

$$
\begin{aligned}
& \sum_{w \neq w' \in \mathcal{H}_{k,1} : |w \cap w'| = 2} \omega_M(w)\omega_M(w') \\
= \;\; & (1/4) \sum_{i,j \in [d]} \Big(\big(\tilde{M}^2 \odot \tilde{M}^2 - (\tilde{M} \odot \tilde{M})^2\big) \odot (\tilde{M} \odot \tilde{M})\Big)_{i,j}\,.
\end{aligned}
\tag{6.87}
$$

Let's denote the quantity in (6.87) by C_1, we have,

$$
\begin{aligned}
& \sum_{w \neq w' \in \mathcal{H}_{k,1} : |w \cap w'| = 1} \omega_M(w)\omega_M(w') \\
= \;\; & (1/8) \sum_{i \in [d]} \Big(\text{diag}(\tilde{M}^3) \odot \text{diag}(\tilde{M}^3) - 2\text{diag}((\tilde{M} \odot \tilde{M})^3)\Big)_i - 2C_1\,.
\end{aligned}
\tag{6.88}
$$

It is easy to verify Equation (6.86) for $k = 3$ and $\ell \in \{1, 2\}$ using the fact that M is a μ incoherent symmetric matrix with its off-diagonals bounded by $\mu\sigma_{\max}(\sqrt{r}/d)$. For $\ell = 0$, quantity in (6.86) is the sum of each pair of disjoint triangles. For sum of all triangles, we have,

$$\sum_{w\in\mathcal{H}_{k,1}} \omega_M(w) = (1/6)\sum_{i\in[d]}\Big(\mathrm{diag}(\tilde{M}^3)\Big)_i \;\leq\; (\mu\sigma_{\max})^3 r\,. \tag{6.89}$$

Using Equations (6.87), (6.88) and (6.89), bound for $\ell = 0$ follows immediately. Bound for $\ell = k$, follows by using the fact that $M_{i,j} \leq \mu\sigma_{\max}(\sqrt{r}/d)$ for $i \neq j \in [d]$.

6.9 k-cyclic pseudographs

We provide an enumeration of all k-cyclic psuedographs for $k \in \{4, 5, 6, 7\}$ in Figures (7.2–6.17).

6.10 Efficient computation of $\omega_M(H)$ for $k \in \{4, 5, 6, 7\}$

In this section we provide the complete matrix oeprations for copmuting $\gamma_M(H)$'s. Equations (6.90) - (6.96) give expressions to compute $\gamma_M(H)$ for $H \in \mathcal{H}_4$ as labeled in Figure 7.2. Equations (6.97) - (6.108) give expressions to compute $\gamma_M(H)$ for $H \in \mathcal{H}_5$ as labeled in Figure 6.11.

For expressions to compute $\gamma_M(H)$ for $H \in \mathcal{H}_6$ and $H \in \mathcal{H}_7$ as labeled in Figure 6.13, we refer the reader to MATLAB code available at `https://github.com/khetan2/Schatten_norm_estimation`.

For brevity of notations and readability, we define the following additional notations. Let $A \odot B$ denote the Hadamard product. For $A \in \mathbb{R}^{d\times d}$, let $\mathrm{sum}(A)$ denote a vector $v \in \mathbb{R}^d$ such that $v_i = \sum_{j\in[d]} A_{i,j}$. With a slight abuse of notation, for $v \in \mathbb{R}^d$, let $\mathrm{sum}(v)$ denote sum of all elements of v that is $\mathrm{sum}(v) = \sum_{i\in[d]} v_i$. Let $\mathrm{sum}(\gamma_M(H_i) : \gamma_M(H_j)) \equiv \sum_{i'=i}^{j} \gamma_M(H_{i'})$. Define $R \equiv \mathbb{1}_{d\times d} - \mathrm{diag}(\mathbb{1}_{d\times d})$, that is R is an all-ones matrix except on diagonals which are zeros. Further, for brevity, we omit the subscript M from the notations $\gamma_M(H), O_M$ and D_M.

$$\gamma(B_1) = \mathrm{sum}(\mathrm{sum}(D{\odot}D{\odot}D{\odot}D)) \tag{6.90}$$
$$\gamma(B_2) = \mathrm{sum}(\mathrm{sum}(O{\odot}O{\odot}O{\odot}O)) \tag{6.91}$$
$$\gamma(B_3) = 4\ \mathrm{tr}(O{*}O{*}D{*}D) \tag{6.92}$$
$$\gamma(B_4) = 2\ \mathrm{sum}(\mathrm{sum}((O{\odot}O){*}(O{\odot}O){\odot}R)) \tag{6.93}$$
$$\gamma(B_5) = 2\ \mathrm{tr}(O{*}D{*}O{*}D) \tag{6.94}$$
$$\gamma(B_6) = \mathrm{tr}(O{*}O{*}O{*}O) - \mathrm{sum}(\gamma(B_2) : \gamma(B_4)) \tag{6.95}$$
$$\gamma(B_7) = \mathrm{tr}(M{*}M{*}M{*}M) - \mathrm{sum}(\gamma(B_1) : \gamma(B_6)) \tag{6.96}$$

$$\gamma(C_1) = \mathrm{tr}(D{\odot}D{\odot}D{\odot}D{\odot}D) \tag{6.97}$$
$$\gamma(C_2) = 5\ \mathrm{sum}(\mathrm{sum}(D{*}O{\odot}O{\odot}O{\odot}O)) \tag{6.98}$$
$$\gamma(C_3) = 5\ \mathrm{sum}(\mathrm{sum}((D{\odot}D{\odot}D){*}(O{\odot}O))) \tag{6.99}$$
$$\gamma(C_4) = 5\ \mathrm{tr}((O{\odot}O{\odot}O){*}O{*}O) \tag{6.100}$$
$$\gamma(C_5) = 5\ \mathrm{sum}(\mathrm{sum}(D{*}(O{\odot}O){*}(D{\odot}D))) \tag{6.101}$$
$$\gamma(C_6) = 5\ \mathrm{sum}(\mathrm{sum}(((O{\odot}O){*}D{*}(O{\odot}O)){\odot}R)) \tag{6.102}$$
$$\gamma(C_7) = 5\ \mathrm{sum}(\mathrm{sum}((D{*}(O{\odot}O){*}(O{\odot}O)){\odot}R)) \tag{6.103}$$
$$\gamma(C_8) = 5\ \mathrm{tr}(O{*}O{*}O{*}(D{\odot}D)) \tag{6.104}$$
$$\begin{aligned}\gamma(C_9) &= 5\ \mathrm{sum}(\mathrm{diag}(O{\odot}O{\odot}O){\odot}\mathrm{sum}(O{\odot}O)) \\ &\quad -10\ \mathrm{tr}((O{\odot}O{\odot}O){*}O{*}O))\end{aligned} \tag{6.105}$$
$$\gamma(C_{10}) = \mathrm{tr}(O{*}O{*}O{*}O{*}O) - \gamma(C_4) - \gamma(C_9) \tag{6.106}$$
$$\gamma(C_{11}) = 5\ \mathrm{tr}(O{*}D{*}O{*}D{*}O) \tag{6.107}$$
$$\gamma(C_{12}) = \mathrm{tr}(M{*}M{*}M{*}M{*}M) - \mathrm{sum}(\gamma(C_1) : \gamma(C_{11})) \tag{6.108}$$

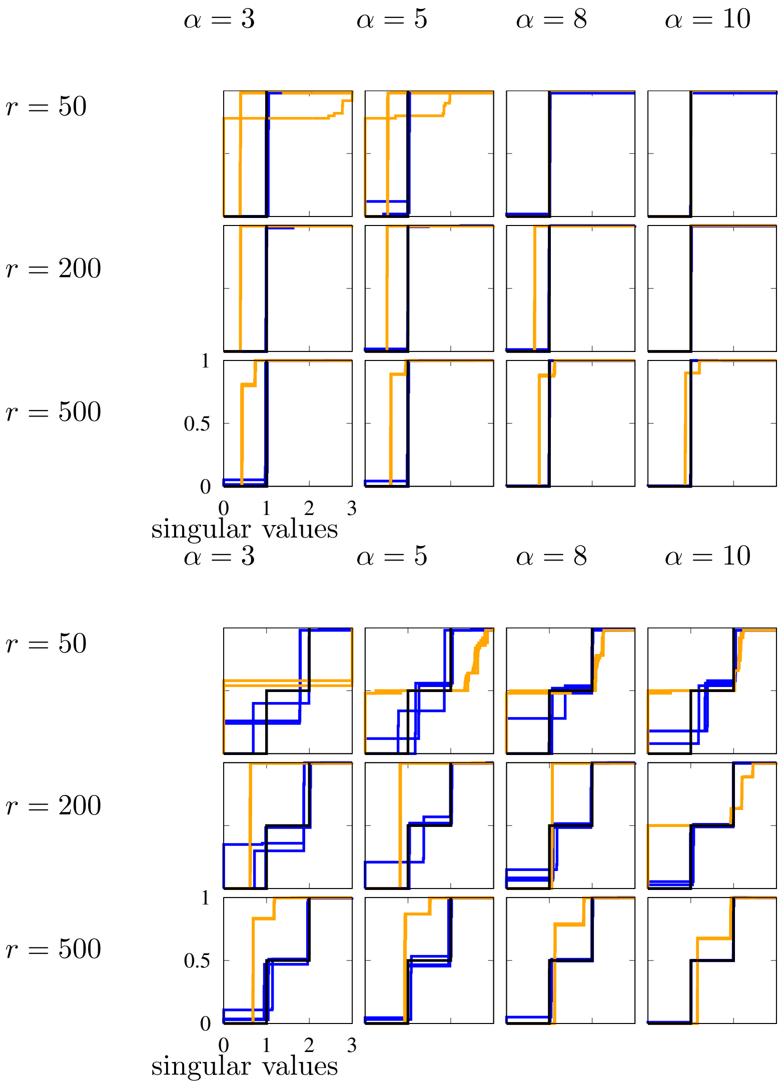


Figure 6.7: The proposed estimator (in blue solid lines) outperforms matrix completion approaches (in orange solid lines) in estimating the ground truths empirical cumulative distribution function of the r strictly positive singular values (in black solid line) for two examples: one peak at $\sigma_i = 1$ on the top and two peaks at $\sigma_i = 1$ or $\sigma_i = 2$ on the bottom.

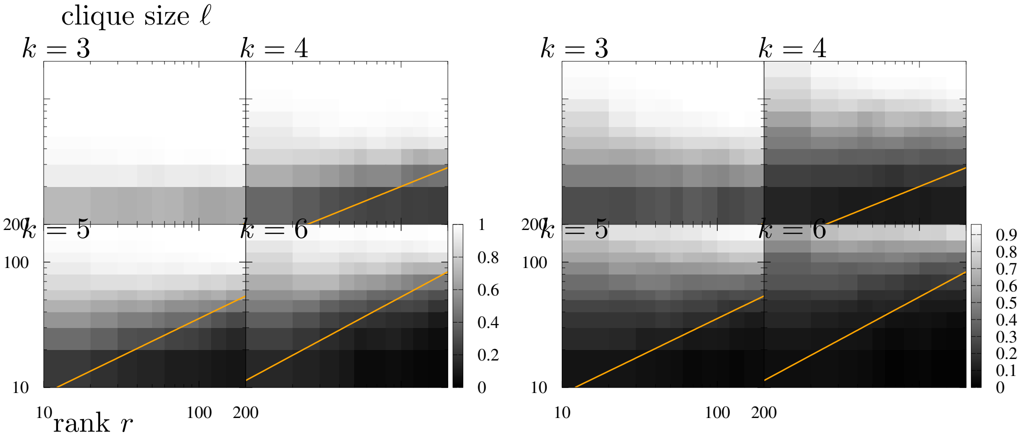


Figure 6.8: Each colormap in each block for $k \in \{3,4,5,6\}$ show empirical probability of the event $\left\{\left|\|M\|_k^k - \widehat{\Theta}_k(\mathcal{P}_\Omega(M))\right|/\|M\|_k^k \leq \delta\right\}$, for $\delta = 0.5$ (left panel) and $\delta = 0.2$ (right panel). Ω is generated by clique sampling of matrix M with a clique of size ℓ (vertical axis). M is a positive semi-definite matrix of size $d = 1000$. The solid lines correspond to our theoretical prediction $\ell = \sqrt{k}r^{1-2/k}$.

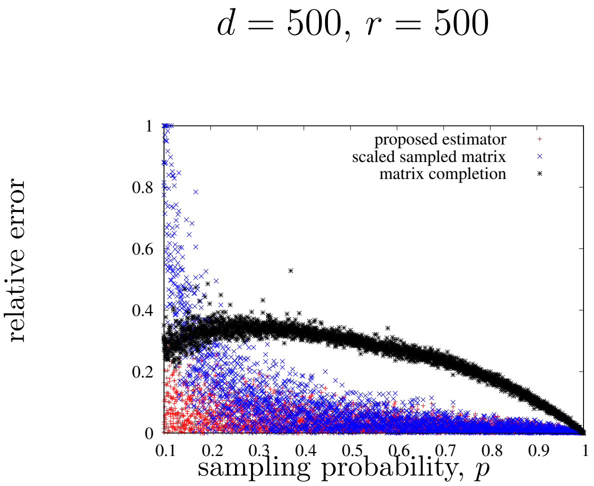


Figure 6.9: For a matrix with a very small effective rank, the gap between the proposed estimator and the simple scaled sampled matrix approach is smaller.

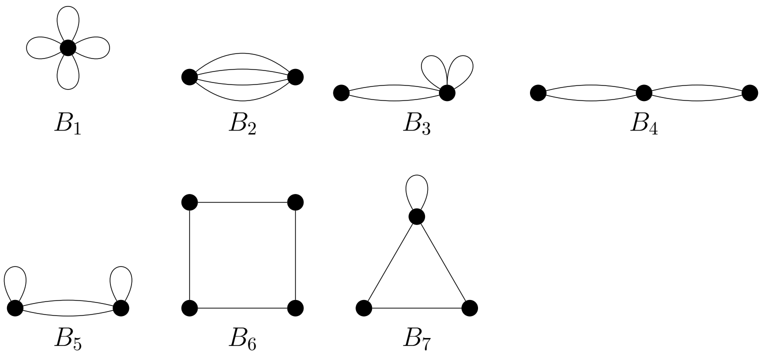


Figure 6.10: The 4-cyclic pseudographs $\mathcal{H}_4$.

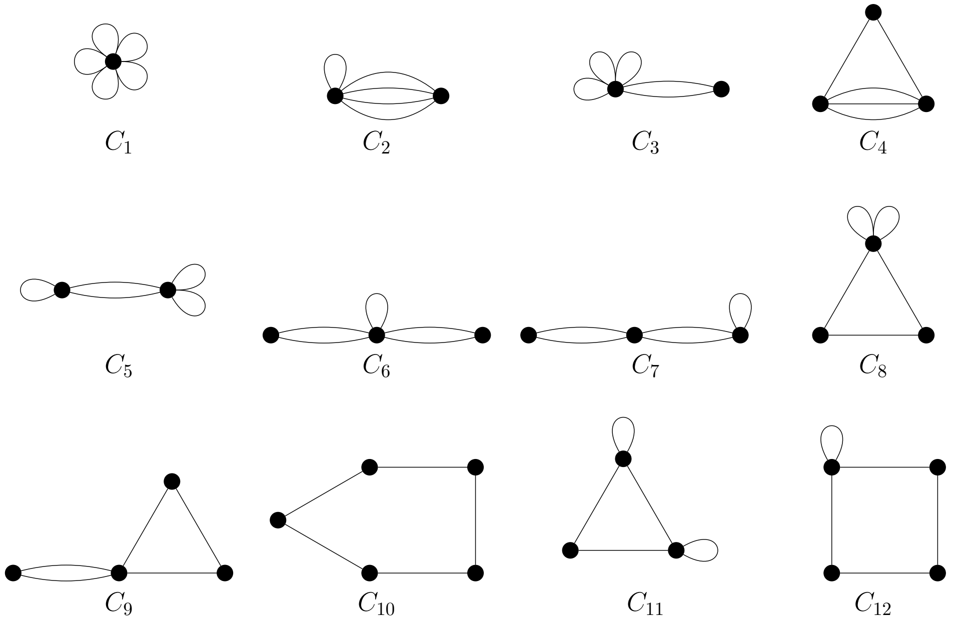


Figure 6.11: The 5-cyclic pseudographs $\mathcal{H}_5$.

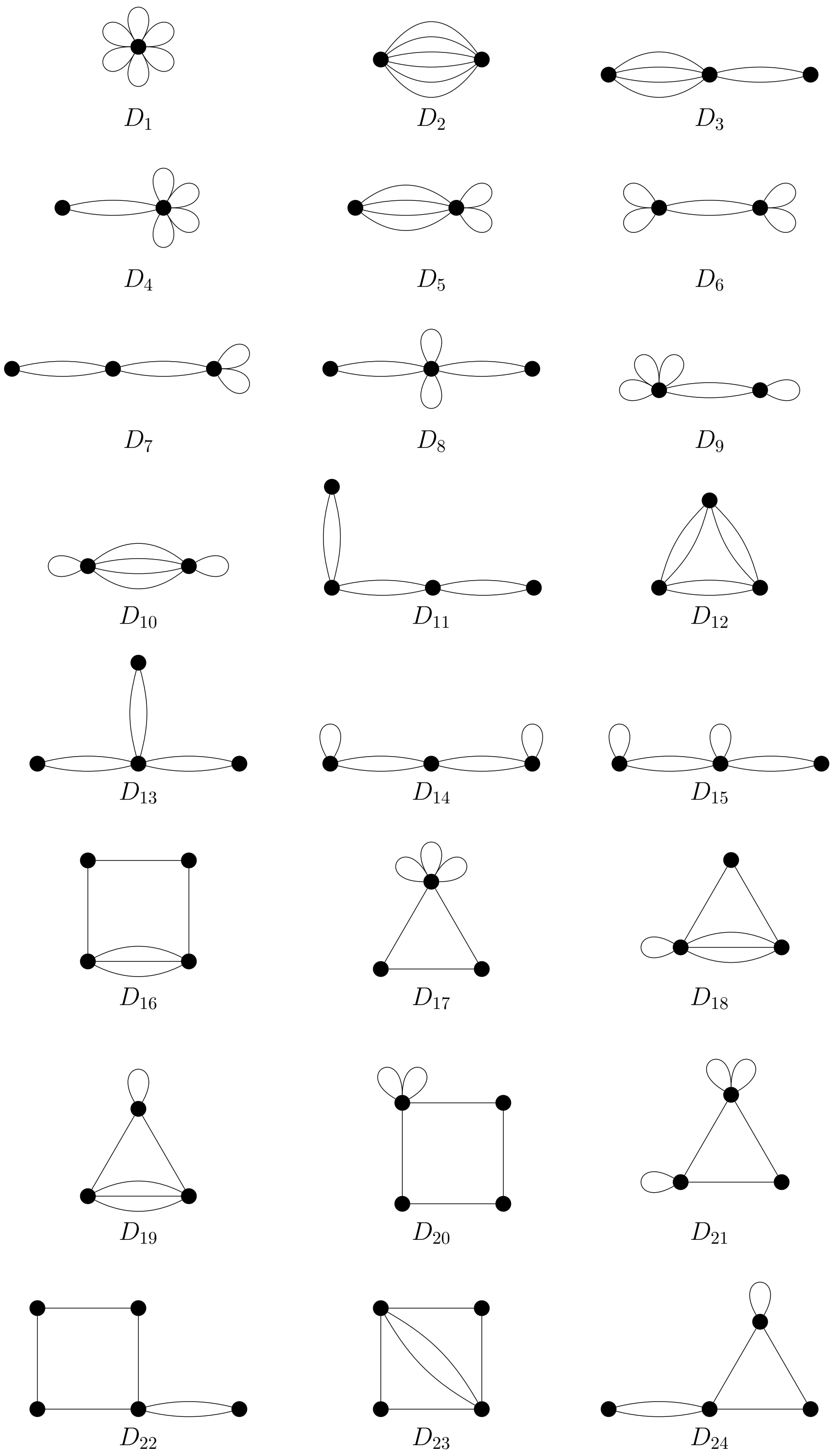


Figure 6.12: The 6-cyclic pseudographs $\mathcal{H}_6$.

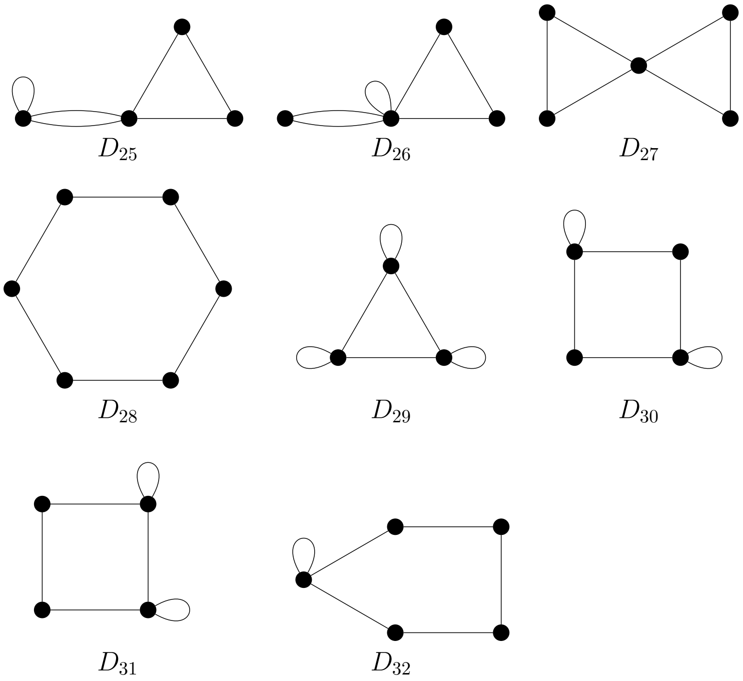
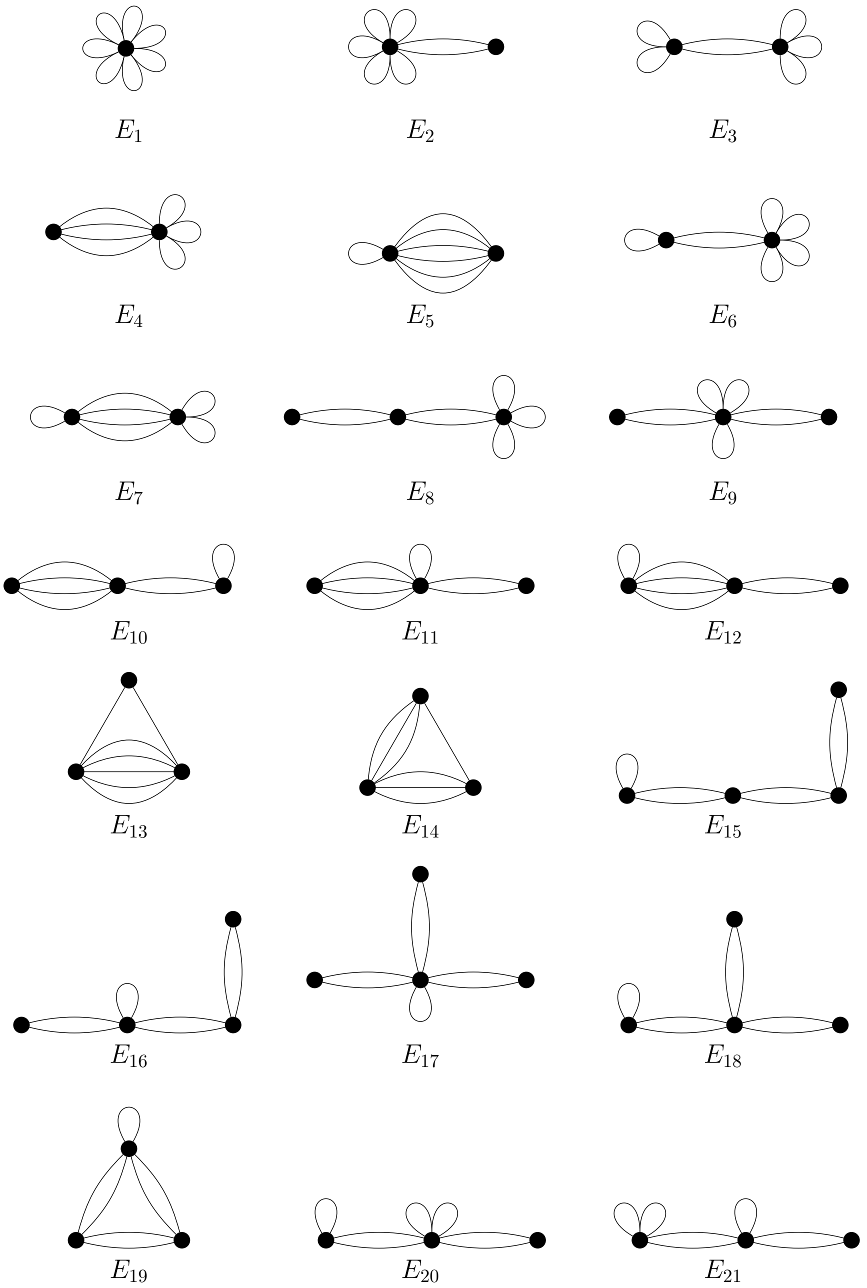


Figure 6.13: The 6-cyclic pseudographs $\mathcal{H}_6$.



E_1 E_2 E_3

E_4 E_5 E_6

E_7 E_8 E_9

E_{10} E_{11} E_{12}

E_{13} E_{14} E_{15}

E_{16} E_{17} E_{18}

E_{19} E_{20} E_{21}

Figure 6.14: The 7-cyclic pseudographs $\mathcal{H}_7$

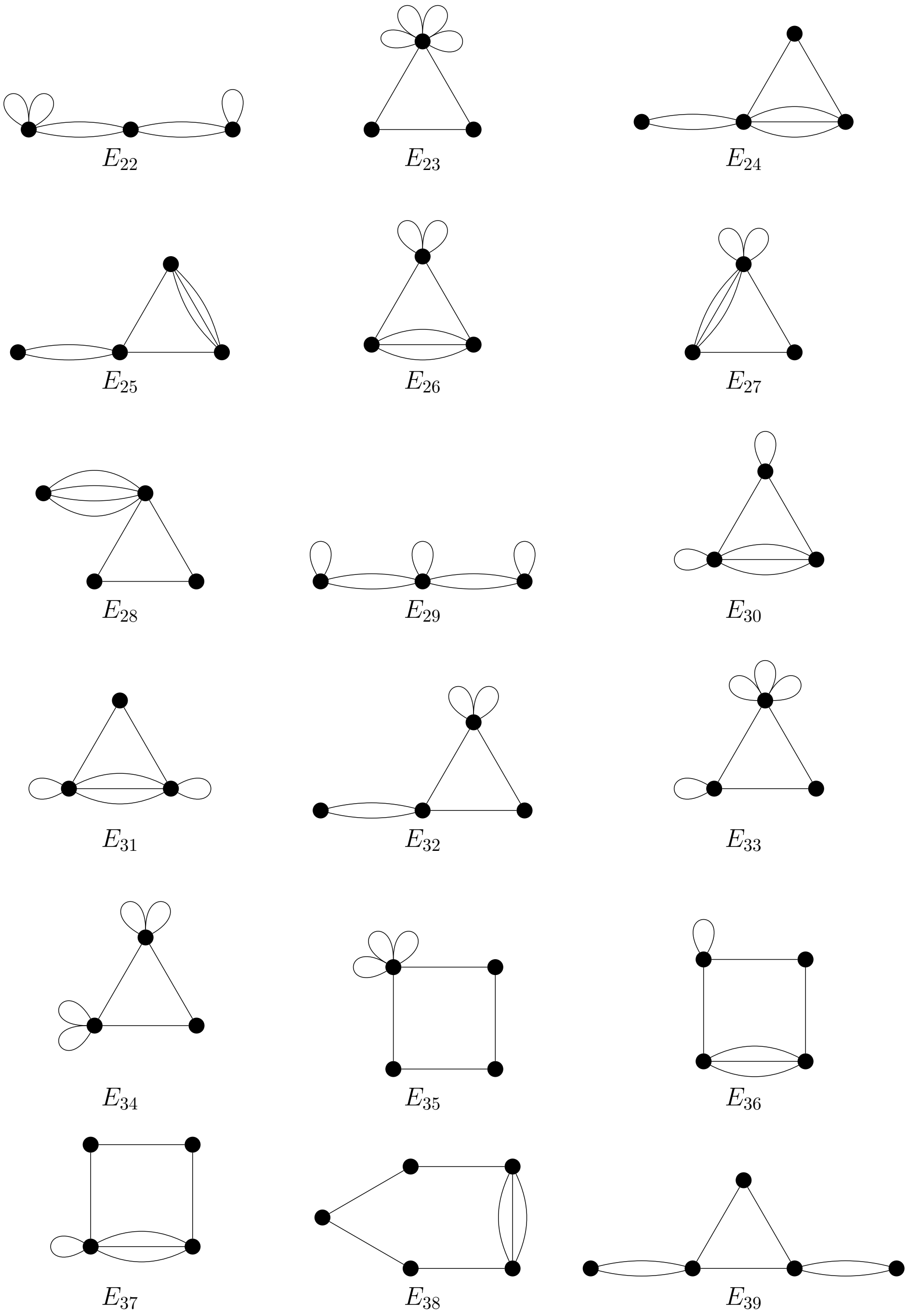


Figure 6.15: The 7-cyclic pseudographs $\mathcal{H}_7$

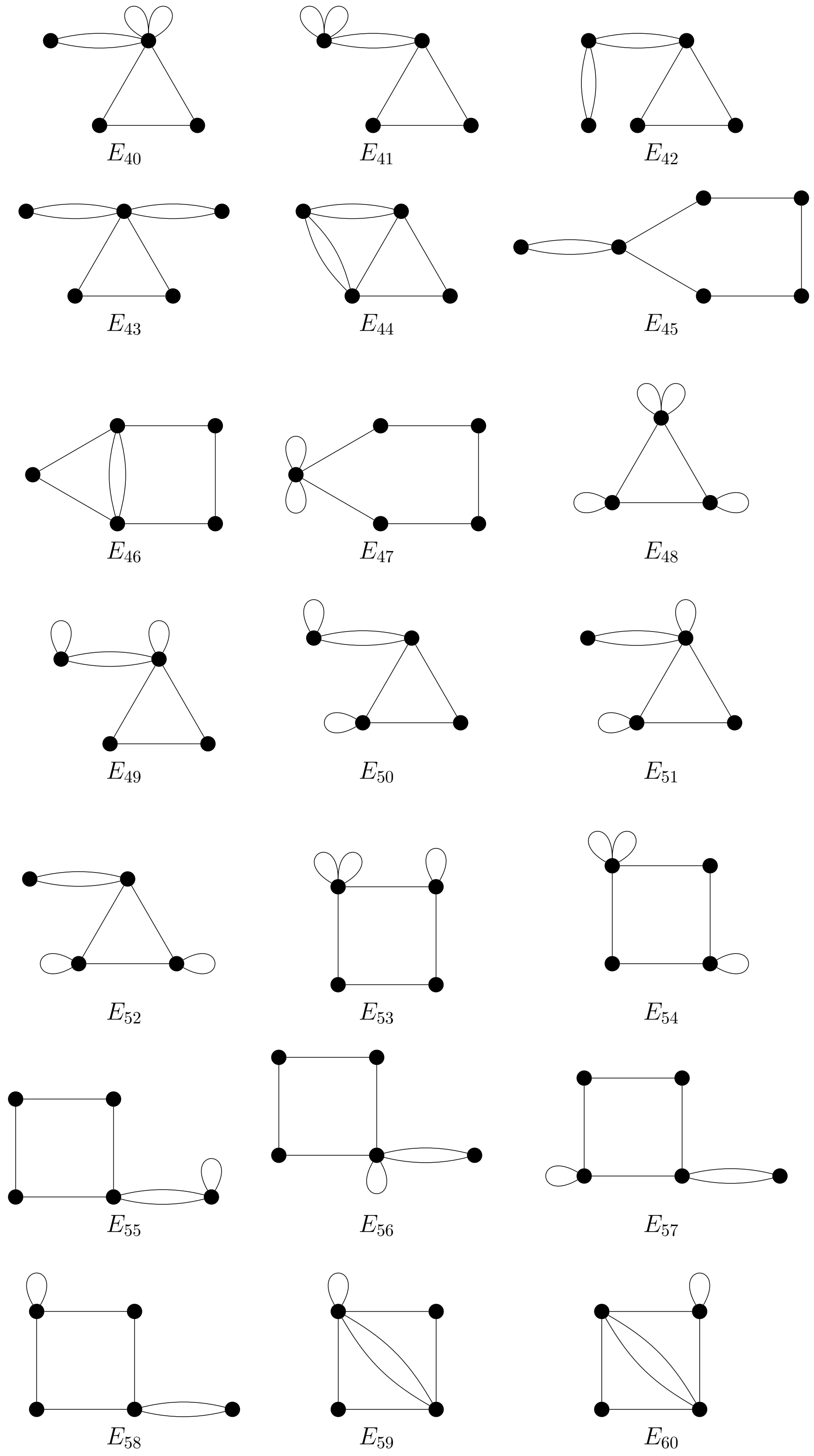
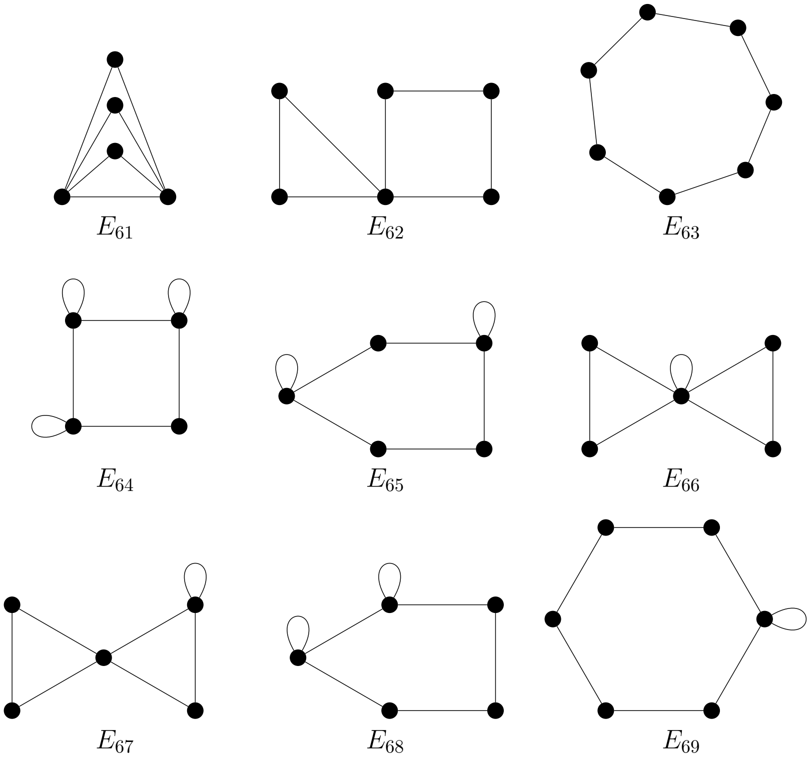


Figure 6.16: The 7-cyclic pseudographs $\mathcal{H}_7$.



E_{61} E_{62} E_{63}

E_{64} E_{65} E_{66}

E_{67} E_{68} E_{69}

Figure 6.17: The 7-cyclic pseudographs $\mathcal{H}_7$.

CHAPTER 7

NUMBER OF CONNECTED COMPONENTS IN A GRAPH: ESTIMATION VIA COUNTING PATTERNS

With the increasing size of modern datasets, a common network analysis task involves sampling a graph, due to restrictions on memory, communication, and computation resources. From such a subgraph with sampled nodes and their interconnections, we want to infer some global properties of the original graph that are relevant to the application in hand. This paper focuses on the task of inferring the number of connected components. It is a fundamental graph property of interest in various applications such as estimating the weight of the minimum spanning trees [35, 17], estimating the number of classes in a population [80], and visualizing large networks [169].

In the sampled subgraph, the count of connected components in general can be smaller as well as larger than the true value. Some connected components might not be sampled at all, whereas the connected nodes in the original graph is not guaranteed to be connected in the subgraph. It is not at all clear how the true number of components is related to the complex structure of the sampled graph. It is unknown how to unravel the complex relationship between the sampled subgraph and the global property of interest, making it challenging to use standard statistical approaches; there is no existing general estimator for the number of connected components. In this paper, we propose encoding the sampled subgraph by counting patterns in the subgraph, and prove that it makes its connection to the number of connected components transparent.

We represent a graph by a vector of counts of all possible patterns, also known as network motifs. For example, the first and second entries in this count vector encodes the number of nodes and (twice) the number of edges, respectively. Later entries encode the count of increasingly complex patterns: the number of times a pattern is repeated in the graph. This vector is clearly a redundant over-representation whose dimension scales super exponentially in the graph size. Perhaps surprisingly, for the purpose of inferring a global property, it suffices to have the first few hundred dimensions of this vector, corresponding to the counts of very small patterns. For counting those patterns, we introduce novel algorithms, and give a precise characterization of how the complexity (the size of the patterns included in the estimation) trades off with accuracy (the mean squared error).

Problem statement and our proposed approach. We want to estimate the number of connected components in a simple graph $G = (V, E)$ from a sampled subset of its nodes and the corresponding subgraph. Let N be the number of vertices and $\texttt{cc}(G)$ the number of connected components in G. We consider the subgraph sampling model, that is, a subset of vertices is sampled at random and the induced subgraph is observed. We consider a Bernoulli sampling model, where each vertex is sampled independently with a probability p. Let Ω be the set of randomly observed vertices, and G_Ω be the corresponding induced subgraph, i.e. $G_\Omega = (\Omega, E_\Omega)$ where $(i, j) \in E_\Omega$ if $i, j \in \Omega$ and $(i, j) \in E$. We want to estimate $\texttt{cc}(G)$ from G_Ω. We propose a novel spectral approach, which makes transparent the relation between the counts of patterns and the number of connected components.

We propose characterizing the number of connected components as the count of zero eigenvalues of its Laplacian matrix $L \in \mathbb{R}^{n\times n}$ given by

$$L \;\equiv\; D - A\,, \tag{7.1}$$

where $D = \text{diag}(A\mathbb{1})$ is the diagonal matrix of the degrees, and A is the adjacency matrix of the graph G. The rank of L reveals $\texttt{cc}(G)$ as

$$\begin{aligned} \texttt{cc}(G) &= N - \text{rank}(L) \\ &= N - \sum_{i\in[N]} \mathbb{I}\big[\sigma_i(L) > 0\big]\,, \end{aligned} \tag{7.2}$$

where the $\sigma_i(L)$'s are the singular values of the graph Laplacian L. Using this relation directly for estimation is an overkill as estimating the singular values is more challenging than estimating $\texttt{cc}(G)$. Instead, we use a few steps of functional approximations to relate to the pattern counts. By Gershgorin's circle theorem, we have $\sigma_i(L) \le 2\text{d}_{\max}$, where $\text{d}_{\max}$ is the maximum degree in G. We therefore normalize L by $1/\beta$ for some $\beta \ge 2d_{\max}$ to ensure all eigenvalues lie in the unit interval $[0, 1]$ and denote it by $\widetilde{L} = (1/\beta)L$. For any constant $0 < \alpha < 1$ that separates the zero and non-zero eigenvalues such that $\alpha < \min_i\{\sigma_i(\widetilde{L}) : \sigma_i(\widetilde{L}) \neq 0\}$, we consider the following approximation of the rank function. We approximate the step function in (7.2) by a continuous piecewise linear function $H_\alpha : [0,1] \to [0,1]$ illustrated in Figure 7.4:

$$\begin{aligned} H_\alpha(x) &= \begin{cases} 1 & \text{if } x \in [\alpha, 1] \;, \\ \frac{x}{\alpha} & \text{if } x \in [0, \alpha] \;. \end{cases} \;, \quad \text{and} \\ \mathsf{cc}(G) &= N - \sum_{i \in [N]} H_\alpha\big(\sigma_i(\widetilde{L})\big) \;, \end{aligned} \tag{7.3}$$

where we used the fact that the approximation is exact under our assumption that the spectral gap is lower bounded by α. To connect it to the pattern counts, we propose a further approximation using a polynomial function $f_\alpha : \mathbb{R} \to \mathbb{R}$ of a finite degree m. Precisely, for $f_\alpha(x) = a_1 x + \cdots + a_m x^m$ (e.g. Figure 7.4), we immediately have the following relation:

$$\sum_{i=1}^{N} f_\alpha(\sigma_i(\widetilde{L})) \;=\; \sum_{k=1}^{m} \frac{a_k}{\beta^k} \|L\|_k^k \,, \tag{7.4}$$

where $\|L\|_k^k$ is the Schatten-k norm of L which is defined as sum of k-th power of its singular values: $\|L\|_k^k \equiv \sum_{i=1}^N \sigma_i(L)^k$. As we choose $f_\alpha(x)$ to be a close approximation of the desired $H_\alpha(x)$, we have the following approximate relation: $\mathsf{cc}(G) \approx N - \sum_{k=1}^m (a_k/\beta^k)\|L\|_k^k$, which can be made arbitrarily close by choosing a larger degree m.

Finally, we propose using the fact that $\|L\|_k^k = \mathrm{Tr}(L^k)$ is a sum of the weights of all length k closed walks. Once we compute the (weighted) count of those walks for each pattern, this gives a direct formula to approximate the number of connected components from the counts. This approximation can be made as accurate as we want, by choosing the right order m in the polynomial approximation. Unlike the singular values, the (weighted) counts can be directly estimated from the sampled subgraph in a statistically efficient manner. We introduce a novel unbiased estimator $\widehat{\Theta}_k(G_\Omega)$ for Schatten-k norms of L in Section 7.1 that uses the counts of patterns in the sampled subgraph, and appropriately aggregates the estimated counts of the original graph. Together with a polynomial approximation $f_\alpha(x)$, this gives a novel estimator:

$$\widehat{\mathsf{cc}}(G_\Omega, \alpha, \beta, m) \;\; \equiv \;\; N - \sum_{k=1}^{m} \frac{a_k}{\beta^k} \widehat{\Theta}_k(G_\Omega)\,, \tag{7.5}$$

where $\widehat{\Theta}_k(G_\Omega)$ is an unbiased estimate of Schatten-k norm of L defined in (7.12) and a_k's are the coefficients in the polynomial approximation $f_\alpha(x) = a_1 x + \cdots + a_m x^m$ as defined as in (7.22).

Related work. It has been suspected that there is a fundamental connection between the number of connected components in the original graph and the counts of various patterns in the sampled graph. Although previous attempts to make this connection precise have been unsuccessful [73, 74], there is enough evidence to suggest that this is plausible. Existing estimators customized for two simple extreme cases of *forests* and *unions of disjoint cliques* all rely only on the counts of a few extremely simple patterns.

For a *forest* $G = (V, E)$, the estimator introduced in [73] exploits the simple relation that the number of connected components is $\mathsf{cc}(G) = |V| - |E|$. Hence, we only need to estimate the number of edges. This is a straightforward procedure that uses the counts of *k-stars* in the sampled subgraph for $k \in \{0, 1, \ldots\}$. A k-star is a graph with one central node with k adjacent nodes, mutually disjoint.

For a *union of disjoint cliques* $G = (V, E)$, the estimator introduced in [73] exploits the simple relation that the number of connected components is

$$\mathsf{cc}(G) = \sum_{k=1}^{|V|} \{\ \#\text{ of cliques of size } k\ \}\,.$$

We only need to estimate the number of cliques of each size k in the original graph. This is straightforward as the observed size of the cliques follow a multinomial distribution. This requires only the counts of *k-cliques* in the sampled subgraph for $k \in \{1, 2, \ldots\}$ [80, 73]. A k-clique is a fully connected graph with k nodes. These approaches have recently been extended in [118] to include chordal graphs, which introduces a novel idea of smoothing to achieve a strong performance guarantees. However, none of these methods can be applied to our setting where we consider the original graph to be a general graph.

Contributions. We pose the problem of estimating the number of connected components as a spectral estimation problem of estimating the rank of the graph Laplacian. This is further split into two tasks of first estimating the Schatten k-norms of the Laplacian and then applying a functional approximation.

We propose an unbiased estimator of the Schatten k-norm $\|L\|_k^k$ based on the counts of patterns in the subsampled graph, known as k-cyclic pseudographs. The main challenge is in estimating the diagonal entries of L (which is the degree of each node), that is critical in computing the weighted counts of the k-cyclic pseudographs. To overcome this challenge, in Section 7.1, we introduce an estimator that uses a novel idea of partitioning the subsampled graph and stitching the estimated degrees in each partition together.

Combining the estimated Schatten norms with polynomial approximation of $H_\alpha(x)$ in (7.3), we introduce a novel estimator of the number of connected components, which to the best of our knowledge is the first estimator that works on general graphs. We provide a sharp characterization of the bias-variance tradeoff of our estimator in Section 7.4. Numerical experiments show that for unions of disjoint cliques where competing estimators exist, the proposed *generic* estimator outperforms (in accuracy) even those estimators *tailored* for this structure. Further, both approaches have comparable runtimes. We also give experimental results for union of Erdös-Rényi graphs, for which there is no algorithm for estimating the number of connected components.

7.1 Unbiased estimator of Schatten-k norms of a graph Laplacian

In this section, we focus on the unnormalized L as Schatten norms are homogeneous and the normalization can be applied afterwards. We first provide an alternative method for computing $\|L\|_k$, and show how it leads to a novel estimator of the Schatten norm from a sampled subgraph. We use an alternative expression of the Schatten k-norm of a positive semidefinite L as the trace of the k-th power:

$$(\|L\|_k)^k \;\; = \;\; \mathrm{Tr}(L^k)\;. \tag{7.6}$$

Such a sum of the diagonal entries is the sum of weights of all closed walks of length k, where the weight of a walk is defined as follows. A length-k closed walk in $G = (V, E)$ is a sequence of vertices $w = (w_1, w_2, \ldots, w_k, w_{k+1})$ with $w_1 = w_{k+1}$ and either $(w_i, w_{i+1}) \in E$ or $w_i = w_{i+1}$ for all $i \in [k]$. Note that we allow repeated nodes and repeated edges. Essentially, these are walks in a graph G augmented by self-loops at each of the nodes. We define the *weight* of a walk w in G to be

$$\mu_G(w) \;\; \equiv \;\; \prod_{i=1}^{k} L_{w_i w_{i+1}} \; , \tag{7.7}$$

which is the product of the weights along the walk and $L = D - A$ is the graph Laplacian. It follows from (7.6) that

$$\|L\|_k^k \;\; = \sum_{w\text{: all length } k \text{ closed walks}} \mu_G(w) \; . \tag{7.8}$$

Even though this formula holds for any general matrix L, it simplifies significantly for graph Laplacians, as its all non-zero off-diagonal entries are -1 (and its diagonal entries are the degrees of the nodes). Consider a length-3 walk $w = (u, v, v, u)$ whose pattern is shown in the subgraph A_2 in Figure 7.1. This walk has weight $\mu_G(w) = (-1)^2 d_v$, where d_v is the degree of node v. Similarly, a walk (u, u, u, u) of pattern A_1 in Figure 7.1 has weight $\mu_G(w) = d_u^3$, and a walk $w = (u, v, x, u)$ of pattern A_3 has weight $\mu_G(w) = (-1)^3$.

In general, for a node u in a walk w of length k, let s_u denote the number of self-loops traversed in the walk on node u. Then, it follows that

$$\mu_G(w) \;\; = \;\; (-1)^{(k - \sum_{u \in w} s_u)} \prod_{u \in w} d_u^{s_u} \; , \tag{7.9}$$

where d_u is the degree of node u in G. The weight of a walk is ±1 if there are no self loops in the walk. Otherwise, its absolute value is the product of the degrees of the vertices corresponding to the self loops, and its sign is determined by how many non-self loop edges there are.

The first critical step in our approach is to partition the summation in Eq. (7.8) according to the pattern of the respective walk, which will make (*i*) counting those walks of the same pattern more efficient; and (*ii*) also de-biasing straight forward (see Equation (7.12)) under ransom sampling.

We refer to component-wise scaling w.r.t. the inverse of the probability of being sampled as de-biasing, which is a critical step in our approach and will be explained in detail later in this section. Following the notations from enumeration of small cycles in [6] and [115], we use the family of patterns called *k-cyclic pseudographs*:

$$\|L\|_k^k \;=\; \sum_{H\in\mathbb{H}_k} \sum_{w:\mathcal{H}(w)=H} \mu_G(w) \;, \tag{7.10}$$

where $\mathbb{H}_k$ is the set of *patterns* that have k edges, and $\{w : \mathcal{H}(w) = H\}$ is the set of walks on G that have the same pattern H. We give formal definitions below. k-cyclic pseudographs expand the standard notion of simple k-cyclic graphs, and include multi-edges and loops, which explains the name pseudograph.

Definition 7.1. Let $C_k = (V_k, E_k)$ denote the undirected simple cycle with k nodes. An unlabelled and undirected pseudograph $H = (V_H, E_H)$ is called a *k-cyclic pseudograph* for $k \geq 3$ if there exists an onto node-mapping from $C_k = (V_k, E_k)$, i.e. $f : V_k \to V_H$, and a one-to-one edge-mapping $g : E_k \to E_H$ such that $g(e) = (f(u_e), f(v_e))$ for all $e = (u_e, v_e) \in E_k$. We use $\mathbb{H}_k$ to denote the set of all k-cyclic pseudographs. We use $c(H)$ to the number of different node mappings f from C_k to a k-cyclic pseudograph H. Each closed walk w of length k is associated with one of the graphs in $\mathbb{H}_k$, as there is a unique H that the walk is an Eulerian cycle of under a one-to-one mapping of the nodes. We denote this graph by $\mathcal{H}(w) \in \mathbb{H}_k$.

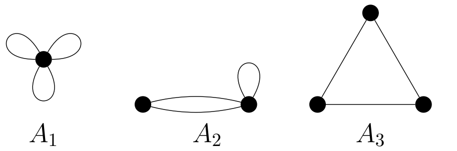


Figure 7.1: The 3-cyclic pseudographs $\mathcal{H}_3 = \{A_1, A_2, A_3\}$.

Figure 7.1 shows examples of all 3-cyclic pseudographs. $\mathbb{H}_3 = \{A_1, A_2, A_3\}$ and each one is a distinct pattern that can be mapped from a triangle graph C_3. In the case of A_1, there is only one mapping from C_3 to A_1 and corresponding multiplicity is $c(A_1) = 1$. Also, a walk $w = (u, u, u, u)$ on the graph

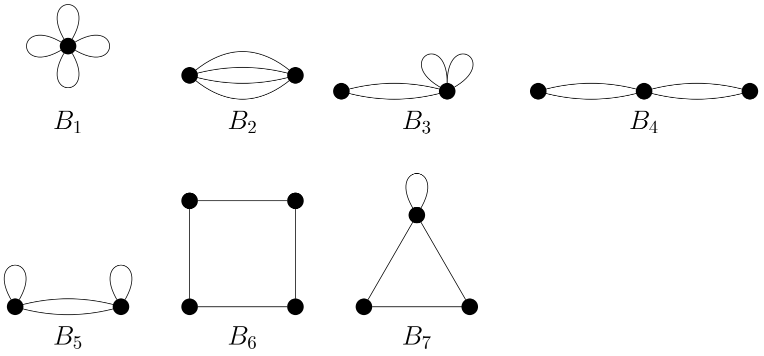


Figure 7.2: The 4-cyclic pseudographs $\mathcal{H}_4$.

G has pattern A_1, which we denote by $\mathcal{H}(w) = A_1$. In the case of A_2, any of the three nodes can be mapped to the left-node of A_2, which gives $c(A_2) = 3$. In the case of A_3, each permutation of the three nodes are distinct, which gives $c(A_3) = 6$. We show more examples of length 4 in Figure 7.2. k-cyclic pseudographs for larger k can be enumerated as well (e.g. [115]).

For a pattern H, let S_H denote the set of self-loops in H, and s_u denote the number of self loops at node u in the walk w. Then the summation of walks can be partitioned according to their patterns as:

$$\|L\|_k^k \;=\; \sum_{H \in \mathbb{H}_k} (-1)^{k-|S_H|} \Big\{ \sum_{w:\mathcal{H}(w)=H} \prod_{u \in w} d_u^{s_u} \Big\}\,, \tag{7.11}$$

which follows from substituting (7.9) in (7.10). This expression does not require the (computation of) singular values and leads to a natural unbiased estimator given a sampled subgraph. As the probability of a walk being sampled depends only on the pattern, we introduce a novel estimator $\widehat{\Theta}_k(G_\Omega)$ of $\|L\|_k^k$ that de-biases each pattern separately:

$$\widehat{\Theta}_k(G_\Omega) = \sum_{H \in \mathbb{H}_k} \frac{(-1)^{k-|S_H|}}{p^{|V_H|}} \Big\{ \sum_{w:\mathcal{H}(w)=H} \theta_w(G_\Omega) \mathbb{I}(w \subseteq G_\Omega) \Big\}\,, \tag{7.12}$$

where $|V_H|$ is the number of *nodes* in H, $p^{|V_H|}$ is the probability that walk with pattern H is sampled (i.e. all edges involves in the walk are present in the sampled subgraph G_Ω), and $\mathbb{I}(w \subseteq G_\Omega)$ denotes the indicator that all nodes in the walk w are sampled. $\theta_w(G_\Omega)$ is defined below.

As the degrees of the nodes in the original graph are unknown, it is challenging to estimate the polynomial of the degrees $\prod_{u\in w} d_u^{s_u}$ in Eq. (7.11), from the sampled graph. To this end, we introduce a novel estimator $\theta_w(G_\Omega)$ in Section 7.2, which is unbiased; it satisfies

$$\mathbb{E}[\theta_w(G_\Omega)|\mathbb{I}(w \subseteq G_\Omega)] \quad = \quad \prod_{u\in w} d_u^{s_u} \; .$$

It immediately follows by taking the expectation of (7.12), that $\widehat{\Theta}_k(G_\Omega)$ is unbiased, i.e.

$$\mathbb{E}_\Omega[\widehat{\Theta}_k(G_\Omega)] \quad = \quad \|L\|_k^k \; . \tag{7.13}$$

7.2 An unbiased estimator of the polynomial of the degrees

Our strategy to get an unbiased estimator of $\prod_{u\in w} d_u^{s_u}$ is to first partitioning the nodes in the original graph G to get a more insightful factorization of $\prod_{u\in w} d_u^{s_u}$ in Eq. (7.16) (see Figure 7.3) that removes dependences between the summands, and next by estimating each term independently in the factorization.

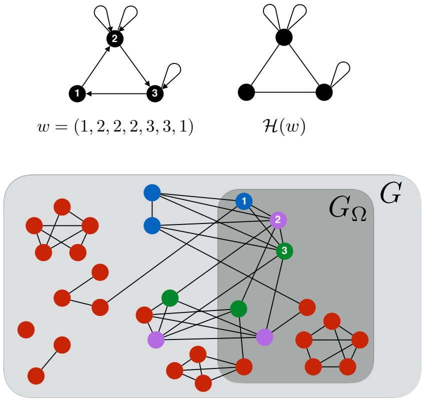


Figure 7.3: We are partitioning the original graph G with respect to a length-($k = 6$) closed walk $w = (1, 2, 2, 2, 3, 3, 1)$. Its corresponding k-cyclic pseudograph $\mathcal{H}(w) \in \mathbb{H}_6$ is shown on the top. Red nodes are not connected to either 2 or 3, which are the nodes of interest in w as they have self loops. Blue nodes are only connected to 2, purple to only 3, and blue to both 2 and 3.

Consider a concrete task of estimating $\prod_{u\in w} d_u^{s_u} = (d_2)^2(d_3)^1 = 6^2 \times 6$, for a walk $w = (1, 2, 2, 2, 3, 3, 1)$ in the observed subgraph G_Ω. Note that we only see G_Ω, whose degrees are very different from the original graph. For instance, node 2 now has degree 3 and node 3 has degree 3 in the sampled graph. Further, these random variables (the observed degrees) are correlated, making estimation challenging. To make such correlations apparent, we first give a novel partitioning of the nodes V in the following.

7.2.1 Partitioning V

Our strategy is first to partition the nodes V in the original graph, with respect to a walk $w = (w_1, \dots, w_{k+1})$ of interest. For a closed walk w, let $U = \{u_1, \dots, u_\ell\}$ denote the set of nodes in w that have at least one self-loop, let $\ell = |U|$ denote its cardinality, and let $\{s_1, \dots, s_\ell\}$ denote the number of self-loops at each node. In the running example, we have $U = \{u_1 = 2, u_2 = 3\}$, $\ell = 2$, $s_1 = 2$, and $s_2 = 1$. As our goal is to estimate $(d_2)^2(d_3)$, we partition the nodes with respect to how they relate to the nodes in $U = \{2, 3\}$. Concretely, there are four partitions: nodes that are not connected to either 2 or 3 (shown in red in Figure 7.3), nodes that are only connected to 2 (shown in green), nodes that are only connected to 3 (shown in purple), and nodes that are connected to both 2 and 3 (shown in blue). Nodes in each partition contribute in different ways to the target quantity $(d_2)^2(d_3)$, which will be precisely captured in the *factorization* in Eq. (7.16). In general, we need to consider all such variations in the partitioning, which gives

$$V \;=\; \bigcup_{T\subseteq U} V_{T,U\setminus T} \;, \tag{7.14}$$

where $V_{T,T'} = \big\{ \bigcap_{v\in T} \partial v \big\} \bigcap \big\{ \bigcap_{v\in T'} \partial v^c \big\}$ is the set of nodes that are adjacent to all nodes in T but are not adjacent to any nodes in T', and ∂v denotes the neighborhood of node v and ∂v^c denotes the complement of ∂v. We let $V_{\emptyset,U} = \bigcap_{v\in U} \partial v^c$ and $V_{U,\emptyset} = \bigcap_{v\in U} \partial v$. Essentially, we are labelling each node according to which nodes in U it is adjacent to, and grouping those nodes with the same label. In the running example, $V = V_{U,\emptyset} \cup V_{\{3\},\{2\}} \cup V_{\{2\},\{3\}} \cup V_{\emptyset,U}$, where the partitions are subset of nodes in blue, purple, green, and red, respectively.

Let $d_{T,U\setminus T} = |V_{T,U\setminus T}|$ denote the size of a partition such that

$$d_u \;\; = \;\; \sum_{T\in\mathcal{T}_u} d_{T,U\setminus T} \; , \tag{7.15}$$

for any $u \in U$ where $\mathcal{T}_u = \{T \subseteq U | u \in T\}$ is the set of subsets of U containing u. For example, $d_2 = 6$ which is the sum of blue and green nodes, and $d_3 = 6$ which is the sum of blue and purple nodes.

We are partitioning the neighborhood of u such that each term can be separately estimated. This ensures we handle the correlations among the degrees of different nodes in w correctly. The quantity of interest is

$$\begin{aligned}\prod_{i\in[\ell]} d_{u_i}^{s_i} &= \prod_{i\in[\ell]} \Big(\sum_{T\in\mathcal{T}_{u_i}} d_{T,U\setminus T} \Big)^{s_i} \\ &= \sum_{\left(T_1^{(1)},\dots,T_1^{(s_1)},\cdots,T_\ell^{(1)},\cdots,T_\ell^{(s_\ell)}\right)\in(\mathcal{T}_{u_1})^{s_1}\times\cdots\times(\mathcal{T}_{u_\ell})^{s_\ell}} \Big\{ \prod_{j=1}^{\ell}\prod_{i=1}^{s_j} d_{T_j^{(i)},U\setminus T_j^{(i)}} \Big\} \; ,\end{aligned} \tag{7.16}$$

where $T_j^{(i)}$ is a i-th choice of a set in $\mathcal{T}_{u_j}$ that contains the node u_j for $i \in [s_j]$, and $[\ell] = \{1, \dots, \ell\}$ denotes the set of positive integers up to ℓ. The second equation follows directly from exchanging the product and the summation. This alternative expression is crucial in designing an unbiased estimator, since each term in the summation can now be estimated separately as follows.

Consider a task of estimating a single term in (7.16), and we merge those $T_j^{(i)}$'s that happen to be identical:

$$\prod_{j=1}^{\ell}\prod_{i=1}^{s_j} d_{T_j^{(i)},U\setminus T_j^{(i)}} \;\; = \;\; \prod_{T\in\mathbb{T}} (d_{T,U\setminus T})^{t_T} \; , \tag{7.17}$$

where $\mathbb{T} = \{T_1^{(1)}, \dots, T_1^{(s_1)}, \cdots, T_\ell^{(1)}, \cdots, T_\ell^{(s_\ell)}\}$ is the current set of partitions allowing for multiple entries of the same set, and t_T is the multiplicity, i.e. how many times a set T appears in the set $\mathbb{T} = (T_1^{(1)}, \dots, T_1^{(s_1)}, \cdots, T_\ell^{(1)}, \cdots, T_\ell^{(s_\ell)})$. Each term in the right-hand side can be now separately estimated, as (a) $V_{T,U\setminus T}$'s are disjoint and (b) we know for the sampled subgraph the membership of each sampled node. This follows from the fact that, conditioned on the event that $\{w \subseteq \Omega\}$, we know how the sampled nodes in Ω are connected to any node in $\{w_i\}_{i=1}^{k+1}$ and in particular those with self-loops denoted by U. Hence, for any node in Ω the membership (or the color in the Figure 7.3) is trivially revealed. Therefore, we can handle (the degrees $d_{T,U\setminus T}$ in) each partition separately, and estimate each monomial in $\prod_{T\in\mathbb{T}}(d_{T,U\setminus T})^{t_T}$. The problem is reduced to the task of estimating $d^s_{T,U\setminus T}$ for some integer s and some partition $V_{T,U\setminus T}$.

7.2.2 Unbiased estimator of $d^s_{T,U\setminus T}$

From the original graph $G = (V, E)$ (where the size of each partition is denoted by $d_{T,U\setminus T}$), we observe a sampled subgraph $G_\Omega = (\Omega, E_\Omega)$ (where the size of each partition in G_Ω is denoted by $d_{T,U\setminus T}(\Omega)$), and we let $d_{T,U\setminus T}(w)$ denote the size of the partition intersecting the walk $w = (w_1, \dots, w_{k+1})$. Precisely, $d_{T,U\setminus T}(\Omega) \equiv |V_{T,U\setminus T} \bigcap \Omega|$, and $d_{T,U\setminus T}(w) \equiv |V_{T,U\setminus T} \bigcap \{w_i\}_{i=1}^{k+1}|$. We do not allow multiple counts when computing the size, such that $d_{\{2\},\{3\}}(\Omega) = 2$ and $d_{\{2\},\{3\}}(w) = 1$, in the example.

Let us focus on a particular walk w on a graph G, its corresponding U and a fixed $T \subseteq U$, such that $V_{T,U\setminus T}$ and $d_{T,U\setminus T}$ are fixed. Now $d_{T,U\setminus T}(\Omega)$ is a random variable representing how many nodes in the partition $V_{T,U\setminus T}$ are sampled. Conditioned on the fact that w is sampled, and hence a $d_{T,U\setminus T}(w)$ sampled nodes are already observed, the remaining $(d_{T,U\setminus T} - d_{T,U\setminus T}(w))$ nodes are sampled i.i.d. with probability p. Hence, conditioned on $\{w \subseteq \Omega\}$, the size of the sampled partition is distributed as

$$d_{T,U\setminus T}(\Omega) \sim \mathrm{Binom}(d_{T,U\setminus T} - d_{T,U\setminus T}(w), p) + d_{T,U\setminus T}(w)\ . \qquad (7.18)$$

This leads to a natural unbiased estimator of the monomial $d^s_{T,U\setminus T}$ as

$$\widehat{d}^{(s)}_{T,U\setminus T} \;=\; \langle (\, A^{-1})_{s+1}\,,\, \overline{d}\,\rangle\ , \qquad (7.19)$$

where $\overline{d} = [1\,,\, d_{T,U\setminus T}(\Omega)\,,\, d_{T,U\setminus T}(\Omega)^2, \dots, d_{T,U\setminus T}(\Omega)^s]^\top$ is a column vector in $\mathbb{R}^{s+1}$ of the monomials of the observed size of the partition, A is the unique matrix satisfying

$$\mathbb{E}[\overline{d}] \;=\; A\,[1\,,\, d_{T,U\setminus T}\,,\, \dots\,,\, d^s_{T,U\setminus T}]^\top\ , \qquad (7.20)$$

and $(A^{-1})_{s+1}$ is the $(s+1)$-th row of A^{-1}. One can check immediately that

$\mathbb{E}[\widehat{d}^{(s)}_{T,U\setminus T}] = \langle (A^{-1})_{s+1}, \mathbb{E}[\overline{d}]\,\rangle = d^{s}_{T,U\setminus T}$, hence giving the desired unbiased estimator. The matrix A is a lower triangular matrix which depends only on s, p and the structure of the walk via $d_{T,U\setminus T}(w)$. In terms of these three parameters, the required vector $(A^{-1})_{s+1}$ has a closed form expression, and hence the estimator can be computed in a straight forward manner. It uses the moments of a binomial distribution, which can be computed immediately.

An example of A for $s = 3$ is given in (7.51), where one should plug-in $\ell = d_{T,U\setminus T}(w) + 1$, $\omega = d_{T,U\setminus T} + 1$ and $\widetilde{\tau} = d_{T,U\setminus T}(\Omega)$. This leads to an unbiased estimator of $\prod_{i\in[\ell]} d^{s_i}_{u_i}$ by replacing (7.19) into (7.17) and (7.16):

$$\theta(w, G_\Omega) = \sum_{\left(T_1^{(1)},\ldots,T_1^{(s_1)},\cdots,T_\ell^{(1)},\cdots,T_\ell^{(s_\ell)}\right)\in(\mathcal{T}_{u_1})^{s_1}\times\cdots\times(\mathcal{T}_{u_\ell})^{s_\ell}} \left\{\prod_{T\in\mathbb{T}} \widehat{d}^{(t_T)}_{T,U\setminus T}\right\}. \tag{7.21}$$

By construction, it is immediate that the estimator is unbiased: $\mathbb{E}[\theta(w, G_\Omega)|I(w \subseteq \Omega)] = \prod_{u\in w} d^{s_u}_{u}$.

7.3 Polynomial approximation

The remaining goal in our approach is to design a polynomial approximation of the target function $H_\alpha : [0, 1] \to [0, 1]$ defined in (7.3) for a fixed scalar $\alpha \in (0, 1)$. Concretely, for a given integer m, we want a degree-m polynomial approximation $f(x)$ of $H_\alpha(x)$ such that (i) $f(0) = 0$; (ii) the approximation error (as measured by the ℓ_∞ norm) is small in the interval $[\alpha, 1]$; and (iii) we can provide an upper bound on the approximation error: $\max_{x\in[\alpha,1]} |H_\alpha(x) - f(x)|$. The first condition can be met by any function with proper scaling and shifting, and strictly enforcing it ensures that we make fair comparisons. The second condition ensures we have a good approximation, as the non-zero singular values only lie in the interval $[\alpha, 1]$. In particular, the approximation error outside of this interval is irrelevant. The last condition ensures we get the desired performance guarantees for the estimation error of the number of connected components. The (upper bound on the) approximation error of the polynomial function directly translates into the end-to-end error on the estimation.

A first attempt might be to use a Chebyshev approximation [36, 144] directly on $H_\alpha(x)$. This is optimal in terms of achieving a target ℓ_∞ error with the smallest degree in all regimes of $[0,1]$. However, we only care about the ℓ_∞ error in $[\alpha,1]$. As shown in magenta curve in Figure 7.4, the Chebyshev approximation $C(x)$ unnecessarily fits the curve in $(0,\alpha]$, resulting in larger error in $[\alpha,1]$.

A natural fix is to use filter design techniques, e.g. Parks-McClellan algorithm [165], where Chebyshev polynomials have been applied to design high pass filters with similar constraints as ours. This will give a polynomial approximation with small approximation error in the desired pass band of $[\alpha,1]$. However, these techniques do not come with the desired approximation guarantee that we seek.

One approach proposed in [217] does come with a provable error bound. This approximation $B(x)$ composes a Chebyshev approximation of a constant degree q with the CDF of a beta distribution of degree $(m/q-1)/2$. The beta distribution boosts the approximation of the function in the interval $[\alpha,1]$, thus providing an error bound of $O((c_\alpha)^m)$, where c_α is a constant that depends on α. Figure 7.4 shows that $B(x)$ (in green) still unnecessarily fits the curve in $(0,\alpha]$, as it starts with a (lower-degree) Chebyshev approximation of $H_\alpha(x)$.

Our goal is to design a new polynomial approximation that ignores the region $(0,\alpha]$ completely, such that it achieves improved performance in $[\alpha,1]$, and also comes with a provable error bound. We propose using a parametric family that ensures $f_b(0)=0$:

$$f_b(x) \;\; = \;\; 1-\prod_{i=1}^{m}(1-b_i x) \;, \tag{7.22}$$

for a vector $b=[b_1,\ldots,b_m]\in\mathbb{R}^m$. We provide an upper bound on the approximation error achieved by the optimal b^*, provide a choice of $\widetilde{b}$ in a closed form that achieves the same error bound, and provide a heuristic for locally searching for the optimal b^* to improve upon the closed-form $\widetilde{b}$.

Proposition 7.2. *For any $\alpha\in(0,1)$ and $m\geq 2$, the optimal parameter $b^*\in\arg\min_{b\in\mathbb{R}^m}\max_{x\in[\alpha,1]}|H_\alpha(x)-f_b(x)|$ achieves error bounded by*

$$\max_{x\in[\alpha,1]}|H_\alpha(x)-f_{b^*}(x)| \;\; \leq \;\; \left(\frac{1-\alpha}{1+\alpha}\right)^m \, . \tag{7.23}$$

A proof is provided in Section 7.6.10. As the optimal b^* is challenging to find, one option is to simplify the optimization by searching over a smaller space. By constraining all b_i's to be the same, solving for minimum ℓ_∞ error gives a closed form solution $\widetilde{b} \equiv (2/(1+\alpha))[1, \dots, 1]$ that achieves the bound in (7.23) with equality, i.e. $\max_{x \in [\alpha,1]} |H_\alpha(x) - f_{\widetilde{b}}(x)| = ((1-\alpha)/(1+\alpha))^m$.

For practical use, we prescribe a slightly better approximation using a local search algorithm in Algorithm 13. The approximation guarantee is compared for $\alpha = 0.2$ and varying m in Figure 7.4 against the analytical choice $f_{\widetilde{b}}(x)$, the standard Chebyshev approximation $C(x)$ of the first kind, and the approximation $B(x)$ from [217]. The proposed $f_{\widehat{b}}(x)$ significantly improves upon both, achieving a faster convergence. The key idea is to exploit the fact that we care about approximating only in the regime of $[\alpha, 1]$. There might be other techniques to design better polynomial approximation than ours, e.g. [165], but might not come with a performance guarantee.

The inset in the top panel of Figure 7.4 illustrates how the proposed $\widehat{b}$ in red admits more fluctuations to achieve smaller ℓ_∞ error, compared to the uniform choice of $\widetilde{b}$. In Algorithm 13, starting from a moderate perturbation around $\widetilde{b}$, we iteratively identify the point x' achieving the maximum error and update b such that the error at x' is decreased. This approximation can be done offline for many random initializations for the desired α and m; the one with minimum error can be stored for later use.

```
**Algorithm 13** Local search for a polynomial approximation
**Input:** degree $m$, $\alpha$ , number of iterations $T$, step size $\delta > 0$
**Output:** $\widehat{b} \in \mathbb{R}^m$
  $b_i \Leftarrow (2/(1+\alpha)) + \mathrm{U}[-1,1]$ for all $i \in [m]$
  **for** $t = 1$ **to** $T$ **do**
    $x' \Leftarrow \arg\max_{x \in [\alpha,1]} \left| \prod_{i \in [m]} (1 - b_i x) \right|$
    $b \Leftarrow b + \mathrm{sign}(1 - f_b(x')) \times \delta \times \nabla_b f_b(x')$
  **end for**
```

7.4 Main results

The polynomial approximation $f_{\widehat{b}}(x)$ of the form (7.22) can easily be translated into the standard polynomial with coefficients $a = (a_1, \dots, a_m)$ such that $f_{\widehat{b}}(x) = a_1 x + \cdots + a_k x^k$. Together with the Schatten norm estimator

$\widehat{\Theta}_k(G_\Omega)$ in (7.12), this gives the proposed estimate $\widehat{\mathsf{cc}}(G_\Omega, \alpha, \beta, m)$ in (7.5). We first give an upper bound on the multiplicative error for a special case of union of cliques, and give a general bound in Theorem 7.4. The overall procedure achieves the following, for a special case of *union of cliques*, which are also called *transitive graphs*:

Theorem 7.3. *If the underlying graph G is a disjoint union of cliques with clique sizes ω_i, for each connected component $1 \leq i \leq \mathsf{cc}(G)$, $\omega_{\max} \equiv \max_i\{\omega_i\}$ and $\omega_{\min} \equiv \min_i\{\omega_i\}$, then for any choice of $\beta \geq \omega_{\max}$ and $\alpha \leq \omega_{\min}/\beta$, and any integer $m \geq 1$, there exist a function $g(m) = O(m!)$ and a constant $C > 0$ such that for $\omega_{\min} > C$,*

$$\frac{\mathbb{E}\big[(\widehat{\mathsf{cc}}(G_\Omega, \alpha, \beta, m) - \mathsf{cc}(G))^2\big]}{\mathsf{cc}(G)^2} \leq$$
$$\frac{g(m)\,(1-p^m)}{\mathsf{cc}(G)^2\, p\, \beta^2} \sum_{i=1}^{\mathsf{cc}(G)} \left(\omega_i^4 \Big(1 + (\omega_i p)^{1-2m}\Big)\right) + \gamma^{2m} \frac{N^2}{\mathsf{cc}(G)^2}, \qquad (7.24)$$

where $\gamma = (1-\alpha)/(1+\alpha)$. Moreover, if there exist some positive constants c_i's such that $\omega_i^3 p \geq c_i m!$ or $\omega_i p^3 \leq c_i/m!$ for all i, then (7.24) holds with $g(m) = O(c^m)$ for some constant $c > 0$.

A proof of Theorem 7.3 is provided in Appendix 7.6.6. This clearly shows the tradeoff between the variance (the first term in the RHS) and the bias (the second term in the RHS). If we choose larger m, our functional approximation becomes more accurate resulting in a smaller bias. However, this will require counting larger patterns in the estimate of $\widehat{\Theta}_k(G_\Omega)$, leading to a larger variance.

In general, the complexity of our estimator for union of cliques is $O(m \times \mathsf{cc}(G))$, as all relevant quantities to compute $\widehat{\Theta}_k(G_\Omega)$ can be pre-computed and stored in a table for all combinations of k and the size of the observed cliques. At execution time, we only need to look up one number for each clique we observe and for each $\widehat{\Theta}_k(G_\Omega)$ we are estimating. Hence, the above guarantee also characterizes the trade-off between the computational complexity and the accuracy. For example, when spectral gap α is small, we need large m with longer run-time to get bias as small as we need. We emphasize here that our estimator is generic and does not assume the true graph is union of cliques. The same generic estimator happens to be more efficient, when the *observed* subgraph is a union of cliques, as shown precisely in Lemma 7.8.

Consider the bias term, which captures how the error increases for graphs with smaller spectral gap in L. The normalized spectral gap for union of cliques is $\omega_{\min}/\omega_{\max}$, and balanced components result in a small spectral gap and a more accurate estimation.

Consider the variance term, and as an extreme example, consider the case when all cliques are of the same size $\omega = N/\texttt{cc}(G)$. It immediately follows that for $\beta = \omega_{\max} = \omega_{\min}$ and $\alpha = 1$, there is no bias and $\gamma = 0$. Further assuming $\omega p > 1$, we can choose some small $m = O(1)$ to minimize the variance which scales as $O(N^2/(\texttt{cc}(G)^3\, p))$. Hence, to achieve arbitrarily small error, it is sufficient to have sample size Np scale as $(N/\texttt{cc}(G))^3$. This implies that finite multiplicative error is guaranteed only for $\texttt{cc}(G) = \Omega(N^{2/3})$.

Such a condition on $\texttt{cc}(G)$ increasing with respect to N seems to be unavoidable in general. Consider a case when $\texttt{cc}(G) = cN$ for some constant c. Then, we need $m = (1/2)\log_\gamma(\delta c^2/2)$ to make the bias as small as we want, say $\delta/2$. Suppose the connected components are balanced such that $\omega_{\max} = O(1)$, then the variance term will be at most $\delta/2$, if $p = \Omega(m^\varepsilon/N)$, where ε depends on γ and c.

Note that the best known guarantees for estimators tailored for union of cliques still require $\texttt{cc}(G) = \Omega(N^{1-\varepsilon})$ for small but strictly positive p, where the ε can be made arbitrarily small with a small sampling probability p (e.g. [73, 118]).

We run synthetic experiments on a graph of $N = 3775$ nodes and union of 50 cliques, each of size $\{51, 52, \ldots, 100\}$. Figure 7.5 shows that we improve upon three competing estimators for a broad range of p. $\widehat{\texttt{cc}}_{\text{chordal}}$ is the best known estimator for chordal graphs from [118], and $\widehat{\texttt{cc}}_{\text{clique}}$ is a smoothed version of $\widehat{\texttt{cc}}_{\text{chordal}}$ explicitly using the knowledge that the underlying graph is a union of cliques. These are tailored for chordal graphs and cliques, respectively, and cannot be applied to general graphs. Our generic algorithm, with an appropriate choices of α, β, and m, outperforms these approaches for unions of cliques. In particular, when p is small, variance dominates and choosing small m helps, whereas when p is large, bias dominates and choosing large m helps.

Theorem 7.4. *For any graph G with size of connected components ω_i, for* *each component* $1 \leq i \leq \mathsf{cc}(G)$*, and degree of each node* $d_j^{(i)}$*, for* $1 \leq j \leq \omega_i$ *with* $d_i \equiv \max_j\{d_j^{(i)}\}$*, and* $d_{\max} \equiv \max_{i,j}\{d_j^{(i)}\}$*,* $d_{\min} \equiv \min_{i,j}\{d_j^{(i)}\}$*, for any choice of* $\beta \geq 2d_{\max}$ *and* $\alpha \leq \sigma_{\min}(L)/\beta$*, and any integer* $m \geq 1$*, there exist a function* $g(m) = O(2^{m^2})$ *such that,*

$$\frac{\mathbb{E}\big[(\widehat{\mathsf{cc}}(G_\Omega, \alpha, \beta, m) - \mathsf{cc}(G))^2\big]}{\mathsf{cc}(G)^2} \leq$$

$$\frac{g(m)\,(1-p^m)}{\mathsf{cc}(G)^2\, p\, \beta^2} \sum_{i=1}^{\mathsf{cc}(G)} \left(\omega_i^2 d_i^2 \Big(1 + (d_i p)^{1-2m}\Big)\right) + \gamma^{2m} \frac{N^2}{\mathsf{cc}(G)^2}, \qquad (7.25)$$

where $\gamma = (1-\alpha)/(1+\alpha)$*. Moreover, if there exist some positive constants* c_i*'s such that* $d_i^3 p \geq c_i 2^{m^2}$ *or* $d_i p^3 \leq c_i/2^{m^2}$ *for all* i*, then* (7.25) *holds with* $g(m) = O(c^m)$ *for some constant* $c > 0$*.*

A proof of Theorem 7.4 is provided in Appendix 7.6.6. This guarantee shows a similar bias-variance tradeoff, with similar dependence on m, which controls the computational complexity and α which is the normalized spectral gap of the original graph Laplacian. The main difference in this generic setting is how computational complexity depends on m. Since we need to estimate $\theta_w(G_\Omega)$ which is an unbiased estimate of $\prod_{u \in w} d_u^{s_u}$, we need to compute it separately for each observed walk w of length k that involves at least one self loop. For the other walks, we exploit a recent algorithm in counting patterns from [115] inspired by a celebrated result from [6], and compute their weighted counts. This can be made as a look-up table, and overall the complexity scales as $O(\mathsf{cc}(G) \times \omega_{\max}^3)$ for $m \leq 7$ and for larger m scales as $O(\mathsf{cc}(G) \times \omega_{\max}^{m/2} \times 2^{m/2})$. If one has faster algorithms for counting patterns those can be seamlessly included in the procedure, for example using recent advances in recursive methods for counting structures from [41]. Our code is publicly available at *url-anonymized.*

We run experiments in Figure 7.5 on a graph of size $N = 5000$ with 50 components each drawn from Erdös-Rényi graph with probability half $G_{100,0.5}$. A moderate $m = 5$ is sufficient to achieve multiplicative error as small as 0.002, which implies that we make a small mistake in one out of ten instances. Note that $\widehat{\mathsf{cc}}_{\text{chordal}}$ and $\widehat{\mathsf{cc}}_{\text{clique}}$ cannot be applied as the observed subgraph is neither cliques nor chordal. We provide the first estimator for such general graphs. As the bias does not depend on p, this experiment implies that with only $m = 5$ the bias is already smaller than 0.002 and the variance is dominating. This is due to the fact that union of Erdös-Rényi graphs exhibit large spectral gaps. The variance decreases linearly in this log-log scale, with respect to the sampling probability p.

For the example of union of Erdös-Rényi graphs $G_{n,q}$ of the same size with $\texttt{cc}(G) = N/n$ connected components, the (normalized) spectral gap is $\Omega(1)$ for a large enough q. This exhibits the desired spectral gap, as long as q is sufficiently large, e.g. $q = \Omega(\log n/n)$. The ideal case is when $q = 1$, which recovers the union of cliques. The normalized spectral gap is one, which is the maximum possible value. On the other hand, the spectral gap can be also made arbitrarily small. Consider a union of n-cycles, where each component is a cycle of length n. In this case, the normalized spectral gap scales as $O(1/n^2)$, which can be quite small. For general graphs, the difficulty (both in computational complexity and sample complexity) depends on the spectral gap of the original graph. If spectral gap is small, then we need higher degree polynomial approximation functions to make the bias small, which in turn requires larger patterns to be counted. This increases the computational complexity and also the variance in the estimate. More samples are required to account for this increased variance.

7.5 Conclusion

We address the problem of estimating the number of connected components in an undirected simple graph, when only a subgraph is observed, where the nodes are chosen uniformly at random. Existing methods relied on special structures of the graphs, and can only be applied to union of disjoint cliques, union of disjoint trees, and chordal graphs. Applying a key insight of viewing the number of connected components as a spectral property that depends on the singular values of the graph Laplacian, we propose a novel spectral approach to this problem. Based on the fact that the number of connected components are the number of zero-valued singular values of the graph Laplacian, we make several innovations. First, we propose weighted count of small patterns (which are called network motifs) to estimate the k-th Schatten norm of the graph Laplacian. Next, to get an estimate of the (monomials of the) degrees, we propose a novel partitioning scheme that gives an unbiased estimate of the desired quantity to be used in the estimate of the k-th Schatten norms. We propose a polynomial approximation of the linearly interpolated step function, and prove a upper bound on the approximation guarantee. Putting these together, we introduce the first estimator with provable performance guarantees, that works for general graphs.

We next discuss several challenges in applying this framework to real world graphs.

Counting patterns. When the underlying graph G is a disjoint union of cliques, computational complexity of our estimator is $O(m \times \texttt{cc}(G))$. In this case, for any clique of size k, count of all possible patterns in it and the estimates of $\theta_w(G_\Omega)$ for any walk w, characterized by the degrees of the self loops it involves, can be pre-computed and stored in a table for look-up at the time of execution. $\theta_w(G_\Omega)$ is an unbiased estimate of of the polynomial of the node degrees $\prod_{u \in w} d_u^{s_u}$. For general graphs, to compute $\widehat{\Theta}_k(G_\Omega)$, we need to compute $\theta_w(G_\Omega)$ separately for each observed walk w that has at least one self loop. For the walks that do not involve any self-loop, we can use matrix multiplication based pattern counting algorithms proposed in [115] for $m \leq 7$. For $m > 7$, one can use homomorphism based a recent recursive algorithm from [41]. Therefore, for general graphs the major computational complexity arises in computing $\theta_w(G_\Omega)$ for walks w that has at least one self-loop. In a different sampling scenario, if we have the additional information of the degree of each node that we observe then computing $\theta_w(G_\Omega)$ can be made fast for all the walks w. Another option is to apply recent advances in sampling-based methods for counting patterns, including wedge sampling [181], the 3-path sampling [99], Moss [203], GRAFT [170], and using Hamiltoniam paths for debiasing [38]. However, it is not immediate how to include the estimation of the monomials of the degrees into these existing fast methods.

Other sampling techniques. In practical settings, sampling nodes uniformly at random might be unrealistic. Our estimator generalizes naturally to a broader class of sampling schemes, which we call *graph sampling*. Consider a scenario where you first sample an *unlabelled* mother graph H_0 of the same size as G (the graph of interest) from any distribution (in particular we do not require any independence on the sampled edges). Then, we apply a permutation drawn uniformly at random to assign node labels to the unlabelled graph H_0. Let H denote this labelled graph, which we use to sample the original graph of interest G. Specifically, for all edges in H, we observe whether the corresponding edge is present or not in G. Namely, we observe the adjacency matrix of G, but masked by the adjacency matrix of H. The random permutation ensures that the sampling probability for a pattern only depends on the shape of the pattern and not the specific labels of the nodes involved, making our algorithm extendible up to properly applying the debiasing as per the new sampling model. The model studied in this paper is a special case of graph sampling where H_0 is a clique of random size, drawn according to a binomial distribution.

On the other hand, more practical sampling scenarios are adaptive to the topology of the graph, creating selection biases. Examples include crawling a connected path from a starting node, sampling higher degree nodes, or sampling via random walks. These create dependencies among the topology and the sampling, which we believe is outside the scope of this paper, but nevertheless poses an interesting new research direction.

7.6 Proofs

We provide proofs for main results and technical lemmas. Recall $\widehat{\mathsf{cc}}(G_\Omega, \alpha, \beta, m) = N - \sum_{k=1}^{m} \frac{a_k}{\beta^k} \widehat{\Theta}_k(G_\Omega)$, where $\widehat{\Theta}_k(G_\Omega)$ is an unbiased estimate of Schatten-k norm of L defined in (7.12) and a_k's are coefficients of polynomial $f_b(x)$ defined in (7.22). We show in (7.28) that for the proposed estimator, bias is bounded as

$$\begin{aligned}\left|\mathsf{cc}(G) - \mathbb{E}\big[\widehat{\mathsf{cc}}(G_\Omega, \alpha, \beta, m)\big]\right| &\leq (N - \mathsf{cc}(G))((1-\alpha)/(1+\alpha))^m \\ &= (N - \mathsf{cc}(G))\gamma^m\,, \end{aligned} \tag{7.26}$$

where $\gamma \equiv (1-\alpha)/(1+\alpha)$. For the choice of $\widetilde{b} \equiv (2/(1+\alpha))[1, \ldots, 1]$, the coefficient of x^k in $f_b(x)$ is bounded as $|a_k| \leq \binom{m}{k} 2^k \leq 2^m 2^k \leq 4^m$. Therefore the mean square error is bounded by

$$\begin{aligned}&\mathbb{E}\big[(\widehat{\mathsf{cc}}(G_\Omega, \alpha, \beta, m) - \mathsf{cc}(G))^2\big] \;= \\ &2^{4m}\mathrm{Var}\bigg(\sum_{k=1}^{m} \frac{1}{\beta^k}\widehat{\Theta}_k(G_\Omega)\bigg) + \Big((N - \mathsf{cc}(G))\gamma^m\Big)^2 .\end{aligned} \tag{7.27}$$

Since the Schatten norm estimator is unbiased, $\mathbb{E}[\widehat{\Theta}_k(G_\Omega)] = \|L\|_k^k$, we have

$\mathbb{E}\big[\widehat{\mathsf{cc}}(G_\Omega,\alpha,\beta,m)\big] = N - \sum_{i=1}^N f_\alpha(\sigma_i(\widetilde{L}))$. where f_α is defined in Equation (7.4). Note that β is chosen such that the non-zero eigenvalues of $\widetilde{L} = L/\beta$ are bounded between α and 1. With the proposed choice of polynomial function f_α, we have $f_\alpha = f_b$. Using, Equation (7.3), $\mathsf{cc}(G) = N - \sum_{i=1}^N H_\alpha(\sigma_i(\widetilde{L}))$, along with $\max_{x\in[\alpha,1]} |H_\alpha(x) - f_{\widetilde{b}}(x)| = ((1-\alpha)/(1+\alpha))^m$, where $\widetilde{b} \equiv (2/(1+\alpha))[1,\dots,1]$, we have,

$$\begin{aligned}
\Big|\mathsf{cc}(G) - \mathbb{E}\big[\widehat{\mathsf{cc}}(G_\Omega,\alpha,\beta,m)\big]\Big| &= \sum_{i=1}^N \Big(H_\alpha(\sigma_i(\widetilde{L})) - f_{\widetilde{b}}(x)\Big) \\
&= \sum_{i:\sigma_i(\widetilde{L})=0} \Big(H_\alpha(\sigma_i(\widetilde{L})) - f_{\widetilde{b}}(x)\Big) + \sum_{i:\sigma_i(\widetilde{L})\neq 0} \Big(H_\alpha(\sigma_i(\widetilde{L})) - f_{\widetilde{b}}(x)\Big) \\
&\le \mathsf{cc}(G)\Big(H_\alpha(0) - f_{\widetilde{b}}(0)\Big) + (N - \mathsf{cc}(G)) \max_{x\in[\alpha,1]} |H_\alpha(x) - f_{\widetilde{b}}(x)| \\
&\le (N - \mathsf{cc}(G))((1-\alpha)/(1+\alpha))^m = (N - \mathsf{cc}(G))\gamma^m\,. \qquad (7.28)
\end{aligned}$$

For the two cases: (a) when the underlying graph G is disjoint union of cliques, and (b) a general graph G with maximum degree $d_{\max}$, we provide bounds on the variance of the Schatten k-norm estimator that leads to the bounds on mean square error using Equation (7.27).

Denote each connected component of G by $G^{(i)}$ for $1 \le i \le \mathsf{cc}(G)$, and let $G^{(i)}_\Omega$ denote the randomly observed subgraph of the connected component $G^{(i)}$. Then, we have,

$$\mathrm{Var}\bigg(\sum_{k=1}^m \frac{1}{\beta^k}\widehat{\Theta}_k(G_\Omega)\bigg) = \sum_{i=1}^{\mathsf{cc}(G)} \mathrm{Var}\bigg(\sum_{k=1}^m \frac{1}{\beta^k}\widehat{\Theta}_k(G^{(i)}_\Omega)\bigg)\,. \qquad (7.29)$$

Note that, our estimator of Schatten k-norm naturally decomposes, and can be computed separately for each connected component $G^{(i)}$ and then added together to get the estimate for the graph G.

7.6.1 Proof of Theorem 7.3

The following lemma provides bound on the variance of Schatten k-norm estimator for a clique graph. We give a proof in Section 7.6.2.

Lemma 7.5. *For a clique graph G on ω vertices, there exists a universal positive constant C such that for $\omega \ge C$, variance of Schatten k-norm estimator*

$\widehat{\Theta}_k(G_\Omega)$ *is bounded by*

$$\mathrm{Var}\big(\widehat{\Theta}_k(G_\Omega)\big) \;\; \leq \;\; g(k)\frac{\omega^{2k+2}}{p}\Big(1+\frac{1}{(\omega p)^{2k-1}}\Big)\,, \tag{7.30}$$

where $g(k) = O(k!)$*. Moreover, if there exists a positive constant* c *such that* $\omega^3 p \geq ck!$ *or* $\omega p^3 \leq c/k!$ *then* (7.30) *holds with* $g(k) = \mathrm{poly}(k)$*.*

Using Equations (7.27) and (7.29) along with Lemma 7.5, and the fact that $\beta \geq \omega$, Theorem 7.3 follows immediately.

7.6.2 Proof of Lemma 7.5

For a k-cyclic pseudograph $H = (V_H, E_H)$, let S_H denote the set of self-loops in H. Recall that in (7.12) Schatten k-norm estimator of Laplacian L is given by

$$\widehat{\Theta}_k(G_\Omega) = \sum_{H\in\mathbb{H}_k} \frac{(-1)^{k-|S_H|}}{p^{|V_H|}}\Big\{\sum_{w:\mathcal{H}(w)=H} \theta(w,G_\Omega)\mathbb{I}(w\subseteq G_\Omega)\Big\}\;.$$

For a clique graph G, the analysis of the above estimator simplifies significantly. We illustrate this with an example in Figure 7.7. Consider a length $k = 6$ walk $w = (1,2,2,2,3,3,1)$ with a corresponding k-cyclic pseudograph $\mathcal{H}(w)$. In general, the degree estimator $\theta(w, G_\Omega)$ is chosen such that $\mathbb{E}[\theta(w,G_\Omega)] = d_2^2 d_3$ where d_2 and d_3 are the degrees of nodes 2 and 3 in G, respectively. This simplifies significantly for a clique graph due to the fact that the degree of those nodes in a closed walk w are the same. Note that our estimator is general, and does not use this information or the fact that the underlying graph component is a clique. It is only the analysis that simplifies. Therefore, for a clique graph G, the degree estimator $\theta(w, G_\Omega)$ satisfies $\mathbb{E}[\theta(w,G_\Omega)|w\subseteq G_\Omega] = (\omega-1)^{|S_H|}$, where ω is the size of the clique G. In the example, we have $\omega = 7$ and $|S_H| = 3$, therefore $\mathbb{E}[\theta(w,G_\Omega)] = 6^3$. Hence, it is best to further partition $\mathbb{H}_k$ according to the number of nodes $\ell = |V_H|$ and the number of self-loops $s = |S_H|$. Precisely, we define

$$\mathbb{H}_{k,\ell,s} \;\; \equiv \;\; \{H(V_H,E_H)\in\mathbb{H}_k \,:\, |V_H| = \ell \text{ and } |S_H| = s\}\;,$$

for $\ell = 1$, $s = k$ and $2 \leq \ell \leq k$, $0 \leq s \leq k-\ell$. There are total $|\{w \in W\,:$

$\mathcal{H}(w) \in \mathbb{H}_{k,\ell,s}\}| \leq \ell^{2k-s}\omega^{\ell}$ corresponding walks in this set. Here, W denotes the collection of all length k closed walks on a complete graph of ω vertices. We slightly overload the notion of complete graph to refer to an undirected graph with not only all the $\omega(\omega-1)/2$ simple edges but also with ω self loops as well. when G is a clique graph, the estimator (7.12) can be re-written as

$$\begin{aligned}\widehat{\Theta}_k(G_\Omega) &= \sum_{\ell=1}^{k}\sum_{s=0}^{k-\ell+1}\sum_{H\in\mathbb{H}_{k,\ell,s}}\Big\{\frac{(-1)^{k-s}}{p^{\ell}}\times \\ &\qquad \sum_{w\in W:\mathcal{H}(w)=H}\theta(w,G_\Omega)\mathbb{I}(w\subseteq G_\Omega)\Big\}\,. \qquad (7.31)\end{aligned}$$

Given this unbiased estimator, we provide an upper bound on the variance of each of the partitions. For any two walks $w, w' \in W$, let $|w\cap w'|$ denote the number of overlapping unique vertices of walks w and w'. We have,

$$\begin{aligned}\mathrm{Var}\big(\widehat{\Theta}_k(G_\Omega)\big) &= \sum_{\ell=1}^{k}\sum_{s=0}^{k-\ell+1}\sum_{H\in\mathbb{H}_{k,\ell,s}}\Big\{\frac{1}{p^{2\ell}} \\ &\qquad \times\sum_{w\in W:\mathcal{H}(w)=H}\mathrm{Var}\Big(\theta(w,G_\Omega)\mathbb{I}(w\subseteq G_\Omega)\Big)\Big\} \\ &+ \; 2\sum_{\tilde{\ell}=0}^{k}\sum_{\substack{w,w'\in W\\ |w\cap w'|=\tilde{\ell}}}\mathrm{Cov}\Big(\theta(w,G_\Omega)\mathbb{I}(w\subseteq G_\Omega)\,,\theta(w',G_\Omega)\mathbb{I}(w'\subseteq G_\Omega)\Big) \\ &\qquad \times\frac{(-1)^{|S_{\mathcal{H}(w)}|+|S_{\mathcal{H}(w')}|}}{p^{|V_{\mathcal{H}(w)}|+|V_{\mathcal{H}(w')}|}}\,. \qquad (7.32)\end{aligned}$$

The following technical lemma provides upper bounds on the variance and covariance terms. We provide a proof in Section 7.6.3.

Lemma 7.6. *Under the hypothesis of Lemma 7.5, for a length-k walk w over ℓ distinct nodes with $s\geq 1$ self-loops, the following holds:*

$$\mathrm{Var}\Big(\theta(w,G_\Omega)\mathbb{I}(w\subseteq G_\Omega)\Big) \leq f(\ell,s)\Big(p^{\ell-1}\omega^{2s-1}+\omega p^{\ell+1-2s}\Big)+p^{\ell}\omega^{2s}, \quad (7.33)$$

and when $\ell=1$, $s=k$, we have,

$$\mathrm{Var}\Big(\theta(w,G_\Omega)\mathbb{I}(w\subseteq G_\Omega)\Big) \leq f(k)\omega^{2k-1}+g(k)\omega p^{2-2k}+p\omega^{2k}, \qquad (7.34)$$

and for any length-k walks w_1, w_2 over ℓ_1, ℓ_2 distinct nodes with $\tilde{\ell}$ unique over-

lapping nodes, $|w_1 \cap w_2| = \tilde{\ell}$*,* $s_1, s_2 \geq 1$ *self-loops respectively, the covariance term can be upper bounded by:*

$$\begin{aligned}\mathrm{Cov}\Big(&\theta(w_1, G_\Omega)\mathbb{I}(w_1 \subseteq G_\Omega)\,, \theta(w_2, G_\Omega)\mathbb{I}(w_2 \subseteq G_\Omega)\Big) \\ &\leq f(\ell', s')p^{((\ell_1+\ell_2-\tilde{\ell})-(s_1+s_2))}\Big((\omega p)^{(s_1+s_2)} + \omega p\Big)\,, \end{aligned} \tag{7.35}$$

for some function $f(\ell, s) = O(k!)$*,* $g(k) = \mathrm{poly}(k)$*, where* p *is the vertex sampling probabiliy.* $\ell' \equiv \max\{\ell_1, \ell_2\}$ *and* $s' \equiv \max\{s_1, s_2\}$*.*

We use this lemma to get bound on $\mathrm{Var}\big(\widehat{\Theta}_k(G_\Omega)\big)$. First, we get a bound on the total variance term. For a walk $w \in W$ with $\mathcal{H}(w) \in \mathbb{H}_{k,\ell,s}$ with $1 \leq s \leq k-2$, using (7.33), we have,

$$\begin{aligned}\frac{\omega^\ell}{p^{2\ell}}&\mathrm{Var}\Big(\theta(w, G_\Omega)\mathbb{I}(w \subseteq G_\Omega)\Big) \\ &\leq f(\ell, s)\Big(\frac{\omega^{\ell+2s-1}}{p^{\ell+1}} + \frac{\omega^{\ell+1}}{p^{\ell+2s-1}}\Big) \;+\; \frac{\omega^{2s+\ell}}{p^\ell} \\ &\leq f(k)\Big(\frac{\omega^{2k-3}}{p^3} + \frac{\omega^3}{p^{2k-3}}\Big) \;+\; \frac{\omega^{2k-2}}{p^2}\,. \end{aligned} \tag{7.36}$$

For a walk $w \in W$ with $\mathcal{H}(w) \in \mathbb{H}_{k,\ell,s}$ with $\ell = 1$, $s = k$, using (7.34), we have,

$$\begin{aligned}\frac{\omega^\ell}{p^{2\ell}}&\mathrm{Var}\Big(\theta(w, G_\Omega)\mathbb{I}(w \subseteq G_\Omega)\Big) \\ &\leq f(k)\omega^{2k}p^{-2} \;+\; g(k)\omega^2 p^{-2k} \;+\; w^{2k+1}p^{-1}\,. \end{aligned} \tag{7.37}$$

For a walk w with $s = 0$, $\theta(w, G_\Omega) = 1$, and, we have,

$$\frac{\omega^\ell}{p^{2\ell}}\mathrm{Var}\Big(\theta(w, G_\Omega)\mathbb{I}(w \subseteq G_\Omega)\Big) \;\leq\; \frac{\omega^\ell}{p^\ell}\,. \tag{7.38}$$

Combining, Equations (7.36), (7.37), and (7.38), and using $|\{w \in W\;:$

$\mathcal{H}(w) \in \mathbb{H}_{k,\ell,s}\}| \leq \ell^{2k-s}\omega^{\ell}$, we have

$$\sum_{\ell=1}^{k}\sum_{s=0}^{k-\ell+1}\sum_{H\in\mathbb{H}_{k,\ell,s}}\Big\{\frac{1}{p^{2\ell}}\sum_{w\in W:\mathcal{H}(w)=H}\mathbb{I}(w\subseteq G)\times$$
$$\mathrm{Var}\Big(\theta(w,G_\Omega)\mathbb{I}(w\subseteq G_\Omega)\Big)\Big\}$$
$$\leq f(k)\omega^2 p^{-2k} \;+\; \omega^{2k+1}p^{-1}\,, \tag{7.39}$$

and if $\omega p^3 \leq 1/f(k)$, then the above quantity is bounded by $g(k)\omega^2 p^{-2k}(1+o(1))$. If $\omega^3 p \geq f(k)$, then the above quantity is bounded by $\omega^{2k+1}p^{-1}(1+o(1))$.

Consider covariance term of two length-k walks w_1, w_2 over ℓ_1, ℓ_2 distinct nodes with $\tilde{\ell}$ unique overlapping nodes, $|w_1 \cap w_2| = \tilde{\ell}$, and $s_1, s_2 \geq 1$ self-loops with $s_1 + s_2 < 2k$. Since there are a total of $f(k)\omega^{\ell_1+\ell_2-\tilde{\ell}}$ such walks, using (7.35), we have

$$\omega^{\ell_1+\ell_2-\tilde{\ell}}\mathrm{Cov}\Big(\theta(w_1,G_\Omega)\mathbb{I}(w_1\subseteq G_\Omega)\,,\theta(w_2,G_\Omega)\mathbb{I}(w_2\subseteq G_\Omega)\Big)$$
$$\times\,\frac{(-1)^{|S_{\mathcal{H}(w_1)}|+|S_{\mathcal{H}(w_2)}|}}{p^{|V_{\mathcal{H}(w_1)}|+|V_{\mathcal{H}(w_2)}|}} \tag{7.40}$$
$$\leq \frac{\omega^{\ell_1+\ell_2-\tilde{\ell}}}{p^{\tilde{\ell}+s_1+s_2}}\Big((\omega p)^{s_1+s_2}+\omega p\Big) \tag{7.41}$$
$$\leq \frac{\omega^{\ell_1+\ell_2+s_1+s_2}}{(\omega p)^{\tilde{\ell}}} + \omega p\frac{\omega^{\ell_1+\ell_2-\tilde{\ell}}}{p^{s_1+s_2+\tilde{\ell}}} \tag{7.42}$$
$$\leq \omega^{2k+1} + \omega p\frac{\omega^2}{p^{2k-1}} \tag{7.43}$$

where in (7.42), for the first term the maximum is achieved at $\ell_1 = 1, s = k, \ell_2 = 2, s_2 = k-2, \tilde{\ell} = 0$, for the second term the maximum is achieved at same ℓ_1, s_1, ℓ_2, s_2 with $\tilde{\ell} = 1$. Note that in the expression (7.41), the first term is dominating when $\omega p \geq 1$, and the second term is dominating when $\omega p < 1$.

When $s_1 + s_2 = 2k$ the two walks are self loop walks on single node with $s_1 = s_2 = k$. Since the graph G is a clique graph, $\theta(w, G_\Omega)$ for self loop walks depends only upon observed size of the clique, and $\theta(w_1, G_\Omega) = \theta(w_2, G_\Omega)$.

Therefore, the expression in (7.40) can be bounded as:

$$\begin{aligned}
&\omega^{\ell_1+\ell_2-\tilde{\ell}}\mathrm{Cov}\Big(\theta(w_1,G_\Omega)\mathbb{I}(w_1\subseteq G_\Omega)\,,\theta(w_2,G_\Omega)\mathbb{I}(w_2\subseteq G_\Omega)\Big)\\
&\qquad\times\frac{(-1)^{|S_{\mathcal{H}(w_1)}|+|S_{\mathcal{H}(w_2)}|}}{p^{|V_{\mathcal{H}(w_1)}|+|V_{\mathcal{H}(w_2)}|}}\\
&\quad\le\frac{\omega^{2\ell}}{p^{2\ell}}\mathrm{Var}\Big(\theta(w,G_\Omega)\mathbb{I}(w\subseteq G_\Omega)\Big)\\
&\quad\le f(k)\omega^{2k+1}p^{-2}\;+\;g(k)\omega^3p^{-2k}\;+\;w^{2k+2}p^{-1}\,, \qquad (7.44)
\end{aligned}$$

where (7.44) follows from (7.37).

Lemma 7.5 follows immediately by combining (7.32) with (7.66), (7.43) and (7.44).

7.6.3 Proof of Lemma 7.6

We use the following technical lemma to get bounds on conditional variance and covariance of the estimator θ. We provide a proof in Section 7.6.4.

Lemma 7.7. *Under the hypothesis of Lemma 7.5, for length-k walks w_1, w_2 over ℓ_1, ℓ_2 distinct nodes with $s_1, s_2 \geq 1$ self-loops respectively, the conditional variance of estimator $\theta(w_1, G_\Omega)$, defined in* (7.21)*, given that all the nodes in the walk are sampled can be upper bounded by*

$$\begin{aligned}
&\mathrm{Var}\Big(\theta(w_1,G_\Omega)\Big|w_1\subseteq G_\Omega\Big)\\
&\quad\le\;\; f(\ell_1,s_1)\Big(p^{-1}(\omega-\ell_1)^{2s_1-1}+(\omega-\ell_1)p^{1-2s_1}\Big)\;,\;\; \textit{and} \qquad (7.45)\\
&\mathbb{E}\Big[\theta(w_1,G_\Omega)\theta(w_2,G_\Omega)\Big|\mathbb{I}(w_1\subseteq G_\Omega)\mathbb{I}(w_2\subseteq G_\Omega)\Big]\\
&\quad\le\;\; f(\ell',s')p^{-(s_1+s_2)}\Big((\omega p)^{s_1+s_2}+\omega p\Big)\;, \qquad (7.46)
\end{aligned}$$

for some function $f(\ell, s) = O(k!)$, where p is the vertex sampling probability. $\ell' \equiv \max\{\ell_1, \ell_2\}$ and $s' \equiv \max\{s_1, s_2\}$. Moreover, for a length k walk w with $\ell = 1$, and $s = k$,

$$\mathrm{Var}\Big(\theta(w,G_\Omega)\Big|w\subseteq G_\Omega\Big)\le\frac{f(k)(\omega-1)^{2k-1}}{p}+\frac{g(k)(\omega-1)}{p^{2k-1}}, \qquad (7.47)$$

for some function $g(k) = \mathrm{poly}(k)$.

Recall for a clique graph, we have $\mathbb{E}[\theta(w,G_\Omega)|w\subseteq G_\Omega]=(\omega-1)^s$. Therefore, we have,

$$\begin{aligned}
&\mathrm{Var}\Big(\theta(w,G_\Omega)\mathbb{I}(w\subseteq G_\Omega)\Big)\\
&=\mathbb{E}\Big[\mathrm{Var}\Big(\theta(w,G_\Omega)\mathbb{I}(w\subseteq G_\Omega)\Big|\mathbb{I}(w\subseteq G_\Omega)\Big)\Big]\\
&\quad+\ \mathrm{Var}\Big(\mathbb{E}\Big[\theta(w,G_\Omega)\mathbb{I}(w\subseteq G_\Omega)\Big|\mathbb{I}(w\subseteq G_\Omega)\Big]\Big)\\
&=p^\ell\,\mathrm{Var}\Big(\theta(w,G_\Omega)\Big|w\subseteq G_\Omega\Big)\ +\ p^\ell(1-p^\ell)(\omega-1)^{2s}\\
&\le f(\ell,s)\Big(p^{\ell-1}(\omega-\ell)^{2s-1}+(\omega-\ell)p^{\ell+1-2s}\Big)\\
&\quad+\ p^\ell(1-p^\ell)(\omega-1)^{2s}\\
&\le f(\ell,s)\Big(p^{\ell-1}\omega^{2s-1}+\omega p^{\ell+1-2s}\Big)\ +\ p^\ell\omega^{2s}\,, \qquad (7.48)
\end{aligned}$$

where the inequality follows from Equation (7.45). Similarly, for a walk w with $\ell=1$, and $s=k$, we have,

$$\mathrm{Var}\Big(\theta(w,G_\Omega)\mathbb{I}(w\subseteq G_\Omega)\Big)\le f(k)\omega^{2k-1}+g(k)\omega p^{2-2k}+p\omega^{2k}\,,$$

where we used the inequality in (7.47).

For covariance term, we have,

$$\begin{aligned}
&\mathrm{Cov}\Big(\theta(w_1,G_\Omega)\mathbb{I}(w_1\subseteq G_\Omega)\,,\theta(w_2,G_\Omega)\mathbb{I}(w_2\subseteq G_\Omega)\Big)\\
\le\ &\mathbb{E}\Big[\theta(w_1,G_\Omega)\theta(w_2,G_\Omega)\Big|\mathbb{I}(w_1\subseteq G_\Omega)\mathbb{I}(w_2\subseteq G_\Omega)\Big]p^{(\ell_1+\ell_2-\tilde{\ell})}\\
\le\ &f(\ell',s')p^{((\ell_1+\ell_2-\tilde{\ell})-(s_1+s_2))}\Big((\omega p)^{(s_1+s_2)}+\omega p\Big)\,, \qquad (7.49)
\end{aligned}$$

where the inequality follows from Equation (7.46).

7.6.4 Proof of Lemma 7.7

When G is a union of disjoint cliques, the estimator $\theta(w,G_\Omega)$ defined in (7.21) has a compact representation. This follows from the fact that for any two nodes i and j that are connected in G_Ω, the neighborhoods of i and j in G_Ω exactly coincide. If this happens, then the estimator $\theta(w,G_\Omega)$ simplifies as follows. Consider a walk w with s self-loops, k edges (including self loops), and ℓ distinct nodes. Define a random integer $\widetilde{\tau}$ as the degree of a node in the clique that w belongs to in the sampled graph G_Ω, conditioned on the fact that all nodes in w are sampled (if a walk w is sampled then it must belong to the same clique.). The randomness comes from the sampling of Ω. It is straightforward that $\widetilde{\tau} \sim \text{Binom}(\omega - \ell, p) + (\ell - 1)$ as there are already $(\ell - 1)$ neighbors from the walk w and the rest of $((\omega - 1) - (\ell - 1))$ nodes are sampled in Ω with probability p, where ω is the size of the clique in the original graph G. For notational simplification, define a random integer $\tau \equiv \widetilde{\tau} - (\ell - 1)$ that is distributed as $\tau \sim \text{Binom}(\omega - \ell, p)$. In the example in Figure 7.7, for the walk $w = (1, 2, 2, 2, 3, 3, 1)$, we have $\ell = 3$, $s = 3$, $\omega = 7$ and a random instance of $\widetilde{\tau} = 4$ that is $\tau = 2$. We claim that the estimator $\theta(w, G_\Omega)$ given in (7.21) is a function of only τ, ℓ, s and p, and can be simplified as follows, and give a proof in Section 7.6.5.

Lemma 7.8. *When the underlying G is a union of disjoint cliques and G_Ω is a subgraph obtained via vertex sampling with a probability p, for a length k walk w in G_Ω with ℓ distinct nodes and s self-loops, we have*

$$\theta(w, G_\Omega) \quad = \quad \langle A^{-1}_{s+1}, \overline{\tau} \rangle \;, \tag{7.50}$$

where $\tau \equiv \widetilde{\tau} - (\ell - 1)$ and $\widetilde{\tau}$ is the degree of any node in the clique that w belongs to in the sampled graph G_Ω, $\overline{\tau} = [1, \tau, \tau^2, \ldots, \tau^s]$ is a column vector of monomials of τ up to degree s, and A^{-1}_{s+1} is the $(s+1)$-th row of the inverse of the matrix $A \in \mathbb{R}^{(s+1)\times(s+1)}$ satisfying $A\overline{\omega} \;=\; \mathbb{E}[\overline{\tau}]$, for $\overline{\omega} = [1, (\omega - 1), (\omega - 1)^2, \ldots, (\omega - 1)^s]$, a column vector of monomials of $(\omega - 1)$. Further, A is a lower-triangular matrix that depends only on ℓ and p, such that $\max_{i \in [s+1]} |(A^{-1})_{s+1,i}| = O(p^{-s})$.

In the running example, $s = 3$, $\widetilde{\tau} = 4$, and $\ell = 3$, and therefore we have

$$A = \begin{bmatrix} 1 & 0 & 0 & 0 \\ p - \ell p & p & 0 & 0 \\ \ell^2 p^2 - \ell p^2 - \ell p + p & -2\ell p^2 + p^2 + p & p^2 & 0 \\ A_{41} & A_{42} & -3\ell p^3 + 3p^3 & p^3 \end{bmatrix} \;, \tag{7.51}$$

where $A_{41} = -\ell^3 p^3 + 3\ell^2 p^2 + \ell p^3 - 3\ell p^2 - \ell p + p$ and $A_{42} = 3\ell^2 p^3 - 6\ell p^2 - p^3 + 3p^2 + p$. For any s, corresponding A can be computed immediately from the moments of a Binomial distribution up to degree s. Since $\overline{\omega} = \mathbb{E}[A^{-1}\overline{\tau}]$, this representation immediately reveals that $\mathbb{E}[\theta(w, G_\Omega) | w \subseteq G_\Omega] =$

$\mathbb{E}[\langle A^{-1}_{s+1}, \overline{\tau}\rangle] = (\omega-1)^s$. Note that τ is conditioned on the event that all the nodes in w are sampled.

With this definition, the variance of $\theta(w, G_\Omega)$ can be upper bounded as follows. We will let $f(\ell, s)$ denote a function over ℓ and s that captures the dependence in ℓ and s that may change from line to line, and only track the dependence in ω and p.

$$\begin{aligned}
\mathrm{Var}\Big(\theta(w,G_\Omega)\Big| w\subseteq G_\Omega\Big) \;&\le\; f(\ell,s) \max_{i\in[s+1]} \mathrm{Var}\big((A^{-1})_{s+1,i}\,\tau^{i-1}\big) \\
&\le\; f(\ell,s)\,p^{-2s}\,\mathrm{Var}(\tau^s) \\
&=\; f(\ell,s)p^{-2s}\Big(\mathbb{E}[\tau^{2s}] - \mathbb{E}[\tau^s]^2\Big) \\
&\le\; f(\ell,s)p^{-2s}\Big((\omega-\ell)^{2s}p^{2s} + f(s)(\omega-\ell)^{2s-1}p^{2s-1} \\
&\quad +\; (\omega-\ell)p - (\omega-\ell)^{2s}p^{2s}\Big) \\
&\le\; f(\ell,s)\Big(p^{-1}(\omega-\ell)^{2s-1} + (\omega-\ell)p^{1-2s}\Big)\,, \qquad (7.52)
\end{aligned}$$

where the first inequality follows from the fact that $\overline{\tau}$ is an $s+1$ dimensional vector, the second inequality follows from that fact that $\max_i |(A^{-1})_{s+1,i}| = O(p^{-s})$ from Lemma 7.8 and $\max_i = \mathrm{Var}(\tau^{i-1}) = \mathrm{Var}(\tau^s)$, and in the third inequality we used the fact that $\tau \sim \mathrm{Binom}(\omega-\ell, p)$ and a result from [18] that $\mathbb{E}[(\mathrm{Binom}(d,p))^s] \le \sum_{j=1}^s S(s,j)(dp)^j$ where $S(s,j)$ is the Sterling number of second kind. $S(s,s)=1$, $S(s,1)=1$ and $S(s,j)\le f(s)$, for $2\le j\le s-1$. We also used Jensen's inequality $\mathbb{E}[(\mathrm{Binom}(d,p))^s] \ge \mathbb{E}[(\mathrm{Binom}(d,p))]^s$. This proves the desired bound in (7.33).

To prove the bound in (7.47), observe that when $\ell=1$, $\tau\sim\mathrm{Binom}(\omega-1,p)$, and we can tighten the above set of inequalities

$$\begin{aligned}
\mathrm{Var}\Big(\theta(w,G_\Omega)\Big| w\subseteq G_\Omega\Big) \;&\le\; g(k) \max_{i\in[k+1]} \mathrm{Var}\big((A^{-1})_{k+1,i}\,\tau^{i-1}\big) \\
&\le\; g(k)\,p^{-2k}\,\mathrm{Var}(\tau^k) \\
&=\; g(k)p^{-2k}\Big(\mathbb{E}[\tau^{2k}] - \mathbb{E}[\tau^k]^2\Big) \\
&\le\; g(k)p^{-2k}\Big((\omega-1)^{2k}p^{2k} + f(k)(\omega-1)^{2k-1}p^{2k-1} \\
&\quad +\; (\omega-1)p - (\omega-1)^{2k}p^{2k}\Big) \\
&\le\; f(k)p^{-1}(\omega-1)^{2k-1} + g(k)(\omega-1)p^{1-2k}\,. \qquad (7.53)
\end{aligned}$$

For the covariance term, conditioned on the event that both the walks w_1 and w_2 are observed, distribution of the random degree integer of each walk is $\tilde{\tau}_1 = \tilde{\tau}_2 = \tilde{\tau}_{12}$, where $\tilde{\tau}_{12} \sim \text{Binom}(\omega - (\ell_1 + \ell_2 - \tilde{\ell}), p) + (\ell_1 + \ell_2 - \tilde{\ell} - 1)$. Therefore, the shifted Binomial random variable of each walk $\tau_1 = \tilde{\tau}_1 - (\ell_1 - 1) = \tilde{\tau}_{12} - (\ell_1 - 1)$, and $\tau_2 = \tilde{\tau}_2 - (\ell_2 - 1) = \tilde{\tau}_{12} - (\ell_2 - 1)$ For the walk w_1, with number of nodes ℓ_1, self loops s_1, lets denote the matrix A in (7.50) by A_1, and similarly for walk w_2, denote it by A_2. Then we have,

$$\begin{aligned}
&\mathbb{E}\Big[\theta(w_1, G_\Omega)\theta(w_2, G_\Omega)\Big|\mathbb{I}(w_1 \subseteq G_\Omega)\mathbb{I}(w_2 \subseteq G_\Omega)\Big] \\
&\leq f(\ell', s') \max_{i_1 \in [s_1+1], i_2 \in [s_2+1]} \mathbb{E}\Big[(A_1^{-1})_{s_1+1,i_1}\, \tau_1^{i_1-1}(A_2^{-1})_{s_2+1,i_2}\, \tau_2^{i_2-1}\Big] && (7.54) \\
&\leq f(\ell', s') p^{-(s_1+s_2)} \mathbb{E}\big[\tau_1^{s_1}\tau_1^{s_2}\big] && (7.55) \\
&\leq f(\ell', s') p^{-(s_1+s_2)} \Big((\omega p)^{s_1+s_2} + \omega p\Big), && (7.56)
\end{aligned}$$

where the first inequality follows from the fact that $\overline{\tau}_1$ is an $s_1 + 1$ dimensional vector, and $\overline{\tau}_2$ is an $s_2 + 1$ dimensional vector, the second inequality follows from that fact that $\max_i |(A_1^{-1})_{s_1+1,i_1}| = O(p^{-s_1})$ from Lemma 7.8 and $\max_{i_1,i_2} = \mathbb{E}[\tau_1^{i_1-1}\tau_2^{i_2-1}] = \mathbb{E}[\tau_1^{s_1}\tau_2^{s_2}]$, and in the third inequality we used a result from [18] that $\mathbb{E}[(\text{Binom}(d,p))^s] \leq \sum_{j=1}^{s} S(s,j)(dp)^j$ where $S(s,j)$ is the Sterling number of second kind. $S(s,j) \leq f(s)$, for $1 \leq j \leq s$. This proves the desired bound in (7.46).

7.6.5 Proof of Lemma 7.8

We are left to prove that the estimator $\theta(w, G_\Omega)$ simplifies as in (7.50) when the original graph is a clique graph(or a union of disjoint cliques).

We use the notations introduced in Section 7.2 and Section 7.6.4. Consider a closed walk w of length k on ℓ distinct nodes with $U = \{u_1, \cdots, u_{\tilde{\ell}}\}$ set of nodes in it that have at least one self-loop, $|U| = \tilde{\ell}$, and a total of s self loops. If the underlying graph is a clique graph the partition of V defined in (7.14), for any $T \subseteq U$, is as follows:

$$V_{T,U\setminus T} = \begin{cases} \emptyset & \text{if } |T| < |U| - 1 \\ d_{U,\emptyset} & \text{if } T = U \\ v & \text{if } T = U \setminus v, \text{ for any } v \in U\,. \end{cases} \qquad (7.57)$$

Recall that $d_{U,\emptyset} \equiv \cap_{v\in U}\partial v$. Therefore, we have,

$$d_{T,U\setminus T}(w) = \begin{cases} 0 & \text{if } |T| < |U|-1 \\ \ell - \tilde{\ell} & \text{if } T = U \\ 1 & \text{if } T = U \setminus v, \text{ for any } v \in U\,. \end{cases} \tag{7.58}$$

If the underlying clique graph is of size ω then $|d_{U,\emptyset}| = \omega - \tilde{\ell}$. Using the fact $d_{T,U\setminus T}(\Omega) \sim \text{Binom}(d_{T,U\setminus T} - d_{T,U\setminus T}(w), p) + d_{T,U\setminus T}(w)$, as explained in Section 7.2, we have,

$$d_{T,U\setminus T}(\Omega) \sim \begin{cases} 0 & \text{if } |T| < |U|-1 \\ \text{Binom}(\omega - \ell, p) + (\ell - \tilde{\ell}) & \text{if } T = U \\ 1 & \text{if } T = U \setminus v, \text{ for any } v \in U\,. \end{cases}$$

Using Equation (7.57) it is immediate that that degree of any node $u \in U$ is $d_u = d_{U,\emptyset} + (\tilde{\ell} - 1)$, and hence, $\mathbb{E}[\theta(w, G_\Omega)|w \in G_\Omega] = \big(d_{U,\emptyset} + (\tilde{\ell}-1)\big)^s = (\omega - 1)^s$. Therefore, an alternative characterization of the estimator defined in (7.21) is as following: $\theta(w, G_\Omega)$, conditioned on the event that all the nodes in the walk w are sampled, is a random variable dependent only upon $d_{U,\emptyset}(\Omega) \sim \text{Binom}(\omega - \ell, p) + (\ell - \tilde{\ell})$ such that its conditional expectation is $\mathbb{E}[\theta(w, G_\Omega)|w \in G_\Omega] = (\omega-1)^s$. With change of notations it is immediate that the estimator defined in (7.21) is same as the estimator in (7.50) when the underlying graph is a clique graph(or a disjoint union of cliques).

By the definition of A, it follows that the diagonal entries are exactly $\text{diag}([1, p, \ldots, p^s])$ and the bottom-left off-diagonal entris are all $\Theta(1)$ with respect to p, and the top-right off-diagonal entries are all zeros. Applying the inverse to this lower triangular matrix, it follows that A^{-1} is also a lower triangular matrix with diagonal entries $\text{diag}([1, p^{-1}, \ldots, p^{-s}])$ and the bottom-left off-diagonal entries are all $\Theta(1)$. It follows that $\max_{i\in[s+1]} |(A^{-1})_{s+1,i}| = O(p^{-s})$.

7.6.6 Proof of Theorem 7.4

The following lemma provides bound on variance of Schatten k-norm estimator for a connected general graph with maximum degree $d_{\max}$. We provide a proof in Section 7.6.7

Lemma 7.9. *For a connected graph G on ω vertices with maximum degree $d_{\max}$, variance of Schatten k-norm estimator $\widehat{\Theta}_k(G_\Omega)$ is bounded by*

$$\mathrm{Var}\big(\widehat{\Theta}_k(G_\Omega)\big) \;\; \leq \;\; h(k)\frac{\omega^2 {d_{\max}}^{2k}}{p}\Big(1+\frac{1}{(d_{\max}p)^{2k-1}}\Big)\,, \tag{7.59}$$

where $h(k) = O(2^{k^2})$. Moreover, if there exists a positive constant c such that ${d_{\max}}^3 p \geq c2^{k^2}$ or $d_{\max}p^3 \leq c/2^{k^2}$ then (7.59) holds with $h(k) = \mathrm{poly}(k)$.

Using Equations (7.27) and (7.29) along with Lemma 7.9, Theorem 7.4 follows immediately.

7.6.7 Proof of Lemma 7.9

We use the notations introduced in Section 7.2 and Section 7.6.2. Denote the size of the connected component by ω and let $d_{\max}$ be the maximum degree of any node in the connected component.

The following technical lemma provides upper bounds on the variance and covariance terms. We provide a proof in Section 7.6.8.

Lemma 7.10. *Under the hypothesis of Lemma 7.9, for a length-k walk w over ℓ distinct nodes with $s \geq 1$ self-loops, the following holds:*

$$\mathrm{Var}\Big(\theta(w,G_\Omega)\mathbb{I}(w \subseteq G_\Omega)\Big) \leq h(k)\Big(p^\ell {d_{\max}}^{2s} + d_{\max}p^{\ell+1-2s}\Big), \tag{7.60}$$

and when $\ell = 1$, $s = k$, we have,

$$\begin{aligned}&\mathrm{Var}\Big(\theta(w,G_\Omega)\mathbb{I}(w \subseteq G_\Omega)\Big)\\ &\qquad \leq f(k){d_{\max}}^{2k-1} \; + \; g(k)d_{\max}p^{2-2k} \; + \; p{d_{\max}}^{2k}\,,\end{aligned} \tag{7.61}$$

and for any length-k walks w_1, w_2 over ℓ_1, ℓ_2 distinct nodes with $\tilde{\ell}$ unique overlapping nodes, $|w_1 \cap w_2| = \tilde{\ell}$, $s_1, s_2 \geq 1$ self-loops respectively, the covariance term can be upper bounded by:

$$\begin{aligned}&\mathrm{Cov}\Big(\theta(w_1,G_\Omega)\mathbb{I}(w_1 \subseteq G_\Omega)\,,\theta(w_2,G_\Omega)\mathbb{I}(w_2 \subseteq G_\Omega)\Big)\\ &\qquad \leq h(k)p^{((\ell_1+\ell_2-\tilde{\ell})-(s_1+s_2))}\Big((d_{\max}p)^{(s_1+s_2)} + d_{\max}p\Big)\,,\end{aligned} \tag{7.62}$$

for some function $h(k) = O(2^{k^2})$, $f(k) = O(k!)$, *and* $g(k) = \text{poly}(k)$, *where* p *is the vertex sampling probabiliy.*

The total count of length k closed cycles on ℓ distinct nodes in a general graph on ω nodes graph with maximum degree $d_{\max}$ is bounded by $f(k)\omega {d_{\max}}^{\ell-1}$. It follows from the observation that fixing a node in the cycle, there are at most ${d_{\max}}^{\ell-1}$ paths to $\ell - 1$-hop neighbors. That is $|w \in W : \mathcal{H}(w) \in \mathbb{H}_{k,\ell,s}| \le f(k)\omega {d_{\max}}^{\ell-1}$ for any $1 \le s \le k$.

We use these inequalities to get bound on variance and covariance terms in (7.32).

For a walk $w \in W$ with $\mathcal{H}(w) \in \mathbb{H}_{k,\ell,s}$ with $1 \le s \le k-2$, using (7.60), we have,

$$\begin{aligned}
&\frac{\omega {d_{\max}}^{\ell-1}}{p^{2\ell}} \text{Var}\Big(\theta(w, G_\Omega)\mathbb{I}(w \subseteq G_\Omega)\Big) \\
&\le h(k)\Big(\frac{{d_{\max}}^{\ell+2s-1}}{p^{\ell}} + \frac{{d_{\max}}^{\ell}}{p^{\ell+2s-1}}\Big) \\
&\le h(k)\omega\Big(\frac{{d_{\max}}^{2k-3}}{p^{2}} + \frac{{d_{\max}}^{2}}{p^{2k-3}}\Big)\,. \qquad (7.63)
\end{aligned}$$

For a walk $w \in W$ with $\mathcal{H}(w) \in \mathbb{H}_{k,\ell,s}$ with $\ell = 1$, $s = k$, using (7.61), we have,

$$\begin{aligned}
&\frac{\omega}{p^{2\ell}} \text{Var}\Big(\theta(w, G_\Omega)\mathbb{I}(w \subseteq G_\Omega)\Big) \;\le \\
&\quad f(k)\omega {d_{\max}}^{2k-1}p^{-2} \;+\; g(k)\omega d_{\max} p^{-2k} \;+\; \omega {d_{\max}}^{2k}p^{-1}\,. \qquad (7.64)
\end{aligned}$$

For a walk w with $s = 0$, $\theta(w, G_\Omega) = 1$, and, we have,

$$\frac{\omega {d_{\max}}^{\ell-1}}{p^{2\ell}} \text{Var}\Big(\theta(w, G_\Omega)\mathbb{I}(w \subseteq G_\Omega)\Big) \;\le\; \frac{\omega {d_{\max}}^{\ell-1}}{p^{\ell}}\,. \qquad (7.65)$$

Combining, Equations (7.63), (7.64), and (7.65), and using $|w \in W :$

$\mathcal{H}(w) \in \mathbb{H}_{k,\ell,s}| \le f(k)\omega {d_{\max}}^{\ell-1}$, we have

$$
\begin{aligned}
\sum_{\ell=1}^{k}\sum_{s=0}^{k-\ell+1}\sum_{H\in\mathbb{H}_{k,\ell,s}} \Big\{\frac{1}{p^{2\ell}} &\sum_{w\in W:\mathcal{H}(w)=H} \mathbb{I}(w\subseteq G)\times \\
&\mathrm{Var}\Big(\theta(w,G_\Omega)\mathbb{I}(w\subseteq G_\Omega)\Big)\Big\} \\
&\le h(k)\omega d_{\max}p^{-2k} \;+\; \omega {d_{\max}}^{2k}p^{-1}\,, \qquad (7.66)
\end{aligned}
$$

and if $d_{\max}p^3 \le 1/h(k)$, then the above quantity is bounded by $g(k)\omega d_{\max}p^{-2k}(1+o(1))$. If ${d_{\max}}^3 p \ge h(k)$, then the above quantity is bounded by $\omega {d_{\max}}^{2k}p^{-1}(1+o(1))$.

Analysis of covariance terms in (7.32) follows along the similar lines as that of the clique graph case and the result in Lemma 7.9 follows immediately.

7.6.8 Proof of Lemma 7.10

We give a lemma similar to Lemma 7.7 for the case of a general graph that provides a bound on conditional variance and conditional covariance terms. We give a proof in Section 7.6.9.

Lemma 7.11. *Under the hypothesis of Lemma 7.9, for length-k walks w_1, w_2 over ℓ_1, ℓ_2 distinct nodes with $s_1, s_2 \ge 1$ self-loops respectively, the conditional variance of estimator $\theta(w_1, G_\Omega)$, defined in* (7.21)*, given that all the nodes in the walk are sampled can be upper bounded by*

$$
\begin{aligned}
\mathrm{Var}\Big(\theta(w_1,G_\Omega)\Big|w_1\subseteq G_\Omega\Big) \;&\le \\
& h(k)\Big({d_{\max}}^{2s_1} + d_{\max}p^{1-2s_1}\Big)\;,\;\; \textit{and} \qquad (7.67)\\
\mathbb{E}\Big[\theta(w_1,G_\Omega)\theta(w_2,G_\Omega)\Big|\mathbb{I}(w_1\subseteq G_\Omega)\mathbb{I}(w_2\subseteq G_\Omega)\Big] \;&\le \\
& h(k)p^{-(s_1+s_2)}\Big((d_{\max}p)^{s_1+s_2} + d_{\max}p\Big)\;, \qquad (7.68)
\end{aligned}
$$

for some function $h(k) = O(2^{k^2})$, where p is the vertex sampling probability. Moreover, for a length k walk w with $\ell = 1$, and $s = k$,

$$
\mathrm{Var}\Big(\theta(w,G_\Omega)\Big|w\subseteq G_\Omega\Big) \le f(k)p^{-1}{d_{\max}}^{2k-1} + g(k)d_{\max}p^{1-2k}\,, \qquad (7.69)
$$

for some function $f(k) = O(k!)$, and $g(k) = \mathrm{poly}(k)$.

Using the above lemma, proof of Lemma 7.10 follows along the lines of the proof of Lemma 7.6.

7.6.9 Proof of Lemma 7.11

Recall that for a general graph $\theta(w, G_\Omega)$ is an unbiased estimator of $\prod_{u \in w} d_u^{s_u}$ and is given in (7.21). It is easy to see that for any given walk w on ℓ distinct nodes and with s self-loops,

$$\mathrm{Var}\Big(\theta(w, G_\Omega)|w \subseteq G_\Omega\Big) \le h(k) \max_{\mathbb{T}} \left\{ \mathrm{Var}\Bigg(\Big\{ \prod_{T \in \mathbb{T}} \widehat{d}^{(t_T)}_{T,U\setminus T} \Big\}\Bigg) \right\},$$

where $h(k) = O(2^{k^2})$. It follows from the fact that there are at most $k/2$ distinct nodes with self loops and hence at most $2^{k/2-1}$ partitions in (7.14) which leads to at most $2^{k^2/4}$ summation terms in (7.16). Further $\prod_{T \in \mathbb{T}} \widehat{d}^{(t_T)}_{T,U\setminus T}$ is the product of independent random variables. Observe that using Lemma 7.7, we have

$$\mathrm{Var}\Big(\widehat{d}^{(t_T)}_{T,U\setminus T}\Big) \;\; \le \;\; f(k)\Big(p^{-1}{d_{\max}}^{2t_T-1} + d_{\max}p^{1-2t_T}\Big) \qquad (7.70)$$

and $\mathbb{E}[\widehat{d}^{(t_T)}_{T,U\setminus T}] \le {d_{\max}}^{t_T}$. Using the fact that for independent random variables $X_1, X_2, \cdots, X_n$,

$$\mathrm{Var}(X_1 X_2 \cdots X_n) = \prod_{i=1}^{n} \Big(\mathrm{Var}(X_i) + (\mathbb{E}[X_i])^2\Big) - \prod_{i=1}^{n} (\mathbb{E}[X_i])^2\,, \qquad (7.71)$$

we have,

$$\begin{aligned} &\max_{\mathbb{T}} \left\{ \mathrm{Var}\Bigg(\Big\{ \prod_{T \in \mathbb{T}} \widehat{d}^{(t_T)}_{T,U\setminus T} \Big\}\Bigg) \right\} \\ &\qquad \le \prod_{T \in \mathbb{T}} f(k)\Big(p^{-1}{d_{\max}}^{2t_T-1} + d_{\max}p^{1-2t_T} + {d_{\max}}^{2t_T}\Big) \\ &\qquad \le f(k)\Big({d_{\max}}^{2s} + d_{\max}p^{1-2s}\Big) \end{aligned} \qquad (7.72)$$

where in the last inequality we used that $\sum_{T \in \mathbb{T}} t_T = s$. (7.67) follows from collecting the above inequalities. (7.68) follows from the definition of $\theta(w, G_\Omega)$ given in (7.21) and the proof of (7.46) of Lemma 7.7. (7.69) follows directly from (7.47) of Lemma 7.7.

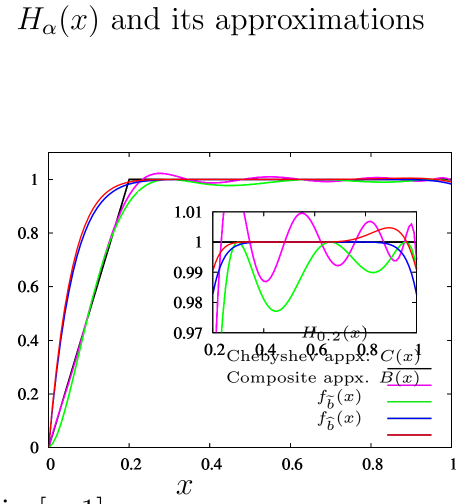
7.6.10 Proof of Proposition 7.2

$$\max_{x\in[\alpha,1]} |H_\alpha(x) - f_{b^*}(x)| \leq \max_{x\in[\alpha,1]} \left|H_\alpha(x) - f_{\widetilde{b}}(x)\right|, \tag{7.73}$$

$$= \max\left\{|H_\alpha(\alpha) - f_{\widetilde{b}}(\alpha)|, |H_\alpha(1) - f_{\widetilde{b}}(1)|\right\} \tag{7.74}$$

$$= \left(\frac{1-\alpha}{1+\alpha}\right)^m \tag{7.75}$$

where $\widetilde{b} \equiv (2/(1+\alpha))[1,\ldots,1]$.



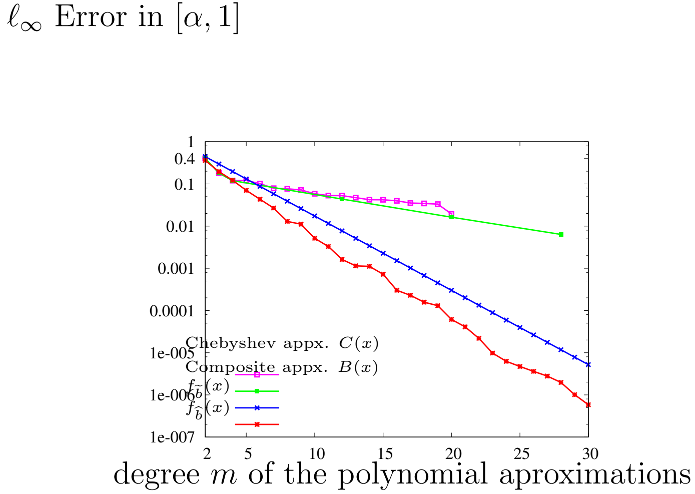


Figure 7.4: Top: four polynomial approximations to $H_{0.2}(x)$ of degree $k = 10$, $f_{\widehat{b}}(x)$ with $\widehat{b}$ chosen according to Algorithm 13, $f_{\widetilde{b}}(x)$ with $\widetilde{b} = (2/1.2)\mathbf{1}$ as prescribed above, Chebyshev approximation $C(x)$, and the composite approximation $B(x)$ from [217]. Bottom: approximation error achieved by the proposed $f_{\widehat{b}}(x)$ improves upon other polynomial functions.

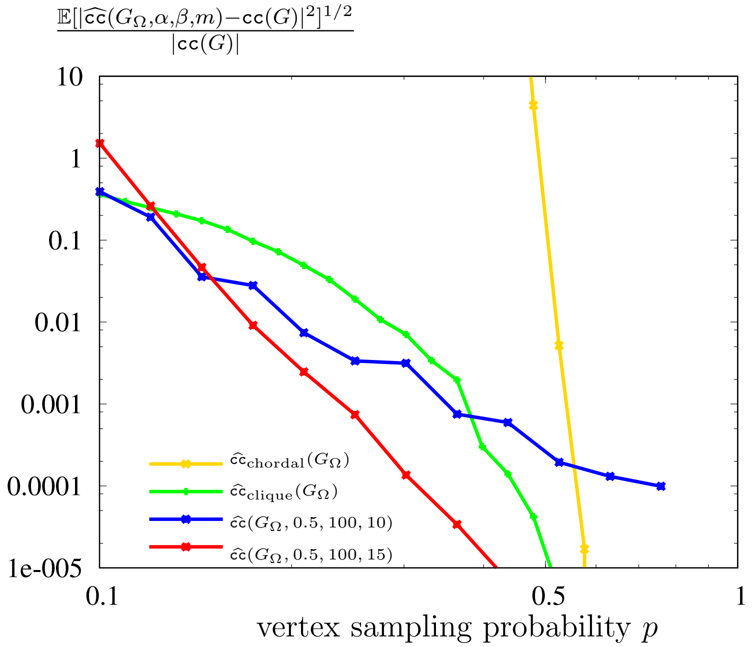


Figure 7.5: The proposed estimator for two choices of the degree m of the polynomial. With the right choice of m, we improve upon competing estimators when the original graph is a union of cliques.

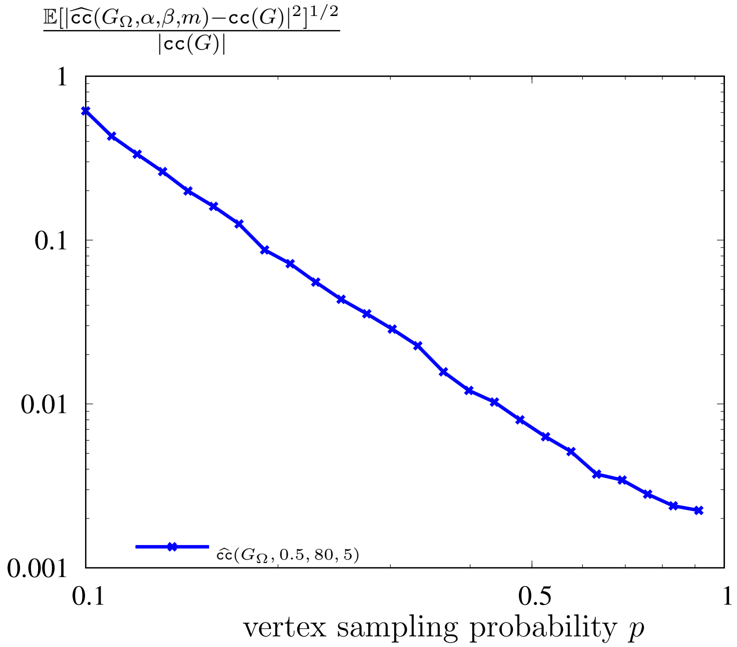


Figure 7.6: For general graphs, there is no competing algorithm and moderate m works when there is a spectral gap.

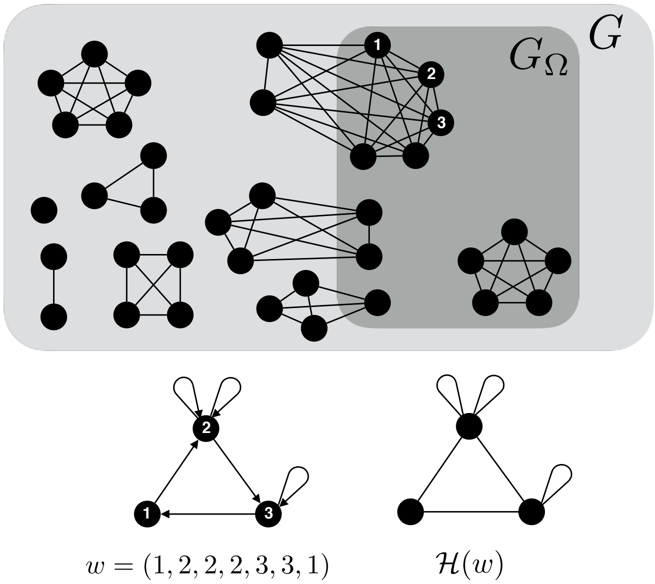


Figure 7.7: For a set of cliques G, we get a sampled G_Ω. An example of a length-($k = 6$) closed walk $w = (1, 2, 2, 2, 3, 3, 1)$ and its corresponding k-cyclic pseudograph $\mathcal{H}(w) \in \mathbb{H}_6$.

REFERENCES

[1] Dimitris Achlioptas and Frank McSherry. Fast computation of low rank matrix approximations. In *Proceedings of the thirty-third annual ACM symposium on Theory of computing*, pages 611–618. ACM, 2001.

[2] A. Agarwal, P. L. Bartlett, and J. C. Duchi. Oracle inequalities for computationally adaptive model selection. *arXiv preprint arXiv:1208.0129*, 2012.

[3] N. Ailon. Active learning ranking from pairwise preferences with almost optimal query complexity. In *Advances in Neural Information Processing Systems*, pages 810–818, 2011.

[4] A. Ali and M. Meilă. Experiments with kemeny ranking: What works when? *Mathematical Social Sciences*, 64(1):28–40, 2012.

[5] N. Alon and J. H. Spencer. *The probabilistic method*. John Wiley and Sons, 2004.

[6] N. Alon, R. Yuster, and U. Zwick. Finding and counting given length cycles. *Algorithmica*, 17(3):209–223, 1997.

[7] A. Ammar, S. Oh, D. Shah, and L. Voloch. What's your choice? learning the mixed multi-nomial logit model. In *Proceedings of the ACM SIGMETRICS/international conference on Measurement and modeling of computer systems*, 2014.

[8] A. Ammar and D. Shah. Ranking: Compare, don't score. In *Communication, Control, and Computing (Allerton), 2011 49th Annual Allerton Conference on*, pages 776–783. IEEE, 2011.

[9] A. Andoni, H. L. Nguyên, Y. Polyanskiy, and Y. Wu. Tight lower bound for linear sketches of moments. In *International Colloquium on Automata, Languages, and Programming*, pages 25–32. Springer, 2013.

[10] E. Aune, D. P. Simpson, and J. Eidsvik. Parameter estimation in high dimensional gaussian distributions. *Statistics and Computing*, 24(2):247–263, 2014.

[11] H. Avron and S. Toledo. Randomized algorithms for estimating the trace of an implicit symmetric positive semi-definite matrix. *Journal of the ACM (JACM)*, 58(2):8, 2011.

[12] H. Azari Soufiani, W. Chen, D. C Parkes, and L. Xia. Generalized method-of-moments for rank aggregation. In *Advances in Neural Information Processing Systems 26*, pages 2706–2714, 2013.

[13] H. Azari Soufiani, D. Parkes, and L. Xia. Computing parametric ranking models via rank-breaking. In *Proceedings of The 31st International Conference on Machine Learning*, pages 360–368, 2014.

[14] H. Azari Soufiani, D. C. Parkes, and L. Xia. Random utility theory for social choice. In *NIPS*, pages 126–134, 2012.

[15] M. E. Ben-Akiva and S. R. Lerman. *Discrete choice analysis: theory and application to travel demand*, volume 9. MIT press, 1985.

[16] Michael A Bender, Martin Farach-Colton, Giridhar Pemmasani, Steven Skiena, and Pavel Sumazin. Lowest common ancestors in trees and directed acyclic graphs. *Journal of Algorithms*, 57(2):75–94, 2005.

[17] P. Berenbrink, B. Krayenhoff, and F. Mallmann-Trenn. Estimating the number of connected components in sublinear time. *Information Processing Letters*, 114(11):639–642, 2014.

[18] Daniel Berend and Tamir Tassa. Improved bounds on bell numbers and on moments of sums of random variables. *Probability and Mathematical Statistics*, 30(2):185–205, 2010.

[19] N. Betzler, R. Bredereck, and R. Niedermeier. Theoretical and empirical evaluation of data reduction for exact kemeny rank aggregation. *Autonomous Agents and Multi-Agent Systems*, 28(5):721–748, 2014.

[20] S. Bhojanapalli and P. Jain. Universal matrix completion. In *Proceedings of the 31st International Conference on Machine Learning (ICML-14)*, pages 1881–1889, 2014.

[21] J. Blanchet, G. Gallego, and V. Goyal. A Markov chain approximation to choice modeling. In *EC*, pages 103–104, 2013.

[22] T. Bonald and R. Combes. Crowdsourcing: Low complexity, minimax optimal algorithms. *arXiv preprint arXiv:1606.00226*, 2016.

[23] T. Bonald and R. Combes. A streaming algorithm for crowdsourced data classification. *arXiv preprint arXiv:1602.07107*, 2016.

[24] C. Bordenave, M. Lelarge, and L. Massoulié. Non-backtracking spectrum of random graphs: community detection and non-regular ramanujan graphs. In *Foundations of Computer Science (FOCS), 2015 IEEE 56th Annual Symposium on*, pages 1347–1357. IEEE, 2015.

[25] O. Bousquet and L. Bottou. The tradeoffs of large scale learning. In *Advances in neural information processing systems*, pages 161–168, 2008.

[26] C. Boutsidis, P. Drineas, P. Kambadur, E.-M. Kontopoulou, and A. Zouzias. A randomized algorithm for approximating the log determinant of a symmetric positive definite matrix. *arXiv preprint arXiv:1503.00374*, 2015.

[27] Jonathan Bragg, Daniel S Weld, et al. Optimal testing for crowd workers. In *Proceedings of the 2016 International Conference on Autonomous Agents & Multiagent Systems*, pages 966–974. International Foundation for Autonomous Agents and Multiagent Systems, 2016.

[28] Steve Branson, Grant Van Horn, and Pietro Perona. Lean crowdsourcing: Combining humans and machines in an online system. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 7474–7483, 2017.

[29] Steve Branson, Catherine Wah, Florian Schroff, Boris Babenko, Peter Welinder, Pietro Perona, and Serge Belongie. Visual recognition with humans in the loop. *Computer Vision–ECCV 2010*, pages 438–451, 2010.

[30] M. Braverman and E. Mossel. Sorting from noisy information. *arXiv:0910.1191*, 2009.

[31] E. J. Candès and B. Recht. Exact matrix completion via convex optimization. *Foundations of Computational Mathematics*, 9(6):717–772, 2009.

[32] Emmanuel J Candès, Xiaodong Li, Yi Ma, and John Wright. Robust principal component analysis? *Journal of the ACM (JACM)*, 58(3):11, 2011.

[33] R. Carbó-Dorca. Smooth function topological structure descriptors based on graph-spectra. *Journal of Mathematical Chemistry*, 44(2):373–378, 2008.

[34] V. Chandrasekaran and M. I. Jordan. Computational and statistical tradeoffs via convex relaxation. *Proceedings of the National Academy of Sciences*, 110(13):E1181–E1190, 2013.

[35] B. Chazelle, R. Rubinfeld, and L. Trevisan. Approximating the minimum spanning tree weight in sublinear time. *SIAM Journal on computing*, 34(6):1370–1379, 2005.

[36] Pafnuti Lvovich Chebyshev. *Théorie of the mechanisms known as parallel e logrammes.* Critical Academy of Science Printing, 1853.

[37] J. Chen. How accurately should i compute implicit matrix-vector products when applying the hutchinson trace estimator? *SIAM Journal on Scientific Computing*, 38(6):A3515–A3539, 2016.

[38] Xiaowei Chen and John CS Lui. Mining graphlet counts in online social networks. In *Data Mining (ICDM), 2016 IEEE 16th International Conference on*, pages 71–80. IEEE, 2016.

[39] Y. Chen and C. Suh. Spectral mle: Top-k rank aggregation from pairwise comparisons. *arXiv:1504.07218*, 2015.

[40] C. Cortes, M. Mohri, and A. Rastogi. Magnitude-preserving ranking algorithms. In *Proceedings of the 24th international conference on Machine learning*, pages 169–176. ACM, 2007.

[41] Radu Curticapean, Holger Dell, and Dániel Marx. Homomorphisms are a good basis for counting small subgraphs. *arXiv preprint arXiv:1705.01595*, 2017.

[42] Artur Czumaj, MirosIaw Kowaluk, and Andrzej Lingas. Faster algorithms for finding lowest common ancestors in directed acyclic graphs. *Theoretical Computer Science*, 380(1-2):37–46, 2007.

[43] N. Dalvi, A. Dasgupta, R. Kumar, and V. Rastogi. Aggregating crowdsourced binary ratings. In *Proceedings of the 22nd international conference on World Wide Web*, pages 285–294, 2013.

[44] 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.

[45] J. V. Davis, B. Kulis, P. Jain, S. Sra, and I. S. Dhillon. Information-theoretic metric learning. In *Proceedings of the 24th international conference on Machine learning*, pages 209–216. ACM, 2007.

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

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

[48] J. C. de Borda. Mémoire sur les élections au scrutin. 1781.

[49] N. De Condorcet. *Essai sur l'application de l'analyse à la probabilité des décisions rendues à la pluralité des voix*. L'imprimerie royale, 1785.

[50] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei. ImageNet: A Large-Scale Hierarchical Image Database. In *CVPR09*, 2009.

[51] Jia Deng, Jonathan Krause, and Li Fei-Fei. Fine-grained crowdsourcing for fine-grained recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 580–587, 2013.

[52] Y. Deshpande and A. Montanari. Improved sum-of-squares lower bounds for hidden clique and hidden submatrix problems. *arXiv preprint arXiv:1502.06590*, 2015.

[53] E. Di Napoli, E. Polizzi, and Y. Saad. Efficient estimation of eigenvalue counts in an interval. *Numerical Linear Algebra with Applications*, 2016.

[54] P. Diaconis. A generalization of spectral analysis with application to ranked data. *The Annals of Statistics*, pages 949–979, 1989.

[55] W. Ding, P. Ishwar, and V. Saligrama. Learning mixed membership mallows models from pairwise comparisons. *arXiv preprint arXiv:1504.00757*, 2015.

[56] P. Donmez, J. G. Carbonell, and J. Schneider. Efficiently learning the accuracy of labeling sources for selective sampling. In *Proceedings of the 15th ACM SIGKDD*, pages 259–268. ACM, 2009.

[57] J. C. Duchi, L. Mackey, and M. I. Jordan. On the consistency of ranking algorithms. In *Proceedings of the ICML Conference*, Haifa, Israel, June 2010.

[58] C. Dwork, R. Kumar, M. Naor, and D. Sivakumar. Rank aggregation methods for the web. In *Proceedings of the 10th international conference on World Wide Web*, pages 613–622. ACM, 2001.

[59] O. Dykstra. Rank analysis of incomplete block designs: A method of paired comparisons employing unequal repetitions on pairs. *Biometrics*, 16(2):176–188, 1960.

[60] Khaled M Elbassioni. A polynomial delay algorithm for generating connected induced subgraphs of a given cardinality. *J. Graph Algorithms Appl.*, 19(1):273–280, 2015.

[61] E. R. Elenberg, K. Shanmugam, M. Borokhovich, and A. G. Dimakis. Beyond triangles: A distributed framework for estimating 3-profiles of large graphs. In *Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 229–238. ACM, 2015.

[62] E. R. Elenberg, K. Shanmugam, M. Borokhovich, and A. G. Dimakis. Distributed estimation of graph 4-profiles. In *Proceedings of the 25th International Conference on World Wide Web*, pages 483–493. International World Wide Web Conferences Steering Committee, 2016.

[63] Ş. Ertekin, H. Hirsh, and C. Rudin. Approximating the wisdom of the crowd. In *Proceedings of the Second Workshop on Computational Social Science and the Wisdom of Crowds (NIPS 2011)*, 2011.

[64] E. Estrada. Characterization of 3d molecular structure. *Chemical Physics Letters*, 319(5):713–718, 2000.

[65] E. Estrada and N. Hatano. Statistical-mechanical approach to subgraph centrality in complex networks. *Chemical Physics Letters*, 439(1):247–251, 2007.

[66] E. Estrada and J. A. Rodriguez-Velázquez. Spectral measures of bipartivity in complex networks. *Physical Review E*, 72(4):046105, 2005.

[67] V. F. Farias, S. Jagabathula, and D. Shah. A data-driven approach to modeling choice. In *NIPS*, pages 504–512, 2009.

[68] Vivek F Farias, Srikanth Jagabathula, and Devavrat Shah. A nonparametric approach to modeling choice with limited data. *Management Science*, 59(2):305–322, 2013.

[69] U. Feige and E. Ofek. Spectral techniques applied to sparse random graphs. *Random Struct. Algorithms*, 27(2):251–275, 2005.

[70] J. B. Feldman and H. Topaloglu. Revenue management under the markov chain choice model. 2014.

[71] J. Flum and M. Grohe. The parameterized complexity of counting problems. *SIAM Journal on Computing*, 33(4):892–922, 2004.

[72] L. R. Ford Jr. Solution of a ranking problem from binary comparisons. *The American Mathematical Monthly*, 64(8):28–33, 1957.

[73] O. Frank. Estimation of the number of connected components in a graph by using a sampled subgraph. *Scandinavian Journal of Statistics*, pages 177–188, 1978.

[74] O. Frank and F. Harary. Cluster inference by using transitivity indices in empirical graphs. *Journal of the American Statistical Association*, 77(380):835–840, 1982.

[75] J. Friedman, J. Kahn, and E. Szemerédi. On the second eigenvalue in random regular graphs. In *Proceedings of the Twenty-First Annual ACM Symposium on Theory of Computing*, pages 587–598, Seattle, Washington, USA, may 1989. ACM.

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

[77] A. Ghosh, S. Kale, and P. 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.

[78] K. Goldberg, T. Roeder, D. Gupta, and C. Perkins. Eigentaste: A constant time collaborative filtering algorithm. *Information Retrieval*, 4(2):133–151, 2001.

[79] Ryan G. Gomes, Peter Welinder, Andreas Krause, and Pietro Perona. Crowdclustering. In J. Shawe-Taylor, R. S. Zemel, P. L. Bartlett, F. Pereira, and K. Q. Weinberger, editors, *Advances in Neural Information Processing Systems 24*, pages 558–566. Curran Associates, Inc., 2011.

[80] L. A. Goodman. On the estimation of the number of classes in a population. *The Annals of Mathematical Statistics*, pages 572–579, 1949.

[81] F. Götze and A. Tikhomirov. On the rate of convergence to the semicircular law, preprint (2011). *arXiv preprint arXiv:1109.0611.*

[82] P. M. Guadagni and J. D. Little. A logit model of brand choice calibrated on scanner data. *Marketing science*, 2(3):203–238, 1983.

[83] Melody Y Guan, Varun Gulshan, Andrew M Dai, and Geoffrey E Hinton. Who said what: Modeling individual labelers improves classification. *arXiv preprint arXiv:1703.08774*, 2017.

[84] B. Hajek, S. Oh, and J. Xu. Minimax-optimal inference from partial rankings. In *Advances in Neural Information Processing Systems 27*, pages 1475–1483, 2014.

[85] T. R. Halford and K. M. Chugg. An algorithm for counting short cycles in bipartite graphs. *IEEE Transactions on Information Theory*, 52(1):287–292, 2006.

[86] Nathan Halko, Per-Gunnar Martinsson, and Joel A Tropp. Finding structure with randomness: Probabilistic algorithms for constructing approximate matrix decompositions. *SIAM review*, 53(2):217–288, 2011.

[87] I. Han, D. Malioutov, H. Avron, and J. Shin. Approximating the spectral sums of large-scale matrices using chebyshev approximations. *arXiv preprint arXiv:1606.00942*, 2016.

[88] I. Han, D. Malioutov, and J. Shin. Large-scale log-determinant computation through stochastic chebyshev expansions. In *ICML*, pages 908–917, 2015.

[89] T. P. Hayes. A large-deviation inequality for vector-valued martingales. *Combinatorics, Probability and Computing*, 2005.

[90] T. P. Hayes. A large-deviation inequality for vector-valued martingales. *Combinatorics, Probability and Computing*, 2005.

[91] C. Ho, S. Jabbari, and J. W. Vaughan. Adaptive task assignment for crowdsourced classification. In *Proceedings of the 30th International Conference on Machine Learning (ICML-13)*, pages 534–542, 2013.

[92] D. R. Hunter. Mm algorithms for generalized bradley-terry models. *Ann. of Stat.*, pages 384–406, 2004.

[93] David R Hunter and Kenneth Lange. Rejoinder. *Journal of Computational and Graphical Statistics*, 9(1):52–59, 2000.

[94] Michael F Hutchinson. A stochastic estimator of the trace of the influence matrix for laplacian smoothing splines. *Communications in Statistics-Simulation and Computation*, 19(2):433–450, 1990.

[95] Yuri Ingster and Irina A Suslina. *Nonparametric goodness-of-fit testing under Gaussian models*, volume 169. Springer Science & Business Media, 2012.

[96] Panagiotis G Ipeirotis, Foster Provost, Victor S Sheng, and Jing Wang. Repeated labeling using multiple noisy labelers. *Data Mining and Knowledge Discovery*, 28(2):402–441, 2014.

[97] P. Jain, P. Netrapalli, and S. Sanghavi. Low-rank matrix completion using alternating minimization. In *STOC*, pages 665–674, 2013.

[98] K. G. Jamieson and R. Nowak. Active ranking using pairwise comparisons. In *Advances in Neural Information Processing Systems*, pages 2240–2248, 2011.

[99] Madhav Jha, C Seshadhri, and Ali Pinar. Path sampling: A fast and provable method for estimating 4-vertex subgraph counts. In *Proceedings of the 24th International Conference on World Wide Web*, pages 495–505. International World Wide Web Conferences Steering Committee, 2015.

[100] R. Jin and Z. Ghahramani. Learning with multiple labels. In *Advances in neural information processing systems*, pages 921–928, 2003.

[101] Rong Jin and Zoubin Ghahramani. Learning with multiple labels. In *Advances in neural information processing systems*, pages 921–928, 2003.

[102] Ishan Jindal, Matthew Nokleby, and Xuewen Chen. Learning deep networks from noisy labels with dropout regularization. In *Data Mining (ICDM), 2016 IEEE 16th International Conference on*, pages 967–972. IEEE, 2016.

[103] Armand Joulin, Laurens van der Maaten, Allan Jabri, and Nicolas Vasilache. Learning visual features from large weakly supervised data. In *European Conference on Computer Vision*, pages 67–84. Springer, 2016.

[104] T. Kamishima. Nantonac collaborative filtering: recommendation based on order responses. In *Proceedings of the ninth ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 583–588. ACM, 2003.

[105] D. R. Karger, S. Oh, and D. Shah. Iterative learning for reliable crowdsourcing systems. In *Advances in neural information processing systems*, pages 1953–1961, 2011.

[106] D. R. Karger, S. Oh, and D. Shah. Efficient crowdsourcing for multi-class labeling. In *Proceedings of the ACM SIGMETRICS/international conference on Measurement and modeling of computer systems*, pages 81–92, 2013.

[107] D. R. Karger, S. Oh, and D. Shah. Budget-optimal task allocation for reliable crowdsourcing systems. *Operations Research*, 62:1–24, 2014.

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

[109] M. Karimi and A. H. Banihashemi. Message-passing algorithms for counting short cycles in a graph. *IEEE Transactions on Communications*, 61(2):485–495, 2013.

[110] R. H. Keshavan, A. Montanari, and S. Oh. Matrix completion from a few entries. *Information Theory, IEEE Transactions on*, 56(6):2980–2998, 2010.

[111] R. H Keshavan, A. Montanari, and S. Oh. Matrix completion from noisy entries. *Journal of Machine Learning Research*, 11(2057-2078):1, 2010.

[112] R. H. Keshavan and S. Oh. A gradient descent algorithm on the Grassman manifold for matrix completion. *arXiv preprint arXiv:0910.5260*, 2009.

[113] R. H. Keshavan, S. Oh, and A. Montanari. Matrix completion from a few entries. In *Information Theory, 2009. ISIT 2009. IEEE International Symposium on*, pages 324–328. IEEE, 2009.

[114] A. Khetan and S. Oh. Data-driven rank breaking for efficient rank aggregation. In *International Conference on Machine Learning*, 2016.

[115] A. Khetan and S. Oh. Spectrum estimation from a few entries. *arXiv preprint arXiv:1703.06327*, 2017.

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

[117] T. Kloks, D. Kratsch, and H. Müller. Finding and counting small induced subgraphs efficiently. *Information Processing Letters*, 74(3):115–121, 2000.

[118] J. M. Klusowski and Y. Wu. Estimating the number of connected components in a graph via subgraph sampling. *Technical report*, 2017. available at http://www.stat.yale.edu/∼yw562/preprints/cc.pdf.

[119] W. Kong and G. Valiant. Spectrum estimation from samples. *arXiv preprint arXiv:1602.00061*, 2016.

[120] Jonathan Krause, Benjamin Sapp, Andrew Howard, Howard Zhou, Alexander Toshev, Tom Duerig, James Philbin, and Li Fei-Fei. The unreasonable effectiveness of noisy data for fine-grained recognition. In *European Conference on Computer Vision*, pages 301–320. Springer, 2016.

[121] Alex Krizhevsky and Geoffrey Hinton. Learning multiple layers of features from tiny images. 2009.

[122] F. Krzakala, C. Moore, E. Mossel, J. Neeman, A. Sly, L. Zdeborová, and P. Zhang. Spectral redemption in clustering sparse networks. *Proceedings of the National Academy of Sciences*, 110(52):20935–20940, 2013.

[123] C. M. Le, E. Levina, and R. Vershynin. Sparse random graphs: regularization and concentration of the laplacian. *arXiv preprint arXiv:1502.03049*, 2015.

[124] T. Le Van, M. van Leeuwen, S. Nijssen, and L. De Raedt. Rank matrix factorisation. In *Advances in Knowledge Discovery and Data Mining*, pages 734–746. Springer, 2015.

[125] G. Lebanon and Y. Mao. Non-parametric modeling of partially ranked data. In *Advances in neural information processing systems*, pages 857–864, 2007.

[126] H. Li and B. Yu. Error rate bounds and iterative weighted majority voting for crowdsourcing. *arXiv preprint arXiv:1411.4086*, 2014.

[127] Y. Li, H. L. Nguyên, and D. P. Woodruff. On sketching matrix norms and the top singular vector. In *Proceedings of the Twenty-Fifth Annual ACM-SIAM Symposium on Discrete Algorithms*, pages 1562–1581. Society for Industrial and Applied Mathematics, 2014.

[128] Y. Li and D. P. Woodruff. On approximating functions of the singular values in a stream. *arXiv preprint arXiv:1604.08679*, 2016.

[129] Y. Li and D. P. Woodruff. Embeddings of schatten norms with applications to data streams. *arXiv preprint arXiv:1702.05626*, 2017.

[130] Christopher H Lin, M Mausam, and Daniel S Weld. Re-active learning: Active learning with relabeling. In *AAAI*, pages 1845–1852, 2016.

[131] Christopher H Lin, Daniel S Weld, et al. To re (label), or not to re (label). In *Second AAAI conference on human computation and crowdsourcing*, 2014.

[132] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco: Common objects in context. In *European conference on computer vision*, pages 740–755. Springer, 2014.

[133] H. Liu and J. Wang. A new way to enumerate cycles in graph. In *AICT/ICIW*, page 57, 2006.

[134] Q. Liu, J. Peng, and A. Ihler. Variational inference for crowdsourcing. In *Advances in Neural Information Processing Systems 25*, pages 701–709, 2012.

[135] Qiang Liu, Jian Peng, and Alexander T Ihler. Variational inference for crowdsourcing. In *Advances in neural information processing systems*, pages 692–700, 2012.

[136] T. Lu and C. Boutilier. Budgeted social choice: From consensus to personalized decision making. In *IJCAI*, volume 11, pages 280–286, 2011.

[137] T. Lu and C. Boutilier. Learning mallows models with pairwise preferences. In *Proceedings of the 28th International Conference on Machine Learning (ICML-11)*, pages 145–152, 2011.

[138] Y. Lu and S. Negahban. Individualized rank aggregation using nuclear norm regularization. *arXiv preprint arXiv:1410.0860*, 2014.

[139] M. Lucic, M. I. Ohannessian, A. Karbasi, and A. Krause. Tradeoffs for space, time, data and risk in unsupervised learning. In *AISTATS*, 2015.

[140] J. Lundell. Second report of the irish commission on electronic voting. *Voting Matters*, 23:13–17, 2007.

[141] Y. Ma, A. Olshevsky, V. Saligrama, and C. Szepesvari. Crowdsourcing with sparsely interacting workers. *arXiv preprint arXiv:1706.06660*, 2017.

[142] Michael W Mahoney et al. Randomized algorithms for matrices and data. *Foundations and Trends® in Machine Learning*, 3(2):123–224, 2011.

[143] J. C. Mason and D. C. Handscomb. *Chebyshev polynomials.* CRC Press, 2002.

[144] John C Mason and David C Handscomb. *Chebyshev polynomials.* CRC Press, 2002.

[145] L. Massoulie and K. Xu. On the capacity of information processing systems. *arXiv preprint arXiv:1603.00544*, 2016.

[146] L. Maystre and M. Grossglauser. Fast and accurate inference of plackett-luce models. In *Advances in Neural Information Processing Systems 28 (NIPS 2015)*, 2015.

[147] L. Maystre and M. Grossglauser. Robust active ranking from sparse noisy comparisons. *arXiv:1502.05556*, 2015.

[148] D. McFadden. Conditional logit analysis of qualitative choice behavior. *Frontiers in Econometrics*, pages 105–142, 1973.

[149] D. McFadden. Econometric models for probabilistic choice among products. *Journal of Business*, 53(3):S13–S29, 1980.

[150] D. McFadden and K. Train. Mixed mnl models for discrete response. *Journal of applied Econometrics*, 15(5):447–470, 2000.

[151] R. Meka, A. Potechin, and A. Wigderson. Sum-of-squares lower bounds for planted clique. In *Proceedings of the Forty-Seventh Annual ACM on Symposium on Theory of Computing*, pages 87–96. ACM, 2015.

[152] M. Mezard and A. Montanari. *Information, physics, and computation.* Oxford University Press, 2009.

[153] T. Mikolov, K. Chen, G. Corrado, and J. Dean. Efficient estimation of word representations in vector space. *arXiv preprint arXiv:1301.3781*, 2013.

[154] K. Miyahara and M. J. Pazzani. Collaborative filtering with the simple bayesian classifier. In *PRICAI 2000 Topics in Artificial Intelligence*, pages 679–689. Springer, 2000.

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

[156] S. Negahban, S. Oh, and D. Shah. Iterative ranking from pair-wise comparisons. In *NIPS*, pages 2483–2491, 2012.

[157] S. Negahban, S. Oh, and D. Shah. Rank centrality: Ranking from pair-wise comparisons. preprint arXiv:1209.1688, 2014.

[158] S. Negahban and M. J. Wainwright. Restricted strong convexity and (weighted) matrix completion: Optimal bounds with noise. *Journal of Machine Learning Research*, 2012. To appear; posted at http://arxiv.org/abs/1009.2118.

[159] Praneeth Netrapalli, UN Niranjan, Sujay Sanghavi, Animashree Anandkumar, and Prateek Jain. Non-convex robust pca. In *Advances in Neural Information Processing Systems*, pages 1107–1115, 2014.

[160] S. Oh and D. Shah. Learning mixed multinomial logit model from ordinal data. In *Advances in Neural Information Processing Systems*, pages 595–603, 2014.

[161] S. Oh, K. K. Thekumparampil, and J. Xu. Collaboratively learning preferences from ordinal data. In *Advances in Neural Information Processing Systems 28*, pages 1900–1908, 2015.

[162] J. Ok, S. Oh, J. Shin, and Y. Yi. Optimality of belief propagation for crowdsourced classification. In *International Conference on Machine Learning*, 2016.

[163] R. K. Pace and J. P. LeSage. Chebyshev approximation of log-determinants of spatial weight matrices. *Computational Statistics & Data Analysis*, 45(2):179–196, 2004.

[164] D. Park, J. Neeman, J. Zhang, S. Sanghavi, and I. S. Dhillon. Preference completion: Large-scale collaborative ranking from pairwise comparisons. In *Proceedings of The 32nd International Conference on Machine Learning*, pages 1907–1916, 2015.

[165] T Parks and J McClellan. Chebyshev approximation for nonrecursive digital filters with linear phase. *IEEE Transactions on Circuit Theory*, 19(2):189–194, 1972.

[166] H. Polat and W. Du. Svd-based collaborative filtering with privacy. In *Proceedings of the 2005 ACM symposium on Applied computing*, pages 791–795. ACM, 2005.

[167] Eric Polizzi. Density-matrix-based algorithm for solving eigenvalue problems. *Physical Review B*, 79(11):115112, 2009.

[168] A. Prékopa. Logarithmic concave measures and related topics. In *Stochastic programming*, 1980.

[169] D. Rafiei. Effectively visualizing large networks through sampling. In *Visualization, 2005. VIS 05. IEEE*, pages 375–382. IEEE, 2005.

[170] Mahmudur Rahman, Mansurul Alam Bhuiyan, and Mohammad Al Hasan. Graft: An efficient graphlet counting method for large graph analysis. *IEEE Transactions on Knowledge and Data Engineering*, 26(10):2466–2478, 2014.

[171] A. Rajkumar and S. Agarwal. A statistical convergence perspective of algorithms for rank aggregation from pairwise data. In *Proceedings of The 31st International Conference on Machine Learning*, pages 118–126, 2014.

[172] Alexander J Ratner, Christopher M De Sa, Sen Wu, Daniel Selsam, and Christopher Ré. Data programming: Creating large training sets, quickly. In *Advances in Neural Information Processing Systems*, pages 3567–3575, 2016.

[173] Paramesh Ray. Independence of irrelevant alternatives. *Econometrica: Journal of the Econometric Society*, pages 987–991, 1973.

[174] Tom Richardson and Ruediger Urbanke. *Modern coding theory*. Cambridge university press, 2008.

[175] F. Roosta-Khorasani and U. Ascher. Improved bounds on sample size for implicit matrix trace estimators. *Foundations of Computational Mathematics*, 15(5):1187–1212, 2015.

[176] H. Rue and L. Held. *Gaussian Markov random fields: theory and applications*. CRC Press, 2005.

[177] A. Saade, F. Krzakala, and L. Zdeborová. Matrix completion from fewer entries: Spectral detectability and rank estimation. In *Advances in Neural Information Processing Systems*, pages 1261–1269, 2015.

[178] Tetsuya Sakurai and Hiroshi Sugiura. A projection method for generalized eigenvalue problems using numerical integration. *Journal of computational and applied mathematics*, 159(1):119–128, 2003.

[179] G. Schofield, J. R. Chelikowsky, and Y. Saad. A spectrum slicing method for the kohn–sham problem. *Computer Physics Communications*, 183(3):497–505, 2012.

[180] W. Schudy and M. Sviridenko. Bernstein-like concentration and moment inequalities for polynomials of independent random variables: multilinear case. *arXiv preprint arXiv:1109.5193*, 2011.

[181] Comandur Seshadhri, Ali Pinar, and Tamara G Kolda. Triadic measures on graphs: The power of wedge sampling. In *Proceedings of the 2013 SIAM International Conference on Data Mining*, pages 10–18. SIAM, 2013.

[182] N. B. Shah, S. Balakrishnan, J. Bradley, A. Parekh, K. Ramchandran, and M. J. Wainwright. Estimation from pairwise comparisons: Sharp minimax bounds with topology dependence. *arXiv:1505.01462*, 2015.

[183] N. B. Shah, S. Balakrishnan, A. Guntuboyina, and M. J. Wainright. Stochastically transitive models for pairwise comparisons: Statistical and computational issues. *arXiv preprint arXiv:1510.05610*, 2015.

[184] N. B. Shah, S. Balakrishnan, A. Guntuboyina, and M. J. Wainright. Stochastically transitive models for pairwise comparisons: Statistical and computational issues. *arXiv preprint arXiv:1510.05610*, 2015.

[185] N. B. Shah, S. Balakrishnan, and M. J. Wainwright. A permutation-based model for crowd labeling: Optimal estimation and robustness. *arXiv preprint arXiv:1606.09632*, 2016.

[186] N. B. Shah and M. J. Wainwright. Simple, robust and optimal ranking from pairwise comparisons. *arXiv preprint arXiv:1512.08949*, 2015.

[187] S. Shalev-Shwartz and N. Srebro. Svm optimization: inverse dependence on training set size. In *Proceedings of the 25th international conference on Machine learning*, pages 928–935. ACM, 2008.

[188] P. Sham and D. Curtis. An extended transmission/disequilibrium test (tdt) for multi-allele marker loci. *Annals of human genetics*, 59(3):323–336, 1995.

[189] S. S. Shen-Orr, R. Milo, S. Mangan, and U. Alon. Network motifs in the transcriptional regulation network of escherichia coli. *Nature genetics*, 31(1):64–68, 2002.

[190] V. S Sheng, F. Provost, and P. G. Ipeirotis. Get another label? improving data quality and data mining using multiple, noisy labelers. In *Proceedings of the 14th ACM SIGKDD*, pages 614–622. ACM, 2008.

[191] Victor S Sheng, Foster Provost, and Panagiotis G Ipeirotis. Get another label? improving data quality and data mining using multiple, noisy labelers. In *Proceedings of the 14th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 614–622. ACM, 2008.

[192] G. Simons and Y. Yao. Asymptotics when the number of parameters tends to infinity in the bradley-terry model for paired comparisons. *The Annals of Statistics*, 27(3):1041–1060, 1999.

[193] P. Smyth, U. Fayyad, M. Burl, P. Perona, and P. Baldi. Inferring ground truth from subjective labelling of venus images. In *NIPS*, pages 1085–1092, 1995.

[194] A. Stathopoulos, J. Laeuchli, and K. Orginos. Hierarchical probing for estimating the trace of the matrix inverse on toroidal lattices. *SIAM Journal on Scientific Computing*, 35(5):S299–S322, 2013.

[195] Sainbayar Sukhbaatar, Joan Bruna, Manohar Paluri, Lubomir Bourdev, and Rob Fergus. Training convolutional networks with noisy labels. *arXiv preprint arXiv:1406.2080*, 2014.

[196] T. Tian, C. R. Jones, J. D. Villasenor, and R. D. Wesel. Selective avoidance of cycles in irregular ldpc code construction. *IEEE Transactions on Communications*, 52(8):1242–1247, 2004.

[197] Ryuhei Uehara et al. The number of connected components in graphs and its applications. *Manuscript. URL: http://citeseerx. ist. psu. edu/viewdoc/summary*, 1999.

[198] J. Ugander, L. Backstrom, and J. Kleinberg. Subgraph frequencies: Mapping the empirical and extremal geography of large graph collections. In *Proceedings of the 22nd international conference on World Wide Web*, pages 1307–1318. ACM, 2013.

[199] S. Van Aelst and P. Rousseeuw. Minimum volume ellipsoid. *Wiley Interdisciplinary Reviews: Computational Statistics*, 1(1):71–82, 2009.

[200] Catherine Wah, Steve Branson, Pietro Perona, and Serge Belongie. Multiclass recognition and part localization with humans in the loop. In *Computer Vision (ICCV), 2011 IEEE International Conference on*, pages 2524–2531. IEEE, 2011.

[201] Joan Walker and Moshe Ben-Akiva. Generalized random utility model. *Mathematical Social Sciences*, 43(3):303–343, 2002.

[202] P. Wang, J. Lui, B. Ribeiro, D. Towsley, J. Zhao, and X. Guan. Efficiently estimating motif statistics of large networks. *ACM Transactions on Knowledge Discovery from Data (TKDD)*, 9(2):8, 2014.

[203] Pinghui Wang, Jing Tao, Junzhou Zhao, and Xiaohong Guan. Moss: A scalable tool for efficiently sampling and counting 4-and 5-node graphlets. *arXiv preprint arXiv:1509.08089*, 2015.

[204] P. Welinder, S. Branson, S. Belongie, and P. Perona. The multidimensional wisdom of crowds. In *Advances in Neural Information Processing Systems*, pages 2424–2432, 2010.

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

[206] Peter Welinder and Pietro Perona. Online crowdsourcing: rating annotators and obtaining cost-effective labels. In *Computer Vision and Pattern Recognition Workshops (CVPRW), 2010 IEEE Computer Society Conference on*, pages 25–32. IEEE, 2010.

[207] J. Whitehill, P. Ruvolo, T. Wu, J. Bergsma, and J. Movellan. Whose vote should count more: Optimal integration of labels from labelers of unknown expertise. In *Advances in Neural Information Processing Systems*, volume 22, pages 2035–2043, 2009.

[208] Jacob Whitehill, Ting-fan Wu, Jacob Bergsma, Javier R Movellan, and Paul L Ruvolo. Whose vote should count more: Optimal integration of labels from labelers of unknown expertise. In *Advances in neural information processing systems*, pages 2035–2043, 2009.

[209] E. P. Wigner. Characteristic vectors of bordered matrices with infinite dimensions. *Annals of Mathematics*, page 548564, 1955.

[210] D. Williams. *Probability with martingales.* Cambridge university press, 1991.

[211] L. Wu, J. Laeuchli, V. Kalantzis, A. Stathopoulos, and E. Gallopoulos. Estimating the trace of the matrix inverse by interpolating from the diagonal of an approximate inverse. *Journal of Computational Physics*, 326:828–844, 2016.

[212] R. Wu, J. Xu, R. Srikant, L. Massoulié, M. Lelarge, and B. Hajek. Clustering and inference from pairwise comparisons. *arXiv preprint arXiv:1502.04631*, 2015.

[213] J. Yi, R. Jin, S. Jain, and A. Jain. Inferring users? preferences from crowdsourced pairwise comparisons: A matrix completion approach. In *First AAAI Conference on Human Computation and Crowdsourcing*, 2013.

[214] E. Zermelo. Die berechnung der turnier-ergebnisse als ein maximumproblem der wahrscheinlichkeitsrechnung. *Mathematische Zeitschrift*, 29(1):436–460, 1929.

[215] Y. Zhang, X. Chen, D. Zhou, and M. I. Jordan. Spectral methods meet em: A provably optimal algorithm for crowdsourcing. In *Advances in neural information processing systems*, pages 1260–1268, 2014.

[216] Y. Zhang and W. E. Leithead. Approximate implementation of the logarithm of the matrix determinant in gaussian process regression. *Journal of Statistical Computation and Simulation*, 77(4):329–348, 2007.

[217] Y. Zhang, M. Wainwright, and M. Jordan. Distributed estimation of generalized matrix rank: Efficient algorithms and lower bounds. In *Proceedings of the 32nd International Conference on Machine Learning (ICML-15)*, pages 457–465, 2015.

[218] 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.

[219] Y. Zheng, S. Scott, and K. Deng. Active learning from multiple noisy labelers with varied costs. In *Data Mining (ICDM), 2010 IEEE 10th International Conference on*, pages 639 –648, dec. 2010.

[220] D. Zhou, Q. Liu, J. C. Platt, C. Meek, and N. B. Shah. Regularized minimax conditional entropy for crowdsourcing. *arXiv preprint arXiv:1503.07240*, 2015.

[221] D. Zhou, J. Platt, S. Basu, and Y. Mao. Learning from the wisdom of crowds by minimax entropy. In *Advances in Neural Information Processing Systems 25*, pages 2204–2212, 2012.

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

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