LEARNING NETWORK STRUCTURE FROM NODE BEHAVIOR

BY

RUI WU

DISSERTATION

Submitted in partial fulfillment of the requirements
for the degree of Doctor of Philosophy in Electrical and Computer Engineering
in the Graduate College of the
University of Illinois at Urbana-Champaign, 2014

Urbana, Illinois

Doctoral Committee:

Professor R. Srikant, Chair
Professor Bruce Hajek
Assistant Professor Sewoong Oh
Professor Venugopal V. Veeravalli

ABSTRACT

Understanding the network structure connecting a group of entities is of interest in applications such as predicting stock prices and making recommendations to customers. The network structure is usually not directly observable. However, due to improvements in technology and the ever-increasing use of the Internet, large amounts of data about individual node behavior is becoming more easily available. Thus, an interesting problem is to devise algorithms to infer network structure from node behavior data. Since network sizes are enormous in typical applications, the learning problem is not tractable for general network topology. In this thesis, we focus on three models with simplifying assumptions on the underlying network.

The first model represents the network as a Markov random field, where each node in the network is viewed as a random variable and the conditional independence relations among them is encoded by a graph. The simplifying assumption is that the underlying graph is loosely connected: the number of short paths between any pair of nodes is small. We point out that many previously studied models are examples of this family. Given i.i.d. samples from the joint distribution, we present a natural low complexity algorithm for learning the structure of loosely connected Markov random fields. In particular, our algorithm learns the graph correctly with high probability using $n = O(\log p)$ samples, where p is the size of the graph. If there are at most D_1 short paths between non-neighbor nodes and D_2 non-direct short paths between neighboring nodes, the running time of our algorithm is $O(np^{D_1+D_2+2})$.

The second model arises from the recommender systems where users give ratings to items. We make the assumption that both users and items form clusters and users in the same cluster give the same binary rating to items in the same cluster. The goal is to recover the user and item clusters by observing only a small fraction of noisy entries. We first derive a lower bound on the minimum number of observations needed for exact cluster

recovery. Then, we study three algorithms with different running time and compare the number of observations needed for successful cluster recovery. Our analytical results show smooth time-data trade-offs: one can gradually reduce the computational complexity when increasingly more observations are available.

The third model considers a similar scenario as the previous one: instead of giving binary ratings, users give pairwise comparisons to items. We assume the users form clusters where users in the same cluster share the same score vector for the items, and the pairwise comparisons obtained from each user are generated according to the Bradley-Terry model with his/her score vector. We propose a two-step algorithm for estimating the score vectors: first cluster the users using projected comparison vectors and then estimate a score vector separately for each cluster by the maximum likelihood estimation for the classical Bradley-Terry model. The key observation is that, though each user is represented by a high-dimensional comparison vector, the corresponding expected comparison vector is determined by only a small number of parameters and it lies close to a low-dimensional linear subspace. When projecting the comparison vectors onto this subspace, it significantly reduces the noise and improves the clustering performance. Moreover, we show that the maximum likelihood estimation is robust to clustering errors.

To my parents

ACKNOWLEDGMENTS

Firstly I would like to thank my advisor Prof. R. Srikant for his guidance throughout my Ph.D. study. He is a very supportive and encouraging mentor. The most important thing I have learned from him is how to look at problems from a more fundamental perspective. I am especially grateful when he sits down patiently to help me edit the papers sentence by sentence.

I would like to thank Prof. Bruce Hajek, Prof. Sewoong Oh and Prof. Venugopal V. Veeravalli for being my thesis committee members. It is a great learning experience for me sitting in Prof. Bruce Hajek's group meetings and Prof. Sewoong Oh has given me many insightful suggestions during our discussions. I would also like to thank Dr. Marc Lelarge and Dr. Laurent Massoulié for the valuable intern opportunity in Paris.

I would like to thank Yuxin Chen for his hearty support. He greatly influenced me at every stage of my graduate study years. His aesthetic has led me to truly appreciate the original beauty of research. I would also like to thank Jiaming Xu for inspiring discussions and fruitful collaborations, which helped me move forward to a new stage in research.

I have been fortunate to have many friends who have made my years at the University of Illinois at Urbana-Champaign an unforgettable experience. My colleagues around the Coordinated Science Laboratory are like a big family and we have shared so much memorable time together. The deepest thanks goes to my friends in the Karaoke and Board Game Club. They have affected almost every aspects of my life. I enjoyed all the time playing games, partying, and traveling with them, and there are just so many moments to remember. I would also like to thank A-Team. Badminton is always my favorite sport and I am so grateful for the experience training with them. In order not to go beyond one page, I chose to keep all the names in my heart.

Finally I would like to thank my parents and Kejia for their understanding and support.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

CHAPTER 1

INTRODUCTION

In many applications of interest, we wish to understand the network structure connecting a group of entities. For example, in the stock market, knowing how the stocks depend on each other allows one to better predict the trend of the stock prices using current information. As another example, consider recommender systems that recommend items to potential customers. Since people with similar tastes behave similarly, recommender systems can increase the chance of a user making a purchase by studying the past behavior of users similar to himself/herself.

Network structures are usually not directly observable. In stock markets, only the individual stock prices are observed; in gene regulatory networks, only the individual gene expression levels are measured; in social networks, even when friendship relationships are available as in Facebook, it is not immediately helpful, as the networks of friendships are not always the same as networks representing the opinions or preferences of the users.

Due to improvements in technology and the ever-increasing use of the Internet, large amounts of data about individual node behavior is becoming more easily available. The data from individual nodes are not independent, and their correlation can provide information about the structure of the network. Thus, an interesting problem is to devise algorithms to infer network structure from node behavior data.

In typical applications, network sizes are enormous, and learning their structures is not tractable without some reasonable assumptions on the network topology. In this thesis, we consider three models with simplifying assumptions.

1.1 Learning Markov Random Fields

In Chapter 2 of the thesis, we consider a problem in which the network corresponds to a Markov random field and we wish to learn the corresponding graph. Each node in the network represents a random variable and the graph encodes the conditional independence relations among the random variables. The lack of an edge between two nodes implies that the two random variables are independent, conditioned on all the other random variables in the network. We observe only the nodes' behaviors, and do not observe or are unable to observe, interactions between the nodes. Our goal is to infer relationships among the nodes in such a network by understanding the correlations among them.

The canonical example used to illustrate such inference problems is the U.S. Senate [1]. Suppose one has access to the voting patterns of the senators over a number of bills (and not their party affiliations or any other information), the question we would like to answer is the following: can we say that a particular senator's vote is independent of everyone else's when conditioned on a few other senators' votes? In other words, if we view the senators' actions as forming a Markov Random Field (MRF), we want to infer the topology of the underlying graph.

Learning the underlying graph structure of a Markov random field, i.e., structure learning, refers to the problem of determining if there is an edge between each pair of nodes, given i.i.d. samples from the joint distribution of the Markov random field. In general, learning high-dimensional densely connected graphical models requires a large number of samples, and is usually computationally intractable. In this thesis, we consider the structure learning problem for graphical models that we call loosely connected Markov random fields [2], in which the number of short paths between any pair of nodes is small. We show that many previously studied models are examples of this family.

However, loosely connected MRFs are not always easy to learn. When there are short cycles in the graph, the dependence over an edge connecting a pair of neighboring nodes can be approximately cancelled by some short non-direct paths between them, in which case correctly detecting this edge is difficult, as shown in the following very simple example. This example is perhaps well known, but we present it here to motivate our algorithm

presented later.

Example 1.1. *Consider three binary random variables* $X_i \in \{0,1\}, i = 1,2,3$. *Assume* X_1, X_2 *are independent Bernoulli*$(\frac{1}{2})$ *random variables and* $X_3 = X_1 \oplus X_2$ *with probability* 0.9*, where* $\oplus$ *means exclusive or. We note that this joint distribution is symmetric, i.e., we get the same distribution if we assume that* X_2, X_3 *are independent Bernoulli*$(\frac{1}{2})$ *and* $X_1 = X_2 \oplus X_3$ *with probability* 0.9*. Therefore, the underlying graph is a triangle. However, it is not hard to see that the three random variables are marginally independent. Therefore, previous methods in [3, 4] would return an empty graph and fail to learn the true graph.* □

We propose a new algorithm that correctly learns the graphs for loosely connected MRFs. For each node, the algorithm loops over all the other nodes to determine if they are neighbors of this node. The key step in the algorithm is a max-min conditional independence test, in which the maximization step is to detect the edges while the minimization step is to detect non-edges. We focus on computational complexity rather than sample complexity in comparing our algorithm with previous algorithms. In fact, it has been shown that $\Omega(\log p)$ samples are required to learn the graph correctly with high probability, where p is the size of the graph [5]. For all the previously known algorithms for which analytical complexity bounds are available, the number of samples required to recover the graph correctly with high probability, i.e, the sample complexity, is $O(\log p)$. Not surprisingly, the sample complexity for our algorithm is also $n = O(\log p)$ under reasonable assumptions.

For loosely connected Markov random fields, if there are at most D_1 short paths between non-neighbor nodes and D_2 non-direct short paths between neighboring nodes, the running time of our algorithm is $O(np^{D_1+D_2+2})$. If in addition the Markov random field has correlation decay and satisfies a pairwise non-degeneracy condition, an extended algorithm with a preprocessing step can be applied and the running time is reduced to $O(np^2)$. In several special cases of loosely connected Markov random fields, our algorithm achieves the same or lower computational complexity than the previously designed algorithms for individual cases.

1.2 Clustering Users and Items

In Chapter 3 of the thesis, we study recommender systems and want to understand both the structure of the users and items. Recommender systems are now in widespread use in online commerce to assist users in finding interesting items and information. For instance, Amazon recommends products, Netflix recommends items, Google recommends articles and so on. These systems predict the interest of a user and make recommendations using the past behavior of all users. The underlying assumption is that, if two users have the same preferences on a set of items, then they are more likely to have the same preferences on another set of items than two randomly picked users.

We consider a simple model introduced in [6, 7] for generating a binary data matrix from underlying row and column clusters. Assumes that both users and items form clusters. Users in the same cluster give the same rating to items in the same cluster, where ratings are either $+1$ or -1 with $+1$ being "like" and -1 being "dislike". Each rating is flipped independently with a fixed flipping probability less than $1/2$, modeling the noisy user behavior and the fact that users (items) in the same cluster do not necessarily give (receive) identical ratings. Each rating is further erased independently with an erasure probability, modeling the fact that some ratings are not observed. Then, from the observed noisy ratings, we aim to *exactly* recover the underlying user and item clusters, i.e., jointly cluster the rows and columns of the observed rating matrix.

Data matrices exhibiting both row and column cluster structure arise in many other applications as well, such as gene expression analysis and text mining. The binary assumption on data matrices is of practical interest. Firstly, in many real datasets like the Netflix dataset and DNA microarrays, estimation of entry values appears to be very unreliable, but the task of determining whether an entry is $+1$ or -1 can be done more reliably [7]. Secondly, in recommender systems like rating music on Pandora or rating posts on sites such as Facebook and MathOverflow, the user ratings are indeed binary [8].

The hardness of our cluster recovery problem is governed by the erasure probability and cluster size. Intuitively, cluster recovery becomes harder when the erasure probability increases, meaning fewer observations, and the

cluster size decreases, meaning that clusters are harder to detect.

We first derive a lower bound on the minimum number of observations needed for exact cluster recovery as a function of matrix dimension and cluster size. Then we propose three algorithms with different running times and compare the number of observations needed by them for successful cluster recovery.

- The first algorithm directly searches for the optimal clustering of rows and columns separately; it is combinatorial in nature and takes exponential time but achieves the best statistical performance among the three algorithms in the noiseless setting.

- By noticing that the underlying true rating matrix is a specific type of low-rank matrix, the second algorithm recovers the clusters by solving a nuclear norm regularized convex optimization problem, which is a popular heuristic for low-rank matrix completion problems; it takes polynomial-time but performs worse than the first algorithm.

- The third algorithm applies spectral clustering to the rows and columns separately and then performs a joint clean-up step; it has lower computational complexity than the previous two algorithms, but less powerful statistical performance.

These algorithms are then compared with a simple nearest-neighbor clustering algorithm proposed in [7]. Our analytical results show a smooth time-data trade-off: when more observations are available, one can gradually reduce the computational complexity by applying simpler algorithms while still achieving the desired performance. Such a time-data trade-off is of great practical interest for statistical learning problems involving large datasets [9].

1.3 Clustering Users and Ranking Items

In Chapter 4 of the thesis, we consider the problem of ranking items with pairwise comparisons obtained from multiple types of users. This scenario is similar to the previous one, but instead of giving binary ratings, users provide pairwise comparisons of items.

The problem of estimating a ranking for items using pairwise comparisons is of interest in many applications. Some typical examples are from sports where pairs of players play against each other and people are interested in knowing which are the best players from past games. There are other examples where comparisons are obtained implicitly. For example, when a user clicks a result from a list returned by a search engine for a given request, it implies that this user prefers this result over nearby results on the list. Similarly, when a customer buys a product from an online retailer, it implies that this customer prefers this product over previously browsed products. Websites providing these services are interested in inferring users' ranking of items and displaying the top choices for each user to maximize their profit.

The Bradley-Terry model is a well-studied ranking model [10] where each item i is associated with a score θ_i and

$$\mathbb{P}\left[\text{item } i \text{ is preferred over item } j\right] = \frac{e^{\theta_i}}{e^{\theta_i} + e^{\theta_j}}.$$

Let R_{ij} be the number of times item i is preferred over item j, then the ranking problem can be solved by maximum likelihood estimation:

$$\hat{\theta} = \arg\max_{\gamma} \sum_{ij} R_{ij} \log \frac{e^{\gamma_i}}{e^{\gamma_i} + e^{\gamma_j}}.$$

The above optimization is convex, thus can be solved efficiently [11]. Further, the recent work [12] provides an error bound for $\hat{\theta}$ when the pairs of items are chosen uniformly and independently.

In examples like search engines and online retailers, users can have different scores for the same item and a single ranking is no longer sufficient to capture individual preferences. It is more realistic to assume that users form clusters and only users in the same cluster share the same score vector. The difficulty in this new problem is that the clusters are not known *a priori*, therefore, simply treating all users as a single type is likely to result in a meaningless global ranking as the user preferences can be conflicting.

To address this issue, we propose a two-step algorithm for estimating the score vectors: it first clusters the users and then estimate a score vector for each cluster separately. Let m be the number of items ranked by the users. The key observation is that, though each user is represented by a high-dimensional comparison vector of length $\binom{m}{2}$, the corresponding expected

comparison vector is completely determined by a length m score vector and it is close to an m-dimensional linear subspace. Therefore, we can first project the comparison vectors onto this subspace to reduce the noise while still preserving the separation between the clusters. The nontrivial part of the result here is that a nonlinear $\binom{m}{2}$-dimensional function of the m-dimensional vector is projected onto an m-dimensional subspace using a linearization approximation, but the projection performs denoising even in regimes where linearization may be a poor approximation to the nonlinear function.

In the first step of our algorithm, we consider two clustering algorithms using the projected comparison vectors. The first one clusters the vectors directly and the second one is a spectral clustering type of algorithm. Both algorithms show superior performance when compared to the standard spectral clustering algorithm using the original comparison vectors. In the second step of our algorithm, we treat each cluster separately and estimate a score vector using the maximum likelihood estimation for the single cluster Bradley-Terry model. We show that the maximum likelihood estimation is robust to clustering errors: as long as the number of misclustered users is small compared to the cluster size, it is still sufficient to recover the score vectors for most users with a good error bound as in [12].

1.4 Notations

A variety of norms on matrices will be used. Assume the matrix $X \in \mathbb{R}^{n\times m}$. Define the l_1 norm of X as $\|X\|_1 = \sum_{i,j} |X_{ij}|$ and the l_∞ norm of X as $\|X\|_\infty = \max_{i,j} |X_{ij}|$. Let $\langle X, Y\rangle = \mathrm{Tr}\left[X^\top Y\right]$ denote the inner product between two matrices X and Y of the same dimension, and the Frobenius norm of X is defined as $||X||_F = \langle X, X\rangle^{1/2}$. Let $X = \sum_{t=1}^{\min\{n,m\}} \sigma_t u_t v_t^\top$ denote the singular value decomposition of X such that $\sigma_1 \geq \cdots \geq \sigma_{\min\{n,m\}}$. The spectral norm of a matrix X is denoted by $\|X\|$, which is equal to the largest singular value. The nuclear norm is denoted by $\|X\|_*$ which is equal to the sum of singular values and is a convex function of X. The best rank r approximation of X is defined as $P_r(X) = \sum_{t=1}^{r} \sigma_t u_t v_t^\top$. For vectors, let $\langle x, y\rangle$ denote the inner product between two vectors and the only norm that will be used is the usual l_2 norm, denoted as $\|x\|_2$.

Throughout the thesis, we say that an event occurs "a.a.s." or "asymptot-

ically almost surely" when it occurs with a probability which tends to one as some parameter goes to infinity.

CHAPTER 2

LEARNING LOOSELY CONNECTED MARKOV RANDOM FIELDS

2.1 Motivation

In this chapter, we view the network as a Markov random field and want to learn the graph structure using i.i.d. samples from the joint distribution. This problem has been studied in [13, 3, 14, 4, 15], and our algorithm is motivated by and builds on the prior work in [13, 3, 14]. We aim to provide a unified framework for structure learning by considering a family of graphical models called loosely connected Markov random fields, and designing robust algorithms to avoid the kind of pitfalls illustrated in Example 1.1. In particular, we show that several previously studied models are special cases of this family of graphical models, and our algorithm achieves the same or lower computational complexity than the algorithms designed for these special cases.

2.2 Preliminaries

2.2.1 Markov Random Fields (MRFs)

Let $X = (X_1, X_2, \ldots, X_p)$ be a random vector with distribution P and $G = (V, E)$ be an undirected graph consisting of $|V| = p$ nodes with each node i associated with the i^{th} element X_i of X. Before we define an MRF, we introduce the notation X_S to denote any subset S of the random variables in X. A random vector and graph pair (X, G) is called an MRF if it satisfies one of the following three Markov properties:

1. Pairwise Markov: $X_i \perp X_j | X_{V\setminus\{i,j\}}, \forall (i,j) \notin E$, where $\perp$ denotes independence.

2. Local Markov: $X_i \perp X_{V\setminus\{i\cup N_i\}}|X_{N_i}, \forall i \in V$, where N_i is the set of neighbors of node i.

3. Global Markov: $X_A \perp X_B|X_S$ if S separates A, B on G. We say G is an *I-map* of X in this case. If G is an I-map of X and the global Markov property does not hold if any edge of G is removed, then G is called a *minimal I-map* of X.

In all three cases, G encodes a subset of the conditional independence relations of X and we say that X is Markov with respect to G. We note that the global Markov property implies the local Markov property, which in turn implies the pairwise Markov property.

When $P(x) > 0, \forall x$, the three Markov properties are equivalent, i.e., if there exists a G under which one of the Markov properties is satisfied, then the other two are also satisfied. Further, in the case when $P(x) > 0, \forall x$, there exists a unique minimal I-map of X. The unique minimal I-map $G = (V, E)$ is constructed as follows:

1. Each random variable X_i is associated with a node $i \in V$.

2. $(i, j) \notin E$ if and only if $X_i \perp X_j|X_{V\setminus\{i,j\}}$.

In this case, we consider the case $P(x) > 0, \forall x$ and are interested in learning the structure of the associated unique minimal I-map. We will also assume that, for each i, X_i takes on values in a discrete, finite set $\mathcal{X}$. We will also be interested in the special case where the MRF is an Ising model, which we describe next.

2.2.2 Ising Model

Ising models are a type of well-studied pairwise Markov random fields. In an Ising model, each random variable X_i takes values in the set $\mathcal{X} = \{-1, +1\}$ and the joint distribution is parameterized by edge coefficients J and external fields h :

$$P(x) = \frac{1}{Z} \exp\Big(\sum_{(i,j)\in E} J_{ij} x_i x_j + \sum_{i\in V} h_i x_i \Big),$$

where Z is a normalization constant to make $P(x)$ a probability distribution. If $h = 0$, we say the Ising model is zero-field. If $J_{ij} \geq 0$, we say the Ising model is ferromagnetic.

Ising models have the following useful property. Given an Ising model, the conditional probability $P(X_{V \setminus S}|x_S)$ corresponds to an Ising model on $V \setminus S$ with edge coefficients $J_{ij}, i, j \in V \setminus S$ unchanged and modified external fields $h_i + h'_i, i \in V \setminus S$, where $h'_i = \sum_{(i,j) \in E, j \in S} J_{ij} x_j$ is the additional external field on node i induced by fixing $X_S = x_S$.

2.2.3 Random Graphs

A random graph is a graph generated from a prior distribution over the set of all possible graphs with a given number of nodes. Let χ_p be a function on graphs with p nodes and let C be a constant. We say $\chi_p \geq C$ almost always for a family of random graphs indexed by p if $P(\chi_p \geq C) \to 1$ as $p \to \infty$. Similarly, we say $\chi_p \to C$ almost always for a family of random graphs if $\forall \epsilon > 0, P(|\chi_p - C| > \epsilon) \to 1$ as $p \to \infty$. This is a slight variation of the definition of almost always in [16].

The Erdős-Rényi random graph $\mathcal{G}(p, \frac{c}{p})$ is a graph on p nodes in which the probability of an edge being in the graph is $\frac{c}{p}$ and the edges are generated independently. We note that, in this random graph, the average degree of a node is c. In this thesis, when we consider random graphs, we only consider the Erdős-Rényi random graph $\mathcal{G}(p, \frac{c}{p})$.

2.2.4 High-Dimensional Structure Learning

Structure learning is the problem of inferring the structure of the graph G associated with an MRF (X, G). We will assume that $P(x) > 0$ for all x, and G will refer to the corresponding unique minimal I-map. The goal of structure learning is to design an algorithm that, given n i.i.d. samples $\{X^{(k)}\}_{k=1}^n$ from the distribution P, outputs an estimate $\hat{G}$ which equals G with high probability when n is large. We say that two graphs are equal when their node and edge sets are identical.

In the classical setting, the accuracy of estimating G is considered only when the sample size n goes to infinity while the random vector dimension p

is held fixed. This setting is restrictive for many contemporary applications, where the problem size p is much larger than the number of samples. A more suitable assumption allows both n and p to become large, with n growing at a slower rate than p. In such a case, the structure learning problem is said to be high-dimensional.

An algorithm for structure learning is evaluated both by its computational complexity and sample complexity. The computational complexity refers to the number of computations required to execute the algorithm, as a function of n and p. When G is a deterministic graph, we say the algorithm has sample complexity $f(p)$ if, for $n = O(f(p))$, there exist constants c and $\alpha > 0$, independent of p, such that $\Pr(\hat{G} = G) \geq 1 - \frac{c}{p^\alpha}$ for all P which are Markov with respect to G. When G is a random graph drawn from some prior distribution, we say the algorithm has sample complexity $f(p)$ if the above is true almost always.

In the high-dimensional setting, n can be much smaller than p. It has been shown that $\Omega(\log p)$ samples are required to learn the graph correctly with high probability, where p is the size of the graph [5]. For all the previously known algorithms for which analytical complexity bounds are available, the number of samples required to recover the graph correctly with high probability, i.e, the sample complexity, is $O(\log p)$. Not surprisingly, the sample complexity for our algorithm is also $O(\log p)$ under reasonable assumptions.

2.3 Loosely Connected MRFs

Loosely connected Markov random fields are undirected graphical models in which the number of short paths between any pair of nodes is small. Roughly speaking, a path between two nodes is short if the dependence between two node is non-negligible even if all other paths between the nodes are removed. Later, we will more precisely quantify the term "short" in terms of the correlation decay property of the MRF. For simplicity, we say that a set S separates some paths between nodes i and j if removing S disconnects these paths. In such a graphical model, if i, j are not neighbors, there is a small set of nodes S separating all the short paths between them, and conditioned on this set of variables X_S the two variables X_i and X_j are approximately independent. On the other hand, if i, j are neighbors, there

is a small set of nodes T separating all the short non-direct paths between them, i.e, the direct edge is the only short path connecting the two nodes after removing T from the graph. Conditioned on this set of variables X_T, the dependence of X_i and X_j is dominated by the dependence over the direct edge hence is bounded away from zero. The following necessary and sufficient condition for the non-existence of an edge in a graphical model shows that both the sets S and T above are essential for learning the graph, which we have not seen in prior work.

Lemma 2.1. *Consider two nodes i and j in G. Then, $(i,j) \notin E$ if and only if $\exists S, \forall T, X_i \perp X_j | X_S, X_T$.* □

Proof. Recall from the definition of the minimal I-map that $(i,j) \notin E$ if and only if $X_i \perp X_j | X_{V\setminus\{i,j\}}$. Therefore, the statement of the lemma is equivalent to

$$I(X_i; X_j | X_{V\setminus\{i,j\}}) = 0 \Leftrightarrow \min_S \max_T I(X_i; X_j | X_S, X_T) = 0,$$

where $I(X_i; X_j | X_S)$ denotes the mutual information between X_i and X_j conditioned on X_S, and we have used the fact that $X_i \perp X_j | X_S$ is equivalent to $I(X_i; X_j | X_S) = 0$. Notice that

$$\min_S \max_T I(X_i; X_j | X_S, X_T) = \min_S \max_{T' \supset S} I(X_i; X_j | X_{T'})$$

and $\max_{T' \supset S} I(X_i; X_j | X_{T'})$ is an increasing function in S. The minimization over S is achieved at $S = V \setminus \{i,j\}$, i.e.,

$$I(X_i; X_j | X_{V\setminus\{i,j\}}) = \min_S \max_T I(X_i; X_j | X_S, X_T).$$

□

Lemma 2.1 tells that, if there is not an edge between node i and j, we can find a set of nodes S such that the removal of S from the graph separates i and j. From the global Markov property, this implies that $X_i \perp X_j | X_S$. However, as Example 1.1 shows, the converse is not true. In fact, for S being the empty set or $S = \emptyset$, we have $X_1 \perp X_2 | X_S$, but $(1,2)$ is indeed an edge in the graph. Lemma 2.1 completes the statement in the converse direction, showing that we should also introduce a set T in addition to the set S to correctly identify the edge.

Motivated by this lemma, we define loosely connected MRFs as follows.

Definition 2.1. *We say a MRF is (D_1, D_2, ϵ)-loosely connected if*

1. *for any $(i,j) \notin E$, $\exists S$ with $|S| \leq D_1$, $\forall T$ with $|T| \leq D_2$,*

$$\Delta(X_i; X_j | X_S, X_T) \leq \frac{\epsilon}{4},$$

2. *for any $(i,j) \in E$, $\forall S$ with $|S| \leq D_1$, $\exists T$ with $|T| \leq D_2$,*

$$\Delta(X_i; X_j | X_S, X_T) \geq \epsilon,$$

for some conditional independence test Δ.

The conditional independence test Δ should satisfy $\Delta(X_i; X_j | X_S, X_T) = 0$ if and only if $X_i \perp X_j | X_S, X_T$. In this thesis, we use two types of conditional independence tests:

- Mutual Information Test:

$$\Delta(X_i; X_j | X_S, X_T) = I(X_i; X_j | X_S, X_T).$$

- Probability Test:

$$\Delta(X_i; X_j | X_S, X_T) = \max_{x_i, x_j, x_j', x_S, x_T} |P(x_i | x_j, x_S, x_T) - P(x_i | x_j', x_S, x_T)|.$$

In Sections 2.5 and 2.6, we will see that the probability test gives lower sample complexity for learning Ising models on bounded degree graphs, while the mutual information test gives lower sample complexity for learning Ising models on graphs with unbounded degree.

Note that the above definition restricts the size of the sets S and T to make the learning problem tractable. We show in the rest of the section that several important Ising models are examples of loosely connected MRFs. Unless otherwise stated, we assume that the edge coefficients J_{ij} are bounded, i.e., $J_{\min} \leq |J_{ij}| \leq J_{\max}$.

2.3.1 Bounded Degree Graph

We assume the graph has maximum degree d. For any $(i,j) \notin E$, the set $S = N_i$ of size at most d separates i and j, and for any set T we have $\Delta(X_i; X_j | X_S, X_T) = 0$. For any $(i,j) \in E$, the set $T = N_i \setminus j$ of size at most $d-1$ separates all the non-direct paths between i and j. Moreover, we have the following lower bound for neighbors from [13, Proposition 2].

Proposition 2.1. *When i, j are neighbors and $T = N_i \setminus j$, there is a choice of x_i, x_j, x_j', x_S, x_T such that*

$$|P(x_i|x_j, x_S, x_T) - P(x_i|x_j', x_S, x_T)| \geq \frac{\tanh(2J_{\min})}{2e^{2J_{\max}} + 2e^{-2J_{\max}}} \triangleq \epsilon.$$

□

Therefore, the Ising model on a bounded degree graph with maximum degree d is a $(d, d-1, \epsilon)$-loosely connected MRF. We note that here we do not use any correlation decay property, and we view all the paths as short.

2.3.2 Bounded Degree Graph, Correlation Decay and Large Girth

In this subsection, we still assume the graph has maximum degree d. From Section 2.3.1, we already know that the Ising model is loosely connected. But we show that when the Ising model is in the correlation decay regime and further has large girth, it is a much sparser model than the general bounded degree case.

Correlation decay is a property of MRFs which says that, for any pair of nodes i, j, the correlation of X_i and X_j decays with the distance between i, j. When a MRF has correlation decay, the correlation of X_i and X_j is mainly determined by the short paths between nodes i, j, and the contribution from the long paths is negligible. It is known that when $J_{\max}$ is small compared with d, the Ising model has correlation decay. More specifically, we have the following lemma, which is a consequence of the strong correlation decay property [17, Theorem 1].

Lemma 2.2. *Assume* $(d-1)\tanh J_{\max} < 1$. $\forall i, j \in V, d(i,j) = l$, *then for any set* S *and* $\forall x_i, x_j, x_j', x_S$,

$$|P(x_i|x_j, x_S) - P(x_i|x_j', x_S)| \leq 4J_{\max} d[(d-1)\tanh J_{\max}]^{l-1} \triangleq \beta\alpha^l,$$

where $\beta = \frac{4J_{\max} d}{(d-1)\tanh J_{\max}}$ *and* $\alpha = (d-1)\tanh J_{\max}$.

Proof. For some given x_i, x_j, x_j', x_S, w.l.o.g. assume $P(x_i|x_j, x_S) \geq P(x_i|x_j', x_S)$. Applying the [17, Theorem 1] with $\Lambda = \{j\} \cup S$, we get

$$\begin{aligned}
|P(x_i|x_j, x_S) - P(x_i|x_j', x_S)| \leq & 1 - \frac{P(x_i|x_j', x_S)}{P(x_i|x_j, x_S)} \\
\leq & 1 - e^{-4J_{\max} d[(d-1)\tanh J_{\max}]^{d(i,j)-1}} \\
\leq & 4J_{\max} d[(d-1)\tanh J_{\max}]^{d(i,j)-1}.
\end{aligned}$$

□

This lemma implies that, in the correlation decay regime $(d-1)\tanh J_{\max} < 1$, the Ising model has exponential correlation decay, i.e., the correlation between a pair of nodes decays exponentially with their distance. We say that a path of length l is short if $\beta\alpha^l$ is above some desired threshold.

The girth of a graph is defined as the length of the shortest cycle in the graph, and large girth implies that there is no short cycle in the graph. When the Ising model is in the correlation decay regime and the girth of the graph is large in terms of the correlation decay parameters, there is at most one short path between any pair of non-neighbor nodes, and no short paths other than the direct edge between any pair of neighboring nodes. Naturally, we can use S of size 1 to approximately separate any pair of non-neighbor nodes and do not need T to block the other paths for neighbor nodes as the correlations are mostly due to the direct edges. Therefore, we would expect this Ising model to be $(1, 0, \epsilon)$-loosely connected for some constant ϵ. In fact, the following theorem gives an explicit characterization of ϵ. The condition on the girth below is chosen such that there is at most one short path between any pair of nodes, so a path is called short if it is shorter than half of the girth.

Theorem 2.1. *Assume* $(d-1)\tanh J_{\max} < 1$ *and the girth* g *satisfies*

$$\beta\alpha^{\frac{g}{2}} \leq A \wedge \ln 2,$$

where $A = \frac{1}{1800}(1 - e^{-4J_{\min}})e^{-8dJ_{\max}}$. *Let* $\epsilon = 48Ae^{4dJ_{\max}}$. *Then* $\forall (i,j) \in E$,

$$\min_{\substack{S \subset V \setminus \{i \cup j\} \\ |S| \leq D_1}} \max_{x_i, x_j, x_j', x_S} |P(x_i|x_j, x_S) - P(x_i|x_j', x_S)| > \epsilon,$$

and $\forall (i,j) \notin E$,

$$\min_{\substack{S \subset V \setminus \{i \cup j\} \\ |S| \leq D_1}} \max_{x_i, x_j, x_j', x_S} |P(x_i|x_j, x_S) - P(x_i|x_j', x_S)| \leq \frac{\epsilon}{4}.$$

Proof. See Appendix A.1. □

2.3.3 Erdős-Rényi Random Graph $\mathcal{G}(p, \frac{c}{p})$ and Correlation Decay

We assume the graph G is generated from the prior $\mathcal{G}(p, \frac{c}{p})$ in which each edge is in G with probability $\frac{c}{p}$ and the average degree for each node is c. For this random graph, the maximum degree scales as $O(\frac{\ln p}{\ln \ln p})$ with high probability [16]. Thus, we cannot use the results for bounded degree graphs even though the average degree remains bounded as $p \to \infty$.

Our analysis of the class of Ising models on sparse Erdős-Rényi random graphs $\mathcal{G}(p, \frac{c}{p})$ was motivated by the results in [14] which studies the special case of the so-called ferromagnetic Ising models defined over an Erdős-Rényi random graph. It is known from [14] that, for ferromagnetic Ising models, i.e, $J_{ij} \geq 0$ for any i and j, when $J_{\max}$ is small compared with the average degree c, the random graph is in the correlation decay regime and the number of short paths between any pair of nodes is at most 2 asymptotically. We show that the same result holds for general Ising models. Our proof is related to the techniques developed in [14], but certain steps in the proof of [14] do rely on the fact that the Ising model is ferromagnetic, so the proof does not directly carry over. We point out similarities and differences as we proceed in Appendix A.3.

More specifically, letting $\gamma_p = \frac{\log p}{K \log c}$ for some $K \in (3,4)$, the following theorem shows that nodes that are at least γ_p hops from each other have negligible impact on each other. As a consequence of the following theorem, we can say that a path is short if it is at most γ_p hops.

Theorem 2.2. *Assume $\alpha = c \tanh J_{\max} < 1$. Then, the following properties are true almost always.*
(1) Let G be a graph generated from the prior $\mathcal{G}(p, \frac{c}{p})$. If i, j are not neighbors in G and S separates all the paths shorter than γ_p hops between i, j, then $\forall x_i, x_j, x'_j, x_S$,

$$|P(x_i|x_j, x_S) - P(x_i|x'_j, x_S)| \leq |B(i, \gamma_p)|(\tanh J_{\max})^{\gamma_p} = o(p^{-\kappa}),$$

for all Ising models P on G, where $\kappa = \frac{\log \frac{1}{\alpha}}{4 \log c}$ and $B(i, \gamma_p)$ is the set of all nodes which are at most γ_p hops away from i.
(2) There are at most two paths shorter than γ_p between any pair of nodes.

Proof. See Appendix A.3. □

The above result suggests that for Ising models on the random graph there are at most two short paths between non-neighbor nodes and one short non-direct path between neighboring nodes, i.e., it is a $(2, 1, \epsilon)$-loosely connected MRF. Further the next two theorems prove that such a constant ϵ exists. The proofs are in Appendix A.3.

Theorem 2.3. *For any $(i, j) \notin E$, let S be a set separating the paths shorter than γ_p between i, j and assume $|S| \leq 3$, then almost always*

$$I(X_i; X_j|X_S) = o(p^{-2\kappa}).$$

□

Theorem 2.4. *For any $(i, j) \in E$, let T be a set separating the non-direct paths shorter than γ_p between i, j and assume $|T| \leq 3$, then almost always*

$$I(X_i; X_j|X_T) = \Omega(1).$$

□

2.4 The Algorithm *CondST*

Learning the structure of a graph is equivalent to learning if there exists an edge between every pair of nodes in the graph. Therefore, we would like

to develop a test to determine if there exists an edge between two nodes or not. From Definition 2.1, it should be clear that learning a loosely connected MRF is straightforward. For non-neighbor nodes, we search for the set S that separates all the short paths between them, while for neighboring nodes, we search for the set T that separates all the non-direct short paths between them.

We first introduce a few notations. Given n i.i.d. samples $\{X^{(k)}\}_{k=1}^{n}$ from the distribution the empirical distribution $\hat{P}$ is defined as follows: for any set A,

$$\hat{P}(x_A) = \frac{1}{n}\sum_{i=1}^{n} \mathbb{I}_{\left\{X_A^{(i)}=x_A\right\}}.$$

Let $\hat{\Delta}$ be the empirical conditional independence test which is the same as Δ but computed using $\hat{P}$. Our algorithm is as follows.

Algorithm 1 $CondST(D_1, D_2, \epsilon)$

for $i, j \in V$ **do**
  if $\exists S$ with $|S| \leq D_1, \forall T$ with $|T| \leq D_2$, $\hat{\Delta}(X_i; X_j | X_S, X_T) \leq \frac{\epsilon}{2}$
  then
    $(i, j) \notin E$
  else
    $(i, j) \in E$
  end if
end for

For clarity, when we specifically use the mutual information test (or the probability test), we denote the corresponding algorithm by $CondST_I$ (or $CondST_P$).

The algorithm $CondST(D_1, D_2, \epsilon)$ runs for each pair (i, j). It compares the empirical min-max conditional independence

$$\min_{|S|\leq D_1} \max_{|T|\leq D_2} \hat{\Delta}(X_i; X_j | X_S, X_T)$$

with the threshold $\frac{\epsilon}{2}$ to determine if there is an edge between node i and j. The maximization step is designed to detect the edges while the minimization step is designed to detect non-edges. The minimization step is used in several previous works such as [14, 4]. The maximization step has been added to explicitly break the short cycles that can cause problems in edge detection.

Intuitively, if the direct edge is the only edge between a pair of neighboring nodes, the dependence over the edge can be detected by the independence test. When there are other short paths between a pair of neighboring nodes, directly performing the independence test might fail. In Example 1.1, X_1 and X_3 are marginally independent as the dependence over edge $(1,3)$ is canceled by the other path $(1,2,3)$. Our algorithm handles this difficulty by first finding a set T of nodes that separates all the short, non-direct paths between them, i.e., after removing the set T from the graph, the direct edge is the only short path connecting to two nodes. Then the dependence over the edge can again be detected by the conditional independence test where the conditioned set is T. In the same example, if we break the short path $(1,2,3)$ by conditioning on X_2, X_1 and X_3 become dependent, so our algorithm is able to detect the edges correctly. When the empirical conditional independence test $\hat{\Delta}$ is close to the exact test Δ, we immediately get the following result.

Fact 2.1. *For a (D_1, D_2, ϵ)-loosely connected MRF, if*

$$|\hat{\Delta}(X_i; X_j|X_A) - \Delta(X_i; X_j|X_A)| < \frac{\epsilon}{4}$$

for any node i, j and set A with $|A| \leq D_1+D_2$, then $CondST(D_1, D_2, \epsilon)$ recovers the graph correctly. The running time for the algorithm is $O(np^{D_1+D_2+2})$.

Proof. The correctness is immediate. We note that, for each pair of i, j in V, we search S, T in V. So the possible combinations of (i, j, S, T) is $O(p^{D_1+D_2+2})$ and we get the running time result. □

When the MRF has correlation decay, it is possible to reduce the computational complexity by restricting the search space for the set S and T to a smaller candidate neighbor set. In fact, for each node i, the nodes which are a certain distance away from i have small correlation with X_i. As suggested in [13], we can first perform a pairwise correlation test to eliminate these nodes from the candidate neighbor set of node i. To make sure the true neighbors are all included in the candidate set, the MRF needs to satisfy an additional pairwise non-degeneracy condition. Our second algorithm is as follows.

The following result provides conditions under which the second algorithm correctly learns the MRF.

Algorithm 2 *CondST_Pre*$(D_1, D_2, \epsilon, \epsilon')$

for $i \in V$ **do**
  $L_i = \{j \in V \setminus i, \max_{x_i, x_j, x_j'} |\hat{P}(x_i|x_j) - \hat{P}(x_i|x_j')| > \frac{\epsilon'}{2}\}$.
  for $j \in L_i$ **do**
    if $\exists S \subset L_i$ with $|S| \leq D_1, \forall T \subset L_i$ with $|T| \leq D_2, \hat{\Delta}(X_i; X_j|X_S, X_T) \leq \frac{\epsilon}{2}$ **then**
      $j \notin N_i$
    else
      $j \in N_i$
    end if
  end for
end for

Fact 2.2. *For a* (D_1, D_2, ϵ)*-loosely connected MRF with*

$$\max_{x_i, x_j, x_j'} |P(x_i|x_j) - P(x_i|x_j')| > \epsilon' \tag{2.1}$$

for any $(i, j) \in E$*, if*

$$|\hat{P}(x_i|x_j) - P(x_i|x_j)| < \frac{\epsilon'}{8}$$

for any node i, j *and* x_i, x_j*, and*

$$|\hat{\Delta}(X_i; X_j|X_A) - \Delta(X_i; X_j|X_A)| < \frac{\epsilon}{4}$$

for any node i, j *and set* A *with* $|A| \leq D_1 + D_2$*, then* *CondST_Pre*$(D_1, D_2, \epsilon, \epsilon')$ *recovers the graph correctly. Let* $L = \max_i |L_i|$*. The running time for the algorithm is* $O(np^2 + npL^{D_1+D_2+1})$*.*

Proof. By the pairwise non-degeneracy condition (2.1), the neighbors of node i are all included in the candidate neighbor set L_i. We note that this pre-processing step excludes the nodes whose correlation with node i is below $\frac{\epsilon'}{4}$. Then in the inner loop, the correctness of the algorithm is immediate. The running time of the correlation test is $O(np^2)$. We note that, for each i in V, we loop over j in L_i and search S and T in L_i. So the possible combinations of (i, j, S, T) is $O(pL^{D_1+D_2+1})$. Combining the two steps, we get the running time of the algorithm. □

Note that the additional non-degeneracy condition (2.1) required for the

second algorithm to execute correctly is not satisfied for all graphs (recall Example 1.1).

In the following lemma we show a set of concentration results for the empirical quantities in the above algorithm for general discrete MRFs, which will be used to obtain the sample complexity results in Section 2.5 and Section 2.6.

Lemma 2.3. *Fix $\gamma > 0$. Let $L = \max_i |L_i|$. For $\forall \alpha > 0$,*

1. *Assume $\gamma \leq \frac{1}{4}$. If*

$$n > \frac{2\big[(2+\alpha)\log p + 2\log|\mathcal{X}|\big]}{\gamma^2},$$

then $\forall i, j \in V, \forall x_i, x_j$,

$$|\hat{P}(x_i|x_j) - P(x_i|x_j)| < 4\gamma$$

with probability $1 - \frac{c_1}{p^\alpha}$ for some constant c_1.

2. *Assume $\forall S \subset V, |S| \leq D_1 + D_2 + 1, P(x_S) > \delta$ for some constant δ, and $\gamma \leq \frac{\delta}{2}$. If*

$$n > \frac{2\big[(1+\alpha)\log p + (D_1+D_2+1)\log L + (D_1+D_2+2)\log|\mathcal{X}|\big]}{\gamma^2},$$

then $\forall i \in V, \forall j \in L_i, \forall S \subset L_i, |S| \leq D_1 + D_2, \forall x_i, x_j, x_S$,

$$|\hat{P}(x_i|x_j, x_S) - P(x_i|x_j, x_S)| < \frac{2\gamma}{\delta}$$

with probability $1 - \frac{c_2}{p^\alpha}$ for some constant c_2.

3. *Assume $\gamma \leq \frac{1}{2|\mathcal{X}|^{D_1+D_2+2}} < 1$. If*

$$n > \frac{2\big[(1+\alpha)\log p + (D_1+D_2+1)\log L + (D_1+D_2+2)\log|\mathcal{X}|\big]}{\gamma^2},$$

then $\forall i, j \in V, |S| \leq D_1 + D_2, \forall x_i, x_j, x_S$,

$$|\hat{I}(X_i; X_j|X_S) - I(X_i; X_j|X_S)| < 8|\mathcal{X}|^{D_1+D_2+2}\sqrt{\gamma}$$

with probability $1 - \frac{c_3}{p^\alpha}$ for some constant c_3,

Proof. See Appendix A.4. □

This lemma could be used as a guideline on how to choose between the two conditional independence tests for our algorithm to get lower sample complexity. The key difference is the dependence on the constant δ, which is a lower bound on the probability of any x_S with the set size $|S| \leq D_1+D_2+1$. The probability test requires a constant $\delta > 0$ to achieve sample complexity $n = O(\log p)$, while the mutual information test does not depend on δ and also achieves sample complexity $n = O(\log p)$. We note that, while both tests have $O(\log p)$ sample complexity, the constants hidden in the order notation may be different for the two tests. For Ising models on bounded degree graphs, we show in Section 2.5 that a constant $\delta > 0$ exists, and the probability test gives a lower sample complexity. On the other hand, for Ising models on the Erdős-Rényi random graph $\mathcal{G}(p, \frac{c}{p})$, we could not get a constant $\delta > 0$ as the maximum degree of the graph is unbounded, and the mutual information test gives a lower sample complexity.

For loosely connected MRF, the required sizes D_1, D_2 of the conditioned sets S, T are typically small, therefore the algorithm has low computational and sample complexity.

2.5 Computational Complexity for General Ising Models

In this section, we apply our algorithm to the Ising models in Section 2.3. We evaluate both the number of samples required to recover the graph with high probability and the running time of our algorithm. The following results are simple combinations of the results in the previous two sections. Unless otherwise stated, we assume that the edge coefficients J_{ij} are bounded, i.e., $J_{\min} \leq |J_{ij}| \leq J_{\max}$. Throughout this section, we use the notation $x \wedge y$ to denote the minimum of x and y.

2.5.1 Bounded Degree Graph

We assume the graph has maximum degree d. First we have the following lower bound on the probability of any finite size set of variables.

Lemma 2.4. $\forall S \subset V, \forall x_S,\ P(x_S) \geq 2^{-|S|} \exp(-2(|S|+d)|S|J_{\max})$.

Proof. See Appendix A.1. □

Our algorithm with the probability test for the bounded degree graph case reproduces the algorithm in [13]. However, our algorithm is more flexible and achieves lower computational complexity for MRFs that are loosely connected but have a large maximum degree as we will see later. For completeness, we state the following result without a proof since it is nearly identical to the result in [13], except for some constants.

Corollary 2.1. *Let ϵ be defined as in Proposition 2.1. Define*

$$\delta = 2^{-2d} \exp(-12d^2 J_{\max}).$$

Let $\gamma = \frac{\epsilon\delta}{16} \wedge \frac{\delta}{2} < 1$. If $n > \frac{2\left[(2d+1+\alpha)\log p + (2d+1)\log 2\right]}{\gamma^2}$, the algorithm $CondST_P(d, d-1, \epsilon_2)$ recovers G with probability $1 - \frac{c}{p^\alpha}$ for some constant c. The running time of the algorithm is $O(np^{2d+1})$. □

2.5.2 Bounded Degree Graph, Correlation Decay and Large Girth

We assume the graph has maximum degree d. We also assume that the Ising model is in the correlation decay regime, i.e., $(d-1)\tanh J_{\max} < 1$, and the graph has large girth. The same setting has been considered in [3]. Combining Theorem 2.1, Fact 2.1 and Lemma 2.3, we can show that the algorithm $CondST_P(1, 0, \epsilon)$ recovers the graph correctly with high probability for some constant ϵ, and the running time is $O(np^3)$ for $n = O(\log p)$.

We can get even lower computational complexity using our second algorithm. The key observation is that, as there is no short path other than the direct edge between neighboring nodes, the correlation over the edge dominates the total correlation hence the pairwise non-degeneracy condition is satisfied. We note that the length of the second shortest path between neighboring nodes is no less than $g-1$.

Lemma 2.5. *Assume that $(d-1)\tanh J_{\max} < 1$, and the girth g satisfies*

$$\beta\alpha^{g-1} \leq A \wedge \ln 2,$$

where $A = \frac{1}{1800}(1 - e^{-4J_{\min}})$. *Let* $\epsilon' = 48A$. $\forall (i,j) \in E$, *we have*

$$\max_{x_i, x_j, x_j'} |P(x_i|x_j) - P(x_i|x_j')| > \epsilon'.$$

Proof. See Appendix A.1. □

Using this lemma, we can apply our second algorithm to learn the graph. Using Lemma 2.2, if node j is of distance $l_{\epsilon'} = \frac{\ln \frac{4\beta}{\epsilon'}}{\ln \frac{1}{\alpha}}$ hops from node i, we have

$$\max_{x_i, x_j, x_j'} |P(x_i|x_j) - P(x_i|x_j')| < \beta \alpha^{l_{\epsilon'}} \leq \frac{\epsilon'}{4}.$$

Therefore, in the correlation test, L_i only includes nodes within distance $l_{\epsilon'}$ from i and the size $|L_i| \leq d^{l_{\epsilon'}}$ since the maximum degree is d; i.e., $L = \max_i |L_i| \leq d^{l_{\epsilon'}}$, which is a constant independent of p. Combining Lemma 2.5 with Theorem 2.1, Fact 2.2 and Lemma 2.3, we get the following result.

Corollary 2.2. *Assume* $(d-1)\tanh J_{\max} < 1$. *Assume* g, ϵ *and* ϵ' *satisfy Theorem 2.1 and Lemma 2.5. Let* δ *be defined as in Theorem 2.1. Let* $\gamma = \frac{\epsilon'}{32} \wedge \frac{\epsilon\delta}{16} \wedge \frac{\delta}{2}$. *If*

$$n > \frac{2\big[(2+\alpha)\log p + 2l_{\epsilon'}\log d + 3\log 2\big]}{\gamma^2},$$

the algorithm $CondST\_Pre_P(1, 0, \epsilon, \epsilon')$ *recovers* G *with probability* $1 - \frac{c}{p^\alpha}$ *for some constant* c. *The running time of the algorithm is* $O(np^2)$. □

By performing a simple correlation test, we can reduce the search space for neighbors from all the nodes to a constant size candidate neighbor set, then our algorithm and the algorithms in [13, 3, 18] all have computational complexity $O(np^2)$, which is lower than what we would get by only applying the greedy algorithm [3]. The results in [18] improve over [3] by proposing two new greedy algorithms that are correct for learning small girth graphs. However, the algorithm in [18] requires a constant size candidate neighbor set as input, which might not be easy to obtain in general. In fact, for MRFs with bad short cycles as in Example 1.1, learning a candidate neighbor set can be as difficult as directly learning the neighbor set.

2.5.3 Erdős-Rényi Random Graph $\mathcal{G}(p, \frac{c}{p})$ and Correlation Decay

We assume the graph G is generated from the prior $\mathcal{G}(p, \frac{c}{p})$ in which each edge is in G with probability $\frac{c}{p}$ and the average degree for each node is c. Because the random graph has unbounded maximum degree, we cannot lower bound for the probability of a finite size set of random variables by a constant, for all p. To get good sample complexity, we use the mutual information test in our algorithm. Combining Theorem 2.3, Theorem 2.4, Fact 2.1 and Lemma 2.3, we get the following result.

Corollary 2.3. *Assume* $c \tanh J_{\max} < 1$. *There exists a constant* $\epsilon > 0$ *such that, for* $\gamma = \left(\frac{\epsilon}{32^2}\right)^2 \wedge \frac{1}{64} < 1$, *if* $n > \frac{2\left[(5+\alpha)\log p + 5\log 2\right]}{\gamma^2}$, *the algorithm* $CondST_I(2, 1, \epsilon)$ *recovers the graph* G *almost always. The running time of the algorithm is* $O(np^5)$. □

The results in [4] extend the results in [14] to general Ising models and more general sparse graphs (beyond the Erdős-Rényi model). We note that the tractable graph families in [4] is similar to our notion of loosely-connected MRFs. For general Ising models over sparse Erdős-Rényi random graphs, our algorithm has computational complexity $O(np^5)$ while the algorithm in [4] has computational complexity $O(np^4)$. The difference comes from the fact that our algorithm has an additional maximization step to break bad short cycles as in Example 1.1. Without this maximization step, the algorithm in [4] fails for this example. The performance analysis in [4] explicitly excludes such difficult cases by noting that these "unfaithful" parameter values have Lebesgue measure zero [4, Section B.3.2]. However, when the Ising model parameters lie close to this Lebesgue measure zero set, the learning problem is still ill posed for the algorithm in [4], i.e., the sample complexity required to recover the graph correctly with high probability depends on how close the parameters are to this set, which is not the case for our algorithm. In fact, the same problem with the argument that the unfaithful set is of Lebesgue measure zero has been observed for causal inference in the Gaussian case [19]. It has been shown in [19] that a stronger notion of faithfulness is required to get uniform sample complexity results, and the set that is not strongly faithful has non-zero Lebesgue measure and can be be surprisingly large.

2.5.4 Sample Complexity

In this subsection, we briefly summarize the number of samples required by our algorithm. According to the results in this section and Section 2.6, $C \log p$ samples are sufficient in general, where the constant C depends on the parameters of the model. When the Ising model is on a bounded degree graph with maximum degree d, the constant C is of order $\exp(-O(d+d^2 J_{\max}))$. In particular, if the Ising model is in the correlation decay regime, then $dJ_{\max} = O(1)$ and the constant C is of order $\exp(-O(d))$. When the Ising model is on an Erdős-Rényi random graph $\mathcal{G}(p, \frac{c}{p})$ and is in the correlation decay regime, then the constant C is lower bounded by some absolute constant independent of the model parameters.

2.6 Computational Complexity for Ferromagnetic Ising Models

Ferromagnetic Ising models are Ising models in which all the edge coefficients J_{ij} are non-negative. We say (i, j) is an edge if $J_{ij} > 0$. One important property of ferromagnetic Ising models is association, which characterizes the positive dependence among the nodes.

Definition 2.2. *[20] We say a collection of random variables $X = (X_1, X_2, \ldots, X_n)$ is associated, or the random vector X is associated, if*

$$\mathrm{Cov}(f(X), g(X)) \geq 0$$

for all nondecreasing functions f and g for which $\mathbb{E}[f(X)], \mathbb{E}[g(X)]$ and $\mathbb{E}[f(X)g(X)]$ exist. □

Proposition 2.2. *[21] The random vector X of a ferromagnetic Ising model (possibly with external fields) is associated.* □

A useful consequence of the Ising model being associated is as follows.

Corollary 2.4. *Assume X is a zero field ferromagnetic Ising model. For any i, j, $P(X_i = 1, X_j = 1) \geq \frac{1}{4} \geq P(X_i = 1, X_j = -1)$.*

Proof. See Appendix A.2. □

Informally speaking, the edge coefficient $J_{ij} > 0$ means that i and j are positively dependent over the edge. For any path between i, j, as all the edge coefficients are positive, the dependence over the path is also positive. Therefore, the non-direct paths between a pair of neighboring nodes i, j make X_i and X_j, which are positively dependent over the edge (i, j), even more positively dependent. This observation has two important implications for our algorithm.

1. We do not need to break the short cycles with a set T in order to detect the edges, so the maximization in the algorithm can be removed.

2. The pairwise non-degeneracy is always satisfied for some constant ϵ', so we can apply the correlation test to reduce the computational complexity.

2.6.1 Bounded Degree Graph

We assume the graph has maximum degree d. We have the following non-degeneracy result for ferromagnetic Ising models.

Lemma 2.6. *$\forall (i,j) \in E, S \subset V \setminus \{i,j\}$ and $\forall x_S$,*

$$\max_{x_i, x_j, x_j'} |P(x_i|x_j, x_S) - P(x_i|x_j', x_S)| \geq \frac{1}{16}(1 - e^{-4J_{\min}})e^{-4|N_S|J_{\max}}.$$

Proof. See Appendix A.2. □

The following theorem justifies the remarks after Corollary 2.4 and shows that the algorithm with the preprocessing step $CondST\_Pre(d, 0, \epsilon, \epsilon')$ can be used to learn the graph, where ϵ, ϵ' are obtained from Lemma 2.6. Recall that L_i is the candidate neighbor set of node i after the preprocessing step and $L = \max_i |L_i|$.

Theorem 2.5. *Let*

$$\epsilon = \frac{1}{16}(1 - e^{-4J_{\min}})e^{-4d^2 J_{\max}}, \quad \epsilon' = \frac{1}{16}(1 - e^{-4J_{\min}}),$$

and δ be defined as in Theorem 2.1. Let $\gamma = \frac{\epsilon'}{32} \wedge \frac{\epsilon\delta}{16} \wedge \frac{\delta}{2}$. If

$$n > \frac{2\big[(1+\alpha)\log p + (d+1)\log L + (d+2)\log 2\big]}{\gamma^2},$$

the algorithm $CondST\_Pre_P(d, 0, \epsilon, \epsilon')$ *recovers* G *with probability* $1 - \frac{c}{p^\alpha}$ *for some constant* c*. The running time of the algorithm is* $O(np^2 + npL^{d+1})$*. If we further assume that* $(d-1)\tanh J_{\max} < 1$*, then the running time of the algorithm is* $O(np^2)$*.*

Proof. We choose $|S| \leq d$ and $T = \emptyset$ in our algorithm, and we have $|N_S| \leq d^2$ as the maximum degree is d. By Lemma 2.6, we have

$$\max_{x_i, x_j, x_j', x_S} |P(x_i|x_j, x_S) - P(x_i|x_j', x_S)| \geq \epsilon$$

for any $|S| \leq d$. Therefore, the Ising model is a $(d, 0, \epsilon)$-loosely connected MRF. Note that Lemma 2.6 is applicable to any set S (not necessarily the set S in the conditional independence test). Applying Lemma 2.6 again with $S = \emptyset$, we get the pairwise non-degeneracy condition

$$\max_{x_i, x_j, x_j'} |P(x_i|x_j) - P(x_i|x_j')| \geq \epsilon'.$$

Combining Fact 2.2 and Lemma 2.3, we get the correctness of the algorithm. The running time is $O(np^2 + npL^{d+1})$, which is at most $O(np^{d+2})$.

When $(d-1)\tanh J_{\max} < 1$, as the Ising model is in the correlation decay regime, $L = \max_i |L_i| \leq d^{l_{\epsilon'}}$ is a constant independent of p as argued for Theorem 2.2. Therefore, the running time is only $O(np^2)$ in this case. □

2.6.2 Erdős-Rényi Random Graph $\mathcal{G}(p, \frac{c}{p})$ and Correlation Decay

When the Ising model is ferromagnetic, the result for the random graph is similar to that of a deterministic graph. For each graph sampled from the prior distribution, the dependence over the edges is positive. If i, j are neighbors in the graph, having additional paths between them makes them more positively dependent, so we do not need to block those paths with a set T to detect the edge and set $D_2 = 0$. In fact, we can prove a stronger result for neighbor nodes than the general case. The following result also appears in [14], but we are unable to verify the correctness of all the steps there and so we present the result here for completeness.

Theorem 2.6. *$\forall i \in V, \forall j \in N_i$, let S be any set with $|S| \leq 2$, then almost always*

$$I(X_i; X_j | X_S) = \Omega(1).$$

Proof. See Appendix A.3. □

Moreover, the pairwise non-degeneracy condition in Theorem 2.5 also holds here. We can thus use algorithm $CondST\_Pre(2, 0, \epsilon, \epsilon')$ to learn the graph. Without the pre-processing step, our algorithm is the same as in [14], which has computational complexity $O(np^4)$. We show in the following theorem that by using the pre-processing step our algorithm reduces the computational complexity to $O(np^2)$.

Theorem 2.7. *Assume $c \tanh J_{\max} < 1$ and the Ising model is ferromagnetic. Let ϵ' be defined as in Theorem 2.5. There exists a constant $\epsilon > 0$ such that, for $\gamma = \frac{\epsilon_1}{32} \wedge \left(\frac{\epsilon_2}{512}\right)^2 \wedge \frac{1}{32} < 1$, if $n > \frac{2\left[(2+\alpha)\log p + 3\log L + 5\log 2\right]}{\gamma^2}$, the algorithm $CondST\_Pre_I(2, 0, \epsilon, \epsilon')$ recovers the graph G almost always. The running time of the algorithm is $O(np^2)$.*

Proof. Combining Theorem 2.3, Theorem 2.4, Fact 2.2, Lemma 2.3 and Lemma 2.6, we get the correctness of the algorithm.

From Theorem 2.2 we know that if j is more than γ_p hops away from i, the correlation between them decays as $o(p^{-\kappa})$. For the constant threshold $\frac{\epsilon'}{2}$, these far-away nodes are excluded from the candidate neighbor set L_i when p is large. It is shown in the proof of [22, Lemma 2.1] that for $\mathcal{G}(p, \frac{c}{p})$, the number of nodes in the γ_p-ball around i is not large with high probability. More specifically, $\forall i \in V, |B(i, \gamma_p)| = O(c^{\gamma_p} \log p)$ almost always, where $B(i, \gamma_p)$ is the set of all nodes which are at most γ_p hops away from i. Therefore we get

$$L = \max_i |L_i| \leq |B(i, \gamma_p)| = O(c^{\gamma_p} \log p) = O(p^{\frac{1}{K}} \log p) = O(p^{\frac{1}{3}}).$$

So the total running time of algorithm $CondST_I(2, 0, \epsilon, \epsilon')$ is $O(np^2 + npL^3) = O(np^2)$. □

2.7 Related Work

Another way to learn the structures of MRFs is by solving l_1-regularized convex optimizations under a set of incoherence conditions [15]. It is shown in [23] that, for some Ising models on a bounded degree graph, the incoherence conditions hold when the Ising model is in the correlation decay regime. But the incoherent conditions do not have a clear interpretation as conditions for the graph parameters in general and are NP-hard to verify for a given Ising model [23]. Using results from standard convex optimization theory [24], it is possible to design a polynomial complexity algorithm to approximately solve the l_1-regularized optimization problem. However, the actual complexity will depend on the details of the particular algorithm used, therefore, it is not clear how to compare the computational complexity of our algorithm with the one in [15].

We note that the recent development of directed information graphs [25] is closely related to the theory of MRFs. Learning a directed information graph, i.e., finding the causal parents of each random process, is essentially the same as finding the neighbors of each random variable in learning a MRF. Therefore, our algorithm for learning the MRFs can potentially be used to learn the directed information graphs as well.

2.8 Experimental Results

In this section, we present experimental results to show the importance of the choice of a non-zero D_2 in correctly estimating the edges and non-edges of the underlying graph of a MRF. We evaluate our algorithm $CondST_I(D_1, D_2, \epsilon)$, which uses the mutual information test and does not have the preprocessing step, for general Ising models on grids and random graphs as illustrated in Figure 2.1. In a single run of the algorithm, we first generate the graph $G = (V, E)$: for grids, the graph is fixed, while for random graphs, the graph is generated randomly each time. After generating the graph, we generate the edge coefficients uniformly from $[-J_{\max}, -J_{\min}] \cup [J_{\min}, J_{\max}]$, where $J_{\min} = 0.4$ and $J_{\max} = 0.6$. We then generate samples from the Ising model by Gibbs sampling. The sample size ranges from 400 to 1000. The

algorithm computes, for each pair of nodes i and j,

$$\hat{I}_{ij} = \min_{|S|\leq D_1} \max_{|T|\leq D_2} \hat{I}(X_i; X_j | X_S, X_T)$$

using the samples. For a particular threshold ϵ, the algorithm outputs (i, j) as an edge if $\hat{I}_{ij} > \epsilon$ and gets an estimated graph $\hat{G} = (V, \hat{E})$. We select ϵ optimally for each run of the simulation, using the knowledge of the graph, such that the number of errors in $\hat{E}$, including both errors in edges and non-edges, is minimized. The performance of the algorithm in each case is evaluated by the probability of success, which is the percentage of the correctly estimated edges, and each point in the plots is an average over 50 runs. We then compare the performance of the algorithm under different choices of D_1 and D_2.

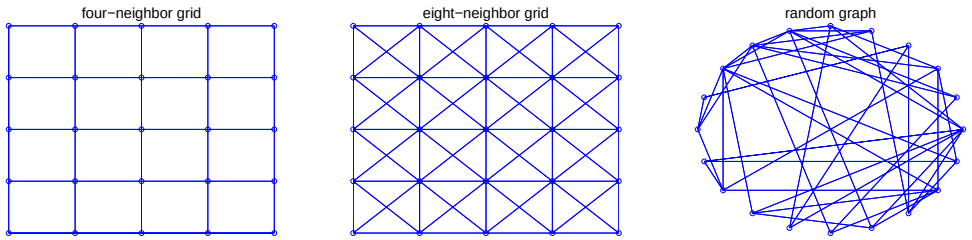

Figure 2.1: Illustrations of four-neighbor grid, eight-neighbor grid and the random graph.

The experimental results for the algorithm with $D_1 = 0, \ldots, 3$ and $D_2 = 0, 1$ applied to eight-neighbor grids on 25 and 36 nodes are shown in Figure 2.2. We omit the results for four-neighbor grids as the performances of the algorithm with $D_2 = 0$ and $D_2 > 0$ are very close. In fact, four-neighbor grids do not have many short cycles and even the shortest non-direct paths are weak for the relatively small $J_{\max}$ we choose, therefore there is no benefit using a set T to separate the non-direct paths for edge detection. However, for eight-neighbor grids which are denser and have shorter cycles, the probability of success of the algorithm significantly improves by setting $D_2 = 1$, as seen from Figure 2.2. It is also interesting to note that increasing from $D_1 = 2$ to $D_1 = 3$ does not improve the performance, which implies that a set S of size 2 is sufficient to approximately separate the non-neighbor nodes in our eight-neighbor grids.

The experimental results for the algorithm with $D_1 = 0, \ldots, 3$ and $D_2 =$

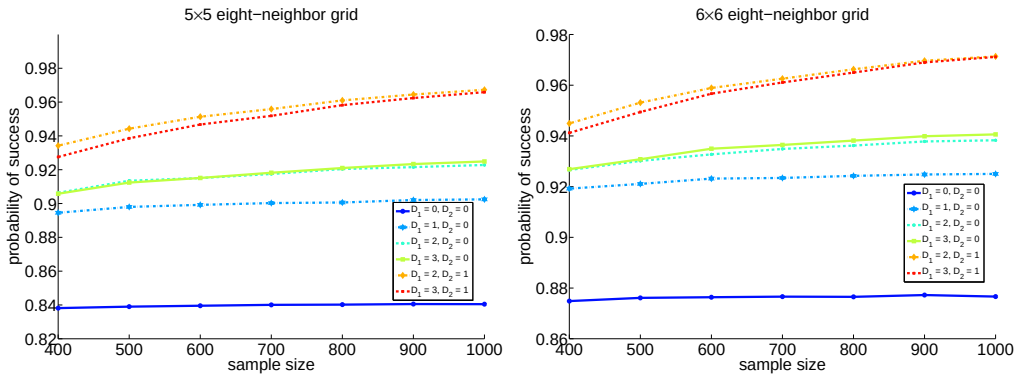

Figure 2.2: Plots of the probability of success versus the sample size for 5×5 and 6×6 eight-neighbor grids with $D_1 = 0, \ldots, 3$ and $D_2 = 0, 1$.

$0, 1$ applied to random graphs on 20 and 30 nodes are shown in Figure 2.3. For a random graph on n nodes with average degree d, each edge is included in the graph with probability $\frac{d}{n-1}$ and is independent of all other edges. In the experiment, we choose average degree 5 for the graphs on 20 nodes and 7 for the graphs on 30 nodes. From Figure 2.3, the probability of success of the algorithm improves a lot when we increase D_2 from 0 to 1, which is very similar to the result of the eight-neighbor grids. We also note that, unlike the previous case, the algorithm with $D_1 = 3$ does have a better performance than with $D_1 = 2$ as there might be more short paths between a pair of nodes in random graphs.

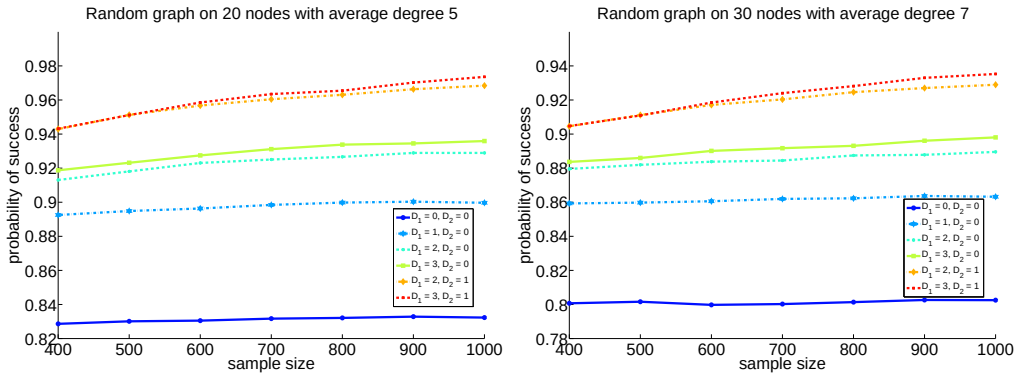

Figure 2.3: Plots of the probability of success versus the sample size for random graphs with $D_1 = 0, \ldots, 3$ and $D_2 = 0, 1$.

In a true experiment where only the data is available and no prior knowledge of the MRF is available, the choice of ϵ itself may affect the performance

of the algorithm. At this time, we do not have any theoretical results to inform the choice of ϵ. We briefly present a heuristic, which seems reasonable. However, extensive testing of the heuristic is required before we can confidently state that the heuristic is reasonable, which is beyond the scope of this thesis. Our proposed heuristic is as follows.

For a given D_1 and D_2, we compute $\hat{I}_{ij}$ for each pair of nodes i and j. If the choice of D_1 and D_2 is good, $\hat{I}_{ij}$ is expected to be close to 0 for non-edges and away from 0 for edges. Therefore, we can view the problem of choosing the threshold ϵ as a two-class hypothesis testing, where the non-edge class concentrates near 0 while the edge class is more spread out. If we view $\hat{I}$, the collection of $\hat{I}_{ij}$ for all i and j, as samples generated from the distribution of some random variable Z, then the hypothesis testing problem can viewed as one of finding the right ϵ such that the density of Z has a big spike below ϵ. One heuristic is to first estimate a smoothed density function from $\hat{I}$ via kernel density estimation [26] and then set ϵ to be the right boundary of the big spike near 0.

In order to choose proper D_1 and D_2 for the algorithm, we can start with $(D_1, D_2) = (0, 0)$. At each step, we run the algorithm with two pairs of values $(D_1 + 1, D_2)$ and $(D_1, D_2 + 1)$ separately, and choose the pair that has a more significant change on the density estimated from $\hat{I}$ as the new value for (D_1, D_2). We continue this process and stop increasing D_1 or D_2 if at some step there is no significant change for either pair of values.

Justifying this heuristic either through extensive experimentation or theoretical analysis is a topic for future research.

CHAPTER 3

CLUSTERING IN RECOMMENDER SYSTEMS

3.1 Introduction

In this chapter, we consider recommender systems where users and items form clusters. The goal is to cluster both users and items using a small fraction of noisy binary ratings that users give to items. We first try to understand the fundamental limit of the number of observations required. Then we propose three clustering algorithms and analyze their performances. In particular, our results show an interesting trade-off between the amount of data available and the running time of the algorithms.

3.2 Model and Main Results

3.2.1 Model

Our model is described in the context of recommender systems, but it is applicable to other systems with binary data matrices having row and column cluster structure. Consider a recommender system with n users and n items. Let R be the rating matrix of size $n \times n$ where R_{ij} is the rating user i gives to item j. Assume both users and items form r clusters of size $K = n/r$. Users in the same cluster give the same rating to items in the same cluster. The set of ratings corresponding to a user cluster and a item cluster is called a block. Let B be the *block rating matrix* of size $r \times r$ where B_{kl} is the *block rating* user cluster k gives to item cluster l. Then the rating $R_{ij} = B_{kl}$ if user i is in user cluster k and item j is in item cluster l. Further assume that entries of B are independent random variables which are $+1$ or -1 with equal probability. Thus, we can imagine the rating matrix as a block-constant matrix with all

Table 3.1: Main results: Comparison of a lower bound and four algorithms.

	regime in (K,ϵ)	regime in (K,m)	running time	remark
lower bound	$nK^2(1-\epsilon)^2 = O(1)$	$m = O(\frac{n^{1.5}}{K})$		
combinatorial method	$nK(1-\epsilon)^2 = \Omega(\log n)$	$m = \Omega(\frac{n^{1.5}\sqrt{\log n}}{\sqrt{K}})$	exponential	assuming noiseless
convex method	$K(1-\epsilon) = \Omega(\log n)$	$m = \Omega(\frac{n^2 \log n}{K})$	polynomial	assuming Conjecture 3.1
spectral method	$K^2(1-\epsilon) = \Omega(n \log^2 n)$	$m = \Omega(\frac{n^3 \log^2 n}{K^2})$	$O(n^3)$	
nearest-neighbor clustering	$n(1-\epsilon)^2 = \Omega(\log n)$	$m = \Omega(n^{1.5}\sqrt{\log n})$	$O(mr)$	

the entries in each block being either $+1$ or -1. Observe that if r is a fixed constant, then users from two different clusters have the same ratings for all items with some positive probability, in which case it is impossible to differentiate between these two clusters. To avoid such situations, assume r is at least $\Omega(\log n)$.

Suppose each entry of R goes through an independent binary symmetric channel with flipping probability $p < 1/2$, representing noisy user behavior, and an independent erasure channel with erasure probability ϵ, modeling the fact that some entries are not observed. The expected number of observed ratings is $m = n^2(1-\epsilon)$. We assume that p is a constant throughout the thesis and ϵ could converge to 1 as $n \to \infty$. Let R' denote the output of the binary symmetric channel and Ω denote the set of non-erased entries. Let $\widehat{R}_{ij} = R'_{ij}$ if $(i,j) \in \Omega$ and $\widehat{R}_{ij} = 0$ otherwise. The goal is to exactly recover the row and column clusters from the observation $\widehat{R}$.

3.2.2 Main Results

The main results are summarized in Table 3.1. Note that these results do not explicitly depend on p. In fact, as p is assumed to be a constant strictly less than $1/2$, it affects the results by constant factors.

The parameter regime where exact cluster recovery is fundamentally impossible for any algorithm is proved in Section 3.4. The combinatorial method, convex method and spectral method are studied in Section 3.5, Section 3.6 and Section 3.7, respectively. We only analyze the combinatorial method in the noiseless case where $p = 0$, but we believe a similar result is true for the noisy case as well. The parameter regime in which the convex method succeeds is obtained by assuming that a technical conjecture holds, which is justified through extensive simulation. The parameter regime in

which the spectral method succeeds is obtained for the first time for exact cluster recovery with a growing number of clusters. The nearest-neighbor clustering algorithm was proposed in [7]. It clusters the users by finding the $K-1$ most similar neighbors for each user. The similarity between users i and i' is measured by the number of items with the same observed rating, i.e.,

$$s_{ii'} = \sum_{j=1}^{n} \mathbb{I}_{\{\widehat{R}_{ij}\neq 0\}} \mathbb{I}_{\{\widehat{R}_{i'j}\neq 0\}} \mathbb{I}_{\{\widehat{R}_{ij}=\widehat{R}_{i'j}\}},$$

where $\mathbb{I}_{\{\cdot\}}$ is an indicator function. Items are clustered similarly. It is shown in [7] that the nearest-neighbor clustering algorithm exactly recovers user and item clusters when $n(1-\epsilon)^2 > C \log n$ for a constant C.

The number of observations needed for successful cluster recovery can be derived from the corresponding parameter regime using the identity $m = n^2(1-\epsilon)$ as shown in Table 3.1. For better illustration, we visualize our results in Figure 3.1. In particular, we take $\log(m/n)$ as x-axis and $\log K$ as y-axis and normalize both axes by $\log n$. Since exact cluster recovery becomes easy when the number of observations m and cluster size K increase, we expect that exact cluster recovery is easy near $(1, 1)$ and hard near $(0, 0)$.

From Figure 3.1, we can observe interesting trade-offs between algorithmic running time and statistical performance. In terms of the running time, the combinatorial method is exponential, while the other three algorithms are polynomial. In particular, the convex method can be casted as a semidefinite programming and solved in polynomial time. For the spectral method, the most computationally expensive step is the singular value decomposition of the observed data matrix which can always be done in time $O(n^3)$ and more efficiently when the observed data matrix is sparse. It is not hard to see that the time complexity for the nearest-neighbor clustering algorithm is $O(n^2 r)$ and more careful analysis reveals that its time complexity is $O(mr)$. On the other hand, in terms of statistical performance, the combinatorial method needs strictly fewer observations than the other three algorithms when there is no noise, and the convex method always needs fewer observations than the spectral method. It is somewhat surprising to see that the simple nearest-neighbor clustering algorithm needs fewer observations than the more sophisticated convex method when the cluster size K is $O(\sqrt{n})$.

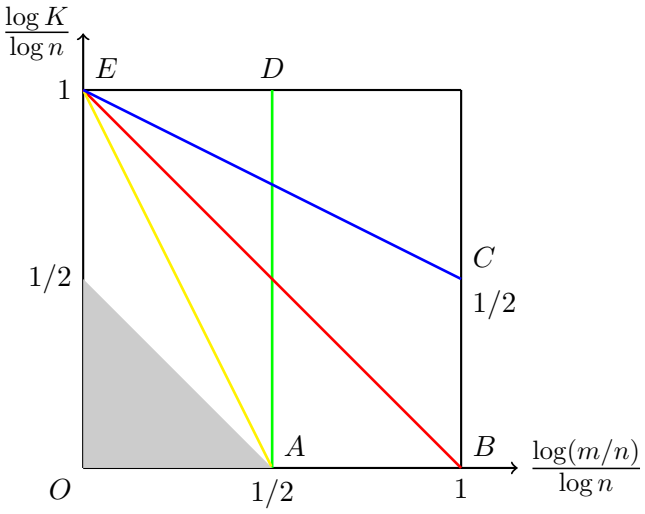

Figure 3.1: Summary of results in terms of number of observations m and cluster size K. The lower bound states that it is impossible for any algorithm to reliably recover the clusters exactly in the shaded regime (gray). The combinatorial method, the convex method, the spectral method and the nearest-neighbor clustering algorithm succeed in the regime to the right of lines AE (yellow), BE (red), CE (blue) and AD (green), respectively.

In summary, we see that when more observations available, one can apply algorithms with less running time while still achieving exact cluster recovery. For example, consider the noiseless case with cluster size $K = n^{0.8}$, the number of observations *per user* required for cluster recovery by the combinatorial method, convex method, spectral method and nearest-neighbor clustering algorithm are $\Omega(n^{0.1})$, $\Omega(n^{0.2})$, $\Omega(n^{0.4})$ and $\Omega(n^{0.5})$, respectively. Therefore, when the number of observations per user increases from $\Omega(n^{0.1})$ to $\Omega(n^{0.5})$, one can gradually reduces the computational complexity from exponential-time to polynomial-time as low as $O(n^{1.7})$.

The main results in this chapter of the thesis can be easily extended to the more general case with n_1 rows and $n_2 = \Theta(n_1)$ columns and r_1 row clusters and $r_2 = \Theta(r_1)$ column clusters. The sizes of different clusters could vary as long as they are of the same order. Likewise, the flipping probability p and the erasure probability ϵ could also vary for different entries of the data matrix as long as they are of the same order. Due to space constraints, such generalizations are omitted in this thesis.

3.3 Related Work

In this section, we point out some connections of our model and results to prior work. There is a vast literature on clustering and we only focus on theoretical works with rigorous performance analysis. More detailed comparisons are provided after we present the theorems.

3.3.1 Graph Clustering

Much of the prior work on graph clustering, as surveyed in [27], focuses on graphs with a single node type, where nodes in the same cluster are more likely to have edges among them. A low-rank plus sparse matrix decomposition approach is proved to exactly recover the clusters with the best-known performance guarantee in [28]. The same approach is used to recover the clusters from a partially observed graph in [29]. A spectral method for exact cluster recovery is proposed and analyzed in [30] with the number of clusters fixed. More recently, [31] proved an upper bound on the number of nodes "mis-clustered" by a spectral clustering algorithm in the high-dimensional setting with a growing number of clusters. An interesting recent work [32] studies the graph clustering problem under both non-adaptive and adaptive sampling strategies of node pairs.

In contrast to the above works, in our model, we have a labeled bipartite graph with two types of nodes (rows and columns). Notice that there are no edges among nodes of the same type and cluster structure is defined for the two types separately. In this sense, our cluster recovery problem can be viewed as a natural generalization of graph clustering problem to labeled bipartite graphs. In fact, our second algorithm via convex programming is inspired by the work [28, 29, 33].

A model similar to ours but with a fixed number of clusters has been considered in [34], where the spectral method plus majority voting is shown to *approximately* predict the rating matrix. However, our third algorithm via spectral method is shown to achieve exact cluster and rating matrix recovery with a growing number of clusters. To our best knowledge, this is the first theoretical result on spectral method for exact cluster recovery with a growing number of clusters to our knowledge.

3.3.2 Biclustering

Biclustering [35, 36, 37, 38] tries to find (overlap) sub-matrices with particular patterns in a data matrix. Many of the proposed algorithms are based on heuristic searches without provable performance guarantees. Our cluster recovery problem can be viewed as a special case where the data matrix consists of non-overlapping sub-matrices with constant binary entries, and this thesis provides a thorough study of this special biclustering problem. Recently, there is a line of work studying another special case of biclustering problem, which tries to detect a single small submatrix with elevated mean in a large fully observed noisy matrix [39]. Interesting statistical and computational trade-offs are summarized in [40].

3.3.3 Low-Rank Matrix Completion

Under our model, the underlying true data matrix is a specific type of low-rank matrix. If we recover the true data matrix, we immediately get the user (or item) clusters by assigning the identical rows (or columns) of the matrix to the same cluster. In the noiseless setting with no flipping, the nuclear norm minimization approach [41, 42, 43] can be directly applied to recover the true data matrix and further recover the row and column clusters. Alternate minimization is another popular and empirically successful approach for low-matrix completion [44]. However, it is harder to analyze and the performance guarantee is weaker than nuclear norm minimization [45]. In the low noise setting with the flipping probability restricting to be a small constant, the low-rank plus sparse matrix decomposition approach [46, 47, 48] can be applied to exactly recover data matrix and further recover the row and column clusters.

The performance guarantee for our convex method is better than these previous approaches and it allows the flipping probability to be any constant less than $1/2$. Moreover, our proof turns out to be much simpler. The recovery of our true data matrix from binary observations can also be viewed as a specific type of one-bit matrix completion problem recently studied in [8]. However, [8] focuses on approximately recovering a low-rank matrix with real-valued entries.

3.4 Lower Bound

In this section, we derive a lower bound for any algorithm to reliably recover the user and item clusters. The lower bound is constructed by considering a genie-aided scenario where the set of flipped entries is revealed as side information, which is equivalent to saying that we are in the noiseless setting with $p = 0$. Hence, the true rating matrix R agrees with $\widehat{R}$ on all non-erased entries. We construct another rating matrix $\tilde{R}$ with the same item cluster structure as R but different user cluster structure by swapping two users in two different user clusters. We show that if $nK^2(1-\epsilon)^2 = O(1)$, then $\tilde{R}$ agrees with $\widehat{R}$ on all non-erased entries with positive probability, which implies that no algorithm can reliably distinguish between R and $\widehat{R}$ and thus recover user clusters.

Theorem 3.1. *Fix $0 < \delta < 1$. If $nK^2(1-\epsilon)^2 < \delta$, then with probability at least $1-\delta$, it is impossible for any algorithms to recover the user clusters or item clusters.*

Intuitively, Theorem 3.1 says that when the erasure probability is high and the cluster size is small that $nK^2(1-\epsilon)^2 = O(1)$, the observed rating matrix $\widehat{R}$ does not carry enough information to distinguish between different possible cluster structures.

3.5 Combinatorial Method

In this section, we study a combinatorial method which clusters users or items by searching for a partition with the least total number of "disagreements". We describe the method in Algorithm 3 for clustering users only. Items are clustered similarly. The number of disagreements $D_{ii'}$ between a pair of users i, i' is defined as the number of items satisfying that: The two ratings given by users i, i' are both observed and the observed two ratings are different. In particular, if for every item, the two ratings given by users i, i' are not observed simultaneously, then $D_{ii'} = 0$.

The idea of Algorithm 3 is to reduce the problem of clustering both users and items to a standard user clustering problem without item cluster structure. In fact, this algorithm looks for the optimal partition of the users which

Algorithm 3 Combinatorial Method

1: For each pair of users i, i', compute the number of disagreements $D_{ii'}$ between them.
2: For each partition of users into r clusters of equal size K, compute its total number of disagreements defined as

$$\sum_{i,i' \text{ in the same cluster}} D_{ii'}.$$

3: Output a partition which has the least total number of disagreements.

has the minimum total in-cluster distance, where the distance between two users is measured by the number of disagreements between them. The following theorem shows that such simple reduction does *not* achieve the lower bound given in Theorem 3.1. The optimal algorithm for our cluster recovery problem might need to explicitly make use of both user and item cluster structures.

Theorem 3.2. *If $nK(1-\epsilon)^2 \leq \frac{1}{4}$, then with probability at least $3/4$, Algorithm 3 cannot recover user and item clusters.*

Next we show that the above necessary condition for the combinatorial method is also sufficient up to a logarithmic factor when there is no noise, i.e., $p = 0$. We suspect that the theorem holds for the noisy setting as well, but we have not yet been able to prove this.

Theorem 3.3. *If $p = 0$ and $nK(1-\epsilon)^2 > C \log n$ for some constant C, then a.a.s. Algorithm 3 exactly recovers user and item clusters.*

This theorem is proved by considering a conceptually simpler greedy algorithm that does not need to know K. After computing the number of disagreements for every pair of users, we search for a largest set of users which have no disagreement between each other, and assign them to a new cluster. We then remove these users and repeat the searching process until there is no user left. In the noiseless setting, the K users from the same true cluster have no disagreement between each other. Therefore, it is sufficient to show that, for any set of K users consisting of users from more than one cluster, they have more than one disagreement with high probability under our assumption.

3.6 Convex Method

In this section, we show that the rating matrix R can be exactly recovered by a convex program, which is a relaxation of the maximum likelihood (ML) estimation. When R is known, we immediately get the user (or item) clusters by assigning the identical rows (or columns) of R to the same cluster.

Let $\mathcal{Y}$ denote the set of binary block-constant rating matrix with r^2 blocks of equal size. As the flipping probability $p < 1/2$, Maximum Likelihood (ML) estimation of R is equivalent to finding a $Y \in \mathcal{Y}$ which best matches the observation $\widehat{R}$:

$$\begin{aligned} &\max_{Y} \sum_{i,j} \widehat{R}_{ij} Y_{ij} \\ &\text{s.t.} \quad Y \in \mathcal{Y}. \end{aligned} \tag{3.1}$$

Since $|\mathcal{Y}| = \Omega(e^n)$, solving (3.1) via exhaustive search takes exponential time. Observe that $Y \in \mathcal{Y}$ implies that Y is of rank at most r. Therefore, a natural relaxation of the constraint that $Y \in \mathcal{Y}$ is to replace it with a rank constraint on Y, which gives the following problem:

$$\begin{aligned} &\max_{Y} \sum_{i,j} \widehat{R}_{ij} Y_{ij} \\ &\text{s.t.} \quad \operatorname{rank}(Y) \le r,\ Y_{ij} \in \{1, -1\}. \end{aligned}$$

Further by relaxing the integer constraint and replacing the rank constraint with the nuclear norm regularization, which is a standard technique for low-rank matrix completion, we get the desired convex program:

$$\begin{aligned} &\max_{Y} \sum_{i,j} \widehat{R}_{ij} Y_{ij} - \lambda \|Y\|_* \\ &\text{s.t.} \quad Y_{ij} \in [-1, 1]. \end{aligned} \tag{3.2}$$

The clustering algorithm based on the above convex program is given in Algorithm 4.

The convex program (3.2) can be casted as a semidefinite program and solved in polynomial time. Thus, Algorithm 4 takes polynomial time. Our performance guarantee for Algorithm 4 is stated in terms of the incoherence

Algorithm 4 Convex Method

1: (Rating matrix estimation) Solve for $\widehat{Y}$ the convex program (3.2).
2: (Cluster estimation) Assign identical rows (columns) of $\widehat{Y}$ to the same cluster.

parameter μ defined as follows. Since the rating matrix R has rank r, the Singular Vector Decomposition (SVD) is $R = U\Sigma V^\top$, where $U, V \in \mathbb{R}^{n\times r}$ are matrices with orthonormal columns and $\Sigma \in \mathbb{R}^{r\times r}$ is a diagonal matrix with non-negative entries. Define incoherence parameter $\mu > 0$ such that $\|UV^\top\|_\infty \le \mu\sqrt{r}/n$. A small value of μ means that the left and right singular vectors of R are unaligned with each other. Denote the SVD of the block rating matrix B by $B = U_B\Sigma_B V_B^\top$. In Lemma 3.1 we show that

$$\|UV^\top\|_\infty = \|U_B V_B^\top\|_\infty / K, \tag{3.3}$$

and thus it is not hard to show that μ is upper bounded by $\sqrt{r}$.

Lemma 3.1. $\mu \le \sqrt{r}$.

Recent studies [41, 42, 43] in low-rank matrix completion have demonstrated that the number of samples needed for exact low-rank matrix recovery depends on the incoherence parameter μ. Not surprisingly, the performance guarantee for Algorithm 4 given by the following theorem also depends on μ.

Theorem 3.4. *If $n(1-\epsilon) \ge C' \log^2 n$ for some constant C', and*

$$m > Cnr \max\{\log n, \mu^2\}, \tag{3.4}$$

where C is a constant and μ is the incoherence parameter for R, then a.a.s. the rating matrix R is the unique maximizer to the convex program (3.2) *with* $\lambda = 3\sqrt{(1-\epsilon)n}$.

Our proof shows that with appropriate choices of λ, the nuclear norm regularization is effective in "de-noising" and the effectiveness depends on $\|UV^\top\|_\infty$. This is exactly why our performance guarantee depends on the incoherence parameter μ. Note that Algorithm 4 is easy to implement as λ only depends on the erasure probability ϵ, which can be reliably estimated from $\widehat{R}$. Moreover, the particular choice of λ in the theorem is just to simplify

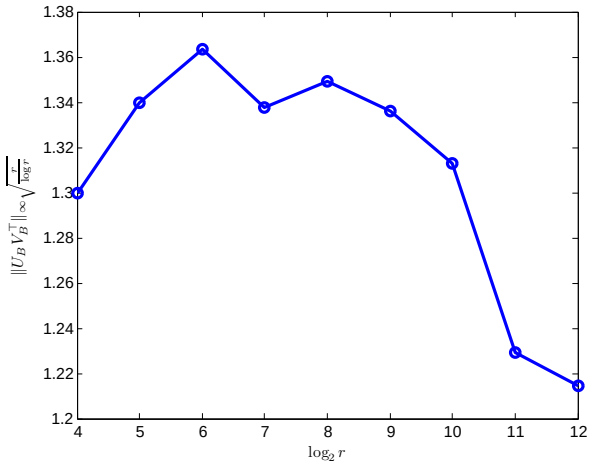

Figure 3.2: Simulation result supporting Conjecture 3.1. The conjecture is equivalent to $\|U_B V_B^\top\|_\infty = \Theta(\sqrt{\frac{\log r}{r}})$.

notations. It is straightforward to generalize our proof to show that the above theorem holds with $\lambda = C_1\sqrt{(1-\epsilon)n}$ for any constant $C_1 \geq 3$.

Using Lemma 3.1, we immediately conclude from the above theorem that the convex program succeeds when $m > Cnr^2$ for some constant C. However, based on extensive simulation in Figure 3.2, we conjecture that the following result is true.

Conjecture 3.1. $\mu = \Theta(\sqrt{\log r})$ *a.a.s.*

Conjecture 3.1 is equivalent to $\|U_B V_B^\top\|_\infty = \Theta(\sqrt{\frac{\log r}{r}})$ due to (3.3). For a fixed r, we simulate 1000 independent trials of B, pick the largest value of $\|U_B V_B^\top\|_\infty$, scale it by dividing $\sqrt{\log r/r}$, and get the plot in Figure 3.2.

Assuming this conjecture holds, Theorem 3.4 implies that

$$m > Cnr \log n$$

for some constant C is sufficient to recover the rating matrix, which is better than the previous condition by a factor of r. We do not have a proof for the conjecture at this time.

Comparison to previous work In the noiseless setting with $p = 0$, the nuclear norm minimization approach [41, 42, 43] can be directly applied to recover data matrix and further recover the row and column clusters. It is shown in [43] that the nuclear norm minimization approach exactly recovers

the matrix with high probability if $m = \Omega(\mu^2 nr \log^2 n)$. The performance guarantee for Algorithm 4 given in (3.4) is better by at least a factor of $\log n$. Theorem 3.3 shows that the combinatorial method exactly recovers the row and column clusters if $m = \Omega(nr^{1/2} \log^{1/2} n)$, which is substantially better than the two previous conditions by at least a factor of $r^{1/2}$.

In the low noise setting with p restricted to be a small constant, the low-rank plus sparse matrix decomposition approach [46, 47, 48] can be applied to exactly recover data matrix and further recover the row and column clusters. It is shown in [48] that a weighted nuclear norm and l_1 norm minimization succeeds with high probability if $m = \Omega(\rho_r \mu^2 nr \log^6 n)$ and $p \leq \rho_s$ for two constants ρ_r and ρ_s. The performance guarantee for Algorithm 4 given in (3.4) is better by several $\log n$ factors and we allow the fraction of noisy entries p to be any constant less than $1/2$. Moreover, our proof turns out to be much simpler. The recovery of our true data matrix from binary observations can also be viewed as a specific type of one-bit matrix completion problem [8]: Given an unknown rank-r matrix M, generate a binary matrix $Y \in \{\pm 1\}^{n \times n}$ such that $Y_{ij} = 1$ with probability $f(M_{ij})$ and the task is to recover M from a partial observation of Y. By taking $f(1) = 1 - p, f(-1) = p$, our problem reduces to the one-bit matrix completion problem. It is shown in [8] that approximate recovery is possible using the maximum likelihood estimation with nuclear norm constraint. In contrast, as shown in Theorem 4, our convex method yields exact recovery.

3.7 Spectral Method

In this section, we study a polynomial-time clustering algorithm based on the spectral projection of the observed rating matrix $\widehat{R}$. The description is given in Algorithm 5.

Step 1 of the algorithm produces two subsets, Ω_1 and Ω_2, of Ω such that: (1) for $i \in \{1, 2\}$, each rating is observed in Ω_i with probability $\frac{1-\epsilon}{2}$, independently of other elements; and (2) Ω_1 is independent of Ω_2. The purpose of Step 1 is to remove dependency between Step 2 and Steps 3 and 4 in our proof. In particular, to establish our theoretical results, we identify the initial clustering of users and items using Ω_1, and then majority voting and reclustering are done using Ω_2. In practice, one can simply use the same set

Algorithm 5 Spectral Method

1: (Producing two subsets, Ω_1 and Ω_2, of Ω via randomly sub-sampling Ω) Let $\delta = \frac{1-\epsilon}{4}$, and independently assign each element of Ω only to Ω_1 with probability $\frac{1}{2} - \delta$, only to Ω_2 with probability $\frac{1}{2} - \delta$, to both Ω_1 and Ω_2 with probability δ, and to neither Ω_1 nor Ω_2 with probability δ. Let $\widehat{R}^{(1)}_{i,j} = \widehat{R}_{i,j}\mathbb{I}_{\{(i,j)\in\Omega_1\}}$ and $\widehat{R}^{(2)}_{i,j} = \widehat{R}_{i,j}\mathbb{I}_{\{(i,j)\in\Omega_2\}}$ for $i, j \in \{1, \ldots, n\}$.

2: (Approximate clustering) Let $P_r(\widehat{R}^{(1)})$ denote the rank r approximation of $\widehat{R}^{(1)}$ and let x_i denote the i-th row of $P_r(\widehat{R}^{(1)})$. Construct user clusters $\widehat{C}_1, \ldots, \widehat{C}_r$ sequentially as follows. For $1 \leq k \leq r$, after $\widehat{C}_1, \ldots, \widehat{C}_{k-1}$ have been selected, choose an initial user not in the first $k-1$ clusters, uniformly at random, and let $\widehat{C}_k = \{i' : ||x_i - x_{i'}|| \leq \tau\}$. (The threshold τ is specified below.) Assign each remaining unclustered user to a cluster arbitrarily. Similarly, construct item clusters $\widehat{D}_1, \ldots, \widehat{D}_r$ based on the columns of $P_r(\widehat{R}^{(1)})$.

3: (Block rating estimation by majority voting) For $k, l \in \{1, \ldots, r\}$, let $\widehat{V}_{kl} = \sum_{i\in\widehat{C}_k}\sum_{j\in\widehat{D}_l}\widehat{R}^{(2)}_{ij}$ be the total vote that user cluster $\widehat{C}_k$ gives to item cluster $\widehat{D}_l$. If $\widehat{V}_{kl} \geq 0$, let $\widehat{B}_{kl} = 1$; otherwise, let $\widehat{B}_{kl} = -1$.

4: (Reclustering by assigning users and items to nearest centers) Recluster users as follows. For $k \in \{1, \ldots, r\}$, define center μ_k for user cluster $\widehat{C}_k$ as $\mu_{kj} = \widehat{B}_{kl}$ if item $j \in \widehat{D}_l$ for all j. Assign user i to cluster k if $\langle \widehat{R}^{(2)}_{i,\cdot}, \mu_k\rangle \geq \langle \widehat{R}^{(2)}_{i,\cdot}, \mu_{k'}\rangle$ for all $k' \neq k$. Recluster items similarly.

of observations, i.e., $\Omega_1 = \Omega_2 = \Omega$.

The following theorem shows that the spectral method exactly recovers the user and item clusters under a condition stronger than (3.4). In particular, we show that Step 3 exactly recovers the block rating matrix B and Step 4 cleans up clustering errors made in Step 2.

Theorem 3.5. *If*

$$n(1-\epsilon) > Cr^2 \log^2 n, \tag{3.5}$$

for a positive constant C, then Algorithm 5 with $\tau = 12(1-\epsilon)^{1/2} r \log n$ a.a.s. exactly recovers user and item clusters, and the rating matrix R.

Algorithm 5 is also easy to implement as τ only depends on parameters ϵ and r. The erasure probability ϵ can be reliably estimated from $\widehat{R}$ using empirical statistics. The number of clusters r can be reliably estimated by searching for the largest eigen-gap in the spectrum of $\widehat{R}$ (see Algorithm 2 and Theorem 3 in [28] for justification). We further note that the threshold τ used in the theorem can be replaced by $C_1(1-\epsilon)^{1/2} r \log n$ for any constant $C_1 \geq 12$.

Comparison to previous work Variants of spectral method are widely used for clustering nodes in a graph. Step 2 of Algorithm 5 for approximate clustering has been previously proposed and it is analyzed in [49]. In [30], an adaptation of Step 1 is shown to exactly recover a fixed number of clusters under the planted partition model. More recently, [31] proves an upper bound on the number of nodes "mis-clustered" by spectral method under the stochastic block model with a growing number of clusters.

Compared to previous work, the main novelty of Algorithm 5 is the Steps 1, 3, and 4 which allow for exact cluster recovery even with a growing number of clusters. To our knowledge, Theorem 3.5 provides the first theoretical result on the spectral method for exact cluster recovery with a growing number of clusters.

3.8 Numerical Experiments

In this section, we illustrate the performance of the convex method and the spectral method using synthetic data.

3.8.1 Convex Method

The convex program (3.2) can be formulated as a semidefinite program (SDP) and solved using a general purpose SDP solver. However this method does not scale well for our problem when the matrix dimension n is large. Instead we apply the accelerated gradient descent method proposed in [50, 51] which aims to solve the optimization problem

$$\min_Y f(Y) + \lambda||Y||_*$$

for some smooth function $f(Y)$. In our case, the smooth function is linear, i.e., $f(Y) = -\langle \hat{R}, Y \rangle$. Define proximal regularization of $f(Y)$ at X as

$$\begin{aligned} P_\mu(X, Y) =& f(X) - \langle Y - X, \hat{R} \rangle + \frac{\mu}{2}||Y - X||_F^2 \\ =& - \langle Y, \hat{R} \rangle + \frac{\mu}{2}||Y - X||_F^2 \end{aligned}$$

for some constant $\mu > 0$. Then it is shown in [50] that (3.2) is solved by the following iterative algorithm:

$$Y_k = \arg \min_{Y_{ij} \in [-1,1]} P_\mu(Y_{k-1}, Y) + \lambda||Y||_*. \tag{3.6}$$

We approximate Y_k by first solving the unconstrained optimization problem

$$\min_Y P_\mu(Y_{k-1}, Y) + \lambda||Y||_* \tag{3.7}$$

and then project each entry of the solution to $[-1, 1]$, where we use $P_{[-1,1]}$ to denote the projection operator. The minimizer of (3.7) can be explicitly written in terms of the soft-thresholding operator D defined as follows. For any $\gamma \geq 0$ and for any matrix X with SVD $X = U\Sigma V^\top$ where $\Sigma = \text{diag}(\{\sigma_i\})$, define

$$D_\gamma(X) = U\text{diag}(\{\max(\sigma_i - \gamma, 0)\})V^\top.$$

Intuitively, the soft-thresholding operator D shrinks the singular values of X towards zero. Applying Theorem 2.1 in [52], we get

$$D_{\frac{\lambda}{\mu}}\left(X + \frac{\widehat{R}}{\mu}\right) = \arg\min_Y P_\mu(X, Y) + \lambda||Y||_*.$$

Thus, the update equation (3.6) of Y_k is approximated by

$$Y_k = P_{[-1,1]}\left(D_{\frac{\lambda}{\mu}}(Y_{k-1} + \frac{\widehat{R}}{\mu})\right).$$

This iterative algorithm can further be accelerated to achieve the optimal convergence rate of $O(1/k^2)$, which results Algorithm 6 [50]. Note that we do not use a fixed regularization parameter λ. The algorithm has better performance when we start with $\lambda_0 = 3\sqrt{(1-\epsilon)n}$ as in Theorem 3.4 and decrease it gradually until it reaches $\bar{\lambda} = \sqrt{(1-\epsilon)n}$. In the experiment, we choose $\mu_k = 1$.

Algorithm 6 Accelerated Gradient Descent Algorithm

Input: $\widehat{R}$
Initialization: Set $Y_0 = Y_{-1} = 0$ and $\alpha_0 = \alpha_{-1} = 1$. Pick $\lambda_0 = 3\sqrt{(1-\epsilon)n}$ and $\bar{\lambda} = \sqrt{(1-\epsilon)n}$. Set $\gamma = 0.95$. Set $\mu_k = 1$.
for $k = 0, 1, 2, \ldots$ **do**
$Z_k = Y_k + \frac{\alpha_{k-1}-1}{\alpha_k}(Y_k - Y_{k-1})$
$Y_{k+1} = P_{[-1,1]}\left(D_{\frac{\lambda_k}{\mu_k}}\left(Z_k + \frac{\hat{R}}{\mu_k}\right)\right)$
$\alpha_{k+1} = \frac{1+\sqrt{1+4\alpha_k^2}}{2}$
$\lambda_{k+1} = \max\{\gamma\lambda_k, \bar{\lambda}\}$.
end for

We simulate Algorithm 6 on the synthetic data. Assume K and ϵ take the form given by

$$K = n^{\beta}, \quad \epsilon = 1 - n^{-\alpha}. \tag{3.8}$$

Theorem 3.4 shows that the convex program (3.4) recovers the rating matrix exactly when $\alpha < \beta$, assuming Conjecture 3.1 holds.

We generate the observed data matrix with $n = 2048$, $p = 0.05$ and various choices of $\beta, \alpha \in (0, 1)$, and apply Algorithm 6. The solution $\widehat{Y}$ is evaluated by the fraction of entries with correct signs, i.e., $\frac{1}{n^2}|\{(i,j) : \mathrm{sign}(\widehat{Y}_{ij}) = R_{ij}\}|$. The result is plotted in grayscale in Figure 3.3. In particular, the white area represents exact recovery and the black area represents around 50% recovery, which is equivalent to random guess. The red line represents $\alpha = \beta$, which shows the performance guarantee given by Theorem 3.4. As we can see, the simulation results roughly match the theoretical performance guarantee.

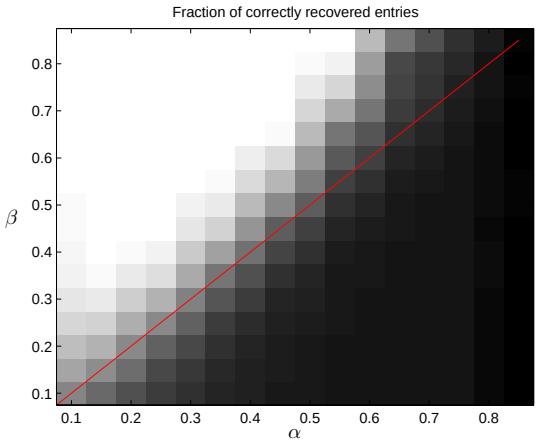

Figure 3.3: Simulation result of the convex method (Algorithm 6) with $n = 2048$ and $p = 0.05$. The x-axis corresponds to erasure probability $\epsilon = 1 - n^{-\alpha}$ and y-axis corresponds to cluster size $K = n^{\beta}$. The grayscale of each area represents the fraction of entries with correct signs, with white representing exact recovery and black representing around 50% recovery. The red line shows the performance of the convex method predicted by Theorem 3.4.

3.8.2 Spectral Method

We simulate the spectral method given in Algorithm 5 on synthetic data. Assume K and ϵ take the form of (3.8). Theorem 3.5 shows that the spectral method exactly recovers the clusters when $\alpha < \frac{1}{2}(\beta + 1)$.

We generate the observed data matrix according to our model with $n = 2^{11}, 2^{12}, 2^{13}$ and $p = 0.05$, and various choices of $\beta, \alpha \in (0, 1)$. We apply Algorithm 5 with slight modifications. Firstly, we do not split the observation as in Step 1 but use all the observations for the later steps, i.e., $\Omega_1 = \Omega_2 = \Omega$. Secondly, in Step 2 we use the more robust k-means algorithm to cluster users and items instead of the thresholding based clustering algorithm. The clustering error is measured by the fraction of mis-clustered users and items. We say the algorithm succeeds if the clustering error is less than 5%.

For each β, we run the algorithm for several values of α and record the largest α for which the algorithm succeeds. The result is depicted in Figure 3.4. The solid blue line represents $\alpha = \frac{1}{2}(\beta + 1)$, which shows the performance guarantee of the spectral method given by Theorem 3.5. The solid red line represents $\alpha = \beta$, which shows the performance guarantee of the convex method given by Theorem 3.4. We can see that the simulation results of the

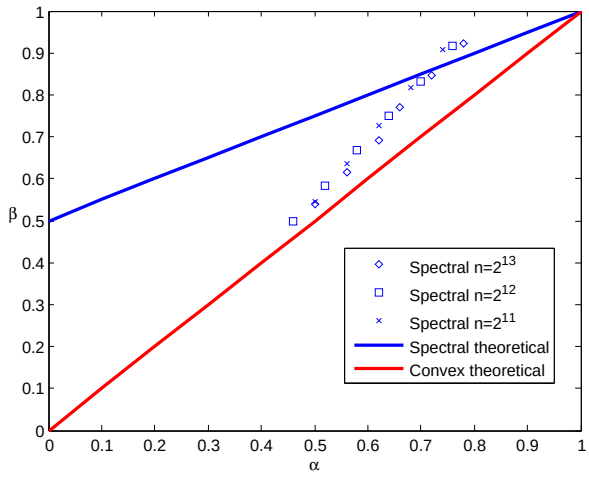

Figure 3.4: Simulation result of the spectral method given in Algorithm 5 with $n = 2^{11}, 2^{12}, 2^{13}$ and $p = 0.05$. The x-axis corresponds to erasure probability $\epsilon = 1 - n^{-\alpha}$ and y-axis corresponds to cluster size $K = n^{\beta}$. Each data point in the plot indicates the maximum value of α for which the spectral method succeeds with a given β. The blue solid line shows the performance of the spectral method predicted by Theorem 3.5. The red solid line shows the performance of the convex method predicted by Theorem 3.4.

spectral method are better than its theoretical performance guarantee, but worse than the theoretical performance guarantee of the convex method.

CHAPTER 4

RANKING ITEMS USING PAIRWISE COMPARISONS FROM MULTIPLE TYPES OF USERS

4.1 Introduction

In this chapter, we consider the problem of ranking items using pairwise comparisons obtained from multiple types of users. This scenario is similar to the previous one, but instead of giving binary ratings, users provide pairwise comparisons of items. We propose a two-step algorithm for estimating the score vectors: first cluster the users using projected comparison vectors and then estimate a score vector separately for each cluster by the maximum likelihood estimation for the classical Bradley-Terry model. The key observation is that, even though each user is represented by a high-dimensional comparison vector, the corresponding expected comparison vector is determined by only a small number of parameters and it lies close to a low-dimensional linear subspace.

4.2 Problem Setup

Consider a system with r user clusters of sizes K and m items and let $n = rK$. Each user u has a score vector for the items $\theta_u = (\theta_{u,1}, \ldots, \theta_{u,m})$, and it compares items according to the Bradley-Terry model: it prefers item i over item j with probability $\frac{e^{\theta_{u,i}}}{e^{\theta_{u,i}}+e^{\theta_{u,j}}}$ and the other way around with probability $\frac{e^{\theta_{u,j}}}{e^{\theta_{u,i}}+e^{\theta_{u,j}}}$. Assume users in the same cluster have the same score vector and denote the common score vector for cluster k by θ_k. Since θ_k is shift invariant, i.e., $(\theta_{k,1}, \ldots, \theta_{k,m})$ and $(\theta_{k,1} + C, \ldots, \theta_{k,m} + C)$ for any C define the same probabilities in the Bradley-Terry model, to eliminate the ambiguity and without loss of generality, we always shift θ_k to ensure that $\sum_i \theta_{k,i} = 0$. In this thesis, we will assume θ_k's are generated independently as follows: for

each k, generate $\theta^0_{k,i}$ i.i.d. uniformly in $[0, b]$, then define

$$\theta_{k,i} = \theta^0_{k,i} - \frac{1}{m}\sum_{i=1}^{m} \theta_{k,i}.$$

Clearly, $\sum_i \theta_{k,i} = 0$ and $|\theta_{k,i} - \theta_{k,j}| \leq b$ for any k, i and j. Though $\theta_{k,i}$ are not independent, we have $\theta_{k,i} - \theta_{k,j} = \theta^0_{k,i} - \theta^0_{k,j}$.

The comparison result is represented an $n \times \binom{m}{2}$ sample comparison matrix R. The u-th row R_u of R is the comparison vector by user u for $u = 1, \ldots, n$. The columns are indexed by two numbers $i, j = 1, \ldots, m$ with $i < j$, and the ij-th column corresponds to the comparisons for item i and j. For each user u and item i and j with $i < j$, we sample user u's comparison of item i and j with probability $1 - \epsilon$ independently. Let $R_{u,ij} = 1$ if u prefers i over j, $R_{u,ij} = -1$ if u prefers j over i, and $R_{u,ij} = 0$ if u's comparison is not sampled. Then

$$R_{u,ij} = \begin{cases} 1 & \text{w.p. } (1-\epsilon)\frac{e^{\theta_{u,i}}}{e^{\theta_{u,i}} + e^{\theta_{u,j}}} \\ 0 & \text{w.p. } \epsilon \\ -1 & \text{w.p. } (1-\epsilon)\frac{e^{\theta_{u,j}}}{e^{\theta_{u,i}} + e^{\theta_{u,j}}}. \end{cases}$$

Our goal is to estimate the score vectors θ_u from R.

4.3 Related Work

The Bradley-Terry model is a probabilistic way of modeling the rank aggregation problem. Here we briefly discuss another old and popular framework for rank aggregation. By viewing the scores for items as potentials that should match the comparison data, [53] proposed a simple least square approach for ranking football teams. The recent work [54] reformulates the problem using combinatorial Hodge theory and further decomposes the residual into local and global inconsistencies. This result is explained with more elementary linear algebra tools in [55] and extensive numerical experimental studies comparing the algorithms can be found therein.

To solve the Bradley-Terry model, in addition to the maximum likelihood estimation, several Markov chain based iterative methods have been proposed [56, 12]. It is shown in [12] that their algorithm has near optimal performance

when applied to estimate the score vectors for the Bradley-Terry model. Several generalizations of the classical Bradley-Terry model are studied in [11]. There are also other works focusing on permutations rather than scores [57, 58]. Most of the literature on rank aggregation assume that there is only one type of users.

A generalized Bradley-Terry model is considered in [59] for the crowdsourced setting where users have different qualities and the maximum likelihood algorithm is studied. We note that under this assumption, users still share the same score vector. A mixture approach is proposed in [60] for clustering heterogeneous ranking data and an effient EM algorithm is derived for parameter estimation. This method can take rankings of different lengths as input. However, no analytical performance guarantee is provided for the clustering.

Another similar line of work considers rating prediction using ratings of items from multiple types of users [7, 61, 62]. Our clustering algorithms are related to the algorithms in [61, 62]. The benefit of pairwise comparisons is that they are usually more reliable and consistent than the ratings. On the other hand, the number of pairwise comparisons is much larger than the number of items, thus the comparison vector is a much higher dimensional vector compared to a ranking of the items, which presents a different challenge in coming up with an algorithm for our problem.

4.4 Summary of Main Results

Before going into the details of our algorithm, we outline the main ideas in this section.

In this problem, we observe roughly $(1-\epsilon)n\binom{m}{2}$ comparisons from n users. Unlike the classical Bradley-Terry model, these users come from r clusters with different score vectors. If one simply treats the comparisons as being from users in a single cluster, the estimated score vector will only represent an aggregate opinion of all clusters and do not tell much about any individual user. Therefore, in this thesis, we consider a two-step algorithm for estimating the score vectors: it first clusters the users and then estimate a score vector for each cluster separately.

Each user is represented by the comparisons he/she provides. For each

user, there are m parameters we are interested in, however, the comparisons are given by a length $\binom{m}{2}$ binary vector which is extremely noisy. So instead of directly clustering comparison vectors, we would like to first denoise these vectors. For the moment, it is easier to understand the expected behavior of the comparison vectors and the difference from the actual comparison vectors can be taken care of later by concentration results. Consider user u with comparison vector R_u. The expectation of each entry is

$$\begin{aligned}\mathbb{E}\left[R_{u,ij}\right] =&(1-\epsilon)\frac{e^{\theta_{u,i}}-e^{\theta_{u,j}}}{e^{\theta_{u,i}}+e^{\theta_{u,j}}}\\ =&(1-\epsilon)\frac{e^{\theta_{u,i}-\theta_{u,j}}-1}{e^{\theta_{u,i}-\theta_{u,j}}+1}\\ \triangleq&(1-\epsilon)f(\theta_{u,i}-\theta_{u,j}),\end{aligned}$$

where $f(x)=\frac{e^x}{e^x+1}$. Let $A\in\{\pm 1,0\}^{m\times\binom{m}{2}}$ be the matrix with the ij-th column being e_i-e_j, where e_i is the length m vector with all 0s except for a 1 in the i-th coordinate. The expectation $\bar{R}_u\triangleq\mathbb{E}\left[R_u\right]=(1-\epsilon)f(\theta_u A)$. Though the dependence of $\bar{R}_u$ on θ_u is nonlinear, the key observation is that, when b is small or $|\theta_{u,i}-\theta_{u,j}|$ is small, we can linearize the function f at 0 and get

$$\bar{R}_{u,ij}\approx(1-\epsilon)\frac{\theta_{u,i}-\theta_{u,j}}{2}\text{ or }\bar{R}_u\approx(1-\epsilon)\frac{\theta_u A}{2}.$$

The important thing to notice is that the matrix A multiplied with θ_u is a known matrix the same for all users and independent of the observed comparisons. In this approximately linear regime, the immediate thing to do to reduce the noise of R_u is to project it onto the linear space spanned by the rows of A or the row space of A, and we denote the projection by $S_u=P_A(R_u)$. The signal θ_u's strength in $\bar{S}_u\triangleq\mathbb{E}\left[S_u\right]$ is roughly the same as in $\bar{R}_u$. However, as A is an $m\times\binom{m}{2}$ matrix, the noise strength is expected to be reduce by factor of m. Because S_u is less noisy compared with R_u, we get more reliable clustering performance when using S_u's. More importantly, it turns out that this projection is also helpful when b is not necessarily small. We remark that this result is somewhat surprising since the expected comparison vector is a nonlinear function of the score vector.

After clustering, we pull together users with the same or similar score vectors. Now it makes sense to view each cluster of users as from a single cluster and we applies the maximum likelihood estimation for the classical

Bradley-Terry model to estimate the score vectors. We show that, when the number of misclustered users is small compared to the cluster size (which happens with high probability), the score vector estimated for each cluster is a good approximation of the true score vector.

Using the above intuition, we derive the following main result of this chapter of the thesis.

Theorem 4.1. *If* $Km^2(1-\epsilon) > C'r\max\{m,n\}\log^5 n \log m$ *and* $b \in [0.6, 5]$, *then*

$$\frac{||\hat{\theta}_u - \theta_u||_2}{||\theta_u||_2} \leq \frac{(e^b+1)^2}{be^b}\frac{C}{\log^2 n}$$

except for $\frac{K}{\log^2 n}$ *users a.a.s.*

Since our problem involves r ranking problems of size m, at least $rm\log m$ comparisons are required to reliably estimate the score vectors. Theorem 4.1 shows that our algorithm needs approximately

$$\frac{1}{2}(1-\epsilon)Km^2 = O(r^2\max\{m,n\}\text{poly}(\log n)\log m)$$

comparisons. Suppose n and m are on the same order. If r is polylog in n or m, then our analysis shows that we require only a polylog factor of m more measurements than the minimum required. In the rest of the thesis, we present the intermediate theorems from which the above theorem immediately follows. As mentioned earlier, the proofs of all the results are in the supplemental material.

4.4.1 Preprocessing the Samples

Recall that our algorithm has two steps: user clustering and score vector estimation. To remove the dependency between these two steps in our analysis, we divide the sampled comparisons into two smaller samples with independent support sets. We emphasize that this is only necessary for the analysis, and in practice one can use R for both steps. Let Ω be the support of R, i.e., $\Omega = \{(u, ij)|R_{u,ij} \neq 0\}$. We construct two sets Ω_1 and Ω_2 by independently assigning each element of Ω only to Ω_1 or Ω_2 with probability b and to both

Ω_1 and Ω_2 with probability a, for some $a, b \in (0,1)$. Lemma 4.1 shows that for proper choice of a and b, Ω_1 and Ω_2 are independent.

Lemma 4.1. *When $a = (1-\epsilon)/4$ and $b = (1+\epsilon)/4$, Ω_1 and Ω_2 are independent and $\mathbb{P}[(u,ij) \in \Omega_1] = \mathbb{P}[(u,ij) \in \Omega_2] = (1-\epsilon)/2$.*

Define $R^{(1)}_{u,ij} = R_{u,ij}\mathbb{I}_{\{(u,ij)\in\Omega_1\}}$ and $R^{(2)}_{u,ij} = R_{u,ij}\mathbb{I}_{\{(u,ij)\in\Omega_2\}}$, and in the algorithm, we use $R^{(1)}$ and $R^{(2)}$ for the first and second step, respectively.

4.5 Clustering

In this section, we present two algorithms for clustering the users using the sampled comparison matrix $R^{(1)}$. To simplify the notation, we abuse the notation by calling the sampled comparison matrix R instead through this section, but we note that the sampling probability is only $(1-\epsilon)/2$.

As mentioned in Section 4.4, the comparison vectors for the users are in high-dimensional space and are very noisy for clustering. When b is small, by a linearization argument, we show that the expected comparison vectors $\bar{R}_u$'s are close to the row space of A, therefore we can project the comparison vectors onto this linear subspace to reduce the noise before clustering. Now we describe the projection in more detail.

4.5.1 Cluster Separation Preserving Projection

We first summarize a few properties of A.

Lemma 4.2. *The matrix A is of rank $m-1$ with SVD $A = \sqrt{m}UV^\top$, where $U \in \mathbb{R}^{n\times(m-1)}$ and $V \in \mathbb{R}^{\binom{m}{2}\times(m-1)}$. Moreover, the l_2-norms of the rows of U and V are $\sqrt{(m-1)/m}$ and $\sqrt{2/m}$, respectively.*

Note that $V^\top$ is an orthonormal basis of the row space of A and the projection of any row vector $\eta^\top$ onto this space is given by $\eta^\top VV^\top$. In particular, the cosine of the angle between $\eta^\top$ and the row space of A is $\frac{||\eta^\top V||_2}{||\eta||_2}$.

For any user u, we showed in Section 4.4 that when $|\theta_{u,i} - \theta_{u,j}|$ are small for any i and j, $\bar{R}_u$ is close to the row space of A. Here we consider the other extreme when $|\theta_{u,i} - \theta_{u,j}| \to \infty$ for any i and j. In this case, we have

$\bar{R}_{u,ij} \approx \frac{1-\epsilon}{2}\text{sign}(\theta_{u,i} - \theta_{u,j})$, and Lemma 4.3 implies that $\bar{R}_u$ is again close to the row space of A by showing that the angle between $\bar{R}_k$ and the row space of A is small.

Lemma 4.3. *For any $\theta_k \in \mathbb{R}^m$ and assume $\theta_{k,i} \neq \theta_{k,j}$ for any i and j. Define $\eta \in \{-1, +1\}^{\binom{m}{2}}$ as $\eta_{ij} = \text{sign}(\theta_{k,i} - \theta_{k,j})$. Then $||\eta^\top V||_2^2 = \frac{1}{3}(m^2 - 1)$. Since $||\eta||_2^2 = \frac{1}{2}m(m-1)$, the angle between $\eta^\top$ and row space of A is $\arccos\sqrt{\frac{2}{3}}$ in the limit as $m \to \infty$.*

Motivated by the linearization argument presented in Section 4.4 and the above observation, we instead represent the users by the projected comparison vectors $S_u = R_u V$ which is of length $m - 1$. By the assumption on θ_u's, the rows of $\bar{R} = \mathbb{E}[R]$ and $\bar{S} = \mathbb{E}[S]$ are the same for users in the same cluster and denote the common row for cluster k by $\bar{R}_k$ and $\bar{S}_k$, respectively. It is not difficult to see that $||\bar{R}_k - \bar{R}_{k'}||_2$ is of order $O((1-\epsilon)m)$. Theorem 4.2 shows that $\bar{S}_k$'s are also separated by a distance of $C(1-\epsilon)m$, which means the separation between $\bar{R}_k$'s are preserved after the projection.

Theorem 4.2. *Assume $m \geq C' \log r$ for some constant C'. If $b \in [0.6, 5]$ or $b \geq C'' m^3 \log m$, then a.a.s. there exists some constant C such that for any $k \neq k'$,*

$$||\bar{S}_k - \bar{S}_{k'}||_2 \geq C(1-\epsilon)m.$$

We note that even though this lemma requires $b \in [0.6, 5]$ or b very large, our experiment shows that the $\bar{S}_k$'s are in fact well separated for any $b \geq 0.6$. Moreover, our analysis does not restrict to the Bradley-Terry model. For any pairwise comparison model, as long as $\bar{R}_{u,ij}$ depends on $\theta_{u,i} - \theta_{u,j}$ through a function with hyperbolic tangent shape, the same result should still hold. We note that a lower bound on b is required since, if b is very small, then all $\theta_{u,i}$'s are close to zero and there is no way to distinguish between the clusters.

4.5.2 Two Clustering Algorithms

Our first algorithm is called projected clustering and it clusters the rows of the matrix $S = RV$.

The following theorem shows that this algorithm clusters the users exactly when the number of observations is large enough.

Algorithm 7 Projected Clustering

Step 1: Define $S = RV$.
Step 2: Construct the clusters $\hat{\mathcal{C}}_1, \ldots, \hat{\mathcal{C}}_r$ sequentially. For $1 \le k \le r$, after $\hat{\mathcal{C}}_1, \ldots, \hat{\mathcal{C}}_{k-1}$ have been selected, choose an initial user u not in the first $k-1$ clusters uniformly at random, and let $\hat{\mathcal{C}}_k = \{u' : ||S_u - S_{u'}||_2 \le \tau\}$ where the threshold τ is specified later. Assign each remaining unclustered user to a cluster arbitrarily.

Theorem 4.3. *If $m(1-\epsilon) > C \log n$ for some constant C and $b \in [0.6, 5]$ or $b \ge C' m^3 \log m$, then a.a.s. Algorithm 7 with $\tau = 6\sqrt{(1-\epsilon)m \log n}$ clusters the users exactly.*

Algorithm 7 applies simple nearest-neighbor clustering and does not make full use of the cluster structure. One would expect that the clustering is easier when the cluster size K is larger but the result in Theorem 4.3 is independent of K. In the following, we will consider a variation of the algorithm based on the spectral method, which we call projected spectral clustering. Let $\tilde{S}$ be the rank r approximation of S, then our new algorithm clusters the rows of $\tilde{S}$ instead of the rows of S.

Algorithm 8 Projected Spectral Clustering

Step 1: Define $S = RV$. Let $\tilde{S}$ be the rank r approximation of S.
Step 2: Construct the clusters $\hat{\mathcal{C}}_1, \ldots, \hat{\mathcal{C}}_r$ sequentially. For $1 \le k \le r$, after $\hat{\mathcal{C}}_1, \ldots, \hat{\mathcal{C}}_{k-1}$ have be selected, choose an initial user u not in the first $k-1$ clusters uniformly at random, and let $\hat{\mathcal{C}}_k = \{u' : ||\tilde{S}_u - \tilde{S}_{u'}||_2 \le \tau\}$ where the threshold τ is specified later. Assign each remaining unclustered user to a cluster arbitrarily.

The following theorem shows that Algorithm 8 clusters users approximately when the number of samples is large enough. In particular, it shows that there are at most $o(K)$ users which are assigned to the wrong clusters.

Theorem 4.4. *Let $\mathcal{C}_k$ denote the true cluster k and $\hat{\mathcal{C}}_k$ denote the k-th cluster generated by Algorithm 8 with $\tau = 32\sqrt{2\frac{(1-\epsilon)r \max\{m,n\}}{K}} \log^{5/2} n$. If $Km^2(1-\epsilon) > Cr \max\{m,n\} \log^5 n$ and $b \in [0.6, 5]$ or $b \ge C' m^3 \log m$, then a.a.s. there exists a permutation π such that $|\mathcal{C}_k \triangle \hat{\mathcal{C}}_{\pi(k)}| \le \frac{K}{\log^2 n}$ and $\sum_k |\mathcal{C}_k \triangle \hat{\mathcal{C}}_{\pi(k)}| \le \frac{2K}{\log^2 n}$, where $\triangle$ denotes the symmetric difference of two sets.*

Compared with the previous result, this theorem shows that Algorithm 8 only approximately clusters the users, i.e., it allows $o(K)$ misclustered users in each cluster, however, it requires fewer observations when $K > Cr\log^4 n$. Based on our experiment, we believe that in practice Algorithm 8 is always better than Algorithm 7, and the requirement of K being large is an artifact of the analysis.

4.6 Score Vector Estimation

In this section, we consider the problem of estimating the score vectors for users using the sampled comparison matrix $R^{(2)}$. After clustering the users as in Section 4.5, we perform the score vector estimation for each cluster separately and will only take one such cluster $\hat{\mathcal{C}}$ as an example. For the clustering step, we will assume either the result of Theorem 4.3 or Theorem 4.4 holds, thus cluster $\hat{\mathcal{C}}$ is at least approximately equal to some true cluster $\mathcal{C}$, i.e., $|\mathcal{C}\Delta\hat{\mathcal{C}}| \leq K/\log^2 n$, and there are at most $K/\log^2 n$ users assigned to the wrong clusters. To simplify the notation, we omit the subscript and use θ to denote the true score vector for the cluster $\mathcal{C}$ throughout this section.

To estimate the score vectors for the users in $\hat{\mathcal{C}}$, we view these users as from a single cluster and apply the maximum likelihood estimation for the Bradley-Terry model to get a common score vector $\hat{\theta}$. Theorem 4.5 shows that when the number of comparisons is large enough, the relative error $\frac{||\hat{\theta}-\theta||_2}{||\theta||_2}$ is of order $o(1)$ when $n \to \infty$. We should emphasize that $\hat{\theta}$ is only a good approximation for the score vectors of the users from cluster $\mathcal{C}$ and it is likely to be a bad estimate for the small number of users from some other clusters.

Theorem 4.5. *Assume $(1-\epsilon)mK > C'\log^2 n \log m$ for some constant C'. Then a.a.s. there exists some constant C such that*

$$\frac{||\hat{\theta}-\theta||_2}{||\theta||_2} \leq \frac{(e^b+1)^2}{be^b}\frac{C}{\log^2 n}.$$

4.7 Experiments

In this section, we illustrate the performance of our algorithm using synthetic data. In the first experiment, we compare Algorithms 7 and 8 with a standard spectral clustering algorithm, and show that the projection is essential for clustering the users. In the second experiment, we demonstrate the performance of score vector estimation and suggest a heuristic for estimating the number of clusters.

4.7.1 Clustering Performance Comparison

In Algorithms 7 and 8, we cluster the rows of S and $\tilde{S}$ using a thresholding type of algorithm, which is easy to analyze. However, in practice, we will always use K-means clustering instead as it is more robust. We initialize the centers for K-means clustering as follows. First, randomly pick a row as a center. Then pick the row whose minimum distance from existing centers is maximized and add it to the centers. Continue this process until we have picked r centers. For comparison purpose, we also consider the standard spectral clustering algorithm that applies the K-means algorithm to cluster the rows of $\tilde{R}$, which is the rank r approximation of R.

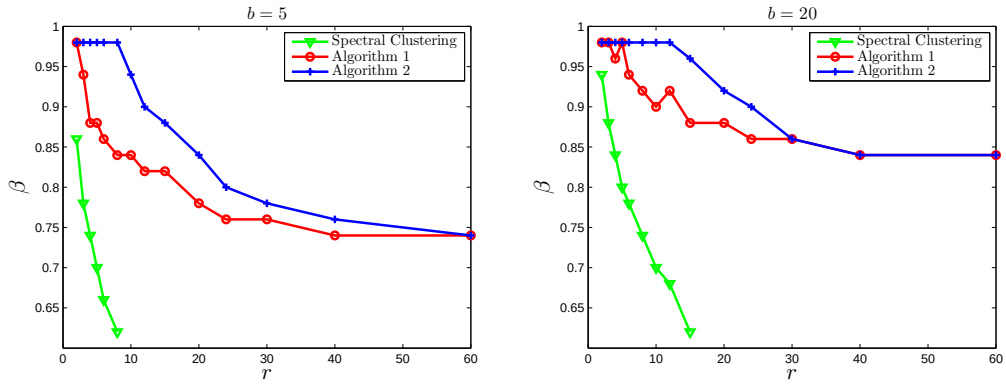

Figure 4.1: Performance comparison of the standard spectral clustering algorithm and Algorithms 7 and 8. The y-axis is β which represents the erasure probability $\epsilon = 1 - \frac{1}{m^\beta}$. The algorithms succeed in the parameter regime below the corresponding curves.

Let $\{C_k\}$ denote the true clusters and $\{\hat{C}_k\}$ denote the clusters generated by some clustering algorithm. For each k, we say $\hat{C}_k$ corresponds to true

cluster k' if the majority of users in $\hat{C}_k$ are from $C_{k'}$, and we count any user who is from a different true cluster as an error. Then the performance is measured by the total number of errors divided by the total number of users, i.e., the fraction of misclustered users.

We fix $m = n = 120$ and $b = 5$ or 20. Figure 4.1 shows the performance of these three algorithms. The x-axis is the number of clusters r and y-axis corresponds to the erasure probability ϵ. To better visualize the result, we choose the normalized log scale $\beta = \frac{\log(1-\epsilon)^{-1}}{\log m}$ as y-axis, i.e., $\epsilon = 1 - \frac{1}{m^\beta}$. Each point on a curve shows, for the given number of clusters r, the largest erasure probability ϵ such that the average fraction of misclustered users of an algorithm over 50 expriments is less than 5%, in which case we say the algorithm succeeds. In this figure, the algorithms succeed in the parameter regime below the corresponding curves.

Compared to our algorithms, the standard spectral clustering algorithm has very poor performance. It directly approximates $\bar{R}$ and requires many more samples, while our algorithms only approximate $\bar{S}$. On the other hand, as mentioned earlier, Algorithm 8 that uses the rank r approximation $\tilde{S}$ performs better than Algorithm 7 which uses S in all range of r. Note that the case $b = 20$ is not covered by our theorems, but Figure 4.1 still illustrates good performance in that case.

4.7.2 Estimating the Number of Clusters r

In practice, the number of user clusters r is usually not known *a priori*. One way to get around this difficulty is to first guess the number of clusters $\tilde{r}$ and then apply our algorithm. In the experiment, we first clusters the rows of $\tilde{S}$ using the K-means algorithm for each $\tilde{r}$ and then apply the maximum likelihood estimation for the score vector in each cluster.

We fix $m = n = 120$, $b = 5$ and $\epsilon = 0.95$. Figure 4.2 shows the simulation results for $r = 1, 2, 4$ and 8. For each r, the blue curve shows how the relative error $\frac{||\hat{\theta}-\theta||_2}{||\theta||_2}$ changes with $\tilde{r}$. When $\tilde{r}$ is smaller than r, two or more true clusters are assigned to one cluster and the error in $\hat{\theta}$ is large. On the other hand, when $\tilde{r}$ is equal or slightly larger than r, the estimation $\hat{\theta}$ approximate θ quite well as each cluster returned by our clustering algorithm is mainly consisted of users from one true cluster. In particular, in the first

plot where there is only one cluster, the relative estimation error does not grow much even for $\tilde{r} = 6$. However, when $\tilde{r}$ is too large, there will be many small clusters and the variance in $\hat{\theta}$ can be very large, which also could result large estimation error.

If we view $\hat{\theta}$ as a function of $\tilde{r}$, the red curve shows how the change of $\hat{\theta}$ in $\tilde{r}$, i.e., $||\hat{\theta}(\tilde{r}) - \hat{\theta}(\tilde{r}-1)||_2$, changes with $\tilde{r}$. For comparison purpose, we normalize this difference by $||\theta||_2$. From the experiment, a good heuristic for identifying the number of clusters r is by looking for the $\tilde{r}$ such that the change $||\hat{\theta}(\tilde{r}) - \hat{\theta}(\tilde{r}-1)||_2$ is minimized.

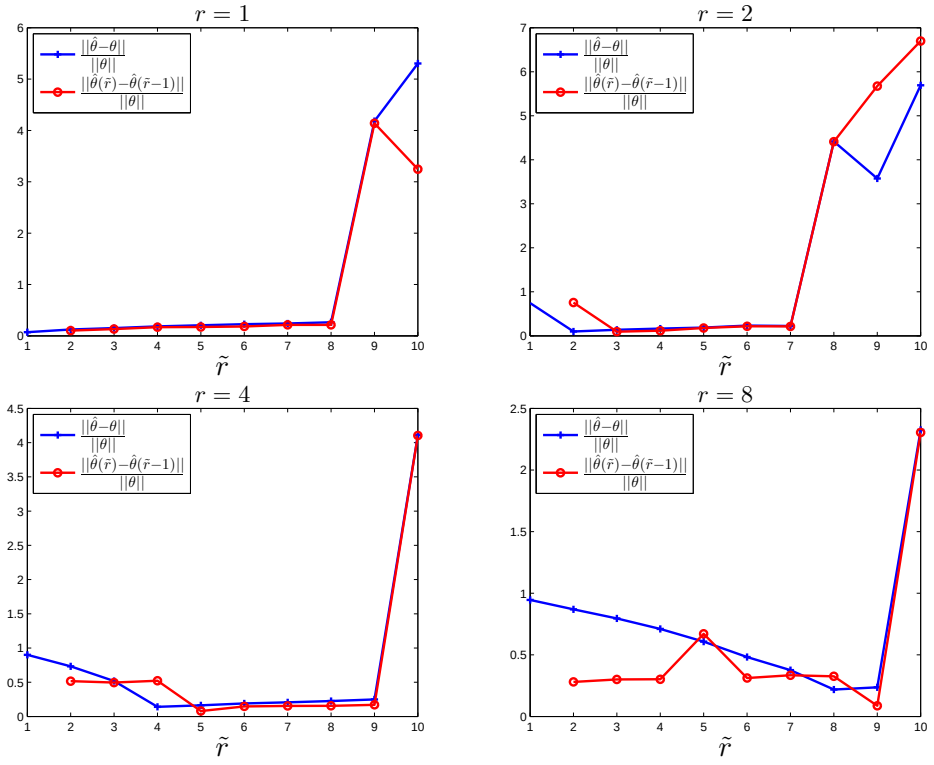

Figure 4.2: Score vector estimation for different r. For each r, the blue curve shows how the relative error $\frac{||\hat{\theta}-\theta||}{||\theta||}$ changes with $\tilde{r}$, and $\frac{||\hat{\theta}-\theta||}{||\theta||}$ is minimized when $\tilde{r} = r$. From the red curve, r can be identified by looking for the $\tilde{r}$ such that the change $||\hat{\theta}(\tilde{r}) - \hat{\theta}(\tilde{r}-1)||$ is minimized.

CHAPTER 5

CONCLUSION AND FUTURE DIRECTIONS

In this thesis we study the problem of learning the underlying network structure from the observed behavior of individual nodes. Such learning problems are computationally intractable, in general, for large networks. To facilitate efficient algorithms, we focus on two types of simplified structure assumptions on the network. In the first part, we consider loosely connected MRFs where the dependence of the nodes is encoded by a graph. The key assumption is that the network is sparse in the sense that the number of short paths between any pair of nodes is small, which allows us to propose a low complexity search-based structure learning algorithm. In the second and third parts, we turn to the case where nodes form clusters and analyzed several practical algorithms such as spectral clustering and convex relaxation of maximum likelihood estimation. Their performances are compared with fundamental limits on the number of observations or some high complexity combinatorial search method. In particular, for recommender systems where both users and items have cluster structure, we show that there is an interesting trade-off between the computational complexity and the statistical performance.

To conclude, we discuss several open questions as future research directions.

5.1 Maximum Likelihood Estimation and Computational Complexity Constraint

In Chapter 3 of the thesis, we see from Figure 3.1 that there is a gap between the combinatorial method and the lower bound. Note that the combinatorial method is suboptimal in the sense that it clusters the users and items separately. We expect the lower bound to be tight, and an interesting problem is to show that the maximum likelihood estimation, which jointly clusters the

users and items, achieves this lower bound.

We also observed that our exponential-time combinatorial method needs substantially fewer observations for successful cluster recovery than the other three polynomial-time counterparts, which suggests that a performance gap might exist between exponential-time algorithms and polynomial-time algorithms. Similar performance gaps due to the computational complexity constraint have also been observed recently in many other inference problems such as graph clustering [63, 33], sparse PCA [64, 65, 66] and sparse submatrix detection [39, 40, 67]. One future direction is to provide an upper bound on the performance that can be achieved by any polynomial-time algorithm.

5.2 Tensor Completion

In Chapter 3 of the thesis, as a byproduct of the convex relaxation approach, we show that the block-constant rating matrix can be recovered exactly under a condition that is slightly better than those required for low-rank matrix completion or low-rank plus sparse matrix decomposition. More importantly, our binary and block-constant assumption significantly simplifies the analysis. But the analysis does not apply to the case when ratings have three or more levels.

To get around this difficulty, we can consider the following tensor generalization for the original rating matrix. Suppose we are interested in only three types of ratings: like, dislike and do not care. Instead of using a single scalar to represent each rating, we user a length-three indicator vector, where each element of the vector corresponds to one type of rating. For example, if a user likes a item, his/her rating is $[1, 0, 0]$. Under this representation, the original $n \times n$ rating matrix becomes an $n \times n \times 3$ rating tensor, which preserves the desired property that each entry of the tensor is binary. Then a natural question is, can we use a similar approach to recover this tensor under noise and erasure?

Tensor completion has received much attention in recent years. Many algorithms have been proposed to solve this problem [68, 69, 70, 71]. The most common technique was first introduced in [68], which generalizes the convex relaxation technique for matrix completion to the tensor case by defining a trace norm with respect to the matrices obtained by unfolding a tensor.

See [70] for a review of tensor and its norms. Recently, another algorithm that performs Riemannian optimization on the manifold of tensors of a fixed multilinear rank is proposed in [71]. However, the derivation of recovery guarantee for tensor completion is still an open problem.

In light of the simple analysis of the matrix recovery problem arising from the recommender systems, one future direction is to provide recovery guarantee for the natural tensor generalization mentioned above.

5.3 Clustering Overlapping Clusters

In Chapters 3 and 4 of the thesis, we make the simplifying assumption that the network is consisted of disjoint clusters. However, in a variety of applications, it is more realistic to assume that the network contains overlapping clusters, i.e., the nodes can belong to multiple clusters. In recommender systems like Netflix, many movies belong to more than one genre, so the clusters for movies are likely to overlap. In biology, genes can influence more than one metabolic pathways, thus it is more reasonable to consider overlapping clusters when clustering genes using microarray data.

The most natural model for overlapping clusters is the latent model, in which each node has a latent vector indicating how much it is associated with each cluster [72, 73, 74]. The observed behavior of a node is determined by a weighted combinations of the properties of the clusters it belongs to. Then one can infer the latent vector for each user from the observations using standard algorithms such as the EM algorithms. Another approach is to consider a different clustering criterion that allows the nodes to belong to multiple clusters and then design algorithms accordingly [75, 76]. For example, in [75], instead of measuring the performance of a clustering using the distance of a node to its nearest cluster center, it uses the distance of a node to the average of several nearest cluster centers, and proposes a generalized K-means algorithm to solve the problem with respect to the new criterion. Some other algorithms can be found in the references in [76]. However, these works do not have theoretical guarantees for the performance of their algorithms.

The recent work [77] applies the tensor method to solve the community detection problem where communities can overlap. It assumes each node has

a latent vector indicating the probability it belongs to each cluster, and the algorithm estimates the latent vectors by decomposing the moment tensor using the power method. Moreover, it provides a performance guarantee for the algorithm by showing an upper bound on the difference between the estimated latent vectors and the original ones. One future direction is to generalize the analysis to the setting considered in this thesis.

APPENDIX A

PROOFS IN CHAPTER 2

A.1 Bounded Degree Graph

A.1.1 Proof of Lemma 2.4

Let N_S be the neighbor nodes of S. Note that each node in S has at most d neighbors in N_S.

$$\begin{aligned}
P(x_S) =& \sum_{x_{N_S}} P(x_{N_S}) P(x_S|x_{N_S}) \\
\geq& \min_{x_S, x_{N_S}} P(x_S|x_{N_S}) \\
=& \min_{x_S, x_{N_S}} \frac{\exp(x_S^T J_{SS} x_S + x_S^T J_{SN_S} x_{N_S})}{\sum_{x'_S} \exp({x'_S}^T J_{SS} x'_S + {x'_S}^T J_{SN_S} x_{N_S})} \\
\geq& \frac{\min_{x_S, x_{N_S}} \exp(x_S^T J_{SS} x_S + x_S^T J_{SN_S} x_{N_S})}{2^{|S|} \max_{x'_S, x_{N_S}} \exp({x'_S}^T J_{SS} x'_S + {x'_S}^T J_{SN_S} x_{N_S})} \\
\geq& \frac{\exp(-|S|^2 J_{\max} - |S| d J_{\max})}{2^{|S|} \exp(|S|^2 J_{\max} + |S| d J_{\max})} \\
=& 2^{-|S|} \exp(-2(|S| + d)|S| J_{\max}).
\end{aligned}$$

A.1.2 Correlation Decay and Large Girth

We assume that the Ising model on the bounded degree graph is further in the correlation decay regime. Both Theorem 2.1 and Lemma 2.5 immediately follow from the following more general result, which characterizes the conditions under which the Ising model is (D_1, D_2, ϵ)-loosely connected. We will make the connections at the end of this subsection.

Theorem A.1. *Assume* $(d-1)\tanh J_{\max} < 1$. *Fix* D_1, D_2. *Let* h *satisfy*

$$\beta\alpha^h \le A \wedge \ln 2,$$

where $A = \frac{1}{1800}(1 - e^{-4J_{\min}})e^{-8(D_1+D_2)dJ_{\max}}$, *and let* $\epsilon = 48Ae^{4(D_1+D_2)dJ_{\max}}$. *Assume that there are at most* D_1 *paths shorter than* h *between non-neighbor nodes and* D_2 *paths shorter than* h *between neighboring nodes. Then* $\forall (i,j) \in E$,

$$\min_{\substack{S\subset V\setminus\{i\cup j\}\\ |S|\le D_1}} \max_{\substack{T\subset V\setminus\{i\cup j\}\\ |T|\le D_2}} \max_{x_i,x_j,x_j',x_S,x_T} |P(x_i|x_j,x_S,x_T) - P(x_i|x_j',x_S,x_T)| > \epsilon,$$

and $\forall (i,j) \notin E$,

$$\min_{\substack{S\subset V\setminus\{i\cup j\}\\ |S|\le D_1}} \max_{\substack{T\subset V\setminus\{i\cup j\}\\ |T|\le D_2}} \max_{x_i,x_j,x_j',x_S,x_T} |P(x_i|x_j,x_S,x_T) - P(x_i|x_j',x_S,x_T)| \le \frac{\epsilon}{4}.$$

Proof. First consider $(i,j) \in E$. Without loss of generality, assume $J_{ij} > 0$. By the assumption that there are at most D_2 paths shorter than h between neighboring nodes, there exists $T' \subset N_i, |T'| \le D_2$ such that, when the set T' is removed from the graph, the length of any path from i to j is no less than h. For any S, let $T = T' \setminus S$. To simplify the notation, let $R = S \cup T$ and $W = V \setminus R$. For any value x_R, let Q be the joint probability of X_W conditioned on $X_R = x_R$, i.e., $Q(X_W) = P(X_W|x_R)$. Q has the same edge coefficients for the unconditioned nodes, but is not zero-field as conditioning induces external fields. Let $\tilde{Q}$ denote the joint probability when edge (i,j) is removed from Q. We note that Q and $\tilde{Q}$ satisfy the same correlation decay property as P, so

$$\begin{aligned}
\tilde{Q}(1,1) =& \tilde{Q}(X_i = 1)\tilde{Q}(X_j = 1|X_i = 1)\\
\ge& \tilde{Q}(X_i = 1)[\tilde{Q}(X_j = 1|X_i = -1) - \beta\alpha^{l_{ij}}]\\
\ge& \tilde{Q}(X_i = 1)[\tilde{Q}(X_j = 1|X_i = -1) - \beta\alpha^{h}].
\end{aligned}$$

Similarly, $\tilde{Q}(-1,-1) \geq \tilde{Q}(X_i=-1)[\tilde{Q}(X_j=-1|X_i=1)-\beta\alpha^h]$. Then,

$$\begin{aligned}
&\tilde{Q}(1,1)\tilde{Q}(-1,-1)\\
\geq&\tilde{Q}(X_i=1)\tilde{Q}(X_i=-1)[\tilde{Q}(X_j=1|X_i=-1)-\beta\alpha^h]\\
&[\tilde{Q}(X_j=-1|X_i=1)-\beta\alpha^h]\\
\geq&\tilde{Q}(1,-1)\tilde{Q}(-1,1)-2\beta\alpha^h.
\end{aligned}$$

Using the above inequality, we have the following lower bound on the P-test quantity.

$$\begin{aligned}
&\max_{x_i,x_j,x_j'}|P(x_i|x_j,x_S,x_T)-P(x_i|x_j',x_S,x_T)|\\
\geq&|Q(x_i=1|x_j=1)-Q(x_i=1|x_j=-1)|\\
=&\left|\frac{Q(x_i=1,x_j=1)}{Q(x_j=1)}-\frac{Q(x_i=1,x_j=-1)}{Q(x_j=-1)}\right|\\
=&\frac{1}{Q(x_j=1)Q(x_j=-1)}|Q(x_i=1,x_j=1)Q(x_i=-1,x_j=-1)\\
&-Q(x_i=1,x_j=-1)Q(x_i=-1,x_j=1)|\\
=&\frac{\left|e^{2J_{ji}}\tilde{Q}(1,1)\tilde{Q}(-1,-1)-e^{-2J_{ji}}\tilde{Q}(1,-1)\tilde{Q}(-1,1)\right|}{\left(e^{J_{ji}}\tilde{Q}(1,1)+e^{-J_{ji}}\tilde{Q}(-1,1)\right)\left(e^{-J_{ji}}\tilde{Q}(1,-1)+e^{J_{ji}}\tilde{Q}(-1,-1)\right)}\\
\geq&e^{-2J_{ij}}\left[(e^{2J_{ij}}-e^{-2J_{ij}})\tilde{Q}(1,-1)\tilde{Q}(-1,1)-2e^{2J_{ij}}\beta\alpha^h\right]\\
=&(1-e^{-4J_{ij}})\tilde{Q}(1,-1)\tilde{Q}(-1,1)-2\beta\alpha^h\\
\geq&(1-e^{-4J_{\min}})\tilde{Q}(1,-1)\tilde{Q}(-1,1)-2\beta\alpha^h.
\end{aligned}$$

Let $\check{Q}$ denote the joint probability when all the external field terms are removed from $\tilde{Q}$; i.e.,

$$\tilde{Q}(X_W)\propto\check{Q}(X_W)e^{h_W^TX_W}.$$

As there are at most $(D_1+D_2)d$ edges between R and W, we have $||h_W||_1\leq$

$(D_1 + D_2)dJ_{\max}$. Hence, for any subset $U \subset W$ and value x_U,

$$\begin{aligned}
\tilde{Q}(x_U) =& \frac{\tilde{Q}(x_U)}{\sum_{x'_U} \tilde{Q}(x'_U)} \\
=& \frac{\sum_{x_{W\setminus U}} \check{Q}(x_U, x_{W\setminus U}) e^{h_W^T x_W}}{\sum_{x'_U} \sum_{x'_{W\setminus U}} \check{Q}(x'_U, x'_{W\setminus U}) e^{h_W^T x'_W}} \\
\geq& \frac{\check{Q}(x_U) e^{-(D_1+D_2)dJ_{\max}}}{e^{(D_1+D_2)dJ_{\max}}} \\
=& e^{-2(D_1+D_2)dJ_{\max}} \check{Q}(x_U).
\end{aligned}$$

Moreover, $\check{Q}$ is zero-field by definition and again has the same correlation decay condition as P, hence

$$\begin{aligned}
\check{Q}(1,-1) + \check{Q}(1,1) =& \check{Q}(X_i = 1) = \frac{1}{2} \\
\frac{\check{Q}(1,-1)}{\check{Q}(1,1)} \geq& e^{-\beta\alpha^h},
\end{aligned}$$

which gives the lower bound $\check{Q}(1,-1) \geq \frac{1}{2(1+e^{\beta\alpha^h})}$. Therefore, we have

$$\tilde{Q}(1,-1) \geq \frac{e^{-2(D_1+D_2)dJ_{\max}}}{2(1+e^{\beta\alpha^h})}.$$

The same lower bound applies for $\tilde{Q}(-1,1)$. Hence,

$$\begin{aligned}
&\max_{x_i, x_j, x'_j} |P(x_i|x_j, x_S, x_T) - P(x_i|x'_j, x_S, x_T)| \\
\geq& \frac{(1 - e^{-4J_{\min}}) e^{-4(D_1+D_2)dJ_{\max}}}{4(1+e^{\beta\alpha^h})^2} - 2\beta\alpha^h \\
\geq& \frac{(1 - e^{-4J_{\min}}) e^{-4(D_1+D_2)dJ_{\max}}}{36} - 2\beta\alpha^h \\
\geq& \frac{(1 - e^{-4J_{\min}}) e^{-4(D_1+D_2)dJ_{\max}}}{36} - 2e^{4(D_1+D_2)dJ_{\max}}\beta\alpha^h \\
>& \epsilon.
\end{aligned}$$

The second inequality uses the fact that $e^{\beta\alpha^h} < 2$. The last inequality is by the choice of h.

Next consider $(i,j) \notin E$. By the choice of h, there exists $S \subset N_i, |S| \leq D_1$ such that, when the set S is removed from the graph, the distance from i to

j is no less than h. Let T set with $|T| \leq D_2$. As there is no edge between i, j, the joint probability Q and $\tilde{Q}$ are the same. Then $\forall x_S, x_T, x_i, x_j$,

$$\begin{aligned}
&|P(x_i|x_j, x_S, x_T) - P(x_i| - x_j, x_S, x_T)| \\
=&|\tilde{Q}(x_i|x_j) - \tilde{Q}(x_i| - x_j)| \\
=&\frac{|\tilde{Q}(x_i, x_j)\tilde{Q}(-x_i, -x_j) - \tilde{Q}(x_i, -x_j)\tilde{Q}(-x_i, x_j)|}{\tilde{Q}(x_j)\tilde{Q}(-x_j)}.
\end{aligned}$$

Similar as above, we have

$$\tilde{Q}(x_j) \geq e^{-2(D_1+D_2)dJ_{\max}}\check{Q}(x_j) = \frac{1}{2}e^{-2(D_1+D_2)dJ_{\max}}.$$

The same bound applies for $\tilde{Q}(-x_j)$. Therefore,

$$\begin{aligned}
&|P(x_i|x_j, x_S, x_T) - P(x_i| - x_j, x_S, x_T)| \\
\leq&4e^{4(D_1+D_2)dJ_{\max}}|\tilde{Q}(x_i, x_j)\tilde{Q}(-x_i, -x_j) - \tilde{Q}(x_i, -x_j)\tilde{Q}(-x_i, x_j)|.
\end{aligned}$$

By correlation decay and the fact $\beta\alpha^h < \ln 2 < 1$,

$$\begin{aligned}
&Q(x_i, x_j)Q(-x_i, -x_j) \\
=&Q(x_i|x_j)Q(x_j)Q(-x_i| - x_j)Q(-x_j) \\
\leq&(Q(x_i| - x_j) + \beta\alpha^h)Q(x_j)(Q(-x_i| - x_j) + \beta\alpha^h)Q(-x_j) \\
\leq&Q(x_i, -x_j)Q(-x_i, x_j) + 3\beta\alpha^h.
\end{aligned}$$

Similarly, we have $Q(x_i, x_j)Q(-x_i, -x_j) \geq Q(x_i, -x_j)Q(-x_i, x_j) - 2\beta\alpha^h$. Hence, by the choice of h,

$$|P(x_i|x_j, x_S, x_T) - P(x_i| - x_j, x_S, x_T)| \leq 12e^{4(D_1+D_2)dJ_{\max}}\beta\alpha^h \leq \frac{\epsilon}{4}.$$

□

Now we specialize this lemma for large girth graphs, in which there is at most one short path between non-neighbor nodes and no short non-direct path between neighboring nodes. Setting $D_1 = 1$ and $D_2 = 0$ in the theorem, we get Theorem 2.1. For the lower bound on the correlation between neighbor nodes, we set $D_1 = D_2 = 0$ in the theorem and get Lemma 2.5.

A.2 Ferromagnetic Ising Models

A.2.1 Proof of Corollary 2.4

By Proposition 2.2, we apply Definition 2.2 to X with $f(X) = X_i$ and $g(X) = X_j$, and get $\mathbb{E}[[]\, X_iX_j] \geq \mathbb{E}[[]\, X_i]\mathbb{E}[[]\, X_j]$. As there is no external field, $P(X_i = 1) = P(X_i = -1) = 0$ for any i and $P(X_i = x_i, X_j = x_j) = P(X_i = -x_i, X_j = -x_j)$ for any i, j. Therefore, $\mathbb{E}[[]\, X_i] = 0$ and

$$\begin{aligned}\mathbb{E}[[]\, X_iX_j] =& 4[P(X_i = 1, X_j = 1) - P(X_i = 1, X_j = -1)][P(X_i = 1, X_j = 1) \\ &+ P(X_i = 1, X_j = -1)].\end{aligned}$$

By the above inequality, noticing that $P(X_i = 1, X_j = 1) + P(X_i = 1, X_j = -1) = \frac{1}{2}$, we get the result.

A.2.2 Proof of Lemma 2.6

For any $i \in V, j \in N_i, S \subset V$, $Q, \tilde{Q}, \check{Q}$ are defined as in the proof of Lemma A.1. When X is ferromagnetic but with external field, as in Corollary 2.4, we can show that

$$\begin{aligned}&P(X_i = 1, X_j = 1)P(X_i = -1, X_j = -1) \\ \geq& P(X_i = 1, X_j = -1)P(X_i = -1, X_j = 1)\end{aligned}$$

for any i, j. Therefore, we have

$$\begin{aligned}&\max_{x_i, x_j, x_j'} |P(x_i|x_j, x_S) - P(x_i|x_j', x_S)| \\ \geq& e^{-2J_{ij}} \left| e^{2J_{ji}} \tilde{Q}(1,1)\tilde{Q}(-1,-1) - e^{-2J_{ij}} \tilde{Q}(1,-1)\tilde{Q}(-1,1) \right| \\ \geq& e^{-2J_{ij}}(e^{2J_{ij}} - e^{-2J_{ij}})\tilde{Q}(1,1)\tilde{Q}(-1,-1) \\ \geq& (1 - e^{-4J_{\min}})\tilde{Q}(1,1)\tilde{Q}(-1,-1).\end{aligned}$$

We note that $\check{Q}$ is zero field, so by Corollary 2.4 we get $\check{Q}(1,1) = \check{Q}(-1,-1) \geq \frac{1}{4}$. As shown in Lemma A.1,

$$\tilde{Q}(1,1) \geq e^{-2|N_S|J_{\max}} \check{Q}(1,1) \geq \frac{1}{4} e^{-2|N_S|J_{\max}}.$$

The same lower bound can be obtained for $\tilde{Q}(-1,-1)$. Plugging the lower bounds to the above inequality, we get the result.

A.3 Random Graphs

The proofs in this section are related to the techniques developed in [14, 4]. The key differences are in adapting the proofs for general Ising models, as opposed to ferromagnetic models. We point out similarities and differences as we proceed with the section.

A.3.1 Self-Avoiding-Walk Tree and Some Basic Results

This subsection introduces the notion of a self-avoiding-walk (SAW) tree, first introduced in [78], and presents some properties of a SAW tree. For an Ising model on a graph G, fix an ordering of all the nodes. We say dge (i,j) is larger (smaller resp.) than (i,l) with respect to node i if j comes after (before resp.) l in the ordering. The SAW tree rooted at node i is denoted as $T_{saw}(i;G)$. It is essentially the tree of self-avoiding walks originated from node i except that the terminal nodes closing a cycle are also included in the tree with a fixed value $+1$ or -1. In particular, a terminal node is fixed to $+1$ (resp. -1) if the closing edge of the cycle is larger (resp. smaller) than the starting edge with respect to the terminal node. Let A denote the set of all terminal nodes in $T_{saw}(i;G)$ and x_A denote the fixed configuration on A. For set $S \subset V$, let $U(S)$ denote the set of all non-terminal copies of nodes in S in $T_{saw}(i;G)$. Notice that there is a natural way to define conditioning on $T_{saw}(i;G)$ according to the conditioning on G; specifically, if node j in graph G is fixed to a certain value, the non-terminal copies of j in tree $T_{saw}(i;G)$ are fixed to the same value.

One important result is [79, Theorem 7], motivated by [78], says that the conditional probability of node i on graph G is the same as the corresponding conditional probability of node i on tree $T_{saw}(i;G)$, which is easier to deal with.

Proposition A.1. *Let S be a subset of V. $\forall x_i, x_S, P(x_i|x_S;G) = P(x_i|x_{U(S)}, x_A; T_{saw}(i;G))$.*

Next we list some basic results which will be used in later proofs. First we have the following lemma about the number of short paths between a pair of nodes from [14]. The second part of Theorem 2.2 is an immediate result of this lemma.

Lemma A.1. *[14] For all $i, j \in V$, the number of paths shorter than γ_p between nodes i, j is at most 2 almost always.*

Let $B(i, l; T_{saw}(i; G))$ be the set of nodes of distance l from i on the tree $T_{saw}(i; G)$. Recall that A is the set of terminal nodes in the tree. Let $\tilde{A}$ be the subset of A that are of distance at most γ_p from i. The size of $B(i, l; T_{saw}(i; G))$ and $\tilde{A}$ are upper bounded as follows.

Lemma A.2. *[22, Lemma 2.2] For $1 \leq l \leq a \log p$, where $0 < a < \frac{1}{2 \log c}$, we have*

$$\max_i |B(i, l; T_{saw}(i; G))| = O(c^l \log p), \text{ almost always.}$$

Lemma A.3. *$\forall i \in V, |\tilde{A}| \leq 1$ in $T_{saw}(i; G)$ almost always.*

Proof. Each terminal node in $\tilde{A}$ corresponds to a cycle connected to i with the total length of the cycle and the path to i at most γ_p. Let OLO_l denote the subgraph consists of two connected circles with total length l. This structure has $l - 1$ nodes and l edges. Let $H = \{OLO_l,\ l \leq 2\gamma_p\}$ and N_H denote the number of subgraphs containing an instance from H. Then it is equivalent to show that there is at most one such small cycle close to each node or $N_H = 0$ almost always.

$$\begin{aligned}\mathbb{E}\left[[]\, N_H\right] \leq & \sum_{l=1}^{2\gamma_p} \binom{p}{l-1} (l-1)!(l-1)^2 (\frac{c}{p})^l \leq O(\sum_{l=1}^{2\gamma_p} p^{-1} l^2 c^l) \\ = & O(p^{-1} \gamma_p^2 c^{2\gamma_p}) \leq O(p^{-\frac{1}{3}}) = o(1).\end{aligned}$$

So, $P(N_H \geq 1) = o(1)$. □

A.3.2 Correlation Decay in Random Graphs

This subsection is to prove the first part of Theorem 2.2 which characterizes the correlation decay property of a random graph.

First we state a correlation decay property for tree graphs. This result shows that having external fields only makes the correlation decay faster.

Lemma A.4. *Let P be a general Ising model with external fields on a tree T. Assume $|J_{ij}| \leq J_{\max}$. $\forall i, j \in T$,*

$$|P(x_i|x_j) - P(x_i|x_j')| \leq (\tanh J_{\max})^{d(i,j)}.$$

Proof. The basic idea in the proof is get an upper bound that does not depend on the external field. To do this, we proceed as in the proof of Lemma 4.1 in [80]. First, as noted in [80], without loss of generality, assume the tree is a line from i to j. Then, we prove the result by induction on the number of hops in the line.

1. $d(i,j) = 1$ or $j \in N_i$. The graph has only two nodes. We have

$$P(x_i|x_j) = \frac{e^{J_{ij}x_ix_j + h_ix_i}}{e^{J_{ij}x_j + h_i} + e^{-J_{ij}x_j - h_i}}.$$

   Hence,

$$\begin{aligned}|P(x_i|x_j) - P(x_i|x_j')| =& \frac{|e^{2J_{ij}} - e^{-2J_{ij}}|}{(e^{J_{ij}+h_i} + e^{-J_{ij}-h_i})(e^{-J_{ij}+h_i} + e^{J_{ij}-h_i})} \\ =& \frac{|e^{2J_{ij}} - e^{-2J_{ij}}|}{e^{2J_{ij}} + e^{-2J_{ij}} + e^{2h_i} + e^{-2h_i}}.\end{aligned}$$

   This function is even in both J_{ij} and h_i. Without loss of generality, assume $J_{ij} \geq 0, h_i \geq 0$. It is not hard to see that the RHS is maximized when $h_i = 0$. So

$$|P(x_i|x_j) - P(x_i|x_j')| \leq \tanh |J_{ij}| \leq \tanh J_{\max}.$$

   The inequality suggests that, when there is external field, the impact of one node on the other is reduced.

2. Assume the claim is true for $d(i,j) \leq k$. For $d(i,j) = k+1$, pick any l on the path from i to j, and note that X_i — X_l — X_j forms a Markov

chain. Moreover, $d(i,l) \leq k$ and $d(l,j) \leq k$.

$$
\begin{aligned}
&|P(x_i|x_j) - P(x_i|x'_j)| \\
=&|\sum_{x_l} P(x_i|x_l)P(x_l|x_j) - \sum_{x_l} P(x_i|x_l)P(x_l|x'_j)| \\
=&|P(x_i|x_l)(P(x_l|x_j) - P(x_l|x'_j)) + P(x_i|x'_l)(P(x'_l|x_j) - P(x'_l|x'_j))| \\
=&|(P(x_i|x_l) - P(x_i|x'_l))(P(x_l|x_j) - P(x_l|x'_j))| \\
\leq&(\tanh J_{\max})^{d(i,l)}(\tanh J_{\max})^{d(l,j)} = (\tanh J_{\max})^{d(i,j)}.
\end{aligned}
$$

The third equality follows by observing that $P(x_l|x_j) - P(x_l|x'_j) = -(P(x'_l|x_j) - P(x'_l|x'_j))$. The last inequality is by induction.

□

Writing the conditional probability on a graph as a conditional probability on the corresponding SAW tree, we can apply the above lemma and show the correlation decay property for random graphs.

Lemma A.5. *Let P be a general Ising model on a graph G. Fix $i \in V$. $\forall j \notin N_i$, let S be the set that separates the paths shorter than γ between i, j and $B = B(i, \gamma; T_{saw}(i; G))$, then $\forall x_i, x_j, x'_j, x_S$,*

$$|P(x_i|x_j, x_S) - P(x_i|x'_j, x_S)| \leq |B|(\tanh J_{\max})^{\gamma}.$$

Proof. Let Z be the subset of $U(j)$ on $T_{saw}(i; G)$ that is not separated by $U(S)$ from i. By the definition of S, Z is of distance at least γ from i. So

the γ-sphere B separates Z and i.

$$
\begin{aligned}
&|P(x_i|x_j, x_S) - P(x_i|x'_j, x_S)| \\
\overset{(a)}{=}&|P(x_i|x_{U(j)}, x_{U(S)}, x_A; T_{saw}(i;G)) - P(x_i|x'_{U(j)}, x_{U(S)}, x_A; T_{saw}(i;G))| \\
\overset{(b)}{=}&|P(x_i|x_Z, x_{U(S)}, x_A; T_{saw}(i;G)) - P(x_i|x'_Z, x_{U(S)}, x_A; T_{saw}(i;G))| \\
\overset{(c)}{=}&|\sum_{x_B} P(x_i|x_B, x_{U(S)}, x_A; T_{saw}(i;G)) P(x_B|x_Z, x_{U(S)}, x_A; T_{saw}(i;G)) \\
&- \sum_{x_B} P(x_i|x_B, x_{U(S)}, x_A; T_{saw}(i;G)) P(x_B|x'_Z, x_{U(S)}, x_A; T_{saw}(i;G))| \\
\leq& \max_{x_B} P(x_i|x_B, x_{U(S)}, x_A; T_{saw}(i;G)) - \min_{x_B} P(x_i|x_B, x_{U(S)}, x_A; T_{saw}(i;G)) \\
\overset{(d)}{=}&P(x_i|x_B^M, x_{U(S)}, x_A; T_{saw}(i;G)) - P(x_i|x_B^m, x_{U(S)}, x_A; T_{saw}(i;G)) \\
\overset{(e)}{\leq}&|B|(\tanh J_{\max})^\gamma.
\end{aligned}
$$

In the above, (a) follows from the property of SAW tree in Prop A.1. Step (b) is by the choice of S and the definition of Z. Step (c) uses the fact that Z is separated from i by B. In (d), x_B^M, x_B^m represent the maximizer and minimizer respectively. Step (e) is by telescoping the sign of x_B. Notice that the Hamming distance between x_B^M, x_B^m is at most $|B|$, and we can apply the above lemma to each pair as the conditioning terms differ only on one node. The above proof is similar to the proof of Lemma 3 in [14]. However, in going from step (c) to step (d) above, it is important to note that our proof holds for general Ising models, whereas the proof in [14] is specific to ferromagnetic Ising models. □

Proof of Theorem 2.2. As in [14], setting $\gamma = \gamma_p$ in the above lemma and noticing that

$$|B(i, \gamma_p; T_{saw}(i;G))| = O(c^{\gamma_p} \log p),$$

we get

$$
\begin{aligned}
&|P(x_i|x_j, x_S) - P(x_i|x'_j, x_S)| \\
\leq& O((c \tanh J_{\max})^{\gamma_p} \log p) = O(p^{-\frac{\log \alpha}{K \log c}} \log p) = o(p^{-\kappa}).
\end{aligned}
$$

□

A.3.3 Asymptotic Lower Bound on $P(x_i|x_R)$ When $|R| \leq 3$

This subsection is to prove that $P(x_i|x_R)$ is lower bounded by some constant when $|R| \leq 3$. This result comes in handy when proving the other two theorems. This result was conjectured to hold in [14] for ferromagnetic Ising models on the random graph $\mathcal{G}(p, \frac{c}{p})$ without a proof. Here we prove that it is also true for general Ising models on the random graph.

Lemma A.6. *$\forall i \in V, \forall R \subset V, |R| \leq 3$, there exists a constant C such that $\forall x_i, x_R, P(x_i|x_R) \geq C$ almost always.*

This basic idea is that the conditional probability $P(x_i|x_R)$ is equal to some conditional probability on a SAW tree, which in turn is viewed as some unconditional probability on the same tree with induced external fields. Then we apply a tree reduction to the SAW tree until only the root is left, and show that the induced external field on the root is bounded, which implies that the probability of the root taking $+1$ or -1 is bounded.

On a tree graph, when calculating a probability which involves no nodes in a subtree, we can reduce the subtree by simply summing (marginalizing) over all the nodes in it. This reduction produces an Ising model on the rest part of the tree with the same J_{ij} and h_i except for the root of the subtree, which would have an induced external field due to the reduction of the subtree. The probability we want to calculate remains unchanged on this new tree. Such induced external fields are bounded according to the following lemma.

Lemma A.7. *Consider a leaf node 2 and its parent node 1. The induced external field h_1' on node 1 due to summation over node 2 satisfies*

$$|h_1'| \leq |h_2| \tanh |J_{12}|.$$

We first prove an inequality which is used in the proof of the above lemma.

Lemma A.8. *$\forall x \geq 0, y \geq 0$,*

$$e^{2x \tanh y} \geq \frac{e^{x+y} + e^{-x-y}}{e^{x-y} + e^{-x+y}}.$$

Proof. Let $u = \tanh y \in [0, 1)$, then $y = \frac{1}{2} \ln \frac{1+u}{1-u}$. The required result is equivalent to showing that

$$e^{2xu}[(1+u)e^{-x} + (1-u)e^{x}] > (1+u)e^{x} + (1-u)e^{-x}.$$

Define

$$f_u(z) = (1+u)e^{uz} + (1-u)e^{(1+u)z} - (1+u)e^z - (1-u).$$

Clearly, $f_u(0) = 0$, and

$$f'_u(z) = (1+u)[ue^{uz} + (1-u)e^{(1+u)z} - e^z].$$

By the convexity of e^z, $ue^{uz} + (1-u)e^{(1+u)z} \geq e^z$. Hence, $f'_u(z) \geq 0$, which implies $f_u(z) \geq 0$. We finish the proof by noticing that the original inequality is equivalent to $f_u(2x) \geq 0$. □

Proof of Lemma A.7.

$$\sum_{x_2} e^{J_{12}x_1x_2 + h_2x_2} = e^{J_{12}x_1 + h_2} + e^{-J_{12}x_1 - h_2} \propto e^{h'_1x_1}.$$

Comparing the ratio of $x_1 = \pm 1$, we get

$$\frac{e^{J_{12}+h_2} + e^{-J_{12}-h_2}}{e^{-J_{12}+h_2} + e^{J_{12}-h_2}} = \frac{e^{h'_1}}{e^{-h'_1}} = e^{2h'_1}.$$

So

$$h'_1 = \frac{1}{2}\log\frac{e^{J_{12}+h_2} + e^{-J_{12}-h_2}}{e^{-J_{12}+h_2} + e^{J_{12}-h_2}} \leq |h_2|\tanh|J_{12}|.$$

The last inequality follows from Lemma A.8. □

It is easy to see that $|h'_1| \leq |h_2|\tanh|J_{\max}| < |h_2|$. By induction, we can bound the external field induced by the whole subtree.

Proof of Lemma A.6. First we have

$$\begin{aligned}
P(x_i|x_R) =& P(x_i|x_{U(R)}, x_A; T_{saw}(i;G)) \\
=& \sum_{x_B} P(x_i|x_B, x_{\tilde{U}(R)}, x_{\tilde{A}}; T_{saw}(i;G)) P(x_B|x_{U(R)}, x_A; T_{saw}(i;G)) \\
\geq& \min_{x_B} P(x_i|x_B, x_{\tilde{U}(R)}, x_{\tilde{A}}; T_{saw}(i;G)) \\
=& P(x_i|x_B^m, x_{\tilde{U}(R)}, x_{\tilde{A}}; T_{saw}(i;G)) \triangleq Q(x_i),
\end{aligned}$$

where Q is the probability on the tree with external fields induced by x_B^m,

$x_{\tilde{U}(R)}$ and $x_{\tilde{A}}$. We only need to consider the external fields on the parent nodes of $B, \tilde{U}(R), \tilde{A}$ as the conditional probability is on a tree. The nodes affected by B are all γ_p away from i and the total number of them is no larger than $|B|$, which is bounded by Lemma A.2. The number of nodes affected by $\tilde{U}(R), \tilde{A}$ is no larger than $|\tilde{U}(R)| + |\tilde{A}|$. By Lemma A.1 and Lemma A.3, $|\tilde{U}(R)| \leq 2|R|$ and $|\tilde{A}| \leq 1$ almost always. Applying the reduction technique to the tree until a single root node i, by Lemma A.7, we bound the induced external field on i as

$$\begin{aligned}|h_i| \leq & [(\tanh J_{\max})^{\gamma_n}|B| + (|\tilde{U}(R)| + |\tilde{A}|)]J_{\max} \\ \leq & O((c \tanh J_{\max})^{\gamma_n} \log n + 2|R| + 1) \\ \leq & O(n^{-\kappa} + 7) = O(1).\end{aligned}$$

So,

$$Q(x_i) = \frac{e^{h_i x_i}}{e^{h_i x_i} + e^{-h_i x_i}} \geq \Omega(e^{-2|h_i|}) = \Omega(1).$$

When p is large enough, there exists some constant C such that $P(x_i|x_R) \geq C$. □

A.3.4 Proof of Theorem 2.3

Let S be the set that separates all the paths shorter than γ_p between nodes i, j with size $|S| \leq 3$. It is straightforward to show that $I(X_i; X_j|X_S) = o(p^{-2\kappa})$ in a manner similar to [14, Lemma 5]. The only difference is that the correlation decay property in Theorem 2.2 takes a different form, which is easier to apply, therefore the proof there needs to be modified accordingly. We also note that the constant C in Lemma A.6 is referred to as $f_{\min}(S)$ in [14]. The details are omitted here.

A.3.5 Proof of Theorem 2.4

When j is a neighbor of i, conditioned on the approximate separator T, there is one copy of j which is a child of the root i in the SAW tree and is the only copy that within γ_p from i. In Theorem 2.4, we show that the effect

of conditioning on T is bounded and this copy of j has a nontrivial impact on i. With a little abuse of notation, we use j to denote this copy of j in $T_{saw}(i;G)$. W.l.o.g assume $J_{ij} > 0$. As in Lemma A.5,

$$
\begin{aligned}
&\max_{x_i,x_j} |P(x_i|x_j, x_T) - P(x_i|x'_j, x_T)| \\
=&\max_{x_i,x_j} |P(x_i|x_{U(j)}, x_{U(T)}, x_A; T_{saw}(i;G)) - P(x_i|x'_{U(j)}, x_{U(T)}, x_A; T_{saw}(i;G))| \\
=&\max_{x_i,x_j} |P(x_i|x_Z, x_{U(T)}, x_A; T_{saw}(i;G)) - P(x_i|x'_Z, x_{U(T)}, x_A; T_{saw}(i;G))| \\
=&\max_{x_i,x_j} |\sum_{x_B} P(x_i|x_j, x_B, x_{\tilde{U}(T)}, x_{\tilde{A}}; T_{saw}(i;G)) P(x_B|x_Z, x_{U(T)}, x_A; T_{saw}(i;G)) \\
&- \sum_{x_B} P(x_i|x_B, x_{\tilde{U}(T)}, x_{\tilde{A}}; T_{saw}(i;G)) P(x_B|x'_Z, x_{U(T)}, x_A; T_{saw}(i;G))| \\
\geq&\min_{x_B} P(x_i = +|x_j = +, x_B, x_{\tilde{U}(T)}, x_{\tilde{A}}; T_{saw}(i;G)) \\
&- \max_{x_B} P(x_i = +|x_j = -, x_B, x_{\tilde{U}(T)}, x_{\tilde{A}}; T_{saw}(i;G)) \\
=&P(x_i = +1|x_j = +1, x_B^m, x_{\tilde{U}(T)}, x_{\tilde{A}}; T_{saw}(i;G)) \\
&- P(x_i = +1|x_j = -1, x_B^M, x_{\tilde{U}(T)}, x_{\tilde{A}}; T_{saw}(i;G)) \\
=&P(x_i = +1|x_j = +1, x_B^m, x_{\tilde{U}(T)}, x_{\tilde{A}}; T_{saw}(i;G)) \\
&- P(x_i = +1|x_j = -1, x_B^m, x_{\tilde{U}(T)}, x_{\tilde{A}}; T_{saw}(i;G)) \\
&+ P(x_i = +1|x_j = -1, x_B^m, x_{\tilde{U}(T)}, x_{\tilde{A}}; T_{saw}(i;G)) \\
&- P(x_i = +1|x_j = -1, x_B^M, x_{\tilde{U}(T)}, x_{\tilde{A}}; T_{saw}(i;G)) \\
\geq&Q(x_i = +1|x_j = +1) - Q(x_i = +1|x_j = -1) - |B|(\tanh J_{\max})^{\gamma_n},
\end{aligned}
$$

where Q is the probability measure on the reduced graph with only nodes i, j. We have

$$
\begin{aligned}
&Q(x_i = +1|x_j = +1) - Q(x_i = +1|x_j = -1) \\
=&\frac{e^{2J_{ij}} - e^{-2J_{ij}}}{e^{2J_{ij}} + e^{-2J_{ij}} + e^{2h_i} + e^{-2h_i}} \\
\geq&\frac{e^{2J_{\min}} - e^{-2J_{\min}}}{e^{2J_{\min}} + e^{-2J_{\min}} + e^{2h_i} + e^{-2h_i}} = \Omega(e^{-2|h_i|}).
\end{aligned}
$$

The external fields in Q are induced by the conditioning on $B, \tilde{U}(T), \tilde{A}$. As in the proof of Lemma A.6, we have $|h_i| \leq O(1)$, so $Q(x_i = +|x_j = +) - Q(x_i =$

$+|x_j = -) = \Omega(1)$. Hence,

$$\max_{x_i, x_j} |P(x_i|x_j, x_S) - P(x_i|x'_j, x_S)| \geq \Omega(1) - O(p^{-\kappa}) = \Omega(1).$$

Using this result, the lower bound $I(X_i; X_j|X_T) = \Omega(1)$ simply follows from the proof of [14, Lemma 7]. Again we note that the constant C in Lemma A.6 is referred to as $f_{\min}(T)$ in [14]. The details are omitted here.

A.3.6 Proof of Theorem 2.6

The proof of the theorem needs the following lemma.

Lemma A.9. *Assume X is a ferromagnetic Ising model (possibly with external fields). $\forall i \in V, \forall S \subset V \setminus i$,*

$$P(x_i = +1|x_S = +1) \geq P(x_i = +1|x_S = -1).$$

Proof. For any node $j \in S$, let probability $\tilde{P}(x_i, x_j) = P(x_i, x_j|x_{S\setminus j})$. The probability $\tilde{P}$ is still ferromagnetic and hence is associated. Then we have

$$\begin{aligned}&\tilde{P}(x_i = +1, x_j = +1)\tilde{P}(x_i = -1, x_j = -1)\\ \geq&\tilde{P}(x_i = +1, x_j = -1)\tilde{P}(x_i = -1, x_j = +1).\end{aligned}$$

After some algebraic manipulation, we get

$$\tilde{P}(x_i = +1|x_j = +1) \geq \tilde{P}(x_i = +1|x_j = -1).$$

This is equivalent saying that

$$P(x_i = +1|x_j = +1, x_{S\setminus j} = +1) \geq P(x_i = +1|x_j = -1, x_{S\setminus j} = +1).$$

So flipping one node from $+1$ to -1 reduces the conditional probability regardless the what value the rest of the nodes take. Continuing this process till we flip all the nodes in S, we get the result

$$P(x_i = +1|x_S = +1) \geq P(x_i = +1|x_S = -1).$$

□

Proof of Theorem 2.6. For $(i,j) \in E$, assume $J_{ij} > 0$. Following the proof of Theorem 2.4,

$$\begin{aligned}
&\max_{x_i,x_j} |P(x_i|x_j, x_S) - P(x_i|x'_j, x_S)| \\
=&\max_{x_i,x_j} |P(x_i|x_{U(j)}, x_{U(S)}, x_A; T_{saw}(i;G)) - P(x_i|x'_{U(j)}, x_{U(S)}, x_A; T_{saw}(i;G))| \\
\geq& P(x_i = +1|x_{\tilde{U}(j)} = +1, x_B^m, x_{\tilde{U}(S)}, x_{\tilde{A}}; T_{saw}(i;G)) \\
&- P(x_i = +1|x_{\tilde{U}(j)} = -1, x_B^M, x_{\tilde{U}(S)}, x_{\tilde{A}}; T_{saw}(i;G)).
\end{aligned}$$

The only difference here is that we might have more than one copy of j in $\tilde{U}(j)$. Let $Z = \tilde{U}(j) \setminus j$. By the above lemma, we have

$$\begin{aligned}
&\max_{x_i,x_j} |P(x_i|x_j, x_S) - P(x_i|x'_j, x_S)| \\
\geq& P(x_i = +1|x_j = +1, x_Z = +1, x_B^m, x_{\tilde{U}(S)}, x_{\tilde{A}}; T_{saw}(i;G)) \\
&- P(x_i = +1|x_j = -1, x_Z = +1, x_B^m, x_{\tilde{U}(S)}, x_{\tilde{A}}; T_{saw}(i;G)) \\
&+ P(x_i = +1|x_j = -1, x_Z = -1, x_B^m, x_{\tilde{U}(S)}, x_{\tilde{A}}; T_{saw}(i;G)) \\
&- P(x_i = +1|x_j = -1, x_Z = -1, x_B^M, x_{\tilde{U}(S)}, x_{\tilde{A}}; T_{saw}(i;G)) \\
\geq& Q(x_i = +1|x_j = +1) - Q(x_i = +1|x_j = -1) - |B|(\tanh J_{\max})^{\gamma_n}.
\end{aligned}$$

As the size of Z is only a constant, by the same reasoning, we finish the theorem. □

A.4 Concentration

Before proving the concentration results in Lemma 2.3, we first present the following lemma which upper bounds the difference between the entropies of two distributions with their l_1-distance. Let P and Q be two probability mass functions on a discrete, finite set $\mathcal{X}$, and $H(P)$ and $H(Q)$ be their entropies respectively. The l_1 distance between P and Q is defined as $||P - Q||_1 = \sum_{x \in \mathcal{X}} |P(x) - Q(x)|$.

Lemma A.10. *[81, Theorem 17.3.3] If $||P-Q||_1 \leq \frac{1}{2}$, then $|H(P)-H(Q)| \leq -||P - Q||_1 \log \frac{||P-Q||_1}{|\mathcal{X}|}$. When $||P - Q||_1 \leq \frac{1}{e}$, the RHS is increasing in $||P - Q||_1$.*

Proof of Lemma 2.3. By definition, $\forall S \subset V$ and $\forall x_S$, $|1_{\{X_S^{(i)} = x_S\}} - P(x_s)| \leq 1$ and

$$\hat{P}(x_S) = \frac{1}{n} \sum_{i=1}^{n} 1_{\{X_S^{(i)} = x_S\}}.$$

By the Hoeffding inequality,

$$\begin{aligned} & P\left(|\hat{P}(x_S) - P(x_S)| \geq \gamma\right) \\ =& P\left(\left|\sum_{i=1}^{n} 1_{\{X_S^{(i)} = x_S\}} - nP(x_S)\right| \geq n\gamma\right) \leq 2e^{-\frac{n^2\gamma^2}{2n}} \leq 2e^{-\frac{n\gamma^2}{2}}. \end{aligned}$$

1. By the union bound, we have

$$\begin{aligned} & P\left(\exists S \subset V, |S| \leq 2, \exists x_S, |\hat{P}(x_S) - P(x_S)| \geq \gamma\right) \\ <& p^2|\mathcal{X}|^2 2e^{-\frac{n\gamma^2}{2}} = 2e^{-\frac{n\gamma^2}{2} + 2\log p|\mathcal{X}|}. \end{aligned}$$

   For our choice of n, $\forall i, j \in V, \forall x_i, x_j$,

$$|\hat{P}(x_i, x_j) - P(x_i, x_j)| < \gamma, |\hat{P}(x_i) - P(x_i)| < \gamma,$$

   with probability $1 - \frac{c_1}{p^\alpha}$ for some constant c_1, which gives $\hat{P}(x_j) > P(x_j) - \gamma \geq \frac{1}{2} - \gamma \geq \frac{1}{4}$ as $\gamma < \frac{1}{4}$. Hence,

$$\begin{aligned} & |\hat{P}(x_i|x_j) - P(x_i|x_j)| \\ =& \frac{|\hat{P}(x_i, x_j)P(x_j) - P(x_i, x_j)\hat{P}(x_j)|}{P(x_j)\hat{P}(x_j)} \\ \leq& \frac{\hat{P}(x_i, x_j)|P(x_j) - P(x_j)|}{P(x_j)\hat{P}(x_j)} + \frac{\hat{P}(x_j)|\hat{P}(x_i, x_j) - P(x_i, x_j)|}{P(x_j)\hat{P}(x_j)} \\ \leq& \frac{2\gamma}{\frac{1}{2}} = 4\gamma. \end{aligned}$$

2. By the union bound, we have

$$\begin{aligned}
&P\left(\begin{array}{c}\exists i\in V,\exists S\subset L_i,|S|\le D_1+D_2+1,\exists x_S,\\ |\hat{P}(x_S)-P(x_S)|\ge\gamma,|\hat{P}(x_i,x_S)-P(x_i,x_S)|\ge\gamma\end{array}\right)\\
<&2pL^{D_1+D_2+1}|\mathcal{X}|^{D_1+D_2+2}2e^{-\frac{n\gamma^2}{2}}\\
<&4e^{-\frac{n\gamma^2}{2}+\log p+(D_1+D_2+1)\log L+(D_1+D_2+2)\log|\mathcal{X}|}.
\end{aligned}$$

For our choice of n, $\forall i\in V,\forall j\in L_i,\forall S\subset L_i,|S|\le D_1+D_2,\forall x_i,x_j,x_S$,

$$|\hat{P}(x_i,x_j,x_S)-P(x_i,x_j,x_S)|\le\gamma,\ |\hat{P}(x_j,x_S)-P(x_j,x_S)|\le\gamma,$$

with probability $1-\frac{c_2}{p^\alpha}$ for some constant c_2, which gives $\hat{P}(x_j,x_S)>P(x_j,x_S)-\gamma\ge\frac{\delta}{2}$ as $\gamma<\frac{\delta}{2}$. Hence,

$$\begin{aligned}
&|\hat{P}(x_i|x_j,x_S)-P(x_i|x_j,x_S)|\\
=&\frac{|\hat{P}(x_i,x_j,x_S)P(x_j,x_S)-P(x_i,x_j,x_S)\hat{P}(x_j,x_S)|}{P(x_j,x_S)\hat{P}(x_j,x_S)}\\
\le&\frac{\hat{P}(x_i,x_j,x_S)|P(x_j,x_S)-P(x_j,x_S)|}{P(x_j,x_S)\hat{P}(x_j,x_S)}\\
&+\frac{\hat{P}(x_j,x_S)|\hat{P}(x_i,x_j,x_S)-P(x_i,x_j,x_S)|}{P(x_j,x_S)\hat{P}(x_j,x_S)}\\
\le&\frac{2\gamma}{\delta}.
\end{aligned}$$

3. As in the previous case, for our choice of n, $\forall i,j\in V,\forall S\subset L_i,|S|\le D_1+D_2,\forall x_i,x_j,x_S$,

$$\begin{aligned}
|\hat{P}(x_i,x_j,x_S)-P(x_i,x_j,x_S)|\le&\gamma,\\
|\hat{P}(x_j,x_S)-P(x_j,x_S)|\le&\gamma,\\
|\hat{P}(x_S)-P(x_S)|\le&\gamma
\end{aligned}$$

with probability $1-\frac{c_3}{p^\alpha}$ for some constant c_3. So we get

$$||\hat{P}(X_i,X_j,X_S)-P(X_i,X_j,X_S)||_1\le|\mathcal{X}|^{D_1+D_2+2}\gamma\le\frac{1}{2}.$$

By Lemma A.10,

$$
\begin{aligned}
&|\hat{H}(X_i, X_j, X_S) - H(X_i, X_j, X_S)| \\
\leq &- ||\hat{P}(X_i, X_j, X_S) - P(X_i, X_j, X_S)||_1 \\
&\log \frac{||\hat{P}(X_i, X_j, X_S) - P(X_i, X_j, X_S)||_1}{|\mathcal{X}|^{D_1+D_2+2}} \\
\leq &- |\mathcal{X}|^{D_1+D_2+2}\gamma \log \gamma = -2|\mathcal{X}|^{D_1+D_2+2}\gamma \log \sqrt{\gamma} \\
\leq &2|\mathcal{X}|^{D_1+D_2+2}\sqrt{\gamma}.
\end{aligned}
$$

The last inequality used the fact that $0 < -\sqrt{\gamma}\log\sqrt{\gamma} < 1$ for $0 < \gamma < 1$. Similarly, we have the same upper bound for $|\hat{H}(X_i, X_S) - H(X_i, X_S)|, |\hat{H}(X_j, X_S) - H(X_j, X_S)|$ and $|\hat{H}(X_S) - H(X_S)|$. We finish the proof by noticing that

$$I(X_i; X_j|X_S) = H(X_i, X_S) + H(X_j, X_S) - H(X_i, X_j, X_S) - H(X_S).$$

□

APPENDIX B

PROOFS IN CHAPTER 3

B.1 Proof of Theorem 3.1

Without loss of generality, suppose users $1, 3, \ldots, 2K-1$ are in cluster 1 and users $2, 4, \ldots, 2K$ are in cluster 2. We construct a block-constant matrix with the same item cluster structure as R but a different user cluster structure. In particular, under $\tilde{R}$, user 1 forms a new cluster with users $2i, i = 2, \ldots, K$ and user 2 forms a new cluster with users $2i-1, i = 2, \ldots, K$.

Let i-th row of $\tilde{R}$ be identical to the i-th row of R for all $i > 2K$. Consider all items j in item cluster l. If the ratings of user 1 to items in item cluster l are all erased, then let $\tilde{R}_{1j} = R_{2j}$ and $\tilde{R}_{ij} = R_{2j}$ for $i = 4, 6, \ldots, 2K$; otherwise let $\tilde{R}_{1j} = R_{1j}$ and $\tilde{R}_{ij} = R_{1j}$ for $i = 4, 6, \ldots, 2K$. If the ratings of user 2 to items in item cluster l are all erased, then let $\tilde{R}_{2j} = R_{1j}$ and $\tilde{R}_{ij} = R_{1j}$ for $i = 3, 5, \ldots, 2K-1$; otherwise let $\tilde{R}_{2j} = R_{2j}$ and $\tilde{R}_{ij} = R_{2j}$ for $i = 3, 5, \ldots, 2K-1$. From the above procedure, it follows that the first row of $\tilde{R}$ is identical to the $(2i)$-th row of $\tilde{R}$ for all $i = 2, \ldots, K$, and the second row of $\tilde{R}$ is identical the $(2i-1)$-th row of $\tilde{R}$ for all $i = 2, \ldots, K$.

We show that $\tilde{R}$ agrees with $\widehat{R}$ on all non-erased entries. We say that item cluster l is conflicting between user 1 and user cluster 2 if (1) user cluster 1 and 2 have different block rating on item cluster l; and (2) the ratings of user 1 to items in item cluster l are not all erased; and (3) the block corresponding to user cluster 2 and item cluster l is not totally erased. Therefore, the probability that item cluster l is conflicting between user 1 and user cluster 2 equals to $\frac{1}{2}(1-\epsilon^{K^2})(1-\epsilon^K)$. By the union bound,

$$\begin{aligned}&\mathbb{P}\{\exists\text{conflicting item cluster between user 1 and cluster 2}\}\\&\leq \frac{r}{2}(1-\epsilon^{K^2})(1-\epsilon^K) \leq \frac{r}{2}K^3(1-\epsilon)^2 \leq \delta/2,\end{aligned}$$

where the third inequality follows because $(1-x)^a \geq 1-ax$ for $a \geq 1$ and $x \geq 0$ and the last inequality follows from the assumption. Similarly, the probability that there exists a conflicting item cluster between user 2 and cluster 1 is also upper bounded by $\delta/2$. Hence, with probability at least $1-\delta$, there is no conflicting item cluster between user 1 and cluster 2 as well as between user 2 and cluster 1, and thus $\tilde{R}$ agrees with $\widehat{R}$ on all non-erased entries.

B.2 Proof of Theorem 3.2

Consider a genie-aided scenario where the set of flipped entries is revealed as side information, which is equivalent to saying that we are in the noiseless setting with $p = 0$. Then the true partition corresponding to the true user cluster structure has zero disagreement. Suppose that users $1, 3, \ldots, 2K-1$ are in true cluster 1 and users $2, 4, \ldots, 2K$ are in true cluster 2. We construct a new partition different from the true partition by swapping user 1 and user 2. In particular, under the new partition, user 1 forms a new cluster $\widehat{C}_2$ with users $2i, i = 2, \ldots, K$, user 2 forms a new cluster $\widehat{C}_1$ with users $2i-1, i = 2, \ldots, K$. It suffices to show that for $k = 1, 2$, any two users in $\widehat{C}_k$ has zero disagreement with probability at least $3/4$, in which case the new partition has zero agreement and Algorithm 3 cannot distinguish between the true partition and the new one.

For $k = 1, 2$, we lower bound the probability that any two users in $\widehat{C}_k$ has zero disagreement.

$$\begin{aligned}
&\mathbb{P}(\text{Any two users in } \widehat{C}_k \text{ have zero disagreement}) \\
=&1 - \mathbb{P}(\text{total number of disagreements in } \widehat{C}_k \geq 1) \\
\geq&1 - \mathbb{E}[\text{total number of disagreements in } \widehat{C}_k] \\
\geq&1 - \frac{1}{2} nK(1-\epsilon)^2 \geq 7/8.
\end{aligned}$$

By union bound, the probability that for $k = 1, 2$, any two users in $\widehat{C}_k$ have zero disagreement is at least $3/4$.

B.3 Proof of Theorem 3.3

Consider a compatibility graph with n vertices representing users. Two vertices i, i' are connected if users i, i' have zero disagreement, i.e., $D_{ii'} = 0$. In the noiseless setting, each user cluster forms a clique of size K in the compatibility graph. We call a clique of size K in the compatibility graph a bad clique if it is formed by users from more than one cluster. Then to prove the theorem, it suffices to show that there is no bad clique a.a.s. Since the probability that bad cliques exist increases in ϵ, without loss of generality, we assume $K(1-\epsilon) < 1$.

Recall that B_{kl} is $+1$ or -1 with equal probability. Define $S_k = \{l : B_{kl} = +1\}$ for $k = 1, \ldots, r$. As $r \to \infty$, by Chernoff bound, we get that a.a.s., for any $k_1 \neq k_2$

$$|S_{k_1} \Delta S_{k_2}| \triangleq |\{l : B_{k_1 l} \neq B_{k_2 l}\}| \geq \frac{r}{4}. \tag{B.1}$$

Assume this condition holds throughout the proof.

Fix a set of K users that consists of users from t different clusters. Without loss of generality, assume these users are from cluster $1, \ldots, t$. Let n_k denote the number of users from the cluster k and define $n_{\max} = \max_k n_k$. By definition, $2 \leq t \leq t_{\max} \triangleq \min\{r, K\}$, $n_{\max} < K$ and $\sum_{k=1}^{t} n_k = K$. For any item j in cluster l, among the K ratings given by these users, there are $\sum_{k=1}^{t} n_k \mathbb{I}_{\{l \in S_k\}}$ ratings being $+1$ and $\sum_{k=1}^{t} n_k \mathbb{I}_{\{l \notin S_k\}}$ ratings being -1. Let E_j denote the event that the observed ratings for item j by these K users are the same. Then,

$$\begin{aligned}
\mathbb{P}[E_j] =& 1 - \left(1 - \epsilon^{\sum_{k=1}^{t} n_k \mathbb{I}_{\{l \in S_k\}}}\right)\left(1 - \epsilon^{\sum_{k=1}^{t} n_k \mathbb{I}_{\{l \notin S_k\}}}\right) \\
\leq& \exp\left(-(1 - \epsilon^{\sum_{k=1}^{t} n_k \mathbb{I}_{\{l \in S_k\}}})(1 - \epsilon^{\sum_{k=1}^{t} n_k \mathbb{I}_{\{l \notin S_k\}}})\right) \\
\leq& \exp\Big(-\frac{1}{4}(1-\epsilon)^2 \sum_{k=1}^{t} n_k \mathbb{I}_{\{l \in S_k\}} \sum_{k=1}^{t} n_k \mathbb{I}_{\{l \notin S_k\}}\Big).
\end{aligned}$$

Let $p_{n_1 \ldots n_t}$ be the probability that K users, out of which n_k are from cluster k, form a bad clique. Because $\{E_j\}$ are independent and there are K items

in each item cluster,

$$
\begin{aligned}
&p_{n_1 \dots n_t} \\
&= \prod_{j=1}^{n} \mathbb{P}[E_j] \\
&\leq \exp\Big(-\frac{1}{4} K(1-\epsilon)^2 \sum_{l=1}^{r} (\sum_{k=1}^{t} n_k \mathbb{I}_{\{l \in S_k\}} \sum_{k=1}^{t} n_k \mathbb{I}_{\{l \notin S_k\}}) \Big) \\
&= \exp\Big(-\frac{1}{4} K(1-\epsilon)^2 \sum_{1 \leq k_1 < k_2 \leq t} n_{k_1} n_{k_2} |S_{k_1} \Delta S_{k_2}| \Big) \\
&\leq \exp\Big(-C_1 n (1-\epsilon)^2 \sum_{k=1}^{t} n_k (K - n_k) \Big) \qquad \text{(B.2)}
\end{aligned}
$$

for some constant C_1. For a large enough constant C in the assumption in the statement of the theorem, we have

$$
\begin{aligned}
K \exp(-C_1 n (1-\epsilon)^2 (K - n_k)) \leq & n^{-3}, \quad n_k \leq \frac{K}{2}, \qquad \text{(B.3)} \\
K \exp(-C_1 n (1-\epsilon)^2 n_k) \leq & n^{-3}, \quad n_k > \frac{K}{2}. \qquad \text{(B.4)}
\end{aligned}
$$

Below we show that the probability of bad cliques existing goes to zero. By the Markov inequality and linearity of expectation,

$$
\begin{aligned}
&\mathbb{P}[\text{Number of bad cliques} \geq 1] \\
&\leq \mathbb{E}[\text{Number of bad cliques}] \\
&= \sum_{t=2}^{t_{\max}} \binom{r}{t} \sum_{n_1 + \dots + n_t = K} \binom{K}{n_1} \cdots \binom{K}{n_t} p_{n_1 \dots n_t} \\
&= \sum_{t=2}^{t_{\max}} \binom{r}{t} \sum_{n_1 + \dots + n_t = K} \binom{K}{n_1} \cdots \binom{K}{n_t} p_{n_1 \dots n_t} \\
&\quad \big[\mathbb{I}_{\{n_{\max} \leq K/2\}} + \mathbb{I}_{\{n_{\max} > K/2\}} \big]. \qquad \text{(B.5)}
\end{aligned}
$$

The first term in (B.5) is bounded as

$$\begin{aligned}
&\sum_{t=2}^{t_{\max}} \binom{r}{t} \sum_{\substack{n_1+\cdots+n_t=K \\ n_{\max}\le K/2}} \binom{K}{n_1}\cdots\binom{K}{n_t} p_{n_1\ldots n_t} \\
&\le \sum_{t=2}^{t_{\max}} r^t \sum_{\substack{n_1+\cdots+n_t=K \\ n_{\max}\le K/2}} \prod_{k=1}^{t} (Ke^{-C_1(1-\epsilon)^2(K-n_k)})^{n_k} \\
&\le \sum_{t=2}^{t_{\max}} r^t K^t n^{-3K} = o(1), \qquad \text{(B.6)}
\end{aligned}$$

where the first inequality follows from the fact that $\binom{K}{n_k} \le K^{n_k}$ and (B.2), and the second inequality follows from (B.3). The second term in (B.5) is bounded as

$$\begin{aligned}
&\sum_{t=2}^{t_{\max}} \binom{r}{t} \sum_{\substack{n_1+\cdots+n_t=K \\ n_{\max}> K/2}} \binom{K}{n_1}\cdots\binom{K}{n_t} p_{n_1\ldots n_t} \\
&\le \sum_{t=2}^{t_{\max}} r^t \sum_{\substack{n_1+\cdots+n_t=K \\ n_{\max}> K/2}} (Ke^{-C_1 n(1-\epsilon)^2 n_{\max}})^{K-n_{\max}} \\
&\quad \prod_{k:n_k<n_{\max}} (Ke^{-C_1(1-\epsilon)^2(K-n_k)})^{n_k} \\
&\le \sum_{t=2}^{t_{\max}} \mathbb{I}_{\{t\le K-n_{\max}+1\}} r^t K^t n^{-6(K-n_{\max})} = o(1), \qquad \text{(B.7)}
\end{aligned}$$

where the first inequality follows from the fact that $\binom{K}{n_k} \le \min\{K^{n_k}, K^{K-n_k}\}$ and (B.2), and the second inequality follows from (B.3) and (B.4) and the fact that $t \le K - n_{\max} + 1$. Therefore we conclude that

$$\mathbb{P}[\text{Number of bad cliques} \ge 1] = o(1).$$

B.4 Proof of Theorem 3.4

We first introduce some notations. Let $u_{C,k}$ be the normalized characteristic vector of user cluster k, i.e., $u_{C,k}(i) = 1/\sqrt{K}$ if user i is in cluster k and $u_{C,k}(i) = 0$ otherwise. Thus, $||u_{C,k}||_2 = 1$. Let $U_C = [u_{C,1}, \ldots, u_{C,r}]$.

Similarly, let $v_{C,l}$ be the normalized characteristic vector of item cluster l and $V_C = [v_{C,1}, \ldots, v_{C,r}]$. It is not hard to see that the rating matrix R can be written as $R = KU_C B V_C^\top$. Denote the SVD of the block rating matrix B by $B = U_B \Sigma_B V_B^\top$, then the SVD of R is simply $R = UK\Sigma_B V^\top$, where $U = U_C U_B$ and $V = V_C V_B$. When $r \to \infty$, B has full rank almost surely [82]. We will assume B is full rank in the following proofs, which implies that $U_B U_B^\top = I$ and $V_B V_B^\top = I$. Note that $UU^\top = U_C U_C^\top, VV^\top = V_C V_C^\top$ and $UV^\top = U_C U_B V_B^\top V_C^\top$.

We now briefly recall the subgradient of the nuclear norm [42]. Define T to be the subspace spanned by all matrices of the form $UA^\top$ or $AV^\top$ for any $A \in \mathbb{R}^{n \times r}$. The orthogonal projection of any matrix $M \in \mathbb{R}^{n \times n}$ onto the space T is given by $\mathcal{P}_T(M) = UU^\top M + MVV^\top - UU^\top MVV^\top$. The projection of M onto the complement space $T^\perp$ is $\mathcal{P}_{T^\perp}(M) = M - \mathcal{P}_T(M)$. Then $M \in \mathbb{R}^{n \times n}$ is a subgradient of $||X||_*$ at $X = R$ if and only if $\mathcal{P}_T(M) = UV^\top$ and $||\mathcal{P}_{T^\perp}(M)|| \leq 1$.

Proof of Lemma 3.1. Assume user i is from user cluster k and item j is in item cluster l, then

$$|(UV^\top)_{ij}| = |(U_B V_B^\top)_{kl}|/K \leq 1/K = r/n,$$

where the inequality follows from the Cauchy-Schwartz inequality. By definition $\mu \leq \sqrt{r}$. □

Next we establish the concentration property of $\widehat{R}$. By definition the conditional expectation of $\widehat{R}$ is given by

$$\mathbb{E}[\widehat{R}|R] = (1-\epsilon)(1-2p)R := \bar{R}.$$

Furthermore, the variance is given by

$$\mathrm{Var}[\widehat{R}_{ij}|R] = (1-\epsilon) - (1-\epsilon)^2(1-2p)^2 := \sigma^2.$$

The following corollary applies Theorem 1.4 in [83] to bound the spectral norm $\|\widehat{R} - \bar{R}\|$.

Corollary B.1. *If $\sigma^2 \geq C' \log^4 n/n$ for a constant C', then conditioned on*

R,

$$\|\widehat{R} - \bar{R}\| \leq 3\sigma\sqrt{n} \quad a.a.s. \tag{B.8}$$

Proof. We adopt the trick called dilations [84]. In particular, define A as

$$A = \begin{bmatrix} \mathbf{0} & \widehat{R} - \mathbb{E}[\widehat{R}|R] \\ \widehat{R}^\top - \mathbb{E}[\widehat{R}^\top|R] & \mathbf{0} \end{bmatrix}. \tag{B.9}$$

Observe that $\|A\| = \|\widehat{R} - \mathbb{E}[\widehat{R}|R]\|$, so it is sufficient to prove the theorem for $\|A\|$. Conditioned on R, A is a random symmetric $2n \times 2n$ matrix with each entry bounded by 1, and a_{ij} $(1 \leq i < j \leq 2n)$ are independent random variables with mean 0 and variance *at most* σ^2. By Theorem 1.4 in [83] , if $\sigma \geq C' n^{-1/2} \log^2 n$, then conditioned on R a.a.s.

$$\begin{aligned} \|\widehat{R} - \mathbb{E}[\widehat{R}|R]\| = \|A\| &\leq 2\sigma\sqrt{2n} + C(2\sigma)^{1/2}(2n)^{1/4}\log(2n) \\ &\leq 3\sigma\sqrt{n}. \end{aligned} \tag{B.10}$$

□

Proof of Theorem 3.4. For any feasible Y that $Y \neq R$, we have to show that $\Delta(Y) = \langle \widehat{R}, R\rangle - \lambda||R||_* - (\langle \widehat{R}, Y\rangle - \lambda||Y||_*) > 0$. Rewrite $\Delta(Y)$ as

$$\begin{aligned} \Delta(Y) =& \langle \bar{R}, R - Y\rangle + \langle \widehat{R} - \bar{R}, R - Y\rangle \\ &+ \lambda(||Y||_* - ||R||_*). \end{aligned} \tag{B.11}$$

The first term in (B.11) can be written as

$$\begin{aligned} \langle \bar{R}, R - Y\rangle &= (1-\epsilon)(1-2p)\langle R, R - Y\rangle \\ &= (1-\epsilon)(1-2p)||R - Y||_1, \end{aligned}$$

where the second equality follows from the fact that $Y_{ij} \in [-1, 1]$ and $R_{ij} = \mathrm{sgn}(R_{ij})$. Define the normalized noise matrix $W = (\widehat{R} - \bar{R})/\lambda$. Note that $||W||_\infty \leq 1/\lambda$ and $\mathrm{Var}(W_{ij}) \leq 1/9n$. The second term in (B.11) becomes $\langle \widehat{R} - \bar{R}, R - Y\rangle = \lambda\langle W, R - Y\rangle$. By Corollary B.1, $||W|| \leq 1$ almost surely. Thus $UV^\top + \mathcal{P}_{T^\perp}(W)$ is a subgradient of $||X||_*$ at $X = R$. Hence, for the third

term in (B.11), $\lambda(||Y||_* - ||R||_*) \geq \lambda\langle UV^\top + \mathcal{P}_{T^\perp}(W), Y - R\rangle$. Therefore,

$$\begin{aligned}
&\Delta(Y) \\
&\geq (1-\epsilon)(1-2p)||R-Y||_1 + \lambda\langle UV^\top - \mathcal{P}_T(W), Y - R\rangle \\
&\geq [(1-\epsilon)(1-2p) - \lambda(||UV^\top||_\infty + ||\mathcal{P}_T(W)||_\infty)]||R-Y||_1 \\
&\geq [(1-\epsilon)(1-2p) - \lambda(\mu\sqrt{r}/n + ||\mathcal{P}_T(W)||_\infty)]||R-Y||_1, \qquad \text{(B.12)}
\end{aligned}$$

where the last inequality follows from definition of the incoherence parameter μ. Below we bound the term $||\mathcal{P}_T(W)||_\infty$. From the definition of $\mathcal{P}_T$ and the fact that $U_B U_B^\top = I$ and $V_B V_B^\top = I$,

$$\begin{aligned}
||\mathcal{P}_T(W)||_\infty \leq &||U_C U_C^\top W||_\infty + ||W V_C V_C^\top||_\infty \\
&+ ||U_C U_C^\top W V_C V_C^\top||_\infty.
\end{aligned}$$

We bound $||U_C U_C^\top W||_\infty$. To bound the term $(U_C U_C^\top W)_{ij}$, assume user i belongs to user cluster k and let $\mathcal{C}_k$ be the set of users in user cluster k. Recall that $u_{C,k}$ is the normalized characteristic vector of user cluster k. Then

$$(U_C U_C^\top W)_{ij} = (u_{C,k} u_{C,k}^\top W)_{ij} = (1/K) \sum_{i' \in \mathcal{C}_k} W_{i'j},$$

which is the average of K independent random variables. By Bernstein's inequality (stated in the supplementary material), with probability at least $1 - n^{-3}$,

$$\left| \sum_{i' \in \mathcal{C}_k} W_{i'j} \right| \leq \sqrt{\frac{2}{3r} \log n} + \frac{2 \log n}{\lambda}.$$

Then $||U_C U_C^\top W||_\infty \leq \frac{1}{K}\left(\sqrt{\frac{2}{3r}\log n} + \frac{2\log n}{\lambda}\right)$ with probability at least $1 - n^{-1}$. Similarly we bound $||W V_C V_C^\top||_\infty$ and $||U_C U_C^\top W V_C V_C^\top||_\infty$. Therefore, with probability at least $1 - 3n^{-1}$,

$$||\mathcal{P}_T(W)||_\infty \leq \frac{C_1}{K}\left(\sqrt{\frac{\log n}{r}} + \frac{\log n}{\lambda}\right) \leq \frac{C_2}{K}\sqrt{\frac{\log n}{r}}, \qquad \text{(B.13)}$$

for some constants C_1 and C_2, where the second inequality follows from the

assumption in (3.4). Substituting (B.13) into (B.12) and by (3.4) again, we conclude that $\Delta(Y) > 0$ a.a.s. □

B.5 Proof of Theorem 3.5

The proof is divided into three parts. Recall that x_i denotes the i-th row of $Pr(\widehat{R}^{(1)})$. We first show that, for most users, x_i is close to the expected value conditioned on R. Then we show that the clusters output by Step 2 are close to the true clusters. Finally, we show that Step 3 exactly recovers the block rating matrix B and Step 4 exactly recovers clusters.

Define $\bar{R}^{(1)} = \mathbb{E}\left[\widehat{R}^{(1)}|R\right] = \frac{1}{2}(1-\epsilon)(1-2p)R$ and let $\bar{x}_i$ be the i-th row of $\bar{R}^{(1)}$. We call user i a *good* user if $\|x_i - \bar{x}_i\|_2 \leq \tau/2$ where the threshold $\tau = 12(1-\epsilon)^{1/2}\log n$; otherwise it is called a *bad* user. Let $\mathcal{I}$ denote the set of all good users and $\mathcal{I}^c$ denote the set of all bad users. Define good items in the same way, and let $\mathcal{J}$ denote the set of all good items and $\mathcal{J}^c$ denote the set of all bad items. The following lemma shows that the number of bad users (items) are bounded by $K\log^{-2} n$.

Lemma B.1. *If $\sigma^2 \geq C'\log^4 n/n$ for a constant C', then a.a.s., $|\mathcal{I}^c| \leq K\log^{-2} n$ and $|\mathcal{J}^c| \leq K\log^{-2} n$.*

Proof. Let $(\sigma^{(1)})^2 = \frac{1}{2}(1-\epsilon)$. By Corollary B.1, $\|\widehat{R}^{(1)} - \bar{R}^{(1)}\| \leq 3\sigma^{(1)}\sqrt{n}$. Note that

$$\begin{aligned}\|P_r(\widehat{R}^{(1)}) - \bar{R}\| &\leq \|P_r(\widehat{R}^{(1)}) - \widehat{R}^{(1)}\| + \|\widehat{R}^{(1)} - \bar{R}\| \\ &\leq 2\|\widehat{R}^{(1)} - \bar{R}\|,\end{aligned}$$

where the second inequality follows from the definition of $P_r(\widehat{R}^{(1)})$ and the fact that $\bar{R}$ has rank r. Since both $P_r(\widehat{R}^{(1)})$ and $\bar{R}$ have rank r, the matrix $P_r(\widehat{R}^{(1)}) - \bar{R}$ has rank at most $2r$, which implies that

$$\|P_r(\widehat{R}^{(1)}) - \bar{R}\|_F^2 \leq 8r\|\widehat{R}^{(1)} - \bar{R}\|^2 \leq 72(\sigma^{(1)})^2 nr.$$

As $\sum_{i=1}^n \|x_i - \bar{x}_i\|_2^2 = \|P_r(\widehat{R}^{(1)}) - \bar{R}\|_F^2$, we conclude that there are at most $K\log^{-2} n$ users with

$$\|x_i - \bar{x}_i\|_2 > 6\sqrt{2}\sigma^{(1)} r\log n = \tau/2.$$

Similarly we can prove the result for items. □

The following proposition upper bounds the set difference between the estimated clusters and the true clusters by $K \log^{-2} n$. Let $C_1^*, \ldots, C_r^*$ be the true user clusters and Δ denote the set difference.

Proposition B.1. *Assume the assumption of Theorem 3.5 holds. Step 2 of Algorithm 5 outputs $\{\widehat{C}_k\}_{k=1}^r$ and $\{\widehat{D}_l\}_{1=1}^r$ such that, up to a permutation of cluster indices, a.a.s., $\widehat{C}_k \Delta C_k^* \subset \mathcal{I}^c$ and $\widehat{D}_l \Delta D_l^* \subset \mathcal{J}^c$ for all k, l. It follows that for all k, l,*

$$|\widehat{C}_k \Delta C_k^*| \leq \frac{K}{\log^2 n}, \quad |\widehat{D}_l \Delta D_l^*| \leq \frac{K}{\log^2 n}. \tag{B.14}$$

Proof. It suffices to prove the conclusion for the user clusters. Consider two good users $i, i' \in \mathcal{I}$. If they are from the same cluster, we have $\bar{x}_i = \bar{x}_{i'}$ and

$$\|x_i - x_{i'}\| \leq \|x_i - \bar{x}_i\| + \|x_{i'} - \bar{x}_{i'}\| \leq \tau, \tag{B.15}$$

where the last inequality follows from Lemma B.1. If they are from different clusters, by (B.1), we have a.a.s.

$$\begin{aligned} \|\bar{x}_i - \bar{x}_{i'}\|_2^2 =& \frac{1}{4}(1-\epsilon)^2(1-2p)^2 ||R_i - R_{i'}||_2^2 \\ \geq& \frac{1}{4}(1-\epsilon)^2(1-2p)^2 n, \end{aligned}$$

where R_i denotes the i-th row of R. Thus,

$$\begin{aligned} \|x_i - x_{i'}\| &\geq \|\bar{x}_i - \bar{x}_{i'}\| - \|x_i - \bar{x}_i\| - \|x_{i'} - \bar{x}_{i'}\| \\ &\geq \frac{1}{2}(1-\epsilon)(1-2p)\sqrt{n} - \tau > \tau, \end{aligned} \tag{B.16}$$

where the last inequality follows from the assumption (3.5). Therefore, in the clustering procedure of Step 2, if we choose a good initial user at some iteration, the corresponding estimated cluster will contain all the good users from the same cluster as the initial user and no good user from other clusters. It is not hard to see that the probability of the event that we choose a good

initial user in every iteration is lower bounded by

$$\begin{aligned}
&\left(1-\frac{1}{r\log^2 n}\right)\left(1-\frac{1}{(r-1)\log^2 n}\right)\cdots\left(1-\frac{1}{\log^2 n}\right)\\
&\geq 1-\frac{1}{\log^2 n}\left(\frac{1}{r}+\frac{1}{r-1}+\cdots+1\right)\\
&\geq 1-\frac{\log r}{\log^2 n}\geq 1-\frac{1}{\log n}.
\end{aligned}$$

Assume the above event holds. Under proper permutation, the initial good user in the k-th iteration is from cluster C_k^* for all k. By the above argument, the set difference $\widehat{C}_k \Delta C_k^* \subset \mathcal{I}^c$. By Lemma B.1, (B.14) follows. □

Proof of Theorem 3.5. We first show that Step 3 of Algorithm 5 exactly recovers the block rating matrix B. Let V_{kl} denote the total vote that the true user cluster k gives to the true item cluster l, i.e.,

$$V_{kl} = \sum_{i\in C_k^*}\sum_{j\in D_l^*} \widehat{R}_{ij}^{(2)}.$$

Then by definition of $\widehat{V}_{kl}$,

$$\begin{aligned}
|\widehat{V}_{kl} - V_{kl}| \leq &\sum_{i\in C_k^*\Delta\widehat{C}_k}\sum_{j\in D_l^*\cup\widehat{D}_l} \mathbb{I}_{\{(i,j)\in\Omega_2\}}\\
&+\sum_{i\in C_k^*\cup\widehat{C}_k}\sum_{j\in D_l^*\Delta\widehat{D}_l} \mathbb{I}_{\{(i,j)\in\Omega_2\}}. \qquad \text{(B.17)}
\end{aligned}$$

Without loss of generality, assume $B_{kl}=1$. By Bernstein inequality and assumption (3.5), $V_{kl}\geq \frac{1}{4}(1-\epsilon)(1-2p)K^2$ a.a.s. On the other hand, as Ω_2 and $\widehat{R}^{(1)}$ are independent, Ω_2 is independent from $\{\widehat{C}_k\}$ and $\{\widehat{D}_l\}$. It follows from (B.14) and the Chernoff bound that each term on the right-hand side of (B.17) is upper bounded by $(1-\epsilon)K^2\log^{-2} n$ a.a.s. Hence, when assumption (3.5) holds for some large enough constant C, we have $\widehat{V}_{kl}>0$ thus $\widehat{B}_{kl}=B_{kl}$.

Next we prove that Step 4 clusters the users and items correctly. Without loss of generality, we only prove the correctness for users. Suppose user i is from cluster k. Recall that R_i denotes the i-th row of R. When $\widehat{B}=B$, we

have $\mu_{kj} = R_{ij}$ for $j \in \mathcal{J}$ by definition and Proposition B.1. Then

$$\begin{aligned}\langle \widehat{R}_i^{(2)}, \mu_k \rangle =& \langle \widehat{R}_i^{(2)}, R_i \rangle + \langle \widehat{R}_i^{(2)}, \mu_k - R_i \rangle \\ \geq& \langle \widehat{R}_i^{(2)}, R_i \rangle - 2 \sum_{j \in \mathcal{J}^c} |\widehat{R}_{ij}^{(2)}|. \end{aligned} \tag{B.18}$$

Similarly, for some user i' from cluster $k' \neq k$,

$$\begin{aligned}\langle \widehat{R}_i^{(2)}, \mu_{k'} \rangle =& \langle \widehat{R}_i^{(2)}, R_{i'} \rangle + \langle \widehat{R}_i^{(2)}, \mu_{k'} - R_{i'} \rangle \\ \leq& \langle \widehat{R}_i^{(2)}, R_{i'} \rangle + 2 \sum_{j \in \mathcal{J}^c} |\widehat{R}_{ij}^{(2)}|. \end{aligned} \tag{B.19}$$

For ease of notation, let $t := \frac{1}{2}(1-\epsilon)(1-2p)n$ and $(\sigma^{(2)})^2 = \frac{1}{2}(1-\epsilon)$. By (B.1), $\langle R_i, R_{i'} \rangle \leq n/2$ for all $i \neq i'$. Then conditioned on R, we have $\mathbb{E}[\langle \widehat{R}_i^{(2)}, R_i \rangle] = t$ and $\mathrm{Var}[\langle \widehat{R}_i^{(2)}, R_i \rangle] \leq n(\sigma^{(2)})^2$, and

$$\mathbb{E}[\langle \widehat{R}_i^{(2)}, R_{i'} \rangle] = \frac{1}{2}(1-\epsilon)(1-2p)\langle R_i, R_i' \rangle \leq t/2$$

and $\mathrm{Var}[\langle \widehat{R}_i^{(2)}, R_{i'} \rangle] \leq n(\sigma^{(2)})^2$. Now by the Bernstein inequality and assumption (3.5), we have that conditioned on R, a.a.s. $\langle \widehat{R}_i^{(2)}, R_i \rangle > 7t/8$ and $\langle \widehat{R}_i^{(2)}, R_{i'} \rangle < 5t/8$ for all $i \neq i'$.

On the other hand, because $\mathcal{J}$ and Ω_2 are independent, by the Chernoff bound, a.a.s. $\sum_{j \in \mathcal{J}^c} |\widehat{R}_{ij}^{(2)}|$ is upper bounded by $(1-\epsilon)K \log^{-2} n < t/16$ for all i, when assumption (3.5) holds for some large enough constant C.

Therefore, from (B.18) and (B.19), $\langle \widehat{R}_i^{(2)}, \mu_k \rangle > \langle \widehat{R}_i^{(2)}, \mu_{k'} \rangle$ for all $k' \neq k$. □

APPENDIX C

PROOFS IN CHAPTER 4

C.1 Proof of Lemma 4.1

Suppose we require $\mathbb{P}[(u,ij) \in \Omega_1] = \mathbb{P}[(u,ij) \in \Omega_2] = (1-\epsilon)\delta$ for some $\delta \in (0,1)$. According to the assignment, we have $a+b=\delta$ and it is not hard to see that the independence of Ω_1 and Ω_2 is equivalent to $(1-\epsilon)a = (1-\epsilon)^2\delta^2$. Therefore, $a = (1-\epsilon)\delta^2$ and $b = \delta - (1-\epsilon)\delta^2$. The constraint $a + 2b \leq 1$ implies that $2\delta - (1-\epsilon)\delta^2 \leq 1$ and we can choose $\delta = 1/2$.

C.2 Proof of Lemma 4.2

Note that $AA^\top = L_m$, which is the Laplacian of a complete graph on m vertices, so A is of rank $m-1$ and all nonzero singular values are $\sqrt{m}$, i.e., the SVD of A is $A = \sqrt{m}UV^\top$. As the first eigenvector of L_m has all entries being $1/\sqrt{m}$, the l_2 norms of the rows of U are $\sqrt{(m-1)/m}$. Let U_i be the i-th row of U and note that $\{[U_i, 1/\sqrt{m}]\}$ is an orthonormal basis, then the l_2 norms of the rows of V are

$$\begin{aligned}
||V_{ij}||_2 =& ||U^\top A_{ij}||_2/\sqrt{m} \\
=& ||U_i - U_j||_2/\sqrt{m} \\
=& ||[U_i, 1/\sqrt{m}] - [U_j, 1/\sqrt{m}]||_2/\sqrt{m} \\
=& \sqrt{2/m}.
\end{aligned}$$

C.3 Proof of Lemma 4.3

Using the properties of A, we have

$$||\eta^\top V||_2^2 = \eta^\top V V^\top \eta = \frac{1}{m}\eta^\top A^\top A\eta = \frac{1}{m}||A\eta||^2.$$

Note that

$$(A\eta)_i = \#\{j : \theta_{k,j} < \theta_{k,i}\} - \#\{j : \theta_{k,j} > \theta_{k,i}\}.$$

By the assumption that $\theta_{k,i} \neq \theta_{k,j}$ for any i and j, the vector $A\eta$ is always a permutation of the deterministic vector

$$[-(m-1), -(m-3), \ldots, m-3, m-1]^\top$$

representing the net wins of the items. Therefore,

$$||\eta^\top V||_2^2 = \frac{1}{m}||[-(m-1), -(m-3), \ldots, m-3, m-1]^\top||^2 = \frac{1}{3}(m^2-1).$$

C.4 Proof of Theorem 4.2

We prove the theorem by considering the two regimes of b separately in the following two lemmas.

Lemma C.1. *Assume $m \geq C' \log r$. If $b \in [0.6, 5]$, then a.a.s. there exists some constant C such that for any $k \neq k'$,*

$$||\bar{S}_k - \bar{S}_{k'}|| \geq C(1-\epsilon)m.$$

Proof. By definition, $\bar{R}_{u,ij} = \frac{1-\epsilon}{2} f(\theta_{u,i} - \theta_{u,j})$, and we have

$$\mathrm{Var}\left[R_{u,ij}\right] \leq \mathbb{E}\left[R_{u,ij}^2\right] = \frac{1-\epsilon}{2}.$$

The function $f(x)$ is nonlinear, but it behaves like $x/2$ for x close to 0. According to the way we generate θ_k, the maximum approximation error is given by $\delta(b) = |f(b) - \frac{b}{2}|$.

By definition, for any k,

$$\begin{aligned}\bar{S}_k =& \frac{1-\epsilon}{2} f(\theta_k^\top A)V \\ =& \frac{1-\epsilon}{2}\frac{1}{2}\theta_k^\top \sqrt{m}U + \frac{1-\epsilon}{2}(f(\theta_k^\top A) - \frac{1}{2}\theta_k^\top A)V.\end{aligned}$$

Then, the difference between $\bar{S}_k$ and $\bar{S}_{k'}$ is lower bounded by

$$\begin{aligned}||\bar{S}_k - \bar{S}_{k'}||_2 \geq& \frac{1-\epsilon}{2}\sqrt{m}||(\theta_k - \theta_{k'})U||_2 \\ & - \frac{1-\epsilon}{2}\left[||(f(\theta_k^\top A) - \frac{1}{2}\theta_k^\top A)V||_2 + ||(f(\theta_{k'}^\top A) - \frac{1}{2}\theta_{k'}^\top A)V||_2\right].\end{aligned}$$

As $\sum_i \theta_{k,i} = \sum_i \theta_{k',i} = 0$,

$$||(\theta_k - \theta_{k'})U||_2 = ||(\theta_k - \theta_{k'})[U, \frac{1}{\sqrt{m}}\mathbf{1}]||_2 = ||\theta_k - \theta_{k'}||_2,$$

where $\mathbf{1}$ is the vector with all ones. Using the fact that

$$|f((\theta_i - \theta_j) - (\theta_i - \theta_j)/2| \leq \frac{\delta(b)}{b}|\theta_i - \theta_j|,$$

we get

$$||f(\theta_k^\top A) - \frac{1}{2}\theta_k^\top A||_2 \leq \frac{\delta(b)}{b}\sqrt{\sum_{i<j}(\theta_{k,i} - \theta_{k,j})^2} = \frac{\delta(b)}{b}\sqrt{m||\theta_k||_2^2}.$$

Therefore,

$$||\bar{S}_k - \bar{S}_{k'}||_2 \geq \frac{1-\epsilon}{2}\sqrt{m}\left[\frac{1}{2}||\theta_k - \theta_{k'}||_2 - \frac{\delta(b)}{b}(||\theta_k||_2 + ||\theta_{k'}||_2)\right].$$

First we bound $||\theta_k - \theta_{k'}||_2$. Recall that θ_k is the centered version of θ_k^0, and $\theta_{k,i}^0$ are generated i.i.d. uniformly in $[0, b]$. When $m > C_1 \log r$, by Hoeffding's inequality,

$$|\sum_i \theta_{k,i}^0 - \frac{mb^2}{2}| \leq C_2\sqrt{m\log r}, \quad |\sum_i \theta_{k',i}^0 - \frac{mb^2}{2}| \leq C_2\sqrt{m\log r}$$

with high probability. By definition,

$$\begin{aligned}||\theta_k - \theta_{k'}||_2 \geq & ||\theta_k^0 - \theta_{k'}^0||_2 - ||\frac{1}{m}(\sum_i \theta_{k,i}^0 - \sum_i \theta_{k',i}^0)\mathbf{1}||_2 \\ \geq & ||\theta_k^0 - \theta_{k'}^0||_2 - 2C_2\sqrt{\log r}.\end{aligned}$$

To bound $||\theta_k^0 - \theta_{k'}^0||_2$, note that $\mathbb{E}\left[||\theta_k^0 - \theta_{k'}^0||_2^2\right] = \mathbb{E}\left[\sum_i (\theta_{k,i}^0 - \theta_{k',i}^0)^2\right] = \frac{mb^2}{6}$. Define $X_i = (\theta_{k,i}^0 - \theta_{k',i}^0)^2 - \frac{b^2}{6}$. Then $|X_i| \leq b^2, \mathbb{E}\left[X_i\right] = 0$ and

$$\begin{aligned}\mathbb{E}\left[X_i^2\right] = & \mathbb{E}\left[(\theta_{k,i}^0 - \theta_{k',i}^0)^4\right] - \frac{b^4}{36} \\ = & \mathbb{E}\left[(\theta_{k,i}^0 - \frac{b}{2})^4\right] + 6\mathbb{E}\left[(\theta_{k,i}^0 - \frac{b}{2})^2(\theta_{k',i}^0 - \frac{b}{2})^2\right] \\ & + \mathbb{E}\left[(\theta_{k',i}^0 - \frac{b}{2})^4\right] - \frac{b^4}{36} \\ \leq & \frac{b^4}{80} + \frac{b^4}{6} + \frac{b^4}{80} - \frac{b^4}{36} \leq \frac{b^4}{6}.\end{aligned}$$

By Bernstein's inequality, when $m \geq C_3 \log r$, with high probability,

$$\left| ||\theta_k^0 - \theta_{k'}^0||_2^2 - \frac{mb^2}{6} \right| \leq C_4\sqrt{m\log r}b^2.$$

When m is large enough, we have $||\theta_k - \theta_{k'}||_2 \geq \sqrt{0.9mb^2/6}$ with high probability.

Next we bound $||\theta_k||_2$. By definition, $||\theta_k||_2^2$ is the sample variance for θ_k^0 and $||\theta_k||_2$ is always bounded by $\sqrt{mb^2/4}$. Using the fact that $\theta_{k,i}^0$ is uniform over $[0, b]$, we can similarly show that, when $m \geq C_5 \log r$, $||\theta_k||_2 \leq \sqrt{1.1mb^2/12}$ with high probability.

Combining the two inequalities, we get

$$||\bar{S}_k - \bar{S}_{k'}||_2 \geq (\frac{1}{2}\sqrt{\frac{0.9}{6}} - \sqrt{\frac{1.1}{3}}\frac{\delta(b)}{b})\frac{b}{2}(1-\epsilon)m \triangleq C(1-\epsilon)m.$$

Note that $b/\delta(b)$ decreases with b. For $b \in [0.6, 5]$, the constant C is larger than 0.05. □

Lemma C.2. *Assume $m \geq C' \log r$. If $b \geq C''m^3 \log m$, then a.a.s. there*

exists some constant C such that for any $k \neq k'$,

$$||\bar{S}_k - \bar{S}_{k'}|| \geq C(1-\epsilon)m.$$

Proof. The assumption that b is large implies that $|\theta_{k,i} - \theta_{k,j}|$ is large for any k and $i < j$. To show this, we note that

$$\mathbb{P}\left[|\theta_{k,i} - \theta_{k,j}| < 1\right] \leq \frac{2}{b},$$

then by union bound we get

$$\mathbb{P}\left[\forall k, i < j, |\theta_{k,i} - \theta_{k,j}| \geq 1\right] \geq 1 - \frac{m^3}{b} \geq 1 - \frac{C''}{\log m}. \tag{C.1}$$

In the following we will assume $\theta_{k,i} \neq \theta_{k,j}$ for $i \neq j$. By definition, $\bar{S}_k = \frac{1-\epsilon}{2} f(\theta_k A)V$. Define $\eta_{k,ij} = \mathbb{I}_{\left\{\theta_{k,i} > \theta_{k,j}\right\}} - \mathbb{I}_{\left\{\theta_{k,i} < \theta_{k,j}\right\}}$ to be the signed indicator variable of the order between $\theta_{k,i}$ and $\theta_{k,j}$. Then

$$\begin{aligned}
||\bar{S}_k - \bar{S}_{k'}|| =& \frac{1-\epsilon}{2}||(f(\theta_k A) - f(\theta_{k'} A))V||_2 \\
\geq& \frac{1-\epsilon}{2}[||(\eta_k - \eta_{k'})V||_2 - ||(f(\theta_k A) - \eta_k)V||_2 \\
& - ||(f(\theta_{k'} A) - \eta_{k'})V||_2] \\
\geq& \frac{1-\epsilon}{2}\left[||(\eta_k - \eta_{k'})V||_2 - ||f(\theta_k A) - \eta_k||_2 - ||f(\theta_{k'} A) - \eta_{k'}||_2\right].
\end{aligned}$$

First we show that $||f(\theta_k A) - \eta_k||_2 \leq C_1\sqrt{m}$. When $|\theta_{k,i} - \theta_{k,j}| \geq t$,

$$|f(\theta_{k,i} - \theta_{k,j}) - \eta_{k,ij}| \leq \frac{2}{e^t + 1} \leq 2e^{-t}.$$

According to (C.1), for any integer $1 \leq t \leq m$, there are $m - t$ pairs of $\theta_{k,i}$ and $\theta_{k,i}$ separated at least by t. Therefore,

$$||f(\theta_k A) - \eta_k||_2^2 \leq \sum_{t=1}^{m}(m-t)4e^{-2t} \leq C_1 m.$$

We bound $||f(\theta_{k'} A) - \eta_{k'}||_2$ similarly.

Next we show that $||(\eta_k - \eta_{k'})V||_2 \geq C_2 m$. Observe that

$$\begin{aligned} ||(\eta_k - \eta_{k'})V||_2^2 =& ||\eta_k V||_2^2 + ||\eta_{k'} V||_2^2 - 2\eta_k V V^\top \eta_{k'}^\top \\ =& \frac{2}{3}(m^2 - 1) - \frac{2}{m}\eta_k A^\top A \eta_{k'}^\top, \end{aligned} \tag{C.2}$$

where the second equality follows from Lemma 4.3. By definition of A, $(\eta_k A^\top)_i$ represents the number of θ_j that are smaller than θ_i minus the number of θ_j that are larger than θ_i. Therefore, $\eta_k A^\top$ and $\eta_{k'} A^\top$ are independent random permutations of the deterministic vector $[-(m-1), -(m-3), \ldots, m-3, m-1]$. Without loss of generality, assume $\eta_k A^\top = [-(m-1), -(m-3), \ldots, m-3, m-1]$ and denote $\eta_{k'} A^\top$ by x which is a random permutation of $\eta_k A^\top$. Let $Z = \eta_k A^\top A \eta_{k'}^\top = -(m-1)x_1 + \cdots + (m-1)x_m$ and define the martingale $y_i = \mathbb{E}[Z|x_1, \ldots, x_i]$. In particular, we have $y_0 = \mathbb{E}[Z] = 0$ and $y_m = Z$. We also note that $|y_{i+1} - y_i| \leq 2m^2$. By Azuma's inequality,

$$\mathbb{P}[|Z| \geq t] = \mathbb{P}[|y_m - y_0| \geq t] \leq 2e^{-\frac{t^2}{8m^5}},$$

i.e., $|Z| \leq C_3 m^{5/2} \log^{1/2} m$ with high probability. Plugging it into (C.2), we get $||(\eta_k - \eta_{k'})V||_2 \geq C_2 m$.

Combining the above two steps, we conclude that

$$||\bar{S}_k - \bar{S}_{k'}|| \geq C(1-\epsilon)m.$$

□

C.5 Proof of Theorem 4.3

By definition, the rows of S are independent, and we have $\mathbb{E}\left[||S_u - \bar{S}_u||_2^2\right] \leq \mathbb{E}\left[||R_u - \bar{R}_u||_2^2\right] \leq (1-\epsilon)m^2$. The following lemma shows that this bound is loose, and $||S_u - \bar{S}_u||_2$ is in fact of order $O(\sqrt{(1-\epsilon)m})$.

Lemma C.3. *If $(1-\epsilon)m^2 > 36 \log n$, then with high probability,*

$$||S_u - \bar{S}_u||_2 \leq 3\sqrt{\frac{1-\epsilon}{2} m \log n}, \quad \forall u.$$

Proof. Rewrite $S_u - \bar{S}_u$ as

$$S_u - \bar{S}_u = \sum_{ij} (R_{u,ij} - \bar{R}_{u,ij}) V_{ij} \triangleq \sum_{ij} Z_{ij}.$$

Note that $||Z_{ij}||_2 \leq ||V_{ij}||_2 = \sqrt{2/m}$. Moreover,

$$\sum_{ij} \mathbb{E}\left[||Z_{ij}||_2^2\right] \leq \frac{1-\epsilon}{2} \binom{m}{2} \frac{2}{m} \leq \frac{1-\epsilon}{2} m \triangleq \sigma^2.$$

Now we apply the vector Bernstein's inequality [85, Theorem 12]. We choose $t = 3\sigma\sqrt{\log n}$ and under our assumption it satisfies $t\sqrt{2/m} \leq \sigma^2$. Then, for any u,

$$\begin{aligned} &\mathbb{P}\left[||S_u - \bar{S}_u||_2 > 4\sigma\sqrt{\log n}\right] \\ \leq &\mathbb{P}\left[||S_u - \bar{S}_u||_2 > \sigma + t\right] \leq \exp(-\frac{t^2}{4\sigma^2}) \leq 1/n^2. \end{aligned}$$

Applying the union bound, we get the result. □

Now Theorem 4.3 immediately follows from Theorem 4.2 and Lemma C.3.

Proof of Theorem 4.3. For large enough C, the condition implies that for any user u from cluster k and any cluster $k' \neq k$,

$$||S_u - \bar{S}_u||_2 \leq \frac{\tau}{2} < \frac{1}{4}||\bar{S}_k - \bar{S}_{k'}||_2.$$

Therefore, for any two users u and u', if they are from the same cluster k, then

$$||S_u - S_{u'}||_2 \leq ||S_u - \bar{S}_k||_2 + ||S_{u'} - \bar{S}_k||_2 \leq \tau;$$

if they are from different clusters k and k', then

$$||S_u - S_{u'}||_2 \geq ||\bar{S}_k - \bar{S}_{k'}||_2 - ||S_u - \bar{S}_k||_2 - ||S_{u'} - \bar{S}_k||_2 > \tau.$$

Suppose the first initial user we choose is from cluster k, then the above two inequalities imply that the set of users in C_1 is the same as the set of users in cluster k. By the same argument, we can show all users are clustered clustered correctly by the algorithm. □

C.6 Proof of Theorem 4.4

The key step in proving this theorem is to show that spectral norm $||S - \bar{S}||$ is small.

Lemma C.4. *If* $(1-\epsilon)m^2 > 36 \log n$, *then with high probability,*

$$||S - \bar{S}|| \leq 8\sqrt{(1-\epsilon)\max\{m,n\}}\log^{3/2} n.$$

Proof. We bound $||S - \bar{S}||$ by the matrix Bernstein's inequality [84]. Let $X_u = e_u(S_u - \bar{S}_u)$, then $S - \bar{S} = \sum_u X_u$. First we bound $||X_u||$. Since

$$||X_u||^2 = ||X_u X_u^\top|| = ||S_u - \bar{S}_u||_2^2 ||e_u e_u^\top|| = ||S_u - \bar{S}_u||_2^2,$$

by Lemma C.3, $||X_u|| \leq 3\sqrt{\frac{1-\epsilon}{2} m \log n}$ with high probability. Next we bound

$$\sigma^2 \triangleq \max\{||\sum_u \mathbb{E}\left[X_u X_u^\top\right]||, ||\sum_u \mathbb{E}\left[X_u^\top X_u\right]||\}.$$

The covariance matrix for S_u is $\Sigma_u = V^\top D_u V$, where

$$D_u = \mathrm{diag}([\mathrm{Var}\left[R_{u,ij}\right]]_{ij}) \leq \frac{1-\epsilon}{2} I.$$

Then

$$\begin{aligned} ||\sum_u \mathbb{E}\left[X_u X_u^\top\right]|| &= ||\sum_u \mathbb{E}\left[||S_u - \bar{S}_u||_2^2 e_u e_u^\top\right]|| \\ &= \max_u \mathbb{E}\left[||S_u - \bar{S}_u||_2^2\right] \\ &= \max_u \mathrm{Tr}\left[V^\top D_u V\right]. \end{aligned}$$

Since $D_u \leq \frac{1-\epsilon}{2} I$ and using the fact that $A \leq B$ implies $\mathrm{Tr}\left[V^\top (B-A) V\right] \geq$

0, we get $||\sum_u \mathbb{E}\left[X_u X_u^\top\right]|| \leq \frac{1-\epsilon}{2}m$. Similarly,

$$\begin{aligned}||\sum_u \mathbb{E}\left[X_u^\top X_u\right]|| =& ||\sum_u \mathbb{E}\left[(S_u - \bar{S}_u)^\top (S_u - \bar{S}_u)\right]|| \\ =& ||\sum_u V^\top D_u V|| \\ =& ||V^\top (\sum_u D_u) V|| \\ \leq& \frac{1-\epsilon}{2}n,\end{aligned}$$

where the last inequality follows from $D_u \leq \frac{1-\epsilon}{2}I$ and the fact that $A \leq B$ implies $||V^\top A V|| \leq ||V^\top B V||$ Therefore, $\sigma^2 \leq \frac{1-\epsilon}{2}\max\{m,n\}$. Now by applying the matrix Bernstein's inequality, we get

$$\begin{aligned}||S - \bar{S}|| \leq& 3\max\{\max_u ||X_u||\log n, \sigma\sqrt{\log n}\} \\ \leq& 8\sqrt{(1-\epsilon)\max\{m,n\}}\log^{3/2} n\end{aligned}$$

with probability at least $1 - 2/n$. □

Using this lemma, we can show that most users $\tilde{S}_u$ are close to the expected comparison vectors $\bar{S}_u$.

Corollary C.1. *Let $\tau = 32\sqrt{\frac{2(1-\epsilon)r\max\{m,n\}}{K}}\log^{5/2} n$, then with high probability, there are at most $\frac{K}{\log^2 n}$ users such that $||\tilde{S}_u - \bar{S}_u||_2 > \frac{\tau}{2}$.*

Proof. By Lemma C.4, with probability at least $1 - 2/n$,

$$||S - \bar{S}|| \leq 8\sqrt{(1-\epsilon)\max\{m,n\}}\log^{3/2} n.$$

Note that

$$\begin{aligned}||\tilde{S} - \bar{S}|| \leq& ||\tilde{S} - S|| + ||S - \bar{S}|| \\ \leq& 2||S - \bar{S}||,\end{aligned}$$

where the second inequality follows from the definition of $\tilde{S}$ and the fact that

$\bar{S}$ has rank r. Since the matrix $\tilde{S} - \bar{S}$ is of rank at most $2r$, we get

$$\begin{aligned}||\tilde{S} - \bar{S}||_F^2 \leq & (\sqrt{2r}||\tilde{S} - \bar{S}||)^2 \\ \leq & 8r||S - \bar{S}||^2 \\ \leq & 512(1-\epsilon)r\max\{m,n\}\log^3 n.\end{aligned}$$

As $||\tilde{S} - \bar{S}||_F^2 = \sum_u ||\tilde{S}_u - \bar{S}_u||_2^2$, we conclude that there are at most $\frac{K}{\log^2 n}$ users with

$$||\tilde{S}_u - \bar{S}_u||_2 > 16\sqrt{\frac{2(1-\epsilon)r\max\{m,n\}}{K}}\log^{5/2} n = \frac{\tau}{2}.$$

□

Combined with the fact that $\bar{S}_k$'s are well separated as shown in Theorem 4.2, we get Theorem 4.4.

Proof of Theorem 4.4. Let τ be defined as above. We say a user is a good user if $||\tilde{S}_u - \bar{S}_u||_2 \leq \frac{\tau}{2}$. Then under the assumption of the theorem, we have

$$||S_u - \bar{S}_u||_2 \leq \frac{\tau}{2} < \frac{1}{4}||\bar{S}_k - \bar{S}_{k'}||_2$$

for all good users. Let $\mathcal{I}$ be the set of good users and Corollary C.1 shows that the number of bad users $|\mathcal{I}^c| \leq \frac{K}{\log^2 n}$. The rest of the analysis is the same as the proof of Proposition 1 in [61]. We conclude that there exists a permutation π such that $|\mathcal{C}_k \Delta \hat{\mathcal{C}}_{\pi(k)}| \leq |\mathcal{I}^c| \leq \frac{K}{\log^2 n}$ and $\sum_k |\mathcal{C}_k \Delta \hat{\mathcal{C}}_{\pi(k)}| \leq 2|\mathcal{I}^c| \leq \frac{2K}{\log^2 n}$. □

C.7 Proof of Theorem 4.5

Let $D_{u,ij} = \mathbb{I}_{\left\{R_{u,ij}^{(2)}=1\right\}}$ and $D_{u,ji} = \mathbb{I}_{\left\{R_{u,ij}^{(2)}=-1\right\}}$ be the random variable indicating u's comparison result of i and j. Then the maximum likelihood estimator is defined as $\hat{\theta} = \arg\max_\gamma L(\gamma)$, where

$$L(\gamma) = \sum_{u,i,j} D_{u,ij} \log \frac{e^{\gamma_i}}{e^{\gamma_i} + e^{\gamma_j}}.$$

Further let $B_{u,ij} = B_{u,ji} = \mathbb{I}_{\left\{R^{(2)}_{u,ij} \neq 0\right\}}$ be the random variables indicating if u compared i and j. By definition of $L(\gamma)$,

$$\begin{aligned}
\frac{\partial L}{\partial \gamma_i} =& \sum_{u,j} (D_{u,ij} - B_{u,ij} \frac{e^{\gamma_i}}{e^{\gamma_i} + e^{\gamma_j}}) \\
\frac{\partial^2 L}{\partial \gamma_i^2} =& - \sum_{j} B_{ij} \frac{e^{\gamma_i} e^{\gamma_j}}{(e^{\gamma_i} + e^{\gamma_j})^2} \\
\frac{\partial^2 L}{\partial \gamma_i \partial \gamma_j} =& B_{ij} \frac{e^{\gamma_i} e^{\gamma_j}}{(e^{\gamma_i} + e^{\gamma_j})^2},
\end{aligned}$$

where $B_{ij} = \sum_u B_{u,ij}$. Let $\Delta = \hat{\theta} - \theta$. As $\hat{\theta}$ is the optimal solution,

$$\begin{aligned}
0 \leq & L(\hat{\theta}) - L(\theta) \\
= & \langle \nabla L(\theta), \Delta \rangle + \frac{1}{2} \Delta^\top (\nabla^2 L(\gamma)) \Delta,
\end{aligned}$$

where the second step is by Taylor expansion and $\gamma = \theta + \lambda \Delta$ for some $\lambda \in [0, 1]$. By Cauchy-Schwartz inequality,

$$\begin{aligned}
||\nabla L(\theta)||_2 ||\Delta||_2 \geq & \frac{1}{2} \Delta^\top (-\nabla^2 L(\gamma) \Delta) \\
\geq & \frac{e^b}{2(e^b + 1)^2} \Delta^\top L_B \Delta,
\end{aligned}$$

where the second inequality follows from the quadratic form of a Laplacian and the fact that $|\theta_i - \theta_j| \leq b$ for any i, j.

Let $Z_{u,ij} = D_{u,ij} - B_{u,ij} \frac{e^{\gamma_i}}{e^{\gamma_i} + e^{\gamma_j}}$. First we bound $||\nabla L(\theta)||_2$. For each i,

$$\frac{\partial L}{\partial \theta_i} = \sum_{j, u \in C} Z_{u,ij} - \sum_{j, u \in C \setminus \hat{C}} Z_{u,ij} + \sum_{j, u \in \hat{C} \setminus C} Z_{u,ij}.$$

The first term is independent of $\hat{\mathcal{C}}$. For $u \in \mathcal{C}$, $\mathbb{E}[Z_{u,ij}] = 0$ and $\mathrm{Var}[Z_{u,ij}] \leq \frac{1-\epsilon}{2}$. By Bernstein's inequality, with high probability for large m,

$$|\sum_{j, u \in C} Z_{u,ij}| \leq C_1 \sqrt{(1-\epsilon) K m \log m}.$$

For the next two terms, note that the matrix B only depends on Ω_2 but

not the comparison results, thus is independent of $R^{(1)}$ or $\hat{\mathcal{C}}$. As $B_{u,ij}$ are independent Bernoulli random variables with parameter $1-\epsilon$, with high probability for large m,

$$|-\sum_{j,u\in C\setminus\hat{C}} Z_{u,ij}+\sum_{j,u\in\hat{C}\setminus C} Z_{u,ij}|\leq\sum_{j,u\in I^C} B_{u,ij}\leq C_2(1-\epsilon)m\frac{K}{\log^2 n}.$$

Under the assumption, we conclude that $|\frac{\partial L}{\partial\theta_i}|\leq C_2(1-\epsilon)m\frac{K}{\log^2 n}$. Therefore,

$$||\nabla L(\theta)||_2\leq C_2(1-\epsilon)m^{3/2}\frac{K}{\log^2 n}.$$

Next we bound $\Delta^\top L_B\Delta$. Again by the fact that B is independent of $R^{(1)}$, we can simply follow the proof of Theorem 4 in [12] and get

$$\Delta^\top L_B\Delta\geq\frac{1}{4}(1-\epsilon)Km||\Delta||_2^2,$$

with high probability for large m. Combining the above results, we get the upper bound on $||\Delta||_2$

$$||\Delta||_2\leq\frac{C_3(e^b+1)^2}{e^b}\frac{\sqrt{m}}{\log^2 n}.$$

On the other hand, similar to the proof of Lemma C.1, we can show that $||\theta||_2\geq\frac{\sqrt{mb^2}}{4}$. Therefore, we get

$$\frac{||\hat{\theta}-\theta||_2}{||\theta||_2}=\frac{||\Delta||_2}{||\theta||_2}\leq\frac{(e^b+1)^2}{be^b}\frac{C}{\log^2 n}.$$

REFERENCES

[1] O. Banerjee, L. El Ghaoui, and A. d'Aspremont, "Model selection through sparse maximum likelihood estimation for multivariate Gaussian or binary data," *Journal of Machine Learning Research (JMLR)*, vol. 9, pp. 485–516, June 2008.

[2] R. Wu, R. Srikant, and J. Ni, "Learning loosely connected Markov random fields," *Stochastic Systems*, vol. 3, pp. 362–404, 2013.

[3] P. Netrapalli, S. Banerjee, S. Sanghavi, and S. Shakkottai, "Greedy learning of Markov network structure," in *Allerton Conf. on Communication, Control and Computing*, 2010.

[4] A. Anandkumar, V. Y. F. Tan, and A. S. Willsky, "High-dimensional structure estimation in Ising models: tractable graph families," 2011. [Online]. Available: http://arxiv.org/abs/1107.1736v1

[5] N. P. Santhanam and M. J. Wainwright, "Information-theoretic limits of selecting binary graphical models in high dimensions," *IEEE Transactions on Information Theory*, vol. 58, no. 7, pp. 4117–4134, 2012.

[6] S. T. Aditya, O. Dabeer, and B. Dey, "A channel coding perspective of collaborative filtering," *IEEE Transactions on Information Theory*, vol. 57, no. 4, pp. 2327–2341, 2011.

[7] K. Barman and O. Dabeer, "Analysis of a collaborative filter based on popularity amongst neighbors," *IEEE Transactions on Information Theory*, vol. 58, no. 12, pp. 7110–7134, 2012.

[8] M. A. Davenport, Y. Plan, E. van den Berg, and M. Wootters, "1-bit matrix completion," 2012. [Online]. Available: http://arxiv.org/abs/1209.3672

[9] V. Chandrasekaran and M. I. Jordan, "Computational and statistical tradeoffs via convex relaxation," *PNAS*, vol. 110, no. 13, pp. E1181–E1190, 2013.

[10] R. A. Bradley and M. E. Terry, "Rank analysis of incomplete block designs: I. the method of paired comparisons," *Biometrika*, vol. 39, no. 3/4, 1952.

[11] D. R. Hunter, "MM algorithms for generalized Bradley-Terry models," *The Annals of Statistics*, vol. 32, pp. 384–406, 2004.

[12] S. Negahban, S. Oh, and D. Shah, "Rank centrality: Ranking from pair-wise comparisons," 2014. [Online]. Available: http://arxiv.org/abs/1209.1688

[13] G. Bresler, E. Mossel, and A. Sly, "Reconstruction of Markov random fields from samples: Some observations and algorithms," in *APPROX-RANDOM*, 2008, pp. 343–356.

[14] A. Anandkumar, V. Tan, and A. Willsky, "High dimensional structure learning of Ising models on sparse random graphs," 2010. [Online]. Available: http://arxiv.org/abs/1011.0129

[15] P. Ravikumar, M. J. Wainwright, and J. Lafferty, "High-dimensional Ising model selection using l_1-regularized logistic regression," *Annals of Statistics*, vol. 38, no. 3, pp. 1287–1319, 2010.

[16] N. Alon and J. H. Spencer, *The Probabilistic Method*. New York: Wiley, 1992.

[17] J. Zhang, H. Liang, and F. Bai, "Approximating partition functions of the two-state spin system," *Inf. Process. Lett.*, vol. 111, pp. 702–710, July 2011.

[18] A. Ray, S. Sanghavi, and S. Shakkottai, "Greedy learning of graphical models with small girth," in *Allerton Conf. on Communication, Control and Computing*, 2012.

[19] C. Uhler, G. Raskutti, P. Bühlmann, and B. Yu, "Geometry of faithfulness assumption in causal inference," 2012. [Online]. Available: http://arxiv.org/abs/1207.0547

[20] J. D. Esary, F. Proschan, and D. W. Walkup, "Association of random variables, with applications," *Annals of Mathematical Statistics*, vol. 38, pp. 1466–1473, 1967.

[21] T. M. Liggett, "Stochastic models for large interacting systems and related correlation inequalities," *Proceedings of the National Academy of Sciences*, vol. 107, no. 38, pp. 16 413–16 419, Sep. 2010.

[22] E. Mossel and A. Sly, "Rapid mixing of Gibbs sampling on graphs that are sparse on average," in *Proceedings of the nineteenth annual ACM-SIAM symposium on Discrete algorithms*, ser. SODA '08, 2008, pp. 238–247.

[23] A. Montanari and J. A. Pereira, "Which graphical models are difficult to learn?" in *Advances in Neural Information Processing Systems 22*, 2009, pp. 1303–1311.

[24] S. Boyd and L. Vandenberghe, *Convex Optimization*. New York, NY, USA: Cambridge University Press, 2004.

[25] C. J. Quinn, N. Kiyavash, and T. P. Coleman, "Directed information graphs," *CoRR*, vol. abs/1204.2003, 2012. [Online]. Available: http://arxiv.org/abs/1204.2003

[26] T. Hastie, R. Tibshirani, and J. Friedman, *The Elements of Statistical Learning*. New York, NY, USA: Springer, 2001.

[27] S. Fortunato, "Community detection in graphs," *Physics Reports*, vol. 486, no. 3-5, pp. 75–174, 2010.

[28] Y. Chen, S. Sanghavi, and H. Xu, "Clustering sparse graphs." in *NIPS*, 2012, pp. 2213–2221.

[29] Y. Chen, A. Jalali, S. Sanghavi, and H. Xu, "Clustering partially observed graphs via convex optimization," in *ICML*, 2011. [Online]. Available: http://arxiv.org/abs/1104.4803

[30] F. McSherry, "Spectral partitioning of random graphs," in *42nd IEEE Symposium on Foundations of Computer Science*, Oct. 2001, pp. 529 – 537.

[31] K. Rohe, S. Chatterjee, and B. Yu, "Spectral clustering and the high-dimensional stochastic blockmodel," *The Annals of Statistics*, vol. 39, no. 4, pp. 1878–1915, 2011.

[32] S.-Y. Yun and A. Proutiere, "Community detection via random and adaptive sampling," 2014. [Online]. Available: http://arxiv.org/abs/1402.3072

[33] Y. Chen and J. Xu, "Statistical-computational tradeoffs in planted problems and submatrix localization with a growing number of clusters and submatrices," 2014. [Online]. Available: http://arxiv.org/abs/1402.1267

[34] D.-C. Tomozei and L. Massoulié, "Distributed user profiling via spectral methods," *SIGMETRICS Perform. Eval. Rev.*, vol. 38, no. 1, pp. 383–384, June 2010.

[35] J. A. Hartigan, "Direct clustering of a data matrix," *Journal of the American Statistical Association*, vol. 67, no. 337, pp. 123–129, 1972.

[36] Y. Cheng and G. M. Church, "Biclustering of expression data," *Proceedings of International Conference on Intelligent Systems for Molecular Biology (ISMB)*, vol. 8, pp. 93–103, 2000.

[37] S. C. Madeira and A. L. Oliveira, "Biclustering algorithms for biological data analysis: A survey," *IEEE/ACM Trans. Comput. Biol. Bioinformatics*, vol. 1, no. 1, pp. 24–45, Jan. 2004.

[38] S. Busygin, O. Prokopyev, and P. M. Pardalos, "Biclustering in data mining," *Computers and Operations Research*, vol. 35, no. 9, pp. 2964 – 2987, 2008.

[39] M. Kolar, S. Balakrishnan, A. Rinaldo, and A. Singh, "Minimax localization of structural information in large noisy matrices," in *NIPS*, 2011.

[40] S. Balakrishnan, M. Kolar, A. Rinalso, and A. Singh, "Statistical and computational tradeoffs in biclustering," in *NIPS 2011 Workshop on Computational Trade-offs in Statistical Learning*, 2011.

[41] E. J. Candès and T. Tao, "The power of convex relaxation: Near-optimal matrix completion," *IEEE Trans. Inf. Theory*, vol. 56, no. 5, pp. 2053–2080, May 2010.

[42] E. J. Candès and B. Recht, "Exact matrix completion via convex optimization," *Found. Comput. Math.*, vol. 9, no. 6, pp. 717–772, Dec. 2009.

[43] B. Recht, "A simpler approach to matrix completion," *J. Mach. Learn. Res.*, vol. 12, pp. 3413–3430, Dec. 2011.

[44] Y. Koren, R. Bell, and C. Volinsky, "Matrix factorization techniques for recommender systems," *Computer*, vol. 42, no. 8, pp. 30–37, 2009.

[45] P. Jain, P. Netrapalli, and S. Sanghavi, "Low-rank matrix completion using alternating minimization," in *Proceedings of the 45th Annual ACM Symposium on Symposium on Theory of Computing*, ser. STOC '13. New York, NY, USA: ACM, 2013, pp. 665–674.

[46] V. Chandrasekaran, S. Sanghavi, P. Parrilo, and A. Willsky, "Rank-sparsity incoherence for matrix decomposition," *SIAM Journal on Optimization*, vol. 21, no. 2, pp. 572–596, 2011.

[47] E. J. Candès, X. Li, Y. Ma, and J. Wright, "Robust principal component analysis?" *J. ACM*, vol. 58, no. 3, pp. 11:1–11:37, June 2011.

[48] Y. Chen, A. Jalali, S. Sanghavi, and C. Caramanis, "Low-rank matrix recovery from errors and erasures," *IEEE Transactions on Information Theory*, vol. 59, no. 7, pp. 4324–4337, 2013.

[49] R. Kannan and S. Vempala, "Spectral algorithms," *Found. Trends Theor. Comput. Sci.*, vol. 4, Mar. 2009.

[50] S. Ji and J. Ye, "An accelerated gradient method for trace norm minimization," in *Proceedings of the 26th Annual International Conference on Machine Learning*, ser. ICML '09, 2009, pp. 457–464.

[51] K.-C. Toh and S. Yun, "An accelerated proximal gradient algorithm for nuclear norm regularized linear least squares problems," *Pacific Journal of Optimization*, no. 6, pp. 615 – 640, 2010.

[52] J.-F. Cai, E. J. Candès, and Z. Shen, "A singular value thresholding algorithm for matrix completion," *SIAM J. on Optimization*, vol. 20, no. 4, pp. 1956–1982, Mar. 2010.

[53] R. J. Leake, "A method for ranking teams: With an application to college football," *Management Science in Sports*, pp. 27–46, 1976.

[54] X. Jiang, L.-H. Lim, Y. Yao, and Y. Ye, "Statistical ranking and combinatorial Hodge theory," *Math. Program.*, vol. 127, no. 1, pp. 203–244, Mar. 2011.

[55] A. N. Hirani, K. Kalyanaraman, and S. Watts, "Least squares ranking on graphs," 2011. [Online]. Available: http://arxiv.org/abs/1011.1716

[56] 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*, ser. WWW '01, 2001, pp. 613–622.

[57] M. Braverman and E. Mossel, "Noisy sorting without resampling," in *Proceedings of the Nineteenth Annual ACM-SIAM Symposium on Discrete Algorithms*, ser. SODA '08, 2008, pp. 268–276.

[58] F. Wauthier, M. Jordan, and N. Jojic, "Efficient ranking from pairwise comparisons," in *Proceedings of the 30th International Conference on Machine Learning (ICML-13)*, vol. 28, no. 3, May 2013, pp. 109–117.

[59] X. Chen, P. N. Bennett, K. Collins-Thompson, and E. Horvitz, "Pairwise ranking aggregation in a crowdsourced setting," in *Proceedings of the Sixth ACM International Conference on Web Search and Data Mining*, ser. WSDM '13, 2013, pp. 193–202.

[60] L. M. Busse, P. Orbanz, and J. M. Buhmann, "Cluster analysis of heterogeneous rank data," in *Proceedings of the 24th International Conference on Machine Learning*, ser. ICML '07, 2007, pp. 113–120.

[61] J. Xu, R. Wu, K. Zhu, B. Hajek, R. Srikant, and L. Ying, "Jointly clustering rows and columns of binary matrices: Algorithms and tradeoffs," *Proceedings of ACM Sigmetrics*, 2014.

[62] L. Massoulié and D.-C. Tomozei, "Distributed user profiling via spectral methods," 2011. [Online]. Available: http://arxiv.org/abs/1109.3318

[63] E. Arias-Castro and N. Verzelen, "Community detection in random networks," 2013. [Online]. Available: http://arxiv.org/abs/1302.7099

[64] Q. Berthet and P. Rigollet, "Optimal detection of sparse principal components in high dimension," *Ann. Statist.*, vol. 41, no. 1, pp. 1780–1815, 2013.

[65] Q. Berthet and P. Rigollet, "Complexity theoretic lower bounds for sparse principal component detection," *J. Mach. Learn. Res.*, vol. 30, pp. 1046–1066 (electronic), 2013.

[66] R. Krauthgamer, B. Nadler, and D. Vilenchik, "Do semidefinite relaxations really solve sparse PCA?" 2013. [Online]. Available: http://arxiv.org/abs/1306.3690

[67] Z. Ma and Y. Wu, "Computational barriers in minimax submatrix detection," 2013. [Online]. Available: http://arxiv.org/abs/1309.5914

[68] J. Liu, P. Musialski, P. Wonka, and J. Ye, "Tensor completion for estimating missing values in visual data." in *ICCV*, 2009, pp. 2114–2121.

[69] S. Gandy, B. Recht, and I. Yamada, "Tensor completion and low-n-rank tensor recovery via convex optimization," *Inverse Problems*, vol. 27, no. 2, Feb. 2011.

[70] M. Signoretto, L. De Lathauwer, and J. A. K. Suykens†, "Nuclear norms for tensors and their use for convex multilinear estimation," 2010. [Online]. Available: ftp://wgs.esat.kuleuven.ac.be/sista/signoretto/Signoretto_nucTensors.pdf

[71] B. Vandereycken, "Low-rank matrix completion by Riemannian optimization," *SIAM Journal on Optimization*, 2013, accepted.

[72] A. Banerjee, C. Krumpelman, J. Ghosh, S. Basu, and R. J. Mooney, "Model-based overlapping clustering," in *Proceedings of the Eleventh ACM SIGKDD International Conference on Knowledge Discovery in Data Mining*, ser. KDD '05, 2005, pp. 532–537.

[73] Q. Fu and A. Banerjee, "Multiplicative mixture models for overlapping clustering." in *ICDM*, 2008. [Online]. Available: http://dblp.uni-trier.de/db/conf/icdm/icdm2008.html#FuB08 pp. 791–796.

[74] K. A. Heller and Z. Ghahramani, "A nonparametric Bayesian approach to modeling overlapping clusters," in *Eleventh International Conference on Artificial Intelligence and Statistics (AISTATS-2007)*, 2007. [Online]. Available: http://learning.eng.cam.ac.uk/zoubin/papers/HelGha07over.pdf

[75] G. Cleuziou, "A generalization of k-means for overlapping clustering," 2007. [Online]. Available: http://www.univ-orleans.fr/lifo/prodsci/rapports/RR/RR2007/RR-2007-15.pdf

[76] A. P. Surez, J. F. M. Trinidad, J. A. Carrasco-Ochoa, and J. E. Medina-Pagola, "Oclustr: A new graph-based algorithm for overlapping clustering." *Neurocomputing*, vol. 121, pp. 234–247, 2013.

[77] A. Anandkumar, R. Ge, D. Hsu, and S. Kakade, "A tensor spectral approach to learning mixed membership community models." in *COLT*, vol. 30, 2013, pp. 867–881.

[78] D. Weitz, "Counting independent sets up to the tree threshold," in *Proceedings of the Thirty-Eighth Annual ACM Symposium on Theory of Computing*, ser. STOC '06, 2006, pp. 140–149.

[79] K. Jung and D. Shah, "Local approximate inference algorithms," 2006. [Online]. Available: http://arxiv.org/abs/cs/0610111

[80] N. Berger, C. Kenyon, E. Mossel, and Y. Peres, "Glauber dynamics on trees and hyperbolic graphs," *Probability Theory and Related Fields*, vol. 131, pp. 311–340, 2005.

[81] T. M. Cover and J. A. Thomas, *Elements of Information Theory*. Wiley-Interscience, 1991.

[82] J. Bourgain, V. H. Vu, and P. M. Wood, "On the singularity probability of discrete random matrices," *Journal of Functional Analysis*, vol. 258, no. 2, pp. 559–603, 2010.

[83] V. H. Vu, "Spectral norm of random matrices," *Combinatorica*, vol. 27, no. 6, pp. 721–736, 2007.

[84] J. Tropp, "User-friendly tail bounds for sums of random matrices," *Foundations of Computational Mathematics*, vol. 12, no. 4, pp. 389–434, 2012.

[85] D. Gross, "Recovering low-rank matrices from few coefficients in any basis," *IEEE Trans. Inf. Theor.*, vol. 57, no. 3, pp. 1548–1566, Mar. 2011.