Towards Provable and Scalable Machine Learning

Jin Lu

University of Connecticut - Storrs, jin.lu@uconn.edu

Follow this and additional works at: https://opencommons.uconn.edu/dissertations

Recommended Citation

https://opencommons.uconn.edu/dissertations/2194
Towards Provable and Scalable Machine Learning

Jin Lu, Ph.D.
University of Connecticut, 2019

In the recent decade, machine learning has been substantially developed and has demonstrated great success in various domains such as web search, computer vision, and natural language processing. Despite of its practical success, many of the applications involve solving NP-hard problems based on heuristics. It is challenging to analyze whether a heuristic scheme has any theoretical guarantee. In this dissertation, we show that if a certain structure occurs in sample data, it is possible to solve the related problem with provable guarantees. We propose to employ granular data structure, e.g., sample clusters or features describing an aspect of the sample, to design new statistical models and algorithms for two learning problems. The first learning problem deals with the commonly-encountered missing data issue by formulating it as a matrix completion problem. When side features describing the data entities are available, we propose a new convex formulation to construct a bilinear model that infers the missing values based on the side features. This approach can be proved that with a much lower sampling rate than that of
the classic matrix completion methods, it can exactly recover or epsilon-recover missing values, depending on whether the side features are corrupted. A novel linearized alternating direction method of multipliers is developed to efficiently solve the proposed convex formulation. For the second learning problem, we build a new generative adversarial network (GAN) to generate data that follow a distribution much closer to the true data distribution than the standard GAN when the data contains underlying clusters. The proposed model consists of multiple smaller GANs as components, each corresponding to a data cluster identified automatically during the construction of the GAN. This GAN approach can recover the true distribution for every cluster if an appropriate Kolmogorov regularization is used. If the GAN complexity is regularized by smoothness with a parameter epsilon, we prove that GAN model can approximate the true data distribution with an epsilon tolerance. We use the Adaptive Momentum (ADAM) algorithm to optimize this model with scalability. The proposed two approaches essentially bring new insights and suggest new methods for provable and scalable machine learning.
Towards Provable and Scalable Machine Learning

Jin Lu

Master, University of Connecticut, USA, 2019

Master, Xi’an Jiaotong University, China, 2014

Bachelor, Northwestern Polytechnical University, China, 2010

A Dissertation

Submitted in Partial Fulfillment of the

Requirements for the Degree of

Doctor of Philosophy

at the

University of Connecticut

2019
Copyright by

Jin Lu
APPROVAL PAGE

Doctor of Philosophy Dissertation
Towards Provable and Scalable Machine Learning

Presented by
Jin Lu, M.S., M.E.

Major Advisor
________________________________________
Jinbo Bi

Associate Advisor
________________________________________
Alexander Russell

Associate Advisor
________________________________________
Sanguthevar Rajasekaran

University of Connecticut

2019
I owe thanks to numerous people for the successful completion of this thesis.

First, to my supervisor Jinbo Bi, for believing in me and supporting me to achieve ever greater personal and professional growth. I will be forever grateful for your dedication to myself and the other students, and truly felt that you were always watching out for our well being both in times of success and struggle. With your help, I have immensely matured as a researcher, professional, and human being.

To my committee members, Dr. Alexander Russell and Dr. Sanguthevar Rajasekaran. Thank you all for your insightful comments and encouragement, but also for the hard question which guided me to widen my research from various perspectives.

To Jiangwen Sun, for all his help with edits and comments in the early drafts of my several papers. Your help has been critical to my works, and you’ve been a great friend who has provided support to me both in and out of the lab.

To Tingyang Xu, Chao Shang, Guannan Liang, Xinyu Wang, and Xin Wang, for invaluable assistance in my research projects, and help with research strategy, project and time management, and improving my writing and programming skills.

To Arron Palmer, Qianqian Tong, and Guoqing Chao for your help on the formulation and insights in letting me be more productive than I would have been on
my own.

To my Mom and Dad, for your unwavering love and support through my long time in school. You have always been wonderful for encouraging me to reach high and try hard. Your advice and encouragement have always given me drive to accomplish my dreams.

Finally, to my wife, Fei Dou. Fei, who has been my best friend and great companion, loved, supported, encouraged, entertained, and helped me get through this study journey in the most positive way.
# TABLE OF CONTENTS

1. Introduction ................................................................................. 1

2. Matrix Completion with Side Information ............................... 6
   2.1 Introduction ........................................................................ 6
   2.2 The Proposed Formulation ...................................................... 12
   2.3 Recovery Analysis ................................................................. 17
      2.3.1 Sampling Rate for Exact Recovery ................................. 17
      2.3.2 Sampling Rate for $\epsilon$-Recovery .............................. 41
   2.4 Adaptive LADMM Algorithm ............................................... 51
      2.4.1 Algorithm .................................................................. 51
      2.4.2 Convergence Analysis ................................................... 56
   2.5 Experimental Results ........................................................... 63
      2.5.1 Synthetic Datasets ......................................................... 64
      2.5.2 Benchmark Datasets ..................................................... 66
      2.5.3 Case Study: Inference of missing diagnostic criteria of substance use disorders ......................................................... 68
   2.6 Summary ............................................................................. 76

3. Statistical Approximation and Learning of Kolmogorov CoupledNet ................................................................. 77
   3.1 Background ......................................................................... 77
LIST OF FIGURES

2.1 The Comparison of RMSE for Experiments I, II, and III 65
2.2 The heatmap of True G and Recovered G matrices in Experiment I 66
2.3 HeatMap of G for MovieLens 69
2.4 HeatMap of sgn(G) log(|G|) for NCI-DREAM for a better illustration 69
2.5 The recovered G by our method for the Cocaine-Opioid SUD dataset.

Columns C1-C11 represent 11 CUD diagnostic criteria, columns O1-O11 represent 11 OUD diagnostic criteria. C1/O1: Larger or longer Cocaine/Opioid use than intended; C2/O2: Failed efforts to stop on Cocaine/Opioid; C3/O3: Much time spent in Cocaine/Opioid related activities; C4/O4: Strong desire to use Cocaine/Opioid; C5/O5: Cocaine/Opioid effect interfered with life; C6/O6: Cocaine/Opioid use despite of its interference; C7/O7: Major activities reduced by Cocaine/Opioid use; C8/O8: Physical hazard caused by Cocaine/Opioid use; C9/O9: Cocaine/Opioid use knowing it threatening health; C10/O10: Cocaine/Opioid tolerance; C11/O11: Cocaine/Opioid withdrawal syndrome 72
2.6 The top 30 rows of the recovered $G$ by our method for the Cocaine-Opioid SUD dataset. Columns correspond to the diagnostic criteria for CUD and OUD whereas rows correspond to the candidate genetic variants. The right-hand side gives the locations of these genetic variants and their p-values obtained in the GWAS.

2.7 Gene expression distribution (RPKM, Reads per Kilobase Million) of C8orf48 across human tissues.

3.1 Synthetic experiments

3.2 Generated results on MNIST(a), SIFA10(b) and COIL20(c).

3.3 SKC vs $\sigma$ of 10 generators on MNIST dataset.
LIST OF TABLES

2.1 The Comparison of RMSE values of different methods on real-world datasets. ........................................ 68

2.2 The comparison of imputation results by different methods on the Opioid-Cocaine SUD dataset. .............. 71

3.1 The Comparison of ACC values of different methods on real-world datasets. ........................................ 107
Chapter 1

Introduction

The momentum of research in modern machine intelligence and data analytics is towards learning from massive amounts of data. Learning via complex black-box models, such as deep learning, has seen a dramatic resurgence of popularity over the past few years because of their impressive performance on big-data problems. Despite their practical success, many of those machine learning methods are computationally intractable and solved on the basis of heuristics, leading to a lack of theoretical foundations and provable guarantees. As a result, there is a great need for developing scalable machine learning methods with theoretical rigor that bridge the gap between the theory and the practice.

The understanding of this gap could be revealed by the fact that computational intractability only refers to the worst case inputs. Therefore, it further raises several questions. Could we design new methods that work provably on “easier” input data? How to define and identify the ”easy” input data? Inspired by these questions, we propose practical machine learning methods with provable guarantees for several real-world learning problems in this proposal, under the
rationale that the input data has certain underlying structure.

We first consider the matrix completion problem, in which the data to be represented are in the 2nd order form such as user-to-user social networks and user-to-item recommender system. One typical example is the movie recommendation. In this problem, users submit ratings on a subset of movies in a database, and the vendor is required to provide recommendations based on the users preferences. Since users only rate a few movies, one would like to infer their preference for unrated items. A partially observed matrix can be defined where the rows index users and the columns index movies. With this definition, recent researches often formulate the problem as low-rank matrix completion, which recovers the matrix by assuming that similar users give similar ratings to the similar products. Besides this low-rank assumption on data, to impute missing entries, granular structures such as auxiliary or side features describing the row or column entities of the matrix are often available and useful as well. Matrix completion methods using these side features have been shown to reduce sample complexity from those classic ones that only use the observed data entries in the matrix. However, to recover a low rank data matrix, it is often assumed that the parameter matrix in their model of using the side features is also low rank, which is unnecessary. We propose a new learning formulation that constructs a bilinear model in terms of the interactions between the row and column side features to approximate the matrix entries, and requires the model parameter matrix to be sparse rather than low rank. It
is proved that when the side features span the latent feature space of the data matrix, the number of observed entries needed for an exact recovery of the data matrix is $O(\log n)$ where $n$ is the size of the matrix. When the side features are corrupted latent features of the matrix with a small perturbation, the proposed approach can achieve an $\epsilon$-recovery with $O(\log n)$ sample complexity. It maintains $O(n^{3/2})$ sample complexity similar to classic matrix completion methods if side features happen to be useless. A linearized Lagrangian algorithm is developed with global convergence guarantee at a linear convergence rate. Both simulation results and real-world data analyses show that the new approach outperforms the state of the art.

Then, we extend our understanding of provable machine learning to the problem of probability density estimation, which corresponds to a typical problem in approximation theory and many engineering fields that aims to find an approximate solution to the problem

$$\min_{q \in \mathcal{Q}} \text{dist}(p, q)$$

given a hypothesis space $\mathcal{Q}$ of functions and given $p$ is the function one needs to approximate. Recent needs of machine learning call for consideration of the above infinite dimensional optimization problem. Especially the mixture models for probability density estimation, as a crucial topic in unsupervised learning, are often developed to discretize a complex probability density function (pdf) $p$ into several distributions, each of which corresponding to a cluster via new parame-
terization. As a powerful discretization technique to reveal the patterns of data without labels or humans' representation, mixture models has been successfully applied to various research fields. Existing explicit mixture models typically approach the targeted distribution by assuming that pdfs in $Q$ follow specific analytic forms $[1, 2, 3, 4]$, or find an intermediate function embedding the samples that follow certain analytical distributions in the embedding space. Though being successful in tractable cases, as data explodes rapidly in the most recent decades, the former methodology might restrict a model’s capacity to approximate clusters with complicated distributions. It is challenging as well to determine the embedding function for the later methodology, which depends highly on the expert domain knowledge $[5]$ and model selection $[6]$. Alternatively, aiming to implicitly approximate the probabilistic distribution, generative adversarial models have been enjoying considerable success as a framework of implicit germinative models for numerous types of tasks and datasets in recent years. However, quantifying the expressive power of deep generative models has not been substantially studied.

In this proposal, we propose a novel metric for GAN’s capacity from approximation theory, named as Surrogate Kolmogorov Complexity. By assuming the local grouping structure (mixtures) of the data, we propose a new direction to approximate the data distribution by functions drawn from a function space $Q$ where $Q$ is generalized to include functions induced by a generative adversarial network and we apply the Surrogate Kolmogorov Complexity to effectively control the capac-
ity of $Q$. The resultant model, named Kolmogorov Coupled Nets, can provably recover the true mixtures as well as approximating both individual mixtures’ distributions and the overall data distribution with a controllable tolerance, showing that the new way offers better expressive power of $Q$. 
2.1 Introduction

Matrix completion is an problem of recovering or imputing missing entries in a matrix. It is commonly used to complete a data matrix when prior knowledge about the data is available, such as correlated data that lead to a low rank data matrix. It has become a fundamental technique in many engineering and scientific domains, such as information retrieval [7], collaborative filtering [8], computer vision [9, 10, 11], recommender systems [12], control systems [13] and signal processing [14]. Among the numerous applications, creating a movie recommender system has been an exemplar illustration based on a numerical matrix filled with the ratings from a group of users for a set of movies. Each row of the matrix represents a user, and each column represents a movie. A recommender system aims to perform the task of predicting the preferences of users to certain movies based on limited observed ratings or reviews.

A certain relationship among data entries in the matrix is typically assumed.
Then the observed data and this relationship form a basis to impute missing entries. Otherwise, a matrix completion method can have infinitely possible solutions to guess the full matrix. For instance, a recommender system is created based on an assumption that similar users give similar ratings to similar products, which leads to rating correlations. These correlations are the basis for drawing the inferences regarding missing ratings. Thus, matrix completion is formulated as an optimization problem to impute the missing entries with values that yield a minimal rank of the completed matrix. However, the rank minimization problem is not convex and can be difficult to solve. By relaxing the matrix rank to its convex surrogate, which is the nuclear norm of the matrix, a convex optimization problem can be formulated. As in classical low-rank matrix completion methods [15, 16, 17], it solves the following optimization problem

$$\min_{E} \|E\|_* \quad \text{subject to} \quad R_\Omega(E) = R_\Omega(F),$$

where $F \in \mathbb{R}^{n_1 \times n_2}$ is the partially observed low-rank matrix (with a rank of $r$) that needs to be recovered, $\Omega \subseteq \{1, \cdots, n_1\} \times \{1, \cdots, n_2\}$ be the set of indexes where the corresponding components in $F$ are observed, the mapping $R_\Omega(M) : \mathbb{R}^{n_1 \times n_2} \to \mathbb{R}^{n_1 \times n_2}$ gives another matrix whose $(i, j)$-th entry is $M_{i,j}$ if $(i, j) \in \Omega$, or 0 otherwise, and $\|M\|_*$ computes the nuclear norm of $M$ which is the sum of all singular values of $M$. Eq. (3.1) is now convex, and the advantage of working with a convex problem is that any local minimum is in fact the global minimum and thus they can be solved exactly and efficiently. Researchers have developed
first order algorithms, including the singular value thresholding (SVT) algorithm [18] for the formulation (3.1) and several variants of the proximal gradient method for solving (3.1) when noise is concerned in the observation of $F$ [16, 19]. The proximal gradient method has been established for the noisy matrix completion with linear convergence [20, 21].

Alternatively, instead of assuming that the matrix $E$ is low rank, studies in [22, 23] consider a decomposition of $F$ into a product of two smaller matrices $AB^T$, where $A \in \mathbb{R}^{n_1 \times r}$, $B \in \mathbb{R}^{n_2 \times r}$. In this case, the optimization problem is often formulated as the minimization of the Euclidean distance between $AB^T$ and $F$ on the observed entries, and then the low-rank requirement is automatically fulfilled (i.e., $\text{rank}(F) \leq r$). Such a matrix factorization model has long been used in many other areas, such as principle component analysis (PCA) [24]. It has several noticeable advantages in practice, such as, the compact representation of the unknown matrix, which reduces the storage space and the per-iteration computation cost. It is also elaborated that the factorization model can be easily modified to incorporate additional application-specific requirements [25]. However, there are two fundamental challenges of the factorization based models: (1) obtaining the best rank $r$ is often intractable; (2) the non-convexity of the related optimization problem makes it difficult to guarantee that a global minimum is acquired in the recovering of the data matrix, unless the optimization starts from an initialization that is close enough to the global minimum.
Early theoretical analysis [26, 27, 28] proves that $O(nr \log^2 n)$ entries are sufficient for an exact recovery of $F$ if the observed entries are uniformly sampled at random where $n = \max\{n_1, n_2\}$. Another work [29] deals with noisy data observations in matrix completion and reveals a similar sampling rate. The approach in [17] studies the matrix recovery from very few observed data entries and derives an effective algorithm that can retrieve $F$ with a high probability and a small relative mean square error using $O(rn)$ observed entries and further with $O(n \log n)$ data entries, it can retrieve the exact $F$. However, these bounds assume that the observed entries are sampled uniformly at random. For sample complexity in a distribution-free manner, Shamir and Shalev-Shwartz [30] recently showed that $O(n^{3/2})$ entries are sufficient for $\epsilon$-recovery, which means the expected recovery error for each entry is less than a small tolerance $\epsilon > 0$.

Recent studies start to explore side information for matrix completion [31, 32, 33, 34]. For example, to infer the missing ratings in a user-movie rating matrix, descriptors of the users and movies are often known and may help to build a content-based recommender system. For instance, kids tend to like cartoons, so the age of a user likely interacts with the cartoon feature of a movie. When few ratings are known, this side information could be the main source for completing the matrix. Based on empirical studies, several works found that side features are helpful [32, 33, 34, 35] via matrix factorization formulations; Berg et al. [37] imposes a Graph Convolutional Network representing the bipartite graph between
features of users and movies [37]. Monti et al. [38] constructs two GCNs, each extracting features of users and movies, respectively; however, all the mentioned methods above involve highly non-convex optimization, in consequence of extreme difficulty of theoretical recovery analysis.

Three recent methods have focussed on convex nuclear-norm regularized objectives, which leads to theoretical guarantees on matrix recovery [39, 40, 41]. These methods all construct an inductive model $XG^TY^T$ so that $R_\Omega(XG^TY^T) = R_\Omega(F)$ where the side matrices $X$ and $Y$ consist of side features, respectively, for the row entities (e.g., users) and column entities (e.g., movies) of a (rating) matrix. This inductive model has a parameter matrix $G$ which is either required to be low rank [39] or to have a minimal nuclear norm $\|G\|_*$ [40]. Recovering $G$ of a (usually) smaller size is argued to be easier than directly recovering the matrix $F$. With a very strong assumption on ‘perfect’ side information, i.e., both $X$ and $Y$ are orthonormal matrices and respectively in the latent column and row space of the matrix $F$, the method [40] is proved to have much reduced sample complexity $O(\log n)$ for an exact recovery of $F$. Because most side features $X$ and $Y$ are not perfect in practice, a very recent work [41] proposes to use a residual matrix $N$ to handle the noisy side features. This method constructs an inductive model $XG^TY^T + N$ to approximate $F$ and requires both $G$ and $N$ to be low rank, or have a low nuclear norm. It uses the nuclear norm of the residual to quantify the usefulness of side information, and proves $O(\log n)$ sampling rate for an $\epsilon$-
recovery when \( X \) and \( Y \) span the full latent feature space of \( F \), and \( o(n) \) sample complexity when \( X \) and \( Y \) have noise features not from the latent space of \( F \). An \( \epsilon \)-recovery is defined as that the expected discrepancy between the predicted matrix and the true matrix is less than an arbitrarily small \( \epsilon > 0 \) under a certain probability.

Comparing with the recent works, our contributions are summarized as follows:

(i) We propose a new formulation in [12] that estimates both \( E \) and \( G \) by imposing a nuclear-norm constraint on \( E \) but a general regularizer on \( G \), e.g., the sparse regularizer \( \| G \|_1 \). The proposed model has theoretical recovery guarantees depending on the quality of the side features: (1) when \( X \) and \( Y \) are full column rank and span the entire latent feature space of \( F \) (but are not required to satisfy the much stronger condition of being orthonormal as in [40]), \( O(\log n) \) observations are still sufficient for our method to achieve an exact recovery of \( F \). (2) When the side matrices are not full rank and corrupted from the original latent features of \( F \), i.e., \( X \) and \( Y \) do not contain enough basis to exactly recover \( F \), \( O(\log N) \) observed entries are sufficient for an \( \epsilon \)-recovery.

(ii) A novel linearized alternating direction method of multipliers (LADM-M) is developed to efficiently solve the proposed convex formulation. Existing methods use side information are solved by a standard block-wise coordinate descent algorithm; This algorithm has convergence guarantee to a global minimum.
when the optimization problem is strictly convex, or converge to a stationary point when the problem is non-convex and non-differentiable while each subproblem has unique solution [43]. Our LADMM achieves the global minimum and has a linear convergence rate [44].

(iii) Prior methods focus on the recovery of $F$, and little light has been shed to understand whether the interactive model of $G$ can be retrieved. Because of the explicit use of $E$ and $G$, our method aims to directly recover both. The unique $G$ in the case of exact recovery of $F$ can be obtained by our algorithm. When $G$ is not unique in the $\epsilon$-recovery case, our algorithm converges to a point in the optimal solution set.

(iv) Our proposed method demonstrates high effectiveness of integrating genotype data with other relevant sources of information for imputing missing phenotypes than other matrix completion methods [45, 46]. As an additional benefit, the proposed method constructs a bi-linear predictive model that can help to identify important interactions that link specific genotypes to diagnostic criteria.

2.2 The Proposed Formulation

Assume that there are $a$ side features $x$ describing a user (row entity of $F$) and $b$ side features $y$ describing a movie (column entity of $F$). Thus, two side feature matrices $X$ of size $n_1 \times a$ and $Y$ of size $n_2 \times b$ are available. To complete $F$, we
propose to build a predictive model, as a function of \( X \) and \( Y \), that is constructed from the observed components of \( F \) to predict the missing ones. This is different from the transductive model commonly used in the method of Eq.(3.1) where the missing entities are directly filled rather than creating an explicit inference model.

The advantage of creating a model is that the model can then be used to predict future (expanded) entries of the matrix. We can simply start with a linear model:

\[
f = x^T u + y^T v + z,
\]

where \( u, v, \) and \( z \) are model parameters. In real life applications, interactive terms between the features in \( X \) and \( Y \) can be very important. For example, kids tend to rate animation pictures high, which means that the interactive term of age and animation genre can be a good predictor. Male users tend to rate action movies higher than female users, indicating gender-genre interaction. These interactive terms can be informative when predicting the ratings for specific movies.

Therefore, a linear model considering no interactive terms may have low predictive power for missing entries. We then add interactive terms by introducing an interaction matrix \( H^{a \times b} \) to the predictive model, which can be written as:

\[
f = x^T H y + x^T u + y^T v + z.
\]

By defining \( \bar{x} = [x^T \ 1] \), \( \bar{y} = [y^T \ 1] \) and \( G^{(d_1 = a + 1) \times (d_2 = b + 1)} = \begin{pmatrix} H & u \\ v^T & g \end{pmatrix} \)

the above model can be simplified to: \( f = \bar{x} G \bar{y}^T \). The following optimization
The problem can be solved to obtain the model parameter \( G \).

\[
\min_{G, E} \ g(G) + \lambda_E \|E\|_*,
\]

subject to \( XGY^T = E, R_\Omega(E) = R_\Omega(F) \),

where \( E \) is a completed version of \( F \), \( X^{n_1 \times d_1} \) and \( Y^{n_2 \times d_2} \) are two matrices that are created by augmenting one column of all ones to \( X \) and to \( Y \), respectively, and \( g(G) \) and \( \|E\|_* \) are used to incorporate the (sparsity) prior of \( G \) and (low rank) prior of \( E \). Because the side information data can be noisy and not all the features and their interactions are helpful to the prediction of \( F \), a sparse \( G \) is often expected. Our implementation has used \( g(G) = \|G\|_1 \). It is natural to impose low rank requirement on \( E \) because it is a completed version of a low rank matrix \( F \). The tuning parameter \( \lambda_E \) is used to balance the two priors in the objective.

Without loss of generality and for notational convenience, we simply use \( X \) and \( Y \) to denote the augmented matrices. Denote the Frobenius norm of a matrix by \( \|A\|_F = \sum_{i=1}^{n_1} \sum_{j=1}^{n_2} |a_{i,j}|^2 \), the one norm by \( \|A\|_1 = \sum_{i=1}^{n_1} \sum_{j=1}^{n_2} |a_{i,j}| \), the nuclear norm by \( \|A\|_* = \text{trace}(\sqrt{A^T A}) \) and the spectral norm by \( \|A\| = \|A\|_2 = \max_{\lambda_i \in \Lambda} \lambda_i \) where \( \lambda_i \) is the \( i \)-th singular value in the set of all singular values \( \Lambda \). The spectral norm of an operator \( T(A) \) will be denoted by \( \|T\| = \max_{\|A\|_F \leq 1} \|T(A)\|_F \). To account for Gaussian noise, we relax the equality constraint \( XGY^T = E \) and replace it by minimizing their squared residual: \( \|XGY^T - E\|^2_F \) and solve the following optimization problem to obtain \( G \) and \( E \):
\[
\min_{G, E} \frac{1}{2} \|XGY^T - E\|_F^2 + \lambda_G g(G) + \lambda_E \|E\|_*,
\]
subject to \( R_{\Omega}(E) = R_{\Omega}(F) \).

(2.2)

where \( \lambda_G \) is another tuning parameter that, together with \( \lambda_E \), balances the three terms in the objective. Especially, the regularizer \( g(\cdot) \) in our formulation can take any general matrix norm and can be chosen according to any prior knowledge of \( G \).

In our experiments, we realized \( g(\cdot) \) to be the \( \ell_1 \)-norm of the matrix \( G \). In the next section when we analyze the recovery property, we show the sample complexity needed for exact recovery when \( \|G\|_1 \) is used in Eq.(2.2). In the \( \epsilon \)-recovery later discussed, as long as \( g(\cdot) \) satisfies \( \|M\|_* \leq C g(M), \forall M \), for a constant \( C \), so for instance \( g(\cdot) \) can be \( \|G\|_1 \), or \( \|G\|_F \), or \( \|G\|_2 \). Throughout this chapter, the matrices \( X \) (and \( Y \)) refer to, i.e., either the original \( X^{n_1 \times a} \) (and \( Y^{n_2 \times b} \)) or the augmented \( \tilde{X}^{n_1 \times d_1} \) (and \( \tilde{Y}^{n_2 \times d_2} \)) depending on the user-specified model. In our formulation \( \lambda_G \) influences the significance of the prior structure of \( G \). One can also observe that when \( \lambda_G \) is sufficiently large, the problem (2.2) regresses to the standard matrix completion problem in [26] without side information as problem (3.1), in addition to a regularizing term in terms of \( \|E\|_F \) equivalent to the upper-bound prior on \( \|E\|_F \). Thanks to Theorem 2, a useful guidance for the hyper-parameter selection that \( \lambda_G / \lambda_E = 1/n \) is given such that the recovery can be achieved theoretically with certain sample complexity proposed in Theorem 2. It suggests that our formulation is more generalizable and adoptable to the standard
matrix completion in the case of no access to useful side information.

Our formulation (2.2) is different from existing methods that make use of side information for matrix completion in many different ways. Existing methods [39, 40, 41] solve the problem by finding $\hat{H}$ that minimizes $\|H\|_*$ subject to $R_\Omega(XHY^T) = R_\Omega(F)$, but we expand it to include the linear term within the interactive model. The proposed model adds the flexibility to consider both linear and interactive terms, and allows the algorithm to determine the terms that should be used in the model by enforcing the sparsity in $H$ (or $G$). Because $E = XGY^T$, the rank of $G$ bounds that of $E$ from above. The existing methods all minimize the rank of $G$ (e.g. by minimizing $\|G\|_*$) to incorporate the prior of low rank $E$ (and thus low rank $F$) in their formulations. However, when the rank of $G$ is not properly chosen during the tuning of hyperparameters, it may not even be a sufficient condition to ensure low rank $E$ (if rank($E$) $\ll$ the pre-specified rank($G$)). It is easy to see that besides $G$ both low-rank $X$ and $Y$ can lead to low-rank $E$ as well, so a low-rank $G$ is not necessary to result in a low-rank $E$. Enforcing a low-rank condition on $H$ or $G$ may limit the search space of the interactive model and thus impairing the predictive performance on missing matrix entries, which are demonstrated in our empirical results.
2.3 Recovery Analysis

Let $E_0$ and $G_0$ be the two matrices such that $R_\Omega(F) = R_\Omega(E_0)$ and $E_0 = XG_0Y^T$.

In this section, we give our theoretical results on the sampling rate for achieving an exact recovery of $E_0$ and $G_0$ when $X$ and $Y$ are both full column rank (i.e., rank($X$) = $d_1$ and rank($Y$) = $d_2$) and span the latent feature space of $E_0$, and an $\epsilon$-recovery of $E_0$ when the two side matrices are corrupted and less informative.

2.3.1 Sampling Rate for Exact Recovery

In this section, we introduce the notion of matrix coherence that measures how singular matrix columns or rows are, or in other words, how uncorrelated any columns (or rows) in the matrix can be with all other columns (or rows). Intuitively, it might be easier to recover ‘incoherent’ matrices, i.e., matrices with relatively low coherence than those with high coherence. We give a few relevant definitions before presenting our results because coherence conditions are important for exact recovery of a matrix. Let $F = U\Sigma V^T$, $X = U_X \Sigma_X V_X^T$ and $Y = U_Y \Sigma_Y V_Y^T$ be the singular value decomposition of $F$, $X$ and $Y$, respectively. Without loss of generality, this decomposition only uses the non-zero eigenvalues and hence only the left and right singular vectors corresponding to non-zero eigenvalues are used in $U$ and $V$. Let

$$P_U = UU^T \in \mathbb{R}^{n_1 \times n_1} \quad P_X = U_X U_X^T = XV_X \Sigma_X^{-2} V_X^T X^T \in \mathbb{R}^{n_1 \times n_1}$$

$$P_Y = VV^T \in \mathbb{R}^{n_2 \times n_2} \quad P_Y = U_Y U_Y^T = YV_Y \Sigma_Y^{-2} V_Y^T Y^T \in \mathbb{R}^{n_2 \times n_2},$$

where $\Sigma$ is the diagonal matrix of singular values of $F$, $X$ or $Y$. The notation $X^T$ denotes the transpose of $X$. The matrices $P_U$, $P_X$, $P_Y$ and $P_Y$ are the projection matrices onto the column spaces of $U$, $X$, $Y$ and $Y$ respectively.
where $P_U$, $P_V$, $P_X$ and $P_Y$ are projector matrices that project a vector onto the subspaces spanned, respectively, by the columns in $U$, $V$, $X$, and $Y$. All projectors $P$ satisfy that $P^2 = P$.

Let $\mu_0$ and $\mu_1$ be the two coherence measurements for a single matrix $F$ and be defined as follows [26]:

$$
\mu_0 = \max \left( \frac{n_1}{r} \max_{1 \leq i \leq n_1} \|P_U e_i\|_2^2, \frac{n_2}{r} \max_{1 \leq j \leq n_2} \|P_V e_j\|_2^2 \right), \quad \mu_1 = \max_{i,j} \frac{n_1 n_2}{r} \left( [U V^T]_{i,j} \right)^2,
$$

where $e_i$ is the unit vector with the $i$th entry equal to 1, which also forms the coordinate axes when $i = 1, \cdots, n_1$. The coherence parameter $\mu_0$ measures the alignment between the column space spanned by $U$ (or the row space spanned by $V$) and the coordinate axes. Let $\mu_2$ be the coherence measurement between two matrices $X$ and $Y$ and be defined as:

$$
\mu_2 = \max \left( \max_{1 \leq i \leq n_1} \frac{n_1 \|x_i\|_2^2}{d_1}, \max_{1 \leq j \leq n_2} \frac{n_2 \|y_j\|_2^2}{d_2} \right),
$$

where $x$ and $y$ are the row vectors in $X$ and $Y$, respectively.

Using the above definitions, we first prove the following theorem that when $X$ and $Y$ are both full column rank, $(G_0, E_0)$ is the unique solution to Problem (2.2) under two deterministic assumptions $A1$ and $A2$ for $F$, $X$ and $Y$, and more precisely, for the projectors defined by these matrices. In several lemmas followed, we prove that with high probabilities, $A1$ and $A2$ hold. Then these probabilities help us identify a high probability, with which $(G_0, E_0)$ is the unique solution to (2.2) as long as there are $O(r \log n)$ observed entries in $F$ in Theorem 2. In other
words, with a sampling rate of $O(r \log n)$, our method can fully recover both $E_0$ and $G_0$ with a high probability when $X$ and $Y$ are full column rank.

**Theorem 1:** For any matrix $M \in \mathbb{R}^{n_1 \times n_2}$, we define two linear operators: $P_T : \mathbb{R}^{n_1 \times n_2} \to \mathbb{R}^{n_1 \times n_2}$ and $P_{T^\perp} : \mathbb{R}^{n_1 \times n_2} \to \mathbb{R}^{n_1 \times n_2}$ as follows:

$$P_T(M) = P_{MY}PM + PXMPY - PUMPV$$

$$P_{T^\perp}(M) = (PX - PU)(PY - PV).$$

Assuming that for any $M \in \mathbb{R}^{n_1 \times n_2}$ satisfying that $M \neq 0$, $R_\Omega(M) = 0$ and $M = PXMPY$, we have

$$A1 \quad \|P_T(M)\|_F \leq \zeta \|P_{T^\perp}(M)\|_F,$$

where $\zeta \leq \sqrt{\frac{d_1}{2r}}$.

We further assume that there exists a matrix $H \in \mathbb{R}^{n_1 \times n_2}$ satisfying $R_\Omega(H) = H$ such that

$$A2 \quad \|P_T(H - \frac{\lambda_G}{\lambda_E}P_Q(sgn(G_0))) - UV^T\|_F \leq \frac{1}{\sqrt{2d_1}},$$

$$\|P_{T^\perp}(H - \frac{\lambda_G}{\lambda_E}P_Q(sgn(G_0)))\| < \frac{1}{2},$$

where $P_Q(G) = X^TY^{GT}$ by defining $X^\dagger = V_X\Sigma_X^{-1}U_{X(1:d_1)}^T$ and defining $Y^\dagger = V_Y\Sigma_Y^{-1}U_{Y(1:d_2)}^T$, where $U_{X(1:d_1)}^T$ and $U_{Y(1:d_2)}^T$ denote the first $d_1$ and $d_2$ rows of $U_X^T$ and $V_Y^T$, respectively. Then $G_0$ and $E_0$ are the unique minimizer to our optimization problem.

Assume the solution $(G_0, E_0)$ is not unique, and hence there exists another optimal solution $G_0 + G_\Delta$ and $E_0 + E_\Delta$ with $G_\Delta \neq 0, E_\Delta \neq 0$. Using proof by
contradiction, we utilize the subgradient of $\|\cdot\|_*$ and $\|\cdot\|_1$ at $E_0$ and $G_0$ to derive a contradiction that $\lambda_G\|G_0 + G_\Delta\|_1 + \lambda_E\|E_0 + E_\Delta\|_* \geq \lambda_G\|G_0\|_1 + \lambda_E\|E_0\|_*$. Recall the definition of a subgradient of a convex function $f : \mathbb{R}^{n_1 \times n_2} \to \mathbb{R}$. We say that $Z$ is a subgradient of $f$ at $E_0$, denoted by $Z \in \partial f(E_0)$, if

$$f(E) \geq f(E_0) + \langle Z, E - E_0 \rangle$$  \hspace{1cm} \text{(2.3)}

for all $E$.

**Subgradient of $\|E\|_*$:** when $E_0$ has rank $r$ with a singular value decomposition given by

$$E_0 = \sum_{k=1}^r \sigma_k u_k v_k^T,$$

one can obtain that $Z_1$ is a subgradient of $\|E\|_*$ at $E_0$ if and only if it takes the form

$$Z_1 = \sum_{k=1}^r u_k v_k^T + W,$$  \hspace{1cm} \text{(2.4)}

where the following two properties hold:

(i) The column space of $W$ is orthogonal to the linear space $\text{span}(u_1, \cdots, u_r)$, and the row space of $W$ is orthogonal to $\text{span}(v_1, \cdots, v_r)$.

(ii) The norm $\|W\|$ is less than or equal to 1.

It is worth noting that one can derive the subgradients of the nuclear norm according to Theorem 2 in Watson [47].

**Subgradient of $\|G\|_1$:** when $G_0 \in \mathbb{R}^{d_1 \times d_2}$ is sparse, we define $\Omega' \subseteq \{1, \cdots, d_1\} \times \{1, \cdots, d_2\}$ as the set of indexes where the corresponding compo-
nents in \( G_0 \) are non-zeroes, the mapping \( R_{\Omega'}(G) : \mathbb{R}^{d_1 \times d_2} \rightarrow \mathbb{R}^{d_1 \times d_2} \) gives another matrix whose \((i,j)\)-th entry is \( G_{i,j} \) if \((i,j) \in \Omega'\), or 0 otherwise. Then one can obtain a subgradient of \( \|G\|_1 \) at \( G_0 \) as follows:

\[
Z_2 = \text{sgn}(G_0) + J
\]  

(2.5)

where the matrix \( J \in \mathbb{R}^{d_1 \times d_2} \) can be any matrix such that \( R_{\Omega'}(J) = 0 \) and \( \|J\|_\infty \leq 1 \). The signum function on the matrix \( G \in \mathbb{R}^{d_1 \times d_2} \) is defined as follows:

\[
\text{sgn}(G_{i,j}) = \begin{cases} 
-1 & \text{if } G_{i,j} \leq 0 \\
0 & \text{if } G_{i,j} = 0 \\
1 & \text{if } G_{i,j} \geq 0 
\end{cases} \quad \forall 1 \leq i \leq d_1, 1 \leq j \leq d_2. 
\]  

(2.6)

In order to derive a contradiction, we illustrate several useful facts (statements) below:

(a) By the constraint in Eq. (2), we have \( R_{\Omega}(F) = R_{\Omega}(E_0) \). If \( E_0 = XG_0Y^T \), we have \( R_{\Omega}(X(G_0 + G_\Delta)Y^T) = R_{\Omega}(XG_0Y^T) \), and \( X(G_0 + G_\Delta)Y^T = U_XU_X^T \Sigma_XV_X^T(G_0 + G_\Delta)V_Y^T \Sigma_YU_Y^TU_Y^T = P_X(X(G_0 + G_\Delta)Y^T)P_Y \), as \( G_0 + G_\Delta \) minimizes the original problem.

(b) \( XG_\Delta Y^T \neq 0 \), since \( X^\dagger XG_\Delta(YY^\dagger)^T = G_\Delta \neq 0 \) for \( X \) and \( Y \) are full column rank.

(c) \( XG_\Delta Y^T = U_XU_X^T \Sigma_XV_X^T G_\Delta V_Y^T \Sigma_YU_Y^TU_Y^T = P_X(XG_\Delta Y^T)P_Y \) and it holds that \( R_{\Omega}(XG_\Delta Y^T) = 0 \).

(d) \( \|P_T(XG_\Delta Y^T)\|_F \leq \zeta \|P_T(XG_\Delta Y^T)\|_F \leq \zeta \|P_{T^\perp}(XG_\Delta Y^T)\|_*, \) since we...
have \( XG\Delta Y^T \neq 0 \) with Assumption A1.

(e) \( U_\perp \) and \( V_\perp \) are the left and right singular vectors of \( P_T(XG\Delta Y^T) \), while 
\( U^T U_\perp = 0 \) and \( V^T V_\perp = 0 \).

(f) Since \( R_\Omega(XG\Delta Y^T) = R_\Omega(X(G_0 + G_\Delta)Y^T) - R_\Omega(X(G)Y^T) = 0 \) we have 
\( R_\Omega(XG\Delta Y^T) = 0 \).

Proof. By making use of the above statements, we have

\[
\lambda_E \|E_0 + E_\Delta\|_* + \lambda_G \|G_0 + G_\Delta\|_1 = \lambda_E \|X(G_0 + G_\Delta)Y^T\|_* + \lambda_G \|G_0 + G_\Delta\|_1
\]

\[
\geq \lambda_E \langle X(G_0 + G_\Delta)Y^T, U^T V + U_\perp V_\perp^T \rangle + \lambda_G \|G_0\|_1 + \lambda_G \langle \text{sgn}(G_0) + J, G_\Delta \rangle \tag{i}
\]

\[
= \lambda_E \left( \langle XG_0 Y^T, U^T V \rangle + \langle XG_0 Y^T, U_\perp V_\perp^T \rangle + \langle XG_\Delta Y^T, U^T V + U_\perp V_\perp^T \rangle \right)
\]

\[
+ \lambda_G \|G_0\|_1 + \lambda_G \langle \text{sgn}(G_0) + J, G_\Delta \rangle
\]

The term (i) and (ii) are obtained since the inequalities are implied by the subgradients and let \( W = U_\perp V_\perp^T \) and \( J \) satisfy \( R_\Omega(J) = 0 \) and \( \|J\|_{\infty} \leq 1 \).

Furthermore one can obtain that

\[
\lambda_E \left( \langle XG_0 Y^T, U^T V \rangle + \langle XG_0 Y^T, U_\perp V_\perp^T \rangle + \langle XG_\Delta Y^T, U^T V + U_\perp V_\perp^T - H \rangle \right)
\]

\[
+ \lambda_G \|G_0\|_1 + \lambda_G \langle \text{sgn}(G_0) + J, G_\Delta \rangle \tag{iii}
\]

\[
= \lambda_E \|E_0\|_* + \lambda_E \langle XG_0 Y^T, U_\perp V_\perp^T \rangle + \lambda_E \langle XG_\Delta Y^T, U^T V + U_\perp V_\perp^T - H \rangle \tag{iv}
\]

\[
+ \lambda_G \|G_0\|_1 + \lambda_G \langle \text{sgn}(G_0) + J, G_\Delta \rangle
\]

\[
= \lambda_E \|E_0\|_* + \lambda_E \|G_0\|_1 + \lambda_E \left( \langle P_{T_\perp}(XG_\Delta Y^T), U_\perp V_\perp^T - P_{T_\perp}(H) \rangle \right)
\]

\[
+ \lambda_G \langle \text{sgn}(G_0) + J, G_\Delta \rangle + \langle P_T(XG_\Delta Y^T), U^T V - P_T(H) \rangle \tag{v}
\]
where the term (iii) is obtained since we have $R_\Omega(H) = H$ in Assumption A2, and $R_\Omega(XG_\Delta Y^T) = 0$ using Statement (f), so that we have $\langle XG_\Delta Y^T, H \rangle = \langle R_\Omega(XG_\Delta Y^T), R_\Omega(H) \rangle + \langle R_\Lambda\Omega(XG_\Delta Y^T), R_\Lambda\Omega(H) \rangle = 0$. The terms (iv) and (v) are derived from Statement (c). Using matrix norm inequality and selecting $J = -\text{sgn}(R_\Lambda\Omega(G_\Delta))$ so that $\langle J, G_\Delta \rangle = ||R_\Lambda\Omega(G_\Delta)||_1$, one can derive by the norm inequality as follows:

$$
\lambda_E \|E_0 + E_\Delta\|_* + \lambda_G \|G_0 + G_\Delta\|_1 - (\lambda_E \|E_0\|_* + \lambda_G \|G_0\|_1) \\
\geq \lambda_G \|R_\Lambda\Omega(G_\Delta)\|_1 + \lambda_E \left(-\|P_T(XG_\Delta Y^T)\|_*\|UV^T - P_T(H - \frac{\lambda_G}{\lambda_E} P_Q(\text{sgn}(G_0)))\|_1 + \|P_T(XG_\Delta Y^T)\|_*\|UV^T - P_T(H - \frac{\lambda_G}{\lambda_E} P_Q(\text{sgn}(G_0)))\|_1 \cdot \|P_T(XG_\Delta Y^T)\|_*\right) \\
\geq \lambda_G \|R_\Lambda\Omega(G_\Delta)\|_1 + \lambda_E \left(\|P_T(XG_\Delta Y^T)\|_*(1 - \|P_T(H - \frac{\lambda_G}{\lambda_E} P_Q(\text{sgn}(G_0)))\|_1) \right) \\
> \lambda_G \|R_\Lambda\Omega(G_\Delta)\|_1 + \lambda_E \left(\|P_T(XG_\Delta Y^T)\|_*(1 - \frac{1}{2}) - \sqrt{r}\|P_T(XG_\Delta Y^T)\|_F \sqrt{\frac{1}{2d_1}}\right)_{(vi)} \\
\text{(by Assumption A2)} \\
> \lambda_G \|R_\Lambda\Omega(G_\Delta)\|_1 + \lambda_E \left(\|P_T(XG_\Delta Y^T)\|_*(1 - \frac{1}{2}) - \sqrt{r}\|P_T(XG_\Delta Y^T)\|_* \sqrt{\frac{1}{2d_1}}\right)_{(vi)} \\
\text{(by Statement (d))} \\
\geq \lambda_G \|R_\Lambda\Omega(G_\Delta)\|_1 + \lambda_E \|P_T(XG_\Delta Y^T)\|_*\left(\frac{1}{2} - \sqrt{\frac{\xi r}{2d_1}}\right) > 0
$$

(2.7)

since $\|P_T(XG_\Delta Y^T)\|_*(\frac{1}{2} - \sqrt{\frac{\xi r}{2d_1}}) \geq 0$ which is implied from Assumption A2. (vi) is obtained from $\|P_T(XG_\Delta Y^T)\|_* \leq \sqrt{r}\|P_T(XG_\Delta Y^T)\|_F$. Therefore, it re-
veals that the \((E_0 + E_\Delta, G_0 + G_\Delta)\) is not the minimizer for our optimization problem.

We need to prove that A1 and A2 hold with high probabilities (in Lemmas 1, 8, and 9). However, before going to the detailed proof, we summarize our first main result that characterizes the exact recovery as follows:

**Theorem 2:** Let \(\mu = \max(\mu_0, \mu_2), \sigma = \max(\|\Sigma_X^{-1}\|_\star, \|\Sigma_Y^{-1}\|_\star), n = \max(n_1, n_2),\)

\[q_0 = \frac{1}{2}(3 + \log d_1), T_0 = \frac{128}{3} \sigma^2 p \mu \max(\mu_1, \mu) r (d_1 + d_2) \log n \quad \text{and} \quad T_1 = \frac{8}{3} \sigma^2 \mu^2 (d_1 d_2 + r^2) \log n, \]

where \(p\) is a constant. Assume \(T_1 \geq q_0 T_0, X\) and \(Y\) are both full column rank. For any \(p > 1\), by setting \(\lambda_E/\lambda_G = \frac{1}{n}\), with a probability at least \(1 - 2q_0 n^{-p+1} - 2q_0 n^{-p+2}\), \((G_0, E_0)\) is the unique optimizer to Problem (2.2) with necessary sampling rate as few as \(O(r \log n)\). More specifically, the sampling size \(|\Omega|\) should satisfy that \(|\Omega| \geq \frac{64}{3} \sigma^2 p \mu \max(\mu_1, \mu) r (3 + \log d_1)(d_1 + d_2) \log n.\)

**Proof.** To satisfy the probabilistic conditions that Assumption A1 and Assumption A2 that could holds in, one can deduce from Lemma 1, Lemma 7, Lemma 8, and Lemma 9 that \(|\Omega| \geq q_0 T_0 = \frac{64}{3} \sigma^2 p \mu \max(\mu_1, \mu) r (3 + \log d_1)(d_1 + d_2) \log n;\)

therefore, \(|\Omega| = O(r \log n).\)

A1 holds with high probability

In this subsection we prove that Lemma 1 holds with a certain probability. Lemma 1 is derived by combining the results from Lemma 4 and Lemma 5, which
upper-bound $\|P_T(M)\|_F$ and lower-bound $\|P_T(M)\|_F$, and clarify the inequality between them. Lemma 2 and 3 are cited from [40] to facilitate the proof.

Let us first describe Lemma 1 as follows.

**Lemma 1:** With a certain probability at least $1 - 4n^{-p+1}$ for $p > 1$ as stated in Theorem 2, for any $M \neq 0$, $M \in \mathbb{R}^{n_1 \times n_2}$ satisfying $R_\Omega(M) = 0$ and $M = P_XMP_Y$, we have

$$\|P_T(M)\|_F \leq \zeta \|P_T(M)\|_F,$$

where $\zeta$ is the same as in Lemma 1, if the sampling rate $|\Omega|$ can be bounded as $T_0 \leq |\Omega| \leq T_1$.

**Proof.** Since $R_\Omega(M) = R_\Omega(P_X(M)P_Y) = R_\Omega P_T(M) + R_\Omega P_{\perp}(M) = 0$, we have $R_\Omega P_T(M) = -R_\Omega P_{\perp}(M)$. Since

$$\langle P_T(M), M \rangle = \langle P_T(M), P_T(M) + (I - P_T)(M) \rangle$$

$$= \langle P_T(M), P_T(M) \rangle + \langle P_T(M), (I - P_T)(M) \rangle = \langle P_T(M), P_T(M) \rangle = \langle M, P_T(M) \rangle,$$

we could attain $\|R_\Omega P_T(M)\|_F^2 = \langle R_\Omega P_T(M), R_\Omega P_T(M) \rangle = \langle M, P_T R_\Omega R_\Omega P_T(M) \rangle = \langle M, P_T R_\Omega P_T(M) \rangle$. Therefore, we have

$$\frac{n_1n_2}{|\Omega|} \langle M, P_T R_\Omega P_T(M) \rangle = \frac{n_1n_2}{|\Omega|} \langle M, P_{T\perp} R_\Omega P_{T\perp}(M) \rangle.$$

According to Lemma 4 and Lemma 5 with a probability at least $1 - 4n^{-p+1}$,
we have
\[
\frac{1}{2} \|P_T(M)\|_F^2 \leq \frac{n_1 n_2}{|\Omega|} \langle M, P_T R_{\Omega} P_T(M) \rangle
\]
\[
\leq \frac{16 \sigma^2 \mu^2 p(d_1 d_2 + r^2) \log n}{3|\Omega|} \|P_{T^\perp}(M)\|_F^2
\]
\[
\leq \frac{16 \sigma^2 \mu^2 p(d_1 d_2 + r^2) \log n}{3 T_0} \|P_{T^\perp}(M)\|_F^2
\]
\[
\leq \frac{1}{4} \|P_{T^\perp}(M)\|_F^2.
\]

Hence, we have
\[
\|P_T(M)\|_F \leq \frac{1}{\sqrt{2}} \|P_{T^\perp}(M)\|_F \leq \sqrt{\frac{d_1}{2r}} \|P_{T^\perp}(M)\|_F \leq \zeta \|P_{T^\perp}(M)\|_F
\]

since \(\frac{1}{\sqrt{2}} \leq \sqrt{\frac{d_1}{2r}}\).

We recite Lemma 2, Lemma 3 that have been previously derived in [40] from the Berstein Inequality [28]. We use these lemmas in proving both Lemmas 4 and 5.

**Lemma 2:** [40] Let \(X_1, \ldots, X_L\) be independent zero-mean random matrices of dimension \(a \times b\). Suppose \(\rho_k^2 \geq \max\{\|E[X_k X_k^T]\|, \|E[X_k^T X_k]\|\}\) and \(\|X_k\| \leq M\) almost surely for all \(k\). If we assume
\[
M^2 \log \frac{a + b}{\xi} \leq \frac{3}{8} \sum_k \rho_k^2,
\]
then with a certain probability at least \(1 - \xi\), we have
\[
\|\sum_{k=1}^L X_k\| \leq \sqrt{\frac{8 \log \frac{a + b}{\xi}}{3} \sum_{k=1}^L \rho_k^2}.
\]
Lemma 3: \[40\] Let $X_1, \ldots, X_L$ be independent zero-mean random matrices of dimension $a \times b$. Suppose $\rho_k^2 \geq \max\{\|E[X_kX_k^T]\|, \|X_k^T X_k\|\}$ and $\|X_k\| \leq M$ almost surely for all $k$. If we assume

$$M^2 \log \frac{a+b}{\xi} \geq \frac{3}{8} \sum \rho_k^2,$$

then with a certain probability at least $1 - \xi$, we have

$$\|\sum_{k=1}^L X_k\| \leq \frac{8}{3} M \log \frac{a+b}{\xi}.$$

Next we will bound $\|P_T - \frac{n_1 n_2}{|\Omega|} P_TR\Omega P_T\|$ by using Lemma 2 and Lemma 3.

Lemma 4: With a certain probability at least $1 - 2n^{-p+1}$, we have

$$\left\|P_T - \frac{n_1 n_2}{|\Omega|} P_TR\Omega P_T\right\| \leq \sqrt{\frac{8\sigma^2 p\mu^2 r(d_1 + d_2) \log n}{3|\Omega|}}$$

if $|\Omega| \geq \frac{8}{3} \sigma^2 p\mu^2 r(d_1 + d_2) \log n$. Therefore, for any $M \in \mathbb{R}^{n_1 \times n_2}$, we have

$$\frac{1}{2}\|P_T(M)\|_F^2 \leq \frac{n_1 n_2}{|\Omega|} \langle M, P_TR\Omega P_T(M) \rangle$$

if $|\Omega| \geq T_0$.

Proof. To simplify the left side of the inequality, for any $i \in [n_1]$ and $j \in [n_2]$, define the linear operator $T_{i,j}$ as

$$T_{i,j}(M) = \langle M, P_T(e_i e_j^T) \rangle P_T(e_i e_j^T) = P_T R_{(i,j)} P_T(M),$$
where $R_{(i,j)}(M) = e_ie_j^T M_{i,j}$ for any $M \in \mathbb{R}^{n_1 \times n_2}$. From Theorem 4.1 in [26] we similarly could obtain

$$P_T R_{i,j} P_T(M) = \sum_{(i,j) \in \Omega} \langle P_T(M), e_i e_j^T \rangle P_T(e_i e_j^T) = \sum_{(i,j) \in \Omega} \langle M, P_T(e_i e_j^T) \rangle P_T(e_i e_j^T)$$

$$= \sum_{(i,j) \in \Omega} P_T R_{(i,j)} P_T(M)$$

$$= \sum_{(i,j) \in \Omega} T_{i,j}(M)$$

To implement Lemma 2, we need to give $M$ and the corresponding $\rho_2^2$’s.

Since $\|P_T - \frac{n_1 n_2}{|\Omega|} P_T R_{i,j} P_T\|$ can be viewed as the spectral norm of $|\Omega|$ independent zero-mean random variables $\frac{1}{|\Omega|} P_T - \frac{n_1 n_2}{|\Omega|} T_{i,j}$, then we have

$$\|\frac{1}{|\Omega|} P_T - \frac{n_1 n_2}{|\Omega|} T_{i,j}\|$$

$$\leq \max\{\|\frac{1}{|\Omega|} P_T\|, \|\frac{n_1 n_2}{|\Omega|} T_{i,j}\|\}$$

$$= \max\{\|\frac{1}{|\Omega|} P_T\|, \frac{n_1 n_2}{|\Omega|} \arg \max_{\|M\|_F = 1} \langle M, P_T(e_i e_j^T) \rangle P_T(e_i e_j^T)\|_F\}$$

$$= \max\{\|\frac{1}{|\Omega|} P_T\|, \frac{n_1 n_2}{|\Omega|} \arg \max_{\|M\|_F = 1} \langle M, P_T(e_i e_j^T) \rangle P_T(e_i e_j^T)\|_F\}$$

$$= \max\{\|\frac{1}{|\Omega|} P_T\|, \frac{n_1 n_2}{|\Omega|} \|P_T(e_i e_j^T)\|_F^2\}.$$
To bound $\|P_T(e_ie_j^T)\|_F^2$, from the definition of $P_T$, we get

\[
\|P_T(e_ie_j^T)\|_F^2 = \langle P_T(e_ie_j^T), e_ie_j^T \rangle
\]

\[
= \langle P_X(e_ie_j^T)P_V, e_ie_j^T \rangle + \langle P_U(e_ie_j^T)P_Y, e_ie_j^T \rangle - \langle P_U(e_ie_j^T)P_V, e_ie_j^T \rangle
\]

\[
= \|P_X(e_ie_j^T)P_V\|_F^2 + \|P_U(e_ie_j^T)P_Y\|_F^2 - \|P_U(e_ie_j^T)P_V\|_F^2
\]

\[
\leq \|P_Xe_i\|_F^2 \|P_Ve_j\|_F^2 + \|P_Ue_i\|_F^2 \|P_Ye_j\|_F^2
\]

\[
\leq \|XV_X\Sigma_X^{-2}V_X^T\|_F^2 \frac{d_1\mu_2 \nu_0}{n_1} \frac{\nu_0 d_2 \mu_2}{n_2} + \|VV_Y\Sigma_Y^{-2}V_Y^T\|_F^2 \frac{\nu_0 d_1 \mu_2}{n_1} \frac{\nu_0 d_2 \mu_2}{n_2}
\]

\[
\leq \|\Sigma_X^{-1}\|_F^2 \frac{d_1\mu_2 \nu_0}{n_1} \frac{\nu_0 d_2 \mu_2}{n_2} \leq \sigma^2 \frac{\mu_2 \nu_0 d_1 + d_2}{n_1 n_2} \leq \frac{\sigma^2 \mu_2 \nu_0 (d_1 + d_2)}{n_1 n_2}.
\]

Therefore, we have

\[
\left\| \frac{1}{|\Omega|} P_T - \frac{n_1 n_2}{|\Omega|} T_{i,j} \right\| \leq \max\left\{ \left\| \frac{1}{|\Omega|} P_T \right\|, \frac{n_1 n_2}{|\Omega|} \left\| P_T(e_ie_j^T) \right\|_F^2 \right\}
\]

\[
\leq \max\left\{ \left\| \frac{1}{|\Omega|} P_T \right\|, \frac{\sigma^2 \mu_2 \nu_0 (d_1 + d_2)}{n_1 n_2} \right\}
\]

\[
= \max\left\{ \frac{1}{|\Omega|}, \frac{\sigma^2 \mu_2 \nu_0 (d_1 + d_2)}{n_1 n_2} \right\} = \max\left\{ \frac{1}{|\Omega|}, \frac{\sigma^2 \mu_2 \nu_0 (d_1 + d_2)}{n_1 n_2} \right\} = \frac{\sigma^2 \mu_2 \nu_0 (d_1 + d_2)}{n_1 n_2} = M.
\]

As $\frac{1}{|\Omega|} \mathbb{E}[P_TR_{\Omega}P_T] = \frac{1}{|\Omega|} [P_T \mathbb{E}(R_{\Omega})P_T] = \frac{1}{|\Omega|} [P_T \frac{|\Omega|}{n_1 n_2} P_T] = \frac{1}{n_1 n_2} P_T$, the corre-
sponding $\rho_{i,j}^2$ can be calculated as

$$
\rho_{i,j}^2 = \| \mathbb{E}[\frac{1}{|\Omega|} P_T - \frac{n_1 n_2}{|\Omega|} T_{i,j}]^T (\frac{1}{|\Omega|} P_T - \frac{n_1 n_2}{|\Omega|} T_{i,j}) \|
$$

$$
= \| \mathbb{E}[\frac{1}{|\Omega|^2} P_T P_T + \frac{n_1 n_2}{|\Omega|^2} T_{i,j} T_{i,j} - \frac{2n_1 n_2}{|\Omega|^2} P_T T_{i,j}] \|
$$

$$
= \| \frac{1}{|\Omega|^2} P_T + \frac{n_1 n_2}{|\Omega|^2} \mathbb{E}[T_{i,j} T_{i,j}] - \frac{2n_1 n_2}{|\Omega|^2} P_T \mathbb{E}[T_{i,j}] \|
$$

$$
= \| \frac{n_1 n_2}{|\Omega|^2} \mathbb{E}[T_{i,j} T_{i,j}] - \frac{1}{|\Omega|^2} P_T \|
$$

$$
\leq \max \left\{ \frac{n_1 n_2}{|\Omega|^2} \mathbb{E}[T_{i,j} T_{i,j}], \frac{1}{|\Omega|^2} P_T \right\}
$$

$$
\leq \max \left\{ \frac{n_1 n_2}{|\Omega|^2} \mathbb{E}[\|P_T (e(e^T))\|_F^2 \|T_{i,j}\|], \frac{1}{|\Omega|^2} \right\}
$$

$$
\leq \max \left\{ \frac{n_1 n_2}{|\Omega|^2} \sigma^2 \mu^2 (d_1 + d_2), \frac{1}{n_1 n_2} \|P_T\|, \frac{1}{|\Omega|^2} \right\}
$$

$$
= \frac{\sigma^2 \mu^2 r (d_1 + d_2)}{|\Omega|^2}.
$$

By Lemma 3 with $M = \sigma^2 \mu^2 r (d_1 + d_2)$ and $\rho_{i,j}^2 = \frac{\sigma^2 \mu^2 r (d_1 + d_2)}{|\Omega|^2}$, we conclude with a certain probability $1 - 2n^{-p+1}$,

$$
\|P_T - \frac{n_1 n_2}{|\Omega|} P_T R \Omega P_T \|
$$

$$
\leq \sqrt{\frac{8}{3} \log \frac{n_1 + n_2 \sigma^2 \mu^2 r (d_1 + d_2)}{2n^{-p+1}|\Omega|}} \leq \sqrt{\frac{8\sigma^2 \mu^2 r (d_1 + d_2) \log n}{3|\Omega|}}
$$

which also should satisfy the condition that

$$
\frac{\sigma^4 \mu^4 r^2 (d_1 + d_2)^2}{|\Omega|^2} \log \frac{n_1 + n_2}{2n^{-p+1}} \leq \frac{3 \sigma^2 \mu^2 r (d_1 + d_2)}{8 |\Omega|}
$$

which means

$$
|\Omega| \geq \frac{8\sigma^2 \mu^2 r (d_1 + d_2) \log n}{3}.
$$
Moreover, if $|\Omega| \geq T_0 \geq \frac{128\sigma^2\mu^2 r (d_1 + d_2) \log n}{3}$, we have

$$\left\| P_T - \frac{n_1 n_2}{|\Omega|} P_T R_\Omega P_T \right\| \leq \sqrt{\frac{8\sigma^2\mu^2 r (d_1 + d_2) \log n}{3|\Omega|}} \leq \frac{1}{2},$$

By utilizing the property of matrix norm, we have

$$\langle M, P_T(M) - \frac{n_1 n_2}{|\Omega|} P_T R_\Omega P_T(M) \rangle \leq \frac{1}{2} \| P_T(M) \|_F^2$$

so that we have

$$\langle M, P_T(M) \rangle - \frac{1}{2} \| P_T(M) \|_F^2 \leq \langle M, \frac{n_1 n_2}{|\Omega|} P_T R_\Omega P_T(M) \rangle,$$

from which we can easily derive

$$\frac{1}{2} \| P_T(M) \|_F^2 \leq \langle M, \frac{n_1 n_2}{|\Omega|} P_T R_\Omega P_T(M) \rangle.$$
Proof. Following the proof of Lemma 4, we can also define \( T_{i,j}^\perp = P_{T \perp} R_{i,j} P_{T \perp} \) so that

\[
\sum_{(i,j) \in \Omega} T_{i,j}^\perp (M) = \sum_{(i,j) \in \Omega} P_{T \perp} R_{i,j} P_{T \perp} (M) = P_{T \perp} R_{\Omega} P_{T \perp} (M),
\]

To derive the bound of the operator norm of \( P_{T \perp} - \frac{n_1 n_2}{|\Omega|} T_{i,j}^\perp \), one can view this term as a sum of \(|\Omega|\) independent zero-mean variables which is

\[
\frac{1}{|\Omega|} P_{T \perp} - \frac{n_1 n_2}{|\Omega|} T_{i,j}^\perp.
\]

Further we have

\[
\| \frac{1}{|\Omega|} P_{T \perp} - \frac{n_1 n_2}{|\Omega|} T_{i,j}^\perp \| \leq \max\{ \| \frac{1}{|\Omega|} P_{T \perp} \|, \| \frac{n_1 n_2}{|\Omega|} T_{i,j}^\perp \| \}
\]

\[
\max\{ \| \frac{1}{|\Omega|} P_{T \perp} \|, \| \frac{n_1 n_2}{|\Omega|} \| P_{T \perp} (e_i e_j^T) \|_F^2 \}.
\]

We can deduce that

\[
\| P_{T \perp} (e_i e_j^T) \|_F^2
\]

\[
= \| P_X (e_i e_j^T) P_Y \|_F^2 + \| P_U (e_i e_j^T) P_V \|_F^2 - \| P_X (e_i e_j^T) P_V \|_F^2 - \| P_U (e_i e_j^T) P_Y \|_F^2
\]

\[
\leq \| P_X (e_i e_j^T) P_Y \|_F^2 + \| P_U (e_i e_j^T) P_V \|_F^2
\]

\[
\leq \| X \Sigma_X^{-2} V_X^T \|_F^2 \frac{d_1 \mu_2 r \mu_0}{n_1 n_2} + \| Y \Sigma_Y^{-2} V_Y^T \|_F^2 \frac{r \mu_0 d_2 \mu_2}{n_1 n_2}
\]

\[
\leq \| \Sigma_X^{-1} \|_F^2 \| \mu_2 \|_2 d_1 d_2 \|_F^2 \frac{r \mu_0}{n_1 n_2} + \| \Sigma_Y^{-1} \|_F^2 \| \mu_2 \|_2 d_2 \|_F^2 \frac{r \mu_0 d_2 \mu_2}{n_1 n_2}
\]

\[
\leq \frac{\sigma^2 \mu^2 (d_1 d_2 + r)}{n_1 n_2}.
\]

Therefore, one can choose the values of \( M \) and \( \rho^2 \) such that

\[
M = \max\left\{ \frac{1}{|\Omega|}, \frac{n_1 n_2 \sigma^2 \mu^2 (d_1 d_2 + r)}{|\Omega|} \right\} = \frac{\sigma^2 \mu^2 (d_1 d_2 + r)}{|\Omega|}.
\]
and
\[
\rho^2 = \| \mathbb{E}\left[ \frac{1}{|\Omega|} P_{T^\perp} - \frac{n_1 n_2}{|\Omega|^2} T_{p,q}^T \left( \frac{1}{|\Omega|} P_{T^\perp} - \frac{n_1 n_2}{|\Omega|^2} T_{i,j} \right) \right] \|
\]
\[
= \left\| \frac{n_1^2 n_2^2}{|\Omega|^2} \mathbb{E}[T_{i,j} T_{i,j}] - \frac{1}{|\Omega|^2} P_{T^\perp} \right\|
\]
\[
\leq \max\left\{ \frac{n_1^2 n_2^2 \sigma^2 \mu^2 (d_1 d_2 + r^2)}{|\Omega|^2}, \frac{1}{n_1 n_2}, \frac{1}{|\Omega|^2} \right\}
\]
\[
= \frac{\sigma^2 \mu^2 (d_1 d_2 + r^2)}{|\Omega|^2}.
\]

By Lemma 2, one can deduce that with a probability at least \(1 - 2n^{-p+1}\), if
\[
\frac{\sigma^4 \mu^4 (d_1 d_2 + r^2)^2}{|\Omega|^2} \log \frac{2n}{2n^{-p+1}} \geq \frac{3 \sigma^2 \mu^2 (d_1 d_2 + r^2)}{8 |\Omega|^2},
\]
that is
\[
|\Omega| \leq \frac{8}{3} \sigma^2 \mu^2 (d_1 d_2 + r^2) \log n = T_1,
\]
we can obtain
\[
\left\| P_{T^\perp} - \frac{n_1 n_2}{|\Omega|} P_{T^\perp} R_\Omega P_{T^\perp} \right\| \leq \frac{8 \sigma^2 \mu^2 (d_1 d_2 + r^2) \log n}{3|\Omega|}.
\]

Further we have
\[
- \frac{8 \sigma^2 \mu^2 (d_1 d_2 + r^2) \log n}{3|\Omega|} \left\| P_{T^\perp} (\mathbf{M}) \right\|_F \leq \left\langle \mathbf{M}, P_{T^\perp} (\mathbf{M}) - \frac{n_1 n_2}{|\Omega|} P_{T^\perp} R_\Omega P_{T^\perp} (\mathbf{M}) \right\rangle
\]
\[
= \left\| P_{T^\perp} (\mathbf{M}) \right\|_F^2 - \left\langle \mathbf{M}, \frac{n_1 n_2}{|\Omega|} P_{T^\perp} R_\Omega P_{T^\perp} (\mathbf{M}) \right\rangle
\]

Hence, we can obtain
\[
\left\langle \mathbf{M}, \frac{n_1 n_2}{|\Omega|} P_{T^\perp} R_\Omega P_{T^\perp} (\mathbf{M}) \right\rangle \leq (1 + \frac{8 \sigma^2 \mu^2 (d_1 d_2 + r^2) \log n}{3|\Omega|}) \left\| P_{T^\perp} (\mathbf{M}) \right\|_F^2.
\]

Due to \(|\Omega| \leq T_1\), we can have
\[
\left\langle \mathbf{M}, \frac{n_1 n_2}{|\Omega|} P_{T^\perp} R_\Omega P_{T^\perp} (\mathbf{M}) \right\rangle \leq (\frac{16 \sigma^2 \mu^2 (d_1 d_2 + r^2) \log n}{3|\Omega|}) \left\| P_{T^\perp} (\mathbf{M}) \right\|_F^2.
\]

\[\blacksquare\]
A2 holds with high probability

In this subsection we aim to investigate the condition when A2 holds with a high probability. We also need to bound the following two terms \( \frac{n_1 n_2}{|\Omega|} \| P_{T \perp} R_{\Omega} P_T(H) \| \) and \( \| P_T(H) - \frac{n_1 n_2}{|\Omega|} P_T R_{\Omega} P_T(H) \|_\infty \) in Lemma 6 and 7 respectively where \( \| \cdot \|_\infty \) is the maximum entry of a matrix.

**Lemma 6:** For a fixed \( H \in \mathbb{R}^{n_1 \times n_2} \) satisfying \( R_{\Omega}(H) = H \), with a probability \( 1 - 2^{n-p+1} \), we have

\[
\frac{n_1 n_2}{|\Omega|} \| P_{T \perp} R_{\Omega} P_T(H) \| \leq \| P_T(H) \|_\infty \sqrt{\frac{8\sigma^2 \mu d_1 n_1 n_2 \log n}{3|\Omega|}},
\]

if \( |\Omega| \geq T_0 \).

**Proof.** We write

\[
P_{T \perp} R_{\Omega} P_T(H) = \sum_{(i,j) \in \Omega} \langle H, P_T(e_i e_j^T) \rangle P_{T \perp}(e_i e_j^T) = \sum_{(i,j) \in \Omega} S_{i,j},
\]

where \( S_{i,j}(H) = \langle H, P_T(e_i e_j^T) \rangle P_{T \perp}(e_i e_j^T) \). Evidently, we have \( \mathbb{E}[P_{T \perp} R_{\Omega} P_T(H)] = 0 \). To use Lemma 2 we compute \( M \) and \( \rho_{i,j}^2 \) as,

\[
M = \max_{i \in [n_1], j \in [n_2]} \| S_{i,j} \|
\]

\[
\leq \max_{i \in [n_1], j \in [n_2]} \max_{\|H\|_F = 1} \| \langle H, P_T(e_i e_j^T) \rangle P_{T \perp}(e_i e_j^T) \|_F
\]

\[
\leq \max_{i \in [n_1], j \in [n_2]} \| H \|_F \max_{i \in [n_1], j \in [n_2]} \| \langle H, P_T(e_i e_j^T) \rangle P_{T \perp}(e_i e_j^T) \|_F
\]

\[
\leq \| P_T(H) \|_\infty \max_{i \in [n_1], j \in [n_2]} \| P_{T \perp}(e_i e_j^T) \|
\]

\[
\leq \| P_T(H) \|_\infty \sqrt{\frac{\mu^2 \sigma^2 (d_1 d_2 + r^2)}{n_1 n_2}},
\]
and

\[ \rho_{i,j}^2 = \max\{ \|E[S_{i,j}, S_{i,j}^T]\|, \|E[S_{i,j}^T, S_{i,j}]\| \}. \]

It suffices to show that

\[
\|E[S_{i,j}, S_{i,j}^T]\| = \|P T(H)\|^2 \|E[P \perp_i e_j e_j^T P \perp_i (e_i e_i^T)]\| = \|P T(H)\|^2 \|E[P Y \perp_i e_i e_i^T P X \perp_i e_j e_j^T P Y \perp_i]\| \leq \|P T(H)\|^2 \|E[\sigma^2 \mu_2 d_1 / n_1] P Y \perp_i P Y \perp_i \| \leq \|P T(H)\|^2 \|E[\sigma^2 \mu_2 d_1 / n_1] P Y \perp_i \| \leq \|P T(H)\|^2 \|E[\sigma^2 \mu_2 d_1 / n_1 / n_2] P Y \perp_i P Y \perp_i \|
\]

The similar inequality on \( \|E[S_{i,j}^T, S_{i,j}]\| \) can be attained as below:

\[
\|E[S_{i,j}^T, S_{i,j}]\| = \|P T(H)\|^2 \|E[P \perp_i e_i e_i^T P \perp_i (e_i e_i^T)]\| = \|P T(H)\|^2 \|E[P X \perp_i e_i e_i^T P Y \perp_i e_j e_j^T P X \perp_i]\| \leq \|P T(H)\|^2 \|E[\sigma^2 \mu_2 d_2 / n_2] P X \perp_i P X \perp_i \| \leq \|P T(H)\|^2 \|E[\sigma^2 \mu_2 d_2 / n_2] P X \perp_i \| \leq \|P T(H)\|^2 \|E[\sigma^2 \mu_2 d_2 / n_1 n_2] P X \perp_i P X \perp_i \|
\]

Therefore, we have

\[ \rho_{i,j}^2 \leq \|P T(H)\|^2 \|E[\sigma^2 \mu_2 \max\{d_1, d_2\} / n_1 n_2] \|. \]
To prove simply without loss of the generality, we assume $d_2 \leq d_1$, so we can get
\[
\rho_{i,j}^2 \leq \|P_T(H)\|_\infty^2 \frac{\sigma^2 \mu d_1}{n_1 n_2}
\]

By Lemma 2 suppose that it satisfies
\[
\|P_T(H)\|_\infty^2 \frac{\sigma^2 \mu^2 (d_1 d_2 + r^2)}{n_1 n_2} \log \frac{2n}{2n^{-p+1}} \leq \frac{3}{8} \|P_T(H)\|_\infty^2 \frac{\sigma^2 \mu d_1 |\Omega|}{n_1 n_2}
\]
which is equivalent to
\[
\frac{8 \sigma^2 p \mu (d_1 d_2 + r^2) \log n}{3 d_1} \leq |\Omega|,
\]
with a probability of $1 - 2n^{-p+1}$ one can obtain
\[
\frac{n_1 n_2}{|\Omega|} \|P_{\perp R_{\Omega} P_T(H)}\| \leq \frac{n_1 n_2}{|\Omega|} \|P_T(H)\|_\infty \sqrt{\frac{8 \sigma^2 p \rho_{i,j}^2 |\Omega| \log n}{3}}
\]
\[
\leq \frac{n_1 n_2}{|\Omega|} \|P_T(H)\|_\infty \sqrt{\frac{8 \sigma^2 p \mu d_1 |\Omega| \log n}{3 n_1 n_2}}
\]
\[
= \|P_T(H)\|_\infty \sqrt{\frac{8 \sigma^2 p \mu d_1 n_1 n_2 \log n}{3 |\Omega|}}.
\]

Observing Lemma 1 we need to satisfy the condition $|\Omega| \geq T_0$. Then, we have
\[
|\Omega| \geq \frac{128 \sigma^2 p \mu^2 r (d_1 + d_2) \log n}{3} \geq \frac{8 \sigma^2 p \mu (d_1 d_2 + r^2) \log n}{3 d_1}
\]
which is because $\mu \geq 1$, $d_1 \geq d_2$, and $d_1 \geq r$. Then we have $\frac{8 \sigma^2 p \mu (d_1 d_2 + r^2) \log n}{3 d_1} \leq |\Omega|$. 

\[\blacksquare\]
Lemma 7: For a fixed $H \in \mathbb{R}^{n_1 \times n_2}$, with a probability $1 - 2n^{-p + 2}$, we have

$$
|\| P_T - \frac{n_1 n_2}{|\Omega|} P_T R_{\Omega} P_T \|\|_\infty \leq \sqrt{\frac{8 \sigma^2 \mu^2 r (d_1 + d_2) \log n}{3|\Omega|}} \| P_T (H) \|_\infty
$$

and therefore if $|\Omega| \geq T_0$,

$$
\| P_T - \frac{n_1 n_2}{|\Omega|} P_T R_{\Omega} P_T \| \leq \frac{1}{2} \| P_T (H) \|_\infty
$$

Proof. For each matrix index $(i, j)$, sample $(i', j')$ uniformly at random to define the random variable $\eta_{i,j} = [(n_1 n_2) P_T R_{(i', j')} P_T (H) - P_T (H)]_{i,j}$. We have

$$
E[\eta_{i,j}] = 0,
$$

$$
|\eta_{i,j}| \leq \| P_T R_{(i', j')} P_T - P_T \| \| P_T (H) \|_\infty \leq \sigma^2 \mu^2 r (d_1 + d_2) \| P_T (H) \|_\infty,
$$

and we have

$$
E[\eta_{i,j}^2]
$$

$$
= E[((n_1 n_2) P_T R_{(i', j')} P_T (H) - P_T (H)]_{i,j})^2]]
$$

$$
= E[((n_1 n_2)^2 P_T R_{(i', j')} P_T (H)]_{i,j})^2]
$$

$$
+ ([P_T (H)]_{i,j})^2 - 2n_1 n_2 E[((P_T R_{(i', j')} P_T (H)]_{i,j}[P_T (H)]_{i,j})^2]
$$

$$
= (n_1 n_2)^2 E[((P_T R_{(i', j')} P_T (H)]_{i,j})^2] - ([P_T (H)]_{i,j})^2
$$

$$
= (n_1 n_2)^2 E[\langle e_i e_j^T, P_T (e_i e_j^T) \rangle \langle H, P_T (e_i e_j^T) \rangle^2] - ([P_T (H)]_{i', j'})^2
$$

$$
= (n_1 n_2) \| P_T (H) \|_F^2 \| P_T (e_i e_j) \|_F^2 - ([P_T (H)]_{i,j})^2
$$

$$
\leq \| P_T (H) \|_F^2 \sigma^2 \mu^2 r (d_1 + d_2).
$$

Using the Bernstein Inequality, we have

$$
P[[n_1 n_2 P_T R_{\Omega} P_T (H) - |\Omega| P_T (H)]_{i,j} > C] \leq 2n^{-p}
$$
where $C = \sqrt{\frac{2}{3}|\Omega|} \|P_T(H)\|_\infty \sigma^2 \mu^2 r (d_1 + d_2) \log \frac{2}{2n-p}$. The bound of the infinity norm, with a probability of $1 - 2n^{-p+2}$, can be given by

$$\| \frac{n_1 n_2}{|\Omega|} P_T R_{\Omega} P_T(H) - P_T(H) \|_\infty \leq \sqrt{\frac{8\sigma^2 p \mu^2 r (d_1 + d_2) \log n}{3|\Omega|}} \|P_T(H)\|_\infty.$$

If $|\Omega| \geq T_0$, we can obtain

$$\| \frac{n_1 n_2}{|\Omega|} P_T R_{\Omega} P_T(H) - P_T(H) \|_\infty \leq \frac{1}{2} \|P_T(H)\|_\infty.$$

In Assumption A2, we assume that there exists a matrix $H$ that satisfies certain conditions. Next, we show that $H$ can be constructed from the gaging scheme in [48] in the following statements.

A sequence of $H_t, t = 1, \cdots, q$ can be generated as follows

$$H_t = \frac{n_1 n_2}{T_0} \sum_{l=1}^{t} R_{\Omega_l}(W_l),$$

where $W_1 = UV^T - \frac{\lambda_G}{\lambda_E} P_T(P_Q(\text{sgn}(G_0)))$ and $W_{t+1}$ is defined as

$$W_{t+1} = UV^T - P_T(H_t) - \frac{\lambda_G}{\lambda_E} P_T(P_Q(\text{sgn}(G_0)))$$

$$= P_T(UV^T - H_t) - \frac{\lambda_G}{\lambda_E} P_T(P_Q(\text{sgn}(G_0))) = P_T(W_t) - \frac{n_1 n_2}{T_0} P_T R_{\Omega_t}(W_t)$$

$$= (P_T - \frac{n_1 n_2}{T_0} P_T R_{\Omega_t}) P_T(W_t)$$

We randomly select $qT_0$ entries from $\Omega$ and partition the selected entries into $q$ subsets as $\Omega_1, \cdots, \Omega_q$ with equal sizes, with $|\Omega_i| = T_0, i = 1, \cdots, q$. We construct $H = H_q$. It can be checked that the constructed $H$ satisfies $H = R_\Omega(H)$. 


Now we are ready to show that $H$ satisfies the other two properties in Assumption A2.

**Lemma 8:** With a probability of $1 - 2qn^{-p+1}$, if $q \geq q_0$, it is satisfied that

$$
\|P_T(H - \frac{\lambda_G}{\lambda_E} P_Q(sgn(G_0))) - UV^T\| \leq \sqrt{\frac{1}{2d_1}}.
$$

**Proof.** One can observe that

$$W_{t+1} = (P_T \frac{n_1n_2}{T_0} P_T R_{\Omega_1} P_T) W_t,$$

then we have

$$
\|P_T(H - \frac{\lambda_G}{\lambda_E} P_Q(sgn(G_0))) - UV^T\| \leq \prod_{i=1}^q \|P_T \frac{n_1n_2}{T_0} P_T R_{\Omega_i} P_T\| \|W_1\|
$$

$$\leq \frac{1}{2q}(\|UV^T\| + \frac{\lambda_G}{\lambda_E} \|P_T(P_Q(sgn(G_0)))\|).$$

From Bernstein’s inequality we have

$$
P(\langle P_T(P_Q(sgn(G_0))), e_i e_j^T \rangle \geq t)
$$

$$= \mathbb{P}(\langle P_Q(sgn(G_0)), P_T(e_i e_j^T) \rangle \geq t) \leq 2 \exp\left(-\frac{t^2}{N + \frac{1}{3}Mt}\right)
$$

where

$$
\begin{align*}
N &:= 2\gamma \|P_Q\| \|P_T(e_i e_j^T)\|_F^2 \leq \frac{2\gamma \sigma^4 \mu^2 r^2 (d_1 + d_2)^2}{n_1^2 n_2^2} \\
M &:= \|P_T(e_i e_j^T)\|_\infty \leq \frac{\sigma^2 \mu^2 r^2 (d_1 + d_2)^2}{n_1^2 n_2^2}.
\end{align*}
$$

Then with a high probability, it satisfies

$$
\|P_T(P_Q(sgn(G_0)))\|_F \leq \sqrt{n_1 n_2} \|P_T(P_Q(sgn(G_0)))\|_\infty \leq \sqrt{n_1 n_2} \frac{\sigma^2 \mu r (d_1 + d_2) \log n_1}{n_1 n_2} \leq \frac{\sigma^2 \mu r (d_1 + d_2) \log n_1}{\sqrt{n_1 n_2}}.
$$
As \( \lambda_G/\lambda_E = 1/n \),

\[
\frac{\lambda_G}{\lambda_E} \| P_T(P_Q(\text{sgn}(G_0)))) \|_F \leq \frac{\sigma^2 \mu r (d_1 + d_2) \log n_1}{\sqrt{\max\{n_1, n_2\}}} \ll 1
\]

whenever \( n_1 \) or \( n_2 \) is large enough. Therefore, from Lemma 4 with a probability \( 1 - 2qn^{-p+1} \), we have the following result:

\[
\| P_T(H - P_Q(\text{sgn}(G_0)))) - UV^T \|
\]

\[
\leq \frac{1}{2^q} (\| UV^T \| + \frac{\lambda_G}{\lambda_E} \| P_T(P_Q(\text{sgn}(G_0)))) \|_F) \leq \sqrt{\frac{1}{2d_1}}
\]

by choosing \( q = q_0 = \frac{1}{2}(3 + \log d_1) \).

**Lemma 9**: With a probability of \( 1 - 2qn^{-p+1} - 2qn^{-p+2} \), it is satisfied that

\[
\| P_{T^\perp}(H - \frac{\lambda_G}{\lambda_E} P_Q(\text{sgn}(G_0)))) \| \leq \frac{1}{2}
\]

if \( q \geq q_0 \).

**Proof.** Following the arguments in [49] about the norm of a matrix with i.i.d. entries, we can obtain \( \| \text{sgn}(G_0) \| \leq 4\sqrt{\beta} \gamma \) where \( \gamma \) is the ratio of non-zero fractions in \( \text{sgn}(G_0) \). For easier proof, our assumption is that the signs of the nonzero entries of \( G_0 \) are independent symmetric Bernoulli variables. Since \( \frac{\lambda_G}{\lambda_E} = 1/n \), we have \( \frac{\lambda_G}{\lambda_E} \| \text{sgn}(G_0) \| \leq 4\sqrt{\beta} \gamma \). Thus, we can attain

\[
\| P_{T^\perp}(\frac{\lambda_G}{\lambda_E} P_Q(\text{sgn}(G_0)))) \| \leq \frac{\lambda_G}{\lambda_E} \| P_Q \| \| \text{sgn}(G_0) \| \leq \sigma^2 \sqrt{\frac{16\gamma}{n}}.
\]

From Lemma 7 we have

\[
\| W_{t+1} \|_\infty = \| (P_T - \frac{n_1n_2}{T_0} P_T R_0 P_T) W_t \|_\infty \leq \frac{1}{2} \| W_t \|_\infty.
\]
We can obtain the bound of \( \| P_{T^\perp} (H - \frac{\lambda G}{\lambda E} P_Q(sgn(G_0))) \| \) as follows:

\[
\| P_{T^\perp} (H - \frac{\lambda G}{\lambda E} P_Q(sgn(G_0))) \| 
\leq \frac{n_1 n_2}{T_0} \sum_{l=1}^{q} \| P_{T^\perp} R_{\Omega_l} (W_l) \| + \| P_{T^\perp} (\frac{\lambda G}{\lambda E} P_Q(sgn(G_0))) \| 
\leq \sum_{l=1}^{q} \frac{n_1 n_2}{T_0} \| P_{T^\perp} R_{\Omega_l} P_T (W_l) \| + \| P_{T^\perp} (\frac{\lambda G}{\lambda E} P_Q(sgn(G_0))) \| 
\leq \sqrt{\frac{8 \sigma^2 p \mu d_1 n_1 n_2 \log n}{3 |\Omega|}} \| W_1 \|_\infty \sum_{i=1}^{q} \frac{1}{2^{i-1}} + \| P_{T^\perp} (\frac{\lambda G}{\lambda E} P_Q(sgn(G_0))) \| 
= 2 \sqrt{\frac{8 \sigma^2 p \mu d_1 n_1 n_2 \log n}{3 |\Omega|}} \| W_1 \|_\infty + \| P_{T^\perp} (\frac{\lambda G}{\lambda E} P_Q(sgn(G_0))) \| 
\leq 2 \sqrt{\frac{8 \sigma^2 p \mu d_1 n_1 n_2 \log n}{3 |\Omega|}} \sqrt{\frac{\mu_1 r}{n_1 n_2}} + \sigma^2 \sqrt{\frac{16 \gamma}{n}} 
\leq 2 \sqrt{\frac{8 \sigma^2 p \mu d_1 n_1 n_2 \log n}{3 |\Omega|}} 
\]

when \( \frac{\lambda G}{\lambda E} = 1/n < 1 \). Thus, when \( |\Omega| \geq T_0 \geq \frac{128 \sigma^2 p \mu_1 \mu_1 \log n}{3} \), it could be guaranteed that \( \| P_{T^\perp} (H - \frac{\lambda G}{\lambda E} P_Q(sgn(G_0))) \| \leq \frac{1}{2} \).

\[ \blacksquare \]

### 2.3.2 Sampling Rate for \( \epsilon \)-Recovery

Because the side information matrices \( X \) and \( Y \) may not be full column rank, i.e., satisfy the conditions for exact recovery of \( E_0 \) (or \( F \)), we also analyze the error bound of the proposed Eq.\((2.2)\) and furthermore provide an \( \epsilon \)-recovery sampling rate when the side information matrices are not full column rank or their rank is difficult to attain. The proposed method still achieves a reduced sample complexity in comparison with standard matrix completion methods.
**Theorem 3:** Denote $\|G\|_1 \leq \alpha$, $\|E\|_* \leq \gamma$, and the perfect side feature matrices (containing latent features of $F$) are corrupted with $\Delta X$ and $\Delta Y$ where $\|\Delta X\|_2 \leq s_1$, $\|\Delta Y\|_2 \leq s_2$ and $S = \max(s_1, s_2)$. To $\epsilon$-recover $F$ that the expected loss $\mathbb{E}[l(f, F)] < \epsilon$ for a given arbitrarily small $\epsilon > 0$, $O(\min((\gamma^2 + r\phi^2) \log n, S^2 \alpha r \sqrt{n})/\epsilon^2)$ observations are sufficient for our model to achieve an $\epsilon$-recovery when corrupted factors of side information are bounded.

**Proof.** The expected risk can be bounded by Eq. (2.17) and Eq. (2.18), which has the complexity of $O(\sqrt{\max((\gamma^2 + r\phi^2) \log n, S^2 \alpha r \sqrt{n})/|\Omega|})$. It can be proved that when $|\Omega| = O(\min((\gamma^2 + r\phi^2) \log n, S^2 \alpha r \sqrt{n})/\epsilon^2)$, the expected risk becomes $O(\epsilon)$. 

Theorem 3 can be inferred from the fact that the trace norm of the original data matrix and the $\ell_1$-norm of the interaction parameter matrix affect sample complexity of our model. It meets the intuition that higher rank matrix ought to require more observations to recover. Besides, to the aspect of the discovery of $G$, a sparse interactive matrix can lead to the decrease of the sample complexity, which implies that the side information, even though is not perfect, could be informative enough such that the original matrix can be compressed by sparse coding via the estimated interaction between the features of both row and column entities of the matrix. In our empirical evaluations on real-world datasets including the movie-recommendation and drug-discovery datasets, our experimental results are in conformity with the physical meaning of the features, either intuitionally...
or biologically. The practical recovery performance is also comparable when the side-information matrices are imperfect in comparison with the requirement of the orthonormality [40] or the full-rankness required by our exact recovery analysis. One can also observe Eq. (2) is equivalent to a problem of minimizing just the loss subject to $||G||_1 \leq \alpha$ and $||E||_* \leq \gamma$ where $\alpha$ and $\gamma$ are determined by the choices of $\lambda_E$ and $\lambda_G$ in (2) by the Tikhonov regularization. The larger $\lambda$’s correspond to smaller $\alpha$ and $\gamma$, when $\lambda_G = \infty$, $G$ will be 0 and (2) becomes (1). The ratio of the $\lambda$’s balances between side model and standard matrix completion component with $E$.

To further clarify the $\epsilon$-recovery and its proof, we first consider the optimization problem below that if the perfect feature matrices $X$ and $Y$ are corrupted by $\Delta X$ and $\Delta Y$ and bounded by a constant $||\Delta X||_F \leq s_1$ and $||\Delta Y||_F \leq s_2$, so that we relax the original problem to the following optimization formulation:

$$\min_G \|R_\Omega((X + \Delta X)G(Y + \Delta Y)^T - F)\|_F^2$$

subject to $E - XGY^T \in B(0, \phi)$,

subject to $||G||_1 \leq \alpha, ||E||_* \leq \gamma$.  \hspace{1cm} (2.6)

where $B(0, \phi) \subset \mathbb{R}^{n_1 \times n_2}$ is a ball with the radius of $\phi$ and center at 0.

The matrix entry $F_{ij}$ is assumed to be observed partially $i.i.d.$ from an index set $\{(i_\alpha, j_\alpha)\}_{\alpha=1}^m$ with an unknown distribution.

We denote $\Theta = \{(G, E) \mid ||G||_1 \leq \alpha, ||E||_* \leq \gamma, E = XGY^T\}$ as the feasible solution set, and $\theta = (G, E) \in \Theta$ as any feasible solution. Let $F_\theta(i, j) = x_i Gy_j^T$ be
the estimation function for $F_{ij}$ with $\theta$ as the parameters, and $F_\Theta = \{ f_\theta | \theta \in \Theta \}$ be
the set of feasible functions. Denote the loss function as $l$ where $l(f_\theta(i, j), F_{ij}) = R_\Omega(XGY^T - F)_{i,j}^2$. Then, we consider two “l-risk” quantities: the expected $l$-risk

$$\mathcal{R}_l(f) = \mathbb{E}_{(i,j)}[l(f_\theta(i, j), F_{ij})],$$

and the empirical $l$-risk

$$\hat{\mathcal{R}}_l(f) = \frac{1}{m} \sum_{(i,j)}[l(f_\theta(i, j), F_{ij})].$$

In this notation, our model is to solve for $\theta$ that exactly parameterizes $f^* = \arg\min_{f \in F_\Theta} \mathcal{R}_l(f)$, and it is sufficient to show that the recovery can be attained if $\hat{\mathcal{R}}_l(f^*)$ approaches to zero. Next we implement Rademacher complexity, a learning theoretic tool to measure the complexity of a function class. Then we will derive the sampling rate. To begin with, we cite the following Lemma [50] to bound the expected risk.

**Lemma 10:** *(Bound on Expected risk).* Let $l$ be a loss function with Lipschitz constant $L_l$ in the compact domain respect to its first argument bounded by $B$, and $p$ be a constant where $0 < p < 1$. Let $\mathfrak{R}(F_\Theta)$ be the Rademacher complexity of the function class $F_\Theta$ defined as:

$$\mathfrak{R}(F_\Theta) = \mathbb{E}[\sup_{f \in F_\Theta} \frac{1}{m} \sum_{i=1}^{m} \sigma_i l(f(i_t, j_t), F)]$$

(2.7)

where each $\sigma_i$ takes values $\{-1, +1\}$ with equal probability. Then with the proba-
bility at least $1 - p$, for all $f \in F_\Theta$ we have:

$$R_l(f) \leq \hat{R}_l(f) + 2\mathbb{E}[\mathcal{R}(F_\Theta)] + B\sqrt{\frac{\log \frac{1}{p}}{2m}}. \tag{2.8}$$

In order to upper-bound $R_l$, both $\hat{R}_l$ and model complexity $\mathbb{E}[\mathcal{R}(F_\Theta)]$ need to be upper-bounded. The next key lemma shows that what affect the model complexity term $\mathbb{E}_\Omega[\mathcal{R}(F_\Theta)]$ in matrix completion context.

The Rademacher complexity can be bounded in terms of $\alpha$ and $\gamma$ by the following lemma:

**Lemma 11:** Let $X = \|X\|_2$, $Y = \|Y\|_2$ and $d = \max(d_1, d_2)$,

$$\mathbb{E}[\mathcal{R}(F_\Theta)] \leq 2L_l\alpha XY\sqrt{\frac{\log 2d}{m}} + \sqrt{CBL_l\sqrt{dpr\alpha}(\sqrt{n_1} + \sqrt{n_2})} (s_1Y + s_2X + s_1s_2) \tag{2.9}$$

For proving clearly we firstly introduce Lemma 12 as below, which is a special case of Theorem 1 in [51];

**Lemma 12:** Let $S_W = \{W \in \mathbb{R}^{n \times n} \mid \|W\|_* \leq W\}$ and $a = \max_i \|A_i\|_2$, where

$\{A_i \mid A_i \in \mathbb{R}^{n \times n}\}_{i=1}^m$ is an arbitrary set, then:

$$\mathbb{E}_\sigma[\sup_{W \in S_W} \frac{1}{m} \sum_{i=1}^m \sigma_i \|WA_i\|_*] \leq 2aW\sqrt{\frac{\log 2n}{m}}, \tag{2.10}$$

where $\{\sigma_i \mid i = 1, ..., m\}$ independently take each of their values in $\{-1, +1\}$ with equal probability.

By using Lemma 12 and Rademacher contraction principle (e.g. Lemma in [52]), we can readily prove Lemma 11 as below.
Proof. Denote the random matrix $P \in \mathbb{R}^{n_1 \times n_2}$ with each entry denoted as $P_{ij} = \sum_{\alpha : i_\alpha = i, j_\alpha = j} \sigma_\alpha$, which means the ‘hit-time’ on the $i, j$-th element of $\Omega$, then $\mathfrak{R}(F_\Theta)$ in (2.7) can be split into two components as:

\[ \mathfrak{R}(F_\Theta) = \mathbb{E}_\sigma[\sup_{f \in F_\Theta} \frac{1}{m} \sum_{(i,j)} P_{ij} l(f(i,j), F_{ij})] \]
\[ = \mathbb{E}_\sigma[\sup_{f \in F_\Theta} \frac{1}{m} \sum_{(i,j)} A_{ij} l(f(i,j), F_{ij})] + \mathbb{E}_\sigma[\sup_{f \in F_\Theta} \frac{1}{m} \sum_{(i,j)} B_{ij} l(f(i,j), F_{ij})]. \tag{2.11} \]

In Eq. (2.11) we define

\[ A_{ij} = \begin{cases} P_{ij}, & \text{if } h_{ij} > p \\ 0, & \text{otherwise.} \end{cases} \quad B_{ij} = \begin{cases} 0, & \text{if } h_{ij} > p \\ P_{ij}, & \text{otherwise.} \end{cases} \]

where $h_{ij} = |\{\alpha : i_\alpha = i, j_\alpha = j\}|$ and $p$ is a thresholding value discussed soon.

Recall that $|l(f(i,j), F_{ij})| \leq B$, we can infer the bound of the first term:

\[ \mathbb{E}_\sigma[\sup_{f \in F_\Theta} \frac{1}{m} \sum_{(i,j)} A_{ij} l(f(i,j), F_{ij})] \leq \frac{B}{m} \mathbb{E}_\sigma[\sum_{(i,j)} |A_{ij}|] \leq \frac{B}{\sqrt{p}}, \tag{2.12} \]

where the last inequality is attained from Lemma 10 in [30].

Next we give the bound of the second term in Eq. (2.11) by using Lemma 12. Using Rademacher contraction principle, we have

\[ \mathbb{E}_\sigma[\sup_{f \in F_\Theta} \frac{1}{m} \sum_{(i,j)} B_{ij} l(f(i,j), F_{ij})] \]
\[ \leq \frac{L_l}{m} \mathbb{E}_\sigma[\sup_{\|G\|_1 \leq \alpha} \sum_{(i,j)} B_{ij} x_i G y_j^T + \sup_{\|G\|_1 \leq \alpha} \sum_{(i,j)} B_{ij} \Delta x_i G y_j^T + \sup_{\|G\|_1 \leq \alpha} \sum_{(i,j)} B_{ij} \Delta x_i G \Delta y_j^T]. \tag{2.13} \]
Since \( \|G\|_* \leq \sqrt{\tau} \|G\|_F \leq r \|G\|_2 \leq \sqrt{d} \sqrt{\tau} \|G\|_1 \), for the last three terms we can give the bound as stated below:

\[
\frac{L_l}{m} E_{\sigma} \left[ \sup_{\|G\| \leq \alpha} \sum_{(i,j)} B_{ij} \Delta x_i G y_j^T + \sup_{\|G\| \leq \alpha} \sum_{(i,j)} B_{ij} x_i G \Delta y_j^T + \sup_{\|G\| \leq \alpha} \sum_{(i,j)} B_{ij} \Delta x_i G \Delta y_j^T \right] \\
\leq \frac{L_l}{m} E_{\sigma} \left[ \sup_{\|G\| \leq \alpha} \|B\|_2 \|\Delta X G Y T\|_* + \sup_{\|G\| \leq \alpha} \|B\|_2 \|X G \Delta Y T\|_* \right] \\
+ \sup_{\|G\| \leq \alpha} \|B\|_2 \|\Delta X G \Delta Y T\|_* \\
\leq \frac{L_l}{m} E_{\sigma} \left[ \sup_{\|G\| \leq \alpha} \|\Delta X G Y T\|_* + \sup_{\|G\| \leq \alpha} \|X G \Delta Y T\|_* + \sup_{\|G\| \leq \alpha} \|\Delta X G Y T\|_* \right] \\
\leq \frac{L_l}{m} E_{\sigma} \left[ \sup_{\|G\| \leq \alpha} \|\Delta X G Y T\|_* + \sup_{\|G\| \leq \alpha} \|X G \Delta Y T\|_* \right] \\
+ \sup_{\|G\| \leq \alpha} \|\Delta X G \Delta Y T\|_* \\
\leq \frac{\sqrt{d} \alpha L_l}{m} E_{\sigma} \left[ \|\Delta X\|_2 \|Y T\|_2 + \|X\|_2 \|\Delta Y\|_2 + \|\Delta X\|_2 \|\Delta Y T\|_2 \right] \\
\leq \frac{\sqrt{d} \alpha L_l}{m} (s_1 Y + s_2 X + s_1 s_2) E_{\sigma} \left[ \|B\|_2 \right] \\
\leq 2.2 C \frac{\sqrt{d} \alpha L_l}{m} (s_1 Y + s_2 X + s_1 s_2). 
\]

(2.14)

where the first and the fourth inequalities are derived by applying Holder’s inequality and using the fact that the spectral norm is the dual to the nuclear norm, and the last inequality is derived by applying Lemma 11 in [30] where.

Next we bound the term \( E_{\sigma} [\sup_{\|G\| \leq \alpha} \sum_{(i,j)} B_{ij} x_i G y_j^T] \) in Eq. (2.13). We
have
\[ \frac{L_i}{m} \mathbb{E}_\sigma \left[ \sup_{\|G\|_1 \leq \alpha} \sum_{\alpha=1}^{m} \sigma_\alpha x_{i\alpha} G y_{j\alpha}^T \right] \leq L_i \mathbb{E}_\sigma \left[ \sup_{\|G\|_1 \leq \alpha} \frac{1}{m} \sum_{\alpha=1}^{m} \sigma_\alpha tr(x_{i\alpha} G y_{j\alpha}^T) \right] \]
\[ \leq L_i \mathbb{E}_\sigma \left[ \sup_{\|G\|_1 \leq \alpha} \frac{1}{m} \sum_{\alpha=1}^{m} \sigma_\alpha tr(G y_{j\alpha}^T x_{i\alpha}) \right] \leq 2L_i \sqrt{d} \alpha r \max_{i,j} \|y_j^T x_i\|_2 \sqrt{\log d} \frac{m}{m} \] \[
(2.15)
\]
\[ \leq 2L_i \sqrt{d} \alpha r \sqrt{\log d} \frac{m}{m}. \]

Combining the upper-bounds in Eq. (2.12), Eq. (2.14) and Eq. (2.15), by letting \( p = \frac{mB}{2.2CL\alpha r \sqrt{d_1 d_2 dp(\sqrt{n_1} + \sqrt{n_2})(s_1 Y + s_2 X + s_1 s_2)}} \) in Eq. (2.14) we can get the bound for \( \mathbb{E}[\mathcal{R}(F_\Theta)] \) as:
\[ \mathbb{E}[\mathcal{R}(F_\Theta)] \leq 2L_i \alpha r \sqrt{d} \mathcal{X} \mathcal{Y} \sqrt{\frac{\log d}{m}} + \sqrt{\frac{CBL_i \sqrt{d} \alpha \sqrt{\max(n_1, n_2)} (s_1 Y + s_2 X + s_1 s_2)}{m}}. \]
\[
(2.16)
\]

Lemma 12 clarifies the upper-bound of the complexity of \( f \). Additionally, with proper chosen \( \lambda_E \) and \( \lambda_G \), the empirical risk \( \hat{\mathcal{R}}(f) \) can be sufficiently small. Therefore we conclude the upper bound of \( \mathcal{R}(f^*) \) as below.

**Lemma 13:** With a probability at least \( 1 - p \), the expected \( l \)-risk of an optimal solution will be bounded by:
\[ \mathcal{R}(f^*) \leq 2L_i \alpha r \sqrt{d} \mathcal{X} \mathcal{Y} \sqrt{\frac{\log d}{m}} + \sqrt{\frac{8CBL_i \sqrt{d} \alpha \sqrt{n}}{m}} (s_1 Y + s_2 X + s_1 s_2) + B \sqrt{\frac{\log \frac{1}{p}}{2m}}. \]
\[
(2.17)
\]

Considering another direction to upper-bound our model, we give Lemma 14 as follows.
Lemma 14: Let $\|G\|_1 \leq \alpha$, $\|E\|_* \leq \gamma$, we have

$$\mathbb{E}[\mathcal{R}(F_\Theta)] \leq 2L_l[\gamma \sqrt{\frac{\log 2n}{m}} + \phi \sqrt{r} \sqrt{\frac{\log 2n}{m}} + \alpha r \sqrt{\frac{d \log 2d}{m}} (s_1 \mathcal{Y} + s_2 \mathcal{X} + s_1 s_2)] \quad (2.18)$$

Again, by using Lemma 12 and Rademacher contraction principle, we prove Lemma 14.

**Proof.** $\mathbb{E}(\mathcal{R}(F_\Theta))$ can be bounded as above, we have

$$\mathbb{E}_\sigma \left[ \sup_{f \in F_\Theta} \frac{1}{m} \sum_{\alpha=1}^{m} \sigma_\alpha l(f(i_\alpha, j_\alpha), F_{i_\alpha j_\alpha}) \right] \leq \frac{L_l}{m} \mathbb{E}_\sigma \left[ \sup_{\|G\|_1 \leq \alpha} \sum_{\alpha=1}^{m} \sigma_\alpha x_{i_\alpha} G y_{j_\alpha}^T + \sup_{\|G\|_1 \leq \alpha} \sum_{\alpha=1}^{m} \sigma_\alpha \Delta x_{i_\alpha} G y_{j_\alpha}^T + \right] \quad (2.19)$$

Then one can derive the bound as:

$$\frac{L_l}{m} \mathbb{E}_\sigma \left[ \sup_{\|G\|_1 \leq \alpha} \sum_{\alpha=1}^{m} \sigma_\alpha x_{i_\alpha} G y_{j_\alpha}^T + \sup_{\|G\|_1 \leq \alpha} \sum_{\alpha=1}^{m} \sigma_\alpha \Delta x_{i_\alpha} G y_{j_\alpha}^T + \right]$$

$$\leq L_l \mathbb{E}_\sigma \left[ \sum_{\alpha=1}^{m} \sigma_\alpha tr(\mathcal{E}_{i_\alpha j_\alpha} e_{i_\alpha} e_{j_\alpha}^T) + \sup_{\|\Phi\| \leq \phi} \sum_{\alpha=1}^{m} \sigma_\alpha tr(\Phi_{i_\alpha j_\alpha} e_{i_\alpha} e_{j_\alpha}^T) + \right]$$

$$2L_l \alpha r \sqrt{\frac{d \log 2d}{m}} \max_{i,j} \|y_j \Delta x_i^T\|_2 + \max_{i,j} \|\Delta y_j x_i^T\|_2 + \max_{i,j} \|\Delta y_j \Delta x_i^T\|_2$$

$$\leq 2L_l[\gamma \sqrt{\frac{\log 2n}{m}} + \phi \sqrt{r} \sqrt{\frac{\log 2n}{m}} + \alpha r \sqrt{\frac{d \log 2d}{m}} (s_1 \mathcal{Y} + s_2 \mathcal{X} + s_1 s_2)],$$

where we let $\Phi = E - X G Y^T$. The last equation is derived by applying Lemma 12. Then, we derive another upper bound of $\mathbb{E}[\mathcal{R}(F_\Theta)]$ as

$$\mathbb{E}[\mathcal{R}(F_\Theta)] \leq 2L_l[\gamma \sqrt{\frac{\log 2n}{m}} + \phi \sqrt{r} \sqrt{\frac{\log 2n}{m}} + \alpha r \sqrt{\frac{d \log 2d}{m}} (s_1 \mathcal{Y} + s_2 \mathcal{X} + s_1 s_2)]. \quad (2.21)$$
Then our Theorem 3 can be attained directly from Lemma 13 and Lemma 14.

For the goal of investigating the recovery guarantee under the generalized frame of our work, it is noted that we can replace any norm-regularizers $\|G\|_\sim$ of $G$ satisfying that $\|G\|_\sim \leq \|G\|_1$. Therefore it is feasible to further explore more structural priors in various situation.

When the rank of the original data matrix $r = O(1)$ ($r \ll n$), and correspondingly $\gamma = O(1)$, Theorem 3 points out that only $O(\log n)$ sampling rate is required for an $\epsilon$-recovery. For completion of an arbitrary matrix without the use of side features, the standard matrix completion analysis shows that under certain conditions, one can achieve $O(npoly \log n)$ sample complexity for both $\epsilon$-recovery and perfect recovery, which is higher than our complexity. However, the condition for these existing bounds is that the observed entries follow a certain distribution. Recent studies found that if no specific distribution is pre-assumed for observed entries, $O(n^{3/2})$ sampling rate is sufficient for an $\epsilon$-recovery. Compared to those results, our analysis does not require any assumptions on the distribution of observed entries. When $X$ and $Y$ contain insufficient interaction information about $F$ and $\|G\|_1 = O(n)$, the sample complexity of our method reaches $O(n^{3/2})$ in the worst case, which still guarantees that our model performs with a comparable sample complexity to the standard matrix completion methods.
This shows that our model can be more generalizable.

2.4 Adaptive LADMM Algorithm

In this section, we propose to use an adaptive LADMM algorithm \[55\] to solve problem \((2.2)\). First, we show that the ADMM is applicable in our problem and we then derive LADMM steps. A convergence proof is established to guarantee the performance of our algorithm.

2.4.1 Algorithm

We first re-write \(C = E - X^TGY\) and use it in Eq. \((2.2)\) so that our formulation has a separable structure, and ADMM is applicable. Then the augmented Lagrangian function of \((2.2)\) is given by

\[
\mathcal{L}(E, G, C, M_1, M_2, \beta) = \frac{1}{2} \|C\|_F^2 + \lambda_E \|E\|_* + \lambda_G \|G\|_1 + \langle M_1, R_{\Omega}(E - F) \rangle + \\
+ \langle M_2, E - X^TGY - C \rangle + \frac{\alpha}{2} \|R_{\Omega}(E - F)\|_F^2 + \frac{\beta}{2} \|E - X^TGY - C\|_F^2
\]

(2.22)

where \(M_1, M_2 \in \mathbb{R}^{m \times n}\) are Lagrange multipliers and \(\beta > 0\) is the penalty parameter. Given \(C^k, G^k, E^k, M_1^k\) and \(M_2^k\) at iteration \(k\), each group of the variables yields their respective subproblems:

\[
C^{k+1} = \arg\min_C \mathcal{L}(E^k, G^k, M_2^k, C, \beta_k),
\]

\[
G^{k+1} = \arg\min_G \mathcal{L}(E^k, G, M_2^k, C^{k+1}, \beta_k),
\]

\[
E^{k+1} = \arg\min_E \mathcal{L}(E, G^{k+1}, M_1, M_2, C^{k+1}, \beta_k),
\]

(2.23)
After solving these subproblems, we update the multipliers $M_1$ and $M_2$ as follows;

$$M^{k+1}_1 = M^k_1 + \beta_k \left( R_{\Omega} (E^{k+1} - F) \right),$$

$$M^{k+1}_2 = M^k_2 + \beta_k \left( E^{k+1} - X^T G^{k+1} Y - C^{k+1} \right).$$

(2.24)

Firstly we solve the subproblem for $C$:

$$C^{k+1} = \arg \min_{C} \frac{1}{2} \|C\|^2_F + \langle M_2^k, E^k - X^T G^k Y - C \rangle + \frac{\beta_k}{2} \|E^k - X^T G^k Y - C\|^2_F$$

(2.25)

which has a closed form solution as:

$$C^{k+1} = \frac{\beta_k}{\beta_k + 1} \left( E^k - X^T G^k Y + M^k_2/\beta_k \right)$$

(2.26)

The $G$-subproblem is equivalent to

$$\min_{G} \lambda_G \|G\|_1 + \langle M_2^k, E^k - X^T G^k Y - C^k \rangle + \frac{\beta_k}{2} \|E^k - X^T G^k Y - C^k\|^2_F,$$  

(2.27)

after adding constant term to (2.27) we obtain

$$\min_{G} \lambda_G \|G\|_1 + \frac{\beta_k}{2} \|B^k - X^T G Y - C^k\|^2_F$$

(2.28)

where $B^k = E^k + M^k_2/\beta_k$. By converting the matrix $b$ into a vector $g = \text{vec}(G)$, $\text{vec}(X^T G Y) = (Y^T \otimes X^T) g$. Further we let $b^k = \text{vec}(B^k)$ and $\otimes$ computes the Kronecker product of two matrices. Thus, if we denote $A = (Y^T \otimes X^T)$, the above subproblem becomes:

$$\min_{g} \lambda_G \|g\|_1 + \frac{\beta_k}{2} \|Ag + c^k - b_1^k\|^2_2$$

(2.29)
Since (2.29) does not have a closed-form solution due to a linear operator \( A \), which results in a lasso problem which could be solved iteratively in practice, by utilizing a linearization technique, we have

\[
\frac{1}{2} \| A g + c^k - b_1^k \|^2_2 \approx \frac{1}{2} \| A g^k + c^k - b_1^k \|^2_2 + \langle f_1^k, g - g^k \rangle + \frac{\tau_A}{2} \| g - g^k \|^2_2
\]  

(2.30)

where \( \tau_A > 0 \) is a proximal parameter and

\[
f_1^k = A^T (A g^k + c^k - b_1^k) = A^T (A g^k + c^k - e^k / \beta_k)
\]  

(2.31)

is the gradient of \( \frac{1}{2} \| A g + c^k - b_1^k \|^2_2 \) at \( g_k \). Eq. (20) can be re-written as:

\[
\min_g \lambda_G \| g \|_1 + \frac{\beta_k \tau_A}{2} \| g - [g^k - f_1^k / \tau_A] \|^2_2
\]  

(2.32)

Obviously the closed-form solution is:

\[
g^{k+1} = \max(|g^k - f_1^k / \tau_A| - \frac{\lambda_G}{\tau_A \beta_k}, 0) \odot sgn(g^k - f_1^k / \tau_A)
\]  

(2.33)

Next, The \( E \)-subproblem is

\[
\min_E \lambda_E \| E \|_* + \langle M_1^k, R_\Omega(E - F) \rangle + \frac{\beta_k}{2} \| R_\Omega(E - F) \|^2_F
\]  

\[
+ \langle M_2^k, E - X^T G^{k+1} Y - C^k \rangle + \frac{\beta_k}{2} \| E - X^T G^{k+1} Y - C^k \|^2_F
\]  

(2.34)

which we can reformulate as:

\[
\min_E \lambda_E \| E \|_* + \frac{\beta_k}{2} \| R_\Omega(E - B_2^k) \|^2_F + \frac{\beta_k}{2} \| E - B_3^k \|^2_F
\]  

(2.35)

where \( B_2^k = R_\Omega(F - M_1^k / \beta_k) \) and \( B_3^k = X^T G^{k+1} Y + C^k - M_2^k / \beta_k \). After linearization, the problem can be approximately optimized by:

\[
\min_E \lambda_E \| E \|_* + \frac{\beta_k \tau_B}{2} \| E - (E^k - f_2^k / \tau_B) \|^2_F + \frac{\beta_k \tau_B}{2} \| E - (E^k - f_3^k / \tau_B) \|^2_F
\]  

(2.36)
where \( f_2^k \) and \( f_3^k \) are the gradients of \( \frac{1}{2} \| R_\Omega (E - B_2^k) \|_F^2 \) and \( \frac{1}{2} \| E - B_3^k \|_F^2 \) at \( E^k \), which are illustrated below:

\[
\begin{align*}
    f_2^k &= R_\Omega (E^k - B_2^k) = R_\Omega (E^k - F + M_1^k / \beta_k), \\
    f_3^k &= E^k - B_3^k = E^k - X^T G^{k+1} Y - C^k + M_2^k / \beta_k.
\end{align*}
\]

(2.37)

The closed-form solution is then readily obtainable as

\[
E^{k+1} = \text{SVT}(E^k - (f_2^k + f_3^k) / (2 \tau_B), \lambda E^k / (2 \beta_k \tau_B)).
\]

(2.38)

To summarize, Given \( C^k, G^k, E^k, M_1^k \) and \( M_2^k \), our algorithm for solving the next iteration \((C, E, G, M_1, M_2)\) is organized in Algorithm 2.

The adaptive parameter in Algorithm 2 are \( \rho > 1 \), and \( \beta_{\text{max}} \) controls the upper bound of \( \{\beta_k\} \). The operator \( \text{reshape}(g) \) converts a vector \( g \in \mathbb{R}^{ab} \) into a matrix \( G \in \mathbb{R}^{a \times b} \), which is the inverse operator of \( \text{vec}(G) \). The operator \( \text{SVT}(E, t) \) is the singular value thresholding process defined in [56] for soft-thresholding the singular values of an arbitrary matrix \( E \) by \( t \). The matrix \( A = Y^T \otimes X^T \) where \( \otimes \) indicates the Kronecker product.

It turns out that the adaptive LADMM can effectively solve the proposed optimization problem in several aspects. First, the convergence of the commonly-used block-wise coordinate descent (BCD) method, sometimes referred to as alternating minimization methods, requires typically that the optimization problem be strictly convex (or quasiconvex and hemivariate). The strongest result for BCD in the convex but non-differentiable problem so far is established in [43] which requires the alternating subproblems to be optimized in each iteration to its unique
Algorithm 1 The adaptive LADMM algorithm to solve $C^k$, $G^k$, $E^k$, $k = 1, \ldots, K$

**Input:** $X$, $Y$ and $R_\Omega(F)$ with parameters $\lambda_G$, $\lambda_E$, $\tau_A$, $\tau_B$, $\rho$ and $\beta_{max}$.

**Output:** $C, G, E$;

1: Initialize $E^0$, $k = 0$,

repeat;

2: $C^{k+1} = \frac{\beta_k}{\beta_{k+1}}(E^k - X^T G^k Y + M^k_2/\beta_k)$;

3: $G^{k+1} = \text{reshape}(\max(|g^k - f^k_1/\tau_A| - \frac{\lambda_G}{\tau_A \beta_k}, 0) \odot \text{sgn}(g^k - f^k_1/\tau_A))$ where $f^k_1 = A^T (Ag^k + c^k - b^k_1)$.

4: $E^{k+1} = \text{SVT}(E^k - (f^k_2 + f^k_3)/(2\tau_B), \lambda_E/2(\beta_k \tau_B))$ where $f^k_2 = R_\Omega(E^k - F + M^k_1/\beta_k)$; $f^k_3 = E^k - X^T G^{k+1} Y - C^{k+1} + M^k_2/\beta_k$.

5: $M_1^{k+1} = M_1^k + \beta_k(R_\Omega(E^{k+1} - F))$.

6: $M_2^{k+1} = M_2^k + \beta_k(E^{k+1} - X^T G^{k+1} Y - C^{k+1})$.

7: $\beta_{k+1} = \min(\beta_{max}, \rho \beta_k)$.

8: $k = k + 1$ until convergence;

Return $C, G, E$;

optimal solution. This requirement is often restrictive in practice. In contrast, our convex (not strictly convex) problem can be solved by the adaptive LADMM with the guarantee of converging to global optimal solution, which is characterized in Theorem 4. Second, two of the sub-optimization problems are non-smooth due to the $\ell_1$-norm or the nuclear norm, so it can be difficult to obtain a closed-form formula to efficiently compute a solution by standard optimization tools; however,
adaptive LADMM utilizes the linearization technique which leads to the closed-form solution for each subproblem, and significantly enhances the efficiency of the iterative process. Third, adaptive LADMM can be practically parallelizable by a similar scheme to that of ADMM. It is also noted that the convergence rate of LADMM and parallelized LADMM is $O(1/K)$ [57] which is in contrast to the BCD method that still lack of clearer theoretical results of its convergence rate while the problem is not strictly convex. Next we provide a rigorous proof of the following theorem that claims the global optimal convergence for our algorithm.

2.4.2 Convergence Analysis

For conveniently writing, we roughly write the Lagrangian function (2.22) in the more compact form as follows:

$$
\mathcal{L}(E, G, C; M_1, M_2, \beta) = \frac{1}{2} \|C\|_F^2 + \lambda_E \|E\|_* + \lambda_G \|G\|_1 +
$$

$$
\langle M, B(E) + A(G) + N(C) - D \rangle + \frac{\beta}{2} \|B(E) + A(G) + N(C) - D\|_F^2
$$

(2.39)

where $B(E) = \begin{pmatrix} R_{\Omega}(E) \\ E \end{pmatrix}$, $A(G) = \begin{pmatrix} 0 \\ -X^TGY \end{pmatrix}$, $N(C) = \begin{pmatrix} 0 \\ C \end{pmatrix}$ and $D = \begin{pmatrix} R_{\Omega}(F) \\ 0 \end{pmatrix}$. $M$ performs the equivalent multiplier function as $\begin{pmatrix} M_1 \\ M_2 \end{pmatrix}$.

The proving framework consists of three steps: The first step includes Lemma [15] for the proof of Lemma [16] and Theorem [4]; the next step is the proof of Lemma [16] which indicates the convergence of our algorithm; the third step is to
clarify our algorithm converges to a global minimum point of problem (4), shown in Theorem 4.

**Lemma 15:** Let $G^k, E^k, C^k$ be the optimal solution for each individual subproblem at the $k$-th iteration, then it satisfies that

$$-\beta_k \tau_A (G^{k+1} - G^k) - A^* (\bar{M}^{k+1}) \in \partial \|G^{k+1}\|_1, -\beta_k \tau_B (E^{k+1} - E^k) - B^* (\hat{M}^{k+1}) \in \partial \|E^{k+1}\|_*$$

where $\bar{M}^{k+1} = M^k + \beta_k [A(G^k) + B(E^{k+1}) + N(C^0) - D]$, $\hat{M}^{k+1} = M^k + \beta_k [A(G^{k+1}) + B(E^{k+1}) + N(C^k) - D]$, here $\partial \| \cdot \|$ denotes the subgradient of an arbitrary $\| \cdot \|$, and $A^*$ is the adjoint operator of $A$.

Note that $A^* = A^T$ if $A$ is a linear operator while $A(X) = AX$. This Lemma is directly derived from the optimality conditions of subproblems when solving $G$ and $E$ individually.

Next we present the lemma implying the convergence.

**Lemma 16:** Given $\beta_k$ is non-decreasing and upper bounded, $\tau_A > \|A\|^2$, $\tau_B > \|B\|^2$, and $(G^*, E^*, C^*, M^*)$ is any KKT point of problem 2.22, then:

$$\{ \tau_A \|G^k - G^*\|_F^2 - \|A(G^k - G^*)\|_F^2 + \tau_B \|E^k - E^*\|_F^2 + \|C^k - C^*\|_F $$

$$- \|N(C^k - C^*)\|_F^2 + \beta_k^{-2} \|M^k - M^*\|_F^2 \}$$

is non-increasing; and

$$\|G^k - G^{k+1}\|_F^2 \to 0, \|E^k - E^{k+1}\|_F^2 \to 0, \|C^k - C^{k+1}\|_F^2 \to 0, \|M^k - M^{k+1}\|_F^2 \to 0.$$

**Proof.** For proving the non-increase property of the first sequence, it is equiv-
alent to investigate the following inequality:

\[
\tau_A \|G^{k+1} - G^*\|_F^2 - \|A(G^{k+1} - G^*)\|_F^2 + \tau_B \|E^{k+1} - E^*\|_F^2 + \|C^{k+1} - C^*\|_F^2 \\
- \|N(C^{k+1} - C^*)\|_F^2 + \beta_k^2 \|M^{k+1} - M^*\|_F^2 - (\tau_A \|G^k - G^*\|_F^2 - \|A(G^k - G^*)\|_F^2) \\
+ \tau_B \|E^k - E^*\|_F^2 + \|C^k - C^*\|_F^2 - \|N(C^k - C^*)\|_F^2 + \beta_k^2 \|M^k - M^*\|_F^2) \leq 0
\]

(2.40)

For proving the above inequality, we list several facts to be used:

\[
M^{k+1} = M^k + \beta_k (A(G^{k+1}) + B(E^{k+1}) + N(C^{k+1}) - D),
\]

\[
2 \left\langle G^{k+1} - G^*, G^{k+1} - G^k \right\rangle = \|G^{k+1} - G^*\|_F^2 - \|G^k - G^*\|_F^2 + \|G^{k+1} - G^k\|_F^2,
\]

\[
A(G^*) + B(E^*) + N(C^*) - D = 0,
\]

\[
\langle M, A(G) \rangle = \langle A^*(M), G \rangle, \langle M, B(E) \rangle = \langle B^*(M), E \rangle.
\]

(2.41)

Then, we have

\[
\tau_A \|G^{k+1} - G^*\|_F^2 - \|A(G^{k+1} - G^*)\|_F^2 + \tau_B \|E^{k+1} - E^*\|_F^2 + \|C^{k+1} - C^*\|_F^2 \\
- \|N(C^{k+1} - C^*)\|_F^2 + \beta_k^2 \|M^{k+1} - M^*\|_F^2 - (\tau_A \|G^k - G^*\|_F^2 - \|A(G^k - G^*)\|_F^2) \\
+ \tau_B \|E^k - E^*\|_F^2 + \|C^k - C^*\|_F^2 - \|N(C^k - C^*)\|_F^2 + \beta_k^2 \|M^k - M^*\|_F^2) \\
= 2\tau_A \left\langle G^{k+1} - G^*, G^{k+1} - G^k \right\rangle - 2 \left\langle A(G^{k+1} - G^*), A(G^{k+1} - G^k) \right\rangle \\
+ \|A(G^{k+1} - G^k)\|_F^2 + 2\tau_B \left\langle E^{k+1} - E^*, E^{k+1} - E^k \right\rangle - \tau_B \|E^{k+1} - E^k\|_F^2 \\
2\tau_N \left\langle C^{k+1} - C^*, C^{k+1} - C^k \right\rangle - 2 \left\langle N(C^{k+1} - C^*), N(C^{k+1} - C^k) \right\rangle \\
+ \|N(C^{k+1} - C^k)\|_F^2 - \tau_A \|C^{k+1} - C^k\|_F^2 - \tau_A \|G^{k+1} - G^k\|_F^2
\]

(2.42)
By reorganizing Eq. (2.42), we have

\[
\begin{align*}
\tau_A \| G^{k+1} - G^* \|^2_F - \| \mathcal{A}(G^{k+1} - G^*) \|^2_F &+ \tau_B \| E^{k+1} - E^* \|^2_F + \| C^{k+1} - C^* \|^2_F \\
- \| \mathcal{N}(C^{k+1} - C^*) \|^2_F &+ \beta_k^{-2} \| M^{k+1} - M^* \|^2_F - \| \mathcal{A}(G^k - G^*) \|^2_F \\
+ \tau_B \| E^k - E^* \|^2_F &+ \| C^k - C^* \|^2_F - \| \mathcal{N}(C^k - C^*) \|^2_F + \beta_k^{-2} \| M^k - M^* \|^2_F \\
= - \{ \beta_k^{-2} \| M^{k+1} - M^k \|^2_F &+ \tau_B \| E^{k+1} - E^k \|^2_F - 2\beta_k^{-1} \langle M^{k+1} - M^k, B(E^{k+1} - E^k) \rangle \} \\
- \{ \tau_A \| G^{k+1} - G^k \|^2_F &- \| \mathcal{A}(G^{k+1} - G^k) \|^2_F \} - \{ \| C^{k+1} - C^k \|^2_F &- \| \mathcal{N}(C^{k+1} - C^k) \|^2_F \} \\
- 2\beta_k^{-1} \langle G^{k+1} - G^* &- \mathcal{A}^*(M^{k+1}) - \mathcal{A}^*(M^*) \rangle \\
- 2\beta_k^{-1} \langle E^{k+1} - E^* &- B^*(M^{k+1}) + B^*(M^*) \rangle \\
- 2\beta_k^{-1} \langle C^{k+1} - C^* &- N^*(M^{k+1}) + N^*(M^*) \rangle \\
\} \geq 0
\end{align*}
\]

since \( \tau_A \geq \| \mathcal{A} \|^2 \), we can check that

\[
\tau_A \| \| - \| \mathcal{A}(\cdot) \|^2_F \geq 0.
\]

and similarly it is clear that

\[
\beta_k^{-2} \| M^{k+1} - M^k \|^2_F + \tau_B \| E^{k+1} - E^k \|^2_F - 2\beta_k^{-1} \langle M^{k+1} - M^k, B(E^{k+1} - E^k) \rangle \geq 0
\]

The last three terms in Eq. (2.43) are nonnegative due to Lemma 15 and the monotonicity of subgradient mapping. So the non-increasing property in Lemma 16 is proved. Because of the non-increasing property and non-negativity, it has a limit. Then we can see that

\[
\tau_A \| G^{k+1} - G^k \|^2_F - \| \mathcal{A}(G^{k+1} - G^k) \|^2_F \to 0,
\]
\[
\|C^{k+1} - C^k\|_F^2 - \|N(C^{k+1} - C^k)\|_F^2 \to 0.
\]

\[
\beta_k^{-2} \|M^{k+1} - M^k\|_F^2 + \tau_B \|E^{k+1} - E^k\|_F^2 - 2\beta_k^{-1} \langle M^{k+1} - M^k, B(E^{k+1} - E^k) \rangle \to 0
\]
due to their non-negativity. So \(\|G^{k+1} - G^k\|_F \to 0\) and \(\|C^{k+1} - C^k\|_F \to 0\) can be obtained from the first two limits. Note that

\[
\beta_k^{-2} \|M^{k+1} - M^k\|_F^2 + \tau_B \|E^{k+1} - E^k\|_F^2 - 2\beta_k^{-1} \langle M^{k+1} - M^k, B(E^{k+1} - E^k) \rangle \\
\geq \beta_k^{-2} \|M^{k+1} - M^k\|_F^2 + \tau_B \|E^{k+1} - E^k\|_F^2 - 2\beta_k^{-1} \|M^{k+1} - M^k\|_F \|B(E^{k+1} - E^k)\|_F \\
= (\beta_k^{-1} \|M^{k+1} - M^k\|_F - \|B(E^{k+1} - E^k)\|_F)^2 + \tau_B \|E^{k+1} - E^k\|_F^2 - \|B(E^{k+1} - E^k)\|_F^2 \\
\geq \tau_B \|E^{k+1} - E^k\|_F^2 - \|B(E^{k+1} - E^k)\|_F^2 \geq 0.
\]

(2.44)

So we have that \(\|E^{k+1} - E^k\|_F \to 0\). Furthermore,

\[
\beta_k^{-2} \|M^{k+1} - M^k\|_F^2 + \tau_B \|E^{k+1} - E^k\|_F^2 - 2\beta_k^{-1} \langle M^{k+1} - M^k, B(E^{k+1} - E^k) \rangle \\
(\beta_k^{-1} \|M^{k+1} - M^k\|_F - \sqrt{\tau_B} \|E^{k+1} - E^k\|_F)^2 + \\
2\beta_k^{-1} (\sqrt{\tau_B} \|M^{k+1} - M^k\|_F \|E^{k+1} - E^k\|_F - \langle M^{k+1} - M^k, B(E^{k+1} - E^k) \rangle) \\
\geq (\beta_k^{-1} \|M^{k+1} - M^k\|_F - \sqrt{\tau_B} \|E^{k+1} - E^k\|_F)^2.
\]

(2.45)

So \(\beta_k^{-2} \|M^{k+1} - M^k\|_F^2 + \tau_B \|E^{k+1} - E^k\|_F^2 - 2\beta_k^{-1} \langle M^{k+1} - M^k, B(E^{k+1} - E^k) \rangle \to 0\).

This results in \(\|M^{k+1} - M^k\|_F \to 0\) noting that \(\|E^{k+1} - E^k\|_F \to 0\).

Based on Lemma 15 and Lemma 16 we can derive the following theorem.
**Theorem 4:** If $\beta_k$ is non-decreasing and upper-bounded, $\tau_A > \|A\|$, and $\tau_B > \|B\|$ then the sequence $\{(C^k, G^k, E^k, M^k)\}$ generated by adaptive LADMM converges to a global optimal point of problem (4).

**Proof.** By Lemma 16, $\{(C^k, G^k, E^k, M^k)\}$ is bounded, hence there is a subsequence that $(C^{k_i}, G^{k_i}, E^{k_i}, M^{k_i}) \to (C^\infty, G^\infty, E^\infty, M^\infty)$. We accomplish the proof in two steps.

We first prove that $(C^\infty, G^\infty, E^\infty, M^\infty)$ is a KKT point of our optimization problem.

By Lemma 16, $A(G^{k+1}) + B(E^{k+1}) + N(C^{k+1}) - D = \beta_k^{-1}(M^{k+1} - M^k) \to 0$. This shows that any accumulation point of $\{(C^k, G^k, E^k, M^k)\}$ is a feasible solution.

Without the loss of generality, suppose $\lambda_G = \lambda_E = \frac{1}{2}$. by letting $k = k_i - 1$ in Lemma 15 and the subgradient definition, we have

$$
\|G^{k_i}\|_1 + \|E^{k_i}\|_* + \|C^{k_i}\|_F \\
\leq \|G^*\|_1 + \|E^*\|_* + \|C^*\|_F + \left\langle G^{k_i} - G^*, -\beta_{k_i-1}\tau_A(G^{k_i} - G^{k_i-1}) - A^*(\tilde{M}^{k_i}) \right\rangle \\
+ \left\langle E^{k_i} - E^*, -\beta_{k_i-1}\tau_B(E^{k_i} - E^{k_i-1}) - B^*(\tilde{M}^{k_i}) \right\rangle \\
+ \left\langle C^{k_i} - C^*, -\beta_{k_i-1}(C^{k_i} - C^{k_i-1}) - N^*(M^{k_i}) \right\rangle
$$

(2.46)
Suppose $i \to \infty$, from Lemma 16, we can observe $G^{k_i} - G^{k_i-1} \to 0$ so that

$$\|G^\infty\|_1 + \|E^\infty\|_* + \|C^\infty\|_F^2 \leq \|G^*\|_1 + \|E^*\|_* + \|C^*\|_F^2$$

since both $(C^\infty, G^\infty, E^\infty)$ and $(C^*, G^*, E^*)$ are feasible solutions. So we conclude that $(C^\infty, G^\infty, E^\infty)$ is an optimal solution to (4).

Similarly we let $k = k_i - 1$ in Lemma 1 and by the definition of subgradient, we have

$$\|G\|_1 \geq \|G^{k_i}\|_1 + \left< G - G^{k_i}, -\beta_{k_i-1} \tau_A(G^{k_i} - G^{k_i-1}) - A^*(\bar{M}^{k_i}) \right>$$

for any $G$. Fix $G$ and let $i \to \infty$, we see that

$$\|G\|_1 \geq \|G^\infty\|_1 + \left< G - G^\infty, -A^*(\bar{M}^\infty) \right>$$

for any $G$. So $-A^*(\bar{M}^\infty) \in \partial\|G^\infty\|_1$. Similarly, $-B^*(\bar{M}^\infty) \in \partial\|E^\infty\|_*$. It is also not difficult to check that $-N^*(\bar{M}^\infty) = C$. Therefore, $(C^\infty, G^\infty, E^\infty, M^\infty)$ is a KKT point of problem (4).
Next we prove that the whole sequence of \( \{(C^k, E^k, G^k, M^k)\} \) converges to \( \{(C^\infty, E^\infty, G^\infty, M^\infty)\} \).

By choosing \((C^*, G^*, E^*, M^*) = (C^\infty, G^\infty, E^\infty, M^\infty)\) in Lemma 16, we have

\[
\tau_A \|G^k - G^\infty\|_F^2 + \tau_B \|G^k - G^\infty\|_F^2 + \beta_{k_i}^{-2} \|M^k - M^\infty\|_F^2 \to 0.
\]

By Lemma 16, we readily have

\[
\tau_A \|G^k - G^\infty\|_F^2 + \|A(G^k - G^\infty)\|_F^2 + \tau_B \|M^k - M^\infty\|_F^2 + \beta_{k_i}^{-2} \|M^k - M^\infty\|_F^2 \to 0.
\]

So \((C^k, G^k, E^k, M^k) \to (C^\infty, G^\infty, E^\infty, M^\infty)\). Since \((C^\infty, G^\infty, E^\infty, M^\infty)\) can be an arbitrary accumulation point of \((C^k, G^k, E^k, M^k)\), we can conclude that

\((C^k, G^k, E^k, M^k)\) converges to a KKT point. Furthermore, the KKT point is a necessary and sufficient condition for global minimum point in the convex problem.

2.5 Experimental Results

We validated our method in both simulations and the analysis of two real world datasets: MovieLens (movie rating) and NCI-DREAM (drug discovery) datasets. Three most recent matrix completion methods that also utilized side information were compared against our method. These methods included MAXIDE [40], IMC [39] and DirtyIMC [11]. For all methods, hyperparameters were tuned via the same cross validation process. The performance of all methods was measured by the relative-mean squared error (RMSE) calculated as:

\[
\|R_{\Omega}(X^TGY - F)\|_F^2 / \|R_{\Omega}(F)\|_F^2.
\]

For both synthetic and real-world datasets, we randomly set \( q \) percent of the components in each observed matrix \( F \) to be missing and all com-
pared methods were compared on the basis of these missing values. For each \( q \), we repeated the procedure six times and the average performance of each method was reported.

2.5.1 Synthetic Datasets

We created two different simulation tests with and without full row rank of \( X \) and \( Y \). For all the datasets that were synthesized, we first randomly created \( X \) and \( Y \). In order to make our simulations mimic real situations where distributions of side features can be heterogeneous, data for each feature in both \( X \) and \( Y \) were generated according to a distribution that was randomly selected from Gaussian, Poisson and Gamma distributions. Then we randomly created the \( G \) matrices. Each simulated \( G \) had 20% non-zero components. Though the location of these non-zero components were randomly picked and their values were generated by multiplying a value drawn from \( \mathcal{N}(0,1) \) by 100, we chose those matrices that showed full or high rank. Lastly, we generated \( F \) with \( F = X^T G Y + N \) where \( N \) represents the noise and each component \( N_{i,j} \) was drawn from \( \mathcal{N}(0,1) \). For each simulated \( F \), we ran all methods with \( q \in [20\% - 80\%] \) with 10% increase interval.

We studied all methods in three different settings, which were labeled as synthetic experiment I, II and III in our results. In the first setting, the dimension of \( X \) and \( Y \) was set to \( 15 \times 50 \) and \( 20 \times 140 \) and all features in these two matrices were randomly generated to make them full row rank. Both the last two settings
corresponded to the second test where $\mathbf{X}$ and $\mathbf{Y}$ were not full row rank. The dimension of $\mathbf{X}$ and $\mathbf{Y}$ was set to $16 \times 50$, $21 \times 140$ and $20 \times 50$, $25 \times 140$, respectively, for these two settings. In both settings, the first 15 features in $\mathbf{X}$ and 20 features in $\mathbf{Y}$ were randomly created. Then the remaining features were generated by arbitrary linear combinations of these randomly created features.

For all three settings, we repeated the methods 10 times and reported mean and standard deviation of RMSE. The results on RMSE for all compared methods are presented in Figure 2.1.

**Fig. 2.1:** The Comparison of RMSE for Experiments I, II, and III.

Our approach outperformed all other compared methods significantly in almost all these settings. When the missing rate $q$ increased, the RMSE of our method growed much slower than other methods. We explored the rank of the recovered $\mathbf{G}$ and $\mathbf{E}$ in the first setting. For all methods, the corresponding $\mathbf{G}$ and $\mathbf{E}$ that gave the best performance were examined. The ranks of $\mathbf{G}$ and $\mathbf{E}$ from our method, MAXIDE, IMC, DirtyIMC were 20, 15, 1, 1 and 15, 15, 1, 2,
respectively. These results suggested that incorporating the strong prior of low ran\nG might hurt the prediction/recovery performance. The retrieved model matrices \nG of all compared methods together with the true G are plotted in Figure 2.2. Only our method was able to recover the true G and all the other methods merely found approximations.

![Heatmap of True G and Recovered G matrices in Experiment I.](image)

**Fig. 2.2:** The heatmap of True G and Recovered G matrices in Experiment I.

### 2.5.2 Benchmark Datasets

5.2.1. **MovieLens.** This dataset was downloaded from [58] and contained 100,000 user ratings (integers from 1 to 5) from 943 users on 1682 movies. There were 20 movie features such as genre and release date, as well as 24 user features describing users’ demographic information such as age and gender. We compared all methods with four different q values: 20-50%. The RMSE values of each method are shown in Table 3.1 which shows that our approach significantly outperformed other methods, especially when q was large. Figure 2.3 shows the
constructed $G$ matrix that shows some interesting observations. For instance, male users tend to rate action, science fiction, thriller and war movies high but low for children’s movies, exhibiting some common intuitions.

5.2.2 NCI-DREAM Challenge. The data on the reactions of 46 breast cancer cell lines to 26 drugs and the expression data of 18633 genes for all the cell lines were provided by NCI-DREAM Challenge [59]. For each drug, we had 14 features that describes their chemical and physical properties such as molecular weight, XLogP3 and hydrogen bond donor count, and were downloaded from National Center for Biotechnology Information (http://pubchem.ncbi.nlm.nih.gov/). For the cell line features, we ran principle component analysis (PCA) and used the top 45 principle components that accounted for more than 99.99% of the total data variance. We compared the four different methods with four different $q$ values: 20-50%. The RMSE values of all methods are provided in Table 3.1 where our method again shows the best performance. We examined the ranks of both $G$ and $E$ obtained by all the methods. They were 15, 20, 1, 1 for $G$ and 2, 20, 1, 2 for $E$, respectively, for our approach, MAXIDE, IMC and DirtyIMC in sequence. This demonstrates that a low rank $E$ but a high rank $G$ give the best performance on this dataset. In other words, requiring a low rank $G$ may hurt the performance of recovering a low rank $E$.

The constructed $G$ by our method is plotted in Figure 2.4, where columns represent cell line features (i.e., principle components) and rows represent drug
Table 2.1: The Comparison of RMSE values of different methods on real-world datasets.

<table>
<thead>
<tr>
<th>Method</th>
<th>MovieLens Data</th>
<th>NCI-Dream Challenge</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>20%  30%  40%  50%</td>
<td>20%  30%  40%  50%</td>
</tr>
<tr>
<td>Our approach</td>
<td>0.2744 0.2759 0.2815 0.3037</td>
<td>0.1225 0.1581 0.1591 0.1848</td>
</tr>
<tr>
<td>MAXIDE</td>
<td>0.6641 0.7592 0.7972 0.8248</td>
<td>0.1467 0.1719 0.1941 0.2020</td>
</tr>
<tr>
<td>IMC</td>
<td>0.9351 0.9413 0.9470 0.9516</td>
<td>0.5664 0.5982 0.6401 0.6729</td>
</tr>
<tr>
<td>DirtyIMC</td>
<td>0.7885 0.8052 0.8216 0.8361</td>
<td>0.4504 0.4881 0.5473 0.5869</td>
</tr>
</tbody>
</table>

features. Please refer to the supplementary material for the names of these features. According to this figure, drug features: XlogP (F2), hydrogen bond donor (HBD) (F3), Hydrogen bond acceptor (HBA) (F4) and Rotatable Bond number (F5) all played important roles in drug sensitivity. This result aligns well with biological knowledge, as all these four features are very important descriptors for cellular entry and retention.

2.5.3 Case Study: Inference of missing diagnostic criteria of substance use disorders

In this section, we explore the use of our approach to impute missing phenotypes for subjects with substance use disorders (SUDs).

Although SUDs are heritable, few genetic risk factors for them have been
Fig. 2.3: HeatMap of $G$ for MovieLens

Fig. 2.4: HeatMap of $\text{sgn}(G) \log(|G|)$ for NCI-DREAM for a better illustration
identified, in part due to the small sample sizes of study populations. To address this limitation, researchers have aggregated subjects from multiple existing genetic studies, but these subjects can have missing phenotypic information, including diagnostic criteria for certain substances that were not originally a focus of study. Recent advances in addiction neurobiology have shown that comorbid SUDs (e.g., the abuse of multiple substances) have similar genetic determinants, which makes our method possible to infer missing SUD diagnostic criteria using criteria from related SUDs and patient genotypes as side information.

In this case study, our dataset contains 7,189 subjects were aggregated from three family-based or case-control genetic studies of cocaine use disorder (CUD) and opioid use disorder (OUD). Of the 7,189 subjects, 7,008 self-reported having used cocaine and were included in a GWAS of CUD \( [60] \); 4,843 self-reported having used an opioid and were included in a GWAS of OUD \( [61] \). In total, 4,662 subjects self-reported having used both cocaine and opioids; of that number, 3,441 subjects who in their lives had used opioids and cocaine more than 11 times were included in the evaluation of the proposed approach to infer cocaine and opioid use behaviors.

We used the proposed approach to analyze the data of 3,441 SUD subjects, which means that we had a fully observed matrix \( \mathbf{F} \). To mimic the real-life situation where the use of a substance might not be reported, thus missing all criteria for that substance, we randomly selected \( q \) percent of SUD patients, for whom
Table 2.2: The comparison of imputation results by different methods on the Opioid-Cocaine SUD dataset.

<table>
<thead>
<tr>
<th>q</th>
<th>LADMM</th>
<th>DirtyIMC</th>
<th>IMC</th>
<th>MAXIDE</th>
<th>NM</th>
</tr>
</thead>
<tbody>
<tr>
<td>20%</td>
<td>0.236</td>
<td>0.297</td>
<td><em>0.230</em></td>
<td>0.235</td>
<td>0.567</td>
</tr>
<tr>
<td>40%</td>
<td><em>0.226</em></td>
<td>0.298</td>
<td>0.235</td>
<td>0.236</td>
<td>0.582</td>
</tr>
<tr>
<td>60%</td>
<td><em>0.228</em></td>
<td>0.301</td>
<td>0.237</td>
<td>0.235</td>
<td>0.581</td>
</tr>
<tr>
<td>80%</td>
<td><em>0.236</em></td>
<td>0.303</td>
<td>0.239</td>
<td>0.241</td>
<td>0.585</td>
</tr>
<tr>
<td>100%</td>
<td><em>0.223</em></td>
<td>0.303</td>
<td>0.246</td>
<td>0.242</td>
<td>0.574</td>
</tr>
</tbody>
</table>

we removed randomly either CUD or OUD diagnostic criteria. We evaluated the performance with 5 different $q$ values: 20%, 40%, 60%, 80%, and 100%. Note that when $q = 100\%$, every patient had either CUD or OUD diagnostic criteria removed but not both. There were 383 genetic variants selected in our GWAS, which were used as side information in $X$. We computed the correlations between each pair of the 22 criteria using all patients and used the correlation matrix as $Y$.

In addition to the four competing methods used in the simulations, we also compared our method to a naive method (NM) in which the missing criteria of a disorder were filled by copying over the patient’s diagnostic symptoms for the other substance. The proposed algorithm was evaluated using the same training and tuning procedure as used in the simulations. The imputation accuracy of all methods are shown in Table 3. The best performance was again obtained by our approach in comparison with other imputation methods.
**Fig. 2.5:** The recovered $\mathbf{G}$ by our method for the Cocaine-Opioid SUD dataset.

Columns C1-C11 represent 11 CUD diagnostic criteria, columns O1-O11 represent 11 OUD diagnostic criteria. C1/O1: Larger or longer Cocaine/Opioid use than intended; C2/O2: Failed efforts to stop on Cocaine/Opioid; C3/O3: Much time spent in Cocaine/Opioid related activities; C4/O4: Strong desire to use Cocaine/Opioid; C5/O5: Cocaine/Opioid effect interfered with life; C6/O6: Cocaine/Opioid use despite of its interference; C7/O7: Major activities reduced by Cocaine/Opioid use; C8/O8: Physical hazard caused by Cocaine/Opioid use; C9/O9: Cocaine/Opioid use knowing it threatening health; C10/O10: Cocaine/Opioid tolerance; C11/O11: Cocaine/Opioid withdrawal syndrome.
**Fig. 2.6:** The top 30 rows of the recovered $G$ by our method for the Cocaine-Opioid SUD dataset. Columns correspond to the diagnostic criteria for CUD and OUD whereas rows correspond to the candidate genetic variants. The right-hand side gives the locations of these genetic variants and their p-values obtained in the GWAS.
Figure 5 shows the parameter matrix $G$ (of size $383 \times 22$) obtained by our algorithm. Note that the genetic variants were ordered in ascending fashion with respect to their association p-values reported in the GWAS, so the most significant variants identified in the GWAS are at the top of the figure. A more saturated color reflects a stronger interaction between a specific genetic variant and a diagnostic criterion. Red denotes positive interactions and blue denotes negative interactions. We further expanded first 30 rows of Figure 5 into Figure 6. It can be observed from Figure 5 and Figure 6 that the first 30 most significant variants from the GWAS had the largest magnitude interactions with the criteria. Another observation on Figure 4 is that genetic variants with lower (stronger) association p-values are more likely to show stronger interactions with the phenotypes.

In Figure 6, 9 of the variants and their interactions with diagnostic criteria received high weights when imputing the unreported criteria. It is also interesting to observe that the interactions between all these variants and the opioid diagnostic criterion “opioid use despite its interference” were negatively proportional to the imputed values of missing criteria for CUD, which may need further investigation in a future study. The SNP rs1481605 at base pair (bp) 13,519,829 on chromosome 8 received the highest weights for its interactions with all 22 phenotypes in the model. Moreover, this SNP was associated with both OUD and CUD at genome-wide significant level ($p < 5 \times 10^{-8}$) in the GWAS. This SNP is located at the downstream (94,032 bp away) of gene $C8orf48$, which, according
to data from GTEx (available at https://www.gtexportal.org/home/), expresses in many brain tissues, and its expression in nucleus accumbens is the highest, as illustrated in Figure 7 copied from the GTEx website.

**Fig. 2.7:** Gene expression distribution (RPKM, Reads per Kilobase Million) of C8orf48 across human tissues.
2.6 Summary

In this chapter, we have proposed a novel sparse inductive model that utilizes side features describing the row and column entities of a partially observed matrix to predict its missing entries. This method models the linear predictive power of side features as well as interaction between the features of row and column entities. Theoretical analysis shows that this model has advantages of reduced sample complexity over classical matrix completion methods, requiring only $O(\log N)$ observed entries to achieve a perfect recovery of the original matrix when the side features reflect the true latent feature space of the matrix. When the side features are less informative, our model requires $O(\log N)$ observations for an $\epsilon$-recovery of the matrix. Unlike early methods that use a BCD algorithm, we have developed a LADMM algorithm to optimize the proposed formulation. Given the optimization problem is convex, this algorithm can converge to a global solution. Computational results demonstrate the superior performance of this method over three recent methods. Future work includes the examination of other types and quality of side information and the understanding of whether our method will benefit a variety of relevant problems, such as multi-label learning, and semi-supervised clustering etc.
Chapter 3

Statistical Approximation and Learning of Kolmogorov Coupled Net

3.1 Background

Mixture density estimation learns how data are generated and represents the whole distribution by combining individual mixtures. To infer the unknown data probability density function (pdf), denoted by $p(x)$ where $x \in \mathbb{R}^d$, from the finite number of samples of $x$, one can approximate $p(x)$ by $q(x)$ where $q(x)$ is composite of multiple $q_i$’s. Probabilistic mixture models aim to find the $K$ pdf’s $q_1:K$ ($K$ is predefined manually or determined by nonparametric approaches) from a function space (or a hypothesis space) $Q$, such that $q(x) = q_i(x)$ if $x \in$ cluster $i$ (i.e. hard clustering), or $q(x) = \sum_{i=1}^{K} \pi_i q_i(x)$ if $\sum_i \pi_i = 1, 0 \leq \pi_{1:K} \leq 1$ (i.e. soft clustering). In general, most generative model based mixture methods solve the similar essential problem in approximation theory:

$$\min_q \: \text{dist}(p,q), \quad \text{subject to } \sum_{i \in I} \pi_i q_i \| \sum_{i \in I} \pi_i = 1, 0 \leq \pi_i \leq 1, q_i \in Q,$$

(3.1)
where $\mathcal{I}$ contains the indexes of the functions selected from $\mathcal{Q}$ for use in the approximation.

Existing mixture models typically assume that $q_i$’s in (3.1) follow specific analytic forms [3, 4]. For instance, Gaussian Mixture Model discretizes the target distribution into a linear combination of $K$ multivariate-Gaussian-distributed clusters. Though being successful in tractable cases, as data explode rapidly in the most recent decades, this kind of hypothesis space $\mathcal{Q}$ can restrict a model’s capacity to approximate clusters with complicated distributions.

To enrich the expressiveness of the space $\mathcal{Q}$, mixture models with hierarchical infrastructure are developed and achieve promising improvement on clustering tasks. Latent Dirichlet Allocation (LDA) methods [62, 63] propose a three-layer Bayesian model to cluster discrete data such as text corpora. However, for many other tasks such as clustering images, either new representation of the data or careful modifications to the probabilistic graphical structures have to be done in order to use LDA, which remains to highly depend on an expert’s domain knowledge [64]. Other pivotal models for clustering are mainly based on Deep Neural Nets (DNN). One of the most widely used DNN methods learns new data representation by Denoising Autoencoder (DAE) [65]. By including the DAE loss, recent DNN clustering methods attempt to construct proper representation of samples that preserves the subspace (clustering) prior [66] or graph prior [6], and then cluster the samples in the new representation using standard clustering.
methods such as K-means and spectral clustering. The two-stage methodology is formed on the belief that either the underlying graph prior or the subspace prior holds. However, it could hardly be effective when no such an assumption can be deployed. Similarly, in other deep clustering methods \[67, 68, 69\], it is challenging to determine an embedding that properly maps from the original space to another space so the sample cluster structures are invariant, and often the clustering result relies highly on model selection.

A recent generative model named Generative Adversarial Network (GAN) shows significant capability in high dimensional distribution approximation \[70, 71\]. The pdf's in the hypothesis space \(\mathcal{Q}\) are explicitly parameterized by DNN coefficients so it directly generates samples instead of using any analytical form. A problem with GANs is that they often suffer the so-called mode collapse problem. To address the mode collapse problem, ADAGAN \[72\], MAD-GAN \[73\] and MGGAN \[74\] improve the approximation by a linear combination written as \[q(x) = \sum_{i=1}^{K} \pi_i q_i(x),\] where \(q_i\)'s are the pdf's induced by multiple GANs. ADAGAN, analogous to standard boosting methods but different from the spirit of clustering, computes the weights \(\pi_i\) in a greedy approximation procedure until the algorithm converges; For MAD-GAN and MGGAN, \(\pi_i\)'s are either set to all ones or pre-defined as the portion of samples generated from each generator. However, because the universal approximation capability of GANs will lead to infinite-many solutions, the output is likely to be unstable due to the ill-defined optimization
problems. As an extreme case, all clusters can be potentially approximated by a single generator.

In this work, a variational optimization model is proposed, namely Kolmogorov Coupled Nets (KCN), which underlines the fundamental nature of data with the local grouping structure (clusters) in two ways: (i) without using $Q$ of functions with analytical forms, we identify distinctive clusters locally by maximizing the pairwise distance between any two clusters’ pdf’s whereas requiring these pdf’s to globally approximate the overall distribution $p$. (ii) The cluster complexity must be well regulated and estimated to avoid an ill-posed optimization problem.

To achieve (i), the linear combination of $q_i$’s in (3.1) is further generalized, which is widely adopted by mixture density estimation, by nonlinearly approximating the target distribution $p(x)$ by $q(x) = \sum_{i=1}^{K} C_i(x)q_i(x)$. In the coupling of $C_i$ and $q_i$, $C_i$ is a function implemented by the standard deep neural network (DNN), capturing the local weight of $q_i$ for any data point $x$, while $q_{1:K}$ constitute $K$ mixtures selected from $Q$ for the best approximation of $p$. In order to explore a more expressive hypothesis space $Q$, each $q_i$ in our method is parameterized by an individual GAN, which means, precisely, the output of the GAN follows a distribution with the pdf $q_i$. We also state that $q_i$ is induced by the GAN. Our proposed optimization problem maximizes the distance between each pair of distributions generated by GANs, measured by Jensen-Shannon-divergence (JSD),
but minimizes the distance between the data distribution $p$ and its approximation $q$.

To achieve (ii) of effectively controlling the complexity of $q_i \in Q$, we propose a new Gaussian Process (GP)-smooth neuron to be used in GANs to interpolate the discrete approximation $q$. Such a type of neurons allows us to directly estimate the pdf induced by a GAN, so the complexity of the pdf can be measured by Surrogate Kolmogorov Complexity (SKC) from approximation theory. Surrogate Kolmogorov Complexity stems from the Kolmogorov Problem of determining the minimal number of unimodal pdf’s needed to approximate a complex pdf. A differential equation system can be used to characterize the Kolmogorov Problem and the steady states of this system are used to calculate the SKC. We can effectively regularize the capacity of a GAN by restricting the SKC of the GAN.

3.2 The Proposed Formulation

Let $p$ and $q : D_x \subseteq \mathbb{R}^d \rightarrow \mathbb{R}$ where $D_x = \{x | p(x) \neq 0\}$ is defined as the support of pdf $p$, and $q$ is the pdf to approximate the complex true $p$. In probability theory, $dist(\cdot)$ in (3.1) can be instantiated by measures such as Jensen-Shannon divergence or Wasserstein distance. The weights $\pi_i$’s from (3.1) uniquely represent the discretized approximation of $p$. In mixture density estimation, $\mathcal{I}$ consists of the indexes of clusters and the pdf $q_i$ describes the cluster $i$. Since $\pi_i$’s are constant on the domain $D_x$, previous research in approximation theory discussed in [75].
suggests a better estimation of the local geometry of $p$ by modifying $\pi_i$ to be adaptive based on $x$, given that the complexity of functions in the space $Q$ is well regularized by certain measure. For instance, $Q = \{\text{polynomials of degree} \leq r\}$ when approximating a function via piecewise approximation. Therefore, using exactly the piecewise strategy, one can re-express the approximation $q$ in (3.1) via a mixture by

$$q = \sum_{i=1}^{K} \pi_i(x_j)q_i, \text{ s.t. } \pi_i(x_j) = \begin{cases} 1, & \text{if } x_j \in D_i, \\ 0, & \text{if } x_j \notin D_i, \end{cases} \forall i \in [K], j \in [N]$$

(3.2)

where the number of mixtures is $K$ and the number of observations is $N$. In (3.2), equivalent to hard-clustering, $D_i \subseteq D_x$ denotes the domain of cluster $i$, $\forall i \in [K]$.

For most practical problems, only finitely $N$ samples $x_{1:N}$ in $D_x$ are observed from the underlying pdf $p(x)$. Estimating a discrete $\pi_i(x_j)$ is difficult whenever $D_x$ is continuous. Therefore, one can interpolate the discrete function $\pi_i(x)$ by learning a continuous manifold parameterized as a neural network function of enough capacity, denoted as $C_i(x)$, which leads to a nonlinear approximation as

$$q = \sum_{i=1}^{K} C_i(x)q_i, \text{ s.t. } C_i(x) \in [0, 1] (\forall i \in [K]), \sum_{i=1}^{K} C_i(x) = 1.$$ 

(3.3)

In (3.3), instead of specifying an explicit analytical form of $q_i$, more flexible GANs are adopted, so that each $q_i$ is induced by the $i$-th generator function $G_{\theta_i}(\xi)$ which is parameterized by $\theta_i$. Here $\xi$ is a random Gaussian or Uniform noise variable as used in the classic GAN.
A GAN consists of two competing networks: a generator network $G$ that generates data examples from random samples of a known distribution; a discriminator network $D$ that distinguishes a generated sample from observed data samples. The minimax objective for GAN fights the adversary between $G$ and $D$ as follows:

$$\min_G \max_D \mathbb{E}_{x \sim p(x)}[\log D(x)] + \mathbb{E}_{z \sim q(z)} \log(1 - D(z)).$$  \hspace{1cm} (3.4)

In (3.4) we adopt the JSD loss as a special case when $D$ has enough capacity \[70\]. Remark that it is generalizable to other metrics like Wasserstein Metric. Substituting (3.3) into (3.4) for $G$ yields a discrete $G$ with multiple components, which introduces a multi-player game between multiple generators $G_{1:K}$ and the discriminators $D$ as well as $C$:

$$\min_{C_{1:K}, G_{1:K}} \max_D \mathbb{E}_{x \sim p(x)} \left\{ \log D(x) + \mathbb{E}_{z \sim \sum_i (C_i(x)q_i(z))} [\log(1 - D(z))] \right\}
\text{s.t. } C_i(x) \in [0, 1], \sum_{i=1}^K C_i(x) = 1, \ G_i(\xi) \text{ has pdf } q_i(z) \text{ where } \xi \sim \mathcal{N}(0, I) \ (\forall i \in [K]).$$  \hspace{1cm} (3.5)

In approximation theory, the functions, e.g., $q_{1:K}$ in $Q$ have to be under certain capacity control, or otherwise the data distribution $p$ itself can be a function in $Q$. Especially, when $q_i$ is implemented by a GAN, due to the universal approximation property of a GAN \[76\], the feasible solution of (3.6) may have an extremely unbalanced decomposition of $q$ (e.g., $q_k$ is close to $p$ and $q_{j \neq k} = 0$) if without control. Prior works in approximation theory \[75, 77\] commonly address this issue by constraining elements $q_{1:K}$ in $Q$ by orthogonality and low-complexity.
We first characterize the orthogonality constraint in the functional space induced by GANs and discuss more on complexity in Section 4.

First it is natural to define the orthogonality of $q_1:K$ respectively as

$$\langle q_i, q_j \rangle = \int_{D_x} q_i(z)q_j(z)dz = 0, \forall i, j \in [K], i \neq j. \quad (3.6)$$

From (3.6) one can deduce that $\{z|q_i(z) > 0\} \cap \{z|q_j(z) > 0\}$ is a set of measure zero, which can be represented by $\text{JSD}(q_i \parallel q_j) = 1$ as well. Therefore, as $C_{1:K}$ are introduced to recognize from which generator an arbitrary sample $z$ is generated, inspired from the JSD loss between $p$ and $q$ induced by $D$, we aim to maximize $\text{JSD}(q_i || q_j)$ in expectation:

$$\max_{C_{1:K}, q_{1:K}}\sum_{i=1}^{K} \left\{ \mathbb{E}_{z \sim q_i(z)} [\log C_i(z)] + \sum_{j \neq i} \mathbb{E}_{z \sim q_j(z)} [\log(1 - C_i(z))] \right\} \quad (3.7)$$

Hence, the overall optimization problem can be written as follows to also control capacity of the coupled GANs:

$$\min_{\{G_i, C_i\}_{i=1}^{K}} \max_{D} \mathbb{E}_{x \sim p(x)} \{ \log D(x) + \mathbb{E}_{z \sim \sum_{i=1}^{K} C_i(x)q_i(z)} [\log(1 - D(z))] \}$$

$$- \lambda_C \sum_{i=1}^{K} \left\{ \mathbb{E}_{z \sim q_i(z)} [\log C_i(z)] + \sum_{j \neq i} \mathbb{E}_{z \sim q_j(z)} [\log(1 - C_i(z))] \right\}$$

$$\text{s.t. } C_i(x) \in [0, 1], \sum_{i=1}^{K} C_i(x) = 1, G_i(\xi) \text{ has pdf } q_i(z) \text{ where } \xi \sim \mathcal{N}(0, I) (\forall i \in [K]) \quad (3.8)$$

where $\lambda_C$ is a positive constant to balance between the two objectives.

Alternatively, (3.8) can be viewed as a multi-player game ruled by two referees $D$ and $\{C_i\}_{i=1}^{K}$. While encouraging the homogeneity within a cluster and the
heterogeneity between clusters through (3.7), the game also requires each player to compete at $D$ so its own $C_i$ is assigned.

The proposed method has four folds of advantage. First, because the method learns explicit forms of $C_{1:K}$, they can be directly used to assign new examples $x$ to clusters. Second, when the generator neurons use continuous activation functions, the overall function of $G$ is continuous, thus difficult to approximate non-continuous $p$, such as a mixture of truncated Gaussians. If the domain in $D_x$ where $p$ is defined contains multiple isolated connected sets as discussed in [78], it is difficult to use a single continuous function $G$ to approximate. With the coupling of multiple $G_i$’s in our method, however, this is no longer an issue. Third, compared with MAD-GAN [73] and MGGAN [74], our method does not depend on any pre-specified weight on each generator so that it is capable to cluster samples with no accessibility to the cluster proportion. Fourth, in Section 4, we prove that the adaptive weight functions $C_{1:K}$ not only learn the assignment of each cluster, but also provide guidance for learning $G_{1:K}$ which identify the generative models for each cluster in the data support $D_x$.

3.3 Capacity Control by GP-smooth Neurons and Surrogate

Kolmogorov Complexity

As discussed earlier, a single GAN without capacity control can approximate extremely complex distributions. Hence, the orthogonality of $q_{1:K}$ by itself cannot
guarantee a well-balanced cluster solution of (3.8); it is also possible to see high sensitivity to the initialization. To control the complexity of $G$, it is effective to enforce smoothness on the neurons. This section proposes a new type of GP-smooth neurons. By adding the GP-smooth layer to a GAN, there is a partial differential equation system that characterizes the dynamics of the GAN. The steady states of this system correspond to the modes of a pdf $q_i$. The number of modes of each pdf $q_i$ in (3.8) estimates the Surrogate Kolmogorov Complexity of a generator. The Surrogate Kolmogorov Complexity reflects the complexity concept of the Kolmogorov Problem in approximation theory [79].

3.3.1 GP-smooth Neurons

Previous nonlinear approximation research has shown that the smoothness of functions in $Q$ corresponds to the approximation ability [75]. One of the best-known ways to describe function smoothness is through differentiability. We propose to include a new GP-smooth layer on top of each generative network in our method, so that the output $z_i$ from $G_i$ can follow a Gaussian process over the function $G_i(\xi)$ with the mean of $\bar{G}_i(\xi)$ and the variance of $\sigma^2 I$, i.e., $G_i(\xi) \sim GP(\bar{G}_i(\xi), \sigma^2 I)$. In other words, the original generator layers learn the mean function $\bar{G}_i(\xi) \sim \bar{q}_i(z_i)$, the last GP-smooth layer translates that into an output whose pdf $q_i(z_i)$ has its smoothness controlled by varying $\sigma$. Increasing $\sigma$ results in a lower resolution of the generator.
Formally, we define the GP-smooth neuron as follows:

**Definition 1:** (GP-smooth neuron) A GP-smooth neuron $\varrho(x, \sigma)$ with the input $x$ is expressed as

$$\varrho(x, \sigma) = \sigma \zeta + (1 - \sigma)x, \text{ s.t. } \zeta \sim \mathcal{N}(x, I),$$

where $\sigma$ denotes the smoothness of $\varrho$.

After adding this layer of GP-smooth neurons to the original generator denoted by $\bar{G}_i$, $z_i$, can be efficiently sampled as

$$z_i = G_i(\xi) = \sigma \zeta + (1 - \sigma)\bar{G}_i(\xi), \text{ s.t. } \zeta \sim \text{GP}(\bar{G}_i(\xi), I), \xi \sim \mathcal{N}(0, I),$$

the corresponding pdf $q_i$ can be obtained in an integral form by the convolution operation:

$$q_i(z_i, \sigma) = \bar{q}_i(z_i) * g(z_i, \sigma) = \int \frac{\bar{q}_i(z_i)}{(\sqrt{2\pi\sigma})^d} e^{-\frac{\|z_i - \bar{z}_i\|^2}{2\sigma^2}} d\bar{z}_i$$

where $g(z_i, \sigma)$ is the pdf of $\mathcal{N}(z_i, \sigma^2 I)$. Despite the intractable $\bar{q}_i$, one can generate from $\bar{q}_i$ a sample set $\{\bar{z}_i(:, j)\}$ in which $\bar{z}_i(:, j)$ denotes the $j$-th sample. With $n_i$ samples concatenating into a matrix $\bar{Z}_i \in \mathbb{R}^{d \times n_i}$, we interpolate the pdf and obtain its empirical discrete approximation of (3.11) as

$$q_i(z_i, \sigma) = \frac{1}{n_i} \sum_{j=1}^{n_i} \frac{1}{(\sqrt{2\pi\sigma})^d} e^{-\frac{\|z_i - \bar{z}_i(:, j)\|^2}{2\sigma^2}}$$

where $\bar{z}_i(:, j)$ denotes the $j$-th column-vector in $\bar{Z}_i$, $\forall j \in [n_i]$. A computable pdf Eq. (3.12) is induced by a GAN with controllable smoothness by $\sigma$. Note that
when $\sigma = 0$, $G_i$ goes back to the original GAN; when $\sigma$ is sufficiently large, the distribution induced by $G_i$ approaches to a Gaussian distribution assuming that the image of $G_i$ is compact. Assume that the target pdf $p$ is smooth, it is reasonable to further require the desired smoothness of $q_{1:K}$ to be bounded, so there exists a value $M > 0$, such that $M < \sigma$ when learning the smoothness parameter $\sigma$ while back-propagating each generator. Alternatively, one can also choose to inject a regularizer $\Omega(\sigma) = -\lambda_\sigma \sigma$ into the objective of (3.8), as existing evidence suggests the equivalence of the two regularization approaches with the correspondence between $\lambda_\sigma$ and $M$.

### 3.3.2 Surrogate Kolmogorov Complexity

**Precondition.** Let us assume that we use a sufficient number of nodes $d$ in the code layer for $z$. The output samples of generator $i$ may likely be lower rank-$l$ ($l \ll d$). We precondition $Z_i$ using a singular value decomposition $Z_i = U_i \Sigma_i W_i$ where $\Sigma_i \in \mathbb{R}^{l \times l}$ is full-rank diagonal and $U_i \in \mathbb{R}^{d \times l}$ and $W_i \in \mathbb{R}^{l \times n_i}$ do not, respectively, include the left and right singular vectors corresponding to the singular value 0.

Denote $V_i = \Sigma_i W_i$. As defined by (3.10) the GP-smooth neuron implements:

$$z_{i,(:j)} = g(v_{i,(:j)}, \sigma) = \sigma U_i \varsigma + (1 - \sigma) U_i v_{i,(:j)} = U_i (\sigma \varsigma + (1 - \sigma) v_{i,(:j)}) = U_i \hat{v}_{i,(:j)},$$

s.t. $\varsigma \sim \mathcal{N}(v_{i,(:j)}, I)$, $\hat{v}_{i,(:j)} = \sigma \varsigma + (1 - \sigma) v_{i,(:j)}$.

$$(3.13)$$

The precondition procedure (3.13) can be used to reduce the dimension in the orthogonal transform and accelerates the subsequent complexity estimation. More-
over, the pdf on $\hat{v}_i$ keeps the geometry of pdf on $z_i$ invariant as characterized by the following proposition.

**Proposition 1:** If $z_i$ is calculated by (3.13), and $z_i$ follows the pdf $q_i(z_i)$, then

$$q_i(z_i) = q_i(\hat{v}_i), \quad \text{for all } i \in [K]. \tag{3.14}$$

This proposition establishes that one can investigate the distribution of $\hat{v}_i$ instead of $z_i$ in a high dimensional space. Especially, $q_i(\hat{v}_i)$ preserves the number of modes of $q_i(z_i)$ through the transform (3.13).

**Proof.**

According to Section 4.2.1, one can alternatively obtain the augmented version of SVD on $\bar{Z}_i$ as:

$$\bar{Z}_i = \bar{U}_i \Sigma_i \bar{W}_i = \begin{bmatrix} U_i & U_i^\perp \end{bmatrix} \begin{bmatrix} \Sigma_i & 0 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} W_i \\ W_i^\perp \end{bmatrix}, \Sigma_i \in \mathbb{R}^{d \times d}, U_i \in \mathbb{R}^{d \times d}, W_i \in \mathbb{R}^{d \times n_i}. \tag{3.15}$$

Therefore, we have

$$z_i(:,j) = q(v_i(:,j), \sigma) = \sigma \begin{bmatrix} U_i & U_i^\perp \end{bmatrix} \begin{bmatrix} \varsigma \\ 0 \end{bmatrix} + (1 - \sigma) \begin{bmatrix} U_i & U_i^\perp \end{bmatrix} \begin{bmatrix} v_i(:,j) \\ 0 \end{bmatrix} = \begin{bmatrix} U_i & U_i^\perp \end{bmatrix} \begin{bmatrix} \hat{v}_i(:,j) \\ 0 \end{bmatrix} = U_i \hat{v}_i(:,j)$$

s.t. $\varsigma \sim \mathcal{N}(v_i(:,j), I), \hat{v}_i(:,j) = \sigma \varsigma + (1 - \sigma)v_i(:,j)$.

Given the joint pdf $q_i(z_i)$ of $z_i(:,j)$ and the transformation $\bar{U}_i$ on $v_i(:,j)$ is invertible, the pdf of $v_i$ is

$$q_i(v_i) = |J|q_i(z_i)$$
where $J = \bar{U}_i$. Hence, we can conclude that the pdf $q_i(v_i) = q_i(z_i)$ with the transformation $\bar{U}_i$.

Estimation of Surrogate Kolmogorov Complexity. Because the GP-smooth layer helps to interpolate the pdf induced by a generator network $G$, we are now ready to estimate the Surrogate Kolmogorov Complexity of $G$. A few recent works examined the global complexity of GANs based on generalization theory \cite{80, 81, 82} and then regularized the complexity of a GAN by learning a function that encodes real data samples into the input space of GAN, which encourages capturing more modes. We move beyond the global and empirical complexity analysis, and estimate a GAN’s complexity (more precisely, the generator’s complexity) from a topological perspective. To address the generative clustering problem, we focus on the modalities (modes) of the pdf induced by $G$. The Kolmogorov Complexity in approximation theory (Problem 1.2 in \cite{79}) determines the smallest number of unimodal (single-mode) pdfs that are needed in a linear combination to approximate a target pdf \cite{83}. Specifically, the Kolmogorov Problem is defined as follows:

**Definition 2:** (Kolmogorov Problem). Determine the smallest integer $k_n$ for which there exists a density $f^n$ with $k_n$ modes and whose distribution $F^n$ satisfies

$$d_{KO}(E_n, F^n) \leq qu(n, \alpha, d_{KO}),$$

(3.15)
where $d_{KO}$ is defined by $d_{KO}(F,G) = \sup\{x : |F(x) - G(x)|\}$, and $qu(n, \alpha, d_{KO})$ denotes the $\alpha$-quantile of the random variable $d_{KO}(F_n, F)$.

We slightly modify the problem and define the Surrogate Kolmogorov Complexity $R(G)$ of a generator $G$:

**Definition 3:** Let $Q$ be a space of pdf functions, and any $q \in Q$ is induced by a generator $G$ and is used to approximate $p$ that has a support $D_p$. We define $R(G)$ as the Surrogate Kolmogorov Complexity (SKC) of $G$ in terms of $p$ iff $R(G)$ is the number of modes of $\hat{q}$ where $\hat{q} = \min_{q \in Q} \|p - q\|$.

The SKC bounds the solution of the Kolmogorov Problem from above, and can be more easily estimated. In a high dimensional space, the number of modes can exceed the number of unimodal components (see an example in [84]). Although there have been methods developed [79, 85] for solving the original Kolmogorov Problem in 1-dimensional case, high-dimensional cases are underexplored.

Let us first introduce some important definitions. For $i = 1, \cdots, K$, we re-express $v_i$, preconditioned from $z_i$ in (3.13) and (3.14), as a time-variant function $v_i(t, v_{i,0})$, where $v_{i,0}$ is the initial value at the time $t = 0$. This $v_i(t, v_{i,0})$ is the solution of the partial differential equation system of the pdf $q_i(v_i, \sigma)$ as follows:

$$
\frac{dv_i}{dt} = \frac{\partial q_i(v_i, \sigma)}{\partial v_i}, \quad v_i(0) = v_{i,0}.
$$

(3.16)

and a steady state $v^*_i$ of the system and the set of the steady states $V_i$ are defined
as
\[ V_i = \{ v_i^* \in \mathbb{R}^d : \lim_{t \to \infty} v_i(t, v_i, 0) = v_i^* \}. \] (3.17)

One can easily verify that the steady-states satisfy \( \frac{\partial q_i(v_i, \sigma)}{\partial v_i} = 0 \). Therefore, the set \( V_i \) of (3.16) except the saddle points of \( q_i(v_i, \sigma) \) contains all local extrema. Substituting (3.11) into (3.16) and employing the Euler method [86], we solve (3.16) by the following numerical steps:

\[ v_i(t + 1) = v_i(t) + \rho \frac{1}{\sigma^2} \int q_i(v) (v - v_i(t)) e^{-\frac{\| v_i(t) - v \|^2}{2\sigma^2}} dv, \quad v_i(0) = v_{i,0}. \] (3.18)

The integral in (3.18) can be estimated based on (3.12), and hence

\[ v_i(t + 1) = v_i(t) + \rho \frac{1}{\sigma^2 N} \sum_{j=1}^{N} \frac{v_i(t) - v_{i,(i,j)}}{(\sigma \sqrt{2\pi})^d} e^{-\frac{\| v_i(t) - v_{i,(i,j)} \|^2}{2\sigma^2}}, \quad v_i(0) = v_{i,0}. \] (3.19)

where \( \rho \) is the step length. We use the gradient-ascent algorithm to solve (3.19) and the gradient-based method almost never converge to a saddle point when the step size is sufficiently small [87]. In other words, we can obtain the modes of \( q_i \).

The convergence criterion used in our setting is \( \| v_i(t + 1) - v_i(t) \|_2 \leq \epsilon \) where \( \epsilon \) is small (e.g., \( 10^{-3} \)). To reduce run-time, note that the process of reaching a steady state of (3.19) from each sample as the initial value can be parallelized. Thus, we can efficiently estimate the SKC of \( G_i \), \( R(G_i) = |V_i| \), i.e., the volume of the set containing all obtained modes. Now, the proposed (3.8) can have further capacity control of \( G_i \)'s by restricting the SKC. We summarize the procedure for training the proposed Kolmogorov Coupled Nets (KCN) in Algorithm 2.
Algorithm 2 KCN

**Input:** Sample set \( \{x_1, \cdots, x_N\} \), \( \lambda_C \), \( \lambda_\sigma \), the number of clusters \( K \), the maximum number of iterations \( \text{iter}_{\text{max}} \), the desired SKC of the \( K \) mixtures as \( r_{1:K} \).

**Output:** \( G_{1:K}, C_{1:K} \), and the actual cluster assignment of \( x_i, i = 1, \cdots, N \).

1: Randomly initialize \( G_{1:K}, C_{1:K} \) and \( D \). \( \text{iter} = 0 \),

repeat;

2: Sample a minibatch of \( n \) data points \( (x_{(1)}, \cdots, x_{(n)}) \) from the data pdf \( p \);

3: Sample a minibatch of \( m \) data points \( (z_{i,(1)}, \cdots, z_{i,(m)}) \) from the pdf \( q_i \) for each \( i \in [K] \);

4: Update \( C_{1:K} \) and \( D \) by Gradient-Descent on (3.8);

5: Compute the \( R(G_1), \cdots, R(G_K) \) via (3.19) after the preconditioning scheme (3.14).

6: Calculate \( C_i(x_{(a)}) = C_i(x_{(a)}) \times (r_i/R(G_i)) \) for all real data samples.

7: Update \( G_{1:K} \) by Gradient-Descent on (3.8).

8: \( \text{iter} = \text{iter} + 1 \) until \( \text{iter} > \text{iter}_{\text{max}} \).
3.4 Recovery Analysis

We first show in Theorem 5 that clusters can be identified when our model reaches a Nash equilibrium, assuming all $C_{1:K}$, $D$, and $G_{1:K}$ have enough capacity. With the smoothness constraint on $G_{1:K}$, Theorem 7 proves that the $\varepsilon$-approximate equilibrium as defined in [80] can be reached. In other words, our method with generators of controllable capacity can approximate each cluster’s distribution as well as the overall data distribution with a controllable tolerance. The tolerance is controlled by the smoothness parameter. We begin by introducing Theorem 5 as below.

**Theorem 5:** Suppose $C_{1:K}$ and $D$ have enough capacity and the pdfs of $K$ true clusters are $p_{1:K}$ with the number of modes respectively as $r_{1:K}$. The global optimum $D^*$ and $C^* = C^*_{1:K}$ for our problem is:

$$
D^*(x) = \frac{p(x)}{p(x) + \sum_{i=1}^{K} T_i q_i(x)}, \quad C^*_k(x) = \frac{q_k(x)}{\sum_{i \in [K]} q_i(x)}, \quad \forall k \in [K]
$$

(3.20)

where $T_i = \int_x C_i(x)p(x)dx \leq 1$ assuming $p(x)$ is piecewise-integrable. Moreover, if $G_{1:K}$ have enough capacity, the pdfs $q^*_{1:K}$ induced by the global optimum $G^*_{1:K}$ satisfy $q^*_{1:k} = p_{1:k}$ when $r_{1:k} = R(G_{1:K})$ and $p_{1:K}$ have separated supports.

Theorem 5 reveals that KCN is capable of recovering the true components $p_{1:K}$ of the target pdf $p$ knowing the complexity measure, without any unnecessarily strong assumption on a pre-defined analytical form of $q_i$. The integral weights $T_{1:K}$ give the proportion or capacity of each cluster w.r.t. the entire population.
As discussed earlier, $C_{1:K}$ are able to guide $G_{1:K}$ with adaptive weights on the region where the generator should focus on to learn instead of parameterizing in the global region. By assuming that $G_{1:K}$ have enough capacity, we mean that there exists a function $\tilde{q}_i$ induced by $\tilde{G}_i$ and $\sigma$ so that $p$ can be written as the convolutions of $\tilde{q}_i$ and $g(\cdot, \sigma)$ (3.11). The theorem also links the smoothness measure $R$ of $p$ and the recovery guarantee of our method.

**Proof.**

**Optimal $D^*$:** First, for simplicity we define

$$
\mathcal{L}(D, C_{1:K}, G_{1:K}) = \mathbb{E}_{x \sim p(x)} \{ \log D(x) + \mathbb{E}_{z \sim \sum_i (C_i(x)q_i)(z)} [\log(1 - D(z))] \} \\
- \lambda C \sum_{i=1}^{K} \left\{ \mathbb{E}_{z \sim q_i(z)} [\log C_i(z)] + \sum_{j \neq i} \mathbb{E}_{z \sim q_j(z)} [\log(1 - C_i(z))] \right\} .
$$

(3.21)
With that definition, one can obtain

\[ \mathcal{L}_D(D, C_{1:K}, G_{1:K}) \]

\[ = \mathbb{E}_{x \sim p} \left[ \log D(x) + \mathbb{E}_{z \sim \sum_i C_i(x)q_i \log(1 - D(z))} \right] \]

\[ = \int_x \left[ \log D(x) + \int_z \log(1 - D(z)) \sum_{i=1}^K C_i(x)q_i(z) dz \right] p(x) dx \]

\[ = \int_x \log D(x)p(x)dx + \int_x \int_z \left[ \log(1 - D(z)) \sum_{i=1}^K C_i(x)q_i(z)p(x) \right] dz dx \]

\[ = \int_x \log D(x)p(x)dx + \sum_{i=1}^K \int_x \int_z \left[ \log(1 - D(z))C_i(x)q_i(z)p(x) \right] dz dx \]

\[ = \int_x \log D(x)p(x)dx + \sum_{i=1}^K \log(1 - D(x))q_i(x)dx \]

\[ = \int_x \left[ \log D(x)p(x) + \sum_{i=1}^K T_i \log(1 - D(x))q_i(x) \right] dx \]

where we denote \( T_i = \int_x C_i(x)p(x)dx \leq 1 \) by assuming \( p(x) \) is piecewise-integrable.

To maximize variational problem of \( \mathcal{L}_D(D, C_{1:K}, G_{1:K}) \) with respect to \( D \), we calculate the partial derivative of the term within the integral:

\[ \frac{\partial}{\partial D} \left[ \log D(x)p(x) + \sum_{i=1}^K T_i \log(1 - D(x))q_i(x) \right] = \frac{p(x)}{D(x)} + \sum_{i=1}^K T_i \frac{q_i(x)}{1 - D(x)} \]

Letting (3.23) equals to 0, we have

\[ D^*(x) = \frac{p(x)}{p(x) + \sum_{i=1}^K T_i q_i(x)} \]
which is optimal for $\mathcal{L}_D(D,C_{1:K},G_{1:K})$.

**Optimal $C^*_k$:**

For an arbitrary $k$ ($1 \leq k \leq K$), we study the optimization problem $\mathcal{L}_C(D,C_{1:K},G_{1:K})$ in terms of $C_k$ as below:

$$
\mathcal{L}_C(D,C_{1:K},G_{1:K}) = \mathbb{E}_{x \sim p} \left[ \mathbb{E}_{x \sim p} \log(1 - D(x)) \right]
- \lambda \left\{ \mathbb{E}_{x \sim q_k} \left[ \log \left( C_k(x) q_k(x) \right) \right] + \sum_{j \neq k} \mathbb{E}_{x \sim q_j} \left[ \log \left( (1 - C_k(x)) q_j(x) \right) \right] \right\}
$$

Suppose the optimal $C_k$ satisfies $\int x C_k(x)p(x)dx \to T_k$ after finite rounds, then the optimum of (3.25) is determined by the last term. Calculating the derivative in the integral and letting it equal to 0, we have

$$
\frac{q_k(x)}{C_k(x)} \frac{\sum_{j \neq k} q_j(x)}{1 - C_k(x)} = 0, \quad \Leftrightarrow \quad C^*_k(x) = \frac{q_k(x)}{q_k(x) + \sum_{j \neq k} q_j(x)}
$$

**Optimal $G^*_k$:**
Since we substitute $C_k(x)$ by $\frac{r_k}{R_k} C_k(x)$ before updating $G_k$, we have

$$
\mathbb{E}_{x \sim p} \left[ \log D(x) + \mathbb{E}_{z \sim \sum_i C_i(x) \frac{1}{R_i} q_i} \log(1 - D(z)) \right]
= \int_x \log D(x) p(x) dx + \int_x \int_z \left[ \log(1 - D(z)) \sum_{i=1}^K \frac{r_i}{R_i} C_i(x) q_i(z) dz p(x) \right] dx
$$

$$
= \int_x \log D(x) p(x) dx + \sum_{i=1}^K \int_x \frac{r_i}{R_i} C_i(x) p(x) dx \int_z \log(1 - D(z)) q_i(z) dz
$$

$$
= \int_x \log D(x) p(x) dx + \sum_{i=1}^K T'_i r_i \int_z \log(1 - D(z)) q_i(z) dz
$$

$$
= \int_x \left[ \log D(x) p(x) + \sum_{i=1}^K T'_i \log(1 - D(x)) q_i(x) \right] dx
$$

$$
= \int_x \left[ \log D(x) p(x) + \sum_{i=1}^K T'_i \log(1 - D(x)) q_i(x) \right] dx
$$

(3.27)

where $T'_i = \frac{T_i r_i}{R_i}$. Plugging $D^*$ and $C^*_i$ into our optimization problem, we have

$$
\mathcal{L}_G(D, C_{1:K}, G_{1:K})
= \mathbb{E}_{x \sim p} \left[ \log \frac{p(x)}{\sum_{i=1}^K T'_i q_i(x)} \right] + \mathbb{E}_{x_{1:K} \sim q_{1:K}} \left[ \log \frac{\sum_{i=1}^K T'_i q_i(x)}{p(x) + \sum_{i=1}^K T'_i q_i(x)} \right]
$$

$$
- \lambda \sum_{k=1}^K \left\{ \mathbb{E}_{x \sim q_k} \left[ \log \frac{q_k(x)}{q_k(x) + \sum_{j \neq k} q_j(x)} \right] + \sum_{j \neq k} \mathbb{E}_{x \sim q_j} \left[ \log \frac{\sum_{j \neq k} q_j(x)}{q_k(x) + \sum_{j \neq k} q_j(x)} \right] \right\}
$$

(3.28)
One can rearrange the last term of (3.28) as follows:

\[
\sum_{j \neq k} \mathbb{E}_{x \sim q_j} \left[ \log \frac{\sum_{j \neq k} q_j(x)}{q_k(x) + \sum_{j \neq k} q_j(x)} \right]
\]

\[
= \sum_{j \neq k} \int_x \left( \log \frac{\sum_{j \neq k} q_j(x)}{q_k(x) + \sum_{j \neq k} q_j(x)} \right) q_j(x) dx
\]

\[
= \int_x \left( \log \frac{\sum_{j \neq k} q_j(x)}{q_k(x) + \sum_{j \neq k} q_j(x)} \right) \sum_{j \neq k} q_j(x) dx
\]

\[
= \mathbb{E}_{x \sim \sum_{j \neq k} q_j} \left[ \log \frac{\sum_{j \neq k} q_j(x)}{q_k(x) + \sum_{j \neq k} q_j(x)} \right].
\]

Plugging (3.29) into (3.28), we have

\[
\mathcal{L}_G(D, C_{1:K}, G_{1:K})
= \mathbb{E}_{x \sim p} \left[ \log \frac{p(x)}{p(x) + \sum_{i=1}^K T'_i q_i(x)} \right] + \mathbb{E}_{x \sim q_1, K \sim q_{1:K}} \left[ \log \frac{\sum_{i=1}^K T'_i q_i(x)}{p(x) + \sum_{i=1}^K T'_i q_i(x)} \right]
- \lambda \sum_{k=1}^K \left\{ \mathbb{E}_{x \sim q_k} \left[ \log \frac{q_k(x)}{q_k(x) + \sum_{j \neq k} q_j(x)} \right] + \mathbb{E}_{x \sim \sum_{j \neq k} q_j} \left[ \log \frac{\sum_{j \neq k} q_j(x)}{q_k(x) + \sum_{j \neq k} q_j(x)} \right] \right\}
= - \log 4 + JSD(p || \sum_{k=1}^K T'_k q_k) - \lambda \sum_{k=1}^K (-\log 4 + JSD(q_k || \sum_{j \neq k} q_j)).
\]

(3.30)

The first Jensen-Shannon Divergence between \( p \) and \( \sum_k T'_k q_k \) is minimized when \( q = \sum_k T'_k q_k \). Further it can be deduced that \( T_k = T'_k \) and \( q_k = p_k \) iff \( r_k = R_k \). The second term is minimized when \( JSD(q_k || \sum_{j \neq k} q_j) = 1 \), \( \forall k \in [K] \), so one can obtain \( \{x | q'_k(x) > 0\} \bigcap \{x | \sum_{j \neq k} q'_j(x) > 0\} = \Phi, \forall k \in [K] \) where \( \Phi \) denotes the empty set. Moreover, one can evidently attain that \( \{x | q'_k(x) > 0\} \bigcap \{x | q'_j(x) > 0\} = \Phi \) holds for \( \forall j \neq k \) and \( \forall k \in [K] \), due to the non-negativeness of \( q_j(x) \).

Next we show the uniqueness of the decomposition on \( p \) with some mild assumptions. We conveniently denote \( D_k = \{x | q'_k(x) > 0\} \) and its closure as \( \bar{D}_k \).
Given \( q_j^*(x) \) and \( q_k^*(x) \) are separated \( \forall 1 \leq j \neq k \leq K \), we have \( \bar{D}_j \cap \bar{D}_k = \Phi \).

From (3.26) we have \( C_i^*(x) = 1 \) if \( x \in D_i \), otherwise \( C_i^*(x) = 0 \). Therefore, acknowledging that \( q_i^* \) is globally continuous, \( p(x) = q^*(x) = \sum_{i=1}^{K} C_i^*(x)q_i^*(x) \) is piecewise-continuous on the domain \( D_1 \cup D_2 \cup \cdots \cup D_K \). Observing the two facts: (1) \( q_i^*(x) \) is continuous, (2) the desired piecewise-continuous decomposition is \( p(x) = \sum_{i=1}^{K} w_i(x)p_i(x) \) where \( w_i(x) = 1 \) if \( x \in \text{Cluster } i \) or 0 otherwise, one can obtain the conclusion that the decomposition of \( p(x) = \sum_{i=1}^{K} C_i^*(x)q_i^*(x) \) is unique on the domain \( D_1 \cup D_2 \cup \cdots \cup D_K \) with the continuous function set \( \{q_i^*\}_{i=1}^{K} \). By finding a proper order of index, one can obtain \( q_i^*(x) = p_i(x) \) and \( C_i^*(x) = w_i(x) \).

By controlling the smoothness of \( G_i \)'s (SKC), one may argue that the GAN may lose the universal approximation property to any \( p \). We further discuss the approximation ability of our method to a pdf function by examining the smoothness of the target \( p \) and approximation \( q \). The following theorem shows the existence of a smooth \( q_i \) in the form of convolution as Eq. (3.11), which can adequately approximate any continuous \( p_i \).

**Theorem 6:** For all \( i \in [K] \), suppose the \( i \)-th cluster has Lipschitz continuous pdf \( p_i \) with Lipschitz constant \( L_{p_i} \). There exists the approximant \( q_i(\cdot, \sigma) \) that can be expressed as the convolution of \( \bar{q}_i \) and \( g(\cdot, \sigma) \) as shown in Eq. (12). Moreover, \( q_i(\cdot, \sigma) \) has the following properties:

1. For all \( \sigma > 0 \), it holds that \( q_i \) is also Lipschitz continuous, with the
Lipschitz constant of $L_{q_i} \leq L_{p_i}$.

2. For all $\varepsilon' > 0$ and $L_{p_i} \sigma < \frac{\varepsilon'}{\alpha_i}$ where $\alpha_i$ denotes the minimum radius of the $d$-dimensional ball containing the set $D_i$, it holds that $\|q_i(\cdot, \sigma) - p_i\|_\infty < \varepsilon'$.

Note that the infinity norm $\|f\|_\infty$ of a function $f$ is defined as $\|f\|_{\infty, S} = \sup\{|f(x)| : x \in S\}$. Theorem 6 confirms that an adequate approximator $q_i$ can be found if $p_i$ is Lipschitz continuous. One can also infer that by choosing a $q_i$ which is smoother than $p_i$, the approximator can approach to the true distribution, which is required in the definition of SKC. With the justification of the adequate approximator, we give the following theorem to manifest that the $\epsilon$-approximate equilibrium defined in [80] can be achieved by our model, instead of an Nash equilibrium reached by standard GANs.

Proof.

For any $z_1, z_2 \in D_i$, we denote the function $q_i$ as $q_i(\cdot, \sigma) = p_i \ast g(\cdot, \sigma)$. Hence, we have

$$
q_i(z, \sigma) = \int_{D_i} p_i(z) g(z - y, \sigma) dy = \int_{D_i} p_i(z) \frac{1}{(\sqrt{2\pi} \sigma)^d} e^{-\frac{\|z - y\|^2}{2\sigma^2}} dy
$$

$$
= \int_{D_i} p_i(z - y) g(y, \sigma) dy = \int_{D_i} p_i(z - y) \frac{1}{(\sqrt{2\pi} \sigma)^d} e^{-\frac{\|y\|^2}{2\sigma^2}} dy
$$

$$
= \int_{D_i} p_i(z - \sigma y') g(y', 1) dy' = \int_{D_i} p_i(z - \sigma y) g(y, 1) dy
$$

where the last line is obtained by substituting $y' = y/\sigma$. 

(3.31)
Clearly $q_i$ is differentiable. Given any $z_1, z_2 \in D_i$ while $z_1 \neq z_2$, we have

$$|q_i(z_1, \sigma) - q_i(z_2, \sigma)| = \left| \int_{D_i} p_i(z_1 - \sigma y) g(y, 1) dy - \int_{D_i} p_i(z_2 - \sigma y) g(y, 1) dy \right|$$

$$= \left| \int_{D_i} [p_i(z_1 - \sigma y) - p_i(z_2 - \sigma y)] g(y, 1) dy \right|$$

$$\leq L_{p_i} \|z_1 - z_2\|_2 \left| \int_{D_i} g(y, 1) dy \right|$$

$$\leq L_{p_i} \|z_1 - z_2\|_2 \left| \int_{\mathbb{R}^d} g(y, 1) dy \right|$$

$$\leq L_{p_i} \|z_1 - z_2\|_2$$

(3.32)

Therefore, one can conclude from (3.32) that $q_i$ is Lipschitz continuous, with its Lipschitz constant as $L_{q_i} \leq L_{p_i}$.

Further, given any $z \in D_i$, it is evident to obtain

$$|q_i(z, \sigma) - p_i(z)| = \left| \int_{D_i} p_i(z - \sigma y) g(y, 1) dy - \int_{D_i} p_i(z) g(y, 1) dy \right|$$

$$= \left| \int_{D_i} [p_i(z - \sigma y) - p_i(z)] g(y, 1) dy \right|$$

$$\leq L_{p_i} \|z - \sigma y - z\|_2 \left| \int_{D_i} g(y, 1) dy \right|$$

$$\leq \frac{\varepsilon'}{\sigma \alpha_i} \|\sigma y\|_2 \left| \int_{\mathbb{R}^d} g(y, 1) dy \right|$$

$$\leq \varepsilon'$$

(3.33)

where the forth line holds due to $L_{p_i} < \frac{\varepsilon'}{\sigma \alpha_i}$. Thus, it is evident that $\|q_i(\cdot, \sigma) - p_i\|_\infty = \sup_z |q_i(z, \sigma) - p_i(z)| < \varepsilon'$.

\[ \Box \]

**Theorem 7:** Define $\beta^{-1} = \sup_{Z \in D_i} \int_{Z} \frac{q_i(z)}{\sigma_i} dz$ and assume $\beta \leq 1$. Suppose
$C_{1:K}$ and $D$ have enough capacity and the pdfs of $K$ true clusters are $p_{1:K}$ with the number of modes respectively as $r_{1:K}$. For any $\varepsilon > 0$, $G_{1:K}$ induce $q_{1:K}(\cdot, \sigma)$ in the setting of $\sigma < \frac{(1-\beta)\varepsilon}{\gamma} \frac{\pi d/2}{(\pi d/2+1) \alpha^d \log \frac{1}{\beta} \min_i \alpha_i}$ defined in Theorem 2, there exists an $\epsilon$-approximate equilibrium for our problem that $D = D^*$, $C_{1:K} = C_{1:K}^*$ identical as in Theorem 5, while it holds that $\|q_{i}^*(\cdot, \sigma) - p_i^*\|_\infty < \varepsilon$ and $\|q^*(\cdot, \sigma) - p^*\|_\infty < \varepsilon$ where $p_i$ is the true pdf of the $i$-th cluster.

Theorem 7 guarantees the approximation ability of KCN on both the cluster distribution and the overall distribution. This is of great importance for a better understanding of the main advantage in our method. In summary, if $p$ is smooth, KCN can assure the adequate approximation of the data distribution that generates clusters by each generator.

**Proof.** From the Reverse Pinsker Inequality (see Theorem 7 in [88]), denoting

$$\beta^{-1} = \sup_{Z \in D_i} \int_Z \frac{q_i(z)}{p_i + q_i(z)} dz,$$

with $\beta \leq 1$, we have

$$\frac{1 - \beta}{\log \frac{1}{\beta}} KL(q_i || \frac{p_i + q_i}{2}) \leq \frac{1}{2} \|q_i - \frac{p_i + q_i}{2}\|_{TV}$$

where $\|q_i - \frac{p_i + q_i}{2}\|_{TV} = \frac{1}{2} \|q_i - p_i\|_{TV} = \sup_{Z \in D_i} |\int_Z q_i(z) dz - \int_Z p_i(z) dz|$ designates the total variation distance between $q_i$ and $\frac{p_i + q_i}{2}$.

To reach the supremum, one can specify $Z = \{z : q_i(z) \geq p_i(z)\}$ and
When $Z = \{z : q_i(z) \leq p_i(z)\}$, using the result of Theorem 2, one can obtain

$$\frac{1}{2} \|q_i - p_i\|_{TV} = \sup_{z \in D_i} \left| \int_Z q_i(z) dz - \int_Z p_i(z) dz \right|$$

$$= \frac{1}{2} \left[ \int_Z q_i(z) dz - \int_Z p_i(z) dz + \left( 1 - \int_Z p_i(z) dz \right) - \left( 1 - \int_Z q_i(z) dz \right) \right]$$

$$= \frac{1}{2} \left[ \int_Z (q_i(z) - p_i(z)) dz + \int_{Z^c} (p_i(z) - q_i(z)) dz \right]$$

$$= \frac{1}{2} \int_{D_i} |q_i(z) - p_i(z)| dz$$

where $\Gamma(\cdot)$ denotes the Gamma function. From Eq. (3.35) and Eq. (3.37), we have

$$\frac{1 - \beta}{\log \beta} KL(q_i || \frac{p_i + q_i}{2}) \leq \frac{\pi^{d/2}}{\Gamma(d/2 + 1)} \alpha_i^d \epsilon', \quad \leftrightarrow \quad KL(q_i || \frac{p_i + q_i}{2}) \leq \frac{\pi^{d/2}}{\Gamma(d/2 + 1)} \alpha_i^d \log \frac{1}{\beta} \epsilon'.$$

In analogy with (3.38), one can easily attain

$$KL(p_i || \frac{p_i + q_i}{2}) \leq \frac{\pi^{d/2}}{\Gamma(d/2 + 1)} \alpha_i^d \log \frac{1}{\beta} \epsilon'.$$  

Hence, Eq. (3.38) and Eq. (3.39) yield an upper-bound of JSD between $p_i$
and $q_i$:

$$JSD(p_i, q_i) = KL(p_i||\frac{p_i + q_i}{2}) + KL(q_i||\frac{p_i + q_i}{2}) \leq 2 \frac{\pi^{d/2}}{\Gamma(d/2+1)} \alpha i \log_\beta \frac{1}{\beta} \leq \varepsilon < \varepsilon$$

(3.40)

where we let $\varepsilon' = \frac{(1-\beta)\varepsilon}{2^{\pi^{d/2}} \min, \alpha i K \log_\beta \frac{1}{\beta}}$. Considering the JSD between the global pdf $s p$ and $q$, we have

$$JSD(p, q) = KL(p||\frac{p + q}{2}) + KL(q||\frac{p + q}{2})$$

$$= KL(\sum_{i=1}^{K} C_i^* p_i || \frac{\sum_{i=1}^{K} [C_i^* p_i + C_i^* q_i]}{2}) + KL(\sum_{i=1}^{K} C_i^* q_i || \frac{\sum_{i=1}^{K} [C_i^* p_i + C_i^* q_i]}{2})$$

$$\leq \frac{1}{2} \log_\beta \frac{1}{2} \sum_{i=1}^{K} [C_i^* (p_i - q_i)] || TV + \frac{1}{2} \log_\beta \frac{1}{2} \sum_{i=1}^{K} [C_i^* (q_i - p_i)] || TV$$

$$\leq \frac{1}{2} \log_\beta \frac{1}{2} \sum_{i=1}^{K} || p_i - q_i || TV + \frac{1}{2} \log_\beta \frac{1}{2} \sum_{i=1}^{K} || q_i - p_i || TV$$

$$\leq \log_\beta \frac{1}{2} \sum_{i=1}^{K} || p_i - q_i || TV \leq 2K \frac{\pi^{d/2}}{\Gamma(d/2+1)} \alpha i \log_\beta \frac{1}{\beta} \leq \varepsilon' < \varepsilon$$

(3.41)

From Definition 3 in [80], we have the definition of $\varepsilon$-approximate pure equilibrium:

**($\varepsilon$-approximate pure equilibrium)** A pair of pure strategies $(u, v) \in (V, U)$ is an $\varepsilon$-approximate pure equilibrium, if for some value $V$

$$\forall v \in V, F(u, v) \leq V + \varepsilon,$$

$$\forall u \in U, F(u, v) \geq V - \varepsilon,$$

where $F(u, v) = \mathbb{E}_{x \sim p} [\log(D_v(x))] + \mathbb{E}_{x \sim q_u} [\log(1 - D_v(x))]$ and $q_u$ is induced by $G_u$ parameterized by $u$. 

With the definition and (3.30), we can conclude that when $C_{1:K} = C^*_{1:K}$ and the discriminator $D$ outputs a constant $1/2$ as $D^*$, it holds that $F(u, v) = V \leq V + \varepsilon$ for all $u \in U$, by letting $V = -\log 4 + \lambda C K \log 4$. On the other hand, $K$ generators satisfy that $F(u, v) = V \geq V - \varepsilon$ since we have

$$F(u, v) = -\log 4 + JSD(p||q) + \lambda C K (\log 4 - \sum_{i=1}^{K} JSD(p_i||q_i)) \leq -\log 4 + \lambda C K \log 4 + \varepsilon \leq V + \varepsilon \tag{3.43}$$

where $q$ is induced by the $K$ generators together with $C_{1:K}$.

Therefore, the $\varepsilon$-approximate pure equilibrium can be achieved.

### 3.5 Experiments

The proposed KCN was validated in a simulation and using real-world datasets. The synthetic data were designed to verify KCN’s capability in comparison with existing multi-GAN models, e.g., MGGAN. We used eight generators respectively in KCN and MGGAN. Samples generated by the eight generators were plotted by different colors in Fig. 3.1 (a) (true clusters). For MGGAN in (b), we observed that all the modes of the true distribution were identified. However, some generators captured multiple mixtures whereas some others captured no mixture at all. On the other hand, the eight generators of KCN captured all modes (Fig. 3.1 (c)), each of which was captured individually by only one generator. This simple synthetic experiment confirmed that our method could better regulate the balance between clusters by controlling the SKC of GANs than MGGAN.
We evaluated KCN on three real-world datasets: MNIST, COIL20, and CIFAR10, and compared it against five other methods, e.g., one-layer generative clustering and DNN-based clustering methods: GMM, SAE+k-means[89], AEVB[90], DAE+k-means[91], and DEC [67]. Table 3.1 summarized the ACC values [67] of all methods. A higher ACC value indicates better performance. The KCN outperformed all other methods and significantly surpassed the classic model-based clustering method GMM. It suggests that KCN, as a generative model based clustering method, is more suitable to cluster samples with complex distributions because of its stronger expressiveness of the hypothesis space \( \mathcal{Q} \) than the classic methods.

**Image Generation.** We present the images generated by KCN trained on the three real-world datasets for qualitative assessment of the approximation quality. We aligned samples generated from a single generator on the same row

<table>
<thead>
<tr>
<th>Methods</th>
<th>Dataset</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>MNIST</td>
</tr>
<tr>
<td>GMM</td>
<td>0.5715</td>
</tr>
<tr>
<td>SAE+k-means[89]</td>
<td>0.8271</td>
</tr>
<tr>
<td>AEVB[90]</td>
<td>0.8251</td>
</tr>
<tr>
<td>DAE+k-means[91]</td>
<td>0.6702</td>
</tr>
<tr>
<td>DEC [67]</td>
<td>0.8391</td>
</tr>
<tr>
<td><strong>KCN</strong></td>
<td><strong>0.8555</strong></td>
</tr>
</tbody>
</table>

**Table 3.1:** The Comparison of ACC values of different methods on real-world datasets.
in Fig. 3.1. In (a), the generated digits from the MNIST dataset on each row appeared indistinguishable from the training digit images. In (b), the left half columns were real images from CIFAR10 and the right half contained the generated images from KCN. It shows that the generated images were highly similar to the true images in the same cluster. We collected results on COIL20 in (c) where each column contained images from the same generator with the last row presenting true examples in the respective cluster. It is interesting to see that images show consistency within its cluster; toy ducks on the 1st row, bottles on the 2nd row and cats on the 11th row.

**Smoothness versus Complexity.** We investigated how the smoothness parameter $\sigma$ influences the resultant SKC of a generator and the generated images on MNIST. We trained six KCN models, each with a fixed $\sigma$ selected in $[0.06, 0.20]$ with a step size of 0.02. In Fig. 3.3 as our expectation, when the smoothness $\sigma$ increased, the complexity of all generators $R(G_{1:K})$ deceased homogeneously.
Fig. 3.2: Generated results on MNIST(a), SIFA10(b) and COIL20(c).

Fig. 3.3: SKC vs $\sigma$ of 10 generators on MNIST dataset.
3.6 Summary

We have proposed a novel generative model - Kolmogorov Coupled Nets - that discretizes a target distribution into coupling of multiple generator networks. Guided by approximation theory, orthogonality and complexity constraints are imposed on the component distributions (in $Q$) induced by generators. To enforce orthogonality, we maximize the distance between each pair of clusters’ distributions. To regularize the complexity of a generator, we introduce a new GP-smooth layer on top of the original GAN, through which we can estimate and control the Surrogate Kolmogorov complexity. The resultant algorithm can guarantee a recovery of the true clusters under regularity conditions and is shown superior in empirical evaluation.
Chapter 4

Conclusion

In this dissertation, we present our studies toward building provable and scalable machine learning methods, where we construct learning models by investigating the specific structure of data inputs. In the first direction of the study, we propose a new method using side information to recover the partially observed matrix. Recent research often formulated the problem as low-rank matrix completion, which recovers the matrix by assuming linear dependency between the users ratings to estimate missing entries in a matrix. Besides this structural assumption on data, we propose a model which builds upon the exploration on the rich side information that increase recovery accuracy. Specifically, we develop a novel sparse bilinear method that explicitly models the interaction between the row and column side features to approximate the matrix entries. In addition, we prove that when the side features are fully coherent with the matrix to be recovered, our method can exactly recover the matrix with orders of magnitude improvements on the necessary number of observed entries (i.e. sampling rate) compared to state-of-the-art methods. If the side features are corrupted, i.e., less coherent with the matrix to
be recovered, our method can still approximately recover the matrix with the same orders of magnitude on the sampling rates. We propose an efficient linearized alternating direction multiplier algorithm (ADMM) to solve the optimization problem with convergence guarantees. The algorithm results in a better performance than existing works and also provides a solution to the cold-start problem where few entries of the matrix to be recovered is observed. Along the second direction, we investigate the new approaches in unsupervised learning, where our goal is to build a generative model to approximate the data distribution. Assuming that the data distribution contains a mixture of $r$ components, we propose a new method where the distribution of each component is generalized to include density functions induced by a generative adversarial network (GAN), but the capacity of each GAN is effectively controlled by the so-called Surrogated Kolmogorov Complexity from approximation theory. The key of Kolmogorov Coupled Nets is its capability of approximating both individual clusters distributions and the overall data distribution with a controllable tolerance, showing that the new way offers better expressive power.
Bibliography


[44] Ethan X Fang, Bingsheng He, Han Liu, and Xiaoming Yuan. Generalized
alternating direction method of multipliers: new theoretical insights and ap-

[45] Jin Lu, Jiangwen Sun, Xinyu Wang, Henry Kranzler, Joel Gelernter, and
Jinbo Bi. Inferring phenotypes from substance use via collaborative matrix

[46] Jin Lu, Jiangwen Sun, Xinyu Wang, Henry R Kranzler, Joel Gelernter, and
Jinbo Bi. Collaborative phenotype inference from comorbid substance use
disorders and genotypes. In *2017 IEEE International Conference on Bioin-


[48] Emmanuel J Candès, Xiaodong Li, Yi Ma, and John Wright. Robust principal

[49] Roman Vershynin. Introduction to the non-asymptotic analysis of random

[50] Peter L Bartlett and Shahar Mendelson. Rademacher and gaussian complex-

[51] Sham M Kakade, Karthik Sridharan, and Ambuj Tewari. On the complexity
of linear prediction: Risk bounds, margin bounds, and regularization. In

[52] Ron Meir and Tong Zhang. Generalization error bounds for bayesian mixture

[53] Nathan Srebro and Adi Shraibman. Rank, trace-norm and max-norm. pages

[54] Ohad Shamir and Shai Shalev-Shwartz. Matrix completion with the trace
norm: learning, bounding, and transducing. *The Journal of Machine Learn-

[55] Junfeng Yang and Xiao-Ming Yuan. Linearized augmented lagrangian and al-
[56] Jian-Feng Cai, Emmanuel J. Candès, and Zuowei Shen. A singular value
thresholding algorithm for matrix completion. SIAM J. on Optimization, 20

[57] W. Shi, Q. Ling, G. Wu, and W. Yin. A proximal gradient algorithm for de-
centralized composite optimization. IEEE Transactions on Signal Processing,

2015.

[59] Anneleen Daemen, Obi L Griffith, Laura M Heiser, Nicholas J Wang, Oana M
Enache, Zachary Sanborn, Francois Pepin, Steffen Durinck, James E Korko-

[60] Joel Gelernter, Richard Sherva, Ryan Koesterer, Laura Almasy, Hongyu
Zhao, Henry R Kranzler, and Lindsay Farrer. Genome-wide association study

[61] Joel Gelernter, Henry R Kranzler, Richard Sherva, Ryan Koesterer, Lau-
ra Almasy, Hongyu Zhao, and Lindsay A Farrer. Genome-wide association
study of opioid dependence: multiple associations mapped to calcium and

latent dirichlet allocation. In advances in neural information processing sys-

[63] David M Blei, Andrew Y Ng, and Michael I Jordan. Latent dirichlet alloca-

[64] Liangliang Cao and Li Fei-Fei. Spatially coherent latent topic model for
concurrent segmentation and classification of objects and scenes. In Computer

[65] Geoffrey E Hinton and Ruslan R Salakhutdinov. Reducing the dimensionality

[66] Fei Tian, Bin Gao, Qing Cui, Enhong Chen, and Tie-Yan Liu. Learning deep
representations for graph clustering. 2014.


