Optimism and Robustness:

Learning From Structured and Semi-Random Inputs

by

Xing Gao
B.S., Georgia Institute of Technology, 2013
M.S., Georgia Institute of Technology, 2014
M.S., Georgia Institute of Technology, 2020

THESIS

Submitted in partial fulfillment of the requirements
for the degree of Doctor of Philosophy in Mathematics
in the Graduate College of the
University of Illinois Chicago, 2025

Chicago, Illinois

Defense Committee:
Prof. Lev Reyzin, Chair and Advisor
Prof. Yu Cheng, Advisor, Brown University
Prof. Vishesh Jain
Prof. Xiaorui Sun, Department of Computer Science
Prof. György Turán

献给我的爷爷

高清元

(1934–2002)

ACKNOWLEDGMENT

I feel extremely blessed during my PhD studies. First and foremost, I would like to express my deepest gratitude to my advisors, Prof. Lev Reyzin and Prof. Yu Cheng. Their advising styles are "orthogonal" in the most beneficial way, each bringing unique perspectives and strengths. Together, their guidance has not only shaped this dissertation but also deeply influenced my growth as a researcher. I thank Lev for his passion for research, optimism in the face of setbacks, and understanding of my special needs, as well as for always encouraging me to pursue what I love and to stay true to myself. Beyond academics, he has inspired me with his integrity and unwavering principles, which have left a lasting impact on me. I thank Yu for his emphasis on strict mathematical rigor, and for challenging me with the advice that I should do things that intimidate me at least once. His high standards have pushed me to refine my work and strive for excellence. I am also deeply grateful to my committee members, Prof. Vishesh Jain, Prof. Xiaorui Sun, and Prof. György Turán, for their guidance and support throughout the process of completing this thesis.

I thank Idan Attias for being a wonderful research partner and for the insights I gained from the way he approaches problems. Working with him has been both inspiring and rewarding. I thank my dear friend Siyu Li, for the Mozart, Stravinsky and Tchaikovsky we enjoyed together, and for sharing the Love and Serenity Labubu. I am lucky to have Shuo Wu in my life. It remains a mystery how two "insulated" personalities could become so

inseparable. We will continue our game of "explore and exploit" and keep arguing about every topic we ever talk about.

Finally, I would like to express my deepest love to my family. I thank my parents, Gui Gao and Qiuxiang Zhou, for their unconditional love and support. To my husband Jin Xu: our double PhD journey has been (challenging)^2. We thought we had lost each other, but we found our way back, and forward. I love you, too.

XG

CONTRIBUTIONS OF AUTHORS

Chapter 2 represents the manuscript *Learning-Augmented Algorithms for Boolean Satisfiability* by Idan Attias, Xing Gao and Lev Reyzin. The work is currently under submission and is available on arXiv (Attias et al., 2025). Idan Attias contributed to most of the writing in the introduction. The remaining of the manuscript, including literature review, formulations of definitions, theorems, algorithms, and writing was done jointly with the coauthors.

Chapter 3 represents the paper *Non-Center-Based Clustering Under Bilu-Linial Stability* (Gao and Reyzin, 2025) by Xing Gao and Lev Reyzin. All content including the introduction, literature review, formulations of definitions, theorems, algorithms, and writing of the manuscript was done jointly with the coauthor.

Chapter 4 represents the manuscript *Semi-Random Noisy and One-Bit Matrix Completion With Nonconvex Primal-Dual Framework* by Xing Gao, Binhao Chen and Yu Cheng. The manuscript is currently under submission. All content including the introduction, literature review, formulations of definitions, theorems, and writing of the manuscript was done jointly with the coauthors.

Chapter 5 represents the paper *Robust Matrix Sensing in the Semi-Random Model* (Gao and Cheng, 2024) by Xing Gao and Yu Cheng. All content including the introduction, literature review, formulations of definitions, theorems, algorithms, and writing of the manuscript was done jointly with the coauthor.

TABLE OF CONTENTS

CHAPTER **PAGE**

CHAPTER **PAGE**

TABLE OF CONTENTS (Continued)

CHAPTER **PAGE**

LIST OF TABLES

LIST OF FIGURES

LIST OF ABBREVIATIONS

CNF	conjunctive normal form
CSP	constraint satisfiability problem
dRIP	decomposable wRIP
ETH	Exponential Time Hypothesis
LIN	linear satisfiability problem
MSD	minimizing sum-of-diameters
MSR	minimizing sum-of-radii
PTAS	polynomial-time approximation scheme
RIP	Restricted Isometry Property
RSC	Restricted Strong Convexity
RSS	Restricted Strong Smoothness
SAT	Boolean satisfiability problem
SDP	semidefinite programming
SETH	Strong Exponential Time Hypothesis
UGC	Unique Games Conjecture
wRIP	weighted RIP

SUMMARY

Traditionally, algorithms have been studied under two regimes: worst-case analysis, which makes no assumptions about the input, and average-case analysis, which assumes that inputs are drawn from a certain distribution. However, real-world inputs rarely conform to either of these extremes. A new paradigm known as "beyond worst-case analysis" seeks to bridge the gap between the two. In this thesis, we study several problems and their algorithms under appropriate beyond worst-case models, aiming to provide more realistic and practically relevant performance guarantees.

In the first part of this thesis, we focus on improving algorithm performance on non-worst-case inputs (structured inputs). In Chapter 2, we study the Boolean satisfiability problem (SAT) in the framework of learning-augmented algorithms, where the problem instance is provided with a prediction that contains partial information of an optimal solution. We study both the decision and optimization problem of SAT under two forms of predictions, namely the subset advice and the label advice. In Chapter 3, we study non-center-based clustering under Bilu-Linial stability assumptions, which assumes that the problem instance has a unique optimal solution that stays unchanged under small perturbation of the input. We focus on the minimizing sum-of-radii (MSR) and minimizing sum-of-diameters (MSD) objectives, and provide polynomial time solutions under stability assumptions.

In the second part of this thesis, we focus on enhancing algorithm robustness to contamination in average-case inputs (semi-random inputs). In the context of low-rank matrix recovery problems, this means a monotone adversary can add arbitrary data from the distribution to break the necessary regularity conditions satisfied by fully random inputs. In Chapter 4, we study the matrix completion problem, whose goal is to recover a ground-truth matrix from incomplete and noisy observations of its entries. In Chapter 5, we study the matrix sensing problem, where the goal is to recover the ground-truth matrix based on linear measurements from a given set of sensing matrices.

CHAPTER 1

INTRODUCTION

1.1 Introduction

The design and analysis of algorithms play a crucial role in theoretical computer science. Traditionally, we analyze algorithms in two contrasting regimes: the worst-case and average-case analysis. In the former, we make no assumptions about the input when designing the algorithm, and evaluate it against the worst-case instances. The advantage of this approach is that we obtain guarantees on the algorithm for all inputs, including adversarial ones, making the algorithm universally applicable. However, since real-world inputs are usually "non-pathological" (Roughgarden, 2019), worst-case analysis is overly pessimistic and may not accurately reflect an algorithm's performance in practice. In the latter approach, we assume inputs are drawn from a fixed distribution, and analyze the algorithm's average-case performance with respect to the distribution. By leveraging domain-specific information on inputs, many algorithms have demonstrated both empirical and theoretical success under distributional assumptions. Yet this approach is susceptible to overfitting to a particular distribution and may rely on assumptions that are too strong or unrealistic. As a result, it is vulnerable to model misspecification and even data-poisoning attacks in practice.

A more recent paradigm known as "beyond worst-case analysis" seeks to bridge the gap between the two, with the goal of providing more realistic and practically relevant performance guarantees. In this thesis, we examine beyond worst-case analysis from two complementary directions.

- **Optimism.** The first half of the thesis adopts an optimistic perspective: we assume that the instances of interest are in some sense better than the worst case. Such optimism might arise because the instance exhibits structural properties intrinsic to the problem, or because we have access to some additional information such as historical data or predictions. From this perspective, we investigate how algorithms can exploit such **structured inputs** to achieve better performance guarantees. These ideas are developed in Chapter 2 and Chapter 3.
- **Robustness.** In contrast, the second half of the thesis emphasizes robustness. Here, we assume that inputs may be worse than the average case. In particular, we are interested in the semi-random model, where inputs drawn from a benign distribution are modified adversarially in an attempt to break algorithms that rely too heavily on distributional assumptions. Our goal in this setting is to design algorithms that remain effective under adversarial contamination, i.e., **semi-random inputs**. This theme is explored in Chapter 4 and Chapter 5.

The subsequent chapters are self-contained, each focusing on a specific problem and its associated algorithms under an appropriate beyond worst-case setting. The remainder of

this chapter provides the necessary background, introduces the problems studied in later chapters, and presents the various beyond worst-case settings considered in this thesis.

1.2 Background on Complexity Theory

Complexity Theory studies the time, space (memory), query and other resources required to solve computational problems. In this thesis, we are primarily concerned with the time complexity, which serves as a criterion for classifying problems into complexity classes. For decision problems (those with yes-or-no answers), the two most well-known classes are P and NP.

Definition 1.2.1 (The Class P (Sipser, 1996))**.** *P is the class of languages (problems) that are decidable in polynomial time on a deterministic single-tape Turing machine.*

Definition 1.2.2 (The Class NP (Sipser, 1996))**.** *NP is the class of languages (problems) that are verifiable in polynomial time given some certificate c. The hardest problems in the class NP are called NP-complete, which means any problem in NP is polynomial time reducible to a problem in NP-complete. The optimization version of an NP-complete decision problem is in the class NP-hard.*

Problems in P are generally considered to be "easy". A famous open question is whether P equals NP, and the prevailing belief is that P $\neq$ NP, i.e., NP-complete problems do not admit polynomial time algorithms. Despite this, NP-complete problems remain at the forefront of theoretical computer science, since many special instances can be solved efficiently. This observation motivates the study of beyond worst-case analysis, which aims

to understand and exploit the structural properties that make certain instances tractable. Identifying these structures and formalizing the assumptions that capture them is a crucial step in beyond worst-case analysis. In the following sections, we introduce the problems studied in this thesis and the respective beyond worst-case settings under which we analyze them.

1.3 Overview of Studied Problems

Here we introduce the specific problems studied in this thesis. All of them are NP-complete or NP-hard in the worst case.

1.3.1 Boolean Satisfiability

Given a Boolean formula on n variables in conjunctive normal form (CNF), the Satisfiability problem (SAT) asks whether there exists a truth assignment on the variables so that the formula evaluates to True. The problem is called k-SAT if each clause contains at most k literals, and the optimization version of the problem is called MAX-(k)-SAT, which asks for the maximum number of satisfiable clauses. We study the Boolean satisfiablility problem in Chapter 2, where we introduce formal definitions including other variants of the problem.

SAT is the first problem shown to be NP-complete by Stephen Cook and Leonid Levin (Cook, 1971; Levin, 1973). There is a stronger conjecture than P$\neq$NP called the Exponential Time Hypothesis (ETH), which postulates that 3-SAT cannot be solved in time $O(2^{cn})$ for some $c > 0$. Assuming P$\neq$NP, MAX-2-SAT cannot be approximated above 0.954 (Hastad, 2001). In light of these hardness results, research efforts have been towards

designing exponential time algorithms with the smallest c for the decision problem, as well as polynomial time algorithms with the best approximation ratio for the optimization problem.

For the decision problem of k-SAT, there has been a line of work using "random restriction algorithms". In particular for 3-SAT, the PPZ algorithm (Paturi et al., 1997) achieves running time of $O^*(2^{\frac{2}{3}n})$, where $O^*(\cdot)$ suppresses polynomial factors. This was later improved by the PPSZ algorithm (Paturi et al., 2005), which runs in time $O^*(2^{0.386n})$. For the optimization problem, a line of research has employed SDP relaxations combined with rounding techniques, achieving progressively improved approximation ratios. In particular for the MAX-2-SAT problem, (Goemans and Williamson, 1994) first showed an approximation ratio of 0.878 beyond the trivial approximation of 0.75 from the random assignment, and (Lewin et al., 2002) achieves state-of-the-art ratio of 0.94.

1.3.2 Non-Center-Based Clustering

In the clustering problem, we are given a set of points P and a metric $d : P \times P \to \mathbb{R}$, and the goal is to partition the points into k clusters while optimizing some objective function on the clusters under the metric. Center-based objectives such as k-center, k-medians and k-means have been studied extensively. In this thesis, we study clustering with non-center-based objectives, namely the minimizing sum-of-radii (MSR) and minimizing sum-of-diameters (MSD) objectives, and the formal definitions are provided in Chapter 3.

MSD and MSR are both known to be NP-hard under general metric. The NP-hardness of MSD is shown by reduction from the Clique problem (Doddi et al., 2000), and MSR is

proven by reduction from planar 3-SAT (Gibson et al., 2010). There are various approximation algorithms for these problems, and an α-approximation to one is a 2α-approximation to another. The work of (Charikar and Panigrahy, 2001) presents a greedy algorithm for MSR that achieves $O(\log \frac{n}{k})$-approximation with at most k clusters. Improving from log-factor to constant-factor approximation, (Charikar and Panigrahy, 2001) gives an LP based primal-dual algorithm that achieves 3.503-approximation for MSR, which is also a 7.006-approximation algorithm for MSD. Moving beyond approximation to exact algorithms, there is a randomized algorithm (Gibson et al., 2010) that gives the optimal solution to MSR with high probability in quasi-polynomial time. If k is considered to be a fixed constant, there are algorithms that take a "brute force" approach. For MSD, (Behsaz and Salavatipour, 2015) gives an algorithm by guessing the diameters of clusters in $n^{O(k^2)}$ time, and for MSR, (Doddi et al., 2000) gives an algorithm by transforming the problem to weighted set cover in $n^{O(k)}$ time and guessing the center-radius pairs.

1.3.3 Low-Rank Matrix Recovery

Low-rank matrix recovery is a popular inverse problem with many applications in machine learning, where the goal is to recover a ground-truth matrix $X^* \in \mathbb{R}^{d_1 \times d_2}$ from n measurements. In general it can be formulated as a constrained optimization problem, where f is a problem-specific objective function:

$$\min_{X \in \mathbb{R}^{d_1 \times d_2}} f(X) \quad \text{subject to } \operatorname{rank}(X) \leq r.$$

The problem has a nonconvex optimization landscape due to the low-rank constraint, making it NP-hard in the worst case. To obtain tractable solutions, analyses typically impose regularity conditions, under which both convex relaxations and nonconvex approaches have been shown to succeed.

The convex relaxation approach uses nuclear norm minimization as a proxy for low-rankness (Recht et al., 2010):

$$\min_{X \in \mathbb{R}^{d_1 \times d_2}} \|X\|_* \quad \text{subject to } f(X) = 0.$$

When formulated as a SDP, it can be solved in time $\widetilde{O}(nd^{2.5})$ where $d = \max(d_1, d_2)$.

The nonconvex approach uses the Burer-Monteiro factorization (Burer and Monteiro, 2003) to enforce the rank requirement. The problem is formulated as unconstrained nonconvex optimization (Recht et al., 2010; Bhojanapalli et al., 2016; Ge et al., 2017):

$$\min_{U \in \mathbb{R}^{d_1 \times r},\ V \in \mathbb{R}^{d_2 \times r}} f(UV^\top).$$

The factorized formulation can be solved by gradient descent, and the running time is more efficient than the SDP solution due to the reduced dimension of the variable.

In this thesis we study two different types of low-rank matrix recovery problems. In Chapter 4 we focus on matrix completion, where the goal is to recover a low-rank matrix X^* based on observations Y on a subset of the indices $\Omega \subset [d_1] \times [d_2]$, i.e., Y_{jk} for $(j, k) \in$

Ω. In particular, the observations might be noisy or quantized (1-bit measurements). Traditionally it is assumed that Ω is uniformly sampled, and X^* satisfies the incoherence condition, which informally means the singular vectors of X^* are "spread out". In Chapter 5 we turn our attention to matrix sensing, where the goal is to reconstruct X^* from a collection of sensing matrices $(A_i)_{i=1}^n$ and the corresponding linear measurements $b_i = \langle A_i, X \rangle$. A standard assumption in the literature is the Restricted Isometry Property (RIP), which means that the sensing matrices approximately preserve the norm of a low-rank matrix. The general formulations (convex and nonconvex) apply to both problems, and there have been provable exact recovery guarantees under appropriate regularity conditions (Candès and Recht, 2009; Recht et al., 2010; Ge et al., 2017; Bhojanapalli et al., 2016).

1.4 Beyond Worst-Case Settings

In this section, we present a few beyond worst-case settings, each corresponding to one of the problems introduced previously.

1.4.1 Learning-Augmented Algorithms

In the framework of "learning-augmented algorithms", a problem instance is provided with additional information about an optimal solution, and the goal is to develop algorithms that leverage the information to improve upon worst-case performances. The additional information is referred to as a prediction or an advice to the input, and often comes from applying machine learning techniques to historic data. In this setting, the goal of algorithm design is to obtain performance guarantees as a function of the quality of the prediction. As machine learning methods advance and the accuracy of predictions improves over time,

the resulting algorithms can achieve better performances (Mitzenmacher and Vassilvitskii, 2022).

Originated in online algorithms (Devanur and Hayes, 2009; Vee et al., 2010), this framework has been applied to many other areas of machine learning in both the online and offline settings. The webpage (Lindermayr and Megow, 2025) complies a comprehensive list of literature on this topic, including problems on mechanism design, caching, paging, scheduling, rent-or-buy, and many others. In particular, we highlight recent work on MAX-CUT and general constraint satisfaction problems (Ghoshal et al., 2025; Cohen-Addad et al., 2024), since they are closely related to the Boolean satisfiability problem (SAT) we study in Chapter 2.

In the context of learning-augmented algorithms for the SAT problem, the prediction (referred to here as advice) is an assignment on some or all of the variables in the CNF formula. We consider two types of advices: the label advice, which predicts each variable with accuracy slightly better than a random guessing, and the subset advice, which reveals the ground-truth assignment on a small subset of the variables. In the literature, these are also referred to as the noisy prediction (label advice) and partial prediction (subset advice).

1.4.2 Bilu-Linial Stability

Motivated by the observation that many problems that are NP-hard in the worst case can be solved efficiently in practice due to structural properties of real-world inputs, Bilu and Linial introduced the Bilu–Linial stability notion, originally for the MAX-CUT prob-

lem (Bilu and Linial, 2012). Informally, it assumes that the problem instance has a unique optimal solution that remains unchanged under small perturbations to the input.

Many problems have been studied under Bilu-Linial stability, including MAX-CUT (Bilu and Linial, 2012; Makarychev et al., 2014), max independent set (Angelidakis et al., 2018), and center-based clustering such as k-means, k-median (Awasthi et al., 2012; Balcan and Liang, 2016; Angelidakis et al., 2017), k-center (Balcan et al., 2020) and min-sum (Ben-David and Reyzin, 2014). Other metric based problems include the traveling salesman problem (Mihalák et al., 2011) and the Steiner tree problem (Freitag et al., 2021). These works are also closely related to robust algorithms (Makarychev et al., 2014) and certified algorithms (Makarychev and Makarychev, 2020), as well as to an interesting connection between stability and independent systems/matroids (Chatziafratis et al., 2017).

In Chapter 3, we apply the Bilu-Linial stability assumption to non-center-based clustering. For a clustering instance, we model perturbations to the input as multiplicative changes to the metric d. Informally, a γ-perturbation means we can change the distance between any pair of points by a factor up to γ, and by convention $\gamma \geq 1$. Closely related to our work, there have been extensive research on center-based clustering, with polynomial time algorithms for k-medians objective, first for $\gamma \geq 3$ (Awasthi et al., 2012), then $\gamma \geq 2.41$ (Balcan and Liang, 2016), and finally $\gamma \geq 2$ for general center-based objective (Angelidakis et al., 2017). On the other hand, there are hardness results for $\gamma < 2$ (Ben-David and Reyzin, 2014; Balcan et al., 2020).

1.4.3 Semi-Random Model

Many machine learning problems have been studied under distributional analysis, where we assume inputs come from some probability distribution, and the algorithms are designed and evaluated based on "good-on-average performance" (Roughgarden, 2019) with respect to fully random inputs from the distribution. However, this approach can be brittle to model mis-specification and input contamination. The semi-random model combines the random (average-case) inputs and adversarial (worst-case) inputs, and the goal is to design more robust algorithms that do not fully rely on distributional input assumptions. In this thesis we consider semi-random models with a "monotone adversary": we assume first the inputs are generated according to a probability distribution, then an adversary can modify the inputs in a restricted, sometimes even "helpful", manner.

First introduced by (Blum and Spencer, 1995), the semi-random model has been studied for various graph problems (Feige and Kilian, 2001; Perry and Wein, 2017; Mathieu and Schudy, 2010; Makarychev et al., 2012). Previously the work of (Cheng and Ge, 2018) applied the semi-random model to the matrix completion problem, and (Kelner et al., 2023a) studied sparse vector recovery, which are closely related to the matrix recovery problems we study in this thesis.

In Chapter 4 we consider variants of the matrix completion problem in the semi-random model. Unlike the traditional uniform observation model, where each entry is observed independently with probability p, here we assume each entry is observed with some unknown

probability at least p, and can otherwise be arbitrary. Equivalently, one may view the process as first sampling indices of the matrix uniformly at random, then an adversary can augment the samples with arbitrary additional indices. In Chapter 5 we turn to the matrix sensing problem, where in addition to the original sensing matrices, the adversary can add arbitrary sensing matrices along with their corresponding measurements. For both problems, the regularity conditions provided by fully random model are no longer guaranteed under the semi-random model, which poses challenges to the efficient nonconvex approaches, since there could be spurious local optimum in the optimization landscape without these regularity conditions.

CHAPTER 2

LEARNING-AUGMENTED ALGORITHMS FOR BOOLEAN SATISFIABILITY

2.1 Introduction

The Boolean satisfiability problem (SAT) is a cornerstone of computational complexity theory and algorithmic research. Given a Boolean formula over variables that can take values true or false, the task is to decide whether there exists an assignment of these variables that makes the formula evaluate to true. SAT is most commonly studied in its conjunctive normal form (CNF), where the formula is represented as a conjunction of clauses, each clause being a disjunction of literals (a variable or its negation). The restricted case where each clause contains at most k literals is known as k-SAT. SAT was the first problem proven to be NP-complete, via the Cook–Levin theorem (Cook, 1971; Levin, 1973), with 3-SAT often serving as its canonical example (Karp, 1972). This foundational result implies that any problem in NP can be efficiently reduced to SAT, making it a central object in the study of computational intractability, reductions, and practical solving techniques.

Since polynomial-time algorithms that solve all instances of SAT optimally are unlikely to exist unless P $\neq$ NP, alternative approaches are necessary. One approach is to abandon the requirement of polynomial runtime and seek exponential-time algorithms that, while still exponential in the worst case, are faster than exhaustive search. For $k \geq 3$, let

$c_k \geq 1$ be a constant such that k-SAT can be solved in time $O^*(c_k)^n$, where n is the number of variables in the given k-SAT instance and $O^*(\cdot)^n$ hides polynomial factors. A well-known conjecture proposed by (Impagliazzo and Paturi, 2001), called the Exponential Time Hypothesis (ETH), posits that 3-SAT cannot be solved in sub-exponential time, i.e., $c_3 > 1$. A positive answer to this conjecture would imply that P $\neq$ NP. A stronger conjecture proposed by (Calabro et al., 2009), known as the Strong Exponential Time Hypothesis (SETH), claims that $\lim_{k\to\infty} c_k = 2$.

Another canonical variant of the Boolean satisfiability problem is its optimization counterpart, MAX-SAT, where the objective is to determine the maximum number of clauses that can be satisfied by any assignment. Since MAX-SAT generalizes the decision problem SAT, it is also computationally intractable unless P = NP. One approach to tackling this problem is to design polynomial-time approximation algorithms. However, it is computationally hard to compute an approximate solution that satisfies a number of clauses arbitrarily close to the optimal. More precisely, MAX-SAT is APX-complete, indicating that it does not admit a polynomial-time approximation scheme (PTAS) unless P = NP (Feige and Goemans, 1995; Arora and Safra, 1998; Arora et al., 1998; Hastad, 2001). Therefore, there has been extensive study to find the best possible approximation factors for this problem, particularly for MAX-k-SAT , the restricted case of MAX-SAT where each clause contains at most k literals.

There has been extensive research on SAT beyond worst-case performance, including random (average-case) (Selman et al., 1996) and semi-random models (Roughgarden, 2021,

Section 9), smoothed analysis (Feige, 2007), stability analysis (Kun and Reyzin, 2014, Section 2), parameterized complexity (Roughgarden, 2021, Section 2), and SAT-solver heuristics such as DPLL and CDCL (Roughgarden, 2021, Section 25). We focus on the emerging paradigm of *learning-augmented algorithms* (Mitzenmacher and Vassilvitskii, 2022), also known as "algorithms with predictions". In this paradigm, a machine learning method provides a prediction (or "advice") about the input or the optimal solution, and the algorithm uses this prediction to improve its performance, with guarantees that depend on the accuracy of the prediction. This approach has been applied to numerous algorithmic tasks, particularly NP-complete problems such as MAX-CUT, MAX-k-LIN, Independent Set and Clustering, among others (Ghoshal et al., 2025; Cohen-Addad et al., 2024; Braverman et al., 2024; Bampis et al., 2024; Dong et al., 2025; Bampis et al., 2025; Ergun et al., 2021; Gamlath et al., 2022; Nguyen et al., 2023). This paper focuses on studying the canonical NP-complete problem of SAT.

2.1.1 Problem Formulation

Let $\phi(x) = C_1 \wedge C_2 \wedge \cdots \wedge C_m$ be a Boolean formula in conjunctive normal form (CNF) over variables $x = x_1, x_2, \ldots, x_n$, consisting of m clauses $C_1, C_2, \ldots, C_m$.

Definition 2.1.1 (SAT)**.** *The Satisfiability Problem asks whether there exists a truth assignment $\sigma : \{x_1, \ldots, x_n\} \to \{0, 1\}$ to a CNF formula ϕ such that $\phi(x)|_\sigma = 1$, i.e., all clauses C_i are satisfied under σ. When each clause in ϕ contains at most k literals, the*

problem is referred to as k-SAT. If every clause contains exactly k literals, it is known as MAX-Ek-SAT.

Definition 2.1.2 (MAX-SAT)**.** *Given a CNF formula ϕ, the MAX-SAT problem seeks a truth assignment $\sigma : \{x_1, \ldots, x_n\} \to \{0, 1\}$ that maximizes the number of clauses C_i for which $C_i|_\sigma = 1$, i.e., the number of clauses satisfied by σ. When each clause in ϕ contains at most k literals, the problem is referred to as MAX-k-SAT. If every clause contains exactly k literals, it is known as MAX-Ek-SAT .*

Let $x^* = \{x_1^*, \ldots, x_n^*\}$ be a fixed optimal solution to either SAT or MAX-SAT. We consider two models of advice: one provides full certainty on a subset of the values in x^*, and the other offers noisy information about all values in x^*.

Definition 2.1.3 (Subset Advice)**.** *The subset advice consists of a random subset of indices $S \subset \{1, \ldots, n\}$ along with the ground-truth assignment x^* restricted to these indices $\{x_i^*\}_{i \in S}$, where each i is included in S independently with probability ϵ for all $i = 1, \ldots, n$.*

Definition 2.1.4 (Label Advice)**.** *The label advice is an assignment $\widetilde{x} \in \{0, 1\}^n$ that contains noise relative to a ground-truth optimal assignment x^*. For each $i = 1, \ldots, n$, independently, we have*

$$\widetilde{x_i} = \begin{cases} x_i^* & \text{with probability } \frac{1+\epsilon}{2}, \\ 1 - x_i^* & \text{with probability } \frac{1-\epsilon}{2}. \end{cases}$$

It is important to note that the randomness in the label advice is sampled once and fixed (i.e., the oracle returns the same answer if queried multiple times), which is standard in the literature on learning-augmented algorithms. Otherwise, one could trivially boost the algorithm's probability of success. See the discussion of persistent vs. non-persistent noise in (Braverman et al., 2024).

Moreover, as noted by (Ghoshal et al., 2025), the subset advice model is stronger than the label advice model, since label advice $\widetilde{x}$ can be simulated given subset advice.

2.1.2 Our Contributions

Under the learning-augmented framework, we study the SAT and MAX-SAT problems with advice. Here, we highlight our contributions and provide a road map of the paper.

In Section 2.2, we study the decision problem of k-SAT with advice and improve the running time of state-of-the-art algorithms. Specifically, we incorporate subset advice into a family of algorithms known as PPZ (Section 2.2.1) and PPSZ (Section 2.2.2). As both algorithms run in exponential time in the worst case, we present our results as improvements on the base constant of the exponent, summarized in Table I. For PPZ, we improve the running time from $2^{(1-\frac{1}{k})}$ to $2^{(1-\frac{1}{k}\frac{1-\epsilon^k}{1-\epsilon})(1-\epsilon)}$, and for PPSZ we improve the base of the exponent by a multiplicative factor of $2^{-c(\epsilon,k)}$, where $c(\epsilon,k)>0$. In particular, for 3-SAT the base constant of the exponent becomes $2^{\left(\frac{\epsilon}{1-\epsilon}+2\ln(2-2\epsilon)-1+o(1)\right)}$ for $\epsilon<\frac{1}{2}$ comparing to $2^{2\ln 2-1+o(1)}$ without advice, and when $\epsilon\geq 1/2$, the running time becomes sub-exponential. We also provide a hardness result for 3-SAT with subset advice under ETH (Section 2.2.3).

In Section 2.3, we study the optimization problem of MAX-SAT and its variants MAX-k-SAT, aiming to improve the approximation factors of polynomial-time algorithms. With subset advice (Section 2.3.1), we show that any α-approximation algorithm for a variant of MAX-SAT can be turned into an approximation of $\alpha+(1-\alpha)\epsilon$ by incorporating the subset advice in a black-box fashion. To complement this result, we establish hardness of approximation for MAX-3-SAT with subset advice, assuming Gap-ETH (Section 2.3.2). With label advice, we focus specifically on the MAX-2-SAT problem (Section 2.3.3), extending the quadratic programming approach for MAX-2-LIN from (Ghoshal et al., 2025) to obtain a near-optimal approximation for instances whose average degree exceeds a threshold depending only on the amount of advice.

Comparison to prior work. Under the subset and label advice models, the recent works of (Ghoshal et al., 2025) and (Cohen-Addad et al., 2024) both study closely related problems: MAX-CUT, MAX-k-LIN, and MAX-2-CSP. In particular, (Ghoshal et al., 2025) studies MAX-k-LIN for $k = 2, 3, 4$, which includes MAX-CUT as a special case. For their positive results, they design algorithms in the weaker label advice model and give near-optimal solutions to MAX-2-LIN, under the assumption that the instance has large average degree. For their negative results, they show conditional hardness for MAX-3-LIN and MAX-4-LIN in the stronger subset advice model. The work of (Cohen-Addad et al., 2024) provides positive results in both advice models. With label advice, they achieve an $\alpha_{GW} + \Omega(\epsilon^4)$ approximation for MAX-CUT and MAX-2-CSP based on a notion of "wide" and "narrow" graphs and their respective properties. Notably, their label advice model

assumes only pairwise independence, whereas both ours and (Ghoshal et al., 2025) assume mutual independence. With subset advice, they achieve $\alpha_{GW} + \Omega(\epsilon^2)$ and $\alpha_{RT} + \Omega(\epsilon)$ approximations for MAX-CUT[1]. Both of these works study the optimization problem, but not the decision problem.

2.1.3 Related Work

Learning-augmented algorithms. The idea of using additional information, such as a prediction about the future or a suggestion about the solution, to improve an algorithm's performance originated in the field of online algorithms (Devanur and Hayes, 2009; Vee et al., 2010). The formal framework was introduced by Lykouris and Vassilvitskii (Lykouris and Vassilvitskii, 2021), who defined the key notions of consistency and robustness for evaluating algorithm performance. Consistency refers to improved performance when the predictions are accurate, while robustness ensures that the algorithm performs comparably to a standard, prediction-free algorithm even when the predictions are unreliable. Numerous algorithmic problems have been studied in this framework, including caching, paging, the ski rental problem, online bipartite matching, scheduling, load balancing, and online facility location (Purohit et al., 2018; Mitzenmacher, 2020; Lattanzi et al., 2020; Rohatgi, 2020; Wei and Zhang, 2020; Lykouris and Vassilvitskii, 2021; Antoniadis et al., 2023). For broader context, see the survey (Mitzenmacher and Vassilvitskii, 2022) and the online database of papers in the field (Lindermayr and Megow, 2025).

---

[1] $\alpha_{GW} = 0.878$ (Goemans and Williamson, 1994), $\alpha_{RT} = 0.858$ (Raghavendra and Tan, 2012).

SAT. Improvements over exhaustive search in the worst case were achieved by a family of deterministic algorithms based on branching (Monien and Speckenmeyer, 1985; Schiermeyer, 1993; Rodošek, 1996; Kullmann, 1999). Another family of algorithms is based on local search, initiated by the randomized algorithm of (Schoning, 1999), and later improved (and in some cases derandomized) by subsequent works (Hofmeister et al., 2002; Baumer and Schuler, 2003; Iwama and Tamaki, 2004; Moser and Scheder, 2011; Liu, 2018). A third family of randomized algorithms is based on random restrictions, initiated by Paturi, Pudlak and Zane (Paturi et al., 1997) and Paturi, Pudlák, Saks, and Zane (Paturi et al., 2005), known as the PPZ and PPSZ algorithms. (Hertli, 2014a; Hertli, 2014b) improved the analysis of PPSZ, which was later simplified by (Scheder and Steinberger, 2017) and slightly improved by (Qin and Watanabe, 2020). A variant of PPSZ, named biased PPSZ, was introduced by Hansen, Kaplan, Zamir, and Zwick (Hansen et al., 2019), and an improved analysis of PPSZ by (Scheder, 2024) currently represents the state of the art, with 1.307^n for 3-SAT.

MAX-SAT. (Goemans and Williamson, 1994) achieved a significant breakthrough by introducing a semidefinite programming (SDP) relaxation combined with randomized rounding, resulting in a 0.878-approximation algorithm for MAX-2-SAT. Building upon this, (Feige and Goemans, 1995) improved the approximation ratio to 0.931 by applying a rotation method prior to randomized rounding. (Zwick, 2000) further refined the analysis of these algorithms. Subsequently, (Matuura and Matsui, 2001) improved the approximation ratio to 0.935 by employing a skewed distribution during the rounding phase. Finally,

Lewin, Livnat, and Zwick (Lewin et al., 2002) combined these techniques (SDP relaxation, rotation, and skewed distribution rounding) to develop an algorithm achieving a 0.940-approximation for MAX-2-SAT. Under the Unique Games Conjecture (UGC), this 0.940-approximation is proven to be optimal, indicating that no polynomial-time algorithm can achieve a better approximation ratio for MAX-2-SAT unless the UGC is false (Brakensiek et al., 2024; Austrin, 2007). Assuming only that P $\neq$ NP, the best achievable approximation ratio is approximately 0.954, as established by (Hastad, 2001).

A semidefinite programming relaxation technique was also applied to MAX-3-SAT, where (Karloff and Zwick, 1997) achieved a 7/8-approximation algorithm[1]. For the case where each clause contains exactly three literals (MAX-Ek-SAT), (Johnson, 1973) showed that a simple random assignment achieves a 7/8-approximation as well. (Hastad, 2001) later proved that this is optimal, showing that no polynomial-time algorithm can achieve a better approximation ratio unless P = NP. For the general MAX-SAT problem, the current best-known approximation factor is 0.796, obtained by Avidor, Berkovitch, and Zwick (Avidor et al., 2005).

2.2 Improvements on the Running Time of the Decision Problem

In this section, we show how subset advice can improve the running time for k-SAT using algorithms in the family of "random restriction algorithms" (Paturi et al., 1997; Paturi et

[1]This guarantee holds for satisfiable instances, and there is strong evidence suggesting the algorithm achieves the same approximation ratio for unsatisfiable instances as well.

al., 2005; Hansen et al., 2019; Scheder and Steinberger, 2017), initiated by the influential PPZ algorithm by Paturi, Pudlák, and Saks. We start with the relatively straightforward PPZ algorithm (Paturi et al., 1997), and then dive into the more involved PPSZ algorithm (Paturi et al., 2005; Hertli, 2014a). A summary of existing results and our improvements can be found in Table I.

	Original	Subset Advice
PPZ	$c_k = 2^{1-1/k}$ $c_3 \approx 1.587$ (Paturi et al., 1997)	$c'_k \leq c_k^{(1-\epsilon)}$ Theorem 2.2.1
PPSZ	$c_k = 2^{1-R_k+o(1)}$ $c_3 \approx 1.308$ (Paturi et al., 2005)	for $\epsilon < \frac{k-2}{k-1}$: $c'_k = c_k \cdot 2^{-\epsilon_k}$, where $\epsilon > \epsilon_k > 0$ for $\epsilon \geq \frac{k-2}{k-1}$: $c'_k = 2^{o(1)}$, sub-exponential time Theorem 2.2.2

TABLE I: Exact exponential-time algorithms for k-SAT.
All algorithms are of the form $O^*(c_k)^n$, where $O^*(\cdot)^n$ hides polynomial factors. R_k and ϵ_k are constants depending on k. See Equations (2.1) and (2.2) for details.

We consider the Unique-k-SAT problem (i.e., instances with at most one satisfying assignment) for simplicity. According to (Hertli, 2014a; Scheder and Steinberger, 2017; Scheder, 2024), Unique-k-SAT bounds can be lifted to general k-SAT with multiple satisfying assignments. In particular, the generalization from (Hertli, 2014a) use the results on

Unique-k-SAT in a black-box fashion and show that the same bounds hold for k-SAT, therefore we analyze the improvements on Unique-k-SAT, and our results apply to k-SAT as well. (Cf. Theorems 1, 2, 7 (Hertli, 2014a) and Theorem 11 (Scheder and Steinberger, 2017).)

In Algorithm 2.1 (Paturi et al., 1997; Hertli, 2014a), we incorporate the subset advice into both PPZ and PPSZ: with inputs $S = \varnothing$ (i.e. $\epsilon = 0$) and $D = 1$, Algorithm 2.1 recovers the PPZ algorithm; with inputs $S = \varnothing$ (i.e. $\epsilon = 0$) and $D = o\left(\frac{n}{\log n}\right)$, Algorithm 2.1 recovers the PPSZ algorithm. We use the notation $\phi = \phi|_\sigma$ to denote reducing a formula ϕ based on some partial assignment σ, i.e., given ϕ and σ, remove the clauses satisfied by variables in σ as well as literals set to false by σ.

Algorithm 2.1: PPZ/PPSZ with subset advice (Paturi et al., 1997; Hertli, 2014a)

Input: (i) k-CNF formula ϕ where V is the set of variables in ϕ, (ii) random subset advice S where each variable is included with probability ϵ, (iii) implication parameter D, (iv) number of iterations T.

Initialize: Let σ be the empty assignment on V.

1. For each assignment $b \in \{0, 1\}$ for variable x_i in S, let $\sigma(x_i) = b$ and $\phi = \phi|_\sigma$.
2. Choose a random permutation π of the remaining variables.
3. For each variable x_i in the order of π:
   - Enumerate over all sets of D clauses: if the value of x_i is forced to be $b \in \{0, 1\}$ by some of these sets (given previously assigned variables), then set $\sigma(x_i) = b$ and $\phi = \phi|_\sigma$.
   - Else, set $\sigma(x_i) = 0$ or 1 uniformly at random and $\phi = \phi|_\sigma$.
4. If σ is a satisfying assignment, return σ. Otherwise, repeat steps 2-3 at most T times. If a satisfying assignment was not found, return "unsatisfiable".

2.2.1 PPZ Algorithm for k-SAT With Subset Advice

First, we briefly introduce some analysis of the PPZ algorithm. Although the original analysis of PPZ (Paturi et al., 1997) applies to general k-SAT, it is more involved, and we adopt the simplified arguments summarized in (Hansen et al., 2019), which assume a unique satisfying assignment.

Under the uniqueness assumption, each variable x_i has a "critical clause" C_{x_i}, where the literal associated with x_i is the only one set to true in the clause. The assignments of variables are either "forced" or "guessed": if a variable x_i appears in the permutation π after all other variables in C_{x_i}, that is, if during the execution of the algorithm there exists a unit clause involving x_i or $\overline{x_i}$, then x_i is forced, and its literal is set to true. Since the permutation is random, the probability that x_i is forced is at least $1/k$. If the variable is not forced, then it is guessed uniformly at random. The running time improves when more variables are forced.

Denote by $G(\pi)$ the number of guessed variables given the order of π. If the formula is satisfiable, we can lower bound the success probability of one iteration of the randomized algorithm

$$\Pr[\text{All guessed variables are correct}] = \mathbb{E}_\pi \left[2^{-G(\pi)}\right] \geq 2^{-\mathbb{E}_\pi[G(\pi)]},$$

which follows from Jensen's inequality. Note that we can analyze the probability of each variable being guessed individually. We repeat this process for $T = O^*\left(2^{\mathbb{E}_\pi[G(\pi)]}\right)$ iterations

to succeed with high probability, ensuring a Monte Carlo algorithm guarantee. Alternatively, we can have a Las Vegas algorithm that is always correct, but its running time becomes a random variable with expectation T. Thus, without advice, the running time is shown to be $O^*\left(2^{\left(1-\frac{1}{k}\right)n}\right)$. In the following Theorem, we show that the running time can be improved exponentially with subset advice.

Theorem 2.2.1 (PPZ (Paturi et al., 1997) with subset advice)**.** *Consider the decision problem of k-SAT. Given subset advice S where each variable is included independently with probability ϵ, the running time of the PPZ algorithm $O^*(c_k)^n$ can be improved exponentially, in particular, the base constant becomes $c'_k = 2^{\left(1-\frac{1}{k}\frac{1-\epsilon^k}{1-\epsilon}\right)(1-\epsilon)}$, comparing to $c_k = 2^{\left(1-\frac{1}{k}\right)}$ without advice.*

Proof. We upper bound $\mathbb{E}_\pi\left[G(\pi)\right]$ by analyzing the probability of a variable being forced in PPZ with subset advice:

$$
\begin{aligned}
&\mathbb{P}_\pi[x_i \notin S \text{ and is forced}] \\
&\geq \mathbb{P}_\pi[x_i \notin S, \text{ and all the other variables in } C_{x_i} \text{ either in } S \text{ or appear in } \pi \text{ before } x_i] \\
&\geq \sum_{j=0}^{k-1} \binom{k-1}{j} \epsilon^j (1-\epsilon)^{k-j} \frac{1}{k-j} \quad \text{(where } j \text{ variables in this clause appear in } S\text{)} \\
&= \frac{1}{k} \sum_{j=0}^{k-1} \binom{k}{j} \epsilon^j (1-\epsilon)^{k-j} \\
&= \frac{1}{k} \left[\sum_{j=0}^{k} \binom{k}{j} \epsilon^j (1-\epsilon)^{k-j} - \binom{k}{k} \epsilon^k (1-\epsilon)^{k-k} \right] \\
&= \frac{1}{k} \left(1-\epsilon^k\right).
\end{aligned}
$$

Therefore $\mathbb{E}_\pi[G(\pi)] = (1-\epsilon)n - \frac{1}{k}\left(1-\epsilon^k\right)n = n(1-\epsilon)\left(1-\frac{1}{k}\frac{1-\epsilon^k}{1-\epsilon}\right)$. The base constant of the running time with subset advice is $c'_k = 2^{\left(1-\frac{1}{k}\frac{1-\epsilon^k}{1-\epsilon}\right)(1-\epsilon)}$. Comparing to $c_k = 2^{1-\frac{1}{k}}$ of the original PPZ algorithm, we have exponential improvement on the running time depending on the advice. ■

2.2.2 PPSZ Algorithm for k-SAT With Subset Advice

The PPSZ algorithm improves upon the PPZ algorithm by introducing a preprocessing step called "D-bounded resolution" (Paturi et al., 2005), which was later relaxed to a concept called "D-implication" in an adapted version of PPSZ by (Hertli, 2014a), which we adopt here. The idea is that a variable can be forced even if not all the variables in the critical clause appeared before it (as analyzed in PPZ). In PPSZ, we force a value for a variable by enumerating over all sets of D clauses to check whether the variable takes the same value in all satisfying assignments consistent with these D clauses. If so, the variable is forced to this value, otherwise, we guess randomly. The probability of a variable being forced increases compared to PPZ, which leads to a better running time. We take $D = D(n)$ to be a slowly growing function of n, e.g., $D = o\left(\frac{n}{\log n}\right)$, so that the enumeration still runs in reasonable time.

First we summarize the analysis of the PPSZ algorithm (Paturi et al., 2005). We choose the random permutation indirectly. For each variable, we choose a uniformly random $r \in [0,1]$ representing its "arrival time", and we determine the order in the permutation by sorting the arrival times. The reason is that the arrival times are completely independent,

in contrast to choosing a random permutation directly, where the positions of two variables are not independent.

By summarizing Lemma 6,7,8 (Paturi et al., 2005), we get

$$\mathbb{P}_\pi[x_i \text{ is forced}] \geq \int_0^1 R_k(r)dr - \Delta_k^{(d)} = R_k - \Delta_k^{(d)}, \tag{2.1}$$

where $R_k(r)$ is the smallest nonnegative solution R to $\left(r + (1-r)R\right)^{k-1} = R$ and $R_k := \int_0^1 R_k(r)dr$. It is shown that $R_k(r)$ is strictly increasing for $r \in \left[0, \frac{k-2}{k-1}\right]$, and $R_k(r) = 1$ for $r \in \left[\frac{k-2}{k-1}, 1\right]$. The asymptotic error of convergence satisfies $0 \leq \Delta_k^{(d)} \leq \frac{3}{(d-1)(k-2)+2}$ where d is the minimum hamming distance between satisfying assignments. Assuming uniqueness, $\Delta_k^{(d)} = o(1)$, and goes to 0 as D goes to infinity.

It is shown that $R_k = \frac{1}{k-1}\sum_{j=1}^{\infty} \frac{1}{j\left(j+\frac{1}{k-1}\right)}$ for $k \geq 3$. Similarly to the PPZ algorithm, we perform $T = O^*\left(2^{\mathbb{E}_\pi[G(\pi)]}\right)$ iterations, where here $\mathbb{E}_\pi[G(\pi)] = n\left(1 - R_k + \Delta_k^{(d)}\right)$. For $k = 3$, we can explicitly evaluate $R_3 = 2 - 2\ln 2$, and $T = O^*\left(2^{(2\ln 2 - 1 + o(1))}\right)^n$.

In Theorem 2.2.2 we show that we can improve the running time for k-SAT with subset advice, then we state the result explicitly for 3-SAT in Corollary 2.2.3.

Theorem 2.2.2 (PPSZ with subset advice for k-SAT)**.** *Consider the decision problem of k-SAT and suppose without advice the PPSZ algorithm runs in time $O^*(c_k)^n$. Given subset advice S with $0 < \epsilon < \frac{k-2}{k-1}$, we improve the base constant of the running time to $c'_k = c_k \cdot 2^{-\epsilon_k}$, where $\epsilon_k = \epsilon - \int_0^\epsilon R_k(r)dr$. In particular, $\epsilon > \epsilon_k > 0$. For $\epsilon \geq \frac{k-2}{k-1}$, the running time becomes $O^*(2^{o(n)})$, i.e. sub-exponential.*

Proof. We follow similar arguments as in the proof of (Paturi et al., 2005) for general $k \geq 3$. For each variable in the permutation, we associate it with a uniformly random $r \in [0, 1]$ which represents its "arrival time" according to π. Following Lemma 6,7,8 (Paturi et al., 2005),

$$\mathbb{P}_\pi[x_i \notin S \text{ and is forced}] \geq (1-\epsilon)\left(\int_0^1 \widetilde{R_k}(r)dr - \Delta_k^{(d)}\right),$$

where $\widetilde{R}_k(r)$ is the smallest nonnegative solution $\widetilde{R}$ to $\left[\epsilon + (1-\epsilon)\Big(r + (1-r)\widetilde{R}\Big)\right]^{k-1} = \widetilde{R}$.

Comparing to the equation for R, here each branch in the "critical clause tree" is more likely to be cut by time r due to the advice: if the variable associated with the branch is in the subset advice, the branch is cut, otherwise the original recursive expression applies. Observe that by a change of variable with $u = g(r) = \epsilon + (1-\epsilon)r$, we can replace $\widetilde{R_k}(r)$ with $R_k(u)$, the the smallest nonnegative solution to $R = [u + (1-u)R]^{k-1}$, which has the same form as in the original PPSZ.

We evaluate the probability of a variable being forced and the value of $\mathbb{E}_\pi[G(\pi)]$,

$$\begin{aligned}
\mathbb{P}_\pi[x_i \notin S \text{ and is forced}] &\geq (1-\epsilon)\left(\int_0^1 \widetilde{R_k}(r)dr - \Delta_k^{(d)}\right) \\
&\geq \int_0^1 \widetilde{R_k}(r)(1-\epsilon)dr - \Delta_k^{(d)} \\
&= \int_{g^{-1}(\epsilon)}^{g^{-1}(1)} R_k(g(r))g'(r)dr - \Delta_k^{(d)} \\
&= \int_\epsilon^1 R_k(u)du - \Delta_k^{(d)} \\
&= R_k - \Delta_k^{(d)} - \int_0^\epsilon R_k(r)dr.
\end{aligned}$$

$$\begin{aligned}
\mathbb{E}_\pi[G(\pi)] &= n\left[(1-\epsilon) - \left(R_k - \Delta_k^{(d)} - \int_0^\epsilon R_k(r)dr\right)\right] \\
&= n\left(1 - R_k + \Delta_k^{(d)} - \epsilon_k\right), \qquad (2.2)
\end{aligned}$$

where $\epsilon_k = \epsilon - \int_0^\epsilon R_k(r)dr$. Since $0 \leq R_k(r) < 1$ for $r \in \left[0, \frac{k-2}{k-1}\right]$, for $\epsilon < \frac{k-2}{k-1}$ we have $\epsilon > \epsilon_k > 0$. Compare to $c_k = 2^{1-R_k+\Delta_k^{(d)}}$ without advice, the improved base constant is $c'_k = c_k \cdot 2^{-\epsilon_k}$.

Notably, in the original PPSZ, variables arrive "late" in the permutation with $r \in [\frac{k-2}{k-1}, 1]$ are forced almost surely. Given subset advice with $\epsilon > \frac{k-2}{k-1}$, if a variable is not included in the advice, the probability that it is forced goes to 1. Under the uniqueness assumption we have $\Delta_k^{(d)} = o(1)$, so $\mathbb{E}_\pi[G(\pi)]$ becomes $o(n)$, i.e., a sub-linear number of variables are guessed, resulting in sub-exponential running time. Refer to (Paturi et al.,

2005; Hertli, 2014a) for more rigorous analyses on PPSZ as well as discussions in (Hansen et al., 2019). ■

Corollary 2.2.3 (PPSZ with subset advice for 3-SAT)**.** *Consider the decision problem of 3-SAT . Given subset advice S where each variable is included independently with probability ϵ, the running time of the PPSZ algorithm $O^*(c_k)^n$ can be improved exponentially, in particular, the base constant becomes $c_3 = 2^{\left(\frac{\epsilon}{1-\epsilon}+2\ln(2-2\epsilon)-1+o(1)\right)}$ for $\epsilon < \frac{1}{2}$ comparing to $c_3 = 2^{2\ln 2-1+o(1)}$ without advice, and for $\epsilon \geq \frac{1}{2}$, the running time becomes $O^*(2^{o(n)})$, i.e. sub-exponential.*

Proof. Recall that $R_k(r)$ is the smallest nonnegative solution R to $\left(r+(1-r)R\right)^{k-1} = R$. Now we focus on the case that $k = 3$ and solve for $R_3(r)$,

$$R_3(r) = \begin{cases} \left(\frac{r}{1-r}\right)^2 & 0 \leq r \leq \frac{1}{2} \\ 1 & \frac{1}{2} \leq r \leq 1 \end{cases}.$$

Then we evaluate ϵ_3,

$$\begin{aligned} \epsilon_3 &= \epsilon - \int_0^{\epsilon} R_3(r)dr \\ &= \epsilon - \left(\frac{1}{1-\epsilon} + 2\ln(1-\epsilon) - 1 + \epsilon\right) \\ &= -\frac{1}{1-\epsilon} - 2\ln(1-\epsilon) + 1. \end{aligned}$$

Now we apply Theorem 2.2.2. Recall that without advice $R_3 = 2 - 2\ln 2$ and $c_3 = 2^{2\ln 2 - 1 + o(1)}$, therefore

$$\begin{aligned} c_3' &= c_3 \cdot 2^{-\epsilon_3} \\ &= 2^{2\ln 2 - 1 + o(1) - \epsilon_3} \\ &= 2^{\left(\frac{\epsilon}{1-\epsilon} + 2\ln(2-2\epsilon) - 1 + o(1)\right)}. \end{aligned}$$

■

2.2.3 Hardness of 3-SAT With Subset Advice

We state a hardness result for 3-SAT given subset advice assuming the Exponential Time Hypothesis (ETH) below.

Conjecture 2.2.4 (ETH (Impagliazzo and Paturi, 2001))**.** *There exists $\delta > 0$ such that no algorithm can solve 3-SAT in $O(2^{\delta n})$ time where n is the number of variables.*

Theorem 2.2.5 (Hardness of 3-SAT with subset advice)**.** *Assuming the ETH, there exists $\epsilon_0 > 0$ such that for all $\epsilon \leq \epsilon_0$, there is no sub-exponential time algorithm for 3-SAT given subset advice where each variable is included with probability ϵ.*

Proof. Let $\epsilon_0 < \delta$, where δ is the constant in the ETH. Suppose there is an algorithm that runs in sub-exponential time $f(n)$ and solves 3-SAT given subset advice with parameter $\epsilon \leq \epsilon_0$. Fix a subset of size ϵn, we can simulate a subset advice by enumerating all possible

assignments then run the algorithm, thereby solving 3-SAT in time $O(2^{\epsilon n} \cdot f(n)) \leq O(2^{\delta n})$, contradicting ETH. ■

2.3 Improving the Approximation Factor for the Optimization Problem

In this section we study the optimization problem of MAX-SAT with advice in order to improve the approximation factors of polynomial time algorithms. First we show that by incorporating the subset advice in a black-box fashion into any approximation algorithm, we gain an $\Omega(\epsilon)$ improvement over the original approximation factor. Then we focus on MAX-2-SAT with label advice. Inspired by the work of (Ghoshal et al., 2025), we extend their work on MAX-2-LIN and adapt their techniques to the more general problem of MAX-2-SAT.

2.3.1 MAX-SAT With Subset Advice

Given a subset advice S, we incorporate it into an approximation algorithm with the following two-step process, and state the performance guarantee in Theorem 2.3.1.

1. Set the variables in S based on the advice, remove all satisfied clauses and unsatisfied literals;
2. Run the approximation algorithm of choice on the reduced instance.

Theorem 2.3.1 (MAX-SAT with subset advice)**.** *Consider a MAX-SAT instance and an α-approximation algorithm. Suppose we have subset advice S where each variable is included independently with probability ϵ, then the approximation ratio can be improved to*

$\alpha + (1 - \alpha)\epsilon$.

In particular, the approximation ratio for MAX-SAT is at least $0.796 + 0.204\epsilon$ *based on* $\alpha \geq 0.796$ *achieved by (Avidor et al., 2005).*

Proof. Given a MAX-SAT instance with m clauses on n variables, suppose $m^* \leq m$ clauses are satisfied in an optimal assignment x^*. For any clause that is satisfied in OPT, we assume (pessimistically) that the clause is satisfied by exactly one of its literals in x^*. Given subset advice S, where each variable is included independently with probability ϵ, each literal's assignment in OPT is revealed with probability ϵ. In particular, the satisfied literal is revealed with probability ϵ, which reduces this clause. In expectation, step 1 reduces ϵm^* of the satisfiable clauses. In step 2, $(1-\epsilon)m^*$ of the satisfiable clauses remain, and an α-approximation algorithm will satisfy at least $\alpha(1-\epsilon)m^*$ of them.

In total, the number of satisfied clauses is at least $\epsilon m^* + \alpha(1-\epsilon)m^*$, and the approximation ratio is $\alpha' = \epsilon + \alpha(1-\epsilon) = \alpha + (1-\alpha)\epsilon$. ■

Our result for MAX-SAT holds in general for the optimization problem of Boolean satisfiability, including MAX-2-SAT , MAX-3-SAT , and the performance ratio depends on the state-of-the-art approximation algorithm for the specific problem, as stated by the following corollaries.

Corollary 2.3.2 (MAX-2-SAT with subset advice)**.** *Given subset advice S where each variable is included independently with probability* ϵ*, the approximation ratio for MAX-2-SAT is at least* $0.940 + 0.06\epsilon$ *based on* $\alpha \geq 0.940$ *achieved by (Lewin et al., 2002).*

Corollary 2.3.3 (MAX-3-SAT with subset advice)**.** *Given subset advice S where each variable is included independently with probability ϵ, the approximation ratio for MAX-3-SAT is at least $\frac{7}{8} + \frac{1}{8}\epsilon$ where $\alpha \geq \frac{7}{8}$ for fully satisfiable instances according to (Karloff and Zwick, 1997)*[1].

Without advice, the corresponding inapproximability bound assuming P $\neq$ NP is 0.9545 for MAX-2-SAT and $\frac{7}{8}$ for MAX-3-SAT (even on fully satisfiable instances) (Hastad, 2001). Assuming the Unique Games Conjecture, the 0.940 approximation for MAX-2-SAT is optimal (Brakensiek et al., 2024).

2.3.2 Hardness of MAX-3-SAT With Subset Advice

We state a hardness result for MAX-3-SAT with subset advice below, assuming the Gap Exponential Time Hypothesis (Gap-ETH). Alternatively, instead of assuming Gap-ETH, we could assume the Exponential Time Hypothesis (ETH) together with the Linear-Size PCP Conjecture (cf. (Dinur, 2016; Manurangsi and Raghavendra, 2016)).

Conjecture 2.3.4 (Gap-ETH (Dinur, 2016; Manurangsi and Raghavendra, 2016))**.** *There exist constants δ, γ such that given MAX-3-SAT instance ϕ, no $O(2^{\delta n})$-time algorithm can distinguish between the case that $\mathrm{sat}(\phi) = 1$ and the case that $\mathrm{sat}(\phi) \leq 1 - \gamma$, where $\mathrm{sat}(\phi)$ denotes the maximum fraction of satisfiable clauses.*

---

[1] *It is conjectured that $\alpha \geq \frac{7}{8}$ holds for arbitrary MAX-3-SAT instances as well.*

Theorem 2.3.5 (Hardness of MAX-3-SAT with subset advice)**.** *Assuming the Gap-ETH, there exists $\epsilon_0 = \epsilon_0(\delta, \gamma) > 0$ such that for all $\epsilon \leq \epsilon_0$, there is no sub-exponential time algorithm for MAX-3-SAT given subset advice with parameter ϵ, such that given a satisfiable instance returns a solution that satisfies at least a $(1-\gamma)$-fraction of the clauses.*

Proof. Let $\epsilon_0 < \delta$ in the Gap-ETH. Suppose there is an algorithm that runs in sub-exponential time $f(n)$, and given a fully satisfiable instance of MAX-3-SAT as well as a subset advice with parameter $\epsilon \leq \epsilon_0$, returns a solution satisfying at least a $(1-\gamma)$-fraction of the clauses. Given input ϕ, we fix a subset of size ϵn and simulate a subset advice by enumerating all possible assignments, then run the algorithm. If ϕ is fully satisfiable, eventually we will get a solution satisfying at least a $(1-\gamma)$-fraction of the clauses; on the other hand, if $\text{sat}(\phi) < 1-\gamma$, no solution can satisfy a $(1-\gamma)$-fraction, thereby we can distinguish between the two cases in time $O(2^{\epsilon n} \cdot f(n)) \leq O(2^{\delta n})$, contradicting Gap-ETH. ■

2.3.3 MAX-2-SAT With Label Advice

Given a label advice to an optimization problem, we first evaluate the performance of directly adopting the advice as a solution. Suppose we are given a label advice $\widetilde{x}$ to a MAX-Ek-SAT problem based on a ground-truth optimal assignment x^*. Consider a clause C satisfied by x^*. We assume the worst-case where exactly one literal in C is set to true

by x^* to obtain a lower-bound on the probability of C being satisfied by $\widetilde{x}$, which gives us the approximation factor,

$$\alpha_k \geq \mathbb{P}_{\widetilde{x}}[C \text{ is satisfied by } \widetilde{x} \mid C \text{ is satisfied by } x^*] = 1 - \left(\frac{1-\epsilon}{2}\right)\left(\frac{1+\epsilon}{2}\right)^{k-1}.$$

For comparison, let β_k denote the approximation factor of a random assignment, where $\beta_k = 1 - \frac{1}{2^k}$. Unlike subset advice, the direct application of label advice does not immediately improve approximation performance. On the one hand, for $k = 2$, following the advice improves upon random assignment ($\alpha_2 > \beta_2$), but does not surpass the current best approximation ratio of 0.94 unless $\epsilon \geq 0.872$. Moreover, for $k \geq 3$, unless ϵ is sufficiently large, following the label advice does not even outperform random assignment. For example, $\alpha_3 < \beta_3$ unless $\epsilon \geq 0.618$.

This motivates more refined methods for incorporating label advice, and in this section we focus exclusively on the MAX-2-SAT problem. We take inspiration from the prior work of (Ghoshal et al., 2025) on the closely related MAX-CUT and MAX-2-LIN problems, which can be viewed as special cases of MAX-2-SAT via reduction. We modify the algorithm of (Ghoshal et al., 2025) and adapt their analysis to achieve similar results for MAX-2-SAT.

Given a MAX-2-SAT formula ϕ with m clauses and n variables, we consider the $2n$ literals corresponding to the variables, i.e., pad the variables with $x_{n+1}, \ldots, x_{2n}$, and replace $\overline{x_i}$ with x_{n+i} in ϕ. Following the convention of (Zwick, 2000) and (Lewin et al., 2002), we define a vector $y \in \{-1, 1\}^{2n+1}$ with respect to an assignment on literals $x \in \{0, 1\}^{2n}$ in

the following way: fix $y_0 = 1$ representing "false", and for $i = 1, \ldots, 2n$: $y_i = 1$ if $x_i = 0$, $y_i = -1$ if $x_i = 1$.

Define the adjacency matrix on the $2n$ literals with an additional row and column of 0 at index 0 to match the dimension of y, i.e., $A \in \mathbb{R}^{(2n+1)\times(2n+1)}$ where

$$A_{ij} = \begin{cases} 1 & \text{if } (x_i \vee x_j) \in \phi, \\ 0 & \text{otherwise.} \end{cases}$$

Given y and A as defined above, the number of satisfied clauses equals to the integer quadratic form formulated by (Goemans and Williamson, 1994):

$$\begin{aligned} \#\text{SAT}(y) &= \sum_{(i,j)\in\phi} \frac{3 - y_0 y_i - y_0 y_j - y_i y_j}{4} \\ &= \frac{3}{4}m - \frac{1}{4}\sum_{(i,j)\in\phi}(y_0 y_i + y_0 y_j) - \frac{1}{4}\sum_{(i,j)\in\phi} y_i y_j \\ &= \frac{3}{4}m - \frac{1}{4}\sum_{i\in[2n]} y_0 y_i d_i - \frac{1}{8}\langle y, Ay\rangle \quad \text{(where } d_i \text{ is the degree of literal } i\text{)} \\ &= \frac{3}{4}m - \frac{1}{8}f(y), \end{aligned}$$

where $f(y) := 2\sum_{i\in[2n]} y_0 y_i d_i + \langle y, Ay\rangle$. Note that d_i's are constants given ϕ.

In Algorithm 2.2, we modify the objective of the quadratic program from (Ghoshal et al., 2025) with the quadratic form for MAX-2-SAT and the result is stated in Theorem

2.3.6. The proof follows the analysis of (Ghoshal et al., 2025), and we include our modified proof for completeness.

Algorithm 2.2: MAX-2-SAT with label advice (Ghoshal et al., 2025)

1: **Input**: (i) Adjacency matrix $A \in \mathbb{R}^{(2n+1)\times(2n+1)}$, (ii) advice vector $\widetilde{y} \in \{-1,1\}^{2n+1}$ based on the advice $\widetilde{x} \in \{0,1\}^n$.
2: **Output**: Solution $\widehat{x} \in \{0,1\}^n$.
3: Solve the quadratic program:

$$\min\ F(y,\widetilde{y}) = 2\sum_{i\in[2n]} y_0 y_i d_i + \langle y, A\widetilde{y}/\epsilon\rangle + \|A(y-\widetilde{y}/\epsilon)\|_1$$
$$\text{subject to: } y_0 = 1,\ y_i \in [-1,1],\ y_i = -y_{i+n}.$$

4: Round the real-valued solution y coordinate-by-coordinate to integer-valued solution $\widehat{y} \in \{-1,1\}^{2n+1}$ such that $f(\widehat{y}) \le f(y)$.
5: **Return** $\widehat{x}$ where $\widehat{x}_i = -\frac{\widehat{y}_i - 1}{2}$ for $i = 1,\dots,n$.

Theorem 2.3.6 (MAX-2-SAT with label advice)**.** *For an unweighted MAX-2-SAT instance, suppose we are given label advice $\widetilde{x}$ with correct probability $\frac{1+\epsilon}{2}$ and the instance has average degree $\Delta \ge \Omega\left(\frac{1}{\epsilon^2}\right)$, then Algorithm 2.2 finds solution $\widehat{x}$ in polynomial time such that at least $\mathrm{OPT}\cdot\left(1 - O\left(\frac{1}{\epsilon\sqrt{\Delta}}\right)\right)$ clauses are satisfied in expectation over the randomness of the advice.*

Proof. The proof follows from a chain of inequalities in expectation over the randomness the advice.

1. $f(\widehat{y}) \leq f(y)$, where y is the QP solution, and $\widehat{y}$ is the rounding of y. This follows from the same argument as in (Ghoshal et al., 2025).

2. $f(y) \leq F(y, \widetilde{y})$, where $F(y, \widetilde{y})$ is the minimum value of the QP attained at solution y. This step follows from Lemma 2.3.7, which is an extension of Claim 3.2 from (Ghoshal et al., 2025).

3. $F(y, \widetilde{y}) \leq F(y^*, \widetilde{y})$, where y^* is the vector corresponding to the ground-truth optimal solution x^*. This step directly follows from the optimality of the QP solution y, and the fact that y^* is feasible to the QP.

4. $F(y^*, \widetilde{y}) \leq f(y^*) + m \cdot O(\frac{1}{\epsilon\sqrt{\Delta}})$. This step follows from Lemma 2.3.8, which is an extension of Lemma 3.3 from (Ghoshal et al., 2025).

Putting these together, and we may assume $\text{OPT} \geq \frac{3}{4}m$,

$$
\begin{aligned}
f(\widehat{y}) &\leq f(y^*) + m \cdot O\left(\frac{1}{\epsilon\sqrt{\Delta}}\right), \\
\#\text{SAT}(\widehat{y}) &= \frac{3}{4}m - \frac{1}{8}f(\widehat{y}) \\
&\geq \frac{3}{4}m - \frac{1}{8}f(y^*) - m \cdot O\left(\frac{1}{\epsilon\sqrt{\Delta}}\right) \\
&\geq \text{OPT} - \text{OPT} \cdot O\left(\frac{1}{\epsilon\sqrt{\Delta}}\right) \\
&\geq \text{OPT} \cdot \left(1 - O\left(\frac{1}{\epsilon\sqrt{\Delta}}\right)\right).
\end{aligned}
$$

■

Lemma 2.3.7. *For $y \in [-1,1]^{2n+1}$, $f(y) \le F(y, \widetilde{y})$.*

Proof. This Lemma is a modified version of Claim 3.2 (Ghoshal et al., 2025), we include a full proof for completeness.

$$\begin{aligned} f(y) &= 2\sum_{i\in[2n]} y_0 y_i d_i + \langle y, Ay\rangle \\ &= 2\sum_{i\in[2n]} y_0 y_i d_i + \langle y, A\widetilde{y}/\epsilon\rangle + \langle y, A(y - \widetilde{y}/\epsilon)\rangle \\ &\le 2\sum_{i\in[2n]} y_0 y_i d_i + \langle y, A\widetilde{y}/\epsilon\rangle + \|y\|_\infty \cdot \|A(y - \widetilde{y}/\epsilon)\|_1 \\ &= 2\sum_{i\in[2n]} y_0 y_i d_i + \langle y, A\widetilde{y}/\epsilon\rangle + \|A(y - \widetilde{y}/\epsilon)\|_1 \\ &= F(y, \widetilde{y}), \end{aligned}$$

where the inequality follows from Hölder's inequality. ■

Lemma 2.3.8. $\mathbb{E}_{\widetilde{y}}[F(y^*, \widetilde{y})] \le f(y^*) + \frac{2}{\epsilon}\sqrt{mn}$.

Note that $\frac{2}{\epsilon}\sqrt{mn} = m \cdot O(\frac{1}{\epsilon\sqrt{\Delta}})$, following the definition of $\Delta = \frac{m}{n}$.

Proof. Let $z = y^* - \widetilde{y}/\epsilon$. First we calculate the mean and variance of $\widetilde{y}_i$ and z_i.

$$\begin{aligned} &\mathbb{E}[\widetilde{y}_i] = \frac{1+\epsilon}{2} y_i^* + \frac{1-\epsilon}{2}(-y_i^*) = \epsilon y_i^*, \\ &\mathbb{E}[z_i] = 0, \\ &\mathbb{E}\big[z_i^2\big] = \frac{1+\epsilon}{2}(y_i^* - y_i^*/\epsilon)^2 + \frac{1-\epsilon}{2}(y_i^* + y_i^*/\epsilon)^2 = \frac{1-\epsilon^2}{\epsilon^2}. \end{aligned}$$

Consider

$$\begin{aligned}
\mathbb{E}_{\widetilde{y}}[F(y^*, \widetilde{y})] &= 2\sum_{i\in[2n]} y_0^* y_i^* d_i + \mathbb{E}_{\widetilde{y}}[\langle y^*, A\widetilde{y}/\epsilon\rangle] + \mathbb{E}_{\widetilde{y}}[\|Az\|_1] \\
&= 2\sum_{i\in[2n]} y_0^* y_i^* d_i + \langle y^*, A\mathbb{E}_{\widetilde{y}}[\widetilde{y}/\epsilon]\rangle + \mathbb{E}_{\widetilde{y}}[\|Az\|_1] \\
&= 2\sum_{i\in[2n]} y_0^* y_i^* d_i + \langle y^*, Ay^*\rangle + \mathbb{E}_{\widetilde{y}}[\|Az\|_1] \\
&= f(y^*) + \mathbb{E}_{\widetilde{y}}[\|Az\|_1]
\end{aligned}$$

According to the proof of Lemma 3.3 (Ghoshal et al., 2025), the expectation term

$$\begin{aligned}
\mathbb{E}_{\widetilde{y}}[\|Az\|_1] &\le \sqrt{2n}\ \mathbb{E}_{\widetilde{y}}[\|Az\|_2] \\
&\le \sqrt{2n}\sqrt{\mathbb{E}\left[z_i^2\right]\cdot\|A\|_F^2} \\
&\le \sqrt{2n}\sqrt{\frac{1-\epsilon^2}{\epsilon^2}}\|A\|_F \\
&\le \frac{2}{\epsilon}\sqrt{mn},
\end{aligned}$$

since $\|A\|_F^2 = 2m$. ■

We formally define the MAX-2-LIN problem and show a folklore reduction to MAX-2-SAT.

Definition 2.3.9 (MAX-2-LIN)**.** *In the (unweighted) MAX-2-LIN problem, we are given a set of binary variables $\{z_i\}_{i=1}^n \in \{\pm 1\}^n$ and m constraints of the form $z_i \cdot z_j = c_{ij}$ where*

$c_{ij} \in \{\pm 1\}$. *The goal is to find an assignment* $\widehat{z} \in \{\pm 1\}^n$ *that maximize the total number of satisfied constraints.*

Given a MAX-2-LIN instance on n variables and m constraints, we can reduce it to MAX-2-SAT on n variables and $M = 2m$ clauses in the following way:

- For MAX-2-LIN constraint $z_i \cdot z_j = +1$:

  add 2 clauses in MAX-2-SAT: $(x_i \vee \overline{x_j}) \wedge (\overline{x_i} \vee x_j)$;

- For MAX-2-LIN constraint $z_i \cdot z_j = -1$:

  add 2 clauses in MAX-2-SAT: $(x_i \vee x_j) \wedge (\overline{x_i} \vee \overline{x_j})$.

This reduction preserves approximation as stated in Proposition 2.3.10 below, and given label advice, our Theorem 2.3.6 is a generalization of Theorem 1.4 from (Ghoshal et al., 2025).

Proposition 2.3.10. *Under the reduction above, a* $(1 - O(\delta))$*-approximation to MAX-2-SAT corresponds to a* $(1 - O(\delta))$*-approximation to MAX-2-LIN.*

Proof. Given a solution x to the MAX-2-SAT problem, we can translate it to a solution z to the MAX-2-LIN problem by setting $z_i = 1$ if $x_i = 1$, and $z_i = -1$ if $x_i = 0$.

Notice that in the reduction, for each of the constraint and its 2 corresponding clauses, the constraint is satisfied if and only if both clauses are true, and the constraint is not satisfied if and only if exactly one of the clauses is true. Therefore, $\#\text{LIN} = \#\text{SAT} - m$. Denote the value of an optimal solution to MAX-2-LIN as m^*, and the value of an optimal

solution to MAX-2-SAT as M^*, and notice that $m^* = M^* - m$. Given an assignment x that satisfies $(1 - O(\delta)) \cdot M^*$ clauses in MAX-2-SAT , the number of constraints in MAX-2-LIN satisfied by corresponding z is

$$\begin{aligned}
\#\text{LIN} &= \#\text{SAT} - m \\
&= (1 - O(\delta)) \cdot M^* - m \\
&= (1 - O(\delta)) \cdot (m^* + m) - m \\
&= (1 - O(\delta)) \cdot m^* - O(\delta) \cdot m \\
&= (1 - O(\delta)) \cdot m^*, \text{ since we may assume } m^* = \Theta(m).
\end{aligned}$$

Given label advice, our Theorem 2.3.6 is a generalization of Theorem 1.4 from (Ghoshal et al., 2025). By mapping $\{-1, 1\}$ to $\{0, 1\}$, an advice to the MAX-2-LIN instance can be translated as an advice to the MAX-2-SAT instance with the same ϵ. Furthermore, the degree of the MAX-2-LIN instance is $\Delta = \frac{2m}{n}$ which is equal to the degree of the MAX-2-SAT instance $\frac{M}{n}$, so the same average degree assumption applies to both problems as well. Therefore we recover the solution to MAX-2-LIN and generalize the previous results from the "symmetric" constraint satisfaction problems to "non-symmetric" SAT problems. ■

2.4 Discussion

In this work, we showed how subset advice can be incorporated to improve the running time of algorithms for k-SAT that currently achieve the best known performance in worst-case analysis. For the optimization problem MAX-SAT and its variants, we incorporated subset advice into any algorithm and showed that the approximation factor improves linearly with the advice parameter. We also proved that, assuming ETH and Gap-ETH, these are the best possible results for 3-SAT and MAX-3-SAT. Using label advice, we obtained near-optimal results for MAX-2-SAT instances where the average degree exceeds a threshold depending only on the amount of advice. This generalizes previous results for MAX-CUT and MAX-2-LIN. Open questions regarding the label advice include designing algorithms for k-SAT, as well as incorporating label advice into SDP-based methods to solve more general problems of MAX-k-SAT. An interesting direction for future work is to explore and compare different advice models, such as proving a formal separation between the label and subset advice models as it is plausible that the label advice model constitutes a weaker model. We would also like to explore a variation on the label advice model, where we are allowed to make a few queries to an oracle with non-persistent noise.

Acknowledgments

We thank Suprovat Ghoshal for helpful discussions. This research was supported in part by the National Science Foundation grant ECCS-2217023.

CHAPTER 3

NON-CENTER-BASED CLUSTERING UNDER BILU-LINIAL STABILITY

This chapter was previously published as Non-Center-Based Clustering Under Bilu-Linial Stability by Xing Gao and Lev Reyzin (Gao and Reyzin, 2025).

3.1 Introduction

In this paper, we give the first results on **minimizing sum-of-diameters (MSD)** and also **minimizing sum-of-radii (MSR)** clustering under a stability assumption first introduced by Bilu and Linial (Bilu and Linial, 2012) that is motivated by the observation that many real-world NP-hard problems can be solved efficiently in practice. Informally, **Bilu-Linial stability** assumes the optimal solution for a problem of interest does not change under small perturbation of the input.

In particular, we give structural properties that show that single-linkage and complete-linkage algorithms give exact solutions to 2-stable sum-of-diameters (MSD) instances, and we show that instances that are strictly less than 2-stable are NP-hard under randomized reductions. For the closely related problem of sum-of-radii clustering (MSR), we also present some structural properties that allow the single-linkage algorithm to solve 2-stable instances and the complete-linkage algorithm to solve 3-stable instances.

Many problems have been studied under Bilu-Linial stability, including MAX-CUT (Bilu and Linial, 2012; Makarychev et al., 2014), max independent set (Angelidakis et al., 2018), and center-based clustering such as k-means, k-median (Awasthi et al., 2012; Balcan and Liang, 2016; Angelidakis et al., 2017), k-center (Balcan et al., 2020) and min-sum (Ben-David and Reyzin, 2014). Other metric based problems include the traveling salesman problem (Mihalák et al., 2011) and the Steiner tree problem (Freitag et al., 2021). These works are also closely related to robust algorithms (Makarychev et al., 2014) and certified algorithms (Makarychev and Makarychev, 2020), as well as to an interesting connection between stability and independent systems/matroids (Chatziafratis et al., 2017). Despite extensive research on center-based clustering, the MSD and MSR problems, which possess distinct, non-center-based structures, have yet to be analyzed under Bilu-Lineal stability.

The MSD and MSR problems are closely related and an exact solution to one is a 2-approximation to the other. Under a general metric, MSD and MSR are both known to be NP-hard (Doddi et al., 2000; Gibson et al., 2010). There are various approximation algorithms for these problems (see e.g. (Charikar and Panigrahy, 2001)), as well as exact algorithms studied under different metrics (Behsaz and Salavatipour, 2015; Hansen and Jaumard, 1987; Capoyleas et al., 1991; Gibson et al., 2012).

3.2 Preliminaries

Given a clustering instance (P, d) where P is a set of n points and $d(\cdot, \cdot)$ is a metric on P, we study the problem of dividing the points into k clusters $\{C_1, \ldots, C_k\}$ under a

non-center-based objective, namely the MSD objective, where the goal is to minimize the sum of diameters of all the clusters. The diameter of a cluster C is

$$\rho(C) := \max_{(x,y)\in C} d(x,y).$$

A closely related objective that minimizes the sum of radii is known as MSR, and the radius is

$$r(C) := \min_{c\in C} \max_{p\in C} d(c,p).$$

Notice that a solution to MSR is a 2-approximation to MSD and vice versa, because for each cluster we have $r \leq \rho \leq 2r$, and

$$\sum_{i=1}^{i=k} r_i^* \leq \sum_{j=1}^{j=k} \rho_j, \ \sum_{i=1}^{i=k} \rho_i^* \leq \sum_{j=1}^{j=k} 2r_j,$$

where r_i^*, ρ_i^* correspond to the radii and diameters of the optimal MSR or MSD solution, and r_j, ρ_j correspond to any feasible solution.

We use $\text{dist}(C_1, C_2)$ to represent the distance between two clusters, which is the distance between the closest pair of points from each cluster, i.e.,

$$\text{dist}(C_1, C_2) := \min_{a\in C_1, b\in C_2} d(a,b).$$

We denote the optimal clustering as $\text{OPT} := \{C_1^*, \ldots, C_k^*\}$ and its value as cost(OPT).

We focus on the MSD problem under the notion of stability first introduced by Bilu and Linial (Bilu and Linial, 2012), which is usually referred to as "perturbation resilience" in the context of clustering (Awasthi et al., 2012).

Definition 3.2.1 (γ-Perturbation)**.** *Given a clustering instance (P, d), we say a function $d' : P \times P \to [0, \infty)$ is a γ-perturbation of (P, d) if $\forall x, y \in P$, we have $d(x, y) \leq d'(x, y) \leq \gamma \cdot d(x, y)$. Note that d' may not be a metric.*

Definition 3.2.2 (Perturbation Resilience)**.** *For $\gamma > 1$, we say a clustering instance (P, d) is γ-perturbation-resilient if for any γ-perturbation d', the unique optimal clustering $\{C_1^*, \ldots, C_k^*\}$ of (P, d) stays the same under d', i.e., $\mathrm{OPT} = \mathrm{OPT}'$ where OPT' is the optimal solution of the perturbed instance.*

3.3 Algorithm for MSD Under Stability

In this section we first present some properties of MSD under stability assumptions, then we use these properties to show that the single-linkage and complete-linkage algorithms combined with dynamic programming finds the optimal clustering of 2-stable instances.

3.3.1 Properties Following Stability

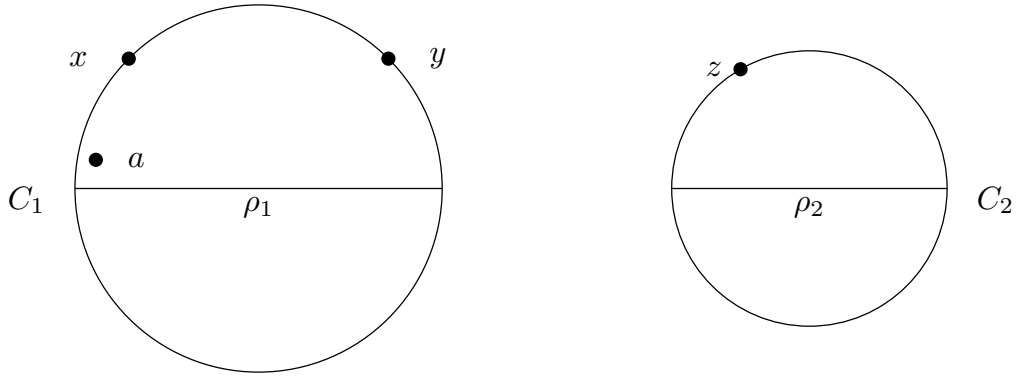

Figure 1: Properties of stable MSD instances.

Lemma 3.3.1 (MSD properties from stability)**.** *Given a γ-stable MSD clustering instance, suppose C_1 and C_2 are clusters in* OPT *with diameters ρ_1 and ρ_2 respectively, then we have the following:*

1. $\forall z \notin C_1, \exists a \in C_1$ *s.t.* $d(a, z) > \gamma \cdot \rho_1$.
2. $\forall x, y \in C_1, \forall z \notin C_1, (\gamma - 1) \cdot d(x, y) < d(y, z)$.

   In particular, if $\gamma \geq 2$, $d(x, y) < d(y, z)$.
3. $(\gamma - 1) \cdot \rho_1 < \text{dist}(C_1, C_2)$.

   In particular, if $\gamma \geq 2$, $\rho_1 < \text{dist}(C_1, C_2)$.

Proof.

1. Suppose not, then under the perturbation where all pair-wise distances in C_1 are perturbed by γ, z can be moved to C_1 in OPT$'$ without increasing the cost so that OPT$' \neq$ OPT, contradicting the stability assumption.

2. Suppose $\exists x, y \in C_1$ and $z \in C_2$ s.t. $(\gamma - 1) \cdot d(x, y) \geq d(y, z)$, which means $d(y, z) \leq (\gamma - 1) \cdot \rho_1$. $\forall a \in C_1$, we have $d(a, y) \leq \rho_1$, therefore $d(a, z) \leq d(a, y) + d(y, z) \leq \gamma \cdot \rho_1$, contradicting property 1.

3. Suppose not, then $\exists y \in C_1$ and $z \in C_2$ s.t. $d(y, z) \leq (\gamma - 1) \cdot \rho_1$. Again, $\forall a \in C_1$ we have $d(a, y) \leq \rho_1$, therefore $d(a, z) \leq d(a, y) + d(y, z) \leq \gamma \cdot \rho_1$, contradicting property 1. ■

3.3.2 Algorithms for 2-Stable MSD Instances

Algorithm 3.1: Single-linkage for MSD (Johnson, 1967)

1: $\mathcal{C} = \{\{p\} \mid p \in P\}$, start with all singletons;

2: **while** $|\mathcal{C}| > k$ **do**

3: Merge $\underset{C_i, C_j}{\operatorname{argmin}}\ \operatorname{dist}(C_i, C_j)$;

4: **end while**

Algorithm 3.2: Complete-linkage for MSD (Johnson, 1967)

1: $\mathcal{C} = \{\{p\} \mid p \in P\}$, start with all singletons;

2: **while** $|\mathcal{C}| > k$ **do**

3: Merge $\underset{C_i, C_j}{\operatorname{argmin}}\ \rho(C_i \cup C_j)$;

4: **end while**

The single-linkage and complete-linkage algorithms (Johnson, 1967) are popular heuristics for clustering, and they both belong to the family of agglomerative hierarchical clustering algorithms (Schütze et al., 2008). In this section we show that for stable MSD instances with $\gamma \geq 2$, these simple heuristics produce a tree structure (a.k.a. dendrogram) where the optimal clustering is a pruning of the tree, and we terminate when there are k clusters remaining. In contrast, for stable instances of center-based-clustering such as k-means and k-median, the cost of a cluster depends on the number of points in it as well as their distances, so the algorithm needs to run until only one cluster remains, then the optimal k clusters can be found by dynamic programming (Cf. (Angelidakis et al., 2017) Section 4.2 and (Awasthi et al., 2012) Section 2.3.)

Theorem 3.3.2 (Algorithms for MSD)**.** *The single-linkage algorithm (Algorithm 3.1) and complete-linkage algorithm (Algorithm 3.2) give exact solutions to MSD instances assuming stability* $\gamma \geq 2$.

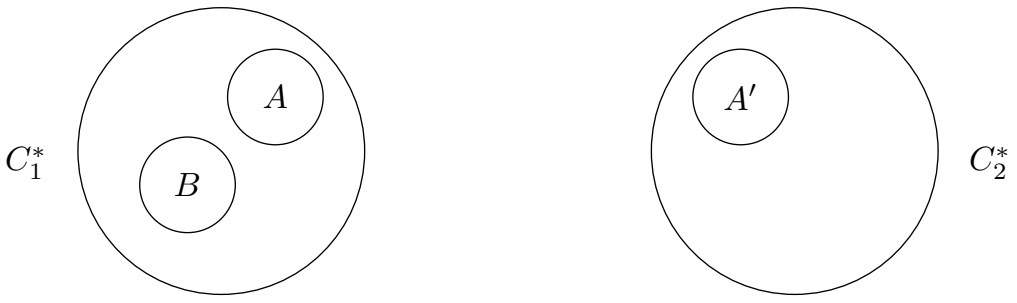

Figure 2: Merge clusters during Algorithm 3.1 and 3.2.

Proof. We show by induction that in both algorithms the clusters after each merge are laminar to OPT, i.e., inside each remaining cluster, all points belong to the same cluster in OPT. This technique is inspired by the analysis in (Balcan and Liang, 2016) for k-median clustering instances.

Base case: singleton clusters are laminar to OPT.

Induction step of merging: consider the clusters formed during the algorithm and a merge step (see Figure 2). Suppose $A \subset C_1^*$ where $\rho(C_1^*) = \rho_1^*$, we know that $\exists B \subset C_1^* \setminus A$ *s.t.* $\text{dist}(A, B) \leq \rho(A \cup B) \leq \rho_1^*$. Let $A' \not\subset C_1^*$, by the induction hypothesis A' is fully contained in some cluster in OPT so without loss of generality we may assume $A' \subset C_2^*$, and $\rho(A \cup A') \geq \text{dist}(A, A') \geq \text{dist}(C_1^*, C_2^*) > \rho_1^*$ (by property 3). This means for single-linkage we have $\text{dist}(A, B) < \text{dist}(A, A')$, and for complete-linkage we have $\rho(A \cup B) < \rho(A \cup A')$, therefore the argmin pair of clusters chosen by the algorithms must belong to the same cluster in OPT, and all the clusters remain laminar to OPT after the merge. ■

3.4 A Matching Lower Bound for MSD

3.4.1 Non-Approximability of MSD Clustering

The following theorem from (Doddi et al., 2000) states the non-approximability result for the MSD problem without any stability assumptions. We restate the theorem and the reduction setup here, and we will use the same reduction to show the NP-hardness result for MSD instances with $2 - \epsilon$ stability.

Theorem 3.4.1 (Prop. 2 (Doddi et al., 2000))**.** *Unless* P = NP*, for any* $\epsilon > 0$*, no polynomial time algorithm for the problem can provide a solution which satisfies the bound on the number of clusters and whose total diameter is within a factor* $2 - \epsilon$ *of the optimal value.*

The result was shown using reduction from the clique problem. Given a clique problem to determine whether there exists a clique of size J in the graph $G = (V, E)$, we can reduce it to a MSD problem using the 2-1-metric: set $P = V$, and $d(u, v) = 1$ if $(u, v) \in E$, otherwise $d(u, v) = 2$. The number of clusters is set to $k = n + 1 - J$. If there exists a clique of size J, $\text{cost}(\text{OPT}_{\text{MSD}}) = 1$ consisting of 1 cluster of diameter 1 containing all the vertices in the clique, and $n - J$ singleton clusters with diameter 0 for each of the remaining vertex; otherwise $\text{cost}(\text{OPT}_{\text{MSD}}) \geq 2$.

3.4.2 Hardness Under Stability Assumptions

In this section, we provide a matching lower-bound of $2 - \epsilon$ on the stability parameter. The result is formally stated in Theorem 3.4.2.

Theorem 3.4.2. *Unless* P = NP = RP*, no polynomial time algorithm can solve a* $(2 - \epsilon)$*-stable instance of the sum-of-diameters clustering problem for any* $\epsilon > 0$*.*

Notice that the reduction used in Theorem 3.4.1 produces a $(2 - \epsilon)$-stable clustering instance if there exists a unique clique of size J in the clique problem. In other words, solving $(2 - \epsilon)$-stable MSD instances is at least as hard as the Clique Promise Problem, which is a variation on the Clique problem where it is promised that there exists a unique

optimal solution. We show the hardness of the Clique Promise Problem in Theorem 3.4.3, and then Theorem 3.4.2 follows.

Theorem 3.4.3 (Clique Promise Problem)**.** *The Clique Promise Problem (CPP), where the instance is promised to have a unique largest clique, is NP-hard under randomized reduction.*

Theorem 3.4.3 follows by combining two existing results. Lemma 3.4.5 states that SAT is parsimoniously reducible to the Clique problem, so we can apply Lemma 3.4.4 and choose A to be the Clique problem, which proves Theorem 3.4.3.

Lemma 3.4.4 (USAT Corollary 3.4 (Valiant and Vazirani, 1985))**.** *Let A be any NP-complete problem to which satisifability is parsimoniously reducible. The following "promise problem" is NP-hard under randomized reduction:*
Input: an instance x of A; Output: a solution to x; Promise: $\#A(x) = 1$.

Lemma 3.4.5 (#Clique is #P-complete (Fortnow and Gasarch, 2023))**.** *There is a parsimonious reduction from SAT to Clique.*

Here we include a modified version of the proof from (Fortnow and Gasarch, 2023) for completeness.

Proof. Step 1: $\#\text{SAT} \leq_{\text{p}} \#\text{3SAT}$.

Consider a SAT instance f, we will reduce it to a 3SAT formula f' where there is a one-to-

one correspondence between any satisfiable assignment to f and f'. First introduce new variables a, b, c and new clauses

$$\overline{(a \vee b \vee c)} \iff (\overline{a} \vee b \vee c) \wedge (a \vee \overline{b} \vee c) \wedge (a \vee b \vee \overline{c})$$
$$\wedge\ (\overline{a} \vee \overline{b} \vee c) \wedge (\overline{a} \vee b \vee \overline{c}) \wedge (a \vee \overline{b} \vee \overline{c}) \wedge (\overline{a} \vee \overline{b} \vee \overline{c})\ ,$$

so that f' is satisfiable if and only if a, b, c are all set to 0.

1. For clauses with 1 literal x_1, replace it with $(x_1 \vee a \vee b) \iff x_1$;
2. For clauses with 2 literals x_1, x_2, replace it with $(x_1 \vee x_2 \vee a) \iff (x_1 \vee x_2)$;
3. For clauses with 3 literals, do nothing;
4. For clauses with ≥ 4 literals $(x_1 \vee x_2 \vee y)$, where y is a disjunction of ≥ 2 literals, repeatedly reduce the number of literals by one by replacing the clause with

$$C = (x_1 \vee x_2 \vee w) \wedge (\overline{x_1} \vee x_2 \vee \overline{w})$$
$$\wedge\ (x_1 \vee \overline{x_2} \vee \overline{w}) \wedge (\overline{x_1} \vee \overline{x_2} \vee \overline{w}) \wedge (\overline{w} \vee y)\ .$$

Consider any satisfiable assignment to f,

- if $\overline{x_1 \vee x_2}$, i.e. $x_1 = 0,\ x_2 = 0,\ y = 1$, and $C \iff w \wedge (\overline{w} \vee y)$, so $w = 1$ in any satisfiable assignment to f' ;
- if $x_1 \vee x_2$, $C \iff \overline{w} \wedge (\overline{w} \vee y)$, so $w = 0$ in any satisfiable assignment to f' .

Step 2: #3SAT $\leq_{\mathrm{p}}$ #Clique.

Consider #3 SAT instance $f = C_1 \wedge \ldots \wedge C_k$. Construct a graph G:

- Vertices: for each clause C_i introduce 7 vertices corresponding to the 7 assignments that satisfy C ;
- Edges: an edge exists between 2 vertices if and only if the assignments represented by the vertices do not contradict each other. In particular, there are no edges among vertices from the same clause.

There is a one-to-one correspondence between a satisfiable assignment to f and a clique of size k in G. ■

It remains an open question to prove a similar lower bound for the MSR objective.

3.5 Algorithm for MSR Under Stability

In this section we show that the MSR objective is also polynomial-time solvable at 2-stability or higher. We begin with showing some properties of MSR following from stability assumptions.

3.5.1 Properties Following Stability

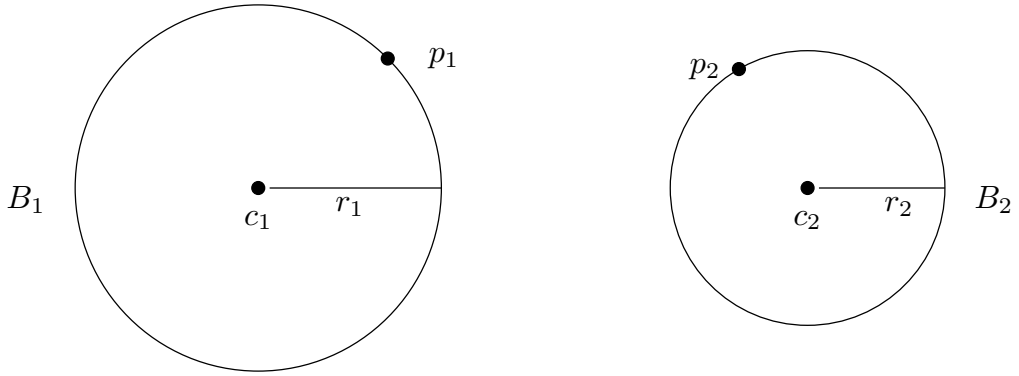

Figure 3: Properties of stable MSR instances.

Lemma 3.5.1 (MSR properties from stability)**.** *Given a γ-stable MSR clustering instance, suppose B_1 and B_2 are clusters in* OPT *centered at c_1, c_2 with radii r_1 and r_2 respectively, then we have the following:*

1. $\forall p_2 \notin B_1$, $d(c_1, p_2) > \gamma \cdot r_1$.
2. $d(c_1, c_2) > \frac{\gamma}{2}(r_1 + r_2)$.

   In particular, if $\gamma > 2$, $d(c_1, c_2) > r_1 + r_2$, i.e., clusters are separated.
3. *If $\gamma \geq 2$, each point belongs to its closest center, i.e., $\forall p_1 \in B_1$, $d(p_1, c_1) < d(p_1, c_2)$ $\forall c_2$ that is a center of another cluster.*

4. $(\gamma - 1) \cdot r_1 < \text{dist}(B_1, B_2)$.

   $(\gamma - 1) \cdot d(p_1, c_1) < d(p_1, p_2)\ \forall p_1 \in B_1,\ p_2 \in B_2$.

   In particular, if $\gamma \geq 2$, $r_1 < \text{dist}(B_1, B_2)$ *and* $d(p_1, c_1) < d(p_1, p_2)$.

   If $\gamma \geq 3$, $\rho(B_1) \leq 2r_1 < \text{dist}(B_1, B_2) \leq \rho(B_1 \cup B_2)$.

5. *Notably we don't have "center proximity", a property implied by perturbation resilience used in (Awasthi et al., 2012) instead of perturbation resilience, i.e., it's possible that* $\gamma \cdot d(p_1, c_1) > d(p_1, c_2)$.

Proof.

1. Suppose not, and consider the perturbation where $\forall p_1 \in B_1, d(c_1, p_1)$ is perturbed by γ, then we can move p_2 to B_1 in OPT$'$ without increasing the cost so that OPT$' \neq$ OPT, contradicting the stability assumption.

2. Following property 1, $d(c_1, c_2) > \gamma \cdot r_1$ and $d(c_1, c_2) > \gamma \cdot r_2$, combined we have $d(c_1, c_2) > \frac{\gamma}{2}(r_1 + r_2)$.

3. Suppose there exists another cluster's center c_2 s.t. $d(p_1, c_2) \leq d(p_1, c_1)$, then $d(c_1, c_2) \leq d(p_1, c_1) + d(p_1, c_2) \leq 2r_1 \leq \gamma \cdot r_1$, contradicting property 1.

4. Suppose $\exists p_1 \in B_1, p_2 \in B_2$ s.t. $d(p_1, p_2) \leq (\gamma - 1) \cdot r_1$, therefore $d(c_1, p_2) \leq d(c_1, p_1) + d(p_1, p_2) \leq \gamma \cdot r_1$, contradicting property 1.

   Suppose $\exists p_1 \in B_1, p_2 \in B_2$ s.t. $d(p_1, p_2) \leq (\gamma - 1) \cdot d(p_1, c_1) \leq (\gamma - 1) \cdot r_1$, therefore $d(c_1, p_2) \leq d(c_1, p_1) + d(p_1, p_2) \leq \gamma \cdot r_1$, contradicting property 1.

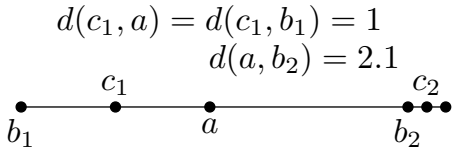

Figure 4: A 3-stable MSR instance without the center proximity property.

5. In Figure 4, we show a counter example where $\gamma \cdot d(p_1, c_1) > d(p_1, c_2)$ with $\gamma = 3$ and the number of clusters $k = 2$.

   $\text{OPT} = d(a, c_1) + d(b_2, c_2) = 1 + \epsilon$. Perturb $d(a, c_1) \to 3$, then $\text{OPT} \to 3 + \epsilon$.

   Consider an alternative solution OPT': move a to c_2, $\text{OPT}' = d(b_1, c_1) + d(a, c_2) = 1 + 2.1 + \epsilon$, so the example is 3 stable, but $3 = 3d(a, c_1) > d(a, c_2) = 2.1 + \epsilon$, violating center proximity. ■

3.5.2 Algorithms for Stable MSR Instances

Now we are ready to analyze Algorithms 3.1 and 3.2 for the MSR objective.

Theorem 3.5.2 (Algorithms for MSR)**.** *The single-linkage algorithm (Algorithm 3.1) gives exact solution to MSR if $\gamma \geq 2$ and the complete-linkage algorithm (Algorithm 3.2) gives exact solution if $\gamma \geq 3$.*

Proof. We show that in both algorithms the clusters after each merge are laminar to OPT by induction.

Single-linkage: Assume $\gamma \geq 2$ and we have $r_1^* < \text{dist}(C_1^*, C_2^*)$ by property 4.

Base case: singleton clusters are laminar to OPT.

Induction step of merging: suppose $A \subset C_1^*$, we know $\exists B \subset C_1^* \setminus A$ *s.t.* $\text{dist}(A, B) \leq r_1^*$ (let either A or B contain the center c_i). Let $A' \not\subset C_1^*$, by induction A' is fully contained in some cluster in OPT so w.o.l.g. we may assume $A' \subset C_2^*$ and $\text{dist}(A, A') \geq \text{dist}(C_1^*, C_2^*) > r_1^*$. This means $\text{dist}(A, B) < \text{dist}(A, A')$, therefore the argmin pair of clusters chosen by the algorithm must belong to the same cluster in OPT, and all the clusters remain laminar to OPT after the merge.

Complete-linkage: Assume $\gamma \geq 3$ and we have $\rho(C_1^*) < \text{dist}(C_1^*, C_2^*)$ by property 4.

Base case: singleton clusters are laminar to OPT.

Induction step of merging: suppose $A \subset C_1^*$, we know $\exists B \subset C_1^* \setminus A$ *s.t.* $\rho(A \cup B) \leq \rho(C_1^*)$. Let $A' \not\subset C_1^*$, by induction A' is fully contained in some cluster in OPT so w.o.l.g. we may assume $A' \subset C_2^*$ and $\rho(A \cup A') \geq \text{dist}(A, A') \geq \text{dist}(C_1^*, C_2^*) > \rho(C_1^*)$. This means $\rho(A \cup B) < \rho(A \cup A')$, therefore the argmin pair of clusters chosen by the algorithm must belong to the same cluster in OPT, and all the clusters remain laminar to OPT after the merge. ■

Acknowledgment

This work was supported in part by the grants ECCS-2217023 and CCF-2307106 from the National Science Foundation.

CHAPTER 4

SEMI-RANDOM NOISY AND ONE-BIT MATRIX COMPLETION WITH NONCONVEX PRIMAL-DUAL FRAMEWORK

4.1 Introduction

We study the problem of recovering a low-rank matrix X^* from incomplete and noisy observations Y, where X^* has rank at most r and Y is supported on a subset of indices Ω. Matrix completion has been well-studied under the uniform observation model, where each entry of X^* is revealed independently with some fixed probability p, known as the sampling rate.

However, the standard assumption that the entries are observed uniformly at random may fail to capture more realistic scenarios in which the observation pattern may exhibit mild or even adversarial non-uniformity. Take collaborative filtering as an example: observations might be heavily concentrated on a few blocks of the matrix, as some groups of users might be more active to provide feedback, or some products are more popular.

In this paper, we focus on a model mis-specification known as the *semi-random model* for matrix recovery, introduced by (Cheng and Ge, 2018), where the observation probabilities are unknown (and not uniform in general) but lower-bounded by some p. Alternatively, the semi-random model can be viewed as a two-stage process: first, each entry is observed

with probability p (same as in the uniform observation model), then additional entries from X^* may be revealed adversarially.

Most existing results on matrix completion under semi-random observations focus on real-valued or noiseless observations. In practice, however, observations are often noisy, quantized, or binary, even though the underlying signal is continuous. For example, user ratings may be restricted to integers between 1 and 5, or in the extreme case, take binary value like an upvote or downvote on a post. This motivates the need to study more general and realistic models, such as noisy matrix completion and one-bit matrix completion, which we formally define below.

Noisy matrix completion. In this setting, each observed entry is subject to additive i.i.d. noise. Specifically, the observed data matrix takes the form:

$$\forall (j,k) \in \Omega : Y_{jk} = X^*_{jk} + E_{jk},$$

where E is a noise matrix in which each entry is i.i.d. sub-exponential[1] with $\mathbb{E}[E_{jk}] = 0$ and $\mathrm{V}ar[E_{jk}] = \frac{\nu^2}{d_1 d_2}$. We consider the least-squares loss function F:

$$F(X) = \frac{1}{2p} \sum_{(j,k)\in\Omega} (X_{jk} - Y_{jk})^2. \tag{4.1}$$

---

[1]Throughout the chapter we restrict to sub-exponential random variables whose sub-exponential norms are on the same order as their standard deviations, which include many classical light-tailed distributions.

One-bit matrix completion. Originally studied by (Davenport et al., 2014), in this setting observations are quantized to binary values in the following way:

$$\forall (j,k) \in \Omega : Y_{jk} = \begin{cases} +1, & X^*_{jk} + E_{jk} \geq 0 \\ \\ -1, & X^*_{jk} + E_{jk} < 0 \end{cases},$$

where E is a noise matrix in which each entry is i.i.d. with mean $\mathbb{E}[E_{jk}] = 0$ and $\mathrm{V}ar[E_{jk}] = \frac{\nu^2}{d_1 d_2}$. Equivalently, we may consider the observations as following some discrete probability distribution parameterized by the original real-valued entries. In this case, let f be the cumulative distribution function (cdf) of $-E_{jk}$, then the observations can be formulated with a probit (probabilistic unit) model:

$$\forall (j,k) \in \Omega,\, Y_{jk} = \begin{cases} +1, & \text{with probability } f(X^*_{jk}) \\ \\ -1, & \text{with probability } 1 - f(X^*_{jk}) \end{cases}.$$

We consider the negative log-likelihood loss function F:

$$F(X) = -\frac{1}{pd_1 d_2} \sum_{(j,k)\in\Omega} \Big[\mathbb{1}(Y_{jk} = 1) \cdot \log\big(f(X_{jk})\big) + \mathbb{1}(Y_{jk} = -1) \cdot \log\big(1 - f(X_{jk})\big) \Big]. \quad (4.2)$$

These two settings have been extensively studied under the uniform observation model, but remain largely unexplored in the semi-random setting [1]. Prior techniques (Cheng and Ge, 2018), which rely on the quadratic structure of the loss, do not apply directly to these more general settings. The noisy and one-bit models introduce new challenges due to their non-quadratic losses and non-linear (often discrete) observations. This leads us to the question we study in this work:

Can we design efficient algorithms for noisy and one-bit matrix completion with provable recovery guarantees under semi-random observations?

To answer this, we consider the following nonconvex optimization problem with loss function $F(X)$, where X is replaced with its Burer-Monteiro factorization (Burer and Monteiro, 2003)

$$\min_{U \in \mathcal{C}_1, V \in \mathcal{C}_2} F(UV^\top), \tag{4.3}$$

where $\mathcal{C}_1 \subset \mathbb{R}^{d_1 \times r}$ and $\mathcal{C}_2 \subset \mathbb{R}^{d_2 \times r}$ guarantee the rank of the product matrix $X = UV^\top$ is at most r. This factorized formulation is preferred in practice, since it avoids the computationally expensive SVD step with convex relaxation approaches as discussed in the related work section.

---

[1] The work of (Kelner et al., 2024) also studies semi-random (noisy) matrix completion, but under a different observation model than the one considered in this paper. See the related work section for further discussion.

Following the nonconvex approach, we adopt the primal-dual framework proposed in (Zhang et al., 2018b), which provides a global landscape analysis via Lagrangian duality. This framework not only characterizes all stationary points of the nonconvex objective, but also supports algorithmic extensions to various loss functions applicable to a broad family of low-rank matrix recovery problems, including the standard least-squares loss for noiseless matrix completion, its noisy variant, and the log-likelihood loss used in one-bit matrix completion. However, their analysis relies on the uniform observation model, where every entry is observed with a fixed probability p, and cannot be directly applied to the semi-random setting. To address the limitations of standard nonconvex methods under semi-random inputs, (Cheng and Ge, 2018) establish a connection to spectral graph theory and design a preprocessing algorithm based on spectral sparsification. This procedure reweights the observed entries to achieve spectral similarity to the uniform observation model and provably eliminates spurious local minima in the case of least-squares loss.

In this work, we unify these two threads to establish recovery guarantees for both noisy and one-bit matrix completion under semi-random model. Specifically, we develop a weighted version of the regularity conditions required in the primal-dual analysis, specialized to the spectrally reweighted observations. We subsequently prove that the approximation error induced by preprocessing does not significantly weaken the final recovery guarantee. This connection allows us to extend the primal-dual paradigm to semi-random models in both the noisy and one-bit observation regimes, which are beyond the scope of either prior work.

4.1.1 Our Contributions

We provide a unified algorithmic framework for a broad class of matrix completion problems, including the noisy and one-bit settings, in the semi-random observation model. Specifically, we show the following result:

Theorem 4.1.1 (Informal, see Theorem 4.3.5)**.** *Let $X^* \in \mathbb{R}^{d\times d}$ be a rank-r, β-incoherent matrix with largest singular value σ_1. Suppose the observation is subject to entry-wise i.i.d. noise $\|E\|_F^2 \le \nu^2$. Assume each entry is observed with probability at least $p = \frac{\mathrm{poly}(r,\log d)}{d}$, with m observations in total. Let $\epsilon = O\left(\sqrt{\frac{\log d}{pd}}\right)$. For a broad class of general loss functions (including those applicable to noisy and one-bit matrix completion), there exists an algorithm that runs in time $\widetilde{O}(m \cdot \mathrm{poly}(r, \log d))$ and with high probability outputs a rank-r factorization $UV^\top$ satisfying,*

$$\|UV^\top - X^*\|_F^2 \le \epsilon\, \mathrm{poly}(\beta, \sigma_1, \nu, r, \log d).$$

This result recovers prior guarantees for the quadratic loss and provides a nearly-linear time solver for semi-random matrix completion under noisy and one-bit settings. We now outline our main contributions below:

- **Spectral similarity-based preprocessing for general losses.** We observe that the spectral sparsification-based reweighting method of (Cheng and Ge, 2018), originally developed for quadratic loss in the noiseless setting, extends to broader observation models. In particular, we apply the reweighting where the entries are scaled

entry-wise to achieve ϵ-spectral similarity to uniform observations and remains compatible with general loss functions, including those with noise.

- **Primal-dual analysis under semi-random observations.** We revisit the primal-dual framework of (Zhang et al., 2018b), originally analyzed under the uniform observation model. Our analysis demonstrates that once the regularity conditions are restored via reweighting, its guarantees remain valid with semi-random inputs.
- **Weighted regularity conditions and error guarantees.** We establish weighted versions of the regularity conditions (4.2.2) and the deviation condition (4.2.3) that are designed for the reweighted observations produced by the preprocessing algorithm of (Cheng and Ge, 2018). In particular, we show that these weighted conditions hold with high probability after the preprocessing (Lemma 4.3.3 and Lemma 4.3.4). Compared to the uniform model, this incurs only a polylogarithmic overhead in recovery error, sample complexity, and runtime.
- **Provable guarantees for noisy and one-bit matrix completion.** As concrete examples, we present results for semi-random noisy matrix completion and for one-bit matrix completion (Corollary 4.4.1 and Corollary 4.4.2), which to our knowledge has not previously been studied in the semi-random setting.

4.1.2 Technical Overview

At a high level, our approach combines two previously disjoint components, global reweighting and primal-dual analysis, into a unified framework for solving matrix com-

pletion problems under the semi-random observation model. We begin by applying the preprocessing algorithm of (Cheng and Ge, 2018), which rescales observed entries to produce a reweighted matrix spectrally similar to uniform observations. This allows us to recover key structural properties that typically fail under semi-random inputs. We then incorporate the resulting weights into the loss function to form a weighted objective, and prove that it satisfies suitable regularity conditions. Finally, we apply the primal-dual framework of (Zhang et al., 2018b) to solve the resulting constrained nonconvex problem. The overall algorithm runs in nearly-linear time and applies broadly to noisy, quantized, and other nonstandard observation models. We elaborate on each component below.

Preprocessing. The primal-dual analysis of (Zhang et al., 2018b) requires the loss function satisfies the RSC, RSS, and deviation conditions. These conditions are no longer guaranteed under the semi-random observation model. To restore them, we apply the preprocessing algorithm of (Cheng and Ge, 2018). The key idea in (Cheng and Ge, 2018) is to relate matrix indices to edges in a bipartite graph. Given a matrix $X \in \mathbb{R}^{d_1 \times d_2}$, define the complete bipartite graph $G = (V_1, V_2, E)$ where $|V_1| = d_1$ and $|V_2| = d_2$ correspond to the row and column indices of X. Let H denote the semi-random subgraph of G obtained by first sampling each edge independently with probability p, followed by the adversarial addition of edges to reach a total of m. The preprocessing algorithm assigns weights to the edges of H so that the resulting weighted graph is ϵ-spectrally-similar to the complete graph G. We represent these weights as a matrix $W \in \mathbb{R}^{d_1 \times d_2}$, where $W_{ij} = w_e$ supported on the edges of H. (Cheng and Ge, 2018) provides a nearly-linear time algorithm that

outputs a weight matrix W which achieves ϵ-spectral-similarity, where $\epsilon = O\left(\sqrt{\frac{\log d}{pd}}\right)$. We include some technical results on the preprocessing algorithm in Section 4.6.1, and provide additional properties on W to accompany prior results, stated as Lemmas 4.5.1 and 4.5.2.

Primal-dual framework. In the analysis of (Zhang et al., 2018b), the nonconvex objective is reformulated by stacking $U \in \mathbb{R}^{d_1 \times r}$ and $V \in \mathbb{R}^{d_2 \times r}$ as $Z = [U; V] \in \mathbb{R}^{(d_1+d_2)\times r}$ and introducing inequality constraints $h_i(Z) = \|Z_i\|_2^2 - \alpha$ for all $i \in [d_1 + d_2]$ to enforce the incoherence conditions. A regularization term $\frac{\gamma}{4}\|U^\top U - V^\top V\|_F^2$ is added to balance U and V, yielding the objective

$$\min_{\substack{Z=[U;V],\\ U\in\mathbb{R}^{d_1\times r}, V\in\mathbb{R}^{d_2\times r}}} G(Z) = F(UV^\top) + \frac{\gamma}{4}\left\|U^\top U - V^\top V\right\|_F^2, \text{ subject to } h_i(Z) \le 0 \ \forall\, i, \quad (4.4)$$

where F is a problem-specific loss function.

(Zhang et al., 2018b) analyze the dual objective under the KKT conditions (Karush, 1939; Kuhn and Tucker, 2013), and show that there are no spurious local minima in the primal objective (4.4) if F satisfies certain regularity conditions. We include the result as Theorem 4.2.4 in the Preliminaries.

Weighted loss and regularity conditions. Suppose we are given a matrix completion problem with loss function F, which satisfies the regularity conditions required by the primal-dual framework under the uniform observation model. Our goal is to solve this problem in the semi-random model, where each entry is observed with probabil-

ity at least p. In our unified framework, we begin by applying a preprocessing algorithm to achieve ϵ-spectral similarity. We then incorporate the resulting weight matrix W into F, yielding a weighted loss function F_W. Specifically, we consider the uniform loss function in the form of $F(X) = \sum_{j,k\in\Omega} \frac{1}{p} F_{jk}(X_{jk})$, and the weighted loss function is $F_W(X) = \sum_{j,k\in\Omega} W_{jk} F_{jk}(X_{jk})$. We then establish weighted versions of the RSC, RSS, and deviation conditions for F_W. This enables the application of the primal-dual framework to solve the optimization problem under the semi-random model. Full details are provided in Section 4.3.

4.1.3 Related Work

Matrix completion. Matrix completion is a popular type of low-rank matrix recovery problems with applications in collaborative filtering, image restoration (Rennie and Srebro, 2005; Zhang et al., 2013), etc. The line of work (Candes and Recht, 2008; Candès and Tao, 2010; Recht, 2011) shows that the convex relaxation via nuclear norm minimization can be solved by SDP in time $\widetilde{O}(md^{2.5})$ (Jiang et al., 2020; Huang et al., 2022) with sample complexity $\Omega(dr)$. Using the more efficient nonconvex approach, success has been shown using gradient descent with good, or sometimes even random, initialization (Keshavan et al., 2010; Jain et al., 2013; Hardt and Wootters, 2014; Chen and Wainwright, 2015; Zhao et al., 2015; Sun and Luo, 2016; Zheng and Lafferty, 2016; Gu et al., 2016; Tu et al., 2016; Wang et al., 2017; Xu et al., 2017; Zhang et al., 2018a; Chen et al., 2020; Gu et al., 2023; Xu et al., 2023). There are also Lagrangian-based analyses of nonconvex objectives (Zhang et al., 2018b; Nie et al., 2018). Another direction of nonconvex research (De Sa et

al., 2015; Ge et al., 2016; Ge et al., 2017; Park et al., 2017; Chen and Li, 2019; Zhu et al., 2018; Zhu et al., 2021) aim to study the global optimization landscape to show that there are no spurious local optima. The recent work of (Kelner et al., 2023b) achieves almost theoretically optimal sample complexity and running time.

One-bit matrix completion. Inspired by one-bit compressive sensing (Boufounos and Baraniuk, 2008), (Davenport et al., 2014) originally formulated the problem for matrices, and provided theoretical minimax error bound of $O(\sqrt{\frac{rd}{m}})$. (Cai and Zhou, 2013), (Eamaz et al., 2023; Eamaz et al., 2024) study the problem using the convex relaxation approach, and (Ni and Gu, 2016), (Zhang et al., 2018b) provided nonconvex solutions that match the minimax statistical error bound up to logarithmic factors and algorithms that converge at linear rate. (Lan et al., 2014), (Shen et al., 2019) study quantized matrix completion with corruption, and (Chen et al., 2023) studies estimators for quantized heavy-tailed data.

Non-uniform model. Apart from the well-studied uniform observation model for matrix completion, another direction of research studies weighted matrix completion under non-uniform observation models, such as deterministic sampling, where the sampling probabilities follow some non-uniform distribution specific to the problem. See e.g. (Lee and Shraibman, 2013; Heiman et al., 2014; Bhojanapalli and Jain, 2014; Li et al., 2016; Foucart et al., 2020; Ashraphijuo et al., 2017; Bhojanapalli et al., 2014; Pimentel-Alarcón et al., 2016; Meka et al., 2009; Liu et al., 2019). Under the non-uniform sampling model where each entry is observed with probability p_{ij} bounded above and below, (Chen and Li,

2022; Chen and Li, 2024) study matrix completion under both Frobenius and entry-wise error bounds.

Semi-random model. While also a non-uniform observation model, the semi-random model does not assume any specific sampling pattern, other than requiring that each entry is observed with probability at least p. This model was first introduced to combinatorial optimization problems such as graph coloring, planted clique, stochastic block models, clustering etc (Blum and Spencer, 1995; Feige and Kilian, 2001; Perry and Wein, 2017; Mathieu and Schudy, 2010; Makarychev et al., 2012). It was later applied to low-rank matrix recovery (Cheng and Ge, 2018), sparse linear regression (Kelner et al., 2023a), and matrix sensing (Gao and Cheng, 2024). Closely related to this work, (Kelner et al., 2024) studies semi-random noisy matrix completion with an non-adaptive adversary, while our work does not require this restriction. In contrast to our global reweighting approach, (Kelner et al., 2024) uses projected gradient descent with iterative reweighting guaranteed by a "short-flat decomposition" technique. They provide close to low-rank solution in nearly-linear time, with an error bound in terms of ℓ_∞ norm (entry-wise measure of noise and error) instead of the more commonly used Frobenius norm (global measure). While the focus and technical tools of (Kelner et al., 2024) differ from ours, both works address similar challenges posed by semi-random observations under noise. Our results are complementary: we provide Frobenius-norm guarantees under a broad family of loss functions using a different reweighting mechanism, and our framework directly supports one-bit matrix completion.

4.2 Preliminaries

4.2.1 Notation

We write $[n]$ for the set of integers $\{1, ..., n\}$. Denote the i^{th} standard basis vector as e_i. For a matrix A, we use $\|A\|_2$, $\|A\|_F$ and $\|A\|_{\max}$ for the operator norm, Frobenius norm and the maximum absolute entry of A respectively. We write $\|A\|_\infty$, $\|A\|_1$ for the maximum ℓ_1-norms of the rows and columns of A respectively. We denote the i^{th} row of A as A_i, and the maximum ℓ_2 norm of rows of A as $\|A\|_{2,\infty}$.

For matrices $A, B \in \mathbb{R}^{d_1 \times d_2}$, we denote their inner product as $\langle A, B\rangle$, which is defined as: $\langle A, B\rangle = \text{tr}(A^\top B) = \sum_{j,k} A_{jk}B_{jk}$. Given subset of entries $\Omega \subset [d_1] \times [d_2]$, define $\langle A, B\rangle_\Omega = \sum_{(j,k)\in\Omega} A_{jk}B_{jk}$; given weight matrix $W \in \mathbb{R}^{d_1\times d_2}$, define $\langle A, B\rangle_W = \sum_{j,k} W_{jk}A_{jk}B_{jk}$. We write $\|A\|_\Omega^2$ for $\langle A, A\rangle_\Omega$, and $\|A\|_W^2$ for $\langle A, A\rangle_W$. Symmetric matrix $A \in \mathbb{R}^{d\times d}$ is called positive semidefinite (PSD) if $x^\top Ax \geq 0 \, \forall x \in \mathbb{R}^d$, and we write $A \preccurlyeq B$ if A and B have the same dimension and $B - A$ is PSD.

For a weighted undirected graph $G = (V, E, w)$ with n vertices and weights $w_e \geq 0$ for each edge $e = (i, j)$, let $D \in \mathbb{R}^{n\times n}$ be a diagonal matrix containing the weighted degree of each vertex, i.e. $D_{ii} = \sum_{(i,j)\in E} w_{(i,j)}$. Let $A \in \mathbb{R}^{n\times n}$ be the adjacency matrix of G, i.e. $A_{ij} = A_{ji} = w_{(i,j)}$. The Laplacian matrix of G is defined as $L := D - A$. Fix some arbitrary orientation for each edge $e = (i, j) \in E$, we can represent it with vector $b_e \in \mathbb{R}^n$ where $b_e[i] = 1$ and $b_e[j] = -1$, and $L = \sum_{e\in E} w_e b_e b_e^\top$.

4.2.2 Assumptions on the Ground Truth Matrix

Throughout the paper we assume the ground truth matrix $X^* \in \mathbb{R}^{d_1 \times d_2}$ and let $d = \max(d_1, d_2)$. Assume X^* has rank $r \ll d$. Suppose X^* has compact SVD $X^* = \widetilde{U}\Sigma\widetilde{V}^\top$, where $\widetilde{U}$ and $\widetilde{V}$ are orthogonal matrices in $\mathbb{R}^{d_1 \times r}$ and $\mathbb{R}^{d_2 \times r}$ respectively, and $\Sigma = \text{diag}(\sigma_1, ..., \sigma_r)$, the diagonal matrix with the singular values of X^* on its diagonal.

It is well known that matrix completion is impossible if the ground truth matrix is too sparse, which means most of the observed entries will be 0. A common solution in previous work is to impose the incoherence condition on X^* defined below.

Definition 4.2.1 (Incoherence Condition (Candès and Recht, 2009))**.** *A rank-r matrix $X \in \mathbb{R}^{d_1 \times d_2}$ with SVD $X = \widetilde{U}\Sigma\widetilde{V}^\top$ is said to be incoherent with parameter β if*

$$\left\|\widetilde{U}\right\|_{2,\infty} \leq \sqrt{\frac{\beta r}{d_1}} \text{ and } \left\|\widetilde{V}\right\|_{2,\infty} \leq \sqrt{\frac{\beta r}{d_2}}.$$

Let $\alpha_1 = \sqrt{\frac{\beta r \sigma_1}{d_1}}$, $\alpha_2 = \sqrt{\frac{\beta r \sigma_1}{d_2}}$. We define the following constraint sets

$$\mathcal{C}_1 := \{U \in \mathbb{R}^{d_1 \times r} \mid \|U\|_{2,\infty} \leq \alpha_1\},$$

$$\mathcal{C}_2 := \{V \in \mathbb{R}^{d_2 \times r} \mid \|V\|_{2,\infty} \leq \alpha_2\},$$

$$\mathcal{C} := \{UV^\top \mid U \in \mathcal{C}_1,\ V \in \mathcal{C}_2\}.$$

Given $X^* = \widetilde{U}\Sigma\widetilde{V}^\top$, we can alternatively write X^* as $X^* = U^*V^{*\top}$, where $U^* = \widetilde{U}\Sigma^{1/2} \in \mathbb{R}^{d_1 \times r}$ and $V^* = \widetilde{V}\Sigma^{1/2} \in \mathbb{R}^{d_2 \times r}$. We assume X^* satisfies the incoherence condition,

therefore $\|U^*\|_{2,\infty} \leq \alpha_1$ and $\|V^*\|_{2,\infty} \leq \alpha_2$, i.e., $X^* \in \mathcal{C}$. Let $\alpha = \alpha_1\alpha_2 = \frac{\beta r \sigma_1}{\sqrt{d_1 d_2}}$, so that $\|X^*\|_{\max} \leq \alpha$.

4.2.3 Conditions on the Loss Function

Quadratic loss function is popular due to the isometric property of its Hessian, which a general loss function is no longer guaranteed to satisfy. A common alternative is to enforce a pair of regularity conditions known as the Restricted Strong Convexity (RSC) and Restricted Strong Smoothness (RSS) conditions (Agarwal et al., 2012), which we formally state below.

Definition 4.2.2 (RSC and RSS Conditions (Agarwal et al., 2012))**.** *The loss function F is said to satisfy the Restricted Strong Convexity condition with parameter μ, and the Restricted Strong Smoothness condition with parameter L if $\forall X_1, X_2 \in \mathbb{R}^{d_1 \times d_2}$ with rank at most $6r$:*

$$F(X_1) \geq F(X_2) + \langle \nabla F(X_2), X_1 - X_2 \rangle + \frac{\mu}{2}\|X_1 - X_2\|_F^2 \ ,$$
$$F(X_1) \leq F(X_2) + \langle \nabla F(X_2), X_1 - X_2 \rangle + \frac{L}{2}\|X_1 - X_2\|_F^2 \ .$$

To control the statistical error due to observation noise, the deviation condition is introduced by (Loh and Wainwright, 2012) for linear regression on vectors. A similar condition is used by (Zhang et al., 2018b) on the gradient of the loss function with respect to X^*, and we restate the definition below.

Definition 4.2.3 (Deviation Condition (Zhang et al., 2018b)). *The loss function F is said to satisfy the deviation condition if $\|\nabla F(X^*)\|_2 \leq \delta$ with high probability, where the deviation bound δ depends on the sampling rate p and the observation noise ν.*

For the one-bit matrix completion problem where the noise $-E_{jk}$ follows distribution with cdf f, denote the entry-wise standard deviation of noise $\mathrm{S}td[E_{jk}] = \tau = \frac{\nu}{\sqrt{d_1 d_2}}$. It is common practice to replace f with its standardization g, where $f(x) = g(x/\tau)$. For example, $g(z) = \Phi(z)$ in the case of Gaussian noise, where $\Phi(z)$ is the standard Gaussian distribution function. We define the dimension free signal-to-noise ratio $\rho = \frac{\alpha}{\tau} = \frac{\beta r \sigma_1}{\nu}$. Note that we only care about the value of g restricted to $|x| \leq \rho$ due to the assumption that $\|X^*\|_{\max} \leq \alpha$. For one-bit matrix completion to be feasible, a steepness assumption (Davenport et al., 2014) is imposed on the distribution function used in the probit model. We control the steepness of the standardized distribution function $g(x)$ by the quantity s_ρ defined as

$$s_\rho = \sup_{|x| \leq \rho} \frac{|g'(x)|}{g(x)(1 - g(x))}, \tag{4.5}$$

which is a constant given ρ and $g(x)$.

4.2.4 Error Guarantee of the Primal-Dual Framework

Here we include the main result from (Zhang et al., 2018b), which states the general error guarantee of the primal-dual framework under regularity conditions.

Theorem 4.2.4 (General error bound, Theorem 3.8 (Zhang et al., 2018b)). *Assume loss function F satisfies the RSC and RSS conditions (4.2.2) with parameter μ and L such that $L/\mu \in (1, 18/17)$, as well as the deviation condition (4.2.3) with deviation bound δ. For all local minima $Z = [U; V]$ of the optimization objective (4.4), with high probability the reconstruction error satisfies*

$$\left\| UV^\top - X^* \right\|_F^2 \leq \Gamma r \delta^2,$$

where Γ is a constant depending on the condition number $\frac{L}{\mu}$. Specifically, $\Gamma = \frac{10}{(10\mu - 9L - \gamma - 3c)c}$ for $c = \frac{18\mu - 17L}{12}$, and γ in the primal objective (4.4) is chosen such that $\mu - L/2 \leq \gamma < \min\{(22\mu - 19L)/4, (3L - 2\mu)/2\}$.

4.3 Our Results Under Semi-Random Preprocessing

4.3.1 Weighted Regularity Conditions

First we study the effect of preprocessing on the conditions required by the primal-dual framework. Given a matrix completion problem which the primal-dual framework is applicable under the uniform observation model, suppose it has objective function (4.4) where the loss function F satisfies the RSC and RSS conditions (4.2.2) with parameters μ and L, as well as the deviation condition (4.2.3) with δ. As discussed before, in the semi-random model with sampling rate at least p, we first run the preprocessing algorithm to achieve spectral similarity $\epsilon = O\left(\sqrt{\frac{\log d}{pd}}\right)$, apply the resulting weight matrix W to the loss function F, and obtain a weighted objective function (4.4) with F_W.

To establish weighted regularity conditions in general, we state the following assumptions on the loss function in general, and later verify that these additional assumptions are satisfied by the two specific examples we study in this paper.

Assumption 4.3.1. *For uniform loss function* $F(X) = \sum_{j,k\in\Omega} \frac{1}{p} F_{jk}(X_{jk})$ *satifying RSC and RSS with parameters* μ *and* L*, for* $X_1, X_2 \in \mathcal{C}$ *we assume*

$$F(X_1) - F(X_2) - \langle \nabla F(X_2), X_1 - X_2 \rangle = \frac{1}{2} \sum_{j,k\in\Omega} \frac{1}{p} K_{jk}(X_{1,jk} - X_{2,jk})^2,$$

for some K_{jk} *where* $\mu \le K_{jk} \le L$ *for all* j, k*.*

Assumption 4.3.2. *For uniform loss function* $F(X) = \sum_{j,k\in\Omega} \frac{1}{p} F_{jk}(X_{jk})$*, we assume*

$$\nabla F(X^*) = \sum_{(j,k)\in\Omega} \frac{1}{p} b_{jk}(X^*) e_j e_k^\top,$$

where each $b_{jk}(X^*)$ *is a sub-exponential random variable given* X^**, and is i.i.d with mean* 0 *and variance* $s^2 = O(\frac{1}{d^2})$ *possibly depending on the distribution of the noise* E_{jk}*.*

The following lemma states the RSC and RSS conditions for F_W.

Lemma 4.3.3 (Weighted RSC and RSS conditions)**.** *Under Assumption 4.3.1, given* F *satisfying the RSC and RSS with parameters* μ *and* L*, and* F_W *weighted by the preprocessing algorithm to* ϵ*-spectral-similarity,* $\forall X_1, X_2 \in \mathcal{C}$ *(Definition 4.2.1), if* $\|X_1 - X_2\|_F^2 \ge O(\beta^2 r^2 \sigma_1^2 \epsilon)$*, then we can establish the RSC and RSS conditions on* F_W *at* X_1 *and* X_2 *with parameters* $\mu_W = (1-c)\mu$ *and* $L_W = (1+c)L$*, for a sufficiently small constant* c*.*

Proof. Let $X_1, X_2 \in \mathcal{C}$ and $D = X_1 - X_2$.

By Assumption 4.3.1,

$$F(X_1) - F(X_2) - \langle \nabla F(X_2), X_1 - X_2 \rangle = \frac{1}{2} \sum_{j,k \in \Omega} \frac{1}{p} K_{jk} D_{jk}^2.$$

For the weighted loss function,

$$F_W(X_1) - F_W(X_2) - \langle \nabla F_W(X_2), X_1 - X_2 \rangle = \frac{1}{2} \sum_{j,k \in \Omega} K_{jk} W_{jk} D_{jk}^2.$$

We first show the upper bound for RSS:

$$\begin{aligned} \frac{1}{2} \sum_{j,k \in \Omega} K_{jk} W_{jk} D_{jk}^2 &\leq \frac{L}{2} \sum_{j,k \in \Omega} W_{jk} D_{jk}^2 \\ &= \frac{L}{2} \|X_1 - X_2\|_W^2 \\ &\leq \frac{L}{2}(1+c)\|X_1 - X_2\|_F^2, \end{aligned}$$

where the last inequality follows from Lemma 4.5.1: since $\|X_1 - X_2\|_F^2 \geq O(\beta^2 r^2 \sigma_1^2 \epsilon)$, we have $\left| \|X_1 - X_2\|_W^2 - \|X_1 - X_2\|_F^2 \right| \leq c \cdot \|X_1 - X_2\|_F^2$ for a small constant c.

The lower bound for RSC follows similarly, thus we establish the RSC and RSS conditions with parameters $\mu_W = (1-c)\mu$ and $L_W = (1+c)L$ provided X_1 and X_2 are sufficiently far apart. ■

The next lemma states the effect of preprocessing on the deviation condition.

Lemma 4.3.4 (Weighted deviation condition)**.** *Under Assumption 4.3.2, given F in the uniform observation model and let F_W be weighted by the preprocessing algorithm to ϵ-spectral-similarity, w.h.p. we have $\delta_W^2 = \|\nabla F_W(X^*)\|_2^2 \le O(s^2 d^2 \epsilon \log d)$.*

Proof. By Assumption 4.3.2, $\nabla F(X^*) = \sum_{(j,k)\in\Omega} \frac{1}{p} b_{jk}(X^*) e_j e_k^\top$, after preprocessing we can write

$$\nabla F_W(X^*) = \sum_{(j,k)\in\Omega} W_{jk} b_{jk}(X^*) e_j e_k^\top = \sum_{(j,k)\in\Omega} Z_{jk},$$

where $Z_{jk} = W_{jk} b_{jk}(X^*) e_j e_k^\top$.

We will bound its operator norm using matrix Bernstein inequality (Theorem 4.6.6). First, we calculate

$$\sum_{(j,k)\in\Omega} \mathbb{E}\Big[Z_{jk} Z_{jk}^\top\Big] = \sum_{(j,k)\in\Omega} \mathbb{E}\Big[b_{jk}^2(X^*) W_{jk}^2 e_j e_j^\top\Big] = s^2 \sum_{(j,k)\in\Omega} W_{jk}^2 e_j e_j^\top,$$

which is a diagonal matrix, therefore

$$\begin{aligned}
\Big\| \sum_{(j,k)\in\Omega} \mathbb{E}\Big[Z_{jk} Z_{jk}^\top\Big] \Big\|_2 &= s^2 \max_j \sum_k W_{jk}^2 \\
&\le s^2 \|W\|_{\max} \cdot \max_j \sum_k W_{jk} \\
&= s^2 \|W\|_{\max} \cdot \|W\|_\infty \\
&= O(s^2 \epsilon \sqrt{d_1 d_2} d_2),
\end{aligned}$$

where the last step is due to Lemma 4.6.2 and 4.5.2.

Similarly, we can show that $\left\| \sum_{(j,k)\in\Omega} \mathbb{E}\left[Z_{jk}^\top Z_{jk}\right] \right\|_2 = O(s^2\epsilon\sqrt{d_1 d_2}d_1)$.

Since $b_{jk}(X^*)$ is sub-exponential with mean 0 and variance s^2, we have $\mathbb{E}[Z_{jk}] = \mathbf{0}$, and $\|Z_{jk}\|_{\Psi_1} \leq O(sW_{jk}) = O(sd\epsilon)$. We can apply Theorem 4.6.6 by setting $\sigma^2 = O(s^2d^2\epsilon)$ and $R = O(sd\epsilon)$. We have $\Pr\left[\left\| \sum_{(j,k)\in\Omega} Z_{jk} \right\|_2 \geq t\right] \leq 2d \cdot \exp(\frac{-t^2}{2\sigma^2+2Rt/3})$, and simple calculation shows that the second order term dominates. Choosing $t = c\sigma\sqrt{\log d}$ for some constant c, we have $\left\| \sum_{(j,k)\in\Omega} Z_{jk} \right\|_2 \leq t$ with probability at least $1 - d^{-(c^2-1)}$. Final result follows by plugging in the value of t. ■

Condition number and deviation bound. The preprocessing step incurs a small cost in the condition number L/μ, increasing it by a factor of $\frac{1+c}{1-c}$ for some small constant c. For the deviation bound, observe that under the same condition as Lemma 4.3.4, the loss function F in the uniform observation model satisfies the deviation condition with $\delta^2 = \|\nabla F(X^*)\|_2^2 \leq O\left(s^2\frac{d\log d}{p}\right)$, as shown in Lemma 4.6.5 (Zhang et al., 2018b; Agarwal et al., 2012). Comparing the two deviation bounds δ_W^2 and δ^2, preprocessing increases the bound by a factor of $O(\epsilon pd)$.

4.3.2 General Error Bound for Semi-Random Matrix Completion

With the weighted RSC and RSS conditions as well as the weighted deviation condition established in the previous section, we now extend Theorem 4.2.4, which states the optimality of the primal-dual framework, to derive a general error bound in the semi-random model.

Theorem 4.3.5 (General semi-random error bound)**.** *Given a matrix completion problem that satisfies the conditions in Theorem 4.2.4 as well as the additional Assumptions 4.3.1 and 4.3.2 under the uniform observation model, consider the same problem under the semi-random model after preprocessing to ϵ-spectral-similarity, where $\epsilon = O\left(\sqrt{\frac{\log d}{pd}}\right)$ given observation probability at least p. For all local minima $[U; V]$ of objective (4.4) with the weighted loss F_W, w.h.p. the error satisfies*

$$\left\|UV^\top - X^*\right\|_F^2 \leq \max\left\{O(\beta^2 r^2 \sigma_1^2 \epsilon), \Gamma r s^2 d^2 \epsilon \log d\right\},$$

where Γ is a constant depending on μ_W and L_W. Given m observed entries, the overall running time is $\widetilde{O}(m \cdot \mathrm{poly}(r, \log d))$.

Proof. In the semi-random model with loss function weighted by preprocessing, first we apply Lemma 4.3.3 to establish the RSC and RSS conditions on F_W, increasing the condition number L/μ by a factor of $\frac{1+c}{1-c}$, which affects Γ by a constant factor as well. Then Lemma 4.3.4 provides the deviation condition. Now we can apply Theorem 4.2.4 to get the error bound $\left\|UV^\top - X^*\right\|_F^2 \leq \Gamma r s^2 d^2 \epsilon \log d$. In addition, in the case that Lemma 4.3.3 does not apply with respect to $X_1 = UV^\top$ and $X_2 = X^*$, we immediately have $\left\|UV^\top - X^*\right\|_F^2 \leq O(\beta^2 r^2 \sigma_1^2 \epsilon)$. Combining these two cases gives the overall error bound. ■

Running time. The objective (4.4), with either uniform or weighted loss, can be solved by the Augmented Lagrangian Method in (Zhang et al., 2018b; Nocedal and Wright, 2006)

with running time $\widetilde{O}(mr + dr^2)$, where the dominant cost is computing $\nabla_Z G$ iteratively. The preprocessing algorithm from (Cheng and Ge, 2018) takes time $\widetilde{O}(m/\epsilon^{O(1)})$, where $\epsilon^{-1} = \mathrm{poly}(r, \log d)$. Therefore our total running time is $\widetilde{O}(m \cdot \mathrm{poly}(r, \log d))$, compared to $\widetilde{O}(mr)$ in the uniform observation model.

4.4 Applications to Noisy and One-Bit Matrix Completion

In this section we showcase two specific examples in the semi-random model, namely noisy matrix completion and one-bit matrix completion. Since these two examples were demonstrated using the primal-dual analysis under the uniform observation model in (Zhang et al., 2018b), we directly apply Theorem 4.3.5 and state the results as corollaries here. We also include a self-contained version of the proofs in Section 4.6.3 and 4.6.4.

Corollary 4.4.1 (Semi-random noisy matrix completion)**.** *For a noisy matrix completion problem under the semi-random observation model with observation probability at least* $p = O\left(\frac{r^4 \log^3 d}{d}\right)$*, suppose the ground truth matrix* $X^* \in \mathbb{R}^{d_1 \times d_2}$ *has rank* r *and satisfies the incoherence condition with parameter* β*, and entries are subject to i.i.d noise* E_{jk} *with variance* $\frac{\nu^2}{d_1 d_2}$*. Using weights from the preprocessing algorithm, w.h.p. all local minima* $UV^\top$ *of the weighted objective satisfy*

$$\left\|UV^\top - X^*\right\|_F^2 \leq \max\left\{O\left(\beta^2\sigma_1^2\sqrt{\frac{r^4 \log d}{pd}}\right), O\left(\nu^2\sqrt{\frac{r^2 \log^3 d}{pd}}\right)\right\}.$$

Proof. In the uniform observation model, it is shown by Corollary 4.1 (Zhang et al., 2018b) that loss function (4.1) satisfies the conditions in Theorem 4.2.4, therefore we only need to verify the additional Assumptions 4.3.1 and 4.3.2.

Recall the uniform loss function (4.1) and compute its gradient:

$$\begin{aligned}\nabla F(X) &= \sum_{(j,k)\in\Omega} \frac{1}{p}(X_{jk} - Y_{jk}) e_j e_k^\top \\ &= \sum_{(j,k)\in\Omega} \frac{1}{p} b_{jk}(X) e_j e_k^\top,\end{aligned}$$

evaluating at X^*, $b_{jk}(X^*) = E_{jk}$. By assumption, each E_{jk} is i.i.d sub-exponential with mean 0, variance $\frac{\nu^2}{d_1 d_2}$, satisfying Assumption 4.3.2 with $s^2 = \frac{\nu^2}{d_1 d_2}$.

For Assumption 4.3.1, we have:

$$\begin{aligned}&F(X_1) - F(X_2) - \langle \nabla F(X_2), X_1 - X_2 \rangle \\ &= \frac{1}{2} \sum_{(j,k)\in\Omega} \frac{1}{p} [(X_{1,jk} - Y_{jk})^2 - (X_{2,jk} - Y_{jk})^2] - \sum_{(j,k)\in\Omega} \frac{1}{p} (X_{2,jk} - Y_{jk})(X_{1,jk} - X_{2,jk}) \\ &= \frac{1}{2} \sum_{(j,k)\in\Omega} \frac{1}{p} [(X_{1,jk} + X_{2,jk} - 2Y_{jk})(X_{1,jk} - X_{2,jk}) - (2X_{2,jk} - 2Y_{jk})(X_{1,jk} - X_{2,jk})] \\ &= \frac{1}{2} \sum_{(j,k)\in\Omega} \frac{1}{p} (X_{1,jk} - X_{2,jk})^2,\end{aligned}$$

satisfying Assumption 4.3.1 with $K_{jk} = 1$ for all j, k. Note that according to (Zhang et al., 2018b), the choice of $L = \frac{44}{43}$ and $\mu = \frac{42}{43}$.

We can apply Theorem 4.3.5, $\left\|UV^\top - X^*\right\|_F^2 \le \max\left\{O(\beta^2 r^2 \sigma_1^2 \epsilon), O(r\nu^2 \epsilon \log d)\right\}$. According to Lemma 4.6.2, given sampling rate p, the preprocessing algorithm can achieve $\epsilon = O\left(\sqrt{\frac{\log d}{pd}}\right)$, plugging in ϵ gives the error bounds. ■

Corollary 4.4.2 (Semi-random one-bit matrix completion)**.** *For a one-bit matrix completion problem under the semi-random observation model with observation probability at least* $p = O\left(\frac{r^4 \log^3 d}{d}\right)$*, suppose the ground truth matrix* $X^* \in \mathbb{R}^{d_1 \times d_2}$ *has rank* r *and satisfies the incoherence condition with parameter* β*, and the cdf in the observation model satisfies the steepness condition (4.5) with parameter* s_ρ*. Using weights from the preprocessing algorithm, w.h.p. all local minima* $UV^\top$ *of the weighted objective function satisfy*

$$\left\|UV^\top - X^*\right\|_F^2 \le \max\left\{O\left(\beta^2 \sigma_1^2 \sqrt{\frac{r^4 \log d}{pd}}\right), O\left(s_\rho^2 \sqrt{\frac{r^2 \log^3 d}{pd}}\right)\right\}.$$

Proof. In the uniform observation model, it is shown by Corollary 4.3 (Zhang et al., 2018b) that loss function (4.2) satisfies the conditions in Theorem 4.2.4, therefore we only need to verify the additional Assumptions 4.3.1 and 4.3.2.

Recall the loss function (4.2) and write it in its standardized form:

$$\begin{aligned} F(X) &= -\frac{1}{pd_1 d_2} \sum_{(j,k)\in\Omega} \Big[\mathbb{1}(Y_{jk} = 1) \cdot \log\big(f(X_{jk})\big) + \mathbb{1}(Y_{jk} = -1) \cdot \log\big(1 - f(X_{jk})\big)\Big] \\ &= \frac{1}{d_1 d_2} \sum_{(j,k)\in\Omega} \frac{1}{p} \Big[\mathbb{1}(Y_{jk} = 1) \cdot \log\big(g(X_{jk}/\tau)\big) + \mathbb{1}(Y_{jk} = -1) \cdot \log\big(1 - g(X_{jk}/\tau)\big)\Big] \end{aligned}$$

Compute its gradient,

$$\nabla F(X) = \frac{1}{d_1 d_2 \tau} \sum_{(j,k)\in\Omega} \frac{1}{p} b_{jk}(X) e_j e_k^\top,$$

where

$$b_{jk}(X) = -\mathbb{1}(Y_{jk} = 1) \cdot \frac{g'(X_{jk}/\tau)}{g(X_{jk}/\tau)} + \mathbb{1}(Y_{jk} = -1) \cdot \frac{g'(X_{jk}/\tau)}{1 - g(X_{jk}/\tau)}.$$

At X^*,

$$b_{jk}(X^*) = \begin{cases} -\frac{g'(X^*_{jk}/\tau)}{g(X^*_{jk}/\tau)} & \text{with probability } g(X^*_{jk}/\tau), \\ \frac{g'(X^*_{jk}/\tau)}{1-g(X^*_{jk}/\tau)} & \text{with probability } 1 - g(X^*_{jk}/\tau). \end{cases}$$

Therefore $\mathbb{E}[b_{jk}(X^*)] = 0$. Next we compute the variance:

$$\begin{aligned} \mathbb{V}ar[b_{jk}(X^*)] &= \mathbb{E}\big[b_{jk}^2(X^*)\big] - \mathbb{E}[b_{jk}(X^*)]^2 \\ &= \left[\frac{g'(X^*_{jk}/\tau)}{g(X^*_{jk}/\tau)}\right]^2 \cdot g(X^*_{jk}/\tau) + \left[\frac{g'(X^*_{jk}/\tau)}{1 - g(X^*_{jk}/\tau)}\right]^2 \cdot (1 - g(X^*_{jk}/\tau)) \\ &= \frac{g'^2(X^*_{jk}/\tau)}{g(X^*_{jk}/\tau)(1 - g(X^*_{jk}/\tau))} \\ &= \left(\frac{|g'(X^*_{jk}/\tau)|}{g(X^*_{jk}/\tau)(1 - g(X^*_{jk}/\tau)}\right)^2 \cdot g(X^*_{jk}/\tau)(1 - g(X^*_{jk}/\tau)) \\ &\leq s_\rho^2. \end{aligned}$$

The last step follows from the definition of s_ρ and the fact that $g(\cdot)$ is a cumulative distribution function and its range is $[0,1]$. Each $\frac{b_{jk}(X^*)}{d_1 d_2 \tau}$ is sub-exponential with mean 0, variance $\frac{s_\rho^2}{d^4\tau^2} = \frac{s_\rho^2}{d^2\nu^2}$, satsifying Assumption 4.3.2.

To verify Assumption 4.3.1, for all $X_1, X_2 \in \mathcal{C}$, applying the Mean Value Theorem to the second order remainder, $\exists M = tX_1 + (1-t)X_2$ for some $t \in [0,1]$ such that:

$$F(X_1) - F(X_2) - \langle \nabla F(X_2), X_1 - X_2 \rangle = \frac{1}{2} \operatorname{vec}(X_1 - X_2)^\top \nabla^2 F(M) \operatorname{vec}(X_1 - X_2).$$

Compute the Hessian of the loss function (4.2):

$$\nabla^2 F(X) = \frac{1}{\nu^2} \sum_{(j,k)\in\Omega} \frac{1}{p} B_{jk}(X) \operatorname{vec}(e_j e_k^\top) \operatorname{vec}(e_j e_k^\top)^\top,$$

where

$$B_{jk}(X) = \mathbb{1}(Y_{jk} = 1) \cdot \left(\frac{g'^2(X_{jk}/\tau)}{g^2(X_{jk}/\tau)} - \frac{g''(X_{jk}/\tau)}{g(X_{jk}/\tau)} \right) + \mathbb{1}(Y_{jk} = -1) \cdot \left(\frac{g''(X_{jk}/\tau)}{1 - g(X_{jk}/\tau)} + \frac{g'^2(X_{jk}/\tau)}{\left(1 - g(X_{jk}/\tau)\right)^2} \right).$$

Two parameters were introduced in (Ni and Gu, 2016) to control the quadratic lower and upper bounds of the second-order Taylor expansion of the log-likelihood function:

$$\mu_\rho = \min\left\{\inf_{|x|\le\rho}\left(\frac{g'^2(x)}{g^2(x)} - \frac{g''(x)}{g(x)}\right), \inf_{|x|\le\rho}\left(\frac{g'^2(x)}{(1-g(x))^2} + \frac{g''(x)}{1-g(x)}\right)\right\},$$
$$L_\rho = \max\left\{\sup_{|x|\le\rho}\left(\frac{g'^2(x)}{g^2(x)} - \frac{g''(x)}{g(x)}\right), \sup_{|x|\le\rho}\left(\frac{g'^2(x)}{(1-g(x))^2} + \frac{g''(x)}{1-g(x)}\right)\right\}.$$

Evaluating at $M = tX_1 + (1-t)X_2$, we have $|M_{jk}/\tau| \le \rho$ (recall the signal to noise ratio $\rho = \frac{\alpha}{\tau}$), so that $\mu_\rho \le B_{jk}(M) \le L_\rho$ for all j, k, and

$$\begin{aligned} F(X_1) - F(X_2) - \langle \nabla F(X_2), X_1 - X_2\rangle &= \frac{1}{2\nu^2}\sum_{(j,k)\in\Omega}\frac{1}{p}B_{jk}(M)\langle e_j e_k^\top, X_1 - X_2\rangle^2 \\ &= \frac{1}{2}\sum_{(j,k)\in\Omega}\frac{1}{p}K_{jk}(X_{1,jk} - X_{2,jk})^2, \end{aligned}$$

where $\frac{\mu_\rho}{\nu^2} \le K_{jk} \le \frac{L_\rho}{\nu^2}$. Note that according to (Zhang et al., 2018b), $\mu = \frac{42}{43}\frac{\mu_\rho}{\nu^2}$ and $L = \frac{44}{43}\frac{L_\rho}{\nu^2}$, therefore we satisfy Assumption 4.3.1 with $\mu \le K_{jk} \le L$.

We can apply Theorem 4.3.5, $\left\|UV^\top - X^*\right\|_F^2 \le \max\left\{O(\beta^2 r^2 \sigma_1^2 \epsilon), O(r s_\rho^2 \epsilon \log d)\right\}$. Note that the $1/\nu^2$ factor cancels out, since the RSS and RSC parameters μ, L are of scale $1/\nu^2$ due to the scaling of the loss function, therefore $\Gamma = O(\nu^2)$. According to Lemma 4.6.2, given sampling rate p, the preprocessing algorithm can achieve $\epsilon = O\left(\sqrt{\frac{\log d}{pd}}\right)$, plugging in ϵ gives the error bounds. ■

Sampling rate and error. In the uniform observation model, for both problems (Zhang et al., 2018b) requires sampling rate $p = O\left(\frac{r^2 \log d}{d}\right)$, and achieves error rate $O\left(\frac{r^2 \log d}{pd}\right)$

and $O\left(\frac{r \log d}{pd}\right)$ for the two respective terms in the bound. In comparison, our results in the semi-random model incur an additional factor of $O(r^2 \log^2 d)$ in both the required sampling rate and the error.

4.5 Supporting Lemmas

In this section, we present our supporting lemmas: Lemmas 4.5.1 and 4.5.2. We make use of prior results from the preprocessing procedure of (Cheng and Ge, 2018), which we include in Section 4.6.1 as Theorem 4.6.1 and Lemmas 4.6.2, 4.6.3, and 4.6.4 for completeness. Our lemmas make use of these prior results, and together they characterize the properties of the weight matrix W that are essential for establishing the parameters used in the weighted RSC, RSS, and deviation conditions.

Lemma 4.5.1 is a consequence of Lemma 4.6.4 and 4.6.2, which bounds the deviation of Frobenius norm of a matrix after applying weight matrix W on it.

Lemma 4.5.1. *Suppose the preprocessing step produces weight matrix W that achieves ϵ-spectral-similarity. For all M_1, M_2 from the constraint set $\mathcal{C}$,*

either $\left| \|M_1 - M_2\|_W^2 - \|M_1 - M_2\|_F^2 \right| \leq c \cdot \|M_1 - M_2\|_F^2$ *for some small constant c,*

or $\|M_1 - M_2\|_F^2 \leq O(\beta^2 r^2 \sigma_1^2 \epsilon)$.

Proof. Based on the definition for constraint set $\mathcal{C}$, we can write $M_1 = U_1 V_1^\top$ and $M_2 = U_2 V_2^\top$ for some $U_1, U_2 \in \mathcal{C}_1$ and $V_1, V_2 \in \mathcal{C}_2$, where $\mathcal{C}_1 = \{U \in \mathbb{R}^{d_1 \times r} \mid \|U\|_{2,\infty} \leq \alpha_1\}$, and $\mathcal{C}_2 = \{V \in \mathbb{R}^{d_2 \times r} \mid \|V\|_{2,\infty} \leq \alpha_2\}$. let $X = [U_1, U_2] \in \mathbb{R}^{d_1 \times 2r}$ and $Y = [V_1, -V_2] \in \mathbb{R}^{d_2 \times 2r}$, so that $M_1 - M_2 = XY^\top$.

Following Lemma 4.6.2, suppose $\|W - J\|_2 \leq C\epsilon\sqrt{d_1 d_2}$ for some constant C. We have the following inequality:

$$\begin{aligned}
\left| \left\| XY^\top \right\|_W^2 - \left\| XY^\top \right\|_F^2 \right| &\leq \|W - J\|_2 \cdot \|X\|_F \cdot \|Y\|_F \cdot \|X\|_{2,\infty} \cdot \|Y\|_{2,\infty} \\
&\leq \|W - J\|_2 \cdot \sqrt{d_1}\|X\|_{2,\infty} \cdot \sqrt{d_2}\|Y\|_{2,\infty} \cdot \|X\|_{2,\infty} \cdot \|Y\|_{2,\infty} \\
&= \sqrt{d_1 d_2} \cdot \|W - J\|_2 \cdot \|X\|_{2,\infty}^2 \cdot \|Y\|_{2,\infty}^2 \\
&\leq C\epsilon d_1 d_2 \alpha_1^2 \alpha_2^2 \\
&\leq C\epsilon d^2 \alpha^2.
\end{aligned}$$

The first inequality is an application of Lemma 4.6.4, the second inequality comes the shape of X and Y, the third inequality is due to the bound on $\|W - J\|_2$, and the last inequality comes from $d = \max(d_1, d_2)$ and the definition of $\mathcal{C}_1$ and $\mathcal{C}_2$.

We consider two separate cases:

Case 1: if $\left\| XY^\top \right\|_F^2 \geq \frac{C}{c}\epsilon d^2 \alpha^2$, then:

$$\left| \left\| XY^\top \right\|_W^2 - \left\| XY^\top \right\|_F^2 \right| \leq C\epsilon d^2 \alpha^2 \leq c \cdot \left\| XY^\top \right\|_F^2.$$

Case 2: otherwise $\left\| XY^\top \right\|_F^2 \leq \frac{C}{c}\epsilon d^2 \alpha^2$, then:

$$\left\| XY^\top \right\|_F^2 \leq \frac{C}{c}\epsilon d^2 \frac{\beta^2 r^2 \sigma_1^2}{d_1 d_2} = O\left(\beta^2 r^2 \sigma_1^2 \epsilon\right). \qquad \blacksquare$$

Lemma 4.5.2 makes use of Lemma 4.6.3 and Theorem 4.6.1 to provide an upper bound on the maximum value in W, which is an additional property of W to accompany Lemma 4.6.2.

Lemma 4.5.2. *Suppose the preprocessing step produces weight matrix W that achieves ϵ-spectral-similarity. Then we have $\|W\|_{\max} = O(\epsilon\sqrt{d_1 d_2})$.*

Proof. Let G denote a $d_1 \times d_2$ complete bipartite graph (corresponding to the full matrix), and H denote the semi-random graph generated by including each edge of G with probability at least p, i.e., edges in H correspond to the the observed indices in Ω. Apply weights W on H to get weighted graph $\widetilde{H}$. Let $L = D - A$ be the Laplacian of G, where D is the degree matrix and A is the adjacency matrix. Similarly write $\widetilde{L} = \widetilde{D} - \widetilde{A}$ for $\widetilde{H}$. Let $J \in \mathbb{R}^{d_1 \times d_2}$ be the all ones matrix, $I_1 \in \mathbb{R}^{d_1 \times d_1}, I_2 \in \mathbb{R}^{d_2 \times d_2}$ be identity matrices of corresponding dimensions, we have:

$$L = \begin{pmatrix} d_2 I_1 & -J \\ -J^\top & d_1 I_2 \end{pmatrix} \qquad \widetilde{L} = \begin{pmatrix} \widetilde{D_1} & -W \\ -W^\top & \widetilde{D_2} \end{pmatrix} \qquad \text{assuming } \widetilde{D} = \begin{pmatrix} \widetilde{D_1} & 0 \\ 0 & \widetilde{D_2} \end{pmatrix}$$

Following Theorem 4.6.1, $(1-\epsilon)L \preccurlyeq \widetilde{L} \preccurlyeq L$. For all $x \in \mathbb{R}^{(d_1+d_2)}$:

$$(1-\epsilon)x^\top L x \leq x^\top \widetilde{L} x \tag{4.6}$$

Consider $x = [\sqrt{d_1}e_j; \sqrt{d_2}e_k]$ where $e_j \in \mathbb{R}^{d_1}, e_k \in \mathbb{R}^{d_2}$ are standard basis vectors. We have the following quadratic forms:

$$x^\top Lx = d_1 d_2 I_{1,jj} + d_2 d_1 I_{2,kk} - 2\sqrt{d_1 d_2} J_{jk}$$

$$x^\top \widetilde{L} x = d_1 \widetilde{D_1}_{jj} + d_2 \widetilde{D_2}_{kk} - 2\sqrt{d_1 d_2} W_{jk}$$

By Lemma 4.6.3, $\widetilde{D_1}_{jj} \leq d_2$ and $\widetilde{D_2}_{kk} \leq d_1$, so the quadratic forms simplify to:

$$x^\top Lx = 2d_1 d_2 - 2\sqrt{d_1 d_2} \tag{4.7}$$

$$x^\top \widetilde{L} x \leq 2d_1 d_2 - 2\sqrt{d_1 d_2} W_{jk} \tag{4.8}$$

Combining (4.6) (4.7) and (4.8), we have:

$$\begin{aligned} 2d_1 d_2 - 2\epsilon d_1 d_2 - 2\sqrt{d_1 d_2} + 2\epsilon\sqrt{d_1 d_2} &\leq 2d_1 d_2 - 2\sqrt{d_1 d_2} W_{jk} \\ \implies W_{jk} &\leq \epsilon\sqrt{d_1 d_2} + 1 - \epsilon \\ &= O(\epsilon\sqrt{d_1 d_2}) \text{ since } \epsilon d = \omega(1) \end{aligned}$$

■

4.6 Relevant Results and Omitted Proofs

4.6.1 Relevant Previous Results

We include some relevant results from (Cheng and Ge, 2018) here. The preprocessing algorithm assigns weights to the edges of the semi-random graph H so that the resulting

weighted graph is ϵ-spectrally-similar to a complete graph G. For sufficiently large p, (Cheng and Ge, 2018) proves the existence of weights w_e^* with spectral similarity parameter $\epsilon_0 = O\left(\sqrt{\frac{\log d}{pd}}\right)$, so that we can Theorem 4.6.1 with L as the Laplacian of G, E as the edge set of H, and $\epsilon = \epsilon_0$.

Theorem 4.6.1 (Semi-random preprocessing, Theorem 3.1 (Cheng and Ge, 2018))**.** *Fix $\epsilon \geq 0$, $\epsilon_0 \leq 1/10$, and let L be the Laplacian of a $d_1 \times d_2$ bipartite graph. Given a set of m edges E, let vector $b_e \in \mathbb{R}^{d_1+d_2}$ represent an edge $e \in E$, where $e = (i, j)$ for some $i \in [d_1]$, $j \in [d_2]$, and $b_e[i] = 1, b_e[j] = -1$. Assume there exists weights $w_e^* \geq 0$ such that*

$$(1 - \epsilon_0)L \preccurlyeq \sum_{e \in E} w_e^* b_e b_e^\top \preccurlyeq (1 + \epsilon_0)L.$$

We can find a set of weights $w_e \geq 0$ in time $\widetilde{O}(m/\epsilon^{O(1)})$, such that with high probability,

$$(1 - O(\epsilon_0) - \epsilon)L \preccurlyeq \sum_{e \in E} w_e b_e b_e^\top \preccurlyeq L.$$

Lemma 4.6.2 (Spectral properties of the weight matrix, Corollary 3.4 (Cheng and Ge, 2018))**.** *For a matrix completion problem under the semi-random setup, given $\epsilon > 0$, there exist $p = O\left(\frac{\log d}{d\epsilon^2}\right)$ such that if each entry of X^* is observed with probability at least p, then with high probability we can compute a weight matrix $W \in \mathbb{R}^{d_1 \times d_2}$ in time $\widetilde{O}(m/\epsilon^{O(1)})$ that achieves ϵ-spectral-similarity, such that W is supported on Ω and $\|W\|_1 \leq d_1, \|W\|_\infty \leq d_2, \|W - J\|_2 = O(\epsilon\sqrt{d_1 d_2})$ where J is the all-ones matrix.*

Lemma 4.6.3 (Closeness of Laplacian and adjacency matrix, Lemma 3.6 (Cheng and Ge, 2018)). *Let $L = D - A$ and $\widetilde{L} = \widetilde{D} - \widetilde{A}$ be two graph Laplacians, where D, $\widetilde{D}$ are the degree matrices and A, $\widetilde{A}$ are the adjacency matrices of the graphs. If $(1-\epsilon)L \preccurlyeq \widetilde{L} \preccurlyeq L$, then we have $(1-\epsilon)D_{ii} \preccurlyeq \widetilde{D}_{ii} \preccurlyeq D_{ii}$ and $\left\|D^{-1/2}(\widetilde{A} - A)D^{-1/2}\right\|_2 \leq 3\epsilon$.*

Lemma 4.6.4 (Preserving the norm via spectral properties, Lemma 4.3 (Cheng and Ge, 2018)). *For any matrices $X \in \mathbb{R}^{d_1 \times r}$, $Y \in \mathbb{R}^{d_2 \times r}$ and $W \in \mathbb{R}^{d_1 \times d_2}$, we have*

$$\left| \left\| XY^\top \right\|_W^2 - \left\| XY^\top \right\|_F^2 \right| \leq \|W - J\|_2 \cdot \|X\|_F \cdot \|Y\|_F \cdot \|X\|_{2,\infty} \cdot \|Y\|_{2,\infty},$$

where J is the all-ones matrix.

4.6.2 Relevant Matrix Concentration Bounds

For completeness we include a lemma in contrast to Lemma 4.3.4, on the operator norm of the gradient (deviation bound) in the uniform case.

Lemma 4.6.5 (Lemma B.2 (Zhang et al., 2018b),(Negahban and Wainwright, 2012)). *Consider noisy matrix completion with the uniform observation model. Suppose the noise entry E_{jk} follows i.i.d zero mean distribution with variance s^2. Then with probability at least $1 - c_1/d$, we have*

$$\left\| \frac{1}{p} \sum_{(j,k)\in\Omega} E_{jk} e_j e_k^\top \right\|_2^2 \leq c_2 s^2 \frac{d \log d}{p},$$

where c_1 c_2 are universal constants and $p = \frac{|\Omega|}{d_1 d_2}$.

The following theorem provides a tail bound on the operator norm of a sum of random matrices.

Theorem 4.6.6 (Matrix Bernstein inequality, Theorem 1.6 (Tropp, 2012))**.** *Consider a finite sequence* $\{Z_k\} \subset \mathbb{R}^{d_1 \times d_2}$ *of independent random matrices. Assume that each random matrix satisfies* $\mathbb{E}[Z_k] = \mathbf{0}$ *and* $\|Z_k\|_2 \leq R$ *almost surely, define*

$$\sigma^2 := \max\left\{ \left\| \sum_k \mathbb{E}\left[Z_k Z_k^\top\right] \right\|_2, \left\| \sum_k \mathbb{E}\left[Z_k^\top Z_k\right] \right\|_2 \right\},$$

then $\forall t > 0$,

$$\Pr\left[\left\| \sum_k Z_k \right\|_2 \geq t \right] \leq (d_1 + d_2) \cdot \exp\left(\frac{-t^2}{2\sigma^2 + 2Rt/3} \right).$$

Note that as mentioned in (Negahban and Wainwright, 2012), according to (Vershynin, 2010) the same bound holds if each Z_k *is sub-exponential with* $R = \|Z_k\|_{\Psi_1}$*, the sub-exponential (Orlicz) norm of* Z_k*.*

4.6.3 Self-Contained Proof of Corollary 4.4.1

Proof of Corollary 4.4.1. Consider the weighted loss function and its gradient:

$$F_W(X) = \frac{1}{2} \sum_{(j,k) \in \Omega} W_{jk}(X_{jk} - Y_{jk})^2,$$

$$\nabla F_W(X) = \sum_{(j,k) \in \Omega} W_{jk}(X_{jk} - Y_{jk}) e_j e_k^\top.$$

In order to achieve the error bound of the primal-dual framework in Theorem 4.2.4, we first need to verify the RSC and RSS conditions (4.2.2). For all $X_1, X_2 \in \mathcal{C}$,

$$
\begin{aligned}
&F_W(X_1) - F_W(X_2) - \langle \nabla F_W(X_2), X_1 - X_2 \rangle \\
&= \frac{1}{2} \sum_{(j,k)\in\Omega} W_{jk}[(X_{1,jk} - Y_{jk})^2 - (X_{2,jk} - Y_{jk})^2] - \sum_{(j,k)\in\Omega} W_{jk}(X_{2,jk} - Y_{jk})(X_{1,jk} - X_{2,jk}) \\
&= \frac{1}{2} \sum_{(j,k)\in\Omega} W_{jk}[(X_{1,jk} + X_{2,jk} - 2Y_{jk})(X_{1,jk} - X_{2,jk}) - (2X_{2,jk} - 2Y_{jk})(X_{1,jk} - X_{2,jk})] \\
&= \frac{1}{2} \sum_{(j,k)\in\Omega} W_{jk}(X_{1,jk} - X_{2,jk})^2 \\
&= \frac{1}{2} \|X_1 - X_2\|_W^2 .
\end{aligned}
$$

By Lemma 4.5.1, either $\|X_1 - X_2\|_W^2 \le (1 \pm c)\|X_1 - X_2\|_F^2$ for some constant $c \le \frac{1}{40}$; or $\|X_1 - X_2\|_F^2 \le O(\beta^2 r^2 \sigma_1^2 \epsilon)$, and we take $X_1 = UV^\top$, $X_2 = X^*$ so that $\left\|UV^\top - X^*\right\|_F^2 = O(\beta^2 r^2 \sigma_1^2 \epsilon)$.

Consider the first case:

$$
(1 - c)\|X_1 - X_2\|_F^2 \le \|X_1 - X_2\|_W^2 \le (1 + c)\|X_1 - X_2\|_F^2,
$$

thus we can establish RSC and RSS conditions with $\mu = 39/40$ and $L = 41/40$.

Next we verify the deviation condition (4.2.3). Consider the gradient at X^*,

$$\begin{aligned}\nabla F_W(X^*) &= \sum_{(j,k)\in\Omega} W_{jk}(X^*_{jk} - Y_{jk})e_j e_k^\top \\ &= \sum_{(j,k)\in\Omega} W_{jk}E_{jk}e_j e_k^\top,\end{aligned}$$

where each E_{jk} is i.i.d with mean 0, variance $\frac{\nu^2}{d_1 d_2}$. According to Lemma 4.3.4, we have $\|\nabla F_W(X^*)\|_2^2 = O(\frac{\nu^2}{d_1 d_2}\epsilon d^2 \log d) = O(\nu^2 \epsilon \log d) =: \delta^2$.

We have established all the necessary conditions to apply the error bound in Theorem 4.2.4, which gives us $\left\|UV^\top - X^*\right\|_F^2 \le \Gamma r \delta^2$ where Γ is a constant depending on μ and L. Given sampling rate p, the preprocessing algorithm can achieve $\epsilon = \sqrt{\frac{\log d}{pd}}$ according to Lemma 4.6.2. Putting things together we have $\left\|UV^\top - X^*\right\|_F^2 = O\left(\nu^2\sqrt{\frac{r^2\log^3 d}{pd}}\right)$. Recall the error bound $\left\|UV^\top - X^*\right\|_F^2 = O(\beta^2 r^2 \sigma_1^2 \epsilon) = O\left(\beta^2\sigma_1^2\sqrt{\frac{r^4\log d}{pd}}\right)$ from the edge case in Lemma 4.5.1, combining both terms gives us the overall error bound. ■

4.6.4 Self-Contained Proof of Corollary 4.4.2

Proof of Corollary 4.4.2. Consider the weighted loss function and write it in its standardized form:

$$\begin{aligned}F_W(X) &= -\frac{1}{d_1 d_2}\sum_{(j,k)\in\Omega} W_{jk}\Big[\mathbb{1}(Y_{jk}=1)\cdot\log\big(f(X_{jk})\big) + \mathbb{1}(Y_{jk}=-1)\cdot\log\big(1-f(X_{jk})\big)\Big] \\ &= \frac{1}{d_1 d_2}\sum_{(j,k)\in\Omega} W_{jk}\Big[\mathbb{1}(Y_{jk}=1)\cdot\log\big(g(X_{jk}/\tau)\big) + \mathbb{1}(Y_{jk}=-1)\cdot\log\big(1-g(X_{jk}/\tau)\big)\Big].\end{aligned}$$

Compute its gradient and Hessian with respect to X:

$$\nabla F_W(X) = \frac{1}{d_1 d_2 \tau} \sum_{(j,k)\in\Omega} W_{jk} b_{jk}(X) e_j e_k^\top,$$
$$\nabla^2 F_W(X) = \frac{1}{d_1 d_2 \tau^2} \sum_{(j,k)\in\Omega} W_{jk} B_{jk}(X) \operatorname{vec}(e_j e_k^\top) \operatorname{vec}(e_j e_k^\top)^\top,$$

where e_j, e_k are standard basis vectors in $\mathbb{R}^{d_1}$, $\mathbb{R}^{d_2}$, and $\operatorname{vec}(\cdot)$ is the vectorization operator, and

$$b_{jk}(X) = -\mathbb{1}(Y_{jk} = 1) \cdot \frac{g'(X_{jk}/\tau)}{g(X_{jk}/\tau)} + \mathbb{1}(Y_{jk} = -1) \cdot \frac{g'(X_{jk}/\tau)}{1 - g(X_{jk}/\tau)},$$
$$B_{jk}(X) = \mathbb{1}(Y_{jk} = 1) \cdot \left(\frac{g'^2(X_{jk}/\tau)}{g^2(X_{jk}/\tau)} - \frac{g''(X_{jk}/\tau)}{g(X_{jk}/\tau)} \right) + \mathbb{1}(Y_{jk} = -1) \cdot \left(\frac{g''(X_{jk}/\tau)}{1 - g(X_{jk}/\tau)} + \frac{g'^2(X_{jk}/\tau)}{\left(1 - g(X_{jk}/\tau)\right)^2} \right).$$

Two parameters were introduced in (Ni and Gu, 2016) to control the quadratic lower and upper bounds of the second-order Taylor expansion of the log-likelihood function:

$$\mu_\rho = \min \left\{ \inf_{|x|\le\rho} \left(\frac{g'^2(x)}{g^2(x)} - \frac{g''(x)}{g(x)} \right), \inf_{|x|\le\rho} \left(\frac{g'^2(x)}{(1-g(x))^2} + \frac{g''(x)}{1-g(x)} \right) \right\},$$
$$L_\rho = \max \left\{ \sup_{|x|\le\rho} \left(\frac{g'^2(x)}{g^2(x)} - \frac{g''(x)}{g(x)} \right), \sup_{|x|\le\rho} \left(\frac{g'^2(x)}{(1-g(x))^2} + \frac{g''(x)}{1-g(x)} \right) \right\}.$$

Recall the signal to noise ratio $\rho = \frac{\alpha}{\tau}$ and $|X_{jk}/\tau| \le \rho$, so that for all indices (j,k), $\mu_\rho \le B_{jk}(X) \le L_\rho$.

We first need to verify the RSC and RSS conditions (4.2.2). For all $X_1, X_2 \in \mathcal{C}$, by applying the Mean Value Theorem to the second order remainder, $\exists M = tX_1 + (1-t)X_2$ for some $t \in [0,1]$ such that

$$\begin{aligned}
&F_W(X_1) - F_W(X_2) - \langle \nabla F_W(X_2), X_1 - X_2 \rangle \\
&= \frac{1}{2} \operatorname{vec}(X_1 - X_2)^\top \nabla^2 F_W(M) \operatorname{vec}(X_1 - X_2) \\
&= \frac{1}{2\nu^2} \sum_{(j,k)\in\Omega} W_{jk} B_{jk}(M) \langle e_j e_k^\top, X_1 - X_2 \rangle^2 \\
&= \frac{1}{2\nu^2} \|X_1 - X_2\|_{W\odot B}^2 ,
\end{aligned}$$

where $W \odot B \in \mathbb{R}^{d_1 \times d_2}$ such that $(W \odot B)_{jk} = W_{jk} B_{jk}(M)$.

By Lemma 4.5.1, either $\|X_1 - X_2\|_W^2 \le (1 \pm c)\|X_1 - X_2\|_F^2$ for some constant $c \le \frac{1}{40}$; or $\|X_1 - X_2\|_F^2 \le O(\beta^2 r^2 \sigma_1^2 \epsilon)$, and we take $X_1 = UV^\top$, $X_2 = X^*$ so that $\left\|UV^\top - X^*\right\|_F^2 = O(\beta^2 r^2 \sigma_1^2 \epsilon)$.

Consider the first case:

$$\begin{aligned}
(1-c)\|X_1 - X_2\|_F^2 \le \|X_1 - X_2\|_W^2 \le (1+c)\|X_1 - X_2\|_F^2, \\
\frac{(1-c)\mu_\rho}{2\nu^2}\|X_1 - X_2\|_F^2 \le \frac{1}{2\nu^2}\|X_1 - X_2\|_{W\odot B}^2 \le \frac{(1+c)L_\rho}{2\nu^2}\|X_1 - X_2\|_F^2,
\end{aligned}$$

therefore we can establish RSC and RSS conditions with parameters $\mu = (1-c)\mu_\rho/\nu^2$ and $L = (1+c)L_\rho/\nu^2$.

Next we verify the deviation condition (4.2.3).

Consider $\nabla F_W(X^*) = \frac{1}{d_1 d_2 \tau} \sum_{(j,k)\in\Omega} W_{jk} b_{jk}(X^*) e_j e_k^\top$ where

$$b_{jk}(X^*) = \begin{cases} -\frac{g'(X^*_{jk}/\tau)}{g(X^*_{jk}/\tau)} & \text{with probability } g(X^*_{jk}/\tau), \\ \frac{g'(X^*_{jk}/\tau)}{1-g(X^*_{jk}/\tau)} & \text{with probability } 1 - g(X^*_{jk}/\tau). \end{cases}$$

Therefore $\mathbb{E}[b_{jk}(X^*)] = 0$. Next we compute the variance:

$$\begin{aligned} \mathrm{V}ar[b_{jk}(X^*)] &= \mathbb{E}\left[b_{jk}^2(X^*)\right] - \mathbb{E}[b_{jk}(X^*)]^2 \\ &= \left[\frac{g'(X^*_{jk}/\tau)}{g(X^*_{jk}/\tau)}\right]^2 \cdot g(X^*_{jk}/\tau) + \left[\frac{g'(X^*_{jk}/\tau)}{1-g(X^*_{jk}/\tau)}\right]^2 \cdot (1 - g(X^*_{jk}/\tau)) - 0 \\ &= \frac{g'^2(X^*_{jk}/\tau)}{g(X^*_{jk}/\tau)(1-g(X^*_{jk}/\tau))} \\ &= \left(\frac{|g'(X^*_{jk}/\tau)|}{g(X^*_{jk}/\tau)(1-g(X^*_{jk}/\tau)}\right)^2 \cdot g(X^*_{jk}/\tau)(1-g(X^*_{jk}/\tau)) \\ &\le s_\rho^2 \end{aligned}$$

The last step follows from the definition of s_ρ and the fact that $g(\cdot)$ is a cumulative distribution function and its range is $[0,1]$. By Lemma 4.3.4, $\|\nabla F_W(X^*)\|_2^2 = \frac{1}{d_1^2 d_2^2 \tau^2} \cdot O(s_\rho^2 \epsilon d^2 \log d) = \frac{1}{d_1 d_2 \nu^2} \cdot O(s_\rho^2 \epsilon d^2 \log d) = O(s_\rho^2 \epsilon \log d/\nu^2) =: \delta^2$.

We have established all the conditions required for Theorem 4.2.4 which gives the error bound $\left\|UV^\top - X^*\right\|_F^2 \le \Gamma r \delta^2$. Based on our choice of μ and L, $\Gamma = \Gamma' \nu^2$ where Γ' is a constant depending on μ_ρ and L_ρ which are constants given standardized distribution function g and the dimension free signal to noise ratio ρ. Given sampling rate p,

the preprocessing algorithm can achieve $\epsilon = \sqrt{\frac{\log d}{pd}}$ according to Lemma 4.6.2. Putting things together we have $\left\|UV^\top - X^*\right\|_F^2 \leq O\left(s_\rho^2 \sqrt{\frac{r^2 \log^3 d}{pd}}\right)$. Recall the error bound $\left\|UV^\top - X^*\right\|_F^2 = O(\beta^2 r^2 \sigma_1^2 \epsilon) = O\left(\beta^2 \sigma_1^2 \sqrt{\frac{r^4 \log d}{pd}}\right)$ from the edge case in Lemma 4.5.1, combining both terms gives us the overall error bound. ■

CHAPTER 5

ROBUST MATRIX SENSING IN THE SEMI-RANDOM MODEL

This chapter was previously published as Robust Matrix Sensing in the Semi-Random Model by Xing Gao and Yu Cheng (Gao and Cheng, 2024).

5.1 Introduction

Low-rank matrix recovery is a popular inverse problem with many applications in machine learning such as collaborative filtering, image compression, and robust principal component analysis (PCA) (Rennie and Srebro, 2005; Fazel et al., 2008; Candès et al., 2011). In this paper, we study one of the most basic low-rank matrix recovery problems namely matrix sensing (Candes et al., 2006; Recht et al., 2010). In the matrix sensing problem, we want to reconstruct a low-rank ground-truth matrix $X^* \in \mathbb{R}^{d_1 \times d_2}$ from a collection of sensing matrices $\{A_i\}_{i=1}^n$ and the corresponding linear measurements $b_i = \langle A_i, X \rangle$.

For notational convenience, we define a sensing operator $\mathcal{A}[\cdot] : \mathbb{R}^{d_1 \times d_2} \to \mathbb{R}^n$ such that $\mathcal{A}[X] = b$ with $b_i = \langle A_i, X \rangle$ for $i = 1 \dots n$. The goal is to solve the following rank-constrained optimization problem:

$$\min_{X \in \mathbb{R}^{d_1 \times d_2}} \|\mathcal{A}[X] - b\|_2^2 \text{ subject to } \operatorname{rank}(X) \leq r.$$

As optimizing over low-rank matrices are often computationally hard, one common approach is to replace the non-convex low-rank constraint with its convex-relaxation, which results in the following nuclear norm minimization problem (Recht et al., 2010):

$$\min_{X \in \mathbb{R}^{d_1 \times d_2}} \|X\|_* \text{ subject to } \mathcal{A}[X] = b. \tag{5.1}$$

Another widely-used approach in practice is to consider the unconstrained non-convex factorized parametrization (Recht et al., 2010; Ge et al., 2017; Bhojanapalli et al., 2016):

$$\min_{U \in \mathbb{R}^{d_1 \times r}, V \in \mathbb{R}^{d_2 \times r}} \left\| \mathcal{A}[UV^\top] - b \right\|_2^2. \tag{5.2}$$

and solve it with some form of gradient descent or alternating minimization.

Existing convex and non-convex approaches all rely on certain assumptions. A standard assumption in the literature is that the sensing matrices satisfy the **Restricted Isometry Property (RIP)**, which means that the sensing matrices approximately preserve the norm of a low-rank matrix. (Formally, $\frac{1}{L} \cdot \|X\|_F^2 \leq \frac{1}{n} \sum_{i=1}^n \langle A_i, X \rangle^2 \leq L \cdot \|X\|_F^2$ given $\text{rank}(X) \leq r$ for some parameters r and L.)

In this paper, we relax the RIP condition on the sensing matrices and study a robust version of the problem, which is often referred to as the **semi-random** model. More specifically, an adversary "corrupts" the input data by providing any number of additional sensing matrices A_i that are adversarially chosen, but the corresponding measurements

$b_i = \langle A_i, X^* \rangle$ remain consistent with the ground truth matrix X^*. Consequently, only a subtset of the sensing matrices satisfy the RIP condition and the rest of them are arbitrary. This is an intermediate scenario between the average case and the worst case, which arises more frequently in practice.

To the best of our knowledge, we are the first to study the matrix sensing problem in this semi-random model. Formally, we consider the following adversary: suppose that originally there was a collection of RIP sensing matrices $\{A_i\}_{i=1}^m$ ("good" matrices), then the adversary augmented some arbitrary $\{A_i\}_{i=m+1}^n$ ("bad" matrices) and then shuffled all the sensing matrices. The algorithm is then given the measurements based on the "good" and "bad" matrices together. The combined sensing matrices are no longer guaranteed to satisfy the RIP condition, but there exists a subset (indicated by an indicator vector w^*) that does, i.e., $\frac{1}{L} \cdot \|X\|_F^2 \leq \sum_{i=1}^n w_i^* \langle A_i, X \rangle^2 \leq L \cdot \|X\|_F^2$, where $w_i^* = \frac{1}{m}$ on the original "good" matrices and $w_i^* = 0$ on the "bad" matrices added by the adversary. In general, the subset may be replaced by a convex combination and the indicator vector by a simplex. Inspired by the adversary for semi-random vector regression in (Kelner et al., 2023a), we refer to this condition as **weighted RIP (wRIP)** and formally define it in Definition 5.2.2.

Since the wRIP condition is a more general assumption than RIP, existing solutions that rely on RIP might fail under the semi-random model with wRIP condition. As stated in (Kelner et al., 2023a), this type of adversary does not break the problem from an "information-theoretic perspective", but affects the problem computationally. In partic-

ular, existing non-convex approaches for matrix sensing (e.g., 5.2) may get stuck at bad local minima as the RIP condition is necessary for proving landscape results regarding the non-convex objective (see, e.g., the counter-examples provided in (Bhojanapalli et al., 2016). The convex relaxation approach (5.1) does continue to work in the semi-random model, because the augmented linear measurements are consistent with the ground-truth matrix X^* which simply provides additional optimization constraints. However, convex approaches are often less desirable in practice and can become computationally prohibitive when $d_1, d_2 > 100$ as pointed out in (Recht et al., 2010).

5.1.1 Our Contributions

The limitations of existing algorithms motivate us to pose and study the problem of semi-random matrix sensing in this paper. We summarize our main contributions below:

- **Pose and study matrix sensing in the semi-random model.** We introduce the more general wRIP condition on matrix sensing as a relaxation of the typical RIP assumption, and provide a solution that is more robust to input contamination. Our work will serve as a starting point for the design of more efficient robust algorithms for matrix sensing, as well as other low-rank matrix problems, in the semi-random model.
- **Design an efficient robust algorithm for semi-random matrix sensing.** Our algorithm is guaranteed to converge to a global optimum which improves on the existing non-convex solution (Bhojanapalli et al., 2016) that can get stuck in bad local optima in the semi-random model, while achieving a comparable running time as existing con-

vex solution (Recht et al., 2010), informally stated in Theorem 5.1.1 below. A formal statement can be found in Theorem 5.3.1.

- **Adapt a reweighting scheme for semi-random matrix sensing.** In contrast to the non-convex approach that failed and the convex approach that avoided the challenge posed by the adversary altogether, we study a new approach that directly targets the semi-random adversary instead. We develop an algorithm using an iterative reweighting approach inspired by (Kelner et al., 2023a): in each iteration, the algorithm reweights the sensing matrices to combat the effect of the adversary and then takes a weighted gradient step that works well based on the current solution.
- **Exploit the connection between sparsity and low-rankness.** Observing a duality between sparse vectors and low-rank matrices, we draw a parallel between linear regression and matrix sensing problems. By exploring the structural similarities and differences between vector and matrix problems, we are able to extend and generalize the work of (Kelner et al., 2023a) on semi-random sparse vector recovery to the higher dimensional problem of semi-random matrix sensing. We emphasize that even though the generalization from vector to matrix problems is natural, the analysis behind the intuition is often nontrivial and involves different mathematical tools.

We state a simplified version of our main algorithmic result assuming Gaussian design. The more general result is stated as Theorem 5.3.1 in Section 5.3.

Theorem 5.1.1 (Semi-random matrix sensing)**.** *Suppose the ground-truth matrix* $X^* \in \mathbb{R}^{d_1 \times d_2}$ *satisfies* $\mathrm{rank}(X^*) \le r$ *and* $\|X^*\|_F \le \mathrm{poly}(d)$*. Let* $A_1, \ldots, A_n$ *be the sensing matrices and let* $b_i = \langle A_i, X^* \rangle$ *be the corresponding measurements. Suppose there exists a (hidden) set of* $\Omega(dr)$ *sensing matrices with i.i.d. standard Gaussian entries, and the remaining sensing matrices are chosen adversarially, where* $d = \max(d_1, d_2)$*.*

There exists an algorithm that can output $X \in \mathbb{R}^{d_1 \times d_2}$ *such that* $\|X - X^*\|_F \le \epsilon$ *with high probability in time* $\widetilde{O}(nd^{\omega+1} r \log(1/\epsilon))$ [1] *where* $\omega < 2.373$ *is the matrix multiplication exponent.*

5.1.2 Overview of Our Techniques

Since there exists a subset (or a convex combination in general) of the sensing matrices that satisfy the RIP condition, a natural strategy is to reverse the effect from the adversary by reweighting the sensing matrices so that they satisfy the RIP condition. However, it is NP-hard to verify RIP condition on all low-rank inputs, so it is unclear how to preprocess and "fix" the input in the beginning and then apply existing solutions to matrix sensing. Instead, we make a trade-off between the frequency of reweighting and the requirement on the weights by adopting an iterative reweighting approach: in each iteration, we only aim to find a set of weights so that the weighted matrices satisfy some desirable properties (not necessarily RIP) with respect to the current estimate X (as opposed to all low-rank matrices).

---

[1] *Throughout the paper, we write* $\widetilde{O}(f(n))$ *for* $O(f(n) \,\mathrm{polylog}\, f(n))$ *and similarly for* $\widetilde{\Omega}(\cdot)$*.*

Inspired by the workflow in (Kelner et al., 2023a), our **semi-random matrix sensing algorithm** (Algorithm 5.1) repeatedly calls a halving algorithm to reduce the error of our estimate arbitrarily small. The **halving algorithm** (Algorithm 5.2) contracts the upper bound on $\|X - X^*\|_F$, which is the error between our current estimate X and the ground truth X^*, each time it is run. Inside this algorithm is a gradient descent style loop, where in each iteration we try to minimize a weighted objective function, which is essentially the weighted ℓ_2-norm of $\mathcal{A}[X_t] - b$ (the distance to X^* "measured" by the sensing matrices), where the weights are provided by an oracle implemented in Algorithm 5.3. The algorithm proceeds by taking a step opposite to the gradient direction, and the step is then projected onto a nuclear-norm-bounded ball which is necessary for the weight oracle to continue working in the next step. As we mentioned before, the weights from the oracle need to satisfy some nice properties with respect to the current iteration estimate X_t. Ideally, the property should: firstly, ensure the gradient step makes enough progress towards X^*; secondly, can be derived from the wRIP condition so that we know such a requirement is feasible; and lastly, be easily verifiable as opposed to the NP-hard RIP condition.

With the first requirement in mind, we define the **weight oracle** as in Definition 5.2.5, adapted from the vector version in (Kelner et al., 2023a). The oracle output should satisfy two properties, namely the progress and decomposition guarantees, and together they ensure the gradient step makes good enough progress toward X^*. Intuitively speaking, the progress guarantee ensures the gradient step is large in the direction parallel to the "actual" deviation $X - X^*$ (as opposed to only reducing the "measured" deviation $\mathcal{A}[X] - b$) and

thus will make significant progress, while the decomposition guarantee ensures the gradient step has small contribution and effect in other directions thus will not cancel the progress after the projection. While the progress guarantee is quantified as an inner product, we introduce a concept called "norm-decomposition" (Definition 5.2.4), the matrix analogy of the "short-flat-decomposition" of vectors (Kelner et al., 2023a), to capture the decomposition guarantee, and we will provide more details later. For the second requirement, we can loosely relate the two oracle guarantees to the wRIP condition: the (large) progress guarantee makes use of the lower bound in wRIP condition $\sum_{i=1}^{n} w_i^* \langle A_i, \frac{X}{\|X\|_F} \rangle^2 \geq \frac{1}{L}$, and the (small) decomposition guarantee makes use of the upper bound $\sum_{i=1}^{n} w_i^* \langle A_i, \frac{X}{\|X\|_F} \rangle^2 \leq L$. We introduce a condition called **decomposable wRIP (dRIP)** (Definition 5.2.3) to formally capture this relation, and we will show that it follows from the wRIP condition thus we can achieve such an oracle. Lastly, we will show that the oracle properties can be easily verified, meeting our third requirement.

A formal statement and a road map that leads to our main result can be found in Section 5.3.

5.1.3 Related Work

Matrix sensing (RIP). There are two main types of existing solutions. The convex-relaxation formulation 5.1 of the problem can be posed as a semidefinite program via the standard form primal-dual pair (Recht et al., 2010), where the primal problem has a $(d_1 + d_2)^2$ semidefinite constraint and n linear constraints. State of the art SDP solver (Jiang et al., 2020) requires running time of $\widetilde{O}(nd^{2.5})$ where $d = \max(d_1, d_2)$. The other

approach uses non-convex formulation 5.2 to reduce the size of the decision variable from d^2 to dr, improving computational efficiency. It is shown in (Bhojanapalli et al., 2016) that there are no spurious local minima given RIP sensing matrices and incoherent linear measurements in the non-convex approach, however, it is no longer applicable in the semi-random model.

Semi-random model. First introduced by (Blum and Spencer, 1995), the semi-random model has been studied for various graph problems (Feige and Kilian, 2001; Perry and Wein, 2017; Mathieu and Schudy, 2010; Makarychev et al., 2012). Previously the work of (Cheng and Ge, 2018) applied the semi-random model to the matrix completion problem, and recently (Kelner et al., 2023a) studied sparse vector recovery in this model.

Semi-random matrix completion. Low-rank matrix problems such as matrix completion and matrix sensing have similar optimization landscapes (Ge et al., 2017), thus development in one often lends insight to another. Prior work (Cheng and Ge, 2018) on the closely-related problem of matrix completion under the semi-random model showed that all bad local optima can be eliminated by reweighting the input data via a preprocessing step. It exploits the connection between the observation data matrix and the Laplacian matrix of a complete bipartite graph, and gives a reweighting algorithm to preprocess the data in a black-box manner. However, the analogous approach for matrix sensing requires reweighting a set of matrices to satisfy RIP, which is a condition that is NP-hard to check, thus is not practical in the matrix sensing problem.

Semi-random linear regression. In order to overcome the barrier of the reweighting or preprocessing approach mentioned earlier, we take inspiration from the work of (Kelner et al., 2023a) on sparse vector recovery under the semi-random model. One of their main contributions is the "short-flat decomposition", which is a property that can be efficiently verified for a given vector (locally), instead of verifying the RIP condition for all sparse vectors (globally). They provide a projected gradient descent style algorithm, where the rows of the sensing matrix are reweighted differently in each iteration to ensure a "short-flat decomposition" exists for the gradient. We draw a parallel between the problem of sparse vector regression and low-rank matrix sensing, and extend their work on linear regression of sparse vectors to the more generalized problem of sensing low-rank matrices.

5.2 Preliminaries

5.2.1 Notation

Throughout the paper, we denote the ground-truth low-rank matrix as X^*. We assume $X^* \in \mathbb{R}^{d_1 \times d_2}$, $\text{rank}(X^*) = r$, and d_1, d_2 have the same order of magnitude. Let $d = \max(d_1, d_2)$.

We write $[n]$ for the set of integers $\{1, ..., n\}$. We use Δ^n for the nonnegative probability simplex in dimension n, and $\mathbb{R}^n_{\geq 0}$ for the set of vectors with nonnegative coordinates in $\mathbb{R}^n$. For a vector x, we denote its ℓ_1, ℓ_2, and ℓ_∞-norms as $\|x\|_1$, $\|x\|_2$ and $\|x\|_\infty$ respectively, and write the i^{th} coordinate in x as x_i. For a matrix A, we use $\|A\|_*$, $\|A\|_2$, and $\|A\|_F$ for the nuclear, spectral (operator), and Frobenius norms of A respectively. For a matrix A,

we use $A_{(k)} = \text{argmin}_{\text{rank}(A')\leq k} \|A - A'\|_F$ to denote the best rank-k approximation of A; or equivalently, given the SVD of $A = \sum_{i=1}^{r} \sigma_i u_i v_i^\top$, we have $A_{(k)} = \sum_{i=1}^{k} \sigma_i u_i v_i^\top$ where $\sigma_1, ..., \sigma_k$ are the top k singular values of A.

We write $\text{tr}(A)$ for the trace of a square matrix A. For matrices $A, B \in \mathbb{R}^{d_1 \times d_2}$, we write $\langle A, B \rangle$ for their entrywise inner product $\langle A, B \rangle = \langle A, B \rangle = \text{tr}(A^\top B) = \sum_{j,k} A_{jk} B_{jk}$. A symmetric matrix $A \in \mathbb{R}^{d \times d}$ is positive semidefinite (PSD) if and only if $A = U^\top U$ for some matrix U, and we write $A \preccurlyeq B$ if A and B have the same dimension and $B - A$ is positive semidefinite. We write $\exp(A)$ as the matrix exponential of A; if A is diagonalizable as $A = UDU^{-1}$ then $\exp(A) = U \exp(D) U^{-1}$.

5.2.2 Definitions

We formally define the matrix sensing operator and observation vector below.

Definition 5.2.1 (Matrix Sensing Operator)**.** *Given a collection of sensing matrices $\mathcal{A} = \{A_i\}_{i=1}^n \subset \mathbb{R}^{d_1 \times d_2}$, we define the sensing operator $\mathcal{A}[\cdot] : \mathbb{R}^{d_1 \times d_2} \to \mathbb{R}^n$ such that $\mathcal{A}[X] = b$ where $b_i = \langle A_i, X \rangle$ for $X \in \mathbb{R}^{d_1 \times d_2}$.*

In other words, we have $b := \sum_{i=1}^{n} \langle A_i, X \rangle e_i$ where e_i is the i^{th} standard basis vector in $\mathbb{R}^n$. Throughout the paper, we use either $\mathcal{A}$ or $\{A_i\}_{i=1}^n$ to represent the sensing matrices.

To consistently recover a rank-r matrix in general, the number of measurements n should be at least $(d_1 + d_2 - r)r$ (Candes and Plan, 2011), hence we assume $n = \widetilde{\Omega}(dr)$ where $\widetilde{\Omega}$ suppresses log factors. In most matrix sensing literature, it is standard to impose the Restricted Isometry Property (RIP) condition on the sensing matrices. The RIP condition

states that $\mathcal{A}[\cdot]$ is approximately an isometry on low-rank matrices, which means the ℓ_2-norm of the observation vector is close to the Frobenius norm of X^*.

In this paper, we consider a semi-random model and relax the RIP condition as follows: we require that there exist weights $\{w_i^*\}_{i=1}^n$ (or $w^* \in \Delta^n$) so that the weighted sensing matrices $\{\sqrt{w_i^*}A_i\}_{i=1}^n$ satisfy the RIP condition. We call this relaxed assumption the wRIP (weighted RIP) condition, following the convention in (Kelner et al., 2023a).

We formally define RIP and wRIP conditions below.

Definition 5.2.2 (RIP and wRIP Conditions)**.** *We say a collection of sensing matrices $\mathcal{A} = \{A_i\}_{i=1}^n \subset \mathbb{R}^{d_1 \times d_2}$ satisfies the **RIP** (Restricted Isometry Property) condition with parameters r, L, and ρ if the following conditions hold for all $X \in \mathbb{R}^{d_1 \times d_2}$ with $\mathrm{rank}(X) \leq r$:*

1. *Boundedness: $\|A_i\|_2 \leq \rho$;*
2. *Isometry: $\frac{1}{L} \cdot \|X\|_F^2 \leq \frac{1}{n}\sum_{i=1}^n \langle A_i, X\rangle^2 \leq L \cdot \|X\|_F^2$.*

*Further, we say $\mathcal{A} = \{A_i\}_{i=1}^n$ satisfies the **wRIP** (weighted RIP) condition with parameters r, L, ρ, if $\exists w^* \in \Delta^n$ such that the following conditions hold for all $X \in \mathbb{R}^{d_1 \times d_2}$ with $\mathrm{rank}(X) \leq r$:*

1. *Boundedness: $\|A_i\|_2 \leq \rho$;*
2. *Isometry: $\frac{1}{L} \cdot \|X\|_F^2 \leq \sum_{i=1}^n w_i^* \langle A_i, X\rangle^2 \leq L \cdot \|X\|_F^2$.*

Notice that wRIP is a relaxation of the RIP condition, because we can choose $w_i^* = 1/n$ for all i in the standard RIP setting. More importantly, wRIP is strictly weaker. For

example, wRIP allows a (possibly majority) fraction of the sensing matrices to be chosen adversarially. We want to emphasize that the algorithm does not know w^* — one of the main challenges of semi-random matrix sensing is that finding w^* seems computationally hard, because it is NP-hard to check the RIP condition.

For presenting our algorithm and analysis, we introduce a variant of the wRIP condition called dRIP (decomposable-wRIP), inspired by Assumption B.1 in (Kelner et al., 2023a).

Definition 5.2.3 (dRIP Condition)**.** *We say a collection of sensing matrices $\mathcal{A} = \{A_i\}_{i=1}^n \subset \mathbb{R}^{d_1 \times d_2}$ satisfies the **dRIP** (decomposable wRIP) condition if $\exists w^* \in \Delta^n$ and constants $L, K, r, \rho \geq 1$, such that for all $V \in \mathbb{R}^{d_1 \times d_2}$ satisfying $\|V\|_F \in [\frac{1}{4}, 1], \|V\|_* \leq 2\sqrt{2r}$:*

1. *Boundedness: $\|A_i\|_2 \leq \rho$;*
2. *Isometry:*

   $\frac{1}{L} \leq \sum_{i=1}^n w_i^* \langle A_i, V \rangle^2 \leq L$, *equivalently* $\frac{1}{L} \leq \sum_{i=1}^n w_i^* u_i^2 \leq L$ *where* $u_i = \langle A_i, V \rangle$;

3. *Decomposability:*

   $\exists (L, \frac{1}{K\sqrt{r}})$*-norm-decomposition of* $G^* = \sum_{i=1}^n w_i^* \langle A_i, V \rangle A_i = \sum_{i=1}^n w_i^* u_i A_i$.

Definition 5.2.4 (Norm Decomposition)**.** *We say a matrix G has a (C_F, C_2)-norm-decomposition if $\exists S$ and E s.t. $G = S + E$, and $\|S\|_F \leq C_F$, $\|E\|_2 \leq C_2$.*

The main difference with wRIP is that dRIP requires the additional "decomposition" property. Observe that G^* is the (weighted) gradient at the point V. At a high level, we would like to decompose the gradient into two matrices, one with small Frobenius norm

and the other one with small operator norm. Our matrix norm-decomposition is inspired by the "short-flat-decomposition" for vectors in (Kelner et al., 2023a).

In Section 5.4, we will explain the motivation behind the norm decomposition as well as how to efficiently verify such a decomposition exists. We will also show that the dRIP condition is closely related to wRIP (by choosing parameters within a constant factor of each other) in Section 5.6.3.

A crucial component in our algorithm is a weight oracle inspired by (Kelner et al., 2023a) that produces a nonnegative weight on each sensing matrix (the weights are in general different in each iteration), such that the weighted gradient step moves the current solution closer to X^*. The oracle should output weights that satisfy certain properties which we term progress and decomposition guarantees. The purpose of these two guarantees is further explained in the proof of Lemma 5.4.2 in Section 5.6.1.

Definition 5.2.5 (Weight Oracle)**.** *We say an algorithm $\mathcal{O}$ is a (C_{prog}, C_F)-oracle, if given as input n matrices $\mathcal{A} = \{A_i\}_{i=1}^n \subset \mathbb{R}^{d_1 \times d_2}$ and an vector $u = \mathcal{A}[V] \in \mathbb{R}^n$ where $V \in \mathbb{R}^{d_1 \times d_2}$, $\|V\|_F \in [\frac{1}{4}, 1]$, and $\|V\|_* \leq 2\sqrt{2r}$, the algorithm $\mathcal{O}(\mathcal{A}, u, \delta)$ returns a weight vector $w \in \mathbb{R}^n_{\geq 0}$ such that the following conditions hold with probability at least $1 - \delta$:*

1. *Progress guarantee: $\sum_{i=1}^n w_i u_i^2 \geq C_{prog}$;*
2. *Decomposition guarantee: $\exists (C_F, \frac{C_{prog}}{6\sqrt{r}})$ norm-decomposition of $G = \sum_{i=1}^n w_i u_i A_i$.*

Note that the progress guarantee is equivalent to $\langle G, V \rangle \geq C_{prog}$.

Finally we define numerical rank which we use in our analysis. Numerical rank serves as a lower bound for the rank of a matrix based on its nuclear norm and Frobenius norm. That is, we always have $\mathrm{Rank}_{\mathrm{n}}(A) \leq \mathrm{rank}(A)$.

Definition 5.2.6 (Numerical Rank)**.** *The numerical rank of A is* $\mathrm{Rank}_{\mathrm{n}}(A) = \frac{\|A\|_*^2}{\|A\|_F^2}$.

5.3 Semi-Random Matrix Sensing

In this section, we present our main algorithm (Algorithm 5.1) for semi-random matrix sensing. With high probability Algorithm 5.1 recovers the ground-truth matrix X^* to arbitrary accuracy, formally stated in Theorem 5.3.1.

Algorithm 5.1: SemiRandomMatrixSensing$(R_0, \epsilon, \delta, \mathcal{A}, b)$

1: Input: $R_0 \geq \|X^*\|_F, b = \mathcal{A}[X^*], \epsilon > 0, \delta \in (0,1)$;
2: Output: X_{out} s.t. $\|X_{\mathrm{out}} - X^*\|_F \leq \epsilon$.
3: $X_0 \leftarrow 0, T \leftarrow \log \frac{R_0}{\epsilon}, \delta' \leftarrow \frac{\delta}{T}, R \leftarrow R_0$;
4: **for** $0 \leq t \leq T$ **do**
5: $\quad X_{t+1} \leftarrow \mathrm{HalveError}(X_t, R, \mathcal{O}, \delta', \mathcal{A}, b)$, $R \leftarrow \frac{R}{2}$;
6: **end for**
7: Return $X_{\mathrm{out}} \leftarrow X_T$;

Theorem 5.3.1 (Matrix sensing under wRIP)**.** *Suppose the ground-truth matrix $X^* \in \mathbb{R}^{d_1 \times d_2}$ satisfies* $\mathrm{rank}(X^*) \leq r$ *and* $\|X^*\|_F \leq R_0$. *Suppose the sensing matrices $\mathcal{A} = (A_i \in \mathbb{R}^{d_1 \times d_2})_{i=1}^n$ satisfy (r, L, ρ)-wRIP (as in Definition 5.2.2). Let $b = \mathcal{A}[X^*] \in \mathbb{R}^n$ be the corresponding measurements. For any $\epsilon, \delta > 0$, Algorithm 5.1 can output $X \in \mathbb{R}^{d_1 \times d_2}$ such that $\|X - X^*\|_F \leq \epsilon$ with probability at least $1 - \delta$. Algorithm 5.1 runs in time*

$O(nd^{\omega}\,\mathrm{polylog}\,(d)\log(\frac{L}{\delta}\log\frac{R_0}{\epsilon})r\rho^2L^4\log\frac{R_0}{\epsilon})$ *where* $d=\max(d_1,d_2)$ *and* $\omega<2.373$ *is the matrix multiplication exponent.*

Theorem 5.1.1 is a direct corollary of Theorem 5.3.1 under Gaussian design.

Proof of Theorem 5.1.1. When there are $\Omega(dr)$ sensing matrices with i.i.d. standard Gaussian entries, the input sensing matrices satisfy (r,L,ρ)-wRIP for $L=O(1)$ and $\rho=O(d^{1/2})$ with probability at least $1-\frac{1}{\mathrm{poly}(d)}$. This follows from a standard proof for RIP and the fact that we can ignore any sensing matrices with $\|A_i\|_2\gg d^{1/2}$. We assume that the wRIP condition is satisfied.

By Theorem 5.3.1, when $L=O(1)$, $\rho=O(d^{1/2})$, $R_0=\mathrm{poly}(d)$ and $\delta=\frac{1}{\mathrm{poly}(d)}$, Algorithm 5.1 can output a solution X such that $\|X-X^*\|_F\leq\epsilon$ with high probability. The runtime of Algorithm 5.1 can be simplified to $\widetilde{O}(nd^{\omega+1}r\log(1/\epsilon))$. ■

We first provide a road map for our analysis for proving Theorem 5.3.1:

- Our main algorithm runs a "halving" subroutine for $\log\frac{R_0}{\epsilon}$ iterations to reduce the error to ϵ. Each call to this subroutine reduces the upper bound on the distance between the current solution and the ground truth X^* by half. This halving subroutine runs in time $O(nd^{\omega}\,\mathrm{polylog}\,(d)\log(\frac{L}{\delta}\cdot\log\frac{R_0}{\epsilon})r\rho^2L^4)$ according to Lemmas 5.4.2 and 5.4.3.
- In Section 5.4, we present the halving algorithm (Algorithm 5.2). It depends on a $(\Omega(1),O(1))$-oracle, and Lemma 5.4.1 shows that the oracle guarantees can be easily verified. The algorithm's correctness and running time are analyzed in Lemma 5.4.2 and Lemma 5.4.3.

- In Section 5.5 we present the weight oracle required by the halving algorithm. We first show in Lemma 5.5.1 that the wRIP condition implies that the sensing matrices satisfy the dRIP condition tailored to the design of the oracle. Then we present an implementation of the oracle in Algorithm 5.3 based on the dRIP condition, and analyze its correctness and running time in Lemma 5.5.3 and Lemma 5.5.6.

5.4 Algorithm for Halving the Error

In this section, we present Algorithm 5.2 (HalveError). Algorithm 5.2 takes an estimate X_{in} with $\|X_{\text{in}} - X^*\|_F \leq R$ and outputs X_{out} such that $\|X_{\text{out}} - X^*\|_F \leq \frac{R}{2}$. This is the matrix version of the HalfRaidusSparse (Kelner et al., 2023a) algorithm for vectors.

Algorithm 5.2: HalveError$(X_{\text{in}}, R, \mathcal{O}, \delta, \mathcal{A}, b)$ (Kelner et al., 2023a)

Input: Rank-r matrix $X_{\text{in}} \in \mathbb{R}^{d_1 \times d_2}$, $\|X_{\text{in}} - X^*\|_F \leq R$, $\mathcal{O}$ is a $(1, 12L^2)$-oracle for $\mathcal{A}$ with failure probability $\delta \in (0,1)$, linear measurements $b = \mathcal{A}[X^*]$.
Output: $X_{\text{out}} \in \mathbb{R}^{d_1 \times d_2}$ *s.t.* $\|X_{\text{out}} - X^*\|_F \leq \frac{R}{2}$ *w.p.* $\geq 1 - \delta$ and $\text{rank}(X_{\text{out}}) \leq r$.
$X_0 \leftarrow X_{\text{in}}$, $\mathcal{X} = \{X \in \mathbb{R}^{d_1 \times d_2} \mid \|X - X_{\text{in}}\|_* \leq \sqrt{2r}R\}$, $\eta = \frac{1}{288L^4}$, $T = \frac{6}{\eta}$.
for $0 \leq t \leq T$ **do**
  $u_t \leftarrow \frac{1}{R}(\mathcal{A}[X_t] - b)$; `/*` $u_t = \mathcal{A}[\frac{X_t - X^*}{R}]$ `where` $(u_t)_i = \frac{1}{R}\langle A_i, X_t - X^* \rangle$ `*/`
  $w_t \leftarrow \mathcal{O}(\mathcal{A}, u_t, \frac{\delta}{T})$;
  $G_t \leftarrow \sum_{i=1}^n (w_t)_i (u_t)_i A_i$;
  if $\mathcal{O}$ output satisfies the progress and decomposition guarantees on u_t **then**
    $X_{t+1} \leftarrow \text{argmin}_{X \in \mathcal{X}} \|X - (X_t - \eta R G_t)\|_F^2$;
  else
    Return $X_{\text{out}} \leftarrow (X_t)_{(r)}$; `/* Rank-r approximation of` X_t `*/`
  end if
end for
Return $X_{\text{out}} \leftarrow (X_T)_{(r)}$;

A crucial requirement of the algorithm is a $(\Omega(1), O(1))$-oracle for $\mathcal{A}$. In each iteration, the oracle takes a vector $u_t = \frac{\mathcal{A}[X_t]-b}{R}$, which is the (normalized) "measured deviation" between current estimate X_t and X^*, and computes a weight vector w_t. The algorithm then tries to minimize the weighted objective function by gradient descent:

$$\text{Objective: } f_t(X) = \frac{1}{2}\sum_{i=1}^{n}(w_t)_i\langle A_i, \frac{X-X^*}{R}\rangle^2, \text{ i.e. } f_t(X_t) = \frac{1}{2}\sum_{i=1}^{n}(w_t)_i(u_t)_i^2,$$

$$\text{Gradient: } \nabla_X f_t(X) = \sum_{i=1}^{n}(w_t)_i\langle A_i, \frac{X-X^*}{R}\rangle A_i, \text{ i.e. } G_t = \nabla_X f_t(X)|_{X_t} = \sum_{i=1}^{n}(w_t)_i(u_t)_i A_i.$$

Ideally in the next iteration, we would like to make a step from X_t in the opposite direction of the gradient G_t with the goal of minimizing the deviation in the next iteration. However, we cannot take a step exactly in the direction of G_t, and our movement is constrained within a ball of (nuclear norm) radius $\sqrt{2r}R$ centered at X_{in}, namely the region $\mathcal{X} = \{X \mid \|X - X_{\text{in}}\|_* \leq \sqrt{2r}R\}$. Nuclear norm is used as a proxy to control the rank and Frobenius norm of X_t simultaneously throughout the algorithm: firstly, since $\|X_{\text{in}} - X^*\|_F \leq R$, it makes sense that in each iteration $\|X_t - X_{\text{in}}\|_F \leq R$ as well; secondly, while trying to minimize the difference between X_t and X^*, we also want to ensure the rank of X_t is relatively small, i.e. $\text{rank}(X_t) \leq O(r)$. To tie things together, we use the following relationship between rank and numerical rank:

$$\text{rank}(X_t - X_{\text{in}}) \geq \text{Rank}_{\text{n}}(X_t - X_{\text{in}}) = \frac{\|X_t - X_{\text{in}}\|_*^2}{\|X_t - X_{\text{in}}\|_F^2}.$$

Assuming $\text{rank}(X_t) \geq \text{rank}(X_{\text{in}})$ and $\|X_t - X_{\text{in}}\|_F \leq R$ throughout, then $\text{rank}(X_t) \geq \frac{\|X_t - X_{\text{in}}\|_*^2}{2R^2}$. Roughly speaking, in order for $\text{rank}(X_t) \leq O(r)$, it is necessary that $\|X_t - X_{\text{in}}\|_* \leq O(\sqrt{r}R)$, i.e. X_t is inside some nuclear norm ball $\mathcal{X}$ of radius $O(\sqrt{r}R)$ centered at X_{in}. We set the radius of $\mathcal{X}$ to be $\sqrt{2r}R$ so that $X^* \in \mathcal{X}$, since $\|X_{\text{in}} - X^*\|_F \leq R$, $\text{rank}(X_{\text{in}} - X^*) \leq 2r$ therefore $\|X^* - X_{\text{in}}\|_* \leq \sqrt{2r}R$. Thus we confine our movement within this nuclear norm ball of radius $\sqrt{2r}R$ centered at X_{in} throughout the algorithm, and take the rank-r approximation of the last X_t to ensure $\text{rank}(X_{\text{out}}) \leq r$ upon the termination of the algorithm.

To analyze the algorithm, first we show how to check whether the weight oracle output satisfies the progress and decomposition guarantees. The progress condition $\sum_{i=1}^n w_i u_i^2 \geq 1$ is trivial to verify, and we check whether G is (C_F, C_2)-decomposable using Lemma 5.4.1, with details and proof deferred to Section 5.6.1.

Lemma 5.4.1 (Verify norm decomposition)**.** *Given a matrix $G = U\Sigma V^\top = \sum_{i=1}^d \sigma_i u_i v_i^\top$ and $C_2 > 0$, suppose $\sigma_1 \geq ... \geq \sigma_k > C_2 \geq \sigma_{k+1} ... \geq \sigma_d$, then for all $\|E\|_2 \leq C_2$, we have $\|G - E\|_F^2 \geq \sum_{i=1}^k (\sigma_i - C_2)^2$.*

The following lemmas analyze the algorithm's correctness and show that it terminates with the desired distance contraction, as well as its running time. The proof is deferred to Section 5.6.1.

Lemma 5.4.2 (Algorithm 5.2: HalveError)**.** *Given a* $(1, 12L^2)$*-oracle for* $\mathcal{A}$ *with failure probability* $\delta \in (0,1)$*, where* $\mathcal{A}$ *satisfies the dRIP condition 5.2.3, and* $b = \mathcal{A}[X^*]$*, Algorithm 5.2 succeeds with probability at least* $1 - \delta$*.*

Lemma 5.4.3 (Running time of Algorithm 5.2)**.** *Algorithm 5.2 with failure probability* δ *runs in time* $O(nd^\omega \operatorname{polylog}(d) \log \frac{L}{\delta} r\rho^2 L^4)$*.*

The crucial part of Lemma 5.4.2 shows that if current estimate X_t is sufficiently far from X^*, i.e. $\|X_t - X^*\|_F \geq \frac{1}{4}R$, then according to Lemma 5.5.3 with high probability the weight oracle produces an output satisfying the progress and decomposition guarantees, and each iteration of Algorithm 5.2 decreases the distance to X^* by a constant factor: $\|X_{t+1} - X^*\|_F^2 \leq \left(1 - \frac{\eta}{2}\right) \cdot \|X_t - X^*\|_F^2$, thus after sufficient number of iterations the distance to X^* will be halved. On the other hand, if the weight oracle fails, with high probability the current estimate X_t is already sufficiently close to X^*, thus the algorithm can terminate early.

5.5 Oracle for Reweighting the Input

In this section, we present an algorithm (Algorithm 5.3) that serves as the weight oracle required by the error-halving algorithm (Algorithm 5.2). Algorithm 5.3 is the matrix version of the StepOracle (Kelner et al., 2023a) algorithm for vectors. We first state that, given proper choices of parameters within a constant factor, the wRIP condition 5.2.2 implies the dRIP condition 5.2.3, which is a more suitable property for our oracle implementation. The proof is deferred to Section 5.6.3.

Lemma 5.5.1 (*wRIP* $\implies$ *dRIP*)**.** *If $\mathcal{A}$ satisfies wRIP condition 5.2.2 with parameters r', L', ρ, then $\mathcal{A}$ satisfies the dRIP condition 5.2.3 with parameters L, K, r, ρ such that $L = \Theta(L')$, $r = \Theta(r')$, and some constant $K \geq 1$.*

Now we are ready to present an implementation of the weight oracle in Algorithm 5.3 based on the dRIP condition. This algorithm takes as inputs the dRIP sensing matrices $\mathcal{A}$ and a vector u. If u is an applicable input to the oracle, with high probability the algorithm outputs a weight vector w satisfying the progress and decomposition guarantees as in Definition 5.2.5.

First we introduce some potential functions used in the algorithm, adapted from vector versions from (Kelner et al., 2023a).

Definition 5.5.2 (Potential Functions in Algorithm 5.3)**.** *For sensing matrices $\mathcal{A} = \{A_i\}_{i=1}^n$ and input $u \in \mathbb{R}^n$ to the oracle, we define the following potential functions on weight vector $w \in \mathbb{R}^n$:*

- *Progress potential:* $\Phi_{prog}(w) = \sum_{i=1}^n w_i u_i^2$.
- *Decomposition potential:* $\Phi_{dc}(w) = \min_{\|S\|_F \leq L\|w\|_1} \left(\mu^2 \log\left[F(G_w - S)\right]\right) + \frac{\|w\|_1}{4CLr}$, *where* $G_w = \sum_{i=1}^n w_i u_i A_i$ *and* $F(E) = \operatorname{tr} \exp\left(\frac{E^\top E}{\mu^2}\right)$.
- *Overall potential:* $\Phi(w) = \Phi_{prog}(w) - Cr\Phi_{dc}(w)$.

Note that $F(E) = \sum_{j=1}^d \exp\left(\frac{\sigma_j^2(E)}{\mu^2}\right)$ where $\sigma_j(E)$ is the j^{th} singular value of E, due to properties of the exponential of a diagonalizable matrix.

The progress and decomposition potential functions control the progress and decomposition guarantees respectively, and later we will show that the termination condition is implied by the overall potential $\Phi \geq 0$. Consequently, by maximizing the overall potential each round, the algorithm tries to make as much progress as possible while ensuring G is decomposable.

Algorithm 5.3: Weight oracle $\mathcal{O}(\mathcal{A}, u, \delta)$ (Kelner et al., 2023a)

Input: Sensing operator $\mathcal{A}$ satisfying dRIP condition 5.2.3, $u \in \mathbb{R}^n$.
Output: $w \in \mathbb{R}^n$ such that the algorithm is a $(1, 12L^2)$-oracle as in Definition 5.2.5 with probability $\geq (1 - \delta)$.
$C \leftarrow 108$, $\mu \leftarrow \frac{1}{\sqrt{Cr \log d}}$, $\eta \leftarrow \frac{1}{Kr\rho^2 \log d}$, $N' \leftarrow \log_2 \frac{1}{\delta}$, $N \leftarrow \frac{8Ln}{\eta}$.
for $0 \leq k \leq N'$ **do**
  $w_0 \leftarrow 0$;
  for $0 \leq t \leq N$ **do**
    if $\Phi_{\text{prog}}(w_t) \geq 1$ **then**
      Return $w \leftarrow w_t$;
    else
      Sample $i \in [n]$ uniformly random;
      $s_t \leftarrow \text{argmax}_{s \in [0,\eta]} \Phi(w_t + se_i)$;
      $w_{t+1} \leftarrow w_t + s_t e_i$;
    end if
  end for
end for
Return $w \leftarrow 0$;

Lemma 5.5.3 (Algorithm 5.3: weight oracle)**.** *Suppose $\mathcal{A}$ satisfies dRIP condition 5.2.3 and u is an applicable input to the weight oracle (that is, $u = \mathcal{A}[V] \in \mathbb{R}^n$ for some $V \in \mathbb{R}^{d_1 \times d_2}$ satisfying $\|V\|_F \in [\frac{1}{4}, 1]$ and $\|V\|_* \leq 2\sqrt{2r}$). Then, Algorithm 5.3 is a $(1, 12L^2)$-oracle for $\mathcal{A}$ (as in Definition 5.2.5) with failure probability at most δ.*

We prove this lemma in two steps: first we show in Lemma 5.5.4 that the output is valid; then in Lemma 5.5.5 we show that the oracle achieves the success probability. Finally we analyze the running time of Algorithm 5.3 in Lemma 5.5.6. The proofs follow similar techniques as the results on vectors (Lemma 11, 13, 14 (Kelner et al., 2023a)), and are deferred to Section 5.6.2.

Lemma 5.5.4 (Correctness of Algorithm 5.3)**.** *If Algorithm 5.3 terminates from the inner loop, the output satisfies the progress and decomposition guarantees as defined in 5.2.5.*

Lemma 5.5.5 (Success probability of Algorithm 5.3)**.** *Given $\mathcal{A}$ satisfying dRIP condition and applicable input u, Algorithm 5.3 terminates from the inner loop with probability at least $1 - \delta$.*

Lemma 5.5.6 (Running time of Algorithm 5.3)**.** *Algorithm 5.3 with failure probability δ runs in time $O(nd^{\omega}\,\mathrm{polylog}\,(d)\log\frac{1}{\delta}r\rho^2)$.*

Although our weight oracle is inspired by the step oracle in (Kelner et al., 2023a) for vectors, it is worth noting that Lemma 5.6.2, a key component used in the proof of Lemma 5.5.5, is significantly different in the matrix case compared to the vector case. Lemma 5.6.2 upper bounds the increase in Φ_{dc} each round, which is then used to provide a lower bound for the increase in Φ. Combining Lemma 5.6.2 with our earlier remark that the algorithm terminates when $\Phi \geq 0$ gives us the number of iterations needed to terminate with high probability.

5.6 Omitted Proofs

5.6.1 Omitted Proofs From Section 5.4

To check whether G is (C_F, C_2)-decomposable using Lemma 5.4.1, we first consider the following construction. Suppose the SVD of G is $G = U\Sigma V^\top = \sum_{i=1}^d \sigma_i u_i v_i^\top$. Let $S = \sum_{i=1}^d \mu_i u_i v_i^\top$ with $\mu_i = \max(\sigma_i - C_2, 0)$ and let $E = \sum_{i=1}^d \lambda_i u_i v_i^\top$ with $\lambda_i = \min(C_2, \sigma_i)$. In other words, suppose $\sigma_1 \geq ... \geq \sigma_k > C_2 \geq \sigma_{k+1} ... \geq \sigma_d$. For $i \leq k$ (i.e., $\sigma_i > C_2$), let $\mu_i = \sigma_i - C_2$ and $\lambda_i = C_2$; for $i > k$ (i.e., $\sigma_i \leq C_2$), let $\mu_i = 0$ and $\lambda_i = \sigma_i$. We have $G = S + E$ with $\|E\|_2 \leq C_2$, and $\|S\|_F = \sqrt{\sum_{i=1}^k (\sigma_i - C_2)^2}$. Then by Lemma 5.4.1, G is (C_F, C_2)-norm-decomposable if and only if $\|S\|_F \leq C_F$ because:

- if $\|S\|_F \leq C_F$, we have a valid (C_F, C_2)-norm-decomposition for G;
- if $\|S\|_F > C_F$, a valid (C_F, C_2)-norm-decomposition does not exist for G.

Proof of Lemma 5.4.1. Fix any E with $\|E\|_2 \leq C_2$. Observe that for all $1 \leq i \leq k$, we have $u_i^\top G v_i = \sigma_i$ and $-C_2 \leq u_i^\top E v_i \leq C_2$. Consequently, for all $i \leq k$, we have $\sigma_i > C_2$ and $u_i^\top (G - E) v_i \geq \sigma_i - C_2 > 0$.

Let $S = G - E$, we have

$$\begin{aligned}
\|S\|_F^2 = \left\|U^\top S V\right\|_F^2 &= \sum_{i,j} \left(U^\top S V\right)_{ij}^2 \\
&\geq \sum_{i=1}^d \left(U^\top S V\right)_{ii}^2 \geq \sum_{i=1}^k \left(U^\top S V\right)_{ii}^2 \\
&= \sum_{i=1}^k (u_i^\top S v_i)^2 \geq \sum_{i=1}^k (\sigma_i - C_2)^2.
\end{aligned}$$

■

Proof of Lemma 5.4.2. We will first show that the distance to X^* decreases by a constant factor after each iteration:

$$\|X_{t+1} - X^*\|_F^2 \leq \left(1 - \frac{\eta}{2}\right) \cdot \|X_t - X^*\|_F^2$$

Consider iteration t in Algorithm 5.2: $X_{t+1} = \text{argmin}_{X \in \mathcal{X}} \|X - (X_t - \eta R G_t)\|_F^2$. Taking the gradient of $\|X - (X_t - \eta R G_t)\|_F^2$ at $X = X_{t+1}$, we get $2[X_{t+1} - (X_t - \eta R G_t)]$. Since $X^* \in \mathcal{X}$ and X_{t+1} is the local minimizer in $\mathcal{X}$:

$$2\langle X_{t+1} - (X_t - \eta R G_t), X_{t+1} - X^* \rangle \leq 0.$$

Rearranging the terms gives

$$2\eta R \langle G_t, X_{t+1} - X^* \rangle \leq -2\langle X_{t+1} - X_t, X_{t+1} - X^* \rangle.$$

To simplify, let $D = X_{t+1} - X_t$, $D_t = X_t - X^*$, $D_{t+1} = X_{t+1} - X^*$. Note that $D + D_t = D_{t+1}$. Then the inequality becomes

$$\begin{aligned} 2\eta R \langle G_t, D_{t+1} \rangle &\leq -2\langle D, D_{t+1} \rangle \\ &= \langle D - D_{t+1}, D - D_{t+1} \rangle - \langle D, D \rangle - \langle D_{t+1}, D_{t+1} \rangle \\ &= \langle D_t, D_t \rangle - \langle D, D \rangle - \langle D_{t+1}, D_{t+1} \rangle. \end{aligned}$$

Rearrange terms, we have

$$\begin{aligned}\|D_t\|_F^2 - \|D_{t+1}\|_F^2 &\geq 2\eta R\langle G_t, D_{t+1}\rangle + \langle D, D\rangle \\ &= 2\eta R\langle G_t, D_t\rangle + 2\eta R\langle G_t, D\rangle + \langle D, D\rangle. \end{aligned} \tag{5.3}$$

Inequality (5.3) provides a lower bound on the distance contraction after each iteration. We break the right-hand side into two parts. The first term $2\eta R\langle G_t, D_t\rangle$ corresponds to the magnitude of the step G_t in the direction $D_t = X_t - X^*$, which is the progress made by this step. To lower bound it, we will use the **progress guarantee** of the $(1, 12L^2)$-oracle. Recall $u_t = \frac{1}{R}(\mathcal{A}[X_t] - b)$ and consider $V_t = \frac{1}{R}(X_t - X^*) = \frac{1}{R}D_t$ so that $u_t = \mathcal{A}[V_t]$. Given that the oracle's output satisfies the progress guarantee, which states that $\sum_{i=1}^n (w_t)_i (u_t)_i^2 \geq 1$, we have:

$$\begin{aligned}2\eta R\langle G_t, D_t\rangle &= 2\eta R^2\langle G_t, V_t\rangle \\ &= 2\eta R^2\Big\langle \sum_{i=1}^n (w_t)_i (u_t)_i A_i, V_t\Big\rangle \\ &= 2\eta R^2 \sum_{i=1}^n (w_t)_i (u_t)_i^2 \\ &\geq 2\eta R^2. \end{aligned} \tag{5.4}$$

The remaining term $2\eta R\langle G_t, D\rangle + \langle D, D\rangle$ might be negative and cancel some of our progress. A natural attempt is to try to bound it using $2\eta R\langle G_t, D\rangle + \langle D, D\rangle \geq -\eta^2 R^2\langle G_t, G_t\rangle$. How-

ever, the wRIP condition of $\mathcal{A}$ does not provide any guarantee on $\|G_t\|_F^2$. (In fact, we can derive that $\|G\|_2 \le L$ from wRIP, but the best we can hope for is $\|G\|_F^2 \le \text{rank}(G) \cdot \|G\|_2^2$ where $\text{rank}(G) \le d$.) This motivates the decomposition property in the dRIP condition 5.2.3 and in the weight oracle. The idea is that even though we cannot directly bound $\|G_t\|_F$, we can in fact lower bound $2\eta R\langle G_t, D\rangle + \langle D, D\rangle$ the term by decomposing G_t into a Frobenius-norm-bounded matrix S_t, and an operator-norm-bounded matrix E_t. Specifically, we will use the **decomposition guarantee** of the $(1, 12L^2)$-oracle, which states that there exists norm-decomposition of $G_t = S_t + E_t$ where $\|S_t\|_F \le 12L^2$ and $\|E_t\|_2 \le \frac{1}{6\sqrt{r}}$. As our movement is confined in $\mathcal{X}$, $D = X_{t+1} - X_t$ is nuclear-norm-bounded so the inner product $\langle E_t, D\rangle$ can be bounded by generalized Holder's inequality. Recall $\eta = \frac{1}{288L^4}$.

$$\begin{aligned}
2\eta R\langle G_t, D\rangle + \langle D, D\rangle &= 2\eta R\langle E_t, D\rangle + 2\eta R\langle S_t, D\rangle + \langle D, D\rangle \\
&\ge -2\eta R \|E_t\|_2 \cdot \|D\|_* - \eta^2 R^2 \langle S_t, S_t\rangle \\
&\ge -2\eta R \cdot \frac{1}{6\sqrt{r}} \cdot 2\sqrt{2r}R - \eta^2 R^2 \cdot 144L^4 \\
&\ge -\frac{3}{2}\eta R^2. \qquad (5.5)
\end{aligned}$$

Putting inequalities 5.3, 5.4 and 5.5 together:

$$\|D_t\|_F^2 - \|D_{t+1}\|_F^2 \ge 2\eta R^2 - \frac{3}{2}\eta R^2 \ge \frac{\eta}{2} \cdot \|D_t\|_F^2,$$
$$\|D_{t+1}\|_F^2 \le \left(1 - \frac{\eta}{2}\right) \cdot \|D_t\|_F^2.$$

In the case that the algorithm terminates after $T = \frac{6}{\eta}$ iterations,

$$\|X_T - X^*\|_F^2 \leq \left(1 - \frac{\eta}{2}\right)^T \cdot \|X_{\text{in}} - X^*\|_F^2 \leq \exp\left(-\frac{\eta T}{2}\right) \cdot R^2 \leq \frac{1}{16}R^2,$$
$$\|X_{\text{out}} - X^*\|_F \leq \|X_{\text{out}} - X_T\|_F + \|X_T - X^*\|_F \leq 2\|X_T - X^*\|_F \leq \frac{1}{2}R.$$

The last inequality comes from X_{out} being the best rank-r approximation of X_T.

In the case that the algorithm terminates early at Line 11, we can assume with probability at least $1 - \frac{\delta}{T}$ that the weight oracle would have succeeded given applicable input u_t. Then failure to satisfy the progress and decomposition guarantees means that u_t is not an applicable input, which means V_t does not satisfy the norm constraint in the weight oracle. $\|V_t\|_* \leq 2\sqrt{2r}$ is guaranteed because $X_t \in \mathcal{X}$, and $\|V_t\|_F = \frac{1}{R}\|X_t - X^*\|_F$ is decreasing in each round, so we must have $\|V_t\|_F < \frac{1}{4}$, which means $\|X_t - X^*\|_F < \frac{1}{4}R$. By the same argument as above, $\|X_{\text{out}} - X^*\|_F \leq \frac{1}{2}R$.

Finally, by a union bound on the failure probability of the weight oracle, the algorithm succeeds with probability at least $1 - \delta$. ■

Proof of Lemma 5.4.3. Algorithm 5.2 has a for-loop that's repeated for $T = O(L^4)$ times.

Inside the loop, line 5 and 7 takes linear time $O(nd^2)$. Computing w_t using the oracle (line 6) runs in time $O(nd^\omega \operatorname{polylog}(d) \log \frac{L}{\delta} r\rho^2)$ according to Lemma 5.5.6. Line 8 through line 12 are all upper bounded by time of SVD, which is on the same order of matrix multiplication $O(d^\omega)$ (Demmel et al., 2007), with current best of $O(d^{2.373})$(Williams, 2014).

In particular, verifying the oracle guarantees (line 8) can be solved as an eigenvalue problem. Finding X_{t+1} (line 9) is equivalent to projecting $X'_{t+1} := \frac{X_t - \eta R G_t - X_{\text{in}}}{\sqrt{2rR}}$ onto the unit nuclear norm ball. We first perform SVD on X'_{t+1} then binary search for the largest truncation from its singular values to reach the nuclear norm sphere in time $O(d \log d)$, and the entire projection step is dominated by SVD. Finally the output step consists of SVD and matrix multiplication.

The overall running time of the algorithm is dominated by the weight oracle, so the total running time is $O(nd^\omega \operatorname{polylog}(d) \log \frac{L}{\delta} r\rho^2 L^4)$. ■

5.6.2 Omitted Proofs From Section 5.5

First we state a couple of lower bounds related to the decomposition potential function, similar to Lemma 11 (Kelner et al., 2023a) for vectors.

Claim 5.6.1. *$\mu^2 \log[F(E)] \geq \mu^2 \log(d)$ and $\mu^2 \log[F(E)] \geq \|E\|_2^2$.*

Proof. For the first lower bound, $\exp\left(\frac{\sigma_j^2(E)}{\mu^2}\right) \geq 1$ for all $j \in [d]$, therefore

$$F(E) = \sum_{j=1}^{d} \exp\left(\frac{\sigma_j^2(E)}{\mu^2}\right) \geq d.$$

For the second lower bound,

$$F(E) = \sum_{j=1}^{d} \exp\left(\frac{\sigma_j^2(E)}{\mu^2}\right) \geq \exp\left(\frac{\sigma_1^2(E)}{\mu^2}\right) = \exp\left(\frac{\|E\|_2^2}{\mu^2}\right).$$ ■

Proof of Lemma 5.5.4. We start with $w_0 = 0$, which means $\Phi_{\text{prog}}(w_0) = 0$, $\Phi_{\text{dc}}(w_0) = \mu^2 \log d$, and $\Phi(w_0) = 0 - Cr\mu^2 \log d = -1$.

At each round, since s_t is chosen to maximize $\Phi(w_t + s_t e_i)$, in particular if we choose $s_t = 0$ then $\Phi(w_{t+1}) = \Phi(w_t)$, so $\Phi(w_{t+1}) \geq \Phi(w_t)$ which is non-decreasing. By definition $\Phi_{\text{prog}}(w_t)$ is also non-decreasing, and increases by at most 1 each round, because $\Phi_{\text{prog}}(w_{t+1}) - \Phi_{\text{prog}}(w_t) = s_t u_i^2 \leq \eta(\|A_i\|_2 \|V\|_*)^2 \leq 8\eta r \rho^2 \leq \frac{8}{K \log d} \leq 1$. $\Phi_{\text{dc}}(w_t)$ may not be monotone, but we have $\Phi_{\text{dc}}(w_t) \geq \mu^2 \log d$.

Suppose the algorithm terminates at round t during one of the inner loops, which means $\Phi_{\text{prog}}(w_{t-1}) < 1$ and $1 \leq \Phi_{\text{prog}}(w_t) < 2$.

Progress guarantee: $\Phi_{\text{prog}}(w_t) = \sum_{i=1}^n (w_t)_i u_i^2 \geq 1$ is satisfied upon termination.

Decomposition guarantee:

$$\Phi(w_t) = \Phi_{\text{prog}}(w_t) - Cr\Phi_{\text{dc}}(w_t) \geq \Phi(w_0) = -1,$$

$$\implies \min_{\|S\|_F \leq L\|w\|_1} \left(\mu^2 \log\left[F(G_w - S)\right]\right) + \frac{\|w_t\|_1}{4CLr} = \Phi_{\text{dc}}(w_t) \leq \frac{\Phi_{\text{prog}}(w_t) + 1}{Cr} \leq \frac{3}{Cr},$$

$$\min_{\|S\|_F \leq L\|w_t\|_1} \left(\mu^2 \log\left[F(G_{w_t} - S)\right]\right) \leq \frac{3}{Cr} \implies \exists\, \|S\|_F \leq L\|w_t\|_1 \;\; s.t. \;\; \|G_{w_t} - S\|_2^2 \leq \frac{3}{Cr},$$

$$\text{and } \frac{\|w_t\|_1}{4CLr} \leq \frac{3}{Cr} \implies \|w_t\|_1 \leq 12L.$$

So there exist $\|S\|_F \leq 12L^2$ and $\|E\|_2 = \|G - S\|_2 \leq \frac{\sqrt{3}}{\sqrt{Cr}} = \frac{1}{6\sqrt{r}}$ which satisfy the decomposition guarantee.

Notice that for any round $t' < t$, $\Phi_{\text{prog}}(w_{t'}) < 1$, we also have $\Phi_{\text{dc}}(w_{t'}) \leq \frac{\Phi_{\text{prog}}(w_{t'})+1}{Cr} \leq \frac{2}{Cr}$, so $\Phi_{\text{dc}}(w_t) \leq \frac{3}{Cr}$ throughout the algorithm, which is a fact we will use later in Lemma 5.6.2. ■

Proof of Lemma 5.5.5. We first show the probability that the algorithm terminates from the inner loop is at least $\frac{1}{2}$, i.e., $\Pr[\Phi_{\text{prog}}(w_t) \geq 1] \geq \frac{1}{2}$ for some $t \leq N$.

Notice that $\Phi(w_t) = \Phi_{\text{prog}}(w_t) - Cr\Phi_{\text{dc}}(w_t) \geq 0 \implies \Phi_{\text{prog}}(w_t) \geq Cr\Phi_{\text{dc}}(w_t) \geq Cr\Phi_{\text{dc}}(w_0) = 1$, therefore the algorithm starts with $\Phi(w_0) = -1$ and will terminate once $\Phi(w_t) \geq 0$. Also notice that throughout the algorithm $\Phi(w_t) < 1$ because $\Phi_{\text{prog}}(w_t) < 2$ and $Cr\Phi_{\text{dc}}(w_t) \geq 1$ (from proof of Lemma 5.5.4).

To prove by contradiction, assume that $\Pr[\Phi_{\text{prog}}(w_t) \geq 1] < \frac{1}{2}$ for all $t \leq N$, i.e., $\Pr[\text{continue}] \geq \frac{1}{2}$ for all rounds. We will lower bound the expected increase in $\Phi(w_t)$ each round, and we will show that with sufficiently large N, $\mathbb{E}[\Phi(w_N)] \geq 1$ contradicting $\Phi(w_t) < 1$ for all $t \leq N$.

Recall that $\Phi(w_t) = \Phi_{\text{prog}}(w_t) - Cr\Phi_{\text{dc}}(w_t)$, the lower bound for increase in $\Phi_{\text{prog}}(w_t)$ is provided by dRIP condition on $\mathcal{A}$ and applicable input u. The upper bound for expected increase in $\Phi_{\text{dc}}(w_t)$ is provided by Lemma 5.6.2.

Given the algorithm continues at round $t \leq N$, consider choosing $s_t = \eta w_i^*$ so that $w' = w_t + \eta w_i^* e_i$, then the expected increase in Φ is at least:

$$\begin{aligned}
\mathbb{E}[\Phi(w_{t+1}) - \Phi(w_t) \mid \text{continue}] &= \mathbb{E}[\Phi_{\text{prog}}(w_{t+1}) - \Phi_{\text{prog}}(w_t)] - Cr\mathbb{E}[\Phi_{\text{dc}}(w_{t+1}) - \Phi_{\text{dc}}(w_t)] \\
&\geq \mathbb{E}\big[\Phi_{\text{prog}}(w') - \Phi_{\text{prog}}(w_t)\big] - Cr\mathbb{E}\big[\Phi_{\text{dc}}(w') - \Phi_{\text{dc}}(w_t)\big] \\
&= \frac{1}{n}\sum_{i=1}^{n} \eta w_i^* u_i^2 - Cr\Big(\mathbb{E}\big[\Phi_{\text{dc}}(w')\big] - \Phi_{\text{dc}}(w_t)\Big) \\
&\geq \frac{\eta}{Ln} - Cr \cdot \frac{\eta}{2CLnr} \\
&= \frac{\eta}{2Ln}.
\end{aligned}$$

Given the algorithm stops after round t, $\mathbb{E}[\Phi(w_{t+1}) - \Phi(w_t) \mid \text{stop}] = 0$. Overall:

$$\begin{aligned}
\mathbb{E}[\Phi(w_{t+1}) - \Phi(w_t)] &= \mathbb{E}[\Phi(w_{t+1}) - \Phi(w_t) \mid \text{continue}] \cdot \Pr[\text{continue}] + 0 \\
&\geq \frac{\eta}{2Ln} \cdot \Pr[\text{continue}]\,.
\end{aligned}$$

By choosing a sufficiently large $N = \frac{8Ln}{\eta}$:

$$\mathbb{E}[\Phi(w_N)] \geq \Phi(w_0) + \frac{\eta N}{2Ln} \cdot \Pr[\text{continue}] \geq -1 + \frac{\eta N}{4Ln} \geq 1,$$

contradicting $\Phi(w_t) < 1$. This means each inner loop of the algorithm will terminate with probability greater than $\frac{1}{2}$. Finally, we boost the success probability to $(1 - \delta)$ using the outer loop with $N' = \log_2 \frac{1}{\delta}$ iterations. ■

Lemma 5.6.2 provides a crucial bound used in the proof. Even though it achieves similar result as Lemma 13 (Kelner et al., 2023a) on the potential functions defined for vectors, analyzing the potential function defined for matrices involves very different techniques.

Lemma 5.6.2 (Potential increase upper bound)**.** *Given $w \in \mathbb{R}^n$ s.t. $\Phi_{dc}(w) \leq \frac{3}{Cr}$, by choosing a sufficiently large value for K, for $w' = w + \eta w_i^* e_i$, we have:*

$$\mathbb{E}_{i\in[n]}\left[\Phi_{dc}(w')\right] \leq \Phi_{dc}(w) + \frac{\eta}{2CLnr}.$$

The assumption $\Phi_{dc}(w) \leq \frac{3}{Cr}$ is justified in the proof of Lemma 5.5.4.

Proof. First we introduce some notation:

Denote $\Phi_{\text{op}}(w) = \min_{\|S\|_F \leq L\|w\|_1} \left(\mu^2 \log\left[F(G_w - S)\right]\right)$, so that $\Phi_{\text{dc}}(w) = \Phi_{\text{op}}(w) + \frac{\|w\|_1}{4CLr}$. Let $G^* = \sum_{i=1}^n w_i^* u_i A_i$, and by dRIP condition 5.2.3, we know $\exists(L, \frac{1}{K\sqrt{r}})$-norm-decomposition of $G^* = S^* + E^*$, where $\|S^*\|_F \leq L$ and $\|E^*\|_2 \leq \frac{1}{K\sqrt{r}}$. Let $G = \sum_{i=1}^n w_i u_i A_i$, and $S = \operatorname{argmin} \Phi_{\text{op}}(w)$ so that $\Phi_{\text{op}}(w) = \mu^2 \log[F(G - S)]$, and let $E = G - S$. Let $G' = \sum_{i=1}^n w_i' u_i A_i$.

Using these notation and $\sum_{i=1}^n w_i^* = 1$:

$$\begin{aligned}\mathop{\mathbb{E}}_{i\in[n]}\left[\Phi_{\text{dc}}(w')\right] &= \mathop{\mathbb{E}}_{i\in[n]}\left[\Phi_{\text{op}}(w') + \frac{\|w'\|_1}{4CLr}\right] = \mathop{\mathbb{E}}_{i\in[n]}\left[\Phi_{\text{op}}(w') + \frac{\|w\|_1 + \eta w_i^*}{4CLr}\right] \\ &= \mathop{\mathbb{E}}_{i\in[n]}\left[\Phi_{\text{op}}(w')\right] + \frac{\|w\|_1}{4CLr} + \frac{\eta}{4CLnr}.\end{aligned}$$

We need to show $\mathbb{E}_{i\in[n]}[\Phi_{\mathrm{dc}}(w')] \le \Phi_{\mathrm{dc}}(w) + \frac{\eta}{2CLnr} = \Phi_{\mathrm{op}}(w) + \frac{\|w\|_1}{4CLr} + \frac{\eta}{2CLnr}$, equivalently

$$\mathbb{E}_{i\in[n]}\big[\Phi_{\mathrm{op}}(w')\big] \le \Phi_{\mathrm{op}}(w) + \frac{\eta}{4CLnr}.$$

Consider $S' = S + \eta w_i^* S^*$. We have $\|S'\|_F \le \|S\|_F + \eta w_i^* \|S^*\|_F \le L\cdot\|w\|_1 + \eta w_i^* L = L\cdot\|w'\|_1$, so S' is a valid argument for $\Phi_{\mathrm{op}}(w') = \min_{\|S\|_F \le L\|w'\|_1}\Big(\mu^2 \log\left[F(G'-S)\right]\Big)$, therefore $\Phi_{\mathrm{op}}(w') \le \mu^2 \log[F(G'-S')]$. Let $E' = G' - S' = G + \eta w_i^* u_i A_i - S - \eta w_i^* S^* = E + Z^{(i)}$ where $Z^{(i)} = \eta w_i^* u_i A_i - \eta w_i^* S^*$. Using these and the concavity of the log function, we have

$$\begin{aligned}\mathbb{E}_{i\in[n]}\big[\Phi_{\mathrm{op}}(w')\big] &\le \frac{1}{n}\sum_{i=1}^n \mu^2 \log[F(G'-S')] = \frac{1}{n}\sum_{i=1}^n \mu^2 \log[F(E+Z^{(i)})] \\ &\le \mu^2 \log\left[\frac{1}{n}\sum_{i=1}^n \Big(F(E+Z^{(i)})\Big)\right].\end{aligned}$$

It suffices to show

$$\mu^2 \log\left[\frac{1}{n}\sum_{i=1}^n \Big(F(E+Z^{(i)})\Big)\right] \le \Phi_{\mathrm{op}}(w) + \frac{\eta}{4CLnr} = \mu^2 \log[F(E)] + \frac{1}{4CL}\frac{\eta}{nr}.$$

Expanding the left hand side:

$$\begin{aligned}
&\mu^2 \log \left[\frac{1}{n} \sum_{i=1}^{n} F(E + Z^{(i)}) \right] \\
&= \mu^2 \log \left[\frac{1}{n} \sum_{i=1}^{n} \operatorname{tr} \exp \left(\frac{E^\top E + Z^{(i)\top} Z^{(i)} + E^\top Z^{(i)} + Z^{(i)\top} E}{\mu^2} \right) \right] \\
&\leq \mu^2 \log \left[\frac{1}{n} \sum_{i=1}^{n} \operatorname{tr} \left[\exp \left(\frac{E^\top E}{\mu^2} \right) \cdot \exp \left(\frac{Z^{(i)\top} Z^{(i)} + E^\top Z^{(i)} + Z^{(i)\top} E}{\mu^2} \right) \right] \right] \\
&= \mu^2 \log \left[\operatorname{tr} \left[\exp \left(\frac{E^\top E}{\mu^2} \right) \cdot \frac{1}{n} \sum_{i=1}^{n} \exp \left(\frac{Z^{(i)\top} Z^{(i)} + E^\top Z^{(i)} + Z^{(i)\top} E}{\mu^2} \right) \right] \right] \\
&\leq \mu^2 \log \left[\operatorname{tr} \exp \left(\frac{E^\top E}{\mu^2} \right) \cdot \left\| \frac{1}{n} \sum_{i=1}^{n} \exp \left(\frac{Z^{(i)\top} Z^{(i)} + E^\top Z^{(i)} + Z^{(i)\top} E}{\mu^2} \right) \right\|_2 \right] \\
&= \mu^2 \log \left[F(E) \right] + \mu^2 \log \left\| \frac{1}{n} \sum_{i=1}^{n} \exp \left(\frac{Z^{(i)\top} Z^{(i)} + E^\top Z^{(i)} + Z^{(i)\top} E}{\mu^2} \right) \right\|_2 .
\end{aligned}$$

The first inequality uses Golden-Thompson Inequality (stated as Lemma 5.6.10), and the second inequality follows from Lemma 5.6.3. Finally it suffices to show

$$\mu^2 \log \left\| \frac{1}{n} \sum_{i=1}^{n} \exp \left(\frac{Z^{(i)\top} Z^{(i)} + E^\top Z^{(i)} + Z^{(i)\top} E}{\mu^2} \right) \right\|_2 \leq \frac{1}{4CL} \frac{\eta}{nr}.$$

We will use the approximation $\exp(X) \preccurlyeq I + X + X^2$ for symmetric X with $\|X\|_2 \leq 1$. The argument in the exponential satisfies this condition as justified in Claim 5.6.5. We will also use $\log(1+x) \leq x \ \forall x \geq 0$.

$$
\begin{aligned}
&\mu^2 \log \left\| \frac{1}{n} \sum_{i=1}^{n} \exp\left(\frac{Z^{(i)\top} Z^{(i)} + E^\top Z^{(i)} + Z^{(i)\top} E}{\mu^2} \right) \right\|_2 \\
&\leq \mu^2 \log \left[\frac{1}{n} \sum_{i=1}^{n} \left\| \exp\left(\frac{Z^{(i)\top} Z^{(i)} + E^\top Z^{(i)} + Z^{(i)\top} E}{\mu^2} \right) \right\|_2 \right] \\
&\leq \mu^2 \log \left[\frac{1}{n} \sum_{i=1}^{n} \left\| I + \frac{Z^{(i)\top} Z^{(i)} + E^\top Z^{(i)} + Z^{(i)\top} E}{\mu^2} + \frac{(Z^{(i)\top} Z^{(i)} + E^\top Z^{(i)} + Z^{(i)\top} E)^2}{\mu^4} \right\|_2 \right] \\
&\leq \mu^2 \log \left[\|I\|_2 + \left\| \frac{1}{n} \sum_{i=1}^{n} \frac{Z^{(i)\top} Z^{(i)}}{\mu^2} \right\|_2 + \left\| \frac{1}{n} \sum_{i=1}^{n} \frac{E^\top Z^{(i)} + Z^{(i)\top} E}{\mu^2} \right\|_2 \right. \\
&\qquad \left. + \left\| \frac{1}{n} \sum_{i=1}^{n} \frac{(Z^{(i)\top} Z^{(i)} + E^\top Z^{(i)} + Z^{(i)\top} E)^2}{\mu^4} \right\|_2 \right] \\
&\leq \mu^2 \log \left[\|I\|_2 + \left\| \frac{1}{n} \sum_{i=1}^{n} \frac{Z^{(i)\top} Z^{(i)}}{\mu^2} \right\|_2 + \left\| \frac{1}{n} \sum_{i=1}^{n} \frac{E^\top Z^{(i)} + Z^{(i)\top} E}{\mu^2} \right\|_2 \right. \\
&\qquad \left. + \left\| \frac{1}{n} \sum_{i=1}^{n} \frac{2(Z^{(i)\top} Z^{(i)})^2}{\mu^4} \right\|_2 + \left\| \frac{1}{n} \sum_{i=1}^{n} \frac{2(E^\top Z^{(i)} + Z^{(i)\top} E)^2}{\mu^4} \right\|_2 \right] \\
&\leq \mu^2 \left\| \frac{1}{n} \sum_{i=1}^{n} \frac{Z^{(i)\top} Z^{(i)}}{\mu^2} \right\|_2 + \mu^2 \left\| \frac{1}{n} \sum_{i=1}^{n} \frac{E^\top Z^{(i)} + Z^{(i)\top} E}{\mu^2} \right\|_2 \\
&\qquad + 2\mu^2 \left\| \frac{1}{n} \sum_{i=1}^{n} \frac{(Z^{(i)\top} Z^{(i)})^2}{\mu^4} \right\|_2 + 2\mu^2 \left\| \frac{1}{n} \sum_{i=1}^{n} \frac{(E^\top Z^{(i)} + Z^{(i)\top} E)^2}{\mu^4} \right\|_2 \\
&= \left\| \frac{1}{n} \sum_{i=1}^{n} Z^{(i)\top} Z^{(i)} \right\|_2 + \left\| \frac{1}{n} \sum_{i=1}^{n} E^\top Z^{(i)} + Z^{(i)\top} E \right\|_2 \\
&\qquad + \frac{2}{\mu^2} \left\| \frac{1}{n} \sum_{i=1}^{n} (Z^{(i)\top} Z^{(i)})^2 \right\|_2 + \frac{2}{\mu^2} \left\| \frac{1}{n} \sum_{i=1}^{n} (E^\top Z^{(i)} + Z^{(i)\top} E)^2 \right\|_2 .
\end{aligned}
$$

These four terms are bounded by Claims 5.6.6, 5.6.7, 5.6.8 and 5.6.9 respectively, notice that the second term dominates the first and the third, and the forth term dominates the second. So finally we have:

$$\begin{aligned}
&\mu^2 \log \left\| \frac{1}{n} \sum_{i=1}^{n} \exp\left(\frac{Z^{(i)\top} Z^{(i)} + E^\top Z^{(i)} + Z^{(i)\top} E}{\mu^2} \right) \right\|_2 \\
\leq & \left(\frac{3 \times 4}{\sqrt{C} K} + \frac{96 L^2}{K} \right) \frac{\eta}{nr} \\
\leq & \frac{97 L^2}{K} \frac{\eta}{nr} \\
= & \frac{1}{4CL} \frac{\eta}{nr}, \text{with choice of } K = 388 C L^3 = O(L^3).
\end{aligned}$$

■

Lemma 5.6.3. *If* $0 \preccurlyeq A$, *then* $\operatorname{tr}(AB) \leq \operatorname{tr}(A) \cdot \|B\|_2$.

Proof. Since $0 \preccurlyeq A$, $A = \sum_j \sigma_j u_j u_j^\top$ with $\sigma_j \geq 0$.

$$\begin{aligned}
\operatorname{tr}(AB) &= \operatorname{tr}\left(\sum_j \sigma_j u_j u_j^\top B \right) \\
&= \sum_j \sigma_j \operatorname{tr}(u_j u_j^\top B) \\
&= \sum_j \sigma_j u_j^\top B u_j \\
&\leq \sum_j \sigma_j \cdot \|B\|_2 \\
&= \operatorname{tr}(A) \cdot \|B\|_2 .
\end{aligned}$$

■

Lemma 5.6.4. *$(A+B)^\top(A+B) \preccurlyeq 2A^\top A+2B^\top B$, and $\left[(A+B)^\top(A+B)\right]^2 \preccurlyeq 8(A^\top A)^2+8(B^\top B)^2$.*

Proof.

$$\begin{aligned} 2A^\top A + 2B^\top B - (A+B)^\top(A+B) &= A^\top A + B^\top B - A^\top B - B^\top A \\ &= (A-B)^\top(A-B) \\ &\succcurlyeq 0. \end{aligned}$$

$$\begin{aligned} \left[(A+B)^\top(A+B)\right]^2 &\preccurlyeq (2A^\top A + 2B^\top B)^2 \\ &\preccurlyeq 2[2(A^\top A)]^2 + 2[2(B^\top B)]^2 \\ &\preccurlyeq 8(A^\top A)^2 + 8(B^\top B)^2. \end{aligned}$$

■

The following claims were used in Lemma 5.6.2. Recall that $\mathcal{A}$ satisfies dRIP condition 5.2.3, $u = \mathcal{A}[V] \in \mathbb{R}^n$ for some $V \in \mathbb{R}^{d_1 \times d_2}$ satisfying $\|V\|_F \in [\frac{1}{4}, 1], \|V\|_* \leq 2\sqrt{2r}$, $Z^{(i)} = \eta w_i^*(u_i A_i - S^*)$, and $\Phi_{\text{dc}}(w) \leq \frac{3}{Cr}$ by assumption of Lemma 5.6.2. We have the following:

- $\|A_i\|_2 \leq \rho$ by the boundedness property of dRIP condition 5.2.3;
- $|u_i| = |\langle A_i, V\rangle| \leq \|A_i\|_2 \|V\|_* \leq \rho 2\sqrt{2r} \leq L\sqrt{r}\rho$ assuming $2\sqrt{2} \leq L$;
- $\|S^*\|_2 \leq \|S^*\|_F \leq L, \|E^*\|_2 \leq \frac{1}{K\sqrt{r}}$ by the decomposition property of dRIP condition;
- $\|E\|_2^2 \leq \Phi_{\text{op}}(w) \leq \Phi_{\text{dc}}(w) \leq \frac{3}{Cr}$ by Claim 5.6.1.

Claim 5.6.5. $\left\|\frac{Z^{(i)\top}Z^{(i)}+E^\top Z^{(i)}+Z^{(i)\top}E}{\mu^2}\right\|_2 \le 1.$

Proof.

$$\begin{aligned}
\left\|Z^{(i)\top}Z^{(i)}\right\|_2 &= \eta^2 w_i^{*2}\left\|(u_iA_i - S^*)^\top(u_iA_i - S^*)\right\|_2 \\
&\le 2\eta^2 w_i^{*2}\left(u_i^2\left\|A_i^\top A_i\right\|_2 + \left\|S^{*\top}S^*\right\|_2\right) \text{ (Lemma 5.6.4)} \\
&\le 2\eta^2 w_i^{*2}(L^2 r\rho^4 + L^2) \\
&\le 4\eta^2 w_i^{*2}L^2 r\rho^4.
\end{aligned}$$

$$\begin{aligned}
\left\|E^\top Z^{(i)} + Z^{(i)\top}E\right\|_2 &\le 2\left\|E^\top Z^{(i)}\right\|_2 \le 2\left\|E\right\|_2 \cdot \left\|Z^{(i)}\right\|_2 \\
&= 2\eta w_i^*\left\|E\right\|_2 \cdot \left\|u_iA_i - S^*\right\|_2 \\
&\le 2\eta w_i^*\left\|E\right\|_2 \cdot (|u_i|\left\|A_i\right\|_2 + \left\|S^*\right\|_2) \\
&\le 2\eta w_i^*\frac{2}{\sqrt{Cr}}(L\sqrt{r}\rho^2 + L) \\
&\le 8\eta w_i^*\frac{L\rho^2}{\sqrt{C}}.
\end{aligned}$$

Putting them together:

$$\begin{aligned}
\left\|\frac{Z^{(i)\top}Z^{(i)} + E^\top Z^{(i)} + Z^{(i)\top}E}{\mu^2}\right\|_2 &\le \frac{1}{\mu^2}\left(\left\|Z^{(i)\top}Z^{(i)}\right\|_2 + \left\|E^\top Z^{(i)} + Z^{(i)\top}E\right\|_2\right) \\
&\le Cr\log d \cdot (4\eta^2 w_i^{*2}L^2 r\rho^4 + 8\eta w_i^*\frac{L\rho^2}{\sqrt{C}}) \\
&\le \frac{16\sqrt{C}L}{K} \le 1 \text{ with sufficiently large } K. \qquad \blacksquare
\end{aligned}$$

Claim 5.6.6. $\left\|\frac{1}{n}\sum_{i=1}^{n} Z^{(i)\top} Z^{(i)}\right\|_2 \leq \frac{4L^2}{K\log d}\cdot\frac{\eta}{nr}$.

Proof.

$$\begin{aligned}
\sum_{i=1}^{n}\left\|Z^{(i)\top}Z^{(i)}\right\|_2 &= \sum_{i=1}^{n}\eta^2 w_i^{*2}\left\|(u_iA_i - S^*)^\top(u_iA_i - S^*)\right\|_2 \\
&\leq 2\sum_{i=1}^{n}\eta^2 w_i^{*2}\left(u_i^2\left\|A_i^\top A_i\right\|_2 + \left\|S^{*\top}S^*\right\|_2\right) \\
&\leq 2\eta\sum_{i=1}^{n}\eta w_i^*(w_i^*u_i^2\rho^2 + w_i^*L^2) \\
&\leq 2\eta\frac{1}{Kr\rho^2\log d}(L\rho^2 + L^2) \\
&\leq \frac{4L^2}{K\log d}\frac{\eta}{r}.
\end{aligned}$$

$$\implies \left\|\frac{1}{n}\sum_{i=1}^{n} Z^{(i)\top}Z^{(i)}\right\|_2 \leq \frac{1}{n}\sum_{i=1}^{n}\left\|Z^{(i)\top}Z^{(i)}\right\|_2 \leq \frac{4L^2}{K\log d}\cdot\frac{\eta}{nr}. \qquad \blacksquare$$

Claim 5.6.7. $\left\|\frac{1}{n}\sum_{i=1}^{n} E^\top Z^{(i)} + Z^{(i)\top} E\right\|_2 \le \frac{4}{\sqrt{C}K} \cdot \frac{\eta}{nr}$.

Proof.

$$\begin{aligned}
\left\|\frac{1}{n}\sum_{i=1}^{n} E^\top Z^{(i)} + Z^{(i)\top} E\right\|_2 &= \frac{1}{n}\left\|E^\top \sum_{i=1}^{n} Z^{(i)} + \sum_{i=1}^{n} {Z^{(i)}}^\top E\right\|_2 \\
&\le \frac{2}{n}\left\|E^\top \sum_{i=1}^{n} Z^{(i)}\right\|_2 \\
&\le \frac{2}{n}\|E\|_2 \cdot \left\|\sum_{i=1}^{n} Z^{(i)}\right\|_2 \\
&= \frac{2}{n}\|E\|_2 \cdot \|\eta G^* - \eta S^*\|_2 \\
&\le \frac{2}{n}\|E\|_2 \cdot \eta \|E^*\|_2 \\
&\le \frac{2}{n} \cdot \frac{2}{\sqrt{Cr}} \cdot \frac{\eta}{K\sqrt{r}} \\
&= \frac{4}{\sqrt{C}K}\frac{\eta}{nr}.
\end{aligned}$$

■

Claim 5.6.8. $\frac{2}{\mu^2}\left\|\frac{1}{n}\sum_{i=1}^n (Z^{(i)\top}Z^{(i)})^2\right\|_2 \le O\left(\frac{CL^4}{K^3 r\rho^2\log^2 d}\right)\cdot\frac{\eta}{nr}$.

Proof.

$$\begin{aligned}
\left\|\sum_{i=1}^n (Z^{(i)\top}Z^{(i)})^2\right\|_2 &\le 8\sum_{i=1}^n \eta^4 w_i^{*4}\left(u_i^4\cdot\left\|(A_i^\top A_i)^2\right\|_2 + \left\|(S^{*\top}S^*)^2\right\|_2\right) \quad \text{(Lemma 5.6.4)}\\
&\le 8\eta\sum_{i=1}^n \eta^3 w_i^{*2}(w_i^{*2}u_i^4\rho^4 + w_i^{*2}L^4)\\
&\le 8\eta\cdot\frac{1}{K^3r^3\rho^6\log^3 d}\left[\sum_{i=1}^n w_i^{*2}\rho^4 u_i^4 + \sum_{i=1}^n w_i^{*2}L^4\right]\\
&\le 8\eta\cdot\frac{1}{K^3r^3\rho^6\log^3 d}\left[\left(\sum_{i=1}^n w_i^*\rho^2 u_i^2\right)^2 + \left(\sum_{i=1}^n w_i^* L^2\right)^2\right]\\
&\le 8\eta\cdot\frac{1}{K^3r^3\rho^6\log^3 d}(\rho^4L^2 + L^4)\\
&\le \frac{16L^4}{K^3r^2\rho^2\log^3 d}\cdot\frac{\eta}{r}\\
&\le O\left(\frac{L^4}{K^3r^2\rho^2\log^3 d}\right)\cdot\frac{\eta}{r},
\end{aligned}$$

$$\implies \frac{2}{\mu^2}\left\|\frac{1}{n}\sum_{i=1}^n (Z^{(i)\top}Z^{(i)})^2\right\|_2 \le O\left(\frac{CL^4}{K^3r\rho^2\log^2 d}\right)\cdot\frac{\eta}{nr}.$$

■

Claim 5.6.9. $\frac{2}{\mu^2}\left\|\frac{1}{n}\sum_{i=1}^{n}(E^\top Z^{(i)} + Z^{(i)\top}E)^2\right\|_2 \le \frac{96L^2}{K}\cdot\frac{\eta}{nr}$.

Proof.

$$\begin{aligned}
\frac{2}{\mu^2}\left\|\frac{1}{n}\sum_{i=1}^{n}(E^\top Z^{(i)} + Z^{(i)\top}E)^2\right\|_2 &\le \frac{2}{\mu^2}\frac{1}{n}\sum_{i=1}^{n}\left\|(E^\top Z^{(i)} + Z^{(i)\top}E)^2\right\|_2 \\
&\le \frac{2}{\mu^2}\frac{1}{n}\sum_{i=1}^{n}4\left\|E^\top Z^{(i)}{Z^{(i)}}^\top E\right\|_2 \\
&\le \frac{2}{\mu^2}\frac{1}{n}\sum_{i=1}^{n}4\|E\|_2^2\left\|Z^{(i)\top}Z^{(i)}\right\|_2 \\
&\le \frac{8}{\mu^2}\|E\|_2^2\cdot\frac{1}{n}\sum_{i=1}^{n}\left\|Z^{(i)\top}Z^{(i)}\right\|_2 \\
&\le \frac{8}{\mu^2}\cdot\frac{3}{Cr}\cdot\frac{4L^2}{K\log d}\frac{\eta}{nr} \text{ (Claim 5.6.6)} \\
&= \frac{96L^2}{K}\cdot\frac{\eta}{nr}.
\end{aligned}$$

■

Lemma 5.6.10 (Golden–Thompson inequality (Thompson, 1965))**.** *For two $n\times n$ Hermitian matrices A and B:*

$$\operatorname{tr}\big(\exp(A+B)\big) \le \operatorname{tr}\big(\exp(A)\exp(B)\big).$$

Proof of Lemma 5.5.6. Algorithm 5.3 has a nested for loop that's repeated for $N'\times N = O(n\log d\log\frac{1}{\delta}r\rho^2)$ times. The major step in the loop is line 11: $s_t \leftarrow \operatorname{argmax}_{s\in[0,\eta]}\Phi(w_t + se_i)$, which is equivalent to $\operatorname{argmin}_{s\in[0,\eta]} Cr\Phi_{\text{op}}(w+se_i) + \frac{s}{4L} - su_i^2$. Recall that $\Phi_{\text{op}}(w) = \min_{\|S\|_F\le L\|w\|_1}\Big(\mu^2\log\left[F(G_w - S)\right]\Big)$ where $F(E) = \operatorname{tr}\exp\left(\frac{E^\top E}{\mu^2}\right)$. Note that $\mu^2\log\left[F(E)\right]$ is convex in E.

First we show that $\Phi_{\text{op}}(w)$ is convex in w, i.e., given w_1, w_2, $\Phi_{\text{op}}\left(\frac{1}{2}(w_1+w_2)\right) \leq \frac{1}{2}\big(\Phi_{\text{op}}(w_1)+\Phi_{\text{op}}(w_2)\big)$.

Let $w_3 = \frac{1}{2}(w_1+w_2)$, and $G_k = \sum_{i=1}^{n}(w_k)_i u_i A_i$ for $k=1,2,3$. Suppose S_1, S_2 attain the minimum for $\Phi_{\text{op}}(w_1)$, $\Phi_{\text{op}}(w_2)$ respectively, i.e., $\Phi_{\text{op}}(w_1) = \mu^2 \log\left[F(G_1 - S_1)\right]$ and $\Phi_{\text{op}}(w_2) = \mu^2 \log\left[F(G_2 - S_2)\right]$.

Let $S_3 = \frac{1}{2}(S_1+S_2)$. Notice that $G_3 = \frac{1}{2}(G_1+G_2)$, so $G_3 - S_3 = \frac{1}{2}\left(G_1 - S_1 + G_2 - S_2\right)$. Since $\|S_3\|_F \leq \frac{1}{2}(\|S_1\|_F + \|S_2\|_F) \leq \frac{1}{2}L(\|w_1\|_1 + \|w_2\|_1) = L\|w_3\|_1$, S_3 is a valid argument for $\Phi_{\text{op}}(w_3)$, therefore

$$\begin{aligned}
\Phi_{\text{op}}(w_3) &\leq \mu^2 \log\left[F(G_3 - S_3)\right] \\
&\leq \frac{1}{2}\Big(\mu^2 \log\left[F(G_1 - S_1)\right] + \mu^2 \log\left[F(G_2 - S_2)\right]\Big) \\
&= \frac{1}{2}\big(\Phi_{\text{op}}(w_1) + \Phi_{\text{op}}(w_2)\big).
\end{aligned}$$

Line 11 is equivalent to minimizing $Cr\Phi_{\text{op}}(w + se_i) + \frac{s}{4L} - su_i^2$, which is convex in s for a fixed w, over a bounded interval $[0, \eta]$, so the minimization needs to evaluate $\Phi_{\text{op}}(w + se_i)$ for $O(\text{polylog}(d))$ different values of s. Evaluating Φ_{op} is also a minimization which can be solved by computing SVD on G_w and evaluating $F(G_w - S)$ in time $O(d^\omega)$ for $\text{polylog}(d)$ various constructions of S. Overall finding the optimal value of s takes time $O(d^\omega \cdot \text{polylog}(d))$, and the algorithm's total running time is $O(nd^\omega \,\text{polylog}\,(d) \log \frac{1}{\delta} r\rho^2)$. ■

5.6.3 Omitted Proofs: From wRIP to dRIP Condition

Here we show that the dRIP condition 5.2.3 is implied by the wRIP condition 5.2.2, given proper choices of parameters within a constant factor. Notice that in the wRIP condition, we have a low-rank constraint on the input matrix, i.e., $\text{rank}(X) \le r$, and in dRIP we have a norm constraint instead, i.e., $\|V\|_F \in [\frac{1}{4}, 1], \|V\|_* \le 2\sqrt{2r}$. To make use of the wRIP condition of $\mathcal{A}$, we will decompose V into low-rank matrices, so that wRIP condition applies to each of the low-rank matrices. Though the rank of V is arbitrary, we can still upper bound its numerical rank based on the norm constraint.

First we will introduce a low-rank decomposition, and an upper bound on the sum of their Frobenius norms. This is the matrix version of the shelling-decomposition in Lemma 15 for vectors in (Kelner et al., 2023a).

Lemma 5.6.11 (Low-rank decomposition)**.** *Given $V \in \mathbb{R}^{d_1 \times d_2}$ with $\text{Rank}_\text{n}(V) = \frac{\|V\|_*^2}{\|V\|_F^2} = \nu$, and let $V = \sum \sigma_i u_i v_i^\top$ be its SVD with σ_i in descending order. Decompose V into sum of rank-r matrices, i.e., write $V = \sum_{\ell=1}^{\ell=k} V^{(\ell)}$ where $V^{(\ell)} = \sum_{i=(\ell-1)r+1}^{i=\ell r} \sigma_i u_i v_i^\top$. Then we have $\sum_{\ell=2}^{k} \left\|V^{(\ell)}\right\|_F \le \sqrt{\frac{\nu}{r}} \|V\|_F$.*

Proof. Note that $V^{(1)}$ is the rank-r approximation of V, and $V^{(\ell)}$'s are constructed using disjoint singular values and vectors in groups of size r, and are orthogonal to each other.

Denote $\sigma_i\left(V^{(\ell)}\right)$ as the i^{th} largest singular value of $V^{(\ell)}$.

$$\left\|V^{(\ell+1)}\right\|_F \leq \sqrt{r}\cdot\sigma_1\left(V^{(\ell+1)}\right) \leq \sqrt{r}\cdot\sigma_r\left(V^{(\ell)}\right) \leq \sqrt{r}\cdot\frac{\left\|V^{(\ell)}\right\|_*}{r},$$
$$\sum_{\ell=2}^{k}\left\|V^{(\ell)}\right\|_F \leq \frac{\sqrt{r}}{r}\cdot\sum_{\ell=1}^{k-1}\left\|V^{(\ell)}\right\|_* \leq \frac{\sqrt{r}}{r}\cdot\sum_{\ell=1}^{k}\left\|V^{(\ell)}\right\|_* = \frac{\sqrt{r}}{r}\cdot\|V\|_*,$$
$$\frac{\|V\|_*^2}{\|V\|_F^2} \leq \nu \implies \sum_{\ell=2}^{k}\left\|V^{(\ell)}\right\|_F \leq \sqrt{\frac{\nu}{r}}\|V\|_F. \qquad \blacksquare$$

Now we are ready to prove Lemma 5.5.1, which states that wRIP implies dRIP condition. The proof uses similar techniques as in the second part of Lemma 17 (Kelner et al., 2023a) for vector recovery.

Proof of Lemma 5.5.1.

Boundedness property: satisfied by assumption $\|A_i\|_2 \leq \rho\ \forall i$.

Isometry property: Consider $V \in \mathbb{R}^{d_1\times d_2}$ *s.t.* $\|V\|_F \in [\frac{1}{4}, 1]$ and $\|V\|_* \leq 2\sqrt{2r}$. Need to show $\frac{1}{L} \leq \sum_{i=1}^{n} w_i^*\langle A_i, V\rangle^2 \leq L$.

Let $L = 25L'$, $K \geq 1$ and $r = \frac{r'}{12800L^2K^2}$. $\nu = \text{Rank}_{\text{n}}(V) = \frac{\|V\|_*^2}{\|V\|_F^2} \leq 128r$. By Lemma 5.6.11, decompose V into rank-r' matrices so that we can apply the (r', L')-wRIP property of $\mathcal{A}$.

$$\sum_{\ell=2}^{k}\left\|V^{(\ell)}\right\|_F \leq \sqrt{\frac{\nu}{r'}}\|V\|_F \leq \frac{1}{10LK}\|V\|_F \leq \frac{1}{10}\|V\|_F,$$
$$\|V\|_F \geq \left\|V_{(r')}\right\|_F = \left\|V^{(1)}\right\|_F = \left\|V - \sum_{\ell=2}^{k}V^{(\ell)}\right\|_F \geq \|V\|_F - \sum_{\ell=2}^{k}\left\|V^{(\ell)}\right\|_F \geq \frac{9}{10}\|V\|_F.$$

Let $B_i = \sqrt{w_i^*} A_i$, so that $\sum_{i=1}^n w_i^* \langle A_i, V\rangle^2 = \sum_{i=1}^n \langle B_i, V\rangle^2 = \|\sum_{i=1}^n \langle B_i, V\rangle e_i\|_2^2$.

Lower bound:

$$\begin{aligned}
\left\|\sum_{i=1}^n \langle B_i, V\rangle e_i\right\|_2 &\geq \left\|\sum_{i=1}^n \langle B_i, V^{(1)}\rangle e_i\right\|_2 - \left\|\sum_{i=1}^n \langle B_i, \sum_{\ell=2}^k V^{(\ell)}\rangle e_i\right\|_2 \\
&\geq \left\|\sum_{i=1}^n \langle B_i, V^{(1)}\rangle e_i\right\|_2 - \sum_{\ell=2}^k \left\|\sum_{i=1}^n \langle B_i, V^{(\ell)}\rangle e_i\right\|_2 \\
&= \sqrt{\sum_{i=1}^n \langle B_i, V^{(1)}\rangle^2} - \sum_{\ell=2}^k \sqrt{\sum_{i=1}^n \langle B_i, V^{(\ell)}\rangle^2} \\
&\geq \sqrt{\frac{1}{L'} \cdot \left\|V^{(1)}\right\|_F^2} - \sum_{\ell=2}^k \sqrt{L' \cdot \left\|V^{(\ell)}\right\|_F^2} \\
&\geq \frac{0.9}{\sqrt{L'}} \cdot \|V\|_F - \frac{0.1}{L}\sqrt{L'} \cdot \|V\|_F \\
&= \frac{4.5}{\sqrt{L}} \cdot \|V\|_F - \frac{0.02}{\sqrt{L}} \cdot \|V\|_F.
\end{aligned}$$

Taking the square: $\sum_{i=1}^n w_i^* \langle A_i, V\rangle^2 \geq \frac{4.48^2}{L} \cdot \|V\|_F^2 \geq \frac{4.48^2}{16L} \geq \frac{1}{L}$.

Upper bound:

$$\begin{aligned}
\left\|\sum_{i=1}^{n}\langle B_i, V\rangle e_i\right\|_2 &\le \left\|\sum_{i=1}^{n}\langle B_i, V^{(1)}\rangle e_i\right\|_2 + \sum_{\ell=2}^{k}\left\|\sum_{i=1}^{n}\langle B_i, V^{(\ell)}\rangle e_i\right\|_2 \\
&= \sqrt{\sum_{i=1}^{n}\langle B_i, V^{(1)}\rangle^2} + \sum_{\ell=2}^{k}\sqrt{\sum_{i=1}^{n}\langle B_i, V^{(\ell)}\rangle^2} \\
&\le \sqrt{L'}\cdot\left\|V^{(1)}\right\|_F + \sum_{\ell=2}^{k}\sqrt{L'}\cdot\left\|V^{(\ell)}\right\|_F \\
&\le \sqrt{L'}\cdot\|V\|_F^2 + \frac{0.1\sqrt{L'}}{L}\cdot\|V\|_F \\
&= \frac{\sqrt{L}}{5}\cdot\|V\|_F + \frac{0.02}{\sqrt{L}}\cdot\|V\|_F.
\end{aligned}$$

Taking the square: $\sum_{i=1}^{n} w_i^*\langle A_i, V\rangle^2 \le \frac{L}{25}\cdot\|V\|_F^2 + \frac{0.02^2}{L}\cdot\|V\|_F^2 + 0.008\|V\|_F^2 \le L\cdot\|V\|_F^2 \le L$.

Combining the lower bound and upper bound: $\frac{1}{L} \le \sum_{i=1}^{n} w_i^*\langle A_i, V\rangle^2 \le L$.

Decomposition property: Let $S = G_{(r')}$, the rank-r' approximation of $G = \sum_{i=1}^{n} w_i^* \langle A_i, V\rangle A_i$. Let $E = G - S$. Suffices to show $\|S\|_F \le L$ and $\|E\|_2 \le \frac{1}{K\sqrt{r}}$. We have

$$\begin{aligned}
\|S\|_F^2 = \langle S, S\rangle = \langle G, S\rangle &= \langle \sum_{i=1}^{n} w_i^* \langle A_i, V\rangle A_i, S\rangle = \langle \sum_{i=1}^{n} \langle B_i, V\rangle B_i, S\rangle = \sum_{i=1}^{n} \langle B_i, V\rangle \langle B_i, S\rangle \\
&= \langle \sum_{i=1}^{n} \langle B_i, V\rangle e_i, \sum_{j=1}^{n} \langle B_j, S\rangle e_j\rangle \\
&\le \left\| \sum_{i=1}^{n} \langle B_i, V\rangle e_i \right\|_2 \cdot \left\| \sum_{i=1}^{n} \langle B_i, S\rangle e_i \right\|_2 \\
&= \sqrt{\sum_{i=1}^{n} \langle B_i, V\rangle^2} \cdot \sqrt{\sum_{i=1}^{n} \langle B_i, S\rangle^2} \\
&= \sqrt{\sum_{i=1}^{n} w_i^* \langle A_i, V\rangle^2} \cdot \sqrt{\sum_{i=1}^{n} w_i^* \langle A_i, S\rangle^2} \\
&\le \sqrt{L} \cdot \sqrt{L'} \cdot \|S\|_F \\
&\le \frac{L}{5} \|S\|_F,
\end{aligned}$$

which implies $\|S\|_F \le \frac{L}{5} \le L$, and consequently,

$$\|E\|_2 = \sigma_{r'+1}(G) \le \sigma_{r'}(G) = \sigma_{r'}(S) \le \sqrt{\frac{\|S\|_F^2}{r'}} \le \frac{L}{5\sqrt{r'}} = \frac{L}{400\sqrt{2}LK\sqrt{r}} \le \frac{1}{K\sqrt{r}}. \quad \blacksquare$$

5.7 Conclusion and Future Work

In this paper, we pose and study the matrix sensing problem in a natural semi-random model. We relax the standard RIP assumption on the input sensing matrices to a much

weaker condition where an unknown subset of the sensing matrices satisfies RIP while the rest are arbitrary.

For this semi-random matrix sensing problem, existing non-convex objectives can have bad local optima. In this work, we employ an iterative reweighting approach using a weight oracle to overcome the influence of the semi-random input. Our solution is inspired by previous work on semi-random sparse vector recovery, where we exploit the structural similarities between linear regression on sparse vectors and matrix sensing on low-rank matrices.

Looking forward, we believe our approach can serve as a starting point for designing more efficient and robust algorithms for matrix sensing, as well as for other low-rank matrix and sparse vector problems in the semi-random model.

Acknowledgement

We thank Rong Ge for helpful discussions. Xing Gao is supported in part by NSF awards ECCS-2217023 and CCF-2307106. Yu Cheng is supported in part by NSF Award CCF-2307106.

APPENDIX

COPYRIGHT AGREEMENT

IEEE COPYRIGHT AND CONSENT FORM

To ensure uniformity of treatment among all contributors, other forms may not be substituted for this form, nor may any wording of the form be changed. This form is intended for original material submitted to the IEEE and must accompany any such material in order to be published by the IEEE. Please read the form carefully and keep a copy for your files.

Non-Center-Based Clustering Under Bilu-Linial Stability

Ms. Xing Gao and Prof. Lev Reyzin

2025 IEEE International Symposium on Information Theory (ISIT)

COPYRIGHT TRANSFER

The undersigned hereby assigns to The Institute of Electrical and Electronics Engineers, Incorporated (the "IEEE") all rights under copyright that may exist in and to: (a) the Work, including any revised or expanded derivative works submitted to the IEEE by the undersigned based on the Work; and (b) any associated written or multimedia components or other enhancements accompanying the Work.

GENERAL TERMS

1. The undersigned represents that he/she has the power and authority to make and execute this form.
2. The undersigned agrees to indemnify and hold harmless the IEEE from any damage or expense that may arise in the event of a breach of any of the warranties set forth above.
3. The undersigned agrees that publication with IEEE is subject to the policies and procedures of the IEEE PSPB Operations Manual.
4. In the event the above work is not accepted and published by the IEEE or is withdrawn by the author(s) before acceptance by the IEEE, the foregoing copyright transfer shall be null and void. In this case, IEEE will retain a copy of the manuscript for internal administrative/record-keeping purposes.
5. For jointly authored Works, all joint authors should sign, or one of the authors should sign as authorized agent for the others.
6. The author hereby warrants that the Work and Presentation (collectively, the "Materials") are original and that he/she is the author of the Materials. To the extent the Materials incorporate text passages, figures, data or other material from the works of others, the author has obtained any necessary permissions. Where necessary, the author has obtained all third party permissions and consents to grant the license above and has provided copies of such permissions and consents to IEEE

You have indicated that you DO wish to have video/audio recordings made of your conference presentation under terms and conditions set forth in "Consent and Release."

CONSENT AND RELEASE

1. ln the event the author makes a presentation based upon the Work at a conference hosted or sponsored in whole or in part by the IEEE, the author, in consideration for his/her participation in the conference, hereby grants the IEEE the unlimited, worldwide, irrevocable

permission to use, distribute, publish, license, exhibit, record, digitize, broadcast, reproduce and archive, in any format or medium, whether now known or hereafter developed: (a) his/her presentation and comments at the conference; (b) any written materials or multimedia files used in connection with his/her presentation; and (c) any recorded interviews of him/her (collectively, the "Presentation"). The permission granted includes the transcription and reproduction of the Presentation for inclusion in products sold or distributed by IEEE and live or recorded broadcast of the Presentation during or after the conference.

2. In connection with the permission granted in Section 1, the author hereby grants IEEE the unlimited, worldwide, irrevocable right to use his/her name, picture, likeness, voice and biographical information as part of the advertisement, distribution and sale of products incorporating the Work or Presentation, and releases IEEE from any claim based on right of privacy or publicity.

BY TYPING IN YOUR FULL NAME BELOW AND CLICKING THE SUBMIT BUTTON, YOU CERTIFY THAT SUCH ACTION CONSTITUTES YOUR ELECTRONIC SIGNATURE TO THIS FORM IN ACCORDANCE WITH UNITED STATES LAW, WHICH AUTHORIZES ELECTRONIC SIGNATURE BY AUTHENTICATED REQUEST FROM A USER OVER THE INTERNET AS A VALID SUBSTITUTE FOR A WRITTEN SIGNATURE.

Xing Gao	18-04-2025
Signature	Date (dd-mm-yyyy)

Information for Authors

AUTHOR RESPONSIBILITIES

The IEEE distributes its technical publications throughout the world and wants to ensure that the material submitted to its publications is properly available to the readership of those publications. Authors must ensure that their Work meets the requirements as stated in section 8.2.1 of the IEEE PSPB Operations Manual, including provisions covering originality, authorship, author responsibilities and author misconduct. More information on IEEE's publishing policies may be found at http://www.ieee.org/publications_standards/publications/rights/authorrightsresponsibilities.html Authors are advised especially of IEEE PSPB Operations Manual section 8.2.1.B12: "It is the responsibility of the authors, not the IEEE, to determine whether disclosure of their material requires the prior consent of other parties and, if so, to obtain it." Authors are also advised of IEEE PSPB Operations Manual section 8.1.1B: "Statements and opinions given in work published by the IEEE are the expression of the authors."

RETAINED RIGHTS/TERMS AND CONDITIONS

- Authors/employers retain all proprietary rights in any process, procedure, or article of manufacture described in the Work.
- Authors/employers may reproduce or authorize others to reproduce the Work, material extracted verbatim from the Work, or derivative works for the author's personal use or for company use, provided that the source and the IEEE copyright notice are indicated, the copies are not used in any way that implies IEEE endorsement of a product or service of any employer, and the copies themselves are not offered for sale.
- Although authors are permitted to re-use all or portions of the Work in other works, this does not include granting third-party requests for reprinting, republishing, or other types of re-use.The IEEE Intellectual Property Rights office must handle all such third-party requests.
- Authors whose work was performed under a grant from a government funding agency are free to fulfill any deposit mandates from

that funding agency.

AUTHOR ONLINE USE

- **Personal Servers**. Authors and/or their employers shall have the right to post the accepted version of IEEE-copyrighted articles on their own personal servers or the servers of their institutions or employers without permission from IEEE, provided that the posted version includes a prominently displayed IEEE copyright notice and, when published, a full citation to the original IEEE publication, including a link to the article abstract in IEEE Xplore. Authors shall not post the final, published versions of their papers.
- **Classroom or Internal Training Use.** An author is expressly permitted to post any portion of the accepted version of his/her own IEEE-copyrighted articles on the author's personal web site or the servers of the author's institution or company in connection with the author's teaching, training, or work responsibilities, provided that the appropriate copyright, credit, and reuse notices appear prominently with the posted material. Examples of permitted uses are lecture materials, course packs, e-reserves, conference presentations, or in-house training courses.
- **Electronic Preprints.** Before submitting an article to an IEEE publication, authors frequently post their manuscripts to their own web site, their employer's site, or to another server that invites constructive comment from colleagues. Upon submission of an article to IEEE, an author is required to transfer copyright in the article to IEEE, and the author must update any previously posted version of the article with a prominently displayed IEEE copyright notice. Upon publication of an article by the IEEE, the author must replace any previously posted electronic versions of the article with either (1) the full citation to the IEEE work with a Digital Object Identifier (DOI) or link to the article abstract in IEEE Xplore, or (2) the accepted version only (not the IEEE-published version), including the IEEE copyright notice and full citation, with a link to the final, published article in IEEE Xplore.

Questions about the submission of the form or manuscript must be sent to the publication's editor.
Please direct all questions about IEEE copyright policy to:
IEEE Intellectual Property Rights Office, copyrights@ieee.org, +1-732-562-3966

CITED LITERATURE

[Agarwal et al. , 2012] Agarwal, A., Negahban, S., and Wainwright, M. J.: Fast global convergence of gradient methods for high-dimensional statistical recovery. The Annals of Statistics, pages 2452–2482, 2012.

[Angelidakis et al. , 2018] Angelidakis, H., Awasthi, P., Blum, A., Chatziafratis, V., and Dan, C.: Bilu-linial stability, certified algorithms and the independent set problem. arXiv preprint arXiv:1810.08414, 2018.

[Angelidakis et al. , 2017] Angelidakis, H., Makarychev, K., and Makarychev, Y.: Algorithms for stable and perturbation-resilient problems. In Proceedings of the 49th Annual ACM SIGACT Symposium on Theory of Computing, pages 438–451, 2017.

[Antoniadis et al. , 2023] Antoniadis, A., Boyar, J., Eliás, M., Favrholdt, L. M., Hoeksma, R., Larsen, K. S., Polak, A., and Simon, B.: Paging with succinct predictions. In International Conference on Machine Learning, pages 952–968. PMLR, 2023.

[Arora et al. , 1998] Arora, S., Lund, C., Motwani, R., Sudan, M., and Szegedy, M.: Proof verification and the hardness of approximation problems. Journal of the ACM (JACM), 45(3):501–555, 1998.

[Arora and Safra, 1998] Arora, S. and Safra, S.: Probabilistic checking of proofs: A new characterization of np. Journal of the ACM (JACM), 45(1):70–122, 1998.

[Ashraphijuo et al. , 2017] Ashraphijuo, M., Aggarwal, V., and Wang, X.: On deterministic sampling patterns for robust low-rank matrix completion. IEEE Signal Processing Letters, 25(3):343–347, 2017.

[Attias et al. , 2025] Attias, I., Gao, X., and Reyzin, L.: Learning-augmented algorithms for boolean satisfiability. arXiv preprint arXiv:2505.06146, 2025.

[Austrin, 2007] Austrin, P.: Balanced max 2-sat might not be the hardest. In Proceedings of the thirty-ninth annual ACM symposium on Theory of computing, pages 189–197, 2007.

[Avidor et al. , 2005] Avidor, A., Berkovitch, I., and Zwick, U.: Improved approximation algorithms for max nae-sat and max sat. In International Workshop on Approximation and Online Algorithms, pages 27–40. Springer, 2005.

[Awasthi et al. , 2012] Awasthi, P., Blum, A., and Sheffet, O.: Center-based clustering under perturbation stability. Information Processing Letters, 112(1-2):49–54, 2012.

[Balcan et al. , 2020] Balcan, M.-F., Haghtalab, N., and White, C.: k-center clustering under perturbation resilience. ACM Transactions on Algorithms (TALG), 16(2):1–39, 2020.

[Balcan and Liang, 2016] Balcan, M. F. and Liang, Y.: Clustering under perturbation resilience. SIAM Journal on Computing, 45(1):102–155, 2016.

[Bampis et al. , 2025] Bampis, E., Escoffier, B., Fotakis, D., Patsilinakos, P., and Xefteris, M.: Polynomial time learning-augmented algorithms for np-hard permutation problems. arXiv preprint arXiv:2502.00841, 2025.

[Bampis et al. , 2024] Bampis, E., Escoffier, B., and Xefteris, M.: Parsimonious learning-augmented approximations for dense instances of np-hard problems. In International Conference on Machine Learning, pages 2700–2714. PMLR, 2024.

[Baumer and Schuler, 2003] Baumer, S. and Schuler, R.: Improving a probabilistic 3-sat algorithm by dynamic search and independent clause pairs. In International Conference on Theory and Applications of Satisfiability Testing, pages 150–161. Springer, 2003.

[Behsaz and Salavatipour, 2015] Behsaz, B. and Salavatipour, M. R.: On minimum sum of radii and diameters clustering. Algorithmica, 73(1):143–165, 2015.

[Ben-David and Reyzin, 2014] Ben-David, S. and Reyzin, L.: Data stability in clustering: A closer look. Theoretical Computer Science, 558:51–61, 2014.

[Bhojanapalli and Jain, 2014] Bhojanapalli, S. and Jain, P.: Universal matrix completion. In International Conference on Machine Learning, pages 1881–1889. PMLR, 2014.

[Bhojanapalli et al. , 2014] Bhojanapalli, S., Jain, P., and Sanghavi, S.: Tighter low-rank approximation via sampling the leveraged element. In Proceedings of the twenty-sixth annual ACM-SIAM symposium on Discrete algorithms, pages 902–920. SIAM, 2014.

[Bhojanapalli et al. , 2016] Bhojanapalli, S., Neyshabur, B., and Srebro, N.: Global optimality of local search for low rank matrix recovery. Advances in Neural Information Processing Systems, 29, 2016.

[Bilu and Linial, 2012] Bilu, Y. and Linial, N.: Are stable instances easy? Combinatorics, Probability and Computing, 21(5):643–660, 2012.

[Blum and Spencer, 1995] Blum, A. and Spencer, J.: Coloring random and semi-random k-colorable graphs. Journal of Algorithms, 19(2):204–234, 1995.

[Boufounos and Baraniuk, 2008] Boufounos, P. T. and Baraniuk, R. G.: 1-bit compressive sensing. In 2008 42nd Annual Conference on Information Sciences and Systems, pages 16–21. IEEE, 2008.

[Brakensiek et al. , 2024] Brakensiek, J., Huang, N., and Zwick, U.: Tight approximability of max 2-sat and relatives, under ugc. In Proceedings of the 2024 Annual ACM-SIAM Symposium on Discrete Algorithms (SODA), pages 1328–1344. SIAM, 2024.

[Braverman et al. , 2024] Braverman, V., Dharangutte, P., Shah, V., and Wang, C.: Learning-augmented maximum independent set. In Approximation, Randomization, and Combinatorial Optimization. Algorithms and Techniques (APPROX/RANDOM 2024), pages 24–1. Schloss Dagstuhl–Leibniz-Zentrum für Informatik, 2024.

[Burer and Monteiro, 2003] Burer, S. and Monteiro, R. D.: A nonlinear programming algorithm for solving semidefinite programs via low-rank factorization. Mathematical Programming, 95(2):329–357, 2003.

[Cai and Zhou, 2013] Cai, T. and Zhou, W.-X.: A max-norm constrained minimization approach to 1-bit matrix completion. Journal of Machine Learning Research, 14:3619–3647, 2013.

[Calabro et al. , 2009] Calabro, C., Impagliazzo, R., and Paturi, R.: The complexity of satisfiability of small depth circuits. In International Workshop on Parameterized and Exact Computation, pages 75–85. Springer, 2009.

[Candès et al. , 2011] Candès, E. J., Li, X., Ma, Y., and Wright, J.: Robust principal component analysis? Journal of the ACM (JACM), 58(3):1–37, 2011.

[Candes and Plan, 2011] Candes, E. J. and Plan, Y.: Tight oracle inequalities for low-rank matrix recovery from a minimal number of noisy random measurements. IEEE Transactions on Information Theory, 57(4):2342–2359, 2011.

[Candes and Recht, 2008] Candes, E. J. and Recht, B.: Exact low-rank matrix completion via convex optimization. In 2008 46th Annual Allerton Conference on Communication, Control, and Computing, pages 806–812. IEEE, 2008.

[Candès and Recht, 2009] Candès, E. J. and Recht, B.: Exact matrix completion via convex optimization. Foundations of Computational mathematics, 9(6):717–772, 2009.

[Candes et al. , 2006] Candes, E. J., Romberg, J. K., and Tao, T.: Stable signal recovery from incomplete and inaccurate measurements. Communications on Pure and Applied Mathematics: A Journal Issued by the Courant Institute of Mathematical Sciences, 59(8):1207–1223, 2006.

[Candès and Tao, 2010] Candès, E. J. and Tao, T.: The power of convex relaxation: Near-optimal matrix completion. IEEE transactions on information theory, 56(5):2053–2080, 2010.

[Capoyleas et al. , 1991] Capoyleas, V., Rote, G., and Woeginger, G.: Geometric clusterings. Journal of Algorithms, 12(2):341–356, 1991.

[Charikar and Panigrahy, 2001] Charikar, M. and Panigrahy, R.: Clustering to minimize the sum of cluster diameters. In Proceedings of the thirty-third annual ACM symposium on Theory of computing, pages 1–10, 2001.

[Chatziafratis et al. , 2017] Chatziafratis, V., Roughgarden, T., and Vondrák, J.: Stability and recovery for independence systems. arXiv preprint arXiv:1705.00127, 2017.

[Chen and Li, 2019] Chen, J. and Li, X.: Model-free nonconvex matrix completion: Local minima analysis and applications in memory-efficient kernel pca. Journal of Machine Learning Research, 20(142):1–39, 2019.

[Chen et al. , 2023] Chen, J., Ng, M. K., and Wang, D.: Quantizing heavy-tailed data in statistical estimation:(near) minimax rates, covariate quantization, and uniform recovery. IEEE Transactions on Information Theory, 70(3):2003–2038, 2023.

[Chen and Li, 2022] Chen, Y. and Li, X.: Determining the number of factors in high-dimensional generalized latent factor models. Biometrika, 109(3), 2022.

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

[Chen and Li, 2024] Chen, Y. and Li, X.: A note on entrywise consistency for mixed-data matrix completion. Journal of Machine Learning Research, 25(343):1–66, 2024.

[Chen et al. , 2020] Chen, Y., Chi, Y., Fan, J., Ma, C., and Yan, Y.: Noisy matrix completion: Understanding statistical guarantees for convex relaxation via nonconvex optimization. SIAM journal on optimization, 30(4):3098–3121, 2020.

[Cheng and Ge, 2018] Cheng, Y. and Ge, R.: Non-convex matrix completion against a semi-random adversary. In Conference On Learning Theory, pages 1362–1394. PMLR, 2018.

[Cohen-Addad et al. , 2024] Cohen-Addad, V., d'Orsi, T., Gupta, A., Lee, E., and Panigrahi, D.: Learning-augmented approximation algorithms for maximum cut and related problems. Advances in Neural Information Processing Systems, 37:25961–25980, 2024.

[Cook, 1971] Cook, S. A.: The complexity of theorem-proving procedures. In Proceedings of the third annual ACM symposium on Theory of computing, pages 151–158, 1971.

[Davenport et al. , 2014] Davenport, M. A., Plan, Y., Van Den Berg, E., and Wootters, M.: 1-bit matrix completion. Information and Inference: A Journal of the IMA, 3(3):189–223, 2014.

[De Sa et al. , 2015] De Sa, C., Re, C., and Olukotun, K.: Global convergence of stochastic gradient descent for some non-convex matrix problems. In International conference on machine learning, pages 2332–2341. PMLR, 2015.

[Demmel et al. , 2007] Demmel, J., Dumitriu, I., and Holtz, O.: Fast linear algebra is stable. Numerische Mathematik, 108(1):59–91, 2007.

[Devanur and Hayes, 2009] Devanur, N. R. and Hayes, T. P.: The adwords problem: online keyword matching with budgeted bidders under random permutations. In Proceedings of the 10th ACM conference on Electronic commerce, pages 71–78, 2009.

[Dinur, 2016] Dinur, I.: Mildly exponential reduction from gap-3sat to polynomial-gap label-cover. In Electronic colloquium on computational complexity ECCC; research reports, surveys and books in computational complexity, page 128, 2016.

[Doddi et al. , 2000] Doddi, S. R., Marathe, M. V., Ravi, S. S., Taylor, D. S., and Widmayer, P.: Approximation algorithms for clustering to minimize the sum of diameters. In Scandinavian Workshop on Algorithm Theory, pages 237–250. Springer, 2000.

[Dong et al. , 2025] Dong, Y., Peng, P., and Vakilian, A.: Learning-augmented streaming algorithms for approximating max-cut. In 16th Innovations in Theoretical Computer Science Conference (ITCS 2025), pages 44–1. Schloss Dagstuhl–Leibniz-Zentrum für Informatik, 2025.

[Eamaz et al. , 2023] Eamaz, A., Yeganegi, F., and Soltanalian, M.: One-bit matrix completion with time-varying sampling thresholds. In 2023 International Conference on Sampling Theory and Applications (SampTA), pages 1–5. IEEE, 2023.

[Eamaz et al. , 2024] Eamaz, A., Yeganegi, F., and Soltanalian, M.: Matrix completion from one-bit dither samples. IEEE Transactions on Signal Processing, 2024.

[Ergun et al. , 2021] Ergun, J. C., Feng, Z., Silwal, S., Woodruff, D., and Zhou, S.: Learning-augmented k-means clustering. In International Conference on Learning Representations, 2021.

[Fazel et al. , 2008] Fazel, M., Candes, E., Recht, B., and Parrilo, P.: Compressed sensing and robust recovery of low rank matrices. In 2008 42nd Asilomar Conference on Signals, Systems and Computers, pages 1043–1047. IEEE, 2008.

[Feige, 2007] Feige, U.: Refuting smoothed 3cnf formulas. In 48th Annual IEEE Symposium on Foundations of Computer Science (FOCS'07), pages 407–417. IEEE, 2007.

[Feige and Goemans, 1995] Feige, U. and Goemans, M.: Approximating the value of two power proof systems, with applications to max 2sat and max dicut. In Proceedings third Israel symposium on the theory of computing and systems, pages 182–189. IEEE, 1995.

[Feige and Kilian, 2001] Feige, U. and Kilian, J.: Heuristics for semirandom graph problems. Journal of Computer and System Sciences, 63(4):639–671, 2001.

[Fortnow and Gasarch, 2023] Fortnow, L. and Gasarch, W.: #cliq is #p-complete. https://www.cs.umd.edu/users/gasarch/BLOGPAPERS/sharpclique.pdf, 2023. Accessed: Dec 2023.

[Foucart et al. , 2020] Foucart, S., Needell, D., Pathak, R., Plan, Y., and Wootters, M.: Weighted matrix completion from non-random, non-uniform sampling patterns. IEEE Transactions on Information Theory, 67(2):1264–1290, 2020.

[Freitag et al. , 2021] Freitag, J., Mohammadi, N., Potukuchi, A., and Reyzin, L.: On the geometry of stable steiner tree instances. arXiv preprint arXiv:2109.13457, 2021.

[Gamlath et al. , 2022] Gamlath, B., Lattanzi, S., Norouzi-Fard, A., and Svensson, O.: Approximate cluster recovery from noisy labels. In Conference on Learning Theory, pages 1463–1509. PMLR, 2022.

[Gao and Cheng, 2024] Gao, X. and Cheng, Y.: Robust matrix sensing in the semi-random model. Advances in Neural Information Processing Systems, 36, 2024.

[Gao and Reyzin, 2025] Gao, X. and Reyzin, L.: Non-center-based clustering under bilu-linial stability. In 2025 IEEE International Symposium on Information Theory (ISIT). IEEE, 2025.

[Ge et al. , 2017] Ge, R., Jin, C., and Zheng, Y.: No spurious local minima in nonconvex low rank problems: A unified geometric analysis. In International Conference on Machine Learning, pages 1233–1242. PMLR, 2017.

[Ge et al. , 2016] Ge, R., Lee, J. D., and Ma, T.: Matrix completion has no spurious local minimum. Advances in neural information processing systems, 29, 2016.

[Ghoshal et al. , 2025] Ghoshal, S., Markarychev, K., and Markarychev, Y.: Constraint satisfaction problems with advice. In Proceedings of the 2025 Annual ACM-SIAM Symposium on Discrete Algorithms (SODA), pages 1202–1221. SIAM, 2025.

[Gibson et al. , 2010] Gibson, M., Kanade, G., Krohn, E., Pirwani, I. A., and Varadarajan, K.: On metric clustering to minimize the sum of radii. Algorithmica, 57:484–498, 2010.

[Gibson et al. , 2012] Gibson, M., Kanade, G., Krohn, E., Pirwani, I. A., and Varadarajan, K.: On clustering to minimize the sum of radii. SIAM Journal on Computing, 41(1):47–60, 2012.

[Goemans and Williamson, 1994] Goemans, M. X. and Williamson, D. P.: . 878-approximation algorithms for max cut and max 2sat. In Proceedings of the twenty-sixth annual ACM symposium on Theory of computing, pages 422–431, 1994.

[Gu et al. , 2016] Gu, Q., Wang, Z. W., and Liu, H.: Low-rank and sparse structure pursuit via alternating minimization. In Artificial Intelligence and Statistics, pages 600–609. PMLR, 2016.

[Gu et al. , 2023] Gu, Y., Song, Z., Yin, J., and Zhang, L.: Low rank matrix completion via robust alternating minimization in nearly linear time. arXiv preprint arXiv:2302.11068, 2023.

[Hansen and Jaumard, 1987] Hansen, P. and Jaumard, B.: Minimum sum of diameters clustering. Journal of Classification, 4(2):215–226, 1987.

[Hansen et al. , 2019] Hansen, T. D., Kaplan, H., Zamir, O., and Zwick, U.: Faster k-sat algorithms using biased-ppsz. In Proceedings of the 51st Annual ACM SIGACT Symposium on Theory of Computing, pages 578–589, 2019.

[Hardt and Wootters, 2014] Hardt, M. and Wootters, M.: Fast matrix completion without the condition number. In Conference on learning theory, pages 638–678. PMLR, 2014.

[Hastad, 2001] Hastad, J.: Some optimal inapproximability results. Journal of the ACM (JACM), 48(4):798–859, 2001.

[Heiman et al. , 2014] Heiman, E., Schechtman, G., and Shraibman, A.: Deterministic algorithms for matrix completion. Random Structures & Algorithms, 45(2):306–317, 2014.

[Hertli, 2014a] Hertli, T.: 3-sat faster and simpler—unique-sat bounds for ppsz hold in general. SIAM Journal on Computing, 43(2):718–729, 2014.

[Hertli, 2014b] Hertli, T.: Breaking the ppsz barrier for unique 3-sat. In International Colloquium on Automata, Languages, and Programming, pages 600–611. Springer, 2014.

[Hofmeister et al. , 2002] Hofmeister, T., Schöning, U., Schuler, R., and Watanabe, O.: A probabilistic 3—sat algorithm further improved. In STACS 2002: 19th Annual Symposium on Theoretical Aspects of Computer Science Antibes-Juan les Pins, France, March 14–16, 2002 Proceedings 19, pages 192–202. Springer, 2002.

[Huang et al. , 2022] Huang, B., Jiang, S., Song, Z., Tao, R., and Zhang, R.: Solving sdp faster: A robust ipm framework and efficient implementation. In 2022 IEEE 63rd Annual Symposium on Foundations of Computer Science (FOCS), pages 233–244. IEEE, 2022.

[Impagliazzo and Paturi, 2001] Impagliazzo, R. and Paturi, R.: On the complexity of k-sat. Journal of Computer and System Sciences, 62(2):367–375, 2001.

[Iwama and Tamaki, 2004] Iwama, K. and Tamaki, S.: Improved upper bounds for 3-sat. In Proceedings of the fifteenth annual ACM-SIAM symposium on Discrete algorithms, pages 328–328, 2004.

[Jain et al. , 2013] Jain, P., Netrapalli, P., and Sanghavi, S.: Low-rank matrix completion using alternating minimization. In Proceedings of the forty-fifth annual ACM symposium on Theory of computing, pages 665–674, 2013.

[Jiang et al. , 2020] Jiang, H., Kathuria, T., Lee, Y. T., Padmanabhan, S., and Song, Z.: A faster interior point method for semidefinite programming. In 2020 IEEE 61st annual symposium on foundations of computer science (FOCS), pages 910–918. IEEE, 2020.

[Johnson, 1973] Johnson, D. S.: Approximation algorithms for combinatorial problems. In Proceedings of the fifth annual ACM symposium on Theory of computing, pages 38–49, 1973.

[Johnson, 1967] Johnson, S. C.: Hierarchical clustering schemes. Psychometrika, 32(3):241–254, 1967.

[Karloff and Zwick, 1997] Karloff, H. and Zwick, U.: A 7/8-approximation algorithm for max 3sat? In Proceedings 38th Annual Symposium on Foundations of Computer Science, pages 406–415. IEEE, 1997.

[Karp, 1972] Karp, R. M.: Reducibility among combinatorial problemst. COMPLEXITY OF COMPUTER COMPUTATIONS, 1972.

[Karush, 1939] Karush, W.: Minima of functions of several variables with inequalities as side constraints. M. Sc. Dissertation. Dept. of Mathematics, Univ. of Chicago, 1939.

[Kelner et al. , 2024] Kelner, J., Li, J., Liu, A., Sidford, A., and Tian, K.: Semi-random matrix completion via flow-based adaptive reweighting. In The Thirty-eighth Annual Conference on Neural Information Processing Systems, 2024.

[Kelner et al. , 2023a] Kelner, J., Li, J., Liu, A. X., Sidford, A., and Tian, K.: Semi-random sparse recovery in nearly-linear time. In The Thirty Sixth Annual Conference on Learning Theory, pages 2352–2398. PMLR, 2023.

[Kelner et al. , 2023b] Kelner, J. A., Li, J., Liu, A., Sidford, A., and Tian, K.: Matrix completion in almost-verification time. In 2023 IEEE 64th Annual Symposium on Foundations of Computer Science (FOCS), pages 2102–2128. IEEE, 2023.

[Keshavan et al. , 2010] Keshavan, R. H., Montanari, A., and Oh, S.: Matrix completion from a few entries. IEEE transactions on information theory, 56(6):2980–2998, 2010.

[Kuhn and Tucker, 2013] Kuhn, H. W. and Tucker, A. W.: Nonlinear programming. In Traces and emergence of nonlinear programming, pages 247–258. Springer, 2013.

[Kullmann, 1999] Kullmann, O.: New methods for 3-sat decision and worst-case analysis. Theoretical Computer Science, 223(1-2):1–72, 1999.

[Kun and Reyzin, 2014] Kun, J. and Reyzin, L.: On coloring resilient graphs. In Mathematical Foundations of Computer Science 2014 - 39th International Symposium, MFCS 2014, Budapest, Hungary, August 25-29, 2014. Proceedings, Part II, eds. E. Csuhaj-Varjú, M. Dietzfelbinger, and Z. Ésik, volume 8635 of Lecture Notes in Computer Science, pages 517–528. Springer, 2014.

[Lan et al. , 2014] Lan, A. S., Studer, C., and Baraniuk, R. G.: Matrix recovery from quantized and corrupted measurements. In 2014 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), pages 4973–4977. IEEE, 2014.

[Lattanzi et al. , 2020] Lattanzi, S., Lavastida, T., Moseley, B., and Vassilvitskii, S.: Online scheduling via learned weights. In Proceedings of the Fourteenth Annual ACM-SIAM Symposium on Discrete Algorithms, pages 1859–1877. SIAM, 2020.

[Lee and Shraibman, 2013] Lee, T. and Shraibman, A.: Matrix completion from any given set of observations. Advances in Neural Information Processing Systems, 26, 2013.

[Levin, 1973] Levin, L. A.: Universal sequential search problems. Problems of Information Transmission, 9(3):265–266, 1973.

[Lewin et al. , 2002] Lewin, M., Livnat, D., and Zwick, U.: Improved rounding techniques for the max 2-sat and max di-cut problems. In International Conference on Integer Programming and Combinatorial Optimization, pages 67–82. Springer, 2002.

[Li et al. , 2016] Li, Y., Liang, Y., and Risteski, A.: Recovery guarantee of weighted low-rank approximation via alternating minimization. In International Conference on Machine Learning, pages 2358–2367. PMLR, 2016.

[Lindermayr and Megow, 2025] Lindermayr, A. and Megow, N.: Online index for algorithms with predictions. https://algorithms-with-predictions.github.io/, 2025.

[Liu et al. , 2019] Liu, G., Liu, Q., Yuan, X.-T., and Wang, M.: Matrix completion with deterministic sampling: Theories and methods. IEEE transactions on pattern analysis and machine intelligence, 43(2):549–566, 2019.

[Liu, 2018] Liu, S.: Chain, generalization of covering code, and deterministic algorithm for k-sat. In 45th International Colloquium on Automata, Languages, and Programming (ICALP 2018), pages 88–1. Schloss Dagstuhl–Leibniz-Zentrum für Informatik, 2018.

[Loh and Wainwright, 2012] Loh, P.-L. and Wainwright, M. J.: High-dimensional regression with noisy and missing data: Provable guarantees with nonconvexity. The Annals of Statistics, 40(3):1637–1664, 2012.

[Lykouris and Vassilvitskii, 2021] Lykouris, T. and Vassilvitskii, S.: Competitive caching with machine learned advice. Journal of the ACM (JACM), 68(4):1–25, 2021.

[Makarychev and Makarychev, 2020] Makarychev, K. and Makarychev, Y.: Certified algorithms: Worst-case analysis and beyond. In 11th Innovations in Theoretical Computer Science Conference (ITCS 2020). Schloss Dagstuhl-Leibniz-Zentrum für Informatik, 2020.

[Makarychev et al. , 2012] Makarychev, K., Makarychev, Y., and Vijayaraghavan, A.: Approximation algorithms for semi-random partitioning problems. In Proceedings of the forty-fourth annual ACM symposium on Theory of computing, pages 367–384, 2012.

[Makarychev et al. , 2014] Makarychev, K., Makarychev, Y., and Vijayaraghavan, A.: Bilu–linial stable instances of max cut and minimum multiway cut. In Proceedings of the

twenty-fifth annual ACM-SIAM symposium on Discrete algorithms, pages 890–906. SIAM, 2014.

[Manurangsi and Raghavendra, 2016] Manurangsi, P. and Raghavendra, P.: A birthday repetition theorem and complexity of approximating dense csps. arXiv preprint arXiv:1607.02986, 2016.

[Mathieu and Schudy, 2010] Mathieu, C. and Schudy, W.: Correlation clustering with noisy input. In Proceedings of the twenty-first annual ACM-SIAM symposium on Discrete Algorithms, pages 712–728. SIAM, 2010.

[Matuura and Matsui, 2001] Matuura, S. and Matsui, T.: 0.935-approximation randomized algorithm for max 2sat and its derandomization. IPSJ SIG technical reports, 2001(115):69–76, 2001.

[Meka et al. , 2009] Meka, R., Jain, P., and Dhillon, I.: Matrix completion from power-law distributed samples. Advances in neural information processing systems, 22, 2009.

[Mihalák et al. , 2011] Mihalák, M., Schöngens, M., Šrámek, R., and Widmayer, P.: On the complexity of the metric tsp under stability considerations. In SOFSEM 2011: Theory and Practice of Computer Science: 37th Conference on Current Trends in Theory and Practice of Computer Science, Novỳ Smokovec, Slovakia, January 22-28, 2011. Proceedings 37, pages 382–393. Springer, 2011.

[Mitzenmacher, 2020] Mitzenmacher, M.: Scheduling with predictions and the price of misprediction. In 11th Innovations in Theoretical Computer Science Conference (ITCS 2020), pages 14–1. Schloss Dagstuhl–Leibniz-Zentrum für Informatik, 2020.

[Mitzenmacher and Vassilvitskii, 2022] Mitzenmacher, M. and Vassilvitskii, S.: Algorithms with predictions. Communications of the ACM, 65(7):33–35, 2022.

[Monien and Speckenmeyer, 1985] Monien, B. and Speckenmeyer, E.: Solving satisfiability in less than 2n steps. Discrete Applied Mathematics, 10(3):287–295, 1985.

[Moser and Scheder, 2011] Moser, R. A. and Scheder, D.: A full derandomization of schöning's k-sat algorithm. In Proceedings of the forty-third annual ACM symposium on Theory of computing, pages 245–252, 2011.

[Negahban and Wainwright, 2012] Negahban, S. and Wainwright, M. J.: Restricted strong convexity and weighted matrix completion: Optimal bounds with noise. The Journal of Machine Learning Research, 13(1):1665–1697, 2012.

[Nguyen et al. , 2023] Nguyen, T., Chaturvedi, A., and Nguyen, H. L.: Improved learning-augmented algorithms for k-means and k-medians clustering. In International Conference on Learning Representations. International Conference on Learning Representations, 2023.

[Ni and Gu, 2016] Ni, R. and Gu, Q.: Optimal statistical and computational rates for one bit matrix completion. In Artificial Intelligence and Statistics, pages 426–434. PMLR, 2016.

[Nie et al. , 2018] Nie, F., Hu, Z., and Li, X.: Matrix completion based on non-convex low-rank approximation. IEEE Transactions on Image Processing, 28(5):2378–2388, 2018.

[Nocedal and Wright, 2006] Nocedal, J. and Wright, S. J.: Penalty and Augmented Lagrangian Methods, pages 497–528. New York, NY, Springer New York, 2006.

[Park et al. , 2017] Park, D., Kyrillidis, A., Carmanis, C., and Sanghavi, S.: Non-square matrix sensing without spurious local minima via the burer-monteiro approach. In Artificial Intelligence and Statistics, pages 65–74. PMLR, 2017.

[Paturi et al. , 2005] Paturi, R., Pudlák, P., Saks, M. E., and Zane, F.: An improved exponential-time algorithm for k-sat. Journal of the ACM (JACM), 52(3):337–364, 2005.

[Paturi et al. , 1997] Paturi, R., Pudlák, P., and Zane, F.: Satisfiability coding lemma. In Proceedings 38th Annual Symposium on Foundations of Computer Science, pages 566–574. IEEE, 1997.

[Perry and Wein, 2017] Perry, A. and Wein, A. S.: A semidefinite program for unbalanced multisection in the stochastic block model. In 2017 International Conference on Sampling Theory and Applications (SampTA), pages 64–67. IEEE, 2017.

[Pimentel-Alarcón et al. , 2016] Pimentel-Alarcón, D. L., Boston, N., and Nowak, R. D.: A characterization of deterministic sampling patterns for low-rank matrix completion. IEEE Journal of Selected Topics in Signal Processing, 10(4):623–636, 2016.

[Purohit et al. , 2018] Purohit, M., Svitkina, Z., and Kumar, R.: Improving online algorithms via ml predictions. Advances in Neural Information Processing Systems, 31, 2018.

[Qin and Watanabe, 2020] Qin, T. and Watanabe, O.: An improvement of the algorithm of hertli for the unique 3sat problem. Theoretical Computer Science, 806:70–80, 2020.

[Raghavendra and Tan, 2012] Raghavendra, P. and Tan, N.: Approximating csps with global cardinality constraints using sdp hierarchies. In Proceedings of the twenty-third annual ACM-SIAM symposium on Discrete Algorithms, pages 373–387. SIAM, 2012.

[Recht, 2011] Recht, B.: A simpler approach to matrix completion. Journal of Machine Learning Research, 12(12), 2011.

[Recht et al. , 2010] Recht, B., Fazel, M., and Parrilo, P. A.: Guaranteed minimum-rank solutions of linear matrix equations via nuclear norm minimization. SIAM review, 52(3):471–501, 2010.

[Rennie and Srebro, 2005] Rennie, J. D. and Srebro, N.: Fast maximum margin matrix factorization for collaborative prediction. In Proceedings of the 22nd international conference on Machine learning, pages 713–719, 2005.

[Rodošek, 1996] Rodošek, R.: A new approach on solving 3-satisfiability. In International Conference on Artificial Intelligence and Symbolic Mathematical Computing, pages 197–212. Springer, 1996.

[Rohatgi, 2020] Rohatgi, D.: Near-optimal bounds for online caching with machine learned advice. In Proceedings of the Fourteenth Annual ACM-SIAM Symposium on Discrete Algorithms, pages 1834–1845. SIAM, 2020.

[Roughgarden, 2019] Roughgarden, T.: Beyond worst-case analysis. Communications of the ACM, 62(3):88–96, 2019.

[Roughgarden, 2021] Roughgarden, T.: Beyond the worst-case analysis of algorithms. Cambridge University Press, 2021.

[Scheder, 2024] Scheder, D.: Ppsz is better than you think. TheoretiCS, 3, 2024.

[Scheder and Steinberger, 2017] Scheder, D. and Steinberger, J. P.: Ppsz for general k-sat-making hertli's analysis simpler and 3-sat faster. In 32nd Computational Complexity Conference (CCC 2017). Schloss-Dagstuhl-Leibniz Zentrum für Informatik, 2017.

[Schiermeyer, 1993] Schiermeyer, I.: Solving 3-satisfiability in less than 1, 579 n steps. In Computer Science Logic: 6th Workshop, CSL'92 San Miniato, Italy, September 28–October 2, 1992 Selected Papers 6, pages 379–394. Springer, 1993.

[Schoning, 1999] Schoning, T.: A probabilistic algorithm for k-sat and constraint satisfaction problems. In 40th Annual Symposium on Foundations of Computer Science (Cat. No. 99CB37039), pages 410–414. IEEE, 1999.

[Schütze et al. , 2008] Schütze, H., Manning, C. D., and Raghavan, P.: Introduction to information retrieval, volume 39. Cambridge University Press Cambridge, 2008.

[Selman et al. , 1996] Selman, B., Mitchell, D. G., and Levesque, H. J.: Generating hard satisfiability problems. Artificial intelligence, 81(1-2):17–29, 1996.

[Shen et al. , 2019] Shen, J., Awasthi, P., and Li, P.: Robust matrix completion from quantized observations. In The 22nd International Conference on Artificial Intelligence and Statistics, pages 397–407. PMLR, 2019.

[Sipser, 1996] Sipser, M.: Introduction to the theory of computation. ACM Sigact News, 27(1):27–29, 1996.

[Sun and Luo, 2016] Sun, R. and Luo, Z.-Q.: Guaranteed matrix completion via non-convex factorization. IEEE Transactions on Information Theory, 62(11):6535–6579, 2016.

[Thompson, 1965] Thompson, C. J.: Inequality with applications in statistical mechanics. Journal of Mathematical Physics, 6(11):1812–1813, 1965.

[Tropp, 2012] Tropp, J. A.: User-friendly tail bounds for sums of random matrices. Foundations of computational mathematics, 12(4):389–434, 2012.

[Tu et al. , 2016] Tu, S., Boczar, R., Simchowitz, M., Soltanolkotabi, M., and Recht, B.: Low-rank solutions of linear matrix equations via procrustes flow. In International Conference on Machine Learning, pages 964–973. PMLR, 2016.

[Valiant and Vazirani, 1985] Valiant, L. G. and Vazirani, V. V.: Np is as easy as detecting unique solutions. In Proceedings of the seventeenth annual ACM symposium on Theory of computing, pages 458–463, 1985.

[Vee et al. , 2010] Vee, E., Vassilvitskii, S., and Shanmugasundaram, J.: Optimal online assignment with forecasts. In Proceedings of the 11th ACM conference on Electronic commerce, pages 109–118, 2010.

[Vershynin, 2010] Vershynin, R.: Introduction to the non-asymptotic analysis of random matrices. arXiv preprint arXiv:1011.3027, 2010.

[Wang et al. , 2017] Wang, L., Zhang, X., and Gu, Q.: A unified computational and statistical framework for nonconvex low-rank matrix estimation. In Artificial Intelligence and Statistics, pages 981–990. PMLR, 2017.

[Wei and Zhang, 2020] Wei, A. and Zhang, F.: Optimal robustness-consistency trade-offs for learning-augmented online algorithms. Advances in Neural Information Processing Systems, 33:8042–8053, 2020.

[Williams, 2014] Williams, V. V.: Multiplying matrices in o (n2. 373) time. preprint, 2014.

[Xu et al. , 2017] Xu, P., Ma, J., and Gu, Q.: Speeding up latent variable gaussian graphical model estimation via nonconvex optimization. Advances in Neural Information Processing Systems, 30, 2017.

[Xu et al. , 2023] Xu, X., Shen, Y., Chi, Y., and Ma, C.: The power of preconditioning in overparameterized low-rank matrix sensing. In International Conference on Machine Learning, pages 38611–38654. PMLR, 2023.

[Zhang et al. , 2013] Zhang, H., He, W., Zhang, L., Shen, H., and Yuan, Q.: Hyperspectral image restoration using low-rank matrix recovery. IEEE transactions on geoscience and remote sensing, 52(8):4729–4743, 2013.

[Zhang et al. , 2018a] Zhang, X., Wang, L., and Gu, Q.: A unified framework for nonconvex low-rank plus sparse matrix recovery. In International Conference on Artificial Intelligence and Statistics, pages 1097–1107. PMLR, 2018.

[Zhang et al. , 2018b] Zhang, X., Wang, L., Yu, Y., and Gu, Q.: A primal-dual analysis of global optimality in nonconvex low-rank matrix recovery. In International conference on machine learning, pages 5862–5871. PMLR, 2018.

[Zhao et al. , 2015] Zhao, T., Wang, Z., and Liu, H.: A nonconvex optimization framework for low rank matrix estimation. Advances in Neural Information Processing Systems, 28, 2015.

[Zheng and Lafferty, 2016] Zheng, Q. and Lafferty, J.: Convergence analysis for rectangular matrix completion using burer-monteiro factorization and gradient descent. arXiv preprint arXiv:1605.07051, 2016.

[Zhu et al. , 2018] Zhu, Z., Li, Q., Tang, G., and Wakin, M. B.: Global optimality in low-rank matrix optimization. IEEE Transactions on Signal Processing, 66(13):3614–3628, 2018.

[Zhu et al. , 2021] Zhu, Z., Li, Q., Tang, G., and Wakin, M. B.: The global optimization geometry of low-rank matrix optimization. IEEE Transactions on Information Theory, 67(2):1308–1331, 2021.

[Zwick, 2000] Zwick, U.: Analyzing the max 2-sat and max di-cut approximation algorithms of feige and goemans. Submitted for Publication, 2000.

VITA

NAME Xing Gao

EDUCATION B.S., Electrical Engineering, Georgia Institute of Technology, Atlanta, GA, 2013
M.S., Electrical Engineering, Georgia Institute of Technology, Atlanta, GA, 2014
M.S., Computer Science, Georgia Institute of Technology, Atlanta, GA, 2020

TEACHING Computer Algorithms (MCS 401, Fall 2022 and Spring 2025)

PUBLICATIONS Xing Gao and Lev Reyzin. "Non-Center-Based Clustering Under Bilu-Linial Stability". In the 58th IEEE International Symposium on Information Theory (ISIT 2025).

Xing Gao and Yu Cheng. "Robust Matrix Sensing in the Semi-Random Model". In the 37th Conference on Neural Information Processing Systems (NeurIPS 2023).

Xing Gao, Thomas Maranzatto and Lev Reyzin. "A Unified Analysis of Dynamic Interactive Learning". In the 59th Annual Allerton Conference on Communication, Control, and Computing (Allerton 2023).

Xing Gao and Lev Reyzin. "An Interactive Search Game with Two Agents". In the 58th Annual Allerton Conference on Communication, Control, and Computing (Allerton 2022).