UNIVERSITY OF CALIFORNIA SAN DIEGO

Learning in the Presence of Adversaries

A dissertation submitted in partial satisfaction of the
requirements for the degree Doctor of Philosophy

in

Electrical Engineering (Communication Theory and Systems)

by

Ayush Jain

Committee in charge:

Professor Alon Orlitsky, Chair
Professor Tara Javidi
Professor Daniel Kane
Professor Arya Mazumdar

2023

The Dissertation of Ayush Jain is approved, and it is acceptable in quality and form for publication on microfilm and electronically.

University of California San Diego

2023

DEDICATION

Dedicated to my mother, Shimla Jain, for her continued love, support and encouragement throughout my life.

TABLE OF CONTENTS

LIST OF FIGURES

ACKNOWLEDGEMENTS

I extend my heartfelt thanks to my parents, Shimla and Kamlesh Jain whose sacrifices and endless encouragement have paved the way for all my endeavors in life. Their love for me has always been my strongest support. I also wish to express my appreciation to my dear sister, Shruti, and my brother, Manan, for infusing my life with joy, care, and unwavering companionship.

I am deeply grateful for the guidance and support of my advisor, Professor Alon Orlitsky. His tireless dedication to discussing research with me has left me indebted, and I've always felt encouraged to express my thoughts openly. Alon's down-to-earth behavior, patience, diligence, enthusiasm, optimism, and unwavering dedication have been a constant source of inspiration. I consider him not only a mentor but also a guardian, as his caring nature has provided me with a sense of security throughout this journey. His teachings have extended far beyond academia, enriching every facet of my life.

I would also like to thank my labmates Dheeraj Pichapati, Moein Falahatgar, Vaishakh Ravindrakumar, and Yi Hao, for fostering an inclusive, supportive, and friendly environment. Our insightful discussions have been invaluable. I have been privileged to collaborate with and receive guidance from Ananda Theertha Suresh, Jayadev Acharya, Gautam Kamath, Jerry Li, Clément Canonne, Rajat Sen, Weihao Kong, Abhimanyu Das during my Ph.D. journey.

During my time at UC San Diego, I was fortunate to have great teachers and mentors. I am thankful to my Ph.D. committee members Prof. Daniel Kane, Prof. Tara Javidi, and Prof. Arya Mazumdar for their willingness to dedicate their valuable time to serve on my committee, engage thoughtfully with my work, and offer invaluable feedback.

I extend my heartfelt thanks to Professor Rakesh K. Bansal, my M. Tech. thesis advisor at IIT Kanpur, whose introduction to information theory ignited my passion for research and kindled my love for this field.

I am fortunate to have been surrounded by an incredibly caring and supportive group of friends. To Akhil, Roshan, Sparsh, Siddharth, Yash, Stuti, Apoorva, Pranav, Mradula, Varun, Waquar, Yeohee, Deeksha, Sameeksha, Rohit, Ashwin, Akshay, Harshita—thank you for being an integral part of my journey and for creating countless cherished memories. I also extend my gratitude to my friends in India—Mukul, Anshik, Shivam, Sumit, Ajay, and Rajat—for their unwavering friendship and support.

I acknowledge the technical help and support of my other co-authors. I also thank them for allowing me to use the following manuscripts for my dissertation.

Chapter 2, in full, is a reprint of the material as it appears in Optimal robust learning of discrete distributions from batches 2020. Ayush Jain, Alon Orlitsky. In ICML 2020. The dissertation author was the primary investigator and author of this paper.

Chapter 3, in full, is a reprint of the material as it appears in A general method for robust learning from batches 2020. Ayush Jain, Alon Orlitsky. In Neurips 2020. The dissertation author was the primary investigator and author of this paper.

Chapter 4, in full, is a reprint of the material as it appears in Robust Density Estimation from Batches: The Best Things in Life are (Nearly) Free 2021. Ayush Jain, Alon Orlitsky. In ICML 2021. The dissertation author was the primary investigator and author of this paper.

Chapter 5, in full, is a reprint of the material as it appears in Efficient list-decodable regression using batches 2023. Abhimanyu Das, Ayush Jain, Weihao Kong, and Rajat Sen. In ICML 2023. The dissertation author was the primary investigator and author of this paper.

Chapter 6, in full, is a reprint of the material as it appears in Linear Regression using Heterogeneous Data Batches 2023. Abhimanyu Das, Ayush Jain, Rajat Sen, Weihao Kong, Abhimanyu Das, and Alon Orlitsky. Submitted in Neurips 2023. The dissertation author was the primary investigator and author of this paper.

Chapter 7, in full is a reprint of the material as it appears in Robust estimation for random graphs 2022. Jayadev Acharya, Ayush Jain, Gautam Kamath, Ananda Theertha Suresh, and Huanyu Zhang. In COLT 2022. The dissertation author was the primary investigator and author of this paper.

VITA

2015	Bachelor of Technology and Masters of Technology (Dual Degree) Electrical Engineering, Indian Institute of Technology, Kanpur
2017–2022	Master of Science, in Electrical Engineering (Communication Theory and Systems) University of California San Diego
2017–2023	Research Assistant, University of California, San Diego
2023	Doctor of Philosophy, in Electrical Engineering (Communication Theory and Systems) University of California San Diego

PUBLICATIONS

Jain, Ayush, and Rakesh K. Bansal. "Point-wise analysis of redundancy in SWLZ algorithm for ϕ-mixing sources." In 2015 IEEE Information Theory Workshop (ITW), pp. 1-5. IEEE, 2015.

Jain, Ayush, and Rakesh K. Bansal. "Exponential rates of convergence for waiting times and generalized AEP." In 2015 IEEE Information Theory Workshop (ITW), pp. 1-5. IEEE, 2015.

Jain, Ayush, and Rakesh K. Bansal. "On redundancy rate of FDLZ algorithm and its variants." In 2015 IEEE International Symposium on Information Theory (ISIT), pp. 1991-1995. IEEE, 2015.

Jain, Ayush, and Rakesh K. Bansal. "On point-wise redundancy rate of Bender-Wolf's variant of SWLZ algorithm." In 2016 IEEE Information Theory Workshop (ITW), pp. 116-120. IEEE, 2016.

Jain, Ayush, and Rakesh K. Bansal. "On optimality and redundancy of side information version of SWLZ." In 2017 IEEE International Symposium on Information Theory (ISIT), pp. 306-310. IEEE, 2017.

Jain, Ayush, and Himanshu Tyagi. "Effective memory shrinkage in estimation." In 2018 IEEE International Symposium on Information Theory (ISIT), pp. 1071-1075. IEEE, 2018.

Falahatgar, Moein, Ayush Jain, Alon Orlitsky, Venkatadheeraj Pichapati, and Vaishakh Ravindrakumar. "The limits of maxing, ranking, and preference learning." In International conference on machine learning, pp. 1427-1436. PMLR, 2018.

Hao, Yi, Ayush Jain, Alon Orlitsky, and Vaishakh Ravindrakumar. "Surf: A simple, universal, robust, fast distribution learning algorithm." Advances in Neural Information Processing Systems 33 (2020): 10881-10890.

Jain, Ayush, and Alon Orlitsky. "Optimal robust learning of discrete distributions from batches." In International Conference on Machine Learning, pp. 4651-4660. PMLR, 2020.

Jain, Ayush, and Alon Orlitsky. "A general method for robust learning from batches." Advances in Neural Information Processing Systems 33 (2020): 21775-21785.

Jain, Ayush, and Alon Orlitsky. "Linear-sample learning of low-rank distributions." Advances in Neural Information Processing Systems 33 (2020): 19201-19211.

Jain, Ayush, and Alon Orlitsky. "Robust density estimation from batches: The best things in life are (nearly) free." In International Conference on Machine Learning, pp. 4698-4708. PMLR, 2021.

Jain, Ayush, Alon Orlitsky, and Vaishakh Ravindrakumar. "Robust estimation algorithms don't need to know the corruption level." arXiv preprint arXiv:2202.05453 (2022).

Canonne, Clément L., Ayush Jain, Gautam Kamath, and Jerry Li. "The price of tolerance in distribution testing." In Conference on Learning Theory, pp. 573-624. PMLR, 2022.

Acharya, Jayadev, Ayush Jain, Gautam Kamath, Ananda Theertha Suresh, and Huanyu Zhang. "Robust estimation for random graphs." In Conference on Learning Theory, pp. 130-166. PMLR, 2022.

Hao, Yi, Ayush Jain, Alon Orlitsky, and Vaishakh Ravindrakumar. "TURF: Two-Factor, Universal, Robust, Fast Distribution Learning Algorithm." In International Conference on Machine Learning, pp. 8427-8445. PMLR, 2022.

Das, Abhimanyu, Ayush Jain, Weihao Kong, and Rajat Sen. "Efficient list-decodable regression using batches." In International Conference on Machine Learning, pp. 7025-7065. PMLR, 2023.

Jain, Ayush, Rajat Sen, Weihao Kong, Abhimanyu Das, and Alon Orlitsky. "Linear Regression using Heterogeneous Data Batches." Submitted to Neurips, 2023.

FIELDS OF STUDY

Major Field: Electrical Engineering (Communication Theory and Systems)

Studies in Data and Information Sciences
Professor Alon Orlitsky

ABSTRACT OF THE DISSERTATION

Learning in the Presence of Adversaries

by

Ayush Jain

Doctor of Philosophy in Electrical Engineering (Communication Theory and Systems)

University of California San Diego, 2023

Professor Alon Orlitsky, Chair

Modern applications, including natural language processing, sensor networks, collaborative filtering, and federated learning, necessitate data collection from diverse sources. However, these sources may be tainted by untrustworthy, erroneous, or adversarial data. Moreover, even in the absence of corruption, the sources might not conform to a shared underlying distribution. They could be categorized into different groups, with distinct and arbitrarily varying data distributions. For instance, consider movie recommendation systems where users rate movies. The ratings provided by different users can exhibit variations based on their genre preferences, highlighting the diversity in data distributions among sources.

In this thesis, we consider a range of issues within the aforementioned contexts:

1. Robust estimation of structured distributions, both discrete and continuous,

2. Robust classification,

3. List-decodable regression,

4. Mixed linear regression with small batches,

5. Robust parameter estimation in graph settings.

Previous approaches to these problems have suffered from limitations in terms of computational complexity, estimation accuracy, and sample complexity due to the presence of corrupted data sources.

This thesis introduces novel methodologies to address the limitations of previous approaches, focusing on robust learning from corrupted data sources. By doing so, it broadens the horizons for achieving precise distribution estimation, regression, classification, and parameter inference across diverse application domains.

Chapter 1

Introduction

Modern machine-learning applications have made remarkable progress, due in large part to the advancements in machine-learning techniques. These applications, however, rely heavily on the availability of a large amount of data. To meet this demand, data are often collected from a multitude of sources. However, this collection process is not without its challenges. The data aggregated from various sources can introduce noise, inaccuracies, faults, or even deliberate adversarial manipulation, compromising the integrity of the data.

Instances of such scenarios are prevalent across different domains. For instance, in sensor-based data collection, multiple sensors contribute data, and malfunctioning sensors might provide inaccurate readings. When estimating author word frequencies from numerous texts, misattributed texts can skew the results. User preference learning involves feedback from various users, some of whom might intentionally offer biased responses. Crowdsourcing platforms can feature unreliable workers, leading to untrustworthy data. Even in federated learning, where data comes from decentralized devices, some users might deviate significantly from the majority's data distribution.

Traditional robust learning setups assume that data is generated independently and identically distributed (i.i.d.) from a common distribution, with a fraction of the data being outliers. However, the presence of outliers places constraints on the learning process. The fraction of corrupt data imposes a fundamental limit on attainable accuracy, even when computational resources and dataset sizes tend towards infinity.

At first glance, the implications of this situation might appear pessimistic: the existence of an adversary corrupting a significant fraction of the data could lead to a large loss that is unavoidable. Yet, this apprehension does not necessarily encapsulate the entire reality.

Fortunately, in the above-mentioned applications, each source typically provides a batch consisting of multiple samples. This means that if a certain fraction of the sources are compromised, the corresponding batches contain corruption, while the remaining batches from the remaining fraction of the sources remain authentic and contain genuine samples.

This thesis delves into various fundamental learning paradigms, such as distribution learning, classification, and regression. By leveraging the inherent batch structure present in the data, we achieve significantly higher accuracy compared to approaches that ignore this structure. Furthermore, this work develops sample-efficient and polynomial-time algorithms for each of these learning scenarios, demonstrating their practical effectiveness through simulations.

Through a combination of innovative algorithms, novel theoretical analyses, and experimental validations, this thesis contributes to advancing robust machine-learning techniques in the face of challenges posed by corrupt, unreliable, diverse, and adversarial data sources.

1.1 Thesis Organization

Qiao and Valiant [125] showed that when batches are of size $\geq n$ and $\leq \beta$ fraction of the batches are corrupt then distribution can be learned to a min-max L_1 distance $\Theta(\beta/\sqrt{n})$, compared to the best possible distance $\Theta(\beta)$ achievable without batches. However, their algorithm ran in exponential time, and for some regimes required a suboptimal number of batches. Chapter 2 provides the first polynomial-time estimator that is optimal in the number of batches and achieves essentially the best possible estimation accuracy.

In the subsequent Chapter 3, we develop a general framework of robust learning from batches, and determine the limits of both distribution estimation, and notably, classification, over arbitrary, including continuous, domains. Building on this framework, we derive the first robust agnostic: (1) polynomial-time distribution estimation algorithms for structured distributions, including piecewise-polynomial, monotone, log-concave, and gaussian-mixtures, and also significantly improve their sample complexity; (2) classification algorithms, and also establish their near-optimal sample complexity; (3) computationally-efficient algorithms for the fundamental problem of interval-based classification that underlies nearly all natural-1-dimensional classification problems.

The results of the previous chapter raise questions regarding the optimal sample complexity for robustly learning structured distributions, stated explicitly in a concurrent work [31]. We answer this question in Chapter 4, showing that, perhaps surprisingly, up to logarithmic factors, the optimal sample complexity is the same as for genuine, non-adversarial, data! To establish the result, we reduce robust learning of approximately piecewise polynomial distributions to robust learning of the probability of all subsets of size at most k of a larger discrete domain and learn these probabilities in optimal sample complexity linear in k regardless of the domain size. In simulations, the algorithm runs very quickly and estimates distributions to essentially the accuracy achieved when all adversarial batches are removed. The results also imply the first polynomial-time sample-optimal algorithm for robust interval-based classification based on batched data.

Chapter 5 shows the efficacy of batch structures in the context of list-decodable linear regression. This chapter tackles scenarios where only a fraction $\alpha \in (0, 1]$ of batches contain genuine samples from a common distribution and the rest can contain arbitrary or even adversarial samples. When genuine batches have $\geq \tilde{\Omega}(1/\alpha)$ samples each, the proposed algorithm can efficiently find a small list of potential regression parameters, with a high probability that one of them is close to the true parameter. This is the first polynomial time algorithm for list-decodable linear regression, and its sample complexity scales nearly linearly with the dimension of the covariates. The polynomial time algorithm is made possible by the batch structure and may not be feasible without it, as suggested by a recent Statistical Query lower bound [53].

Chapter 6 examines scenarios where batches align with one of the k unknown subgroups, each potentially possessing distinct input distributions and linear regression models. Prior work [95] showed that with abundant small-batches, the regression vectors can be learned with only few, $\tilde{\Omega}(k^{3/2})$, batches of medium-size with $\tilde{\Omega}(\sqrt{k})$ samples each. However, the paper requires that the input distribution for all k subgroups be isotropic Gaussian, and states that removing this assumption is an “interesting and challenging problem". This chapter introduces a novel gradient-based algorithm that improves on the existing results in several ways. It extends the applicability of the algorithm by: (1) allowing the subgroups' underlying input distributions to be different, unknown, and heavy-tailed; (2) recovering all subgroups followed by a significant proportion of batches even for infinite k; (3) removing the separation requirement between the regression vectors; (4) reducing the number of batches and allowing smaller batch sizes. Moreover, the algorithm also accommodates sub-groups that are not targeted for recovery to exhibit arbitrary input-output relationships. Note that in contrast to the previous chapter, in this chapter we assume that no batches are adversarial, and develop algorithm that can operate with smaller batches and has a better sample complexity for learning linear regression models that generates α fraction of batches.

Finally, the concluding chapter 7 considers a situation where data exhibits a structure akin to batch structures in the presence of adversarial agents. In this chapter, we study the problem of robustly estimating the parameter p of an Erdős-Rényi random graph on n nodes, where a β fraction of nodes may be adversarially corrupted. After showing the deficiencies of canonical estimators, we design a computationally efficient spectral algorithm that estimates p up to accuracy $\tilde{O}(\sqrt{p(1-p)}/n + \beta\sqrt{p(1-p)}/\sqrt{n} + \beta/n)$ for $\beta < 1/60$. Furthermore, we give an inefficient algorithm with similar accuracy for all $\beta < 1/2$, the information-theoretic limit. Finally, we prove a nearly-matching statistical lower bound, showing that the error of our algorithms is optimal up to logarithmic factors.

Chapter 2

Optimal Robust Learning of Discrete Distributions from Batches

2.1 Introduction

2.1.1 Motivation

Estimating discrete distributions from their samples is a fundamental modern-science tenet. [86] showed that as the number of sample s grows, a k-symbol distribution can be learned to expected L_1 distance $\sim \sqrt{2(k-1)/(\pi s)}$ that we call the *information-theoretic limit*.

In many applications, some samples are inadvertently or maliciously corrupted. A simple and intuitive example shows that this erroneous data limits the extent to which a distribution can be learned, even with infinitely many samples.

Consider the extremely simple case of just two possible binary distributions: $(1, 0)$ and $(1-\beta, \beta)$. An adversary who observes a $1-\beta$ fraction of the samples and can determine the rest, could use the observed samples to learn the underlying distribution, and set the remaining samples to make the distribution appear to be $(1-\beta, \beta)$. By the triangle inequality, even with arbitrarily many samples, any estimator for p incurs an L_1 loss $\geq \beta$ for at least one of the two distributions. We call this the *adversarial lower bound*.

The example may seem to suggest a pessimistic conclusion. If an adversary can corrupt a β fraction of the data, a loss $\geq \beta$ is unavoidable. Fortunately, that is not necessarily so.

In many applications data is collected in batches, most of which are genuine, but some possibly corrupted. Here are a few examples. Data may be gathered by sensors, each providing a large amount of data, and some sensors may be faulty. The word frequency of an author may be estimated from several large texts, some of which are mis-attributed. Or user preferences may be learned by querying several users, but some users may intentionally bias their feedback.

Interestingly, even when a β-fraction of the batches are corrupted, the underlying distribution can be estimated to L_1 distance much lower than β. Consider for example just three n-sample batches, of which one is chosen adversarially. The underlying distribution can be learned from each genuine batch to expected L_1 distance $\sim \sqrt{2(k-1)/(\pi n)}$. It is easy to see that the average of the two estimates pairwise-closest in L_1 distance achieves a comparable expected distance that for large batch size n is much lower than β.

This raises the natural question of whether estimates from even more batches can be combined effectively to estimate distributions to within a distance that is not only much smaller than the β achieved when no batch information was utilized, but also significantly smaller than the $O(\sqrt{k/n})$ distance derived above when two batches were used. For example can the underlying distribution be learned to a small L_1 distance when, as in many practical examples, $n \le k$?

To formalize the problem, [125] considered learning a k-symbol distribution p whose samples are provided in batches of size $\ge n$. A total of m batches are provided, of which a fraction $\le \beta$ may be arbitrarily and adversarially corrupted, while in every other batch b the samples are drawn according a distribution p_b satisfying $||p_b - p||_1 \le \eta$, allowing for the possibility that slightly different distributions generate samples in each batch.

For this adversarial batch setting, they showed that for any alphabet size $k \ge 2$, and any number m of batches, the lowest achievable L_1 distance is $\ge \eta + \frac{\beta}{\sqrt{2n}}$. We refer to this as the *adversarial batch lower bound.*

For $\beta < 1/900$, they also derived an estimation algorithm that approximates p to L_1 distance $O(\max\{\eta+\beta/\sqrt{n}, \sqrt{(n+k)/(nm)}\})$, achieving the adversarial batch lower bound, for m large enough. Surprisingly therefore, not only can the underlying distribution be approximated to L_1 distance $O(\sqrt{k/n})$ that falls below β, but the distance diminishes as $\beta/\sqrt{n}$, independent of the alphabet size k.

Yet, the algorithm in [125] had three significant drawbacks. 1) it runs in time exponential in the alphabet size, hence impractical for most relevant applications; 2) its guarantees are limited to very small fractions of corrupted batches $\beta \geq 1/900$, hence do not apply to practically important ranges; 3) with m batches of size $\geq n$ each, the total number of samples is $\geq nm$, and for alphabet size $k \ll n$, the algorithm's distance guarantee falls short of the information-theoretic $\Theta(\sqrt{k/(nm)})$ limit.

In this paper we derive an algorithm that 1) runs in polynomial time in all parameters; 2) can tolerate any fraction of adversarial batches $\beta < 1/2$, though to derive concrete constant factors in the theoretical analysis, we assume $\beta \leq 0.4$; 3) achieves distortion $O(\max\{\eta + \beta\sqrt{\frac{\log(1/\beta)}{n}}, \sqrt{\frac{k}{nm}}\})$ that achieves the statistical limit in terms of the number nm of samples, and is optimal up to a small $O(\sqrt{\log(1/\beta)})$ factor from the adversarial batch lower bound.

The algorithm's computational efficiency, enables the first experiments of learning with adversarial batches. We tested the algorithm on simulated data with various adversarial-batch distributions and adversarial noise levels up to $\beta = 0.49$. The algorithm runs in a fraction of a second, and as shown in Section 2.3, estimates p nearly as well as an oracle that knows the identity of the adversarial batches.

To summarize, the algorithm runs in polynomial time, works for any adversarial fraction $\beta < 0.5$, is optimal in number of samples, and essentially optimal in batch size. It opens the door to practical robust estimation in sensor networks, federated learning, and collaborative filtering.

2.1.2 Problem Formulation

Let Δ_k be the collection of all distributions over $[k] = \{1, \ldots, k\}$. The L_1 distance between two distributions $p, q \in \Delta_k$ is

$$||p - q||_1 \triangleq \sum_{i \in [k]} |p(i) - q(i)| = 2 \cdot \max_{S \subseteq [k]} |p(S) - q(S)|.$$

We would like to estimate an unknown *target distribution* $p \in \Delta_k$ to a small L_1 distance from samples, some of which may be corrupted or even adversarial.

Specifically, let B be a collections of m batches of n samples each. Among these batches is an unknown collection of *good batches* $B_G \subseteq B$; each batch $b \in B_G$ in this collection has n independent samples $X_1^b, X_2^b, ..., X_n^b \sim p_b$ with $||p_b - p||_1 \leq \eta$. Furthermore, the batches of samples in B_G are independent of each other.

For the special case where $\eta = 0$, all samples in the good batches are generated by the target distribution $p = p_b$. Since the proofs and techniques are essentially the same for $\eta = 0$ and $\eta > 0$, for simplicity of presentation we assume that $\eta = 0$. We briefly discuss, at the end, how these results translate to the case $\eta > 0$.

The remaining set $B_A = B \setminus B_G$ of *adversarial batches* consists of arbitrary n samples each, that may even be chosen by an adversary, possibly based on the samples in the good batches. Let $\alpha = |B_G|/m$, and $\beta = |B_A|/m = 1 - \alpha$ be the fractions of good and adversarial batches, respectively.

Our goal is to use the m batches to return a distribution p^* such that $||p^* - p||_1$ is small or equivalently $|p(S) - p^*(S)|$ is small for all $S \subseteq [k]$.

2.1.3 Result Summary

In section 2.2 we derive a polynomial-time algorithm that returns an estimate p^* of p with the following properties.

Theorem 1. *For any given $\beta \leq 0.4$, n, k, and $m = \Omega(\frac{k}{\beta^2 \log(1/\beta)})$, Algorithm 2 runs in time polynomial in all parameters and its estimate p^* satisfies $||p^* - p||_1 \leq 100\beta\sqrt{\frac{\log(1/\beta)}{n}}$ with probability $\geq 1 - O(e^{-k})$.*

The theorem implies that our algorithm can achieve the adversarial lower bound to a small factor of $O(\sqrt{\log(1/\beta)})$ using the optimal number of samples. The next theorem shows that when the number of samples is not enough to achieve the adversarial batch lower bound our algorithm achieves the statistical lower bound.

Theorem 2. *For any given $\beta \leq 0.4$, n and k and m, Algorithm 2 runs in polynomial time, and its estimate p^* satisfies $||p^* - p||_1 \leq O(\max\{\beta\sqrt{\frac{\ln(1/\beta)}{n}}, \sqrt{\frac{k}{mn}}\})$ with probability $\geq 1 - O(e^{-k})$.*

The above theorem follows from Theorem 1 and a short proof appears in Appendix 2.7.

Note that our polynomial time algorithm achieves the statistical limits for L_1 distance and achieves the adversarial batch lower bounds to a small multiplicative factor of $O(\sqrt{\log(1/\beta)})$.

2.1.4 Comparison to Recent Results and Techniques

In a paper concurrent and independent of this work, [30] propose an algorithm that uses the sum of squares methodology to estimate p to the same distance as ours. Their algorithm needs $\tilde{\mathcal{O}}(\frac{(nk)^{\mathcal{O}(\log(1/\beta))}}{\beta^4})$ batches and has a run-time $\tilde{\mathcal{O}}(\frac{(nk)^{\mathcal{O}(\log^2(1/\beta))}}{\beta^{\mathcal{O}(\log(1/\beta))}})$. Both the sample complexity and run time are much higher than ours, and is quasi-polynomial. They also considered certain structured distributions, namely t-piecewise degree-d polynomial, not addressed in this paper. For this distribution class they provide an algorithm with similar quasi-polynomial run time and the number of batches required was quasi-polylogarithmic in domain size k, and quasi-polynomial in other parameters.

In the follow up work [76], we generalized our techniques to improve both the run time and the number of batches required for learning piece-wise polynomial distributions. We gave an algorithm that runs in polynomial time in all parameters and uses the number of batches $\Omega(\frac{t \cdot d \cdot \sqrt{n} \cdot \log(n/\beta)}{\beta^3})$, which has an optimal linear dependence on t and d and is independent of domain size k. Further, we developed first algorithm for robust *classification* in a similar adversarial batch setting.

Another follow up work [31], concurrent and independent to [76], combined their previous work [30] with the techniques presented here, and also obtained a polynomial time algorithm for learning piecewise-polynomial distributions, which requires $\Omega(\frac{t^2 \cdot d^2 \log^3 k \cdot \text{polylog}(n/\beta)}{\beta^2})$ batches.

2.1.5 Other Related Work

The current results extend several long lines of work on learning distributions and their properties.

The best approximation of a distribution with a given number of samples was determined up to the exact first-order constant for KL loss [22], and L_1 loss and χ^2 loss [86]. These settings do not allow adversarial examples, and some modification of the empirical estimates of the samples is often shown to be near optimal. This is not the case in the presence of adversarial samples, where the challenge is to devise algorithms that are efficient from both computational and sample viewpoints.

Our results also relate to classical robust-statistics work [142, 74]. There has also been significant recent work leading to practical distribution learning algorithms that are robust to adversarial contamination of the data. For example, [47, 99] presented algorithms for learning the mean and covariance matrix of high-dimensional sub-gaussian and other distributions with bounded fourth moments in presence of the adversarial samples. Their estimation guarantees are typically in terms of L_2, and do not yield the L_1- distance results required for discrete distributions.

The work was extended in [29] to the case when more than half of the samples are adversarial. Their algorithm returns a small set of candidate distributions one of which is a good approximate of the underlying distribution. For more extensive survey on robust learning algorithms in the continuous setting, see [135, 46].

Another motivation for this work derives from the practical federated-learning problem, where information arrives in batches [108, 109].

2.1.6 Preliminaries

We introduce notation that will help outline our approach and will be used in rest of the paper.

Throughout the paper, we use B' to denote a sub-collection of batches in B and use B'_G and B'_A for a sub-collection of batches in B_G and B_A, respectively. And S is used to denote a subset of $[k]$, we abbreviate singleton set of $[k]$ such as $\{j\}$ by j.

For any batch $b \in B$, we let $\bar{\mu}_b$ denote the empirical measure defined by samples in batch b. And for any sub-collection of batches $B' \subseteq B$, let $\bar{p}_{B'}$ denote the empirical measure defined by combined samples in all the batches in B'. We use two different symbols to distinguish the empirical distribution defined by an individual batch and the empirical distribution defined by a sub-collection of batches. Let $\mathbf{1}_S(.)$ denote the indicator random variable for set S. Thus, for any subset $S \subseteq [k]$,

$$\bar{\mu}_b(S) \triangleq \frac{1}{n} \sum_{i \in [n]} \mathbf{1}_S(X_i^b)$$

and

$$\bar{p}_{B'}(S) \triangleq \frac{1}{|B'|n} \sum_{b \in B'} \sum_{i \in [n]} \mathbf{1}_S(X_i^b) = \frac{1}{|B'|} \sum_{b \in B'} \bar{\mu}_b(S).$$

Note that $\bar{p}_{B'}$ is the mean of the empirical measures $\bar{\mu}_b$ defined by the batches $b \in B'$. For subset $S \subseteq [k]$, let $\text{med}(\bar{\mu}(S))$ be the median of the set of estimates $\{\bar{\mu}_b(S) : b \in B\}$. Note that the median has been computed using the estimates $\bar{\mu}_b(S)$ for all the batches in $b \in B$.

For $r \in [0, 1]$, we let $\mathrm{V}(r) \triangleq \frac{r(1-r)}{n}$, which we use to denote the variance of sum of n i.i.d. random variables distributed according to Bernoulli(r).

We pause briefly to note the following two properties of the function $\mathrm{V}(r)$ that we use later.

$$\forall\, r, s \in [0, 1],\ \mathrm{V}(r) \leq \frac{1}{4n} \text{ and } |\mathrm{V}(r) - \mathrm{V}(s)| \leq \frac{|r - s|}{n}. \tag{2.1}$$

Here the second property made use of the fact that the derivative $|V'(r)| \le 1/n$, $\forall\, r \in [0,1]$.

For $b \in B_G$, $\mathbf{1}_S(X_i^b)$ for $i \in [n]$ are i.i.d. with distribution $\mathbf{1}_S(X_i^b) \sim$ Bernoulli$(p(S))$. For $b \in B_G$, since $\bar{\mu}_b(S)$ is average of $\mathbf{1}_S(X_i^b)$, $i \in [n]$, therefore,

$$E[\,\bar{\mu}_b(S)\,] = p(S) \quad \text{and} \quad E[(\bar{\mu}_b(S) - p(S))^2] = \mathrm{V}(p(S)).$$

For any collection of batches $B' \subseteq B$ and subset $S \subseteq [k]$, the empirical probability $\bar{\mu}_b(S)$ of S based on batches $b \in B'$ will differ for the different batches. The empirical variance of these empirical probabilities $\bar{\mu}_b(S)$ for batches $b \in B'$ is denoted as

$$\overline{\mathrm{V}}_{B'}(S) \triangleq \frac{1}{|B'|} \sum_{b \in B'} (\bar{\mu}_b(S) - \bar{p}_{B'}(S))^2.$$

2.1.7 Organization of the Paper

In Section 2.2 we present the algorithm, its analysis along with the key insights used in developing the algorithm. Section 2.3 reports the performance of the algorithm on experiments performed on the simulated data.

2.2 Algorithm and its Analysis

At a high level, our algorithm removes the adversarial batches — which are "outliers" — possibly losing a small number of good batches as well in the process. The outlier removal method forms the backbone of many robust learning algorithms. Notably [47, 48] have used this idea to learn the mean of a high dimensional sub-gaussian distribution up to a small L_2 distance, even in an adversarial setting. The main challenge in designing a robust learning algorithm is actually the task of finding the outlier batches efficiently. Several new ideas are needed to identify the outlier batches in the setting considered here.

We begin by illustrating the difficulty of identifying the adversarial batches. Even if p is known, in general, one cannot determine whether a batch b has samples from p or from a distribution at a large L_1 distance from p. The key difficulty is that, for a batch having n samples from p, typically the difference between $\bar{\mu}_b(S)$ and $p(S)$ is large for some of the subsets among 2^k subsets of $[k]$. For example, consider batches of samples from a uniform distribution over k. The empirical distribution of the samples in any batch of size n is at an L_1 distance $\geq 2(1-n/k)$, which for the distributions with large domain size k can be up to two, which is the maximum L_1 distance between two distributions. To address this challenge, we use the following observation.

For a fixed subset $S \subseteq [k]$ and a good batch $b \in B_G$, $\bar{\mu}_b(S)$ has a sub-gaussian distribution subG$(p(S), \frac{1}{4n})$ and the variance is V$(p(S))$. Therefore, for a fixed subset S, most of the good batches assign the empirical probability $\bar{\mu}_b(S) \in p(S) \pm \tilde{O}(1/\sqrt{n})$. Moreover, the mean and the variance of $\bar{\mu}_b(S)$ for $b \in B_G$ converges to the expected values $p(S)$ and V$(p(S))$, respectively.

The collection of batches B along with good batches also includes a sub-collection B_A of adversarial batches that constitute up to an $\beta-$fraction of B. If for adversarial batches $b \in B_A$, the average difference between $\bar{\mu}_b(S)$ and $p(S)$ is within a few standard deviations $\tilde{O}(\frac{1}{\sqrt{n}})$, then these adversarial batches can only deviate the overall mean of empirical probabilities $\bar{\mu}_b(S)$ by $\tilde{O}(\frac{\beta}{\sqrt{n}})$ from $p(S)$. Hence, the mean of $\bar{\mu}_b(S)$ will deviates significantly from $p(S)$ only if for a large number of adversarial batches $b \in B_A$ empirical probability $\bar{\mu}_b(S)$ differ from $p(S)$ by quantity much larger than the standard deviation $\tilde{O}(\frac{1}{\sqrt{n}})$.

We quantify this effect by defining the *corruption score*. For a subset $S \subseteq [k]$, let

$$\text{med}(\bar{\mu}(S)) \triangleq \text{median}\{\bar{\mu}_b(S) : b \in B\}.$$

For a subset $S \subseteq [k]$ and a batch b, *corruption score* $\psi_b(S)$ is defined as

$$\psi_b(S) \triangleq \begin{cases} 0, \quad \text{if } |\bar{\mu}_b(S) - \text{med}(\bar{\mu}(S))| \leq 3\sqrt{\frac{\ln(6e/\beta)}{n}}, \\ (\bar{\mu}_b(S) - \text{med}(\bar{\mu}(S)))^2, \quad \text{else.} \end{cases}$$

Because $p(S)$ is not known, the above definition use median of $\bar{\mu}_b(S)$ as its proxy.

From the preceding discussion, it follows that for a fixed subset $S \subseteq [k]$, corruption score of most good batches w.r.t. S is zero, and adversarial batches that may have a significant effect on the overall mean of empirical probabilities have high corruption score $\psi_b(S)$.

The *corruption score* of a sub-collection B' w.r.t. a subset S is defined as the sum of the *corruption score* of batches in it, namely

$$\psi(B', S) \triangleq \sum_{b \in B'} \psi_b(S).$$

A high corruption score of B' w.r.t. a subset S indicates the presence of many batches $b \in B'$ for which the difference $|\bar{\mu}_b(S) - \text{med}(\bar{\mu}(S))|$ is large. Finally, for a sub-collection B' we define *corruption* as

$$\psi(B') \triangleq \max_{S \subseteq [k]} \psi(B', S).$$

Note that removing batches from a sub-collection reduces corruption. We can simply make corruption zero by removing all batches, but we would lose all the information as well. The proposed algorithm reduces the corruption below a threshold by removing a few batches while not sacrificing too many good batches in the process.

The remainder of this section assumes that the sub-collection of good batches B_G satisfies certain deterministic conditions. Lemma 3 shows that the stated conditions hold with high probability for sub-collection of good batches in B_G. Nothing is assumed about the adversarial batches, except that they form a $\leq \beta$ fraction of the overall batches B.

Conditions: Consider a collection of m batches B, each containing n samples. Among these batches, there is a collection $B_G \subseteq B$ of good batches of size $|B_G| \geq (1 - \beta)m$ and a distribution $p \in \Delta_k$ such that the following deterministic conditions hold for all subsets $S \subseteq [k]$:

1. The median of the estimates $\{\bar{\mu}_b(S) : b \in B\}$ is not too far from $p(S)$.

$$|\text{med}(\bar{\mu}(S)) - p(S)| \leq \sqrt{\ln(6)/n}.$$

2. For all sub-collections $B'_G \subseteq B_G$ of good batches of size $|B'_G| \geq (1-\beta/6)|B_G|$,

$$|\bar{p}_{B'_G}(S) - p(S)| \leq \frac{\beta}{2}\sqrt{\frac{\ln(6e/\beta)}{n}},$$
$$\Big|\frac{1}{|B'_G|}\sum_{b\in B'_G}(\bar{\mu}_b(S) - p(S))^2 - \mathrm{V}(p(S))\Big| \leq \frac{6\beta\ln(\frac{6e}{\beta})}{n}.$$

3. The corruption for good batches B_G is small, namely

$$\psi(B_G) \leq \frac{\beta m \ln(6e/\beta)}{n}.$$

Condition 1 and 3 above are self-explanatory. Condition 2 illustrates that for any sub-collection of good batches that retains all but a small fraction of good batches, empirical mean and variance estimate the actual values $p(S)$ and $\mathrm{V}(p(S))$.

Lemma 3. *When samples in B_G come from p and $|B_G| = \Omega(\frac{k}{\beta^2 \ln(1/\beta)})$, then conditions 1- 3 hold simultaneously with probability $\geq 1 - O(e^{-k})$.*

We prove the above lemma by using the observation that for $b \in B_G$, $\bar{\mu}_b(S)$ has a sub-gaussian distribution subG$(p(S), \frac{1}{4n})$, and it has variance $\mathrm{V}(p(S))$. The proof is in Appendix 2.4.

For easy reference, in the remaining paper, we will denote the upper bound in Condition 3 on the corruption of B_G as

$$\kappa_G \triangleq \frac{\beta m \ln(6e/\beta)}{n}.$$

Assuming that the above stated conditions hold, the next lemma bounds the L_1 distance between the empirical distribution $\bar{p}_{B'}$ and p for any sub-collection B' in terms of how large its corruption is compared to κ_G.

Lemma 4. *Suppose the conditions 1- 3 holds. Then for any B' such that $|B' \cap B_G| \geq (1 - \frac{\beta}{6})|B_G|$*

and let $\psi(B') = t \cdot \kappa_G$, for some $t \geq 0$, then

$$||\bar{p}_{B'} - p||_1 \leq (10 + 3\sqrt{t})\beta\sqrt{\frac{\ln(6e/\beta)}{n}}.$$

Observe that for any sub-collection B' retaining a major portion of good batches, from condition 2, the mean of $\bar{\mu}_b$ of the good batches $B' \cap B_G$ approximates p. Then showing that a small corruption score of B' w.r.t. all subsets S imply that the adversarial batches $B' \cap B_A$ have limited effect on $\bar{p}_{B'}$ proves the above lemma. A complete proof is in Appendix 2.5.

We next exhibit a Batch Deletion procedure in Algorithm 1 that lowers the corruption score of a sub-collection B' w.r.t. a given subset S by deleting a few batches from the sub-Collection. This will be a subroutine of our main algorithm. Lemma 5 characterizes its performance.

Algorithm 1. Batch Deletion

```
Input: Sub-Collection of Batches $B'$, subset $S \subseteq [k]$, med=med($\bar{\mu}(S)$), and $\beta$.
Output: A collection $DEL \subseteq B'$ of batches to delete.
$DEL = \{\}$;
while $\psi(B', S) \geq 20\kappa_G$ do
    Samples batch $b \in B'$ such that probability of picking a batch $b \in B'$ is $\frac{\psi_b(S)}{\psi(B',S)}$;
    $DEL \leftarrow DEL \cup b$;
    $B' \leftarrow \{B' \setminus b\}$;
end while
return $(DEL)$;
```

Lemma 5. *For a given B' and subset S procedure 1 returns a sub-collection $DEL \subset B'$, such that*

1. *For subset S the corruption score $\psi(B' \setminus DEL, S)$ of the new sub-collection is $< 20\kappa_G$.*

2. *Each batch $b \in B'$ that gets included in DEL is an adversarial batch with probability ≥ 0.95.*

3. *The subroutine deletes at-least $\psi(B', S) - 20\kappa_G$ batches.*

Proof. Step 4 in the algorithm ensures the first property. Next, to prove property 2, we bound the probability of deleting a good batch as

$$\sum_{b \in B' \cap B_G} \frac{\psi_b(S)}{\psi(B', S)} \leq \frac{\sum_{b \in B_G} \psi_b(S)}{\psi(B', S)} \leq \frac{\kappa_G}{20\kappa_G},$$

here the last step follows from condition 3 and while loop conditional in step 4. Property 3 follows from the observation that the total corruption score reduced is $\geq (\psi(B', S) - 20\kappa_G)$ and corruption score of one batch is bounded as $\psi_b(S) \leq 1$. ■

We will use procedure 1 to successively update B to decrease the corruption score for different subsets $S \subseteq [k]$. The next lemma show that even after successive updates the resultant sub-collection retains most of the good batches.

Lemma 6. *Let B' be the sub-collection after applying any number of successive deletion updates suggested by the Algorithm 1 on B, for any sequence of input subsets $S_1, S_2, ... \subseteq [k]$, then $|B' \cap B_G| \geq (1 - \beta/6)|B_G|$, with probability $\geq 1 - O(e^{-k})$.*

Therefore, one can make successive updates to the collection of all batches B by deleting the batches suggested by procedure 1 for all subsets in $S \subseteq [k]$ one by one. This will result in a sub-collection $B' \subseteq B$, which still has most of the good batches and corruption score $\psi(B', S)$ bounded w.r.t. each subset S. However, this will take time exponential in k as there are 2^k subsets, and therefore, we want a computationally efficient method to find a subsets S with high corruption score and use procedure 1 for only those subsets. Next, we derive a novel method to achieve this objective.

We start with the following observation. A high corruption score of sub-collection B' with respect to an affected subset S implies a higher empirical variance of $\bar{\mu}_b(S)$ for such S than the expected value of the variance of $\bar{\mu}_b(S)$. While an affected subset S the empirical variance $\overline{\mathrm{V}}_B(S)$ is higher than expected, it is not necessarily higher than the empirical variance observed for all non-affected subset. This is because $\mathrm{V}(p(S))$, the expected value of the variance of $\bar{\mu}_b(S)$, for some subsets S may be larger compared to the other. Hence, simply finding the subset S with the largest variance doesn't work.

We use the following key insight to address this. Recall that the mean of empirical probabilities $\bar{\mu}_b(S)$ for good batches $b \in B_G$ converges, or equivalently $\bar{p}_{B_G}(S) \to p(S)$. This implies that $\mathrm{V}(\bar{p}_{B_G}(S)) \to \mathrm{V}(p(S))$. Also, since the empirical variance $\overline{\mathrm{V}}_{B_G}(S)$ converges to $\mathrm{V}(p(S))$, we get $\overline{\mathrm{V}}_{B_G}(S) - \mathrm{V}(\bar{p}_{B_G}(S)) \to 0$. Therefore, without corruption by the adversarial batches the difference between two estimators of the variance would be small for all subsets $S \subseteq [k]$, and its large value, we show in Lemma 7, can reliably detect any significant adversarial corruption. This happens because empirical variance of $\bar{\mu}_b(S)$ depends on the second moment whereas the other estimator $\mathrm{V}(\bar{p}_{B'}(S))$ of variance depends on the mean of $\bar{\mu}_b(S)$, hence the corruption affects the second estimator less severely. The next Lemma shows that the difference between the two variance estimators for subset S can indicate the corruption score w.r.t. subset S

Lemma 7. *Suppose the conditions 1- 3 holds. Then for any $B' \subseteq B$ such that $|B' \cap B_G| \geq (1 - \frac{\beta}{6})|B_G|$ and let $\psi(B', S) = t \cdot \kappa_G$ for some $t \geq 0$, then following holds.*

$$\overline{V}_{B'}(S) - V(\bar{p}_{B'}(S)) \leq \Big(t + 4\sqrt{t} + 28\Big)\kappa_G,$$

$$\overline{V}_{B'}(S) - V(\bar{p}_{B'}(S)) \geq \Big(0.5t - 8\sqrt{t} - 25\Big)\kappa_G.$$

The next Lemma shows that a subset for which $\overline{\mathrm{V}}_{B'}(S) - \mathrm{V}(\bar{p}_{B'}(S))$ is large, can be found using a polynomial-time algorithm. In subsection 2.2.2 we derive the algorithm. We refer to this algorithm as $Detection - Algorithm$. The next lemma characterizes the performance of this algorithm. In subsection 2.2.2, we show that the algorithm achieves the performance guarantees of the next Lemma.

Lemma 8. *$Detection - Algorithm$ has run time polynomial in number of batches in its input* *sub-collection B' and alphabet size k, and returns $S^*_{B'}$ such that*

$$|\overline{V}_{B'}(S^*_{B'}) - V(\bar{p}_{B'}(S^*_{B'}))|$$
$$\geq 0.56 \max_{S\subseteq[k]} |\overline{V}_{B'}(S) - V(\bar{p}_{B'}(S))|.$$

This leads us to the Robust distribution Learning Algorithm 2. Theorem 9 characterizes its performance.

Algorithm 2. Robust Distribution Estimator

```
Input: All batches $b \in B$, batch size $n$, alphabet size $k$, and $\beta$.
Output: Estimate $p^*$ of the distribution $p$.
$i \leftarrow 1$ and $B'_i \leftarrow B$.
while True do
    $S^*_{B'_i} = Detection - Algorithm(B'_i)$
    if $|\Delta_{B'_i}(S^*_{B'_i})| \leq 75\kappa_G$ then
        Break;
    end if
    med $\leftarrow$ med($\bar{\mu}(S^*_{B'_i})$).
    $DEL \leftarrow$Batch-Deletion($B'_i, S^*_{B'_i}$, med).
end while
return $(p^* \leftarrow \bar{p}_{B'_i})$.
```

Theorem 9. *Suppose the conditions 1- 3 holds. Then Algorithm 2 runs in polynomial time and with probability $\geq 1 - O(e^{-k})$ returns a sub-collection $B'_f \subseteq B$ such that $|B'_f \cap B_G| \geq (1 - \frac{\beta}{6})|B_G|$ and for $p^* = \bar{p}_{B'_f}$,*

$$||p^* - p||_1 \leq 100\beta\sqrt{\frac{\ln(6e/\beta)}{n}}.$$

Outline of the Proof of Theorem 9: In each round of the algorithm, Subroutine $Detection - Algorithm$ finds subsets for which the difference between the two variance estimates is large. Lemma 7 implies that the corruption w.r.t. this subset is large. The deletion subroutine updates the sub-collection of batches by removing some batches from it and reduces the corruption w.r.t. the detected subset S.

The algorithm terminates when for some sub-collection B'_f subroutine $Detection-Algorithm$ returns a subset S small difference between the two variance estimators. Then Lemma 8 implies that the difference is small for all subsets. Lemma 7 further implies that if the difference between the two variance estimators is small then the corruption is bounded w.r.t. all subsets for sub-collection B'_f. Finally, Lemma 4 bounds the L_1 distance between $\bar{p}_{B'_f}$ and p. □

Proof of Theorem 1: Combining Lemma 3 and Theorem 9 yields Theorem 1.

2.2.1 Extension to $\eta > 0$

Recall that when $\eta > 0$, for each good batch $b \in B_G$, the distribution p_b of samples in batch b is close to the common target distribution p, such that $||p_b - p|| \le \eta$, instead of necessarily being the same. For simplicity, we have given the algorithm and the proof for only $\eta = 0$. The algorithm and the proof naturally extend to this more general case; here we get an extra additive dependence on η for the bounds in the lemmas and the theorems, and for the parameters of the algorithm. And with this slight modification in the parameters algorithm estimates p to a distance $O(\eta + \beta\sqrt{\ln(1/\beta)/n})$, and has the same sample and time complexity.

2.2.2 Efficient Detection Algorithm

In this subsection, we derive the procedure $Detection - Algorithm$, that runs in the polynomial time and achieves the performance in Lemma 8.

Given a collection B' of batches, we construct two covariance matrices $C^{EV}_{B'}$ and $C^{EM}_{B'}$ of size $k \times k$.

For an alphabet size k, we can treat the empirical probabilities estimates $\bar{\mu}_b$ and $\bar{p}_{B'}$ as a k-dimensional vector such that j^{th} entry denote the empirical probability of the j^{th} symbol. Recall that $\bar{p}_{B'}$ is the mean of $\bar{\mu}_b$, $b \in B'$.

The first covariance matrix, $C^{EV}_{B'}$, is the covariance matrix of $\bar{\mu}_b$ for $b \in B'$, with entries for $j, l \in [k]$,

$$C^{EV}_{B'}(j,l) = \frac{1}{|B'|}\sum_{b\in B'}(\bar{\mu}_b(j) - \bar{p}_{B'}(j))(\bar{\mu}_b(l) - \bar{p}_{B'}(l)).$$

The second covariance matrix $C_{B'}^{EM}$, is an expected covariance matrix of $\bar{\mu}_b$ if samples in the batches b were drawn from the distribution $\bar{p}_{B'}$. Hence, its entries are

$$C_{B'}^{EM}(j,l) = -\frac{\bar{p}_{B'}(j)\bar{p}_{B'}(l)}{n} \text{ for } j,l \in [k],\ j \neq l,$$

and

$$C_{B'}^{EM}(j,j) = \frac{\bar{p}_{B'}(j)(1-\bar{p}_{B'}(j))}{n}.$$

Let $D_{B'}$ be the difference of the two matrices:

$$D_{B'} = C_{B'}^{EV} - C_{B'}^{EM}.$$

For a vector $x \in \{0,1\}^k$, let

$$S(x) \triangleq \{j \in [k] : x(j) = 1\},$$

be the subset of $[k]$ corresponding to the vector x.

Observations

1. The sum of elements in any row and or column for both the covariance matrices, and hence also for the difference matrix, is zero, hence

$$C_{B'}^{EV}\mathbf{1} = C_{B'}^{EM}\mathbf{1} = D_{B'}\mathbf{1} = \mathbf{0}.$$

Proof: We show for $C_{B'}^{EV}$, the proof for $C_{B'}^{EM}$ is similar. For any $j \in [k]$,

$$\begin{aligned}\sum_{l \in [k]} C_{B'}^{EV}(j,l) &= \frac{1}{|B'|} \sum_{l \in [k]} \sum_{b \in B'} (\bar{\mu}_b(j) - \bar{p}_{B'}(j))(\bar{\mu}_b(l) - \bar{p}_{B'}(l)) \\ &= \sum_{b \in B'} (\bar{\mu}_b(j) - \bar{p}_{B'}(j)) \sum_{l \in [k]} (\bar{\mu}_b(l) - \bar{p}_{B'}(l)) \\ &= \sum_{b \in B'} (\bar{\mu}_b(j) - \bar{p}_{B'}(j))(1-1) = 0.\end{aligned}$$

2. It is easy to verify that for any vector $x \in \{0,1\}^k$,

$$\begin{aligned}\langle C_{B'}^{EV}, xx^\intercal \rangle &= \frac{1}{|B'|} \sum_{b \in B'} (\bar{\mu}_b(S(x)) - \bar{p}_{B'}(S(x)))^2 \\ &= \overline{\mathrm{V}}_{B'}(S(x)),\end{aligned}$$

the empirical variance of $\bar{\mu}_b(S(x))$ for $b \in B'$. Similarly,

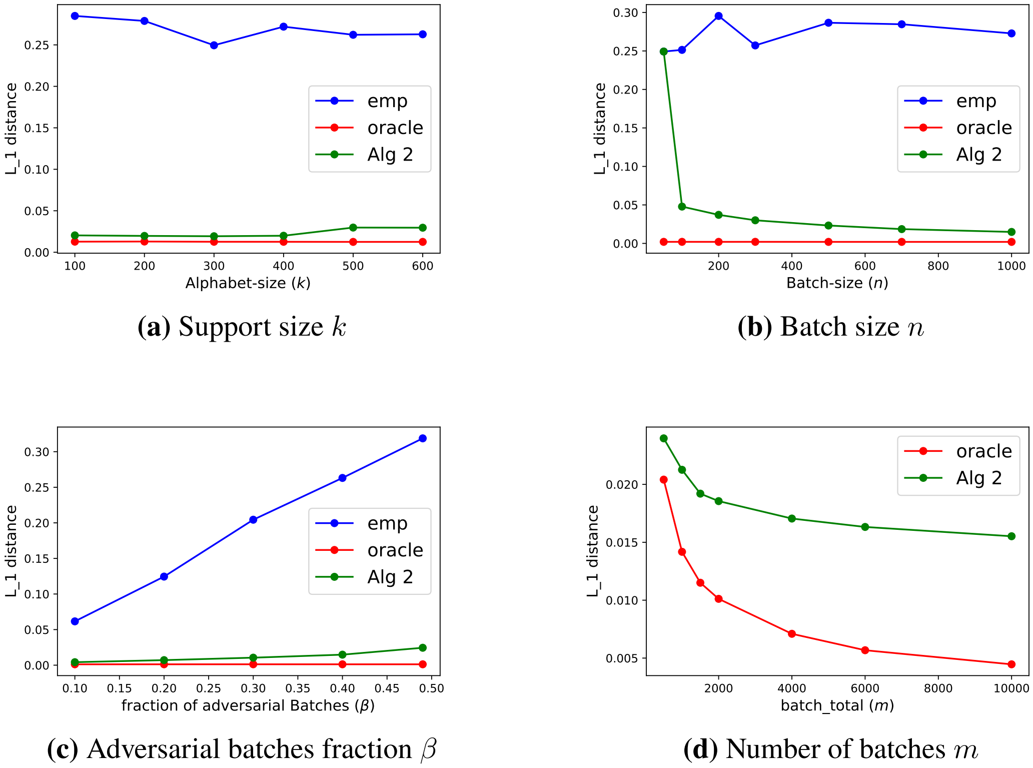
$$\begin{aligned}\langle C_{B'}^{EM}, xx^\intercal \rangle &= \frac{\bar{p}_{B'}(S(x))(1 - \bar{p}_{B'}(S(x)))}{n} \\ &= \mathrm{V}(\bar{p}_{B'}(S(x))).\end{aligned}$$

Therefore,

$$\begin{aligned}\langle D_{B'}, xx^\intercal \rangle &= \langle C_{B'}^{EV} - C_{B'}^{EM}, xx^\intercal \rangle \\ &= \overline{\mathrm{V}}_{B'}(S(x)) - \mathrm{V}(\bar{p}_{B'}(S(x))).\end{aligned}$$

3. Note that $y \to \frac{1}{2}(y + \mathbf{1})$ is a 1-1 mapping from $\{-1,1\}^k \to \{0,1\}^k$, and that

$$\begin{aligned}\langle C_{B'}^{EV}, \frac{1}{2}(y+\mathbf{1})\frac{1}{2}(y+\mathbf{1})^\intercal \rangle &= \langle C_{B'}^{EV}, \frac{1}{4}(yy^\intercal + \mathbf{1}y^\intercal + y\mathbf{1}^\intercal + \mathbf{1}\mathbf{1}^\intercal) \rangle \\ &= \frac{1}{4}\langle C_{B'}^{EV}, yy^\intercal \rangle.\end{aligned}$$



(a) Support size k

(b) Batch size n

(c) Adversarial batches fraction β

(d) Number of batches m

Figure 2.1. L_1 estimation error with different Parameters

Let

$$y = \arg \max_{y \in \{-1,1\}^k} |\langle D_{B'}, yy^\intercal \rangle|.$$

Then from y one can recover the corresponding subset $S(x)$, with $x = \frac{1}{2}(y + \mathbf{1})$, maximizing

$$|\overline{\mathrm{V}}_{B'}(S(x)) - \mathrm{V}(\bar{p}_{B'}(S(x)))|.$$

In [5], Alon et al. derives a polynomial-time approximation algorithm for the above optimization problem. The algorithm first uses a semi-definite relaxation of the problem and then uses randomized integer rounding techniques based on Grothendieck's Inequality. Their algorithm recovers $y_{B'}$ such that

$$|\langle D_{B'}, y_{B'} y_{B'}^\intercal \rangle| \ge 0.56 \max_{y \in \{-1,1\}^k} |\langle D_{B'}, yy^\intercal \rangle|.$$

Let $x_{B'} = \frac{1}{2}(y + \mathbf{1})$. Then from observation 3 it follows that

$$|\langle D_{B'}, x_{B'} x_{B'}^\intercal \rangle| \ge 0.56 \max_{x \in \{0,1\}^k} |\langle D_{B'}, xx^\intercal \rangle|.$$

Therefore for $S^*_{B'} = S(x_{B'})$ we get

$$|\overline{\mathrm{V}}_{B'}(S^*_{B'}) - \mathrm{V}(\bar{p}_{B'}(S^*_{B'}))| \ge 0.56 \max_{S \subseteq [k]} |\overline{\mathrm{V}}_{B'}(S) - \mathrm{V}(\bar{p}_{B'}(S))|.$$

2.3 Experiments

We evaluate the algorithm's performance on synthetic data.

We compare the estimator's performance with two others: 1) an oracle that knows the identity of the adversarial batches. The oracle ignores the adversarial batches and computes the empirical estimators based on remaining batches and is not affected by the presence of adversarial batches. The estimation error achieved by the oracle is the best one could get, even without the adversarial corruptions. 2) a naive-empirical estimator that computes the empirical distribution of all samples across all batches.

Two non-trivial estimators have been derived for this problem. Both have prohibitively large sample and/or computational complexity. The estimator in [125] has run time exponential in k, making it impractical. The time and sample complexities of the estimator in [30] are either super-polynomial or a high-degree polynomial, depending on the range of the parameters (k,n,$1/\beta$), rendering their simulation prohibitively high as well.

We tried different adversarial distributions and found that the major determining factor of the effectiveness of the adversarial batches is the distance between the adversarial distribution and the target distribution. If the adversarial distribution is too far, then adversarial batches are easier to detect. For this scenario our algorithm is even more effective than the performance limits shown in Theorem 1 and the performance between our algorithm and the oracle is almost indistinguishable. When the adversarial distribution is very close to the target distribution p, the adversarial batches don't affect the estimation error by much. The estimator has the worst performance when the adversary chooses the distribution of its batches at an optimal distance from target distribution. This optimal distance differs with the value of the algorithm's parameters. Hence for each choice of algorithm parameters, we tried adversarial distributions at varying distances and reported the worst performance of our estimator.

All experiments were performed on a laptop with a configuration of 2.3 GHz Intel Core i7 CPU and 16 GB of RAM. We choose the parameters for the algorithm by using a small simulation. We provide all codes and implementation details in the supplementary material.

We show four plots here. In each plot we vary one parameter and plot the L_1 loss incurred by all three estimators. For each experiment, we ran ten trials and reported the average L_1 distance achieved by each estimator.

For the first plot we fix batch-size $n = 1000$ and $\beta = 0.4$ and vary alphabet size k. We generate $m = k/(0.4)^2$ batches for each k. Our algorithm's performance show no significant change as the size of alphabet increases and its performance nearly matches the performance of the Oracle and outperforms the naive estimator by order of magnitudes.

In the the second plot we fix $\beta = 0.4$ and $k = 200$ and vary batch size n. We choose $m = 40 \times \frac{k}{\beta^2} \times \frac{1000}{n}$, this keeps the total number of samples $n \times m$, constant for different n. We see that the L_1 loss incurred by our estimator is much smaller than the naive empirical estimator and it diminishes as the batch size increases and comes very close to the performance of the oracle. Note that this roughly matches the decay $O(1/\sqrt{n})$ of L_1 error characterized in both the lower and the upper bounds.

For the next plot we fix batch size $n = 1000$ and $k = 200$. The number of good batches $(1 - \beta)m = 400k$ is kept same. We vary the adversarial noise level and plot the performance of all estimators. We tested our estimator for fraction of adversarial batches as high as 0.49 and still our estimator recovered p to a good accuracy and in fact at the lower noise level it is essentially similar to the oracle and it increases (near) linearly with the noise level β as in Theorem 1,

In the last plot we fixed all other parameters $n = 1000$, $k = 200$, and $\beta = 0.4$ and varied the number of batches. We see that the performance of oracle keep improving as number of bathes increases. But for our algorithm it decreases initially but later it saturates as predicted by adversarial batch lower bound.

Appendix

2.4 Proof of Lemma 3

In this section, we show that conditions 1-3 holds with high probability and prove Lemma 3. To prove the lemma we first prove three auxiliary lemmas; each of these three Lemma will lead to one of the three conditions in Lemma 3. These three lemmas characterizes the statistical properties of the collection of good batches B_G. We state and prove these lemmas in the next subsection.

2.4.1 Statistical Properties of the Good Batches

Recall that, for a good batch $b \in B_G$ and subset $S \subseteq [k]$, $\mathbf{1}_S(X_i^b)$, for $i \in [n]$, are i.i.d. indicator random variables and $\bar{\mu}_b(S)$ is the mean of these n indicator variables. Since the indicator random variables are sub-gaussian, namely $\mathbf{1}_S(X_i^b) \sim \text{subG}(p(S), 1/4)$, the mean $\bar{\mu}_b(S)$ satisfies $\bar{\mu}_b(S) \sim \text{subG}(p(S), 1/4n)$. subG(.) is used to denote a sub-gaussian distribution. This observation plays the key role in the proof of all three auxiliary lemmas in this section.

The first lemma among these three lemmas show that for any fixed subset $S \subseteq [k]$, $\bar{\mu}_b(S)$ for most of the good batches is close to $p(S)$. This lemma is used to show Condition 1.

Lemma 10. *For any* $\epsilon \in (0, 1/4]$ *and* $|B_G| \geq 12k/\epsilon$, $\forall\, S \subseteq [k]$, *with probability* $\geq 1 - e^{-k}$,

$$\left|\left\{b \in B_G : |\bar{\mu}_b(S) - p(S)| \geq \sqrt{\frac{\ln(1/\epsilon)}{n}}\right\}\right| \leq \epsilon |B_G|.$$

Proof. From Hoeffding's inequality, for $b \in B_G$ and $S \subseteq [k]$,

$$\Pr\left[|\bar{\mu}_b(S) - p(S)| \geq \sqrt{\frac{\ln(1/\epsilon)}{n}}\right] \leq 2e^{-2\ln(1/\epsilon)} \leq 2\epsilon^2 \leq \epsilon/2.$$

Let $\mathbf{1}_b(S)$ be the indicator random variable that takes the value 1 iff $|\bar{\mu}_b(S) - p(S)| \geq \sqrt{\ln(1/\epsilon)/n}$. Therefore, for $b \in B_G$, $E[\mathbf{1}_b(S)] \leq \epsilon/2$. Using the Chernoff bound,

$$\Pr[\sum_{b \in B_G} \mathbf{1}_b(S) \geq \epsilon |B_G|] \leq e^{-\frac{1}{3}\cdot\frac{\epsilon}{2}|B_G|} \leq e^{-2k}.$$

Taking the union bound over all 2^k subsets S completes the proof. ■

The next lemma show that even upon removal of any small fraction of good batches from B_G, the empirical mean and the variance of the remaining sub-collection of batches approximate the distribution mean and the variance well enough.

Lemma 11. *For any* $\epsilon \in (0, 1/4]$, *and* $|B_G| \geq \frac{k}{\epsilon^2 \ln(e/\epsilon)}$. *Then* $\forall\, S \subseteq [k]$ *and* $\forall\, B_G' \subseteq B_G$ *of size*

$|B'_G| \geq (1-\epsilon)|B_G|$, *with probability* $\geq 1 - 6e^{-k}$,

$$\Big|\bar{p}_{B'_G}(S) - p(S)| \leq 3\epsilon\sqrt{\frac{\ln(e/\epsilon)}{n}} \tag{2.2}$$

and

$$\Big|\frac{1}{|B'_G|}\sum_{b\in B'_G}(\bar{\mu}_b(S) - p(S))^2 - V(p(S))\Big| \leq 32\frac{\epsilon\ln(e/\epsilon)}{n}. \tag{2.3}$$

Proof. From Hoeffding's inequality,

$$\begin{aligned}
&\Pr\Big[|B_G||\bar{p}_{B_G}(S) - p(S)| \geq |B_G|\epsilon\sqrt{\frac{\ln(e/\epsilon)}{n}}\Big] \\
&= \Pr\Big[\Big|\sum_{b\in B_G}(\bar{\mu}_b(S) - p(S))\Big| \geq |B_G|\epsilon\sqrt{\frac{\ln(e/\epsilon)}{n}}\Big] \\
&\leq 2e^{-\frac{|B_G|\epsilon^2}{2/(4n)}\cdot\frac{\ln(e/\epsilon)}{n}} = 2e^{-2|B_G|\epsilon^2\ln(e/\epsilon)} \leq 2e^{-2k}.
\end{aligned} \tag{2.4}$$

Similarly, for a fix sub-collection $U_G \subseteq B_G$ of size $1 \leq |U_G| \leq \epsilon|B_G|$,

$$\begin{aligned}
&\Pr\Big[|U_G|\cdot|\bar{p}_{U_G}(S) - p(S)| \geq \epsilon|B_G|\sqrt{\frac{\ln(e/\epsilon)}{n}}\Big] \\
&= \Pr\Big[\Big|\sum_{b\in U_G}(\bar{\mu}_b(S) - p(S))\Big| \geq \epsilon|B_G|\sqrt{\frac{\ln(e/\epsilon)}{n}}\Big] \\
&\leq 2e^{-2\ln(e/\epsilon)\frac{(\epsilon|B_G|)^2}{|U_G|}} \leq 2e^{-2\epsilon|B_G|\ln(e/\epsilon)},
\end{aligned}$$

where the last inequality used $|U_G| \leq \epsilon|B_G|$. Next, the number of sub-collections (non-empty) of B_G with size $\leq \epsilon|B_G|$ is bounded by

$$\begin{aligned}
\sum_{j=1}^{\lfloor\epsilon|B_G|\rfloor}\binom{|B_G|}{j} \leq \epsilon|B_G|\binom{|B_G|}{\lfloor\epsilon|B_G|\rfloor} \leq \epsilon|B_G|\Big(\frac{e|B_G|}{\epsilon|B_G|}\Big)^{\epsilon|B_G|} \\
\leq e^{\epsilon|B_G|\ln(e/\epsilon)+\ln(\epsilon|B_G|)} < e^{\frac{3}{2}\epsilon|B_G|\ln(e/\epsilon)},
\end{aligned} \tag{2.5}$$

where last of the above inequality used $\ln(\epsilon|B_G|) < \epsilon|B_G|/2$ and $\ln(e/\epsilon) \geq 1$. Then, using the union bound, $\forall\, U_G \subseteq B_G$ such that $|U_G| \leq \epsilon|B_G|$, we get

$$\Pr\Big[|U_G| \cdot |\bar{p}_{U_G}(S) - p(S)| \geq \epsilon|B_G|\sqrt{\frac{\ln(e/\epsilon)}{n}}\Big] \leq 2e^{-\frac{1}{2}\epsilon|B_G|\ln(e/\epsilon)} < 2e^{-\frac{k}{2\epsilon}} < 2e^{-2k}. \quad (2.6)$$

For any sub-collection $B'_G \subseteq B_G$ with $|B'_G| \geq (1-\epsilon)|B_G|$,

$$\begin{aligned}
|\sum_{b \in B'_G}(\bar{\mu}_b(S) - p(S))| &= |\sum_{b \in B_G}(\bar{\mu}_b(S) - p(S)) - \sum_{b \in B_G/B'_G}(\bar{\mu}_b(S) - p(S))| \\
&\leq \Big|\sum_{b \in B_G}(\bar{\mu}_b(S) - p(S))\Big| + \Big|\sum_{b \in B_G/B'_G}(\bar{\mu}_b(S) - p(S))\Big| \\
&\leq |B_G| \times |\bar{p}_{B_G}(S) - p(S)| + \max_{U_G : |U_G| \leq \epsilon|B_G|} |U_G| \times |\bar{p}_{U_G}(S) - p(S)| \\
&\leq 2\epsilon|B_G|\sqrt{\frac{\ln(e/\epsilon)}{n}},
\end{aligned}$$

with probability $\geq 1 - 2e^{-2k} - 2e^{-2k} \geq 1 - 4e^{-2k}$. Then

$$\begin{aligned}
|\bar{p}_{B'_G}(S) - p(S)| = \frac{1}{|B'_G|}\Big|\sum_{b \in B'_G}(\bar{\mu}_b(S) - p(S))\Big| &\leq 2\frac{\epsilon|B_G|}{|B'_G|}\sqrt{\frac{\ln(e/\epsilon)}{n}} \\
&\leq \frac{2\epsilon}{(1-\epsilon)}\sqrt{\frac{\ln(e/\epsilon)}{n}} < 3\epsilon\sqrt{\frac{\ln(e/\epsilon)}{n}},
\end{aligned}$$

with probability $\geq 1 - 4e^{-2k}$. The last step used $\epsilon \leq 1/4$. Since there are 2^k different choices for $S \subseteq [k]$, from the union bound we get,

$$\Pr\Big[\bigcup_{S \subseteq [k]}\Big\{|\bar{p}_{B'_G}(S) - p(S)| > 4\epsilon\sqrt{\frac{\ln(e/\epsilon)}{n}}\Big\}\Big] \leq 4e^{-2k} \times 2^k = 4e^{-k}.$$

This completes the proof of (2.2).

Let $Y_b = (\bar{\mu}_b(S) - p(S))^2 - \mathrm{V}(p(S))$. For $b \in B_G$, $\bar{\mu}_b(S) - p(S) \sim \mathrm{subG}(1/4n)$, therefore

$$(\bar{\mu}_b(S) - p(S))^2 - E(\bar{\mu}_b(S) - p(S))^2 = Y_b \sim \text{subE}(\frac{16}{4n}) = \text{subE}(\frac{4}{n}).$$

Here subE is sub exponential distribution [121]. Then Bernstein's inequality gives:

$$\Pr[\Big|\sum_{b\in B_G} Y_b\Big| \geq 8|B_G|\frac{\epsilon}{n}\ln(e/\epsilon)] \leq 2e^{-\frac{|B_G|}{2}\left(\frac{8\epsilon\ln(e/\epsilon)/n}{4/n}\right)^2} = 2e^{-2|B_G|\epsilon^2\ln^2(e/\epsilon)} \leq 2e^{-2k}.$$

Next, for a fix sub-collection $U_G \subseteq B_G$ of size $1 \leq |U_G| \leq \epsilon|B_G|$,

$$\Pr\Big[\Big|\sum_{b\in U_G} Y_b\Big| \geq 16\epsilon|B_G|\frac{\ln(e/\epsilon)}{n}\Big] \leq 2e^{-\frac{16\epsilon|B_G|\frac{\ln(e/\epsilon)}{n}}{2\times 4/n}}$$
$$\leq 2e^{-2\epsilon|B_G|\ln(e/\epsilon)}.$$

Then following the same steps as in the proof of (2.2) one can complete the proof of (2.3). ■

To state the next lemma, we make use of the following definition. For a subset $S \subseteq [k]$, let

$$B_G^d(S,\epsilon) \triangleq \big\{b \in B_G : |\bar{\mu}_b(S) - p(S)| \geq 2\sqrt{\frac{\ln(6e/\epsilon)}{n}})\big\}$$

be the sub-collection of batches for which empirical probabilities $\bar{\mu}_b(S)$ are far from $p(S)$ for a given set S.

The last lemma of the section upper bounds the total squared deviation of empirical probabilities $\bar{\mu}_b(S)$ from $p(S)$ for batches in sub-collection $B_G^d(S,\epsilon)$. It helps in upper bounding the corruption for good batches and show that Condition 3 holds with high probability.

Lemma 12. *For any $0 < \epsilon < 1/2$, and $|B_G| \geq \frac{120k}{\epsilon\ln(e/\epsilon)}$. Then $\forall\, S \subseteq [k]$, with probability $\geq 1 - 2e^{-k}$,*

$$|B_G^d(S,\epsilon)| \leq \frac{\epsilon}{40}|B_G|, \tag{2.7}$$

and

$$\sum_{b\in B_G^d(S,\epsilon)} (\bar{\mu}_b(S) - p(S))^2 < \frac{\epsilon}{2}|B_G|\frac{\ln(e/\epsilon)}{n}. \tag{2.8}$$

Proof. The proof of the first part is the same as (with different constants) Lemma 10 and we skip it to avoid repetition.

To prove the second part we bound the total squared deviation of any subset of size $\leq \frac{\epsilon}{40}|B_G|$.

Let $Y_b = (\bar{\mu}_b(S) - p(S))^2 - \mathrm{V}(p(S))$. Similar to the previous lemma, for a fix sub-collection $U_G \subseteq B_G$ of size $1 \leq |U_G| \leq \frac{\epsilon}{40}|B_G|$, Bernstein's inequality gives:

$$\begin{aligned}\Pr\Big[\Big|\sum_{b\in U_G} Y_b\Big| \geq 8\frac{\epsilon}{20}|B_G|\frac{\ln(e/\epsilon)}{n}\Big] &\leq 2e^{-\frac{8\epsilon|B_G|\frac{\ln(e/\epsilon)}{n}}{20\times 2\times 4/n}} \\ &\leq 2e^{-\frac{\epsilon}{20}|B_G|\ln(e/\epsilon)}.\end{aligned}$$

From (2.5), there are $e^{\frac{3}{80}\epsilon|B_G|\ln(e/\epsilon)}$ many sub-collections of size $\leq \frac{\epsilon}{40}|B_G|$. Then taking the union bound for all sub-collections of this size and all subsets $S \subseteq [k]$ we get,

$$\Big|\sum_{b\in U_G}\Big((\bar{\mu}_b(S) - p(S))^2 - \mathrm{V}(p(S))\Big)\Big| \leq \frac{2\epsilon}{5}|B_G|\frac{\ln(e/\epsilon)}{n},$$

for all U_G of size $\leq \frac{\epsilon}{40}|B_G|$. Then using the fact that V(.) is upper bounded by $\frac{1}{4n}$, and therefore $|U_G|\mathrm{V}(p(S)) \leq \frac{\epsilon}{4\times 40}|B_G|$, completes the proof. ■

2.4.2 Completing the proof of Lemma 3

We first show condition 1 holds with high probability.

It is easy to verify that $|p(S) - \mathrm{med}(\bar{\mu}(S))| \geq \sqrt{\ln 6/n}$, only if the sub-collection

$T = \{b : |p(S) - \bar{\mu}_b(S)| \geq \sqrt{\ln 6/n}\}$ has at-least $0.5m$ batches. But

$$|T| = |T \cap B_G| + |T \cap B_A| \overset{\text{(a)}}{<} |B_G|/6 + |B_A| = \frac{m}{6} + \frac{5}{6}|B_A| \overset{\text{(b)}}{\leq} \frac{m}{6} + \frac{2m}{6} = 0.5m,$$

where inequality (a) follows from Lemma 10 by choosing $\epsilon = 1/6$ and (b) follows since $|B_A| \leq \beta m \leq 0.4m$.

Using $\epsilon = \beta/6$ in Lemma 11 gives Condition 2.

Finally, we show the last condition. To show it we use $\epsilon = \beta$ in Lemma 12. From Condition 1, note that for $b \in B_G \setminus B_G^d(S, \beta)$

$$\begin{aligned}|\bar{\mu}_b(S) - \text{med}(\bar{\mu}(S))| \leq |\bar{\mu}_b(S) - p(S)| + |p(S) - \text{med}(\bar{\mu}(S))| &\leq 2\sqrt{\frac{\ln(6e/\beta)}{n}} + \sqrt{\frac{\ln 6}{n}} \\ &\leq 3\sqrt{\frac{\ln(6e/\beta)}{n}},\end{aligned}$$

Then, for $b \in B_G \setminus B_G^d(S, \beta)$, from the definition of corruption score it follows that

$\psi_b(S) = 0$. Next set of inequalities complete the proof of condition 3.

$$\begin{aligned}
\psi(B_G) &= \sum_{b\in B_G} \psi_b(S) = \sum_{b\in B_G\setminus B_G^d(S,\beta)} \psi_b(S) + \sum_{b\in B_G^d(S,\beta)} \psi_b(S)\\
&= \sum_{b\in B_G^d(S,\beta)} \psi_b(S)\\
&\overset{\text{(a)}}{\le} \sum_{b\in B_G^d(S,\beta)} (\bar{\mu}_b(S) - \text{med}(\bar{\mu}(S)))^2\\
&= \sum_{b\in B_G^d(S,\beta)} (\bar{\mu}_b(S) - p(S) + p(S) - \text{med}(\bar{\mu}(S)))^2\\
&\overset{\text{(b)}}{\le} \sum_{b\in B_G^d(S,\beta)} (\bar{\mu}_b(S) - p(S))^2 + \sum_{b\in B_G^d(S,\beta)} (\text{med}(\bar{\mu}(S)) - p(S))^2\\
&+ 2\sqrt{\left(\sum_{b\in B_G^d(S,\beta)} (\bar{\mu}_b(S) - p(S))^2\right)\left(\sum_{b\in B_G^d(S,\beta)} (\text{med}(\bar{\mu}(S)) - p(S))^2\right)}\\
&\overset{\text{(c)}}{\le} \frac{\beta}{2}|B_G|\frac{\ln(e/\beta)}{n} + \frac{\beta}{40}|B_G|\frac{\ln 6}{n} + \sqrt{\frac{\beta}{2}|B_G|\frac{\ln(e/\beta)}{n} \times \frac{\beta}{40}|B_G|\frac{\ln 6}{n}} < \beta|B_G|\frac{\ln(e/\beta)}{n},
\end{aligned}$$

here (a) follows from the definition of the corruption score, (b) uses Cauchy-Schwarz inequality and (c) follows from Lemma 12 and Condition 1.

2.5 Proof of the other Lemmas

We first prove an auxiliary Lemma that will be useful in other proofs. For a given sub-collection B' and subset S, the next lemma bounds the total squared distance of $\bar{\mu}_b(S)$ from $p(S)$ over the adversarial batches $b \in B' \cap B_A$ in terms of corruption score $\psi(B', S)$.

Lemma 13. *Suppose the conditions 1 and 3 holds. For subset S, let $\psi(B', S) = t \cdot \kappa_G$, for some $t \ge 0$, then*

$$(t - 3 - 2\sqrt{t})\kappa_G \le \sum_{b\in B'\cap B_A} (\bar{\mu}_b(S) - p(S))^2 \le (t + 17 + 2\sqrt{t})\kappa_G.$$

Proof. For the purpose of this proof, let $B'_G = B' \cap B_G$ and $B'_A = B' \cap B_A$. Then

$$\sum_{b\in B'_A} (\bar{\mu}_b(S) - p(S))^2 = \sum_{b\in B'_A:\psi_b(S)>0} (\bar{\mu}_b(S) - p(S))^2 + \sum_{b\in B'_A:\psi_b(S)=0} (\bar{\mu}_b(S) - p(S))^2 \quad (2.9)$$

From the definition of corruption score, for batch $b \in B'$, with zero corruption score $\psi_b(S)$, we have $|\bar{\mu}_b(S) - \text{med}(\bar{\mu}(S))| \leq 3\sqrt{\frac{\ln(6e/\beta)}{n}}$. Then using Condition 1 and the triangle inequality, for such batches with zero corruption score, we get

$$|\bar{\mu}_b(S) - p(S)| \leq \sqrt{\ln(6)/n} + 3\sqrt{\frac{\ln(6e/\beta)}{n}} < 4\sqrt{\frac{\ln(6e/\beta)}{n}}. \quad (2.10)$$

Next,

$$\begin{aligned}
&\sum_{b\in B'_A:\psi_b(S)>0} (\bar{\mu}_b(S) - p(S))^2 \\
&= \sum_{b\in B'_A:\psi_b(S)>0} (\bar{\mu}_b(S) - \text{med}(\bar{\mu}(S)) + \text{med}(\bar{\mu}(S)) - p(S))^2 \\
&\overset{\text{(a)}}{\leq} \sum_{b\in B'_A:\psi_b(S)>0} (\bar{\mu}_b(S) - \text{med}(\bar{\mu}(S)))^2 + \sum_{b\in B'_A:\psi_b(S)>0} (\text{med}(\bar{\mu}(S)) - p(S))^2 \\
&+ 2\sqrt{\left(\sum_{b\in B'_A:\psi_b(S)>0} (\bar{\mu}_b(S) - \text{med}(\bar{\mu}(S)))^2\right)\left(\sum_{b\in B'_A:\psi_b(S)>0} (\text{med}(\bar{\mu}(S)) - p(S))^2\right)} \\
&\overset{\text{(b)}}{\leq} \sum_{b\in B'_A} \psi_b(S) + \sum_{b\in B'_A} \frac{\ln 6}{n} + 2\sqrt{\left(\sum_{b\in B'_A} \psi_b(S)\right)\left(\sum_{b\in B'_A} \frac{\ln 6}{n}\right)} \\
&\overset{\text{(c)}}{\leq} \psi(B'_A, S) + \kappa_G + 2\sqrt{\psi(B'_A, S)\cdot\kappa_G}, \qquad (2.11)
\end{aligned}$$

here (a) uses Cauchy-Schwarz inequality, (b) follows from the definition of corruption score and Condition 1, and (c) uses $|B'_A| \leq \beta m$ and $(\beta m \ln 6)/n \leq \kappa_G$.

A similar calculation as the above leads to the following

$$\sum_{b \in B'_A : \psi_b(S) > 0} (\bar{\mu}_b(S) - p(S))^2 \geq \psi(B'_A, S) - 2\sqrt{\psi(B'_A, S) \cdot \kappa_G}, \tag{2.12}$$

Next, we show the upper bound in the lemma. Combining equations (2.9), (2.10) and (2.11) gives

$$\begin{aligned}\sum_{b \in B'_A} (\bar{\mu}_b(S) - p(S))^2 &\leq \psi(B'_A, S) + \kappa_G + 2\sqrt{\psi(B'_A, S) \cdot \kappa_G} + \sum_{b \in B'_A : \psi_b(S) = 0} 4\sqrt{\frac{\ln(6e/\beta)}{n}} \\ &\leq \psi(B', S) + \kappa_G + 2\sqrt{\psi(B', S) \cdot \kappa_G} + 16|B_A|\frac{\ln(6e/\beta)}{n} \\ &\leq (t + 17 + 2\sqrt{t})\kappa_G,\end{aligned}$$

here the second last inequality used $B'_A \subseteq B'$ and $B'_A \subseteq B_A$. This completes the proof of the upper bound.

To prove the lower bound, we first note that

$$\begin{aligned}&\psi(B', S) = \sum_{b \in B'} \psi_b(S) = \sum_{b \in B'_G} \psi_b(S) + \sum_{b \in B'_A} \psi_b(S) \\ &\leq \sum_{b \in B_G} \psi_b(S) + \psi(B'_A, S) \\ &\leq \psi(B_G) + \psi(B'_A, S) \leq \frac{\beta m \ln(6e/\beta)}{n} + \sum_{b \in B'_A} \psi_b(S),\end{aligned}$$

here the last inequality uses condition 3. The above equation implies that

$$\psi(B'_A, S) \geq \psi(B', S) - \beta m \frac{\ln(6e/\beta)}{n} = (t - 1)\kappa_G. \tag{2.13}$$

By combining, equations (2.9) (2.12) and (2.13), we get the lower bound

$$\sum_{b\in B'_A}(\bar{\mu}_b(S)-p(S))^2 \geq (t-1)\kappa_G - 2\sqrt{|t-1|\kappa_G\cdot\kappa_G} \qquad = (t-1-2\sqrt{|t-1|})\kappa_G$$

$$\geq (t-3-2\sqrt{t})\kappa_G.$$

■

2.5.1 Proof of Lemma 4

Proof. For the purpose of this proof, let $B'_G = B'\cap B_G$ and $B'_A = B'\cap B_A$. Note that $|B'| \geq |B'_G| \geq (1-\beta/6)B_G$.

Fix subset $S\subseteq[k]$. Next,

$$\begin{aligned}
&\bar{p}_{B'}(S)-p(S) = \frac{1}{|B'|}\sum_{b\in B'}\bar{\mu}_b(S)-p(S) = \frac{1}{|B'|}\sum_{b\in B'}(\bar{\mu}_b(S)-p(S))\\
&= \frac{1}{|B'|}\sum_{b\in B'_G}(\bar{\mu}_b(S)-p(S)) + \frac{1}{|B'|}\sum_{b\in B'_A}(\bar{\mu}_b(S)-p(S))\\
&= \frac{|B'_G|}{|B'|}(\bar{p}_{B'_G}(S)-p(S)) + \frac{1}{|B'|}\sum_{b\in B'_A}(\bar{\mu}_b(S)-p(S))
\end{aligned}$$

Therefore,

$$\begin{aligned}
|\bar{p}_{B'}(S) - p(S)| &\leq \frac{|B'_G|}{|B'|}|\bar{p}_{B'_G}(S) - p(S)| + \frac{1}{|B'|}\sum_{b\in B'_A}|\bar{\mu}_b(S) - p(S)| \\
&\overset{(a)}{\leq} \frac{\beta}{2}\sqrt{\frac{\ln(6e/\beta)}{n}} + \frac{1}{|B'|}\sum_{b\in B'_A}|\bar{\mu}_b(S) - p(S)| \\
&\overset{(b)}{\leq} \frac{\beta}{2}\sqrt{\frac{\ln(6e/\beta)}{n}} + \frac{1}{|B'|}\sqrt{|B'_A|\sum_{b\in B'_A}(\bar{\mu}_b(S) - p(S))^2} \\
&\overset{(c)}{\leq} \frac{\beta}{2}\sqrt{\frac{\ln(6e/\beta)}{n}} + \frac{1}{|B'|}\sqrt{|B'_A|\cdot(t + 17 + 2\sqrt{t})\kappa_G} \\
&\overset{(d)}{\leq} \frac{\beta}{2}\sqrt{\frac{\ln(6e/\beta)}{n}} + \frac{1}{|B'|}\sqrt{|B'_A|\cdot(t + 17 + 2\sqrt{t})\frac{\beta m\ln(6e/\beta)}{n}} \\
&\leq \frac{\beta}{2}\sqrt{\frac{\ln(6e/\beta)}{n}} + \sqrt{\frac{|B'_A|\cdot m}{|B'|^2}\cdot(t + 17 + 2\sqrt{t})\frac{\beta\ln(6e/\beta)}{n}}, \qquad (2.14)
\end{aligned}$$

here in (a) uses Condition 2 and $|B'_G| \leq |B'|$, inequality (b) follows from the Cauchy-Schwarz inequality, inequality (c) uses Lemma 13, and (d) uses the definition of κ_G. Let $|B'_A| = |B_A| - \text{D}$, for some $\text{D} \in [0, |B_A|]$. Also from Lemma note that

$$|B'_G| \geq (1 - \beta/6)|B_G| = |B_G| - |B_G|\beta/6 = |B_G| - m\beta(1-\beta)/6.$$

Therefore,

$$\begin{aligned}
\frac{|B'_A|\cdot m}{|B'|^2} = \frac{|B'_A|\cdot m}{(|B'_A| + |B'_G|)^2} &\leq \frac{(|B_A| - \text{D})m}{(|B_A| - \text{D} + |B_G| - m\beta(1-\beta)/6)^2} \\
&= \frac{(\beta m - \text{D})m}{(m - \text{D} - m\beta(1-\beta)/6)^2} \\
&\overset{(a)}{\leq} \frac{(\beta m - \text{D})m}{(m - \text{D} - 0.04m)^2} \\
&\overset{(b)}{\leq} \frac{\beta m^2}{(0.96m)^2} \leq \frac{\beta}{0.96^2},
\end{aligned}$$

here (a) follows since $\beta(1-\beta)$ takes maximum value at $\beta = 0.4$ in range $\beta \in (0, 0.4]$, and (b) follows since the expression is maximized at D $= 0$.

Then combining above equation with (2.14) gives

$$
\begin{aligned}
|\bar{p}_{B'}(S) - p(S)| &\leq \frac{\beta}{2}\sqrt{\frac{\ln(6e/\beta)}{n}} + \sqrt{(t+17+2\sqrt{t})\frac{\beta^2\ln(6e/\beta)}{0.96^2 n}} \\
&\leq \Big(1/2 + \frac{1}{0.96}\sqrt{(t+17+2\sqrt{t})}\Big)\beta\sqrt{\frac{\ln(6e/\beta)}{n}} \qquad (2.15)\\
&\overset{\text{(a)}}{\leq} \Big(5 + \sqrt{2.1t}\Big)\beta\sqrt{\frac{\ln(6e/\beta)}{n}}, \qquad (2.16)
\end{aligned}
$$

here inequality (a) uses the fact that $2t^{1/2} \leq t+1$ and $\sqrt{x^2+y^2} \leq |x| + |y|$. Finally, using the definition of L_1 distance between two distributions complete the proof of the Theorem. ■

2.5.2 Proof of Lemma 6

Proof. From the second statement in Lemma 5, each batch that gets removed is adversarial with probability ≥ 0.95. Batch deletion deletes more than $0.1\beta m$ good batches in total over all runs iff it samples $0.1\beta m$ good batches in first $0.1\beta m + |B_A|$ batches removed as otherwise all adversarial batches would have been exhausted already and Batch deletion algorithm would not remove batches any further. But the expected number of good batches sampled is $\leq 0.05(\times 0.1\beta m + |B_A|) \leq 0.005\beta m + 0.05\beta m < 0.06\beta m$.

Then using the Chernoff-bound, probability of sampling (removing) more than $0.1\beta m$ good batches in $0.1\beta m + |B_A|$ deletions is $\leq e^{-O(\beta m)} \leq e^{-O(k)}$. Hence, with high probability the algorithm deletes less than $0.1\beta m = 0.6\beta m/6 \leq |B_G|\beta/6$ batches. ■

2.5.3 Proof of Lemma 7

Proof. For the purpose of this proof, let $B'_G = B' \cap B_G$ and $B'_A = B' \cap B_A$. For batches b in a sub-collection B', the next equation relates the empirical variance of $\bar{\mu}_b(S)$ to sum of their squared deviation from $p(S)$.

$$\begin{aligned}
&|B'|\overline{\mathrm{V}}_{B'}(S) = \sum_{b\in B'}(\bar{\mu}_b(S) - \bar{p}_{B'}(S))^2 = \sum_{b\in B'}(\bar{\mu}_b(S) - p(S) - (\bar{p}_{B'}(S) - p(S)))^2 \\
&= \sum_{b\in B'}\Big((\bar{\mu}_b(S) - p(S))^2 + (\bar{p}_{B'}(S) - p(S))^2 - 2(\bar{p}_{B'}(S) - p(S))(\bar{\mu}_b(S) - p(S))\Big) \\
&= \sum_{b\in B'}(\bar{\mu}_b(S) - p(S))^2 + |B'|(\bar{p}_{B'}(S) - p(S))^2 - 2(\bar{p}_{B'}(S) - p(S))\sum_{b\in B'}(\bar{\mu}_b(S) - p(S)) \\
&= \sum_{b\in B'}(\bar{\mu}_b(S) - p(S))^2 + |B'|(\bar{p}_{B'}(S) - p(S))^2 - 2(\bar{p}_{B'}(S) - p(S))(|B'|\bar{p}_{B'}(S) - |B'|p(S)) \\
&= \sum_{b\in B'}(\bar{\mu}_b(S) - p(S))^2 - |B'|(\bar{p}_{B'}(S) - p(S))^2 \\
&= \sum_{b\in B'_A}(\bar{\mu}_b(S) - p(S))^2 + \sum_{b\in B'_G}(\bar{\mu}_b(S) - p(S))^2 - |B'|(p(S) - \bar{p}_{B'}(S))^2. \qquad (2.17)
\end{aligned}$$

The next set of inequalities lead to the upper bound in the Lemma.

$$\begin{aligned}
&|B'|(\overline{\mathrm{V}}_{B'}(S) - \mathrm{V}(\bar{p}_{B'}(S))) \\
&\overset{\text{(a)}}{=} \sum_{b\in B'_A}(\bar{\mu}_b(S) - p(S))^2 + \sum_{b\in B'_G}(\bar{\mu}_b(S) - p(S))^2 - |B'|(p(S) - \bar{p}_{B'}(S))^2 - |B'|\mathrm{V}(\bar{p}_{B'}(S)) \\
&\overset{\text{(b)}}{\le} (t + 17 + 2\sqrt{t})\kappa_G + |B'_G|\mathrm{V}(p(S)) + |B'_G|\frac{6\beta\ln(\frac{6e}{\beta})}{n} - |B'|\mathrm{V}(\bar{p}_{B'}(S)) \\
&\overset{\text{(c)}}{\le} (t + 17 + 2\sqrt{t})\kappa_G + 6\beta m\frac{\ln(6e/\beta)}{n} + |B'|\mathrm{V}(p(S)) - |B'|\mathrm{V}(\bar{p}_{B'}(S)) \\
&\overset{\text{(d)}}{\le} (t + 23 + 2\sqrt{t})\kappa_G + m\frac{|p(S) - \bar{p}_{B'}(S)|}{n},
\end{aligned}$$

here inequality (a) follows from (2.17), (b) follows from Lemma 13 and condition 2, and (c) follows since $|B'_G| \le |B'|$ and $\mathrm{V}(\cdot) \ge 0$, and inequality (d) uses (2.1) and $|B'| \le m$. Next, from equation (2.16) we have,

$$\begin{aligned}
|\bar{p}_{B'}(S) - p(S)| &\le (5 + \sqrt{2.1t})\beta\sqrt{\frac{\ln(6e/\beta)}{n}} \\
&= (5 + \sqrt{2.1t})\beta\ln(6e/\beta)\sqrt{\frac{1}{n\ln(6e/\beta)}} \\
&\le (5 + \sqrt{2.1t})\frac{n\kappa_G}{m}. \qquad (2.18)
\end{aligned}$$

Combining the above two equations gives the upper bound in the lemma.

Next showing the lower bound,

$$\begin{aligned}
&|B'|(\overline{\mathrm{V}}_{B'}(S) - \mathrm{V}(\bar{p}_{B'}(S))) \\
&\overset{(a)}{=} \sum_{b\in B'_A} (\bar{\mu}_b(S) - p(S))^2 + \sum_{b\in B'_G} (\bar{\mu}_b(S) - p(S))^2 - |B'|(p(S) - \bar{p}_{B'}(S))^2 - |B'|\mathrm{V}(\bar{p}_{B'}(S)) \\
&\overset{(b)}{\ge} (t - 3 - 2\sqrt{t})\kappa_G + |B'_G|\mathrm{V}(p(S)) - |B'_G|\frac{6\beta\ln(\frac{6e}{\beta})}{n} - |B'|(p(S) - \bar{p}_{B'}(S))^2 \\
&\quad - |B'|\mathrm{V}(\bar{p}_{B'}(S)) \\
&\ge (t - 9 - 2\sqrt{t})\kappa_G + |B'_G|\mathrm{V}(p(S)) - |B'|(p(S) - \bar{p}_{B'}(S))^2 - |B'_G|\mathrm{V}(\bar{p}_{B'}(S)) \\
&\quad - |B'_A|\mathrm{V}(\bar{p}_{B'}(S)) \\
&\ge (t - 9 - 2\sqrt{t})\kappa_G - |B'_G|(\mathrm{V}(\bar{p}_{B'}(S)) - \mathrm{V}(p(S))) - |B'|(p(S) - \bar{p}_{B'}(S))^2 - \\
&\quad |B'_A|\mathrm{V}(\bar{p}_{B'}(S)) \\
&\overset{(c)}{\ge} (t - 9 - 2\sqrt{t})\kappa_G - |B'_G|\frac{|p(S) - \bar{p}_{B'}(S)|}{n} - \frac{|B'_A|}{4n} - |B'|(p(S) - \bar{p}_{B'}(S))^2 \\
&\ge (t - 9 - 2\sqrt{t})\kappa_G - m\frac{|p(S) - \bar{p}_{B'}(S)|}{n} - \frac{\beta m}{4n} - m(p(S) - \bar{p}_{B'}(S))^2 \\
&\overset{(d)}{\ge} (t - 15 - 2\sqrt{t} - \sqrt{2.1t})\kappa_G - m(p(S) - \bar{p}_{B'}(S))^2,
\end{aligned}$$

here inequality (a) follows from (2.17), (b) follows from Lemma 13 and condition 2, (c) follows from (2.1) and $\mathrm{V}(\cdot) \le \frac{1}{4n}$, and inequality (d) follows from (2.18).

Next, we bound the last tem in the above equation to complete the proof. From equation (2.15),

$$
\begin{aligned}
(p(S)-\bar{p}_{B'}(S))^2 &\le \Big(1/2+\frac{1}{0.96}\sqrt{(t+17+2\sqrt{t})}\Big)^2\beta^2\frac{\ln(6e/\beta)}{n}\\
&\le \Big(1/4+\frac{1}{0.96^2}(t+17+2\sqrt{t})+\frac{1}{0.96}\sqrt{(t+17+2\sqrt{t})}\Big)\beta\cdot\kappa_G\\
&\overset{\text{(a)}}{\le} \Big(1/4+1.1(t+17+2\sqrt{t})+5+\sqrt{2.1t}\Big)\beta\cdot\kappa_G\\
&\le \Big(24+1.1t+2.2\sqrt{t}+\sqrt{2.1t}\Big)\beta\cdot\kappa_G\\
&\overset{\text{(b)}}{\le} 0.4\Big(24+1.1t+2.2\sqrt{t}+\sqrt{2.1t}\Big)\kappa_G,
\end{aligned}
$$

here inequality (a) uses the fact that $2t^{1/2}\le t+1$ and $\sqrt{x^2+y^2}\le|x|+|y|$ and inequality (b) uses $\beta\le 0.4$. Combining above two equations give us the lower bound in the Lemma. ■

2.6 Proof of Theorem 9

First, we restate the statement of the main theorem.

Theorem 14. *Suppose the conditions 1- 3 holds. Then Algorithm 2 runs in polynomial time and with probability* $\ge 1-O(e^{-k})$ *returns a sub-collection* $B'_f\subseteq B$ *such that* $|B'_f\cap B_G|\ge(1-\frac{\beta}{6})|B_G|$ *and for* $p^*=\bar{p}_{B'_f}$,

$$||p^*-p||_1\le 100\beta\sqrt{\frac{\ln(6e/\beta)}{n}}.$$

Proof. Lemma 6 show that for the sub-collection B'_i at each iteration i, $|B'_i\cap B_G|\ge(1-\frac{\beta}{6})|B_G|$, hence, for sub-collection B'_f returned by the algorithm $|B'_f\cap B_G|\ge(1-\frac{\beta}{6})|B_G|$, with probability $\ge 1-O(e^{-k})$. This also implies that the total number of deleted batches are $<(1+1/6)\beta m$.

To complete the proof of the above Theorem, we state the following corollary, which is a direct consequence of Lemma 7.

Corollary 15. *Suppose the conditions 1- 3 holds. Then following hold for any* $B'\subseteq B$ *such that* $|B'\cap B_G|\ge(1-\frac{\beta}{6})|B_G|$.

1. $|\overline{V}_{B'}(S)-V(\bar{p}_{B'}(S))|\ge 75\kappa_G$ *implies that* $\psi(B',S)\ge 25\kappa_G$.

2. *$|\overline{V}_{B'}(S) - V(\bar{p}_{B'}(S))| \le 150\kappa_G$ implies that $\psi(B', S) \le 900\kappa_G$.*

In each iteration of Algorithm 2, except the last, $Detection - Algorithm$ returns a subset for which the difference between two variance estimate is $\ge 75\kappa_G$. The first statement in the above corollary implies that corruption is high for this subset. Batch Deletion removes batches from the sub-collection to reduce the corruption for such subset. From Statement 3 of Lemma 5, in each iteration Batch Deletion removes $\ge 25\kappa_G - 20\kappa_G$ batches. Since the total batches removed are $< 7/6\beta m$, this implies that the algorithm runs for at-max $\frac{7\beta m}{6\times 5\kappa_G} < n$ iterations.

The algorithm terminates when $Detection - Algorithm$ returns a subset for which the difference between two variance estimate is $\le 75\kappa_G$. Then Lemma 8 implies that the difference between two variance estimate is $\le 150\kappa_G$ for all subsets. Then the above corollary shows that corruption for all subsets is $\le 900\kappa_G$. Therefore, $\psi(B') \le 900\kappa_G$. Then Lemma 4 bounds the L_1 distance. ■

2.7 Proof of Theorem 2

We restate the theorem and give a short proof.

Theorem 16. *For any given $\beta \le 0.4$, n and k and m, Algorithm 2 runs in polynomial time, and its estimate p^* satisfies $||p^* - p||_1 \le O(\max\{\beta\sqrt{\frac{\ln(1/\beta)}{n}}, \sqrt{\frac{k}{mn}}\})$ with probability $\ge 1 - O(e^{-k})$.*

Proof. First we prove the theorem for $m \ge \Omega(k)$. We further divide it into two case depending on number of batches, m.

1. When the number of batches $m \ge \Omega(\frac{k}{\beta^2 \log(1/\beta)})$, then Theorem 1 implies the above result.
2. When the number of batches $m \le \mathcal{O}(\frac{k}{\beta^2 \log(1/\beta)})$, then let β_* such that $m = \Theta(\frac{k}{\beta_*^2 \log(1/\beta_*)})$. Clearly, $\beta_* \gg \beta$. From Theorem 1, the algorithm would achieve a distance $O(\beta_*\sqrt{\frac{\log(1/\beta_*)}{n}}) = O(\sqrt{\frac{k}{nm}})$.

This proves the theorem for $m \ge \Omega(k)$.

For $m \le \mathcal{O}(k)$, there are two possibilities depending on the total number of samples, mn.

1. When $mn \le \mathcal{O}(k)$, one cannot learn the distribution, hence the L_1 error is $= \Omega(1)$, and the guarantees of the theorem trivially hold.

2. When $mn \ge \Omega(k)$, divide each of the m batches into $\Theta(k/m)$ smaller batches so that there are $m' = \Theta(k)$ batches of $n' = \Theta(mn/k)$ samples each. This operation preserves the fraction β of adversarial batches. Since we already proved the theorem for $m' > \Omega(k)$, applying this result for the updated batches yields the following bound:

$$\begin{aligned}\max\{\beta\cdot\sqrt{\frac{\log(1/\beta)}{n'}}, \sqrt{\frac{k}{m'n'}}\} &= \max\{\beta\cdot\sqrt{\frac{k\cdot\log(1/\beta)}{mn}}, \sqrt{\frac{k}{mn}}\}\\ &= \sqrt{\frac{k}{mn}}\\ &\le \max\{\beta\cdot\sqrt{\frac{\log(1/\beta)}{n}}, \sqrt{\frac{k}{mn}}\},\end{aligned}$$

where the second equality follows as $\beta < 1/2$ implies $\beta\cdot\sqrt{\log(1/\beta)} < 1$.

Thereby proving the theorem for the $m \le \mathcal{O}(k)$ range. ■

Chapter 2, in full, is a reprint of the material as it appears in Optimal robust learning of discrete distributions from batches 2020. Ayush Jain, Alon Orlitsky. In ICML 2020. The dissertation author was the primary investigator and author of this paper.

Chapter 3

A General Method for Robust Learning from Batches

3.1 Introduction

3.1.1 Motivation

In many learning applications, some samples are inadvertently or maliciously corrupted. A simple and intuitive example shows that this erroneous data limits the extent to which a distribution can be learned, even with infinitely many samples. Consider p that could be one of two possible binary distributions: $(\frac{1}{2} - \frac{\beta}{2}, \frac{1}{2} + \frac{\beta}{2})$ and $(\frac{1}{2} + \frac{\beta}{2}, \frac{1}{2} - \frac{\beta}{2})$. Given any number of samples from p, an adversary who observes a $1 - \beta$ fraction of the samples and can determine the rest, could use the observed samples to learn p, and set the remaining samples to make the distribution always appear to be $(0.5, 0.5)$. Even with arbitrarily many samples, any estimator for p fails to decide which p is in effect, hence incurs a *total-variation (TV)* distance $\geq \frac{\beta}{2}$, that we call the *adversarial lower bound*.

The example may seem to suggest the pessimistic conclusion that if an adversary can corrupt a β fraction of the data, a TV-loss of $\geq \frac{\beta}{2}$ is inevitable. Fortunately, in many applications it can be avoided.

In the following applications, and many others, data is collected in batches, most of which are genuine, but some possibly corrupted. Data may be gathered by sensors, each providing a large amount of data, and some sensors may be faulty. The word frequency of an author may be estimated from several large texts, some of which are mis-attributed. User preferences may be learned by querying several individuals, some intentionally biasing their feedback. Multiple agents may contribute to a crowd-sourcing platform, but some may be unreliable or malicious. Interestingly, for data arriving in batches, even when a β-fraction of which are corrupted, more can be said.

Recently, [125] formalized the problem for finite domains. They considered estimating a distribution p over $[k]$ in TV-distance when the samples are provided in batches of size $\geq n$. Out of a total of m batches, a fraction $\leq \beta$ may be arbitrarily and adversarially corrupted, while in every other batch b the samples are drawn according to a distribution p.

For $\beta < 1/900$, they derived an estimation algorithm that approximates any p over a finite domain to TV-distance $\mathcal{O}(\beta/\sqrt{n})$, surprisingly, much lower than the individual samples limit of $\Theta(\beta)$. They also derived a matching lower bound, showing that even for binary distributions, and hence for general finite distributions, given any number m of batches, the lowest achievable TV distance is $\Delta_{\text{min}} := \Delta_{\text{min}}(\beta, n) := \frac{\beta}{2\sqrt{2n}}$. We refer to Δ_{min} as the *adversarial batch lower bound*.

Their estimator requires $\Omega(\frac{n+k}{n \cdot \Delta_{\text{min}}^2})$ batches of samples, or equivalently $\Omega(\frac{n+k}{\Delta_{\text{min}}^2})$ samples, which is not optimal if $n >> k$. It also runs in time exponential in the domain size, rendering it impractical.

Recently, [30] used a novel application of the sum-of-squares technique to reduce the exponential time complexity. Using quasi-polynomial sample size and run time, both roughly $(k/\Delta)^{\mathcal{O}(\log(1/\beta))}$, they derived an estimator that achieves TV distance $\mathcal{O}(\Delta)$, where $\Delta := \Delta(\beta, n) := \Delta_{\text{min}} \cdot \sqrt{\ln(1/\beta)}$.

Concurrently, [77] derived the first polynomial-time and optimal $\Omega(k/\Delta^2)$ sample estimator, that achieves the same $\mathcal{O}(\Delta)$ TV distance. To limit the impact of adversarial batches, the algorithm *filters* the data by removing batches that skews the estimator.

For general distributions, the sample complexity of both TV-distance estimation, and Bayes-optimal classification, grows linearly in the domain size, even when all samples are genuine.

Hence, general estimation and classification over large discrete, let alone continuous domains, is infeasible. Since most modern applications are over very large or continuous domains, this may again lead to the pessimistic conclusion that not much can be done.

Fortunately, typical distributions are not arbitrary and possess some structure. For example, they may be monotone, smooth, Lipchitz, etc., or well approximated by structured distributions. These structural properties enable learning over large and even infinite domains. For example, as is well known, classifiers can be learned using a number of samples proportional to the VC-dimension of the classifier class. But so far, our understanding of how to incorporate the distribution structure in Robust batch learning has been quite limited.

The first application of structure to reduce the linear dependence of the sample complexity [30] considered robust batch learning of t-piecewise degree-d polynomials over the finite set $[k] = \{1,\ldots,k\}$. It learned these distributions with number of samples that grows only quasi-poly-logarithmically in the domain size k. Yet this number still grows with k, hence does not extend to continuous distributions. It is also quasi-polynomial in the other parameters t, d, batch size n, and $1/\beta$, much larger than in the non-robust setting. And the algorithm's computational complexity is quasi-polynomial in these parameters and the domain size k.

This leaves several natural questions: (1) Can other non-finite, and even continuous, structured distribution classes, be learned robustly to an estimation error comparable to the adversarial batch lower $\Delta_{\min}$? (2) Can it be achieved with sample complexity comparable to the non-adversarial learning? (3) Can robust learning of structured distributions be accomplished in strict polynomial time? (4) Even more generally can other tasks such as classification be accomplished with adversarial batches? (5) Most importantly, is there a general and systematic theory of learning with adversarial batches?

3.1.2 Summary of techniques and contributions

VC theory helps answer some of the above questions when all the samples are generated i.i.d. from a distribution. We adapt the theory to address robust batch learning as well. Let $\mathcal{F}$ be a family of subsets of an Euclidean domain Ω. The *$\mathcal{F}$-distance* between two distributions p and q over Ω is the largest difference between the probabilities p and q assign to any subset in $\mathcal{F}$,

$$||p-q||_{\mathcal{F}} := \sup_{S\in\mathcal{F}} |p(S)-q(S)|.$$

It is easy to see that TV, and hence L_1, distances are a special case of $\mathcal{F}$-distance where $\mathcal{F}$ is the collection Σ of all Borel subsets of Ω, $||p-q||_\Sigma = ||p-q||_{\text{TV}} = \frac{1}{2}||p-q||_1$.

Without adversarial batches, the VC inequality guarantees that for a subset family $\mathcal{F}$ with finite VC-dimension, the empirical distribution of samples from p estimates p to a small $\mathcal{F}$-distance. But with adversarial batches, the $\mathcal{F}$-distance between the empirical distribution and p could be large.

For learning with adversarial batches over finite domains, [77] presented an algorithm that learns the distribution to a small TV distance with a number of batches proportional to the domain size. We generalize this algorithm to learn any finite-VC subset family $\mathcal{F}$ to a small $\mathcal{F}$-distance using samples linear in the family's VC-dimension, rather than the domain size.

Recall that $\Delta_{\text{min}} = \beta/(2\sqrt{2n})$ is the adversarial batch lower bound for TV-distance learning. No algorithm achieves an error below Δ_{min}, even with the number of batches $\to \infty$. Since the Δ_{min} lower bound applies even to binary domains, it can be shown to also lower bound $\mathcal{F}$-distance learning.

Our proposed algorithm *filters* the batches and returns a sub-collection of batches whose empirical distribution estimates p to $\mathcal{F}$-distance $\mathcal{O}(\Delta)$, where $\Delta = \Delta_{\text{min}} \cdot \sqrt{\log(1/\beta)}$ is only a small factor above the lower bound. The number of batches it requires for any VC family $\mathcal{F}$ is only a logarithmic factor more than needed to achieve the same error without adversarial batches, showing that robustness can be incorporated at little extra cost. This provides the first demonstration that distributions can be learned (1) robustly and (2) sample-efficiently, over infinite, and even continuous domains.

As expected from the setting's vast generality, as in the non-adversarial setting, for some

VC families, one cannot expect to find a computationally efficient algorithm. We, therefore, consider a natural and important VC family over the reals that, as we shall soon see, translates into efficient and robust algorithms for TV-learning and classification over $\mathbb{R}$.

Let $\mathcal{F}_k$ be the family of all unions of at most k intervals over $\mathbb{R}$. We derive a computationally efficient algorithm that estimates distributions to $\mathcal{F}_k$-distance $\mathcal{O}(\Delta)$ using only $\tilde{\mathcal{O}}(1/\Delta)$ times more samples than the non-adversarial, or information-theoretic adversarial cases.

Building on these techniques, we return to estimation in total variation (TV) distance. We consider the family of distributions whose Yatracos Class [151] have finite VC dimension. This family consists of both discrete and continuous distributions, and includes piecewise polynomials, Gaussians in one or more dimensions, and arguably most practical distribution families. We show that all these distributions can be learned robustly from batches to a TV distance $\mathcal{O}(\Delta)$, which is only a factor $\sqrt{\log(1/\beta)}$ above the adversarial TV-distance lower bound of $\Delta_{\min}$. It also achieves sample complexity that is at most a logarithmic factor more than required for non-adversarial case.

These results too are very general, hence as in the non-adversarial case, one cannot expect a computationally efficient algorithm for all cases. We therefore consider the natural and important general class $\mathcal{P}_{t,d}$ of t-piecewise degree-d polynomial distributions over the reals.

To agnostically learn distributions $\mathcal{P}_{t,d}$, we combine the results above with an existing, non-adversarial, polynomial-learning algorithm [1]. We derive a polynomial-time algorithm for estimating polynomials in $\mathcal{P}_{t,d}$ to a TV distance $\mathcal{O}(\Delta)$. The algorithm's sample complexity is linear in td, which is the best possible, and similar to learning in $\mathcal{F}_k$-distance, only $\tilde{\mathcal{O}}(1/\Delta)$ times above the non-adversarial, or information-theoretic adversarial sample complexity.

This is the first algorithm that achieves polynomial sample and time complexity for robust learning for this class, and the first that applies to the non-finite domains.

The general formulation also allows us to use batch-structure for robustness in other learning tasks. We apply this framework to derive the first robust agnostic classifiers. The goal is to minimize the excess loss in comparison to the best hypothesis, in the presence of adversarial batches.

We first modify the lower bound on distribution learning to show that any classification algorithm with adversarial batches must incur an excess loss $\mathcal{O}(\Delta_{\min})$, even with the number of batches $\to \infty$. We then derive a general algorithm that achieves additive excess loss $\mathcal{O}(\Delta)$ for general binary classification using a number of samples that is again only a logarithmic factor larger than required to achieve the same excess loss in the non-adversarial setting.

Finally, we consider classification over $\mathbb{R}$. Many natural and practical classifiers have decision regions consisting of finitely many disjoint intervals. We apply the above results to derive a computationally efficient algorithm for hypotheses consisting of k intervals. Similar to previous results, its sample complexity is linear in k and only a factor $\mathcal{O}(1/\Delta)$ larger than required in the non-adversarial setup.

The rest of the paper is organized as follows. Section 3.2 describes the main technical results and their applications to distribution estimation and classification. Section 3.3 discusses the other related work. Section 3.4 provides an overview of the filtering algorithm that enables these results. Proofs and more details are relegated to the appendix.

3.2 Results

We consider learning from batches of samples, when a $\beta-$fraction of batches are adversarial.

More precisely, B is a collection of m batches, composed of two *unknown* sub-collections. A *good sub-collection* $B_G \subseteq B$ of $\geq (1-\beta)m$ *good batches*, where each batch b consists of n independent samples from a common distribution p over Ω. And an *adversarial sub-collection* $B_A = B \setminus B_G$ of the remaining $\leq \beta m$ batches, each consisting of the same number n of arbitrary Ω elements, that for simplicity we call *samples* as well. Note that the adversarial samples may be chosen in any way, including after observing the good samples.

The next subsection describes the main technical results for learning in $\mathcal{F}$ distance.

Subsequent subsections apply these results to learn distributions in TV distance and to achieve robust binary classification.

3.2.1 Estimating distributions in $\mathcal{F}$ distance

Our goal is to use samples generated by a target distribution p to approximate it to a small $\mathcal{F}$-distance. For general families $\mathcal{F}$, this goal cannot be accomplished even with just good batches. Let $\mathcal{F} = \Sigma$ be the collection of all subsets of the real interval domain $\Omega = [0, 1]$. For any total number t of samples, with high probability, it is impossible to distinguish the uniform distribution over $[0, 1]$ from a uniform discrete distribution over a random collection of $\gg t^2$ elements in $[0, 1]$. Hence any estimator must incur TV-distance 1 for some distribution.

This difficulty is addressed by Vapnik-Chervonenkis (VC) Theory. The collection $\mathcal{F}$ *shatters* a subset $S \subseteq \Omega$ if every subset of S is the intersection of S with a subset in $\mathcal{F}$. The VC-dimension $V_{\mathcal{F}}$ of $\mathcal{F}$ is the size of the largest subset shattered by $\mathcal{F}$.

Let $X^t = X_1, \dots, X_t$, be i.i.d. samples from a distribution p. The empirical probability of $S \subseteq \Omega$ is

$$\bar{p}_t(S) := |\{i : X_i \in S\}|/t.$$

The fundamental *Uniform deviation inequality* of VC theory [145, 138] states that if $\mathcal{F}$ has finite VC-dimension $V_{\mathcal{F}}$, then $\bar{p}_t$ estimates p well in $\mathcal{F}$ distance. For all $\delta > 0$, with probability $> 1 - \delta$,

$$||p - \bar{p}_t||_{\mathcal{F}} \leq \mathcal{O}\big(\sqrt{(V_{\mathcal{F}} + \log 1/\delta)/t}\,\big).$$

The above is also the lowest achievable $\mathcal{F}$-distance, hence we call it the *information-theoretic limit*.

In the adversarial-batch scenario, a fraction β of the batches may be corrupted. It is easy to see that for any number m of batches, however large, the adversary can cause $\bar{p}_t$ to approximate p to $\mathcal{F}$-distance $\geq \beta/2$, namely $||\bar{p}_t - p||_{\mathcal{F}} \geq \beta/2$.

Let $\bar{p}_{B'}$ be the empirical distribution induced by the samples in a collection $B' \subseteq B$. Our first result states that if $\mathcal{F}$ has a finite VC-dimension, for total samples $m \cdot n \geq \tilde{\mathcal{O}}(V_{\mathcal{F}}/\Delta^2)$, the batches in B can be "cleaned" to a sub-collection B^* where $||p - \bar{p}_{B^*}||_{\mathcal{F}} = \mathcal{O}(\Delta)$, namely, a simple empirical estimator of the samples in B^* recovers p to a small $\mathcal{F}$-distance.

Theorem 17. *For any $\mathcal{F}$, $\beta \leq 0.4$, $\delta > 0$, and $mn \geq \tilde{\mathcal{O}}\left(\frac{V_{\mathcal{F}}+\log 1/\delta}{\Delta^2}\right)$, there is an algorithm that w.p. ≥ 1–δ returns a sub-collection $B^* \subseteq B$ s.t. $|B^* \cap B_G| \geq (1-\frac{\beta}{6})|B_G|$ and $||p-\bar{p}_{B^*}||_{\mathcal{F}} \leq \mathcal{O}(\Delta)$.*

The $\mathcal{F}$-distance bound matches the lower bound Δ_{min} up to a small $\mathcal{O}(\sqrt{\log(1/\beta)})$ factor. The number $m \cdot n$ of samples required to achieve this estimation error are the same (up to a logarithmic factor) as the minimum required to achieve the same estimation error even for the non-adversarial setting.

The theorem applies to all families with finite VC dimension, and like most other results of this generality, it is necessarily non-constructive in nature. Yet it provides a road map for constructing efficient algorithms for many specific natural problems. In Section 3.4 we use this approach to derive a polynomial-time algorithm that learns distributions with respect to one of the most important and practical VC classes, where $\Omega = \mathbb{R}$, and $\mathcal{F} = \mathcal{F}_k$ is the collection of all unions of at most k intervals.

Theorem 18. *For any $k > 0$, $\beta \leq 0.4$, $\delta > 0$, and $mn \geq \tilde{\mathcal{O}}\left(\frac{k+\log 1/\delta}{\Delta^3}\right)$, there is an algorithm that runs in time polynomial in all parameters, and with probability $\geq 1 - \delta$ returns a sub-collection $B^* \subseteq B$, such that $|B^* \cap B_G| \geq (1 - \frac{\beta}{6})|B_G|$ and $||p - \bar{p}_{B^*}||_{\mathcal{F}_k} \leq \mathcal{O}(\Delta)$.*

The above polynomial-time algorithm can achieve $\mathcal{F}_k$ error Δ using the number of samples only $\tilde{\mathcal{O}}(1/\Delta)$ times the minimum required to achieve the same estimation error by any algorithm even for the non-adversarial setting. Note that the sample complexity in both Theorems 17 and 18 are independent of the domain size and depends linearly on the VC dimension of the subset family.

Section 3.4 provides a short overview of the algorithms used in the above theorems. The complete algorithms and proof of the two theorems appear in the appendix.

3.2.2 Learning distributions in total-variation distance

Our ultimate objective is to estimate the target distribution in total variation (TV) distance, one of the most common measures in distribution estimation. In this and the next subsection, we follow a framework developed in [43], see also [44].

As noted earlier, the sample complexity of estimating distributions in TV-distance grows with the domain size, becoming infeasible for large discrete domains and impossible for continuous domains. A natural approach to address this intractability is to assume that the underlying distribution belongs to, or is near, a structured class $\mathcal{P}$ of distributions.

Let $\text{opt}_{\mathcal{P}}(p) := \inf_{q\in\mathcal{P}} ||p-q||_{TV}$ be the TV-distance of p from the closest distribution in $\mathcal{P}$. For example, for $p \in \mathcal{P}$, $\text{opt}_{\mathcal{P}}(p) = 0$. Given $\epsilon, \delta > 0$, we try to use samples from p to find an estimate $\hat{p}$ such that, with probability $\geq 1-\delta$,

$$||p-\hat{p}||_{TV} \leq \alpha \cdot \text{opt}_{\mathcal{P}}(p) + \epsilon$$

for a universal constant $\alpha \geq 1$, namely, to approximate p about as well as the closest distribution in $\mathcal{P}$.

Following [43], we utilize a connection between distribution estimation and VC dimension. Let $\mathcal{P}$ be a class of distributions over Ω. The *Yatracos class* [151] of $\mathcal{P}$ is the family of Ω subsets

$$\mathcal{Y}(\mathcal{P}) := \{\{\omega \in \Omega : p(\omega) \geq q(\omega)\} : \, p, q \in \mathcal{P}\}.$$

It is easy to verify that for distributions $p, q \in \mathcal{P}$, $||p-q||_{TV} = ||p-q||_{\mathcal{Y}(\mathcal{P})}$. The *Yatracos minimizer* of a distribution p is its closest distribution, by $\mathcal{Y}(\mathcal{P})$-distance, in $\mathcal{P}$,

$$\psi_{\mathcal{P}}(p) := \arg\min_{q\in\mathcal{P}} ||q-p||_{\mathcal{Y}(\mathcal{P})},$$

where ties are broken arbitrarily. Theorem 6.3 in [43] uses these definitions and a sequence of triangle inequalities to show that for any distributions p, p', and any distribution class $\mathcal{P}$,

$$||p - \psi_{\mathcal{P}}(p')||_{TV} \leq 3 \cdot \text{opt}_{\mathcal{P}}(p) + 4||p - p'||_{\mathcal{Y}(\mathcal{P})}. \tag{3.1}$$

Therefore, given a distribution p' that approximates p in $\mathcal{Y}(\mathcal{P})$-distance, its Yatracos minimizer $\psi_{\mathcal{P}}(p')$ approximates p in TV-distance.

If the Yatracos class $\mathcal{Y}(\mathcal{P})$ has finite VC dimension, the VC-bound ensures that for the empirical distribution $\bar{p}_t$ of t i.i.d. samples from p, $||\bar{p}_t - p||_{\mathcal{Y}(\mathcal{P})}$ decreases to zero as t increases, and $\psi_{\mathcal{P}}(\bar{p}_t)$ can be used to approximate p in TV-distance. This general method has led to many sample-and computationally-efficient algorithms for estimating structured distributions, *e.g.,* [1].

However, as discussed earlier, with a β-fraction of adversarial batches, the empirical distribution of all samples can be at a $\mathcal{Y}(\mathcal{P})$-distance as large as $\Theta(\beta)$ from p, leading to a large TV-distance.

Yet Theorem 17 shows that data can be "cleaned" to remove outlier batches and retain batches $B^* \subseteq B$ whose empirical distribution $\bar{p}_{B^*}$ approximates p to a much smaller $\mathcal{Y}(\mathcal{P})$-distance of $\mathcal{O}(\Delta)$. Letting $p^* = \psi_{\mathcal{P}}(\bar{p}_{B^*})$ and using Equation (3.1), we obtain a much better approximation of p in TV distance.

Theorem 19. *For a distribution class $\mathcal{P}$ with Yatracos Class of finite VC dimension v, for any $\beta \leq 0.4$, $\delta > 0$, and $mn \geq \tilde{\mathcal{O}}\left(\frac{v+\log 1/\delta}{\Delta^2}\right)$, there is an algorithm that w. p. $\geq 1-\delta$ returns a distribution $p^* \in \mathcal{P}$ such that $||p - p^*||_{TV} \leq 3 \cdot opt_{\mathcal{P}}(p) + \mathcal{O}(\Delta)$.*

The estimation error achieved in the theorem for TV-distance matches the lower bound to a small log factor of $O(\sqrt{\log(1/\beta)})$, and is valid for any class $\mathcal{P}$ with finite VC Dimensional Yatracos Class.

Moreover, the upper bound on the number of samples (or batches) required by the algorithm to estimate p to the above distance matches a similar general upper bound obtained for non-adversarial setting to a log factor. This results for the first time shows that it is possible to learn a wide variety of distributions robustly using batches, even over continuous domains.

3.2.3 Learning univariate structured distributions

We apply the general results in the last two subsections to estimate distributions over the real line. We focus on one of the most studied, and important, distribution families, the class of piecewise-polynomial distributions. A distribution p over $[a, b]$ is t-piecewise, degree-d, if there is a partition of $[a, b]$ into t intervals $I_1, \ldots, I_t$, and degree-d polynomials $r_1, \ldots, r_t$ such that $\forall j$ and $x \in I_j$, $p(x) = r_j(x)$. The definition extends naturally to finite distributions over $[k] = \{1, \ldots, k\}$.

Let $\mathcal{P}_{t,d}$ denote the collection of all t-piecewise degree d distributions. $\mathcal{P}_{t,d}$ is interesting in its own right, as it contains important distribution classes such as histograms. In addition, it approximates other important distribution classes, such as monotone, log-concave, Gaussians, and their mixtures, arbitrarily well, *e.g.,* [1].

Note that for any two distributions $p, q \in \mathcal{P}_{t,d}$, the difference $p - q$ is a $2t$-piecewise degree-d polynomial, hence every set in the Yatracos class of $\mathcal{P}_{t,d}$, is the union of at most $2t \cdot d$ intervals in $\mathbb{R}$. Therefore, $\mathcal{Y}(\mathcal{P}_{t,d}) \subseteq \mathcal{F}_{2t \cdot d}$. And since $V_{\mathcal{F}_k} = O(k)$ for any k, $\mathcal{Y}(\mathcal{P}_{t,d})$ has VC dimension $\mathcal{O}(td)$.

Theorem 19 can then be applied to show that any target distribution p can be estimated by a distribution in $\mathcal{P}_{t,d}$ to a TV-distance Δ, using a number of samples, that is within a logarithmic factor from the minimum required [27] even when all samples are i.i.d. generated from p.

Corollary 20. *For any distribution p over $\mathbb{R}$, t, d, $\beta \leq 0.4$, $\delta > 0$, and $mn \geq \tilde{\mathcal{O}}\left(\frac{td+\log 1/\delta}{\Delta^2}\right)$, there is an algorithm that with probability $\geq 1-\delta$ returns a distribution $p^* \in \mathcal{P}_{t,d}$ such that $||p - p^*||_{TV} \leq 3 \cdot opt_{\mathcal{P}_{t,d}}(p) + \mathcal{O}(\Delta)$.*

Next we provide a polynomial-time algorithm for estimating p to the same $\mathcal{O}(\Delta)$ TV-distance, but with an extra $\tilde{\mathcal{O}}(1/\Delta)$ factor in sample complexity.

Theorem 18 provides a polynomial time algorithm that returns a sub-collection $B^* \subseteq B$ of batches whose empirical distribution $\bar{p}_{B^*}$ is close to p in $\mathcal{F}_{2td}$-distance. [1] provides a polynomial time algorithm that for any distribution q returns a distribution in $p' \in \mathcal{P}_{t,d}$ minimizing $||p' - q||_{\mathcal{F}_{2td}}$

to a small additive error. Then equation (3.1) and Theorem 18 yield the following result. We provide formal proof of the theorem in the appendix.

Theorem 21. *For any distribution p over $\mathbb{R}$, n, m, $\beta \leq 0.4$, t, d, $\delta > 0$, and $mn \geq \tilde{\mathcal{O}}\Big(\frac{td+\log 1/\delta}{\Delta^3}\Big)$, there is a polynomial time algorithm that w. p. $\geq 1-\delta$ returns a distribution $p^* \in \mathcal{P}_{t,d}$ such that $||p - p^*||_{TV} \leq \mathcal{O}(opt_{\mathcal{P}_{t,s}}(p)) + \mathcal{O}(\Delta)$.*

3.2.4 Binary classification

Our framework extends beyond distribution estimation. Here we describe its application to Binary classification. Consider a family $\mathcal{H} : \Omega \to \{0, 1\}$ of Boolean functions, and a distribution p over $\Omega \times \{0, 1\}$. Let $(X, Y) \sim p$, where $X \in \Omega$ and $Y \in \{0, 1\}$. The loss of classifier $h \in \mathcal{H}$ for distribution p is $r_p(h) := \Pr_{(X,Y)\sim p}[h(X) \neq Y]$. The *optimal classifier* for distribution p is $h^{\text{opt}}(p) := \arg\min_{h\in\mathcal{H}} r_p(h)$, and hence the *optimal loss* is $r_p^{\text{opt}}(\mathcal{H}) := r_p(h^{\text{opt}}(p))$. The goal is to return a classifier $h \in \mathcal{H}$ whose *excess loss* $r_p(h) - r_p^{\text{opt}}(\mathcal{H})$ compared to the optimal loss is small.

Consider the following natural extension of VC-dimension from families of subsets to families of Boolean functions. For a boolean-function family $\mathcal{H}$, define the family

$$\mathcal{F}_{\mathcal{H}} := \{(\{\omega \in \Omega : h(\omega) = y\}, \bar{y}) : h \in \mathcal{H}, y \in \{0, 1\}\}$$

of subsets of $\Omega \times \{0, 1\}$, and let the VC dimension of $\mathcal{H}$ be $V_{\mathcal{H}} := V_{\mathcal{F}_{\mathcal{H}}}$.

The next simple lemma, proved in the appendix, upper bounds the excess loss of the optimal classifier in $\mathcal{H}$ for a distribution q for another distribution p in terms of $\mathcal{F}_{\mathcal{H}}$ distance between the distributions.

Lemma 22. *For any class $\mathcal{H}$ and distributions p and q, $r_p(h^{opt}(q)) - r_p^{opt}(\mathcal{H}) \leq 4||p - q||_{\mathcal{F}_{\mathcal{H}}}$.*

When q is an empirical distribution of the samples, $h^{\text{opt}}(q)$ is called the *empirical-risk minimizer*. If q is the empirical distribution of the samples generated i.i.d. from p, from VC inequality, the excess loss of the empirical-risk minimizer in the above equation goes to zero if VC dimension of $\mathcal{H}$ is finite.

Yet as discussed earlier, when a β-fractions of the batches, and hence samples, are chosen by an adversary, the empirical distribution of all samples can be at a large $\mathcal{F}_{\mathcal{H}}$-distance $\mathcal{O}(\beta)$ from p, leading to an excess-classification-loss up to $\Omega(\beta)$ for the empirical-risk minimizer.

Theorem 17 states that the collection of batches can be "cleaned" to obtain a sub-collection $B^* \subseteq B$ whose empirical distribution has a lower $\mathcal{F}_{\mathcal{H}}$-distance from p. The above lemma then implies that the optimal classifier $h^{\text{opt}}(\bar{p}_{B^*})$ for the empirical distribution $\bar{p}_{B^*}$ of the cleaner batches will have a small-excess-classification-loss for p as well. The resulting non-constructive algorithm has excess-classification-loss and sample complexity that are optimal to a logarithmic factor.

Theorem 23. *For any $\mathcal{H}$, p, $\beta \leq 0.4$, $\delta > 0$, and $mn \geq \tilde{\mathcal{O}}\left(\frac{V_{\mathcal{H}}+\log 1/\delta}{\Delta^2}\right)$, there is an algorithm that with probability $\geq 1-\delta$ returns a classifier h^*, whose excess lose is $r_p(h^*) - r_p^{opt}(\mathcal{H}) \leq \mathcal{O}(\Delta)$.*

To complement this result, we show an information-theoretic lower bound of $\Omega(\Delta_{\text{min}})$ on the excess loss. The proof is in the appendix. Recall that a similar lower bound holds for learning distribution.

Theorem 24. *For any β, n, and $\mathcal{H}$ s.t. $V_{\mathcal{H}} \geq 1$, there are a distribution p and an adversary, such that any algorithm, with probability $\geq 1/2$, incurs an excess loss $\Omega(\Delta_{min})$, even as number of batches $m \to \infty$.*

To derive a computationally-efficient algorithm, we focus on the following class of binary functions. For $k \geq 1$, let $\mathcal{H}_k$ denote the collection of all binary functions over $\mathbb{R}$ whose decision region, namely values mapping to 0 or 1, consists of at most k-intervals. The VC dimension of $\mathcal{F}_{\mathcal{H}_k}$ is clearly $\mathcal{O}(k)$.

Theorem 18 describes a polynomial-time algorithm that returns a cleaner data w.r.t. $\mathcal{F}_{\mathcal{H}_k}$ distance. From Lemma 22, the classifier that minimizes the loss for the empirical distribution of this cleaner data will have a small excess loss. Furthermore, [105] derived a polynomial-time algorithm to find the empirical risk minimizer $h \in \mathcal{H}_k$ for any given samples. Combining these results, gives a robust computationally efficient classifier in $\mathcal{H}_k$. We provide a formal proof in the appendix.

Theorem 25. *For any k, p, $\beta \leq 0.4$, $\delta > 0$, and $mn \geq \tilde{\mathcal{O}}\left(\frac{k+\log 1/\delta}{\Delta^3}\right)$, there is a polynomial-time algorithm that w. p. ≥ 1–δ returns a classifier h^*, whose excess loss is $r_p(h^*) - r_p^{opt}(\mathcal{H}_k) \leq \mathcal{O}(\Delta)$.*

3.3 Other related and concurrent work

The current results extend several long lines of work on estimating structured distributions, including [115, 44, 8, 1]. The results also relate to classical robust-statistics work [142, 74]. There has also been significant recent work leading to practical distribution learning algorithms that are robust to adversarial contamination of the data. For example, [47, 99] presented algorithms for learning the mean and covariance matrix of high-dimensional sub-gaussian and other distributions with bounded fourth moments in presence of the adversarial samples. Their estimation guarantees are typically in terms of L_2, and do not yield the L_1- distance results required for discrete distributions. The work was extended in [29] to the case when more than half of the samples are adversarial. Their algorithm returns a small set of candidate distributions one of which is a good approximate of the underlying distribution. The filtering based method has also played a key role in other robust learning algorithms in high dimension [48, 50, 135, 46]. These works apply filtering on samples instead on batches of samples, as in [77] and in this paper, and recover in a different metric. For a more extensive survey on robust learning algorithms see [135, 46].

Another motivation for this work derives from the practical federated-learning problem, where information arrives in batches [108, 109].

Concurrent work Concurrent to our work, [31] also extends the filtering algorithm of [77] to obtain robust batch learning algorithms for estimating piecewise polynomials. They derive a polynomial-time algorithm that learns distributions in $\mathcal{P}_{t,d}$ over a finite domain $[k]$ to the same TV distance $\mathcal{O}(\Delta)$ as we do, but requires $\tilde{\mathcal{O}}((td)^2 \log^3(k)/\Delta^2)$ samples, where $\tilde{O}$ hides a logarithmic factor in $1/\Delta$. In contrast, our results show that this accuracy can be achieved using $\tilde{\mathcal{O}}(td/\Delta^2)$ samples, and by a polynomial-time algorithm with sample complexity is $\tilde{\mathcal{O}}(td/\Delta^3)$. Importantly, our algorithms' complexity does not depend on the alphabet size $[k]$, which allows us to extend them to more general non-finite and even continuous domains. In addition, we considered other distribution classes and learning tasks such as classification.

Another concurrent work [96] focuses on the sample complexity of robust batch classification using adversarial batches. Their results achieve an excess loss of $\mathcal{O}(\sqrt{V_{\mathcal{H}}} \cdot \Delta)$, where $V_{\mathcal{H}}$ is the VC-dimension of the hypothesis class, whereas we achieve an excess loss only $\mathcal{O}(\Delta)$.

3.4 Overview of the filtering framework for learning in $\mathcal{F}$ distance

To derive both the information-theoretic and computationally-efficient algorithms for general robust learning from batches, we generalize a finite filtering-based approach in [77]. We first describe the original algorithm and outline how it can be extended to general learning problems. A more complete and formal presentation appears in the appendix.

Recall that B is the collection of all m batches and each batch $b \in B$ has n samples from the domain Ω. A batch b estimates the probability $p(S)$ of a subset $S \in \Sigma$ by its empirical probability. Each subset $S \in \Sigma$, assigns to every batch $b \in B$, a *corruption score* $\psi_b(S)$, defined in the appendix, based on how far the batch's estimate of $p(S)$ is from the median of the estimates for all batches. Similarly, each subset S assigns to every sub-collection $B' \subseteq B$ of batches a corruption score $\psi_{B'}(S) := \sum_{b \in B'} \psi_b(S)$, the sum of individual corruption score of each batch.

We first describe a general *filtering* approach to robust learning from batches. A collection $C \subseteq \Sigma$ of subsets, is *learnable via filtering* if one can "filter out" bad batches in B and find a "good" subset $B^* \subseteq B$ of batches that approximates p to a small C-distance,

$$||p - \bar{p}_{B^*}||_C = \max_{S \in C} |p(S) - \bar{p}_{B^*}(S)| \leq \mathcal{O}(\Delta). \tag{3.2}$$

We describe two properties ensuring that C is learnable via filtering. A finite $C \subseteq \Sigma$ is learnable via filtering if there is a threshold τ such that all subsets $S \in C$ and all sub-collection $B' \subseteq B$ that contain most good batches, namely $|B' \cap B_G| \geq (1 - \beta/6)|B_G|$, satisfy the following two properties:

1. If the corruption score is low, $\psi_{B'}(S) < \tau$, then B' estimates $p(S)$ well, $|p(S) - \bar{p}_{B'}(S)| = \mathcal{O}(\Delta)$.

2. If $\psi_{B'}(S) > \tau$, then there is a (probabilistic) method that removes batches in B', while ensuring that and each batch removed is adversarial with probability at least 0.95, until $\psi_{B'}(S) < \tau$.

A simple algorithm shows that these two properties imply that C is learnable by filtering. Start with $B' = B$, find a filter $S \in C$ with $\psi_{B'}(S) > \tau$, remove the batches from B', and repeat the process until the corruption is small, $\psi_{B'}(S) < \tau$, for all filters in C. By property 2, each deleted batch is adversarial with probability > 0.95. Since there are at most βm adversarial batches, w.h.p. at most $0.1\beta m$ good batches are deleted. Consequently $|B' \cap B_G| \geq (1 - \beta/6)|B_G|$. By property 1, when the algorithm ends, $B^* = B'$ achieves (3.2).

While this algorithm describes the core of the technique, three significant challenges remain.

The above algorithm applies for finite classes C. However, the VC class $\mathcal{F}$ may be infinite, or even uncountable. To apply the algorithm we need to find a finite subset C such that learning in C distance implies learning in $\mathcal{F}$ distance. In the appendix, we prove an essential *Robust Covering Theorem*, showing that for an appropriate ϵ, letting C be an ϵ-cover of $\mathcal{F}$ under empirical density $\bar{p}_B$, suffices to learn p in $\mathcal{F}$ distance. This is despite the fact that a fraction β of the batches in B may be adversarially chosen, and even depend on good samples.

The next key challenge is to show that the two properties hold for all subsets in the ϵ-cover. We establish this fact by showing that with sufficiently many batches, w.h.p., the two properties hold for all subsets $S \in \mathcal{F}$. The proof requires addressing additional technical challenges, as number of subsets in $\mathcal{F}$ could be infinite.

Choosing any finite ϵ-cover $C \subseteq \mathcal{F}$ under density $\bar{p}_B$, therefore yields an information-theoretic algorithm with near-optimal sample complexity. This gives us the near sample optimal algorithm in Theorem 17. However, computationally-efficient algorithms pose one additional challenge. The size of C may be exponential in the VC dimension, and hence searching for a subset in C with a high corruption score may be computationally infeasible.

For the VC class $\mathcal{F}_k$, we overcome this difficulty by choosing the set C of filters from a larger class than $\mathcal{F}_k$ itself so that that still obeys the two properties, but allows for an efficient search. Though C is chosen from a larger class, we ensure that the sample complexity increase is small. Specifically, we let C be the collection of all subsets of a k'-partition of Ω, for an appropriate k' that is linear in k. Subsets in such a cover C correspond to binary vectors in $\{0,1\}^{k'}$. A novel semi-definite-programming based algorithm derived in [77] finds a subset $S \in C$ with nearly the highest corruption $\psi_{B'}(S)$ in time only polynomial in k'. This allows us to obtain the polynomial-time algorithm in Theorem 18.

To summarize, this universal filtering approach allows us to "clean" the data and enables the general robust distribution estimators and classifiers we construct.

Remark. In some applications the distributions underlying genuine batches may differ from the common target distribution p by a small TV distance, say $\eta > 0$. For simplicity, in this paper we presented the analysis for $\eta = 0$, where all the good batches have the same distribution p. For $\eta > 0$, even for binary alphabets, [125] derived the adversarial batch lower bound of $\Omega(\eta + \beta/\sqrt{n})$ on TV distance. And even the trivial empirical estimator achieves $\mathcal{O}(\eta + \beta)$ TV-error, which has optimal linear dependence on η. Therefore, filtering algorithms do not need to do anything sophisticated for general η and incurs only an extra $\mathcal{O}(\eta)$ error as noted in [77] for unstructured distributions, and the same holds for our algorithms for learning structured distributions and binary classification.

Appendix

The Appendix is organized as follows: Section 3.5 introduces notation and states some useful facts. Section 3.6 recounts basic tools from VC theory used to derive the results. Section 3.7 derives a framework for robust distribution estimation in $\mathcal{F}$-distance and proves Theorem 17. Building on this framework it then develops computationally efficient algorithms for learning in $\mathcal{F}_k$ distance and proves Theorem 18. Section 3.8 gives the proof of the filtration properties and other results used in Section 3.7. Section 3.9 gives the other remaining proofs of the main paper.

3.5 Preliminaries

We introduce terminology that helps describe the approach and results. Some of the work builds on results in [77], and we keep the notation consistent.

Recall that B, B_G, and B_A are the collections of all-, good-, and adversarial-batches. Let $B' \subseteq B$, $B'_G \subseteq B_G$, and $B'_A \subseteq B_A$, denote sub-collections of all-, good-, and bad-batches. We also let S denote a subset in the Borel σ-field Σ on domain Ω.

Let $X_1^b, X_2^b, ..., X_n^b$ denote the n samples in a batch b, and let $\mathbf{1}_S$ denote the indicator random variable for a subset $S \in \Sigma$. Every batch $b \in B$ induces an empirical measure $\bar{\mu}_b$ over the domain Ω, where for each $S \in \Sigma$,

$$\bar{\mu}_b(S) := \frac{1}{n} \sum_{i \in [n]} \mathbf{1}_S(X_i^b).$$

Similarly, any sub-collection $B' \subseteq B$ of batches induces an empirical measure $\bar{p}_{B'}$ defined by

$$\bar{p}_{B'}(S) := \frac{1}{|B'|n} \sum_{b \in B'} \sum_{i \in [n]} \mathbf{1}_S(X_i^b) = \frac{1}{|B'|} \sum_{b \in B'} \bar{\mu}_b(S).$$

We use two different symbols to denote empirical distribution defined by single batch and a sub-collection of batches to make them easily distinguishable. Note that $\bar{p}_{B'}$ is the mean of the empirical measures $\bar{\mu}_b$ defined by the batches $b \in B'$.

Recall that n is the batch size. For $r \in [0,1]$, let $\mathrm{V}(r) := \frac{r(1-r)}{n}$, the variance of a Binomial(r, n) random variable. Observe that

$$\forall\, r, s \in [0,1],\ \mathrm{V}(r) \le \frac{1}{4n} \quad \text{and} \quad |\mathrm{V}(r) - \mathrm{V}(s)| \le \frac{|r-s|}{n}, \tag{3.3}$$

where the second property follows as $|r(1-r) - s(1-s)| = |r-s| \cdot |1-(r+s)| \le |r-s|$.

For $b \in B_G$, the random variables $\mathbf{1}_S(X_i^b)$ for $i \in [n]$ are distributed i.i.d. Bernoulli$(p(S))$, and since $\bar{\mu}_b(S)$ is their average,

$$E[\,\bar{\mu}_b(S)\,] = p(S) \quad \text{and} \quad \mathrm{Var}[\,\bar{\mu}_b(S)\,] = E[(\bar{\mu}_b(S) - p(S))^2] = \mathrm{V}(p(S)).$$

For batch collection $B' \subseteq B$ and subset $S \in \Sigma$, the empirical probability $\bar{\mu}_b(S)$ of S will vary with the batch $b \in B'$. The *empirical variance* of these empirical probabilities is

$$\overline{\mathrm{V}}_{B'}(S) := \frac{1}{|B'|} \sum_{b \in B'} (\bar{\mu}_b(S) - \bar{p}_{B'}(S))^2.$$

3.6 Vapnik-Chervonenkis (VC) theory

We recall some basic concepts and results in VC theory, and derive some of their simple consequences that we use later in deriving our main results.

The *VC shatter coefficient* of $\mathcal{F}$ is

$$S_{\mathcal{F}}(t) := \sup_{x_1,x_2,..,x_t \in \Omega} |\{\{x_1, x_2, .., x_t\} \cap S : S \in \mathcal{F}\}|,$$

the largest number of subsets of t elements in Ω obtained by intersections with subsets in $\mathcal{F}$. The VC dimension of $\mathcal{F}$ is

$$V_{\mathcal{F}} := \sup\{t : S_{\mathcal{F}}(t) = 2^t\},$$

the largest number of Ω elements that are "fully shattered" by $\mathcal{F}$. The following Lemma [43] bounds the Shatter coefficient for a VC family of subsets.

Lemma 26 ([43])**.** *For all* $t \geq V_{\mathcal{F}}$*,* $\quad S_{\mathcal{F}}(t) \leq \left(\frac{t\,e}{V_{\mathcal{F}}}\right)^{V_{\mathcal{F}}}$.

Next we state the VC-inequality for relative deviation [144, 7].

Theorem 27. *Let* p *be a distribution over* (Ω, Σ)*, and* $\mathcal{F}$ *be a VC-family of subsets of* Ω *and* $\bar{p}_t$ *denote the empirical distribution from* t *i.i.d samples from* p*. Then for any* $\epsilon > 0$*, with probability* $\geq 1 - 8S_{\mathcal{F}}(2t)e^{-t\epsilon^2/4}$*,*

$$\sup_{S \in \mathcal{F}} \max\left\{\frac{\bar{p}_t(S) - p(S)}{\sqrt{\bar{p}_t(S)}}, \frac{p(S) - \bar{p}_t(S)}{\sqrt{p(S)}}\right\} \leq \epsilon.$$

Another important ingredient commonly used in VC Theory is the concept of covering number that reflects the smallest number of subsets that approximate each subset in the collection.

Let p be any probability measure over (Ω, Σ) and let $\mathcal{F} \subseteq \Sigma$ be a family of subsets. A collection $\mathcal{C} \subseteq \Sigma$ of subsets is an ϵ*-cover* of $\mathcal{F}$ under distribution p if for any $S \in \mathcal{F}$, there exists a $S' \in \mathcal{C}$ with $p(S \triangle S') \leq \epsilon$. The ϵ*-covering number* of $\mathcal{F}$ is

$$N(\mathcal{F}, p, \epsilon) := \inf\{|\mathcal{C}| : \mathcal{C} \text{ is an } \epsilon\text{-cover of } \mathcal{F}\}.$$

If $\mathcal{C} \subseteq \mathcal{F}$ is an *ϵ-cover* of $\mathcal{F}$, then $\mathcal{C}$ is an *ϵ-self cover* of $\mathcal{F}$. The *ϵ-self-covering number* of $\mathcal{F}$ is

$$N^s(\mathcal{F}, p, \epsilon) := \inf\{|\mathcal{C}| : \mathcal{C} \text{ is an } \epsilon\text{-self-cover of } \mathcal{F}\}.$$

Clearly, $N^s(\mathcal{F}, p, \epsilon) \geq N(\mathcal{F}, p, \epsilon)$, and we establish the reverse relation.

Lemma 28. *For any $\epsilon \geq 0$, $N^s(\mathcal{F}, p, \epsilon) \leq N(\mathcal{F}, p, \epsilon/2)$.*

Proof. If $N(\mathcal{F}, p, \epsilon/2) = \infty$, the lemma clearly holds. Otherwise, let $\mathcal{C}$ be an $\epsilon/2$-cover of size $N(\mathcal{F}, p, \epsilon/2)$. We construct an ϵ-self-cover of equal or smaller size.

For every subset $S_{\mathcal{C}} \in \mathcal{C}$, there is a subset $S = f(S_{\mathcal{C}}) \in \mathcal{F}$ with $p(S_{\mathcal{C}} \triangle f(S_{\mathcal{C}})) \leq \epsilon/2$. Otherwise, $S_{\mathcal{C}}$ could be removed from $\mathcal{C}$ to obtain a strictly smaller $\epsilon/2$ cover, which is impossible.

The collection $\{f(S_{\mathcal{C}}) : S_{\mathcal{C}} \in \mathcal{C}\} \subseteq \mathcal{F}$ has size $\leq |\mathcal{C}|$, and it is an ϵ-self-cover of $\mathcal{F}$ because for any $S \in \mathcal{F}$, there is an $S_{\mathcal{C}} \in \mathcal{C}$ with $p(S \triangle S_{\mathcal{C}}) \leq \epsilon/2$, and by the triangle inequality, $p\big(S \triangle f(S_{\mathcal{C}})\big) \leq \epsilon$. ■

Let $N_{\mathcal{F},\epsilon} := \sup_p N(\mathcal{F}, p, \epsilon)$ and $N^s_{\mathcal{F},\epsilon} := \sup_p N^s(\mathcal{F}, p, \epsilon)$ be the largest covering numbers under any distribution.

The next theorem bounds the covering number of $\mathcal{F}$ in terms of its VC-dimension.

Theorem 29 ([143])**.** *There exists a universal constant c such that for any $\epsilon > 0$, and any family $\mathcal{F}$ with VC dimension $V_{\mathcal{F}}$,*

$$N_{\mathcal{F},\epsilon} \leq cV_{\mathcal{F}}\Big(\frac{4e}{\epsilon}\Big)^{V_{\mathcal{F}}}.$$

Combining the theorem and Lemma 28, we obtain the following corollary.

Corollary 30. *There exists a universal constant c such that for any $\epsilon > 0$, and any family $\mathcal{F}$ with VC dimension $V_{\mathcal{F}}$,*

$$N^s_{\mathcal{F},\epsilon} \leq cV_{\mathcal{F}}\Big(\frac{8e}{\epsilon}\Big)^{V_{\mathcal{F}}}.$$

The above corollary implies that for any distribution p, a VC class $\mathcal{F}$ has an ϵ self cover, under distribution p, of size $\mathcal{O}\Big(V_{\mathcal{F}}\Big(\frac{8e}{\epsilon}\Big)^{V_{\mathcal{F}}}\Big)$.

3.7 A framework for distribution estimation from corrupted sample batches

We develop a general framework for learning distributions in $\mathcal{F}$ distance, leading to Theorem 17. Building on this framework, we derive a computationally efficient algorithm for learning in $\mathcal{F}_k$ distance, yielding Theorem 18.

Recall that the $\mathcal{F}$ distance between two distributions p and q is

$$||p - q||_{\mathcal{F}} = \sup_{S \in \mathcal{F}} |p(S) - q(S)|.$$

Our goal is to estimate p to $\mathcal{F}$-distance $\mathcal{O}(\Delta)$, where $\Delta = \mathcal{O}\big(\beta\sqrt{\frac{\ln(1/\beta)}{n}}\big)$ is essentially the lower bound.

At a high level, the filtering algorithm removes the adversarial, or "outlier" batches, and returns a sub-collection $B' \subseteq B$ of batches whose empirical distribution $\bar{p}_{B'}$ is close to p in $\mathcal{F}$ distance. The uniform deviation inequality in VC theory states that the sub-collection B_G of good batches has empirical distribution $\bar{p}_{B_G}$ that approximates p in $\mathcal{F}$ distance, thereby ensuring the existence of such a sub-collection when the number of batches m is sufficiently large.

[77] developed a filtering algorithm for learning in TV-distance for a finite domain $\Omega = [k]$. The main drawback of this approach is that applying filtering algorithm directly for Σ-distance requires a number of samples linear in domain size, which is prohibitive for non-finite domains. Here we focus on general domains Ω and any collection of its subsets that has a finite VC-dimension.

Subsection 3.7.1 describes certain filtration properties for a subset of Ω and using the subset that has these filtration properties as a filter. This can be viewed as a reinterpretation of the similar properties used in the filtering algorithm of [77]. Subsection 3.7.2 uses these properties to develop a filtering algorithm for any finite collection of subsets. Subsection 3.7.3 proves a Robust covering theorem to extends the filtering algorithm to VC family of subsets and proves Theorem 17. Subsection 3.7.4 gives a computationally efficient filtering algorithm for the collection of subsets generated by a finite partition of the domain. Building on this, the next subsection 3.7.5 gives an efficient algorithm for learning in $\mathcal{F}_k$ distance and proves Theorem 18.

3.7.1 Using subsets as filters

We discuss how a subset $S \in \Sigma$ can be used as a filter. For this section, we fix a subset $S \in \Sigma$.

We show that if empirical estimates $\bar{\mu}_b(S)$ that batches $b \in B$ assigns to this subset S satisfy certain properties then we can accurately learn its probability and use this subset as a filter. The following discussion develops some notation and intuitions that lead to these properties.

We start with the following observation. For every good batch $b \in B_G$, the empirical estimate $n \cdot \bar{\mu}_b(S)$ has a binomial distribution $\text{Bin}(p(S), n)$, which implies that $\bar{\mu}_b(S)$ has a sub-gaussian distribution $\text{subG}(p(S), \frac{1}{4n})$ with variance $\text{V}(p(S))$. Hence, the empirical mean and variance of $\bar{\mu}_b(S)$ over $b \in B_G$ converges to the expected values $p(S)$ and $\text{V}(p(S))$, respectively. Moreover, sub-gaussian property of the distribution of $\bar{\mu}_b(S)$ implies that, most of the good batches $b \in B_G$ assign the empirical probability $\bar{\mu}_b(S) \in p(S) \pm \tilde{O}(1/\sqrt{n})$.

In addition to the good batches, the collection B of batches also includes an adversarial sub-collection B_A of batches that constitute up to a $\beta-$fraction of B. If the difference between $p(S)$ and the average of $\bar{\mu}_b(S)$ over all adversarial batches $b \in B_A$ is $\le \tilde{O}(\frac{1}{\sqrt{n}})$, namely comparable to the standard deviation of $\bar{\mu}_b(S)$ for the good batches $b \in B_G$, then the adversarial batches can change the overall mean of empirical probabilities $\bar{\mu}_b(S)$ by at most $\tilde{O}(\frac{\beta}{\sqrt{n}})$, which is within our tolerance. Hence, the mean of $\bar{\mu}_b(S)$ will deviate significantly from $p(S)$ only in the presence of a large number of adversarial batches $b \in B_A$ whose empirical probability $\bar{\mu}_b(S)$ differs from $p(S)$ by $\gg \tilde{\Omega}(\frac{1}{\sqrt{n}})$.

To quantify this effect, for a subset $S \in \Sigma$, let

$$\text{med}(\bar{\mu}(S)) := \text{median}\{\bar{\mu}_b(S) : b \in B\}$$

be the median empirical probability of S over all batches. Property 1 (defined later) shows that w.h.p., the absolute difference between $\text{med}(\bar{\mu}(S))$ and $p(S)$ is $\leq \mathcal{O}(1/\sqrt{n})$. The *corruption score* of batch b for S is

$$\psi_b(S) := \begin{cases} 0 & \text{if } |\bar{\mu}_b(S) - \text{med}(\bar{\mu}(S))| \leq \mathcal{O}\Big(\sqrt{\frac{\ln(1/\beta)}{n}}\Big), \\ (\bar{\mu}_b(S) - \text{med}(\bar{\mu}(S)))^2 & \text{otherwise.} \end{cases}$$

The preceding discussion shows that the corruption score of most good batches for the subset S is zero and that adversarial batches that may significantly change the overall mean of empirical probabilities have high corruption score.

The *corruption score* of a sub-collection $B' \subseteq B$ for a subset S is the sum of the *corruption score* of its batches,

$$\psi_{B'}(S) := \sum_{b \in B'} \psi_b(S).$$

A high corruption score of B' for a subset S indicates that B' has many batches b with large difference $|\bar{\mu}_b(S) - \text{med}(\bar{\mu}(S))|$.

Next, we describe some essential properties that allows to a use subset S as a filter. We later show that regardless of the samples in adversarial batches, with high probability, the empirical estimates $\bar{\mu}_b(S)$ for $b \in B$ satisfies the following four *filtration properties*.

1. The median of the estimates $\{\bar{\mu}_b(S) : b \in B\}$ is close to $p(S)$,

$$|\text{med}(\bar{\mu}(S)) - p(S)| \leq \mathcal{O}(1/\sqrt{n}).$$

2. For every sub-collection $B'_G \subseteq B_G$ containing a large portion of the good batches, $|B'_G| \geq (1 - \beta/6)|B_G|$, the empirical mean of $\bar{\mu}_b(S)$ estimate $p(S)$ well,

$$|\bar{p}_{B'_G}(S) - p(S)| \leq \mathcal{O}\Big(\beta\sqrt{\frac{\ln(1/\beta)}{n}}\Big) = \mathcal{O}(\Delta),$$

3. The corruption score of the collection B_G of good batches for subset S is small,

$$\psi_{B'}(S) \leq \kappa_G := \mathcal{O}\Big(\frac{\beta m \ln(1/\beta)}{n}\Big).$$

4. For every sub-collection $B'_G \subseteq B_G$ s.t. $|B'_G| \geq (1-\beta/6)|B_G|$, the empirical variance of $\bar{\mu}_b(S)$ estimate $\mathrm{V}(p(S))$ well,

$$\Big|\frac{1}{|B'_G|}\sum_{b \in B'_G}(\bar{\mu}_b(S) - p(S))^2 - \mathrm{V}(p(S))\Big| \leq \mathcal{O}\Big(\frac{\beta \ln(1/\beta)}{n}\Big).$$

If any of the four filtration properties holds for subset S, we say that S has that particular property.

Next we show how a subset S with the first three of the filtration properties, can be used as a filter. The last filtration property will be used later for deriving computationally efficient algorithms.

For subset S that has filtration properties and for every sub-collection $B' \subseteq B$ that contain most good batches, the next lemma upper bounds the absolute difference between $p(S)$ and the empirical estimate $\bar{p}_{B'}(S)$ of the batches in B' in terms of the corruption score of B'.

Lemma 31. *If subset S has filtration properties 1- 3, then for any B' such that $|B' \cap B_G| \geq (1-\frac{\beta}{6})|B_G|$ such that $\psi_{B'}(S) \leq t \cdot \kappa_G$, for some $t \geq 0$, then*

$$|\bar{p}_{B'}(S) - p(S)| \leq \mathcal{O}\Big((\sqrt{t}+1)\Delta\Big).$$

The lemma is related to Lemma 4 in [77], hence we provide only a high-level argument. For any sub-collection B' retaining a major portion of good batches, from filtration property 2, the mean of $\bar{\mu}_b(S)$ of the good batches $B' \cap B_G$ approximates $p(S)$. Showing that a small corruption score of B' implies that the adversarial batches $B' \cap B_A$ have limited effect on $\bar{p}_{B'}(S)$ proves the lemma.

Next, we describe the *Batch-Deletion* algorithm of [77] and its performance guarantees.

Given a subset S with filtration property 3 and any sub-collection B', the algorithm successively removes batches from B', ensuring that each batch removed is adversarial with high probability. The algorithm stops deleting batches when the corruption score of the remaining sub-collection for S is small.

Algorithm 3. Batch-Deletion

```
**Input:** Sub-Collection $B'$ of Batches, subset $S$, med=med($\bar{\mu}(S)$), and $\kappa_G$
**Output:** A smaller sub-collection $B'$ of batches
**Comment:** The terms $\kappa_G$, $\psi_b(S)$, and $\psi_{B'}(S)$ used below are defined earlier in this section, and computing $\psi_b(S)$ and $\psi_{B'}(S)$ require med($\bar{\mu}(S)$) as input (that depends on all batches $B$).
while $\psi_{B'}(S) \geq 20\kappa_G$ do
    Select a single batch $b \in B'$ where batch $b$ is selected with probability $\frac{\psi_b(S)}{\psi_{B'}(S)}$;
    $B' \leftarrow \{B' \setminus b\}$;
end while
return $(B')$;
```

The next lemma, characterizes the performance of the Batch-Deletion algorithm.

Lemma 32. *Let $B' \subseteq B$ and subset S be the input of the Batch-Deletion algorithm. If subset S has filtration property 3, then:*

1. *Each batch that gets removed from B' by Batch-Deletion algorithm is an adversarial batch with probability ≥ 0.95.*

2. *Batch-Deletion returns updated sub-collection B' such that $\psi_{B'}(S) < 20\kappa_G$.*

Proof. The first statement in the lemma follows as

$$\Pr[\text{Deleting a batch from } B_G \cap B'] = \sum_{b \in B' \cap B_G} \frac{\psi_b(S)}{\psi_{B'}(S)} \leq \frac{\sum_{b \in B_G} \psi_b(S)}{\psi_{B'}(S)} \leq \frac{\kappa_G}{20\kappa_G} \leq 0.05,$$

here we used filtration property 3. The second statement in the Lemma follows from step 4 of Batch-Deletion algorithm. ■

Lemma 31 implies that if a sub-collection B' has most of the good batches and has a small corruption score for subset S, then $\bar{\mu}_b(S)$ is close to $p(S)$.

Lemma 32 implies that if sub-collection B' has large corruption for subset S, then there is a probabilistic method that removes more adversarial batches from B' then good batches and lowers the corruption.

The next subsection builds on these two Lemma and gives a simple filtering algorithm for any finite collection of subsets $C \subseteq \Sigma$ whose subsets $S \in C$ has filtration properties 1-3.

3.7.2 Filtering algorithms for finite collection of subsets

Given any finite collection of subsets $C \subseteq \mathcal{F}'$, algorithm 4, described next, uses the Batch-Deletion algorithm to successively update B and decrease the corruption score for each subset $S \in C$.

Algorithm 4. Filtering Algorithm

```
Input: Collection $B$ of Batches, finite subset family $C \subseteq \Sigma$, adversarial batches fraction $\beta$
Output: A sub-collection $B^*$ of batches.
Comment: The terms $\kappa_G$, $\psi_{B'}(S)$, and med$(\bar{\mu}(S))$ used below are defined earlier in this section
$B' = B$;
for $S \in \mathcal{C}$ do
    if $\psi_{B'}(S) \geq 20\kappa_G$ then
        med $\leftarrow$ med$(\bar{\mu}(S))$;
        $B' \leftarrow$Batch-Deletion$(B', S,$ med$)$;
    end if
end for
$B^* \leftarrow B'$
return $(B^*)$;
```

The next lemma characterizes the algorithm's performance.

Lemma 33. *Let $C \subseteq \Sigma$ be a finite collection of subsets. If all subsets in C have filtration properties 1, 2 and 3, then algorithm 4 returns a sub-collection of batches B^* such that with probability $\geq 1 - e^{-O(\beta m)}$, $|B^* \cap B_G| \geq (1 - \frac{\beta}{6})|B_G|$ and*

$$||p - p_{B^*}||_C = \max_{S \in C} |p(S) - p_{B^*}(S)| \leq \mathcal{O}(\Delta).$$

The proof of the lemma is immediate from Lemmas 31 and 32.

We note that $|B^*| \geq (1 - \frac{\beta}{6})|B_G| \geq (1 - \frac{\beta}{6})(1 - \beta)m > m/2$, as $\beta \in (0, 0.4]$. Therefore, w.h.p. B^* retains at least half of the overall batches.

3.7.3 Robust covering theorem for learning in $\mathcal{F}$ distance and Proof of Theorem 17

A subset family $\mathcal{F}$, with finite VC dimension, can have potentially uncountable subsets, hence, even if all subsets in $\mathcal{F}$ have filtration properties 1-3, we may not be able to use filtering algorithm directly for subset family $\mathcal{F}$. The *Robust covering* theorem proved here overcomes this challenge.

Recall that the collection B includes adversarial batches that can cause the empirical distribution of all batches $\bar{p}_B$ to be at an $\mathcal{F}$-distance $\mathcal{O}(\beta)$ from p.

Yet for any $\epsilon > 0$, any sub-collection $B' \subseteq B$ consisting of at least half of the batches, and for any ϵ-cover $\mathcal{C}$ of $\mathcal{F}$ under the empirical distribution $\bar{p}_B$ of all batches B, the next theorem upper bounds, $||\bar{p}_{B'} - p||_{\mathcal{F}}$, the $\mathcal{F}$-distance between p and the empirical distribution induced by B' in terms of $||\bar{p}_{B'} - p||_{\mathcal{C}}$, the $\mathcal{C}$-distance between them.

Let $\mathcal{G}$ be a VC-class of subsets such that $\mathcal{F} \subseteq \mathcal{G}$. The theorem allows the ϵ-cover $\mathcal{C}$ of $\mathcal{F}$ to include subsets from a larger class of subsets $\mathcal{G}$. Although, one can always choose a cover of $\mathcal{F}$ from within the class, as we will see in later subsections, for computationally efficient algorithms some additional structure in the cover may be desired. And to choose such a cover, we will choose its elements (subsets) from a larger class of subsets than $\mathcal{F}$.

Theorem 34 (Robust covering). *For any $\epsilon > 0$, any subset family $\mathcal{G} \supseteq \mathcal{F}$ with VC dimension $V_{\mathcal{G}}$, and $m \cdot n \geq \mathcal{O}(\frac{V_{\mathcal{G}} \log(1/\epsilon) + \log(1/\delta)}{\epsilon^2})$, let $\mathcal{C} \subseteq \mathcal{G}$ be an ϵ-cover of family $\mathcal{F}$ under the empirical distribution $\bar{p}_B$. With probability $\geq 1 - \delta$, for every sub-collection of batches $B' \subseteq B$ of size $|B'| \geq m/2$,*

$$||\bar{p}_{B'} - p||_{\mathcal{F}} \leq ||\bar{p}_{B'} - p||_{\mathcal{C}} + 5\epsilon.$$

Proof. Consider any batch sub-collection $B' \subseteq B$. For every $S, S' \in \Sigma$, by the triangle inequality,

$$\begin{aligned}
&|\bar{p}_{B'}(S) - p(S)| \qquad (3.4)\\
&= \Big|\Big(\bar{p}_{B'}(S') + \bar{p}_{B'}(S \setminus S') - \bar{p}_{B'}(S' \setminus S)\Big) - \Big(p(S') + p(S \setminus S') - p(S' \setminus S)\Big)\Big|\\
&\le |\bar{p}_{B'}(S') - p(S')| + \bar{p}_{B'}(S \setminus S') + \bar{p}_{B'}(S' \setminus S) + p(S \setminus S') + p(S' \setminus S)\\
&= |\bar{p}_{B'}(S') - p(S')| + \bar{p}_{B'}(S \triangle S') + p(S \triangle S'). \qquad (3.5)
\end{aligned}$$

Since $\mathcal{C}$ is an ϵ-cover under $\bar{p}_B$, for every $S \in \mathcal{F}$ there is an $S' \in \mathcal{C}$ such that $\bar{p}_B(S \triangle S') \le \epsilon$. For such pairs, we bound the second term on the right in the above equation.

$$\begin{aligned}
\bar{p}_{B'}(S \triangle S') &= \frac{1}{|B'|n} \sum_{b \in B'} \sum_{i \in [n]} \mathbf{1}_{S \triangle S'}(X_i^b)\\
&\le \frac{1}{|B'|n} \sum_{b \in B} \sum_{i \in [n]} \mathbf{1}_{S \triangle S'}(X_i^b)\\
&= \frac{|B|}{|B'|} \cdot \frac{1}{|B|n} \sum_{b \in B} \sum_{i \in [n]} \mathbf{1}_{S \triangle S'}(X_i^b)\\
&= \frac{m}{|B'|} \bar{p}_B(S \triangle S') \le \frac{m\epsilon}{|B'|}. \qquad (3.6)
\end{aligned}$$

Choosing $B' = B_G$ in the above equation and using $B_G = (1 - \beta)m \ge m/2$ gives,

$$\bar{p}_{B_G}(S \triangle S') < 2\epsilon. \qquad (3.7)$$

Then

$$\begin{aligned}
p(S \triangle S') &\le |p(S \triangle S') - \bar{p}_{B_G}(S \triangle S')| + \bar{p}_{B_G}(S \triangle S')\\
&\overset{\text{(a)}}{\le} \sup_{S, S' \in \mathcal{G}} |p(S \triangle S') - \bar{p}_{B_G}(S \triangle S')| + 2\epsilon\\
&\overset{\text{(b)}}{\le} \epsilon + 2\epsilon,
\end{aligned}$$

with probability $\geq 1-\delta$, here (a) used the fact that $\mathcal{C}, \mathcal{F} \subseteq \mathcal{G}$ and equation (3.7) and (b) follows from Lemma 40. Combining equations (3.5), (3.6) and the above equation completes the proof. ■

In contrast to the class $\mathcal{F}$, which could be infinite, we can always choose a cover $\mathcal{C}$ of finite size and therefore run filtering algorithm 4 for $C = \mathcal{C}$ to learn in $\mathcal{C}$ distance. Robust covering theorem implies that if $\mathcal{C}$ is ϵ-cover of family $\mathcal{F}$, under distribution $\bar{p}_B$, where $\epsilon = \mathcal{O}(\Delta)$, then for learning in $\mathcal{C}$ distance suffices to learn in $\mathcal{F}$ distance.

The only step that remains is to find a cover whose subsets have filtration properties. The next lemma establishes that every subsets in a given VC-subset family $\mathcal{G}$ has filtration properties.

Lemma 35. *For any given subset family $\mathcal{G}$ with finite VC dimension and the number of batches $m \geq \mathcal{O}(\frac{V_{\mathcal{G}} \log(n/\beta)+\log(1/\delta)}{\beta^2})$. With probability $\geq 1-\delta$, all subsets in $\mathcal{G}$ has filtration properties 1- 4.*

The proof of the lemma appears in section 3.8.

Note that the number of samples required in the lemma increase with the VC-complexity of $\mathcal{G}$. Therefore, to obtain sample optimal algorithm, we choose $\mathcal{G} = \mathcal{F}$, and $\mathcal{C}$ to be any finite ϵ-self-cover of $\mathcal{F}$ under distribution $\bar{p}_B$, where $\epsilon \leq \mathcal{O}(\Delta)$. The existence of such a self-cover is guaranteed by Corollary 30.

The above lemma implies that w.h.p. all subsets in C has filtration property. Therefore, we run algorithm 4 for $C = \mathcal{C}$. Then combining Lemma 33 and robust covering theorem 34 implies learning in $\mathcal{F}$ distance and gives Theorem 17.

Theorem 36 (Theorem 17 restated)**.** *For any $\beta \leq 0.4$, $\delta > 0$, $\mathcal{F}$, and $m \cdot n \geq \mathcal{O}\Big(\frac{V_{\mathcal{F}} \log(1/\Delta)+\log 1/\delta}{\Delta^2} \cdot \log(\frac{1}{\beta})\Big)$, there is a non-constructive algorithm that with probability $\geq 1-\delta$ returns a sub-collection of batches B^* such that $|B^* \cap B_G| \geq (1-\frac{\beta}{6})|B_G|$ and*

$$||p - \bar{p}_{B^*}||_{\mathcal{F}} \leq \mathcal{O}(\Delta).$$

3.7.4 Computationally efficient algorithm for subsets generated by a partition

For estimating p in $\mathcal{F}$-distance, in the previous subsection, we chose C to be a cover of $\mathcal{F}$ and estimated p in C distance. Then to estimate p in C distance, algorithm 4 iterates through all subsets in C one by one, and therefore, has run-time at least linear in the size of the subset family C. But the size of the cover of $\mathcal{F}$ may grow exponentially with the VC-dimension of family $\mathcal{F}$. This makes the algorithm 4 computationally prohibitive even for subset family $\mathcal{F}$ with moderate VC-dimension. Here we show that if subset collection C has a certain structure then this time complexity can be reduced significantly.

For any $\ell > 0$, we consider C which is the collection of all subsets generated by an ℓ-partition of the domain Ω. Here we give a filtering algorithm that has run time only polynomial in ℓ, whereas the size of subset collection C is 2^ℓ.

For any integer $\ell > 0$, let $\xi : \Omega \to [\ell]$ be any function. This function ξ partitions the domain Ω into ℓ disjoint parts. For $j \in [\ell]$, let $\xi_j := \xi^{-1}(j)$ denote the j^{th} *partition element* in the partition created by ξ. Clearly the partition elements ξ_j's are disjoint and their union is Ω. We refer to ξ as *partition function*. Note that a partition function ξ is uniquely determined by the corresponding partition elements ξ_j's.

For a subset $D \subseteq [\ell]$, let

$$S_D^\xi := \cup_{j\in D}\xi_j,$$

be the union of the partition elements ξ_j's corresponding to the elements of D. Define the collection of subsets

$$C^\xi := \{S_D^\xi : D \in 2^{[\ell]}\}$$

to be the family of all possible unions of ξ_j's. Clearly, $|C^\xi| = 2^\ell$.

We show that if all subsets $S \in C^\xi$ have filtration properties 1- 4, then p can be estimated to a small C^ξ-distance in time polynomial in ℓ rather than exponential.

For finite domain $\Omega' = [\ell]$, [77] derived a method that for any batch sub-collection B', containing a majority of good batches, can find a subset in $2^{[\ell]}$ for which the corruption score of B' is within a constant times the maximum in time only polynomial in the domain size ℓ, when all subsets in $2^{[\ell]}$ have filtration properties 1- 4. Then instead of iterating over all 2^ℓ subsets, as in algorithm 4, they find the subsets with high corruption score efficiently and use the Batch Deletion procedure for these subsets. This leads to a computationally efficient algorithm for learning discrete distributions p.

To obtain a computationally efficient algorithm for learning in C^ξ distance, we first reduce this problem to that of robustly learning distributions over finite domains in total variation distance and then use the algorithm in [77].

Theorem 37. *Let $\xi : \Omega \to [\ell]$ be any partition function and let C^ξ be the collection of all possible unions of the partition elements ξ_j's. If all subsets in C^ξ have filtration properties 1- 4, then there is an algorithm that runs in time polynomial in all parameters ℓ, m, and n, and with probability $\geq 1 - e^{-O(\beta m)}$ returns a sub-collection of batches $B^* \subseteq B$ such that $|B^* \cap B_G| \geq (1-\beta/6)|B_G|$ and*

$$||p - \bar{p}_{B^*}||_{C^\xi} \leq \mathcal{O}(\Delta).$$

Proof. First note that ξ transforms any distribution q over Ω to the discrete distribution q^ξ over $\Omega' = [\ell]$, where $q^\xi(j) := q(\xi_j)$ for each $j \in [\ell]$. Recall that any subset $D \subseteq [\ell]$, corresponds one to one with a subset $S^\xi_D = \cup_{j \in D} \xi_j$ in C^ξ. It follows that for any distribution q over Ω, and $D \subseteq [\ell]$,

$$q(S^\xi_D) = q^\xi(D).$$

Recall that $\bar{p}_{B'}$ denotes the empirical distribution induced by a sub-collection B', therefore $\bar{p}^\xi_{B'}$ denotes the empirical distribution induced by a sub-collection B' over the transformed domain $[\ell]$.

From the one-to-one correspondence between subsets in C^ξ and subsets in $2^{[\ell]}$ it follows that all subsets in C^ξ have filtration properties iff all subsets in $2^{[\ell]}$ have filtration properties for the transformed distributions p^ξ and transformed empirical distribution of the sample batches.

Theorem 9 in [77] implies that, if all subsets in $2^{[\ell]}$ have filtration properties 1- 4 then algorithm 2 therein runs in time polynomial in the domain size ℓ, the number of batches m, and the batch-size n, and with probability $\geq 1 - e^{-O(\beta m)}$ returns a sub-collection of batches $B^* \subseteq B$ such that $|B^* \cap B_G| \geq (1 - \beta/6)|B_G|$ and

$$||p^\xi - \bar{p}^\xi_{B^*}||_{TV} \leq \mathcal{O}(\Delta).$$

Next, for any pair of distributions q_1 and q_2 over the domain Ω, we show that C^ξ-distance between them is the same as the total variation distance between q_1^ξ and q_2^ξ. For every distribution pair q_1, q_2 over Ω,

$$\begin{aligned}
||q_1 - q_2||_{C^\xi} &= \max_{S \in C^\xi} |q_1(S) - q_2(S)| \\
&= \max_{S^\xi_D \in C^\xi} |q_1(S^\xi_D) - q_2(S^\xi_D)| \\
&= \max_{D \in 2^{[\ell]}} |q_1^\xi(D) - q_2^\xi(D)| \\
&= ||q_1^\xi - q_2^\xi||_{TV}.
\end{aligned}$$

Therefore,

$$||p - \bar{p}_{B^*}||_{C^\xi} = ||p^\xi - \bar{p}^\xi_{B^*}||_{TV} \leq \mathcal{O}(\Delta). \qquad \blacksquare$$

3.7.5 Computationally efficient algorithm for learning in $\mathcal{F}_k$ distance and proof of Theorem 18

Recall that $\mathcal{F}_k$ is the collection of all unions of at most k intervals over $\mathbb{R}$.

In the previous subsection we showed that for a partition function ξ, we can learn in C^ξ-distance efficiently. To obtain a computationally efficient algorithm for learning in $\mathcal{F}_k$ distance, we give a partition function $\xi^* : \mathbb{R} \to [\ell]$, for an appropriate ℓ to be chosen later, such that the collection of subsets C^{ξ^*} forms an ϵ-cover of $\mathcal{F}_k$ under the empirical distribution $\bar{p}_B$.

Recall that B is a collection of m batches and each batch has n samples. Let $s = n \cdot m$ and let $x^s = x_1, x_2, \ldots, x_s \in \mathbb{R}$ be the samples of B arranged in non-decreasing order. And recall that the points x^s induce an empirical measure $\bar{p}_B$ over $\mathbb{R}$, where for $S \subseteq \mathbb{R}$,

$$\bar{p}_B(S) = |\{i : x_i \in S\}|/s.$$

Let $t := \frac{s}{\ell}$, and for simplicity assume that it is an integer. Recall that a partition function ξ is uniquely determined by the corresponding partition elements ξ_j's. Let $\xi^* : \mathbb{R} \to [\ell]$ be the partition function with partition elements $\{\xi_1^*, \ldots, \xi_\ell^*\}$ of $\mathbb{R}$, where

$$\xi_j^* := \begin{cases} (-\infty, x_t] & j = 1, \\ (x_{(j-1)t}, x_{jt}] & 2 \le j < \ell, \\ (x_{s-t}, \infty) & j = \ell. \end{cases}$$

Note that all elements of the partition $\{\xi_1^*, \ldots, \xi_\ell^*\}$ are intervals of $\mathbb{R}$. Recall that C^{ξ^*} is is formed by all possible unions of these ℓ intervals. Clearly $C^{\xi^*} \subseteq \mathcal{F}_\ell$, as $\mathcal{F}_\ell$ contains all unions of ℓ intervals over $\mathbb{R}$.

We show that C^{ξ^*} is an $2k/\ell-$cover of $\mathcal{F}_k$ under the empirical distribution $\bar{p}_B$ of points x_1^s.

Lemma 38. *For any k, and ℓ, C^{ξ^*} is a $\frac{2k}{\ell}$-cover of $\mathcal{F}_k$ under $\bar{p}_B$.*

Proof. Any set $S \in \mathcal{F}_k$ is a union of k real intervals $I_1 \cup I_2 \cup \ldots \cup I_k$. Let $S^* \subseteq \mathbb{R}$ be the union of all ξ_j^*-partition elements (intervals) that are fully contained in one of the intervals $I_1, \ldots, I_k$. By definition, $S^* \in C^\xi$, and we show that $\bar{p}_B(S \triangle S^*) \le 2k/\ell$. By construction, $S^* \subseteq S$, hence,

$$\bar{p}_B(S \triangle S^*) = \bar{p}_B(S \setminus S^*) = \sum_{j=1}^{k} \bar{p}_B(I_j \setminus S^*) = \sum_{j=1}^{k} \frac{|\{x_i \in I_j \setminus S^*\}|}{s} \le \sum_{j=1}^{k} 2 \cdot \frac{t}{s} = \frac{2k}{\ell},$$

where the inequality follows as each $I_j \setminus S^*$ contains at most t points and the left and right. ■

Next choose $\ell = \frac{2k}{\epsilon}$ then the lemma implies that the corresponding C^{ξ^*} is an ϵ-cover of $\mathcal{F}_k$ under $\bar{p}_B$. As discussed earlier $C^{\xi^*} \subseteq \mathcal{F}_\ell$. Then choosing $\mathcal{G} = \mathcal{F}_\ell$ in Lemma 35 implies that $w.h.p.$ all subsets in C^{ξ^*} has filtering properties. Then combining Theorem 37 and robust covering theorem 34, and choosing $\epsilon = \mathcal{O}(\Delta)$, we get the following theorem that implies learning in $\mathcal{F}_k$ distance.

We note that this computationally efficient algorithm uses $\mathcal{O}(1/\Delta)$ times more sample than information theoretic algorithm in section 3.7.3, because here we chose the cover of $\mathcal{F}_k$ from the class $\mathcal{G} = \mathcal{F}_{k/\Delta}$. And $\mathcal{F}_{k/\Delta}$ has VC dimension $\mathcal{O}(k/\Delta)$, which is $\mathcal{O}(1/\Delta)$ times the VC-dimension of the class $\mathcal{F}_k$.

Theorem 39 (Theorem 18 restated)**.** *For any given $\beta \leq 0.4$, $\delta > 0$, $k > 0$, and $m \cdot n \geq \mathcal{O}\Big(\frac{k\log(1/\Delta)+\log 1/\delta}{\Delta^3} \cdot \log(\frac{1}{\beta})\Big)$, there is an algorithm that runs in time polynomial in all parameters, and with probability $\geq 1-\delta$ returns a sub-collection of batches B^* such that $|B^* \cap B_G| \geq (1-\frac{\beta}{6})|B_G|$ and*

$$||\bar{p}_{B^*} - p||_{\mathcal{F}_k} \leq \mathcal{O}(\Delta).$$

3.8 Properties of the Collection of Good Batches

Lemma 40. *Let $\mathcal{G}$ be a VC family of subsets of Ω. Then for any $\delta > 0$ and $|B_G| \cdot n \geq \mathcal{O}(\frac{V_{\mathcal{G}}\log(1/\epsilon)+\log(1/\delta)}{\epsilon^2})$, with probability $\geq 1-\delta$,*

$$\sup_{S,S' \in \mathcal{G}} \max\left\{\frac{\bar{p}_{B_G}(S\triangle S') - p(S\triangle S')}{\sqrt{\bar{p}_{B_G}(S\triangle S')}}, \frac{p(S\triangle S') - \bar{p}_{B_G}(S\triangle S')}{\sqrt{p(S\triangle S')}}\right\} \leq \epsilon.$$

Proof. Consider the collection of symmetric differences of subsets in $\mathcal{G}$,

$$\mathcal{G}_\triangle := \{S\triangle S' : S, S' \in \mathcal{G}\}.$$

The next auxiliary lemma bounds the shatter coefficient of $\mathcal{G}_\triangle$.

Lemma 41. *For $t \geq V_{\mathcal{G}}$, $S_{\mathcal{G}_\triangle}(t) \leq \big(\frac{t\,e}{V_{\mathcal{G}}}\big)^{2V_{\mathcal{G}}}$.*

Proof. For $t \geq V_{\mathcal{G}}$ and $x_1, x_2, .., x_t \in \Omega$, let

$$\mathcal{G}(x_1^t) = \{\{x_1, x_2, .., x_t\} \cap S : S \in \mathcal{G}\}.$$

Note that $S_{\mathcal{G}}(t) = \max_{x_1,\dots,x_t} |\mathcal{G}(x_1^t)|$.

From the definition of shatter coefficient $|\mathcal{G}(x_1^t)| \leq S_{\mathcal{G}}(t)$. Then

$$|\mathcal{G}_{\triangle}(x_1^t)| = |\{\{x_1, \dots, x_t\} \triangle \{x'_1, \dots, x'_t\} : S, S' \in \mathcal{G}(x_1^t)\}| \leq (S_{\mathcal{G}}(t))^2 \leq \big(\frac{t\,e}{V_{\mathcal{G}}}\big)^{2V_{\mathcal{G}}}. \qquad \blacksquare$$

Applying Theorem 27 for family of subsets $\mathcal{G}_{\triangle}$, and using Lemma 41, for $|B_G| \cdot n \geq \mathcal{O}(\frac{V_{\mathcal{G}} \log(1/\epsilon) + \log(1/\delta)}{\epsilon^2})$, with probability $\geq 1 - \delta$,

$$\sup_{S \in \mathcal{G}_{\triangle}} \max \Big\{ \frac{\bar{p}_{B_G}(S) - p(S)}{\sqrt{\bar{p}_{B_G}(S)}}, \sup_{S \in \mathcal{G}} \frac{p(S) - \bar{p}_{B_G}(S)}{\sqrt{p(S)}} \Big\} \leq \epsilon. \qquad \blacksquare$$

3.8.1 Proof of Lemma 35

First we list some auxiliary properties for a subset S, each of which is either one of the filtration property or helps in deriving one of the filtration property.

(i) For every $B'_G \subseteq B_G$, such that $|B'_G| \geq (1 - \beta/6)|B_G|$

$$|\bar{p}_{B'_G}(S) - p(S)| \leq \mathcal{O}\left(\beta \sqrt{\frac{\ln(1/\beta)}{n}}\right).$$

(ii) For every $B'_G \subseteq B_G$, such that $|B'_G| \geq (1 - \beta/6)|B_G|$

$$\Big| \frac{1}{|B'_G|} \sum_{b \in B'_G} (\bar{\mu}_b(S) - p(S))^2 - \mathrm{V}(p(S')) \Big| \leq \mathcal{O}\left(\frac{\beta \ln(\frac{1}{\beta})}{n}\right).$$

(iii)

$$\left|\left\{b \in B_G : |\bar{\mu}_b(S) - p(S)| \geq \mathcal{O}\left(\sqrt{\frac{\ln(1/\beta)}{n}}\right)\right\}\right| \leq \mathcal{O}(\beta) \cdot |B_G|.$$

(iv)

$$\left|\left\{b \in B_G : |\bar{\mu}_b(S) - p(S)| \geq \mathcal{O}\left(\frac{1}{\sqrt{n}}\right)\right\}\right| \leq \mathcal{O}(1) \cdot |B_G|.$$

(v) For every $B'_G \subseteq B_G$, such that $|B'_G| \leq \mathcal{O}(\beta) \cdot |B_G|$

$$\sum_{b \in B'_G} (\bar{\mu}_b(S) - p(S))^2 < \mathcal{O}\left(\beta |B_G| \frac{\ln(1/\beta)}{n}\right),$$

The next lemma shows that these properties hold for a fix subset S.

Lemma 42. *For any given subset $S \in \Sigma$ and for $|B_G| \geq O(\frac{\log 1/\delta}{\beta^2 \ln(1/\beta)})$. With probability $\geq 1 - \delta$, subset S has all auxiliary properties* ***(i)–(v)****. Further, if these auxiliary properties hold for subset S then subset S has filtration properties 1- 4.*

The above Lemma, though not stated explicitly, is implied by Section A.1 and Section A.2 in [77]. In particular, the auxiliary properties **(i)** and **(ii)** are implied by Lemma 11, **(iii)** and **(iv)** are implied by Lemma 10, and **(v)** is implied by Lemma 12, and section A.2 therein showed that these auxiliary properties imply filtration properties 1- 4. Hence, we use the lemma without proving it again here.

Therefore, to prove Lemma 35, it suffices to show these auxiliary properties for subsets in $\mathcal{G}$.

The next Lemma extends the auxiliary properties to all subsets in given a VC class $\mathcal{G}$.

Lemma 43. *For any given subset family $\mathcal{G}$ with finite VC-dimension and $|B_G| \geq$*

$O(\frac{V_{\mathcal{G}}\log(n/\beta)+\log 1/\delta}{\beta^2})$. *With probability* $\geq 1-\delta$*, all subsets in* $\mathcal{G}$ *has all auxiliary properties* ***(i)**–**(v)**.*

Proof. From Corollary 30, there exist a self ϵ-cover $\mathcal{C}^*$ of $\mathcal{G}$ under the distribution p of size $\mathcal{O}\big(V_{\mathcal{G}}(\frac{8e}{\epsilon})^{V_{\mathcal{G}}}\big)$. For this section, fix $\epsilon = \mathcal{O}(\frac{\beta^2}{n})$.

For any $S \in C^*$, for $|B_G| \geq O\Bigg(\frac{log\frac{2|\mathcal{C}^*|}{\delta}}{\beta^2\ln(1/\beta)}\Bigg) = O(\frac{V_{\mathcal{G}}\log(n/\beta)+\log 1/\delta}{\beta^2\ln(1/\beta)})$, Lemma 42 implies that the auxiliary properties **(i)**–**(v)** with probability $\geq 1-\frac{\delta}{2|\mathcal{C}^*|}$.

Therefore, taking the union bound over the complement, the auxiliary properties **(i)**–**(v)** hold for all subsets in $\mathcal{C}^*$ with probability $\geq 1-\frac{\delta}{2}$.

Next, we extend these properties for all subsets in $\mathcal{G}$.

For subset $S \in \mathcal{G}$ choose $S' \in \mathcal{C}^*$ such that $p(S\triangle S') \leq \epsilon$. Existence of such a subset $S' \in \mathcal{C}^*$ is guaranteed for all $S \in \mathcal{G}$ as $\mathcal{C}^*$ is an $\epsilon-$cover under p. The properties for S' holds, since it is a part of the cover $\mathcal{C}'$. To extend the auxiliary properties to all subsets in $\mathcal{G}$, we show that if the properties hold for S', then they also hold for subset S.

Note that for any subset $S, S' \in \mathcal{G}$ with $p(S\triangle S') \leq \mathcal{O}(\frac{\beta^2}{n}) = \mathcal{O}(\epsilon)$.

For $|B_G|\cdot n \geq O(\frac{V_{\mathcal{G}}\log(n/\beta)+\log 1/\delta}{\beta^2}\cdot n)$, Lemma 40 implies that with probability $\geq 1-\delta/2$

$$\bar{p}_{B_G}(S\triangle S') \leq \mathcal{O}(\frac{\beta^2}{n}) = \mathcal{O}(\epsilon). \tag{3.8}$$

For any batch $b \in B$

$$\begin{aligned}\bar{\mu}_b(S) - p(S) &= \Big(\bar{\mu}_b(S') + \bar{\mu}_b(S\setminus S') - \bar{\mu}_b(S'\setminus S)\Big) - \Big(p(S') + p(S\setminus S') - p(S'\setminus S)\Big)\\ &= \Big(\bar{\mu}_b(S') - p(S')\Big) + \Big(\bar{\mu}_b(S\setminus S') - \bar{\mu}_b(S'\setminus S)\Big) - \Big(p(S\setminus S') - p(S'\setminus S)\Big).\end{aligned}$$

From the above equation, we get

$$\begin{aligned}\Big|\Big(\bar{\mu}_b(S)-p(S)\Big)-\Big(\bar{\mu}_b(S')-p(S')\Big)\Big| &\le \bar{\mu}_b(S\setminus S')+\bar{\mu}_b(S'\setminus S)+p(S\setminus S')+p(S'\setminus S)\\ &= \bar{\mu}_b(S\triangle S')+p(S\triangle S')\\ &\le \bar{\mu}_b(S\triangle S')+\mathcal{O}(\epsilon). \qquad (3.9)\end{aligned}$$

Next, we extend property **(i)** to subset S.

$$\begin{aligned}|\bar{p}_{B'_G}(S)-p(S)| &= \Big|\frac{1}{|B'_G|}\sum_{b\in B'_G}\bar{\mu}_b(S)-p(S)\Big| = \Big|\frac{1}{|B'_G|}\sum_{b\in B'_G}\Big(\bar{\mu}_b(S)-p(S)\Big)\Big|\\ &\overset{\text{(a)}}{\le} \Big|\frac{1}{|B'_G|}\sum_{b\in B'_G}\Big(\bar{\mu}_b(S')-p(S')\Big)\Big| + \Big|\frac{1}{|B'_G|}\sum_{b\in B'_G}\Big(\bar{\mu}_b(S\triangle S')+\mathcal{O}(\epsilon)\Big)\Big|\\ &\le \Big|\frac{1}{|B'_G|}\sum_{b\in B'_G}\bar{\mu}_b(S')-p(S')\Big| + \Big|\frac{1}{|B'_G|}\sum_{b\in B_G}\bar{\mu}_b(S\triangle S')\Big| + \mathcal{O}(\epsilon)\\ &\le |\bar{p}_{B'_G}(S')-p(S')| + \frac{|B_G|}{|B'_G|}\bar{p}_{B_G}(S\triangle S')+\mathcal{O}(\epsilon)\\ &\overset{\text{(b)}}{\le} \mathcal{O}\left(\beta\sqrt{\frac{\ln(1/\beta)}{n}}\right)+\frac{1}{(1-\beta/6)}\cdot\mathcal{O}(\epsilon)+\mathcal{O}(\epsilon)\\ &\le \mathcal{O}\left(\beta\sqrt{\frac{\ln(1/\beta)}{n}}\right),\end{aligned}$$

here (a) uses (3.9) and (b) uses that the property **(i)** holds for S'.

Next, we extend property **(i)** to subset S. From equation (3.9) we get

$$\begin{aligned}&(\bar{\mu}_b(S)-p(S))^2 \le \Big(|\bar{\mu}_b(S')-p(S')|+(\bar{\mu}_b(S\triangle S')+\mathcal{O}(\epsilon))\Big)^2\\ &= (\bar{\mu}_b(S')-p(S'))^2+2|\bar{\mu}_b(S')-p(S')|(\bar{\mu}_b(S\triangle S')+\mathcal{O}(\epsilon))+(\bar{\mu}_b(S\triangle S')+\mathcal{O}(\epsilon))^2.\end{aligned}$$

Therefore,

$$
\begin{aligned}
&\sum_{b\in B'_G} (\bar{\mu}_b(S) - p(S))^2 - \sum_{b\in B'_G} (\bar{\mu}_b(S') - p(S'))^2 \\
&\leq \sum_{b\in B'_G} 2|\bar{\mu}_b(S') - p(S')|(\bar{\mu}_b(S\triangle S') + \mathcal{O}(\epsilon)) + \sum_{b\in B'_G} (\bar{\mu}_b(S\triangle S') + \mathcal{O}(\epsilon))^2 \\
&\leq 2\sqrt{\sum_{b\in B'_G} (\bar{\mu}_b(S') - p(S'))^2}\sqrt{\sum_{b\in B'_G} (\bar{\mu}_b(S\triangle S') + \mathcal{O}(\epsilon))^2} + \sum_{b\in B'_G} (\bar{\mu}_b(S\triangle S') + \mathcal{O}(\epsilon))^2,
\end{aligned}
$$

here the last inequality follows from Cauchy-Schwarz inequality. Next, we bound the last terms in the above expression.

$$
\begin{aligned}
\sum_{b\in B'_G} (\bar{\mu}_b(S\triangle S') + \mathcal{O}(\epsilon))^2 &\leq \sum_{b\in B'_G} (\bar{\mu}_b(S\triangle S') + \mathcal{O}(\epsilon))(1 + \mathcal{O}(\epsilon)) \\
&\leq 2 \cdot \sum_{b\in B'_G} (\bar{\mu}_b(S\triangle S') + \mathcal{O}(\epsilon)) \\
&\leq 2 \cdot \left(|B'_G|\mathcal{O}(\epsilon) + \sum_{b\in B_G} (\bar{\mu}_b(S\triangle S')\right) \\
&\leq 2|B'_G| \left(\mathcal{O}(\epsilon) + \frac{|B_G|}{|B'_G|}\bar{p}_{B_G}(S\triangle S')\right) \\
&\leq |B'_G|\mathcal{O}(\epsilon).
\end{aligned}
$$

Also, from the property **(ii)** for S' implies

$$
\begin{aligned}
\sum_{b\in B'_G} (\bar{\mu}_b(S') - p(S'))^2 &\leq |B'_G|\mathrm{V}(p(S')) + |B'_G|\mathcal{O}\left(\frac{\beta \ln(\frac{1}{\beta})}{n}\right) \\
&\leq |B'_G|\mathcal{O}\left(\frac{1}{n}\right),
\end{aligned}
$$

here we used equation (3.3), that implies $V(\cdot) \leq 1/4n$, and $\beta \ln(1/\beta) = \mathcal{O}(1)$. Combining the above three equations we get

$$\begin{aligned}
&\sum_{b\in B'_G} (\bar{\mu}_b(S) - p(S))^2 - \sum_{b\in B'_G} (\bar{\mu}_b(S') - p(S'))^2 \\
&\le 2\sqrt{|B'_G|\mathcal{O}\left(\frac{1}{n}\right)}\sqrt{|B'_G|\mathcal{O}(\epsilon)} + |B'_G|\mathcal{O}(\epsilon) < |B'_G|\mathcal{O}\left(\sqrt{\frac{\epsilon}{n}}\right).
\end{aligned}$$

Similarly, one can prove the other direction

$$\sum_{b\in B'_G} (\bar{\mu}_b(S') - p(S'))^2 - \sum_{b\in B'_G} (\bar{\mu}_b(S) - p(S))^2 < |B'_G|\mathcal{O}\left(\sqrt{\frac{\epsilon}{n}}\right).$$

Combining the two equations gives

$$\Big|\sum_{b\in B'_G} (\bar{\mu}_b(S) - p(S))^2 - \sum_{b\in B'_G} (\bar{\mu}_b(S') - p(S'))^2\Big| < |B'_G|\mathcal{O}\left(\sqrt{\frac{\epsilon}{n}}\right).$$

And from equation (3.3) we get

$$|\mathrm{V}(p(S)) - \mathrm{V}(p(S'))| \le \frac{|p(S) - p(S')|}{n} \le \frac{|p(S\triangle S')|}{n} \le \mathcal{O}\left(\frac{\epsilon}{n}\right).$$

Combining the above two equations we get

$$\begin{aligned}
&\Big|\frac{1}{|B'_G|}\sum_{b\in B'_G} (\bar{\mu}_b(S) - p(S))^2 - \mathrm{V}(p(S))\Big| \\
&\le \Big|\frac{1}{|B'_G|}\sum_{b\in B'_G} (\bar{\mu}_b(S') - p(S'))^2 - \mathrm{V}(p(S'))\Big| + \mathcal{O}\left(\sqrt{\frac{\epsilon}{n}}\right) + \mathcal{O}\left(\frac{\epsilon}{n}\right) \\
&\overset{\text{(a)}}{\le} \mathcal{O}\left(\frac{\beta\ln(\frac{1}{\beta})}{n}\right) + \mathcal{O}\left(\sqrt{\frac{\epsilon}{n}}\right) + \mathcal{O}\left(\frac{\epsilon}{n}\right) \\
&\overset{\text{(b)}}{\le} \mathcal{O}\left(\frac{\beta\ln(\frac{1}{\beta})}{n}\right), \qquad (3.10)
\end{aligned}$$

here inequality (a) uses that the property **(ii)** holds for S', (b) uses $\epsilon = \mathcal{O}\Big(\frac{\beta^2}{n}\Big)$.

This completes the proof of the extension of property **(ii)** to subset S and in a similar fashion property **(v)** can be extended.

Next, we extend property **(iii)** to subset S.

Note that

$$
\begin{aligned}
&\left|\left\{b \in B_G : |\bar{\mu}_b(S) - p(S)| \geq t\right\}\right| \\
&\overset{\text{(a)}}{\leq} \left|\left\{b \in B_G : |\bar{\mu}_b(S') - p(S')| + \bar{\mu}_b(S \triangle S') + \mathcal{O}(\epsilon) \geq t\right\}\right| \\
&\leq \left|\left\{b \in B_G : |\bar{\mu}_b(S') - p(S')| \geq \frac{2}{3} \cdot t\right\}\right| + \left|\left\{b \in B_G : \bar{\mu}_b(S \triangle S') \geq \frac{t}{3} - \mathcal{O}(\epsilon)\right\}\right| \\
&\leq \left|\left\{b \in B_G : |\bar{\mu}_b(S') - p(S')| \geq \frac{2}{3} \cdot t\right\}\right| + \frac{\sum_{b \in B_G} \bar{\mu}_b(S \triangle S')}{\frac{t}{3} - \mathcal{O}(\epsilon)} \\
&\leq \left|\left\{b \in B_G : |\bar{\mu}_b(S') - p(S')| \geq \frac{2}{3} \cdot t\right\}\right| + |B_G| \frac{\bar{p}_{B_G}(S \triangle S')}{\frac{t}{3} - \mathcal{O}(\epsilon)} \\
&\leq \left|\left\{b \in B_G : |\bar{\mu}_b(S') - p(S')| \geq \frac{2}{3} \cdot t\right\}\right| + |B_G| \frac{\mathcal{O}(\epsilon)}{\frac{t}{3} - \mathcal{O}(\epsilon)}.
\end{aligned}
$$

here inequality (a) uses (3.9).

Choosing $t = \mathcal{O}\left(\sqrt{\frac{\ln(1/\beta)}{n}}\right)$ in the above equation and putting $\epsilon = \mathcal{O}(\beta^2/n)$ gives

$$
\begin{aligned}
&\left|\left\{b \in B_G : |\bar{\mu}_b(S) - p(S)| \geq \mathcal{O}\left(\sqrt{\frac{\ln(1/\beta)}{n}}\right)\right\}\right| \\
&\leq \left|\left\{b \in B_G : |\bar{\mu}_b(S') - p(S')| \geq \mathcal{O}\left(\sqrt{\frac{\ln(1/\beta)}{n}}\right)\right\}\right| + |B_G| \frac{\mathcal{O}(\beta^2/n)}{\mathcal{O}\left(\sqrt{\ln(1/\beta)/n}\right) - \mathcal{O}(\beta^2/n)} \\
&\leq \mathcal{O}(\beta)|B_G|. \qquad (3.11)
\end{aligned}
$$

here the last step uses property **(ii)** for S'. This extends property **(iii)** to subset S. Property **(iv)** can be extended similarly. ■

Proof of Lemma 35. The previous lemma showed that the auxiliary properties hold for all subsets in $\mathcal{G}$. Lemma 42 showed that these auxiliary properties implies the filtration properties. Combining the two Lemmas completes the proof of Lemma 35.

3.9 Remaining proofs

3.9.1 Proof of Theorem 21

To prove the above theorem we use the following result.

Theorem 44 ([1]). *There is an algorithm which, given any t samples $x_1, x_2, ..., x_s \in \mathbb{R}$, returns an t-piecewise degree-d polynomial p' which minimizes $||p' - \bar{p}_s||_{\mathcal{F}_{2td}}$ distance between p' and the empirical distribution $\bar{p}_s$, to within additive error γ in time poly$(s, t, d, 1/\gamma)$.*

We note that the t-piecewise degree-d polynomial p' returned in the above theorem may not always integrate to 1 and is only an approximate Yatracos minimizer, and hence we can not directly use equation (3.1).

But there is a simple generalization of this equation in [43], which applies even when p' returned in the above theorem doesn't integrate to 1 and is only an approximate Yatracos minimizer.

Recall that $\mathcal{Y}(\mathcal{P})$ is Yatracos class of $\mathcal{P}$. Let $p' \in \mathcal{P}$ be such that $||p' - \bar{p}||_{\mathcal{Y}(\mathcal{P})} = \min_{q \in \mathcal{P}} ||q - \bar{p}||_{\mathcal{Y}(\mathcal{P})} + \gamma$ Then [43] (exercise 6.2) implies that

$$||p - p'||_{TV} \leq 5 \cdot \text{opt}_{\mathcal{P}}(p) + 4||p - \bar{p}||_{\mathcal{Y}(\mathcal{P})} + 5\gamma.$$

Recall that Yatracos class of t-piecewise degree d polynomials, (including those that don't integrate to 1), is $\mathcal{F}_{2td}$.

Theorem 18 provides a polynomial time algorithm that returns a sub-collection $B^* \subseteq B$ of batches whose empirical distribution $\bar{p}_{B^*}$ is close to p in $\mathcal{F}_{2td}$-distance. Then running the algorithm in Theorem 44 for samples in $\bar{p}_{B^*}$ returns a t-piecewise degree-d polynomial p^*. Then the above equation implies that p^* approximates p in TV distance, to complete the proof of the theorem.

3.9.2 Proof of Lemma 22

Proof. For two distributions p and q over $\Omega \times \{0,1\}$, the largest difference between the loss of any classifier $h \in \mathcal{H}$ is related to their $\mathcal{F}_{\mathcal{H}}$-distance,

$$\begin{aligned}
&\sup_{h\in\mathcal{H}} |r_p(h) - r_q(h)| \\
&= \sup_{h\in\mathcal{H}} |\Pr_{(X,Y)\sim p}[h(X) \neq Y] - \Pr_{(X,Y)\sim q}[h(X) \neq Y]| \\
&\leq \sup_{h\in\mathcal{H}} \sum_{y\in\{0,1\}} |\Pr_{(X,Y)\sim p}(h(X) = \bar{y}, Y = y) - \Pr_{(X,Y)\sim q}(h(X) = \bar{y}, Y = y)| \\
&\leq 2||p - q||_{\mathcal{F}_{\mathcal{H}}}. \qquad (3.12)
\end{aligned}$$

Then,

$$\begin{aligned}
&r_p(h^{\text{opt}}(q)) - r_p^{\text{opt}}(\mathcal{H}) \\
&= r_p(h^{\text{opt}}(q)) - r_p(h^{\text{opt}}(p)) \\
&= r_p(h^{\text{opt}}(q)) - r_q(h^{\text{opt}}(q)) + r_q(h^{\text{opt}}(q)) - r_q(h^{\text{opt}}(p)) + r_q(h^{\text{opt}}(p)) - r_p(h^{\text{opt}}(p)) \\
&\leq r_q(h^{\text{opt}}(q)) - r_q(h^{\text{opt}}(p)) + 2\sup_{h\in\mathcal{H}} |r_q(h) - r_p(h)| \\
&\leq 2\sup_{h\in\mathcal{H}} |r_q(h) - r_p(h)| \\
&\leq 4||p - q||_{\mathcal{F}_{\mathcal{H}}},
\end{aligned}$$

here the last inequality uses (3.12). ■

3.9.3 Proof of Theorem 24

Proof. Let $\mathcal{H} : \Omega \to \{0,1\}$ of Boolean functions with VC dimension $\mathcal{V}_{\mathcal{H}} \geq 1$. And let $(X,Y) \sim p$, where $X \in \Omega$ and $Y \in \{0,1\}$.

Since $\mathcal{V}_{\mathcal{H}} \geq 1$, then there is at-least one $\omega^* \in \Omega$ and $h_1, h_2 \in \mathcal{H}$, s.t. $h_1(\omega^*) \neq h_2(\omega^*)$, w.l.o.g., let $h_1(\omega^*) = 1$ and $h_2(\omega^*) = 0$.

Next, we define two distributions p_1 and p_2. Let $\gamma = c\frac{\beta}{\sqrt{n}}$, for some small enough constant $c > 0$ to be chosen later. Let $p_1(\omega^*, 1) = p_2(\omega^*, 0) = \frac{1}{2} + \gamma$, and $p_1(\omega^*, 0) = p_2(\omega^*, 1) = \frac{1}{2} - \gamma$. Both p_1 and p_2 assigns zero probability to all other points in $\Omega \times \{0, 1\}$.

It is easy to see that, for distribution p_1, hypothesis h_1 achieves the optimal loss $\frac{1}{2} - \gamma$ and similarly for distribution p_2, hypothesis h_2 achieves the optimal loss $\frac{1}{2} - \gamma$.

Next, note that for distribution p_1 the loss of any classifier $f : \Omega \to \{0, 1\}$ is

$$\Pr_{(X,Y)\sim p_1}(f(\omega^*) \neq Y) = \Pr(f(\omega^*) = 1) \times (\frac{1}{2} - \gamma) + \Pr(f(\omega^*) = 0) \times (\frac{1}{2} + \gamma).$$

Similarly its loss for distribution p_2 is

$$\Pr_{(X,Y)\sim p_2}(f(\omega^*) \neq Y) = \Pr(f(\omega^*) = 1) \times (\frac{1}{2} + \gamma) + \Pr(f(\omega^*) = 0) \times (\frac{1}{2} - \gamma).$$

Adding the two losses we get

$$\Pr_{(X,Y)\sim p_1}(f(\omega^*) \neq Y) + \Pr_{(X,Y)\sim p_2}(f(\omega^*) \neq Y) = 1$$

Therefore, every classifier incurs a loss of $\geq 1/2$ for at least one of the two distributions. Since the optimal loss for both distributions is $1/2 - \gamma$, any classifier incurs an excess loss of γ for at least one of the distributions among p_1 and p_2.

The distribution p of the data (X, Y), is chosen to be one of the two distributions p_1 and p_2 each with probability $1/2$. Then we show that depending on which distribution is chosen as p, the adversary can choose its batches such that, even with infinitely many batches, the two distributions are indistinguishable. Therefore, any classifier incurs an excess loss of γ with probability $\geq 1/2$.

Note that for every batch, the number of $Y = 1$'s is a sufficient statistic for determining weather p is p_1 or p_2, and it is distributed either $B(n, \frac{1}{2} + \gamma)$ or $B(n, \frac{1}{2} - \gamma)$. From equation 2.15 in [3], for any $c < 1/12$ and $\gamma = c\beta/\sqrt{n}$, the total variation distance between $B(n, \frac{1}{2} + \gamma)$ or

$B(n, \frac{1}{2} - \gamma)$ is $\leq 2\beta$.

Therefore, the adversary can choose distributions q_1 and q_2, over the number of $Y = 1$'s in the adversarial batches, such that

$$(1-\beta)B(n, \frac{1}{2} + \gamma) + \beta q_1 = (1-\beta)B(n, \frac{1}{2} - \gamma) + \beta q_2.$$

Hence, if the good batches are distributed as $B(n, \frac{1}{2} + \gamma)$ then adversary chooses q_1 as distribution of the adversarial batches and if good batches are distributed as $B(n, \frac{1}{2} - \gamma)$ then adversary chooses q_2 and in both the cases the resultant joint distribution of all the batches is same. Hence the two cases are indistinguishable. ■

The theorem implies that even with access to infinitely many batches, even for the simplest of the hypothesis class, no algorithm can avoid an excess loss $\Omega(\beta/\sqrt{n})$ with probability $1/2$.

3.9.4 Proof of Theorem 25

Proof. To prove the theorem, we show how to use algorithm in Theorem 18 that gives "cleaner" batches for $\mathcal{F}_k$-distance, to get "cleaner" batches for $\mathcal{F}_{\mathcal{H}_k}$-distance.

Recall that

$$\mathcal{F}_{\mathcal{H}_k} = \{(\{x \in \mathbb{R} : h(x) = y\}, \bar{y}) : h \in \mathcal{H}_k, y \in \{0,1\}\}.$$

Divide the collection of sets $\mathcal{F}_{\mathcal{H}_k}$ into two parts: $\mathcal{F}^0_{\mathcal{H}_k} := \{(\{x \in \mathbb{R} : h(x) = 0\}, 1) : h \in \mathcal{H}_k\}$ and $\mathcal{F}^1_{\mathcal{H}_k} := \{(\{x \in \mathbb{R} : h(x) = 1\}, 0) : h \in \mathcal{H}_k\}$. Note that $\mathcal{F}_{\mathcal{H}_k} = \mathcal{F}^0_{\mathcal{H}_k} \cup \mathcal{F}^1_{\mathcal{H}_k}$. Then, from the definition of $\mathcal{F}$ distance, it follows

$$||p-q||_{\mathcal{F}_{\mathcal{H}_k}} = \max\{||p-q||_{\mathcal{F}^0_{\mathcal{H}_k}}, ||p-q||_{\mathcal{F}^1_{\mathcal{H}_k}}\}$$

Hence, it suffices to estimate p in both $\mathcal{F}^0_{\mathcal{H}_k}$ and $\mathcal{F}^1_{\mathcal{H}_k}$ distances.

Since decision regions for each hypothesis $h \in \mathcal{H}_k$, consists of at most k-intervals, these collections can be rewritten as $\mathcal{F}^0_{\mathcal{H}_k} := \{(S, 0) : S \in \mathcal{F}_k\}$ and $\mathcal{F}^1_{\mathcal{H}_k} := \{(S, 1) : S \in \mathcal{F}_k\}$.

To learn in $\mathcal{F}^0_{\mathcal{H}_k}$ distance, w.l.o.g., we can remap all points of the form $(x, 1)$ to $(\infty, 0)$. Then this problem is identical to learning in $\mathcal{F}_k$ distance as $y = 0$ is the same for all samples after remapping. Similarly to learn in $\mathcal{F}^1_{\mathcal{H}_k}$ distance we remap all points of the form $(x, 0)$ to $(\infty, 1)$.

Then use the algorithm in Theorem 18 to first remove the adversarial batches for $\mathcal{F}^0_{\mathcal{H}_k}$ distance, and then for the remaining batches again use the same algorithm to remove adversarial batches for $\mathcal{F}^1_{\mathcal{H}_k}$ distance. The empirical distribution $\bar{p}_{B^*}$ of the batches $B^* \subseteq B$ remaining in the end, approximates p in both $\mathcal{F}^0_{\mathcal{H}_k}$ and $\mathcal{F}^1_{\mathcal{H}_k}$ distances to an accuracy $\mathcal{O}(\Delta)$. Therefore, it estimates p in $\mathcal{F}_{\mathcal{H}_k}$ distance to the same accuracy.

Then use the polynomial-time algorithm [105] to find the empirical risk minimizer $h \in \mathcal{H}_k$ for empirical distribution $\bar{p}_{B^*}$. Then Lemma 22 implies that the optimal classifier $h^{\text{opt}}(\bar{p}_{B^*})$ for the empirical distribution $\bar{p}_{B^*}$, of the cleaner batch collection B^*, will have a small-excess-classification-loss $\mathcal{O}(\Delta)$ for p. This completes the proof of the theorem. ■

Chapter 3, in full, is a reprint of the material as it appears in A general method for robust learning from batches 2020. Ayush Jain, Alon Orlitsky. In Neurips 2020. The dissertation author was the primary investigator and author of this paper.

Chapter 4

Robust Density Estimation from Batches: The Best Things in Life are (Nearly) Free

4.1 Overview

4.1.1 Robust learning

In many learning applications, some samples are inadvertently or maliciously corrupted. A natural and intuitive example shows that regardless of the number of samples available, such corruption severely curtails the learning accuracy even for the simplest of tasks, a binary hypothesis test.

Consider independent binary samples distributed either all Ber$(1/2+\beta/2)$ or all Ber$(1/2-\beta/2)$. With genuine samples, the underlying distribution can be identified with error that plummets to 0 exponentially fast in the number of samples.

However, if an adversary can observe a fraction $1-\beta$ of the samples and select the rest, our best error is destined to remain a half, regardless of the number of samples available. The poltergeist could simply use the observed samples to determine the underlying distribution, and set the rest so the whole sequence appears to be generated by a Ber$(1/2)$ distribution, leaving us with no better than a random guess.

This elemental example propagates to essentially all learning tasks, hard-limiting the performance of all learning algorithms. For example, the total variation (TV) distance between the two indistinguishable distributions above is β. Hence the triangle inequality implies that for any number of samples, if a β fraction are adversarial, then even binary, let alone general discrete and continuous, distributions cannot be learned to TV distance less than $\beta/2$. Similar hard limits follow for classification and other learning tasks.

The foregoing seems to suggest the discouraging conclusion that with a β fraction of adversarial data, an $\Omega(\beta)$ loss is inevitable, which as real-life β may be quite large, could be rather foreboding. Fortunately, that is not necessarily so.

In the following and many other applications, data are collected from multiple sources, most typically genuine, but some possibly corrupted or adversarial. Data may be gathered by sensors, each providing a large amount of data, and some sensors may be faulty. The word frequency of an author may be estimated from several large texts, some of which are mis-attributed. User preferences may be learned by querying several individuals, some intentionally biasing their feedback. Multiple agents may contribute to a crowd-sourcing platform, but some may be unreliable or malicious.

The collection of data generated by each source, or during a time period, is called a *batch*. Interestingly, for data generated in batches, a fraction β of which are corrupted or adversarial, significantly higher accuracy can be achieved.

4.1.2 Robust learning from batches

To formalize this setting, [125] considered estimating an unknown distribution p over the finite domain $[\ell] = \{1, \ldots, \ell\}$ in TV-distance. Estimation is based on m batches with $\geq n$ samples each. In most batches, the samples are drawn independently according to p, but a fraction $\beta < 0.5$ of the batches are *adversarial* and may be arbitrarily corrupted, possibly even with knowledge of the good batches.

Unlike the strict $\Theta(\beta)$ accuracy limit for individual samples, they derived a batch-setting algorithm that approximates p to a much lower TV-distance $\mathcal{O}(\beta/\sqrt{n})$, where the implied constant factor is independent of ℓ. They also showed a matching *adversarial lower bound* (for batches), that even for binary distributions, and hence for general finite ones, the lowest achievable TV distance with any number of batches is $\geq \Delta_{\text{min}} := \Delta_{\text{min}}(\beta, n) := \beta/(2\sqrt{2n})$.

However, their estimator had some limitations as well. When all samples are genuine, and none is adversarial, estimating p to TV distance ϵ requires $\Theta(\ell/\epsilon^2)$ samples, *e.g.,* [86]. Since robust learning is at least as hard, this also forms a *statistical lower bound* on the number of samples required to achieve error ϵ with adversarial batches.

To achieve TV distance $\mathcal{O}(\beta/\sqrt{n}) = \mathcal{O}(\Delta_{\text{min}})$, the estimator in [125] required $\Omega(\frac{n+\ell}{n \cdot \Delta_{\text{min}}^2})$ batches, hence $\Omega(\frac{n+\ell}{\Delta_{\text{min}}^2})$ samples that for $n \gg \ell$ exceeds the statistical lower bound. Crucially, and much more significantly, its run-time was exponential in the domain size ℓ, rendering its application, or even simulation, infeasible for even moderate size domains.

The first polynomial-time, and practical, algorithm for the problem was derived in [77]. The algorithm efficiently finds and removes, or *filters*, "outlier" adversarial batches that significantly perturb the empirical distribution away from the underlying p, and then estimates p as the empirical distribution of the remaining batches. It achieves TV distance $\mathcal{O}(\Delta)$, where $\Delta := \Delta(\beta, n) := \Delta_{\text{min}} \cdot \sqrt{\ln(1/\beta)}$ is essentially the adversarial lower bound. To achieve this error they require $\mathcal{O}(\ell/\Delta^2)$ samples, matching the statistical lower bound even when all samples are genuine.

4.1.3 Robust learning large and continuous distributions

Since many modern applications utilize very large, often continuous, domains, even linear ℓ/Δ^2 dependence of the sample complexity on the domain size may be prohibitive.

Fortunately, common distributions often possess some structure that facilitates more efficient learning. One of the most popular, and important structures is piecewise polynomials.

A distribution q over $[a, b]$ is *t-piecewise degree-d* if for some partition of $[a, b]$ into t intervals $I_1, \ldots, I_t$, and degree-d polynomials $r_1, \ldots, r_t$, $\forall j$, $x \in I_j$, $q(x) = r_j(x)$. Let $\mathcal{P}_{t,d}$ denote the set of all t-piecewise degree d distributions.

Piecewise-polynomials include important distribution families, *e.g.,* $\mathcal{P}_{t,0}$ for histograms and $\mathcal{P}_{t,1}$ for piecewise-linear distributions. They can also approximate any piecewise continuous distribution. Importantly, with very low t and d, they arbitrarily closely approximate many staple one-dimensional distribution families, including Gaussians and their mixtures, log-concave, low-modal, and monotone hazard *e.g.,* [1].

For genuine, non-adversarial, samples, several works, *e.g.,* [1, 67], derived efficient algorithms that learn t-piecewise degree-d polynomials to TV distance ϵ, with optimal $\mathcal{O}(td/\epsilon^2)$ sample complexity.

$\mathcal{P}_{t,d}$ can be similarly defined as discrete distributions over the interval domain $[\ell]$. [30] showed that these distributions can be robustly learned from batches to TV-distance $\mathcal{O}(\Delta)$ with sample complexity only quasi-poly-logarithmic in ℓ. However their sample complexity was quasi-polynomial in the other parameters t, d, batch size n, and $1/\beta$. And the algorithm's computational complexity was quasi-polynomial in these parameters and the domain size ℓ.

If computation time is no object, [76] presented an exponential-time estimator that achieves TV distance $\mathcal{O}(\Delta)$ with $\tilde{\mathcal{O}}(td/\Delta^2)$ samples, the same, up to log logarithmic factors, as the minimum genuine samples required.

To obtain polynomial-time algorithms with low sample complexity, subsequent works adapted the filtering approach of [77]. For example, [31] achieved TV-distance $\mathcal{O}(\Delta)$ using $\tilde{\mathcal{O}}((td\log\ell)^2/\Delta^2)$ samples, and concurrently [76] achieved the same TV distance using $\tilde{\mathcal{O}}(td/\Delta^3)$ samples, in particular, removing the dependence of sample complexity on the domain size, and for the first time, enabling robust learning over infinite and continuous domains.

4.1.4 Overview of results and applications

Still, both $\tilde{\mathcal{O}}((td\log\ell)^2/\Delta^2)$ and $\tilde{\mathcal{O}}(td/\Delta^3)$ exceed the $\mathcal{O}(td/\Delta^2)$ optimal sample complexity of genuine samples, leading [31] to raise the open question of the optimal sample complexity of robust polynomial $\mathcal{P}_{t,d}$ estimators.

This paper essentially answers this question. We derive a filter-based polynomial-time algorithm that achieves TV distance $\mathcal{O}(\Delta)$ using only $\tilde{\mathcal{O}}(td/\Delta^2)$ samples, that up to poly-logarithmic factors matches the statistical lower bound of even genuine samples. It therefore essentially determines the sample complexity of robust and efficient learning of piecewise polynomials to optimal accuracy. It also shows that for this large and general class, robustness can be achieved at the small cost of at most a poly-logarithmic increase in the number of samples.

These results apply to both continuous and discreet distributions, and as described in Subsection 4.2.2 also learn distributions that can be approximated by $\mathcal{P}_{t,d}$, hence apply to monotone, log-concave, Gaussian, Gaussian mixtures, and other fundamental distribution classes.

While we present the results in terms of robust density estimation, their distance to other fundamental learning staples is minute. We demonstrate two such applications.

The first is to robust classification. We show that a simple extension of the results yields the first sample-optimal, polynomial-time, robust, classifier based on batched training data. We demonstrate the method's efficacy on the fundamental and practical problem of interval-based classification over the real line.

The second application is to the common *top k* or *heavy hitters* problem that calls for finding the k highest-probability elements in a distribution over a large domain. The problem arises in many applications ranging from caching, to recommendation systems, and vaccine design. We show that in the batch setting, the top k elements can be approximated robustly with sample complexity linear in k regardless of the domain size.

4.1.5 Other related works

This paper builds on several long and impressive lines of work, briefly summarized herein. Structured-distribution estimation was studied in [27, 115, 44, 8, 1, 67]. Robust-statistics was introduced in the classical works of [142, 74]. Efficient algorithms for learning the mean and covariance matrices of high-dimensional sub-gaussian and other distributions with bounded fourth moments in the presence of the adversarial samples was studied in [99, 47]. When more than half of the samples are adversarial, the underlying distribution cannot be estimated well, and instead, [29] returned a small set of candidate distributions one of which is a good approximate of the underlying distribution. For extensive surveys on robust learning algorithms see [135, 46].

The filtering approach to robust estimation was introduced in [47], and used in several subsequent applications including high dimensional estimation [48, 50, 135, 46]. These estimators applied to single samples and learned in L_2 distance. By contrast, the results in this paper and those in [77, 76] address batch learning under TV-distance.

Several recent works considered related "multi-source" or "collaborative" PAC learning scenarios. As in our setting, they assume multiple sources, some genuine and others possibly adversarial, where each source provides multiple labeled samples, but some specific assumptions differ. [10] considers only the realizable case and allow actively acquisition of more data from the source of choice. [124] also focuses on realizable case where sources share a common labelling function, but may have different input distributions. [96] considers the setting that most closely resembles ours and the more general prior work [76], but they do not present efficient algorithms and incur sub-optimal $\mathcal{O}(\sqrt{k}\Delta)$ excess loss, higher than the $\mathcal{O}(\Delta)$ we achieve.

4.1.6 Organization of the paper

In the next section we describe the main results we obtain, the techniques used to derive them, and some of their applications. In Section 4.3, we simplify the learning problem to that of learning all k-element subsets of a large discrete set. In Section 4.4 we describe an efficient filtering algorithm for this problem. In Section 4.5 we describe the experiments. The appendix contains most of the proofs.

4.2 Main techniques, results, and applications

While we would like to learn continuous distribution in TV distance, as in [30, 76, 31], it will prove advantageous to first learn them in a weaker (smaller) distance.

4.2.1 Density estimation in $\mathcal{A}_k$ distance

Recall that the TV distance between two real distributions q and q' is the maximum of $|q(S) - q'(S)|$ over all Borel sets $S \subseteq \mathbb{R}$. This notion generalizes to arbitrary collections $\mathcal{S}$ of real sets. The $\mathcal{S}$-*distance* between q and q' is

$$||q - q'||_{\mathcal{S}} := \max_{S \in \mathcal{S}} |q(S) - q'(S)|.$$

For $k \geq 1$, let $\mathcal{A}_k$ be the collection of all unions of at most k real intervals. Clearly $||q - q'||_{\mathcal{A}_k} \leq ||q - q'||_{\text{TV}}$ for any q, q', with equality when the domain size $\ell \leq k$. Hence from now on we assume $\ell > k$, and can also be infinite.

One nice property of $\mathcal{A}_k$ distance is that with only genuine samples, the empirical distribution itself, already estimates any discrete or continuous distribution to $\mathcal{A}_k$ distance ϵ with $\mathcal{O}(k/\epsilon^2)$ samples, which is also optimal.

To learn distributions in $\mathcal{A}_k$ distance, [76] and [31] adapted the filtering algorithm in [77]. First removing outlier batches, and retaining batches whose empirical distribution approximates p in $\mathcal{A}_k$, rather than TV, distance.

However, both algorithms had suboptimal sample complexity. For $\Delta = \Delta_{\text{min}}\sqrt{\ln(1/\beta)} = \Theta(\beta\sqrt{\ln(1/\beta)/n})$, essentially the best $\mathcal{A}_k$ distance achievable with n-sample batches, they required $\tilde{\mathcal{O}}(k/\Delta^3)$, and $\tilde{\mathcal{O}}((k \log \ell)^2/\Delta^2)$ samples, respectively.

Our fundamental contribution is an algorithm that learns any discrete or continuous distribution to $\mathcal{A}_k$-distance Δ with sample complexity $\tilde{\mathcal{O}}(k/\Delta^2)$, optimal up to logarithmic factors.

Theorem 45. *For some constants $c < 1/2$ and $C > 1$, for any k, $\beta < c$, $\delta < 1$, $n > \Omega(\log^C(1/\beta))$, and discrete or continuous p, the algorithm uses $m \cdot n = \tilde{\mathcal{O}}(\frac{k+\log(1/\delta)}{\Delta^2})$ total samples, and in time* poly(k, n, m, β, δ) *outputs an estimate $\hat{p}$ that with probability $\geq 1 - \delta$ satisfies*

$$||\hat{p} - p||_{\mathcal{A}_k} \leq \mathcal{O}(\Delta).$$

Remark.

Theorem 45 achieves $\mathcal{A}_K$ distance $\mathcal{O}(\Delta)$, within a small $O(\sqrt{\log(1/\beta)})$ factor from the adversarial lower bound for unlimited samples. The algorithm uses a poly-logarithmic factor more samples than the min-max number required for this distance even with strictly genuine data. When the number of samples does not suffice to achieve the minimal $\mathcal{A}_K$ distance of $\mathcal{O}(\Delta)$, the algorithm can be modified to achieve $\mathcal{A}_K$ distance $\tilde{\mathcal{O}}(\sqrt{k/(mn)})$, again within a poly-logarithmic factor from the statistical lower bound as achieving $\mathcal{A}_K$ distance ϵ requires at least k/ϵ^2 genuine samples. This result can be derived by augmenting Theorem 45 with the steps taken in the derivation of Theorem 2 in [77]. A similar observation also holds for all the applications stated next.

To derive the algorithm, we first reduce robust $\mathcal{A}_k$ learning over any domain, even continuous, to robust learning the probability of all $2k$-element subsets of discrete distributions over large domains. We propose a filtering algorithm that learns these probabilities with optimal sample complexity linear in k and independent of the domain's size. The new, simpler, formulation allows for a tight SDP relaxation, that with more refined analysis yields near optimal sample complexity.

The algorithm has several important implications. We apply it to three robust-learning tasks using batched data: (i) learning distributions in or near $\mathcal{P}_{t,d}$, (ii) interval-based binary classification, (iii) learning the top-k heavy hitters. For all three problems we achieve the nearly best possible TV distance $\mathcal{O}(\Delta)$ with the same sample complexity as with genuine samples up to logarithmic factors.

4.2.2 Density estimation in TV distance

Theorem 45 described an optimal, robust, batch-based, algorithm for learning any distribution over the reals in $\mathcal{A}_k$ distance. Yet $||q - q'||_{\mathcal{A}_k} \leq ||q - q'||_{\text{TV}}$ for any q, q'. This section extends the results to robustly learn in the more standard, and stringent, TV-distance.

In Theorem 47 we present a batch-based algorithm that robustly learns $\mathcal{P}_{t,d}$ and related distributions $\mathcal{P}_{t,d}$, including monotone, log-concave, Gaussian, Gaussian mixtures, and other fundamental distributions.

For real distribution p, let $\text{opt}_{t,d}(p) := \inf_{q \in \mathcal{P}_{t,d}} ||p - q||_{TV}$ be p's TV-distance to its nearest distribution in $\mathcal{P}_{t,d}$. We wish to find a distribution $\hat{p}$ such that for a small ϵ and universal constant α, with probability $\geq 1-\delta$,

$$||\hat{p}-p||_{TV} \leq \alpha \cdot \text{opt}_{t,d}(p)+\epsilon.$$

This ensures that we learn distributions not just in $\mathcal{P}_{t,d}$, but also nearby. While not emphasized here, the α we derive is roughly 3, and same as the best known factor for learning $\mathcal{P}_{t,d}$ with only genuine samples.

To convert learning $\mathcal{A}_k$- to TV-distance, we use a transformation that maps $\mathcal{A}_k$ neighborhoods of distributions in or near $\mathcal{P}_{t,d}$ to TV-neighborhoods.

Theorem 46. *[1] For a constant α (roughly 3) and any t, d, and ϵ, an algorithm they describe runs in time* $\text{poly}(t,d,\epsilon)$ *and converts any real distribution p' to a distribution p'' such that for every distribution p,*

$$||p-p''||_{TV} \leq \alpha \cdot \text{opt}_{t,d}(p)+\mathcal{O}(||p-p'||_{\mathcal{A}_{t(d+1)}})+\epsilon.$$

The theorem shows that if p is near $\mathcal{P}_{t,d}$, then an $\mathcal{A}_{t(d+1)}$ distance approximation of p can be converted to a TV-distance approximation of p, hence it suffices to approximate p in the weaker $\mathcal{A}_{t(d+1)}$ distance.

Combining Theorems 45 and 46 for $k=t(d+1)$, we derive a polynomial-time algorithm that robustly estimates any real distribution nearly as well as its best $\mathcal{P}_{t,d}$ approximation, using the optimal number of samples.

Theorem 47. *For some constants α (roughly 3), $c<1/2$, and $C>1$, for any t, d, $\beta<c$, $\delta<1$, $n>\Omega(\log^C(\frac{1}{\beta}))$, and real or discrete distribution p, a simple combination of the above algorithms uses $m\cdot n=\tilde{\mathcal{O}}(\frac{t(d+1)+\log(1/\delta)}{\Delta^2})$ total samples, and in time poly(t,d,n,m,β,δ) outputs an estimate $\hat{p}$ that with probability $\geq 1-\delta$ satisfies*

$$||\hat{p}-p||_{TV} \leq \alpha \cdot \text{opt}_{t,d}(p)+\mathcal{O}(\Delta).$$

Note that the adversarial-batch lower bound on the approximation's TV distance is $\Delta_{\text{min}} = \beta/(2\sqrt{2n})$, while the theorem, like all other robust-learning results so far, applies to a slightly higher TV distance $\Delta = \mathcal{O}(\Delta_{\text{min}}\sqrt{\log(1/\beta)})$. Based on evidence from Gaussian robust mean estimation, [31] suggested that the extra $\mathcal{O}(\sqrt{\log(1/\beta)})$ factor may be necessary for any polynomial time algorithm.

4.2.3 Application to interval-based classification

We now show that though presented for density estimation, a simple extension of our results yields the first polynomial-time, sample-optimal, robust batch classifier, and demonstrate it on the fundamental and practical problem of interval-based binary classification over the reals.

Without loss of generality let the observations be distributed over $[0, 1]$. Each good batch therefore contain n labeled samples from a distribution p over $[0, 1] \times \{-1, 1\}$, while the adversarial batches contain n arbitrary pairs.

Consider a hypothesis family of Boolean functions $\mathcal{H}_k : [0, 1] \to \{-1, 1\}$ whose decision regions, the inverse images of -1 and 1, consist of at most k-intervals. The loss of classifier $h \in \mathcal{H}_k$ for any distribution q over $[0, 1] \times \{-1, 1\}$ is $r_q(h) := \Pr_{(X,Y)\sim q}[h(X) \neq Y]$. The *optimal* $\mathcal{H}_k$ *classifier* for a distribution q is $h^{\text{opt}}(q) := \arg\min_{h\in\mathcal{H}_k} r_q(h)$, and the *optimal loss* is $r_q^{\text{opt}}(\mathcal{H}_k) := r_q(h^{\text{opt}}(q))$.

Given samples from an underlying distribution p, the goal is to return a classifier $h \in \mathcal{H}_k$ whose *excess loss* $r_p(h) - r_p^{\text{opt}}(\mathcal{H}_k)$ relative to the optimal loss is small.

Map any distribution q over $[0, 1] \times \{-1, 1\}$, to a new distribution $q^{[-1,1]}$ over $[-1, 1]$, where $q^{[-1,1]}(z) := \Pr(X \cdot Y = z)$ for $(X, Y) \sim q$. Note that there is a 1-1 correspondence between q and $q^{[-1,1]}$, and that we can define $\mathcal{A}_k$ distance over the new domain $[-1, 1]$.

Lemma 6 in [76] upper bounds the excess loss when the optimal classifier for distribution q is applied to distribution p in terms of $\mathcal{A}_k$ distance between $p^{[-1,1]}$ and $q^{[-1,1]}$. For completeness we present a short proof in Appendix 4.11.

Lemma 48. *For any distributions p, q over* $[0,1] \times \{-1,1\}$,

$$r_p(h^{opt}(q)) - r_p^{opt}(\mathcal{H}_k) \leq 2||p^{[-1,1]} - q^{[-1,1]}||_{\mathcal{A}_{2k}}.$$

Furthermore, [105] derived an algorithm that for any empirical distribution q over $[0,1] \times \{-1,1\}$ finds the optimal classifier $h^{\text{opt}}(q)$ in polynomial time in the number of samples and k. Then from the above Lemma to obtain an excess loss $\mathcal{O}(\Delta)$ it suffices to estimate $p^{[-1,1]}$ to $\mathcal{A}_k$ distance $\mathcal{O}(\Delta)$.

Theorem 45 provides an algorithm to learn any real distribution to $\mathcal{A}_k$ distance $\mathcal{O}(\Delta)$ using $\tilde{\mathcal{O}}(k/\Delta^2)$ samples, implying the following.

Theorem 49. *For some constants $c < 1/2$ and $C > 1$, for any k, $\beta < c$, $\delta < 1$, $n > \Omega(\log^C(1/\beta))$, and p over* $[0,1] \times \{-1,1\}$, *the above algorithm uses $m \cdot n = \tilde{\mathcal{O}}(\frac{k+\log(1/\delta)}{\Delta^2})$ pairs, and in* $\text{poly}(k, n, m, \beta, \delta)$ *time outputs a classification h^* with excess loss $r_p(h^*) - r_p^{\text{opt}}(\mathcal{H}_k) \leq \mathcal{O}(\Delta)$.*

Since the VC-dimension of the collection $\mathcal{H}_k$ is $\mathcal{O}(k)$, any algorithm achieving excess loss ϵ requires $\Omega(k/\epsilon^2)$ samples, even with genuine data. Therefore, achieving excess loss $\mathcal{O}(\Delta)$ requires $\Omega(k/\Delta^2)$ samples, even with genuine data, showing that our algorithm is sample optimal up to logarithmic factors.

[76] showed that the best possible excess loss for this problem is $\Omega(\Delta_{\text{min}})$. They used a similar reduction from $\mathcal{A}_k$ distance, to derive a polynomial-time algorithm with $\mathcal{O}(\Delta)$ excess loss, but required a suboptimal $\tilde{\mathcal{O}}(k/\Delta^3)$ number of samples.

4.2.4 Application to the top k heavy hitters problem

Our last application is to the prevalent *top k*, or *heavy hitters*, problem. Given samples from a distribution over a large domain, we would like to find the k elements with highest probability. This problem arises in numerous applications including deciding which pages to store in a cache, results to show on the front page of a web search, viruses to inoculate for in an influenza vaccine [150, 25], and products to recommend to online shoppers.

As in the rest of the paper, we consider samples that arrive in batches, some possibly corrupt or adversarial. For example, some shoppers biasing consumer ratings towards select products.

The top k elements clearly have the highest total probability among all k-element subsets. However, this set cannot always be found as some elements with nearly identical probabilities cannot be identified. Instead, we therefore aim to robustly find a k-element subset whose total probability is maximal up to a $\mathcal{O}(\Delta)$ difference.

The results in this section apply to all discrete distributions, that without loss of generality we assume range over the integers. They can be trivially extended to mixed distributions over the reals as well.

A natural approach may be to learn p robustly to TV distance Δ as in [77], and return the k element subset with highest estimated probability. However, this approach would require number of samples proportional to the domain size, while in a typical k-hitter problem, k is significantly smaller.

Instead, we first estimate p to an $\mathcal{A}_k$ distance $\mathcal{O}(\Delta)$, which from Theorem 45 can be done efficiently using $\tilde{\mathcal{O}}(k/\Delta^2)$ samples. We then return the k-element subset with highest estimated probability. Since the collection $\mathcal{A}_k$ is a superset of the collection of all subsets of size $\leq k$, learning to an $\mathcal{A}_k$ distance $\mathcal{O}(\Delta)$ implies learning the probability of all such subsets to accuracy $\mathcal{O}(\Delta)$. By the triangle inequality, the k-element subset with highest estimated probability is maximal up to a $2\mathcal{O}(\Delta)$ probability difference.

4.3 Two simplifications of A_k-distance learning

4.3.1 Discretization using partitioning

Let B denote a collection of all m batches. Recall that each batch has n samples. For $s = n \cdot m$, let $x^s = x_1, x_2, \ldots, x_s \in \mathbb{R}$ be the samples of B sorted in non-decreasing order, and define $\bar{p}_B$ to be the empirical distribution of x^s.

Given samples x^s, for $j \geq 1$, let $\mathcal{P}^j = P_1^j, P_2^j, \ldots, P_{k \cdot j}^j$, partition $\mathbb{R}$ into $k \cdot j$ disjoint intervals, or *parts*, each containing $\approx \frac{s}{k\cdot j}$ samples, and given by

$$P_i^j := \begin{cases} (-\infty, x_{\lfloor \frac{s}{k\cdot j}\rfloor}] & i = 1, \\ (x_{\lfloor \frac{(i-1)s}{k\cdot j}\rfloor}, x_{\lfloor \frac{i\cdot s}{k\cdot j}\rfloor}] & 2 \le i < k\cdot j, \\ (x_{\lfloor \frac{(i-1)s}{k\cdot j}\rfloor}, \infty) & i = k\cdot j. \end{cases}$$

Let $\mathcal{C}(\mathcal{P}^j)$ be the collection of real subsets formed by unions of parts of $\mathcal{P}^j$. Unions of consecutive parts of $\mathcal{P}^j$ are themselves intervals in $\mathbb{R}$ that we call *intervals over* $\mathcal{P}^j$.

Let $\mathcal{A}_k(\mathcal{P}^j)$ be the collection of all unions of at most k intervals over $\mathcal{P}^j$. Clearly, $\mathcal{A}_k(\mathcal{P}^j) \subseteq \mathcal{A}_k$, hence $||q - q'||_{\mathcal{A}_k(\mathcal{P}^j)} \le ||q - q'||_{\mathcal{A}_k}$ for any distributions q and q'. Interestingly a reverse relation holds for the underlying distribution p.

Lemma 50. *For $m \cdot n = \tilde{\Omega}(k/\Delta^2)$, w.h.p., for all $j \ge \frac{1}{\Delta}$ and all distributions q over $\mathbb{R}$,*

$$||q - p||_{\mathcal{A}_k} \le ||q - p||_{\mathcal{A}_k(\mathcal{P}^j)} + \mathcal{O}(\Delta).$$

To prove the lemma we need the following results, proved in Appendix 4.11.

Lemma 51. *For any subset $S \in \mathcal{A}_k$, there are sets $S', S'' \in \mathcal{A}_k(\mathcal{P}^j)$ such that $S' \subseteq S \subseteq S''$ and $\bar{p}_B(S'' \setminus S') \le 2/j$.*

Lemma 52. *For $\beta < 1/2$, $m \cdot n = \tilde{\Omega}(\frac{k+\log 1/\delta}{\Delta^2})$, with probability $> 1 - \delta$, for all $S \in \mathcal{C}(\mathcal{P}^j)$, $p(S) \le 2 \cdot \bar{p}_B(S) + \mathcal{O}(\Delta)$.*

Proof of Lemma 50. From Lemma 51 for any subset $S \in \mathcal{A}_k$, let $S', S'' \in \mathcal{A}_k(\mathcal{P}^j)$ be the sets such that $S' \subseteq S \subseteq S''$ and $\bar{p}_B(S'' \setminus S') \le \mathcal{O}(1/j)$. Clearly $S'' \setminus S' \subseteq \mathcal{C}(\mathcal{P}^j)$, then from

Lemma 52, $w.h.p.$, $p(S'' \setminus S') \le 2 \cdot \bar{p}_B(S'' \setminus S') + \mathcal{O}(\Delta) \le \mathcal{O}(1/j + \Delta)$. Then

$$\begin{aligned} p(S) - q(S) &\le p(S) - q(S') \\ &= p(S') - q(S') + p(S \setminus S') \\ &\le p(S') - q(S') + p(S'' \setminus S') \\ &\le ||q - p||_{\mathcal{A}_k(\mathcal{P}^j)} + \mathcal{O}(1/j + \Delta). \end{aligned}$$

A similar bound for $q(S) - p(S)$ completes the proof. ■

The lemma shows that to approximate p in $\mathcal{A}_k$-distance it suffices to estimate it in $\mathcal{A}_k(\mathcal{P}^j)$-distance for any $j = \Omega(\frac{1}{\Delta})$. The advantage of this reduction is that the set $\mathcal{A}_k(\mathcal{P}^j)$ is finite in contrast to $\mathcal{A}_k$.

Given a distribution q on $\mathbb{R}$, for any $j \ge 1$ let $q^j \in \mathbb{R}^{k \cdot j}$ be the discrete distribution over the indices of partition $\mathcal{P}^j$, defined by $q^j(i) = q(P_i^j)$ for $i \in [k \cdot j]$.

Map every subset $S \in \mathcal{C}(\mathcal{P}^j)$ to the binary vector $v_S \in \{0,1\}^{k \cdot j}$ whose ith coordinate indicates whether $P_i^j \subseteq S$. Observe that for any distribution q over $\mathbb{R}$, we can express $q(S)$ as the inner product $q^j \cdot v_S$. Let $\mathcal{V}_k^\ell$ denotes the collection of binary vectors $\{0,1\}^\ell$ with at most k runs of ones. Since each interval over $\mathcal{P}^j$ corresponds to a single run of ones, if $S \in \mathcal{A}_k(\mathcal{P}^j)$, then $v_S \in \mathcal{V}_k^{k \cdot j} \subseteq \{0,1\}^{kj}$.

This discussion and Lemma 50 show that if for the discretized versions of an estimator $\hat{p}$ and underlying distribution p, $\max_{v \in \mathcal{V}_k^{k \cdot j}} |\hat{p}^j \cdot v - p^j \cdot v| \le \mathcal{O}(\Delta)$ then $||\hat{p} - p||_{\mathcal{A}_k} \le \mathcal{O}(\Delta)$. However, the collection $\mathcal{V}_k^{k \cdot j}$ is rather complicated and does not have a tight convex relaxation. Previous relaxations of $\mathcal{V}_k^{k \cdot j}$ [31] lead to sub-optimal sample complexities. Instead, we show in the next section that this problem can be further reduced to robust learning of the probabilities of all subsets of a fixed size $2k$ over a large discrete domain. In Section 4.4, we show that these probabilities can be robustly estimated with optimal sample-complexity $\tilde{\mathcal{O}}(k)$.

4.3.2 Reduction to learning k element subset

Let $\mathcal{I}(\mathcal{P}^j) \subseteq \mathcal{C}(\mathcal{P}^j)$ consist of all unions of at most $2k$ parts of $\mathcal{P}^j$. Let $\{0,1\}_k^\ell$ denote the set of binary vectors of length ℓ with at most k ones. Observe that every subset in $S \in \mathcal{I}(\mathcal{P}^j)$ corresponds to a binary vector $v_S \in \{0,1\}_{2k}^{k\cdot j}$. Note that $\mathcal{I}(\mathcal{P}^2) = \mathcal{C}(\mathcal{P}^2)$, as $\{0,1\}_{2k}^{2k} = \{0,1\}^{2k}$.

We now show that to estimate p in $\mathcal{A}_k$ distance it suffices find a q such that $\forall j \in 2^{[\log(1/\Delta)]}$, the powers of two between 2 and $1/\Delta$, the distances $||p-q||_{\mathcal{I}(\mathcal{P}^j)}$ are small.

Theorem 53. *For every $m \cdot n = \tilde{\Omega}(\frac{k+\log 1/\delta}{\Delta^2})$ and distribution q over $\mathbb{R}$, with probability $> 1-\delta$,*

$$||q-p||_{\mathcal{A}_k} \le \sum_{j \in 2^{[\log(1/\Delta)]}} \max_{v \in \{0,1\}_{2k}^{k\cdot j}} |q^j \cdot v - p^j \cdot v| + \mathcal{O}(\Delta).$$

Note that for any j, the set $\mathcal{I}(\mathcal{P}^j) \subset \mathcal{A}_{2k}$, therefore the sample complexity of estimating p in $\mathcal{I}(\mathcal{P}^j)$ distance is at most that of learning in $\mathcal{A}_{2k}$-distance.

Importantly, this reduces the more complicated set $\mathcal{V}_k^{k\cdot j}$ to more manageable sets $\{0,1\}_{2k}^{k\cdot j}$, which, as we see in the next section, have nice convex relaxations.

To prove Theorem 53, note a simple geometric observation, proved in Appendix 4.11.

Lemma 54. *For any $i \ge 1$, any interval over partition $\mathcal{P}^{2^i}$ is the union of at-most 2 parts from each partition $\mathcal{P}^{2^i}, \mathcal{P}^{2^{i-1}}, ..., \mathcal{P}^{2^2}$ and one interval over $\mathcal{P}^2$.*

The following result is a simple consequence.

Lemma 55. *For any $i \ge 1$, any subset in $\mathcal{A}_k(\mathcal{P}^{2^i})$ is the union of one subset from each of $\mathcal{I}(\mathcal{P}^{2^i})$, $\mathcal{I}(\mathcal{P}^{2^{i-1}})$,...,$\mathcal{I}(\mathcal{P}^{2^1})$.*

Proof of Lemma 55. Any subset in $\mathcal{A}_k(\mathcal{P}^{2^i})$ is a union of at most k intervals over partition $\mathcal{P}^{2^i}$, and Lemma 54 implies that it can be expressed as a union of at-most $2k$ parts from each partition $\mathcal{P}^{2^i}, \ldots, \mathcal{P}^{2^2}$ and at most k intervals over $\mathcal{P}^2$. The lemma follows as any union of intervals over $\mathcal{P}^2$ is in $\mathcal{C}(\mathcal{P}^2)$, and $\mathcal{C}(\mathcal{P}^2) = \mathcal{I}(\mathcal{P}^2)$. ■

Proof of Theorem 53. For any distribution q over $\mathbb{R}$, Lemma 55 and the triangle inequality imply

$$\begin{aligned} ||q-p||_{\mathcal{A}_k(\mathcal{P}^{2^i})} &= \max_{S\in\mathcal{A}_k(\mathcal{P}^{2^i})} |q(S)-p(S)| \\ &\le \sum_{\ell=1}^{i} \max_{S\in\mathcal{I}(\mathcal{P}^{2^\ell})} |q(S)-p(S)|. \end{aligned}$$

Letting $i = \lfloor \log_2(\frac{2}{\Delta}) \rfloor$ and Lemma 50 complete the proof. ■

4.4 Filtering algorithm for $\mathcal{A}_k$ distance

4.4.1 Notation

We begin with notation that helps describe the filtering algorithm. Recall that B is the collection of m batches, each consisting of $\geq n$ samples. Let B_G denote the collection of all *good* batches in B whose samples are drawn independently from common unknown real distribution p. We refer to the batches in remaining set $B_A := B \setminus B_G$ as *adversarial*. Note that $|B_A| \leq \beta m$.

Let $\bar{\mu}_b$ denote the empirical distribution of samples in batch $b \in B$. Note that $\bar{\mu}_b$ is a collection of n Dirac delta functions. Let B' denote any sub-collection of B. For a batch sub-collection $B' \subseteq B$, consider the average of the empirical distributions of batches in B'.

$$\bar{p}_{B'} \triangleq \frac{1}{|B'|} \sum_{b\in B'} \bar{\mu}_b.$$

Note that $\bar{p}_{B'}$ is also the empirical distribution of all samples in batches of B'.

Recall that for any distribution q over $\mathbb{R}$, $q^j \in \mathbb{R}^{k\cdot j}$ is the discrete distribution induced over the the parts of partition $\mathcal{P}^j$, and let $\bar{\mu}_b^j$ and $\bar{p}_{B'}^j$ be the corresponding empirical distributions of batch b and batch collection B', respectively.

For any discrete distribution, or normalized frequency vector, q, let $\mathrm{Mul}^N(q,n)$ denote the distribution of a normalized multinomial frequency vector μ, where $n \cdot \mu \sim \mathrm{Mul}(q,n)$. Also, let $C(q) := \frac{1}{n}(\text{Diag}(q) - qq^\intercal)$ be the covariance of $\text{Mul}^N(q, n)$.

Let $\mu_1, \ldots, \mu_m \sim \text{Mul}^N(q, n)$ be m i.i.d. normalized frequency vectors, and let $\bar{\mu}$ and V be the mean and covariance of the μ_i's. Intuitively speaking, both V and $C(\bar{\mu})$ converge to the covariance of $\text{Mul}^N(q, n)$, hence their difference tends to zero.

If the partition $\mathcal{P}^j$ was fixed beforehand, not after obtaining the samples, then for $b \in B_G$, the frequency vector $\bar{\mu}_b^j$ would follow a normalized multinomial distribution $\text{Mul}^N(p^j, n)$. Even though the partition depends on the samples, the above multinomial-distribution intuition is still useful as the distribution of $\bar{\mu}_b^j$ is still essentially $\text{Mul}^N(p^j, n)$.

For any batch b, and sub-collection B', let $C_{b,B'}^j := (\bar{\mu}_b^j - \bar{p}_{B'}^j)(\bar{\mu}_b^j - \bar{p}_{B'}^j)^\intercal$ be the *deviation* of batch b relative to batch collection B'.

The *filtering statistics* of a batch b w.r.t. a sub collection B', $F_{b,B'}^j = C_{b,B'}^j - C(\bar{p}_{B'}^j)$ is the difference between the deviation of batch b relative to batch collection B' and covariance matrix of a frequency vector μ generated using the distribution $\mu \sim \text{Mul}(\bar{p}_{B'}^j, n)$. Finally, the *filtering statistics* of a batch sub collection $B' \subseteq B$ is the average $F_{B'}^j := \frac{1}{|B'|}\sum_{b \in B'} F_{b,B'}^j$ of the filtering scores of all batches $b \in B'$ w.r.t. this sub collection B'.

Note that $F_{B'}^j = \frac{1}{|B'|}\sum_{b \in B'} C_{b,B'}^j - C(\bar{p}_{B'}^j)$ is the difference between the empirical covariance matrix of $\{\bar{\mu}_b^j\}_{b \in B'}$, and the covariance matrix of the normalized multinomial distribution with parameter $q = \bar{p}_{B'}^j$, the mean of frequency vectors $\bar{\mu}_b^j$ in B'.

We note that this filtering statistics was first used in [77] to robustly learn discrete distributions in TV distance, and later used in [76, 31] for learning in $\mathcal{A}_k$ distance.

4.4.2 The filtering algorithm

If there were no adversarial batches, the empirical distribution $\bar{p}_B$ of all batches would estimate p in $\mathcal{A}_k$ distance. However, the presence of adversarial outlier batches can move the empirical distribution $\bar{p}_B$ away from p.

We derive a filtering algorithm that finds a sub-collection B' of batches such that

$\forall\, j \in 2^{[\log(1/\Delta)]}$

$$\max_{v\in\{0,1\}^{k\cdot j}_{2k}} |\bar{p}^{j}_{B'} \cdot v - p^{j} \cdot v| \le \mathcal{O}(\tfrac{\beta}{\log^2 j}\sqrt{\tfrac{\log(\frac{1}{\beta})}{n}}) = \mathcal{O}(\tfrac{\Delta}{\log^2 j}). \tag{4.1}$$

Note that $\sum_{j\in 2^{[\log(1/\Delta)]}} \frac{1}{\log^2 j} \le \sum_i \frac{1}{i^2} = \mathcal{O}(1)$ and $\Delta = \beta\sqrt{(1/n)\cdot\log(1/\beta)}$. Hence Theorem 53 implies that $\bar{p}_{B'}$ estimates p to $\mathcal{A}_k$ distance $\mathcal{O}(\Delta)$.

Inequality (4.1) characterizes B' whose empirical distribution approximates the underlying distribution p in $\mathcal{A}_k$ distance. However, its definition involves the unknown p itself. It is naturally more convenient to work with inequalities that does not include p.

One attempt at such an inequality is

$$\max_{v\in\{0,1\}^{k\cdot j}_{2k}} \langle vv^{\intercal}, F^{j}_{B'}\rangle \le \mathcal{O}(\tfrac{\beta\log\frac{1}{\beta}}{n\cdot\log^4 j}) = \mathcal{O}(\tfrac{\Delta^2}{\beta\log^4 j}).$$

While under mild conditions this inequality can be shown to imply (4.1), it is still not easy to use as the set $\{vv^{\intercal} : v \in \{0,1\}^{k\cdot j}_{2k}\}$ is not convex, hence it is unclear how to efficiently optimize the left hand side.

To circumvent this difficulty, we define a semi-definite programing (SDP) relaxation of $\{vv^{\intercal} : v \in \{0,1\}^{k\cdot j}_{2k}\}$ as

$$\boldsymbol{R}^{j} := \{M \in \mathbb{R}^{k\cdot j\times k\cdot j} : M \succcurlyeq 0, M_{ii} \le 1, \textstyle\sum_i M_{ii} \le 2k\}.$$

This leads to the following B' inequality, $\forall\, j \in 2^{[\log(1/\Delta)]}$,

$$\max_{M\in\boldsymbol{R}^{j}} \langle M, F^{j}_{B'}\rangle \le \mathcal{O}(\tfrac{\beta\log\frac{1}{\beta}}{n\cdot\log^4 j}) = \mathcal{O}(\tfrac{\Delta^2}{\beta\log^4 j}). \tag{4.2}$$

Lemma 60 in the appendix shows that any B' with $|B_G \cap B'| \le (1-2\beta)|B_G|$ that satisfies this inequality also satisfies Inequality (4.1).

Next, we describe a filtering algorithm that finds $B' \subseteq B$ satisfying the new inequality.

To find such a batch sub-collection, we show that for all $B' \subseteq B$ such that $|B_G \cap B'| \leq (1-2\beta)|B_G|$ good batches, the following conditions hold:

1. There is a computationally efficient algorithm for finding $\mathrm{argmax}\{\langle M, F^j_{B'}\rangle : M \in \boldsymbol{R}^j\}$.
2. Given an M for which $\langle M, F^j_{B'}\rangle$ is large, we can delete batches from B' such that in expectation we delete 3 times more adversarial batches than good.
3. If B' has no adversarial batches, it satisfies (4.2).

The algorithm consists of a main part (Algorithm 5) that sequentially over $j \in 2^{[\log(1/\Delta)]}$ checks if Equation (4.2) is satisfied for partition $\mathcal{P}^j$. If not, it iteratively calls sub-routine Batch-Deletion (Algorithm 6), to delete the appropriate batches. Due to space limitations we present the pseudo code for Algorithm 5 and Algorithm 6 in the appendix. Next, we argue that the algorithm identifies B' for which (4.2) holds.

It starts with $B' = B$, and sequentially over $j \in 2^{[\log(1/\Delta)]}$, perform the following recursive algorithm. Efficiently find M maximizing $\langle M, F^j_{B'}\rangle$ (condition 1). Use M to delete batches $b \in B'$ for which $\langle M, C^j_{b,B'}\rangle$ is high. Continue until Equation (4.2) holds for j. As the algorithm proceeds, so long as Equation (4.2) fails to hold, Condition 2 ensures that the algorithm removes more adversarial batches than good batches (in expectation). Observe that without adversarial batches, Equation (4.2) holds. Hence, at the latest, when all adversarial batches are removed, the condition 3 ensures Equation (4.2) will hold and algorithm will stop. The second condition ensures that w.h.p. the algorithm does not remove more than more than $|B_A|/2 = \beta(1-\beta)B_G/2$, which for $\beta \leq 1/6$ is $\leq 2\beta B_G$ good batches, before removing all adversarial batches.

Hence in the end B' will satisfy Equation (4.2), and therefore Equation (4.1). The empirical distribution $\bar{p}_{B'}$ achieves the guarantee in Theorem 45.

In the appendix, we derive the above filtering conditions by using the following concentration properties of good batches.

Essential Properties of good batches: For all sub-collections $B'_G \subseteq B_G$ of good batches, $j \in 2^{[\log(1/\Delta)]}$, and $M \in \boldsymbol{R}^j$:

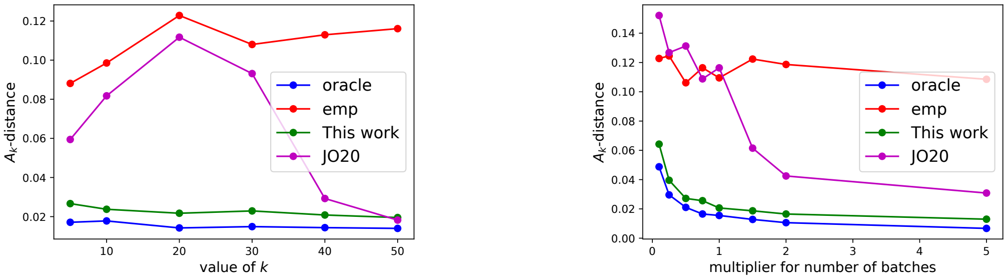
1. If $|B'_G| \geq (1-2\beta)|B_G|$, then

   (a) $\langle M, (\bar{p}^j_{B'_G} - p^j)^{\otimes 2}\rangle \leq \mathcal{O}\Big(\frac{\Delta^2}{\log^4 j}\Big)$,

   (b) $\langle M, F^j_{B'_G}\rangle \leq \mathcal{O}\Big(\frac{\Delta^2}{\beta \log^4 j}\Big)$.

2. If $|B'_G| \leq 2\beta|B_G|$, then

   $\sum_{b \in B'_G} \langle M, (\bar{\mu}^j_b - p^j)^{\otimes 2}\rangle \leq \mathcal{O}\Big(|B_G| \cdot \frac{\Delta^2}{\beta \log^4 j}\Big)$.



(a) $\mathcal{A}_k$ distance vs. k with constant no. of samples to k ratio

(b) $\mathcal{A}_k$ distance vs. number of samples (batches)

Figure 4.1. Learning distributions in $\mathcal{A}_k$ distance

The next theorem shows that w.h.p. the good batch collection B_G satisfies the above properties.

Theorem 56. *For some constants $c < 1/2$ and $C > 1$, for any k, $\beta < c$, $\delta < 1$, $n > \Omega(\log^C(1/\beta))$, and discrete or continuous p. If $|B_G| \cdot n = \tilde{\Omega}(\frac{k+\log(1/\delta)}{\Delta^2})$, then the essential properties hold with probability $\geq 1-\delta$.*

Crucially, the Theorem shows that for carefully chosen SDP relaxation $\boldsymbol{R}^j$ of the set of $2k$ sparse binary vectors, the filtering properties hold with only $\tilde{\Omega}(k)$ samples. By comparison, [31] used a convex relaxation of binary vectors that are sparse in Haar basis, and for that relaxation they showed $\tilde{\mathcal{O}}(k^2)$ sample complexity.

Let $\mathcal{L}^j_i : \{v \in \mathbb{R}^j : ||v||_\infty = 1, ||v||_2^2 \leq i\}$. The next theorem shows that to prove that the above properties hold for all elements in $\boldsymbol{R}^j$ it suffices to show that the property holds for the following strictly smaller set $\{vv^\intercal : v \in \mathcal{L}^{k \cdot j}_{2k}\}$.

Theorem 57. *Consider an $n \times n$ symmetric matrix A of real numbers. Then there is a universal constant $K_G \leq 1.7822$ such that*

$$\max_{M \in \boldsymbol{R}^j} |\langle M, A \rangle| \leq 2 \cdot K_G \max_{v \in \mathcal{L}_{2k}^{k \cdot j}} |\langle vv^\intercal, A \rangle|.$$

We derive the above theorem in Appendix 4.12 using Grothendieck's inequality.

The set $\{vv^\intercal : v \in \mathcal{L}_{2k}^{k \cdot j}\}$ is still infinite. Even its $o(1)$ cover can be shown to have size exponential in $\tilde{\Omega}(k \cdot j)$. Taking the union bound on the cover elements, as in the previous works, would yield only a sub-optimal $\max_j \tilde{\mathcal{O}}(k \cdot j/\Delta^2) = \tilde{\mathcal{O}}(k/\Delta^3)$ sample complexity. But applying a much more nuanced and complex technique, we obtain the optimal sample complexity $\tilde{\mathcal{O}}(k/\Delta^2)$. Due to space constraint we leave the details to Appendix 4.12 and 4.14.

4.5 Experiments

We corroborate our results by performing simulations.

We present here experiments for our main technical contribution, robustly learning arbitrary distributions to $\mathcal{A}_k$ distance using just $\mathcal{O}(k)$ samples, even when the domain size is much larger than k. The simulations for learning continuous distribution in TV distance are relegated to the appendix.

For discrete distributions we set the domain size ℓ to 500. We select this rather large value to show that the algorithm is practical for large domains, where exploiting the structure becomes more important.

We show two plots, for both we set the fraction of adversarial batches to a relatively high value $\beta = 0.4$ and the batch size to a moderate value of 500. This shows that the algorithms perform well even when corruption is high and batch size is only moderate. Note that the algorithm's performance will improve if we increase the batch size or decrease β.

We compare the performance of our algorithm with three other estimators. The first is a powerful oracle, who knows which batches are good batches and uses their empirical distribution as its estimate. The performance of Oracle shows the information theoretic limit in absence of adversarial batches. The second estimator is the standard empirical estimator that simply returns the empirical distribution of all samples in B. The third estimator is the [76] filtering-based estimator. We also considered the estimator of [31], however for the large domain size we test our algorithm on, the implementation of their algorithm provided with their paper took several hours even for a single run, while our estimator took on average less than three minutes.

The simulations were performed on a laptop with a configuration of 2.3 GHz Intel Core i7 CPU and 16 GB of RAM. We took the average of 10 runs to plot the results. For both plots we select p by generating a random vector in $[0, 1]^\ell$ and normalizing it. We tried various adversarial distribution: a randomly chosen distribution similar to p; a randomly generated k piecewise histogram; and their linear combination with p. For each estimator we plot the results for worst adversarial distribution.

In our first simulation we verify that our algorithm can learn large discrete distributions in $\mathcal{A}_k$ distance, with a number of samples only linear in k. We choose the a rather large alphabet size $\ell = 500$ and test for various values of k from $10, 20, 30, 40, 50$. For each k we choose the number of good batches to be k/β^2. Our plots show that the error achieved by our algorithm essentially remains the same as k increases, demonstrating the linear dependence of the sample complexity on k. Our algorithm nearly achieves the performance of the oracle that enjoys the best statistical guarantee, even for the non-adversarial setting. Note that results in $\mathcal{A}_k$ learning imply the other results.

In the second plot, we keep k constant and increase the number of good batches as fk/β^2, for factor $f = [0.01, 0.25, 0.5, 0.75, 1, 1.5, 2, 5]$.

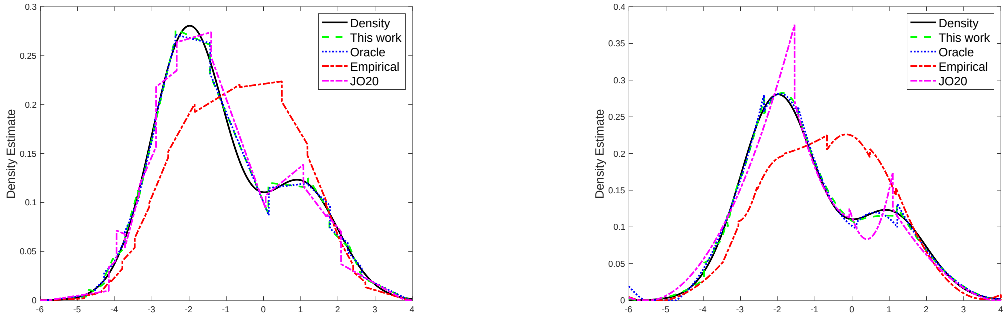
Appendix

4.6 Overview of supplementary material

The paper's main part motivated robust learning, described our major results, outlined some of their implications, and presented experiments confirming the theoretical results and showing that our algorithms accurately and rapidly recover discrete distributions. It relegated experiments for continuous distributions and almost all the proofs to this supplement.

Continuing where we left off, the next section describes experiments for continuous distributions, again showing excellent distribution reconstruction with relatively few batches. The rest, and bulk, of this supplement is devoted to proving results stated in the main paper. Section 4.8 restates essential properties of good batches that play a central role in establishing the filtering performance. Section 4.9 introduces two useful results that will be used in several parts of the proof. Section 4.10 describes the filtering algorithm and proves its correctness. Section 4.11 proves a few simple lemmas stated in the main paper. Section 4.12 shows that the essential properties follow from a concentration bound. Section 4.13 recalls some facts from VC theory and derives some of their implications. Section 4.14 is rather long and establishes the concentration bounds required to prove the essential properties.

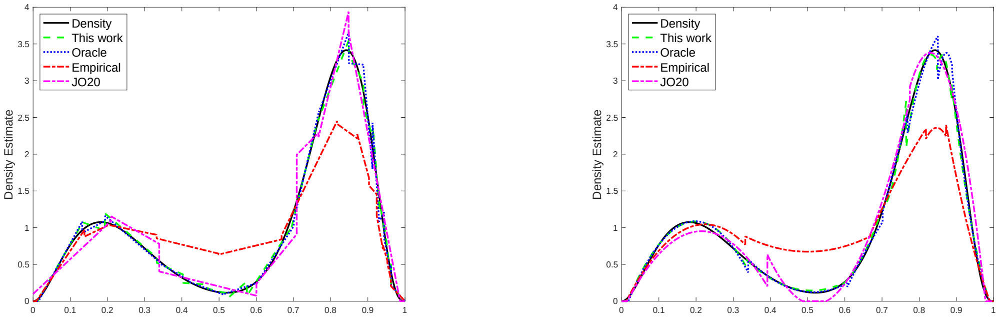
4.7 Experiments for continuous distributions



(a) Using piecewise-linear polynomials

(b) Using piecewise-degree 2 polynomials

Figure 4.2. Learning Gaussian mixture $0.7\mathcal{N}(-2,1) + 0.3\mathcal{N}(1,1)$ using different filters



(a) Using piecewise-linear polynomials **(b)** Using piecewise-degree 2 polynomials

Figure 4.3. Learning Beta mixture $0.7\,\text{Beta}(17,4) + 0.3\,\text{Beta}(3,10)$ using different filters

Table 4.1. TV error of estimating Gaussian and Beta mixtures using various batch filters, extrapolated by polynomials of degree 1 and 2

Distribution →	$0.7 \cdot \mathcal{N}(-2,1) + 0.3 \cdot \mathcal{N}(1,1)$		$0.7 \cdot \text{Beta}(17,4) + 0.3 \cdot \text{Beta}(3,10)$	
Running SURF on samples in ↓	degree 1	degree 2	degree 1	degree 2
All batches (Empirical)	0.1859	0.1863	0.1724	0.1730
Batches filtered using [76]	0.0724	0.0719	0.0573	0.0520
Batches filtered using this work	0.0223	0.0188	0.0218	0.0190
Good batches (Oracle)	0.0205	0.0188	0.0222	0.0202

We now show the algorithm's actual performance for recovering continuous distributions in the presence of adversarial batches. These are the first experiments performed for learning continuous distributions robustly using batches. To show the algorithm's efficacy, we applied it to two of the most common and practical continuous distributions, Gaussian mixtures and beta mixtures.

To estimate the distributions robustly, we first used our algorithm to filter out suspicious batches. In all experiment in this section we run our $\mathcal{A}_k$ filtering algorithm for $k = 10$. The algorithm does not need to know anything else about the underlying distribution.

We then used the remaining batches as input to the recent SURF Algorithm [67] for learning piecewise-polynomial distributions for genuine samples. SURF is an alternative to the algorithm in [1] and has the advantage that we need to specify only the degree d of the estimating polynomial, SURF then automatically finds the best number of pieces t.

We compared the results of our filtering algorithm to those obtained by the following filters. The naive *empirical* filter that keeps all batches, the *oracle* filter that knows the identity of the good batches and keeps only these batches, and the filtering algorithm in [76].

One can not run algorithm of [31] for continuous distributions. We tried combining it with the partition idea. However the size of the partition required is the power of 2 closest to $\frac{k\sqrt{n}}{\beta}$, and for this partition size the implementation of their algorithm provided with their paper does not terminate even after running for several hours, while our estimator took on average less than three minutes.

As for the discrete experiments in the main paper, we used adversarial fraction $\beta = 0.4$ and batch size $n = 500$. For Gaussian mixtures, we considered the distribution $0.7 \cdot \mathcal{N}(-2, 1) + 0.3 \cdot \mathcal{N}(1, 1)$ for the good batches and $\mathcal{N}(0, 1)$ for the adversarial batches. For Beta mixtures, we considered the distribution $0.7 \cdot \text{Beta}(17, 4) + 0.3 \cdot \text{Beta}(3, 10)$ for the good batches, and $\text{Beta}(2, 2)$ for the adversarial batches.

For both distributions, we ran both our algorithm and [76] to filter for $\mathcal{A}_k$ distance for the moderate value $k = 10$. From Theorem 45, our algorithm uses only $\lfloor \frac{k}{\beta^2} \rfloor = 62$ good batches, and $\lfloor \frac{62}{1-\beta} \rfloor = 104$ total batches. We ran SURF with degree-1 (linear) and degree-2 (quadratic) piecewise polynomials.

Each of Figures 4.2 and 4.3 plots the underlying distribution and its four estimates for a single run. Table 4.1 shows the TV-distance for the four estimators, each averaged over 10 runs.

As can be observed from the figures, the empirical filter that keeps all samples tries to estimate the mixture of the underlying and adversarial distributions and lands far from its target. The [76] filter has sub-optimal sample complexity and also misses the mark. Both the current estimator and the unrealizable oracle filters perform very well in all four cases, and are barely distinguishable from the underlying distributions.

Similar conclusions can be drawn from Table 4.1. Consider for example Gaussian distribution and degree 1 approximation. The empirical estimator achieves TV distance 0.1859, the [76] filter achieves TV distance 0.0724, the current filter 0.02223, and the oracle filter a slightly better 0.0205. Curiously, for the Beta distribution the current filter slightly outperforms the optimal oracle filter, in the third decimal place. We do not know whether that is because the estimator removes outlier batches, or the difference, because of the interaction with SURF, or if this tiny difference is within statistical tolerance.

Note that the oracle filter assumes knowledge of the good and adversarial batches, information that we do not assume to be available. As shown in [67] when combined with SURF it achieves the information theoretic limit for learning these densities (to log factors), showing that our algorithm essentially matches the best performance even with knowledge of the adversarial batches.

4.8 Essential properties of good batches

In this short section, we restate important properties that hold for large collections of good batches that play an essential role in the analysis of the filtering algorithm.

Let $v^{\otimes 2}$ denote the outer product $vv^\intercal$ of a vector v with itself.

Theorem 56 (Essential properties of good batches). *For some constants $c < 1/2$ and $C > 1$, for any k, $\beta < c$, $\delta < 1$, $n > \Omega(\log^C(1/\beta))$, and discrete or continuous p. If $|B_G| \cdot n = \tilde{\Omega}(\frac{k+\log(1/\delta)}{\Delta^2})$, then with probability $\geq 1 - \delta$, the following properties hold for all sub-collections $B'_G \subseteq B_G$ of good batches, $j \in 2^{[\log(1/\Delta)]}$, and $M \in \boldsymbol{R}^j$:*

1. *If $|B'_G| \geq (1 - 2\beta)|B_G|$, then*
   (a) $\langle M, (\bar{p}^j_{B'_G} - p^j)^{\otimes 2} \rangle \leq \mathcal{O}\left(\dfrac{\Delta^2}{\log^4 j}\right)$,
   (b) $\langle M, F^j_{B'_G} \rangle \leq \mathcal{O}\left(\dfrac{\Delta^2}{\beta \log^4 j}\right)$.
2. *If $|B'_G| \leq 2\beta|B_G|$, then*

$$|B'_G|\langle M, (\bar{p}^j_{B'_G} - p^j)^{\otimes 2}\rangle \leq \sum_{b \in B'_G} \langle M, (\bar{\mu}^j_b - p^j)^{\otimes 2}\rangle \leq \mathcal{O}\left(|B_G| \cdot \frac{\Delta^2}{\beta \log^4 j}\right).$$

4.9 Two useful results

The following lemmas help prove the essential properties of the good batches and establish the filtering conditions.

Recall that $v^{\otimes 2} := vv^\intercal$ denotes the outer product of a vector v with itself, and that for any discrete distribution q, the covariance matrix of the normalized mutational distribution $\text{Mul}^N(q, n)$ is

$$C(q) = \frac{1}{n}(\text{Diag}(q) - q^{\otimes 2}).$$

The following result bounds the change in $C(q)$ as q changes. The symbol $\odot$ denotes element-wise product of two matrices or vectors.

Lemma 58. *For any $j \geq 0$, pair of discrete distributions $q, \hat{q} \in \mathbb{R}^j$, and collection of vectors $U \subseteq \{u \in \mathbb{R}^j : ||u||_\infty \leq 1\}$,*

$$\max_{u,v \in U} \langle uv^\intercal, C(q) - C(\hat{q})\rangle \leq \frac{3}{n} \max_{u \in U, w \in \{-1,1\}^j} |(u \odot w) \cdot (q - \hat{q})|.$$

Proof. Note that for any vectors u, v and w

$$\langle uv^\intercal, w^{\otimes 2}\rangle = (u \cdot w)(v \cdot w).$$

Then, for any $u, v \in U$ and distributions $q, q' \in \mathbb{R}^j$

$$\begin{aligned}\langle uv^\intercal, q^{\otimes 2} - \hat{q}^{\otimes 2}\rangle &= (u \cdot q)(v \cdot q) - (u \cdot \hat{q})(v \cdot \hat{q}) \\ &= (u \cdot q)(v \cdot (q - \hat{q})) + (v \cdot \hat{q})(u \cdot (q - \hat{q})) \\ &\leq (||u||_\infty ||q||_1) \cdot \Big| v \cdot (q - \hat{q})\Big| + (||v||_\infty ||\hat{q}||_1) \cdot \Big| u \cdot (q - \hat{q})\Big| \\ &\leq \Big| v \cdot (q - \hat{q})\Big| + \Big| u \cdot (q - \hat{q})\Big| \\ &\leq 2 \max_{u \in U} \Big| u \cdot (q - \hat{q})\Big|,\end{aligned}$$

where the last inequality used the facts that $||u||_\infty, ||v||_\infty \leq 1$ and for any pair of distributions $||q + \hat{q}||_1 = 2$.

Hence for any $u, v \in U$,

$$\begin{aligned}\langle uv^\intercal, C(q) - C(\hat{q})\rangle &= \frac{1}{n}\langle uv^\intercal, \text{Diag}(q - \hat{q}) - q^{\otimes 2} + \hat{q}^{\otimes 2}\rangle \\ &\leq \frac{1}{n}|\langle uv^\intercal, \text{Diag}(q - \hat{q})\rangle| + \frac{1}{n}|\langle uv^\intercal, q^{\otimes 2} - \hat{q}^{\otimes 2}\rangle| \\ &= \frac{1}{n}\Big|\sum_{i=1}^{j} u_i \cdot v_i \cdot (q_i - \hat{q}_i)\Big| + \frac{2}{n}\max_{u \in U}|u \cdot (q - \hat{q})| \\ &\leq \frac{1}{n}\Big|\sum_{i=1}^{j} u_i \cdot |q_i - \hat{q}_i|\Big| + \frac{2}{n}\max_{u \in U}|u \cdot (q - \hat{q})| \\ &= \frac{1}{n}\max_{w \in \{-1,1\}^j} |(u \odot w) \cdot (q - \hat{q})| + \frac{2}{n}\max_{u \in U}|u \cdot (q - \hat{q})|,\end{aligned}$$

where the second inequality uses the fact that $||v||_\infty \leq 1$ and the last equality follows by letting $w_i = \text{sign}(q_i - \hat{q}_i)$ and $w = (w_1, \ldots, w_j)$. The statement of lemma follows. ■

Generalizing a familiar result for scalars, the next lemma decomposes the "squared distance" of a set of vectors from some given vector to their squared distance from their mean and the squared distance between their mean and the given vector.

Lemma 59. *Let $\hat{\mu} = \frac{1}{t}\sum_{i=1}^{t} \mu_i$ be the average of $\mu_1, \ldots, \mu_t \in \mathbb{R}^j$. Then for any $\mu \in \mathbb{R}^j$,*

$$\sum_{i=1}^{t} (\mu_i - \mu)^{\otimes 2} = \sum_{i=1}^{t} (\mu_i - \hat{\mu})^{\otimes 2} + t \cdot (\hat{\mu} - \mu)^{\otimes 2}.$$

Proof.

$$\begin{aligned}
\sum_{i=1}^{t} (\mu_i - \mu)(\mu_i - \mu)^{\intercal} &= \sum_{i=1}^{t} (\mu_i - \hat{\mu} + \hat{\mu} - \mu)^{\otimes 2} \\
&= \sum_{i=1}^{t} \Big((\mu_i - \hat{\mu})^{\otimes 2} + 2(\hat{\mu} - \mu)(\mu_i - \hat{\mu})^{\intercal} + (\hat{\mu} - \mu)^{\otimes 2} \Big) \\
&= \sum_{i=1}^{t} (\mu_i - \hat{\mu})^{\otimes 2} + t \cdot (\hat{\mu} - \mu)^{\otimes 2} + 2(\hat{\mu} - \mu) \Big(\sum_{i=1}^{t} \mu_i - t \cdot \hat{\mu} \Big)^{\intercal} \\
&= \sum_{i=1}^{t} (\mu_i - \hat{\mu})^{\otimes 2} + t \cdot (\hat{\mu} - \mu)^{\otimes 2} + 2(\hat{\mu} - \mu)(t \cdot \hat{\mu} - t \cdot \hat{\mu})^{\intercal} \\
&= \sum_{i=1}^{t} (\mu_i - \hat{\mu})^{\otimes 2} + t \cdot (\hat{\mu} - \mu)^{\otimes 2}. \qquad \blacksquare
\end{aligned}$$

4.10 The filtering algorithm and its analysis

We present the filtering algorithm, explain how it works, and prove its correctness.

The algorithm consists of a main part (Algorithm 5) that sequentially over $j \in 2^{[\log(1/\Delta)]}$ checks if Equation (4.2) is satisfied for partition $\mathcal{P}^j$. If not, it iteratively calls sub-routine Batch-Deletion (Algorithm 6), to delete the appropriate batches.

As stated in the main paper, Theorem 53 implies that any distribution satisfying Equation (4.1) learns p to the desired $\mathcal{A}_k$ distance $\mathcal{O}(\Delta)$. Hence to prove the algorithm's correctness it suffices to show that it returns a batch sub-collection whose empirical distribution satisfies Equation (4.1).

The next technical lemma, proved in the next subsection, shows that if a batch sub-collection $B' \subseteq B$ retains at least a $(1 - 2\beta)$ fraction of all good batches in B, and its empirical distribution satisfies Equation (4.2), then it also satisfies Equation (4.1).

Algorithm 5. Robust Estimator in $\mathcal{A}_k$ distance

```
Input: Batch size $n$, $\beta$, $k$, and Batch collection $B$ such that an unknown $B_G \subseteq B$ of size $\geq (1-\beta)|B|$ satisfy essential properties.
Output: $B' \subseteq B$ such that $||p - \bar{p}_{B'}||_{\mathcal{A}_k} \leq \mathcal{O}(\Delta)$.
$B' \leftarrow B$ ;
for $j \in 2^{[\log(1/\Delta)]}$ {Running for partition size $k \cdot j$.} do
    while $\max_{M \in \boldsymbol{R}^j} |\langle M, F^j_{B'} \rangle| \geq \Omega(\frac{\beta \log \frac{1}{\beta}}{n \cdot \log^4 j})$ {Checking if Equation (4.2) holds for $j$} do
        $M^* = \underset{M \in \boldsymbol{R}^j}{\operatorname{argmax}} \langle M, F^j_{B'} \rangle$ ;
        $\forall\, b \in B'$ calculate $\xi_b = |\langle M^*, C^j_{b,B'} \rangle|$ ;
        $B^o = \{$ Batches with top $\beta|B|$ scores$\}$ ;
        {Suspect batches with top $\beta|B|$ scores as the possible outliers.}
        $B_{del} \leftarrow$Batch-Deletion($B^0$, $\{\xi_b\}$) ;
        $B' \leftarrow B' \setminus B_{del}$ ;
    end while
end for
return $(B' \leftarrow B')$.
```

Algorithm 6. Batch-Deletion

```
Input: Sub-Collection $B^0$ of suspected outliers, and the scores $\{\xi_b\}$ for all $b \in B^0$.
Output: Batches $B_{del} \subseteq B^0$ to delete.
$B_{del} = \{\}$, $\xi_{total} = \sum_{b \in B^0} \xi_b$ and $\xi_{del} = 0$ ;
while $\xi_{del} \leq 0.5 \cdot \xi_{total}$ do
    Samples batch $b \in B^0$ such that probability of picking a batch $b \in B^0$ is $\propto \xi_b$;
    $\xi_{del} \leftarrow \xi_{del} + \xi_b$;
    $B^0 \leftarrow B^0 \setminus \{b\}$ and $B_{del} \leftarrow B_{del} \cup \{b\}$ ;
end while
return $(B_{del})$;
```

Lemma 60. *If the essential properties of good batches hold, then* $\forall\, j \in 2^{[\log(1/\Delta)]}$, $\forall\, B'$ *s.t.* $|B' \cap B_G| \geq (1-2\beta)|B_G|$,

$$\max_{v \in \{0,1\}^{k \cdot j} : ||v||_0 \leq 2k} |\bar{p}^j_{B'} \cdot v - p^j \cdot v| \leq \mathcal{O}\left(\frac{\Delta}{\log^2 j} + \sqrt{\beta \cdot \max_{M \in \boldsymbol{R}^j} \langle M, F^j_{B'} \rangle}\right).$$

In the main paper we argued that one can use filtering to find a batch sub-collection B' achieving Equation (4.2) if for all $B' \subseteq B$ such that $|B_G \setminus B'| \leq 2\beta \cdot |B_G|$, the following conditions hold:

1. There is a computationally efficient algorithm for finding $M^* := \operatorname{argmax}\{\langle M, F^j_{B'}\rangle : M \in \boldsymbol{R}^j\}$.

2. If Equation (4.2) do not hold for B', then using M^* we can delete batches from B' such that in expectation we delete 3 times more adversarial batches than good.

3. If B' has no adversarial batches, it satisfies Equation (4.2).

For completeness we first repeat the argument. Suppose that the three condition hold. To find the collection B' achieving Equation (4.2), start with $B' = B$, and sequentially over $j \in 2^{[\log(1/\Delta)]}$, perform the following recursive algorithm. Efficiently find M maximizing $\langle M, F^j_{B'}\rangle$ (condition 1). Use M to delete batches $b \in B'$ for which $\langle M, C^j_{b,B'}\rangle$ is high. Continue until Equation (4.2) holds for j. As the algorithm proceeds, so long as Equation (4.2) fails to hold, Condition 2 ensures that in expectation we remove more adversarial batches than good batches. Observe that without adversarial batches, Equation (4.2) holds. Hence, at the latest, when all adversarial batches are removed, Condition 3 ensures that Equation (4.2) will hold and the algorithm will stop. The second condition ensures that w.h.p. the algorithm do not remove more than more than $|B_A|/2 = \beta(1-\beta)B_G/2 \le 2\beta B_G$ good batches before removing all adversarial batches.

Next we show that with high probability the three condition hold for any $B' \subseteq B$ such that $|B_G \setminus B'| \le 2\beta \cdot |B_G|$.

The first filtering condition always holds since the set $\boldsymbol{R}^j$ is convex, hence for any matrix A one can solve the optimization problem $\operatorname{argmax}\{\langle M, A\rangle : M \in \boldsymbol{R}^j\}$ efficiently.

The third condition holds with high probability as it is essential property 1b of good batches.

To establish the second condition, we now describe a corresponding Batch-Deletion procedure. Its Pseudo-code is presented in Algorithm 6 at the end of this section.

Suppose Equation (4.2) do not hold for a batch sub collection B' such that $|B' \cap B_G| \ge$

$(1-2\beta)|B_G|$. Given $M^* = \operatorname*{argmax}_{M \in \boldsymbol{R}^j} \langle M, F^j_{B'} \rangle$, compute the *score* $\langle M^*, C^j_{b,B'} \rangle$ of each batch $b \in B'$. Let B^o be the sub collection of $\beta|B|$ batches with highest scores in B'.

We delete batches in B^o with probability proportional to their score until total score of the remaining batches is half the initial total score of all batches in B^0.

Since B' has at most $\beta|B|$, adversarial batches, and B^o has the $\beta|B|$ batches with highest scores in B', the total score of batches in B^o is at least as large as the total scores of adversarial batches in B'.

Lemma 61 shows that if Equation (4.2) does not hold, then the ratio of the total score of any collection of $\beta|B|$ good batches to the total score of all adversarial batches in B' is $\leq \frac{1}{8}$.

Lemma 61. *There exists an absolute constant c such that if the essential properties of good batches hold for B_G, then $\forall\, j \in 2^{[\log(1/\Delta)]}$ and $\forall\, B'$ s.t. $|B' \cap B_G| \geq (1-2\beta)|B_G|$, if $\max_{M \in \boldsymbol{R}^j} \langle M, F^j_{B'} \rangle \geq c\frac{\Delta^2}{\beta \log^4 j}$, then $\forall B''_G \subseteq B_G \cap B'$ s.t. $|B''_G| \leq \beta|B|$, $M^* := \operatorname*{argmax}_{M \in \boldsymbol{R}^j} \langle M, F^j_{B'} \rangle$ satisfies*

$$\sum_{b \in B''_G} \langle M^*, C^j_{b,B'} \rangle \leq \frac{1}{8} \sum_{b \in B' \cap B_A} \langle M^*, C^j_{b,B'} \rangle.$$

Together these results imply that the score of all good batches in B^o is at most $\frac{1}{8}$ of the total score of all batches in B^o. Recall that sub-routine Batch Deletion starts with B^0 that contains the $\beta|B|$ batches with highest scores, and then removes batches with probability proportional to their score, until it is left with batches with half the initial total score. It follows that at any point the ratio of total scores of good batches in B^0 to scores of all batches in B^0 always remains below $1/4$. Therefore, in each deletion step the probability of deleting a good batch is $\leq 1/4$. And the second property follows. This completes the proof of the correctness of the Algorithm and shows that it learns p to $\mathcal{A}_k$ distance $\mathcal{O}(\Delta)$.

4.10.1 Proofs of Lemmas 60 and 61

To prove the two lemmas we first derive some useful relations and establish a preliminary technical lemma.

For any β less than some small absolute constant. Let B' be any batch sub-collection such that $|B' \cap B_G| \le (1-2\beta)|B_G|$. For the purpose of this section, let $B'_G = B' \cap B_G$ and $B'_A = B' \cap B_A$. Then for $\beta \le 1/6$,

$$|B'| \ge |B'_G| \ge (1-2\beta)B_G \ge (1-2\beta)(1-\beta)|B| \ge \frac{|B|}{2}, \tag{4.3}$$

and

$$|B'_A| \le |B_A| \le \beta|B| \le 2\beta|B'_G|. \tag{4.4}$$

Note that

$$\begin{aligned}\bar{p}^j_{B'} &= \frac{1}{|B'|}\sum_{b\in B'}\bar{\mu}^j_b = \frac{1}{|B'|}\sum_{b\in B'_G}\bar{\mu}^j_b + \frac{1}{|B'|}\sum_{b\in B'_A}\bar{\mu}^j_b \\ &= \frac{|B'_G|}{|B'|}\bar{p}^j_{B'_G} + \frac{|B'_A|}{|B'|}\bar{p}^j_{B'_A}.\end{aligned}$$

Hence

$$|B'_G|(\bar{p}^j_{B'_G} - \bar{p}^j_{B'}) = -|B'_A|(\bar{p}^j_{B'_A} - \bar{p}^j_{B'}).$$

Next,

$$\sum_{b\in B'_A} C^j_{b,B'} - \sum_{b\in B'_A} C^j_{b,B'_A} = \sum_{b\in B'_A} (\bar{\mu}^j_b - \bar{p}^j_{B'})^{\otimes 2} - \sum_{b\in B'_A} (\bar{\mu}^j_b - \bar{p}^j_{B'_A})^{\otimes 2}$$
$$= |B'_A|(\bar{p}^j_{B'_A} - \bar{p}^j_{B'})^{\otimes 2}$$
$$= \frac{|B'_G|^2}{|B'_A|}(\bar{p}^j_{B'_G} - \bar{p}^j_{B'})^{\otimes 2}, \tag{4.5}$$

where the second last step uses Lemma 59 and the last equation follows from the previous equation.

Similarly,

$$\sum_{b\in B'_G} C^j_{b,B'} - \sum_{b\in B'_G} C^j_{b,B'_G} = \sum_{b\in B'_G} (\bar{\mu}^j_b - \bar{p}^j_{B'})^{\otimes 2} - \sum_{b\in B'_G} (\bar{\mu}^j_b - \bar{p}^j_{B'_G})^{\otimes 2}$$
$$= |B'_G|(\bar{p}^j_{B'_G} - \bar{p}^j_{B'})^{\otimes 2}. \tag{4.6}$$

In addition to the essential properties of good batches, we will need the following simple consequence of our partition scheme. It will be convenient to prove this lemma along with the essential properties.

Lemma 62. *For any batch sub-collection $B' \subseteq B$ of size $|B'| \geq |B|/2$, and for any $j \geq 1$ the following bound holds*

$$\max_{M\in \boldsymbol{R}^j} \langle M, C(\bar{p}_{B'})\rangle \leq \mathcal{O}\left(\frac{1}{jn}\right).$$

The following technical lemma will be useful in several proofs.

Lemma 63. *If the essential properties of good batches hold, then $\forall\, j \in 2^{[\log(1/\Delta)]}$, $\forall\, B'$ s.t. for $B'_G = B' \cap B_G$ $|B'_G| \geq (1-2\beta)|B_G|$,*

$$\max_{M\in \boldsymbol{R}^j} \langle M, (\bar{p}^j_{B'_G} - \bar{p}^j_{B'})^{\otimes 2}\rangle \leq 8\beta \max_{M\in \boldsymbol{R}^j} \langle M, F^j_{B'}\rangle + \mathcal{O}\left(\frac{\Delta^2}{\log^4 j}\right).$$

Proof. Recall that $F^j_{B'} = \frac{1}{|B'|}\sum_{b\in B'} C^j_{b,B'} - C(\bar{p}^j_{B'})$. Let $B'_A = B' \setminus B'_G$. We use the essential properties to lower bound $F^j_{B'}$. Let

$$\widetilde{M} := \operatorname*{argmax}_{M\in \boldsymbol{R}^j} \langle M, (\bar{p}^j_{B'_G} - \bar{p}^j_{B'})^{\otimes 2}\rangle \quad \text{and} \quad \tau := \frac{\langle \widetilde{M}, (\bar{p}^j_{B'_G} - \bar{p}^j_{B'})^{\otimes 2}\rangle}{\Delta^2/\log^4 j}.$$

Then

$$\max_{M\in \boldsymbol{R}^j} \langle M, F^j_{B'}\rangle = \frac{1}{|B'|}\max_{M\in \boldsymbol{R}^j}\sum_{b\in B'}\langle M, C^j_{b,B'} - C(\bar{p}^j_{B'})\rangle \geq \frac{1}{|B'|}\sum_{b\in B'}\langle \widetilde{M}, C^j_{b,B'} - C(\bar{p}^j_{B'})\rangle.$$

For any $M \in \boldsymbol{R}^j$,

$$\begin{aligned}
&\sum_{b\in B'}\langle M, C^j_{b,B'} - C(\bar{p}^j_{B'})\rangle \\
&= \sum_{b\in B'_G}\langle M, C^j_{b,B'} - C(\bar{p}^j_{B'})\rangle + \sum_{b\in B'_A}\langle M, C^j_{b,B'} - C(\bar{p}^j_{B'})\rangle \\
&= \sum_{b\in B'_G}\langle M, C^j_{b,B'_G} - C(\bar{p}^j_{B'_G})\rangle + \sum_{b\in B'_G}\langle M, C^j_{b,B'} - C^j_{b,B'_G}\rangle \\
&+ |B'_G|\langle M, C(\bar{p}^j_{B'_G}) - C(\bar{p}^j_{B'})\rangle + \sum_{b\in B'_A}\langle M, C^j_{b,B'}\rangle - |B'_A|\langle M, C(\bar{p}^j_{B'})\rangle.
\end{aligned}$$

We individually bound each of the five terms on the right. The bound on the first term is implied by essential property 1b

$$|\sum_{b\in B'_G}\langle M, C^j_{b,B'_G} - C(\bar{p}^j_{B'_G})\rangle| = |B'_G||\langle M, F^j_{B'_G}\rangle| \leq |B'_G|\max_{M\in \boldsymbol{R}^j}|\langle M, F^j_{B'_G}\rangle| \leq |B'_G|\mathcal{O}(\tfrac{\Delta^2}{\beta\log^4 j}).$$

The second term is evaluated exactly using Equation (4.6),

$$\sum_{b\in B'_G}\langle M, C^j_{b,B'} - C^j_{b,B'_G}\rangle = |B'_G|\langle M, (\bar{p}^j_{B'_G} - \bar{p}^j_{B'})^{\otimes 2}\rangle.$$

For the third term,

$$\begin{aligned}
|B'_G||\langle M, C(\bar{p}^j_{B'_G}) - C(\bar{p}^j_{B'})\rangle| &\overset{(a)}{\leq} |B'_G| \cdot 2K_G \cdot \max_{u \in \mathcal{L}^{k\cdot j}_{2k}} |\langle u^{\otimes 2}, C(\bar{p}^j_{B'_G}) - C(\bar{p}^j_{B'})\rangle| \\
&\overset{(b)}{\leq} |B'_G| K_G \cdot \frac{6}{n} \max_{u \in \mathcal{L}^{k\cdot j}_{2k}} |u \cdot (\bar{p}^j_{B'_G} - \bar{p}^j_{B'})| \\
&= |B'_G| K_G \cdot \frac{6}{n} \max_{u \in \mathcal{L}^{k\cdot j}_{2k}} \sqrt{\langle u^{\otimes 2}, (\bar{p}^j_{B'_G} - \bar{p}^j_{B'})^{\otimes 2}\rangle} \\
&\overset{(c)}{\leq} |B'_G| K_G \cdot \frac{6}{n} \max_{M \in \boldsymbol{R}^j} \sqrt{\langle M, (\bar{p}^j_{B'_G} - \bar{p}^j_{B'})^{\otimes 2}\rangle} \\
&= |B'_G| K_G \cdot \frac{6}{n} \sqrt{\tau \frac{\Delta^2}{\log^4 j}} \\
&= 6K_G \cdot \sqrt{\tau} |B'_G| \frac{\Delta^2}{\beta \log^4 j} \cdot \frac{\beta \log^2 j}{n \cdot \Delta} \\
&\overset{(d)}{\leq} 6K_G \cdot \sqrt{\tau} |B'_G| \frac{\Delta^2}{\beta \log^4 j} \cdot \frac{\sqrt{\log^2 j}}{\sqrt{n \log(1/\beta)}} \\
&\overset{(e)}{=} 6K_G \cdot \sqrt{\tau} |B'_G| \frac{\Delta^2}{\beta \log^4 j}, \qquad (4.7)
\end{aligned}$$

where (a) uses Theorem 57, (b) uses Lemma 58 and the observation that $u \in \mathcal{L}^{k\cdot j}_{2k}$ and $w \in \{-1, 1\}^{k\cdot j}$ implies $u \odot w \in \mathcal{L}^{k\cdot j}_{2k}$, (c) follows as for all $u \in \mathcal{L}^{k\cdot j}_{2k}$, $uu^\intercal \in \boldsymbol{R}^j$, (d) uses $\Delta = \frac{\beta\sqrt{\log(1/\beta)}}{\sqrt{n}}$, and finally (e) uses $j \leq \frac{1}{\Delta}$ and $n \geq \log^4 \frac{1}{\Delta}$.

For the fourth term, using Equation (4.5),

$$\begin{aligned}
\sum_{b \in B'_A} \langle M, C^j_{b,B'}\rangle &= \sum_{b \in B'_A} \langle M, C^j_{b,B'_A}\rangle + \frac{|B'_G|^2}{|B'_A|} \langle M, (\bar{p}^j_{B'_G} - \bar{p}^j_{B'})^{\otimes 2}\rangle \\
&\geq \frac{|B'_G|^2}{|B'_A|} \langle M, (\bar{p}^j_{B'_G} - \bar{p}^j_{B'})^{\otimes 2}\rangle \\
&\geq \frac{|B'_G|}{2\beta} \langle M, (\bar{p}^j_{B'_G} - \bar{p}^j_{B'})^{\otimes 2}\rangle, \qquad (4.8)
\end{aligned}$$

where the first inequality uses the fact that M and the C^j_{b,B'_A}'s are PSD matrices and the second inequality uses Inequality (4.4).

Finally the last term is bounded using Lemma 62

$$|B'_A||\langle M, C(\bar{p}^j_{B'})\rangle| \leq |B'_A|\mathcal{O}(\frac{1}{jn}) \leq \mathcal{O}(\beta|B|\frac{1}{jn}) \leq \mathcal{O}(|B'_G|\frac{\Delta^2}{\beta\log^4 j}),$$

where in the last inequality we used $|B'_G| \geq \frac{|B|}{2}$, and $\frac{\Delta^2}{\beta\log^2 j} = \frac{\beta\log(1/\beta)}{n\log^2 j} \geq \frac{\beta}{n\cdot j}$.

Combining the bounds on all five terms,

$$\begin{aligned}
&\sum_{b\in B'}\langle M, C^j_{b,B'} - C(\bar{p}^j_{B'})\rangle \\
&\geq -\mathcal{O}(|B'_G|\frac{\Delta^2}{\log^4 j}) + |B'_G|\langle M, (\bar{p}^j_{B'_G} - \bar{p}^j_{B'})^{\otimes 2}\rangle - 6K_G\cdot\sqrt{\tau}|B'_G|\frac{\Delta^2}{\beta\log^4 j} \\
&+ \frac{|B'_G|}{2\beta}\langle M, (\bar{p}^j_{B'_G} - \bar{p}^j_{B'})^{\otimes 2}\rangle - \mathcal{O}(|B'_G|\frac{\Delta^2}{\beta\log^4 j}).
\end{aligned} \tag{4.9}$$

Choosing $M = \widetilde{M}$,

$$\begin{aligned}
\sum_{b\in B'}\langle \widetilde{M}, C^j_{b,B'} - C(\bar{p}^j_{B'})\rangle &\geq -\mathcal{O}(|B'_G|\frac{\Delta^2}{\log^4 j}) + \tau|B'_G|\frac{\Delta^2}{\log^4 j} - 6K_G\cdot\sqrt{\tau}|B'_G|\frac{\Delta^2}{\beta\log^4 j} \\
&+ \frac{1}{2\beta}\tau|B'_G|\frac{\Delta^2}{\log^4 j} - \mathcal{O}(|B'_G|\frac{\Delta^2}{\beta\log^4 j}) \\
&\geq |B'_G|\frac{\Delta^2}{\beta\cdot\log^4 j}(\frac{\tau}{2} + \beta\tau - 6K_G\cdot\sqrt{\tau} - \mathcal{O}(\beta+1)) \\
&\geq |B'_G|\frac{\Delta^2}{\beta\cdot\log^4 j}(\frac{\tau}{4} - \mathcal{O}(1)).
\end{aligned}$$

Therefore,

$$\begin{aligned}
&\max_{M\in\boldsymbol{R}^j}\langle M, F^j_{B'}\rangle \ge \frac{1}{|B'|}\sum_{b\in B'}\langle \widetilde{M}, C^j_{b,B'} - C(\bar{p}^j_{B'})\rangle\\
&\ge \frac{|B'_G|}{|B'|}\frac{\Delta^2}{\beta\cdot\log^4 j}(\frac{\tau}{4} - \mathcal{O}(1))\\
&= \frac{|B'_G|}{|B'|}\frac{\Delta^2}{\beta\cdot\log^4 j}\Big(\frac{\langle \widetilde{M}, (\bar{p}^j_{B'_G} - \bar{p}^j_{B'})^{\otimes 2}\rangle}{4\Delta^2/\log^4 j} - \mathcal{O}(1)\Big)\\
&= \frac{|B'_G|}{|B'|}\frac{\langle \widetilde{M}, (\bar{p}^j_{B'_G} - \bar{p}^j_{B'})^{\otimes 2}\rangle}{4\beta} - \mathcal{O}\Big(\frac{|B'_G|}{|B'|}\frac{\Delta^2}{\beta\cdot\log^4 j}\Big)\\
&\ge \frac{\langle \widetilde{M}, (\bar{p}^j_{B'_G} - \bar{p}^j_{B'})^{\otimes 2}\rangle}{8\beta} - \mathcal{O}\Big(\frac{\Delta^2}{\beta\cdot\log^4 j}\Big),
\end{aligned}$$

where the last step uses $|B'_G| \ge 2|B| \ge 2|B'|$ and $|B'_G| \le |B'|$.

Hence,

$$\max_{M\in\boldsymbol{R}^j}\langle M, (\bar{p}^j_{B'_G} - \bar{p}^j_{B'})^{\otimes 2}\rangle \le 8\beta\max_{M\in\boldsymbol{R}^j}\langle M, F^j_{B'}\rangle + \mathcal{O}\Big(\frac{\Delta^2}{\log^4 j}\Big). \qquad \blacksquare$$

We now prove Lemmas 60 and 61. For completness, we repeat their statements.

Lemma 60. *If the essential properties of good batches hold, then* $\forall\, j \in 2^{[\log(1/\Delta)]}$, $\forall\, B'$ *s.t.* $|B'\cap B_G| \ge (1-\mathcal{O}(\beta))|B_G|$,

$$\max_{v\in\{0,1\}^{k\cdot j}:||v||_0\le 2k}|\bar{p}^j_{B'}\cdot v - p^j\cdot v| \le \mathcal{O}\left(\frac{\Delta}{\log^2 j} + \sqrt{\beta\cdot\max_{M\in\boldsymbol{R}^j}\langle M, F^j_{B'}\rangle}\right).$$

Proof. From Lemma 63,

$$\max_{M\in\boldsymbol{R}^j}\langle M, (\bar{p}^j_{B'_G} - \bar{p}^j_{B'})^{\otimes 2}\rangle \le 8\beta\max_{M\in\boldsymbol{R}^j}\langle M, F^j_{B'}\rangle + \mathcal{O}\Big(\frac{\Delta^2}{\log^4 j}\Big).$$

Next,

$$\begin{aligned}\max_{M\in\boldsymbol{R}^j}\langle M,(p^j-\bar{p}^j_{B'})^{\otimes 2}\rangle &= \max_{M\in\boldsymbol{R}^j}\langle M,(p^j-\bar{p}^j_{B'_G}+\bar{p}^j_{B'_G}-\bar{p}^j_{B'})^{\otimes 2}\rangle\\ &\le 2\max_{M\in\boldsymbol{R}^j}\langle M,(\bar{p}^j_{B'_G}-\bar{p}^j_{B'})^{\otimes 2}\rangle+2\max_{M\in\boldsymbol{R}^j}\langle M,(p^j-\bar{p}^j_{B'_G})^{\otimes 2}\rangle\\ &\le 16\beta\max_{M\in\boldsymbol{R}^j}\langle M,F^j_{B'}\rangle+\mathcal{O}\Big(\frac{\Delta^2}{\log^4 j}\Big),\end{aligned}$$

here the last step uses the previous equation and essential property 1a.

Next note that if $v\in\{0,1\}^{k\cdot j}_{2k}$ then $v^{\otimes 2}\in\boldsymbol{R}^j$. Hence,

$$\begin{aligned}\max_{v\in\{0,1\}^{k\cdot j}_{2k}}(v\cdot(p-\bar{p}^j_{B'}))^2 &= \max_{v\in\{0,1\}^{k\cdot j}_{2k}}\langle v^{\otimes 2},(p-\bar{p}^j_{B'})^{\otimes 2}\rangle\\ &\le \max_{M\in\boldsymbol{R}^j}\langle M,(p-\bar{p}^j_{B'})^{\otimes 2}\rangle\\ &\le 16\beta\max_{M\in\boldsymbol{R}^j}\langle M,F^j_{B'}\rangle+\mathcal{O}\Big(\frac{\Delta^2}{\log^4 j}\Big).\end{aligned}$$

The lemma follows by taking square roots. ■

Lemma 61. *There exists an absolute constant c such that if the essential properties of good batches hold for B_G, then $\forall\, j\in 2^{[\log(1/\Delta)]}$ and $\forall\, B'$ s.t. $|B'\cap B_G|\ge(1-2\beta)|B_G|$, if $\max_{M\in\boldsymbol{R}^j}\langle M,F^j_{B'}\rangle\ge c\frac{\Delta^2}{\beta\log^4 j}$, then $\forall B''_G\subseteq B_G\cap B'$ s.t. $|B''_G|\le\beta|B|$, $M^*:=\operatorname*{argmax}_{M\in\boldsymbol{R}^j}\langle M,F^j_{B'}\rangle$ satisfies*

$$\sum_{b\in B''_G}\langle M^*,C^j_{b,B'}\rangle\le\frac{1}{8}\sum_{b\in B'\cap B_A}\langle M^*,C^j_{b,B'}\rangle.$$

Proof. Let

$$\kappa:=\frac{\langle M^*,F^j_{B'}\rangle}{\frac{\Delta^2}{\beta\cdot\log^4 j}}\quad\text{and}\quad\tau:=\max_{M\in\boldsymbol{R}^j}\frac{\langle M,(\bar{p}^j_{B'_G}-\bar{p}^j_{B'})^{\otimes 2}\rangle}{\frac{\Delta^2}{\log^4 j}}.$$

For any $M \in \boldsymbol{R}^j$, from Lemma 63

$$\max_{M \in \boldsymbol{R}^j} \langle M, (\bar{p}^j_{B'_G} - \bar{p}^j_{B'})^{\otimes 2} \rangle \leq 8\kappa \frac{\Delta^2}{\log^4 j} + \mathcal{O}\Big(\frac{\Delta^2}{\log^4 j}\Big).$$

Hence

$$\tau \leq 8\kappa + \mathcal{O}(1).$$

A calculation similar to Equation (4.9) shows that

$$\begin{aligned}
&\sum_{b \in B'} \langle M, C^j_{b,B'} - C(\bar{p}^j_{B'}) \rangle \\
&\leq \mathcal{O}(|B'_G| \frac{\Delta^2}{\log^4 j}) + |B'_G| \langle M, (\bar{p}^j_{B'_G} - \bar{p}^j_{B'})^{\otimes 2} \rangle + 6K_G \cdot \sqrt{\tau} |B'_G| \frac{\Delta^2}{\beta \log^4 j} + \sum_{b \in B'_A} \langle M, C^j_{b,B'} \rangle.
\end{aligned}$$

Then

$$\begin{aligned}
&\sum_{b \in B'_A} \langle M, C^j_{b,B'} \rangle \\
&\geq \sum_{b \in B'} \langle M, C^j_{b,B'} - C(\bar{p}^j_{B'}) \rangle - \mathcal{O}(|B'_G| \frac{\Delta^2}{\log^4 j}) - |B'_G| \langle M, (\bar{p}^j_{B'_G} - \bar{p}^j_{B'})^{\otimes 2} \rangle \\
&\quad - 6K_G \cdot \sqrt{\tau} |B'_G| \frac{\Delta^2}{\beta \log^4 j} \\
&\geq |B'| \langle M, F^j_{B'} \rangle - \mathcal{O}(|B'_G| \frac{\Delta^2}{\log^4 j}) - |B'_G| (8\kappa + \mathcal{O}(1)) \frac{\Delta^2}{\log^4 j} \\
&\quad - 6K_G \cdot \sqrt{8\kappa + \mathcal{O}(1)} |B'_G| \frac{\Delta^2}{\beta \log^4 j}
\end{aligned}$$

For $M = M^*$,

$$
\begin{aligned}
&\sum_{b\in B'_A} \langle M^*, C^j_{b,B'}\rangle \\
&\geq |B'|\frac{\kappa\Delta^2}{\beta\cdot\log^4 j} - \mathcal{O}(|B'_G|\frac{\Delta^2}{\log^4 j}) - |B'_G|(8\kappa + \mathcal{O}(1))\frac{\Delta^2}{\log^4 j} \\
&\quad - 6K_G \cdot \sqrt{8\kappa + \mathcal{O}(1)}|B'_G|\frac{\Delta^2}{\beta\log^4 j} \\
&\geq \frac{|B_G|\Delta^2}{\beta\cdot\log^4 j}\left(\frac{\kappa|B'|}{|B_G|} - \mathcal{O}(\frac{|B'_G|\beta}{|B_G|}) - \frac{(8\kappa+\mathcal{O}(1))|B'_G|\beta}{|B_G|} - \frac{6K_G\cdot\sqrt{8\kappa+\mathcal{O}(1)}|B'_G|}{|B_G|}\right) \\
&\geq \Omega\bigg(|B_G|\frac{\Delta^2}{\beta\log^4 j}(\kappa - \sqrt{\kappa} - 1)\bigg),
\end{aligned}
$$

where in the last step we assumed that β is sufficiently small, and Equation (4.3).

Next, for any $B''_G \subseteq B'_G$ of size $\leq \beta|B| = \beta\frac{|B_G|}{(1-\beta)} \leq 2\beta|B_G|$,

$$
\begin{aligned}
\sum_{b\in B''_G} C^j_{b,B'} - \sum_{b\in B''_G} C^j_{b,B''_G} &= \sum_{b\in B''_G}(\bar{\mu}^j_b - \bar{p}^j_{B'})^{\otimes 2} - \sum_{b\in B''_G}(\bar{\mu}^j_b - \bar{p}^j_{B''_G})^{\otimes 2} \\
&= |B''_G|(\bar{p}^j_{B''_G} - \bar{p}^j_{B'})^{\otimes 2} \\
&= |B''_G|\big((\bar{p}^j_{B''_G} - p^j) + (p^j - p^j_{B'_G}) + (p^j_{B'_G} - \bar{p}^j_{B'})\big)^{\otimes 2}.
\end{aligned}
$$

where the second to the last equality uses Lemma 59. It follows that for any PSD matrix M

$$
\begin{aligned}
&\sum_{b\in B''_G}\langle M, C^j_{b,B'}\rangle \\
&\leq |\sum_{b\in B''_G}\langle M, C^j_{b,B''_G}\rangle| + 3|B''_G|\Big(\langle M, (\bar{p}^j_{B''_G} - p^j)^{\otimes 2}\rangle + \langle M, (p^j - p^j_{B'_G})^{\otimes 2}\rangle \\
&\quad + \langle M, (p^j_{B'_G} - \bar{p}^j_{B'})^{\otimes 2}\rangle\Big) \\
&\leq \mathcal{O}(|B_G|\frac{\Delta^2}{\beta\log^4 j}) + \mathcal{O}(|B_G|\frac{\Delta^2}{\beta\log^4 j}) + \mathcal{O}(|B''_G|\frac{\Delta^2}{\log^4 j}) + |B''_G|(8\kappa + \mathcal{O}(1))\frac{\Delta^2}{\log^4 j} \\
&\leq \mathcal{O}(|B_G|(\beta^2\kappa + 1)\frac{\Delta^2}{\beta\log^4 j}),
\end{aligned}
$$

where in the second inequality, the first two terms are bounded using the essential property 2, third term using the essential property 1a of the good batches, and the last term is bounded from the definition of τ and the relation $\tau \le 8\kappa + \mathcal{O}(1)$, and the last inequality follows as $|B''_G| \le 2\beta|B_G|$. The above bound holds for all M including M^*

Note that for β smaller than some absolute constant and κ larger than some absolute constant the ratio

$$\frac{\sum_{b\in B''_G}\langle M^*, C^j_{b,B'}\rangle}{\sum_{b\in B'_A}\langle M^*, C^j_{b,B'}\rangle},$$

can be made smaller than 1/8, completing the lemma's proof. ■

4.11 Proof of some simple results in the main paper

In the previous section we established the filtering conditions and used them to show the correctness of the filtering Algorithm. In the next section we show that the essential properties of good batches has essential properties with high probability. In this section we establish some of the less intricate results, Lemmas 48, 51, 52, and 54. For the reader's convenience we restate each lemma before proving it.

Lemma 48. *For any k and distributions p, q over $[0,1] \times \{-1,1\}$,*

$$r_p(h^{opt}(q)) - r_p^{opt}(\mathcal{H}_k) \le 2||p^{[-1,1]} - q^{[-1,1]}||_{\mathcal{A}_{2k}}.$$

Proof. For $h \in \mathcal{H}_k$, let $S_h^{+1} := \{x : x \in [0,1] : h(x) = -1\}$, similarly $S_h^{-1} := \{-x : x \in [0,1] : h(x) = 1\}$, and finally $S_h := S_h^{+1} \cup S_h^{-1}$. Observe that $h(x) \ne y$ iff $x \cdot y \in S_h$.

The definition of $\mathcal{H}_k$ implies that S_h^{+1} and S_h^{-1} consist of at-most k intervals. Therefore, S_h consists of at most $2k$ intervals, and $S_h \in \mathcal{A}_{2k}$.

For two distributions p and q over $[0,1] \times \{-1,1\}$, the largest difference between the loss of any classifier $h \in \mathcal{H}_k$ is

$$\begin{aligned}
\sup_{h\in\mathcal{H}_k} |r_p(h) - r_q(h)| &= \sup_{h\in\mathcal{H}_k} |\Pr_{(X,Y)\sim p}[h(X) \neq Y] - \Pr_{(X,Y)\sim q}[h(X) \neq Y]| \\
&= \sup_{h\in\mathcal{H}_k} |\Pr_{(X,Y)\sim p}[X \cdot Y \in S_h] - \Pr_{(X,Y)\sim q}[X \cdot Y \in S_h]| \\
&\leq \sup_{S\in\mathcal{A}_{2k}} |\Pr_{(X,Y)\sim p}[X \cdot Y \in S] - \Pr_{(X,Y)\sim q}[X \cdot Y \in S]| \\
&\leq ||p^{[-1,1]} - q^{[-1,1]}||_{\mathcal{A}_{2k}}.
\end{aligned}$$

Using a sequence of triangle inequalities, followed by this result,

$$\begin{aligned}
&r_p(h^{\text{opt}}(q)) - r_p^{\text{opt}}(\mathcal{H}_k) \\
&= r_p(h^{\text{opt}}(q)) - r_p(h^{\text{opt}}(p)) \\
&= r_p(h^{\text{opt}}(q)) - r_q(h^{\text{opt}}(q)) + r_q(h^{\text{opt}}(q)) - r_q(h^{\text{opt}}(p)) + r_q(h^{\text{opt}}(p)) - r_p(h^{\text{opt}}(p)) \\
&\leq r_q(h^{\text{opt}}(q)) - r_q(h^{\text{opt}}(p)) + 2 \sup_{h\in\mathcal{H}_k} |r_q(h) - r_p(h)| \\
&\leq 2 \sup_{h\in\mathcal{H}_k} |r_q(h) - r_p(h)| \\
&\leq 2 \cdot ||p^{[-1,1]} - q^{[-1,1]}||_{\mathcal{A}_{2k}}. \qquad \blacksquare
\end{aligned}$$

Lemma 51. *For any subset $S \in \mathcal{A}_k$, there are sets $S', S'' \in \mathcal{A}_k(\mathcal{P}^j)$ such that $S' \subseteq S \subseteq S''$ and $\bar{p}_B(S'' \setminus S') \leq 2/j$.*

Proof. Any set $S \in \mathcal{A}_k$ is a union of k real intervals $I_1 \cup I_2 \cup \ldots \cup I_k$. Let I_j' be the largest interval over $\mathcal{P}^j$ that is fully contained in I_j, and let $S' = I_1' \cup I_2' \cup \ldots \cup I_k'$. By definition, $S' \in \mathcal{A}_k(\mathcal{P}^j)$. Similarly, let I_j'' be the smallest interval over $\mathcal{P}^j$ that is fully contained in I_j, and let $S'' = I_1'' \cup I_2'' \cup \ldots \cup I_k''$. Again, $S'' \in \mathcal{A}_k(\mathcal{P}^j)$. It is easy to see that $I_j'' \setminus I_j'$ consists of at most two parts, one each on the right and left. Therefore, $S'' \setminus S'$ contains at most $2k$ parts. Since all $k \cdot j$ parts of the partition contain an equal number of samples in B, we obtain $\bar{p}_B(S'' \setminus S') \leq 2/j$. ■

Lemma 52. *For $\beta \le 1/2$, $j \le 1/\Delta$, and $m \cdot n = \tilde{\Omega}(\frac{k+\log 1/\delta}{\Delta^2})$, with probability $> 1-\delta$, for all $S \in \mathcal{C}(\mathcal{P}^j)$, $p(S) \le 3 \cdot \bar{p}_B(S) + \mathcal{O}(\Delta)$.*

Proof. First we bound the empirical probability assigned bu good batches B_G to subset S.

$$\begin{aligned}\bar{p}_{B_G}(S) &= \frac{\text{Number of samples in } B_G \text{ that lie in } S}{|B_G| \cdot n} \\ &\le \frac{\text{Number of samples in } B \text{ that lie in } S}{|B_G| \cdot n} = \frac{\bar{p}_B(S) \cdot |B| \cdot n}{|B_G| \cdot n} = \frac{\bar{p}_B(S)}{(1-\beta)}.\end{aligned}$$

Collection of all samples in B_G can be thought of as $|B_G| \cdot n$ i.i.d. samples from underlying distribution from p. Next note that $\mathcal{C}(\mathcal{P}^j) \subseteq \mathcal{A}_{k\cdot j}$. The set $\mathcal{A}_{k\cdot j}$ has VC dimension $\mathcal{O}(k \cdot j)$.

Applying Theorem 67 for $\mathcal{A}_{k\cdot j}$, for $|B_G| \cdot n \ge \Omega(\frac{k\cdot j \log(1/\epsilon)+\log(1/\delta)}{\epsilon^2})$, with probability $\ge 1-\delta$,

$$\sup_{S \in \mathcal{A}_{k\cdot j}} \frac{p(S) - \bar{p}_{B_G}(S)}{\sqrt{p(S)}} \le \epsilon.$$

Since $|B_G| \cdot n = (1-\beta) m \cdot n = \tilde{\Omega}(\frac{k+\log 1/\delta}{\Delta^2})$ and $j \le \frac{1}{\Delta}$, we can choose $\epsilon = \frac{\sqrt{\Delta}}{4}$ in the above equation. We get

$$p(S) \le \bar{p}_{B_G}(S) + \frac{\sqrt{\Delta \cdot p(S)}}{4} \le \bar{p}_{B_G}(S) + \frac{p(S)}{8} + \frac{\Delta}{8},$$

here the last inequality used $\sqrt{ab} = |a| + |b|/2$. Hence

$$p(S) \le \frac{8\bar{p}_B(S)}{7(1-\beta)} + \frac{\Delta}{7} \le \frac{16}{7}\bar{p}_B(S) + \frac{\Delta}{7}.$$

■

Lemma 54. *For all $i \ge 1$, any interval over partition $\mathcal{P}^{2^i}$ can be expressed as the union of at-most 2 parts from each partition $\mathcal{P}^{2^i}, \mathcal{P}^{2^1}, ..., \mathcal{P}^{2^2}$ and one interval over $\mathcal{P}^2$.*

Proof. First we show the following claim.

Claim: For all j, any interval over $\mathcal{P}^{2j}$ can be expressed as the union of an interval over $\mathcal{P}^{j}$, and at-most two parts of $\mathcal{P}^{2j}$.

Proof of claim: Recall that any interval I over $\mathcal{P}^{2j}$ is a union of consecutive elements of $\mathcal{P}^{2j}$, namely $I = \cup_{i=i_s}^{i_e} P_i^{2j}$ for some $0 \leq i_s \leq i_e \leq k \cdot 2j$. Observe that by definition, $P_{2i-1}^{2j} \cup P_{2i}^{2j} = P_i^j$. We consider four cases based on whether the numbers i_s and i_e are even or odd, and prove the claim for each case separately.

First, when both i_s and i_e are odd,

$$\begin{aligned}\bigcup_{i=i_s}^{i_e} P_i^{2j} &= \Big(\bigcup_{i=\frac{i_s+1}{2}}^{\frac{i_e-1}{2}} \Big(P_{2i-1}^{2j} \bigcup P_{2i}^{2j}\Big)\Big) \bigcup P_{i_e}^{2j} \\ &= \Big(\bigcup_{i=\frac{i_s+1}{2}}^{\frac{i_e-1}{2}} P_i^j\Big) \bigcup P_{i_e}^{2j}.\end{aligned}$$

Noting that $\bigcup_{i=\frac{i_s+1}{2}}^{\frac{i_e-1}{2}} P_i^j$ is an interval over partition $\mathcal{P}^j$, and $P_{i_e}^{2j}$ is a part of $\mathcal{P}^{2j}$ proves the claim for this case.

Next, for even i_s and odd i_e,

$$\begin{aligned}\bigcup_{i=i_s}^{i_e} P_i^{2j} &= P_{i_s}^{2j} \bigcup \Big(\bigcup_{i=\frac{i_s+2}{2}}^{\frac{i_e-1}{2}} \Big(P_{2i-1}^{2j} \bigcup P_{2i}^{2j}\Big)\Big) \bigcup P_{i_e}^{2j} \\ &= \Big(\bigcup_{i=\frac{i_s+2}{2}}^{\frac{i_e-1}{2}} P_i^j\Big) \bigcup P_{i_s}^{2j} \bigcup P_{i_e}^{2j}.\end{aligned}$$

Noting that $\bigcup_{i=\frac{i_s+2}{2}}^{\frac{i_e-1}{2}} P_i^j$ is an interval over partition $\mathcal{P}^j$, and $P_{i_s}^{2j}$ and $P_{i_e}^{2j}$ are two parts of $\mathcal{P}^{2j}$ proves the claim for this case. The two remaining cases can be proved similarly to complete the proof of the claim.

From the claim, any interval I over $\mathcal{P}^{2^i}$ can be expressed as a union of some interval I' over $\mathcal{P}^{2^{i-1}}$ and at-most 2 parts of $\mathcal{P}^{2^i}$. Applying the claim again, interval I' can be expressed as a union of an interval I'' over $\mathcal{P}^{2^{i-2}}$ and at-most 2 parts of $\mathcal{P}^{2^{i-1}}$. Applying the claim i times, we obtain the lemma. ■

4.12 Essential properties of good batches

This section is devoted to establishing the Essential properties of the good batches and proving Theorem 56. In the process we also derive Lemma 62, used in Section 4.10.1. To establish these properties we use Theorem 57, stated in the main paper. We start by proving this theorem.

Recall $\mathcal{L}_i^j : \{v \in \mathbb{R}^j : ||v||_\infty = 1, ||v||_2^2 \leq i\}$. The next lemma shows that if the essential properties hold for $\{v^{\otimes 2} : v \in \mathcal{L}_{2k}^{k \cdot j}\}$, the self outer products of vectors in $\mathcal{L}_{2k}^{k \cdot j}$, then they hold for all $M \in \boldsymbol{R}^j$. The following well known result will be useful.

Theorem 64 (Grothendieck's inequality)**.** *For any matrix $A \in \mathbb{R}^{n \times n}$,*

$$\max_{M \succcurlyeq 0, M_{ii} \leq 1} |\langle M, A \rangle| \leq K_G \max_{u,v : ||v||_\infty \leq 1, ||u||_\infty \leq 1} |\langle uv^\intercal, A \rangle|,$$

where $K_G \leq 1.783$ is an absolute constant.

We use this inequality to derive Theorem 57 in the main paper. Recall that $\boldsymbol{R}^j := \{M \in \mathbb{R}^{k \cdot j \times k \cdot j} : M \succcurlyeq 0, M_{ii} \leq 1, \sum_i M_{ii} \leq 2k\}$.

Theorem 57. *There is a universal constant $K_G \leq 1.7822$ such that for all real symmetric matrices $A \in \mathbb{R}^{k \cdot j \times k \cdot j}$,*

$$\max_{M \in \boldsymbol{R}^j} |\langle M, A \rangle| \leq 2 \cdot K_G \cdot \max\{|\langle v^{\otimes 2}, A \rangle| : v \in \mathcal{L}_{2k}^{k \cdot j}\}.$$

Proof. Any $M \in \boldsymbol{R}^j$ is PSD and satisfies $M_{ii} \leq 1$. Consider the diagonal matrix $D = \text{Diag}(M_{ii})$, and let $\hat{M} = D^{-1/2} M D^{-1/2}$. Observe that $\hat{M}$ is also PSD and $\hat{M}_{ii} \leq 1$. Then

$$\langle M, A \rangle = \langle D^{1/2} \hat{M} D^{1/2}, A \rangle = \langle \hat{M}, D^{1/2} A D^{1/2} \rangle,$$

where the last step follows from the standard property that $\langle A_1A_2A_3, A_4\rangle = \langle A_2, A_3A_4A_1\rangle$.

From Grothendieck's inequality, there is some u, v such that $||u||_\infty, ||v||_\infty \le 1$ and

$$|\langle \hat{M}, D^{1/2}AD^{1/2}\rangle| \le K_G|\langle uv^\intercal, D^{1/2}AD^{1/2}\rangle|$$
$$= K_G|\langle D^{1/2}uv^\intercal D^{1/2}, A\rangle| = K_G|\langle D^{1/2}u(D^{1/2}v)^\intercal, A\rangle| = |\langle \hat{u}\hat{v}^\intercal, A\rangle|,$$

where $\hat{u} = D^{1/2}u$ and $\hat{v} = D^{1/2}v$. Note that since $D_{ii} = \sqrt{M_{ii}} \le 1$, hence $||\hat{u}||_\infty \le ||u||_\infty \le 1$ and

$$||\hat{u}||_2^2 = \sum_i D_{ii}^2 u_i^2 \le \sum_i M_{ii} \le 2k,$$

where the last inequality used the facts that $|u_i| \le ||u||_\infty \le 1$ and that $\sum_i M_{i,i} \le 2k$ for all $M \in \boldsymbol{R}^j$. A similar observation holds for v. Hence,

$$\max_{M\in \boldsymbol{R}^j} |\langle M, A\rangle| \le K_G \cdot \max\{|\langle uv^\intercal, A\rangle| : u, v \in \mathcal{L}_{2k}^{k\cdot j}\}.$$

Let $x = \frac{u+v}{2}$ and $y = \frac{u-v}{2}$. Since for any norm $||\frac{u+v}{2}||, ||\frac{u-v}{2}|| \le \frac{||u||+||v||}{2}$, if $u, v \in \mathcal{L}_{2k}^{k\cdot j}$ then $x, y \in \mathcal{L}_{2k}^{k\cdot j}$. Also note that $u = x + y$ and $v = x - y$. Then, since A is symmetric

$$\langle uv^\intercal, A\rangle = \langle x^{\otimes 2}, A\rangle - \langle y^{\otimes 2}, A\rangle.$$

Combining the last two statements, we obtain

$$\max_{M\in \boldsymbol{R}^j} |\langle M, A\rangle| \le K_G \cdot \max\{|\langle uv^\intercal, A\rangle| : u, v \in \mathcal{L}_{2k}^{k\cdot j}\} \le 2K_G \cdot \max\{|\langle xx^\intercal, A\rangle| : x \in \mathcal{L}_{2k}^{k\cdot j}\}. \blacksquare$$

Next, we represent outer products of vectors in $\mathcal{L}_{2k}^{k\cdot j}$ as sums of outer products of binary vectors. This conversion has the advantage that, for example, the product of binary bit vectors with $\bar{\mu}_b^j$ and p^j, corresponds to $\bar{\mu}_b(S)$ and $p(S)$ for some set S of reals. Since for any $S \in \mathbb{R}$ and good batch b, $n \cdot \bar{\mu}_b(S)$ has binomial distribution $\text{Bin}(p(S), n)$, this allow us to use the concentration properties of binomial random variables.

Let y^i denote the i^{th} bit in the binary representation of $y \in [0,1]$. Represent 1 as 0.1111.... and if any other number has two such representations, either can be used. Note that $y = \sum_{i=1}^{\infty} 2^{-i} y^i$.

For any $x \in [-1,1]$ let $x^+ = \max\{0, x\}$ and $x^- = \max\{0, -x\}$. It is easy to see that $0 \le x^+, x^- \le |x|$, and $x = x^+ - x^-$. Therefore, for any $x \in [-1,1]$, $x = \sum_{i=1}^{\infty} 2^{-i}(x^{+,i} - x^{-,i})$.

Next we extend these these definition to vectors with bounded elements. For any vector $v = (v_1, v_2, ...)$ of reals such that $||v||_\infty \le 1$, let $v^{+,i} = (v_1^{+,i}, v_2^{+,i}, ...)$ and $v^{-,i} = (v_1^{-,i}, v_2^{-,i}, ...)$. It follows that $v = \sum_{i=1}^{\infty} 2^{-i}(v^{+,i} - v^{-,i})$.

Then for any vector v such that $||v||_\infty \le 1$

$$\begin{aligned} v^{\otimes 2} &= \Big(\sum_{i=1}^{\infty} 2^{-i}(v^{+,i} - v^{-,i})\Big)\Big(\sum_{i=1}^{\infty} 2^{-i}(v^{+,i} - v^{-,i})\Big)^\intercal \\ &= \sum_{i'=1}^{\infty}\sum_{i=1}^{\infty} 2^{-(i+i')}(v^{+,i} - v^{-,i})(v^{+,i'} - v^{-,i'})^\intercal \end{aligned}$$

Observe that $v^{+,i}$ are binary bit vectors, and $2^{-i}||v^{+,i}||_2 \le ||v||_2$. Further, since $v^{+,i}$ is binary bit vectors $||v^{+,i}||_2 = \sqrt{||v^{+,i}||_0}$, we get $||v^{+,i}||_0 \le 2^{2i}||v||_2^2$. The same observation also holds for $v^{-,i}$.

For any $v \in \mathcal{L}_{2k}^{k\cdot j}$, and any matrix $A \in \mathbb{R}^{k\cdot j \times k\cdot j}$,

$$\begin{aligned} \langle v^{\otimes 2}, A\rangle \le \sum_{i'=1}^{\infty}\sum_{i=1}^{\infty} 4 \cdot 2^{-(i+i')} \cdot \max\{|\langle \hat{u}\hat{v}^\intercal, A\rangle| : \hat{v}, \hat{u} \in \{0,1\}^{k\cdot j}, ||\hat{u}||_\infty \\ \le \min\{2k \cdot 2^{2i}, 2kj\}, ||\hat{v}||_\infty \le \min\{2k \cdot 2^{2i'}, 2kj\}\}. \end{aligned} \tag{4.10}$$

We recall a few definitions from Section 4.3. For any distribution q on $\mathbb{R}$, for any $j \ge 1$ $q^j \in \mathbb{R}^{k\cdot j}$ is the discrete distribution over the indices of partition $\mathcal{P}^j$, defined by $q^j(i) = q(P_i^j)$

for $i \in [k \cdot j]$. Every subset $S \in \mathcal{C}(\mathcal{P}^j)$ to the binary vector $v_S \in \{0,1\}^{k\cdot j}$ whose ith coordinate indicates whether $P^j_i \subseteq S$ and the inner product $q(S) = q^j \cdot v_S$.

For any $j \geq 1$ and $1 \leq i \leq j$ let $\mathcal{I}(\mathcal{P}^j_i) \subseteq \mathcal{C}(\mathcal{P}^j)$ consist of all unions of at most $k \cdot i$ parts of $\mathcal{P}^j$. Recall that for any $0 \leq i \leq j$, $\{0,1\}^j_i$ denote the set of binary vectors of length j with at most j ones. Observe that every subset in $S \in \mathcal{I}(\mathcal{P}^j_i)$ corresponds to a binary vector $v_S \in \{0,1\}^{k\cdot j}_{k\cdot i}$ and vice versa.

For any $i \leq j$ the subsets in $\mathcal{I}(\mathcal{P}^j_i)$ are a union of at most $k \cdot i$ parts, hence the set $\mathcal{I}(\mathcal{P}^j_i) \subseteq \mathcal{A}_{k\cdot i}$. Next recall that all $k \cdot j$ parts of partition $\mathcal{P}^k$ has equal number of samples among all samples in B. Therefore, for any subset $S \in \mathcal{I}(\mathcal{P}^j_i)$, $\bar{p}_B(S) = \mathcal{O}\Big(\frac{k\cdot i}{k\cdot j}\Big) = \mathcal{O}\Big(\frac{i}{j}\Big)$. Then Lemma 52 implies that $p(S) = \mathcal{O}\Big(\frac{i}{j} + \Delta\Big) = \mathcal{O}\Big(\frac{i}{j}\Big)$, as $j = \mathcal{O}(\frac{1}{\Delta})$.

We use these observations to make the following reductions:

For any B' any $j \leq \frac{1}{\Delta}$, any $1 \leq i' \leq i \leq j$,

$$\begin{aligned}
&\max_{u\in\{0,1\}^{k\cdot j}_{k\cdot i}} \max_{v\in\{0,1\}^{k\cdot j}_{k\cdot i'}} \langle uv^\intercal, (\bar{p}^j_{B'} - p^j)^{\otimes 2}\rangle \\
&= \max_{u\in\{0,1\}^{k\cdot j}_{k\cdot i}} \max_{v\in\{0,1\}^{k\cdot j}_{k\cdot i'}} (\bar{p}^j_{B'}\cdot u - p^j \cdot u)(\bar{p}^j_{B'}\cdot v - p^j\cdot v) \\
&= \max_{S\in\mathcal{I}(\mathcal{P}^j_i)} \max_{S'\in\mathcal{I}(\mathcal{P}^j_{i'})} (\bar{p}_{B'}(S) - p(S))(\bar{p}_{B'}(S') - p(S')) \\
&\leq \max_{S\in\mathcal{A}_{k\cdot i}, p(S)\leq\mathcal{O}(\frac{i}{j}),} \max_{S'\in\mathcal{A}_{k\cdot i'}, p(S')\leq\mathcal{O}(\frac{i'}{j})} (\bar{p}_{B'}(S) - p(S))(\bar{p}_{B'}(S') - p(S'))
\end{aligned} \tag{4.11}$$

Recall that for any binary vectors u, v and any discrete distribution q,

$$\langle uv^\intercal, C(q)\rangle = \frac{1}{n}\langle uv^\intercal, \text{Diag}(q) - q^{\otimes 2}\rangle = \frac{1}{n}\langle uv^\intercal, \text{Diag}(q)\rangle - \langle uv^\intercal, q^{\otimes 2}\rangle$$

Using the above equation, and following the similar calculation as (4.11), it is easy to show

$$
\begin{aligned}
&\max_{u\in\{0,1\}^{k\cdot j}_{k\cdot i}} \max_{v\in\{0,1\}^{k\cdot j}_{k\cdot i'}} \langle uv^{\intercal}, C(\bar{p}^{j}_{B'})\rangle \\
&\leq \max_{S\in\mathcal{A}_{k\cdot i}, p(S)\leq\mathcal{O}(\frac{i}{j})} \max_{S'\in\mathcal{A}_{k\cdot i'}, p(S')\leq\mathcal{O}(\frac{i'}{j})} \left(\frac{1}{n}(\bar{p}_{B'}(S\cap S') - \bar{p}_{B'}(S)\cdot\bar{p}_{B'}(S'))\right) \\
&\leq \max_{S:p(S)\leq\mathcal{O}(\frac{i}{j})} \max_{S':p(S')\leq\mathcal{O}(\frac{i'}{j})} \left(\frac{1}{n}(\bar{p}_{B'}(S\cap S') - \bar{p}_{B'}(S)\cdot\bar{p}_{B'}(S'))\right) \leq \frac{\min\{i,i'\}}{jn} \qquad (4.12)
\end{aligned}
$$

Similarly, one can also show

$$
\begin{aligned}
&\max_{u\in\{0,1\}^{k\cdot j}_{k\cdot i}} \max_{v\in\{0,1\}^{k\cdot j}_{k\cdot i'}} \frac{1}{|B'|}\sum_{b\in B'}\langle uv^{\intercal}, (\bar{\mu}^{j}_{b} - p^{j})^{\otimes 2}\rangle - \langle uv^{\intercal}, C(p^{j})\rangle \\
&\leq \max_{S\in\mathcal{A}_{k\cdot i}, p(S)\leq\mathcal{O}(\frac{i}{j})} \max_{S'\in\mathcal{A}_{k\cdot i'}, p(S')\leq\mathcal{O}(\frac{i'}{j})} \left(\frac{1}{|B'|}\sum_{b\in B'}(\bar{\mu}_b(S) - p(S))(\bar{\mu}_b(S') - p(S'))\right. \\
&\quad \left. - \frac{1}{n}(p(S\cap S') - p(S)\cdot p(S'))\right). \qquad (4.13)
\end{aligned}
$$

In Section 4.14 we obtain the following concentration bound.

Let

$$\xi(S,i) := \max\left(\sqrt{p(S)\log\left(\frac{2}{p(S)}\right)}, \frac{\sqrt{i}}{\log^{3}\frac{1}{\Delta}}\right).$$

Theorem 65. *For* $|B_G| = \tilde{\Omega}(\frac{k+\log(1/\delta)}{\beta^2})$*, with probability* $\geq 1-\delta$*, for all* $1 \leq i' \leq i \leq \frac{1}{\Delta}$*,*

1. *For all* $B'_G \subseteq B_G$ *such that* $|B_G \setminus B'_G| \leq \mathcal{O}(\beta|B_G|)$*,*

$$\max_{S\in\mathcal{C}_i} |\bar{p}_{B'_G}(S) - p(S)| \leq \mathcal{O}\bigg(\Delta\cdot\xi(S,i)\bigg),$$

and

$$\max_{S\in\mathcal{C}_i}\max_{S'\in\mathcal{C}_{i'}:p(S')\leq\frac{2i'}{i}}\left|\frac{1}{|B'_G|}\sum_{b\in B'_G}(\bar{\mu}_b(S)-p(S))(\bar{\mu}_b(S')-p(S')) - \frac{1}{n}(p(S\cap S')-p(S)\cdot p(S'))\right| \leq \mathcal{O}\bigg(\frac{\Delta^2}{\beta}\cdot\xi(S,i)\xi(S,i')\bigg).$$

2. *For all $B'_G \subseteq B_G$, such that $|B_G| \leq \mathcal{O}(\beta|B_G|)$,*

$$\max_{S\in\mathcal{C}_i}\max_{S'\in\mathcal{C}_{i'}}\left|\sum_{b\in B'_G}(\bar{\mu}_b(S)-p(S))(\bar{\mu}_b(S')-p(S'))\right| \leq \mathcal{O}\bigg(|B_G|\frac{\Delta^2}{\beta}\cdot\xi(S,i)\xi(S,i')\bigg).$$

Now we are ready to prove Theorem 56.

4.12.1 Proof of Theorem 56

For $B'_G \subseteq B_G$ such that $|B_G \setminus B'_G| \le \mathcal{O}(\beta |B_G|)$ and $A = (\bar{p}^j_{B'_G} - p^j)^{\otimes 2}$,

$$
\begin{aligned}
&|\langle M, (\bar{p}^j_{B'_G} - p^j)^{\otimes 2}\rangle| = |\langle M, A\rangle| \\
&\overset{(a)}{\le} 2 \cdot K_G \cdot \max_{v \in \mathcal{L}^{k\cdot j}_{2k}} |\langle v^{\otimes 2}, F^j_{B'_G}\rangle| \\
&\overset{(b)}{\le} 2 \cdot K_G \cdot \sum_{i'=1}^{\infty}\sum_{i=1}^{\infty} 4 \cdot 2^{-(i+i')}. \\
&\ \max\{|\langle uv^{\intercal}, A\rangle| : v, u \in \{0,1\}^{k\cdot j}, ||u||_\infty \le \min\{2k \cdot 2^{2i}, 2kj\}, ||v||_\infty \le \min\{2k \cdot 2^{2i'}, 2kj\}\} \\
&\overset{(c)}{\le} 2 \cdot K_G \cdot \sum_{i'=1}^{\infty}\sum_{i=1}^{\infty} 4 \cdot 2^{-(i+i')} \max_{S \in \mathcal{A}_{k\cdot\min\{2^i, j\}}, p(S) \le \mathcal{O}(\frac{\min\{2^i, j\}}{j})}, \max_{S' \in \mathcal{A}_{k\cdot\min\{2^{i'}, j\}}, p(S') \le \mathcal{O}(\frac{\min\{2^{i'}, j\}}{j})} \\
&\qquad |(\bar{p}_{B'_G}(S) - p(S))(\bar{p}_{B'_G}(S') - p(S'))| \\
&\overset{(d)}{\le} 2 \cdot K_G \cdot \sum_{i'=1}^{\infty}\sum_{i=1}^{\infty} 4 \cdot 2^{-(i+i')} \max_{S \in \mathcal{A}_{k\cdot\min\{2^i, j\}}, p(S) \le \mathcal{O}(\frac{\min\{2^i, j\}}{j})} \max_{S' \in \mathcal{A}_{k\cdot\min\{2^{i'}, j\}}, p(S') \le \mathcal{O}(\frac{\min\{2^{i'}, j\}}{j})} \\
&\qquad \Big(\xi(S, \min\{2^i, j\})\xi(S', \min\{2^{i'}, j\}) \cdot \mathcal{O}(\Delta^2)\Big) \\
&\overset{(e)}{\le} \mathcal{O}\bigg(\frac{\Delta^2}{\log^4 j}\bigg),
\end{aligned}
\tag{4.14}
$$

where (a) uses Theorem 57, (b) uses Equation (4.10), (c) uses Equation (4.11), (d) uses the first statement of Theorem 65, and the last step (e) can be verified by a slightly lengthy but standard calculation. This proves the essential property 1a.

For $B'_G \subseteq B_G$ such that $|B_G \setminus B'_G| \le \mathcal{O}(\beta |B_G|)$ and $A = \frac{1}{|B'_G|}\sum_{b \in B'_G}(\bar{\mu}^j_b - p^j)(\bar{\mu}^j_b - p^j) - C(p^j)$, a similar calculation as the above uses the second statement of Theorem 65, and shows

$$
\Big|\langle M, \frac{1}{|B'_G|}\sum_{b \in B'_G}(\bar{\mu}^j_b - p^j)^{\otimes 2} - C(p^j)\rangle\Big| \le \mathcal{O}\bigg(\frac{\Delta^2}{\beta \log^4 j}\bigg). \tag{4.15}
$$

Next, from lemma 59

$$\sum_{b \in B'_G} (\bar{\mu}_b^j - p^j)^{\otimes 2} = \sum_{b \in B'_G} (\bar{p}^j_{B'_G} - p^j)^{\otimes 2} + |B'_G|(\bar{\mu}_b^j - \bar{p}^j_{B'_G})^{\otimes 2}. \tag{4.16}$$

Then

$$\Big|\langle M, \sum_{b \in B'_G} (\bar{\mu}_b^j - p^j)^{\otimes 2} - \sum_{b \in B'_G} (\bar{\mu}_b^j - \bar{p}^j_{B'_G})^{\otimes 2}\rangle\Big| = |B'_G|\Big|\langle M, (\bar{p}^j_{B'_G} - p^j)^{\otimes 2}\rangle\Big| \le |B'_G|\mathcal{O}\bigg(\frac{\Delta^2}{\log^4 j}\bigg),$$

where the last inequality follows from the essential property 1a, that we already proved.

A similar calculation as (4.7), gives

$$|\langle M, C(p^j) - C(\bar{p}_{B'_G})\rangle| \le \frac{\Delta^2}{\beta \log^4 j}$$

Combining the last Equations and (4.15) we get

$$|\langle M, F^j_{B'_G}\rangle| = \Big|\langle M, \frac{1}{|B'_G|} \sum_{b \in B'_G} (\bar{\mu}_b^j - \bar{p}_{B'_G})^{\otimes 2} - C(\bar{p}_{B'_G})\rangle\Big| \le \mathcal{O}\bigg(\frac{\Delta^2}{\beta \log^4 j}\bigg).$$

This proves the essential property 1b.

For $B'_G \subseteq B_G$ such that $|B'_G| \le \mathcal{O}(\beta |B_G|)$ and $A = \frac{1}{|B'_G|} \sum_{b \in B'_G} (\bar{\mu}_b^j - p^j)(\bar{\mu}_b^j - p^j)$, a similar calculation as in Equation (4.14) using the third statement of Theorem 65, shows

$$\langle M, \sum_{b \in B'_G} (\bar{\mu}_b^j - p^j)^{\otimes 2}\rangle \le \mathcal{O}\bigg(|B_G| \frac{\Delta^2}{\beta \log^4 j}\bigg).$$

Then from Equation (4.16)

$$|B'_G|\langle M, (\bar{p}^j_{B'_G} - p^j)^{\otimes 2}\rangle = \langle M, \sum_{b\in B'_G} (\bar{\mu}^j_b - p^j)^{\otimes 2}\rangle - \langle M, \sum_{b\in B'_G} (\bar{\mu}^j_b - \bar{p}^j_{B'_G})^{\otimes 2}\rangle$$
$$\leq \langle M, \sum_{b\in B'_G} (\bar{\mu}^j_b - p^j)^{\otimes 2}\rangle,$$

Combining the last two equations we get the last essential property 2. ■

4.13 Cover of set $\mathcal{A}_k$

We recall some basic concepts and results in VC theory, and use them to derive some simple consequences for the set $\mathcal{A}_k$ that we use to derive Theorem 65 in the next section. Let $\mathcal{S}$ be a collection of subsets of $\mathbb{R}$. The *VC shatter coefficient* of $\mathcal{S}$ is

$$SC_{\mathcal{S}}(t) := \sup_{x_1,x_2,..,x_t\in\mathbb{R}} |\{\{x_1, x_2, .., x_t\} \cap S : S \in \mathcal{S}\}|,$$

the largest number of subsets of t elements in obtained by intersections with subsets in $\mathcal{S}$. The VC dimension of $\mathcal{S}$ is

$$V_{\mathcal{S}} := \sup\{t : SC_{\mathcal{S}}(t) = 2^t\},$$

the largest number of $\mathbb{R}$ elements that are "fully shattered" by $\mathcal{S}$. The following Lemma [43] bounds the Shatter coefficient for a VC family of subsets.

Lemma 66 ([43])**.** *For all $t \geq V_{\mathcal{S}}$,* $\quad SC_{\mathcal{S}}(t) \leq \left(\frac{t\,e}{V_{\mathcal{S}}}\right)^{V_{\mathcal{S}}}$.

Next we state the VC-inequality for relative deviation [144, 7].

Theorem 67. *Let q be a distribution over $\mathbb{R}$, and $\mathcal{S}$ be a VC-family of subsets of $\mathbb{R}$ and $\bar{q}_t$ denote the empirical distribution from t i.i.d samples from q. Then for any $\epsilon, \delta > 0$ and $t = \Omega(\frac{V_{\mathcal{S}}+\log 1/\epsilon}{\epsilon^2})$,* *with probability* $\geq 1-\delta$,

$$\sup_{S\in\mathcal{S}} \max\left\{\frac{\bar{q}_t(S)-q(S)}{\sqrt{\bar{q}_t(S)}}, \frac{q(S)-\bar{q}_t(S)}{\sqrt{q(S)}}\right\} \leq \epsilon.$$

Another important ingredient commonly used in VC Theory is the concept of covering number that reflects the smallest number of subsets that approximate each subset in the collection.

Let q be any probability measure over $\mathbb{R}$ and $\mathcal{S}$ be any arbitrary collection of real sets. A collection of real sets $\mathcal{C}$ is an ϵ*-cover* of $\mathcal{S}$ under distribution q if for any $S\in\mathcal{S}$, there exists a $S'\in\mathcal{C}$ with $q(S\triangle S')\leq\epsilon$. The ϵ*-covering number* of $\mathcal{S}$ is

$$N(\mathcal{S}, q, \epsilon) := \inf\{|\mathcal{C}| : \mathcal{C} \text{ is an } \epsilon\text{-cover of } \mathcal{S}\}.$$

If $\mathcal{C}\subseteq\mathcal{S}$ is an ϵ*-cover* of $\mathcal{S}$, then $\mathcal{C}$ is an ϵ*-self cover* of $\mathcal{S}$. The ϵ*-self-covering number* of $\mathcal{S}$ is

$$N^s(\mathcal{S}, q, \epsilon) := \inf\{|\mathcal{C}| : \mathcal{C} \text{ is an } \epsilon\text{-self-cover of } \mathcal{S}\}.$$

Clearly, $N^s(\mathcal{S}, q, \epsilon)\geq N(\mathcal{S}, q, \epsilon)$, and we establish the reverse relation.

Lemma 68. *For any* $\epsilon\geq 0$, $N^s(\mathcal{S}, q, \epsilon)\leq N(\mathcal{S}, q, \epsilon/2)$.

Proof. If $N(\mathcal{S}, q, \epsilon/2)=\infty$, the lemma clearly holds. Otherwise, let $\mathcal{C}$ be an $\epsilon/2$-cover of size $N(\mathcal{S}, q, \epsilon/2)$. We construct an ϵ-self-cover of equal or smaller size.

For every subset $S_{\mathcal{C}}\in\mathcal{C}$, there is a subset $S=f(S_{\mathcal{C}})\in\mathcal{S}$ with $q(S_{\mathcal{C}}\triangle f(S_{\mathcal{C}}))\leq\epsilon/2$. Otherwise, $S_{\mathcal{C}}$ could be removed from $\mathcal{C}$ to obtain a strictly smaller $\epsilon/2$ cover, which is impossible.

The collection $\{f(S_{\mathcal{C}}) : S_{\mathcal{C}}\in\mathcal{C}\}\subseteq\mathcal{S}$ has size $\leq|\mathcal{C}|$, and it is an ϵ-self-cover of $\mathcal{S}$ because for any $S\in\mathcal{S}$, there is an $S_{\mathcal{C}}\in\mathcal{C}$ with $q(S\triangle S_{\mathcal{C}})\leq\epsilon/2$, and by the triangle inequality, $q\big(S\triangle f(S_{\mathcal{C}})\big)\leq\epsilon$. ■

Let $N_{\mathcal{S},\epsilon} := \sup_q N(\mathcal{S}, q, \epsilon)$ and $N^s_{\mathcal{S},\epsilon} := \sup_q N^s(\mathcal{S}, q, \epsilon)$ be the largest covering numbers under any distribution.

The next theorem bounds the covering number of $\mathcal{S}$ in terms of its VC-dimension.

Theorem 69 ([143]). *For some universal constant c, for all families $\mathcal{S}$ and $\epsilon > 0$,*

$$N_{\mathcal{S},\epsilon} \le c \cdot V_{\mathcal{S}} \cdot (4e/\epsilon)^{V_{\mathcal{S}}}.$$

Combining the theorem and Lemma 68, we obtain the following corollary.

Corollary 70. *For some universal constant c, for all families $\mathcal{S}$ and $\epsilon > 0$,*

$$N^s_{\mathcal{S},\epsilon} \le c \cdot V_{\mathcal{S}} \cdot (8e/\epsilon)^{V_{\mathcal{S}}}.$$

It is easy to see that the VC dimension of $\mathcal{A}_k$ is $\mathcal{O}(k)$, hence

Corollary 71. *For any k and $0 < \epsilon < 1$,*

$$N^s_{\mathcal{A}_k,\epsilon} \le \mathcal{O}(k) \cdot \left(\frac{8e}{\epsilon}\right)^{\mathcal{O}(k)} \le \exp\left(\mathcal{O}\left(k \log \frac{2}{\epsilon}\right)\right).$$

Therefore for any distribution q, $\mathcal{A}_k$ has an ϵ self cover of the above size.

4.14 Concentration inequalities for good batches

We use the sub-gaussian distribution of the empirical frequencies $\bar{\mu}_b(S)$ of good batches $b \in B_G$ to derive Theorem 65.

Recall B_G denotes the collection of all good batches. We use B'_G, B''_G, etc to denote sub-collections of good batches, and S, S', etc. to denote subsets of $\mathbb{R}$.

For any subset S, let $p(S)$, $\bar{\mu}_b(S)$, and $\bar{p}_{B'_G}(S)$ denote the probabilities assigned to S by the underlying distribution p, the empirical distribution $\bar{\mu}_b$ of the n samples in a batch b, and the empirical distribution $\bar{p}_{B'_G}$ of a sub-collection $B'_G \subseteq B_G$, respectively.

One cannot hope for a meaningful concentration for $\bar{p}_{B'_G}(S)$ for all $S \subseteq \mathbb{R}$. For the collection S of all samples in good batches, $\bar{p}_{B'_G}(S) = 1$, and yet, since p is continuous and S is finite, $p(S) = 0$. Recall that $\Delta = \frac{\beta}{\sqrt{n}}\sqrt{\log\frac{1}{\beta}}$ is our desired TV-distance accuracy. To achieve it, it suffices to prove concentration bounds for all subsets $S \in \mathcal{A}_{k\cdot i}$ for $1 \le i \le \frac{1}{\Delta}$. As expected, since $\mathcal{A}_{k\cdot i}$ grows with i, so will the slack in the bounds we derive.

Since the size of sets $\mathcal{A}_{k\cdot i}$ is infinite, we first show the concentration for the appropriate covers of sets $\mathcal{A}_{k\cdot i}$, and then we extend it to the set $\mathcal{A}_{k\cdot i}$ itself.

Corollary 71 showed that under any distribution q and for every $\epsilon > 0$, $\mathcal{A}_k$ has an ϵ self cover of size $\exp\mathcal{O}(k\log\frac{2}{\epsilon})$. It follows that for any $i \ge 1$, the set $\mathcal{A}_{k\cdot i}$ has a Δ^2-self cover $\mathcal{C}_i$ of size $\le \exp\mathcal{O}(k\log\frac{1}{\Delta})$ under the underlying distribution p.

First we show concentration of all good batches B_G, later we extend this concentration to any of their large enough sub-collections B'_G.

For a good batch $b \in B_G$ and $S \subseteq \mathbb{R}$, $n \cdot \bar{\mu}_b(S)$ follows the binomial distribution $\text{Bin}(p(S), n)$. It follows that $E[\bar{\mu}_b(S)] = p(S)$ and $\bar{\mu}_b(S) - p(S) \sim \text{subG}(1/4n)$ has sub-gaussian tails, *e.g.,* [121]. We use this fact to show that for all subsets in cover $\mathcal{C}_i$ the difference $|\sum_{b\in B_G}(\bar{\mu}_b(S) - p(S))|$ is small. Recall that $|B_G| = \tilde{\Omega}(\frac{k+\log(1/\delta)}{\beta^2})$ and $\Delta = \frac{\beta}{\sqrt{n}}\sqrt{\log\frac{1}{\beta}}$.

Lemma 72. *For any $1 \le i \le \frac{1}{\Delta}$, with probability $\ge 1-\delta$,*

$$\max_{S\in\mathcal{C}_i}\Big|\sum_{b\in B_G}(\bar{\mu}_b(S) - p(S))\Big| \le \mathcal{O}\Big(|B_G|\frac{\sqrt{i}\cdot\beta}{\sqrt{n}\log^3(\frac{1}{\Delta})}\Big).$$

Proof. From Hoeffding's inequality for subgaussians, *e.g.,* [121],

$$\Pr\Big[\Big|\sum_{b\in B_G}(\bar{\mu}_b(S) - p(S))\Big| \ge |B_G|\epsilon\Big] \le e^{-\Omega(|B_G|\epsilon^2 n)}.$$

Taking a union over all subsets in cover of $\mathcal{C}_i$, we get $\forall S \in \mathcal{C}_i$

$$\Pr\Big[\max_{S\in\mathcal{C}_i}\Big|\sum_{b\in B_G}(\bar{\mu}_b(S) - p(S))\Big| \ge |B_G|\epsilon\Big] \le |\mathcal{C}_i|\exp\big(-\Omega(|B_G|\epsilon^2 n)\big)$$
$$\le \exp\Big(-\Omega(|B_G|\epsilon^2 n - k\cdot i\cdot\log\frac{1}{\Delta})\Big).$$

If $|B_G| = \tilde{\Omega}(\frac{k+\log(1/\delta)}{\beta^2})$ and $\epsilon = \Omega(\frac{\sqrt{i}\cdot\beta}{\sqrt{n}\log^3(\Delta)})$, then $|B_G|\epsilon^2 n = \Omega(k \cdot i \cdot \log \Delta)$, and the lemma follows. ■

For any subsets S and S' of reals, let

$$Y_b(S,S') := (\bar{\mu}_b(S) - p(S))(\bar{\mu}_b(S') - p(S')), \text{ and } Y_b^c(S,S') := Y_b(S,S') - \mathbb{E}[Y_b(S,S')].$$

Note that random variables $Y_b(S,S')$ and its centered version $Y_b^c(S,S')$, both are symmetric in S and S'. Let $Y_b(S) := Y_b(S,S)$ and $Y_b^c(S) := Y_b^c(S,S)$.

We note a few properties of $\mathbb{E}[Y_b(S,S')]$ that will be useful. First note that $\mathbb{E}[Y_b(S,S')]$ is the covariance of $\bar{\mu}_b(S)$ and $\bar{\mu}_b(S')$ and $\mathbb{E}[Y_b(S)]$ is the covariance of $\bar{\mu}_b(S)$. It can be easily shown that

$$\mathbb{E}[Y_b(S,S')] = \frac{p(S\cap S') - p(S)p(S')}{n}.$$

It follows

$$\mathbb{E}[Y_b(S)] = \frac{p(S)(1-p(S))}{n} \le \frac{p(S)}{n}. \tag{4.17}$$

From Cauchy schwarz inequality

$$\left|\mathbb{E}[Y_b(S,S')]\right| \le \mathbb{E}[|Y_b(S,S')|] \le \sqrt{\mathbb{E}[Y_b(S)]\cdot\mathbb{E}[Y_b(S')]} \le \frac{\sqrt{p(S)\cdot p(S')}}{n}. \tag{4.18}$$

For a good batch $b \in B_G$ and $S \subseteq R$, recall that $\bar{\mu}_b(S) - p(S) \sim$ subG$(1/4n)$, and since the product of two sub exponential random variables follows sub expnential [121] distribution, it follows

$$Y_b(S,S') - E[Y_b(S,S')] = Y_b^c(S,S') \sim \text{subE}(\frac{16}{4n}) = \text{subE}(\frac{4}{n}).$$

Here subE is sub exponential distribution.

We first focus on the case when $S = S'$. We obtain the following concentration bound on sum of $Y_b^c(S)$.

Lemma 73. *For any* $1 \leq i \leq \frac{1}{\Delta}$ *and* $|B_G| = \tilde{\Omega}(\frac{k+\log(1/\delta)}{\beta^2})$*, with probability* $\geq 1-\delta$

$$\max_{S \in \mathcal{C}_i} \Big| \sum_{b \in B_G} Y_b^c(S) \Big| \leq \mathcal{O}\Big(|B_G| \frac{i \cdot \beta}{n \log^6 \frac{1}{\Delta}}\Big).$$

Proof. Bernstein's inequality gives:

$$\Pr[\Big| \sum_{b \in B_G} Y_b^c(S) \Big| \geq |B_G|\epsilon] \leq \exp\left(-\Omega(|B_G|\epsilon n \cdot \min\{1, \epsilon n\})\right).$$

Taking union over all subsets S in $\mathcal{C}_i$, we get

$$\begin{aligned} \Pr\left[\max_{S \in \mathcal{C}_i} \Big| \sum_{b \in B_G} Y_b^c(S) \Big| \geq |B_G|\epsilon\right] &\leq |\mathcal{C}_i| \exp\left(-\Omega(|B_G|\epsilon n \cdot \min\{1, \epsilon n\})\right) \\ &\leq \exp\left(-\Omega(|B_G|\epsilon n \cdot \min\{1, \epsilon n\} - k \cdot i \cdot \log \frac{1}{\Delta})\right). \end{aligned}$$

If $|B_G| = \tilde{\Omega}(\frac{k+\log(1/\delta)}{\beta^2})$ and $\epsilon = \Omega(\frac{i \cdot \beta}{n \log^6 \frac{1}{\Delta}})$, then

$$|B_G|\epsilon n \cdot \min\{1, \epsilon n\} = \Omega(k \cdot i \cdot \log \frac{1}{\Delta}),$$

hence the lemma follows. ■

The next lemma shows concentration of sum of $Y_b^c(S, S')$ for good batches.

Lemma 74. *For any* $1 \leq i' \leq i \leq \frac{1}{\Delta}$ *and* $|B_G| = \tilde{\Omega}(\frac{k+\log(1/\delta)}{\beta^2})$*, with probability* $\geq 1-\delta$

$$\max_{S \in \mathcal{C}_i} \max_{S' \in \mathcal{C}_{i'} : p(S') \leq \frac{2i'}{i}} \Big| \sum_{b \in B_G} Y_b^c(S, S') \Big| \leq \mathcal{O}\Big(|B_G| \frac{\sqrt{i \times i'} \cdot \beta}{n \log^6(\frac{1}{\Delta})}\Big).$$

Proof. Bernstein's inequality yields,

$$\Pr[\Big|\sum_{b\in B_G} Y_b^c(S,S')\Big| \geq |B_G|\epsilon] \leq \exp\left(-\Omega(|B_G|\epsilon n \cdot \min\{1,\epsilon n\})\right).$$

Taking union over all subsets S in $\mathcal{C}_i$ and $S' \in \mathcal{C}_{i'}$, we get

$$\Pr\left[\max_{S\in\mathcal{C}_i}\max_{S'\in\mathcal{C}_{i'}}\Big|\sum_{b\in B_G} Y_b^c(S,S')\Big| \geq |B_G|\epsilon\right] \leq |\mathcal{C}_i||\mathcal{C}_{i'}|\exp\left(-\Omega(|B_G|\epsilon n\cdot\min\{1,\epsilon n\})\right)$$
$$\leq \exp\left(-\Omega(|B_G|\epsilon n\cdot\min\{1,\epsilon n\} - k\cdot i\cdot\log\frac{1}{\Delta} - k\cdot i'\cdot\log\frac{1}{\Delta})\right)$$
$$\leq \exp\left(-\Omega(|B_G|\epsilon n\cdot\min\{1,\epsilon n\} - k\cdot i\cdot\log\frac{1}{\Delta})\right),$$

here the last inequality uses $i \leq i'$.

Let $\epsilon = \Omega(\frac{\sqrt{ii'}\cdot\beta}{n\log^6(n/\beta)})$. First consider the case when $\epsilon n \leq 1$, then using $|B_G| = \tilde{\Omega}(\frac{k+\log 1/\delta}{\beta^2})$, we get

$$|B_G|(\epsilon n)^2 = \Omega(k\cdot i\cdot\log\frac{1}{\Delta}),$$

hence the lemma holds with probability $\geq 1-\delta$ for this case.

Next, consider the case $\epsilon n \geq 1$. This implies $\frac{\sqrt{ii'}\cdot\beta}{\log^6\frac{1}{\Delta}} = \Omega(1)$. Since $i \geq i'$, hence

$$\frac{i\cdot\beta}{\log^6\frac{1}{\Delta}} = \Omega(1). \tag{4.19}$$

The previous lemma showed that for all $S \in \mathcal{C}_i$, with probability $\geq 1-\delta$,

$$\Big|\sum_{b\in B_G} Y_b^c(S)\Big| \leq \mathcal{O}\Big(|B_G|\frac{i\cdot\beta}{n\log^6(\frac{1}{\Delta})}\Big).$$

Therefore,

$$\begin{aligned}
\Big|\sum_{b\in B_G} Y_b(S)\Big| &\le \Big|\sum_{b\in B_G}(Y_b^c(S)+\mathbb{E}[Y_b(S)])\Big| \\
&\overset{(a)}{\le} \mathcal{O}\Big(|B_G|\frac{i\cdot\beta}{n\log^6(\frac{1}{\Delta})}\Big)+|B_G|\mathbb{E}[Y_b(S)] \\
&\overset{(b)}{\le} \mathcal{O}\Big(|B_G|\frac{i\cdot\beta}{n\log^6(\frac{1}{\Delta})}\Big)+|B_G|\frac{1}{n} \\
&\le \frac{|B_G|}{n}\mathcal{O}\Big(1+\frac{i\cdot\beta}{\log^6(\frac{1}{\Delta})}\Big) \\
&\overset{(c)}{\le} \frac{|B_G|}{n}\mathcal{O}\Big(\frac{i\cdot\beta}{\log^6(\frac{1}{\Delta})}\Big),
\end{aligned}$$

here (a) uses the previous inequality, (b) uses (4.17) and $p(S)\le 1$ and (c) uses (4.19).

Similarly,

$$\begin{aligned}
\Big|\sum_{b\in B_G} Y_b(S')\Big| &\le \mathcal{O}\Big(|B_G|\frac{i\cdot\beta}{n\log^6(\frac{1}{\Delta})}\Big)+|B_G|\frac{p(S')}{n} \\
&\overset{(a)}{\le} \mathcal{O}\Big(|B_G|\frac{i'\cdot\beta}{n\log^6(\frac{1}{\Delta})}\Big)+\mathcal{O}(|B_G|\frac{i'}{i\cdot n}) \\
&\le |B_G|\frac{i'}{i\cdot n}\mathcal{O}\Big(1+\frac{i\cdot\beta}{\log^6(\frac{1}{\Delta})}\Big) \\
&\overset{(b)}{\le} \frac{|B_G|}{n}\mathcal{O}\Big(\frac{i'\cdot\beta}{\log^6(\frac{1}{\Delta})}\Big),
\end{aligned}$$

here (a) uses $p(S)\le\frac{2i'}{i}$ and (b) uses (4.19).

Applying the Cauchy-Schwarz inequality gives

$$\sum_{b\in B_G} Y_b(S,S') \le \sqrt{(\sum_{b\in B_G} Y_b(S))(\sum_{b\in B_G} Y_b(S'))} \le \mathcal{O}\Big(|B_G|\frac{\sqrt{i\cdot i'}\cdot\beta}{n\log^6(\frac{1}{\Delta})}\Big).$$

Next, from (4.18)

$$|\mathbb{E}[Y_b(S,S')]| \leq \frac{\sqrt{p(S)\cdot p(S')}}{n} \overset{(a)}{\leq} \frac{\sqrt{2i'/i}}{n} = \frac{\sqrt{2i\cdot i'}}{i\cdot n} \overset{(b)}{\leq} \mathcal{O}\Big(\frac{\sqrt{i\cdot i'}\cdot\beta}{n\log^6(\frac{1}{\Delta})}\Big), \tag{4.20}$$

here (a) uses $p(S) \leq \frac{2i'}{i}$ and (b) uses (4.19). Combining the last two equations we get, for the case $\epsilon n \geq 1$ and $p(S') \leq \frac{2i'}{i}$

$$\begin{aligned}|\sum_{b\in B_G} Y_b^c(S,S')| &= |\sum_{b\in B_G}(Y_b(S,S') - \mathbb{E}[Y_b(S,S')])| \\ &\leq |\sum_{b\in B_G} Y_b(S,S')| + |B_G|\mathbb{E}[Y_b(S,S')] \leq \mathcal{O}\Big(\frac{\sqrt{i\cdot i'}\cdot\beta}{n\log^6(\frac{1}{\Delta})}\Big).\end{aligned}$$ ■

Next, to establish the concentration for all sub-collection of size $(1-\beta)|B_G|$, first we establish the concentration for all sub-collections of size $\beta|B_G|$.

The following bounds on the number of sub-collection of B_G smaller than a particular size will be useful.

Lemma 75. *For any $f \leq 1$ and collection B_G the number of sub-collections of B_G of size $\leq f|B_G|$*

$$|\{B'_G : B'_G \subseteq B_G, |B'_G| \leq f|B_G|\}| \leq \exp\big(\mathcal{O}(|B_G|\cdot f\log\frac{e}{f})\big).$$

Proof. For $f \geq \frac{1}{2}$, the lemma follows as number of subsets of B_G are $2^{|B_G|}$. For $f \leq \frac{1}{2}$, the proof follows by combining a simple counting argument and the Stirling's approximation,

$$\begin{aligned}\sum_{j=1}^{\lfloor f|B_G|\rfloor}\binom{|B_G|}{j} &\leq f|B_G|\binom{|B_G|}{\lfloor f|B_G|\rfloor} \\ &\leq f|B_G|\Big(\frac{e|B_G|}{f|B_G|}\Big)^{f|B_G|} \leq e^{f|B_G|\ln(e/f)+\ln(f|B_G|)} < e^{2f|B_G|\ln(e/f)},\end{aligned}$$

where last of the above inequality used $\ln(f|B_G|) < f|B_G|$ and $\ln(e/f) \geq 1$. ■

First we deal with the subsets in $\mathcal{C}_i$ for $i = \Omega(\log^7(\frac{1}{\Delta}))$, and get the following concentration bound.

Lemma 76. *For any $\Omega(\log^7(\frac{1}{\Delta})) \leq i \leq \frac{1}{\Delta}$ and $|B_G| = \tilde{\Omega}(\frac{k+\log 1/\delta}{\beta^2})$, with probability $\geq 1-\delta$,*

$$\max_{B'_G : |B'_G| \leq 2\beta|B_G|} \max_{S \in \mathcal{C}_i} \Big| \sum_{b \in B'_G} Y_b^c(S) \Big| \leq \mathcal{O}\Big(|B_G| \frac{i \cdot \beta}{n \log^6(\frac{1}{\Delta})}\Big).$$

Proof. Consider any sub-collection $B'_G \subseteq B_G$ of size $|B'_G| \leq f|B_G|$. Applying Bernstein's inequality for B'_G gives:

$$\begin{aligned}
&\Pr[\Big| \sum_{b \in B'_G} Y_b^c(S) \Big| \geq |B_G|\epsilon] \leq \Pr[\Big| \sum_{b \in B'_G} Y_b^c(S) \Big| \geq |B'_G| \frac{|B_G|}{|B'_G|}\epsilon] \\
&= \exp\left(-\Omega(|B_G|\epsilon n \cdot \min\{1, \frac{|B_G|}{|B'_G|}\epsilon n\}) \right) = \exp\left(-\Omega(|B_G|\epsilon n \cdot \min\{1, \frac{\epsilon n}{f}\}) \right).
\end{aligned}$$

Taking union over all subsets S in cover $\mathcal{C}_i$, we get

$$\begin{aligned}
\Pr\left[\max_{S \in \mathcal{C}_i} \Big| \sum_{b \in B'_G} Y_b^c(S) \Big| \geq |B_G|\epsilon \right] &\leq |\mathcal{C}_i| \exp\left(-\Omega(|B_G|\epsilon n \cdot \min\{1, \frac{\epsilon n}{f}\}) \right) \\
&\leq \exp\left(-\Omega(|B_G|\epsilon n \cdot \min\{1, \frac{\epsilon n}{f}\} - k \cdot i \cdot \log\frac{1}{\Delta}) \right).
\end{aligned}$$

Taking union over all sub-collections B' of size $f|B_G|$, we get

$$\begin{aligned}
&\Pr\left[\max_{S \in \mathcal{C}_i} \max_{B'_G : |B'_G| \leq f|B_G|} \Big| \sum_{b \in B'_G} Y_b^c(S) \Big| \geq |B_G|\epsilon \right] \\
&\leq |\{B'_G : B'_G \subseteq B_G, |B'_G| \leq f|B_G|\}| \exp\left(-\Omega(|B_G|\epsilon n \cdot \min\{1, \frac{\epsilon n}{f}\} - k \cdot i \cdot \log\frac{1}{\Delta}) \right) \\
&\leq \exp\left(-\Omega(|B_G|\epsilon n \cdot \min\{1, \frac{\epsilon n}{f}\} - k \cdot i \cdot \log\frac{1}{\Delta} - |B_G| \cdot f \log\frac{e}{f}) \right). \qquad (4.21)
\end{aligned}$$

For $|B_G| = \tilde{\Omega}(\frac{k+\log 1/\delta}{\beta^2})$, $\epsilon = \Omega(\frac{i \cdot \beta}{n \log^6(\frac{1}{\Delta})})$, $f = 2\beta$ and $i = \Omega(\log^7(\frac{1}{\Delta}))$,

$$|B_G|\epsilon n \cdot \min\{1, \frac{\epsilon n}{f}\} = \Omega\Big(\max\Big\{ k \cdot i \cdot \log\frac{1}{\Delta}, |B_G| \cdot f \log\frac{e}{f} \Big\} \Big)$$

hence, the statement of the lemma follows. ■

We will need a more nuanced analysis for subsets $S \in \mathcal{C}_i$ for $i = \mathcal{O}(\log^5(\frac{1}{\Delta}))$.

Lemma 77. *For any* $n = \Omega(\log^6 \frac{1}{\Delta})$*,* $1 \le i \le \mathcal{O}(\log^7(\frac{1}{\Delta}))$ *and* $|B_G| = \tilde{\Omega}(\frac{k+\log 1/\delta}{\beta^2})$*, with probability* $\ge 1-\delta$*,* $\forall\, S \in \mathcal{C}_i$*,*

$$\max_{B'_G:|B'_G|\le 2\beta|B_G|}\Big|\sum_{b\in B'_G} Y_b^c(S)\Big| \le \mathcal{O}\bigg(\beta|B_G|\cdot\frac{\log(1/\beta)}{n}\cdot\max\Big(p(S)\log\Big(\frac{2}{p(S)}\Big),\frac{i}{\log^6\frac{1}{\Delta}}\Big)\bigg).$$

Proof. Recall that for any batch $b \in B_G$ and any subset S of reals $\bar{\mu}_b(S)$ is the mean of n samples from $\text{Ber}(p(S))$. For any $\epsilon > 0$, Chernoff bound implies

$$\Pr\Big[|\bar{\mu}_b(S)-p(S)|\ge\epsilon\Big] \le \exp\bigg(-\Omega\Big(\min\{\epsilon\cdot n,\frac{\epsilon^2\cdot n}{p(S)}\}\Big)\bigg).$$

Let

$$\varepsilon_1 = \sqrt{\frac{\log(1/\beta)}{n}\cdot\frac{1}{\log^6\frac{1}{\Delta}}},\ \varepsilon_2 = \sqrt{\frac{\log(1/\beta)}{n}\cdot p(S)\log\Big(\frac{2}{p(S)}\Big)},\ \text{and}\quad \varepsilon = \max\{\varepsilon_1,\varepsilon_2\}.$$

The above bound implies that

$$\begin{aligned}\Pr\Big[|\bar{\mu}_b(S)-p(S)|\ge\varepsilon\Big] &\le \exp\bigg(-\Omega\Big(\min\Big\{\varepsilon\cdot n,\frac{\varepsilon^2\cdot n}{p(S)}\Big\}\Big)\bigg)\\ &\le \exp\bigg(-\Omega\Big(\min\Big\{\varepsilon_1\cdot n,\frac{\varepsilon_2^2\cdot n}{p(S)}\Big\}\Big)\bigg)\\ &\le \exp\Bigg(-\Omega\bigg(\min\Big\{\sqrt{\frac{n\cdot\log(1/\beta)}{\log^6\frac{1}{\Delta}}},\Big(\log\frac{1}{\beta}\Big)\cdot\Big(\log\frac{2}{p(S)}\Big)\Big\}\bigg)\Bigg)\\ &\le \mathcal{O}\Big(\max\Big\{\frac{\beta}{\log^5\frac{1}{\Delta}},\beta\cdot p(S)\Big\}\Big),\end{aligned}$$

here the last inequality uses $n = \Omega(\log^6\frac{1}{\Delta})$.

Let ${B_G}^{\varepsilon}(S) = \{b \in B_G : |\bar{\mu}_b(S)-p(S)| \ge \varepsilon\}$. Then the above equation shows that if conditions in the lemma holds then $E[|{B_G}^{\varepsilon}(S)|] \le \mathcal{O}\Big(|B_G|\max\{\frac{\beta}{\log^5\frac{1}{\Delta}},\beta\cdot p(S)\}\Big)$.

From the Chernoff bound

$$\Pr\left[|{B_G}^{\varepsilon}(S)| \geq 2 \times E[|{B_G}^{\varepsilon}(S)|]\right] \leq \exp\left(-\Omega(E[|{B_G}^{\varepsilon}(S)|])\right) \leq \exp\left(-\Omega(|B_G|\frac{\beta}{\log^5 \frac{1}{\Delta}})\right).$$

Taking union bound over all subsets $S \in \mathcal{C}_i$

$$\Pr\left[\max_{S \in \mathcal{C}_i}\left\{|{B_G}^{\varepsilon}(S)| - 2 \times E[|{B_G}^{\varepsilon}(S)|]\right\} \geq 0\right] \leq \exp\left(-\Omega(|B_G|\frac{\beta}{\log^5 \frac{1}{\Delta}}) + \mathcal{O}(k \cdot i \log \frac{1}{\Delta})\right). \tag{4.22}$$

Provided $i = \mathcal{O}(\log^4 \frac{1}{\Delta})$ and $|B_G| = \tilde{\Omega}(\frac{k+\log 1/\delta}{\beta^2})$, then $|B_G|\frac{\beta}{\log^5 \frac{1}{\Delta}} = \Omega(k \cdot i \log \frac{1}{\Delta})$, hence the above probability is small.

For any subset $S \in \mathcal{C}_i$ and $b \in B_G \setminus {B_G}^{\varepsilon}(S)$,

$$\begin{aligned}|Y_b^c(S)| &\leq |Y_b(S)| + |\mathbb{E}[Y_b(S)]| \\ &= |\frac{p(S)(1-p(S))}{n}| + |\bar{\mu}_b(S) - p(S)|^2 \leq |\frac{p(S)}{n}| + \varepsilon^2 \leq \varepsilon_2^2 + \varepsilon^2 \leq 2\varepsilon^2,\end{aligned} \tag{4.23}$$

here we used the definition of sub collection ${B_G}^{\varepsilon}(S)$.

Next ,

$$\begin{aligned}\Big|\sum_{b \in B'_G} Y_b^c(S)\Big| &\leq \Big|\sum_{b \in B'_G \setminus {B_G}^{\varepsilon}(S)} Y_b^c(S)\Big| + \sum_{b \in {B_G}^{\varepsilon}(S)} \Big|Y_b^c(S)\Big| \\ &\leq |B'_G| \cdot \max_{b \in B'_G \setminus {B_G}^{\varepsilon}(S)} |Y_b^c(S)| + \max_{B''_G : |B''_G| \leq |{B_G}^{\varepsilon}(S)|} \sum_{b \in B''_G} \Big|Y_b^c(S)\Big|.\end{aligned} \tag{4.24}$$

Combining the two equations we get

$$\max_{B'_G : |B'_G| \leq 2\beta|B_G|} \Big|\sum_{b \in B'_G} Y_b^c(S)\Big| \leq 2\varepsilon^2 \cdot 2\beta|B_G| + \max_{B''_G : |B''_G| \leq |{B_G}^{\varepsilon}(S)|} \sum_{b \in B''_G} \Big|Y_b^c(S)\Big|.$$

Equation (4.22) showed that w.h.p. the for all sets $\mathcal{S} \in \mathcal{C}_i$, $|{B_G}^{\varepsilon}(S)| \leq \mathcal{O}\Big(|B_G| \max\{\frac{\beta}{\log^4 \frac{1}{\Delta}}, \beta \cdot$

$p(S)\}\Big)$. To complete the proof we bound the last term in the above equation using the concentration inequality in (4.21).

Let $f = \max\{\mathcal{O}(\frac{\beta}{\log^5 \frac{1}{\Delta}}), \mathcal{O}(\beta \cdot p(S))\}$. Then

$$
\begin{aligned}
f \log \frac{e}{f} &= \max\{\mathcal{O}(\frac{\beta}{\log^4 5\frac{1}{\Delta}} \log(\frac{\log^4 \frac{1}{\Delta}}{\beta})), \mathcal{O}(\beta \cdot p(S) \log(\frac{1}{\beta p(S)}))\} \\
&= \mathcal{O}\Big(\max\{\frac{\beta}{\log^4 \frac{1}{\Delta}}, \beta \cdot p(S) \log \frac{1}{\beta \cdot p(S)}\}\Big)
\end{aligned}
$$

Choose $\epsilon = \Omega\Big(\frac{\beta \log(1/\beta)}{n} \cdot \max\left(p(S) \log\left(\frac{2}{p(S)}\right), \frac{i}{\log^4 \frac{1}{\Delta}}\right)\Big)$, then

$$
\epsilon n = \Omega\Bigg(\max\left(\beta \cdot p(S)\Big(\log \frac{2}{p(S)}\Big)\Big(\log \frac{1}{\beta}\Big), \frac{i \cdot \beta \log(1/\beta)}{\log^4 \frac{1}{\Delta}}\right)\Bigg) = \Omega(f \log \frac{e}{f})
$$

Since $\epsilon n = \Omega(f \log \frac{e}{f})$,

$$
\epsilon n \cdot \min\{1, \frac{\epsilon n}{f}\} = \epsilon n = \Omega(f \log \frac{e}{f}).
$$

Further,

$$
\epsilon n \cdot \min\{1, \frac{\epsilon n}{f}\} = \epsilon n = \Omega\Big(\frac{i\beta \log(1/\beta)}{\log^4 \frac{1}{\Delta}}\Big).
$$

Then, for $|B_G| = \tilde{\Omega}(\frac{k + \log 1/\delta}{\beta^2})$

$$
|B_G| \epsilon n \cdot \min\{1, \frac{\epsilon n}{f}\} = |B_G| \epsilon n = \Omega\Big(|B_G| \frac{i\beta \log(1/\beta)}{\log^4 \frac{1}{\Delta}}\Big) = \Omega\Big(\max\Big\{k \cdot i \cdot \log \frac{1}{\Delta}\Big\}\Big)
$$

Then for $f = \max\{\mathcal{O}(\frac{\beta}{\log^5 \frac{1}{\Delta}}), \mathcal{O}(\beta \cdot p(S))\}$, the concentration inequality (4.21) shows that with probability $\geq 1 - \delta$, for all $S \in \mathcal{C}_i$

$$
\max_{B'_G : |B'_G| \leq f|B_G|} \Big| \sum_{b \in B'_G} Y_b^c(S) \Big| \leq |B_G| \epsilon.
$$

Combining the above equation with Equations (4.24) and (4.22) completes the proof. ■

By combining Lemma 76 and Lemma 77, we get

Corollary 78. *For any* $n = \Omega(\log^6 \frac{1}{\Delta})$, $1 \leq i \leq \frac{1}{\Delta}$ *and* $|B_G| = \tilde{\Omega}(\frac{k+\log 1/\delta}{\beta^2})$, *with probability* $\geq 1-\delta$, $\forall\, S \in \mathcal{C}_i$,

$$\max_{B'_G:|B'_G|\leq 2\beta|B_G|} \Big|\sum_{b\in B'_G} Y_b^c(S)\Big| \leq \mathcal{O}\bigg(\beta|B_G| \cdot \frac{\log(1/\beta)}{n} \cdot \max\Big(p(S)\log\Big(\frac{2}{p(S)}\Big), \frac{i}{\log^4 \frac{1}{\Delta}}\Big)\bigg).$$

Recall that $Y_b(S) = Y_b^c(S) + \mathbb{E}[Y_b(S)]$. And

$$|\mathbb{E}[Y_b(S)]| \leq \frac{p(S)}{n} \leq \mathcal{O}\bigg(\frac{1}{n} \cdot \max\Big(p(S)\log\Big(\frac{2}{p(S)}\Big), \frac{i}{\log^4 \frac{1}{\Delta}}\Big)\bigg).$$

Then the following corollary follows from the above.

Corollary 79. *For any* $n = \Omega(\log^6 \frac{1}{\Delta})$, $1 \leq i \leq \frac{1}{\Delta}$ *and* $|B_G| = \tilde{\Omega}(\frac{k+\log 1/\delta}{\beta^2})$, *with probability* $\geq 1-\delta$, $\forall\, S \in \mathcal{C}_i$,

$$\max_{B'_G:|B'_G|\leq 2\beta|B_G|} \Big|\sum_{b\in B'_G} Y_b(S)\Big| \leq \mathcal{O}\bigg(\beta|B_G| \cdot \frac{\log(1/\beta)}{n} \cdot \max\Big(p(S)\log\Big(\frac{2}{p(S)}\Big), \frac{i}{\log^4 \frac{1}{\Delta}}\Big)\bigg).$$

Next, recall that $Y_b(S) = (\bar{\mu}_b(S) - p(S))^2$ and

$$\xi(S,i) = \max\bigg(\sqrt{p(S)\log\Big(\frac{2}{p(S)}\Big)}, \frac{\sqrt{i}}{\log^3 \frac{1}{\Delta}}\bigg).$$

We derive the following results using the concentration bounds derived so far.

Theorem 80. *For* $|B_G| = \tilde{\Omega}(\frac{k+\log(1/\delta)}{\beta^2})$, *with probability* $\geq 1-\delta$, *for all* $1 \leq i' \leq i \leq \frac{1}{\Delta}$,

1. *For all* $B'_G \subseteq B_G$ *such that* $|B_G \setminus B'_G| \leq 2\beta|B_G|$,

$$\max_{S\in\mathcal{C}_i} |\bar{p}_{B'_G}(S) - p(S)| \leq \mathcal{O}\bigg(\Delta \cdot \xi(S,i)\bigg),$$

and

$$\max_{S\in\mathcal{C}_i}\ \max_{S'\in\mathcal{C}_{i'}:p(S')\le\frac{2i'}{i}}\Bigg|\frac{1}{|B'_G|}\sum_{b\in B'_G}(\bar{\mu}_b(S)-p(S))(\bar{\mu}_b(S')-p(S')) - \frac{1}{n}(p(S\cap S')-p(S)\cdot p(S'))\Bigg| \le \mathcal{O}\Big(\frac{\Delta^2}{\beta}\cdot\xi(S,i)\xi(S,i')\Big).$$

2. *For all $B'_G \subseteq B_G$, such that $|B_G| \le 2\beta|B_G|$,*

$$\max_{S\in\mathcal{C}_i}\ \max_{S'\in\mathcal{C}_{i'}}\Bigg|\sum_{b\in B'_G}(\bar{\mu}_b(S)-p(S))(\bar{\mu}_b(S')-p(S'))\Bigg| \le \mathcal{O}\Big(|B_G|\frac{\Delta^2}{\beta}\cdot\xi(S,i)\xi(S,i')\Big).$$

Proof. Using a simple relation $\sum_{i=1}^{j}|x_i| \le \sqrt{j\times\sum_{i=1}^{j}x_i^2}$, implied by the Jensen's inequality, in the above corollary gives:

$$\max_{B'_G:|B'_G|\le 2\beta|B_G|}\sum_{b\in B'_G}|\bar{\mu}_b(S)-p(S)| \le \mathcal{O}\Big(\xi(S,i)\beta|B_G|\sqrt{\frac{\log(1/\beta)}{n}}\Big) \le \mathcal{O}\Big(\xi(S,i)|B_G|\Delta\Big).$$

Lemma 72 showed

$$\Big|\sum_{b\in B_G}(\bar{\mu}_b(S)-p(S))\Big| \le \mathcal{O}\Big(|B_G|\frac{\sqrt{i}\cdot\beta}{\sqrt{n}\log^3(\frac{1}{\Delta})}\Big) \le \mathcal{O}\Big(\xi(S,i)|B_G|\Delta\Big).$$

Then for any $B'_G \subseteq B_G$ such that $|B_G\setminus B'_G| \le \mathcal{O}(\beta|B_G|)$,

$$\begin{aligned}|B'_G||\bar{p}_{B'_G}(S)-p(S)| &= \Big|\sum_{b\in B'_G}(\bar{\mu}_b(S)-p(S))\Big| \\ &\le \Big|\sum_{b\in B_G}(\bar{\mu}_b(S)-p(S))\Big| + \Big|\sum_{b\in B_G\setminus B'_G}(\bar{\mu}_b(S)-p(S))\Big| \\ &\le \mathcal{O}\Big(\xi(S,i)|B_G|\Delta\Big),\end{aligned}$$

where the last inequality uses $|B_G\setminus B'_G| \le 2\beta|B_G|$ and the previous two inequality. Finally noting that $\frac{|B_G|}{B'_G} = \frac{1}{(1-2\beta)} = \mathcal{O}(1)$ shows the first part

$$|\bar{p}_{B'_G}(S) - p(S)| \leq \mathcal{O}\Big(\xi(S,i)\Delta\Big).$$

Recall $Y_b(S,S') = (\bar{\mu}_b(S) - p(S))(\bar{\mu}_b(S') - p(S'))$. From Cauchy Schwarz inequality, it follows that

$$\sum_{b \in B'_G} |Y_b(S,S')| \leq \sum_{b \in B'_G} \sqrt{Y_b(S)Y_b(S')}.$$

From the above equation and Corollary 79, we get

$$\max_{B'_G : |B'_G| \leq 2\beta|B_G|} \sum_{b \in B'_G} \Big|Y_b(S,S')\Big| \leq \mathcal{O}\bigg(\beta|B_G| \cdot \frac{\log(1/\beta)}{n} \cdot \xi(S,i)\xi(S,i')\bigg).$$

This completes the proof of the third part of the Theorem.

Next, using the equation (4.20), and above equation

$$\begin{aligned}
\max_{B'_G : |B'_G| \leq 2\beta|B_G|} \sum_{b \in B'_G} \Big|Y^c_b(S,S')\Big| &\leq \max_{B'_G : |B'_G| \leq 2\beta|B_G|} \sum_{b \in B'_G} \Big|Y^c_b(S,S')\Big| + \sum_{b \in B'_G} \Big|\mathbb{E}[Y_b(S,S')]\Big| \\
&\leq \mathcal{O}\bigg(\beta|B_G| \cdot \frac{\log(1/\beta)}{n} \cdot \xi(S,i)\xi(S,i')\bigg).
\end{aligned}$$

Combining the above equation with Lemma 74 we get

$$\begin{aligned}
\max_{S \in \mathcal{C}_i} \max_{S' \in \mathcal{C}_{i'} : p(S') \leq \frac{2i'}{i}} \Big|\sum_{b \in B_G} Y^c_b(S,S')\Big| &\leq \mathcal{O}\Big(|B_G| \frac{\sqrt{i \times i'} \cdot \beta}{n \log^6(\frac{1}{\Delta})}\Big) \\
&\leq \mathcal{O}\bigg(\beta|B_G| \cdot \frac{\log(1/\beta)}{n} \cdot \xi(S,i)\xi(S,i')\bigg).
\end{aligned}$$

Combining the last two bounds we get the second part of the Theorem. ■

The previous Theorem shows the concentration for covers of sets $\mathcal{A}_{k \cdot i}$. These properties can be then extended to all elements in the sets $\mathcal{A}_{k \cdot i}$, using the standard VC theoretic arguments as in [76].

Extending these properties to sets $\mathcal{A}_{k\cdot i}$ completes the proof of the Theorem 65.

Chapter 4, in full, is a reprint of the material as it appears in Robust Density Estimation from Batches: The Best Things in Life are (Nearly) Free 2021. Ayush Jain, Alon Orlitsky. In ICML 2021. The dissertation author was the primary investigator and author of this paper.

Chapter 5

Efficient List-Decodable Regression using Batches

5.1 Introduction

Linear regression is one of the most fundamental tasks in supervised learning with applications in various sciences and industries [107, 58]. In the standard linear regression setup, one is given m samples (x_i, y_i) such that $y_i = \langle w^*, x_i \rangle + n_i$ where n_i is the observation noise with bounded variance and the covariates $x_i \in \mathbb{R}^d$ are drawn i.i.d from some fixed distribution. For this setup, the commonly used least-squares estimator that minimizes the square loss $\sum_i (y_i - \langle w, x_i \rangle)^2$, provides a good estimate of the unknown regression vector w^*.

In many applications, some samples are inadvertently or maliciously corrupted, for example, due to mislabeling or measurement errors, or data poisoning attacks. For instance, such corruptions are commonplace in biology [129, 119] and machine learning security [15, 19]. Even a small number of corrupt samples in the data can cause the least-squares estimator to fail catastrophically. Classical robust estimators have been proposed in [75, 130] but they suffer from exponential runtime. Recent works [99, 46, 48] have derived efficient algorithms for robust mean estimation with provable guarantees even when a small fraction of the data can be corrupt or adversarial. These works have inspired the efficient algorithms for robust regression [123, 50, 57, 120] under the same corruption model. [38, 80] have obtained robust regression algorithms with near-optimal run time and sample complexity.

In this paper, we are interested in the setting where a small fraction α, potentially even less than half, of the data is considered inlier, and the majority of the data may be influenced by factors such as adversarial manipulation, corruption, bias, or being drawn from a diverse distribution. This setting also encompasses the problem of learning a mixture of regressions [84, 154, 95, 117] because any solution of the former immediately yields a solution to the latter by setting α to be the proportion of the data from the smallest mixture component.

However, it is information-theoretically impossible to output a single accurate estimate of regression parameter when $\alpha < 1/2$. Instead, it may be possible to generate a short list of estimates such that at least one of them is accurate. This relaxed notion of learning is known as *list-decodable learning* and is useful since a learner can identify a single accurate estimate from the list given a small number of reliable samples.

For high dimensional mean estimation, [29] derived the first polynomial time algorithm for list decodable setting. List-decodable linear regression has been studied in [87, 126] yielding algorithms with runtime and sample complexity of $O(d^{\text{poly}(1/\alpha)})$. In contrast to list-decodable mean estimation, recent work [53] has shown that a sub-exponential runtime and sample complexity might be impossible for linear regression. These prior results may lead to a pessimistic conclusion for obtaining practical algorithms for the fundamental learning paradigm of linear regression when less than half of the data may be inlier or genuine.

However, our work demonstrates that it can be overcome in various real-world applications such as federated learning [147], learning from multiple sensors [149], and crowd-sourcing [136]. In these and many other applications individual data sources often provide multiple samples. We refer to a collection of samples from a single source as a *batch*. If a fraction α of the sources follow the underlying distribution we aim to learn, then α fraction of the batches will contain independent samples from that distribution, while the remaining batches may contain arbitrary samples.

When each batch contains $\tilde{\Omega}(d)$ samples then one can get the estimate of the regression vectors for each batch. However, typically in modern applications the dimension of the data is high and only a moderate number of samples are available per batch [66, 118, 95]. As we show in this paper, for any $\alpha \in (0, 1]$, as long as the number of samples provided by each genuine source is more than a small threshold of $\tilde{\Omega}(1/\alpha)$, we can use the grouping of samples in batches to develop a polynomial-time algorithm.

The batch setting has a natural advantage in the context of list-decodable learning. When there are multiple possible inlier distributions for the data sources, the list will include regression vectors for all distributions that underlie more than α fraction of sources. To determine the best-fitting solution for a specific source from the short list generated by the list-decodable algorithm, a small hold-out portion of the batch provided by that source can be used. This post hoc identification of the best weight for a source/batch is naturally not feasible in the single sample setting.

This motivates the problem of list-decodable linear regression using batches. Formally, there are m batches. Each batch has a collection of $\geq n$ regression samples which can either all come from a global regression model with true weight w^* (good batch) and noise variance σ^2 or are arbitrarily corrupted (adversarial batch). The task is to output a small list of regression vectors at least one of which is approximately correct given that only α fraction of the batches are good. It is important to highlight that in this scenario, any algorithm aiming to provide reasonable estimation guarantees must return a list of estimates. This is because the formulation allows for data to stem from $\Theta(1/\alpha)$ different distributions, each of which generates at least α fraction of the batches. The regression parameters for each of these distributions can vary arbitrarily. Without any method to identify the genuine distribution among these $\Theta(1/\alpha)$ possibilities, any algorithm providing a single estimate of the regression parameter would fail to offer a meaningful estimation guarantee.

Our main result is the following theorem:

Theorem 81 (Informal). *For any $\alpha \in (0, 1]$, there exists a polynomial time algorithm for list-decodable regression, that uses $m = \tilde{O}_{n,\alpha}(d)$ batches each of size $n = \tilde{\Omega}(1/\alpha)$, and* *outputs $O(1/\alpha^2)$ weights such that with high probability at least one of them, $\tilde{w}$, satisfies* $\|\tilde{w} - w^*\|_2 = \tilde{\mathcal{O}}(\sigma/\sqrt{n\alpha})$.

We formally state the problem in Section 5.2, introduce necessary notation in Section 5.3, and present our main result in Section 5.4. In Section 5.5, we describe the main ideas behind our algorithm and provide a comprehensive overview of our technical contributions. We present our algorithm and prove its performance guarantee in Section 5.6. We provide a detailed discussion of related work in Appendix 5.8.

5.2 Problem formulation

We have m sources. Of these m sources at least α-fraction of the sources are genuine and provide $\geq n$ i.i.d. samples from a common distribution. The remaining sources may provide arbitrary data. Since, we can use only the first n samples from each source and ignore the rest, hence, w.l.o.g. we assume that each source provides exactly n samples. We will refer to the collection of all samples from a single source as a *batch*.

To formalize the setting, let B be a collection of m batches. Each batch $b \in B$ in this collection, has n samples $\{(x_i^b, y_i^b)\}_{i=1}^n$, where $x_i^b \in \mathbb{R}^d$ and $y_i^b \in \mathbb{R}$.

Among these batches B, there is a sub-collection G of *good batches* such that for each $b \in G$ and $i \in [n]$ samples (x_i^b, y_i^b) are generated independently from a common distribution $\mathcal{D}$ and the size of this sub-collection is $|G| \geq \alpha|B|$. The remaining batches $B \setminus G$ are *adversarial batches* and have arbitrary samples that may be selected by an adversary depending on good batches.

Next, we describe the assumption of distribution $\mathcal{D}$. We require the same set of general assumptions on the distribution, as in the recent work [38], which focuses on the case when $n = 1$ and $1 - \alpha$ is small, that is when all but a small fraction of data is genuine.

Distribution Assumptions.

For an unknown d-dimensional vector w^*, the *sample noises* n_i^b, the *covariates* x_i^b and the outputs y_i^b are random variables that are related as $y_i^b = x_i^b \cdot w^* + n_i^b$. Let $\Sigma = \mathbb{E}_{\mathcal{D}}[x_i^b (x_i^b)^\intercal]$. For scaling purposes, we assume $\|\Sigma\| = 1$. We have the following general assumptions.

1. x_i^b is $L4$-$L2$ hypercontractive, that is for some $C \geq 1$ and all vectors u, $\mathbb{E}_{\mathcal{D}}[(x_i^b \cdot u)^4] \leq C \mathbb{E}_{\mathcal{D}}[(x_i^b \cdot u)^2]^2$.
2. For some constant $C_1 > 0$, $\|x_i^b\| \leq C_1 \sqrt{d}$ a.s.
3. The condition number of Σ is at most C_3, that is for each unit vector u, we have $u^\intercal \Sigma u \geq \frac{\|\Sigma\|}{C_3} = \frac{1}{C_3}$.
4. Sample noise$n_i^b$is independent of x_i^b, has zero mean $\mathbb{E}_{\mathcal{D}}[n_i^b]=0$, and bounded covariance $\mathbb{E}_{\mathcal{D}}[(n_i^b)^2] \leq \sigma^2$.
5. The distribution of noise n_i^b is symmetric around 0.

We note that the assumptions 1,3, and 4 are standard in heavy-tailed linear regression [38, 100]. Assumptions 2 and 5, on the other hand, are introduced solely for the ease of presentation and we discuss in Appendix 5.14 that these two assumptions can be eliminated without any impact on our results.

5.3 Notation

We use h^b to denote a function over batches. For a function h^b, we use $\mathbb{E}_{\mathcal{D}}[h^b]$ and $\mathrm{Cov}_{\mathcal{D}}(h^b)$ to denote the expected value and covariance of h^b for a random batch b of n independent samples from $\mathcal{D}$.[1]

Next, we define the expectation and covariance w.r.t. the collection of batches B. When batches are chosen uniformly from a sub-collection $B' \subseteq B$, the expected value and co-variance

[1] With slight abuse of notation, instead of $h(b)$, we use h^b to denote function over batches. Note that h^b may be a function of some or all the samples in the batch b.

of a function h^b are denoted as $\mathbb{E}_{B'}[h^b] = \sum_{b\in B'} \frac{1}{|B'|} h^b$ and $\text{Cov}_{B'}(h^b) = \sum_{b\in B'} \frac{1}{|B'|}(h^b - \mathbb{E}_{B'}[h^b])(h^b - \mathbb{E}_{B'}[h^b])^\intercal$, respectively.

To allow for more general samplings, the definition is extended to use a weight vector. A *weight vector*, denoted by β, is a collection of weights, β^b, for each batch, $b \in B$, such that β^b is between 0 and 1. The *total weight* of the vector is represented by $\beta^B = \sum_{b\in B} \beta^b$. It can be helpful to think of β as a soft cluster of batches, with its components denoting the membership weight of batches in the cluster.

When defining expectation or covariance of a function w.r.t. a weight vector β, the probability of sampling a batch, b, is $\frac{\beta^b}{\beta^B}$. The expectation of a function, h^b, over batches, when using a weight vector β, is represented by $\mathbb{E}_\beta[h^b] := \sum_{b\in B} \frac{\beta^b}{\beta^B} h^b$, and the covariance is represented by $\text{Cov}_\beta(h^b) := \sum_{b\in B} \frac{\beta^b}{\beta^B}(h^b - \mathbb{E}_\beta[h^b])(h^b - \mathbb{E}_\beta[h^b])^\intercal$.

For weight vector β, the *weight of all batches of a subset* B' is denoted as $\beta^{B'} := \sum_{b\in B'} \beta^b$.

We use $f(x) = \tilde{\mathcal{O}}(g(x))$ as a shorthand for $f(x) = \mathcal{O}(g(x)\log^k x)$, where k is some integer, and $f(x) = \mathcal{O}_y(g(x))$ implies that if y is bounded then $f(x) = \mathcal{O}(g(x))$. Throughout the paper, we use the notation c_i, with $i \geq 1$, to represent universal constants.

5.4 Main Results

Recently there has been a significant interest in the problem of list decodable linear regression. The prior works considered only the non-batch setting. The sample and time complexity of algorithm in [87, 126] are $d^{\mathcal{O}(1/\alpha^4)}$ and $d^{\mathcal{O}(1/\alpha^8)}$, respectively. [126] achieves an error $\mathcal{O}(\sigma/\alpha^{3/2})$ with a list of size $(1/\alpha)^{\mathcal{O}(\log(1/\alpha))}$, and [87] obtains an error guarantee $\mathcal{O}(\sigma/\alpha)$ with a list of size $\mathcal{O}(1/\alpha)$.

[53] improved the sample complexity. For Gaussian noise and covariates distributed according to standard Gaussian, they gave an information-theoretic algorithm that uses $\mathcal{O}(d/\alpha^3)$ samples and estimates w to an accuracy $\mathcal{O}(\sigma\sqrt{\log(1/\alpha)}/\alpha)$ using a list of size $\mathcal{O}(1/\alpha)$. They also showed that no algorithm, even with infinite samples, can achieve an error

$\ll \sigma/\alpha\sqrt{\log(1/\alpha)}$ with a Poly$(1/\alpha)$ size list.

As these works considered the non-batch setting, they do not obtain a polynomial time algorithm for this problem, which may in fact be impossible [53].

Our main result shows that using batches one can achieve a polynomial time algorithm for this setting, moreover, the algorithm requires only $\tilde{\mathcal{O}}_{n,\alpha}(d)$ genuine samples.

Theorem 82. *For any* $0 < \alpha < 1$, $n \geq \Theta(\frac{{C_3}^2 C^2 \log^2(2/\alpha)}{\alpha})$ *and* $|G| = \Omega_C(dn^2\log(d))$, *Algorithm 7 runs in time poly*$(|G|, \alpha, d, n)$ *and returns a list* M *of size at most* $4/\alpha^2$ *such that with probability* $\geq 1 - 4/d^2$,

$$\min_{w\in L} \|w - w^*\| \leq \mathcal{O}\Big(\frac{C_3 C \log(2/\alpha)}{\sqrt{n\alpha}}\sigma\Big).$$

Interestingly, for $n = \tilde{\Omega}(1/\alpha)$, the estimation error of our polynomial algorithm has a better dependence on α than the best possible $\sigma/\alpha\sqrt{\log(1/\alpha)}$ [53] by any algorithm (even with infinite resources) in the non-batch setting (i.e. $n = 1$).

We restate the above result as the following corollary, which for a given ϵ, d and α characterizes the number of good batches $|G|$ and n required by Algorithm 7 to achieve an estimation error $\mathcal{O}(\epsilon\sigma)$.

Corollary 83. *For any* $0 < \alpha < 1$, $0 \leq \epsilon \leq 1$, $n_{\min} = \Theta_{C_3,C}(\frac{\log^2(2/\alpha)}{\alpha\epsilon^2})$, $n \geq n_{\min}$, *and* $|G| = \Omega_C(dn_{\min}^2\log(d))$, *Algorithm 7 runs in time poly*(α, d, ϵ) *and returns a list* M *of size at most* $4/\alpha^2$ *such that with probability* $\geq 1 - 4/d^2$,

$$\min_{w\in L} \|w - w^*\| \leq \mathcal{O}(\epsilon\sigma).$$

For $\epsilon = \Theta(1)$ in the above corollary, we get $n = \tilde{\Omega}(\frac{1}{\alpha})$ and $|G| = \tilde{\Omega}_C(d\log(d)/\alpha^2)$.

Remark 1. As discussed earlier, for the case where a majority of data is genuine, i.e. $\alpha > 1/2$, polynomial time algorithms have been developed in prior works [123, 50, 38] to estimate the regression parameter even in a non-batch setting. Since the majority of data is genuine, these algorithms can return a single estimate of the regression parameter instead of a list. In particular, the algorithm in [38] requires $\mathcal{O}(d/(1-\alpha)^2)$ genuine samples, and estimates the regression parameter w^* to an ℓ_2 distance of $\mathcal{O}(C_3\sqrt{(1-\alpha)}\sigma)$ for any $1-\alpha = \mathcal{O}(\frac{1}{C_3{}^2})$, where C_3 is the condition number of the covariance matrix Σ of the covariates. A lower bound of $\Omega(\sqrt{(1-\alpha)}\sigma)$ is also known for the non-batch setting. We note that the algorithm in [38] for the case $\alpha > 1/2$, can easily be extended to the batch setting, where by using batch gradients instead of sample gradients in their algorithm, the regression parameter w^* can be estimated to a much smaller ℓ_2 distance of $\mathcal{O}(C_3\sqrt{(1-\alpha)}\sigma/\sqrt{n})$.

5.5 Technical Overview

This section presents the main ideas behind our algorithm.

For a given batch b from B, the square loss of its ith sample at point w in the parameter space is represented by $f_i^b(w) := (w \cdot x_i^b - y_i^b)^2/2$.

If all batches in B had samples generated from $\mathcal{D}$ then the minimizer of the average loss across all batches, represented by $\mathbb{E}_B[f_i^b(w)]$, would converge to the optimal solution w^*. However, the presence of even a single outlier sample can cause this method to fail. In our setting, a majority of batches may contain potentially outlier samples.

The gradient of the loss function $f_i^b(w)$ is $\nabla f_i^b(w) = (w \cdot x_i^b - y_i^b) \cdot x_i^b$. For good batches, which has i.i.d. samples from distribution $\mathcal{D}$, the expected value of this gradient is $\mathbb{E}_{\mathcal{D}}[\nabla f_i^b(w)] = \Sigma(w - w^*)$.

When $|G|$ is sufficiently large, then

$$\begin{aligned}\|\mathbb{E}_G[\nabla f_i^b(w)]\| &= \left\| \tfrac{1}{|G|n} \textstyle\sum_{b\in G} \sum_{i\in[n]} \nabla f_i^b(w) \right\| \\ &\approx \|\mathbb{E}_{\mathcal{D}}[\nabla f_i^b(w)]\| = \|\Sigma\,(w - w^*)\|. \end{aligned} \tag{5.1}$$

Suppose $\tilde{w}$ is a stationary point of all samples, i.e. $\mathbb{E}_B[\nabla f_i^b(\tilde{w})] = 0$. If $\tilde{w}$ is far from w^*, then the above equation implies that the mean of gradients good samples will be large. Then norm of the co-variance of the sample gradients at $\tilde{w}$ will be at least

$$\begin{aligned}\|\text{Cov}_B[\nabla f_i^b(\tilde{w})]\| &\geq \tfrac{|G|}{|B|}\|\mathbb{E}_G[\nabla f_i^b(\tilde{w})] - \mathbb{E}_B[\nabla f_i^b(\tilde{w})]\|^2 \\ &= \alpha\|\mathbb{E}_G[\nabla f_i^b(\tilde{w})]\|^2 \overset{\text{(a)}}{\approx} \alpha\|\Sigma\left(\tilde{w} - w^*\right)\|^2. \end{aligned} \tag{5.2}$$

When the co-variance of good sample points is much smaller than the overall co-variance of all samples it is possible to iteratively divide or filter samples in two (possibly overlapping) clusters such that one of the clusters is “cleaner" than the original [136, 51]. Hence, if we had $\|\text{Cov}_B[\nabla f_i^b(\tilde{w})]\| \gg \|\text{Cov}_G[\nabla f_i^b(\tilde{w})]\|$ then we could have obtained a “cleaner version" of B, that had a higher fraction of good batches.

For batch $b \in G$ the norm of co-variance of gradients (of a single sample) is $\|\text{Cov}_{\mathcal{D}}[\nabla f_i^b(\tilde{w})]\| = \Theta(\sigma^2 + \|\Sigma(\tilde{w} - w^*)\|^2)$ (using L_4-L_2 hypercontractivity). Even if we had $\|\text{Cov}_G[\nabla f_i^b(\tilde{w})]\| \approx \|\text{Cov}_{\mathcal{D}}[\nabla f_i^b(\tilde{w})]\|$, it does not guarantee $|\text{Cov}_B[\nabla f_i^b(\tilde{w})]| \gg |\text{Cov}_G[\nabla f_i^b(\tilde{w})]|$, as $\alpha\|\Sigma\left(\tilde{w} - w^*\right)\|^2 \ll \sigma^2 + \|\Sigma(\tilde{w} - w^*)\|^2$, regardless of how large the difference between the stationary point w^* for the distribution $\mathcal{D}$ and the stationary point $\tilde{w}$ for all samples is. We will now see that focusing on batch gradients rather than single sample gradients can alleviate this problem.

5.5.1 How Batches Help

In the preceding approach, we didn’t leverage the batch structure. In fact, the SQ lower bound in [53] suggests that it may be impossible to achieve a polynomial-time algorithm for the non-batch setting.

To take the advantage of the batch structure instead of considering the loss function and its gradient for each sample individually, we consider the loss of a batch and the gradient of the batch loss. The loss function of a batch b is $f^b(w) = \frac{1}{n}\sum_{i=1}^n f_i^b(w)$ i.e the average of the loss function in its samples. From the linearity of differentiation, the gradient of the batch loss function is $\nabla f^b(w) = \frac{1}{n}\sum_{i=1}^n \nabla f_i^b(w)$.

Then from the linearity of expectation, $\|\mathbb{E}_G[\nabla f_i^b(w)]\| = \|\mathbb{E}_G[\nabla f^b(w)]\|$ for any w. However, averaging over n samples reduces the co-variance by a factor n, therefore, $\text{Cov}_{\mathcal{D}}[\nabla f^b(w)] = \text{Cov}_{\mathcal{D}}[\nabla f_i^b(w)]/n$.

For $|G|$ large enough, we will have population covariance $\|\text{Cov}_G[\nabla f^b(\tilde{w})]\| \approx \|\text{Cov}_{\mathcal{D}}[\nabla f^b(\tilde{w})]\|$. Further, as $\|\text{Cov}_{\mathcal{D}}[\nabla f_i^b(\tilde{w})]\| = \mathcal{O}(\sigma^2 + \|\Sigma(\tilde{w} - w^*)\|^2)$, it follows

$$\|\text{Cov}_G[\nabla f^b(\tilde{w})] = \mathcal{O}\Big(\tfrac{\sigma^2 + \|\Sigma(\tilde{w}-w^*)\|^2}{n}\Big).$$

If the batch size $n = \Omega(\log^2(1/\alpha)/\alpha)$ and $\tilde{w}$ is stationary point of average loss of all samples in B, then for a large value of $\|\tilde{w} - w^*\| = \Omega(\sigma \log(1/\alpha)/\sqrt{n\alpha})$, it can be shown that $\alpha\|\Sigma\,(\tilde{w} - w^*)\|^2 \geq \log^2(\frac{1}{\alpha})\mathcal{O}(\frac{\sigma^2 + \|\Sigma(\tilde{w}-w^*)\|^2)}{n})$. Since the expectation of batch and sample gradients are the same over any batch sub-collection, using a similar argument as for Equation (5.2) one can show that $\|\text{Cov}_B[\nabla f^b(\tilde{w})]\| \approx \alpha\|\Sigma\,(\tilde{w} - w^*)\|^2$, Combining this bound with the above bound gives

$$\|\text{Cov}_B[\nabla f^b(\tilde{w})]\| \geq \log^2(1/\alpha)\|\text{Cov}_G[\nabla f^b(\tilde{w})]\|.$$

Therefore, either the distance between the stationary point of this cluster and w^* is $\leq \mathcal{O}(\frac{\sigma \log(1/\alpha)}{\sqrt{n\alpha}})$ or the covariance of gradients for the set of all batches is much larger than that for good batches. If it is the former, then we have a good approximation of w^* and if it is the latter, we can divide B into two (possibly overlapping) clusters, where at least one of the new clusters contains a majority of good batches and has a higher proportion of good batches than the initial cluster. The same argument can be extended from B to any sub-collection of B that retains a major portion of good batches G.

To divide the clusters, we use the MULTIFILTER routine from [51]. Instead of hard clustering, this routine does soft clustering. The soft clustering produces a membership or weight vector β of length $|B|$ with each entry between $[0, 1]$ that denotes the membership weight of the corresponding batch in the cluster.

The above discussion leads to the following algorithm. We begin with the initial cluster of all batches B. We keep applying MULTIFILTER routine iteratively on the clusters (or weight vectors) until, for all the clusters, the covariance of gradients at the stationary points of the respective clusters becomes small. MULTIFILTER routine ensures that at least one of the clusters retains a major portion of good batches and it doesn't have more than $\mathcal{O}(1/\alpha^2)$ clusters at any stage.

As discussed, for a cluster that retains a major portion of good batches G, the covariance of batch gradients is small only if stationary point $\tilde{w}$ of that cluster approximates w^* with an accuracy of $\|\tilde{w} - w^*\| = \mathcal{O}(\frac{\sigma \log(1/\alpha)}{\sqrt{n\alpha}})$. Since the final set of $\mathcal{O}(1/\alpha^2)$ clusters includes at least one such cluster, the stationary point of at least one of the clusters should approximate w^* to the desired accuracy.

However, applying the MULTIFILTER routine for this purpose presents additional challenges, which we address through various technical contributions in the next section.

5.5.2 Clipping to Improve Sample Complexity

We would like to obtain a high probability concentration bound of $\|\mathrm{Cov}_G[\nabla f^b(w)]\| \leq \mathcal{O}(\frac{\|w-w^*\|^2+\sigma^2}{n})$ on the empirical covariance of the batch gradients in the good batches. No such bounds for general n are known in previous literature. And even for $n = 1$, using known concentration bounds would require a large number of good batches or samples. For example, [50] needed d^5 samples in total, and in fact, a minimum requirement of d^2 samples can be shown for such a bound to hold. [38] required $O(d)$ samples (for n fixed to 1) for a related bound, but for each point w they need to ignore certain samples from the calculation of empirical covariance. These samples can be different depending on w. While such guarantees sufficed for their application where a majority of data was genuine, it is unclear if it can be extended to the list-decodable setting.

To address these challenges, we use *clipped loss* instead. For *clipping parameter* $\kappa > 0$

and any batch $b \in B$, the clipped loss of its i^{th} sample at point w is given by

$$f_i^b(w, \kappa) := \begin{cases} \frac{(w \cdot x_i^b - y_i^b)^2}{2} & \text{if } |w \cdot x_i^b - y_i^b| \leq \kappa \\ \kappa |w \cdot x_i^b - y_i^b| - \kappa^2/2 & \text{otherwise.} \end{cases}$$

We specify the choice of clipping parameter κ later. The clipped loss defined above is known as Huber's loss in literature. The gradient of this clipped loss is

$$\nabla f_i^b(w, \kappa) := \frac{(x_i^b \cdot w - y_i^b)}{|x_i^b \cdot w - y_i^b| \vee \kappa} \kappa x_i^b. \tag{5.3}$$

We refer to the gradient of the clipped loss above as the *clipped gradient*.

For a batch b, its *clipped loss* is simply the average of clipped loss over all its samples. *Clipped loss* for a batch b at point w is $f^b(w, \kappa) := \frac{1}{n} \sum_{i \in [n]} f_i^b(w, \kappa)$. By the linearity of gradients, the gradient of the clipped loss, or *clipped gradient*, $\nabla f^b(w, \kappa)$ is the average of clipped loss over all its samples, i.e. $\nabla f^b(w, \kappa) := \sum_{i \in [n]} \frac{1}{n} \nabla f_i^b(w, \kappa)$.

Ideal choice of Clipping parameter.

When $\kappa \to \infty$, the clipped loss is the same as the squared loss, hence clipping will have no effect in reducing the number of samples required. On the other hand, if $\kappa \to 0$, the loss function is overly clipped, which can lead to the expected norm of the clipped gradient being much smaller than that of the unclipped gradients in Equation (5.1). Theorem 89 shows that as long as the clipping parameter is set to $\Omega(\|w - w^*\|) + \Omega_{n\alpha}(\sigma)$, the expected norm of the clipped gradient will be $\Omega(\|w - w^*\|) - \tilde{\mathcal{O}}(\sigma/\sqrt{n\alpha})$. This means that for any point w whose distance from w^* is greater than $\tilde{\Omega}(\sigma/\sqrt{n\alpha})$, the expected norm of the clipped gradient at w is $\|\mathbb{E}_{\mathcal{D}}[\nabla f_i^b(w, \kappa)]\| = \Omega(\|w - w^*\|)$, which is of the same order as that of unclipped gradients in Equation (5.1).

Furthermore, taking advantage of clipping, in Theorem 84 we show that for all points w, and for any clipping parameter $\kappa = \mathcal{O}(\|w - w^*\|) + \mathcal{O}_{n\alpha}(\sigma)$ the covariance of the clipped gradients satisfies $\|\mathrm{Cov}_G[\nabla f^b(w,\kappa)]\| \le \mathcal{O}(\frac{\|w-w^*\|^2+\sigma^2}{n})$ with only $\tilde{O}_{n,\alpha}(d)$ samples. As discussed previously, the same bound on the covariance of the un-clipped gradients would instead require $\Omega(d^2)$ samples.

From the preceding discussion, in order for both the requirements of $\|\mathbb{E}_{\mathcal{D}}[\nabla f_i^b(w,\kappa)]\| = \Omega(\|w-w^*\|)$ and $\|\mathrm{Cov}_G[\nabla f^b(w,\kappa)]\| \le \mathcal{O}(\frac{\|w-w^*\|^2+\sigma^2}{n})$ to be met using only $\tilde{O}_{n,\alpha}(d)$ samples, the clipping parameter must be set to $\kappa = \Theta(\|w-w^*\|) + \Theta_{n\alpha}(\sigma)$. This requires a constant factor approximation of $\|w-w^*\|$ to be obtained.

Additionally, when using the MULTIFILTER on a cluster, a tight approximation of $\|w-w^*\|$ is necessary to obtain a tight upper bound on $\|\mathrm{Cov}_G[\nabla f^b(w,\kappa)]\|$, which is required by MULTIFILTER as an input parameter.

Furthermore, recall that when applying the MULTIFILTER on any cluster, we set w to a stationary point of the clipped loss for that cluster. This stationary point w will depend on the clipping parameter κ, and the appropriate range for κ depends on w, creating a cyclic dependence that we must also overcome when estimating $\|w-w^*\|$.

5.5.3 Estimating Parameters for Multifilter

Recall that our goal is to return a small set of (soft) clusters, such that at least one of them retains a major portion of good batches, and its stationary point closely approximates w^*. When the MULTIFILTER routine is applied to a cluster, it generates sub-clusters. Hence, the sub-clusters that originate from a cluster that has already lost a majority of good batches are not relevant for us. Therefore, we will need accurate parameter estimation only for clusters that have retained a substantial weight of good batches, and will only consider such clusters in the remaining section.

Let $v^b(w) := \frac{1}{n}\sum_{i\in[n]} |w \cdot x_i^b - y_i^b|$ denote the mean absolute loss of a batch at point w.

We developed a subroutine called FINDCLIPPINGPPARAMETER (Algorithm 8) that overcomes the cyclic dependence to find appropriate stationary point w and clipping parameter κ for a given soft cluster β. These values ensure that w is a stationary point for clipped gradients $\nabla f^b(w,\kappa)$ for batches in the cluster and κ falls in a range determined by the expected absolute loss of batches in cluster β at the stationary point w, specifically, $\kappa = \Theta(\mathbb{E}_\beta\big[v^b(w)\big]) + \Theta_{n\alpha}(\sigma)$.

For $|G| = \tilde{\Omega}(d)$, we prove that w.h.p.

$$\begin{aligned}\mathrm{Var}_G\big(v^b(w)\big) &\leq \mathbb{E}_G\big[(v^b(w) - \mathbb{E}_{\mathcal{D}}\big[v^b(w)\big]\big)^2\big] \\ &= \mathcal{O}\Big(\tfrac{\sigma^2 + \mathbb{E}_{\mathcal{D}}[v^b(w)]^2}{n}\Big). \end{aligned} \tag{5.4}$$

From the above bound, it follows that for most of the good batches, $v^b(w)$ is very close to $\mathbb{E}_{\mathcal{D}}[v^b(w)]$.

Further, it can be shown that $\mathbb{E}_{\mathcal{D}}[v^b(w)] = \Theta(\|w - w^*\|) \pm \mathcal{O}(\sigma)$, where $\mathbb{E}_{\mathcal{D}}[v^b(w)]$ is expectation of $v^b(w)$ for a batch sampled from $\mathcal{D}$.

We derive a novel way that given a weight vector β estimates upper bound θ_1 on variance $\mathrm{Var}_G\big(v^b(w)\big)$. Further, the upper bound is tight enough to ensure that if $\mathrm{Var}_\beta(v^b(w)) = \tilde{\mathcal{O}}(\theta_1)$ then for most batches b in β, $v^b(w)$ will be close to its expectation $\mathbb{E}_\beta[v^b(w)]$ over β. As the soft cluster contains a significant proportion of good batches, and since $v^b(w)$ for most of these good batches is close to $\mathbb{E}_{\mathcal{D}}[v^b(w)]$, then $\mathbb{E}_\beta[v^b(w)]$ would also be close to $\mathbb{E}_{\mathcal{D}}[v^b(w)]$. Furthermore, since $\mathbb{E}_{\mathcal{D}}[v^b(w)] = \Theta(\|w - w^*\|) \pm \mathcal{O}(\sigma)$, it follows that $\mathbb{E}_\beta[v^b(w)] = \Theta(\|w - w^*\|) \pm \mathcal{O}(\sigma)$.

Therefore, if for a certain weight vector β the variance $\mathrm{Var}_\beta(v^b(w))$, is close to our estimated variance of good batches, θ_1, then we can ensure that $\kappa = \Theta(\|w - w^*\|) + \Theta_{n\alpha}(\sigma)$ and use $\mathbb{E}_\beta[v^b(w)]$ as an estimate for $\|w - w^*\|$.

However, if the variance $\mathrm{Var}_\beta(v^b(w))$ for a β, is significantly greater than our estimated variance of good batches, θ_1, then we will not use the MULTIFILTER routine for gradients on that cluster. Instead, we will apply MULTIFILTER routine on this cluster w.r.t. average absolute loss $v^b(w)$. As a result, the estimation of $\|w - w^*\|$ and ensuring that κ is within the correct range, which is necessary for using the MULTIFILTER routine for gradients, is no longer relevant.

Hence, in the estimation part, we either apply MULTIFILTER routine on the cluster for average absolute loss to obtain new clusters with one of them being cleaner, or else our estimate of the parameters is in the desired range to apply MULTIFILTER routine w.r.t. the gradients.

5.6 Algorithm and Proof of Theorem 82

In subsection 5.6.2, a triplet (β, κ, w) is defined as nice if it meets certain criteria: β retains a substantial amount of weight among good batches, κ falls within a specific range, w is a stationary point of the clipped loss, and the covariance of gradients of the clipped loss for the cluster β is bounded at w. It is noted that any such triplet's point w is a good approximation of w^*. In Section 5.5.1, we provided intuition for the same without the clipping.

To identify a cluster with bounded covariance of clipped gradients, we require that the covariance of clipped gradients for G is bounded. And to estimate the correct range of κ and an upper bound on the covariance of clipped gradients for G, as described in Section 5.5.3, we require that the variance of mean absolute loss is bounded for the set of good batches. We formalize these requirements in the next subsection in form of two regularity conditions.

To specify the range in which the clipping parameter κ should be set, we define $\kappa_{\max}$ and $\kappa_{\min}$ in Definition 1 in the appendix, which are functions of w, and other distribution parameters. Finally, in the last two subsections, we describe the algorithm and show that it finds a nice triplet.

5.6.1 Regularity Conditions.

The first condition is that for all unit vectors u, all vectors w and for all $\kappa \leq \kappa_{\max}$,

$$\mathbb{E}_G\Big[\big(\nabla f^b(w,\kappa)\cdot u - \mathbb{E}_\mathcal{D}[\nabla f^b(w,\kappa)\cdot u]\big)^2\Big] \leq U_1,$$

where $U_1 := c_4 \frac{\sigma^2 + C\mathbb{E}_\mathcal{D}[|(w-w^*)\cdot x_i^b|^2]}{n}$. The second regularity condition is that for all vectors w,

$$\mathbb{E}_G\Big[\big(v^b(w) - \mathbb{E}_\mathcal{D}[|w\cdot x_i^b - y_i^b|]\big)^2\Big] \leq U_2,$$

where $U_2 := c_2 \frac{\sigma^2 + C\mathbb{E}_\mathcal{D}[|w\cdot x_i^b - y_i^b|]^2}{n}$. We will repeatedly refer to the upper bounds U_1 and U_2 in the regularity conditions throughout this section.

In Section 5.9, we show that even with a minimal number of good batches, $|G| = \tilde{\Omega}_{n,\alpha}(d)$, the two regularity conditions hold w.h.p.

As a simple consequence, the first regularity condition implies that

$$\|\mathrm{Cov}_G(\nabla f^b(w,\kappa))\| \leq U_1, \tag{5.5}$$

and similarly, the second regularity condition implies that

$$\mathrm{Var}_G\big(v^b(w)\big) \leq U_2. \tag{5.6}$$

We note that the expressions for U_1 and U_2 simplify to $\Theta(\sigma^2 + \|w - w^*\|^2/n)$ and the expressions for $\kappa_{\max}$ and $\kappa_{\min}$ simplify to $\Theta(\|w - w^*\| + \sigma)$ if one is not concerned with the dependence on distribution parameters C, C_3, C_p.

5.6.2 Nice Triplet

First, we introduce the notion of *nice* weight vector. A weight vector β is considered *nice* if the total weight assigned to all good batches by it is at least $\beta^G \geq 3|G|/4$.

We term a combination of a weight vector β, a clipping parameter κ, and an estimate w as a *triplet*. Next, we introduce the concept of a *nice* triplet.

Condition 1. A triplet (β, κ, w) is considered nice if

(a) β is a nice weight vector, i.e. $\beta^G \geq 3|G|/4$.

(b) Clipping parameter is in the range, $\kappa_{\min} \leq \kappa \leq \kappa_{\max}$.

(c) w is an approximate stationary point, namely mean clipped loss for weight vector β at w is at most $\|\mathbb{E}_\beta[\nabla f^b(w,\kappa)]\| \leq \log(2/\alpha)\sigma/8\sqrt{n\alpha}$.

(d) Covariance of the clipped gradients over β at stationary point w is at most $\|\mathrm{Cov}_\beta(\nabla f^b(w,\kappa))\| \leq c_5 C^2 \log^2(\frac{2}{\alpha})\frac{(\sigma^2 + \mathbb{E}_{\mathcal{D}}[|(w-w^*)\cdot x_i^b|]^2)}{n}$, where c_5 is a positive universal constant.

According to these conditions, a triplet (β, κ, w) is nice if weight vector β is considered nice, clipping parameter κ is within the appropriate range, w is an approximate stationary point for clipped loss for this weight vector and covariance of clipped gradient over weight vector β at this point w is small. As discussed briefly at the beginning of this section, for a triplet satisfying these conditions w is a good approximation of w^*. Theorem 88 formally shows that for any nice triplet (β, w, κ), we have $\|w - w^*\| \leq \mathcal{O}(\frac{C_3 C\sigma \log(2/\alpha)}{\sqrt{n\alpha}})$. Then to prove Theorem 82, it is sufficient to show that the algorithm returns a small list of triplets such that at least one of them is nice.

In the next two subsections, we will describe the algorithm and demonstrate that it returns a small list of triplets, at least one of which is nice.

5.6.3 Description of the Algorithm

MainAlgorithm starts with $L = \beta_{init}$, where the initial weight vector β_{init} assigns an equal weight of 1 to each batch in B. This initial weight vector is nice since $\beta^G_{init} = |G|$. In each iteration of the while loop, the algorithm selects one of the weight vectors β from the list L, until the list L is empty. Then, it uses the subroutine FindClippingPparameter on this weight vector β, which returns the values of clipping parameter κ and approximate stationary point of clipped loss as w.

Next, the algorithm uses the Multifilter subroutine on β. Given a weight vector and a function over batches, as well as an estimate of the variance of the function for good batches, this subroutine divides the cluster to produce new clusters, such that each of them is shorter than the original.

To apply this subroutine, the algorithm first calculates parameters θ_1 and θ_2, which are estimates of the upper bounds U_2 and U_1 in the two regularity conditions for the point w.

If the variance of the mean absolute loss at w for batches in this weight vector β is much larger than the estimate θ_1, namely $\mathrm{Var}_\beta(v^b) \geq c_3 \log^2(2/\alpha)\theta_1$, the algorithm applies the Multifilter subroutine for the function $v^b(w)$. This is referred to as a Type-1 use of this subroutine.

If instead, the variance of v^b in the weight vector is small, the algorithm defines a new

Algorithm 7. MainAlgorithm

```
Input: Data $\{\{(x_i^b, y_i^b)\}_{i\in[n]}\}_{b\in B}$, $\alpha$, $C$, $\sigma$.
For each $b \in B$, $\beta_{init}^b \leftarrow 1$ and $\beta_{init} \leftarrow \{\beta_{init}^b\}_{b\in B}$.
List $L \leftarrow \{\beta_{init}\}$ and $M \leftarrow \emptyset$.
while $L \neq \emptyset$ do
    Pick any element $\beta$ in $L$ and remove it from $L$.
    $a_1 = \frac{256C\sqrt{2}}{3}$ and $a_2 = \frac{a_1}{4} + 64$.
    $\kappa, w \leftarrow$ FindClippingPparameter$(B, \beta, a_1, a_2\ \{\{(x_i^b, y_i^b)\}_{i\in[n]}\}_{b\in B})$
    Find top approximate unit eigenvector $u$ of $\mathrm{Cov}_\beta(\nabla f^b(w,\kappa))$.
    For each batch $b \in B$, let $v^b = \frac{1}{n}\sum_{i\in[n]} |w \cdot x_i^b - y_i^b|$ and $\tilde{v}^b = \nabla f^b(w,\kappa) \cdot u$.
    $\theta_0 \leftarrow \inf\{v : \beta(\{b : v^b \geq v\}) \leq \alpha|B|/4$ and
        $\theta_1 \leftarrow \frac{c_2}{n}\Big(\sigma^2 + \Big(\frac{8\sqrt{C}\theta_0}{7} + \frac{\sigma}{7}\Big)^2\Big),$   (5.7)
        $\theta_2 \leftarrow \frac{c_4}{n}\Big(\sigma^2 + 16C^2\big(\mathbb{E}_\beta[v^b] + \sigma\big)^2\Big).$   (5.8)
    if $\mathrm{Var}_{B,\beta}(v^b) > c_3 \log^2(2/\alpha)\theta_1$ then
        NewWeights $\leftarrow$Multifilter$(B, \alpha, \beta, \{v^b\}, \theta_1)$. {Type-1 use}
        Append each weight vector $\widetilde{\beta} \in$ NewWeights that has total weight $\widetilde{\beta}^B \geq \alpha|B|/2$ to
        list $L$.
    else if $\mathrm{Var}_{B,\beta}(\tilde{v}^b) > c_3 \log^2(2/\alpha)\theta_2$ then
        NewWeights $\leftarrow$Multifilter$(B, \alpha, \beta, \{\tilde{v}^b\}, \theta_2)$. {Type-2 use}
        Append each weight vector $\widetilde{\beta} \in$ NewWeights that has total weight $\widetilde{\beta}^B \geq \alpha|B|/2$ to
        list $L$.
    else
        Append $(\beta, \kappa, w)$ to $M$.
    end if
end while
Return $M$
```

function on batches, $\tilde{v}^b := \nabla f^b(w,\kappa) \cdot u$, where u is a top approximate unit eigenvector of $\mathrm{Cov}_\beta(\nabla f^b(w,\kappa))$ such that $u^\intercal \mathrm{Cov}_\beta(\nabla f^b(w,\kappa))u \geq 0.5\|\mathrm{Cov}_\beta(\nabla f^b(w,\kappa))\|$. This function $\tilde{v}^b$ is a projection of clipped batch gradients along the direction in which covariance is nearly the highest. From (5.5), it follows that variance of this new function $\tilde{v}^b$ in good batch collection G will be bounded by U_1. If the variance of $\tilde{v}^b$ over the weight vector β is much larger than estimate θ_2 of U_1, namely $\mathrm{Var}_\beta\big(\tilde{v}^b\big) \geq c_3 \log^2(2/\alpha)\theta_2$, then the algorithm applies Multifilter subroutine for function $\tilde{v}^b(w)$. This is referred to as a Type-2 use of this subroutine.

When Multifilter is applied to a weight vector, it returns a list NewWeights of weight vectors as a result. The MAINALGORITHM appends weight vectors in NEWWEIGHTS that have total weights more than $\alpha|B|/2$ to list L and the iteration terminates. The weight vectors that have total weights less than $\alpha|B|/2$ are ignored as they can't be nice weight vectors and can not result in any nice weight vector in future iterations.

If the variances of both v^b and $\tilde{v}^b$ are small, then the iteration ends by appending (β, κ, w) to M. Next, we argue that M ends up with at least one nice triplet.

5.6.4 Finding Nice Triplet

We first show that Type-1 application of MULTIFILTER on a nice weight β only occurs when,

$$\mathrm{Var}_\beta\big(v^b\big) \geq c_3 \log^2(2/\alpha)\mathrm{Var}_G\big(v^b\big). \tag{5.9}$$

Recall that Type-1 application of MULTIFILTER on β takes place when $\mathrm{Var}_\beta\big(v^b\big) \geq c_3 \log^2(2/\alpha)\theta_1$. From Equation (5.6), we have $\mathrm{Var}_G\big(v^b\big) \leq U_2$ and Theorem 94 shows that for a nice weight vector β the parameter θ_1 upper bounds U_2. Thus, Type-1 use of MULTIFILTER on a nice weight β only takes place when Equation (5.9) holds.

The subroutine FINDCLIPPINGPPARAMETER returns κ and w for a given weight vector β. Theorem 93 in the Appendix 5.11 shows that these parameters w and κ satisfy:

1. w is an approximate stationary point for $\{f^b(\cdot, \kappa)\}$ w.r.t. weight vector β.
2. $\big(\frac{a_1}{2}\mathbb{E}_\beta[v^b(w)] \vee a_2\sigma\big) \leq \kappa \leq \big(4a_1^2\mathbb{E}_\beta[v^b(w)] \vee a_2\sigma\big)$, where a_1 and a_2 are input parameters of FINDCLIPPINGPPARAMETER.

The first guarantee implies that if a triplet (β, κ, w) ends in set M, then it must satisfy condition (c) for a nice triplet.

Theorem 97 shows that if Type-1 filtering did not occur for a nice weight vector, then for this weight vector the range of κ specified in the second guarantee of subroutine FINDCLIPPINGPPARAMETER is a subset of the desired range $(\kappa_{\min}, \kappa_{\max})$. Specifically, if for a nice weight vector β, $\mathrm{Var}_\beta\big(v^b\big) \leq c_3 \log^2(2/\alpha)\theta_1$, then $\kappa \in (\kappa_{\min}, \kappa_{\max})$ and

$$U_1 \leq \theta_2 \leq \tfrac{c_5}{2c_3} \tfrac{C^2(\sigma^2 + \mathbb{E}_{\mathcal{D}}[|(w-w^*)\cdot x_i^b|]^2)}{n}. \tag{5.10}$$

Recall that a triplet (β, κ, w) ends up in M only when $\mathrm{Var}_\beta\big(v^b\big) \leq c_3 \log^2(2/\alpha)\theta_1$ and $\mathrm{Var}_\beta\big(\tilde{v}^b\big) \leq c_3 \log^2(2/\alpha)\theta_2$ are both satisfied.

From the above discussion, it follows that if a triplet (β, κ, w) is in M such that β is nice then $\kappa \in (\kappa_{\min}, \kappa_{\max})$ and it satisfies,

$$\mathrm{Var}_\beta\big(\tilde{v}^b\big) \leq \tfrac{c_5}{2} \log^2(2/\alpha) \tfrac{C^2(\sigma^2 + \mathbb{E}_{\mathcal{D}}[|(w-w^*)\cdot x_i^b|]^2)}{n}.$$

From the definition of $\tilde{v}^b$, it follows that $\|\mathrm{Cov}_\beta(\nabla f^b(w,\kappa))\| \leq 2\mathrm{Var}_\beta\big(\tilde{v}^b\big)$. Therefore, for any triplet (β, κ, w) in M such that β is a nice weight vector, conditions (b) and (d) are also satisfied. This means that any such triplet is a nice triplet. Finally, it remains to be shown that M contains at least one triplet with a nice weight vector, which we do next.

Recall that Type-2 application of Multifilter on a weight β only takes place when, $\mathrm{Var}_\beta\big(\tilde{v}^b\big) \geq c_3 \log^2(2/\alpha)\theta_2$. Since for a nice β, from Equation (5.10), $U_1 \leq \theta_2$, from Equation (5.5), $\|\mathrm{Cov}_G(\nabla f^b(w,\kappa))\| \leq U_1$, and from the definition of $\tilde{v}^b$, $\mathrm{Var}_G\big(\tilde{v}^b\big) \leq \|\mathrm{Cov}_G(\nabla f^b(w,\kappa))\|$. Therefore, $\theta_2 \geq \mathrm{Var}_G\big(\tilde{v}^b\big)$, and hence Type-2 application on a nice weight β only takes place when,

$$\mathrm{Var}_\beta\big(\tilde{v}^b\big) \geq c_3 \log^2(2/\alpha)\mathrm{Var}_G\big(\tilde{v}^b\big). \tag{5.11}$$

Theorem 100 in Appendix 5.13 states that if Equation (5.9) holds for all Type-1 uses and Equation (5.11) holds for all Type-2 uses when using subroutine Multifilter on nice weight vectors, then at least one of the triplets in the final list M will include a nice weight vector. Since we have already shown that these two equations hold, it follows that M will contain a nice triplet. The theorem also shows that the size of M is at most $4/\alpha^2$ and the total number of iterations of the while loop is at most $\mathcal{O}(|B|/\alpha^2)$, implying a small list size and a polynomial runtime for the algorithm

5.7 Conclusion

In summary, this paper addresses the problem of linear regression in the setting when data is presented in batches and only a small fraction of the batches contain genuine data. The paper presents a polynomial time algorithm to identify a small list containing a good approximation of the true regression parameter when genuine batches have at least $\tilde{\Omega}(1/\alpha)$ samples each. By utilizing the batch structure, the paper introduces the first polynomial-time algorithm for list decodable linear regression. Additionally, the algorithm requires a number of genuine samples that increase nearly linearly with the dimension of the covariates.

SQ lower bounds in [53] for the non-batch setting suggests that a polynomial time algorithm is impossible with batch size 1, and the paper demonstrates that a batch size of $\geq \tilde{\Omega}(1/\alpha)$ is sufficient to obtain a polynomial time algorithm. This poses the question of what the smallest batch size required is to obtain a polynomial time algorithm, which is a promising direction for future work.

Appendix

5.8 Related Work

Robust Estimation and Regression. Designing estimators which are robust under the presence of outliers has been broadly studied since 1960s [142, 6, 73]. However, most prior works either requires exponential time or have a dimension dependency on the error rate, even for basic problems such as mean estimation. Recently, [46] proposed a filter-based algorithm for mean estimation which achieves polynomial time and has no dependency on the dimensionality in the estimation error. There has been a flurry of research on robust estimation problems, including mean estimation [99, 48, 59, 68, 69, 49], covariance estimation [36, 101], linear regression and sparse regression [18, 17, 11, 64, 123, 90, 50, 104, 88, 40, 112, 57, 87, 120, 38], principal component analysis [94, 81], mixture models [45, 83, 97, 71]. The results on robust linear regression are particularly related to the setting of this work, though those papers considered non-batch settings and the fraction of good examples $\alpha > 1/2$. [123, 50, 57, 120, 38, 80] considered the setting when both both covariate x_i and label y_i are corrupted. When there are only label corruptions, [18, 40, 93] achieve nearly optimal rates with $O(d)$ samples. Under the oblivious label corruption model, i.e., the adversary only corrupts a fraction of labels in complete ignorance of the data, [17, 137] provide a consistent estimator whose approximate error goes to zero as the sample size goes to infinity.

Robust Learning from Batches. [125] introduced the problem of learning discrete distribution from untrusted batches and derived an exponential time algorithm. Subsequent works [32] improved the run-time to quasi-polynomial and [77] obtained polynomial time with an optimal sample complexity. [78, 31] extended these results to one-dimensional structured distributions. [76, 96] studied the problem of classification from untrusted batches. [2] studies a closely related problem of learning parameter of Erdős-Rényi random graph when a fraction of nodes are corrupt. All these works focus on different problems than ours and only consider the case when a majority of the data is genuine.

List Decodable Mean Estimation and Regression. List decodable framework was first introduced in [29] to obtain learning guarantees when a majority of data is corrupt. They derived the first polynomial algorithm for list decodable mean estimation under co-variance bound. Subsequent works [51, 39, 52] obtained a better run time. [56, 97] improved the error guarantees, however, under stronger distributional assumptions and has higher sample and time complexities.

[87] studies the problem of list-decodable linear regression with batch-size $n = 1$ and derive an algorithm with sample complexity $(d/\alpha)^{O(1/\alpha^4)}$ and runtime $(d/\alpha)^{O(1/\alpha^8)}$. [126] show a sample complexity of $(d/\alpha)^{O(1/\alpha^4)}$ with runtime $(d/\alpha)^{O(1/\alpha^8)}(1/\alpha)^{\log(1/\alpha)}$. Polynomial time might indeed be impossible for the single sample setting owing to the statistical query lower bounds in [53].

Mixed Linear Regression. When each batch has only one sample, (i.e. $n = 1$) and contains samples of one of the k regression components the problem becomes the classical mixed linear regression which has been widely studied [55, 32, 102, 134, 154, 153, 26]. It is worth noting that no algorithm is known to achieve polynomial sample complexity in this setting. The problem is only studied very recently in the batched setting with $n > 1$ by [95, 94], where all the samples in the batch are from the same component. [95] proposed a polynomial time algorithm which requires $O(d)$ batches each with size $O(\sqrt{k})$. [94] leveraged sum-of-squares hierarchy to introduce a class of algorithms which is able to trade off the batch size n and the sample complexity. Both of these works assume that the distributions of covariates for all components is identical and Gaussian. Since the above problem is a special case of the list-decodable linear regression, our algorithm is able to recover the k regression components with batch size $n = O(k)$ and $O(d)$ number of batches. Our algorithms allow more general distributions for the covariates than allowed by the Gaussian assumption in the previous works. Further, our algorithms allow the distributions of covariates for the different components to differ. It is worth noting that list-decodable linear regression is a strictly harder problem than mixed linear regression as shown in [53] and thus our result is incomparable to the ones in the mixed linear regression setting. Leaning mixture of linear dynamical systems has been studied in [33].

5.9 Regularity conditions

In this section, we state regularity conditions for genuine data used in proving the guarantees of our algorithm. Before we proceed we will define upper and lower bounds on the clipping parameter κ that are functions of w and other distribution parameters,

Definition 1. We define the following upper and lower bounds on the clipping parameter κ as a function of w and other distribution parameters:

$$\begin{aligned}\kappa_{\max} =& c_7 C^2 \Big(\sqrt{\mathbb{E}_{\mathcal{D}}[|x_i^b \cdot (w - w^*)|^2]} + \sigma \Big), \\ \kappa_{\min} =& \max \Big\{ 8\sqrt{C\mathbb{E}_{\mathcal{D}}[|x_i^b \cdot (w - w^*)|^2]}, 8\sigma \Big\}.\end{aligned}$$

$\kappa_{\max}$ will be used in this section to define our first regularity condition, while $\kappa_{\min}$ will be used in Section 5.10 for defining a nice triplet.

Regularity Conditions.

1. For all $\kappa \leq \kappa_{\max}$, all unit vectors u and all vectors w

$$\mathbb{E}_G\Big[\big(\nabla f^b(w,\kappa) \cdot u - \mathbb{E}_{\mathcal{D}}[\nabla f^b(w,\kappa) \cdot u]\big)^2\Big] \leq c_4 \frac{\sigma^2 + C\mathbb{E}_{\mathcal{D}}[((w - w^*) \cdot x_i^b)^2]}{n},$$

2. For all vectors w,

$$\mathbb{E}_G\left[\left(\frac{1}{n}\sum_{i\in[n]} |w \cdot x_i^b - y_i^b| - \mathbb{E}_{\mathcal{D}}[|w \cdot x_i^b - y_i^b|]\right)^2\right] \leq c_2\left(\frac{\sigma^2 + C\mathbb{E}_{\mathcal{D}}[|w \cdot x_i^b - y_i^b|]^2}{n}\right).$$

The first regularity condition on the set of good batches G, bounds the mean squared deviation of projections of clipped batch gradients from its true population mean. The regularity condition requires clipping parameter κ to be upper bounded, with the upper bound depending on $\|w - w^*\|$ and σ.

As discussed in Section 5.5, when $\kappa \to \infty$, the clipping has no effect, and establishing such regularity condition for unclipped gradients would require $\Omega(d^2)$ samples. By using clipping, and ensuring that clipping parameter κ is in the desired range we are able to achieve $\tilde{O}_{n,\alpha}(d)$ sample complexity.

Theorem 84 characterizes the number of good batches required for regularity condition 1 as a function of the upper bound on κ.

Theorem 84. *There exist a universal constant c_4 such that for $\mu_{\max} \in [1, \frac{d^4 n^2}{C}]$ and $|G| = \Omega(\mu_{\max}^4 n^2 d \log(d))$, with probability $\geq 1 - \frac{4}{d^2}$, for all unit vectors u, all vectors w and for all $\kappa^2 \leq \mu_{\max}(\sigma^2 + C\mathbb{E}_{\mathcal{D}}[((w - w^*) \cdot x_i^b)^2])$,*

$$\mathbb{E}_G\Big[\big(\nabla f^b(w,\kappa) \cdot u - \mathbb{E}_{\mathcal{D}}[\nabla f^b(w,\kappa) \cdot u]\big)^2\Big] \leq c_4 \frac{\sigma^2 + C\mathbb{E}_{\mathcal{D}}[((w - w^*) \cdot x_i^b)^2]}{n}. \qquad (5.12)$$

We prove the above theorem in Section 5.15.

The second regularity condition on the set of good batches G, bounds the mean squared deviation of average absolute error for a batch from its true population mean. Theorem 85 characterizes the number of good batches required for regularity condition 2.

Theorem 85. *For $|G| = \Omega(n^2 d \log(d))$ and universal constant $c_2 > 0$, with probability $\geq 1 - \frac{4}{d^2}$, for all vectors w,*

$$\mathbb{E}_G\left[\left(\frac{1}{n}\sum_{i\in[n]} |w \cdot x_i^b - y_i^b| - \mathbb{E}_{\mathcal{D}}[|w \cdot x_i^b - y_i^b|]\right)^2\right] \leq c_2\bigg(\frac{\sigma^2 + C\mathbb{E}_{\mathcal{D}}[|w \cdot x_i^b - y_i^b|^2]}{n}\bigg).$$

Proof. Proof of the above theorem is similar to the proof of Theorem 84, and for brevity, we skip it. ■

Combining the two theorems shows that the two regularity conditions hold with high probability with $\tilde{O}_{n,\alpha}(d)$ batches.

Corollary 86. *For $|G| \geq \Omega_C(dn^2 \log(d))$, both regularity conditions hold with probability $\geq 1 - \frac{8}{d^2}$.*

We conclude the sections with the following Lemma which lists some simple consequences of regularity conditions, that we use in later sections.

Lemma 87. *If regularity conditions hold then*

1. *For all vectors w and for all $\kappa \leq \kappa_{\max}$,*

$$\|\mathrm{Cov}_G(\nabla f^b(w,\kappa))\| \leq c_4 \frac{\sigma^2 + C\mathbb{E}_{\mathcal{D}}[((w-w^*)\cdot x_i^b)^2]}{n},$$

2. *For all vectors w*

$$Var_G\left(\frac{1}{n}\sum_{i\in[n]} |w\cdot x_i^b - y_i^b|\right) \leq c_2\left(\frac{\sigma^2 + C\mathbb{E}_{\mathcal{D}}[|w\cdot x_i^b - y_i^b|]^2}{n}\right).$$

3. *For all $G' \subseteq G$ of size $\geq |G|/2$,*

$$\|\mathbb{E}_{G'}[\nabla f^b(w,\kappa)] - \mathbb{E}_{\mathcal{D}}[\nabla f^b(w,\kappa)]\| \leq \sqrt{2c_4}\frac{\sigma + \sqrt{C\mathbb{E}_{\mathcal{D}}[((w-w^*)\cdot x_i^b)^2]}}{\sqrt{n}}.$$

Proof. The first item in the lemma follows as

$$\begin{aligned}\|\mathrm{Cov}_G(\nabla f^b(w,\kappa))\| &= \max_{u:\|u\|\leq 1} \mathbb{E}_G\Big[\big(\nabla f^b(w,\kappa)\cdot u - \mathbb{E}_G[\nabla f^b(w,\kappa)\cdot u]\big)^2\Big]\\ &\leq \max_{u:\|u\|\leq 1} \mathbb{E}_G\Big[\big(\nabla f^b(w,\kappa)\cdot u - \mathbb{E}_{\mathcal{D}}[\nabla f^b(w,\kappa)\cdot u]\big)^2\Big]\\ &\leq c_4\frac{\sigma^2 + C\mathbb{E}_{\mathcal{D}}[((w-w^*)\cdot x_i^b)^2]}{n},\end{aligned}$$

where the first inequality follows as the expected squared deviation along the mean is the smallest and the second inequality follows from the first regularity condition.

Similarly, the second item follows from the second regularity condition.

Finally, we prove the last item using the first regularity condition. Let u be any unit vector and $Z^b(u) := \big(\nabla f^b(w,\kappa)\cdot u - \mathbb{E}_{\mathcal{D}}[\nabla f^b(w,\kappa)\cdot u]\big)^2$. Then

$$\|\mathbb{E}_G[Z^b](u)\| = \left\|\frac{1}{|G|}\sum_{b\in G} Z^b(u)\right\| \geq \left\|\frac{1}{|G|}\sum_{b\in B'} Z^b(u)\right\| = \frac{|G'|}{|G|}\|\mathbb{E}_{G'}[Z^b(u)]\| \geq \frac{1}{2}\|\mathbb{E}_{G'}[Z^b(u)]\|,$$

where the first inequality used the fact that $Z^b(u)$ is a positive and the second inequality used

$|G'| \geq |G|/2$. Then using the bound on $\|\mathbb{E}_G[Z^b(u)]\|$ in in the first regularity condition, we get

$$\|\mathbb{E}_{G'}[Z^b(u)]\| \leq 2c_4 \frac{\sigma^2 + C\mathbb{E}_{\mathcal{D}}[((w-w^*)\cdot x_i^b)^2]}{n}.$$

Using the Cauchy–Schwarz inequality and the above bound,

$$\begin{aligned}
\mathbb{E}_{G'}[\left|\nabla f^b(w,\kappa)\cdot u - \mathbb{E}_{\mathcal{D}}[\nabla f^b(w,\kappa)\cdot u]\right|] &= \mathbb{E}_{G'}[\sqrt{Z^b(u)}] \\
&\leq \sqrt{\mathbb{E}_{G'}[Z^b(u)]} \\
&\leq \sqrt{2c_4 \frac{\sigma^2 + C\mathbb{E}_{\mathcal{D}}[((w-w^*)\cdot x_i^b)^2]}{n}}.
\end{aligned}$$

Since the above bound holds for each unit vector u, we have

$$\begin{aligned}
\mathbb{E}_{G'}[\left|\nabla f^b(w,\kappa) - \mathbb{E}_{\mathcal{D}}[\nabla f^b(w,\kappa)]\right|] &\leq \sqrt{2c_4 \frac{\sigma^2 + C\mathbb{E}_{\mathcal{D}}[((w-w^*)\cdot x_i^b)^2]}{n}} \\
&\leq \sqrt{2c_4} \frac{\sigma + \sqrt{C\mathbb{E}_{\mathcal{D}}[((w-w^*)\cdot x_i^b)^2]}}{\sqrt{n}}.
\end{aligned}$$

■

5.10 Guarantees for nice triplet

For completeness, we first restate the conditions a nice triplet (β, κ, w) satisfy.

A triplet (β, κ, w) is *nice* if

(a) β is a nice weight vector, i.e. $\beta^G \geq 3|G|/4$.

(b) $\kappa_{\min} \leq \kappa \leq \kappa_{\max}$.

(c) w is any approximate stationary point w.r.t. β for clipped loss with clipping parameter κ, namely $\|\mathbb{E}_\beta[\nabla f^b(w,\kappa)]\| \leq \frac{\log(2/\alpha)\sigma}{8\sqrt{n\alpha}}$.

(d) $\|\mathrm{Cov}_\beta(\nabla f^b(w,\kappa))\| \leq \frac{c_5 C^2 \log^2(2/\alpha)(\sigma^2+\mathbb{E}_{\mathcal{D}}[\|(w-w^*)\cdot x_i^b\|]^2)}{n}$.

In this section, we establish the following guarantees for any nice triplets. In doing so we assume regularity conditions hold for G.

Theorem 88. *Suppose (β, κ, w) is a nice triplet,* $n \geq \max\{32c_4CC_3, \frac{256}{\alpha}c_5C^2{C_3}^2\log^2(2/\alpha)\}$ *and regularity conditions holds, then* $\|w - w^*\| \leq \mathcal{O}(\frac{C_3C\sigma\log(2/\alpha)}{\sqrt{n\alpha}})$.

In the remainder of this section, we prove the theorem. First, we provide an overview of the proof and state some auxiliary lemma that we use to prove the theorem.

In this section, we show that for any nice triplet (β, κ, w) if $\|w - w^*\| = \tilde{\Omega}(\sigma/\sqrt{n\alpha})$ then the following lower bound on clipped gradient co-variance, $\|\text{Cov}_\beta(\nabla f^b(w, \kappa))\| \geq \Omega(\alpha\|w - w^*\|^2)$ holds. For $n = \tilde{\Omega}(\frac{1}{\alpha})$ and $\|w - w^*\| = \tilde{\Omega}(\sigma/\sqrt{n\alpha})$ this lower bound contradicts the upper bound in condition (d). Hence, the theorem concludes that $\|w - w^*\| = \tilde{\mathcal{O}}(\sigma/\sqrt{n\alpha})$.

To show the lower bound $\|\text{Cov}_\beta(\nabla f^b(w, \kappa))\| \geq \Omega(\alpha\|w - w^*\|^2)$, we first show $\|\mathbb{E}_\mathcal{D}[\nabla f_i^b(w, \kappa)]\| = \Omega(\|w - w^*\|) - \tilde{\mathcal{O}}(\sigma/\sqrt{n\alpha})$ in Theorem 89. Since $\|\mathbb{E}_\mathcal{D}[\nabla f_i^b(w, \kappa)]\| = \|\mathbb{E}_\mathcal{D}[\nabla f^b(w, \kappa)]\|$, the same bound will hold for the norm of expectation of clipped batch gradients.

When clipping parameter $\kappa \to \infty$ then $\nabla f_i^b(w, \kappa) = \nabla f_i^b(w)$ and for unclipped gradients, a straightforward calculation shows the desired lower bound $\|\mathbb{E}_\mathcal{D}[\nabla f_i^b(w, \kappa)]\| = \Omega(\|w - w^*\|)$. However, if κ is too small then clipping may introduce a large bias in the gradients and such a lower bound may no longer hold.

Yet, the lower bound on κ in condition (b) ensures that κ is much larger than the typical error which is of the order $\|w - w^*\| + \sigma$. And when clipping parameter κ is much larger than the typical error, it can be shown that with high probability clipped and unclipped gradients for a random sample from $\mathcal{D}$ would be the same. The next theorem uses this observation and for the case when κ satisfies the lower bound in condition (b) it shows the desired lower bound on the norm of expectation of clipped gradient.

Theorem 89. *If $\kappa \geq \max\{8\sqrt{C\mathbb{E}_{\mathcal{D}}[|x_i^b \cdot (w-w^*)|^2]}, 8\sigma\}$, then*

$$\left\|\mathbb{E}_{\mathcal{D}}[\nabla f_i^b(w,\kappa)]\right\| \geq \frac{3}{4C_3}\|w-w^*\|.$$

We prove the above theorem in subsection 5.10.1

Since $\mathbb{E}_{\mathcal{D}}[\nabla f_i^b(w,\kappa)] = \mathbb{E}_{\mathcal{D}}[\nabla f^b(w,\kappa)]$, the same bound holds for the clipped batch gradients.

Next, in Lemma 90 we show that for any sufficiently large collection $G' \subseteq G$ of the good batches $\|\mathbb{E}_{G'}[\nabla f^b(w,\kappa)]\| \approx \|\mathbb{E}_{\mathcal{D}}[\nabla f^b(w,\kappa)]\|$.

Lemma 90. *Suppose κ and w are part of a nice triplet, $n \geq 32c_4CC_3$ and regularity conditions holds, then for all $G' \subseteq G$ of size $\geq |G|/2$,*

$$\left\|\mathbb{E}_{G'}[\nabla f^b(w,\kappa)]\right\| \geq \frac{1}{2C_3}\|w-w^*\| - \frac{\sqrt{2c_4}\sigma}{\sqrt{n}}.$$

Proof. From item 3 in Lemma 87,

$$\begin{aligned}\|\mathbb{E}_{G'}[\nabla f^b(w,\kappa)] - \mathbb{E}_{\mathcal{D}}[\nabla f^b(w,\kappa)]\| &\leq \sqrt{2c_4}\cdot\frac{\sigma + \sqrt{C\mathbb{E}_{\mathcal{D}}[((w-w^*)\cdot x_i^b)^2]}}{\sqrt{n}} \\ &\leq \frac{\sqrt{2c_4}\sigma}{\sqrt{n}} + \frac{\sqrt{2c_4C\|w-w^*\|^2\|\Sigma\|}}{\sqrt{n}} \\ &\leq \frac{\sqrt{2c_4}\sigma}{\sqrt{n}} + \|w-w^*\|\cdot\frac{\sqrt{2c_4C}}{\sqrt{n}}.\end{aligned}$$

Using $n \geq 32c_4C{C_3}^2$,

$$\|\mathbb{E}_{G'}[\nabla f^b(w,\kappa)] - \mathbb{E}_{\mathcal{D}}[\nabla f^b(w,\kappa)]\| \leq \frac{1}{4C_3}\|w-w^*\| + \frac{\sqrt{2c_4}\sigma}{\sqrt{n}}.$$

From Theorem 89, and the observation $\mathbb{E}_{\mathcal{D}}[\nabla f_i^b(w,\kappa)] = \mathbb{E}_{\mathcal{D}}[\nabla f^b(w,\kappa)]$, we get

$$\left\|\mathbb{E}_{\mathcal{D}}[\nabla f^b(w,\kappa)]\right\| \geq \frac{3}{4C_3}\|w-w^*\|.$$

The lemma follows by combining the above equation using triangle inequality. ■

Next, the general bound on the co-variance will be useful in proving Theorem 88.

Lemma 91. *For any weight vector β, any set of vectors z^b associated with batches, and any sub-collection of vectors $B' \subseteq \{b \in B : \beta^b \geq 1/2\}$,*

$$Cov_\beta(z^b) \geq \frac{|B'|}{2|B|}\|\mathbb{E}_\beta[z^b] - \mathbb{E}_{B'}[z^b]\|^2.$$

The proof of the lemma appears in Section 5.10.2.

In Theorem 88 we show that since $\beta^G \geq 3/4|G|$, we can find a sub-collection G' of size $|G|/2$ such that for each $b \in G'$, its weight $\beta^b \geq 1/2$. The we use the previous results for $B' = G'$ and $z = \nabla f^b(w, \kappa)$ to get, $\mathrm{Cov}_\beta(\nabla f^b(w, \kappa)) \geq \frac{|G'|}{4|B|}\|\mathbb{E}_\beta[\nabla f^b(w, \kappa)] - \mathbb{E}_{G'}[\nabla f^b(w, \kappa)]\| \geq \frac{|G|}{8|B|}\|\mathbb{E}_\beta[\nabla f^b(w, \kappa)] - \mathbb{E}_{G'}[\nabla f^b(w, \kappa)]\| \geq \frac{\alpha}{8}\|\mathbb{E}_\beta[\nabla f^b(w, \kappa)] - \mathbb{E}_{G'}[\nabla f^b(w, \kappa)]\|^2$.

From condition (c) of nice triplets we have $\mathbb{E}_\beta[\nabla f^b(w, \kappa)] \approx 0$ and from Lemma 90 we have $\mathbb{E}_{G'}[\nabla f^b(w, \kappa)] \gtrsim \|w - w^*\|$. Then from Lemma 91, we get an upper bound $\mathrm{Cov}_\beta(\nabla f^b(w, \kappa)) \gtrsim \alpha \cdot \|w - w^*\|^2$.

As discussed before, combining this lower bound with the upper bound in condition (d), the theorem concludes $\|w - w^*\| = \tilde{\mathcal{O}}(\sigma/\sqrt{n\alpha})$. Next, we formally prove Theorem 88 using the above auxiliary lemmas and theorems.

Proof of Theorem 88. Let $G' := \{b \in G : \beta^b \geq 1/2\}$. Next, we show that $|G'| \geq |G|/2$. To prove it by contradiction assume the contrary that $|G'| < |G|/2$. Then

$$\beta^G = \sum_{b \in G}\beta^b = \sum_{b \in G \setminus G'}\beta^b + \sum_{b \in G'}\beta^b \overset{\text{(a)}}{\leq} \sum_{b \in G \setminus G'}\frac{1}{2} + \sum_{b \in G'} 1 \leq \frac{|G| - |G'|}{2} + |G'| < \frac{3|G|}{4},$$

here (a) follows as the definition of G' implies that for any $b \notin G'$, $\beta^b < 1/2$ and for all batches $\beta^b \leq 1$. Above is a contradiction, as we assumed in the Theorem that $\beta^G \geq 3|G|/4$.

Applying Lemma 91 for $B' = G'$ and $z^b = \nabla f^b(w,\kappa)$ we have

$$\begin{aligned}\|\mathrm{Cov}_\beta(\nabla f^b(w,\kappa))\| &\geq \frac{G'}{2|B|}\big(\|\mathbb{E}_{G'}[\nabla f^b(w,\kappa)]\| - \|\mathbb{E}_\beta[\nabla f^b(w,\kappa)]\|\big)^2\\ &\geq \frac{|G|}{4|B|}\big(\|\mathbb{E}_{G'}[\nabla f^b(w,\kappa)]\| - \|\mathbb{E}_\beta[\nabla f^b(w,\kappa)]\|\big)^2\\ &\geq \frac{\alpha}{4}\big(\|\mathbb{E}_{G'}[\nabla f^b(w,\kappa)]\| - \|\mathbb{E}_\beta[\nabla f^b(w,\kappa)]\|\big)^2. \qquad (5.13)\end{aligned}$$

In the above equation, using the bound in Lemma 90 and bound on $\|\mathbb{E}_\beta[\nabla f^b(w,\kappa)]\|$ in condition (c) for nice triplet we get,

$$\|\mathrm{Cov}_\beta(\nabla f^b(w,\kappa))\| \geq \frac{\alpha}{4}\left(\max\left\{0, \frac{1}{2C_3}\|w-w^*\| - \frac{\sqrt{2c_4}\sigma}{\sqrt{n}} - \frac{\log(2/\alpha)\sigma}{8\sqrt{n\alpha}}\right\}\right)^2.$$

We show that when $\|w-w^*\| \leq \mathcal{O}(\frac{C_3 C\sigma\log(2/\alpha)}{\sqrt{n\alpha}})$, the above upper bound contradicts the following lower bound in condition (d),

$$\begin{aligned}\|\mathrm{Cov}_\beta(\nabla f^b(w,\kappa))\| &\leq \frac{c_5 C^2\log^2(2/\alpha)(\sigma^2 + \mathbb{E}_{\mathcal{D}}[|(w-w^*)\cdot x_i^b|]^2)}{n}\\ &\leq \frac{c_5 C^2\log^2(2/\alpha)(\sigma^2 + \|w-w^*\|^2)}{n}.\end{aligned}$$

To prove the contradiction assume

$$\frac{\|w-w^*\|}{8C_3} > \max\left\{\frac{\log(2/\alpha)\sigma}{8\sqrt{n\alpha}}, \frac{\sqrt{2c_4}\sigma}{\sqrt{n}}, \frac{2\sqrt{c_5}C\sigma\log(2/\alpha)}{\sqrt{n\alpha}}\right\}.$$

Using this lower bound on $\|w-w^*\|$, we lower bound the co-variance. Combining the above lower bound on $\|w - w^*\|$ and equation (5.13), we get,

$$\begin{aligned}
\|\text{Cov}_\beta(\nabla f^b(w,\kappa))\| &\geq \frac{\alpha}{4}\bigg(\frac{1}{4C_3}\|w-w^*\|\bigg)^2 \\
&\geq \frac{\alpha}{4}\bigg(\frac{2\sqrt{c_5}C\sigma\log(2/\alpha)}{\sqrt{n\alpha}} + \frac{1}{8C_3}\|w-w^*\|\bigg)^2 \\
&\geq \frac{\alpha}{4}\bigg(\frac{2\sqrt{c_5}C\sigma\log(2/\alpha)}{\sqrt{n\alpha}}\bigg)^2 + \frac{\alpha}{4}\bigg(\frac{1}{8C_3}\|w-w^*\|\bigg)^2 \\
&\geq \frac{c_5C^2\log^2(2/\alpha)\sigma^2}{n} + \frac{\alpha}{256}\frac{\|w-w^*\|^2}{{C_3}^2} \\
&\geq \frac{c_5C^2\log^2(2/\alpha)\sigma^2}{n} + \frac{c_5C^2\log^2(2/\alpha)\|w-w^*\|^2}{n},
\end{aligned}$$

here the last step used $n \geq \frac{256}{\alpha}c_5C^2{C_3}^2\log^2(2/\alpha)$.

This completes the proof of the contradiction. Hence,

$$\frac{\|w-w^*\|}{8C_3} \leq \max\bigg\{\frac{\log(2/\alpha)\sigma}{8\sqrt{n\alpha}}, \frac{\sqrt{2c_4}\sigma}{\sqrt{n}}, \frac{2\sqrt{c_5}C\sigma\log(2/\alpha)}{\sqrt{n\alpha}}\bigg\}.$$

The above equation implies $\|w-w^*\| \leq \mathcal{O}(\frac{C_3C\sigma\log(2/\alpha)}{\sqrt{n\alpha}})$. ■

5.10.1 Proof of Theorem 89

The following auxiliary lemma will be useful in the proof of the theorem.

Lemma 92. *For any $z_1 \in \mathbb{R}$, $z_2 > 0$ and a symmetric random variable Z,*

$$\bigg|\mathbb{E}\bigg[(z_1+Z) - \frac{(z_1+Z)z_2}{\max(|z_1+Z|,z_2)}\bigg]\bigg| \leq 2|z_1|\Pr(Z > z_2 - |z_1|)$$

Proof. We consider $z_1 \geq 0$ and prove the lemma for this case. The proof for $z_1 < 0$ case then follows from the symmetry of the distribution of Z around 0.

The term inside the expectation can be expressed in terms of indicator random variables as follows:

$$
\begin{aligned}
&(z_1 + Z) - \frac{(z_1 + Z)z_2}{\max(|z_1 + Z|, z_2)} \\
&= (z_1 + Z - z_2) \cdot \mathbb{1}(Z > z_2 - z_1) + (z_1 + Z + z_2) \cdot \mathbb{1}(Z < -z_2 - z_1) \\
&= (z_1 + Z - z_2) \cdot \mathbb{1}(z_2 - z_1 < Z \leq z_2 + z_1) + (z_1 + Z - z_2) \cdot \mathbb{1}(Z > z_2 + z_1) \\
&\quad + (z_1 + Z + z_2) \cdot \mathbb{1}(Z < -z_2 - z_1).
\end{aligned}
$$

Next, taking the expectation on both sides in the above equation,

$$
\begin{aligned}
&\mathbb{E}\left[(z_1 + Z) - \frac{(z_1 + Z)z_2}{\max(|z_1 + Z|, z_2)}\right] \\
&= \mathbb{E}[(z_1 + Z - z_2) \cdot \mathbb{1}(z_2 - z_1 < Z \leq z_2 + z_1)] + \mathbb{E}[(z_1 + Z - z_2) \cdot \mathbb{1}(Z > z_2 + z_1)] \\
&\quad + \mathbb{E}[(z_1 + Z + z_2) \cdot \mathbb{1}(Z < -z_2 - z_1)] \\
&= \mathbb{E}[(z_1 + Z - z_2) \cdot \mathbb{1}(z_2 - z_1 < Z \leq z_1 + z_2)] + 2|z_1| \Pr(Z > z_2 + z_1),
\end{aligned}
$$

where the last step follows because Z is symmetric and $z_1 = |z_1|$ since we assumed $z_1 \geq 0$.

Then,

$$
\begin{aligned}
&\left|\mathbb{E}\left[(z_1 + Z) - \frac{(z_1 + Z)z_2}{\max(|z_1 + Z|, z_2)}\right]\right| \\
&= \mathbb{E}[|z_1 + Z - z_2| \cdot \mathbb{1}(z_2 - z_1 < Z \leq z_2 + z_1)] + 2|z_1| \Pr(Z > z_2 + z_1) \\
&\leq 2|z_1| \Pr(z_2 - z_1 < Z \leq z_2 + z_1) + 2|z_1| \Pr(Z > z_2 + z_1) \\
&= 2|z_1| \Pr(Z > z_2 - z_1).
\end{aligned}
$$

■

Next, using the above lemma we prove Theorem 89.

Proof of Theorem 89. Consider a random sample (x_i^b, y_i^b) from distribution $\mathcal{D}$. Recall that

$n_i^b = y_i^b - w^* \cdot n_i^b$ denote the random noise and is independent of x_i^b.

Consider $(x_i^b \cdot w - y_i^b)x_i^b - \nabla f_i^b(w, \kappa)$. the difference between the unclipped and the clipped gradient for the sample:

$$\begin{aligned}(x_i^b \cdot w - y_i^b)x_i^b - \nabla f_i^b(w, \kappa) &= (x_i^b \cdot w - y_i^b)x_i^b - \frac{(x_i^b \cdot w - y_i^b)}{|x_i^b \cdot w - y_i^b| \vee \kappa}\kappa x_i^b \\ &= \left((x_i^b \cdot (w - w^*) - n_i^b)x_i^b - \frac{(x_i^b \cdot (w - w^*) - n_i^b)}{|x_i^b \cdot (w - w^*) - n_i^b| \vee \kappa}\kappa\right)x_i^b,\end{aligned} \tag{5.14}$$

where in the last equality we used the relation between x_i^b, y_i^b and n_i^b.

Next, by applying Lemma 92, we get:

$$\begin{aligned}&\mathbb{E}_{\mathcal{D}}\left[(x_i^b \cdot (w - w^*) - n_i^b)x_i^b - \frac{(x_i^b \cdot (w - w^*) - n_i^b)}{|x_i^b \cdot (w - w^*) - n_i^b| \vee \kappa}\kappa \middle| x_i^b\right] \\ &\leq 2|x_i^b \cdot (w - w^*)| \cdot \Pr\big(n_i^b > \kappa - |x_i^b \cdot (w - w^*)|\big),\end{aligned}$$

note that in the above expectation x_i^b is fixed and expectation is taken over n_i^b.

Let $Z := \mathbb{1}\big(|x_i^b \cdot (w - w^*)| \geq \kappa/2\big)$. Observe that $\Pr(n_i^b > \kappa - |(w - w^*) \cdot x_i^b|) \leq Z + \Pr(n_i^b > \kappa/2)$. Combining this observation with the above equation, we have:

$$\begin{aligned}&\mathbb{E}_{\mathcal{D}}\left[((w - w^*) \cdot x_i^b - n_i^b) - \frac{((w - w^*) \cdot x_i^b - n_i^b)}{|(w - w^*) \cdot x_i^b - n_i^b| \vee \kappa}\kappa \middle| x_i^b\right] \\ &\leq 2|(w - w^*) \cdot x_i^b| \cdot \big(\Pr(n_i^b > \kappa/2) + Z\big).\end{aligned} \tag{5.15}$$

When $w \neq w^*$ the bound holds trivially. Hence, in the remainder of the proof, we assume $w \neq w^*$. Let $v := \frac{w - w^*}{\|w - w^*\|}$ and $Z_i^b := \mathbb{1}\big((|x_i^b \cdot (w - w^*)| \geq \kappa/2) \cup (|n_i^b| \geq \kappa/2)\big)$. Then, for unit vector $v \in \mathbb{R}^d$, we have

$$\begin{aligned}
&|\mathbb{E}_{\mathcal{D}}[((w \cdot x_i^b - y_i^b)x_i^b - \nabla f_i^b(w,\kappa)) \cdot v]| \\
&= |\mathbb{E}_{\mathcal{D}}[\mathbb{E}_{\mathcal{D}}[((w \cdot x_i^b - y_i^b)x_i^b - \nabla f_i^b(w,\kappa)) \cdot v|x_i^b]]| \\
&\leq \mathbb{E}_{\mathcal{D}}[|\mathbb{E}_{\mathcal{D}}[((w \cdot x_i^b - y_i^b)x_i^b - \nabla f_i^b(w,\kappa)) \cdot v\big|x_i^b]]|] \\
&\overset{\text{(a)}}{\leq} \mathbb{E}_{\mathcal{D}}\big[2|(w-w^*) \cdot x_i^b| \cdot |x_i^b \cdot v|\big(Z + \Pr(n_i^b > \kappa/2)\big)\big] \\
&\overset{\text{(b)}}{\leq} \mathbb{E}_{\mathcal{D}}\left[\frac{2|(w-w^*) \cdot x_i^b|^2}{\|w-w^*\|}\big(Z + \Pr(n_i^b > \kappa/2)\big)\right] \\
&\leq \frac{2}{\|w-w^*\|}\big(\mathbb{E}_{\mathcal{D}}[Z \cdot |(w-w^*) \cdot x_i^b|^2] + \Pr(n_i^b > \kappa/2)\mathbb{E}_{\mathcal{D}}[|(w-w^*) \cdot x_i^b|^2]\big), \qquad (5.16)
\end{aligned}$$

here (a) follows from Equation (5.14) and Equation (5.15), and (b) follows from the definition of vector v. Next, we bound the two terms on the right one by one. We start with the first term:

$$\begin{aligned}
\mathbb{E}_{\mathcal{D}}[Z \cdot |(w-w^*) \cdot x_i^b|^2] &\overset{\text{(a)}}{\leq} \big(\mathbb{E}[(Z)^2] \cdot \mathbb{E}_{\mathcal{D}}[(x_i^b \cdot (w-w^*))^4]\big)^{1/2} \\
&\overset{\text{(b)}}{\leq} \big(\mathbb{E}[Z] \cdot C\mathbb{E}_{\mathcal{D}}[(x_i^b \cdot (w-w^*))^2]^2\big)^{1/2} \\
&\overset{\text{(c)}}{\leq} \big(C\Pr[|x_i^b \cdot (w-w^*)| \geq \kappa/2]\big)^{1/2} \cdot \mathbb{E}_{\mathcal{D}}[(x_i^b \cdot (w-w^*))^2], \quad (5.17)
\end{aligned}$$

where (a) used the Cauchy-Schwarz inequality, (b) used the fact that Z is an indicator random variable, hence, $Z^2 = Z$ and $L4 - L2$ hypercontractivity, and (c) follows from the definition of Z.

Applying the Markov inequality to $(n_i^b)^2$ we get:

$$\Pr[|n_i^b| \geq \kappa/2] \leq \frac{\mathbb{E}_{\mathcal{D}}[(n_i^b)^2]}{(\kappa/2)^2} \leq \frac{\sigma^2}{(\kappa/2)^2}. \qquad (5.18)$$

Similarly, applying the Markov inequality to $|x_i^b \cdot (w-w^*)|^4$ yields:

$$\Pr[|x_i^b \cdot (w-w^*)| \geq \kappa/2] \leq \frac{\mathbb{E}_{\mathcal{D}}[|x_i^b \cdot (w-w^*)|^4]}{(\kappa/2)^4} \leq \frac{C\mathbb{E}_{\mathcal{D}}[|x_i^b \cdot (w-w^*)|^2]^2}{(\kappa/2)^4}, \qquad (5.19)$$

where the last inequality uses $L4 - L2$ hypercontractivity.

Combining Equations (5.16), (5.17), (5.18) and (5.19), we have

$$\begin{aligned}
&|\mathbb{E}_{\mathcal{D}}[((w \cdot x_i^b - y_i^b)x_i^b - \nabla f_i^b(w,\kappa)) \cdot v]| \\
&\leq \frac{8\mathbb{E}_{\mathcal{D}}[(x_i^b \cdot (w - w^*))^2]}{\kappa^2\|w - w^*\|}\big(C\mathbb{E}_{\mathcal{D}}[(x_i^b \cdot (w - w^*))^2] + \sigma^2\big).
\end{aligned}$$

Next,

$$\begin{aligned}
\mathbb{E}_{\mathcal{D}}[(w \cdot x_i^b - y_i^b)x_i^b \cdot v] &\overset{\text{(a)}}{=} \mathbb{E}_{\mathcal{D}}[((w - w^*) \cdot x_i^b - n_i^b)x_i^b \cdot v] \\
&\overset{\text{(b)}}{=} \mathbb{E}_{\mathcal{D}}[((w - w^*) \cdot x_i^b)x_i^b \cdot v] - \mathbb{E}_{\mathcal{D}}[n_i^b] \cdot \mathbb{E}_{\mathcal{D}}[x_i^b \cdot v] \\
&\overset{\text{(c)}}{=} \mathbb{E}_{\mathcal{D}}[((w - w^*) \cdot x_i^b)x_i^b \cdot v] \\
&\overset{\text{(d)}}{=} \frac{\mathbb{E}_{\mathcal{D}}\big[((w - w^*) \cdot x_i^b)^2\big]}{\|w - w^*\|},
\end{aligned}$$

here (a) follows from the relationship between x_i^b, y_i^b and n_i^b, (b) follows from as x_i^b and n_i^b are independent, (c) uses $\mathbb{E}_{\mathcal{D}}[n_i^b] = 0$ and (d) follows from the definition of v.

Combining the previous two equations using the triangle inequality:

$$\begin{aligned}
|\mathbb{E}_{\mathcal{D}}[\nabla f_i^b(w,\kappa) \cdot v]| &\geq |\mathbb{E}_{\mathcal{D}}[(w \cdot x_i^b - y_i^b)x_i^b \cdot v]| - |\mathbb{E}_{\mathcal{D}}[((w \cdot x_i^b - y_i^b)x_i^b - \nabla f_i^b(w,\kappa)) \cdot v]| \\
&\geq \mathbb{E}_{\mathcal{D}}\left[\frac{((w - w^*) \cdot x_i^b)^2}{\|w - w^*\|}\right] |\left(1 - \frac{8}{\kappa^2}\big(C\mathbb{E}_{\mathcal{D}}[(x_i^b \cdot (w - w^*))^2] + \sigma^2\big)\right) \\
&\geq \mathbb{E}_{\mathcal{D}}\left[\frac{((w - w^*) \cdot x_i^b)^2}{\|w - w^*\|}\right]\left(1 - \frac{1}{4}\right) \\
&\geq \frac{3}{4}\|w - w^*\| \cdot \frac{\|\Sigma\|}{C_3} = \frac{3}{4C_3}\|w - w^*\|,
\end{aligned}$$

here the second last inequality follows from lower bound on κ.

The theorem then follows by observing,

$$\|\mathbb{E}_{\mathcal{D}}[\nabla f_i^b(w,\kappa)]\| \geq \max_{\|u\|=1} |\mathbb{E}_{\mathcal{D}}[\nabla f_i^b(w,\kappa) \cdot u]| \geq |\mathbb{E}_{\mathcal{D}}[\nabla f_i^b(w,\kappa) \cdot v]| \geq \frac{3}{4C_3}\|w - w^*\|.$$

■

5.10.2 Proof of Lemma 91

Proof. Note that

$$\begin{aligned}
\|\mathrm{Cov}_\beta(z^b)\| &= \left\|\sum_{b\in B}\frac{\beta^b}{\beta^B}(z^b-\mathbb{E}_\beta[z^b])(z^b-\mathbb{E}_\beta[z^b])^\intercal\right\| \\
&\geq \left\|\sum_{b\in B'}\frac{\beta^b}{\beta^B}(z^b-\mathbb{E}_\beta[z^b])(z^b-\mathbb{E}_\beta[z^b])^\intercal\right\| \\
&\overset{\text{(a)}}{\geq} \left\|\sum_{b\in B'}\frac{1}{2|B|}(z^b-\mathbb{E}_\beta[z^b])(z^b-\mathbb{E}_\beta[z^b])^\intercal\right\| \\
&\overset{\text{(b)}}{\geq} \frac{1}{2|B|}\big\||B'|(\mathbb{E}_{B'}[z^b]-\mathbb{E}_\beta[z^b])(\mathbb{E}_{B'}[]z^b]-\mathbb{E}_\beta[z^b])^\intercal\big\| \\
&= \frac{|B'|}{2|B|}\|\mathbb{E}_\beta[z^b]-\mathbb{E}_{B'}[z^b]\|^2,
\end{aligned}$$

where (a) used $\beta^b \geq 1/2$ for $b \in B'$ and the trivial bound $\beta^B \leq |B|$ and (b) follows from the fact that any Z,

$$\left\|\sum_{b\in B'}(z^b-Z)(z^b-Z)^\intercal\right\| \geq |B'|\cdot\big\|\big(\mathbb{E}_{B'}[z^b]-Z\big)\big(\mathbb{E}_{B'}[z^b]-Z\big)^\intercal\big\|.$$

We complete the proof of the lemma by proving the above fact.

$$\begin{aligned}
&\left\|\sum_{b\in B'}(z^b-Z)(z^b-Z)^\intercal\right\| \\
&= \left\|\sum_{b\in B'}(z^b-\mathbb{E}_{B'}[z^b]+\mathbb{E}_{B'}[z^b]-Z)(z^b-\mathbb{E}_{B'}[z^b]+\mathbb{E}_{B'}[z^b]-Z)^\intercal\right\| \\
&\overset{\text{(a)}}{=} \left\|\sum_{b\in B'}\big((z^b-\mathbb{E}_{B'}[z^b])(z^b-\mathbb{E}_{B'}[z^b])^\intercal+(\mathbb{E}_{B'}[z^b]-Z)(\mathbb{E}_{B'}[z^b]-Z)^\intercal\big)\right\| \\
&\overset{\text{(b)}}{\geq} |B'|\cdot\big\|(\mathbb{E}_{B'}[z^b]-Z)(\mathbb{E}_{B'}[z^b]-Z)^\intercal\big\|,
\end{aligned}$$

here (a) follows as $\sum_{b\in B'} z^b = |B'|\mathbb{E}_{B'}[z^b]$ and hence, $\sum_{b\in B'}(z^b - \mathbb{E}_{B'}[z^b])(\mathbb{E}_{B'}[z^b] - Z)^\intercal = \sum_{b\in B'}(\mathbb{E}_{B'}[z^b] - Z)(z^b - \mathbb{E}_{B'}[z^b])^\intercal = 0$, and (b) follows as $(z^b - \mathbb{E}_{B'}[z^b])(z^b - \mathbb{E}_{B'}[z^b])^\intercal$ are positive semi-definite matrices.

■

5.11 Subroutine FindClippingParameter and its analysis

Algorithm 8. FindClippingPparameter

```
Input: Set $B$, $\beta$, $\sigma$, $a_1 \geq 1$, $a_2$ data $\{\{(x_i^b, y_i^b\}_{i\in[n]}\}_{b\in B}$.
$\kappa \leftarrow \infty$
while True do
    $w_\kappa \leftarrow$ any approximate stationary point of clipped losses $\{f^b(\,\cdot\,,\kappa)\}$ w.r.t. weight vector $\beta$ such that $\|\mathbb{E}_\beta[f^b(w_\kappa,\kappa)]\| \leq \frac{\log(2/\alpha)\sigma}{8\sqrt{n\alpha}}$
    $\kappa_{new} \leftarrow \max\Big\{a_1\sqrt{\mathbb{E}_\beta[f^b(w_\kappa,\kappa)]}, a_2\sigma\Big\}$.
    if $\kappa_{new} \geq \kappa/2$ then
        Break
    end if
    $\kappa \leftarrow \kappa_{new}$
end while
Return$(\kappa, w_\kappa)$
```

Theorem 93. *For any weight vector β, $a_1 \geq 1$, and $a_2 > 0$, Algorithm FindClippingParameter runs at most $\log\Big(\mathcal{O}\Big(\frac{\max_{i,b}|y_i^b|}{\sigma}\Big)\Big)$ iterations of the while loop and returns κ and w_κ such that*

1. *w_κ is a (approximate) stationary point for $\{f^b(\cdot,\kappa)\}$ w.r.t. weight vector β such that*

$$\|\mathbb{E}_\beta[f^b(w_\kappa,\kappa)]\| \leq \tfrac{\log(2/\alpha)\sigma}{8\sqrt{n\alpha}}.$$

2. $\max\Big\{a_1\sqrt{\mathbb{E}_\beta[f^b(w_\kappa,\kappa)]}, a_2\sigma\Big\} \leq \kappa \leq 2\max\Big\{a_1\sqrt{\mathbb{E}_\beta[f^b(w_\kappa,\kappa)]}, a_2\sigma\Big\}$.

3. $\kappa \geq \max\Big\{\frac{a_1}{2}\mathbb{E}_\beta\Big[\frac{1}{n}\sum_{i\in[n]}|w_\kappa\cdot x_i^b - y_i^b|\Big], a_2\sigma\Big\}$ *and*
   $\kappa \leq \max\Big\{4a_1^2\mathbb{E}_\beta\Big[\frac{1}{n}\sum_{i\in[n]}|w_\kappa\cdot x_i^b - y_i^b|\Big], a_2\sigma\Big\}$.

Proof. First, we bound the number of iterations of the while loop. Since w_κ is a stationary point for $f^b(.,\kappa)$, hence its will achieve a smaller loss than $w = 0$, hence $\mathbb{E}_\beta[f^b(w_\kappa,\kappa)] \le \mathbb{E}_\beta[f^b(0,\kappa)]$. And, since the clipped loss is smaller than unclipped loss, $\mathbb{E}_\beta[f^b(0,\kappa)] \le \mathbb{E}_\beta[f^b(0)] = \mathbb{E}_\beta[\frac{1}{n}\sum_{i\in[n]}(y_i^b)^2] \le \max_{i,b}(y_i^b)^2$. Therefore after the first iteration $\kappa \le \max\{a_1 \max_{i,b}|y_i^b|, a_2\sigma\}$. Also in each iteration apart from the last one κ decreases by a factor 2 and κ can't be smaller than $a_2\sigma$. Hence, the number of iterations between the first one and the last one are at most $\log(\frac{a_1\max_{i,b}|y_i^b|}{a_2\sigma}))$. Therefore the total number of iterations are at most $\log(\frac{a_1\max_{i,b}|y_i^b|}{a_2\sigma})) + 2$.

The first item follows from the definition of w_κ in the subroutine FindClippingPparameter.

Next to prove the lower bound in item 2 we prove the claim that if in an iteration $\kappa \ge \max\left\{a_1\sqrt{\mathbb{E}_\beta[f^b(w_\kappa,\kappa)]}, a_2\sigma\right\}$ then the same condition will hold in the next iteration.

The condition $\kappa \ge \max\left\{a_1\sqrt{\mathbb{E}_\beta[f^b(w_\kappa,\kappa)]}, a_2\sigma\right\}$ in the claim implies that $\kappa \ge \kappa_{new}$. Then from the definition of clipped loss, for each w and each b we have $f^b(w,\kappa) \ge f^b(w,\kappa_{new})$. It follows that $\mathbb{E}_\beta[f^b(w_\kappa,\kappa)] \ge \mathbb{E}_\beta[f^b(w_\kappa,\kappa_{new})]$. And further $w_{\kappa_{new}}$ is stationary point for $f^b(.,\kappa_{new})$, hence it will achieve a smaller loss, $\mathbb{E}_\beta[f^b(w_{\kappa_{new}},\kappa_{new})] \le \mathbb{E}_\beta[f^b(w_\kappa,\kappa_{new})]$. Therefore, $\mathbb{E}_\beta[f^b(w_{\kappa_{new}},\kappa_{new})] \le \mathbb{E}_\beta[f^b(w_\kappa,\kappa)]$. Hence, $\kappa_{new} = \max\left\{a_1\sqrt{\mathbb{E}_\beta[f^b(w_\kappa,\kappa)]}, a_2\sigma\right\} \ge \max\left\{a_1\sqrt{\mathbb{E}_\beta[f^b(w_{\kappa_{new}},\kappa_{new})]}, a_2\sigma\right\}$. This completes the proof of the claim.

Since the initial value of κ is infinite the claim must hold in the first iteration, and therefore in each iteration thereafter. Therefore it must hold in the iteration when the algorithm terminates. This completes the proof of the lower bound in item 2.

The upper bound in the second item follows by observing that when the algorithm ends $\kappa \le 2\kappa_{new}$ and $\kappa_{new} = a_1\sqrt{\mathbb{E}_\beta[f^b(w_\kappa,\kappa)]} + a_2\sigma$.

Finally, we prove item 3 using item 2. We start by proving the lower bound in item 3. From the lower bound in item 2, we have, $\kappa \ge a_2\sigma$. Then to complete the proof of the lower bound in item 3, it suffices to prove $\kappa > \frac{a_1}{2}\mathbb{E}_\beta\left[\frac{1}{n}\sum_{i\in[n]}|w_\kappa \cdot x_i^b - y_i^b|\right]$. To prove this by contradiction suppose $\kappa < \frac{a_1}{2}\mathbb{E}_\beta\Big[\frac{1}{n}\sum_{i\in[n]}|w_\kappa \cdot x_i^b - y_i^b|\Big]$. Then

$$\begin{aligned}
&\mathbb{E}_\beta[f^b(w_\kappa,\kappa)] \\
&= \mathbb{E}_\beta\left[\frac{1}{n}\sum_{i\in[n]} f_i^b(w_\kappa,\kappa)\right] \\
&= \frac{1}{n}\sum_{i\in[n]}\mathbb{E}_\beta\Bigg[\mathbb{1}(|w_\kappa \cdot x_i^b - y_i^b| \le \kappa)\cdot\frac{(w_\kappa \cdot x_i^b - y_i^b)^2}{2} \\
&\quad + \mathbb{1}(|w_\kappa \cdot x_i^b - y_i^b| > \kappa)\cdot\left(\kappa|w_\kappa \cdot x_i^b - y_i^b| - \frac{\kappa^2}{2}\right)\Bigg] \\
&\ge \frac{1}{n}\sum_{i\in[n]}\Bigg(\mathbb{E}_\beta\Bigg[\mathbb{1}(|w_\kappa \cdot x_i^b - y_i^b| \le \kappa)\cdot\frac{(w_\kappa \cdot x_i^b - y_i^b)^2}{2}\Bigg] \\
&\quad + \mathbb{E}_\beta\Bigg[\mathbb{1}(|w_\kappa \cdot x_i^b - y_i^b| > \kappa)\cdot\left(\frac{\kappa|w_\kappa \cdot x_i^b - y_i^b|}{2}\right)\Bigg]\Bigg) \\
&\overset{\text{(a)}}{\ge} \frac{1}{2n}\sum_{i\in[n]}\mathbb{E}_\beta\big[\mathbb{1}(|w_\kappa \cdot x_i^b - y_i^b| \le \kappa)\cdot|w_\kappa \cdot x_i^b - y_i^b|\big]^2 \\
&\quad + \frac{\kappa}{2n}\sum_{i\in[n]}\mathbb{E}_\beta\big[\mathbb{1}(|w_\kappa \cdot x_i^b - y_i^b| > \kappa)\cdot|w_\kappa \cdot x_i^b - y_i^b|\big] \\
&\overset{\text{(b)}}{\ge} \frac{1}{2}\left(\frac{1}{n}\sum_{i\in[n]}\mathbb{E}_\beta\big[\mathbb{1}(|w_\kappa \cdot x_i^b - y_i^b| \le \kappa)\cdot|w_\kappa \cdot x_i^b - y_i^b|\big]\right)^2 \\
&\quad + \frac{\kappa}{2n}\sum_{i\in[n]}\mathbb{E}_\beta\big[\mathbb{1}(|w_\kappa \cdot x_i^b - y_i^b| > \kappa)\cdot|w_\kappa \cdot x_i^b - y_i^b|\big] \\
&\overset{\text{(c)}}{\ge} \frac{\kappa}{a_1}\left(\frac{1}{n}\sum_{i\in[n]}\mathbb{E}_\beta\big[\mathbb{1}(|w_\kappa \cdot x_i^b - y_i^b| \le \kappa)\cdot|w_\kappa \cdot x_i^b - y_i^b|\big]\right) \\
&\quad + \frac{\kappa}{2n}\sum_{i\in[n]}\mathbb{E}_\beta\big[\mathbb{1}(|w_\kappa \cdot x_i^b - y_i^b| > \kappa)\cdot|w_\kappa \cdot x_i^b - y_i^b|\big] \\
&\overset{\text{(d)}}{\ge} \frac{\kappa}{2a_1 n}\sum_{i\in[n]}\Big(\mathbb{E}_\beta\big[\mathbb{1}(|w_\kappa \cdot x_i^b - y_i^b| \le \kappa)\cdot|w_\kappa \cdot x_i^b - y_i^b|\big] \\
&\quad + \mathbb{E}_\beta\big[\mathbb{1}(|w_\kappa \cdot x_i^b - y_i^b| > \kappa)\cdot|w_\kappa \cdot x_i^b - y_i^b|\big]\Big) \\
&= \frac{\kappa}{2a_1}\mathbb{E}_\beta\left[\frac{1}{n}\sum_{i\in[n]}|w_\kappa \cdot x_i^b - y_i^b|\right] \\
&\overset{\text{(e)}}{\ge} \frac{\kappa^2}{a_1^2},
\end{aligned} \tag{5.20}$$

here (a) and (b) follows the Cauchy-Schwarz inequality, (c) and (e) follows from our assumption $\kappa < \frac{a_1}{2}\mathbb{E}_\beta\left[\frac{1}{n}\sum_{i\in[n]}|w_\kappa \cdot x_i^b - y_i^b|\right]$ and (d) follows since $a_1 \geq 1$.

This contradicts the lower bound $\kappa \geq a_1\sqrt{\mathbb{E}_\beta[f^b(w_\kappa,\kappa)]}$ in item 2. Hence we conclude, $\kappa \geq \frac{a_1}{2}\mathbb{E}_\beta\left[\frac{1}{n}\sum_{i\in[n]}|w_\kappa \cdot x_i^b - y_i^b|\right]$. This completes the proof of the lower bound in item 3.

Next, we prove the upper bound in item 3. We consider two cases. For the case when $a_1\sqrt{\mathbb{E}_\beta[f^b(w_\kappa,\kappa)]} \leq a_2\sigma$ then upper bound in item 3 follows from the upper bound in item 2. Next we prove for the other case, when $a_1\sqrt{\mathbb{E}_\beta[f^b(w_\kappa,\kappa)]} > a_2\sigma$. For this case item 2 implies $\mathbb{E}_\beta[f^b(w_\kappa,\kappa)] \geq \frac{\kappa^2}{4a_1^2}$.

Next, from the definition of $f^b(w,\kappa)$,

$$\begin{aligned}\mathbb{E}_\beta[f^b(w_\kappa,\kappa)] &= \mathbb{E}_\beta\left[\frac{1}{n}\sum_{i\in[n]} f_i^b(w_\kappa,\kappa)\right]\\ &\leq \mathbb{E}_\beta\left[\frac{1}{n}\sum_{i\in[n]}\kappa|w_\kappa \cdot x_i^b - y_i^b|\right] \leq \kappa\mathbb{E}_\beta\left[\frac{1}{n}\sum_{i\in[n]}|w_\kappa \cdot x_i^b - y_i^b|\right]. \qquad (5.21)\end{aligned}$$

Combining the above equation and $\mathbb{E}_\beta[f^b(w_\kappa,\kappa)] \geq \frac{\kappa^2}{4a_1^2}$, we get,

$$\frac{\kappa^2}{4a_1^2} \leq \kappa\mathbb{E}_\beta\left[\frac{1}{n}\sum_{i\in[n]}|w_\kappa \cdot x_i^b - y_i^b|\right].$$

The upper bound in item 3 then follows from the above equation. ■

5.12 Correctness of estimated parameters for nice weight vectors

For batch $b \in B$, let $v^b(w) := \frac{1}{n}\sum_{i\in[n]}|w \cdot x_i^b - y_i^b|$. Since w will be fixed in the proofs, we will often denote $v^b(w)$ as v^b.

In this section, we state and prove Theorems 94, 95 and 97. For any triplet with a nice weight vector, Theorem 94 ensures the correctness of parameters calculated for Type-1 use of

Multifilter. For any triplet with a nice weight vector, Theorem 97 ensures the correctness of parameters calculated for the case when it gets added to M or goes through Type-2 use of Multifilter. Theorem 95 serves as an intermediate step in proving Theorem 97.

Theorem 94. *In Algorithm 7 if the weight vector β is such that $\beta^G \geq 3|G|/4$, $n \geq (16)^2 c_2 C$, and Theorem 85's conclusion holds, then for any w, the parameter θ_1 computed in the subroutine satisfies*

$$\theta_1 \geq c_2\left(\frac{\sigma^2 + C\mathbb{E}_{\mathcal{D}}[|w \cdot x_i^b - y_i^b|]^2}{n}\right),$$

where c_2 is the same universal positive constant as item 2 in Lemma 87.

Proof. To prove the theorem we first show that θ_0 calculated in the algorithm is $\geq \frac{7\mathbb{E}_{\mathcal{D}}[|w \cdot x_i^b - y_i^b|]}{8} - \frac{\sigma}{8\sqrt{C}}$.

Let MED denote median of the set $\{v^b : b \in G\}$. From Theorem 85 and Markov's inequality, it follows that

$$\begin{aligned}\left|\text{MED} - \mathbb{E}_{\mathcal{D}}[|w \cdot x_i^b - y_i^b|]\right| &\leq 2\sqrt{c_2\left(\frac{\sigma^2 + C\mathbb{E}_{\mathcal{D}}[|w \cdot x_i^b - y_i^b|]^2}{n}\right)} \\ &\leq \frac{\mathbb{E}_{\mathcal{D}}[|w \cdot x_i^b - y_i^b|]}{8} + \frac{\sigma}{8\sqrt{C}}. \end{aligned} \tag{5.22}$$

where the last inequality uses $n \geq (16)^2 c_2 C$. It follows that

$$\text{MED} \geq \frac{7\mathbb{E}_{\mathcal{D}}[|w \cdot x_i^b - y_i^b|]}{8} - \frac{\sigma}{8\sqrt{C}}.$$

Then to complete the proof we show that MED $\leq \theta_0$. Note that

$$\sum_{b \in G : v^b < \text{MED}} \beta^b \leq |\{b \in G : v^b < \text{MED}\}| < \frac{|G|}{2}.$$

Then,

$$\sum_{b\in B:v^b\geq \text{MED}} \beta^b \geq \sum_{b\in G:v^b\geq \text{MED}} \beta^b = \sum_{b\in G} \beta^b - \sum_{b\in G:v^b< \text{MED}} \beta^b > \beta^G - \frac{|G|}{2} \geq \frac{3|G|}{4} - \frac{|G|}{2} \geq \frac{|G|}{4}. \tag{5.23}$$

And since from the definition of θ_0, we have $\sum_{b:v^b>\theta_0} \beta^b \leq \alpha|B|/4 \leq \frac{|G|}{4}$, it follows that MED $\leq \theta_0$.

Therefore, $\theta_0 \geq \frac{7\mathbb{E}_{\mathcal{D}}[|w\cdot x_i^b - y_i^b|]}{8} - \frac{\sigma}{8\sqrt{C}}$. The lower bound in the theorem on θ_1 then follows from the relation between θ_0 and θ_1. ■

Theorem 95. *Suppose regularity conditions holds, and β, w and n satisfy $n \geq \max\{(16)^2 c_2 C, \frac{(32)^2 c_3 c_2 C \log^2(2/\alpha)}{\alpha}\}$, $\beta^G \geq 3|G|/4$, and*

$$\mathit{Var}_\beta\big(v^b(w)\big) \leq c_3 \log^2(2/\alpha)\theta_1,$$

then

$$\frac{3\mathbb{E}_{\mathcal{D}}[|(w-w^*)\cdot x_i^b|]}{4} - \sigma \leq \mathbb{E}_\beta\big[v^b(w)\big] \leq \frac{4\mathbb{E}_{\mathcal{D}}[|(w-w^*)\cdot x_i^b|]}{3} + 2\sigma.$$

In proving Theorem 95 the following auxiliary lemma will be useful. We prove this lemma in Subsection 5.12.1.

Lemma 96. *Let Z be any random variable over the reals. For any $z \in \mathbb{R}$, such that* $\Pr[Z > z] \leq 1/2$*, we have*

$$z - \sqrt{\frac{\mathit{Var}(Z)}{\Pr[Z \geq z]}} \leq \mathbb{E}[Z] \leq z + \sqrt{2\mathit{Var}(Z)}.$$

and for all $z \in Z$,

$$|\mathbb{E}[Z] - z| \leq \sqrt{\frac{\mathit{Var}(Z)}{\min\{\Pr[Z \leq z], \Pr[Z \geq z], 0.5\}}}.$$

Now we prove Theorem 95 using the above Lemma.

Proof of Theorem 95. Let MED denote median of the set $\{v^b : b \in G\}$. In Equation (5.23) we showed,

$$\sum_{b \in B : v^b \geq \text{MED}} \beta^b \geq \frac{|G|}{4}.$$

Hence,

$$\frac{\sum_{b \in B : v^b \geq \text{MED}} \beta^b}{\beta^B} \geq \frac{|G|}{4|B|} \geq \frac{\alpha}{4}.$$

Similarly, by symmetry, one can show

$$\frac{\sum_{b \in B : v^b \leq \text{MED}} \beta^b}{\beta^B} \geq \frac{\alpha}{4}.$$

Then from the second bound in Lemma 96,

$$|\mathbb{E}_\beta[v^b] - \text{MED}| \leq \sqrt{\frac{4\text{Var}_\beta[v^b]}{\alpha}}. \tag{5.24}$$

From Equation (5.22), the above equation, and the triangle inequality,

$$|\mathbb{E}_\beta[v^b] - \mathbb{E}_\mathcal{D}[|w \cdot x_i^b - y_i^b|]| \leq \sqrt{\frac{4\text{Var}_\beta[v^b]}{\alpha}} + \frac{\mathbb{E}_\mathcal{D}[|w \cdot x_i^b - y_i^b|]}{8} + \frac{\sigma}{8\sqrt{C}}. \tag{5.25}$$

Next, from the definition of θ_0, we have $\sum_{b : v^b \geq \theta_0} \beta^b \geq \alpha|B|/4$ and $\sum_{b : v^b > \theta_0} \beta^b < \alpha|B|/4$. Then

$$\frac{\sum_{b : v^b \geq \theta_0} \beta^b}{\beta^B} \geq \frac{\alpha|B|}{4\beta^B} \geq \frac{\alpha|B|}{4|B|} \geq \frac{\alpha}{4},$$

and

$$\frac{\sum_{b : v^b > \theta_0} \beta^b}{\beta^B} < \frac{\alpha|B|}{4\beta^B} \leq \frac{\alpha|B|}{4\beta^G} \leq \frac{\alpha|B|}{4(3|G|/4)} \leq \frac{1}{3}.$$

Then from the first bound in Lemma 96,

$$\theta_0 - \sqrt{\frac{4\mathrm{Var}_\beta[v^b]}{\alpha}} \leq \mathbb{E}_\beta[v^b]. \tag{5.26}$$

In this lemma, we had assumed the following bound on the variance of v^b,

$$\mathrm{Var}_\beta[v^b] \leq c_3 \log^2(2/\alpha)\theta_1.$$

Next,

$$\begin{aligned}
\frac{\mathrm{Var}_\beta[v^b]}{\alpha} &\leq \frac{c_3 \log^2(2/\alpha)\theta_1}{\alpha} \\
&= \frac{c_3 \log^2(2/\alpha) c_2(\sigma^2 + (2\sqrt{C}\theta_0 + \sigma)^2)}{n\alpha} \\
&\leq \frac{(\sigma^2 + 4C\theta_0^2 + 2\sigma^2)}{32^2 C} \leq \frac{\sigma^2}{256C} + \frac{\theta_0^2}{256},
\end{aligned}$$

here the equality follows from the relation between θ_0 and θ_1 and the first inequality follows as $n \geq \frac{(32)^2 C c_3 c_2 \log^2(2/\alpha)}{\alpha}$.

Then

$$\sqrt{\frac{\mathrm{Var}_\beta[v^b]}{\alpha}} \leq \sqrt{\frac{\sigma^2}{256C} + \frac{\theta_0^2}{256}} \leq \frac{\sigma}{16\sqrt{C}} + \frac{\theta_0}{16} \leq \frac{\sigma}{16\sqrt{C}} + \frac{1}{16}\left(\mathbb{E}_\beta[v^b] + 2\sqrt{\frac{\mathrm{Var}_\beta[v^b]}{\alpha}}\right),$$

here the second inequality used $\sqrt{a^2+b^2} \leq |a| + |b|$ and the last inequality used (5.26). From the above equation, it follows that

$$\sqrt{\frac{\mathrm{Var}_\beta[v^b]}{\alpha}} \leq \frac{\sigma}{14\sqrt{C}} + \frac{1}{14}\mathbb{E}_\beta[v^b].$$

Combining the above bound and Equation (5.25)

$$|\mathbb{E}_\beta[v^b] - \mathbb{E}_\mathcal{D}[|w \cdot x_i^b - y_i^b|]| \le \frac{\sigma}{7\sqrt{C}} + \frac{1}{7}\mathbb{E}_\beta[v^b] + \frac{\mathbb{E}_\mathcal{D}[|w \cdot x_i^b - y_i^b|]}{8} + \frac{\sigma}{8\sqrt{C}}.$$

From the above equation it follows that

$$\frac{49\mathbb{E}_\mathcal{D}[|w \cdot x_i^b - y_i^b|]}{64} - \frac{15\sigma}{64\sqrt{C}} \le \mathbb{E}_\beta[v^b] \le \frac{21\mathbb{E}_\mathcal{D}[|w \cdot x_i^b - y_i^b|]}{16} + \frac{5\sigma}{16\sqrt{C}}. \qquad (5.27)$$

Finally, we upper bound and lower bound $\mathbb{E}_\mathcal{D}[|w \cdot x_i^b - y_i^b|]$ to complete the proof. To prove the upper bound, note that,

$$\begin{aligned}\mathbb{E}_\mathcal{D}[|w \cdot x_i^b - y_i^b|] &= \mathbb{E}_\mathcal{D}[|(w - w^*) \cdot x_i^b - n_i^b|] \le \mathbb{E}_\mathcal{D}[|(w - w^*) \cdot x_i^b|] + \mathbb{E}_\mathcal{D}[|n_i^b|] \\ &\le \mathbb{E}_\mathcal{D}[|(w - w^*) \cdot x_i^b|] + \sigma,\end{aligned}$$

here the last inequality used $\mathbb{E}_\mathcal{D}[|n_i^b|] \le \sqrt{\mathbb{E}_\mathcal{D}[|n_i^b|^2]}$. Combining the above upper bound with the upper bound in (5.27) and using $C \ge 1$ proves the upper bound in the lemma. Similarly, we can show

$$\mathbb{E}_\mathcal{D}[|w \cdot x_i^b - y_i^b|] \ge \mathbb{E}_\mathcal{D}[|(w - w^*) \cdot x_i^b|] - \sigma,$$

Combining the above lower bound with the lower bounds in (5.27) and using $C \ge 1$ proves the lower bound in the lemma. ■

Theorem 97. *Suppose regularity conditions holds, and* β, w *and* n *satisfy* $n \ge \max\{\frac{(32)^2 c_3 c_2 C \log^2(2/\alpha)}{\alpha}, (16)^2 c_2 C\}$, $\beta^G \ge 3|G|/4$, *and*

$$Var_\beta\left(\frac{1}{n}\sum_{i\in[n]} |w \cdot x_i^b - y_i^b|\right) \le c_3 \log^2(2/\alpha)\theta_1,$$

then for κ, w *returned by subroutine* FindClippingParameter *and* θ_2 *calculated by* MainAlgo-

RITHM, we have

1. $c_4\frac{\sigma^2+C\mathbb{E}_{\mathcal{D}}[((w-w^*)\cdot x_i^b)^2]}{n} \le \theta_2 \le \frac{c_6C^2(\sigma^2+\mathbb{E}_{\mathcal{D}}[|(w-w^*)\cdot x_i^b|]^2)}{n}$, *where* c_4 *is the same positive constant as in item 1 of Lemma 87 and* c_6 *is some other positive universal constant.*

2. $\max\{8\sqrt{C\mathbb{E}_{\mathcal{D}}[|x_i^b\cdot(w-w^*)|^2]}, 8\sigma\} \le \kappa$ *and* $\kappa \le c_7C^2\Big(\sqrt{\mathbb{E}_{\mathcal{D}}[|x_i^b\cdot(w-w^*)|^2]}+\sigma\Big)$, *where* c_7 *is some other positive universal constant.*

Note that the range of κ in item 2 of the above Theorem is the same as that in (b).

In proving the theorem the following lemma will be useful.

Lemma 98. *For any vectors* u*, we have*

$$\sqrt{\frac{\mathbb{E}_{\mathcal{D}}[|u\cdot x_i^b|^2]}{8C}} \le \mathbb{E}_{\mathcal{D}}[|u\cdot x_i^b|] \le \sqrt{\mathbb{E}_{\mathcal{D}}[|u\cdot x_i^b|^2]}$$

We prove the above auxiliary lemma in Section 5.12.2 using the Cauchy-Schwarz inequality for the upper bound and $L4-L2$ hypercontractivity for the lower bound.

Next, we prove Theorem 97 using the above lemma and Theorem 95.

Proof of Theorem 97. We start by proving the first item. For convenience, we recall the definition of θ_2 in (5.8),

$$\theta_2 = \tfrac{c_4}{n}\Big(\sigma^2+16C^2\big(\mathbb{E}_\beta[v^b]+\sigma\big)^2\Big).$$

The upper bound in the item follows from this definition of θ_2 and the upper bound on $\mathbb{E}_\beta[v^b]$ in Lemma 95.

Using the lower bound bound on $\mathbb{E}_\beta[v^b]$ in Lemma 95 and definition of θ_2,

$$\theta_2 \ge \tfrac{c_4}{n}\big(\sigma^2+9C^2\mathbb{E}_{\mathcal{D}}[|(w-w^*)\cdot x_i^b|]^2\big) \ge \tfrac{c_4}{n}\big(\sigma^2+\tfrac{9}{8}C^2\mathbb{E}_{\mathcal{D}}[|(w-w^*)\cdot x_i^b|^2]\big),$$

where the last step used Lemma 98. This completes the proof of lower bound item 1.

Next, we prove item 2. From Theorem 93,

$$\max\Big\{\tfrac{a_1}{2}\mathbb{E}_\beta\Big[\tfrac{1}{n}\textstyle\sum_{i\in[n]}|w\cdot x_i^b - y_i^b|\Big], a_2\sigma\Big\} \le \kappa \le \max\Big\{4a_1^2\mathbb{E}_\beta\Big[\tfrac{1}{n}\textstyle\sum_{i\in[n]}|w\cdot x_i^b - y_i^b|\Big], a_2\sigma\Big\}.$$

Since for any $a, b > 0$, $(a+b)/2 \le \max(a,b) \le a+b$. Then from the above bound,

$$\tfrac{a_1}{4}\mathbb{E}_\beta\Big[\tfrac{1}{n}\textstyle\sum_{i\in[n]}|w_\kappa\cdot x_i^b - y_i^b|\Big] + \tfrac{a_2\sigma}{2} \le \kappa \le 4a_1^2\mathbb{E}_\beta\Big[\tfrac{1}{n}\textstyle\sum_{i\in[n]}|w_\kappa\cdot x_i^b - y_i^b|\Big] + a_2\sigma.$$

Using the bound on $\mathbb{E}_\beta[v^b]$ in Lemma 95 in the above equation

$$\tfrac{a_1}{4}\Big(\tfrac{3\mathbb{E}_\mathcal{D}[|(w-w^*)\cdot x_i^b|]}{4} - \tfrac{\sigma}{2}\Big) + \tfrac{a_2\sigma}{2} \le \kappa \le 4a_1^2\Big(\tfrac{4\mathbb{E}_\mathcal{D}[|(w-w^*)\cdot x_i^b|]}{3} + 2\sigma\Big) + a_2\sigma.$$

Using Lemma 98, and the above equation,

$$\tfrac{3a_1}{32\sqrt{2C}}\sqrt{\mathbb{E}_\mathcal{D}[|(w-w^*)\cdot x_i^b|^2]} + \tfrac{(4a_2-a_1)\sigma}{8} \le \kappa \le \tfrac{16a_1^2}{3}\sqrt{\mathbb{E}_\mathcal{D}[|(w-w^*)\cdot x_i^b|^2]} + (8a_1^2 + a_2)\sigma.$$

The upper bound and lower bound in item 2 then follow by using the values $a_1 = \frac{256C\sqrt{2}}{3}$ and $a_2 = \frac{a_1}{4} + 64$. ■

5.12.1 Proof of Lemma 96

Proof of Lemma 96. We only prove the first statement as the second statement and then follow from the symmetry.

We start by proving the upper bound in the first statement. We consider two cases, $\mathbb{E}[Z] \le z$ and $\mathbb{E}[Z] > z$. For the first case, the upper bound automatically follows. Next, we prove the second case. In this case,

$$\begin{aligned}\mathrm{Var}(Z) &= \mathbb{E}[(Z-\mathbb{E}[Z])^2] \\ &\ge \mathbb{E}[\mathbb{1}(Z\le z)(Z-\mathbb{E}[Z])^2] \ge \mathbb{E}[\mathbb{1}(Z\le z)(z-\mathbb{E}[Z])^2] = \Pr[Z\le z](z-\mathbb{E}[Z])^2.\end{aligned}$$

Then using $\Pr[Z \leq z] = 1 - \Pr[Z > z] \geq 1/2$, we get

$$\text{Var}(Z) \geq \frac{(z - \mathbb{E}[Z])^2}{2}.$$

The upper bound from the above equation.

Next, we prove the lower bound. Again, we consider two cases, $\mathbb{E}[Z] \geq z$ and $\mathbb{E}[Z] < z$. For the first case, the lower bound automatically follows. Next, we prove the second case. In this case,

$$\begin{aligned}\text{Var}(Z) &= \mathbb{E}[(Z - \mathbb{E}[Z])^2] \\ &\geq \mathbb{E}[\mathbb{1}(Z \geq z)(Z - \mathbb{E}[Z])^2] \geq \mathbb{E}[\mathbb{1}(Z \geq z)(z - \mathbb{E}[Z])^2] = \Pr[Z \geq z](z - \mathbb{E}[Z])^2,\end{aligned}$$

from which the lower bound follows.

By symmetry, for any $z \in \mathbb{R}$, such that $\Pr[Z < z] \leq 1/2$, one can show that

$$z - \sqrt{2\text{Var}(Z)} \leq \mathbb{E}[Z] \leq z + \sqrt{\frac{\text{Var}(Z)}{\Pr[Z \leq z]}}.$$

Since for any z, either $\Pr[Z > z] \leq 1/2$ or $\Pr[Z < z] \leq 1/2$, Hence, either the first bound in the Lemma or the above bound holds for each z, therefore for any $z \in \mathbb{R}$,

$$z - \max\left\{\sqrt{\frac{\text{Var}(Z)}{\Pr[Z \geq z]}}, \sqrt{2\text{Var}(Z)}\right\} \leq \mathbb{E}[Z] \leq z + \max\left\{\sqrt{\frac{\text{Var}(Z)}{\Pr[Z \leq z]}}, \sqrt{2\text{Var}(Z)}\right\}.$$

The second bound in the lemma is implied by the above bound. ■

5.12.2 Proof of Lemma 98

Proof of Lemma 98. The upper bound on $\mathbb{E}_{\mathcal{D}}[|u \cdot x_i^b|]$ follows from the Cauchy-Schwarz inequality,

$$\mathbb{E}_{\mathcal{D}}[|u \cdot x_i^b|] \leq \sqrt{\mathbb{E}_{\mathcal{D}}[|u \cdot x_i^b|^2]} \leq \|\Sigma\| \leq 1.$$

Next, we prove the lower bound. From Markov's inequality

$$\begin{aligned}\Pr{}_{\mathcal{D}}[\|x_i^b \cdot u\|^2 \geq 2C\mathbb{E}_{\mathcal{D}}[\|x_i^b \cdot u\|^2]] &= \Pr{}_{\mathcal{D}}[\|x_i^b \cdot u\|^4 \geq 4C^2(\mathbb{E}_{\mathcal{D}}[\|x_i^b \cdot u\|^2])^2] \\ &= \frac{\mathbb{E}_{\mathcal{D}}[\|x_i^b \cdot u\|^4]}{4C^2(\mathbb{E}_{\mathcal{D}}[\|x_i^b \cdot u\|^2])^2} \leq \frac{1}{4C},\end{aligned}$$

where the last step uses $L4 - L2$ hypercontractivity.

Then, from the Cauchy-Schwarz inequality,

$$\begin{aligned}&\mathbb{E}_{\mathcal{D}}[\|x_i^b \cdot u\|^2 \cdot \mathbb{1}(\|x_i^b \cdot u\|^2 \geq 2C\mathbb{E}_{\mathcal{D}}[\|x_i^b \cdot u\|^2])] \\ &\leq \sqrt{\mathbb{E}_{\mathcal{D}}\big[\mathbb{1}\big(\|x_i^b \cdot u\|^2 \geq 2C\mathbb{E}_{\mathcal{D}}[\|x_i^b \cdot u\|^2]\big)\big] \cdot \mathbb{E}_{\mathcal{D}}[\|x_i^b \cdot u\|^4]} \\ &\leq \sqrt{\Pr{}_{\mathcal{D}}\big[\|x_i^b \cdot u\|^2 \geq 2C\mathbb{E}_{\mathcal{D}}[\|x_i^b \cdot u\|^2]\big] \cdot C\mathbb{E}_{\mathcal{D}}[\|x_i^b \cdot u\|^2]^2} \\ &\leq \frac{1}{2}\mathbb{E}_{\mathcal{D}}[\|x_i^b \cdot u\|^2].\end{aligned}$$

Then,

$$\begin{aligned}&\mathbb{E}_{\mathcal{D}}[\|x_i^b \cdot u\|^2 \cdot \mathbb{1}(\|x_i^b \cdot u\|^2 < 2C\mathbb{E}_{\mathcal{D}}[\|x_i^b \cdot u\|^2])] \\ &= \mathbb{E}_{\mathcal{D}}[\|x_i^b \cdot u\|^2] - \mathbb{E}_{\mathcal{D}}[\|x_i^b \cdot u\|^2 \cdot \mathbb{1}(\|x_i^b \cdot u\|^2 \geq 2C\mathbb{E}_{\mathcal{D}}[\|x_i^b \cdot u\|^2])] \\ &\geq \frac{1}{2}\mathbb{E}_{\mathcal{D}}[\|x_i^b \cdot u\|^2]\end{aligned}$$

Next,

$$\begin{aligned}&\mathbb{E}_{\mathcal{D}}\big[\|x_i^b \cdot u\|^2 \cdot \mathbb{1}\big(\|x_i^b \cdot u\|^2 < 2C\mathbb{E}_{\mathcal{D}}[\|x_i^b \cdot u\|^2]\big)\big] \\ &\leq \mathbb{E}_{\mathcal{D}}\left[\|x_i^b \cdot u\| \cdot \sqrt{2C\mathbb{E}_{\mathcal{D}}[\|x_i^b \cdot u\|^2]} \cdot \mathbb{1}\big(\|x_i^b \cdot u\|^2 < 2C\mathbb{E}_{\mathcal{D}}[\|x_i^b \cdot u\|^2]\big)\right] \\ &\leq \sqrt{2C\mathbb{E}_{\mathcal{D}}[\|x_i^b \cdot u\|^2]} \cdot \mathbb{E}_{\mathcal{D}}[\|x_i^b \cdot u\|].\end{aligned}$$

Combining the above two equations we get

$$\mathbb{E}_{\mathcal{D}}[\|x_i^b \cdot u\|] \geq \frac{\mathbb{E}_{\mathcal{D}}[\|x_i^b \cdot u\|^2]}{2\sqrt{2C\mathbb{E}_{\mathcal{D}}[\|x_i^b \cdot u\|^2]}} = \frac{\sqrt{\mathbb{E}_{\mathcal{D}}[\|x_i^b \cdot u\|^2]}}{2\sqrt{2C}}.$$

■

5.13 Multi-filtering

In this section, we state the subroutine MULTIFILTER, a simple modification of BASICMULTIFILTER algorithm in [51].

The subroutine takes a weight vector β, a real function z^b on batches, and a parameter θ as input and produces new weight vectors.

This subroutine is used only when:

$$\text{Var}_{B,\beta}(z^b) > c_3 \log^2(2/\alpha)\theta, \tag{5.28}$$

where c_3 is an universal constant (Same as $2 * C$, where C is the constant in BASICMULTIFILTER algorithm in [51]).

When the variance of z^b for good batches is smaller than θ and the weight vector β is nice that is $\beta^G \geq 3/4|G|$, then at least one of the new weight vectors produced by this subroutine has a higher fraction of weights in good vector than the original weight vector β.

In BASICMULTIFILTER subroutine of [51] input is not restricted by the condition in Equation (5.28). However, when input meets this condition BASICMULTIFILTER and its modification MULTIFILTER behaves the same.

Therefore, the guarantees for weight vectors returned by MULTIFILTER follows from the guarantees of BASICMULTIFILTER in [51]. We characterize these guarantees in Theorem 99.

Theorem 99. *Let* $\{z^b\}_{b\in B}$ *be collection of real numbers associated with batches,* β *be a weight vector, and threshold* $\theta > 0$ *be such that condition in* (5.28) *holds. Then*

Algorithm 9. MULTIFILTER

```
Input: Set $B$, $\alpha$, $\beta$, $\{z^b\}_{b\in B}$, $\theta$. {Input must satisfy Condition (5.28)}
Let $a = \inf\{z : \sum_{b:z^b<z} \beta^b \le \alpha\beta^B/8\}$ and $b = \sup\{z : \sum_{b:z^b>z} \beta^b \le \alpha\beta^B/8\}$
Let $B' = \{b \in B : z^b \in [a, b]\}$
if $\mathrm{Var}_{B',\beta}(z^b) \le \frac{c_3 \log^2(2/\alpha)\theta}{2}$ then
    Let $f^b = \min_{z\in[a,b]} |z^b - z|^2$, and the new weight of each batch $b \in B$ be

        $\beta^b_{\text{new}} = \Big(1 - \frac{f^b}{\max_{b\in B:\beta^b>0} f^b}\Big)\beta^b$        (5.29)

    NEWWEIGHTS ← $\{\beta_{\text{new}}\}$
else
    Find $z \in \mathbb{R}$ and $R > 0$ such that sets $B' = \{b \in B : z^b \ge z - R\}$ and $B'' = \{b \in B : z^b < z + R\}$ satisfy

        $(\beta^{B'})^2 + (\beta^{B''})^2 \le (\beta^B)^2,$        (5.30)

    and

        $\min\Big(1 - \frac{\beta^{B'}}{\beta^B}, 1 - \frac{\beta^{B''}}{\beta^B}\Big) \ge \frac{48\log(\frac{2}{\alpha})}{R^2}.$        (5.31)

    {Existence of such $z$ and $R$ is guaranteed as shown in Lemma 3.6 of [51].}
    For each $b \in B$, let $\beta^b_1 = \beta^b \cdot \mathbb{1}(b \in B')$ and $\beta^b_2 = \beta^b \cdot \mathbb{1}(b \in B'')$. Let $\beta_1 = \{\beta^b_1\}_{b\in B}$ and $\beta_2 = \{\beta^b_2\}_{b\in B}$.
    NEWWEIGHTS ← $\{\beta_1, \beta_2\}$
end if
Return(NEWWEIGHTS)
```

MULTIFILTER($B, \beta, \{z^b\}_{b\in B}, \theta_1$) returns a list NEWWEIGHTS containing either one or two new weight vectors such that,

1. *Sum of square of the total weight of new weight vectors is bounded by the square of the total weight of β, namely*

$$\sum_{\widetilde{\beta}\in \textit{NewWeights}} (\widetilde{\beta}^B)^2 \le (\beta^B)^2. \tag{5.32}$$

2. *In the new weight vectors returned the weight of at least one of the weight vectors has been*

set to zero, that is for each weight vector $\widetilde{\beta} \in$ NewWeights,

$$\{b : \tilde{\beta}^b > 0\} \subset \{b : \beta^b > 0\}, \tag{5.33}$$

3. *If weight vector β is such that $\beta^G \geq 3|G|/4$ and for good batches the variance $Var_G(z^b) \leq \theta$ is bounded, then for at least one of the weight vector $\widetilde{\beta} \in$ NewWeights,*

$$\frac{\beta^G - \tilde{\beta}^G}{\beta^G} \leq \frac{\beta^B - \widetilde{\beta}^B}{\beta^B} \cdot \frac{1}{24\log(2/\alpha)}. \tag{5.34}$$

Proof. When the list NewWeights contains one weight vector it is generated using Equation (5.29), and when the list NewWeights contains one weight vector it is generated using Equations (5.30) and (5.31). In both cases, item 1 and item 2 of the Theorem follow immediately from these equations. The last item follows from Corollary 3.8 in [51]. ■

5.13.1 Guarantees for the use of Multifilter in Algorithm 7

The following Theorem characterizes the use Multifilter by our algorithm. The proof of the theorem is similar to the proofs for the main algorithm in [51].

Theorem 100. *At the end of Algorithm 7 the size of M is at most $4/\alpha^2$ and the algorithm makes at most $\mathcal{O}(|B|/\alpha^2)$ calls to Multifilter. And, if for every use of subroutine Multifilter by the algorithm we have $Var_G(z^b) \leq \theta$ then there is at least one triplet (β, w, κ) in M such that $\beta^G \geq 3|G|/4$.*

Proof. First note that the if blocks in Algorithm 7 ensures that for every use of subroutine Multifilter Equation (5.28) is satisfied, therefore we can use the guarantees in Theorem 99.

First we upper bound the size of M.

The progress of Algorithm 7 may be described using a tree. The internal nodes of this tree are the weight vectors that have gone through subroutine Multifilter at some point of the algorithm, and children of these internal nodes are new weight vectors returned by Multifilter.

Observe that any weight vector β encountered in Algorithm 7 is ignored iff $\beta^B < \alpha|B|/2$. If it is not ignored then either it is added to M (in form of a triplet), or else it goes through subroutine MULTIFILTER.

It follows that, if a node β is an internal node or a leaf in M then

$$\beta^B \geq \alpha|B|/2. \tag{5.35}$$

From Equation (5.32), it follows that the total weight squared for each node is greater than equal to that of its children. It follows that the total weight squared of the root, β_{init} is greater than equal to the sum of the square of weights of all the leaves. And since all weight vectors in M are among the leaves of the tree, and have total weight at least $\alpha|B|/2$,

$$(\beta_{\text{init}}^B)^2 \geq \sum_{\beta \in M} (\beta^B)^2 \geq \sum_{\beta \in M} (\frac{\alpha|B|}{2})^2,$$

here the last step follows from Equation (5.35). Using $\beta_{\text{init}}^B = |B|$, in the above equation we get $|M| \leq 4/\alpha^2$.

Similarly, it can be shown that the number of branches in the tree is at most $\mathcal{O}(1/\alpha^2)$. Item 2 in Theorem 99 implies that each iteration of MULTIFILTER zeroes out the weight of one of the batches. Hence for any weight β at depth d, we have $\beta^B \leq |B| - d$. Therefore, the depth of the tree can't be more than $|B|$. Hence, the number of nodes in the tree is upper bounded by $\mathcal{O}(|B|/\alpha^2)$. And since each call to MULTIFILTER corresponds to a non-leaf node in the tree, the total calls to MULTIFILTER by Algorithm 7 are upper bounded by $\mathcal{O}(|B|/\alpha^2)$.

Next, we show that if for each use of MULTIFILTER we have $\text{Var}_G(z^b) \leq \theta$ then one of the weight vector $\beta \in M$ must satisfy $\beta^G \geq 3|G|/4$.

Let $\beta_0 = \beta_{\text{init}}$ and suppose for each i, weight vectors β_i and β_{i+1} are related as follows:

$$\frac{\beta_i^G - \beta_{i+1}^G}{\beta_i^G} \leq \frac{\beta_i^B - \beta_{i+1}^B}{\beta_i^B} \cdot \frac{1}{24\log(2/\alpha)}. \tag{5.36}$$

Then Lemma 3.12 in [51] showed that under the above relation, for each i, we have $\beta_i^G \geq 3|G|/4$.

We show that there is a branch of the tree such that β_i and β_{i+1} are related using the above equation, where for each i, β_i denote the weight vector corresponding to the node at i^{th} level in this branch. From the preceding discussion, this would imply that for each i, $\beta_i^G \geq 3|G|/4$.

We prove it by induction. For $i = 0$, we select $\beta_i = \beta_{\text{init}}$. Note that $\beta_{\text{init}}^G = |G|$, hence $\beta_i^G \geq 3|G|/4$.

If β_i is a leaf then the branch is complete. Else, since $\beta_i^G \geq 3|G|/4$, item 3 in Theorem 99 implies that we can select one of the child of β_i as β_{i+1} so that (5.36) holds. Then from the preceding discussion, we have $\beta_{i+1}^G \geq 3|G|/4$. By repeating this argument, we keep finding the next node in the branch, until we reach the leaf. Next, we argue that the leaf at the end of this branch must be in M.

Let β denote the weight vector for the leaf. From the above discussion, it follows that $\beta^G \geq 3|G|/4$. Hence, $\beta^B \geq \beta^G \geq 3|G|/4 \geq 3\alpha|B|/4 > \alpha|B|/2$.

As discussed earlier any leaf β is not part of M iff $\beta^B \leq \alpha|B|/2$. Hence, the leaf at the end of the above branch must be in M. This concludes the proof of the Theorem. ■

5.14 Eliminating Additional Distributional Assumptions

In this section, we discuss how we can remove assumptions 2 and 5 regarding the distribution of data in Section 5.2 of the main paper. We demonstrate that our results can still be achieved without these assumptions.

Assumption 2 states that there exists a constant $C_1 > 0$ such that for random samples $(x_i^b, y_i^b) \sim \mathcal{D}$, we have $\|x_i^b\| \leq C_1\sqrt{d}$ almost surely. In the non-batch setting, Cherapanamjeri et al. (2020) [38] have shown that this assumption is not limiting. They have proven that if other assumptions are met, then there exists a constant C_1 such that the probability of $\|x_i^b\| \leq C_1\sqrt{d}$ exceeds 0.99. Thus, discarding the samples where $\|x_i^b\| > C_1\sqrt{d}$ does not significantly reduce the dataset's size. Additionally, it has minimal impact on the covariance matrix and hypercontractivity constants of the distribution This reasoning can be easily extended to the batch setting. In the batch setting, we first exclude samples from batches where $\|x_i^b\| > C_1\sqrt{d}$. We then remove batches that have been reduced by more than 10% of their original size. Since, on average, this operation would remove $\leq 1\%$ of samples from genuine batches, a simple argument using the Markov inequality shows that the probability of removing a genuine batch is at most 10%. It can be demonstrated that with high probability, the fraction of genuine batches that are removed for any component is $\lesssim 10\%$. Therefore, assumption 2 regarding data distribution is not required, and this simple procedure can be used to enforce assumption 2, resulting in a decrease in batch size and α by at most 10%. Consequently, the guarantees in our theorem are altered by only a small factor.

Assumption 5 states that the noise distribution is symmetric. We can address this by employing a simple technique. Let's consider two independent samples (x_i^b, y_i^b) and (x_{i+1}^b, y_{i+1}^b), where $y_j^b = w^* \cdot x_j^b + n_j^b$ for $j \in \{i, i+1\}$. We define $\tilde{x}_i^b = (x_i^b - x_{i+1}^b)/\sqrt{2}$, $\tilde{y}_i^b = (y_i^b - y_{i+1}^b)/\sqrt{2}$, and $\tilde{n}_i^b = (n_i^b - n_{i+1}^b)/\sqrt{2}$. Since n_i^b and n_{i+1}^b are i.i.d., the distribution of $\tilde{n}_i^b$ is symmetric around 0 and the variance of $\tilde{n}_i^b$ matches that of n_i^b. Moreover, the covariance of $\tilde{x}_i^b$ is the same as that of x_i^b, and we have $\tilde{y}_i^b = w^* \cdot \tilde{x}_i^b + \tilde{n}_i^b$. Therefore, the new sample $(\tilde{x}_i^b, \tilde{y}_i^b)$ obtained by combining two i.i.d. samples (x_i^b, y_i^b) and (x_{i+1}^b, y_{i+1}^b) in a batch satisfies the same distributional assumptions as before, and in addition, ensures a symmetric noise distribution. We can apply this approach to combine every two samples in a batch, which only reduces the batch size by a constant factor of 1/2. Thus, the assumption of symmetric noise can be eliminated by increasing the required batch sizes in our theorems by a factor of 2.

5.15 Proof of Theorem 84

In Section 5.15.1, we state and prove two auxiliary lemmas that will be used in proving Theorem 84, and in Section 5.15.2, we prove Theorem 84.

We will use the following notation in describing the auxiliary lemmas and in the proofs.

Let $S := \{(x_i^b, y_i^b) : b \in G, i \in [n]\}$ denote the collection of all good samples. Note that $|S| = |G|n$.

For any function h over (x, y), we denote the expectation of h w.r.t. uniform distribution on subset $S' \subseteq S$ by $\mathbb{E}_{S'}[h(x_i^b, y_i^b)] := \sum_{(x_i^b, y_i^b)\in S'} \frac{h(x_i^b, y_i^b)}{|S'|}$.

5.15.1 Auxiliary lemmas

In this subsection, we state and prove Lemmas 101 and 102. We will use these lemmas in proof of Theorem 84 in the following subsection.

In the next lemma, for any unit vectors u, we bound the expected second moment of the tails of $|x_i^b \cdot u|$, for covariate x_i^b of a random sample from the distribution $\mathcal{D}$.

Lemma 101. *For all $\theta > 1$, and all unit vectors $u \in \mathbb{R}^d$,*

$$\Pr{}_{\mathcal{D}}[\|x_i^b \cdot u\|^2 \geq \sqrt{C}\theta] \leq \frac{1}{\theta^2} \textit{ and } \mathbb{E}_{\mathcal{D}}[\mathbb{1}(\|x_i^b \cdot u\|^2 \geq \sqrt{C}\theta) \cdot \|x_i^b \cdot u\|^2] \leq \frac{\sqrt{C}}{\theta}$$

Proof. The first part of the lemma follows from Markov's inequality,

$$\Pr{}_{\mathcal{D}}[\|x_i^b \cdot u\|^2 \geq \sqrt{C}\theta] = \Pr{}_{\mathcal{D}}[\|x_i^b \cdot u\|^4 \geq C\theta^2] = \frac{\mathbb{E}_{\mathcal{D}}[\|x_i^b \cdot u\|^4]}{C\theta^2} \leq \frac{1}{\theta^2},$$

where the last step uses $L4 - L2$ hypercontractivity. This proves the first bound in the lemma.

For the second bound, note that

$$\begin{aligned}\mathbb{E}_{\mathcal{D}}[\mathbb{1}(\|x_i^b \cdot u\|^2 \geq \sqrt{C}\theta) \cdot \|x_i^b \cdot u\|^2] &\overset{\text{(a)}}{\leq} \sqrt{\mathbb{E}_{\mathcal{D}}[\mathbb{1}(\|x_i^b \cdot u\|^2 \geq \sqrt{C}\theta)] \cdot \mathbb{E}_{\mathcal{D}}[\|x_i^b \cdot u\|^4]} \\ &\overset{\text{(b)}}{\leq} \sqrt{\Pr{}_{\mathcal{D}}[\|x_i^b \cdot u\|^2 \geq \sqrt{C}\theta] \cdot C\mathbb{E}_{\mathcal{D}}[\|x_i^b \cdot u\|^2]^2} \\ &\overset{\text{(c)}}{\leq} \frac{\sqrt{C}}{\theta}\mathbb{E}_{\mathcal{D}}[\|x_i^b \cdot u\|^2],\end{aligned}$$

here (s) follows from the Cauchy-Schwarz inequality, (b) uses $L4 - L2$ hypercontractivity, and (c) follows from the first bound in the lemma. ■

In the next lemma, for any unit vectors u, we provide a high probability bound on the expected second moment of the tails of $|x_i^b \cdot u|$, wheres x_i^b are covariates of samples in good batches G.

Lemma 102. *For any given $\theta > 1$, and $|G|n = \Omega(d\theta^2 \log(\frac{C_1 d\theta}{C}))$, with probability at least $1 - 2/d^2$, for all unit vectors u,*

$$\mathbb{E}_S\Big[\mathbb{1}\Big(\|x_i^b \cdot u\|^2 \geq 3\sqrt{C}\theta\Big) \cdot \|x_i^b \cdot u\|^2\Big] \leq \mathcal{O}\left(\frac{\sqrt{C}}{\theta}\right).$$

The following lemma restates Lemma 5.1 of [38]. The lemma shows that for any large subset of S, the covariance of covariates x_i^b in S is close to the true covariance for distribution $\mathcal{D}$ of samples. We will use this result in proving Lemma 102.

Lemma 103. *For any fix $\theta > 1$, and $|G|n = \Omega(d\theta^2 \log(d\theta))$, with probability at least $1 - 1/d^2$ for all subsets of $S' \subseteq S$ of size $\geq (1 - \frac{1}{\theta^2})|S|$, we have*

$$\Sigma - \mathcal{O}\left(\frac{\sqrt{C}}{\theta}\right) \cdot I \preceq \mathbb{E}_{S'}[x_i^b (x_i^b)^\intercal] \preceq \Sigma + \mathcal{O}\left(\frac{\sqrt{C}}{\theta}\right) \cdot I.$$

Remark 2. Lemma 5.1 of [38] assumes that hypercontractive parameter C is a constant and its dependence doesn't appear in their lemma but is implicit in their proof. hides/ignores its dependence.

The following corollary is a simple consequence. We will use this corollary in proving Lemma 102.

Corollary 104. *For any fix $\theta > 1$, and $|G|n = \Omega(d\theta^2 \log(d\theta))$, with probability at least $1 - 1/d^2$ for all subsets $S' \subseteq S$ of size $\leq \frac{|S|}{\theta^2}$ and all unit vectors u, we have*

$$\frac{|S'|}{|S|} \cdot \mathbb{E}_{S'}[(x_i^b \cdot u)^2] \preceq \mathcal{O}\left(\frac{\sqrt{C}}{\theta}\right).$$

Proof. Consider any set S' of size $\leq \frac{|S|}{\theta^2}$. Since $|S \setminus S'| \geq (1 - \frac{1}{\theta^2})|S|$, applying Lemma 103 for $S \setminus S'$ and S,

$$\Sigma - \mathcal{O}\left(\frac{\sqrt{C}}{\theta}\right) \cdot I \preceq \mathbb{E}_{S \setminus S'}[x_i^b (x_i^b)^\intercal],$$

and

$$\mathbb{E}_S[x_i^b (x_i^b)^\intercal] \preceq \Sigma + \mathcal{O}\left(\frac{\sqrt{C}}{\theta}\right) \cdot I.$$

Next,

$$\begin{aligned} \mathbb{E}_S[x_i^b (x_i^b)^\intercal] &= \frac{|S'|}{|S|}\mathbb{E}_{S'}[x_i^b (x_i^b)^\intercal] + \frac{|S \setminus S'|}{|S|}\mathbb{E}_{S \setminus S'}[x_i^b (x_i^b)^\intercal] \\ \implies |S'|\mathbb{E}_{S'}[x_i^b (x_i^b)^\intercal] &= |S|\mathbb{E}_S[x_i^b (x_i^b)^\intercal] - |S \setminus S'|\mathbb{E}_{S \setminus S'}[x_i^b (x_i^b)^\intercal]). \end{aligned}$$

Combining the previous three equations,

$$\begin{aligned} |S'|\mathbb{E}_{S'}[x_i^b (x_i^b)^\intercal] &\preceq |S|\left(\Sigma + \mathcal{O}\left(\frac{\sqrt{C}}{\theta}\right) \cdot I\right) - |S \setminus S'|\left(\Sigma - \mathcal{O}\left(\frac{\sqrt{C}}{\theta}\right) \cdot I\right) \\ &\preceq |S'|\Sigma + (|S| + |S \setminus S'|)\mathcal{O}\left(\frac{\sqrt{C}}{\theta}\right) \cdot I \\ &\preceq \frac{1}{\theta^2}|S|\Sigma + 2|S|\mathcal{O}\left(\frac{\sqrt{C}}{\theta}\right) \cdot I \preceq 3|S|\mathcal{O}\left(\frac{\sqrt{C}}{\theta}\right) \cdot I, \end{aligned}$$

where the last line used $\Sigma \preceq I$, $|S'| \leq |S|/\theta^2$, $C \geq 1$, and $1/\theta^2 \leq 1/\theta$ for $\theta \geq 1$.

Finally, observing that for any unit vector $u^\intercal \mathbb{E}_{S'}[x_i^b (x_i^b)^\intercal]u = \mathbb{E}_{S'}[(x_i^b \cdot u)^2]$ completes the proof. ■

Now we complete the proof of the Lemma 102 with help of the above corollary.

Proof of Lemma 102. From Lemma 101 we have $\mathbb{E}_{\mathcal{D}}[1(\|x_i^b \cdot u\|^2 \geq \sqrt{C}\theta)] = \Pr[\|x_i^b \cdot u\|^2 \geq$

$\sqrt{C}\theta] \le \frac{1}{\theta^2}$ Applying Chernoff bound for random variable $\mathbb{1}(\|x_i^b \cdot u\|^2 \ge \sqrt{C}\theta)$,

$$\begin{aligned}\Pr\left[\mathbb{E}_S[1(\|x_i^b \cdot u\|^2 \ge \sqrt{C}\theta)] \le \frac{2}{\theta^2}\right] &= \Pr\left[\frac{1}{|S|}\sum_{(i,b)\in S} 1(\|x_i^b \cdot u\|^2 \ge \sqrt{C}\theta) \le \frac{2}{\theta^2}\right] \\ &\le \exp\left(-\frac{|S|}{3\theta^2}\right).\end{aligned}$$

Hence, for a fix unit vector u, with probability $\ge 1 - \exp\left(-\frac{|S|}{3\theta^2}\right)$

$$\mathbb{E}_S[1(\|x_i^b \cdot u\|^2 \le \sqrt{C}\theta)] \le |S|\frac{2}{\theta^2}.$$

Next, we show that this bound holds uniformly over all unit vectors u.

Consider an $\sqrt{\frac{\sqrt{C}\theta}{2C_1 d}}-$ net of unit sphere $\{u \in \mathbb{R}^d : \|u\| \le 1\}$ such that for any vector u in this ball there exist a u' in the net such that $\|u - u'\| \le \sqrt{\frac{\sqrt{C}\theta}{2C_1 d}}$. The standard covering argument [146] shows the existence of such a net of size $e^{\mathcal{O}(d\log(\frac{C_1 d}{C\theta}))}$. Then from the union bound, for all vectors u in this net with probability at least $1 - e^{\mathcal{O}(d\log(\frac{C_1 d}{C\theta}))}e^{-\frac{|S|}{3\theta^2}}$,

$$\mathbb{E}_S[1(\|x_i^b \cdot u\|^2 \le \sqrt{C}\theta)] \le |S|\frac{2}{\theta^2}.$$

Since $\frac{|S|}{3\theta^2} = \frac{|G|n}{3\theta^2} \gg d\log(\frac{C_1 d\theta}{C}) \ge d\log(\frac{C_1 d}{C\theta}))$, therefore, $e^{\mathcal{O}(d\log(\frac{C_1 d}{C\theta}))}e^{-\frac{|S|}{3\theta^2}} \ll e^{-\frac{|S|}{6\theta^2}} \ll 1/d^2$.

Now consider any vector u in unit ball and u' in the net such that $\|u - u'\| \le \sqrt{\frac{\sqrt{C}\theta}{2C_1 d}}$. Then

$$\begin{aligned}(x_i^b \cdot u)^2 = (x_i^b \cdot (u' + (u - u')))^2 = 2(x_i^b \cdot u')^2 + (x_i^b \cdot (u - u'))^2 \\ &\le 2(x_i^b \cdot u')^2 + 2\|u - u'\|^2\|x_i^b\|^2 \\ &\le 2(x_i^b \cdot u')^2 + 2\frac{\sqrt{C}\theta}{2C_1 d}C_1 d \le 2(x_i^b \cdot u')^2 + \sqrt{C}\theta,\end{aligned}$$

where in the last line we used the assumption that $\|x_i^b\| \le C_1\sqrt{d}$. When $(x_i^b \cdot u')^2 \le \sqrt{C}\theta$, then above sum is bounded by $2\sqrt{C}\theta$. It follows that with probability $\geq 1 - 1/d^2$, for all unit vectors u,

$$\mathbb{E}_S[1(\|x_i^b \cdot u\|^2 \leq 3\sqrt{C}\theta)] \leq |S|\frac{2}{\theta^2}.$$

Applying Corollary 104 for $S' = \{\|x_i^b \cdot u\|^2 \leq 3\sqrt{C}\theta\}$, proves the lemma

$$\mathbb{E}_S[\mathbb{1}(\|x_i^b \cdot u\|^2 \geq 3\sqrt{C}\theta) \cdot \|x_i^b \cdot u\|^2] = \frac{|S'|}{|S|}\mathbb{E}_{S'}[\|x_i^b \cdot u\|^2] \leq \mathcal{O}\left(\frac{\sqrt{C}}{\theta}\right).$$

■

5.15.2 Proof of Theorem 84

Proof of Theorem 84. Note that

$$\begin{aligned}
&\mathbb{E}_G\Big[\big(\nabla f^b(w,\kappa) \cdot u - \mathbb{E}_{\mathcal{D}}[\nabla f^b(w,\kappa) \cdot u]\big)^2\Big] \\
&= \frac{1}{|G|}\sum_{b \in G}\big(\nabla f^b(w,\kappa) \cdot u - \mathbb{E}_{\mathcal{D}}[\nabla f^b(w,\kappa) \cdot u]\big)^2 \\
&= \frac{1}{|G|}\sum_{b \in G}\left(\frac{1}{n}\sum_{i \in n}\nabla f_i^b(w,\kappa) \cdot u - \mathbb{E}_{\mathcal{D}}[\nabla f^b(w,\kappa) \cdot u]\right)^2 \\
&= \frac{1}{|G|}\sum_{b \in G}\left(\frac{1}{n}\sum_{i \in n}\big(\nabla f_i^b(w,\kappa) \cdot u - \mathbb{E}_{\mathcal{D}}[\nabla f_i^b(w,\kappa) \cdot u]\big)\right)^2,
\end{aligned}$$

where in the last step we used the expectation of batch and sample gradients are the same, namely $\mathbb{E}_{\mathcal{D}}[\nabla f_i^b(w,\kappa) \cdot u] = \mathbb{E}_{\mathcal{D}}[\nabla f^b(w,\kappa) \cdot u]$.

For any positive $\rho > 0$ and unit vector u, define

$$g_i^b(w,\kappa,u,\rho) := \frac{\nabla f_i^b(w,\kappa) \cdot u}{\|x_i^b \cdot u\| \vee \rho}\rho.$$

Recall that for a good batch $b \in G$, $y_i^b = w^* \cdot x_i^b + n_i^b$. Using this in equation (5.3), for any good batch $b \in G$, we have

$$\nabla f_i^b(w, \kappa) = \frac{(x_i^b \cdot (w - w^*) - n_i^b)}{|x_i^b \cdot (w - w^*) - n_i^b| \vee \kappa} \kappa x_i^b. \tag{5.37}$$

Combining the above two equations,

$$g_i^b(w, \kappa, u, \rho) = \kappa\rho \left(\frac{(x_i^b \cdot (w - w^*) - n_i^b)}{\|x_i^b \cdot (w - w^*) - n_i^b\| \vee \kappa} \right) \left(\frac{x_i^b \cdot u}{\|x_i^b \cdot u\| \vee \rho} \right). \tag{5.38}$$

From the above expression it follows that $|g_i^b(w, \kappa, u, \rho)| \leq \kappa\rho$ a.s.

We will choose ρ later in the proof. Let

$$Z_i^b(w, \kappa, u, \rho) := g_i^b(w, \kappa, u, \rho) - \mathbb{E}_{\mathcal{D}}\big[g_i^b(w, \kappa, u, \rho)\big].$$

and

$$\begin{aligned} \tilde{Z}_i^b(w, \kappa, u, \rho) &:= \nabla f_i^b(w, \kappa) \cdot u - \mathbb{E}_{\mathcal{D}}\big[\nabla f_i^b(w, \kappa) \cdot u\big] - Z_i^b(w, \kappa, u, \rho) \\ &= \nabla f_i^b(w, \kappa) \cdot u - g_i^b(w, \kappa, u, \rho) - \mathbb{E}_{\mathcal{D}}\big[\nabla f_i^b(w, \kappa) \cdot u - g_i^b(w, \kappa, u, \rho)\big]. \end{aligned}$$

When w, u, κ, and ρ are fixed or clear from the context, we will omit them from the notation of Z_i^b and $\tilde{Z}_i^b$. Then,

$$\begin{aligned} \mathbb{E}_G\Big[\big(\nabla f^b(w, \kappa) \cdot u - \mathbb{E}_{\mathcal{D}}[\nabla f^b(w, \kappa) \cdot u]\big)^2\Big] &= \frac{1}{|G|} \sum_{b \in G} \left(\frac{1}{n} \sum_{i \in n} (Z_i^b + \tilde{Z}_i^b) \right)^2 \\ &\leq \frac{2}{|G|} \sum_{b \in G} \left(\frac{1}{n} \sum_{i \in n} Z_i^b \right)^2 + \frac{2}{|G|} \sum_{b \in G} \left(\frac{1}{n} \sum_{i \in n} \tilde{Z}_i^b \right)^2 \\ &\leq \frac{2}{|G|} \sum_{b \in G} \left(\frac{1}{n} \sum_{i \in n} Z_i^b \right)^2 + \frac{2}{|G|} \sum_{b \in G} \frac{1}{n} \sum_{i \in n} (\tilde{Z}_i^b)^2, \end{aligned} \tag{5.39}$$

here in the last step we used Jensen's inequality $(\mathbb{E}[Z])^2 \le \mathbb{E}[Z^2]$.

We bound the two summations separately. To bound the first summation we first show that Z_i^b are bounded, and then use Bernstein's inequality. We bound the second term using Lemma 102 and Lemma 101.

From (5.38), it follows that $|g_i^b(w, \kappa, u, \rho)| \le \kappa\rho$ a.s., and therefore, $|Z_i^b| \le 2\kappa\rho$.

Since $|Z_i^b|$ is bounded by $2\kappa\rho$, it is a $(2\kappa\rho)^2$ sub-gaussian random variable. Using the fact that the sum of sub-gaussian random variables is sub-gaussian, the sum $\sum_{i=1}^n Z_i^b$ is $n(2\kappa\rho)^2$ sub-gaussian random variable. Since square of a sub-gaussian is sub-exponential [121] (Lemma 1.12), hence $(\sum_{i=1}^n Z_i^b)^2 - \mathbb{E}_{\mathcal{D}}(\sum_{i=1}^n Z_i^b)^2$ is sub-exponential with parameter $16n(2\kappa\rho)^2$.

Bernstein's inequality [121] (Theorem 1.12) for sub-Gaussian random variables implies that with probability $\ge 1 - \delta$,

$$\frac{1}{|G|}\sum_{b\in G}\left(\left(\sum_{i=1}^n Z_i^b\right)^2 - \mathbb{E}_{\mathcal{D}}\left[\left(\sum_{i=1}^n Z_i^b\right)^2\right]\right) \le 16n(2\kappa\rho)^2 \max\left\{\frac{2\ln(1/\delta)}{|G|}, \sqrt{\frac{2\ln(1/\delta)}{|G|}}\right\}.$$

Since Z_i^b are zero mean independent random variables,

$$\mathbb{E}_{\mathcal{D}}\left[\left(\sum_{i=1}^n (Z_i^b)\right)^2\right] = n\mathbb{E}_{\mathcal{D}}\left[(Z_i^b)^2\right].$$

We bound the expectation on the right,

$$\begin{aligned}
&\mathbb{E}_{\mathcal{D}}\big[(Z_i^b)^2\big] \\
&= \mathbb{E}_{\mathcal{D}}\Big[\big(g_i^b(w,\kappa,u,\rho) - \mathbb{E}_{\mathcal{D}}\big[g_i^b(w,\kappa,u,\rho)\big]\big)^2\Big] \\
&\overset{\text{(a)}}{\leq} \mathbb{E}_{\mathcal{D}}\Big[\big(g_i^b(w,\kappa,u,\rho)\big)^2\Big] \\
&\overset{\text{(b)}}{\leq} \mathbb{E}_{\mathcal{D}}[(n_i^b + (w-w^*)\cdot x_i^b)^2(x_i^b\cdot u)^2] \\
&\overset{\text{(c)}}{=} \mathbb{E}_{\mathcal{D}}[(n_i^b)^2(x_i^b\cdot u)^2] + \mathbb{E}_{\mathcal{D}}[((w-w^*)\cdot x_i^b)^2(x_i^b\cdot u)^2] \\
&\overset{\text{(d)}}{\leq} \mathbb{E}_{\mathcal{D}}[(n_i^b)^2]\mathbb{E}_{\mathcal{D}}[(x_i^b\cdot u)^2] + \sqrt{\mathbb{E}_{\mathcal{D}}[((w-w^*)\cdot x_i^b)^4]\mathbb{E}_{\mathcal{D}}[(u\cdot x_i^b)^4]} \\
&\overset{\text{(e)}}{\leq} \sigma^2\mathbb{E}_{\mathcal{D}}[(x_i^b\cdot u)^2] + \sqrt{C^2\mathbb{E}_{\mathcal{D}}[((w-w^*)\cdot x_i^b)^2]^2\mathbb{E}_{\mathcal{D}}[(u\cdot x_i^b)^2]^2} \\
&\overset{\text{(f)}}{\leq} \sigma^2 + C\mathbb{E}_{\mathcal{D}}[((w-w^*)\cdot x_i^b)^2],
\end{aligned}$$

here inequality (a) uses that squared deviation is smaller than mean squared deviation, inequality (b) follows from the definition of g_i^b in (5.38), inequality (c) follows from the independence of n_i^b and x_i^b, inequality (d) follows the Cauchy–Schwarz inequality, (e) uses the L-4 to L-2 hypercontractivity assumption $\mathbb{E}_{\mathcal{D}}[(u\cdot(x_i^b))^4] \leq C$, and (f) follows as for any unit vector $\mathbb{E}_{\mathcal{D}}[(x_i^b\cdot u)^2] \leq \|\Sigma\| \leq 1$.

Combining the last three equations, we get that with probability $\geq 1-\delta$,

$$\begin{aligned}
&\frac{1}{|G|}\sum_{b\in G}\left(\sum_{i=1}^n Z_i^b\right)^2 \\
&\leq n(\sigma^2 + C\mathbb{E}_{\mathcal{D}}[((w-w^*)\cdot x_i^b)^2]) + 64n(\kappa\rho)^2\max\left\{\frac{2\ln(1/\delta)}{|G|}, \sqrt{\frac{2\ln(1/\delta)}{|G|}}\right\}.
\end{aligned} \tag{5.40}$$

The above bound holds for given fixed values of parameters κ, w, and u. To extend the bound for all values of these parameters (for appropriate ranges of interest), we will use the covering argument.

With the help of the covering argument, we show that with probability

$\geq 1 - \delta e^{\mathcal{O}(d \log(C_1 dn)} - \frac{1}{d^2}$, for all unit vectors u, all vectors w and $\kappa \leq (\sigma + \|w - w^*\|)d^2 n$,

$$\begin{aligned}
&\frac{1}{|G|}\sum_{b\in G}\left(\sum_{i=1}^{n} Z_i^b(w,\kappa,u,\rho)\right)^2 \\
&\leq \frac{5}{2}\sigma^2 n + 13Cn\mathbb{E}_{\mathcal{D}}[((w-w^*)\cdot x_i^b)^2] + 384n(\kappa\rho)^2 \max\left\{\frac{2\ln(1/\delta)}{|G|}, \sqrt{\frac{2\ln(1/\delta)}{|G|}}\right\}. \quad (5.41)
\end{aligned}$$

We delegate the proof of Equation (5.41) using Equation (5.40) and the covering argument to the very end. The use of covering argument is rather standard. The main subtlety is that the above bound holds for all vectors w. The cover size of all d dimensional vectors is infinite. To overcome this difficulty we first take union bound for vectors for all w such that $\|w - w^*\| \leq R$ for an appropriate choice of R. To extend it to any w for which $\|w - w^*\| > R$ is large we show that the behavior of the above quantity on the left for such a w can be approximated by its behavior for $w' = w^* + (w - w^*)\frac{R}{\|w-w^*\|}$.

Note that dividing Equation (5.41) by n^2 bounds the first term in Equation (5.39). Next, we bound the second term in Equation(5.39). Note that

$$\begin{aligned}
&\frac{1}{n|G|}\sum_{b\in G}\sum_{i\in n}(\tilde{Z}_i^b)^2 \\
&\leq \frac{1}{n|G|}\sum_{b\in G}\sum_{i\in n}\big(\nabla f_i^b(w,\kappa)\cdot u - g_i^b(w,\kappa,u,\rho) - \mathbb{E}_{\mathcal{D}}\big[\nabla f_i^b(w,\kappa)\cdot u - g_i^b(w,\kappa,u,\rho)\big]\big)^2 \\
&\leq \frac{2}{n|G|}\sum_{b\in G}\sum_{i\in n}\Big(\big(\nabla f_i^b(w,\kappa)\cdot u - g_i^b(w,\kappa,u,\rho)\big)^2 + \big(\mathbb{E}_{\mathcal{D}}\big[\nabla f_i^b(w,\kappa)\cdot u - g_i^b(w,\kappa,u,\rho)\big]\big)^2\Big) \\
&\leq \frac{2}{n|G|}\sum_{b\in G}\sum_{i\in n}\Big(\big(\nabla f_i^b(w,\kappa)\cdot u - g_i^b(w,\kappa,u,\rho)\big)^2 + \mathbb{E}_{\mathcal{D}}\Big[\big(\nabla f_i^b(w,\kappa)\cdot u - g_i^b(w,\kappa,u,\rho)\big)^2\Big]\Big).
\end{aligned}$$

From the definitions of $g_i^b(w,\kappa,u,\rho)$ and $\nabla f_i^b(w,\kappa)$,

$$\begin{aligned}|\nabla f_i^b(w,\kappa)\cdot u - g_i^b(w,\kappa,u,\rho)| &= \mathbb{1}(\|x_i^b\cdot u\| \geq \rho)\left|\nabla f_i^b(w,\kappa)\cdot u - \frac{\rho}{\|x_i^b\cdot u\|}\nabla f_i^b(w,\kappa)\cdot u\right| \\ &\leq \mathbb{1}(\|x_i^b\cdot u\| \geq \rho)\big|\nabla f_i^b(w,\kappa)\cdot u\big| \\ &\leq \kappa\big|x_i^b\cdot u\big|\cdot\mathbb{1}(\|x_i^b\cdot u\| \geq \rho).\end{aligned}$$

From the above equation, it follows that

$$\mathbb{E}_{\mathcal{D}}\Big[\big(\nabla f_i^b(w,\kappa)\cdot u - g_i^b(w,\kappa,u,\rho)\big)^2\Big] \leq \kappa^2\mathbb{E}_{\mathcal{D}}\Big[\mathbb{1}(\|x_i^b\cdot u\| \geq \rho)\big|x_i^b\cdot u\big|^2\Big].$$

Combining the above three bounds,

$$\begin{aligned}&\frac{1}{n|G|}\sum_{b\in G}\sum_{i\in n}(\tilde{Z}_i^b)^2 \\ &\leq \frac{2\kappa^2}{n|G|}\sum_{b\in G}\sum_{i\in n}\Big(\mathbb{1}(\|x_i^b\cdot u\| \geq \rho)\big|x_i^b\cdot u\big|^2 + \mathbb{E}_{\mathcal{D}}\Big[\mathbb{1}(\|x_i^b\cdot u\| \geq \rho)\big|x_i^b\cdot u\big|^2\Big]\Big) \\ &= 2\kappa^2\Big(\mathbb{E}_S\Big[\mathbb{1}(\|x_i^b\cdot u\| \geq \rho)\big|x_i^b\cdot u\big|^2\Big] + \mathbb{E}_{\mathcal{D}}\Big[\mathbb{1}(\|x_i^b\cdot u\| \geq \rho)\big|x_i^b\cdot u\big|^2\Big]\Big),\end{aligned}$$

here the last line uses the fact that S is the collection of all good samples.

For $\rho^2 \geq 3\sqrt{C}$, and $|G|n = \Omega(d\rho^4\log(\frac{C_1 d\rho}{C}))$, Lemma 102 implies that with probability at least $1-2/d^2$, for all unit vectors u, we have

$$\mathbb{E}_S\Big[\mathbb{1}(\|x_i^b\cdot u\| \geq \rho)\big|x_i^b\cdot u\big|^2\Big] = \mathbb{E}_S\Big[\mathbb{1}(\|x_i^b\cdot u\|^2 \geq \rho^2)\big|x_i^b\cdot u\big|^2\Big] \leq \mathcal{O}(\sqrt{C}/\rho^2).$$

And from Lemma 101, for $\rho^2 \geq \sqrt{C}$ and any unit vectors u,

$$\mathbb{E}_{\mathcal{D}}\Big[\mathbb{1}(\|x_i^b\cdot u\| \geq \rho)\big|x_i^b\cdot u\big|^2\Big] \leq \mathcal{O}(\sqrt{C}/\rho^2).$$

By combining the above three bounds it follows that, if $\rho^2 \geq \sqrt{C}$, and $|G|n =$

$\Omega(d\rho^4 \log(\frac{C_1 d\rho}{C}))$, with probability at least $1 - 2/d^2$, for all unit vectors u,

$$\frac{1}{n|G|}\sum_{b\in G}\sum_{i\in n}(\tilde{Z}_i^b(w,\kappa,u,\beta))^2 \le \mathcal{O}\left(\frac{\sqrt{C}\kappa^2}{\rho^2}\right).$$

Combining the above bound, Equation (5.41) and (5.39) we get that if $\rho^2 = \Omega(\sqrt{C})$, and $|G| = \Omega(\frac{d\rho^4}{n}\log(\frac{C_1 d\rho}{C}))$ then with probability $\ge 1 - \delta e^{\mathcal{O}(d\log(C_1 dn)} - \frac{3}{d^2}$, for all unit vectors u, all vectors w and $\kappa \le (\sigma + \|w - w^*\|)d^2 n$,

$$\begin{aligned}&\mathbb{E}_G\Big[\big(\nabla f^b(w,\kappa)\cdot u - \mathbb{E}_{\mathcal{D}}[\nabla f^b(w,\kappa)\cdot u]\big)^2\Big]\\ &\le \frac{2}{n^2}\left(\frac{5}{2}\sigma^2 n + 13Cn\mathbb{E}_{\mathcal{D}}[((w-w^*)\cdot x_i^b)^2] + 384n(\kappa\rho)^2\max\left\{\frac{2\ln(1/\delta)}{|G|}, \sqrt{\frac{2\ln(1/\delta)}{|G|}}\right\}\right)\\ &\quad + \mathcal{O}\left(\frac{\sqrt{C}\kappa^2}{\rho^2}\right).\end{aligned}$$

Recall that $1 \le \mu_{\max} \le \frac{d^4 n^2}{C}$. Choose $\rho^2 = \mu_{\max}\sqrt{C}n$. Note that $\sqrt{\mu_{\max}(\sigma^2 + C\mathbb{E}_{\mathcal{D}}[((w-w^*)\cdot x_i^b)^2])} \le (\sigma + \|w - w^*\|)d^2 n$. Then from the above equation choosing $\rho^2 = \mu_{\max}\sqrt{C}n$, for all

$$\kappa \le \sqrt{\mu_{\max}(\sigma^2 + C\mathbb{E}_{\mathcal{D}}[((w-w^*)\cdot x_i^b)^2])},$$

with probability $\ge 1 - \delta e^{\mathcal{O}(d\log(C_1 dn)} - \frac{3}{d^2}$, for all unit vectors u, all vectors w ,

$$\begin{aligned}&\mathbb{E}_G\Big[\big(\nabla f^b(w,\kappa)\cdot u - \mathbb{E}_{\mathcal{D}}[\nabla f^b(w,\kappa)\cdot u]\big)^2\Big]\\ &\le \mathcal{O}\left(\frac{\sigma^2 + C\mathbb{E}_{\mathcal{D}}[((w-w^*)\cdot x_i^b)^2]}{n}\right)\left(1 + n\mu_{\max}^2\sqrt{C}\max\left\{\frac{2\ln(1/\delta)}{|G|}, \sqrt{\frac{2\ln(1/\delta)}{|G|}}\right\}\right).\end{aligned}$$

Choose $\delta = e^{-\Theta(d\log(C_1 dn)}$, and $|G| = \Omega(\frac{d\rho^4}{n}\log(\frac{C_1 d\rho}{C}) + C\mu_{\max}^4 dn^2\log(C_1 dn)) = \Omega(\rho_{\max}^4 n^2 d\log(d))$. Then with probability $\ge 1 - \frac{4}{d^2}$, for all unit vectors u, all vectors w and for all $\kappa^2 \leq \mu_{\max}(\sigma^2 + C\mathbb{E}_{\mathcal{D}}[((w-w^*)\cdot x_i^b)^2])$,

$$\mathbb{E}_G\Big[\big(\nabla f^b(w,\kappa)\cdot u - \mathbb{E}_{\mathcal{D}}[\nabla f^b(w,\kappa)\cdot u]\big)^2\Big] \leq \mathcal{O}\bigg(\frac{\sigma^2 + C\mathbb{E}_{\mathcal{D}}[((w-w^*)\cdot x_i^b)^2]}{n}\bigg),$$

which is the desired bound.

We complete the proof by proving Equation (5.41).

Proof of Equation (5.41)

To complete the proof of the theorem next we prove Equation (5.41) with the help of Equation (5.40) and covering argument. To use the covering argument, we first show that $g_i^b(w,\kappa,u,\rho)$ do not change by much by slight deviation of these parameters. From the definition of $Z_i^b(w,\kappa,u,\rho)$, the same conclusion would then hold for it.

By the triangle inequality,

$$\begin{aligned}
&|g_i^b(w,\kappa,u,\rho) - g_i^b(w',\kappa',u',\rho)|\\
&\leq |g_i^b(w',\kappa',u,\rho) - g_i^b(w',\kappa',u',\rho)| + |g_i^b(w,\kappa',u,\rho) - g_i^b(w',\kappa',u,\rho)|\\
&\quad + |g_i^b(w,\kappa,u,\rho) - g_i^b(w,\kappa',u,\rho)|.
\end{aligned}$$

We bound each term on the right one by one. To bound these terms we use Equation (5.38), the assumption that $\|x_i^b\| \leq C_1\sqrt{d}$ and the definition of the function $g()$. For the first term,

$$|g_i^b(w',\kappa',u,\rho) - g_i^b(w',\kappa',u',\rho)| \leq \|(u-u')x_i^b\|\kappa' \leq C_1\|u-u'\|\sqrt{d}\kappa',$$

for the second term,

$$|g_i^b(w,\kappa',u,\rho) - g_i^b(w',\kappa',u,\rho)| \leq |u\cdot x_i^b|\cdot|(w-w')\cdot x_i^b| \leq \|x_i^b\|^2\cdot\|w-w'\| \leq C_1^2 d\|w-w'\|,$$

and for the last term

$$|g_i^b(w,\kappa,u,\rho) - g_i^b(w,\kappa',u,\rho)| \le |\kappa-\kappa'| \cdot |u \cdot x_i^b| \le C_1\sqrt{d}|\kappa-\kappa'|.$$

Combining the three bounds,

$$|g_i^b(w,\kappa,u,\rho) - g_i^b(w',\kappa',u',\rho)| \le C_1\|u-u'\|\sqrt{d}\kappa' + C_1^2 d\|w-w'\| + C_1\sqrt{d}|\kappa-\kappa'|.$$

For $\|u-u'\| \le 1/(24C_1d^5n^3)$, $\kappa' \le 2d^4\sigma n^2$, $\|w-w'\| \le \sigma/(12dC_1^2n)$ and $|\kappa-\kappa'| \le \sigma/(12C_1dn)$,

$$|g_i^b(w,\kappa,u,\rho) - g_i^b(w',\kappa',u',\rho)| \le \sigma/4n \text{ a.s.}$$

This would imply,

$$|Z_i^b(w,\kappa,u,\rho) - Z_i^b(w',\kappa',u',\rho)| \le \sigma/2n \text{ a.s.}$$

Using this bound,

$$\begin{aligned}
\frac{1}{|G|}\sum_{b\in G}\left(\sum_{i=1}^n Z_i^b(w,\kappa,u,\rho)\right)^2 &\le \frac{1}{|G|}\sum_{b\in G}\left(\sum_{i=1}^n\Big(Z_i^b(w',\kappa',u',\rho') + \frac{\sigma}{2n}\Big)\right)^2 \\
&\le \frac{2}{|G|}\sum_{b\in G}\left(\sum_{i=1}^n Z_i^b(w',\kappa',u',\rho')\right)^2 + \frac{2}{|G|}\sum_{b\in G}\left(\sum_{i=1}^n \frac{\sigma}{2n}\right)^2 \\
&\le \frac{2}{|G|}\sum_{b\in G}\left(\sum_{i=1}^n Z_i^b(w',\kappa',u',\rho')\right)^2 + \frac{\sigma^2}{2}. \qquad (5.42)
\end{aligned}$$

Let $\mathcal{U} := \{u \in \mathbb{R}^d : \|u\| = 1\}$, $\mathcal{W} := \{w \in \mathbb{R}^d : \|w - w^*\| \le d^2\sigma n)\}$, and $\mathcal{K} := [0, 2d^4\sigma n^2]$.

Standard covering argument shows that there exist covers such that

$$\mathcal{U}' \subseteq \mathcal{U} : \forall u \in \mathcal{U}, \min_{u' \in \mathcal{U}'} \|u - u'\| \leq \frac{1}{(24 C_1 d^5 n^3)}, \tag{5.43}$$

$$\mathcal{W}' \subseteq \mathcal{W} : \forall w \in \mathcal{W}, \min_{w' \in \mathcal{W}'} \|w - w'\| \leq \frac{\sigma}{12 C_1^2 dn}, \tag{5.44}$$

and

$$\mathcal{K}' \subseteq \mathcal{K} : \forall \kappa \in \mathcal{K}, \min_{\kappa' \in \mathcal{K}', \kappa' \geq \kappa} |\kappa - \kappa'| \leq \frac{\sigma}{12 C_1 dn}, \tag{5.45}$$

and the size of each is $|\mathcal{U}'|, |\mathcal{W}'|, |\mathcal{K}'| \leq e^{\mathcal{O}(d \log(C_1 dn)}$.

In equation (5.40), taking the union bound over all elements in $\mathcal{U}'$, $\mathcal{W}'$ and $\mathcal{K}'$, it follows that with probability $\geq 1 - \delta e^{\mathcal{O}(d \log(C_1 dn)}$, for all $u' \in \mathcal{U}'$, $w' \in \mathcal{W}'$ and $\kappa' \in \mathcal{K}'$

$$\begin{aligned}
&\frac{1}{|G|} \sum_{b \in G} \left(\sum_{i=1}^{n} Z_i^b(w', \kappa', u', \rho) \right)^2 \\
&\leq n(\sigma^2 + C\mathbb{E}_{\mathcal{D}}[((w' - w^*) \cdot x_i^b)^2]) + 64n(\kappa'\rho)^2 \max\left\{ \frac{2\ln(1/\delta)}{|G|}, \sqrt{\frac{2\ln(1/\delta)}{|G|}} \right\}.
\end{aligned}$$

Combining the above bound with Equation (5.42), it follows that with probability $\geq 1 - \delta e^{\mathcal{O}(d \log(C_1 dn)}$, for all $u \in \mathcal{U}$, $w \in \mathcal{W}$ and $\kappa \in \mathcal{K}$ and elements u', w' and κ' in the respective nets satisfying equations (5.43),(5.44), and (5.45),

$$
\begin{aligned}
&\frac{1}{|G|}\sum_{b\in G}\left(\sum_{i=1}^{n} Z_i^b(w,\kappa,u,\rho)\right)^2 \\
&\leq 2n(\sigma^2 + C\mathbb{E}_{\mathcal{D}}[((w'-w^*)\cdot x_i^b)^2]) + 128n(\kappa'\rho)^2 \max\left\{\frac{2\ln(1/\delta)}{|G|}, \sqrt{\frac{2\ln(1/\delta)}{|G|}}\right\} + \frac{\sigma^2}{2} \\
&\leq 2n(\sigma^2(1+\frac{1}{4n}) + C\mathbb{E}_{\mathcal{D}}[((w'-w^*)\cdot x_i^b)^2]) + 128n(\kappa\rho)^2 \max\left\{\frac{2\ln(1/\delta)}{|G|}, \sqrt{\frac{2\ln(1/\delta)}{|G|}}\right\} \\
&\leq 2n(\frac{5}{4}\sigma^2 + 2C\mathbb{E}_{\mathcal{D}}[((w-w^*)\cdot x_i^b)^2]) + 128n(\kappa\rho)^2 \max\left\{\frac{2\ln(1/\delta)}{|G|}, \sqrt{\frac{2\ln(1/\delta)}{|G|}}\right\}. \quad (5.46)
\end{aligned}
$$

here (a) follows from the bound $\kappa \geq \kappa'$ in Equation (5.45), and (b) follows by first writing $w' - w^* = (w - w^*) + (w' - w)$ and then using the bound $\|w' - w\| \leq \frac{\sigma}{12C_1^2 dn}$ in Equation (5.44).

Next, we further remove the restriction $w \in \mathcal{W}$ and extend the above bound to all vectors w.

Consider a $w \notin \mathcal{W}$ and $\kappa \in [0, (\sigma + \|w - w^*\|)d^2 n]$. From the definition of $\mathcal{W}$, we have $\|w - w^*\| > d^2\sigma n$. Let $w' = w^* + \frac{w-w^*}{\|w-w^*\|}d^2\sigma n$ and $\kappa' = \frac{d^2\sigma n}{\|w-w^*\|}\kappa$. Observe that $\|w' - w\| = d^2\sigma n$ and

$$
\kappa' \leq (\sigma + \|w - w^*\|)d^2 n\frac{d^2\sigma n}{\|w-w^*\|} \leq d^2\sigma n\frac{d^2\sigma n}{\|w-w^*\|} + d^4\sigma n^2 \leq d^2\sigma n + d^4\sigma n^2 \leq 2d^4\sigma n^2,
$$

hence, $w' \in \mathcal{W}$ and $\kappa' \in \mathcal{K}$. From Equation (5.38),

$$\begin{aligned}
&\left|\frac{\|w-w^*\|}{d^2\sigma n} g_i^b(w',\kappa',u,\rho) - g_i^b(w,\kappa,u,\rho)\right| \\
&\overset{\text{(a)}}{=} \frac{\rho\|x_i^b \cdot u\|}{\|x_i^b \cdot u\| \vee \rho}\left|\frac{\|w-w^*\|}{d^2\sigma n}\kappa'\left(\frac{(x_i^b \cdot (w'-w^*) - n_i^b)}{\|x_i^b \cdot (w'-w^*) - n_i^b\| \vee \kappa'}\right) - \left(\frac{(\kappa x_i^b \cdot (w-w^*) - n_i^b)}{\|x_i^b \cdot (w-w^*) - n_i^b\| \vee \kappa}\right)\right| \\
&\overset{\text{(b)}}{\leq} \|x_i^b \cdot u\| \cdot \left|\kappa\left(\frac{(x_i^b \cdot \frac{w-w^*}{\|w-w^*\|} d^2\sigma n - n_i^b)}{\|x_i^b \cdot \frac{w-w^*}{\|w-w^*\|} d^2\sigma n - n_i^b\| \vee \frac{d^2\sigma n}{\|w-w^*\|}\kappa}\right) - \kappa\left(\frac{(x_i^b \cdot (w-w^*) - n_i^b)}{\|x_i^b \cdot (w-w^*) - n_i^b\| \vee \kappa}\right)\right| \\
&= \|x_i^b \cdot u\| \cdot \left|\kappa\left(\frac{(x_i^b \cdot (w-w^*) - \frac{d^2\sigma n}{\|w-w^*\|} n_i^b)}{\|x_i^b \cdot (w-w^*) - \frac{d^2\sigma n}{\|w-w^*\|} n_i^b\| \vee \kappa}\right) - \kappa\left(\frac{(x_i^b \cdot (w-w^*) - n_i^b)}{\|x_i^b \cdot (w-w^*) - n_i^b\| \vee \kappa}\right)\right| \\
&\overset{\text{(c)}}{\leq} \|x_i^b \cdot u\| \cdot \left|\frac{d^2\sigma n}{\|w-w^*\|} n_i^b - n_i^b\right| \\
&\overset{\text{(d)}}{\leq} C_1\sqrt{d}|n_i^b| \frac{d^2\sigma n}{\|w-w^*\|} \\
&\overset{\text{(e)}}{\leq} C_1\sqrt{d}|n_i^b|,
\end{aligned}$$

here (a) follows from the definition of g_i^b, inequality (b) follows as $\rho \leq \|x_i^b \cdot u\| \vee \rho$, inequality (c) uses the fact that for any a, Δ and $b \geq 0$, we have $|b\frac{a+\Delta}{(a+\Delta)\vee b} - b\frac{a}{a \vee b}| \leq |\Delta|$, inequality (c) uses $\|x_i^b\| \leq C_1\sqrt{d}$ and the last inequality (e) uses $\|w-w^*\| > d^2\sigma n$.

Therefore,

$$\left|\frac{\|w-w^*\|}{d^2\sigma n} Z_i^b(w',\kappa',u,\rho) - Z_i^b(w,\kappa,u,\rho)\right| \leq C_1\sqrt{d}\big(|n_i^b| + \mathbb{E}[|n_i^b|]\big) \leq C_1\sqrt{d}\big(|n_i^b| + \sigma\big).$$

From the above equation,

$$
\begin{aligned}
&\frac{1}{|G|}\sum_{b\in G}\left(\sum_{i=1}^{n} Z_i^b(w,\kappa,u,\rho)\right)^2 \\
&\leq \frac{1}{|G|}\sum_{b\in G}\left(\sum_{i=1}^{n}\left(\frac{\|w-w^*\|}{d^2\sigma n} Z_i^b(w',\kappa',u,\rho) + C_1\sqrt{d}|n_i^b| + C_1\sqrt{d}\sigma\right)\right)^2 \\
&\overset{\text{(a)}}{\leq} \frac{1}{|G|}\sum_{b\in G} 3\left(\left(\sum_{i=1}^{n}\frac{\|w-w^*\|}{d^2\sigma n} Z_i^b(w',\kappa',u,\rho)\right)^2 + \left(\sum_{i=1}^{n} C_1\sqrt{d}|n_i^b|\right)^2 + \left(\sum_{i=1}^{n} C_1\sqrt{d}\sigma\right)^2\right) \\
&\overset{\text{(b)}}{\leq} \frac{3\|w-w^*\|^2}{d^4\sigma^2 n^2}\frac{1}{|G|}\sum_{b\in G}\left(\sum_{i=1}^{n} Z_i^b(w',\kappa',u,\rho)\right)^2 + \frac{3dC_1^2}{|G|}\sum_{b\in G}\left(n\sum_{i=1}^{n}|n_i^b|^2\right) + 3dC_1^2 n^2\sigma^2 \\
&\overset{\text{(c)}}{\leq} \frac{3\|w-w^*\|^2}{d^4\sigma^2 n^2}\frac{1}{|G|}\sum_{b\in G}\left(\sum_{i=1}^{n} Z_i^b(w',\kappa',u,\rho)\right)^2 + 3dC_1^2 n^2\sigma^2(d^2+1),
\end{aligned}
$$

here (a) and (b) uses $(\sum_{i=1}^{t} z_i) \leq t\sum_{i=1}^{t} z_i^2$ and inequality (c) holds with with probability $\geq 1-\frac{1}{d^2}$ by Markov inequality, as $Pr[\frac{1}{n|G|}\sum_{b\in G}\sum_{i=1}^{n}|n_i^b|^2 > d^2\mathbb{E}_{\mathcal{D}}[(n_i^b)^2)] \leq \frac{1}{d^2}$.

Recall that $\|w'-w\| \leq d^2\sigma n$ and $\kappa \leq 2d^4\sigma n^2$, therefore in the above equation, we can bound the first term on the right by using high probability bound in Equation (5.46). Then, with probability $\geq 1 - \delta e^{\mathcal{O}(d\log(C_1 dn)} - \frac{1}{d^2}$,

$$\begin{aligned}
&\frac{1}{|G|}\sum_{b\in G}\left(\sum_{i=1}^{n} Z_i^b(w,\kappa,u,\rho)\right)^2 \\
&\leq \frac{3\|w-w^*\|^2}{d^4\sigma^2 n^2}\Bigg(2n(\frac{5}{4}\sigma^2 + 2C\mathbb{E}_{\mathcal{D}}[((w'-w^*)\cdot x_i^b)^2]) \\
&\quad + 128n(\kappa'\rho)^2\max\left\{\frac{2\ln(1/\delta)}{|G|},\sqrt{\frac{2\ln(1/\delta)}{|G|}}\right\}\Bigg) + 3dC_1^2 n^2\sigma^2(d^2+1) \\
&\overset{\text{(a)}}{=} 12Cn\mathbb{E}_{\mathcal{D}}[((w-w^*)\cdot x_i^b)^2] + \frac{15\|w-w^*\|^2}{2d^4\sigma^2 n^2}n\sigma^2 \\
&\quad + 384n(\kappa\rho)^2\max\left\{\frac{2\ln(1/\delta)}{|G|},\sqrt{\frac{2\ln(1/\delta)}{|G|}}\right\} + 3dC_1^2 n^2\sigma^2(d^2+1) \\
&\overset{\text{(b)}}{\leq} 13Cn\mathbb{E}_{\mathcal{D}}[((w-w^*)\cdot x_i^b)^2] + 384n(\kappa\rho)^2\max\left\{\frac{2\ln(1/\delta)}{|G|},\sqrt{\frac{2\ln(1/\delta)}{|G|}}\right\},
\end{aligned}$$

here equality (a) uses the relation $w'-w^* = \frac{w-w^*}{\|w-w^*\|}d^2\sigma n$ and $\kappa' = \frac{d^2\sigma n}{\|w-w^*\|}\kappa$, and (b) follows as $C\mathbb{E}_{\mathcal{D}}[((w-w^*)\cdot x_i^b)^2] \geq C\frac{\|w-w^*\|^2\|\Sigma\|}{C_3} = C\frac{\|w-w^*\|^2}{C_3} \geq Cd^4\sigma^2n^2/C_3$, where C_3 is condition number of Σ, hence $C\mathbb{E}_{\mathcal{D}}[((w-w^*)\cdot x_i^b)^2] \gg \frac{15\|w-w^*\|^2}{2d^4\sigma^2n^2}n\sigma^2$ and $C\mathbb{E}_{\mathcal{D}}[((w-w^*)\cdot x_i^b)^2] \geq C\frac{\|w-w^*\|^2}{C_3} \gg \frac{15\|w-w^*\|^2}{2d^4\sigma^2n^2}n\sigma^2$ and $C\mathbb{E}_{\mathcal{D}}[((w-w^*)\cdot x_i^b)^2] \gg 3dC_1^2n^2\sigma^2(d^2+1)$.

The above bound holds for all unit vectors u, $w \notin \mathcal{W}$ and $\kappa \leq (\sigma + \|w-w^*\|)d^2n$,

$$\begin{aligned}
&\frac{1}{|G|}\sum_{b\in G}\left(\sum_{i=1}^{n} Z_i^b(w,\kappa,u,\rho)\right)^2 \\
&\leq 13Cn\mathbb{E}_{\mathcal{D}}[((w-w^*)\cdot x_i^b)^2] + 450n(\kappa\rho)^2\max\left\{\frac{2\ln(1/\delta)}{|G|},\sqrt{\frac{2\ln(1/\delta)}{|G|}}\right\}.
\end{aligned}$$

Recall that bound in Equation (5.46) holds for all unit vectors u, $w \in \mathcal{W}$ and $\kappa \leq \mathcal{K}'$ with probability $\geq 1 - \delta e^{\mathcal{O}(d\log(C_1dn)}$. Note that for $w \in \mathcal{W}$, $(\sigma + \|w-w^*\|)d^2n \leq 2d^4\sigma n^2$, hence $[0, (\sigma + \|w-w^*\|)d^2n] \subseteq \mathcal{K}'$. Hence the above bound holds for all unit vectors u, $w \in \mathcal{W}$ and $\kappa \leq (\sigma + \|w-w^*\|)d^2n$. Combining the two bounds, with probability $\geq 1 - \delta e^{\mathcal{O}(d\log(C_1dn)} - \frac{1}{d^2}$, for all unit vectors u, all vectors w and $\kappa \leq (\sigma + \|w - w^*\|)d^2 n$,

$$\frac{1}{|G|}\sum_{b\in G}\left(\sum_{i=1}^{n} Z_i^b(w,\kappa,u,\rho)\right)^2$$
$$\leq \frac{5}{2}\sigma^2 n + 13Cn\mathbb{E}_{\mathcal{D}}[((w-w^*)\cdot x_i^b)^2] + 384n(\kappa\rho)^2 \max\left\{\frac{2\ln(1/\delta)}{|G|}, \sqrt{\frac{2\ln(1/\delta)}{|G|}}\right\}.$$

This completes the proof of Equation (5.41). ■

Chapter 5, in full, is a reprint of the material as it appears in Efficient list-decodable regression using batches 2023. Abhimanyu Das, Ayush Jain, Weihao Kong, and Rajat Sen. In ICML 2023. The dissertation author was the primary investigator and author of this paper.

Chapter 6

Linear Regression using Heterogeneous Data Batches

6.1 Introduction

In numerous applications, including federated learning [147], sensor networks [149], crowd-sourcing [136] and recommendation systems [148], data are collected from multiple sources, each providing a *batch* of samples. For instance, in movie recommendation systems, users typically rate multiple films. Since all samples in a batch are generated by the same source, they are often assumed to share the same underlying distribution. However, the batches are frequently very small, e.g., many users provide only few ratings. Hence, it may be impossible to learn a different model for each batch.

A common approach has therefore assumed [140] that all batches share the same underlying distribution and learn this common model by pooling together the data from all batches. While this may work well for some applications, in others, it may fail, or lack personalization. For instance, in recommendation systems, it may not capture the characteristics of individual users.

A promising alternative that allows for personalization even with many small batches assumes that batches can be categorized into k sub-populations. In each sub-population, all batches are generated by distributions that are close to each other and hence can be represented by a single distribution. Henceforth, we identify the sub-population with this common distribution. Even when k is large, our work allows the recovery of models for sub-populations with a significant fraction of batches. For example, in the recommendation setting, most users can be classified into a few sub-populations such that the distribution of users in the sub-population is close, for instance, those preferring certain genres.

This paper focuses on the canonical model of linear regression in supervised learning. A distribution $\mathcal{D}$ of samples (x, y) follows a linear regression model if, for some regression vector $w \in \mathbb{R}^d$, the output is $y = w \cdot x + \eta$ where input x is a random d dimensional vector and η is a zero-mean noise. The goal is to recover the regression vectors for all large sub-populations that follows the linear regression model.

6.1.1 Our Results

This setting was first considered in [95] for meta-learning applications, where they view and term batches as *tasks*. [95] argue that in meta-learning applications task or batch lengths follow a long tail distribution and in the majority of the batches only a few labeled examples are available. Only a few batches have medium size labeled samples available, and almost all of them have length $\ll d$. Note that similar observations have been made in the recommender system literature where the distribution of a number of ratings per user follows a long-tailed distribution with an overwhelming number of users rating only a few items while rare tail users rating hundreds of items [66]. The same has been observed for the distribution of the number of ratings per item [118]. Therefore, it is reasonable to assume that in these applications of interest, a handful of medium-size batches along with a large number, $\Omega(d)$, batches of constant size are available. Under this setting our main results allow recovery of all sub-populations that has a significant fraction of batches and follow a linear regression model:

Let $k \in \mathbb{N}$ be the number of distinct sub-populations. For $\alpha > 0$, let I be the collection of all sub-populations that contribute at least α fraction of the batches and satisfy a linear regression model with an output-noise variance $\leq \sigma^2$. For $i \in I$, let w_i be the regression parameter of sub-population i. Our goal is to estimate w_i's.

Theorem 105 (Informal)**.** *Given $\tilde{\Omega}(d/\alpha^2)$ small batches of size ≥ 2, and $\tilde{\Omega}(\min(\sqrt{k}, 1/\sqrt{\alpha})/\alpha)$ medium batches of size $\geq \tilde{\Omega}(\min(\sqrt{k}, 1/\sqrt{\alpha}))$, our algorithm runs in time $poly(d, 1/\alpha, k)$ and outputs a list L of size $\tilde{O}(1/\alpha)$ such that w.h.p., for each sub-population $i \in I$, there is at least one estimate in L that is within a distance of $\mathcal{O}(\sigma)$ from w_i and has an expected prediction error $\sigma^2(1 + o(1))$ for the sub-population i. Furthermore, given $\Omega(\log L)$ samples from the sub-population i, we can identify such an estimate from L.*

Note that to recover regression vectors for all sub-populations I, our algorithm only requires $\tilde{\Omega}(d/\alpha + \min(k, 1/\alpha))$ samples from each sub-population and $\tilde{\Omega}(d/\alpha^2 + \min(k, 1/\alpha)/\alpha)$ samples in total. Note that $\Omega(d)$ samples are required by any algorithm even when $k = 1$. To the best of our knowledge, ours is the best sample complexity for recovering the linear regression models in the presence of multiple sub-populations using batch sizes smaller than d.

6.1.2 Comparison to Prior Work

The only work that provides a polynomial time algorithm in dimension, in the same generality as ours is [41]. They even allow the presence of adversarial batches. However, they require $\tilde{\Omega}(d/\alpha^2)$ batches from the sub-population of size $\tilde{\Omega}(1/\alpha)$ each, and therefore, $\tilde{\Omega}(d/\alpha^3)$ samples in total, which exceeds our sample complexity by a factor of $1/\alpha^2$. Note that the batch length in their setting is at least quadratically larger than ours. All other works place strong assumptions on the distributions of the sub-population and still require a number of samples much larger than ours, which we discuss next.

Most of the previous works [26, 134, 154, 153, 102, 32, 55, 117] have addressed the widely studied *mixed linear regression (MLR)* model where all batches are of size 1, and adhere to the following three assumptions:

1. All k sub-populations have $\geq \alpha$ fraction of data. This assumption implies $k \leq 1/\alpha$.

2. All k distributions follow a linear regression model.

3. All k regression coefficients are well separated, namely $\|w_i - w_j\| \geq \Delta, \forall\, i \neq j$.

Even for $k = 2$, solving MLR, in general, is NP-hard [152]. Hence all these works on mixed linear regression, except [102], also made the following assumption:

4. All input distributions (i.e., the distribution over x) are the same for every sub-population, in fact, the same isotropic Gaussian distribution. This implies the distribution of movies that users rate is the same across every user.

With this additional isotropic Gaussian assumption, they provided algorithms that have runtime and sample complexity polynomial in the dimension. However, even with these four strong assumptions, their sample complexity is super-polynomial overall. In particular, the sample complexity in [154, 32, 55] is quasi-polynomial in k and [26, 134, 55] require at least a quadratic scaling in d. In [26, 134, 153] the sample complexity scales as a large negative power of the minimum singular value of certain moment matrix of regression vectors that can be zero even when the gap between the regression vectors is large. In addition, [154, 153, 32] required zero-noise i.e $\eta = 0$. The only work we are aware of that can avoid Assumption 4 and handle different input distributions for different sub-populations under MLR is [102]. However, they still require all distributions to be Gaussian and $\eta = 0$, and their sample size, and hence run-time is exponential, $\Omega(\exp(k^2))$ in k.

The work that most closely relates to ours is [95], which considers batch sizes > 1. While it achieves the same dependence as us on d, k, and $1/\alpha$, on the length and number of medium and small batches, the sample complexity of the algorithms and the length of medium-size batches had an additional multiplicative dependence on the inverse separation parameter $1/\Delta$. It also required Assumption 4 mentioned in the section. The follow-up work [94] which still assumes all four assumptions can handle the presence of a small fraction $\ll 1/k^2\alpha^2$ of adversarial batches, but requires $\tilde{\Omega}(dk^2/\alpha^2 + k^5/\alpha^4)$ samples. It also suffers from similar strong assumptions as earlier works and the sum of squares approach makes it impractical. The sum of the square approach, and stronger isotropic Gaussian assumption, allow it to achieve a better dependence on $1/\alpha$ on medium-size batch lengths, however, causing a significant increase in the number of medium-size batches required.

Our improvement over prior work. In contrast, our work avoids all four assumptions, and can recover any sufficiently large sub-populations that follow a linear regression model. In particular: (1) Even when a large number of different sub-populations are present, (e.g., $k \geq 1/\alpha$), we can still recover the regression coefficient of a sub-population with sufficient fraction of batches. (2) The k distributions do not even need to follow a linear regression model. In particular, our algorithm is robust to the presence of sub-populations for which the conditional distribution of output given input is arbitrary. (3) Our work requires no assumption on the separation of regression coefficient Δ, and our guarantees as well have no dependence on the separation. (4) We allow different input distributions for different sub-populations. (5) In addition to removing the four assumptions, the algorithm doesn't require all batches in a sub-population to have identical distributions, it only requires them to be close so that the expected value of gradient for a batch is close to one of the sub-population.

6.1.3 Techniques and Organization

We sample a medium-size batch randomly and recover the regression vector of the population that the sampled batch corresponds to. We estimate the regression vector w.h.p. if there are enough batches in the collection of medium and small-size batches from that sub-populations and the sub-population follows a linear regression model.

The regression vector minimizes the expected squared loss for the sub-population. Therefore, we use a gradient-descent-based approach to estimate such a vector. We start with an initial estimate (all zero) and improve this estimate by performing multiple rounds of gradient descent steps.

Our approach to estimating the gradient in each step is inspired by [95]. However, they used it to directly estimate regression vectors of all sub-populations simultaneously. First, using a large number of smaller batches we estimate a smaller subspace of $\mathbb{R}^d$ that preserves the norm of the gradient. Next, using the sampled medium-size batch from the sub-population, we test which of the remaining medium-size batches has a projection of gradient close to the sampled batch, and use them to estimate the gradient in this smaller subspace. The advantage of sub-space reduction is that testing and estimation of the gradient in the smaller subspace is easier, and reduces the minimum length of medium-size batches required for testing and the number of medium-size batches required for estimation. A crucial ingredient of our algorithm is clipping, which limits the effect of other components and allows the algorithm to work for heavy-tailed distributions.

Sampling more than $\tilde{\Omega}(1/\alpha)$ medium-size batches and repeating this process for all the sampled batches ensures that we recover a list containing regression vector estimates for all large subgroups.

We describe the algorithm in detail in Section 6.3 after having presented our main theorems in Section 6.2. Then in Section 6.4 we compare our algorithm with the one in [95] on simulated datasets, to show that our algorithm performs better in the setting of the latter paper as well as generalizes to settings that are outside the assumptions of [95].

6.2 Problem Formulation and Main Results

6.2.1 Problem Formulation

Consider distributions $\mathcal{D}_0, \ldots, \mathcal{D}_{k-1}$ over input-output pairs $(x, y) \in \mathbb{R}^d \times \mathbb{R}$. A *batch* b consists of i.i.d. samples from one of the distributions. Samples in different batches are independent. There are two sets of batches. Batches in B_s are *small* and contain at least two samples each, while batches in B_m are of medium size and contain at least n_m samples. Next, we describe the distributions. To aid this description and the remaining paper we first introduce some notation.

6.2.2 Notation

The L_2 *norm* of a vector u is denoted by $\|u\|$ and represents the length of the vector. The *norm*, or *spectral norm*, of a matrix M is denoted by $\|M\|$ and is defined as the maximum value of $\|Mu\|$ for all unit vectors u. If M is a symmetric matrix, the norm simplifies to $\|M\| = \max_{\|u\|=1} |u^\intercal Mu|$, and for a positive semidefinite matrix M, we have $\|M\| = \max_{\|u\|=1} u^\intercal Mu$. The trace of a symmetric matrix M is $\mathrm{Tr}(M) := \sum_i M_{ii}$, the sum of the elements on the main diagonal of M. We will use the symbol S to denote an arbitrary collection of samples. For a batch denoted by b, we will use S^b to represent the set of all n^b samples in the batch.

6.2.3 Data Distributions

Let $\Sigma_i := \mathbb{E}_{\mathcal{D}_i}[xx^\intercal]$ denote the second-moment matrix of input for distribution $\mathcal{D}_i$.

Let $I \subseteq \{0, 1, .., k-1\}$ denote the collection of indices of distributions sampled in at least α_s and α_m fractions of the batches in B_s and B_m, respectively, and satisfy the following assumptions standard in heavy-tailed linear regression [38, 41].

1. (Input distribution) There are constants C and C_1 such that for all $i \in I$,
   (a) $L4$-$L2$ hypercontractivity: For all $u \in \mathbb{R}^d$, $\mathbb{E}_{\mathcal{D}_i}[(x \cdot u)^4] \leq C(\mathbb{E}_{\mathcal{D}_i}[(x \cdot u)^2])^2$.
   (b) Bounded condition number: For normalization purpose we assume $\min_{\|u\|=1} u^\intercal \Sigma_i u \geq 1$ and to bound the condition number we assume that $\|\Sigma_i\| \leq C_1$.
2. (Input-output relation) There is a $\sigma > 0$ s.t. for all $i \in I$, $y = w_i \cdot x + \eta$, where $w_i \in \mathbb{R}^d$ is an unknown regression vector, and η is a noise independent of x, with zero mean $\mathbb{E}_{\mathcal{D}_i}[\eta] = 0$, and $\mathbb{E}_{\mathcal{D}_i}[\eta^2] \leq \sigma^2$. Note that by definition, the distribution of η may differ for each i.

We will recover the regression vectors w_i for all $i \in I$. For $i \notin I$, we require only that the input distribution satisfies $\|\Sigma_i\| \leq C_1$, same as the second half of assumption 1(b). The input-output relation for samples generated by $\mathcal{D}_i$ for $i \notin I$ *may be arbitrary*, and in particular, does not even need to follow a linear regression model, and the fraction of batches with samples from $\mathcal{D}_i$ in B_s and B_m may be arbitrary.

To simplify the presentation, we make two additional assumptions. First, there is a constant $C_2 > 0$ such that for all components $i \in \{0, 1, .., k-1\}$, and random sample $(x, y) \sim \mathcal{D}_i$,

$\|x\| \leq C_2\sqrt{d}$, a.s. Second, for all $i \in I$ and a random sample $(x, y) \sim \mathcal{D}_i$, the noise distribution $\eta = y - w_i \cdot x$ is symmetric around 0. As discussed in Appendix 6.15, these assumptions are not limiting.

Remark 3. To simplify the presentation, we assumed that the batches exactly follow one of the k distributions. However, our techniques can be extended to more general scenarios. Let $\mathcal{D}^b$ denote the underlying distribution of batch b. Instead of requiring $\mathcal{D}^b = \mathcal{D}_i$ for some $i \in \{0, 1, .., k-1\}$, our methods can be extended to cases when the expected value of the gradients for $\mathcal{D}^b$ and $\mathcal{D}_i$ are close and if $i \in I$, regression vector w_i achieves a small mean square error of at most σ^2. This is guaranteed if (1) $\|\mathbb{E}_{\mathcal{D}^b}[xx^\intercal] - \Sigma_i\|$ is small, (2) for all $x \in \mathbb{R}^d$, $|\mathbb{E}_{\mathcal{D}^b}[y|x] - \mathbb{E}_{\mathcal{D}_i}[y|x]|$ is small, and (3) if $i \in I$ then for all $x \in \mathbb{R}^d$, $\mathbb{E}_{\mathcal{D}^b}[(y - w_i \cdot x)^2|x] \leq \sigma^2$. The strict identity requirement $\mathcal{D}^b = \mathcal{D}_i$ can therefore be replaced by these three approximation conditions.

6.2.4 Main Results

Estimating regression vectors

We begin by presenting our result for estimating the regression vector of a component $\mathcal{D}_i$, for any $i \in I$. This result assumes that in addition to the batch collections B_s and B_m, we have an extra medium-sized batch denoted as b^* which contains samples from $\mathcal{D}_i$. W.l.o.g, we assume $i = 0$.

Theorem 106. *Suppose index* 0 *is in set* I *and let* b^* *be a batch of* $\geq n_m$ *i.i.d. samples from* $\mathcal{D}_0$. *For* $\delta, \epsilon \in (0, 1]$, *if* $|B_s| = \tilde{\Omega}(\frac{d}{{\alpha_s}^2\epsilon^4})$, $n_m = \tilde{\Omega}(\min\{\sqrt{k}, \frac{1}{\epsilon\sqrt{\alpha_s}}\} \cdot \frac{1}{\epsilon^2})$, *and* $|B_m| = \tilde{\Omega}(\frac{1}{\alpha_m} \min\{\sqrt{k}, \frac{1}{\epsilon\sqrt{\alpha_s}}\})$, *then Algorithm 10 runs in polynomial time and returns estimate* $\hat{w}$, *such that with probability* $\geq 1 - \delta$, $\|\hat{w} - w_0\| \leq \epsilon\sigma$.

We provide a proof sketch of Theorem 106 and the description of Algortihm 10 in Section 6.3, and a formal proof in Appendix 6.13. Algorithm 10 can be used to estimate w_i for all $i \in I$, and the requirement of a separate batch b^* is not crucial. It can be obtained by repeatedly sampling a batch from B_m and running the algorithm for these sampled b^*. Since all the components in I have $\geq \alpha_m$ fraction of batches in B_m, then randomly sampling b^* from B_m, $\tilde{\Theta}(1/\alpha_m)$ times would ensure that, with high probability, we have b^* corresponding to each component. We can then return a list of size $\tilde{\Theta}(1/\alpha_m)$ containing estimates corresponding to each sampled b^*. Then, with high probability, the list will have an estimate of the regression vectors for all components. Note that in this case, returning a list is unavoidable as there is no way to assign an appropriate index to the regression vector estimates. The following corollary follows from the above discussion and Theorem 106.

Corollary 107. *For $\delta, \epsilon \in (0,1]$, if $|B_s| = \tilde{\Omega}(\frac{d}{{\alpha_s}^2\epsilon^4})$, $n_m = \tilde{\Omega}(\min\{\sqrt{k}, \frac{1}{\epsilon\sqrt{\alpha_s}}\} \cdot \frac{1}{\epsilon^2})$, and $|B_m| \geq \tilde{\Omega}(\frac{1}{\alpha_m}\min\{\sqrt{k}, \frac{1}{\epsilon\sqrt{\alpha_s}}\})$, the above modification of Algorithm 10 runs in polynomial-time and outputs a list L of size $\tilde{\mathcal{O}}(1/\alpha_m)$ such that with probability $\geq 1-\delta$, the list has an accurate estimate for regression vectors w_i for each $i \in I$, namely $\max_{i\in I}\min_{\hat{w}\in L}\|\hat{w} - w_i\| \leq \epsilon\sigma$.*

In particular, this corollary implies that for any $i \in I$, the algorithm requires only $\tilde{\Omega}(d/\alpha_s)$ batches of size two and $\tilde{\Omega}(\min\{\sqrt{k}, \frac{1}{\sqrt{\alpha_s}}\})$ medium-size batches of size $\tilde{\Omega}(\min\{\sqrt{k}, \frac{1}{\sqrt{\alpha_s}}\})$ from distribution $\mathcal{D}_i$ to estimate w_i within an accuracy $o(\sigma)$. Furthermore, it is easy to show that any $o(\sigma)$ accurate estimate of regression parameter w_i achieves an expected prediction error of $\sigma^2(1+o(1))$ for output y given input x generated from this $\mathcal{D}_i$.

Note that results work even for infinite k and without any separation assumptions on regression vectors. The $\min(\sqrt{k}, 1/\sqrt{\alpha_s})$ dependence is the best of both words. This dependence is reasonable for recovering components with a significant presence or if the number is few.

The total number of samples required by the algorithm from $\mathcal{D}_i$ in small size batches B_s and medium size batches B_m are only $\tilde{\mathcal{O}}(d/\alpha_s)$ and $\tilde{\mathcal{O}}(\min\{k, 1/\alpha_s\})$. Note that any estimator would require $\Omega(d)$ samples for such estimation guarantees even in the much simpler setting with just i.i.d. data. Therefore, in the high-dimensional regime, where $d \gg \tilde{\mathcal{O}}(\min\{k, 1/\alpha_s\})$, the samples in the medium-size batches in themselves have $\ll d$ samples and are insufficient to learn w_i. Note that the total number of samples required from $\mathcal{D}_i$ in B_s and B_m by the algorithm is within $\tilde{\mathcal{O}}(1/\alpha_s)$ factor from that required in a much simpler single component setting.

Prediction using list of regression vector estimates

The next theorem shows that given a list L containing estimates of w_i for all $i \in I$ and $\Omega(\log(1/\alpha_s))$ samples from $\mathcal{D}_i$ for some $i \in I$, we can identify an estimate of regression vector achieving a small prediction error for $\mathcal{D}_i$. The proof of the theorem and the algorithm is in Appendix 6.7.

Theorem 108. *For any $i \in I$, $\beta > 0$, and list L that contains at least one β good estimate of regression parameter of $\mathcal{D}_i$, namely $\min_{w \in L} \|w - w_i\| \leq \beta$. Given $\mathcal{O}(\max\{1, \frac{\sigma^2}{\beta^2}\} \log \frac{L}{\delta})$ samples from $\mathcal{D}_i$ Algorithm 12 identifies an estimate w, s.t. with probability $\geq 1 - \delta$, $\|w - w_i\| = \mathcal{O}(\beta)$ and it achieves an expected estimation error $\mathbb{E}_{\mathcal{D}_i}[(\hat{w} \cdot x - y)^2] \leq \sigma^2 + \mathcal{O}(\beta^2)$.*

Combining the above theorem and Theorem 106, we get

Theorem 109. *For $\delta, \epsilon \in (0, 1]$, suppose that $|B_s| = \tilde{\Omega}(\frac{d}{{\alpha_s}^2 \epsilon^4})$, $n_m = \tilde{\Omega}(\min\{\sqrt{k}, \frac{1}{\epsilon\sqrt{\alpha_s}}\} \cdot \frac{1}{\epsilon^2})$, and $|B_m| \geq \tilde{\Omega}(\frac{1}{\alpha_m} \min\{\sqrt{k}, \frac{1}{\epsilon\sqrt{\alpha_s}}\})$. Then, there exists a polynomial-time algorithm that, with probability $\geq 1 - \delta$, outputs a list L of size $\tilde{\mathcal{O}}(1/\alpha_m)$ containing estimates of w_i's for $i \in I$. Further, given $|S| \geq \Omega(\frac{1}{\epsilon^2} \log \frac{1}{\delta\alpha_m})$ samples from $\mathcal{D}_i$, for any $i \in I$, Algorithm 12 returns $\hat{w} \in L$ that with probability $\geq 1 - \delta$ satisfies $\|w_i - \hat{w}\| \leq \mathcal{O}(\epsilon\sigma)$ and achieves an expected estimation error $\mathbb{E}_{\mathcal{D}_i}[(\hat{w} \cdot x - y)^2] \leq \sigma^2 + \mathcal{O}(\epsilon^2\sigma^2)$*

When $\epsilon = o(1)$, the corollary implies that for $|B_s| = \tilde{\Omega}(\frac{d}{{\alpha_s}^2})$, $n_m = \tilde{\Omega}(\min\{\sqrt{k}, \frac{1}{\sqrt{\alpha_s}}\})$, and $|B_m| \geq \tilde{\Omega}(\frac{1}{\alpha_m} \min\{\sqrt{k}, \frac{1}{\sqrt{\alpha_s}}\})$, Algorithm 10 can be used to obtain a list L of size $\tilde{\mathcal{O}}(1/\alpha_s)$. Given this list, and $|S| \geq \Omega(\log \frac{1}{\alpha_s\delta})$ samples from $\mathcal{D}_i$ for any $i \in I$, Algorithm 12 returns $\hat{w} \in L$ that achieves an expected estimation error $\mathbb{E}_{\mathcal{D}_i}[(\hat{w} \cdot x - y)^2] \leq \sigma^2(1 + o(1))$.

6.3 Algorithm for recovering regression vectors

This section provides an overview and pseudo-code of Algorithm 10, along with an outline of the proof that achieves the guarantee stated in Theorem 106. As per the theorem, we assume that index 0 belongs to I, and we have a batch b^* containing n_m samples from the distribution $\mathcal{D}_0$. Note that $\mathcal{D}_0$ satisfies the conditions mentioned in Section 6.2.3 and that B_s and B_m each have $\geq |B_s|\alpha_s$ and $\geq |B_m|\alpha_m$ batches with i.i.d. samples from $\mathcal{D}_0$. However, the identity of these batches is unknown.

Gradient Descent. Note that w_0 minimizes the expected square loss for distribution $\mathcal{D}_0$. Our algorithm aims to estimate w_0 by taking a gradient descent approach. It performs a total of R gradient descent steps. Let $\hat{w}^{(r)}$ denote the algorithm's estimate of w_0 at the beginning of step r. Without loss of generality, we assume that the algorithm starts with an initial estimate of $\hat{w}^{(1)} = 0$. At step r, the algorithm produces an estimate $\Delta^{(r)}$ of the gradient of the expected square loss for distribution $\mathcal{D}_0$ at its current estimate $\hat{w}^{(r)}$. We refer to this estimate as the expected gradient for $\mathcal{D}_0$ at $\hat{w}^{(r)}$, or simply the expected gradient. The algorithm then updates its current estimate for the next round as $\hat{w}^{(r+1)} = \hat{w}^{(r)} - \Delta^{(r)}/C_1$.

The main challenge the algorithm faces is the accurate estimation of the expected gradients in each step. Accurately estimating the expected gradients at each step would require $\Omega(d/\epsilon^2)$ i.i.d. samples from $\mathcal{D}_0$. However, our algorithm only has access to a medium-size batch b^* that is guaranteed to have samples from $\mathcal{D}_0$ and this batch contains far fewer samples. And for batches in B_s and B_m, the algorithm doesn't know which of the batches has samples from $\mathcal{D}_0$. Despite these challenges, we demonstrate an efficient method to estimate the expected gradients accurately.

The algorithm randomly divides sets B_s and B_m into R disjoint equal subsets, denoted as $\{B_s^{(r)}\}_{r=1}^R$ and $\{B_m^{(r)}\}_{r=1}^R$, respectively. The samples in batch b^* are divided into two collections of equal disjoint parts, denoted as $\{S_1^{b^*,(r)}\}_{r=1}^R$ and $\{S_2^{b^*,(r)}\}_{r=1}^R$. At each iteration r, the algorithm uses the collections of medium and small batches $B_s^{(r)}$ and $B_m^{(r)}$, respectively, along with the two collections of i.i.d. samples $S_1^{b^*,(r)}$ and $S_2^{b^*,(r)}$ from $\mathcal{D}_0$ to estimate the gradient at point $w^{(r)}$. While this division may not be necessary for practical implementation, this ensures independence between the stationary point $\hat{w}^{(r)}$ and the gradient estimate which facilitates our theoretical analysis and only incurs a logarithmic factor in sample complexity.

Next, we describe how the algorithm estimates the gradient and the guarantees of this estimation. Due to space limitations, we provide a brief summary here, and a more detailed

Algorithm 10. MainAlgorithm

```
Input: Collections of batches $B_s$ and $B_m$, $\alpha_s$, $k$, a medium size batch $b^*$ of i.i.d. samples from $\mathcal{D}_0$, distribution parameters (upper bounds) $\sigma$, $C$, $C_1$, an upper bound $M$ on $\|w_0\|$, $\epsilon$ and $\delta$,
Output: Estimate of $w_0$
$R \leftarrow \Theta(C_1 \log \frac{M}{\sigma})$, $\epsilon_1 \leftarrow \Theta(1)$, $\epsilon_2 \leftarrow \Theta\Big(\frac{1}{C_1\sqrt{C+1}}(\epsilon_1 + \frac{1}{\sqrt{C_1}})\Big)$, $\ell \leftarrow \min\{k, \frac{1}{2\alpha_s\epsilon_2^2}\}$ $\delta' \leftarrow \frac{\delta}{5R}$
Partition the collection of batches $B_s$ into $R$ disjoint same size random parts $\{B_s^{(r)}\}_{r\in[R]}$.
Similarly partition $B_m$ into $R$ disjoint same size random parts $\{B_m^{(r)}\}_{r\in[R]}$.
Divide samples $S^{b^*}$ into $2R$ disjoint same size random parts $\{S_1^{b^*,(r)}\}_{r\in[R]}$ and $\{S_2^{b^*,(r)}\}_{r\in[R]}$.
Initilize $\hat{w}^{(1)} \leftarrow 0$
for $r$ from 1 to $R$ do
    $\kappa^{(r)} \leftarrow \text{ClipEst}(S_1^{b^*,(r)}, \hat{w}^{(r)}, \epsilon_1, \delta', \sigma, C, C_1)$
    $P^{(r)} \leftarrow \text{GradSubEst}(B_s^{(r)}, \kappa^{(r)}, \hat{w}^{(r)}, \ell\ )$
    $\Delta^{(r)} \leftarrow \text{GradEst}(B_m^{(r)}, S_2^{b^*,(r)}, \kappa^{(r)}, \hat{w}^{(r)}, P^{(r)}, \epsilon_2, \delta')$
    $\hat{w}^{(r+1)} \leftarrow \hat{w}^{(r)} - \frac{1}{C_1}\Delta^{(r)}$
end for
$\hat{w} \leftarrow \hat{w}^{(R+1)}$ and Return $\hat{w}$
```

description, along with formal proofs, can be found in the appendix. We start by introducing a clipping operation on the gradients, which plays a crucial role in the estimation process.

Clipping. Recall the squared loss of samples (x, y) on point w is $(w \cdot x - y)^2/2$ and its gradient is $(x \cdot w - y)x$. Instead of directly working with the gradient of the squared loss, we work with its clipped version. Given a *clipping parameter* $\kappa > 0$, the *clipped gradient* for a sample (x, y) evaluated at point w is defined as

$$\nabla f(x, y, w, \kappa) := \tfrac{(x\cdot w - y)}{|x\cdot w-y|\vee\kappa}\kappa x.$$

For a collection of samples S, the *clipped gradient* $\nabla f^b(w, \kappa)$ is the average of the clipped gradients of all samples in S, i.e., $\nabla f(S, w, \kappa) := \frac{1}{|S|}\sum_{(x,y)\in S} \nabla f(x, y, w, \kappa)$.

The clipping parameter κ controls the level of clipping and for $\kappa = \infty$, the clipped and the unclipped gradients are the same. The clipping step is necessary to make our gradient estimate more robust, by limiting the influence of the components other than $\mathcal{D}_0$ (in lemma 111), and as a bonus, we also obtain better tail bounds for the clipped gradients. Theorem 112 shows that for $\kappa \geq \Omega(\sqrt{\mathbb{E}_{\mathcal{D}_0}[(y - x \cdot w)^2]})$, the difference between the expected clipped gradient and the expected gradient $\|\mathbb{E}_{\mathcal{D}_0}[(\nabla f(x,y,w,\kappa)] - \mathbb{E}_{\mathcal{D}_0}[(x \cdot w - y)x]\|$ is small. Therefore, the ideal value of κ at point w is $\Theta(\sqrt{\mathbb{E}_{\mathcal{D}_0}[(y - x \cdot w)^2]})$.

For the estimate $\hat{w}^{(r)}$ at step r, the choice of the clipping parameter is represented by $\kappa^{(r)}$. To estimate a value for $\kappa^{(r)}$ that is close to its ideal value, the algorithm employs the subroutine ClipEst (presented as Algorithm 13 in the appendix). The subroutine estimates the expected value of $(y - x \cdot \hat{w}^{(r)})^2$ by using i.i.d. samples $S_1^{b^*,(r)}$ from the distribution $\mathcal{D}_0$. According to Theorem 114 in Appendix 6.9, the subroutine w.h.p. obtains $\kappa^{(r)}$ that is close to the ideal value. This ensures that the difference between the expectation of clipped and unclipped gradients is small, and thus, estimating the expectation of clipped gradients can replace estimating the actual gradients.

Subspace Estimation. The algorithm proceeds by using subroutine GradSubEst (presented as Algorithm 14 in Appendix 6.10) with $\widehat{B} = B_s^{(r)}$, $w = \hat{w}^{(r)}$, and $\kappa = \kappa^{(r)}$ to estimate a smaller subspace $P^{(r)}$ of $\mathbb{R}^d$.The expected projection of the clipped gradient on $P^{(r)}$ is nearly the same as the expected value of the clipped gradient, hence to estimate the expected gradient, it suffices to estimate the expected projection of the clipped gradient on $P^{(r)}$, which now requires fewer samples since $P^{(r)}$ is a lower dimensional subspace. The subroutine constructs a matrix A such that $\mathbb{E}[A] = \sum_i p_i \mathbb{E}_{\mathcal{D}_i}[\nabla f(x,y,w,\kappa)]\mathbb{E}_{\mathcal{D}_i}[\nabla f(x,y,w,\kappa)]^\intercal$, where p_i denotes the fraction of batches in $\widehat{B}$ that have samples from $\mathcal{D}_i$. Since $B_s^{(r)}$ are obtained by randomly partitioning B_s, w.h.p. $p_0 \approx \alpha_s$. It is crucial for the success of the subroutine that the expected contribution of every batch in the above expression is a PSD matrix. The clipping helps in bounding the contribution of other components and statistical noise.

The subroutine returns the projection matrix $P^{(r)}$ for the subspace spanned by the top ℓ singular vectors of A, where $\ell = \min\{k, \Theta(1/\alpha_s)\}$. It is worth noting that when $1/\alpha_s$ is much smaller than k (thinking of the extreme case $k = \infty$), our algorithm still only requires estimating the top $\ell = 1/\alpha_s$ dimensional subspace, since those infinitely many components can create at most $(1/\alpha_s - 1)$ directions with weight greater than α_s, therefore the direction of $\mathcal{D}_0$ must appear in the top $\Theta(1/\alpha_s)$ subspace. Theorem 116 in Appendix 6.10 characterizes the guarantees for this subroutine. Informally, if $\widehat{B} \geq \tilde{\Omega}(d/\alpha^2)$, then w.h.p., the expected value of the projection of the clipped gradient on this subspace is nearly the same as the expected value of the clipped gradient, namely $\|\mathbb{E}_{\mathcal{D}_0}[P^{(r)}\nabla f(x, y, w, \kappa)] - \mathbb{E}_{\mathcal{D}_0}[\nabla f(x, y, w, \kappa)]\|$ is small.

We note that our construction of matrix A for the subroutine is inspired by a similar construction in [95], where they used it for directly estimating regression vectors. Our results generalize the applicability of the procedure to provide meaningful guarantees even when the number of components $k = \infty$. Additionally, Lemma 118 improves matrix perturbation bounds in Lemma 5.1 of [95], which is crucial for applying this procedure for heavy-tailed distributions and reducing the number of required batches.

Algorithm 11. GradEst

```
Input: A collection of medium batches $\widehat{B}$, a collection of samples $S^*$ from $\mathcal{D}_0$, $\kappa$, $w$, projection matrix $P$ for subspace of $\mathbb{R}^d$, parameter $\epsilon$, $\delta'$
Output: An estimate of clipped gradient at point $w$.
$T_1 \leftarrow \Theta(\log \frac{|\widehat{B}|}{\delta'})$ and $T_2 \leftarrow \Theta(\log \frac{1}{\delta'})$
For each $b$ divide $S^b$ into two equal random parts $S^b_1$ and $S^b_2$
For each $b$ further divide $S^b_1$ into $2T_1$ equal random parts, and denote them as $\{S^b_{1,j}\}_{j\in[2T_1]}$
Divide $S^*$ into $2T_1$ equal random parts, and denote them as $\{S^*_j\}_{j\in[2T_1]}$
$\zeta^b_j := \big(\nabla f(S^b_{1,j}, w, \kappa) - \nabla f(S^*_j, w, \kappa)\big)^\intercal P^\intercal P\big(\nabla f(S^b_{1,T_1+j}, w, \kappa) - \nabla f(S^*_{T_1+j}, w, \kappa)\big)$
Let $\widetilde{B} \leftarrow \Big\{b \in \widehat{B} : median\{\zeta^b_j : j \in [T_1]\} \leq \epsilon^2\kappa^2 C_1\Big\}$
For each $b$ divide $S^b_2$ into $T_2$ equal parts randomly, and denote them as $\{S^b_{2,j}\}_{j\in[T_2]}$
For $i \in [T_2]$, let $\Delta_i \leftarrow \frac{1}{|\widetilde{B}|}\sum_{b\in\widetilde{B}} P\nabla f(S^b_{2,i}, w, \kappa)$.
Let $\xi_i \leftarrow median\{j \in [T_2] : \|\Delta_i - \Delta_j\|\}$
Let $i^* \leftarrow \arg\min\{i \in [T_2] : \xi_i\}$ and $\Delta \leftarrow \Delta_{i^*}$
Return $\Delta$
```

Estimating expectation of clipped gradient projection. The last subroutine, called GradEst, estimates the expected projection of the clipped gradient using medium-size batches $B^{(r)}_m$ and i.i.d. samples $S^{b^*,(r)}_2$ from $\mathcal{D}_0$. First, GradEst divides each batch in $B^{(r)}_m$ into two equal parts and uses the first half of the samples in each batch b and samples $S^{b^*,(r)}_2$ to test whether the expected projection of clipped gradient for the distribution batch b was sampled from and $\mathcal{D}_0$ are close or not. With high probability, the algorithm retains all the batches from $\mathcal{D}_0$ and rejects batches from all distributions for which the difference between the two expectations is large. This test requires $\tilde{\Omega}(\sqrt{\ell})$ samples in each batch, where ℓ is the dimension of the projected clipped gradient.

After identifying the relevant batches, GRADEST estimates the projection of the clipped gradients using the second half of the samples in these batches. Since the projections of the clipped gradients lie in an ℓ dimensional subspace, $\Omega(\ell)$ samples suffice for the estimation. To obtain high-probability guarantees, the procedure uses the median of means approach for both testing and estimation.

The guarantees of the subroutine are described in Theorem 119, which implies that the estimate $\Delta^{(r)}$ of the gradient satisfies $\|\Delta^{(r)} - \mathbb{E}_{\mathcal{D}_0}[P^{(r)}\nabla f(x, y, w, \kappa)]\|$ is small.

Estimation guarantees for expected gradient.

Using the triangle inequality, we have:

$$\begin{aligned}\|\Delta^{(r)} - \mathbb{E}_{\mathcal{D}_0}[(x \cdot w - y)x]\| &\leq \|\mathbb{E}_{\mathcal{D}_0}[\nabla f(x, y, w, \kappa)] - \mathbb{E}_{\mathcal{D}_0}[(x \cdot w - y)x]\| \\ &+ \|\mathbb{E}_{\mathcal{D}_0}[\nabla f(x, y, w, \kappa)] - \mathbb{E}_{\mathcal{D}_0}[P^{(r)}\nabla f(x, y, w, \kappa)]\| + \|\Delta^{(r)} - \mathbb{E}_{\mathcal{D}_0}[P^{(r)}\nabla f(x, y, w, \kappa)]\|.\end{aligned}$$

As previously argued, all three terms on the right side of the inequality are small, hence $\Delta^{(r)}$ provides an accurate estimate of the gradient. Moreover, Lemma 126 shows that with an accurate estimation of expected gradients, gradient descent reaches an ϵ-accurate estimation of w_0 after $\mathcal{O}(\log \frac{\|w_0\|}{\sigma})$ steps. Therefore, setting $R = \Omega(\log \frac{\|w_0\|}{\sigma})$ suffices. This completes the description and proof sketch of Theorem 106. A more formal proof can be found in Appendix 6.13.

As mentioned before, given a new batch of only logarithmically many samples from subgroup $i \in I$, we can identify the weight vector $\hat{w}$ in the list L that is close to w_i. In the interest of space, we include the algorithm for selecting the appropriate weight vector from the list in Appendix 6.7 along with a discussion about how the algorithm (Algorithm 12) achieves the guarantees in Theorem 108.

6.4 Empirical Results

Setup. We have sets B_s and B_m of small and medium size batches and k distributions $\mathcal{D}_i$ for $i \in \{0, 1, \ldots, k-1\}$. For a subset of indices $I \subseteq \{0, 1, \ldots, k-1\}$, both B_s and B_m have a fraction of α batches that contain i.i.d. samples from $\mathcal{D}_i$ for each $i \in I$. And for each $i \in \{0, 1, \ldots, k-1\} \setminus I$ in the remaining set of indices, B_s and B_m have $(1 - |I|/16)/(k - |I|)$ fraction of batches, that have i.i.d samples from $\mathcal{D}_i$. In all figures, the output noise is distributed as $\mathcal{N}(0, 1)$. All small batches have 2 samples each, while medium-size batches have n_m samples each, which we vary from 4 to 32, as shown in the plots. We fix data dimension $d = 100$, $\alpha = 1/16$, number of small batches to $|B_s| = \min\{8dk^2, 8d/\alpha^2\}$ and the number of medium batches to $|B_m| = 256$. In all the plots, we average over 10 runs and report the standard error.

Evaluation. Our objective is to recover a small list containing good estimates for the regression vectors of $\mathcal{D}_i$ for each $i \in I$. We compare our proposed algorithm's performance with that of the algorithm in [95]. Given a new batch, we can choose the weight vector from the returned list, L that achieves the best error[1]. Then the MSE of the chosen weight is reported on another new batch drawn from the same distribution. The size of the new batch can be either 4 or 8 as marked in the plot. More details about our setup can be found in Appendix 6.16.

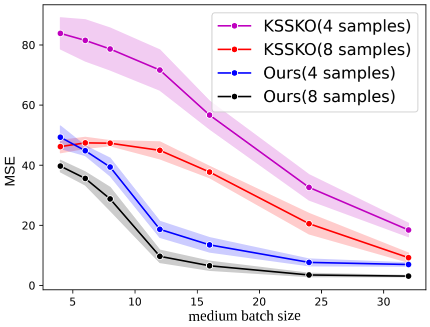


Figure 6.1. Same input dist., $k = 16$, large minimum distance between regression vectors.

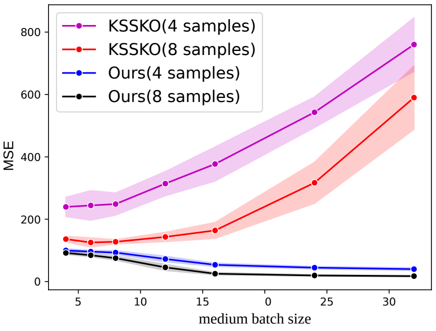


Figure 6.2. Different input dist., $k = 16$, large minimum distance between regression vectors.

Setting in [95]. We first compare our algorithm with the one in [95] in the same setting

[1] This simple approach showed better empirical performance than Algorithm 12, whose theoretical guarantees we described in Section 6.2.4

as the latter paper i.e. with more restrictive assumptions. The results are displayed in Figure 6.1, where $I = \{0, 1, \ldots, 15\}$ and all 16 distributions have been used to generate $1/16$ fraction of the batches. All the $\mathcal{D}_i$'s are equal to $\mathcal{N}(0, I)$, and the minimum distance between the regression vectors is comparable to their norm. It can be seen that even in the original setting of [95] our algorithm significantly outperforms the other at all the different medium batch sizes plotted on the x-axis.

Input distributions. Our algorithm can handle different input distributions for different subgroups. We test this in our next experiment presented in Figure 6.2. Specifically, for each i, we randomly generate a covariance matrix Σ_i such that its eigenvalues are uniformly distributed in $[1, C_1]$, and the input distribution for $\mathcal{D}_i$ is chosen as $\mathcal{N}(0, \Sigma_i)$. We set $C_1 = 4$. It can be seen that [95] completely fails in this case, while our algorithm retains its good performance.

In the interest of space, we provide additional results in Appendix 6.16 which include even more general settings: (i) when the minimum distance between regression vectors can be much smaller than their norm (ii) when the number of subgroups k can be very large but the task is to recover the regression weights for the subgroups that appear in a sufficient fraction of the batches. In both these cases, our algorithm performs much better than the baseline.

6.5 Conclusion

We study the problem of learning linear regression from batched data in the presence of sub-populations. In this work, we remove several restrictive assumptions from prior work and provide better guarantees in terms of overall sample complexity. Moreover, we require relatively fewer medium batches that need to contain less number of samples compared to prior work. Finally, in our empirical results, we show that our algorithm is both practical and more performant compared to a prior baseline.

It would be interesting to study robust algorithms for a similar setting where a fraction of batches can be corrupted i.e. they follow an arbitrary distribution. It can serve as a middle ground between our setting and list-decodable regression from batches, which would be a great direction for future work.

Appendix

6.6 Other related work

Meta Learning. The setting we considered in this paper is closely related to *meta learning* if we treat each batch as a task. Meta-learning approaches aim to jointly learn from past experience to quickly adapt to new tasks with little available data [132, 139]. This is particularly significant in our setting when each task is associated with only a few training examples. By leveraging structural similarities among those tasks (e.g. sub-population structure), meta-learning algorithms achieve far better accuracy than what can be achieved for each task in isolation [63, 127, 92, 116, 141, 131]. Learning mixture of linear dynamical systems has been studied in [33].

Robust and List decodable Linear Regression. Several recent works have focused on obtaining efficient algorithms for robust linear regression and sparse liner regression when a small fraction of data may be adversarial [18, 17, 11, 64, 123, 90, 50, 104, 88, 40, 112, 57, 87, 120, 38, 80, 93].

In scenarios where over half of the data may be arbitrary or adversarial, it becomes impossible to return a single estimate for the underlying model. Consequently, the requirement is relaxed to return a small list of estimates such that at least one of them is a good estimate for the underlying model. This relaxed framework, called “List decodable learning," was first introduced in [29]. List-decodable linear regression has been studied by [87, 126, 53], who have provided exponential runtime algorithms. Additionally, [53] has established statistical query lower bounds, indicating that polynomial-time algorithms may be impossible for this setting. However, as mentioned earlier, the problem can be solved in polynomial time in the batch setting as long as the batch size is greater than the inverse of the fraction of genuine data, as demonstrated in [41].

It's worth noting that an algorithm for list-decodable linear regression can be used to obtain a list of regression vector estimates for mixed linear regression.

Robust Learning from Batches. [125] presented the problem of robust learning of discrete distributions from untrustworthy batches, where a majority of batches share the same distribution and a small fraction are adversarial. They developed an exponential time algorithm for the problem. Subsequent works [32] improved the run-time to quasi-polynomial, while and [77] derived a polynomial time algorithm with an optimal sample complexity. The results were extended to learning one-dimensional structured distributions in [78, 31], and classification in [76, 96]. [2] examined a closely related problem of learning the parameters of an Erdős-Rényi random graph when a portion of nodes may be corrupted and their edges are maybe be chosen by an adversary.

6.7 Selecting a regression vector from a given list

In this section, we introduce Algorithm 12 and prove that it achieves the guarantees presented in Theorem 108.

Algorithm 12. Selecting the regression vector

```
Input: Samples $S$ from $\mathcal{D}_i$ for some $i \in I$, $C_1$, a list $L$ of possible estimates of $w_i$, and
  $\beta \geq 0$ s.t. $\beta \geq \min_{w \in L} \|w - w_i\|$.
Output: An estimate of $w_{i^*}$ from list $L$
Divide $S$ into $T_3 = \Theta(\log(|L|/\delta))$ equal parts $\{S_j\}_{j=1}^{T_3}$
while $\max\{\|w - w'\| : w, w' \in L\} \geq 12C_1\beta$ do
    pick any $w, w' \in L$ s.t. $\|w - w'\| \geq 12C_1\beta$.
    For $j \in [T_3]$, let $a_j \leftarrow \sum_{(x,y)\in S_j} \frac{1}{|S_j|}(x \cdot w - y)x \cdot (w - w')$
    $a \leftarrow$ Median$\{a_j : j \in [T_3]\}$
    If $a > \|w - w'\|^2/4$ remove $w$, else remove $w'$ from $L$
end while
Return any of the remaining $w \in L$
```

Without loss of generality, assume $i = 0$, and let $w^* = \arg\min_{w \in L} \|w - w_0\|$. From the condition in the theorem, we know that $\|w^* - w_0\| \leq \beta$. The algorithm is given access to $|S| = \Omega(\max 1, \frac{\sigma^2}{\beta^2} \log \frac{|L|}{\delta})$ samples. The algorithm chooses any two vectors $w, w' \in L$ that are more than $12C_1\beta$ distance apart and tests which of them is more likely to be within β distance from w_0 using samples in S. The algorithm performs $T_3 = \Theta(\log \frac{|L|}{\delta})$ such tests and takes the majority vote. It retains the vector that is more likely to be closer to w and discards the other from L. The algorithm terminates when all the vectors in L are within a distance of $12C_1\beta$ from each other, by choosing a vector from those remaining in L and returning it as an estimate of w_0. If w^* is retained in L until the end, using the simple triangle inequality for all w that remain in L at the end, we have $\|w - w_0\| \le \|w - w^*\| + \|w^* - w_0\| \le 12C_1\beta + \beta \le 13C_1\beta = \mathcal{O}(\beta)$. Therefore, the estimate returned by the algorithm achieves the desired accuracy in estimating w_0. Hence, it suffices to show that w^* is retained at the end with high probability.

Suppose w^* is not in the final list L. Then it must have been discarded by the test in favor of $\tilde{w} \in L$ such that $\|w^* - \tilde{w}\| \ge 12C_1\beta$. The following theorem shows that for any $\tilde{w}$ such that $\|w^* - \tilde{w}\| \ge 12C_1\beta$, the probability of the testing procedure rejecting w^* in favor of $\tilde{w}$ is at most $\delta/|L|$.

Theorem 110. *Given $\beta > 0$, list L, and samples S from $\mathcal{D}_0$, if $\min_{w\in L} \|w - w_0\| \le \beta$ and $|S| = \max\{1, \frac{\sigma^2}{\beta^2}\} \log \frac{L}{\delta}$, then for the parameter a computed in the while loop of Algorithm 12, with probability $1 - \delta/|L|$, we have $a \le \|w - w'\|^2/4$ if $w = w_0$ and $a > \|w - w'\|^2/4$ if $w' = w_0$.*

The testing procedure utilized in the algorithm is based on gradients. Specifically, it calculates the average of the gradient computed on samples at point w projected onto the vector $(w - w')$. The expected value of the gradient at w, and its projection onto $(w - w')$, are $(w - w_0)^\intercal \Sigma_0$ and $(w - w_0)^\intercal \Sigma_0 (w - w')$, respectively. If $w \approx w_0$, then the expected projection will be small. On the other hand. if $w' \approx w_0$ and w, then expected value of projection is $\approx (w - w')^\intercal \Sigma_0 (w - w') \gtrsim \|w - w'\|^2$. Using these observations, we prove Theorem 110 in the next subsection.

Finally, since the maximum number of comparisons made by the algorithm is $|L| - 1$, a union bound ensures that w^* will be retained until the end with probability greater than $1 - \delta$, completing the proof of Theorem 108.

6.7.1 Proof of Theorem 110

Proof. Note that a is the median of the set $\{a_j : j \in [T_3]\}$, where each a_j is computed using different sets of i.i.d. samples. Consequently, $\{a_j\}_{j\in[T_3]}$ are also i.i.d random variables.

We begin by calculating the expected value of a_j. Using the linearity of expectations, we have:

$$\begin{aligned}
\mathbb{E}[a_j] &\overset{(a)}{=} \mathbb{E}\Big[\textstyle\sum_{(x,y)\in S_j} \frac{1}{|S_j|}(x \cdot w - y)x \cdot (w - w')\Big] \\
&\overset{(b)}{=} \mathbb{E}_{\mathcal{D}_0}[(x \cdot w - y)x \cdot (w - w')] \\
&= \mathbb{E}_{\mathcal{D}_0}[(x \cdot (w - w_0) - (y - x \cdot w_0))x \cdot (w - w')] \\
&= \mathbb{E}_{\mathcal{D}_0}[(x \cdot (w - w_0)x \cdot (w - w')] - \mathbb{E}_{\mathcal{D}_0}[(y - x \cdot w_0)x \cdot (w - w')] \\
&= \mathbb{E}_{\mathcal{D}_0}[x \cdot (w - w_0)x \cdot (w - w')] \qquad (6.1) \\
&\overset{(c)}{=} \mathbb{E}_{\mathcal{D}_0}[(x \cdot (w - w'))^2] + \mathbb{E}_{\mathcal{D}_0}[x \cdot (w' - w_0)x \cdot (w - w')], \qquad (6.2)
\end{aligned}$$

here, (a) follows from the definition of a_j, (b) follows from the linearity of expectation, and since S_j contains i.i.d. samples from $\mathcal{D}_0$, (c) follows as the noise $y - x \cdot w_0$ has a zero mean and is independent of x.

Next, we compute the variance of a_j. Since a_j represents the average of $(x{\cdot}w{-}y)x{\cdot}(w{-}w')$ over $|S_j|$ i.i.d. samples, we have

$$\begin{aligned}
\mathrm{Var}(a_j) &= \frac{1}{|S_j|}\mathrm{Var}_{\mathcal{D}_0}((x \cdot w - y)x \cdot (w - w')) \\
&\le \frac{1}{|S_j|}\mathbb{E}_{\mathcal{D}_0}\Big[((x \cdot w - y)x \cdot (w - w'))^2\Big]. \qquad (6.3)
\end{aligned}$$

By applying Chebyshev's inequality, with a probability $\geq$ 3/4, the following holds for each a_j:

$$\mathbb{E}[a_j] - 2\text{Var}(a_j) \leq a_j \leq \mathbb{E}[a_j] + 2\text{Var}(a_j). \tag{6.4}$$

First, we consider the case when $w = w^*$. In this case, we have $\|w_0 - w\| \leq \beta$.

Using Equation (6.3), we can express the variance of a_j as follows:

$$\begin{aligned}
\text{Var}(a_j) &\leq \frac{1}{|S_j|}\mathbb{E}_{\mathcal{D}_0}\Big[\big((x \cdot w - y)x \cdot (w - w')\big)^2\Big] \\
&= \frac{1}{|S_j|}\mathbb{E}_{\mathcal{D}_0}\Big[\big(x \cdot (w - w_0)x \cdot (w - w') + (w_0 \cdot x - y)x \cdot (w - w')\big)^2\Big] \\
&\leq \frac{2}{|S_j|}\Big(\mathbb{E}_{\mathcal{D}_0}\Big[\big(x \cdot (w - w_0)x \cdot (w - w')\big)^2\Big] + \mathbb{E}_{\mathcal{D}_0}\Big[\big((w_0 \cdot x - y)x \cdot (w - w')\big)^2\Big]\Big),
\end{aligned}$$

where the last step uses the fact that for any $u, v \in \mathbb{R}$, $(u+v)^2 \leq 2u^2 + 2v^2$.

Next, we bound the two terms on the right. For the first term, we have

$$\begin{aligned}
\mathbb{E}_{\mathcal{D}_0}\Big[\big(x \cdot (w - w_0)x \cdot (w - w')\big)^2\Big] &\leq \sqrt{\mathbb{E}_{\mathcal{D}_0}[(x \cdot (w - w_0))^4]\mathbb{E}_{\mathcal{D}_0}[(x \cdot (w - w'))^4]} \\
&\leq C\mathbb{E}_{\mathcal{D}_0}[(x \cdot (w - w_0))^2]\mathbb{E}_{\mathcal{D}_0}[(x \cdot (w - w'))^2]. \qquad (6.5)
\end{aligned}$$

For the second term, we have:

$$\begin{aligned}
\mathbb{E}_{\mathcal{D}_0}\Big[\big((w_0 \cdot x - y)x \cdot (w - w')\big)^2\Big] &= \mathbb{E}_{\mathcal{D}_0}\big[(w_0 \cdot x - y)^2\big]\mathbb{E}_{\mathcal{D}_0}\Big[\big(x \cdot (w - w')\big)^2\Big] \\
&= \sigma^2\mathbb{E}_{\mathcal{D}_0}\Big[\big(x \cdot (w - w')\big)^2\Big], \qquad (6.6)
\end{aligned}$$

where the first inequality follows from assumption 1a and the second inequality follows from assumption 1b.

Combining the above three equations, we obtain:

$$\begin{aligned}\mathrm{Var}(a_j) &\leq \frac{2}{|S_j|}\mathbb{E}_{\mathcal{D}_0}\Big[(x\cdot(w-w'))^2\Big]\big(C\mathbb{E}_{\mathcal{D}_0}[(x\cdot(w-w_0))^2]+\sigma^2\big)\\ &\leq \frac{2}{|S_j|}C_1\|w-w'\|^2\big(CC_1\|w-w_0\|^2+\sigma^2\big),\end{aligned}$$

where the last inequality uses assumption 1b.

Using Equation (6.1), the Cauchy-Schwarz inequality, and assumption 1b, we have:

$$\mathbb{E}[a_j] \leq C_1\|w-w_0\|\cdot\|w-w'\|. \tag{6.7}$$

Combining the two equations above, we obtain:

$$\begin{aligned}\mathbb{E}[a_j]+2\sqrt{\mathrm{Var}(a_j)} &\leq \|w-w'\|\left(C_1\|w-w_0\|+\frac{2\sqrt{2C_1}}{\sqrt{|S_j|}}\Big(\sigma+\sqrt{CC_1}\|w-w_0\|\Big)\right)\\ &\overset{(a)}{\leq} \|w-w'\|\left(C_1\beta+\frac{\sqrt{8C_1}}{\sqrt{|S_j|}}\sigma+\frac{\sqrt{8C}C_1}{\sqrt{|S_j|}}\beta\right)\\ &\overset{(b)}{\leq} 3C_1\|w-w'\|\beta\\ &\overset{(c)}{\leq} \frac{\|w-w'\|^2}{4},\end{aligned}$$

here, in (a), we use $w=w^*$, which implies $\|w-w_0\|\leq\beta$, in (b), we utilize $|S_j|\geq 48C$ and $|S_j|\geq\frac{12\sigma^2}{C_1\beta^2}$, in (c), we use the fact that for any w and w' in the while loop of the algorithm, $\|w-w'\|\geq 12C_1\beta$. Consequently, it follows from Equation (6.4) that each $a_j\leq\frac{\|w-w'\|^2}{4}$ with probability $\geq 3/4$. Hence, with probability $\geq 1-\delta$ the median of a_j is $\leq\frac{\|w-w'\|^2}{4}$.

Next, we consider the case when $w'=w^*$. Firstly, we bound the variance using

Equation (6.3):

$$
\begin{align*}
&\text{Var}(a_j) \tag{6.8}\\
&\le \frac{1}{|S_j|}\mathbb{E}_{\mathcal{D}_0}\Big[((x\cdot w - y)x\cdot(w-w'))^2\Big]\\
&= \frac{1}{|S_j|}\mathbb{E}_{\mathcal{D}_0}\Big[\big((x\cdot(w-w'))^2 + x\cdot(w'-w_0)x\cdot(w-w') + (w_0\cdot x - y)x\cdot(w-w')\big)^2\Big]\\
&\overset{(a)}{\le} \frac{3}{|S_j|}\Bigg(\mathbb{E}_{\mathcal{D}_0}\Big[((w-w')\cdot x)^4\Big] + \mathbb{E}_{\mathcal{D}_0}\Big[(x\cdot(w'-w_0)x\cdot(w-w'))^2\Big] \tag{6.9}\\
&\quad + \mathbb{E}_{\mathcal{D}_0}\Big[((w_0\cdot x - y)x\cdot(w-w'))^2\Big]\Bigg)\\
&\overset{(b)}{\le} \frac{3}{|S_j|}\Bigg(C\mathbb{E}_{\mathcal{D}_0}\Big[(x\cdot(w-w'))^2\Big]^2 + \big(C\mathbb{E}_{\mathcal{D}_0}[(x\cdot(w'-w_0))^2]+\sigma^2\big)\mathbb{E}_{\mathcal{D}_0}[(x\cdot(w-w'))^2]\Bigg)\\
&\overset{(c)}{\le} \frac{3}{|S_j|}\Big(\sqrt{C}\mathbb{E}_{\mathcal{D}_0}\Big[(x(w-w'))^2\Big] + \Big(\sqrt{C\mathbb{E}_{\mathcal{D}_0}[(x(w'-w_0))^2]}+\sigma\Big)\sqrt{\mathbb{E}_{\mathcal{D}_0}[(x(w-w'))^2]}\Big)^2.
\end{align*}
$$

In (a), we use the fact that for any $t, u, v \in \mathbb{R}$, $(t+u+v)^2 \le 3t^2 + 3u^2 + 3v^2$. In (b) the first term is bounded using assumption 1a, the bound on the second term can be obtained similarly to Equation (6.5), and the bound on the last term is from Equation (6.6). Finally, in (c) we use the fact that for any $t, u, v \ge 0$, $(t+u+v)^2 \le t^2 + u^2 + v^2$.

Using Equation (6.2) and the equation above, we get

$$\begin{aligned}
&\mathbb{E}[a_j]-2\sqrt{\mathrm{Var}(a_j)}\\
&\geq \mathbb{E}_{\mathcal{D}_0}[(x\cdot(w-w'))^2]+\mathbb{E}_{\mathcal{D}_0}[x(w'-w_0)x\cdot(w-w')]\\
&-\frac{2\sqrt{3}}{\sqrt{|S_j|}}\Big(\sqrt{C}\mathbb{E}_{\mathcal{D}_0}\Big[(x(w-w'))^2\Big]+\big(\sqrt{C\mathbb{E}_{\mathcal{D}_0}[(x(w'-w_0))^2]}+\sigma\big)\sqrt{\mathbb{E}_{\mathcal{D}_0}[(x\cdot(w-w'))^2]}\Big)\\
&\overset{(a)}{\geq}\left(1-\frac{\sqrt{12C}}{\sqrt{|S_j|}}\right)\mathbb{E}_{\mathcal{D}_0}[(x\cdot(w-w'))^2]-\sqrt{\mathbb{E}_{\mathcal{D}_0}[(x\cdot(w'-w_0))^2]}\sqrt{\mathbb{E}_{\mathcal{D}_0}[(x\cdot(w-w'))^2]}\\
&\quad-\frac{\sqrt{12}}{\sqrt{|S_j|}}\Big(\Big(\sqrt{C\mathbb{E}_{\mathcal{D}_0}[(x\cdot(w'-w_0))^2]}+\sigma\Big)\sqrt{\mathbb{E}_{\mathcal{D}_0}[(x\cdot(w-w'))^2]}\Big)\\
&\overset{(b)}{\geq}\left(\frac{1}{2}\sqrt{\mathbb{E}_{\mathcal{D}_0}[(x(w-w'))^2]}-\frac{3}{2}\sqrt{\mathbb{E}_{\mathcal{D}_0}[(x(w'-w_0))^2]}-\frac{\sqrt{12}}{\sqrt{|S_j|}}\sigma\right)\sqrt{\mathbb{E}_{\mathcal{D}_0}[(x\cdot(w-w'))^2]}\\
&\overset{(c)}{\geq}\left(\frac{1}{2}\sqrt{\mathbb{E}_{\mathcal{D}_0}[(x\cdot(w-w'))^2]}-\frac{3}{2}\sqrt{\mathbb{E}_{\mathcal{D}_0}[(x\cdot(w'-w_0))^2]}-\sqrt{C_1}\beta\right)\sqrt{\mathbb{E}_{\mathcal{D}_0}[(x\cdot(w-w'))^2]},
\end{aligned}$$

here, in (a) we use the Cauchy-Schwarz inequality, (b) follows from $|S_j|\geq 48C$, and (c) utilizes $|S_j|\geq\frac{12\sigma^2}{C_1\beta^2}$. Next, we have:

$$\begin{aligned}
&\frac{1}{2}\sqrt{\mathbb{E}_{\mathcal{D}_0}[(x\cdot(w-w'))^2]}-\frac{3}{2}\sqrt{\mathbb{E}_{\mathcal{D}_0}[(x\cdot(w'-w_0))^2]}-\sqrt{C_1}\beta && (6.10)\\
&\overset{(a)}{\geq}\frac{1}{2}\|w-w'\|-\frac{3}{2}\sqrt{C_1}\|w'-w_0\|-\sqrt{C_1}\beta\\
&\overset{(b)}{\geq}\frac{1}{2}\|w-w'\|-\frac{5}{2}\sqrt{C_1}\beta\\
&\overset{(c)}{>}\frac{1}{4}\|w-w'\|, && (6.11)
\end{aligned}$$

here in (a), we use assumption 1b, (b) relies on $w'=w^*$, which implies $\|w'-w_0\|\leq\beta$, and (c) uses the fact that for any w and w' in the while loop of the algorithm, $\|w-w'\|\geq 12C_1\beta$ and $C_1\geq 1$.

Combining the above two equations, we obtain

$$\mathbb{E}[a_j]-2\sqrt{\mathrm{Var}(a_j)}>\frac{1}{4}\|w-w'\|^2.$$

Then from Equation (6.4) it follows that each $a_j > \frac{\|w-w'\|^2}{4}$ with probability $\geq 3/4$. Hence, with probability $\geq 1-\delta/|L|$ the median of a_j is $> \frac{\|w-w'\|^2}{4}$. ■

6.8 Properties of Clipped Gradients

The norm of the expected value and covariance of unclipped gradients for components other than $\mathcal{D}_0$ can be significantly larger than $\mathcal{D}_0$, acting as noise in the recovery process of $\mathcal{D}_0$. When using unclipped gradients, the algorithm's batch size and the number of batches must increase to limit the effect of these noisy components. And while the norm of the expected value and covariance of the unclipped gradient for $\mathcal{D}_0$ follows desired bounds, the maximum value of the unclipped gradient is unbounded, posing difficulties in applying concentration bounds. The following lemma shows that the clipping operation described in the main paper is able to address these challenges.

Lemma 111. *Let S be a collection of random samples drawn from distribution $\mathcal{D}_i$ for some $i \in \{0, 1, ..., k-1\}$. For any $\kappa \geq 0$ and $w \in \mathbb{R}^d$, the clipped gradient satisfies the following properties:*

1. $\|\mathbb{E}[\nabla f(S, w, \kappa)]\| \leq \kappa\sqrt{C_1}$,
2. $\|Cov(\nabla f(S, w, \kappa))\| \leq \frac{1}{|S|}\kappa^2 C_1$,
3. $\|\nabla f(S, w, \kappa)\| \leq \kappa C_2\sqrt{d}$ *almost surely*,
4. $\mathbb{E}[\|\nabla f(S, w, \kappa)\|^2] \leq C_1\kappa^2 d$,
5. *for all unit vectors u,* $\|\mathbb{E}[(\nabla f(S, w, \kappa) \cdot u)^2]\| \leq \kappa^2 C_1$.

This lemma implies that for smaller values of κ, the norm of the expectations and covariance of clipped gradients is bounded by a smaller upper limit. The proof of the lemma is presented in Subsection 6.8.1.

The following theorem demonstrates that by appropriately choosing a sufficiently large value of κ, the norm of the expected difference between the clipped and unclipped gradients for distribution $\mathcal{D}_0$ can be small:

Theorem 112. *For any $\epsilon > 0$, $\kappa^2 \geq 8CC_1\mathbb{E}_{\mathcal{D}_0}[(y - x \cdot w)^2]/\epsilon$ the norm of difference between expected clipped gradient $\mathbb{E}_{\mathcal{D}_0}[(\nabla f(x, y, w, \kappa)]$ and expected unclipped gradient $\mathbb{E}_{\mathcal{D}_0}[(w \cdot x - y)x]$ is at most,*

$$\|\mathbb{E}_{\mathcal{D}_0}[(\nabla f(x, y, w, \kappa)] - \mathbb{E}_{\mathcal{D}_0}[(w \cdot x - y)x]\| \leq \epsilon \|w - w_0\|,$$

where $\mathbb{E}_{\mathcal{D}_0}[(w \cdot x - y)x] = \Sigma_0(w - w_0)$.

The theorem shows in order to estimate the expectation of gradients at point w for distribution $\mathcal{D}_0$, it is sufficient to estimate the expectation of clipped gradients at point w, as long as the clipping parameter κ is chosen to be at least $\Omega\left(\sqrt{\frac{\mathbb{E}_{\mathcal{D}_0}[(y-x\cdot w)^2]}{\epsilon}}\right)$.

Intuitively, when κ is much larger than $\mathbb{E}_{\mathcal{D}_0}[|y - x \cdot w|]$, with high probability the clipped and unclipped gradients at point w for a random sample from $\mathcal{D}_0$ will be identical. The proof of the theorem is a bit more nuanced and involves leveraging the symmetry of noise distribution and $L4 - L2$ hypercontractivity of distribution of x. The proof appears in Subsection 6.8.2.

In the algorithm, we set κ to approximately $\Theta(\sqrt{(\mathbb{E}_{\mathcal{D}_0}[(y - x \cdot w)^2] + \sigma^2)/\epsilon})$. This choice ensures that κ is close to the minimum value recommended by Theorem 112 for preserving the gradient expectation of $\mathcal{D}_0$. By selecting a small κ, we ensure a tighter upper bound on the expectation and covariance of the clipped gradient for other components, as described in Lemma 111. The use of the clipping operation also assists in obtaining improved bounds on the tails of the gradient by limiting the maximum possible norm of the gradients after clipping, as stated in item 3 of the lemma.

6.8.1 Proof of Lemma 111

Proof. Since $\nabla f(S, w, \kappa)$ is average of clipped gradients of $|S|$ independent samples from $\mathcal{D}_i$, it follows that

a) $\mathbb{E}[\nabla f(S, w, \kappa)] = \mathbb{E}_{\mathcal{D}_i}[\nabla f(x, y, w, \kappa)]$,

b) $\text{Cov}(\nabla f(S, w, \kappa)) = \frac{1}{|S|}\text{Cov}_{\mathcal{D}_i}(\nabla f(x, y, w, \kappa))$,

c) $\|\nabla f(S, w, \kappa)\| \leq \text{ess sup}_{(x,y)\sim\mathcal{D}_i} \|\nabla f(x, y, w, \kappa)\|$ a.s.,

d) $\mathbb{E}[\|\nabla f(S, w, \kappa)\|^2] \leq \mathbb{E}_{\mathcal{D}_i}\left[\|\nabla f(x, y, w, \kappa)\|^2\right]$, and

e) for all vectors u, $\|\mathbb{E}[(\nabla f(S, w, \kappa) \cdot u)^2]\| \leq \left\|\mathbb{E}_{\mathcal{D}_i}\left[(\nabla f(x, y, w, \kappa) \cdot u)^2\right]\right\|$.

We will now proceed to prove the five claims in the lemma by using these properties. Firstly, we can analyze the expected norm of $\mathbb{E}_{\mathcal{D}_i}[\nabla f(x, y, w, \kappa)]$ as follows:

$$\begin{aligned}
\|\mathbb{E}_{\mathcal{D}_i}[\nabla f(x, y, w, \kappa)]\| &= \max_{\|u\|} \|\mathbb{E}_{\mathcal{D}_i}[(\nabla f(x, y, w, \kappa) \cdot u)]\| \\
&\leq \max_{\|u\|} \|\mathbb{E}_{\mathcal{D}_i}[(\nabla f(x, y, w, \kappa) \cdot u)^2]^{1/2}\| \\
&\leq \max_{\|u\|} \|\mathbb{E}_{\mathcal{D}_i}[(\kappa x \cdot u)^2]^{1/2}\| \leq \kappa\sqrt{C_1},
\end{aligned}$$

here the first inequality follows from the Cauchy–Schwarz inequality and the last inequality follows from assumptions on distributions $\mathcal{D}_i$. Combining the above inequality with item a) above proves the first claim in the lemma.

Next, to prove the second claim in the lemma, we first establish bounds for the norm of the covariance of the clipped gradient of a random sample:

$$\begin{aligned}\|\mathrm{Cov}_{\mathcal{D}_i}(\nabla f(x,y,w,\kappa))\| &= \max_{\|u\|} \mathrm{Var}_{\mathcal{D}_i}(\nabla f(x,y,w,\kappa)\cdot u)\\ &\le \max_{\|u\|} \mathbb{E}_{\mathcal{D}_i}[(\nabla f(x,y,w,\kappa)\cdot u)^2]\\ &\le \max_{\|u\|} \mathbb{E}_{\mathcal{D}_i}[(\kappa x\cdot u)^2] \le \kappa^2 C_1.\end{aligned}$$

By using the above bound and combining it with item b), we establish the second claim in the lemma.

To prove the third item in the lemma, we first bound the norm of the clipped gradient:

$$\operatorname*{ess\,sup}_{(x,y)\sim\mathcal{D}_i} \|\nabla f(x,y,w,\kappa)\| \le \kappa \operatorname*{ess\,sup}_{(x,y)\sim\mathcal{D}_i} \|x\| \le \kappa C_2\sqrt{d}.$$

We then combine this bound with item c) to prove the third claim in the lemma.

Next, we bound the expected value of the square of the norm of the clipped gradient of a random sample,

$$\mathbb{E}_{\mathcal{D}_i}\left[\|\nabla f(x,y,w,\kappa)\|^2\right] \le \mathbb{E}_{\mathcal{D}_i}\left[\kappa^2\|x\|^2\right] = \kappa^2\mathrm{Tr}(\Sigma_i) \le \kappa^2 d\|\Sigma_i\| \le C_1\kappa^2 d.$$

This bound, combined with item d), proves the fourth claim in the lemma.

Finally, for any unit vector u, we bound

$$\left\|\mathbb{E}_{\mathcal{D}_i}\left[(\nabla f(x,y,w,\kappa)\cdot u)^2\right]\right\| \le \kappa^2\left\|\mathbb{E}_{\mathcal{D}_i}\left[(x\cdot u)^2\right]\right\| \le \kappa^2\|\Sigma_i\| \le \kappa^2 C_1.$$

This bound, combined with item e), shows the fifth claim in the lemma. ■

6.8.2 Proof of Theorem 112

We will utilize the following auxiliary lemma in the proof of the theorem. This lemma applies to general random variables.

Lemma 113. *For any $a \in \mathbb{R}$, $b > 0$ and a symmetric random variable z,*

$$\left|\mathbb{E}\left[(a+z) - \frac{(a+z)b}{\max(|a+z|, b)}\right]\right| \leq 2|a| \Pr(z > b - |a|)$$

Proof. We assume $a \geq 0$ and prove the lemma for this case. The statement for $a < 0$ case then follows from symmetry.

We rewrite the term inside the expectation in terms of indicator random variables:

$$\begin{aligned}
&(a+z) - \frac{(a+z)b}{\max(|a+z|, b)} \\
&= (a+z-b) \cdot \mathbb{1}(z > b-a) + (a+z+b) \cdot \mathbb{1}(z < -b-a) \\
&= (a+z-b) \cdot \mathbb{1}(b-a < z \leq b+a) + (a+z-b) \cdot \mathbb{1}(z > b+a) \\
&\quad + (a+z+b) \cdot \mathbb{1}(z < -b-a).
\end{aligned}$$

Taking the expectation on both sides,

$$\begin{aligned}
&\mathbb{E}\left[(a+z) - \frac{(a+z)b}{\max(|a+z|, b)}\right] \\
&= \mathbb{E}[(a+z-b) \cdot \mathbb{1}(b-a < z \leq b+a)] + \mathbb{E}[(a+z-b) \cdot \mathbb{1}(z > b+a)] \\
&\quad + \mathbb{E}[(a+z+b) \cdot \mathbb{1}(z < -b-a)] \\
&= \mathbb{E}[(a+z-b) \cdot \mathbb{1}(b-a < z \leq b+a)] + 2a \Pr(z > b+a),
\end{aligned}$$

where the last step follows because z is symmetric.

Next,

$$\begin{aligned}
&\left|\mathbb{E}\left[(a+z) - \frac{(a+z)b}{\max(|a+z|, b)}\right]\right| \\
&= \mathbb{E}[|a+z-b| \cdot \mathbb{1}(b-a < z \leq b+a)] + 2|a| \Pr(z > b+a) \\
&\leq \mathbb{E}[|2a| \cdot \mathbb{1}(b-a < z \leq b+a)] + 2|a| \Pr(z > b+a) \\
&\leq 2|a| \Pr(b-a < z \leq b+a) + 2|a| \Pr(z > b+a) \\
&= 2|a| \Pr(z > b-a).
\end{aligned}$$

■

Next, we proceed with the proof of Theorem 112 using the aforementioned lemma.

Proof of Theorem 112. Let (x, y) be a random sample from distribution $\mathcal{D}_0$, and let $\eta = y - w_0 \cdot x$ denote the noise. Recall that η is independent of x.

Note that:

$$(w \cdot x - y)x = ((w - w_0) \cdot x - \eta)x.$$

We will now evaluate the expected value of the unclipped gradient.

$$\mathbb{E}_{\mathcal{D}_0}[(w \cdot x - y)x] = \mathbb{E}_{\mathcal{D}_0}[((w - w_0) \cdot x - \eta)x] = \mathbb{E}_{\mathcal{D}_0}[((w - w_0) \cdot x)x] = \Sigma_0(w - w_0). \tag{6.12}$$

Next, we will bound the norm of the expected value of $(w \cdot x - y)x - \nabla f(x, y, w, \kappa)$, which represents the difference between the clipped gradient and the true gradient. We first expand this expression:

$$\begin{aligned}(w\cdot x-y)x-\nabla f(x,y,w,\kappa)&=\left((w\cdot x-y)-\frac{(w\cdot x-y)}{|w\cdot x-y|\vee\kappa}\kappa\right)x\\&=\left(((w-w_0)\cdot x-\eta)-\frac{((w-w_0)\cdot x-\eta)}{|(w-w_0)\cdot x-\eta|\vee\kappa}\kappa\right)x.\quad(6.13)\end{aligned}$$

Next, by applying Lemma 113, we have

$$\begin{aligned}&\mathbb{E}_\eta\left[((w-w_0)\cdot x-\eta)-\frac{((w-w_0)\cdot x-\eta)}{|(w-w_0)\cdot x-\eta|\vee\kappa}\kappa\right]\\&\le 2|(w-w_0)\cdot x|\cdot\Pr(\eta>\kappa-|(w-w_0)\cdot x|).\end{aligned}$$

Note that in the above expectation, we fixed x and took the expectation over noise η.

Let $Z:=\mathbb{1}(|x\cdot(w-w_0)|\ge\kappa/2)$. Observe that $\Pr(\eta>\kappa-|(w-w_0)\cdot x|)\le Z+\Pr(\eta>\kappa/2)$. Combining this observation with the above equation, we have:

$$\mathbb{E}_\eta\left[((w-w_0)\cdot x-\eta)-\frac{((w-w_0)\cdot x-\eta)}{|(w-w_0)\cdot x-\eta|\vee\kappa}\kappa\right]\le 2|(w-w_0)\cdot x|\cdot(\Pr(\eta>\kappa/2)+Z).\quad(6.14)$$

Then, for any unit vector $v\in\mathbb{R}^d$, we have

$$\begin{aligned}&|\mathbb{E}_{\mathcal{D}_0}[((w\cdot x-y)x-\nabla f(x,y,w,\kappa))\cdot v]|\\&=|\mathbb{E}_{x\sim\mathcal{D}_0}[\mathbb{E}_{\eta\sim\mathcal{D}_0}[((w\cdot x-y)x-\nabla f(x,y,w,\kappa))\cdot v]]|\\&\le\mathbb{E}_{x\sim\mathcal{D}_0}[|\mathbb{E}_{\eta\sim\mathcal{D}_0}[((w\cdot x-y)x-\nabla f(x,y,w,\kappa))\cdot v]|]\\&\le\mathbb{E}_{x\sim\mathcal{D}_0}[2|(w-w_0)\cdot x|\cdot|x\cdot v|(Z+\Pr(\eta>\kappa/2))]\\&\le 2\mathbb{E}_{\mathcal{D}_0}[Z\cdot|(w-w_0)\cdot x|\cdot|x\cdot v|]+2\Pr(\eta>\kappa/2)\mathbb{E}_{\mathcal{D}_0}[|(w-w_0)\cdot x|\cdot|x\cdot v|],\quad(6.15)\end{aligned}$$

here the second last inequality follows from Equation (6.13) and Equation (6.14). Next, we bound the two terms on the right one by one. We start with the first term:

$$\begin{aligned}\mathbb{E}_{\mathcal{D}_0}[Z \cdot |(x \cdot (w - w_0))(x \cdot v)|] &\overset{\text{(a)}}{\leq} \left(\mathbb{E}[(Z)^2] \cdot \mathbb{E}_{\mathcal{D}_0}[(x \cdot (w - w_0))^2 (x \cdot v)^2]\right)^{1/2} \\ &\overset{\text{(b)}}{\leq} \left(\mathbb{E}[Z] \cdot \mathbb{E}_{\mathcal{D}_0}[(x \cdot (w - w_0))^4]^{1/2} \mathbb{E}_{\mathcal{D}_0}[(x \cdot v)^4]^{1/2}\right)^{1/2} \\ &\overset{\text{(c)}}{\leq} \left(\mathbb{E}[Z] \cdot C \mathbb{E}_{\mathcal{D}_0}[(x \cdot (w - w_0))^2] \mathbb{E}_{\mathcal{D}_0}[(x \cdot v)^2]\right)^{1/2} \\ &\overset{\text{(d)}}{\leq} \left(C C_1 \Pr[|x \cdot (w - w_0)| \geq \kappa/2] \cdot \mathbb{E}_{\mathcal{D}_0}[(x \cdot (w - w_0))^2]\right)^{1/2},\end{aligned} \tag{6.16}$$

where (a) used the Cauchy-Schwarz inequality, (b) used the fact that Z is an indicator random variable, hence, $Z^2 = Z$, and the Cauchy-Schwarz inequality, (c) uses $L4 - L2$ hypercontractivity, and (d) follows from the definition of Z and the assumption that $\|\Sigma_0\| \leq C_1$.

Similarly, we can show that

$$\mathbb{E}_{\mathcal{D}_0}[|(w - w_0) \cdot x| \cdot |x \cdot v|] \leq \left(C_1 \mathbb{E}_{\mathcal{D}_0}[(x \cdot (w - w_0))^2]\right)^{1/2}. \tag{6.17}$$

Applying the Markov inequality to η^2 we get:

$$\Pr[|\eta| \geq \kappa/2] \leq \frac{\mathbb{E}_{\mathcal{D}_0}[\eta^2]}{(\kappa/2)^2}. \tag{6.18}$$

Similarly, applying the Markov inequality to $|x \cdot (w - w_0)|^4$ yields:

$$\Pr[|x \cdot (w - w_0)| \geq \kappa/2] \leq \frac{\mathbb{E}_{\mathcal{D}_0}[|x \cdot (w - w_0)|^4]}{(\kappa/2)^4} \leq \frac{C \mathbb{E}_{\mathcal{D}_0}[|x \cdot (w - w_0)|^2]^2}{(\kappa/2)^4}, \tag{6.19}$$

where the last inequality uses $L4 - L2$ hypercontractivity.

Combining Equations (6.15), (6.16), (6.17), (6.18) and (6.19), we have

$$\begin{aligned}
&|\mathbb{E}_{\mathcal{D}_0}[((w\cdot x-y)x-\nabla f(x,y,w,\kappa))\cdot v]| \\
&\le \frac{8C\sqrt{C_1}(\mathbb{E}_{\mathcal{D}_0}[(x\cdot(w-w_0))^2])^{3/2}}{\kappa^2}+\frac{8\sqrt{C_1}\mathbb{E}_{\mathcal{D}_0}[\eta^2](\mathbb{E}_{\mathcal{D}_0}[(x\cdot(w-w_0))^2])^{1/2}}{\kappa^2} \\
&\le \frac{8C\sqrt{C_1}(\mathbb{E}_{\mathcal{D}_0}[(x\cdot(w-w_0))^2])^{1/2}((\mathbb{E}_{\mathcal{D}_0}[(x\cdot(w-w_0))^2])+\mathbb{E}_{\mathcal{D}_0}[\eta^2]/C)}{\kappa^2} \\
&\overset{(a)}{\le} \frac{\epsilon(\mathbb{E}_{\mathcal{D}_0}[(x\cdot(w-w_0))^2])^{1/2}}{\sqrt{C_1}}\cdot\frac{((\mathbb{E}_{\mathcal{D}_0}[(x\cdot(w-w_0))^2])+\mathbb{E}_{\mathcal{D}_0}[\eta^2]/C)}{\mathbb{E}_{\mathcal{D}_0}[(y-x\cdot w)^2]} \\
&\overset{(b)}{\le} \epsilon\|w-w_0\|\cdot\frac{((\mathbb{E}_{\mathcal{D}_0}[(x\cdot(w-w_0))^2])+\mathbb{E}_{\mathcal{D}_0}[\eta^2])}{\mathbb{E}_{\mathcal{D}_0}[(y-x\cdot w)^2]} \\
&\overset{(c)}{=} \epsilon\|w-w_0\|,
\end{aligned}$$

here inequality (a) follows from the lower bound on κ^2 in theorem, inequality (b) follows from Assumption 1b and $C\ge 1$, and the last equality follows since $y-x\cdot w=x(w-w_0)+\eta$, and x and η are independent.

Note that the above bound holds for all unit vectors v, therefore,

$$\begin{aligned}
&\|\mathbb{E}_{\mathcal{D}_0}[((w\cdot x-y)x-\nabla f(x,y,w,\kappa))]\| \\
&\le \max_{\|v\|}\mathbb{E}_{\mathcal{D}_0}[((w\cdot x-y)x-\nabla f(x,y,w,\kappa))]\cdot v\le\epsilon\|w-w_0\|.
\end{aligned}$$

■

6.9 Estimation of clipping parameter

In round r, to set $\kappa\approx\sqrt{\mathbb{E}_{\mathcal{D}_0}[(y-x\cdot w)^2]/\epsilon}$ at point $w=\hat{w}^{(r)}$, the main algorithm 10 runs subroutine ClipEst 13 for $S^*=S_1^{b^*,(r)}$ and $w=\hat{w}^{(r)}$. Recall that $S_1^{b^*,(r)}$ is collection of i.i.d. samples from $\mathcal{D}_0$. Using these samples this subroutine estimates $\mathbb{E}_{\mathcal{D}_0}[(y-x\cdot w)^2]$ at $w=\hat{w}^{(r)}$ using the median of means and then use it to obtain κ in the desired range. The following theorem provides the guarantees on the estimation of κ by this subroutine.

Algorithm 13. ClipEst

```
Input: A collection of samples $S^*$ from $\mathcal{D}_0$, $w$, $\epsilon$, $\delta'$, $\sigma$, $C$, and $C_1$.
Output: clipping parameter $\kappa$
$T \leftarrow \Theta(\log 1/\delta')$
Divide $S^*$ into $T$ equal parts randomly, and denote them as $\{S_j^*\}_{j\in[T]}$
$\theta \leftarrow \text{Median}\Big\{\frac{1}{|S_j^*|}\sum_{(x,y)\in S_j^*}(x\cdot w - y)^2 : j \in [T]\Big\}$
$\kappa \leftarrow \sqrt{\frac{32(C+1)C_1(\theta+17\sigma^2)}{\epsilon}}$
Return $\kappa$
```

Theorem 114. *For $\epsilon > 0$, $T \geq \Omega(\log 1/\delta')$ and $|S^*| \geq 64C^2T$ and $w \in \mathbb{R}^d$. With probability $\geq 1-\delta'$, the clipping parameter κ returned by subroutine ClipEst satisfy,*

$$\sqrt{\tfrac{8(C+1)C_1\cdot\mathbb{E}_{\mathcal{D}_0}[(y-x\cdot w)^2]}{\epsilon}} \leq \kappa \leq 28\sqrt{\tfrac{2(C+1)C_1\left(\mathbb{E}_{\mathcal{D}_0}[((w-w_0)\cdot x)^2]+\sigma^2\right)}{\epsilon}}.$$

To prove the theorem, we will make use of the following lemma:

Lemma 115. *Let S be a collection of $m \geq 64C^2$ i.i.d. samples from $\mathcal{D}_0$ and $w \in \mathbb{R}^d$, then with probability at least $7/8$, the following holds:*

$$\frac{1}{4}\mathbb{E}_{\mathcal{D}_0}[(w\cdot x - y)^2] - 17\sigma^2 \leq \frac{1}{m}\sum_{(x,y)\in S}(y - w\cdot x)^2 \leq 3\mathbb{E}_{\mathcal{D}_0}[((w-w_0)\cdot x)^2] + 32\sigma^2.$$

Proof. We start by expanding the expression:

$$\begin{align}
&\frac{1}{m}\sum_{(x,y)\in S}(w\cdot x - y)^2 \tag{6.20}\\
&= \frac{1}{m}\sum_{(x,y)\in S}((w-w_0)\cdot x + (w_0\cdot x - y))^2 \notag\\
&= \frac{1}{m}\sum_{(x,y)\in S}\left(((w-w_0)\cdot x)^2 + 2((w-w_0)\cdot x)(w_0\cdot x - y) + (w_0\cdot x - y)^2\right) \notag\\
&\geq \frac{1}{m}\sum_{(x,y)\in S}\left(\frac{1}{2}((w-w_0)\cdot x)^2 - (w_0\cdot x - y)^2\right), \tag{6.21}
\end{align}$$

where the last inequality follows since for any a, b, we have $a^2 + 2ab + b^2 \geq a^2/2 - b^2$.

Similarly, we can show:

$$\frac{1}{m} \sum_{(x,y) \in S} (w \cdot x - y)^2 \leq \frac{1}{m} \sum_{(x,y) \in S} \big(2((w - w_0) \cdot x)^2 + 2(w_0 \cdot x - y)^2 \big). \tag{6.22}$$

Since S contains independent samples from $\mathcal{D}_0$, we have:

$$\mathbb{E}\left[\frac{1}{m} \sum_{(x,y) \in S} (((w - w_0) \cdot x)^2 \right] = \mathbb{E}_{\mathcal{D}_0}[((w - w_0) \cdot x)^2],$$

and

$$\begin{aligned} \text{Var}\left(\frac{1}{m} \sum_{(x,y) \in S} ((w - w_0) \cdot x)^2 \right) &= \frac{\text{Var}_{\mathcal{D}_0}(((w - w_0) \cdot x)^2)}{m} \\ &\leq \frac{\mathbb{E}_{\mathcal{D}_0}[((w - w_0) \cdot x)^4]}{m} \leq \frac{C\mathbb{E}_{\mathcal{D}_0}[((w - w_0) \cdot x)^2]^2}{m}, \end{aligned}$$

where the last inequality follows from $L4$-$L2$ hypercontractivity.

For any $a > 0$, using Chebyshev's inequality,

$$\Pr\left[\left| \frac{1}{m} \sum_{(x,y) \in S} ((w - w_0) \cdot x)^2 - \mathbb{E}_{\mathcal{D}_0}[((w - w_0) \cdot x)^2] \right| \geq a \frac{C\mathbb{E}_{\mathcal{D}_0}[((w - w_0) \cdot x)^2]}{\sqrt{m}} \right] \leq \frac{1}{a^2}. \tag{6.23}$$

Using the Markov inequality, for any $a > 0$, we have:

$$\Pr\left[\frac{1}{m} \sum_{(x,y) \in S} (w_0 \cdot x - y)^2 > a^2 \sigma^2 \right] \leq \frac{\mathbb{E}_{\mathcal{D}_0}[(w_0 \cdot x - y)^2]}{a^2 \sigma^2} \leq \frac{1}{a^2}. \tag{6.24}$$

By combining the equations above, we can derive the following inequality:

With probability $\geq 1 - \frac{2}{a^2}$, the following holds:

$$\frac{1}{2}\mathbb{E}_{\mathcal{D}_0}[((w - w_0) \cdot x)^2](1 - a\frac{C}{\sqrt{m}}) - a^2\sigma^2 \leq \frac{1}{m} \sum_{(x,y)\in S} (w \cdot x - y)^2$$
$$\leq 2\mathbb{E}_{\mathcal{D}_0}[((w - w_0) \cdot x)^2](1 + a\frac{C}{\sqrt{m}}) + 2a^2\sigma^2.$$

By choosing $a = 4$ and using $m \geq 64C^2$ in the above equation, we can conclude that with probability $\geq 1 - \frac{2}{a^2}$, the following holds:

$$\frac{1}{4}\mathbb{E}_{\mathcal{D}_0}[((w - w_0) \cdot x)^2] - 16\sigma^2 \leq \frac{1}{m} \sum_{(x,y)\in S} (w \cdot x - y)^2 \leq 3\mathbb{E}_{\mathcal{D}_0}[((w - w_0) \cdot x)^2] + 32\sigma^2.$$

Next, note that

$$\begin{aligned}\mathbb{E}_{\mathcal{D}_0}[(w \cdot x - y)^2] &= \mathbb{E}_{\mathcal{D}_0}[((w - w_0) \cdot x - (y - w_0 \cdot x))^2] \\ &\overset{(a)}{=} \mathbb{E}_{\mathcal{D}_0}[((w - w_0) \cdot x)^2] + \mathbb{E}_{\mathcal{D}_0}[(y - w_0 \cdot x)^2] \\ &\overset{(b)}{\leq} \mathbb{E}_{\mathcal{D}_0}[((w - w_0) \cdot x)^2] + \sigma^2,\end{aligned}$$

here (a) follows since x is independent of the output noise $y - w_0 \cdot x$, and (b) follows since the output noise $y - w_0 \cdot x$ is zero mean and has a variance at most σ^2. Combining the above two equations completes the proof. ■

Now we prove Theorem 114 using the above lemma:

Proof of Theorem 114. From the previous lemma and Chernoff bound it follows that with probability $\geq 1 - \delta'$,

$$\frac{1}{4}\mathbb{E}_{\mathcal{D}_0}[(w \cdot x - y)^2] - 17\sigma^2 \leq \theta \leq 3\mathbb{E}_{\mathcal{D}_0}[((w - w_0) \cdot x)^2] + 32\sigma^2.$$

Then bound on κ follows from the relation $\kappa = \sqrt{\frac{32(C+1)C_1(\theta+17\sigma^2)}{\epsilon}}$. ■

6.10 Subspace Estimation

Algorithm 14. GradSubEst

```
Input: A collection of medium batches $\widehat{B}$, $\kappa$, $w$, $\ell$
Output: A rank $\ell$ projection matrix.
For each $b \in \widehat{B}$ divide its samples $S^b$ into two equal random parts $S_1^b$ and $S_2^b$
$A \leftarrow \frac{1}{2|\widehat{B}|}\sum_{b\in\widehat{B}}\big(\nabla f(S_1^b, w, \kappa)\nabla f(S_2^b, w, \kappa)^\intercal + \nabla f(S_2^b, w, \kappa)\nabla f(S_1^b, w, \kappa)^\intercal\big)$
$U \leftarrow [u_1, u_2, ..., u_\ell]$, where $\{u_i\}$'s are top $\ell$ singular vectors of $A$
Return $UU^\intercal$
```

As a part of gradient estimation in step r, the main algorithm 10 uses subroutine GradSubEst for $\widehat{B} = B_s^{(r)}$ and $w = \hat{w}^{(r)}$. Recall that $B_s^{(r)}$ is a random subset of the collection of small batches B_s.

The purpose of this subroutine is to estimate a smaller subspace of $\mathbb{R}^d$ such that for distribution $\mathcal{D}_0$, the expectation of the projection of the clipped gradient onto this subspace closely approximates the true expectation of the clipped gradient, for distribution $\mathcal{D}_0$. This reduction to a smaller subspace helps reduce the number of medium-sized batches and their required length in the subsequent part of the algorithm.

The following theorem characterizes the final guarantee for subroutine GradSubEst.

Theorem 116. *Let p_0 denote the fraction of batches in $\widehat{B}$ that are sampled from $\mathcal{D}_0$. For any $\epsilon, \delta' > 0$, and $\widehat{B} = \Omega\Big(\frac{d}{\alpha_s\epsilon^2}\Big(\frac{1}{\alpha_s\epsilon^2} + \frac{C_2^2}{C_1}\Big)\log\frac{d}{\delta'}\Big)$, $p_0 \geq \alpha_s/2$ and $\ell \geq \min\{k, \frac{1}{2\alpha_s\epsilon^2}\}$, with probability $\geq 1 - \delta'$, the projection matrix $UU^\intercal$ returned by subroutine GradSubEst satisfy*

$$\|(I - UU^\intercal)\mathbb{E}_{\mathcal{D}_0}[\nabla f(x, y, w, \kappa)]\| \leq 4\epsilon\kappa\sqrt{C_1}.$$

The above theorem implies that the difference between the expectation of the clipped gradient and the expectation of projection of the clipped gradient for distribution $\mathcal{D}_0$ is small. Next, we present the description of the subroutine GradSubEst and provide a brief outline of the proof for the theorem before formally proving it in the subsequent subsection.

The subroutine divides samples in each batch $b \in \widehat{B}$ into two parts, namely S_1^b and S_2^b. Then it computes the clipped gradients $u^b := \nabla f(S_1^b, w, \kappa)$ and $v^b := \nabla f(S_2^b, w, \kappa)$. From linearity of expectation, for any i and batch b that contain i.i.d. samples from $\mathcal{D}_i$, $\mathbb{E}[u^b] = \mathbb{E}[v^b] = \mathbb{E}_{\mathcal{D}_i}[\nabla f(x, y, w, \kappa)]$. The subroutine defines $A = \sum_{b \in \widehat{B}} \frac{1}{2|\widehat{B}|} u^b (v^b)^\intercal + v^b (u^b)^\intercal$. Let p_i denote the fraction of batches in $\widehat{B}$ that have samples from $\mathcal{D}_i$. Then using the linearity of expectation, we have:

$$\mathbb{E}[A] = \sum_{i=0}^{k-1} p_i \mathbb{E}_{\mathcal{D}_i}[\nabla f(x, y, w, \kappa)] \mathbb{E}_{\mathcal{D}_i}[\nabla f(x, y, w, \kappa)]^\intercal.$$

It is evident that if the matrix U is formed by selecting the top k singular vectors of $\mathbb{E}[A]$, then the projection of $\mathbb{E}_{\mathcal{D}_0}[\nabla f(x, y, w, \kappa)]$ onto $UU^\intercal$ corresponds to itself, and the guarantee stated in the theorem holds. However, we do not have access to $\mathbb{E}[A]$, and furthermore, when the number of components k is large, it may be desirable to obtain a subspace of smaller size than k.

To address the first challenge, Theorem 117 in the next subsection shows that $\|A - \mathbb{E}[A]\|$ is small. This theorem permits the usage of A as a substitute for $\mathbb{E}[A]$. The clipping operation, introduced in the previous subsection, plays a crucial role in the proof of Theorem 117 by controlling the norm of the expectation and the covariance of the clipped gradient for other components, and the maximum length of clipped gradients across all components. This is crucial for obtaining a good bound on the number of small-size batches required. Additionally, the clipping operation ensures that the subroutine remains robust to arbitrary input-output relationships for other components.

Furthermore, the clipping operation assists in addressing the second challenge by ensuring a uniform upper bound on the norm of the expectation of all components, i.e., $\|\mathbb{E}_{\mathcal{D}_i}[\nabla f(x, y, w, \kappa)]\| \leq \mathcal{O}(\kappa)$. Leveraging this property, Lemma 118 demonstrates that it suffices to estimate the top $\approx 1/p_0$-dimensional subspace. Intuitively, this is because the infinitely many components can create at most approximately $1/p_0$ directions with weights greater than p_0, indicating that the direction of $\mathcal{D}_0$ must be present in the top $\Theta(1/p_0)$ subspace.

Since $\widehat{B} = B_s^{(r)}$ is obtained by randomly partitioning B_s into R subsets, and B_s contains a fraction of at least α_s batches with samples from $\mathcal{D}_0$, it holds with high probability that $p_0 \gtrsim \alpha_s$. Consequently, when $\ell \geq \min\{k, \Omega(\frac{1}{\alpha_s})\}$, the subspace corresponding to the top ℓ singular vectors of A satisfies the desired property in the Theorem 116.

We note that the construction of matrix A in subroutine GRADSUBEST is inspired by previous work [95]. However, while they employed it to approximate the k-dimensional subspace of the true regression vectors for all components, we focus exclusively on one distribution $\mathcal{D}_0$ at a time and recover a subspace such that, for distribution $\mathcal{D}_0$, the expectation of the projection of the clipped gradient on this subspace closely matches the true expectation of the clipped gradient.

It is worth noting that, in addition to repurposing the subroutine from [95], we achieve four significant improvements:

1) A more meticulous statistical analysis and the use of clipping enable our algorithm to handle heavy-tailed distributions for both noise and input distributions. 2) Clipping also facilitates the inclusion of arbitrary input-output relationships for other components. The next two improvements are attributed to an improved linear algebraic analysis. Specifically, our Lemma 118 enhances the matrix perturbation bounds found in [] and [95]. These enhancements enable us to: 3) Provide meaningful guarantees even when the number of components k is very large, 4) reduce the number of batches required when the distance between the regression vectors is small.

6.10.1 Proof of Theorem 116

To prove Theorem 116, in the following theorem, we will first demonstrate that the term $\|A - \mathbb{E}[A]\|$ is small when given enough batches.

Theorem 117. *For $0 \leq i \leq k-1$, let $z_i = \mathbb{E}_{\mathcal{D}_i}[\nabla f(x, y, w, \kappa)]$, and p_i denote the fraction of batches in $\widehat{B}$ that have samples from $\mathcal{D}_i$. For any $\epsilon, \delta' > 0$, and $\widehat{B} = \Omega\Big(\frac{d}{\alpha_s \epsilon^2}\Big(\frac{1}{\alpha_s \epsilon^2} + \frac{C_2^2}{C_1}\Big) \log \frac{d}{\delta'}\Big)$,* *with probability at least* $1-\delta'$*,*

$$\left\|A-\sum_{i=0}^{k-1} p_i z_i z_i^{\intercal}\right\| \le \alpha_s \epsilon^2 \kappa^2 C_1,$$

where A *is the matrix defined in subroutine* GradSubEst*.*

Proof. Let $Z^b := \nabla f(S_1^b, w, \kappa)\nabla f(S_2^b, w, \kappa)^{\intercal}$.

Note that

$$A = \frac{1}{2|\widehat{B}|}\sum_{b\in\widehat{B}}(Z^b + (Z^b)^{\intercal}).$$

Then, from the triangle inequality, we have:

$$\begin{aligned}\left\|A-\sum_{i=0}^{k-1} p_i z_i z_i^{\intercal}\right\| &\le \frac{1}{2}\left\|\frac{1}{|\widehat{B}|}\sum_{b\in\widehat{B}} Z^b - \sum_{i=0}^{k-1} p_i z_i z_i^{\intercal}\right\| + \frac{1}{2}\left\|\frac{1}{|\widehat{B}|}\sum_{b\in\widehat{B}}(Z^b)^{\intercal} - \sum_{i=0}^{k-1} p_i z_i z_i^{\intercal}\right\| \\ &= \left\|\frac{1}{|\widehat{B}|}\sum_{b\in\widehat{B}} Z^b - \sum_{i=0}^{k-1} p_i z_i z_i^{\intercal}\right\|.\end{aligned}$$

For a batch b sampled from distribution $\mathcal{D}_i$, we have:

$$\begin{aligned}\mathbb{E}[Z^b] &= \mathbb{E}[\nabla f(S_1^b, w, \kappa)\nabla f(S_2^b, w, \kappa)^{\intercal}] \\ &= \mathbb{E}[\nabla f(S_1^b, w, \kappa)]\mathbb{E}[\nabla f(S_2^b, w, \kappa)^{\intercal}] \\ &= \mathbb{E}_{\mathcal{D}_i}[\nabla f(x, y, w, \kappa)]\mathbb{E}_{\mathcal{D}_i}[\nabla f(x, y, w, \kappa)^{\intercal}] = z_i z_i^{\intercal},\end{aligned}$$

where the second inequality follows since samples in S_1^b and S_2^b are independent, and the third equality follows from the linearity of expectation.

It follows that

$$\frac{1}{|\widehat{B}|}\sum_{b\in\widehat{B}}\mathbb{E}[Z^b] = \sum_{i=0}^{k-1} p_i z_i z_i^{\intercal},$$

and

$$\left\|A-\sum_{i=0}^{k-1}p_iz_iz_i^{\mathsf{T}}\right\| \leq \left\|\frac{1}{|\widehat{B}|}\sum_{b\in\widehat{B}}Z^b-\frac{1}{|\widehat{B}|}\sum_{b\in\widehat{B}}\mathbb{E}[Z^b]\right\|. \tag{6.25}$$

To complete the proof, we will prove a high probability bound on the term on the right by applying the Matrix Bernstein inequality. To apply this inequality, we first upper bound $|Z^b|$ as follows:

$$\|Z^b\| = \|\nabla f(S_1^b,w,\kappa)\nabla f(S_2^b,w,\kappa)^{\mathsf{T}}\| \leq \|\nabla f(S_1^b,w,\kappa)\|\cdot\|\nabla f(S_2^b,w,\kappa)^{\mathsf{T}}\|.$$

From item 3 in Lemma 111, we have $\|\nabla f(S_1^b,w,\kappa)\| \leq \kappa C_2\sqrt{d}$ almost surely, and $\|\nabla f(S_2^b,w,\kappa)\| \leq \kappa C_2\sqrt{d}$ almost surely. It follows that $\|Z^b\| \leq \kappa^2C_2^2d$. Therefore, $\|Z^b-\mathbb{E}[Z^b]\| \leq 2\kappa^2C_2^2d$.

Next, we provide an upper bound for $\big\|\mathbb{E}\big[\big(\sum_{b\in\widehat{B}}(Z^b-\mathbb{E}[Z^b])\big)\big(\sum_{b\in\widehat{B}}(Z^b-\mathbb{E}[Z^b])\big)^{\mathsf{T}}\big]\big\|$:

$$\begin{aligned}
&\big\|\mathbb{E}\big[\big(\textstyle\sum_{b\in\widehat{B}}(Z^b-\mathbb{E}[Z^b])\big)\big(\sum_{b\in\widehat{B}}(Z^b-\mathbb{E}[Z^b])\big)^{\mathsf{T}}\big]\big\|\\
&=\big\|\mathbb{E}\big[\textstyle\sum_{b\in\widehat{B}}(Z^b-\mathbb{E}[Z^b])(Z^b-\mathbb{E}[Z^b])^{\mathsf{T}}\big]\big\|\\
&\leq|\widehat{B}|\max_{b\in\widehat{B}}\big\|\mathbb{E}\big[(Z^b-\mathbb{E}[Z^b])(Z^b-\mathbb{E}[Z^b])^{\mathsf{T}}\big]\big\|\\
&\leq|\widehat{B}|\max_{b\in\widehat{B}}\big\|\mathbb{E}\big[(Z^b(Z^b)^{\mathsf{T}}\big]\big\|\\
&\leq|\widehat{B}|\max_{b\in\widehat{B}}\big(\mathbb{E}[\|\nabla f(S_2^b,w,\kappa)\|^2]\cdot\big\|\mathbb{E}\big[\nabla f(S_1^b,w,\kappa)\nabla f(S_1^b,w,\kappa)^{\mathsf{T}}\big]\big\|\big)\\
&\leq|\widehat{B}|\max_{b\in\widehat{B},u:\|u\|=1}\big(\mathbb{E}[\|\nabla f(S_2^b,w,\kappa)\|^2]\cdot\big\|\mathbb{E}\big[(\nabla f(S_1^b,w,\kappa)\cdot u)^2\big]\big\|\big).
\end{aligned}$$

From item 4 and item 5 in lemma 111, we have:

$$\mathbb{E}[\|\nabla f(S_2^b,w,\kappa)\|^2] \leq C_1\kappa^2d,$$

and

$$\left\|\mathbb{E}\left[(\nabla f(S_1^b, w, \kappa) \cdot u)^2\right]\right\| \leq \kappa^2 C_1.$$

Combining these two bounds, wee obtain:

$$\left\|\mathbb{E}\left[\left(\sum_{b\in\widehat{B}}(Z^b - \mathbb{E}[Z^b])\right)\left(\sum_{b\in\widehat{B}}(Z^b - \mathbb{E}[Z^b])\right)^{\intercal}\right]\right\| \leq |\widehat{B}|d\kappa^4{C_1}^2.$$

Due to symmetry, the same bound holds for $\left\|\mathbb{E}\left[\left(\sum_{b\in\widehat{B}}(Z^b - \mathbb{E}[Z^b])\right)^{\intercal}\left(\sum_{b\in\widehat{B}}(Z^b - \mathbb{E}[Z^b])\right)\right]\right\|$.

Finally, by applying the Matrix Bernstein inequality, we have:

$$\Pr\left[\left\|\frac{1}{|\widehat{B}|}\sum_{b\in\widehat{B}}(Z^b - \mathbb{E}[Z^b])\right\| \geq \alpha_s\epsilon^2\kappa^2 C_1\right] \leq 2d\exp\left\{-\frac{|\widehat{B}|^2\theta^2}{|\widehat{B}|d\kappa^4{C_1}^2 + |\widehat{B}|\theta(2C_2^2\kappa^2 d)}\right\}.$$

For $\widehat{B} = \Omega\left(\frac{d}{\alpha_s\epsilon^2}\left(\frac{1}{\alpha_s\epsilon^2} + \frac{C_2^2}{C_1}\right)\log\frac{d}{\delta'}\right)$, the quantity on the right-hand side is bounded by δ'.

Therefore, with probability at least $1 - \delta'$, we have:

$$\left\|\frac{1}{|\widehat{B}|}\sum_{b\in\widehat{B}}(Z^b - \mathbb{E}[Z^b])\right\| \leq \alpha_s\epsilon^2\kappa^2 C_1.$$

Combining the above equation with Equation (6.25) completes the proof of the Theorem. ■

In the proof of Theorem 116, we will utilize the following general linear algebraic result:

Lemma 118. *For $z_0, z_1, ..., z_{k-1} \in \mathbb{R}^d$ and a probability distribution $(p_0, p_1, ..., p_{k-1})$ over k elements, let $Z = \sum_{i=0}^{k-1} p_i z_i z_i^{\intercal}$. For a symmetric matrix M and $\ell > 0$, let $u_1, u_2, .., u_\ell$ be top ℓ singular vectors of M and let $U = [u_1, u_2, ..., u_\ell] \in \mathbb{R}^{d\times\ell}$, then we have:*

$$\|(I - UU^{\intercal})z_0\|^2 \leq \begin{cases} \frac{2(\ell+1)\|M-Z\|+\max_j \|z_j\|^2}{(\ell+1)p_0} & \ell < k \\ \frac{2\|M-Z\|}{p_0} & \text{if } \ell \geq k. \end{cases}$$

Lemma 118 provides a bound on the preservation of the component z_0 by the subspace spanned by the top-ℓ singular vectors of a symmetric matrix M. This bound is expressed in terms of the spectral distance between matrices Z and M, the maximum norm of any z_i, and the weight of the component corresponding to z_0 in Z. The proof of Lemma 118 can be found in Section 6.14.

Utilizing Lemma 118 in conjunction with Theorem 117, we proceed to prove Theorem 116.

Proof of Theorem 116. From Lemma 118, we have the following inequality:

$$\|(I - UU^\intercal)\mathbb{E}_{\mathcal{D}_0}[\nabla f(x,y,w,\kappa)]\|^2 \leq \begin{cases} \frac{2(\ell+1)\|A - \sum_{i=0}^{k-1} p_i z_i z_i^\intercal\| + \max_j \|\mathbb{E}_{\mathcal{D}_j}[\nabla f(x,y,w,\kappa)]\|^2}{(\ell+1)p_0} & \ell < k \\ \frac{2\|A - \sum_{i=0}^{k-1} p_i z_i z_i^\intercal\|}{p_0} & \text{if } \ell \geq k. \end{cases}$$

By applying Theorem 117 and utilizing item 1 of Lemma 111, it follows that with a probability of at least $1 - \delta'$, we have:

$$\|(I - UU^\intercal)\mathbb{E}_{\mathcal{D}_0}[\nabla f(x,y,w,\kappa)]\|^2 \leq \begin{cases} \frac{2\alpha_s \epsilon^2 \kappa^2 C_1}{p_0} + \frac{\kappa^2 C_1}{(\ell+1)p_0} & \ell < k \\ \frac{2\alpha_s \epsilon^2 \kappa^2 C_1}{p_0} & \text{if } \ell \geq k. \end{cases}$$

The theorem then follows by using $p_0 \geq \alpha_s/2$ and $\ell \geq \min\{k, \frac{1}{2\alpha_s \epsilon^2}\}$. ■

6.11 Grad Estimation

Recall that in gradient estimation for step r, Algorithm 10 utilizes the subroutine GradSubEst to find a projection matrix $P^{(r)}$ for an ℓ-dimensional subspace. In the previous section, we showed that the difference between the expectation of the clipped gradient and the expectation of projection of the clipped gradient on the subspace for distribution $\mathcal{D}_0$ is small. Therefore, it suffices to estimate the expectation of projection of the clipped gradient on the subspace.

The main algorithm 10 passes the medium-sized batches $\widehat{B} = B_m^{(r)}$, the ℓ-dimensional projection matrix $P = P^{(r)}$, and a collection of i.i.d. samples $S^* = S_2^{b^*,(r)}$ from $\mathcal{D}_0$ to the subroutine GradEst. Here, $B_m^{(r)}$ is a random subset of the collection of medium-sized batches B_m.

The purpose of the GradEst subroutine is to estimate the expected value of the projection of the clipped gradient onto the ℓ-dimensional subspace defined by the projection matrix P. Since the subroutine operates on a smaller ℓ-dimensional subspace, the minimum batch size required for the batches in B_m and the number of batches required depend on ℓ rather than d.

The following theorem characterizes the final guarantee for the GradEst subroutine:

Theorem 119. *For subroutine GradEst, let n_m denote the length of the smallest batch in $\widehat{B}$, N denote the number of batches in $\widehat{B}$ that has samples from $\mathcal{D}_0$ and P be a projection matrix for some ℓ dimensional subspace of $\mathbb{R}^d$. If $T_1 \geq \Omega(\log \frac{|\widehat{B}|}{\delta'})$, $T_2 \geq \Omega(\log \frac{1}{\delta'})$, $n_m \geq 4T_1\Omega(\frac{\sqrt{\ell}}{\epsilon^2})$, $|S^*| \geq 2T_1\Omega(\frac{\sqrt{\ell}}{\epsilon^2})$ samples, and $N \cdot n_m \geq T_2\Omega(\frac{\ell}{\epsilon^2})$, then with probability $\geq 1 - 2\delta'$ the estimate Δ returned by subroutine GradEst satisfy*

$$\|\Delta - \mathbb{E}_{\mathcal{D}_0}[P\nabla f(x, y, w, \kappa)]\| \leq 9\epsilon\kappa\sqrt{C_1}.$$

The above theorem implies that when the length of medium-sized batches is $\tilde{\Omega}(\sqrt{\ell})$ and the number of batches in $\widehat{B}$ containing samples from $\mathcal{D}_0$ is $\tilde{\Omega}(\ell)$, the GradEst subroutine provides a reliable estimate of the projection of the clipped gradient onto the ℓ-dimensional subspace defined by the projection matrix P.

Next, we provide a description of the GradEst subroutine and present a brief outline of the proof for the theorem before formally proving it in the subsequent subsection.

In the GradEst subroutine, the first step is to divide the samples in each batch of $\widehat{B}$ into two equal parts. By utilizing the first half of the samples in a batch b along with the samples S^*, it estimates whether the expected values of the projection of the clipped gradient for $\mathcal{D}_0$ and the distribution used for the samples in b are close or not. With high probability, the algorithm retains all the batches from $\mathcal{D}_0$ while rejecting batches from distributions where the difference between the two expectations is large. To achieve this with an ℓ-dimensional subspace, we require $\tilde{\Omega}(\sqrt{\ell})$ samples in each batch (see Lemma 124).

Following the rejection process, the GRADEST subroutine proceeds to estimate the projection of the clipped gradients within this ℓ-dimensional subspace using the second half of the samples from the retained batches. To estimate the gradient accurately in the ℓ-dimensional subspace, $\Omega(\ell)$ samples are sufficient (see Lemma 125). To obtain guarantees with high probability, the procedure employs the median of means approach, both for determining which batches to keep and for estimation using the retained batches.

We prove the theorem formally in the next subsection.

6.11.1 Proof of Theorem 119

The following lemma provides an upper bound on the covariance of the projection of the clipped gradients.

Lemma 120. *Consider a collection S of m i.i.d. samples from distribution $\mathcal{D}_i$. For $\kappa > 0$, $w \in \mathbb{R}^d$ and a projection matrix P for an ℓ dimensional subspace of $\mathbb{R}^d$, we have*

$$\mathbb{E}[P\nabla f(S, w, \kappa)] = \mathbb{E}_{\mathcal{D}_i}[P\nabla f(x, y, w, \kappa)],$$

and $\|Cov(P\nabla f(S, w, \kappa))\| \leq \frac{2\kappa^2}{m} C_1$ and $Tr\left(Cov(P\nabla f(S, w, \kappa))\right) \leq \ell\|Cov(P\nabla f(S, w, \kappa))\|$.

Proof. Note that,

$$\mathbb{E}\big[P\big(\nabla f(S, w, \kappa)\big)\big] = P\mathbb{E}[\nabla f(S, w, \kappa)]] = P\mathbb{E}_{\mathcal{D}_i}[\nabla f(x, y, w, \kappa)] = \mathbb{E}_{\mathcal{D}_i}[P\nabla f(x, y, w, \kappa)],$$

where the second-to-last equality follows from Lemma 111.

This proves the first part of the lemma. To prove the second part, we bound the norm of the covariance matrix:

$$\begin{aligned}\text{Cov}(P\nabla f(S,w,\kappa)) &= \max_{\|u\|\leq 1} \text{Var}(u^\intercal P\nabla f(S,w,\kappa)) \\ &= \max_{\|v\|\leq 1} \text{Var}(v^\intercal \nabla f(S,w,\kappa)) \\ &\leq \|\text{Cov}(\nabla f(S,w,\kappa))\| \\ &\leq \frac{\kappa^2}{m} C_1,\end{aligned}$$

where the last inequality follows from Lemma 111. Similarly,

$$\text{Cov}(P\nabla f(S',w,\kappa)) \leq \frac{\kappa^2}{m} C_1.$$

Finally, since random vector $P\nabla f(S',w,\kappa)$ lies in ℓ dimensional subspace of $\mathbb{R}^d$, corresponding to projection matrix P, hence its covariance matrix has rank $\leq \ell$. Hence, the relation $\text{Tr}\left(\text{Cov}(P\nabla f(S,w,\kappa))\right) \leq \ell\|\text{Cov}(P\nabla f(S,w,\kappa))\|$ follows immediately. ■

The following corollary is a simple consequence of the previous lemma:

Corollary 121. *Consider two collections S and S' each consisting of m i.i.d. samples from distributions $\mathcal{D}_i$ and $\mathcal{D}_0$, respectively. For $\kappa > 0$, $w \in \mathbb{R}^d$ and a projection matrix P for an ℓ dimensional subspace of $\mathbb{R}^d$, let $z = P\big(\nabla f(S,w,\kappa) - \nabla f(S',w,\kappa)\big)$, we have:*

$$\mathbb{E}[z] = \mathbb{E}_{\mathcal{D}_i}[P\nabla f(x,y,w,\kappa)] - \mathbb{E}_{\mathcal{D}_0}[P\nabla f(x,y,w,\kappa)],$$

and $\|Cov(z)\| \leq \frac{4\kappa^2}{m} C_1$ and $Tr\left(Cov(z)\right) \leq \ell\|Cov(z)\|$.

Proof. The expression for $\mathbb{E}[z]$ can be obtained from the previous lemma and the linearity of expectation.

To prove the second part, we bound the norm of the covariance matrix of z.

$$\begin{aligned}\text{Cov}(z) &= \text{Cov}(P\big(\nabla f(S, w, \kappa) - \nabla f(S', w, \kappa)\big)) \\ &\leq 2(\text{Cov}(P\nabla f(S, w, \kappa)) + \text{Cov}(P\nabla f(S', w, \kappa))).\end{aligned}$$

Using the bounds from the previous lemma, we can conclude that $\|\text{Cov}(z)\| \leq \frac{4\kappa^2}{m} C_1$.

Finally, since the random vector z lies in the ℓ-dimensional subspace of $\mathbb{R}^d$ defined by the projection matrix P, its covariance matrix has rank $\leq \ell$. Therefore, we have Tr, $(\text{Cov}(z)) \leq \ell \|\text{Cov}(z)\|$. ■

The following theorem bounds the variance of the dot product of two independent random vectors. It will be helpful in upper bounding the variance of ζ_j^b (defined in subroutine GradEst).

Theorem 122. *For any two independent random vectors z_1 and z_2, we have:*

$$\mathit{Var}(z_1 \cdot z_2) \leq 3\mathit{Tr}(\mathit{Cov}(z_1)) \cdot \|\mathit{Cov}(z_2)\| + 3\|\mathbb{E}[z_1]\|^2 \|\mathit{Cov}(z_2)\| + 3\|\mathbb{E}[z_2]\|^2 \|\mathit{Cov}(z_1)\|.$$

Proof. We start by expanding the variance expression:

$$\begin{aligned}&\text{Var}(z_1 \cdot z_2) \\ &= \text{Var}(z_1 \cdot z_2 - \mathbb{E}[z_1 \cdot z_2]) \\ &= \text{Var}(z_1 \cdot z_2 - \mathbb{E}[z_1] \cdot \mathbb{E}[z_2]) \\ &= \text{Var}((z_1 - \mathbb{E}[z_1]) \cdot (z_2 - \mathbb{E}[z_2]) + \mathbb{E}[z_1] \cdot z_2 + \mathbb{E}[z_2] \cdot z_1) \\ &\leq 3\text{Var}((z_1 - \mathbb{E}[z_1]) \cdot (z_2 - \mathbb{E}[z_2])) + 3\text{Var}(\mathbb{E}[z_1] \cdot z_2) + 3\text{Var}(\mathbb{E}[z_2] \cdot z_1) \\ &= 3\text{Var}((z_1 - \mathbb{E}[z_1]) \cdot (z_2 - \mathbb{E}[z_2])) + 3\mathbb{E}[z_1]^\intercal \text{Cov}(z_2) \mathbb{E}[z_1] + \mathbb{E}[z_2]^\intercal \text{Cov}(z_1) \mathbb{E}[z_2] \\ &\leq 3\text{Var}((z_1 - \mathbb{E}[z_1]) \cdot (z_2 - \mathbb{E}[z_2])) + 3\|\mathbb{E}[z_1]\|^2 \|\text{Cov}(z_2)\| + 3\|\mathbb{E}[z_2]\|^2 \|\text{Cov}(z_1)\|.\end{aligned}$$

To complete the proof, we bound the first term in the last expression:

$$\begin{aligned}
\text{Var}((z_1 - \mathbb{E}[z_1]) \cdot (z_2 - \mathbb{E}[z_2])) &= \mathbb{E}[((z_1 - \mathbb{E}[z_1]) \cdot (z_2 - \mathbb{E}[z_2]))^2] \\
&= \mathbb{E}[(z_1 - \mathbb{E}[z_1])^\intercal \text{Cov}(z_2)(z_1 - \mathbb{E}[z_1])] \\
&\leq \mathbb{E}[\|z_1 - \mathbb{E}[z_1]\|^2] \cdot \|\text{Cov}(z_2)\| \\
&= \text{Tr}(\text{Cov}(z_1)) \cdot \|\text{Cov}(z_2)\|.
\end{aligned}$$

■

Using the two previous results, we can establish a bound on the expectation and variance of ζ_j^b.

Lemma 123. *In subroutine* GRADEST*, let P be a projection matrix of an ℓ dimensional subspace. Suppose S_j^* has $\geq m$ i.i.d. samples from $\mathcal{D}_0$ and, $S_{1,j}^b$ and $S_{1,j+T_1}^b$ have $\geq m$ i.i.d. samples from $\mathcal{D}_i$ for some $i \in \{0, 1, ..., k-1\}$. Than we have:*

$$\mathbb{E}[\zeta_j^b] = \|\mathbb{E}_{\mathcal{D}_i}[P\nabla f(x, y, w, \kappa)] - \mathbb{E}_{\mathcal{D}_0}[P\nabla f(x, y, w, \kappa)]\|^2$$

and

$$\textit{Var}(\zeta_j^b) \leq \frac{48}{m^2}\kappa^4 \ell {C_1}^2 + \frac{24}{m}\kappa^2 \mathbb{E}[\zeta_j^b] C_1.$$

Proof. Let $z_1 = P\big(\nabla f(S_{1,j}^b, w, \kappa) - \nabla f(S_j^*, w, \kappa)\big)$ and $z_2 = P\big(\nabla f(S_{1,T_1+j}^b, w, \kappa) - \nabla f(S_{T_1+j}^*, w, \kappa)\big)$.

From Corollary 121, we know that

$$\mathbb{E}[z_1] = \mathbb{E}[z_2] = \mathbb{E}_{\mathcal{D}_i}[P\nabla f(x, y, w, \kappa)] - \mathbb{E}_{\mathcal{D}_0}[P\nabla f(x, y, w, \kappa)],$$

$$\text{Cov}(z_1) = \text{Cov}(z_2) = \frac{4\kappa^2}{m} C_1$$

and

$$\text{Tr}(\text{Cov}(z_1)) = \text{Tr}(\text{Cov}(z_2)) = \frac{4}{m}\ell\kappa^2 C_1.$$

Note that $\zeta_j^b = z_1 \cdot z_2$. Then bound on the variance of ζ_j^b follows by combining the above bounds with Theorem 122. Finally, the expected value of ζ_j^b is:

$$\mathbb{E}[\zeta_j^b] = \mathbb{E}[z_1] \cdot \mathbb{E}[z_2] = \|\mathbb{E}[z_1]\|^2.$$

■

The following lemma provides a characterization of the minimum batch length in $\widehat{B}$ and the size of the collection S^* required for successful testing in subroutine GradEst.

Lemma 124. *In subroutine GradEst, let P be a projection matrix of an ℓ dimensional subspace, $T_1 \geq \Omega(\log \frac{|\widehat{B}|}{\delta'})$, and each batch $b \in \widehat{B}$ has at least $|S^b| \geq 4T_1\Omega(\frac{\sqrt{\ell}}{\epsilon^2})$ samples, and $|S^*| = 2T_1\Omega(\frac{\sqrt{\ell}}{\epsilon^2})$. Then with probability $\geq 1 - \delta'$, the subset $\tilde{B}$ in subroutine GradEst satisfy the following:*

1. *$|\tilde{B}|$ retains all the batches in $\widehat{B}$ that had samples from $\mathcal{D}_0$.*

2. *$\tilde{B}$ does not contain any batch that had samples from $\mathcal{D}_i$ if i is such that*

$$\|\mathbb{E}_{\mathcal{D}_i}[P\nabla f(x, y, w, \kappa)] - \mathbb{E}_{\mathcal{D}_0}[P\nabla f(x, y, w, \kappa)]\| > 2\epsilon\kappa\sqrt{C_1}.$$

Proof. The lower bound on $|S^b|$ in the lemma ensures that for each batch b and all $j \in [2T_1]$, we have $S_{1,j}^b = \Omega(\frac{\sqrt{\ell}}{\epsilon^2})$, and the lower bound on $|S^{|}$ ensures that for all $j \in [2T_1]$, $S_j = \Omega(\frac{\sqrt{\ell}}{\epsilon^2})$.

First, consider the batches that have samples from the distribution $\mathcal{D}_0$.

For any such batch b and $j \in [T_1]$, from Lemma 123, we have $\mathbb{E}[\zeta_j^b] = 0$ and $\text{Var}(\zeta_j^b) = \mathcal{O}(\epsilon^4\kappa^2{C_1}^2)$. Therefore, for $T_1 \geq \Omega(\log \frac{|\widehat{B}|}{\delta'})$, it follows that with probability $\geq 1 - \delta'/2$ for every batch $b \in \widehat{B}$ that has samples from $\mathcal{D}_0$ the median of $\{\zeta_j^b\}_{j\in[T_1]}$ will be less than $\epsilon^2\kappa^2 C_1$, and it will be retained in $\tilde{B}$. This completes the proof of the first part.

Next, consider the batches that have samples from any distribution $\mathcal{D}_i$ for which

$$\|\mathbb{E}_{\mathcal{D}_i}[P\nabla f(x,y,w,\kappa)] - \mathbb{E}_{\mathcal{D}_0}[P\nabla f(x,y,w,\kappa)]\| > 2\epsilon\kappa\sqrt{C_1}.$$

For any such batch b and $j \in [T_1]$, according to Lemma 123, we have $\mathbb{E}[\zeta_j^b] \geq 4\epsilon^2\kappa^2 C_1$ and $\mathrm{Var}(\zeta_j^b) = \mathcal{O}(\mathbb{E}[\zeta_j^b]^2)$. Hence, for $T_1 \geq \Omega(\log \frac{|\widehat{B}|}{\delta'})$, it follows that with probability at least $1-\delta'/2$, the median of $\{\zeta_j^b\}_{j\in[T_1]}$ for every batch will be greater than $\epsilon^2\kappa^2 C_1$, and those batches will not be included in $\tilde{B}$. This completes the proof of the second part. ■

The following theorem characterizes the number of samples required in $\tilde{B}$ for an accurate estimation of Δ.

Lemma 125. *Suppose the conclusions in Lemma 124 hold for $\tilde{B}$ defined in subroutine GradEst, $T_2 \geq \Omega(\log \frac{1}{\delta'})$, each batch $b \in \tilde{B}$ has size $\geq n_m$, and $|\tilde{B}| \cdot n_m \geq 2T_2\Omega(\frac{\ell}{\epsilon^2})$, then with probability $\geq 1-\delta'$ the estimate Δ returned by subroutine GradEst satisfy*

$$\|\Delta - \mathbb{E}_{\mathcal{D}_0}[P\nabla f(x,y,w,\kappa)]\| \leq 9\epsilon\kappa\sqrt{C_1}.$$

Proof. Recall that in subroutine GradEst, we defined

$$\Delta_i = \frac{1}{|\tilde{B}|}\sum_{b\in\tilde{B}} P\nabla f(S_{2,i}^b, w, \kappa).$$

Let $z_i^b = P\nabla f(S_{2,i}^b, w, \kappa)$. From Lemma 124, for all $b \in \tilde{B}$, we have

$$\left\|\mathbb{E}[z_i^b] - \mathbb{E}_{\mathcal{D}_0}[P\nabla f(x,y,w,\kappa)]\right\| \leq 2\epsilon\kappa\sqrt{C_1}.$$

Therefore,

$$\begin{aligned}\|\mathbb{E}[\Delta_i] - \mathbb{E}_{\mathcal{D}_0}[P\nabla f(x,y,w,\kappa)]\| &= \left\| \frac{1}{|\tilde{B}|}\sum_{b\in\tilde{B}} \mathbb{E}[z_i^b] - \mathbb{E}_{\mathcal{D}_0}[P\nabla f(x,y,w,\kappa)] \right\| \\ &\le \max_{b\in\tilde{B}} \left\|\mathbb{E}[z_i^b] - \mathbb{E}_{\mathcal{D}_0}[P\nabla f(x,y,w,\kappa)]\right\| \le 2\epsilon\kappa\sqrt{C_1}.\end{aligned} \tag{6.26}$$

Next, from Lemma 120,

$$\|\text{Cov}(z_i^b)\| \le \frac{\kappa^2}{|S_{2,i}^b|}C_1 = \frac{T_2\kappa^2}{|S_2^b|}C_1 = \frac{2T_2\kappa^2}{|S^b|}C_1 \le \frac{2T_2\kappa^2 C_1}{\min_{b\in\tilde{B}}|S^b|}, \tag{6.27}$$

where the two equalities follow because for all batches $b \in \tilde{B}$, $|S_{2,i}^b| = |S_2^b|/T_2$ and $|S_2^b| = |S^b|/2$.

Then

$$\|\text{Cov}(\Delta_i)\| = \frac{1}{|\tilde{B}|}\max_{b\in\tilde{B}} \|\text{Cov}(z_i^b)\| \le \frac{2T_2\kappa^2 C_1}{|\tilde{B}|\cdot \min_{b\in\tilde{B}}|S^b|}$$

Since Δ_i lies in an ℓ dimensional subspace of $\mathbb{R}^d$, it follows that

$$\text{Tr}(\text{Cov}(\Delta_i)) \le \ell\|\text{Cov}(\Delta_i)\| \le \frac{2\ell T_2\kappa^2 C_1}{|\tilde{B}|\cdot \min_{b\in\tilde{B}}|S^b|}$$

Note that $\text{Var}(\|\Delta_i - \mathbb{E}[\Delta_i]\|) = \text{Tr}(\text{Cov}(\Delta_i))$. Then, from Chebyshev's bound:

$$\Pr[\|\Delta_i - \mathbb{E}[\Delta_i]\| \ge \epsilon\kappa\sqrt{C_1}] \le \frac{\text{Var}(\|\Delta_i - \mathbb{E}[\Delta_i]\|)}{\epsilon^2\kappa^2 C_1} \le \frac{2\ell T_2}{\epsilon^2|\tilde{B}|\cdot\min_{b\in\tilde{B}}|S^b|} \le 1/8.$$

Combining above with Equation (6.26),

$$\Pr[\|\Delta_i - \mathbb{E}_{\mathcal{D}_0}[P\nabla f(x,y,w,\kappa)]\| \ge 3\epsilon\kappa\sqrt{C_1}] \le 1/4.$$

Let $D := \{i \in [T_2] : \|\Delta_i - \mathbb{E}_{\mathcal{D}_0}[P\nabla f(x,y,w,\kappa)]\| \le 3\epsilon\kappa\sqrt{C_1}\|\}$. Then, for $T_2 =$

$\Omega(\log \frac{1}{\delta'})$, with probability $\geq 1 - \delta'$, we have

$$|D| \geq \frac{1}{2} T_2.$$

Recall that in the subroutine, we defined $\xi_i = median\{j \in [T_2] : \|\Delta_i - \Delta_j\|\}$ and $i^* = \arg\min\{i \in [T_2] : \xi_i\}$.

From the definition of D, and triangle inequality, for all $i, j \in D$, we have $\|\Delta_i - \Delta_j\| \leq 6\epsilon\kappa\sqrt{C_1}$. Therefore, if $|D| \geq \frac{1}{2}T_2$, then for any $i \in D$, $\xi_i \leq 6\epsilon\kappa\sqrt{C_1}$. This would imply $\xi_{i^*} \leq 6\epsilon\kappa\sqrt{C_1}$. Furthermore, since $|D| \geq \frac{1}{2}T_2$, there exist at least one $i \in D$ such that $\|\Delta_i - \Delta_{i^*}\| \leq 6\epsilon\kappa\sqrt{C_1}$. Using the definition of D, and the triangle inequality, we can conclude that

$$\|\Delta i^* - \mathbb{E}_{\mathcal{D}_0}[P\nabla f(x,y,w,\kappa)]\| \leq \|\Delta_i - \mathbb{E}_{\mathcal{D}_0}[P\nabla f(x,y,w,\kappa)]\| + \|\Delta_i - \Delta_{i^*}\| \leq 9\epsilon\kappa\sqrt{C_1}.$$

■

Theorem 119 then follows by combining lemmas 124 and 125.

6.12 Number of steps required

The following lemma shows that with a sufficiently accurate estimation of the expectation of gradients, a logarithmic number of gradient descent steps are sufficient in the main algorithm 10.

Lemma 126. *For $\epsilon > 0$, suppose $\|\Delta^{(r)} - \Sigma_0(\hat{w}^{(r)} - w_0)\| \leq \frac{1}{2}\|\hat{w}^{(r)} - w_0\| + \frac{\epsilon\sigma}{4}$, and $R = \Omega(C_1 \log \frac{\|w_0\|}{\sigma})$, then $\|\hat{w}^{(r)} - w_0\| \leq \epsilon\sigma$.*

Proof. Recall that $\hat{w}^{(r+1)} = \hat{w}^{(r)} - \frac{1}{C_1}\Delta^{(r)}$. Then we have:

$$\begin{aligned}\hat{w}^{(r+1)} - w_0 &= \hat{w}^{(r)} - w_0 - \frac{1}{C_1}\Delta^{(r)} \\ &= (\hat{w}^{(r)} - w_0)\left(I - \frac{1}{C_1}\Sigma_0\right) + \frac{1}{C_1}(\Sigma_0(\hat{w}^{(R+1)} - w_0) - \Delta^{(r)}).\end{aligned}$$

Using triangle inequality, we obtain:

$$\begin{aligned}
\|\hat{w}^{(r+1)} - w_0\| &\leq \|\hat{w}^{(r)} - w_0\| \left\| I - \frac{1}{C_1}\Sigma_0 \right\| + \frac{1}{C_1}\|\Sigma_0(\hat{w}^{(r)} - w_0) - \Delta^{(r)}\| \\
&\leq \|\hat{w}^{(r)} - w_0\| \left(1 - \frac{1}{C_1}\right) + \frac{1}{C_1}\left(\frac{\|\hat{w}^{(r)} - w_0\|}{2} + \frac{\epsilon\sigma}{4}\right) \\
&\leq \|\hat{w}^{(r)} - w_0\| \left(1 - \frac{1}{2C_1}\right) + \frac{\epsilon\sigma}{4C_1}.
\end{aligned}$$

Using recursion, we have:

$$\begin{aligned}
\|\hat{w}^{(R+1)} - w_0\| &\leq \|\hat{w}^{(1)} - w_0\| \left(1 - \frac{1}{2C_1}\right)^R + \sum_{i=0}^{R-1}\left(1 - \frac{1}{2C_1}\right)^i \frac{\epsilon\sigma}{4C_1} \\
&\leq \|\hat{w}^{(1)} - w_0\| \exp\left(-\frac{R}{2C_1}\right) + 2C_1\frac{\epsilon\sigma}{4C_1} \\
&\leq \epsilon\sigma,
\end{aligned}$$

where the second inequality follows from the upper bound on the sum of infinite geometric series and the last inequality follows from the bound on R and $\hat{w}^{(1)} = 0$. ■

6.13 Final Estimation Guarantees

Proof of Theorem 106. We show that with probability $\geq 1 - \delta$, for each $r \in [R]$, the gradient computed by the algorithm satisfies $\|\Delta^{(r)} - \Sigma_0(\hat{w}^{(r)} - w_0)\| \leq \frac{1}{2}\|\hat{w}^{(r)} - w_0\| + \frac{\epsilon\sigma}{4}$. Lemma 126 then implies that for $R = \Omega(C_1 \log \frac{M}{\sigma})$, the output returned by the algorithm $\hat{w} = \hat{w}^{(R+1)}$ satisfy $\|\hat{w} - w_0\| \leq \epsilon\sigma$.

To show this, we fix r, and for this value of r, we show that with probability $\geq 1 - \delta/R$, $\|\Delta^{(r)} - \Sigma_0(\hat{w}^{(r)} - w_0)\| \leq \frac{1}{2}\|\hat{w}^{(r)} - w_0\| + \frac{\epsilon\sigma}{4}$. Since each round uses an independent set of samples, the theorem then follows by applying the union bound.

First, we determine the bound on the clipping parameter. From Theorem 114, for

$|S^{b^*}|/R = \Omega(C^2 \log 1/\delta')$, with probability $\geq 1-\delta'$, we have

$$\sqrt{\tfrac{8(C+1)C_1\left(\mathbb{E}_{\mathcal{D}_0}[(y-x\cdot w^{(r)})^2]\right)}{\epsilon_1}} \leq \kappa^{(r)} \leq 28\sqrt{\tfrac{2(C+1)C_1\left(\mathbb{E}_{\mathcal{D}_0}[(x\cdot(w^{(r)}-w_0))^2]+\sigma^2\right)}{\epsilon_1}}. \tag{6.28}$$

Next, employing Theorem 112 and utilizing the lower bound on the clipping parameter in the above equation, we obtain the following bound on the norm of the expected difference between clipped and unclipped gradients:

$$\left\|\mathbb{E}_{\mathcal{D}_0}[(\nabla f(x,y,w^{(r)},\kappa^{(r)}) - \Sigma_0(w^{(r)}-w_0)\right\| \leq \epsilon_1 \|w^{(r)}-w_0\|. \tag{6.29}$$

Recall that in B_s, at least α_s fraction of the batches contain samples from $\mathcal{D}_0$. When B_s is divided into R equal random parts, w.h.p. each part $B_s^{(r)}$ will have at least α_s fraction of the batches containing samples from $\mathcal{D}_0$.

From Theorem 116, if $|B_s^{(r)}| = \frac{|B_s|}{R} = \Omega\Big(\frac{d}{\alpha_s\epsilon_2^2}\Big(\frac{1}{\alpha_s\epsilon_2^2}+\frac{C_2^2}{C_1}\Big)\log\frac{d}{\delta'}\Big)$, then with probability $\geq 1-\delta'$, the projection matrix $P^{(r)}$ satisfies

$$\|\mathbb{E}_{\mathcal{D}_0}[\nabla f(x,y,w^{(r)},\kappa^{(r)})] - P^{(r)}\mathbb{E}_{\mathcal{D}_0}[\nabla f(x,y,w^{(r)},\kappa^{(r)})]\| \leq 4\epsilon_2\kappa^{(r)}\sqrt{C_1}. \tag{6.30}$$

The above equation shows subroutine GradSubEst finds projection matrix $P^{(r)}$ such that the expected value of clipped gradients projection is roughly the same as the expected value of the clipped gradient.

Next, we show that subroutine GradEst provides a good estimate of the expected value of clipped gradients projection. Let N denote the number of batches in B_m that have samples from $\mathcal{D}_0$. If $N \geq \Omega(R + \log 1/\delta')$ then with probability $\geq 1-\delta'$, $B_m^{(r)}$ has $\Theta(N/R)$ batches sampled from $\mathcal{D}_0$. If each batch in B_m and batch b^* has more than n_m samples, $\frac{n_m}{R} = \Omega(\frac{\sqrt{\ell}}{\epsilon_2^2}\log(\frac{|B_m|}{\delta'}))$, and $\frac{N \cdot n_m}{R} \geq \Omega(\frac{\ell}{\epsilon_2^2} \log 1/\delta')$, then from Theorem 119, with probability $\geq 1 - \delta'$

$$\left\|\Delta^{(r)} - \mathbb{E}_{\mathcal{D}_0}[P^{(r)} \nabla f(x, y, w^{(r)}, \kappa^{(r)})]\right\| \leq 9\epsilon_2 \kappa^{(r)} \sqrt{C_1}. \tag{6.31}$$

Combining the above three equations using triangle inequality,

$$\|\Delta^{(r)} - \Sigma_0(\hat{w}^{(r)} - w_0)\| \leq 13\epsilon_2 \kappa^{(r)} \sqrt{C_1} + \epsilon_1 \|w^{(r)} - w_0\|, \tag{6.32}$$

with probability $\geq 1 - 5\delta'$.

In equation (6.28) using the upper bound, $\mathbb{E}_{\mathcal{D}_0}[(x \cdot (w^{(r)} - w_0))^2] \leq \|w^{(r)} - w_0\|^2 \|\Sigma_0\| \leq C_1 \|w^{(r)} - w_0\|^2$ we get

$$\begin{aligned} \kappa^{(r)} &\leq 28 \sqrt{\frac{2(C+1){C_1}^2 \|w^{(r)} - w_0\|^2 + (C+1) C_1 \sigma^2}{\sqrt{\epsilon_1}}} \\ &\leq \frac{28\sqrt{2(C+1)}}{\sqrt{\epsilon_1}} (C_1 \|w^{(r)} - w_0\| + \sqrt{C_1}\sigma). \end{aligned}$$

Combining the two equations,

$$\|\Delta^{(r)} - \Sigma_0(\hat{w}^{(r)} - w_0)\| \leq \frac{364\epsilon_2 \sqrt{2(C+1)C_1}}{\sqrt{\epsilon_1}} (C_1 \|w^{(r)} - w_0\| + \sqrt{C_1}\sigma) + \epsilon_1 \|w^{(r)} - w_0\|, \tag{6.33}$$

with probability $\geq 1 - 5\delta'$ There exist universal constants $c_1, c_2 > 0$ such that for $\epsilon_1 = c_1$ and $\epsilon_2 = \frac{c_2}{C_1\sqrt{C+1}}\left(\epsilon + \frac{1}{\sqrt{C_1}}\right)$, the quantity on the right is bounded by $\|w^{(r)} - w_0\|/2 + \epsilon\sigma/4$. We choose these values for ϵ_1 and ϵ_2 and $\delta' = \frac{\delta}{5R}$.

From the above discussion, it follows that if $|B_s| = \tilde{\Omega}\left(\frac{d}{{\alpha_s}^2 \epsilon^4}\right)$, $n_m \geq \tilde{\Omega}(\frac{\sqrt{\ell}}{\epsilon^2})$, and B_m has $\geq \frac{1}{n_m}\tilde{\Omega}\left(\frac{\ell}{\epsilon^2}\right)$ batches sampled from $\mathcal{D}_0$, then with probability $\geq 1 - \delta/R$,

$$\|\Delta^{(r)} - \Sigma_0(\hat{w}^{(r)} - w_0)\| \leq \frac{1}{2}\|\hat{w}^{(r)} - w_0\| + \frac{\epsilon\sigma}{4}.$$

Using $\ell = \min\{k, \frac{1}{\epsilon^2 \alpha_s}\}$, we get the bounds on the number of samples and batches required by the algorithm. ■

6.14 Proof of Lemma 118

To establish the lemma, we first introduce and prove two auxiliary lemmas.

Lemma 127. *For $k > 0$, and a probability distribution $(p_0, p_1, ..., p_{k-1})$ over k elements, let $Z = \sum_{i=0}^{k-1} p_0 z_i z_i^\intercal$, where z_i are d-dimensional vectors. Then for all $\ell \geq 0$, ℓ^{th} largest singular value of Z is bounded by $\max_i \|z_i\|^2/\ell$.*

Proof. Note that Z is a symmetric matrix, so its left and right singular values are the same. Let $v_1, v_2, ...$ be the singular vectors in the SVD decomposition of Z, and let $a_1 \leq a_2 \leq a_3 \leq ...$ be the corresponding singular values. Using the properties of SVD, we have:

$$\sum_i a_i = \sum_i v_i^\intercal Z v_i = \sum_i v_i^\intercal \left(\sum_{j=0}^{k-1} p_j z_j z_j^\intercal \right) v_i = \sum_{j=0}^{k-1} p_j \sum_i (v_i \cdot z_j)^2 \leq \sum_{j=0}^{k-1} p_j \|z_j\|^2$$
$$\leq \max_j \|z_j\|^2.$$

Next, we have:

$$\sum_i a_i \geq \sum_{i \leq \ell} a_i \geq \sum_{i \leq \ell} a_\ell = \ell \cdot a_\ell.$$

Combining the last two equations yields the desired result. ■

Lemma 128. *Let $u_1, u_2, .., u_\ell \in \mathbb{R}^d$ be ℓ mutually orthogonal unit vectors, and let $U = [u_1, u_2, ..., u_\ell] \in \mathbb{R}^{d \times \ell}$. For any set of k vectors $z_0, z_1, ..., z_{k-1} \in \mathbb{R}^d$, non-negative reals $p_0, p_1, ..., p_{k-1}$, and reals $a_1, a_2, ..., a_\ell$, we have:*

$$\|(I - UU^\intercal) z_0\|^2 \leq \frac{\left\| \sum_{i=1}^{k-1} p_i z_i z_i^\intercal - \sum_{j \in [\ell]} a_j u_j u_j^\intercal \right\|}{p_0}.$$

Proof. Let $v = (I - UU^\intercal)z_0$. First we show that for all $j \in [\ell]$, the vectors v and u_j are orthogonal,

$$u_j^\intercal(I - UU^\intercal)z_0 = (u_j^\intercal \cdot z_0) - (u_j^\intercal \cdot z_0) = 0.$$

Then,

$$\left\| v^\intercal \left(\sum_{i=0}^{k-1} p_i z_i z_i^\intercal - \sum_{j \in [\ell]} a_j u_j u_j^\intercal \right) v \right\| = \left\| v^\intercal \left(\sum_{i=0}^{k-1} p_i z_i z_i^\intercal \right) v \right\| = \left\| \sum_{i=0}^{k-1} p_i (z_i^\intercal v)^2 \right\| \geq \left\| p_0 (z_0^\intercal v)^2 \right\|$$

Next, we have:

$$z_0^\intercal v = z_0^\intercal (I - UU^\intercal)v + z_0 UU^\intercal v = z_0^\intercal (I - UU^\intercal)v = v^\intercal v = \|v\|^2$$

Combining the last two equations, we obtain:

$$\|v\|^2 \cdot \left\| \sum_{i=0}^{k-1} p_i z_i z_i^\intercal - \sum_{j \in [\ell]} a_j u_j u_j^\intercal \right\| \geq \left\| v^\intercal \left(\sum_{i=0}^{k-1} z_i z_i^\intercal - \sum_{j \in [\ell]} a_j u_j u_j^\intercal \right) v \right\| \geq p_0 \|v\|^4.$$

Dividing both sides by $\|v\|^2$ completes the proof. ■

Next, combining the above two auxiliary lemmas we prove Lemma 118.

Proof of Lemma 118. Let $\Lambda_i(\cdot)$ denote the i^{th} largest singular value of a matrix. Let $\hat{M}$ be rank ℓ truncated-SVD of M, then it follows that,

$$\|M - \hat{M}\| = \Lambda_{\ell+1}(M).$$

First, we consider the case $\ell < k$. By applying Weyl's inequality for singular values, we have

$$\Lambda_{\ell+1}(M) \leq \Lambda_{\ell+1}(Z) + \Lambda_1(M - Z) \leq \frac{\max_j \|z_j\|^2}{\ell + 1} + \|M - Z\|,$$

where the last equation follows from Lemma 127.

First applying the triangle inequality, and then using the above two equations, we have

$$\|\hat{M} - Z\| \le \|M - \hat{M}\| + \|M - Z\| \le \frac{\max_j \|z_j\|^2}{\ell + 1} + 2\|M - Z\|.$$

Combining the above equation with Lemma 128, we have:

$$\|(I - UU^\intercal)z_0\|^2 \le \frac{2(\ell+1)\|M - Z\| + \max_j \|z_j\|^2}{(\ell+1)p_0}.$$

This completes the proof for $\ell < k$. To prove for the case $\ell > k$, we use $\Lambda_{\ell+1}(Z) = 0$ in place of the bound $\Lambda_{\ell+1}(Z) \le \frac{\max_j \|z_j\|^2}{\ell+1}$ in the above proof for the case $\ell < k$. ■

6.15 Removing the Additional Assumptions

To simplify our analysis, we made two assumptions about the data distributions. We now argue that these assumptions are not limiting.

The first additional assumption was that there exists a constant $C_2 > 0$ such that for all components $i \in \{0, 1, \ldots, k-1\}$ and random samples $(x, y) \sim \mathcal{D}_i$, we have $\|x\| \le C_2\sqrt{d}$ almost surely. In the non-batch setting, Cherapanamjeri et al. (2020) [38] have shown that this assumption is not limiting. They showed that if other assumptions are satisfied, then there exists a constant C_2 such that with probability ≥ 0.99, we have $\|x\| \le C_2\sqrt{d}$. Therefore, disregarding the samples for which $|x| > C_2\sqrt{d}$ does not significantly reduce the data size. Moreover, it has minimal impact on the covariance matrix and hypercontractivity constants of the distributions. This argument easily extends to the batch setting. In the batch setting, we first exclude samples from batches where $\|x\| > C_2\sqrt{d}$. Then we remove small-sized batches with fewer than or equal to 2 samples and medium-sized batches that have been reduced by more than 10% of their original size. It is easy to show that w.h.p. the fraction of medium and small size batches that gets removed for any component is at most 10%. THence, this assumption can be removed with a small increase in the batch size and the number of required samples in our main results.

Next, we address the assumption that the noise distribution is symmetric. We can handle this by employing a simple trick. Consider two independent and identically distributed (i.i.d.) samples (x_1, y_1) and (x_2, y_2), where $y_i = w^* \cdot x_i + \eta_i$. We define $x = (x_1 - x_2)/\sqrt{2}$, $y = (y_1 - y_2)/\sqrt{2}$, and $\eta = (\eta_1 - \eta_2)/\sqrt{2}$. It is important to note that the distribution of η is symmetric around 0, and the covariance of x is the same as that of x_i, while the variance of η is the same as that of η_i. Furthermore, we have $y = w^* \cdot x + \eta$. Therefore, the new sample (x, y) obtained by combining two i.i.d. samples satisfies the same distributional assumptions as before, and in addition, the noise distribution is symmetric. We can combine every two samples in a batch using this approach, which only reduces the batch size of each batch by a constant factor of 1/2. Thus, the assumption of symmetric noise can be eliminated by increasing the required batch sizes in our theorems by a factor of 2.

6.16 More Simulation Details

Setup. We have sets B_s and B_m of small and medium size batches and k distributions $\mathcal{D}_i$ for $i \in \{0, 1, \ldots, k-1\}$. For a subset of indices $I \subseteq \{0, 1, \ldots, k-1\}$, both B_s and B_m have a fraction of α batches that contain i.i.d. samples from $\mathcal{D}_i$ for each $i \in I$. And for each $i \in \{0, 1, \ldots, k-1\} \setminus I$ in the remaining set of indices, B_s and B_m have $(1 - |I|/16)/(k - |I|)$ fraction of batches, that have i.i.d samples from $\mathcal{D}_i$. In all figures the output noise is distributed as $\mathcal{N}(0, 1)$.

All small batches have 2 samples each, while medium-size batches have n_m samples each, which we vary from 4 to 32, as shown in the plots. We fix data dimension $d = 100$, $\alpha = 1/16$, number of small batches to $|B_s| = \min\{8dk^2, 8d/\alpha^2\}$ and the number of medium batches to $|B_m| = 256$. In all the plots, we average our 10 runs.

Evaluation. Our objective is to recover a small list containing good estimates for the regression vectors of $\mathcal{D}_i$ for each $i \in I$. We compare our proposed algorithm's performance with that of the algorithm in [95]. We generate lists of regression vector estimates L_{Ours} and L_{KSSKO}

using our algorithm and [95], respectively. Then, we create 1600 new batches, each containing n_{new} i.i.d samples randomly drawn from the distribution $\mathcal{D}_i$, where for each batch, index i is chosen randomly from I.

Each list enables the clustering of the new sample batches. To cluster a batch using a list, we assign it to the regression vector in the list that achieves the lowest mean square error (MSE) for its samples.

To evaluate the average MSE for each algorithm, for each clustered batch, we generate additional samples from the distribution that the batch was generated from and calculate the error achieved by the regression vector in the list that the batch was assigned to. We then take the average of this error over all sets. We evaluate both algorithms' performance for new batch sizes $n_{new} = 4$ and $n_{new} = 8$, as shown in the plots.

Minimum distance between regression vectors. Our theoretical analysis suggests that our algorithm is robust to the case when the minimum distance between the regression vectors are much smaller than their norms. In order to test this, in Figure 6.3, we generate half of the regression vectors with elements independently and randomly distributed in $U[9, 11]$, and the other half with elements independently and randomly distributed in $U[-11, -9]$. Notably, the minimum gap between the vectors, in this case, is much smaller than their norm. It can be seen that the performance gap between our algorithm and the one in [95] increases significantly as we deviate from the assumptions required for the latter algorithm to work.

Number of different distributions. Our algorithm can notably handle very large k (even infinite) while still being able to recover regression vectors for the subgroups that represent sufficient fraction of the data. In the last plot, we set $k = 100$ and $I = \{0, 1, 2, 3\}$ to highlight this ability. In this case, the first four distributions each generate a $1/16$ fraction of batches, and the remaining 96 distributions each generate a $1/128$ fraction of batches. We provide the algorithm with one additional medium-size batch from $\mathcal{D}_i$ for each $i \in I$ for identification of a list of size I. The results are plotted in Figure 6.4, where we can see that the performance gets better with medium batch size as expected. Note that the algorithm in [95] cannot be applied to this scenario.

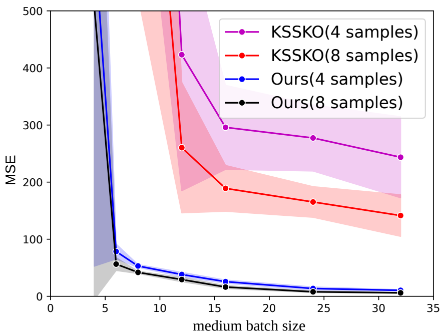


Figure 6.3. Same input dist. (standard normal), $k = 16$, small minimum distance between regression vectors, recovering all

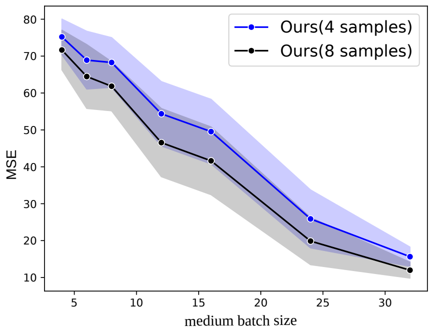


Figure 6.4. Different input dist, $k = 100$, large minimum distance between regression vectors, recovering 4 components that have $1/16$ fraction of batches each

Chapter 6, in full, is a reprint of the material as it appears in Linear Regression using Heterogeneous Data Batches 2023. Abhimanyu Das, Ayush Jain, Rajat Sen, Weihao Kong, Abhimanyu Das, and Alon Orlitsky. Submitted in Neurips 2023. The dissertation author was the primary investigator and author of this paper.

Chapter 7

Robust Estimation for Random Graphs

7.1 Introduction

Finding underlying patterns and structure in data is a central task in machine learning and statistics. Typically, such structures are induced by modelling assumptions on the data generating procedure. While they offer mathematical convenience, real data generally does not match with these idealized models, for reasons ranging from model misspecification to adversarial data poisoning. Thus for learning algorithms to be effective in the wild, we require methods that are *robust* to deviations from the assumed model.

With this motivation, we initiate the study of robust estimation for random graph models. Specifically, we will be concerned with the Erdős-Rényi (ER) random graph model [65, 60].[1]

Definition 2 (Erdős-Rényi graphs)**.** The Erdős-Rényi random graph model on n nodes with parameter $p \in [0, 1]$, denoted as $G(n, p)$, is the distribution over graphs on n nodes where each edge is present with probability p, independently of the other edges.

We consider graphs generated according to the Erdős-Rényi random graph model, but which then have a constant fraction of their nodes *corrupted* by an adversary. When a node is corrupted, the adversary can arbitrarily modify its neighborhood. This setting is naturally motivated by social networks, where random graphs are a common modelling assumption [114]. Even if a

[1] This model was introduced in [65], simultaneously with the related $G(n, m)$ model in [60]. Nevertheless, the community refers to both models Erdős-Rényi graphs.

fraction of individuals in the network are malicious actors, we still wish to perform inference with respect to the regular users. Apart from adversarial settings, tools for robust analysis of graphs may also assist in addressing deficiencies of existing models, such as in model misspecification. For example, certain random graph models have been criticized for not capturing various statistics of real-world networks [114], and some notion of robustness may facilitate better modelling.

7.1.1 Problem Setup

Let $\beta \in [0, 1]$ denote the fraction of corrupted nodes, and $G \sim G(n, p)$ be a random graph, where p is unknown. Without loss of generality, we assume that the node set is $[n] := \{1, \ldots, n\}$. An adversary $\mathcal{A}$ is then given G, and is allowed to arbitrarily 'rewire' the edges adjacent to a set $B \subseteq [n]$ of nodes of size at most βn, resulting in a graph $\mathcal{A}(G)$. In other words, the adversary can change the status of any edge with at least one end point in B. We call B the set of *corrupted nodes*. We consider two kinds of adversaries.

- *β-omniscient adversary*: The adversary knows the true value of the edge probability p and observes the realization of the graph $G \sim G(n, p)$. They then choose B and how to rewire its edges.
- *β-oblivious adversary*: The adversary knows the true value of the edge probability p. They must choose B and the distribution of edges from B without knowing the realization G.

Note that the oblivious adversary is weaker than the omniscient adversary. Given a corrupted graph $\mathcal{A}(G)$, our goal is to output $\hat{p}\left(\mathcal{A}(G)\right)$, an estimate of the true edge probability p.

7.1.2 Results

We first analyze standard estimators from the robust statistics toolkit, and show that they provide sub-optimal rates. We then propose a computationally-efficient spectral algorithm to estimate p with improved rates. Finally, we prove a lower bound for this problem, showing that our algorithms are optimal up to logarithmic factors. We note that our upper bounds hold for omniscient adversaries, whereas the lower bounds are tight even against the weaker oblivious adversary.

Standard Robust Estimators and Natural Variants

At first glance, the problem appears deceptively simple, as our goal is to estimate a single univariate parameter p. A standard technique is the maximum likelihood estimator, which in this case is the empirical edge density. We call the following the *mean estimator*

$$\hat{p}_{\text{mean}}(\mathcal{A}(G)) = \frac{\text{\# of edges present in } \mathcal{A}(G)}{\binom{n}{2}}. \tag{7.1}$$

In robust statistics, the median often provides better guarantees than the mean. Let $\deg(i)$ denote the degree of node $i \in [n]$ in $\mathcal{A}(G)$. The *median estimator* is given by

$$\hat{p}_{\text{med}}(\mathcal{A}(G)) = \frac{\text{Median}\{\deg(1), \ldots, \deg(n)\}}{n-1}. \tag{7.2}$$

Absent corruptions (i.e., $\beta = 0$), we have $\mathcal{A}(G) = G$. In this simple setting, the mean and median are both very accurate. Specifically, it is not hard to show that $|\hat{p}_{\text{mean}}(G) - p| \leq O\left(\sqrt{p(1-p)}/n\right)$ and $|\hat{p}_{\text{med}}(G) - p| \leq O\left(1/n\right)$ (Lemma 133). However, both estimators perform much worse under even mild corruption. In Lemma 134 we describe and analyze a simple oblivious adversary $\mathcal{A}$ such that both the mean and median estimator have $|\hat{p}(\mathcal{A}(G)) - p| \geq \beta/2$. Note that if even a single node is corrupted (i.e., $\beta = 1/n$), the "price of robustness" (informally, the additional error term(s) introduced in the corrupted setting) dominates the baseline $O(1/n)$ error in the uncorrupted setting.

The adversary against the mean and median estimators is easy to describe: either add or remove all edges incident to the nodes in B. This suggests the strategy of first pruning a set of $c\beta n$ nodes with the largest and smallest degrees and then applying either the mean or median estimator to the resulting graph. These *prune-then-mean/median* algorithms are described in Algorithm 15. Despite this additional step, the pruned estimators are still deficient. We design an oblivious adversary such that the prune-then-median estimate satisfies $|\hat{p}(\mathcal{A}(G)) - p| \geq \Omega(\beta)$ and the prune-then-mean estimate satisfies $|\hat{p}(\mathcal{A}(G)) - p| \geq \Omega(\beta^2)$ (Theorem 136). Interestingly, we show the tightness of both these bounds, showing that prune-then-mean improves the error to $O\left(\beta^2\right)$ (Theorem 135). These results are summarized in the theorem below.

Theorem 129 (Informal)**.** *The price of robustness of the prune-then-mean/median estimators are $\Theta(\beta^2)$ and $\Theta(\beta)$, respectively.*

A Spectral Algorithm for Robust Estimation

Given the failings of the approaches described so far, it may appear that a poly(β) cost for robustness may be unavoidable. Our main result is a computationally-efficient algorithm that bypasses this barrier.

Theorem 130. *Suppose $\beta < 1/60$ and $p \in [0,1]$. Let $G \sim G(n,p)$ and $\mathcal{A}(G)$ be a rewiring of G by a β-omniscient adversary $\mathcal{A}$. There exists a polynomial-time estimator $\hat{p}(\mathcal{A}(G))$ such that with probability at least $1 - 10n^{-2}$,*

$$|\hat{p}(\mathcal{A}(G)) - p| \leq C \cdot \left(\frac{\sqrt{p(1-p)\log n}}{n} + \frac{\beta\sqrt{p(1-p)\log(1/\beta)}}{\sqrt{n}} + \frac{\beta}{n}\log n \right),$$

for some constant C. This estimate can be computed in $\tilde{O}(\beta n^3 + n^2)$ time.

The first term is the error without corruptions, while the other two terms capture the price of robustness. Except at extreme values of p, the last term will be dominated by one of the other two. In this case, note that the cost of robustness in the second term decreases as the number of nodes n increases. This is in contrast to the previously described approaches, for which the price of robustness did not decrease with n. Observe that the non-robust error will dominate for most regimes when $\beta \leq 1/\sqrt{n}$.

As our lower bounds will establish, our algorithm provides a nearly-tight solution to the problem. Note that while this algorithm requires knowledge of β, [79] recently proposed a simple argument which using Lepski's method generically removes the need to know the corruption parameter for robust estimation tasks, leading to such an algorithm with the same rates.

Our upper bound requires $\beta < 1/60$.[2] On the other hand, note that if $\beta \geq 0.5$, an identifiability argument implies that no estimator can achieve error better than 0.5.[3] This raises the question of whether the above rates are achievable for all $\beta < 0.5$. We show that this is indeed the case, providing a computationally inefficient algorithm with the following guarantees.

Theorem 131. *Suppose $\beta < 1/2$. There exists an algorithm such that with probability at least $1 - n^{-2}$,*

$$|\hat{p}(\mathcal{A}(G)) - p| \leq \frac{C}{1/2 - \beta} \cdot \left(\frac{\sqrt{p(1-p)}}{\sqrt{n}} + \frac{\sqrt{\log n}}{n} \right),$$

for some constant C.

Note that for $\beta > 1/60$, the error bound above matches that presented in Theorem 130 up to a factor of $1/(0.5 - \beta)$, and therefore extends the error rates of Theorem 130 to the regime $\beta \in [1/60, 1/2)$ at the cost of computational efficiency.

Information-theoretic Lower Bounds

We provide a lower bound to establish near-optimality of our algorithms. While our upper bounds are against an omniscient adversary, the lower bounds hold for the weaker oblivious adversary.

Theorem 132. *For every $\beta < 1/2$, $p \in [0, 1]$, and $n \geq 0$ and a universal constant C', let*

$$\Delta = C' \cdot \left(\frac{\sqrt{p(1-p)}}{n} + \frac{\beta\sqrt{p(1-p)}}{\sqrt{n}} + \frac{\beta}{n} \right).$$

[2] We have not tried to optimize the value of β for computationally efficient algorithms, and could likely be made larger than $1/60$ through a more careful analysis.

[3] Consider an empty graph $G(n, 0)$. An adversary can corrupt half the graph into a clique, making it look like it came from $G(n, 1)$. No algorithm can identify which half of the graph was the original.

For any $p' \in [p + \Delta, p - \Delta]$ and $G \sim G(n,p)$ and $G' \sim G(n,p')$, there exists an oblivious adversary $\mathcal{A}$ such that no algorithm can distinguish between $\mathcal{A}(G)$ and $\mathcal{A}(G')$ with probability more than 0.65.

7.1.3 Techniques

Upper bound techniques. Broadly speaking, robust estimation is only possible when samples from the (uncorrupted) distribution enjoy some nice structure. Work in this area generally proceeds by imposing some regularity conditions on the uncorrupted data, which hold with high probability over samples from the distribution. The algorithm subsequently relies solely on these regularity conditions to make progress. For example, for mean estimation problems, it is common to assume that the mean and covariance of the uncorrupted samples are close to the true mean and covariance. However, the appropriate regularity conditions in our setting are far less obvious. We employ conditions which bound the empirical edge density and spectral norm for submatrices of the adjacency matrix, when appropriately centered around the true parameter p (Definition 3), which can be proven using tools from random matrix theory.

With our regularity conditions established, the algorithmic procedure proceeds in two stages: a coarse estimator, followed by a fine estimator.

Stage 1: A coarse estimate. Our regularity conditions are suggestive of the following intuition about how one might estimate the value of p. If one could locate a sufficiently large subgraph S of the uncorrupted nodes, such that their adjacency matrix centered around p has small spectral norm, then the empirical edge density of this subgraph would give a good estimate for the true parameter p. More precisely, we let A be the (corrupted) adjacency matrix of $\mathcal{A}(G)$, let $A_{S\times S}$ be the submatrix of A indexed by the set S, and p_S be the empirical edge density of the subgraph S. The goal is to obtain an S where $\|A_{S\times S} - p\|$ is small,[4] at which point we can output p_S.

There are two clear challenges with this approach. First off, we can not center the adjacency matrix around the unknown parameter p, since estimating that parameter is our goal.

[4] For clarity: in the expression $A_{S\times S} - p$, p is subtracted entry-wise.

However, we demonstrate that it instead suffices to center around p_S (Theorem 139). The other issue is that it is not clear how to identify such a set S of uncorrupted nodes. One (inefficient) approach is to simply inspect all sufficiently large subgraphs. This will be accurate (quantified in Theorem 140), but not computationally tractable.

Instead, our main algorithmic contribution is an efficient algorithm which achieves this same goal. We give an iterative spectral approach, which starts with $S = [n]$. In Lemma 142 we show that if the spectral norm of $A_{S\times S} - p_S$ is large, then the top eigenvector assigns significant weight to the set of corrupted nodes. Normalizing this eigenvector and sampling from the corresponding probability distribution identifies a corrupted node with constant probability. We eliminate this node from S and repeat the process. Finally, using this approach, we obtain a subset $S^* \subset [n]$ of nodes such that p_{S^*} is a coarse estimate of p.

Stage 2: Pruning the coarse estimate. It turns out that the above coarse estimate gives a price of robustness which is roughly $O(1/\sqrt{n})$, rather than the $O(\beta/\sqrt{n})$ we are trying to achieve. However, a simple pruning step allows us to complete the argument. Specifically, our coarse estimator gave us a set S^* such that the spectral norm of $A_{S^*\times S^*} - p_{S^*}$ is small and p_{S^*} is close to p. We employ this to show that most nodes must have degree close to p (Lemma 157). Thus, we remove $\Theta(\beta n)$ nodes whose degree (restricted to the subgraph S^*) is furthest from p_{S^*}. Our final estimate is the empirical density of the resulting pruned subgraph.

Lower bound techniques. A strategy for proving lower bounds is the following: Suppose there exists an adversary that with βn corruptions can convert the distribution $G(n, p)$ and $G(n, p + \delta)$ into the same distribution of random graphs, then we cannot estimate p to accuracy better than $\delta/2$. This is akin to couplings between $G(n, p)$ and $G(n, p + \delta)$ by corrupting only a βn nodes. Designing these couplings over Erdős-Rényi graphs can be tricky due to the fact that degrees of nodes are not independent of each other.

We instead consider directed Erdős-Rényi graphs, where an edge from a node i to j is present independently of all others. Then, the (outgoing) degrees of all the nodes are independent

Binomial distributions. Using total variation bounds between Binomial distributions we can design couplings between directed ER graphs with different parameters, thus showing a lower bound on the error of robustly estimating directed ER graphs. Our final argument is a reduction showing that estimating the parameters of undirected of graphs is at least as hard as estimating the parameters of directed ER graphs. Combining these bounds we obtain the lower bounds.

7.1.4 Related Work

Due to the wealth of study in robust estimation and the page limits, we mention here only a fraction of the most relevant related work. For additional discussion, please see Section 7.6.

Robust statistics is a classic and mature branch of statistics which focuses on precisely this type of setting since at least the 1960s [142, 74]. However, since the classic literature typically did not take into account computational considerations, proposed estimators were generally intractable for settings of even moderate dimensionality [16]. Recently, results in [47] and [99] overcame this barrier, producing the first algorithms which are both accurate and computationally efficient for robust estimation in multivariate settings. While they focused primarily on parameter estimation of Gaussian data, a flurry of subsequent works have provided efficient and accurate robust algorithms for a vast array of settings.

A common tool in several of these robust estimation results is to prune suspected outliers from the dataset so that a natural estimator over the remaining points has a small error. We also use this meta technique in this paper. We note that as in the previous works, the main challenge lies in designing efficient schemes to detect and remove corrupted data-points for the particular task at hand. In most prior works, the uncorrupted data-points are unaffected by corruptions. In our setting however, the edges from the good nodes are also affected by corruptions to the corrupted nodes. This presents a new challenge requiring new insights.

7.2 Notation and Preliminaries

Problem Formulation.

Let $G \sim G(n, p)$. An adversary observes G and chooses a subset $B \subseteq [n]$ of nodes with $|B| \leq \beta n$. It can then change the status (i.e., presence or non-presence) of any edge with at least one node in B to get a graph $\mathcal{A}(G)$. Let $F = [n] \setminus B$. We call B the *corrupted* nodes, and F the *uncorrupted* nodes. Let $\tilde{A}$ and A be the $n \times n$ adjacency matrix of the original graph G and the modified graph $\mathcal{A}(G)$ respectively. Then $A_{F\times F} = \tilde{A}_{F\times F}$ and the remaining entries of A can be arbitrary. Given A, the goal is to estimate p, the parameter of the underlying random graph model. The algorithm does not know the set B, though we assume that it knows the value of β.

Notation.

The ℓ_2 norm of a vector $v = [v_1, \ldots, v_n] \in \mathbb{R}^n$ is $\|v\| := \sqrt{\sum_{i=1}^n v_i^2}$. Suppose M is an $m \times n$ real matrix. The spectral norm of M is

$$\|M\| := \max_{u\in\mathbb{R}^m, v\in\mathbb{R}^n : \|u\|=1, \|v\|=1} |u^T M v|. \tag{7.3}$$

It is easy to check that $\|M\| = \max_{v\in\mathbb{R}^n:\|v\|=1} \|Mv\|$. For a matrix M and real number $a \in \mathbb{R}$, let $M - a$ be the matrix obtained by subtracting a from each entry of M. For $S \subseteq [m]$, $S' \subseteq [n]$, let $M_{S\times S'}$ be the $m \times n$ matrix that agrees with M on $S \times S'$ and is zero elsewhere. Similarly for a vector $v \in \mathbb{R}^n$ and $S \subseteq [n]$, let vector v_S be the vector that agrees with with v on S and has zero entries elsewhere. Our proofs will use several standard properties of the matrix spectral norm, which we state in Appendix 7.7 for completeness.

7.3 Mean- and Median-based Algorithms

To demonstrate the need for our more sophisticated algorithms in Section 7.4, we first analyze canonical robust estimators for univariate settings – specifically, approaches based on trimming and order statistics (i.e., the median).

Recall the mean and median estimators for p in (7.1) and (7.2). The following simple lemma quantifies their guarantees in the setting absent corruptions.

Lemma 133. *Suppose $\beta = 0$. There exists a constant $C > 0$ such that with probability at least 0.99, $|\hat{p}_{\mathrm{mean}}(G) - p| \leq C \cdot \frac{\sqrt{p(1-p)}}{n}$, and $|\hat{p}_{\mathrm{med}}(G) - p| \leq C \cdot \frac{1}{n}$.*

The analysis of these estimators is not difficult, but we include them for completeness in Section 7.9.1. Analysis of the median estimator is slightly more involved due to correlations between nodes.

While both estimators are optimal up to constant factors (for constant p) without corruptions, their performance decays rapidly in the presence of an adversary, scaling at least linearly in the corruption fraction β. In particular, consider an adversary that picks βn nodes at random and either adds all the edges with at least one endpoint in B or removes all of them. In Section 7.9.2 we prove the following lower bound on the performance of the mean and median estimators for such an adversary. Observe that if even one node is corrupted (i.e., $\beta \geq 1/n$), the error in Lemma 134 dominates the error without corruptions in Lemma 133.

Lemma 134. *There exists an adversary $\mathcal{A}$ such that for $\hat{p} \in \{\hat{p}_{\mathrm{mean}}(\mathcal{A}(G)), \hat{p}_{\mathrm{med}}(\mathcal{A}(G))\}$ with probability at least 0.5, we have $|\hat{p} - p| \geq \beta/2$.*

A common strategy in robust statistics is to prune or trim the most extreme outliers. Accordingly, in our setting, one may prune the nodes with the most extreme degrees, described in Algorithm 15. This strategy bypasses the adversary which provides the lower bound in Lemma 134.

Algorithm 15. Prune-then-mean/median algorithm

```
    Input: A graph $\mathcal{A}(G)$, corruption parameter $\beta$, a constant $c > 0$
    Remove $c\beta n$ nodes with largest and smallest degrees from $\mathcal{A}(G)$
    Apply the mean/median estimator from (7.1)/(7.2) to the resulting graph on $(1 - 2c\beta) \cdot n$ nodes
```

However, this strategy can only go so far. Roughly speaking, pruning improves the mean's robust accuracy from $\Theta(\beta)$ to $\Theta(\beta^2)$, while pruning does not improve the median's robust accuracy. The upper and lower bounds are described in Theorems 135 and 136, and proved in Sections 7.9.3 and 7.9.4, respectively.

Theorem 135. *For $c \geq 1$ and $0 < \beta \cdot c < 0.25$, the prune-then-mean and prune-then-median estimators described in Algorithm 15 prune $2c\beta n$ nodes in total and with probability $1 - n^{-2}$ estimates p to an accuracy $\mathcal{O}\big(c\beta^2 + \frac{\log n}{n}\big)$ and $\mathcal{O}\big(c\beta + \sqrt{\frac{\log n}{n}}\big)$, respectively.*

Theorem 136. *Let $p = 0.5$, $\beta > 100 \cdot \sqrt{\log n/n}$, and $c > 0$ be such that $c\beta < 0.25$. There exists an adversary such that with probability at least 0.99, the prune-then-median estimate that deletes $c\beta n$ satisfies $|\hat{p}(\mathcal{A}(G)) - p| \geq C'\beta$, and the prune-then-mean estimate satisfies $|\hat{p}(\mathcal{A}(G)) - p| \geq C'\beta^2$.*

To summarize: none of the standard univariate robust estimators we have explored are able to achieve error better than $\Omega(\beta^2)$. To bypass this barrier, we turn to more intricate techniques in designing our main estimator in Section 7.4.

7.4 An Algorithm for Robust Estimation

Non-trivial robust estimation in Erdős-Rényi graphs is possible because even if the set of edges connected to a small set of nodes is changed arbitrarily, the subgraph between the remaining nodes retains a certain structure. In Section 7.4.1, we formalize this structure as deterministic regularity conditions and show that the subgraph corresponding to the set of uncorrupted nodes satisfy them with high probability. In the following subsections, we use only the fact that the subgraph of uncorrupted nodes satisfy these regularity conditions to derive our robust algorithms for estimating p.

In Section 7.4.2, we first derive a simple but novel inefficient spectral algorithm for coarse estimation of p. Our efficient algorithm consists of two parts: an efficient version of the spectral algorithm in Section 7.4.2 that, as its inefficient counterpart, provides a coarse estimate of p, followed by a trimming algorithm which achieves near-optimal error rates for estimating p. We describe and analyze the spectral and trimming components of the algorithm in Sections 7.4.3 and 7.4.4, respectively. Finally, in Appendix 7.10.8, we put the pieces together to show that guarantees for these algorithms imply our upper bound in Theorem 130.

7.4.1 Regularity Conditions

In this section we state a set of three deterministic regularity conditions. We will then show that the set of uncorrupted nodes of a random Erdős-Rényi graph satisfy these regularity conditions with high probability. First, we define the following quantities κ and η, which we use in stating the regularity conditions and in the bounds of several lemmas and theorems. For $p \in [0,1]$ and $n > 0$, let

$$\eta(p,n) := c \cdot \max\Big(\sqrt{\frac{p(1-p)}{n}}, \frac{\sqrt{\ln n}}{n}\Big). \tag{7.4}$$

For $\alpha \in (0,1]$, $p \in [0,1]$ and $n > 0$, let

$$\kappa(\alpha,p,n) := c_1 \cdot \max\Big(\alpha\sqrt{\frac{p}{n}\ln\frac{e}{\alpha}}, \frac{\alpha}{n}\ln\frac{e}{\alpha}, \frac{\sqrt{p\ln n}}{n}\Big). \tag{7.5}$$

In the above definitions c and c_1 are some constants that we determine in Theorem 138.

We employ the following regularity conditions.

Definition 3. Given $\alpha_1 \in [0,1/2)$, $\alpha_2 \in [0,1/2)$, and an $[n] \times [n]$ adjacency matrix A, a set of nodes $F \subseteq [n]$ of the graph corresponding to A satisfy (α_1, α_2, p)-*regularity* if

1. $|F^c| \leq \alpha_1 n$.

2. For all $F' \subseteq F$,

$$\|(A-p)_{F'\times F'}\| \leq n \cdot \eta(p,n).$$

3. For all $F', F'' \subseteq F$ such that $|F'|, |F''| \in [0, \alpha_2 n] \cup [n - \alpha_2 n, n]$, then

$$\Big|\sum_{i \in F'} \sum_{j \in F''} (A_{i,j} - p)\Big| \leq n^2 \cdot \kappa(\alpha_2, p, n).$$

Item 2 implies that upon subtracting p from each entry of the adjacency matrix A, the spectral norm of the matrix corresponding to all subgraphs of the subgraph $F \times F$ is bounded. Item 3 implies that upon subtracting p from each entry of the adjacency matrix A, the sum of the entries over any of its submatrices $F' \times F'' \subseteq F \times F$ has a small absolute value, as long as each of F' and F'' either leave out or include at most $\alpha_2 n$ nodes. We will informally refer to nodes in the set $F \subseteq [n]$ that satisfy (α_1, α_2, p)-regularity as *good nodes*.

For a subset $S \subseteq [n]$ and adjacency matrix A, we will use $p_S := \frac{\sum_{i,j \in S} A_{i,j}}{|S|^2}$ to denote (approximately) the empirical fraction of edges present in the subgraph induced by a set S. Note that this differs slightly from expression one might anticipate, $\binom{|S|}{2}^{-1} \left(\sum_{i<j:\, i,j \in S} A_{i,j}\right)$. For convenience, our sum double-counts each edge and also includes the $A_{i,i}$ terms (which are always 0 due to the lack of self-loops). The double counting is accounted for since the denominator is scaled by a factor of 2. The inclusion of the diagonal 0's is *not* accounted for, thus leading to p_S being a slight under-estimate of the empirical edge parameter for this subgraph, but not big enough to make a significant difference.

The following lemma lists some simple but useful consequences of the regularity conditions that we use in later proofs. We prove it in Appendix 7.10.1.

Lemma 137. *Suppose $0 \leq \alpha_1, \alpha_2 < 1/2$ and adjacency matrix A has a node subset $F \subseteq [n]$ that satisfies (α_1, α_2, p)-regularity, then*

1. *For all $F' \subseteq F$,*

$$\|(A - p_{F'})_{F' \times F'}\| \leq 2n \cdot \eta(p, n). \tag{7.6}$$

2. *For all $F' \subseteq F$ of size $\geq (1-\alpha_2)n$,*

$$|p_{F'} - p| \leq 4\kappa(\alpha_2, p, n). \tag{7.7}$$

Equation (7.6) implies that if the adjacency matrix of any subset of good nodes is centered around its empirical fraction of the edges, then its spectral norm is bounded. Equation (7.7) implies that for any subset of good nodes that excludes at most $\alpha_2 n$ nodes, the empirical fraction of edges in the subgraph induced by it estimates p accurately.

The next theorem shows that the set of uncorrupted nodes of a random Erdős-Rényi graph satisfy these regularity conditions with high probability. The proof of the Theorem is in Appendix 7.10.2.

Theorem 138. *For any $\beta \in [0, 1/2)$, $n > 0$ and $p > 0$, let A be a β-corrupted adjacency matrix of a sample from $G(n,p)$. There exist universal constants c and c_1 in Equations* (7.4) *and* (7.5)*, respectively, such that with probability at least $1 - 4n^{-2}$ the set of uncorrupted nodes F satisfy (α_1, α_2, p)-regularity for all $\alpha_1 \in [\beta, 1/2]$ and $\alpha_2 \in [0, 1/2]$.*

7.4.2 An Inefficient Coarse Estimator

In this section we propose a simple inefficient algorithm to recover a coarse estimate of p, which has an optimal dependence on all parameters other than α_1.

The following theorem serves as the foundation of our coarse estimator. It shows that if, for any subset $S \subseteq [n]$ of size $\geq n/2$ nodes, the spectral norm of its submatrix centered with respect to p_S is small, then p_S is a reasonable estimate of p.

Theorem 139. *Suppose $0 \leq \alpha_1, \alpha_2 < 1/2$, and let A be an adjacency matrix containing a (α_1, α_2, p)-regular subgraph. Then for all $S \subseteq [n]$ such that $|S| \geq n/2$, we have*

$$|p_S - p| \leq \frac{\|(A - p_S)_{S\times S}\| + n \cdot \eta(p,n)}{(1/2 - \alpha_1)n}.$$

Proof. Let F be the (α_1, α_2, p)-regular subgraph of A. From the triangle inequality,

$$\|(A - p_S)_{(S\cap F)\times(S\cap F)}\| \geq |p - p_S| \cdot |S \cap F| - \|(A - p)_{(S\cap F)\times(S\cap F)}\|.$$

Then by Lemma 147,

$$\begin{aligned} |p - p_S| \cdot |S \cap F| &\leq \|(A - p_S)_{(S\cap F)\times(S\cap F)}\| + \|(A - p)_{(S\cap F)\times(S\cap F)}\| \\ &\leq \|(A - p_S)_{S\times S}\| + \|(A - p)_{F\times F}\|. \end{aligned}$$

Finally, noting that $|S \cap F| \geq |S| - |F^c| \geq |S| - \alpha_1 n \geq n/2 - \alpha_1 n$ proves the theorem. ■

With this in hand, it suffices to locate a subset of nodes S such that $\|(A - p_S)_{S\times S}\|$ is small. We provide the accuracy guarantee of our inefficient algorithm in the following theorem.

Theorem 140. *Suppose* $0 \leq \alpha_1, \alpha_2 < 1/2$*, and let* A *be an adjacency matrix containing a* (α_1, α_2, p)*-regular subgraph. Let*

$$\hat{S} = \arg\min_{S\subseteq[n]:|S|\geq n/2} \|(A - p_S)_{S\times S}\|.$$

Then $\|(A - p_{\hat{S}})_{\hat{S}\times\hat{S}}\| \leq 2n \cdot \eta(p, n)$ *and* $|p_{\hat{S}} - p| \leq \frac{3}{(1/2-\alpha_1)} \cdot \eta(p, n)$.

Proof. Let F be the (α_1, α_2, p)-regular subgraph of A. From the definition of $\hat{S}$,

$$\|(A - p_{\hat{S}})_{\hat{S}\times\hat{S}}\| \leq \|(A - p_F)_{F\times F}\| \leq 2n \cdot \eta(p, n),$$

where the last inequality uses Equation (7.6). The proof follows from Theorem 139. ■

Theorem 140 implies the following simple algorithm to estimate p: compute $\hat{S}$ by iterating over all subsets of $[n]$, and then output $p_{\hat{S}}$. Combining with Theorem 138, this proves Theorem 131. The clear downside of this approach is that it is not computationally efficient, with a running time that depends exponentially on n. Also, as we will later establish, while this algorithm gives near-optimal rates for all constant β bounded away from $1/2$ by a constant, it may be sub-optimal for smaller β. In the following sections, we address both of these issues: we provide a computationally efficient algorithm which provides near-optimal rates for $\beta < 1/60$.

7.4.3 An Efficient Coarse Spectral Algorithm

In this section, we propose an efficient spectral method (Algorithm 16) which finds a subset $S^* \subseteq [n]$ such that both the set $(S^*)^c$ and the spectral norm $\|(A - p_{S^*})_{S^* \times S^*}\|$ are small. Note that the latter guarantee is comparable to the inefficient algorithm from Section 7.4.2. Then Theorem 139 implies that p_{S^*} is an accurate estimate of p. We note that this is still a *coarse* estimate of p, which has a sub-optimal dependence on α_1.[5] In the following section, we will post-process the set S^* returned by Algorithm 16 to provide our near-optimal bounds.

Theorem 141. *Suppose* $\alpha_1 \in [\frac{1}{n}, \frac{1}{60}]$, $\alpha_2 \in [0, 1/2]$ *and let* A *be an adjacency matrix containing an* (α_1, α_2, p)*-regular subgraph. With probability at least* $1 - 1/n^2$,[6] *Algorithm 16 returns a subset* S^* *with* $|S^*| \geq (1 - 9\alpha_1)n$ *such that* $\|(A - p_{S^*})_{S^* \times S^*}\| \leq 20n \cdot \eta(p, n)$. *Furthermore, these conditions on* S^* *imply* $|p_{S^*} - p| \leq 45 \cdot \eta(p, n)$.

[5] The guarantees are comparable to Theorem 140, up to constant factors.

[6] The probability of success of Algorithm 16 is $\Pr[\text{Bin}(\lfloor 9\alpha_1 n \rfloor, 0.15) \geq \lfloor \alpha_1 n \rfloor] \geq 1 - \exp(-\Omega(\alpha_1 n))$. Note that for all values of $\alpha_1 n$ the success probability is $> 1/2$. When $\alpha_1 n = \Omega(\log n)$ then it gives the probability of success at least $1 - 1/n^2$. When $\alpha_1 n = \mathcal{O}(\log n)$, to get the probability of success $\geq 1 - 1/n^2$ one can run Algorithm 16 $\mathcal{O}(\log n)$ times and choose an S^* for which $\|(A - p_{S^*})_{S^* \times S^*}\|$ is the minimum among all runs.

Algorithm 16. Spectral algorithm for estimating p

```
**Input:** number of nodes $n$, parameter $\alpha_1 \in [1/n, 1/60]$, adjacency matrix $A$
$S \leftarrow [n]$, Candidates $\leftarrow \{\}$
Candidates $\leftarrow$ Candidates $\cup \{S\}$
**for** $t = 1$ to $9\alpha_1 n$ **do**
    Compute a top normalized eigenvector $v$ of the matrix $(A - p_S)_{S\times S}$
    Draw $i_t$ from the distribution where $i \in S$ is selected with probability $v_i^2$
    $S \leftarrow S \setminus \{i_t\}$
    Candidates $\leftarrow$ Candidates $\cup \{S\}$
**end for**
$S^* \leftarrow \arg\min_{S\in\text{Candidates}} \|(A - p_S)_{S\times S}\|$
**Output:** $S^*$
```

In the remainder of this section we will prove that Algorithm 16 indeed outputs a subset S^* with the guarantee in Theorem 141. Let F (unknown) be the (α_1, α_2, p)-regular subgraph of A. The key technical argument is that if the spectral norm of $(A-p_S)_{S\times S}$ is large, the normalized top eigenvector v of $(A-p_S)_{S\times S}$ places constant weight on the subset $S\cap F^c$. Thus, if at a given iteration Algorithm 16 possesses an unsatisfactory set S, it will remove a node from $S\cap F^c$ with a constant probability. We formalize this argument in the following key Lemma 142. The proof of the lemma appears in Appendix 7.10.3.

Lemma 142. *Suppose* $\alpha_1 \in [\frac{1}{n}, \frac{1}{60}]$, $\alpha_2 \in [0, 1/2]$ *and let* A *be an adjacency matrix containing an* (α_1, α_2, p)*-regular subgraph* F*. Let* $S \subseteq [n]$ *be of size* $|S| \geq (1-9\alpha_1)n$*, and* v *be the normalized top eigenvector of* $(A - p_S)_{S\times S}$*. If* $\|(A - p_S)_{S\times S}\| \geq 20n\cdot\eta(p,n)$ *then* $\|v_{S\cap F^c}\|^2 \geq 0.15$.

We conclude this section with the proof of Theorem 141.

Proof of Theorem 141. It suffices to show that at least one of the sets S encountered by Algorithm 16 satisfies the condition $\|(A-p_S)_{S\times S}\| \leq 20n\cdot\eta(p,n)$. From Lemma 142 it follows that until the algorithm finds such a subset S, in each deletion step the probability of deleting a node from F^c is at least 0.15. Since there are $9\alpha_1 n$ steps, a standard Chernoff-style argument implies that either a subset S (including nodes from both F^c and F) satisfying the conditions of the theorem will be created, or with probability at least $\Pr[\text{Bin}(\lfloor 9\alpha_1 n\rfloor, 0.15) \geq \lfloor \alpha_1 n\rfloor] \geq 1-\exp(-\Omega(\alpha_1 n))$, all nodes from F^c will be deleted and thus $S \subseteq F$. In the latter case we apply Equation (7.6), which implies that $\|(A - p_S)_{S\times S}\| \leq 20n \cdot \eta(p, n)$ and the theorem. ■

Remark 4. Algorithm 16 runs for $9\alpha_1 n$ rounds, and in each round the algorithm finds the top eigenvector of an $n \times n$ matrix. This may be expensive to compute when the spectral gap is small. However, Lemma 155 shows it suffices to find any unit vector $v \in \mathbb{R}^n$ such that $|v^\intercal (A - p_S)_{S\times S} v| \geq 0.99||(A - p_S)_{S\times S}||$. Note that such a unit vector can be found in $\tilde{O}(n^2)$ time [113]. Therefore, one can implement Algorithm 16 to run in $\tilde{O}(\alpha_1 n^3)$ time.

7.4.4 A Fine Trimming Algorithm

In this section, we provide a trimming method (Algorithm 17), which refines the output of Algorithm 16, improving its guarantee (quantified in Theorem 141) by up to a factor of α_1.

The algorithm (Algorithm 17) is easy to describe. For a subset $S^* \subseteq [n]$ and a node $i \in S^*$, we define $p_{S^*}^{(i)} := \frac{\sum_{j\in S^*} A_{i,j}}{|S^*|}$ to be the normalized degree of node i in the subgraph induced by S^*. We remove the $3\alpha_1 n$ nodes for which this normalized degree deviate furthest from the average parameter p_{S^*}. Its guarantees are quantified in Theorem 143, whose proof appears in Appendix 7.10.7.

Theorem 143. *Let $\alpha_1 \in [\frac{1}{n}, \frac{1}{60}]$, and A be an adjacency matrix containing an $(\alpha_1, 13\alpha_1, p)$-regular subgraph. Suppose we have some S^* such that $|S^*| \geq (1-9\alpha_1)n$ and $\|(A-p_{S^*})_{S^*\times S^*}\| \leq 20n \cdot \eta(p, n)$, Algorithm 17 outputs p_{S^f} such that for some universal constants $c_2, c_3 > 0$,*

$$\left|p_{S^f} - p\right| \leq c_2\alpha_1\eta(p, n) + c_3\kappa(13\alpha_1, p, n).$$

Algorithm 17. Trimming Algorithm

```
Input:number of nodes $n$, parameter $\alpha_1 \in [1/n, 1/60]$, adjacency matrix $A$, subset $S^* \subseteq [n]$
Define the score for each node $i \in S^*$ to be $|p_{S^*} - p_{S^*}^{(i)}|$
Remove the $3\alpha_1 n$ nodes in $S^*$ with the highest scores to obtain $S^f$
return $p_{S^f}$
```

At this point, we have all the pieces to prove our main upper bound (Theorem 130). The argument first reasons that a random graph will satisfy certain regularity conditions with high probability. With these guarantees, we feed it into our coarse spectral algorithm (Algorithm 16), followed by our fine trimming algorithm (Algorithm 17). Some (mundane) case analysis is required to achieve the optimal bounds in certain parameter regimes; the full argument is rigorously described in Appendix 7.10.8.

7.5 Lower Bounds

In this section, we prove our main lower bound for robust parameter estimation in Erdős-Rényi random graphs establishing that our algorithms are tight up to logarithmic factors.

First we consider the problem of parameter estimation for directed version of Erdős-Rényi random graphs. Such graphs have independent (outgoing) degrees across the nodes. We then show a reduction showing that the directed version of the problem is at least as hard as the standard version. We start by describing the directed Erdős-Rényi graphs.

Definition 4 (Directed Erdős-Rényi graphs)**.** The directed Erdős-Rényi random graph model on n nodes with parameter p, denoted as $DG(n, p)$, is the distribution over directed graphs on n nodes where each edge is present with probability p, independently of the other edges.

We show the following reduction from the directed problem to standard. We provide the proof in Appendix 7.12.

Lemma 144. *If there exists an algorithm that estimates p in $G(n,p)$ to within $\pm\Delta$ with probability at most $1-\delta$ under β-corruptions, then there exists an algorithm for estimating p in $DG(n,p)$ to within $\pm\Delta$ with probability at most $1-\delta$ under β-corruptions.*

Then to prove the lower bound in Theorem 132, we prove its analogue for directed Erdős-Rényi graphs.

Theorem 145. *Let $p \le 0.5$. Then there exists a β-oblivious adversary such that no algorithm can distinguish between $DG(n,p)$ and $DG\left(n, p + 0.1\max\left(\beta\sqrt{p/n}, \beta/n, \sqrt{p}/n\right)\right)$ with probability at least 0.65.*

By symmetry, a similar statement holds for $p > 0.5$, with p replaced by $1-p$. Combining these two statements and Lemma 144 gives the lower bound in Theorem 132.

We prove Theorem 145 formally in Appendix 7.12, and conclude the section with a proof sketch. We consider a weaker β-oblivious adversary for $DG(n,p)$ that does the following: (a) randomly choose a subset B of βn nodes, (b) for each node $i \in B$, remove all the outgoing edges from i, and draw a number d_i independently from a different distribution over $\{0,1,\ldots,n\}$, and (c) select d_i nodes from $[n]\setminus\{i\}$ at random and add an edge from i to them. Note that both for the uncorrupted nodes and for nodes corrupted by such an adversary the out-degrees of nodes completely determine its distribution and is therefore a sufficient statistic. For a random directed graph $DG \sim DG(n,p)$, the out-degree of a node is distributed $Bin(n-1,p)$. We can think of observed degrees of n nodes of an uncorrupted directed Erdős-Rényi random graph $DG \sim DG(n,p)$ as independent samples from binomial distribution $Bin(n-1,p)$. Let $\Delta_p = 0.1\max\left(\beta\sqrt{p/n}, \beta/n\right)$. We show that for $p \le 0.5$, the TV distance between $Bin(n-1,p)$ and $Bin(n-1,p+\Delta)$ is less than 0.15β. Then we show that an adversary that chooses a random set of size $Bin(n, 0.15\beta)$, can choose the distribution of their out-degree in a way that the overall distribution of out-degrees is same for the both graphs $DG \sim DG(n,p)$ and $DG \sim DG(n, p+\Delta_p)$ after corruption. Finally, we show that even without corruption (when $\beta = 0$), no algorithm can reliably distinguish between $DG(n,p)$ and $DG(n, p+0.1\sqrt{p}/n)$.

Appendix

7.6 Additional Related Work

Beyond the aforementioned results on efficient robust estimation [47, 99], several works have focused on similar estimation tasks in a variety of related settings, including under weaker moment assumptions [48], with a larger fraction of corrupted data [29], under sparsity constraints [11, 104], for regression or other supervised learning tasks [90, 50, 123, 120], under more general robustness conditions [135], with alternate perturbation models [155], for mixture models [71, 97, 56], approaching information-theoretic barriers to accuracy [49], fast algorithms for robust estimation [34, 36, 59], and with gradient descent algorithms [35]. See [54] for a survey.

Our algorithm relies on a spectral outlier-removal technique common to several works in robust estimation. Prior to this line of work, similar approaches were employed for robust supervised learning tasks, namely learning halfspaces with malicious noise [91, 9].

There has been significant work on robust community detection in the presence of adversaries [111, 106, 135, 13]. Most of this focuses on monotone adversaries (which make only "helpful" changes to the graph) or edge corruptions. It is not clear how to define monotone adversaries for the Erdős-Rényi setting, and for our estimation problem under edge corruptions, the empirical estimator is trivially optimal in the worst case. [23] also considers a node corruption model similar to ours. However, all of the aforementioned work studies community detection in stochastic block models, which is different from our goal of parameter estimation.

Our corruption model may seem reminiscent of the classic planted clique problem [89, 82, 98], in which an algorithm must distinguish between a) $G(n, 1/2)$ and b) $G(n, 1/2)$ with the addition of a planted clique of size βn. Our adversary is given much more power (i.e., they can make arbitrary changes to the neighbourhoods of their selected nodes), though the two goals are incomparable. The planted clique problem is known to be information-theoretically solvable for any $\beta > \frac{2\log n}{n}$. However, polynomial-time algorithms are only known for $\beta > 1/\sqrt{n}$ [4], and there is strong evidence that efficient algorithms do not exist for smaller values of β [61, 62, 110, 42, 70, 14]. We have not run into issues in our setting related to this intractability, though deeper connections between our model and the planted clique problem would be interesting. Note that our task of parameter estimation is not interesting for the cases of the planted clique problem when $\beta \leq 1/\sqrt{n}$. Simply using the empirical estimator on the two instances would give error $\approx 1/n$ and $\approx 1/n + \beta^2 = O(1/n)$, which are identical up to constant factors.

Some prior works have studied robust estimation for graphical models, including Ising models [103, 122] and Bayesian networks [37]. Despite the common nomenclature, these works are rather different from our work on random graph models. Graphical models are distributions over vectors, where correlations between coordinates exist based on some latent graph structure. On the other hand, random graph models are distributions over graphs, sampled according to some underlying parameters. While existing work on graphical models necessitates many samples from the same distribution (due to parameters outnumbering the samples), our setting requires a single sample from a random graph model.

Our setting is related to the untrusted batches setting in [125], where many batches of samples are drawn from a distribution, but a constant fraction of batches may be adversarially corrupted, see also followup works [77, 76, 78] and [30, 31]. This is somewhat similar to our setting, where each batch is the set of edges connected to a node. However, the key difference is that in our setting, each edge belongs to both its two endpoint nodes, whereas in the untrusted batches setting, a sample is only associated with a single batch.

Estimation in random graph models has also been studied under the constraint of differential privacy [20, 21, 133]. Despite superficial similarities between the two settings, we are unaware of deeper technical connections.

Our setting bears some conceptual similarity to a line of robustness work focused on decomposing a matrix as a sum of a low rank matrix and a sparse matrix [28, 24, 72]. Our true parameter matrix is the rank-1 matrix pJ, where J is the all-ones matrix. However, the uncorrupted adjacency matrix is a sample from the distribution where each entry is a Bernoulli with the corresponding parameter, which is in general not low rank. Furthermore, our corruption model allows for a bounded number of rows/columns to be changed, whereas this line of work requires that the corruptions satisfy some further sparsity, such as a limited number of changed entries per row/column, or that the corruption positions are chosen randomly.

7.7 Spectral norm properties

Matrix Properties.

We state some useful properties of matrix spectral norm that will be useful in our proofs.

Lemma 146. *Let $M, M' \in \mathbb{R}^{m\times n}$, then $\|M + M'\| \le \|M\| + \|M'\|$.*

Lemma 147. *For any $M \in \mathbb{R}^{m\times n}$, $S \subseteq [m]$, $S' \subseteq [n]$, $\|M_{S\times S'}\| \le \|M\|$.*

Proof. For any unit vectors $u \in \mathbb{R}^m$ and $v \in \mathbb{R}^n$, let $\tilde{u} = u_S/\|u_S\|$ and $\tilde{v} = v_{S'}/\|v_{S'}\|$. Then

$$|u^{\intercal} M_{S\times S'} v| = |u_S^{\intercal} M v_{S'}| = \|u_S\| \cdot \|v_{S'}\| \cdot |\tilde{u}^{\intercal} M \tilde{v}| \le |\tilde{u}^{\intercal} M \tilde{v}| \le \|M\|,$$

where the second last inequality used $\|u_S\| \le \|u\| = 1$ and $\|v_{S'}\| \le \|v\| = 1$ and the last inequality used that $\tilde{u}$ and $\tilde{v}$ are unit vectors. Finally, in the above equation taking maximum over all unit vectors u, v completes the proof. ■

Lemma 148. *For any $M \in \mathbb{R}^{m\times n}$, $\|M\| \ge \frac{|\sum_{i,j} M_{i,j}|}{\sqrt{mn}}$.*

Proof. Consider $u = \frac{1}{\sqrt{m}}[1, 1, \dots, 1]$ and $v = \frac{1}{\sqrt{n}}[1, 1, \dots, 1]^T$, which are unit vectors in $\mathbb{R}^m$ and $\mathbb{R}^n$, respectively. Then $|u^T M v| = \frac{|\sum_{i,j} M_{i,j}|}{\sqrt{mn}} \le \|M\|$ by (7.3). ■

7.8 Concentration Inequalities

Lemma 149 (Chernoff bound)**.** *Let $X_1, X_2, ..., X_t \sim$ Ber (p) be t independent Bernoulli random variables. Then for any $\lambda > 0$*

$$\Pr\left[\left|\sum_{i=1}^{t} X_i - tp\right| \geq \lambda\right] \leq 2\exp\left(-\min\left(\frac{\lambda^2}{3tp}, \frac{\lambda}{3}\right)\right). \tag{7.8}$$

7.9 Proofs for Mean- and Median-Based Algorithms

In this section we provide the proofs for algorithms based on mean and medians. Throughout this section we assume that n is at least 14400 for computational simplifications.

7.9.1 Upper Bounds for Mean and Median Estimators without Corruptions

Mean estimate. The total number of edges in $G \sim G(n,p)$ is a Binomial distribution with parameters $\binom{n}{2}$ and p. Therefore, its expectation and variance are $\binom{n}{2}p$ and $\binom{n}{2}p(1-p)$, respectively. Thus, $\mathbb{E}\left[\hat{p}_{\text{mean}}(G)\right] = p$ and $\textbf{Var}(p_{\text{mean}}(G)) = p(1-p)/\binom{n}{2} \leq 4p(1-p)/n^2$. By Chebyshev's inequality,

$$\Pr\left(|\hat{p}_{\text{mean}}(G) - p| \geq 20 \cdot \frac{\sqrt{p(1-p)}}{n}\right) \leq 0.01.$$

Median estimate. We will show that with probability at least 0.995, the median degree of G is at least $(n-1)p - C$ for some constant C. The main hurdle in showing this is the fact that the node degrees $\deg(i)$ are not independent, which requires a careful analysis. For $i \in [n]$, let $Y_i := \mathbb{I}\left(\deg(i) \leq p(n-1) - 121\right)$. Then, $\sum_i Y_i$ is the number of nodes with degree at most $p(n-1) - 121$.

We establish the following bounds for $n \geq 14400$:

$$\mathbb{E}\left[\sum_i Y_i\right] \leq \frac{n}{2} - 15\sqrt{n} \tag{7.9}$$

$$\textbf{Var}\left(\sum_i Y_i\right) \leq n \tag{7.10}$$

With these, we can apply Cantelli's inequality to obtain:

$$\Pr\left(\sum_i Y_i \geq \frac{n}{2}\right) \leq \frac{\textbf{Var}\left(\sum Y_i\right)}{\textbf{Var}\left(\sum Y_i\right) + (15\sqrt{n})^2} < 0.005.$$

This shows that with probability at least 0.995 the median degree is at least $(n-1)p - 121$. By symmetry, with probability at least 0.995 the median degree is at most $(n-1)p + 121$. By the union bound, with probability at least 0.99 the error of the median estimate is at most $121/(n-1)$.

We now prove (7.9) and (7.10) to complete the proof.

To prove (7.9), note that $\deg(i) \sim Bin(n-1, p)$ and $\mathbb{E}\left[Y_i\right] = \Pr[Bin(n-1,p) \leq p(n-1) - 121]$.

We show that for any n', $\Pr[Bin(n',p) \leq pn' - 121] \leq \frac{1}{2} - \frac{15}{\sqrt{n'+1}}$, then (7.9) follows from the linearity of expectation. If $\Pr(Bin(n',p) \leq pn' - 1) \leq \frac{1}{2} - \frac{15}{\sqrt{n'+1}}$ then we are done. We prove for the case when $\Pr(Bin(n',p) \leq pn' - 1) \geq \frac{1}{2} - \frac{15}{\sqrt{n'+1}}$. By Chebyshev's inequality,

$$\Pr\left(Bin(n',p) \leq n'p - \sqrt{n}\right) \leq \frac{1}{4}.$$

Then, for $n' \geq 14400$,

$$\begin{aligned}
&\Pr\left(Bin(n',p) \in [n'p - \sqrt{n'}, pn' - 1)\right) \\
&= \Pr(Bin(n',p) \leq pn' - 1) - \Pr\left(Bin(n',p) \leq n'p - \sqrt{n}\right) \\
&\geq \frac{1}{2} - \frac{15}{\sqrt{n'+1}} - \frac{1}{4} \geq \frac{1}{8}.
\end{aligned}$$

Since the binomial distribution has a unique mode $\geq pn' - 1$, then for any $t \leq \sqrt{n'}$,

$$\Pr\left(Bin(n',p) \in [n'p - t, pn' - 1)\right) \geq \frac{t-1}{\sqrt{n'}-1} \cdot \frac{1}{8} \geq \frac{t-1}{\sqrt{n'+1}} \cdot \frac{1}{8}.$$

Since the median of $Bin(n',p)$ is $\geq n'p - 1$, [85], hence $\Pr[Bin(n',p) \leq pn' - 1] \leq 1/2$. From it subtracting the above equation for $t - 1 = 15 \cdot 8 = 120$, we get $\Pr[Bin(n',p) \leq pn' - 121] \leq \frac{1}{2} - \frac{15}{\sqrt{n'+1}}$.

We now prove (7.10). Since Y_i's are identically distributed indicator random variables,

$$\mathbf{Var}\left(\sum_i Y_i\right) = n\mathbf{Var}\left(Y_1\right) + n(n-1)\text{Cov}(Y_1,Y_2) \leq \frac{n}{4} + n(n-1)\text{Cov}(Y_1,Y_2). \quad (7.11)$$

Let $t = (n-1)p - 121$, then $Y_i = \mathbb{I}\left(deg(i) \leq t\right)$. Let Y_{12} be the number of edges from node 1 to $[n] \setminus \{2\}$ and $\mathbb{I}\left(E_{1,2}\right)$ be the indicator that edge between 1 and 2 is present. Then $Y_{12} \sim Bin(n-2,p)$. Elementary computations using the observation that $Y_1 = \mathbb{I}\left(Y_{12} \leq t-1\right) + \mathbb{I}\left(Y_{12} = t\right) \cdot \left(1 - \mathbb{I}\left(E_{1,2}\right)\right)$ show that

$$\text{Cov}(Y_1,Y_2) = p(1-p) \cdot \Pr\left(Y_{12} = t\right)^2.$$

From Stirling's approximation at $t = np$, we have $\Pr\left(Y_{12} = t\right) \leq 1/\sqrt{\pi p(1-p)(n-2)}$, and therefore,

$$\text{Cov}(Y_1,Y_2) = p(1-p) \cdot \Pr\left(Y_{12} = t\right)^2 \leq \frac{1}{\pi(n-2)} \leq \frac{1}{3n}$$

for $n > 120^2$. Plugging this in (7.11) proves (7.10).

7.9.2 Lower Bounds for Mean and Median Estimators under Corruptions

We will prove the $\beta/2$ lower bound for the mean and median estimates. Consider the following oblivious adversary $\mathcal{A}$.

- Pick a random subset $B \subset [n]$ of size βn.
- Let $\mathcal{A}_1(G)$ be the graph obtained by adding all edges (u, v) that have at least one node in B to the graph G, and let $\mathcal{A}_2(G)$ be the graph obtained by removing all edges that have at least one node in B from the graph G.
- Output $\mathcal{A}_1(G)$ or $\mathcal{A}_2(G)$ chosen uniformly at random.

Any node in $\mathcal{A}_1(G)$ has degree at least βn more than the corresponding node in $\mathcal{A}_2(G)$. Therefore, $|\hat{p}_{\text{mean}}(\mathcal{A}_1(G)) - \hat{p}_{\text{mean}}(\mathcal{A}_2(G))| \geq \beta$, and $|\hat{p}_{\text{med}}(\mathcal{A}_1(G)) - \hat{p}_{\text{med}}(\mathcal{A}_2(G))| \geq \beta$. Therefore by the triangle inequality, with probability 0.5, $|\hat{p}_{\text{mean}}(\mathcal{A}(G)) - p| \geq \beta/2$, and $|\hat{p}_{\text{med}}(\mathcal{A}(G)) - p| \geq \beta/2$.

7.9.3 Upper Bounds for Prune-then-Mean/Median Algorithms

Recall the prune-then mean/median algorithm in Algorithm 15. We remove $c\beta$ fraction of nodes with the highest and lowest degrees, and then output the median (or mean) of the remaining subgraphs. We restate the performance bound of the algorithm here.

Theorem 135. *For $c \geq 1$ and $0 < \beta \cdot c < 0.25$, the prune-then-mean and prune-then-median estimators described in Algorithm 15 prune $2c\beta n$ nodes in total and with probability $1 - n^{-2}$ estimates p to an accuracy $\mathcal{O}\big(c\beta^2 + \frac{\log n}{n}\big)$ and $\mathcal{O}\big(c\beta + \sqrt{\frac{\log n}{n}}\big)$, respectively.*

Proof. Let $G \sim G(n, p)$. By Chernoff bound (Lemma 149) and the union bound, with probability $\geq 1 - 1/n^2$,

$$\deg(i) \in \Big(np - 100\sqrt{n\log n}, np + 100\sqrt{n\log n}\Big)$$

for all nodes $i \in [n]$ of G. We condition on this event.

Suppose an adversary converts G into $\mathcal{A}(G)$ by corrupting nodes in $B \subset [n]$ with $|B| \leq \beta n$. Note that the degree of a node in $F = [n] \setminus B$ cannot change by more than βn. Therefore, for all nodes $i \in F$ in $\mathcal{A}(G)$,

$$\deg(i) \in \left(np - 100\sqrt{n\log n} - \beta n, np + 100\sqrt{n\log n} + \beta n\right). \tag{7.12}$$

Therefore, at most βn nodes do not satisfy (7.12). Since we remove $c\beta n$ nodes with the highest and the lowest degrees for $c \geq 1$ all such nodes are pruned. The degree of any node not pruned decreases by at most $2c\beta n$, and after pruning all degrees are in the following interval

$$\left(np - 100\sqrt{n\log n} - (2c+1)\beta n, np + 100\sqrt{n\log n} + \beta n\right). \tag{7.13}$$

We can rewrite this interval as follows

$$\left(n(1-2c\beta)p - 100\sqrt{n\log n} + (2cp - 2c - 1)\beta n, n(1-2c\beta)p + 100\sqrt{n\log n} + (2cp+1)\beta n\right).$$

The prune-then-median estimator outputs one of these degrees (normalized), and its error is at most

$$\left(\frac{100\sqrt{n\log n} + (4c+1)\beta n}{(1-2c\beta)n}\right) = \mathcal{O}\left(\sqrt{\frac{\log n}{n}} + c\beta\right).$$

We now bound the performance of prune-then-mean estimator. Let $V' \subseteq [n]$ be the nodes that are not pruned, so $|V'| = (1-2c\beta)n$. Let $F^p := V' \cap F$ and $B^p := V' \cap B$ be the uncorrupted and corrupted nodes that remain after pruning. We have $|B^p| \leq |B| \leq \beta n$ and $|F^p| \geq (1-(2c+1)\beta)n$.

There are three types of edges among the nodes in V': (i) $\mathcal{E}_1$: edges whose both end points are good nodes (in F^p), (ii) $\mathcal{E}_2$: edges with at least one end point in B^p. The mean estimator outputs

$$\frac{|\mathcal{E}_1| + |\mathcal{E}_2|}{\binom{|V'|}{2}}.$$

Its error is at most

$$\begin{aligned}\left|\frac{|\mathcal{E}_1| + |\mathcal{E}_2|}{\binom{|V'|}{2}} - p\right| &= \left|\frac{|\mathcal{E}_1| - \binom{|F^p|}{2}p}{\binom{|V'|}{2}}\right| + \left|\frac{|\mathcal{E}_2| - (|V'| - |F^p|)((|V'| + |F^p| - 1)/2)p}{\binom{|V'|}{2}}\right| \\ &= \left|\frac{|\mathcal{E}_1| - \binom{|F^p|}{2}p}{\binom{|V'|}{2}}\right| + \left|\frac{|\mathcal{E}_2| - |B^p|((|V'| + |F^p| - 1)/2)p}{\binom{|V'|}{2}}\right|\end{aligned}$$

We will bound each term individually. Since the subgraph $F^p \times F^p$ between the uncorrupted nodes remains unaffected from the original graph G, then Theorem 152 implies that, with probability $\geq 1 - 3n^{-2}$,

$$\left|\frac{|\mathcal{E}_1|}{\binom{|F^p|}{2}} - p\right| = \mathcal{O}\left(\max\left\{c\beta\sqrt{\frac{\ln(e/c\beta)}{n}}, \frac{c\beta\log n}{n}, \frac{1}{n}\right\}\right) \leq \mathcal{O}\left(c\beta^2 + \frac{\log n}{n}\right).$$

Therefore,

$$\left||\mathcal{E}_1| - \binom{|F^p|}{2}p\right| = \binom{|F^p|}{2} \cdot O\left(c\beta^2 + \frac{\log n}{n}\right) \leq \binom{|V'|}{2} \cdot O\left(c\beta^2 + \frac{\log n}{n}\right).$$

This shows that the first error term is at most $O\left(c\beta^2 + \frac{\log n}{n}\right)$.

We now consider the second term. Note that $|n - (|V'| + |F^p| - 1)/2| \leq 3c\beta n$. By the triangle inequality,

$$\left||\mathcal{E}_2| - \frac{1}{2} \cdot |B^p|(|V'| + |F^p| - 1)p\right| \leq ||\mathcal{E}_2| - |B^p| \cdot np| + 3\beta np \cdot |B^p|. \tag{7.14}$$

Let $\deg(i)$ be the degree of node i after pruning. By the triangle inequality adding and subtracting $\sum_{i\in B^p}\deg(i)$ to the first term we obtain,

$$||\mathcal{E}_2| - |B^p|\cdot np| \le \left||\mathcal{E}_2| - \sum_{i\in B^p}\deg(i)\right| + \sum_{i\in B^p}|\deg(i) - np|.$$

Now note that $|\mathcal{E}_2|$ is the number of edges with at least one endpoint in B^p. Therefore $\left||\mathcal{E}_2| - \sum_{i\in B^p}\deg(i)\right|$ is the number of edges inside $B^p \times B^p$ and is at most $|B^p|^2$. For the second term we use the fact that each node in B^p satisfies (7.13), and $|B^p| \le \beta n$. This gives

$$\begin{aligned}||\mathcal{E}_2| - |B^p|\cdot np| &\le \left||\mathcal{E}_2| - \sum_{i\in B^p}\deg(i)\right| + \sum_{i\in B^p}|\deg(i) - np| \\ &\le |B^p|\cdot\left(100\sqrt{n\log n} + (2c+2)\beta n\right).\end{aligned}$$

Plugging this along with the fact that $|B^p| \le \beta n$ in (7.14), we obtain

$$\left||\mathcal{E}_2| - \frac{1}{2}\cdot|B^p|(|V'| + |F^p| - 1)p\right| \le \beta n\cdot\left(\left(100\sqrt{n\log n} + (5c+2)\beta n\right)\right).$$

Since $\binom{|V'|}{2} > (n/2)^2$, the second term can be bounded by

$$\mathcal{O}\left(4\beta\cdot\left(\sqrt{\frac{\log n}{n}} + (5c+2)\beta\right)\right) = \mathcal{O}\left(c\beta^2 + \frac{\log n}{n}\right),$$

thus proving the result.

■

7.9.4 Lower Bounds for Prune-then-Mean/Median Algorithms

We will prove the following result showing the tight dependence of the upper bounds on β.

Theorem 136. *Let $p = 0.5$, $\beta > 100\cdot\sqrt{\log n/n}$, and $c > 0$ be such that $c\beta < 0.25$. There* *exists an adversary such that with probability at least 0.99, the prune-then-median estimate that deletes $c\beta n$ satisfies $|\hat{p}(\mathcal{A}(G)) - p| \geq C'\beta$, and the prune-then-mean estimate satisfies $|\hat{p}(\mathcal{A}(G)) - p| \geq C'\beta^2$.*

Let $G \sim G(n, 0.5)$. The oblivious adversary $\mathcal{A}$ operates as follows. It partitions G into five random sets B, S_0, S_1, S_2, and S_3 with $|B| = \beta n$, $|S_0| = c\beta n, |S_1| = c\beta n, |S_2| = \frac{2}{3}(1-(2c+1)\beta)n, |S_3| = \frac{1}{3}(1-(2c+1)\beta)n$.

- Remove all edges with at least one endpoint in B.
- Remove all edges between S_0 and B.
- Add all edges between S_1 and B.
- Connect each node in B to each node in S_2 independently with probability $3/5$.
- Connect each node in B to each node in S_3 independently with probability $3/10$.
- Connect nodes within B to each other with probability $3/5$.

By the Chernoff bound (Lemma 149) and the union bound, we obtain the following bounds on the node degrees in $\mathcal{A}(G)$.

Lemma 150. *In $\mathcal{A}(G)$, the following hold with probability at least $1 - 3n^{-3}$*

$$\begin{aligned}
\deg(u) &= n\left(\frac{1}{2} + \frac{\beta}{10}\right) \pm 4\sqrt{n \log n} && \textit{for } u \in B,\\
\deg(u) &= n \cdot \left(\frac{1}{2} - \frac{\beta}{2}\right) \pm 4\sqrt{n \log n} && \textit{for } u \in S_0,\\
\deg(u) &= n \cdot \left(\frac{1}{2} + \frac{\beta}{2}\right) \pm 4\sqrt{n \log n} && \textit{for } u \in S_1,\\
\deg(u) &= n \cdot \left(\frac{1}{2} + \frac{\beta}{10}\right) \pm 4\sqrt{n \log n} && \textit{for } u \in S_2,\\
\deg(u) &= n \cdot \left(\frac{1}{2} - \frac{\beta}{5}\right) \pm 4\sqrt{n \log n} && \textit{for } u \in S_3.
\end{aligned}$$

Since $\beta > 100\sqrt{\log n/n}$, the nodes in S_0 are the $c\beta n$ nodes with the lowest degrees and the nodes in S_1 are the $c\beta n$ nodes with the highest degrees, and they are pruned by the algorithm. Now since the sets S_0 and S_1 were randomly chosen ahead of time, in the pruned graph, once again by the Chernoff bound (Lemma 149) and the union bound, the following holds with probability at least $1 - 3n^{-3}$

$$\begin{aligned}
\deg(u) &= n\left(\frac{1-2c\beta}{2} + \frac{\beta}{10}\right) \pm 8\sqrt{n\log n} && \text{for } u \in B,\\
\deg(u) &= n \cdot \left(\frac{1-2c\beta}{2} + \frac{\beta}{10}\right) \pm 8\sqrt{n\log n} && \text{for } u \in S_2,\\
\deg(u) &= n \cdot \left(\frac{1-2c\beta}{2} - \frac{\beta}{5}\right) \pm 8\sqrt{n\log n} && \text{for } u \in S_3.
\end{aligned}$$

Since we assume that $c\beta < 0.25$, there are more nodes in S_3 than in $S_2 \cup B$ and every node in $S_2 \cup B$ had a higher degree than any node in S_3. Therefore a node in S_3 is chosen as the median node, thus deviating from the median degree by at least $\beta/5 \pm 8\sqrt{\log n/n} > \beta/10$ for $\beta > 100\sqrt{\log n/n}$. This proves the lower bound for prune-then-median estimate.

Now for the prune-then-mean estimate, note that each edge that remains after pruning is chosen at random, independent of all other edges. The total expected number of edges after pruning is $\frac{1}{2} \cdot \frac{n^2(1-2c\beta)^2}{2} + \frac{n^2\beta^2}{20}$ and the variance is at most $n^2/4$. Therefore, the total error of the prune-then-mean estimate is at least $\beta^2/20 \pm O(1/n)$, and since $\beta > 100\sqrt{\log n/n}$, the error is at least $\beta^2/40$.

7.10 Upper Bound Proofs

7.10.1 Proof of Lemma 137

Proof. We first prove Equation (7.6). From the triangle inequality

$$\|(A - p_{F'})_{F' \times F'}\| \le \|(A - p)_{F' \times F'}\| + |p - p_{F'}| \cdot F'.$$

From Lemma 148 we have

$$\|(A-p)_{F'\times F'}\| \geq |F'| \cdot |p_{F'} - p|.$$

Combining the above two equations with regularity proves Equation (7.6),

$$\|(A-p_{F'})_{F'\times F'}\| \leq 2\|(A-p)_{F'\times F'}\| \leq 2n \cdot \eta(p,n),$$

where the last inequality follows from regularity condition 2.

Next, Equation (7.7) is obtained by using $F' = F''$ in regularity condition 3 and $|F'| \geq n/2$. ■

7.10.2 Proof of Theorem 138

Proof. In a β-corrupted graph the set of uncorrupted nodes F has size $\geq (1-\beta)n$, which proves regularity condition 1.

We use the following bound on the spectral norm of a centered version of $\tilde{A}$, which follows from Remark 3.13 of [12].

Lemma 151. *Let $\tilde{A}$ be the adjacency matrix of a sample from $G(n,p)$ and I be the $n \times n$ identity matrix. There exist a universal constant c such that with probability at least $1 - n^{-2}$, $\|\tilde{A} - p + pI\| \leq c\sqrt{np(1-p) + \ln n}$.*

To establish regularity condition 2, note that A and $\tilde{A}$ agree on $(i,j) \in F \times F$, and therefore by Lemma 147 and Lemma 151, $\|(A-p)_{F'\times F'}\| = \|(\tilde{A}-p)_{F'\times F'}\| \leq \|(\tilde{A}-p)\| \leq \|(\tilde{A}-p+pI)\| + p\|I\| \leq c\sqrt{np(1-p)+\ln n} + 1$.

The following theorem implies regularity condition 3. The proof uses a Chernoff and union bound style argument, and is provided in Section 7.11.

Theorem 152. *Let $\tilde{A}$ be the adjacency matrix of a sample from $G(n, p)$. With probability at least $1 - 3n^{-2}$, simultaneously for all $\alpha \in [0, \frac{1}{2}]$, we have*

$$\max_{|S|,|S'| \in C_\alpha} \left| \sum_{i \in S,\, j \in S'} (\tilde{A}_{i,j} - p) \right| \leq 6 \max \left\{ 16\alpha n \sqrt{pn \ln \frac{e}{\alpha}}, 60\alpha n \ln \frac{e}{\alpha}, 5n\sqrt{p \ln(en)} \right\},$$

where we define $C_\alpha := [0, \alpha n] \cup [n - \alpha n, n]$.

■

7.10.3 Proofs for Lemma 142

Proof. We first require the following lemma, which lower bounds the spectral norm of a matrix $(A - p_S)_{S \times S}$ primarily in terms of the empirical estimates of p corresponding to the submatrices induced by S and $S \cap F$. The proof appears in Section 7.10.4.

Lemma 153. *Given any symmetric matrix A, and subsets $S, F \subseteq [n]$*

$$\|(A - p_S)_{S \times S}\| \geq \frac{|p_{S \cap F} - p_S| \cdot |S \cap F|}{3} \cdot \min \left\{ \sqrt{\frac{|S \cap F|}{|S \cap F^c|}}, \frac{|S \cap F|}{|S \cap F^c|} \right\}.$$

For $\alpha_1 \leq 1/60$ and $|S| \geq (1 - 9\alpha_1)n$, we can deduce that $|S \cap F| \geq n(1 - 10\alpha_1) \geq 5n/6$ and $|S \cap F^c| \leq |F^c| \leq \alpha_1 n \leq n/60$. Therefore, $|S \cap F^c|/|S \cap F| \leq 1/50$. By Lemma 153,

$$|S \cap F| \cdot |p_{S \cap F} - p_S| \leq 3\|(A - p_S)_{S \times S}\| \max \left\{ \sqrt{\frac{|S \cap F^c|}{|S \cap F|}}, \frac{|S \cap F^c|}{|S \cap F|} \right\} \leq \frac{3}{\sqrt{50}} \cdot \|(A - p_S)_{S \times S}\|.$$

Applying Equation (7.6) with $F' = S \cap F$, we have

$$\|(A - p_{S \cap F})_{(S \cap F) \times (S \cap F)}\| \leq 2n \cdot \eta(p, n).$$

This implies $\|(A - p_{S\cap F})_{(S\cap F)\times(S\cap F)}\| \leq 0.1\|(A - p_S)_{S\times S}\|$. Next, by the triangle inequality,

$$\begin{aligned}\|(A - p_S)_{(S\cap F)\times(S\cap F)}\| &\leq \|(A - p_{S\cap F})_{(S\cap F)\times(S\cap F)}\| + |S \cap F| \cdot |p_{S\cap F} - p_S| \\ &\leq \left(\frac{1}{10} + \frac{3}{\sqrt{50}}\right)\|(A - p_S)_{S\times S}\|.\end{aligned}$$

To interpret the derivation above: we have reasoned that if the spectral norm of $(A-p_S)_{S\times S}$ is large, the contribution due to $S \cap F$ (i.e., the submatrix induced by the intersection with the good nodes) is relatively small. This suggests that any top eigenvector must place a constant mass on $S \cap F^c$. Indeed, the following theorem formalizes this reasoning, showing that the normalized top eigenvector contains significant weight in this complementary subset of indices. The proof appears in Section 7.10.5.

Theorem 154. *Let M be a non-zero $n \times n$ real symmetric matrix such that for some set $S \subseteq [n]$ and $0 \leq \rho \leq 1$ we have $\|M_{S\times S}\| \leq \rho\|M\|$. Let v be any normalized top eigenvector of M. Then $\|v_{S^c}\|^2 \geq \frac{(1-\rho)^2}{1+(1-\rho)^2}$.*

Applying Theorem 154 with $\rho = \frac{1}{10} + \frac{3}{\sqrt{50}}$ implies that $\|v_{S\setminus(S\cap F)}\|^2 = \|v_{S\cap F^c}\|^2 \geq \frac{(1-\rho)^2}{1+(1-\rho)^2} > 0.15$. ■

7.10.4 Proof of Lemma 153

First note that

$$0 = \sum_{i,j\in S}(A_{i,j} - p_S) = \sum_{i,j\in S\cap F}(A_{i,j} - p_S) + \sum_{i,j\in S\cap F^c}(A_{i,j} - p_S) + 2\sum_{i\in S\cap F,\, j\in S\cap F^c}(A_{i,j} - p_S).$$

Therefore,

$$\left|\sum_{i,j\in S\cap F}(A_{i,j} - p_S)\right| \leq \left|\sum_{i,j\in S\cap F^c}(A_{i,j} - p_S)\right| + 2\left|\sum_{i\in S\cap F,\, j\in S\cap F^c}(A_{i,j} - p_S)\right|.$$

Hence,

$$\frac{|\sum_{i,j\in S\cap F}(A_{i,j}-p_S)|}{3}\le \max\left\{\left|\sum_{i,j\in S\cap F^c}(A_{i,j}-p_S)\right|,\left|\sum_{i\in S\cap F,\, j\in S\cap F^c}(A_{i,j}-p_S)\right|\right\}. \tag{7.15}$$

From Lemma 147 , Lemma 148 and the above inequality, it follows that

$$\begin{aligned}
\|(A-p_S)_{S\times S}\| &\ge \max\left\{\|(A-p_S)_{(S\cap F^c)\times(S\cap F^c)}\|,\|(A-p_S)_{(S\cap F^c)\times(S\cap F)}\|\right\} &(7.16)\\
&\ge \max\left\{\frac{|\sum_{i,j\in S\cap F^c}(A_{i,j}-p_S)|}{|S\cap F^c|},\frac{|\sum_{i\in S\cap F^c,\, j\in S\cap F}(A_{i,j}-p_S)|}{\sqrt{|S\cap F|\cdot|S\cap F^c|}}\right\} &(7.17)\\
&\ge \min\left\{\frac{|\sum_{i,j\in S\cap F}(A_{i,j}-p_S)|}{3|S\cap F^c|},\frac{|\sum_{i,j\in S\cap F}(A_{i,j}-p_S)|}{3\sqrt{|S\cap F|\cdot|S\cap F^c|}}\right\} &(7.18)\\
&= \frac{|\sum_{i,j\in S\cap F}(A_{i,j}-p_S)|}{3\sqrt{|S\cap F|\cdot|S\cap F^c|}}\cdot\min\left\{\sqrt{\frac{|S\cap F|}{|S\cap F^c|}},1\right\}\\
&= \frac{|p_{S\cap F}-p_S||S\cap F|}{3}\cdot\min\left\{\frac{|S\cap F|}{|S\cap F^c|},\sqrt{\frac{|S\cap F|}{|S\cap F^c|}}\right\},
\end{aligned}$$

where (7.16) is from Lemma 147, (7.17) follows from Lemma 148, (7.18) from (7.15).

7.10.5 Proof of Theorem 154

Since eigenvalues of symmetric matrices are real, let $v\in\mathbb{R}^n$ be the normalized top eigenvector of M with eigenvalue $\lambda\in\mathbb{R}$ such that $Mv=\lambda v$ and $\|M\|=|\lambda|$. Since $Mv=\lambda v$, we have $M_{S\times[n]}\,v\;=\lambda v_S$, and

$$M_{S\times[n]}\,v\;=M_{S\times S}\,v_S+M_{S\times S^c}\,v_{S^c} \tag{7.19}$$

By Lemma 146 on (7.19),

$$\begin{aligned} & \|M_{S\times[n]}\, v\| \leq \|M_{S\times S}\, v_S\| + \|M_{S\times S^c}\, v_{S^c}\| \\ \Rightarrow \quad & |\lambda| \cdot \|v_S\| \leq \rho|\lambda| \cdot \|v_S\| + |\lambda| \cdot \|v_{S^c}\| \qquad (7.20) \\ \Rightarrow \quad & (1-\rho)\|v_S\| \leq \|v_{S^c}\| \\ \Rightarrow \quad & (1-\rho)^2\|v_S\|^2 \leq \|v_{S^c}\|^2 \end{aligned}$$

where (7.20) uses the assumption of the lemma. Finally using $\|v_S\|^2 + \|v_{S^c}\|^2 = 1$ gives the bound.

7.10.6 An Approximate Top Eigenvector Suffices

As discussed in Remark 4, computing an exact top eigenvector in Algorithm 16 may be costly. The guarantees associated with this top eigenvector are quantified in Lemma 142, which relies upon Theorem 154. In this section, we prove a variant of Theorem 154, which works with an approximate rather than an exact top eigenvector. By repeating the proof of Lemma 142 with Lemma 155 swapped in place of Theorem 154, we can instead use approximate top eigenvector procedures, reducing the runtime.

Lemma 155. *Let M be a nonzero $n \times n$ real matrix such that for some set $S \subset [n]$ we have $\|M_{S\times S}\| \leq 0.53\|M\|$. Let $v \in \mathbb{R}^n$ be a unit vector such that $\|Mv\| \geq 0.99\|M\|$, then $\|v_{S^c}\|^2 \geq \frac{1}{8}$.*

Proof. Let $u = Mv$. Note that $M_{S\times[n]}\, v = u_S$ and $M_{S^c\times[n]}\, v = u_{S^c}$, therefore

$$v^T\, M\, v = v^T\left(M_{S\times[n]} + M_{S^c\times[n]}\right) v = v^T(u_S + u_{S^c}) = v_S^T\, u_S + v_{S^c}^T\, u_{S^c}.$$

Then by the triangle inequality,

$$\begin{aligned}
& |v^T\, M\, v| \leq \|v_S\| \cdot \|u_S\| + \|v_{S^c}\| \cdot \|u_{S^c}\| \\
\Rightarrow \quad & 0.99\|M\| \leq \|v_S\| \cdot \|u_S\| + \|v_{S^c}\| \cdot \|u_{S^c}\| \\
\Rightarrow \quad & 0.99\|M\| \leq \sqrt{1 - \|v_{S^c}\|^2} \cdot \|u_S\| + \|v_{S^c}\| \cdot \sqrt{\|M\|^2 - \|u_S\|^2}.
\end{aligned}$$

In the last line, we used the fact that $\|u\| \leq \|M\| \cdot \|v\| = \|M\|$ and $\|u\|^2 = \|u_S\|^2 + \|u_{S^c}\|^2$. Rearranging this expression, it is easy to show that in the case $\|u_S\|^2 \leq \frac{3\|M\|^2}{4}$, the inequality is violated if $||v_{S^c}\|^2 \leq \frac{1}{8}$. Therefore, $\|u_S\|^2 \leq \frac{3\|M\|^2}{4}$ implies $\|v_{S^c}\|^2 \geq \frac{1}{8}$.

To prove the lemma, we must handle the remaining case: we show that if $\|u_S\|^2 \geq \frac{3\|M\|^2}{4}$, then $\|v_{S^c}\|^2 \geq \frac{1}{8}$.

Note that

$$M_{S\times[n]}\, v \;= M_{S\times S}\, v_S + M_{S\times S^c}\, v_{S^c}.$$

Then

$$\begin{aligned}
& \|M_{S\times[n]}\, v\| \leq \|M_{S\times S}\, v_S\| + \|M_{S\times S^c}\, v_{S^c}\| \\
\Rightarrow \quad & \|u_S\| \leq 0.53\|M\| \cdot \|v_S\| + \|M\| \cdot \|v_{S^c}\| \\
\Rightarrow \quad & \|u_S\|^2 \leq 2(0.53^2)\|M\|^2 \cdot \|v_S\|^2 + 2\|M\|^2 \cdot \|v_{S^c}\|^2 \\
\Rightarrow \quad & \|u_S\|^2 \leq 0.5618\|M\|^2(1 - \|v_{S^c}\|^2) + 2\|M\|^2 \cdot \|v_{S^c}\|^2 \\
\Rightarrow \quad & \|u_S\|^2 \leq 0.5618\|M\|^2 + 1.4382\|M\|^2 \cdot \|v_{S^c}\|^2.
\end{aligned}$$

When $\|u_S\|^2 \geq 3\|M\|^2/4$, the above equation implies $\|v_{S^c}\|^2 \geq 1/8$, which completes the proof of the lemma. ■

7.10.7 Proofs for Theorem 143

Before proving the Theorem we state and prove two auxiliary lemmas. The first lemma shows that the average of entries of all *small submatrices* of $S^* \times S^*$ are close to p_{S^*}.

Lemma 156. *Assume the conditions of Theorem 143 hold. For all $S_1, S_2 \subseteq S^*$ with $|S_1|, |S_2| \leq 3\alpha_1 n$ we have*

$$\left| \sum_{i \in S_1,\, j \in S_2} (A_{i,j} - p_{S^*}) \right| \leq 60\alpha_1 n^2 \cdot \eta(p, n).$$

Proof. Since $\|(A - p_{S^*})_{S^* \times S^*}\| \leq 20n \cdot \eta(p, n)$, and using Lemma 147 and Lemma 148, we get

$$\begin{aligned}\left| \sum_{(i,j) \in S_1 \times S_2} (A_{i,j} - p_{S^*}) \right| &\leq \sqrt{|S_1| \cdot |S_2|} \cdot \|(A - p_{S^*})_{S_1 \times S_2}\| \\ &\leq 3\alpha_1 n \|(A - p_{S^*})_{S^* \times S^*}\| \\ &\leq 60\alpha_1 n^2 \cdot \eta(p, n).\end{aligned}$$

■

We now show that all the nodes in S^f have normalized degree close to p_{S^*}.

Lemma 157. *Assume the conditions of Theorem 143 hold, and let S^f be the output of Algorithm 17, then for every node $i \in S^f$,*

$$|p_{S^*}^{(i)} - p_{S^*}| \leq \Big(\frac{2\kappa(13\alpha_1, p, n)}{\alpha_1} + 210\eta(p, n) \Big).$$

Proof. Suppose to the contrary that after $3\alpha_1 n$ nodes are deleted by Algorithm 17, there is a node $i \in S^f$ such that $|p_{S^*}^{(i)} - p_{S^*}| > \Big(\frac{2\kappa(13\alpha_1, p, n)}{\alpha_1} + 210\eta(p, n) \Big)$. Therefore, all the nodes deleted by Algorithm 17 are such that $|p_{S^*}^{(i)} - p_{S^*}| > \Big(\frac{2\kappa(13\alpha_1, p, n)}{\alpha_1} + 210\eta(p, n) \Big)$. Let D^+ be the set of nodes deleted by Algorithm 17 such that $p_{S^*}^{(i)} > p_{S^*}$ for $i \in D^+$ and D^- be the set of deleted nodes i such that $p_{S^*}^{(i)} < p_{S^*}$ for $i \in D^-$. Since $|D^+| + |D^-| = 3\alpha_1 n$ and $|(D^+ \cup D^-) \setminus F| \leq |F^c| \leq \alpha_1 n$, we have that $|D^+ \cap F| \geq \alpha_1 n$ or $|D^- \cap F| \geq \alpha_1 n$. Suppose $|D^+ \cap F| \geq \alpha_1 n$. Let $F' = D^+ \cap F$. Then, using $|F'| \geq \alpha_1 n$ and $|S^*| > n/2$, we have

$$\begin{aligned}\sum_{i \in F', j \in S^*} (A_{i,j} - p_{S^*}) = \sum_{i \in F'} |S^*|(p_{S^*}^{(i)} - p_{S^*}) &> |F'| \cdot |S^*| \cdot \Big(\frac{2\kappa(13\alpha_1, p, n)}{\alpha_1} + 210\eta(p,n)\Big) \\ &\geq n^2 \kappa(13\alpha_1, p, n) + 105|F'|n \cdot \eta(p,n).\end{aligned}$$

Now, note that

$$\begin{aligned}&\sum_{i \in F', j \in S^*} (A_{i,j} - p_{S^*}) \\ &= \sum_{i \in F', j \in S^* \cap F} (A_{i,j} - p) + |F'| \cdot |S^* \cap F| \cdot (p - p_{S^*}) + \sum_{i \in F', j \in S^* \cap F^c} (A_{i,j} - p_{S^*})\end{aligned}$$

By Lemma 156 with $S_1 = F'$ and $S_2 = S^* \cap F^c$ the last term in the expression above is at most $60\,\alpha_1\, n^2 \cdot \eta(p,n)$. For the second term note that $|p - p_{S^*}| < 45 \cdot \eta(p,n)$ and therefore, the second term is at most $45|F'|n \cdot \eta(p,n)$. Finally using regularity condition 3 with F' and $F'' = S^* \cap F$ and $\alpha_2 = 13\alpha_1$ bounds the first term by $n^2 \cdot \kappa(13\alpha_1, p, n)$. Combining the three bounds and using $|F'| \geq \alpha_1 n$,

$$\begin{aligned}\sum_{i \in F', j \in S^*} (A_{i,j} - p_{S^*}) &\leq n^2 \kappa(13\alpha_1, p, n) + (45|F'| + 60\alpha_1 n)n \cdot \eta(p,n) \\ &\leq n^2 \kappa(13\alpha_1, p, n) + 105|F'|n \cdot \eta(p,n),\end{aligned}$$

This shows the contradiction and completes the proof for the case $|D^+ \cap F| > \alpha_1 n$. The case when $|D^- \cap F| > \alpha_1 n$ has a similar argument and is omitted. ■

Combining these lemmas appropriately allows us to conclude our main result on the guarantees of Algorithm 17.

Proof of Theorem 143. We will partition $S^f \times S^f$ into the following groups and bound each term separately.

$$\sum_{i,j\in S^f} A_{i,j} = \sum_{i,j\in S^f\cap F} A_{i,j} + 2 \sum_{i\in S^f, j\in S^f\cap F^c} A_{i,j} - \sum_{i,j\in S^f\cap F^c} A_{i,j}.$$

Since $p_{S^f} = \sum_{i,j\in S^f} A_{i,j}/|S^f|^2$, by the triangle inequality,

$$\big|p_{S^f} - p\big| \le \left|\frac{\sum_{i,j\in S^f\cap F}(A_{i,j}-p)}{|S^f|^2}\right| + 2\left|\frac{\sum_{i\in S^f\, j\in S^f\cap F^c}(A_{i,j}-p)}{|S^f|^2}\right| + \left|\frac{\sum_{i,j\in S^f\cap F^c}(A_{i,j}-p)}{|S^f|^2}\right|.$$

For the first term, $|S^f\cap F| \ge (1-13\alpha_1)n \ge n/2$. Using Equation (7.7) with $F' = S^f\cap F$ and $\alpha_2 = 13\alpha_1$,

$$\left|\frac{\sum_{i,j\in S^f\cap F}(A_{i,j}-p)}{|S^f|^2}\right| \le \left|\frac{\sum_{i,j\in S^f\cap F}(A_{i,j}-p)}{|S^f\cap F|^2}\right| = |p_{S^f\cap F} - p| \le 4\kappa(13\alpha_1, p, n).$$

Since $|S^f \times (S^f\cap F^c)| \le \alpha_1 n^2$, and $|S^f| \ge n/2$, by the triangle inequality, the second term is bounded by

$$\begin{aligned}
&2\left|\frac{\sum_{i\in S^f\, j\in S^f\cap F^c}(A_{i,j}-p)}{|S^f|^2}\right| && (7.21)\\
&\le 2\left|\frac{\sum_{i\in S^f\, j\in S^f\cap F^c}(A_{i,j}-p_{S^*})}{n^2/4}\right| + \frac{2\alpha_1 n^2}{n^2/4}|p_{S^*} - p| \\
&\le 8\left|\frac{\sum_{i\in S^*\, j\in S^f\cap F^c}(A_{i,j}-p_{S^*})}{n^2}\right| + 8\left|\frac{\sum_{i\in S^*\setminus S^f\, j\in S^f\cap F^c}(A_{i,j}-p_{S^*})}{n^2}\right| + 8\alpha_1\cdot 45\cdot \eta(p,n).
\end{aligned}$$

Since $|S^*\setminus S^f| \le 3\alpha_1 n$ and $|S^f\cap F^c| \le \alpha_1 n$, by taking $S_1 = S^*\setminus S^f$ and $S_2 = S^f\cap F^c$

in Lemma 156 bounds the second term above by $8(60\alpha_1 \cdot \eta(p,n))$. For the first term,

$$\begin{aligned}\left|\frac{\sum_{i\in S^*\, j\in S^f\cap F^c}(A_{i,j}-p_{S^*})}{n^2}\right| &\leq \sum_{j\in S^f\cap F^c}\left|\frac{\sum_{i\in S^*}(A_{i,j}-p_{S^*})}{n^2}\right| \\ &\leq \sum_{j\in S^f\cap F^c}\frac{|S^*|}{n^2}\left|\frac{\sum_{i\in S^*}(A_{i,j}-p_{S^*})}{|S^*|}\right| \\ &\leq \sum_{j\in S^f\cap F^c}\frac{1}{n}|p_{S^*}^{(j)}-p_{S^*}| \\ &\leq \alpha_1\cdot\Big(\frac{2\kappa(13\alpha_1,p,n)}{\alpha_1}+210\eta(p,n)\Big),\end{aligned}$$

where we use Lemma 157 and $|S^f \cap F^c| \leq \alpha_1 n$.

For the final term, since $|(S^f\cap F^c)\times(S^f\cap F^c)| \leq \alpha_1^2 n^2$,

$$\left|\frac{\sum_{i,j\in S^f\cap F^c}(A_{i,j}-p)}{|S^f|^2}\right| \leq \left|\frac{\sum_{i,j\in S^f\cap F^c}(A_{i,j}-p_{S^*})}{|S^f|^2}\right| + |p_{S^*}-p|\cdot\frac{|S^f\cap F^c|^2}{|S^f|^2},$$

which can be bounded again by taking $S_1 = S_2 = S^f \cap F^c$ in Lemma 156. ■

7.10.8 Putting Things Together: Proof of Theorem 130

We now combine our methods from previous sections to prove our main upper bound. This primarily consists of running Algorithm 16 followed by Algorithm 17, as described by Algorithm 18 and quantified by Theorem 158. For technical reasons, to get the correct scaling of the error with respect to the parameter p, we run this procedure on both the graph and its complement, and output the appropriate of the two estimates. This is described in Algorithm 19, and quantified in Theorem 159. This theorem implies our upper bound (Theorem 130).

Theorem 158. *Suppose $\alpha_1 \in [\frac{1}{n}, \frac{1}{60}]$ and let A be an adjacency matrix containing an $(\alpha_1, 13\alpha_2, p)$-regular subgraph. With probability at least $1-n^{-2}$, Algorithm 18 outputs p_{S^f} such that for some* *universal constants $c_2, c_3 > 0$,*

$$\left|p_{S^f} - p\right| \leq c_2\alpha_1\eta(p,n) + c_3\kappa(13\alpha_1, p, n).$$

The running time of this algorithm is $\tilde{O}(\alpha_1 n^3)$.

Proof. The estimation guarantees in Theorem 158 follows by combining the guarantees of Theorems 141, and 143. We conclude the proof by analyzing the running time. As discussed in Remark 4, Algorithm 16 can be implemented in $\tilde{\mathcal{O}}(\alpha_1 n^3)$ time. Algorithm 17 takes $\mathcal{O}(n^2)$ time. Hence, Algorithm 18 runs in $\tilde{\mathcal{O}}(\alpha_1 n^3)$ time. ■

Algorithm 18. Algorithm for estimating p

```
Input:number of nodes $n$, parameter $\alpha_1 \in [1/n, 1/60]$, adjacency matrix $A$
$S^* \leftarrow$ run the spectral algorithm (Algorithm 16) with inputs $n$, $\alpha_1$, $A$
$p_{S^f} \leftarrow$ run the trimming algorithm (Algorithm 17) with inputs $n$, $\alpha_1$, $A$, $S^*$
return $p_{S^f}$
```

Observe that the $\kappa(13\alpha_1, p, n)$ error term in Theorem 158 scales proportional to $\sqrt{p}$, which gives improved error when p is close to 0. To enjoy the same improvement for p close to 1, we can run the algorithm on the complement of the graph. Theorem 159 describes the resulting guarantees, and the procedure appears as Algorithm 19. Note that we apply Theorem 138 to convert from adjacency matrices containing regular subgraphs (which we have considered up to this point) back to our original problem.

Theorem 159. *Suppose $\beta \in [\frac{1}{n}, \frac{1}{60}]$ and $p \in [0,1]$. Let $G \sim G(n,p)$, and A be the adjacency matrix of a rewiring of G by a β-omniscient adversary. With probability at least $1 - 10n^{-2}$, running Algorithm 19 will output a $\hat{p}$ such that*

$$|\hat{p} - p| \leq C \cdot \left(\frac{\sqrt{p(1-p)\log n}}{n} + \frac{\beta\sqrt{p(1-p)\log(1/\beta)}}{\sqrt{n}} + \frac{\beta}{n}\log n \right),$$

for some universal constant C. The running time of this algorithm is $\tilde{O}(\beta n^3)$.

Proof. Theorem 158 and Theorem 138 imply that with probability $\geq 1-5n^{-2}$, p^* in Algorithm 19 satisfies:

$$\left|p^* - p\right| \leq c_2\beta\eta(p,n) + c_3\kappa(13\beta, p, n). \tag{7.22}$$

By symmetry, with probability $\geq 1 - 5n^{-2}$, q^* in Algorithm 19 satisfies:

$$\left|q^* - (1-p)\right| \leq c_2\beta\eta(1-p,n) + c_3\kappa(13\beta, 1-p, n). \tag{7.23}$$

When $p \leq 0.1$, equation (7.22) implies $p^* \leq 0.5$, and hence $\hat{p} = p^*$ and $|\hat{p}-p| = |p^*-p|$. Similarly, when $p \geq 0.9$, (7.22) implies $p^* > 0.5$, and hence $\hat{p} = 1 - q^*$ and $|\hat{p} - p| = |(1-q^*) - p| = |(1-p) - q^*|$. Finally, for $0.1 \leq p \leq 0.9$, we have $|\hat{p} - p| \leq \max\{|p^* - p|, |q^* - (1-p)|\}$. Combining the bound for the three cases completes the proof. ■

Algorithm 19. Algorithm for Robust Erdős-Rényi parameter estimation

```
Input:number of nodes $n$, parameter $\beta \in [1/n, 1/60]$, adjacency matrix $A$
$p^* \leftarrow$ run Algorithm 18 with inputs $n$, $\beta$, $A$
$q^* \leftarrow$ run Algorithm 18 with inputs $n$, $\beta$, $(1 - I - A)$ ($1$ and $I$ are the $n \times n$ all-ones and identity matrix)
if $p^* \leq 0.5$ then
    $\hat{p} \leftarrow p^*$
else
    $\hat{p} \leftarrow 1 - q^*$
end if
return $\hat{p}$
```

7.11 Proof of Theorem 152

Throughout this proof, let $\gamma = \max\left\{16\alpha n\sqrt{pn\ln\frac{e}{\alpha}}, 60\alpha n\ln\frac{e}{\alpha}, 5n\sqrt{p\ln(en)}\right\}$. First fix $\alpha \in [0, 1/2]$.

We first consider the entire matrix $\tilde{A}$, namely $S = S' = [n]$. Recall that the diagonal entries of $\tilde{A}$ are zero. Then, note that $\sum_{(i,j)\in[n]\times[n]}(\tilde{A}_{i,j} - p) = 2\cdot\sum_{(i,j)\in[n]\times[n]:\ i>j}(\tilde{A}_{i,j} - p) - np$. Now since all the entries $\tilde{A}_{ij}$ are independent for $i > j$, we can apply the Chernoff bound (Equation (7.8)) with $\lambda = \gamma$ over these entries and with probability at least $1 - n^{-3}$,

$$\left|\sum_{(i,j)\in[n]\times[n]:\ i>j}(\tilde{A}_{i,j} - p)\right| \leq \gamma. \tag{7.24}$$

Since $np \leq n\sqrt{p} \leq \gamma$, then from the above equation we get $|\sum_{(i,j)\in[n]\times[n]}(\tilde{A}_{i,j} - p)| \leq 3\gamma$, with probability at least $1 - n^{-3}$. Note that for $\alpha < 1/n$ the statement only applies to $S = S' = [n]$, and thus this case is handled. In the remaining proof $\alpha \in [1/n, 1/2]$.

Conditioned on the event $|\sum_{(i,j)\in[n]\times[n]}(\tilde{A}_{i,j} - p)| \leq 3\gamma$, note that for all $T \subset [n] \times [n]$,

$$\left|\sum_{(i,j)\in T}(\tilde{A}_{i,j} - p)\right| > 6\gamma \Rightarrow \left|\sum_{(i,j)\in T^c}(\tilde{A}_{i,j} - p)\right| > 3\gamma, \tag{7.25}$$

where $T^c = [n] \times [n] \setminus T$. In particular, if $T = S \times S'$ with $|S| \geq n - \alpha n$ and $|S'| \geq n - \alpha n$, then $|T^c| < 2\alpha n^2$ and if $\min\{|S|, |S'|\} \leq \alpha n$, then $|T| \leq \alpha n^2$. Therefore, for $T = S \times S'$ with $|S|$, $|S'| \in C_\alpha$, either $|T|$ or $|T^c|$ is smaller than $2\alpha n^2$. With this in hand, the theorem will follow from the following lemmas.

Lemma 160. *Let $T \subset [n] \times [n]$ be a given subset of size at most $2\alpha n^2$, then*

$$\Pr\left[\left|\sum_{(i,j)\in T}(\tilde{A}_{i,j} - p)\right| \geq 3\gamma\right] \leq 4\exp\left(-20\alpha n\ln e/\alpha\right).$$

We now bound the number of subsets of interest.

Lemma 161. *For a given $\alpha \in [1/n, 1/2]$, the number of sets S, S' with $|S|, |S'| \in C_\alpha$ is at most* $4\exp(4\alpha n \ln(e/\alpha))$.

For a given $\alpha \in [1/n, 1/2]$ and $T = S \times S'$ such that $|S|, |S'| \in C_\alpha$, since either of T or T^c have size $\leq 2\alpha n^2$, therefore, combining the two lemmas implies that with probability $\geq 1 - 16\exp\left(-16\alpha n \ln e/\alpha\right) \geq 1 - n^3$,

$$\min\left\{\left|\sum_{(i,j)\in T}(\tilde{A}_{i,j} - p)\right|, \left|\sum_{(i,j)\in T^c}(\tilde{A}_{i,j} - p)\right|\right\} \leq 3\gamma.$$

Then from Equation (7.25), with probability $\geq 1 - n^3 - n^3$, $\left|\sum_{(i,j)\in T}(\tilde{A}_{i,j} - p)\right| \leq 6\gamma$. This completes the proof for a given value of α. To extend it to all $\alpha \in [1/n, 1/2]$ first note that it suffices to prove the theorem for $\alpha \in \{\frac{1}{n}, \frac{2}{n}, ..., \frac{\lfloor 0.5n \rfloor}{n}\}$, and then upon taking the union bound over these values of α completes the proof.

We now prove Lemma 160. Note that

$$\sum_{(i,j)\in T}(\tilde{A}_{i,j} - p) = \sum_{(i,j)\in T: i>j}(\tilde{A}_{i,j} - p) + \sum_{(i,j)\in T: i<j}(\tilde{A}_{i,j} - p) - \sum_{(i,i)\in T} p. \tag{7.26}$$

Then using the triangle inequality, $\{(i,i) \in T\} \leq n$ and $np \leq \gamma$ to disregard the third term (as done before),

$$\begin{aligned}
&\Pr\left[\left|\sum_{(i,j)\in T}(\tilde{A}_{i,j} - p)\right| \geq 2\gamma\right] \\
&\leq \Pr\left[\left|\sum_{(i,j)\in T: i>j}(\tilde{A}_{i,j} - p)\right| \geq \gamma\right] + \Pr\left[\left|\sum_{(i,j)\in T: i<j}(\tilde{A}_{i,j} - p)\right| \geq \gamma\right].
\end{aligned}$$

The two events on the right hand side are for sums of independent mean-centered Bernoulli random variables. We will now apply the Chernoff bound (Equation (7.8)). Note that for a fixed λ the right hand side of (7.8) is a non-decreasing function of t. Further note that

$|\{(i,j) \in T : i > j\}|, |\{(i,j) \in T : i < j\}| \le |T| < 2\alpha n^2$. Therefore,

$$\Pr\left[\left|\sum_{(i,j)\in T:i>j} (\tilde{A}_{i,j} - p)\right| \ge \gamma\right] \le 2\exp\left(-\min\left(\frac{\gamma^2}{6\alpha n^2 p}, \frac{\gamma}{3}\right)\right) \le 2\exp\left(-20\alpha n \ln \frac{e}{\alpha}\right).$$

Similarly,

$$\Pr\left[\left|\sum_{(i,j)\in T:i<j} (\tilde{A}_{i,j} - p)\right| \ge \gamma\right] \le 2\exp\left(-20\alpha n \ln \frac{e}{\alpha}\right).$$

Combining the two bounds completes the proof of Lemma 160.

We finally prove Lemma 161. The number of such sets can be upper bounded by $4 \cdot \left(\sum_{j=0}^{\lfloor \alpha n \rfloor} \binom{n}{j}\right)^2$, where

$$\begin{aligned}\sum_{j=0}^{\lfloor \alpha n \rfloor} \binom{n}{j} &\le (\alpha n + 1) \cdot \binom{n}{\lfloor \alpha n \rfloor} \\ &\le (\alpha n + 1) \cdot \left(\frac{e}{\alpha}\right)^{\alpha n} \\ &\le e^{\alpha n \ln\left(\frac{e}{\alpha}\right) + \ln(\alpha n + 1)} \le e^{\alpha n \ln\left(\frac{e}{\alpha}\right) + \alpha n} \le e^{2\alpha n \ln(e/\alpha)}.\end{aligned}$$

7.12 Lower bound proofs

Proof of Lemma 144. We prove this lemma by converting a β-corrupted graph from $DG(n,p)$ to a β-corrupted graph from $G(n,p)$. Then one can run the algorithm for the undirected setting to obtain an estimate of p, which implies the same error guarantees for the directed instance.

Suppose there exists a random directed graph $DG \sim DG(n,p)$ which is β-corrupted by an adversary. Assume there exists some lexicographic ordering of the nodes (e.g., they are numbered from 1 to n). We define a corresponding undirected graph G as follows: let there be an edge between nodes i and j in G if there exists an edge from i to j in DG and $i < j$. Sans corruptions, this converts $DG(n,p)$ into $G(n,p)$ since the edges are still independent and the probability of each edge existing is p. Furthermore, when at most βn nodes in the original directed graph are modified, at most βn nodes are changed in the corresponding undirected graph. ■

Proof of Theorem 145. Our β-oblivious adversary for the directed graph model works as follows. The adversary picks a set B of size $Bin(n, 0.15\beta)$ to corrupt, by independently picking each node in $[n]$ with probability 0.15β. Note that it is possible that the size of the set of corrupted nodes B may exceed βn with small probability. We will address this issue later.

The adversary will corrupt outgoing edges of the nodes in B. The adversary's strategy to corrupt the neighborhood of node $i \in B$ is as follows. They first choose node i's new out-degree $\deg(i)$ independently from some distribution P over $\{0, \ldots, n-1\}$. Then, they select an independent random subset S_i of nodes $[n] \setminus \{i\}$ of size $\deg(i)$. Finally, they introduce the directed edge (i, j) for each $j \in S_i$, and remove the directed edge (i, j) for each $j \notin S_i$. The distribution of the degree of corrupted nodes, P, depends on the parameter p of the Erdős-Rényi graph and will be specified later.

By this construction, all graphs with a given outgoing degree distribution $d_1, d_2, \ldots, d_n$ have the same probability, and they form a sufficient statistic for estimating p. The out-degree of any uncorrupted node is distributed as $Bin(n-1, p)$ and the out-degree of any corrupted node has distribution P. Since each node is corrupted with probability 0.15β, the out-degree of each node is an i.i.d. sample from the mixture distribution $(1-0.15\beta) \cdot Bin(n-1, p) + 0.15\beta \cdot P$.

Next, we show that for any $p_1 \leq 1/2$ and $p_2 = p_1 + 0.1 \max\left(\beta\sqrt{p/n}, 0.1\beta/n\right)$ there exist distributions P_1 and P_2 such that

$$(1-0.15\beta) \cdot Bin(n-1, p_1) + 0.15\beta \cdot P_1 = (1-0.15\beta) \cdot Bin(n-1, p_2) + 0.15\beta \cdot P_2. \tag{7.27}$$

This will imply that, with the aforementioned adversary, any estimator that distinguishes between the two cases will be correct with probability at most $1/2$. At this point, we account for the probability that the adversary selects a set B of size $> \beta n$, which is not allowed according to the corruption model. By Markov's inequality, this occurs with probability at most 0.15. Therefore, even counting such violations as a success at distinguishing the two cases, it still succeeds with probability at most $0.5 + 0.15 = 0.65$.

To prove the existence of P_1 and P_2 satisfying (7.27) we use the following folklore fact: given any two distributions D_1 and D_2 and $\epsilon > 0$, if $d_{\mathrm{TV}}(D_1, D_2) \le \epsilon$, then there exist distributions Q_1 and Q_2 such that $(1-\epsilon)D_1 + \epsilon Q_1 = (1-\epsilon)D_2 + \epsilon Q_2$.

Hence, it suffices to show that

$$d_{\mathrm{TV}}\left(Bin(n-1, p_1), Bin(n-1, p_2)\right) \le 0.15\beta. \tag{7.28}$$

The total variation distance between two binomials can be bounded as [128], [3, Eq (2.16)].

$$d_{\mathrm{TV}}\left(Bin(n', p), Bin(n', p+x)\right) \le \sqrt{\frac{e}{2}\frac{\tau(x)}{(1-\tau(x))^2}}, \tag{7.29}$$

where $\tau(x) = x\sqrt{\frac{n'+2}{2p(1-p)}}$. We also use the trivial upper bound $d_{\mathrm{TV}}\left(Bin(n', p), Bin(n', p+x)\right) \le n'x$.

For the case when $\beta\sqrt{p/n} \ge 0.1\beta/n$, applying the first bound for $x = 0.1\beta\sqrt{p/n}$ and $n' = n-1$ we get

$$\tau(x) = 0.1\beta\sqrt{\frac{p}{n}}\sqrt{\frac{n+1}{2p(1-p)}} \le 0.1 \cdot 1.1\beta = 0.11\beta.$$

For this case, using (7.29) gives

$$d_{\mathrm{TV}}\left((Bin(n-1, p_1), Bin(n-1, p_2)\right) \le 0.15\beta.$$

For the other case when $\beta\sqrt{p/n} < 0.1\beta/n$, applying the trivial bound gives

$$d_{\mathrm{TV}}\left((Bin(n-1, p_1), Bin(n-1, p_2)\right) \le 0.1\beta(n-1)/n < 0.1\beta.$$

This proves (7.28) and shows the existence of P_1 and P_2, which completes the proof of the first two terms in Δ_p.

Finally, we show that the third term in the max in Δ_p holds even when there is no corruption. To show this we first note that in absence of corruption the sufficient statistics for estimating p is the total number of edges in the directed graph, which has a distribution $Bin((n-1)^2, p)$. Then to show that for $p \leq 0.5$ no algorithm can distinguish between between $DG(n, p)$ and $DG(n, p + 0.1\sqrt{p}/n)$ with probability ≥ 0.6 it suffices to show that $d_{\text{TV}}(Bin((n-1)^2, p), Bin((n-1)^2, p + 0.1\sqrt{p}/n)) < 0.2$, which can be verified using (7.29) for $x = 0.1\sqrt{p}/n$, $n' = (n-1)^2$, and any $p < 1/2$. ■

Chapter 7, in full is a reprint of the material as it appears in Robust estimation for random graphs 2022. Jayadev Acharya, Ayush Jain, Gautam Kamath, Ananda Theertha Suresh, and Huanyu Zhang. In COLT 2022. The dissertation author was the primary investigator and author of this paper.

Bibliography

[1] Jayadev Acharya, Ilias Diakonikolas, Jerry Li, and Ludwig Schmidt. Sample-optimal density estimation in nearly-linear time. In *Proceedings of the Twenty-Eighth Annual ACM-SIAM Symposium on Discrete Algorithms*, pages 1278–1289. SIAM, 2017.

[2] Jayadev Acharya, Ayush Jain, Gautam Kamath, Ananda Theertha Suresh, and Huanyu Zhang. Robust estimation for random graphs. In *Conference on Learning Theory*, pages 130–166. PMLR, 2022.

[3] José A Adell and Pedro Jodrá. Exact kolmogorov and total variation distances between some familiar discrete distributions. *Journal of Inequalities and Applications*, 2006(1):64307, 2006.

[4] Noga Alon, Michael Krivelevich, and Benny Sudakov. Finding a large hidden clique in a random graph. *Random Structures and Algorithms*, 13(3-4):457–466, 1998.

[5] Noga Alon and Assaf Naor. Approximating the cut-norm via grothendieck's inequality. In *Proceedings of the thirty-sixth annual ACM symposium on Theory of computing*, pages 72–80. ACM, 2004.

[6] Frank J Anscombe. Rejection of outliers. *Technometrics*, 2(2):123–146, 1960.

[7] Martin Anthony and John Shawe-Taylor. A result of vapnik with applications. *Discrete Applied Mathematics*, 47(3):207–217, 1993.

[8] Hassan Ashtiani and Abbas Mehrabian. Some techniques in density estimation. *arXiv preprint arXiv:1801.04003*, 2018.

[9] Pranjal Awasthi, Maria Florina Balcan, and Philip M. Long. The power of localization for efficiently learning linear separators with noise. In *Proceedings of the 46th Annual ACM Symposium on the Theory of Computing*, STOC '14, pages 449–458, New York, NY, USA, 2014. ACM.

[10] Pranjal Awasthi, Avrim Blum, Nika Haghtalab, and Yishay Mansour. Efficient pac learning from the crowd. In *Conference on Learning Theory*, pages 127–150. PMLR, 2017.

[11] Sivaraman Balakrishnan, Simon S. Du, Jerry Li, and Aarti Singh. Computationally efficient robust sparse estimation in high dimensions. In *Proceedings of the 30th Annual Conference on Learning Theory*, COLT '17, pages 169–212, 2017.

[12] Afonso S. Bandeira and Ramon Van Handel. Sharp nonasymptotic bounds on the norm of random matrices with independent entries. *The Annals of Probability*, 44(4):2479–2506, 2016.

[13] Jess Banks, Sidhanth Mohanty, and Prasad Raghavendra. Local statistics, semidefinite programming, and community detection. In *Proceedings of the 32nd Annual ACM-SIAM Symposium on Discrete Algorithms*, SODA '21, pages 1298–1316, Philadelphia, PA, USA, 2021. SIAM.

[14] Boaz Barak, Samuel Hopkins, Jonathan Kelner, Pravesh K Kothari, Ankur Moitra, and Aaron Potechin. A nearly tight sum-of-squares lower bound for the planted clique problem. *SIAM Journal on Computing*, 48(2):687–735, 2019.

[15] Marco Barreno, Blaine Nelson, Anthony D Joseph, and J Doug Tygar. The security of machine learning. *Machine Learning*, 81(2):121–148, 2010.

[16] Thorsten Bernholt. Robust estimators are hard to compute. Technical report, Technische Universität Dortmund, 2006.

[17] Kush Bhatia, Prateek Jain, Parameswaran Kamalaruban, and Purushottam Kar. Consistent robust regression. *Advances in Neural Information Processing Systems*, 30, 2017.

[18] Kush Bhatia, Prateek Jain, and Purushottam Kar. Robust regression via hard thresholding. *Advances in neural information processing systems*, 28, 2015.

[19] Battista Biggio, Blaine Nelson, and Pavel Laskov. Poisoning attacks against support vector machines. *arXiv preprint arXiv:1206.6389*, 2012.

[20] Christian Borgs, Jennifer Chayes, and Adam Smith. Private graphon estimation for sparse graphs. In *Advances in Neural Information Processing Systems 28*, NIPS '15, pages 1369–1377. Curran Associates, Inc., 2015.

[21] Christian Borgs, Jennifer Chayes, Adam Smith, and Ilias Zadik. Revealing network structure, confidentially: Improved rates for node-private graphon estimation. In *Proceedings of the 59th Annual IEEE Symposium on Foundations of Computer Science*, FOCS '18, pages 533–543, Washington, DC, USA, 2018. IEEE Computer Society.

[22] Dietrich Braess and Thomas Sauer. Bernstein polynomials and learning theory. *Journal of Approximation Theory*, 128(2):187–206, 2004.

[23] T. Tony Cai and Xiaodong Li. Robust and computationally feasible community detection in the presence of arbitrary outlier nodes. *The Annals of Statistics*, 43(3):1027–1059, 2015.

[24] Emmanuel J Candès, Xiaodong Li, Yi Ma, and John Wright. Robust principal component analysis? *Journal of the ACM*, 58(3):1–37, 2011.

[25] Center for Disease Control. CDC influenza vaccine 2020/2021. https://www.cdc.gov/flu/season/faq-flu-season-2020-2021.htm, 2020.

[26] Arun Tejasvi Chaganty and Percy Liang. Spectral experts for estimating mixtures of linear regressions. In *International Conference on Machine Learning (ICML)*, pages 1040–1048, 2013.

[27] Siu-On Chan, Ilias Diakonikolas, Rocco A Servedio, and Xiaorui Sun. Efficient density estimation via piecewise polynomial approximation. In *Proceedings of the forty-sixth annual ACM symposium on Theory of computing*, pages 604–613, 2014.

[28] Venkat Chandrasekaran, Sujay Sanghavi, Pablo A Parrilo, and Alan S Willsky. Rank-sparsity incoherence for matrix decomposition. *SIAM Journal on Optimization*, 21(2):572–596, 2011.

[29] Moses Charikar, Jacob Steinhardt, and Gregory Valiant. Learning from untrusted data. In *Proceedings of the 49th Annual ACM SIGACT Symposium on Theory of Computing*, pages 47–60. ACM, 2017.

[30] Sitan Chen, Jerry Li, and Ankur Moitra. Efficiently learning structured distributions from untrusted batches. In *Proceedings of the 52nd Annual ACM Symposium on the Theory of Computing*, STOC '20, pages 960–973, New York, NY, USA, 2020. ACM.

[31] Sitan Chen, Jerry Li, and Ankur Moitra. Learning structured distributions from untrusted batches: Faster and simpler. *arXiv preprint arXiv:2002.10435*, 2020.

[32] Sitan Chen, Jerry Li, and Zhao Song. Learning mixtures of linear regressions in subexponential time via Fourier moments. In *STOC*. https://arxiv.org/pdf/1912.07629.pdf, 2020.

[33] Yanxi Chen and H. Vincent Poor. Learning mixtures of linear dynamical systems. In Kamalika Chaudhuri, Stefanie Jegelka, Le Song, Csaba Szepesvari, Gang Niu, and Sivan Sabato, editors, *Proceedings of the 39th International Conference on Machine Learning*, volume 162 of *Proceedings of Machine Learning Research*, pages 3507–3557. PMLR, 17–23 Jul 2022.

[34] Yu Cheng, Ilias Diakonikolas, and Rong Ge. High-dimensional robust mean estimation in nearly-linear time. In *Proceedings of the 30th Annual ACM-SIAM Symposium on Discrete*

Algorithms, SODA '19, pages 2755–2771, Philadelphia, PA, USA, 2019. SIAM.

[35] Yu Cheng, Ilias Diakonikolas, Rong Ge, and Mahdi Soltanolkotabi. High-dimensional robust mean estimation via gradient descent. In *Proceedings of the 37th International Conference on Machine Learning*, ICML '20, pages 1768–1778. JMLR, Inc., 2020.

[36] Yu Cheng, Ilias Diakonikolas, Rong Ge, and David P Woodruff. Faster algorithms for high-dimensional robust covariance estimation. In *Conference on Learning Theory*, pages 727–757. PMLR, 2019.

[37] Yu Cheng, Ilias Diakonikolas, Daniel M. Kane, and Alistair Stewart. Robust learning of fixed-structure Bayesian networks. In *Advances in Neural Information Processing Systems 31*, NeurIPS '18, pages 10304–10316. Curran Associates, Inc., 2018.

[38] Yeshwanth Cherapanamjeri, Efe Aras, Nilesh Tripuraneni, Michael I Jordan, Nicolas Flammarion, and Peter L Bartlett. Optimal robust linear regression in nearly linear time. *arXiv preprint arXiv:2007.08137*, 2020.

[39] Yeshwanth Cherapanamjeri, Sidhanth Mohanty, and Morris Yau. List decodable mean estimation in nearly linear time. *arXiv preprint arXiv:2005.09796*, 2020.

[40] Arnak Dalalyan and Philip Thompson. Outlier-robust estimation of a sparse linear model using ℓ_1 -penalized huber's m-estimator. *Advances in neural information processing systems*, 32, 2019.

[41] Abhimanyu Das, Ayush Jain, Weihao Kong, and Rajat Sen. Efficient list-decodable regression using batches. *arXiv preprint arXiv:2211.12743*, 2022.

[42] Yash Deshpande and Andrea Montanari. Improved sum-of-squares lower bounds for hidden clique and hidden submatrix problems. In *Proceedings of the 28th Annual Conference on Learning Theory*, COLT '15, pages 523–562, 2015.

[43] Luc Devroye and Gabor Lugosi. *Combinatorial Methods in Density Estimation*. Springer Science & Business Media, 2001.

[44] Ilias Diakonikolas. Learning structured distributions. *Handbook of Big Data*, 267, 2016.

[45] Ilias Diakonikolas, Samuel B Hopkins, Daniel Kane, and Sushrut Karmalkar. Robustly learning any clusterable mixture of gaussians. *arXiv preprint arXiv:2005.06417*, 2020.

[46] Ilias Diakonikolas, Gautam Kamath, Daniel Kane, Jerry Li, Ankur Moitra, and Alistair Stewart. Robust estimators in high-dimensions without the computational intractability. *SIAM Journal on Computing*, 48(2):742–864, 2019.

[47] Ilias Diakonikolas, Gautam Kamath, Daniel M Kane, Jerry Li, Ankur Moitra, and Alistair Stewart. Robust estimators in high dimensions without the computational intractability. In *2016 IEEE 57th Annual Symposium on Foundations of Computer Science (FOCS)*, pages 655–664. IEEE, 2016.

[48] Ilias Diakonikolas, Gautam Kamath, Daniel M Kane, Jerry Li, Ankur Moitra, and Alistair Stewart. Being robust (in high dimensions) can be practical. In *Proceedings of the 34th International Conference on Machine Learning-Volume 70*, pages 999–1008. JMLR. org, 2017.

[49] Ilias Diakonikolas, Gautam Kamath, Daniel M Kane, Jerry Li, Ankur Moitra, and Alistair Stewart. Robustly learning a gaussian: Getting optimal error, efficiently. In *Proceedings of the Twenty-Ninth Annual ACM-SIAM Symposium on Discrete Algorithms*, pages 2683–2702. SIAM, 2018.

[50] Ilias Diakonikolas, Gautam Kamath, Daniel M. Kane, Jerry Li, Jacob Steinhardt, and Alistair Stewart. Sever: A robust meta-algorithm for stochastic optimization. In *Proceedings of the 36th International Conference on Machine Learning*, ICML '19, pages 1596–1606. JMLR, Inc., 2019.

[51] Ilias Diakonikolas, Daniel Kane, and Daniel Kongsgaard. List-decodable mean estimation via iterative multi-filtering. *Advances in Neural Information Processing Systems*, 33:9312–9323, 2020.

[52] Ilias Diakonikolas, Daniel Kane, Daniel Kongsgaard, Jerry Li, and Kevin Tian. List-decodable mean estimation in nearly-pca time. *Advances in Neural Information Processing Systems*, 34:10195–10208, 2021.

[53] Ilias Diakonikolas, Daniel Kane, Ankit Pensia, Thanasis Pittas, and Alistair Stewart. Statistical query lower bounds for list-decodable linear regression. *Advances in Neural Information Processing Systems*, 34:3191–3204, 2021.

[54] Ilias Diakonikolas and Daniel M. Kane. Recent advances in algorithmic high-dimensional robust statistics. *arXiv preprint arXiv:1911.05911*, 2019.

[55] Ilias Diakonikolas and Daniel M Kane. Small covers for near-zero sets of polynomials and learning latent variable models. In *2020 IEEE 61st Annual Symposium on Foundations of Computer Science (FOCS)*, pages 184–195. IEEE, 2020.

[56] Ilias Diakonikolas, Daniel M Kane, and Alistair Stewart. List-decodable robust mean estimation and learning mixtures of spherical gaussians. In *Proceedings of the 50th Annual ACM SIGACT Symposium on Theory of Computing*, pages 1047–1060. ACM, 2018.

[57] Ilias Diakonikolas, Weihao Kong, and Alistair Stewart. Efficient algorithms and lower

bounds for robust linear regression. In *Proceedings of the Thirtieth Annual ACM-SIAM Symposium on Discrete Algorithms*, pages 2745–2754. SIAM, 2019.

[58] Terry E Dielman. *Applied regression analysis for business and economics*. Duxbury/Thomson Learning Pacific Grove, CA, 2001.

[59] Yihe Dong, Samuel Hopkins, and Jerry Li. Quantum entropy scoring for fast robust mean estimation and improved outlier detection. *Advances in Neural Information Processing Systems*, 32, 2019.

[60] Paul Erdős and Alfréd Rényi. On random graphs i. *Publicationes Mathematicate*, 6:290–297, 1959.

[61] Uriel Feige and Robert Krauthgamer. The probable value of the lovász–schrijver relaxations for maximum independent set. *SIAM Journal on Computing*, 32(2):345–370, 2003.

[62] Vitaly Feldman, Elena Grigorescu, Lev Reyzin, Santosh S Vempala, and Ying Xiao. Statistical algorithms and a lower bound for detecting planted cliques. *Journal of the ACM*, 64(2):1–37, 2017.

[63] Chelsea Finn, Pieter Abbeel, and Sergey Levine. Model-agnostic meta-learning for fast adaptation of deep networks. In *Proceedings of the 34th International Conference on Machine Learning (ICML)*, pages 1126–1135, 2017.

[64] Chao Gao. Robust regression via mutivariate regression depth. *Bernoulli*, 26(2):1139–1170, 2020.

[65] Edgar N. Gilbert. Random graphs. *The Annals of Mathematical Statistics*, 30(4):1141–1144, 1959.

[66] Michael Grottke, Julian Knoll, and Rainer Groß. How the distribution of the number of items rated per user influences the quality of recommendations. In *2015 15th International Conference on Innovations for Community Services (I4CS)*, pages 1–8. IEEE, 2015.

[67] Yi Hao, Ayush Jain, Alon Orlitsky, and Vaishakh Ravindrakumar. Surf: A simple, universal, robust, fast distribution learning algorithm. *arXiv preprint arXiv:2002.09589*, 2020.

[68] Sam Hopkins, Jerry Li, and Fred Zhang. Robust and heavy-tailed mean estimation made simple, via regret minimization. *Advances in Neural Information Processing Systems*, 33, 2020.

[69] Samuel B Hopkins et al. Mean estimation with sub-gaussian rates in polynomial time. *Annals of Statistics*, 48(2):1193–1213, 2020.

[70] Samuel B Hopkins, Pravesh Kothari, Aaron Henry Potechin, Prasad Raghavendra, and Tselil Schramm. On the integrality gap of degree-4 sum of squares for planted clique. *ACM Transactions on Algorithms*, 14(3):1–31, 2018.

[71] Samuel B Hopkins and Jerry Li. Mixture models, robustness, and sum of squares proofs. In *Proceedings of the 50th Annual ACM SIGACT Symposium on Theory of Computing*, pages 1021–1034, 2018.

[72] Daniel Hsu, Sham M Kakade, and Tong Zhang. Robust matrix decomposition with sparse corruptions. *IEEE Transactions on Information Theory*, 57(11):7221–7234, 2011.

[73] Peter J Huber. Robust estimation of a location parameter. *Annals Mathematics Statistics, 35*, 1964.

[74] Peter J Huber. Robust estimation of a location parameter. In *Breakthroughs in statistics*, pages 492–518. Springer, 1992.

[75] Peter J Huber. Robust statistics. In *International encyclopedia of statistical science*, pages 1248–1251. Springer, 2011.

[76] Ayush Jain and Alon Orlitsky. A general method for robust learning from batches. In *Advances in Neural Information Processing Systems 33*, NeurIPS '20, pages 21775–21785. Curran Associates, Inc., 2020.

[77] Ayush Jain and Alon Orlitsky. Optimal robust learning of discrete distributions from batches. In *Proceedings of the 37th International Conference on Machine Learning*, ICML '20, pages 4651–4660. JMLR, Inc., 2020.

[78] Ayush Jain and Alon Orlitsky. Robust density estimation from batches: The best things in life are (nearly) free. In *International Conference on Machine Learning*, pages 4698–4708. PMLR, 2021.

[79] Ayush Jain, Alon Orlitsky, and Vaishakh Ravindrakumar. Robust estimation algorithms don't need to know the corruption level. *arXiv preprint arXiv:2202.05453*, 2022.

[80] Arun Jambulapati, Jerry Li, Tselil Schramm, and Kevin Tian. Robust regression revisited: Acceleration and improved estimation rates. *Advances in Neural Information Processing Systems*, 34:4475–4488, 2021.

[81] Arun Jambulapati, Jerry Li, and Kevin Tian. Robust sub-gaussian principal component analysis and width-independent schatten packing. *Advances in Neural Information Processing Systems*, 33, 2020.

[82] Mark Jerrum. Large cliques elude the Metropolis process. *Random Structures and*

Algorithms, 3(4):347–359, 1992.

[83] He Jia and Santosh Vempala. Robustly clustering a mixture of gaussians. *arXiv preprint arXiv:1911.11838*, 2019.

[84] Michael I Jordan and Robert A Jacobs. Hierarchical mixtures of experts and the em algorithm. *Neural computation*, 6(2):181–214, 1994.

[85] Rob Kaas and Jan M Buhrman. Mean, median and mode in binomial distributions. *Statistica Neerlandica*, 34(1):13–18, 1980.

[86] Sudeep Kamath, Alon Orlitsky, Dheeraj Pichapati, and Ananda Theertha Suresh. On learning distributions from their samples. In *Conference on Learning Theory*, pages 1066–1100, 2015.

[87] Sushrut Karmalkar, Adam Klivans, and Pravesh Kothari. List-decodable linear regression. *Advances in neural information processing systems*, 32, 2019.

[88] Sushrut Karmalkar and Eric Price. Compressed sensing with adversarial sparse noise via l1 regression. In *2nd Symposium on Simplicity in Algorithms*, 2019.

[89] Richard Karp. Probabilistic analysis of some combinatorial search problems. traub, jf (ed.): Algorithms and complexity: New directions and recent results, 1976.

[90] Adam Klivans, Pravesh K Kothari, and Raghu Meka. Efficient algorithms for outlier-robust regression. In *Conference On Learning Theory*, pages 1420–1430. PMLR, 2018.

[91] Adam R Klivans, Philip M Long, and Rocco A Servedio. Learning halfspaces with malicious noise. *Journal of Machine Learning Research*, 10(12):2715–2740, 2009.

[92] Gregory Koch, Richard Zemel, and Ruslan Salakhutdinov. Siamese neural networks for one-shot image recognition. In *ICML deep learning workshop*, volume 2, 2015.

[93] Weihao Kong, Rajat Sen, Pranjal Awasthi, and Abhimanyu Das. Trimmed maximum likelihood estimation for robust learning in generalized linear models. *arXiv preprint arXiv:2206.04777*, 2022.

[94] Weihao Kong, Raghav Somani, Sham Kakade, and Sewoong Oh. Robust meta-learning for mixed linear regression with small batches. *Advances in Neural Information Processing Systems*, 33, 2020.

[95] Weihao Kong, Raghav Somani, Zhao Song, Sham Kakade, and Sewoong Oh. Meta-learning for mixed linear regression. In *International Conference on Machine Learning*, pages 5394–5404. PMLR, 2020.

[96] Nikola Konstantinov, Elias Frantar, Dan Alistarh, and Christoph Lampert. On the sample complexity of adversarial multi-source pac learning. In *International Conference on Machine Learning*, pages 5416–5425. PMLR, 2020.

[97] Pravesh K Kothari, Jacob Steinhardt, and David Steurer. Robust moment estimation and improved clustering via sum of squares. In *Proceedings of the 50th Annual ACM SIGACT Symposium on Theory of Computing*, pages 1035–1046, 2018.

[98] Luděk Kučera. Expected complexity of graph partitioning problems. *Discrete Applied Mathematics*, 57(2-3):193–212, 1995.

[99] Kevin A Lai, Anup B Rao, and Santosh Vempala. Agnostic estimation of mean and covariance. In *2016 IEEE 57th Annual Symposium on Foundations of Computer Science (FOCS)*, pages 665–674. IEEE, 2016.

[100] Guillaume Lecué and Shahar Mendelson. Performance of empirical risk minimization in linear aggregation. 2016.

[101] Jerry Li and Guanghao Ye. Robust gaussian covariance estimation in nearly-matrix multiplication time. *Advances in Neural Information Processing Systems*, 33, 2020.

[102] Yuanzhi Li and Yingyu Liang. Learning mixtures of linear regressions with nearly optimal complexity. In *COLT*. arXiv preprint arXiv:1802.07895, 2018.

[103] Erik M. Lindgren, Vatsal Shah, Yanyao Shen, Alexandros G. Dimakis, and Adam Klivans. On robust learning of Ising models. In *NeurIPS Workshop on Relational Representation Learning*, 2018.

[104] Liu Liu, Yanyao Shen, Tianyang Li, and Constantine Caramanis. High dimensional robust sparse regression. In *Proceedings of the 23rd International Conference on Artificial Intelligence and Statistics*, AISTATS '20, pages 411–421. JMLR, Inc., 2020.

[105] Wolfgang Maass. Efficient agnostic pac-learning with simple hypothesis. In *Proceedings of the seventh annual conference on Computational learning theory*, pages 67–75, 1994.

[106] Konstantin Makarychev, Yury Makarychev, and Aravindan Vijayaraghavan. Learning communities in the presence of errors. In *Proceedings of the 29th Annual Conference on Learning Theory*, COLT '16, pages 1258–1291, 2016.

[107] John H McDonald. *Handbook of biological statistics*, volume 2. sparky house publishing Baltimore, MD, 2009.

[108] H Brendan McMahan, Eider Moore, Daniel Ramage, Seth Hampson, et al. Communication-efficient learning of deep networks from decentralized data. *arXiv preprint*

arXiv:1602.05629, 2016.

[109] H Brendan McMahan and Daniel Ramage. research.google.com/pubs/pub44822.html. 2017.

[110] Raghu Meka, Aaron Potechin, and Avi Wigderson. Sum-of-squares lower bounds for planted clique. In *Proceedings of the 47th Annual ACM Symposium on the Theory of Computing*, STOC '15, pages 87–96, New York, NY, USA, 2015. ACM.

[111] Ankur Moitra, William Perry, and Alexander S Wein. How robust are reconstruction thresholds for community detection? In *Proceedings of the 48th Annual ACM Symposium on the Theory of Computing*, STOC '16, pages 828–841, New York, NY, USA, 2016. ACM.

[112] Bhaskar Mukhoty, Govind Gopakumar, Prateek Jain, and Purushottam Kar. Globally-convergent iteratively reweighted least squares for robust regression problems. In *The 22nd International Conference on Artificial Intelligence and Statistics*, pages 313–322, 2019.

[113] Cameron Musco and Christopher Musco. Randomized block Krylov methods for stronger and faster approximate singular value decomposition. In *Advances in Neural Information Processing Systems 28*, NeurIPS '15, pages 1396–1404. Curran Associates, Inc., 2015.

[114] Mark E. J. Newman, Duncan J. Watts, and Steven H. Strogatz. Random graph models of social networks. *Proceedings of the National Academy of Sciences*, 99(suppl 1):2566–2572, 2002.

[115] Carl M O'Brien. Nonparametric estimation under shape constraints: Estimators, algorithms and asymptotics. *International Statistical Review*, 84(2):318–319, 2016.

[116] Boris Oreshkin, Pau Rodríguez López, and Alexandre Lacoste. Tadam: Task dependent adaptive metric for improved few-shot learning. In *Advances in Neural Information Processing Systems*, pages 721–731, 2018.

[117] Soumyabrata Pal, Arya Mazumdar, Rajat Sen, and Avishek Ghosh. On learning mixture of linear regressions in the non-realizable setting. In *International Conference on Machine Learning*, pages 17202–17220. PMLR, 2022.

[118] Yoon-Joo Park and Alexander Tuzhilin. The long tail of recommender systems and how to leverage it. In *Proceedings of the 2008 ACM conference on Recommender systems*, pages 11–18, 2008.

[119] Peristera Paschou, Jamey Lewis, Asif Javed, and Petros Drineas. Ancestry informative markers for fine-scale individual assignment to worldwide populations. *Journal of Medical*

Genetics, 47(12):835–847, 2010.

[120] Ankit Pensia, Varun Jog, and Po-Ling Loh. Robust regression with covariate filtering: Heavy tails and adversarial contamination. *arXiv preprint arXiv:2009.12976*, 2021.

[121] Rigollet Philippe. 18.s997 high-dimensional statistics. *Massachusetts Institute of Technology: MIT OpenCourseWare, https://ocw.mit.edu. License: Creative Commons BY-NC-SA*, 2015.

[122] Adarsh Prasad, Vishwak Srinivasan, Sivaraman Balakrishnan, and Pradeep Ravikumar. On learning Ising models under Huber's contamination model. In *Advances in Neural Information Processing Systems 33*, NeurIPS '20. Curran Associates, Inc., 2020.

[123] Adarsh Prasad, Arun Sai Suggala, Sivaraman Balakrishnan, and Pradeep Ravikumar. Robust estimation via robust gradient estimation. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 82(3):601–627, 2020.

[124] Mingda Qiao. Do outliers ruin collaboration? In *International Conference on Machine Learning*, pages 4180–4187. PMLR, 2018.

[125] Mingda Qiao and Gregory Valiant. Learning discrete distributions from untrusted batches. In *Proceedings of the 9th Conference on Innovations in Theoretical Computer Science*, ITCS '18, pages 47:1–47:20, Dagstuhl, Germany, 2018. Schloss Dagstuhl–Leibniz-Zentrum fuer Informatik.

[126] Prasad Raghavendra and Morris Yau. List decodable learning via sum of squares. In *Proceedings of the Fourteenth Annual ACM-SIAM Symposium on Discrete Algorithms*, pages 161–180. SIAM, 2020.

[127] Sachin Ravi and Hugo Larochelle. Optimization as a model for few-shot learning. In *International Conference on Representation Learning*, 2017.

[128] Bero Roos. Binomial approximation to the poisson binomial distribution: The krawtchouk expansion. *Theory of Probability & Its Applications*, 45(2):258–272, 2001.

[129] Noah A Rosenberg, Jonathan K Pritchard, James L Weber, Howard M Cann, Kenneth K Kidd, Lev A Zhivotovsky, and Marcus W Feldman. Genetic structure of human populations. *science*, 298(5602):2381–2385, 2002.

[130] Peter J Rousseeuw. Tutorial to robust statistics. *Journal of chemometrics*, 5(1):1–20, 1991.

[131] Andrei A Rusu, Dushyant Rao, Jakub Sygnowski, Oriol Vinyals, Razvan Pascanu, Simon Osindero, and Raia Hadsell. Meta-learning with latent embedding optimization. *arXiv preprint arXiv:1807.05960*, 2018.

[132] Jürgen Schmidhuber. *Evolutionary principles in self-referential learning, or on learning how to learn: the meta-meta-... hook.* PhD thesis, Technische Universität München, 1987.

[133] Adam Sealfon and Jonathan Ullman. Efficiently estimating Erdos-Renyi graphs with node differential privacy. In *Advances in Neural Information Processing Systems 32*, NeurIPS '19, pages 3765–3775. Curran Associates, Inc., 2019.

[134] Hanie Sedghi, Majid Janzamin, and Anima Anandkumar. Provable tensor methods for learning mixtures of generalized linear models. In *Artificial Intelligence and Statistics (AISTATS)*, pages 1223–1231, 2016.

[135] Jacob Steinhardt, Moses Charikar, and Gregory Valiant. Resilience: A criterion for learning in the presence of arbitrary outliers. *arXiv preprint arXiv:1703.04940*, 2017.

[136] Jacob Steinhardt, Gregory Valiant, and Moses Charikar. Avoiding imposters and delinquents: Adversarial crowdsourcing and peer prediction. *Advances in Neural Information Processing Systems*, 29, 2016.

[137] Arun Sai Suggala, Kush Bhatia, Pradeep Ravikumar, and Prateek Jain. Adaptive hard thresholding for near-optimal consistent robust regression. In *Conference on Learning Theory*, pages 2892–2897. PMLR, 2019.

[138] Michel Talagrand. Sharper bounds for gaussian and empirical processes. *The Annals of Probability*, pages 28–76, 1994.

[139] Sebastian Thrun and Lorien Pratt. *Learning to learn*. Springer Science & Business Media, 2012.

[140] Kai Ming Ting, Boon Toh Low, and Ian H Witten. Learning from batched data: Model combination versus data combination. *Knowledge and Information Systems*, 1:83–106, 1999.

[141] Eleni Triantafillou, Tyler Zhu, Vincent Dumoulin, Pascal Lamblin, Kelvin Xu, Ross Goroshin, Carles Gelada, Kevin Swersky, Pierre-Antoine Manzagol, and Hugo Larochelle. Meta-dataset: A dataset of datasets for learning to learn from few examples. *arXiv preprint arXiv:1903.03096*, 2019.

[142] John W Tukey. A survey of sampling from contaminated distributions. *Contributions to probability and statistics*, pages 448–485, 1960.

[143] Aad W Vaart and Jon A Wellner. *Weak convergence and empirical processes: with applications to statistics*. Springer, 1996.

[144] Vladimir Vapnik and Alexey Chervonenkis. Theory of pattern recognition, 1974.

[145] VN Vapnik and A Ya Chervonenkis. On the uniform convergence of relative frequencies of events to their probabilities. *Theory of Probability & Its Applications*, 16(2):264–280, 1971.

[146] Roman Vershynin. Introduction to the non-asymptotic analysis of random matrices. In Yonina C. Eldar and Gitta Kutyniok, editors, *Compressed Sensing*, pages 210–268. Cambridge University Press, 2012.

[147] Jianyu Wang, Zachary Charles, Zheng Xu, Gauri Joshi, H Brendan McMahan, Maruan Al-Shedivat, Galen Andrew, Salman Avestimehr, Katharine Daly, Deepesh Data, et al. A field guide to federated optimization. *arXiv preprint arXiv:2107.06917*, 2021.

[148] Jun Wang, Arjen P De Vries, and Marcel JT Reinders. Unifying user-based and item-based collaborative filtering approaches by similarity fusion. In *Proceedings of the 29th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 501–508, 2006.

[149] Mati Wax and Ilan Ziskind. On unique localization of multiple sources by passive sensor arrays. *IEEE Transactions on Acoustics, Speech, and Signal Processing*, 37(7):996–1000, 1989.

[150] Wikipedia. Historical annual reformulations of the influenza vaccine. https://en.wikipedia.org/wiki/Historical_annual_reformulations_of_the_influenza_vaccine, 2020.

[151] Yannis G Yatracos. Rates of convergence of minimum distance estimators and kolmogorov's entropy. *The Annals of Statistics*, pages 768–774, 1985.

[152] Xinyang Yi, Constantine Caramanis, and Sujay Sanghavi. Alternating minimization for mixed linear regression. In *International Conference on Machine Learning*, pages 613–621. PMLR, 2014.

[153] Xinyang Yi, Constantine Caramanis, and Sujay Sanghavi. Solving a mixture of many random linear equations by tensor decomposition and alternating minimization. *arXiv preprint arXiv:1608.05749*, 2016.

[154] Kai Zhong, Prateek Jain, and Inderjit S Dhillon. Mixed linear regression with multiple components. In *Advances in neural information processing systems (NIPS)*, pages 2190–2198, 2016.

[155] Banghua Zhu, Jiantao Jiao, and Jacob Steinhardt. Generalized resilience and robust statistics. *arXiv preprint arXiv:1909.08755*, 2019.