UNIVERSITY OF CALIFORNIA

Los Angeles

Efficient Algorithms
for Linear Regression and Spectrum Estimation

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

by

William Joseph Swartworth

2023

ABSTRACT OF THE DISSERTATION

Efficient Algorithms for Linear Regression and Spectrum Estimation

by

William Joseph Swartworth
Doctor of Philosophy in Mathematics
University of California Los Angeles, 2023
Professor Deanna M. Hunter, Chair

In this thesis we study efficient algorithms for solving very large linear algebra problems. We first consider the Kaczmarz method for solving linear systems, and develop a variant that is robust to a small number of large corruptions, while still requiring only a small working memory. We provide both theoretical guarantees for certain data distributions as well as empirical results showing that our approach works well in practice. We then turn our attention to problems of quickly learning spectral information about a matrix. The first such problem is PSD-testing where we give optimal query complexity bounds (with respect to types of types of queries) for distinguishing between a matrix being positive semi-definite versus having a large negative eigenvalue. Building on part of this work, we then develop optimal sketches for learning the entire spectrum of a matrix to within additive error. Finally we return our attention to solving linear systems and give new algorithms that achieve optimal communication complexity for solving least-squares regression problems.

The dissertation of William Joseph Swartworth is approved.

Georg Menz

Michael Anthony Hill

Guido Fra Montufar Cuartas

Deanna M. Hunter, Committee Chair

University of California, Los Angeles

2023

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

ACKNOWLEDGMENTS

I would first like to thank my advisor Deanna Needell for her supervision during my time at UCLA. I am extremely grateful to her and the entire research group for providing such a a supportive and welcoming environment.

I would also like to thank my other collaborators for the many stimulating discussions, and for contributing to the work contained in this thesis. Jamie Haddock and Liza Rebrova contributed to the work presented in Chapter 2, and David Woodruff contributed to the work in chapters 3, 4, and 5.

I am grateful to the Nation Science Foundation for financial support during my studies (NSF DMS #2011140 and NSF DMS #2108479).

VITA

2017	B.S. Mathematics, University of Texas at Austin
2017	B.S. Computer Science, University of Texas at Austin
2019	M.A. Mathematics, UCLA
2017 - 2023	Teaching Assistant, UCLA
2020 - 2024	Graduate Student Researcher, UCLA
2023	Graduate Student Instructor, UCLA

PUBLICATIONS

W. Swartworth, D. Woodruff. Optimal Eigenvalue Approximation via Sketching. ACM Symposium on Theory of Computing (STOC). 2023.

J. Haddock, D. Needell, E. Rebrova, W. Swartworth. Quantile-based iterative methods for corrupted systems of linear equations. SIAM Journal on Matrix Analysis and Applications. 2022.

D. Needell, W. Swartworth, D. Woodruff. Testing Positive Semidefiniteness Using Linear Measurements. IEEE 63rd Annual Symposium on Foundations of Computer Science (FOCS). 2022.

S. Li, W. Swartworth, M. Takac, D. Needell, R. Gower. SP2: A second order stochastic Polyak method. International Conference on Learning Representations (ICLR). 2023.

Y. Yaniv, J. Moorman, W. Swartworth, T. Tu, D. Landis, D. Needell. Selectable set randomized Kaczmarz. Numerical Linear Algebra with Applications. 2023.

CHAPTER 1

Introduction

As data continues to grow larger, it becomes increasing important to have fast algorithms capable of processing that data. In this thesis we focus on *extremely efficient* algorithms for solving various linear algebra problems. Depending on the context, “efficient” can refer to different metrics. For example when presented with a huge dataset one might not have the ability to load it all from disk into RAM. Or one might be receiving huge amounts of data over a stream but not possess the resources to store it all. In such cases it is useful to have algorithms that are space-efficient.

In other circumstances the data might all be available, but the dataset may be so large that carrying out an exact computation on it takes too long. In this situation we are interested in time-efficient algorithms. For very large datasets, the fastest exact algorithms may not be fast enough. It is therefore natural to relax the problem slightly and search for algorithms that approximate the desired behavior in exchange for much faster runtimes.

Finally, one might have a large amount of data that is spread over many servers or processors. In this situation, sending the data between nodes may be a bottleneck and so we are interested in finding algorithms that are efficient with respect to communication.

We will often be concerned with algorithms that are *provably optimal* with respect to these metrics. As such, much of work emphasizes lower bounds. Such results are useful to algorithm developers as they tell us whether one should invest energy in developing better algorithms for a problem, or instead finding new ways of relaxing the problem.

In Chapter 2 we consider a variant of the Kaczmarz algorithm for solving linear systems that are too large to load into memory. We propose a novel variant that can handle a small fraction of corruptions. This chapter is based off of work in [Had+22].

In Chapter 3 we consider the problem of testing if a matrix is positive-semi-definite with a small number of measurements to the matrix. We also show that our algorithms are optimal in terms of the number of measurements that are made to the matrix. This chapter is based off of work in [NSW22].

In Chapter 4 we consider the more general problem of approximating the entire spectrum of a matrix. Here we show how to recover spectral information after compressing a matrix. In a streaming setting, this yields an algorithm with low storage requirements. When the matrix is given explicitly, our algorithm allows for approximating the spectrum with nearly linear time complexity. This chapter is based off of work in [SW23].

In Chapter 5 we study solving regression problems where the rows of the system are spread across many different servers. In this setting, we are interested in minimizing the amount of communication that occurs between the servers.

CHAPTER 2

Quantile Based Iterative Methods for Solving Corrupted Systems of Linear Equations

In many situations one is interested in solving large scale systems of equations. If the system is too large however, this can pose problems for algorithms that need the entire system to operate.

In such situations it is natural to consider “row-action methods” which can make progress from only viewing a single row of the system at a time. The classic example of such an algorithm is the Randomized Kaczmarz method, which at each step chooses a random row of the system and projects onto the solution space for that row. For well-conditioned, consistent systems, the Kaczmarz method has been shown to enjoy linear convergence at a rate determined by the conditioning of the system.

Unfortunately in real-world data, overdetermined systems are rarely consistent. For example in imaging applications, a small amount of movement by the sample, or simply small inconsistencies in the detectors, result in noisy measurements. Previous work has studied analogues of Kaczmarz that are robust to such small perturbations.

However, what if a small fraction measurements are not just noisy, but completely wrong? For example in imaging applications, one might worry that the occasional measurement captures a speck of dust floating by the detector. We refer to such incorrect measurements as *corrupt* to distinguish from simply having noise. Existing versions of Kaczmarz are not robust to corrupt measurements, since projecting onto a single corrupt row of the system could erase all the progress made until that point.

In this work, we propose two novel row-action methods which are robust to corruptions. The first of these is QuantileRK which uses a version of “quantile thresholding” to avoid making projections with the potential to erase too much progress. The second is QuantileSGD which runs stochastic gradient descent with respect to the ℓ_1 loss for the system, and uses the quantile thresholding idea as inspiration to select a good step size.

2.1 Contributions

This chapter presents joint work with Deanna Needell, Jamie Haddock, and Liza Rebrova [Had+22]. Deanna Needell proposed the problem of adapting Kaczmarz-type methods to handle corrupt measurements. I proposed the main algorithms and gave an initial analysis. Jamie Haddock and Liza Rebrova both worked on refining the analysis. Jammie Haddock and I both carried out experiments. All authors contributed to writing and formatting the manuscript.

2.2 Introduction

One of the most ubiquitous problems arising across the sciences is that of solving large-scale systems of linear equations. These problems arise in many areas of data science including machine learning, as subroutines of several optimization methods [BV04], medical imaging [GBH70; HM93], sensor networks [SHS01], statistical analysis, and many more. A practical challenge in all of these settings is that there is almost always corruption present in any such large scale data, either due to data collection, transmission, adversarial components, or modern storage systems that can introduce corruptions into otherwise consistent systems of equations. For example, sensors can malfunction, or survey responses can be inconsistent. If the measurements are taken in a distributed setting by a collection of agents, some of these agents may coordinate in a distributed attack to corrupt the model. We seek methods that are robust to such corruption but scalable to big data.

In this work, we develop scalable methods for solving corrupted systems of linear equations. Here, we consider the problem of solving large scale systems of equations $\mathbf{A}\mathbf{x} = \tilde{\mathbf{b}}$ where a subset of equations have been contaminated with arbitrarily large corruptions in the measurement vector, thereby constructing an inconsistent system of equations defined by measurement matrix $\mathbf{A}$ and observed measurement vector $\mathbf{b} = \tilde{\mathbf{b}} + \mathbf{b}_C$ ($\tilde{\mathbf{b}}$ being unobserved but corresponding to the desired system of equations and $\mathbf{b}_C$ being an arbitrary corruption vector of the same dimension). Our work is motivated by the setting where the uncorrupted system of equations $\mathbf{A}\mathbf{x} = \tilde{\mathbf{b}}$ is highly overdetermined and the number of measurements is very large. In such settings, the full matrix may be too large to load into RAM. Therefore we seek methods that operate using only a small number of rows of $\mathbf{A}$ at a time.

We focus on variants of the popular iterative methods, *stochastic gradient descent* (SGD) or *randomized Kacmarz* (RK), that have gained popularity recently due to their small memory footprint and good theoretical guarantees [SV09; Bot10; NSW16]. We propose variants of both RK and SGD based upon use of *quantile statistics*. We focus on proving theoretical convergence guarantees for these variants, but additionally discuss their implementation, and present numerical experiments evidencing their promise.

The SGD method is a widely-used first-order iterative method for convex optimization [RM51]. The classical method seeks to minimize a separable objective function $f(\mathbf{x}) = \sum_{i=1}^{m} f_i(\mathbf{x})$ by accessing (stochastically) selected components of the objective and using a gradient step for this component. That is, SGD constructs iterates $\mathbf{x}_k$ given by

$$\mathbf{x}_{k+1} = \mathbf{x}_k - \gamma_k \nabla f_i(\mathbf{x}_k) \tag{2.1}$$

where γ_k is the learning rate (or step-size) and i is the selected component for the kth iteration. When the objective function $f(\mathbf{x})$ represents error in the solution of a system of equations, SGD generally updates in the direction of the sampled row, i.e., $\mathbf{x}_{k+1} - \mathbf{x}_k = \alpha_k \mathbf{a}_i$ for some α_k which depends upon the iterate $\mathbf{x}_k$. Our variants apply SGD to the *least absolute deviations (LAD)* error and *least squares (LS)* error,

$$f(\mathbf{x}) = \|\mathbf{A}\mathbf{x} - \mathbf{b}\|_1 = \sum_{i=1}^{m} |\langle \mathbf{a}_i, \mathbf{x}\rangle - b_i| \quad \text{and} \quad f(\mathbf{x}) = \frac{1}{2}\|\mathbf{A}\mathbf{x} - \mathbf{b}\|^2 = \frac{1}{2}\sum_{i=1}^{m} \left(\langle \mathbf{a}_i, \mathbf{x}\rangle - b_i\right)^2,$$

respectively. For these objectives, the SGD updates (2.1) take the form

$$\mathbf{x}_{k+1} = \mathbf{x}_k - \gamma_k \text{sign}(\langle \mathbf{a}_i, \mathbf{x}_k\rangle - b_i)\mathbf{a}_i \quad \text{and} \quad \mathbf{x}_{k+1} = \mathbf{x}_k - \gamma_k(\langle \mathbf{a}_i, \mathbf{x}_k\rangle - b_i)\mathbf{a}_i,$$

respectively, where $\text{sign}(\cdot)$ denotes the function that returns 1 if its argument is positive and -1 otherwise. The RK updates are a specific instance of the SGD updates for the LS error where $\gamma_k = 1/\|\mathbf{a}_i\|^2$ [NSW16]; that is

$$\mathbf{x}_{k+1} = \mathbf{x}_k + \frac{b_i - \langle \mathbf{a}_i, \mathbf{x}_k\rangle}{\|\mathbf{a}_i\|^2}\mathbf{a}_i. \tag{2.2}$$

In [SV09], the authors showed that when applied to a consistent system of equations with a unique solution $\mathbf{x}^*$ and with a specific sampling distribution, RK converges at least linearly in expectation. Indeed, denoting $\mathbf{e}_k := \mathbf{x}_k - \mathbf{x}^*$ as the difference between the k-th iterate of the method and the exact solution of the system, the method guarantees

$$\mathbb{E}\|\mathbf{e}_k\|^2 \leq \left(1 - \frac{\sigma_{\min}^2(\mathbf{A})}{\|\mathbf{A}\|_F^2}\right)^k \|\mathbf{e}_0\|^2, \tag{2.3}$$

where $\|\cdot\|_F$ denotes the Frobenius norm and $\sigma_{\min}(\mathbf{A})$ the smallest (nonzero) singular value of $\mathbf{A}$. Standard SGD results (e.g., [SZ13]) provide similar convergence rates for SGD on these objectives when the stepsizes are chosen according to an appropriately decreasing schedule. See Section 2.2.3 below for more details and a discussion of related work.

Here, we consider variants of the SGD and RK methods that converge to the solution of the uncorrupted system even in the presence of large corruptions in the measurement vector $\mathbf{b}$. We prove convergence rates in the same form as (2.3). It is worth noting that both our experimental and theoretical results illustrate that the size of the corruptions do not negatively impact the convergence of the proposed methods. Our methods will make use of SGD and RK steps but will use a quantile of the residual entries in order to determine the step-size.

2.2.1 Organization

The rest of our paper is organized as follows. In the remainder of the introduction, we present our main contributions in Section 2.2.2, discuss related works in Section 2.2.3, and briefly describe our notations and give required definitions in Section 2.2.4. We then provide the detailed pseudocode of our proposed methods, QuantileRK(q) and QuantileSGD(q), in Section 2.3. We state and prove our theoretical results in Section 2.4. Within this section, we highlight some new results for random matrices as useful tools in Subsection 2.4.1 and then include the proofs of our main convergence results in Subsections 2.4.2 and 2.4.3. In Section 2.5, we discuss several implementation considerations that affect the efficiency and convergence of our proposed methods. In Section 2.6, we empirically demonstrate the promise of our methods with experiments on synthetic and real data. Finally, we conclude and offer some future directions in Section 2.7.

2.2.2 Contributions

In this section, we provide summaries of foundational results we prove in high-dimensional probability, then state our main convergence results for the proposed methods. Our main convergence results rely on the following assumptions about the linear system $\mathbf{A}\mathbf{x} = \mathbf{b}$. Let $\mathbf{A} \in \mathbb{R}^{m \times n}$ be a random matrix with $m \geq n$. We direct the reader to [Ver18] for the random matrix theory definitions involved; we also provide summaries in Section 2.2.4.

Assumption 1. *All the rows $\mathbf{a}_i$ of the matrix $\mathbf{A}$ have unit norm and are independent. Additionally, for all $i \in [m]$, $\sqrt{n}\mathbf{a}_i$ is mean zero isotropic and has uniformly bounded subgaussian norm*

$\|\sqrt{n}\mathbf{a}_i\|_{\psi_2} \leq K$.

Assumption 2. *Each entry a_{ij} of $\mathbf{A}$ has probability density function ϕ_{ij} which satisfies $\phi_{ij}(t) \leq D\sqrt{n}$ for all $t \in \mathbb{R}$. (The quantity D is a constant which we will use throughout when referring to this assumption.)*

The prototypical example of a matrix satisfying both assumptions is a normalized Gaussian matrix, i.e., a matrix whose rows are sampled uniformly over S^{n-1}. In this case, the entries are effectively standard normal, scaled by $\frac{1}{\sqrt{n}}$, so one can take $D \approx \frac{1}{\sqrt{2\pi}}$ and $K \approx 2$

Assumptions 1 and 2 extract the properties of Gaussian matrices that are required for our theory. As such, our work applies to more general distributions, whenever there is enough independence between the entries of the matrix and their distributions are regular enough.

By Assumptions 1 and 2, the matrices we consider will be full rank almost surely so the uncorrupted system $\mathbf{A}\mathbf{x} = \tilde{\mathbf{b}}$ will always have a unique solution $\mathbf{x}^*$.

High-dimensional probability results

Our main convergence guarantees build upon several useful results related to the non-asymptotic properties of random matrices that appear to be new and that may be of independent interest.

In particular, Proposition 2.7 shows that for a class of random matrices, *all* large enough submatrices uniformly have smallest singular values that are at least on the order of $\sqrt{m/n}$. For matrices which satify Assumptions 1 and 2, this generalizes standard bounds on the smallest singular value, but does not follow directly from these bounds.

Proposition 2.12 is more specialized, but may also be of independent interest. For a random matrix $\mathbf{A}$, we show that the average magnitude of the entries of $\mathbf{A}\mathbf{x}$ is well concentrated *uniformly* in $\mathbf{x}$. In fact, $\mathbf{A}$ does not need to be very tall for this result to hold; a constant aspect ratio suffices.

Main results

We first introduce two new methods for iteratively solving linear systems with corruptions and give the formal statements of our main results. The first method we introduce is **QuantileRK**, which builds upon the RK method. Recall that the iteration of RK given by (2.2) implies that the method proceeds by sampling rows of the matrix $\mathbf{A}$ and projecting onto the corresponding hyperplane given by the linear constraint. When some of the entries in $\mathbf{b}$ are corrupted by a large amount, RK periodically projects onto the associated corrupted hyperplanes and therefore does not converge. Our solution is to avoid making projections that result in $\|\mathbf{x}_{k+1} - \mathbf{x}_k\|$ being abnormally large. Specifically, for each iterate $\mathbf{x}_k$ we consider the set of distances from $\mathbf{x}_k$ to a set of t sampled hyperplane constraints.[1] We assign a threshold value to be the q-quantile of these distances, where q is a parameter of the method. If the distance from $\mathbf{x}_k$ to the sampled hyperplane is greater than this threshold then the method avoids projecting during that iteration. Otherwise it projects in the same manner as RK.

Theorem 2.1 states that the QuantileRK method converges for random matrices satisfying Assumptions 1 and 2 above, as long as the fraction of corrupted entries is a sufficiently small constant (which does not depend on the dimensions of the matrix). Here and throughout, $c, C, c_1, C_1, \ldots$ denote absolute constants that may denote different values from one use to the next. Variable subscripts on constants will indicate quantities that a given constant may depend on. For example, C_1 only depends on q.

Theorem 2.1. *Let the system be defined by random matrix* $\mathbf{A} \in \mathbb{R}^{m \times n}$ *satisfying Assumptions 1 and 2, with the constant parameters* D *and* K.[2] *Then with probability* $1 - c\exp(-c_q m)$, *the iterates produced by the QuantileRK*(q) *Method 1 with* $q \in (0, 1)$, *where in each iteration the quantile is computed using the full corrupted residual (instead of subsampling, we use* $t = m$*), and initialized*

[1] In order to allow more efficient implementations, we empirically show that considering a small subset of hyperplanes is sufficient. One could extend the theory to this setting as well, with a slightly more complicated analysis.

[2] In other words we do not track the dependencies on D and K.

with arbitrary point $\mathbf{x}_0 \in \mathbb{R}^n$ *satisfy*

$$\mathbb{E}\left(\|\mathbf{e}_k\|^2\right) \leq \left(1 - \frac{C_q}{n}\right)^k \|\mathbf{e}_0\|^2$$

as long as the fraction of corrupted entries $\beta = |\text{supp}(\mathbf{b}_C)|/m < \min(c_1 q^2, 1-q)$ *and* $m \geq Cn$. *(Recall that* $\mathbf{e}_k$ *denotes the error vector* $\mathbf{x}_k - \mathbf{x}^*$*.)*

The second method we introduce is **QuantileSGD**, which is a variant of SGD in which the step-size used in each iteration is chosen to avoid abnormally large steps. Specifically, for each iterate $\mathbf{x}_k$, we consider the set of distances from $\mathbf{x}_k$ to the set of t sampled hyperplane constraints specified by our linear system.[1] We choose the step-size as the q-quantile of these distances, where q is a parameter of the method. This prevents projections that are on the order of distances associated to corrupted equations.

Under nearly the same assumptions for the system and slightly more restrictive assumptions on the quantile parameter, we also guarantee an RK-type convergence rate for QuantileSGD(q). Our second main result is Theorem 2.2, which shows that QuantileSGD converges, again when the fraction of corruptions is sufficiently small.

Theorem 2.2. *Let the system be defined by random matrix* $\mathbf{A} \in \mathbb{R}^{m \times n}$ *satisfying Assumptions 1 and 2 with the constant parameters* D *and* K.[2] *Then with probability at least* $1 - c\exp(-c_q m)$*, the iterates produced by the QuantileSGD*$(q-\beta)$ *Method 2 with* $q \in (0, 1/2)$*, where in each iteration the quantile is computed using the full corrupted residual (instead of subsampling, we use* $t = m$*), and initialized with arbitrary point* $\mathbf{x}_0 \in \mathbb{R}^n$ *satisfy*

$$\mathbb{E}\left(\|\mathbf{e}_k\|^2\right) \leq \left(1 - \frac{C_q}{n}\right)^k \|\mathbf{e}_0\|^2$$

as long as the fraction of corrupted entries $\beta = |\text{supp}(\mathbf{b}_C)|/m$ *is a sufficiently small constant and* $m \geq Cn\log n$. *(Recall that* $\mathbf{e}_k$ *denotes the error vector* $\mathbf{x}_k - \mathbf{x}^*$*.)*

In order to prove this result, we first introduce a method that we call OptSGD, which adaptively chooses an optimal step size at each iteration. This method cannot be run in practice as it requires knowledge of $\mathbf{x}^*$. However, we are able to show that QuantileSGD approximates OptSGD and therefore performs similarly well. OptSGD may also serve as a useful benchmark when considering other SGD-type solvers for linear systems.

In each of these results, we make extensive use of theorems in high dimensional probability and do not attempt to track constants. In particular we allow our constants to depend on parameters of the underlying distributions on the rows. We remedy this with our empirical results, which show that QuantileRK and QuantileSGD work well for practical sets of parameters.

Finally, we consider a simpler setting that we call the *streaming* setting, which may be viewed as the limiting case when the number of rows of $\mathbf{A}$ tends to infinity. In this situation we do not rely on the non-asymptotic properties of random matrices and are able to give an analysis with better constants for the case when the matrix has Gaussian rows. In particular, Theorem 2.21 shows that our methods can handle a 0.35 fraction of corruptions, even when the values of the corruptions are chosen by an adversary. In practice, we see that the proposed methods (including the non-streaming setting) are able to accommodate much more complex cases when up to one half of the equations are corrupted.

Remark 2.3. *We get the same standard convergence rate for both methods; however, for QuantileSGD(q) we have a slightly stronger requirement on the aspect ratio of the matrix* $\mathbf{A}$*, and an additional restriction for the quantile* $q < 1/2$ *(whereas QuantileRK is proved for any quantile* $q \in (0,1)$*) In practice, QuantileSGD indeed diverges for the value of a quantile too close to one (see Figure 2.1 (b)); however, one could safely use a much wider range of quantiles. We note that for a normalized Gaussian model (when the rows of* $\mathbf{A}$ *are sampled from the uniform distribution on the unit sphere) one can use the QuantileSGD(q) method for all* $q \leq 0.75$ *(see Remark 2.20).*

2.2.3 Related Works

There are many extensions and analyses of the SGD and RK methods; we review some of the results most relevant to our contributions. The first two sections deal with consistent or *noisy* systems, while the last section deals with methods for the problem of *corrupted* systems. We distinguish between *corruption*, in which there are few but relatively large errors in the measurement vector, and *noise*, in which there are many but relatively small errors in the measurement vector; the latter is more commonly considered within the SGD and RK literature.

Randomized Kaczmarz variants. The Kaczmarz method was proposed in the late 30s by Stefan Kaczmarz [Kac37]. The iterative method for solving consistent systems of equations was rediscovered and popularized for computed tomography as *algebraic reconstruction technique (ART)* [GBH70; HM93]. While it has enjoyed research focus since that time [CEG83; Nat86; SS87; Fei+92; HN90; FS95], the elegant analysis of the *randomized Kaczmarz* method of [SV09] has spurred a surge of research into variants of the Kaczmarz method. In [SV09], the authors proved the first exponential convergence rate in expectation (2.3) in the case of full-rank and consistent systems of equations. This result was generalized to the case when $\mathbf{A}$ is not full-rank in [ZF13]. Block methods which make use of several rows in each iteration have also become popular [EHL81; Elf80; Pop99; Pop01; NT14; RN20].

One relevant and well-studied variant of the Kaczmarz method is that in which the row selection is performed greedily rather than randomly. This greedy variant goes by the name *Motzkin's relaxation method for linear inequalities (MM)* in the linear programming literature [MS54; Agm54], where convergence analyses coinciding with (2.3) have been shown [Agm54]. MM has been rediscovered in the Kaczmarz literature under the name "most violated constraint control" or "maximal-residual control" [Cen81; Nut+16; PP15]. Several greedy extensions and hybrid randomized and greedy methods have been proposed and analyzed [BW18a; BW18b; DHN17; MIN19; LR19; MI+20; HM19]. Like our methods, these greedy approaches require that sufficiently large entries of the residual be identified; however, these methods differ from ours in how these residual entries are used.

Another relevant direction in the Kaczmarz literature are convergence analyses for systems in which the measurement matrices $\mathbf{A}$ have entries sampled according to a given probability distribution [CP12; HN19; HM19; RN20]. Our main results will make mild assumptions on the distribution of the entries of the measurement matrix.

The convergence of many of the previously mentioned methods has been analyzed in the case that there is a small amount of noise in the system. Generally, these analyses provide a *convergence horizon* around the solution that depends upon the size of the entries of the noise. In [Nee10], the author proves that RK converges on inconsistent linear systems to a horizon which depends upon the size of the largest entry of the noise; a similar result is shown in [HN19] for MM. In [ZF13; DSS20], the authors develop methods that converge to the least-squares solution of a noisy system. Meanwhile, in this work, our focus will be developing methods for systems in which there is *corruption* rather than noise. We will exploit the fact that the overdetermined system of equations has few corruptions in order to solve the uncorrupted system of equations.

Stochastic gradient descent variants. There has been an abundance of work developing and analyzing variants of SGD (e.g., step-size schedules, variants for specific and non-smooth objectives, etc.). This is not meant to be a thorough survey of the literature in this area; we direct the reader to [BCN18] and the references therein for a survey of recent advances, and outline here those most relevant to our approach.

In [RM51], the authors provide a convergence analysis for SGD in the case that the objective is smooth and strongly convex and the step-size schedule diminishes at the appropriate rate. Such convergence results hold for fixed step-size schedules, but include a constant error term akin to the convergence horizon of RK for inconsistent systems [NSW16]. Similar convergence rates can be proved in the case of non-smooth and non-strongly convex objectives [SZ13]; this result assumes an appropriately decreasing step-size schedule, and prove bounds on the objective value optimality gap. Our results, unlike these, will use an iterate dependent step-size and will provide bounds on the distance between iterates and the solution of the uncorrupted system.

Recently, batch variants that use several samples in each iteration have become popular and enjoy similar rates [Dek+12]. In [KL20], the authors propose and analyze a greedy variant of SGD known as *ordered SGD* that selects batches of the gradient according to the value of the associated objective components.

An important branch of advances in the analysis of SGD deal with robustness to corruption and outliers in the objective defining data and sampled gradients, see e.g., [Chi+19; Li+20]. Similar to our work, the aforementioned papers use quantile statistics, namely, a median-truncated SGD. Our methods differ from these in how we use the quantile statistic to achieve robustness to corruption and in our specification to linear systems.

Here, we focus on the SGD variants developed for the LAD error; this problem is often known as LAD regression. It has been previously noted that the ℓ_1 objective is more robust to outliers than the ℓ_2 objective [WGZ06]; for this reason, there have been many algorithmic approaches to LAD regression. These approaches have been motivated by maximum likelihood approaches [LA04], rescaling techniques for low-dimensional problems [BS80], iterative re-weighted least-squares [Sch73], descent approaches [Wes81], dimensionality reduction [KS18], or linear programming approaches [BR73]; see [GSN88] and references therein.

Corrupted linear systems approaches. The corrupted linear system problem has been studied within the error-correction literature and has been formulated in the compressed sensing framework. Many recovery results build upon and resemble those within the compressed sensing literature [CT05]. In particular, the optimization problem $\min \|\mathbf{A}\mathbf{x} - \mathbf{b}\|_0$ is a special case of the NP-hard MAX-FS problem [AK95]. However, if the measurement matrix $\mathbf{A}$ and the support of the corruption vector $\mathbf{b}_C$ satisfy appropriate properties, then the minimizer of the ℓ_0 problem and the ℓ_1 problem coincide and the problem can be solved using e.g., linear programming methods [CT05; Can+05; WM10]. Such methods however are quite slow and have large memory requirements if the system is extremely large. In addition, we do not place such assumptions on our systems here.

Previous work has developed and analyzed iterative methods for corrupted systems of equations. As mentioned previously, much of the focus on this problem has been in the error correction and compressed sensing literature [FR13; EK12]. However, there has been work that has focused on iterative row-action methods; previous work in this direction includes [HN18a; JCC15; ABH05].

Our work was inspired by [HN18b; HN18a], in which the authors propose and analyze randomized Kaczmarz variants that detect and remove corrupted equations in the system. These methods differ from ours in that they exploit the ability of the standard RK method to detect and avoid few corruptions. Meanwhile, our work develops variants of RK and SGD that use quantiles of the residual to converge even in the presence of corruptions. In [Had+20], we present several methods related to those here; our results will significantly improve and generalize those in [Had+20].

2.2.4 Notation and Definitions

We consider a system with measurement matrix $\mathbf{A} \in \mathbb{R}^{m \times n}$ and corrupted measurement vector $\mathbf{b} \in \mathbb{R}^m$ and $m \gg n$. We denote the ith row of $\mathbf{A}$ by $\mathbf{a}_i$. If $\mathbf{A}$ is an $m \times n$ matrix and $S \subset [m]$, then let $\mathbf{A}_S$ denote the matrix obtained by restricting to the rows S.

The corrupted measurement vector $\mathbf{b}$ is the sum of the ideal (uncorrupted) measurement vector $\tilde{\mathbf{b}}$ and the corruptions $\mathbf{b}_C$. The number of corruptions is a fraction $\beta \in (0, 1)$ of the total number of measurements, $|\text{supp}(\mathbf{b}_C)| = \beta m$. Here $\text{supp}(\mathbf{x})$ denotes the set of indices of nonzero entries of $\mathbf{x}$. The ideal measurement vector $\tilde{\mathbf{b}}$ defines a consistent system of equations with ideal solution $\mathbf{x}^*$. We denote the k-th error as $\mathbf{e}_k := \mathbf{x}_k - \mathbf{x}^*$, where $\mathbf{x}_k$ denotes the k-th iterate of a method.

The notation $\|\mathbf{v}\|$ denotes the Euclidean norm of a vector $\mathbf{v}$. We denote the sphere in $\mathbb{R}^n$ as S^{n-1}, so $S^{n-1} = \{\mathbf{x} \in \mathbb{R}^n : \|\mathbf{x}\| = 1\}$. For a matrix $\mathbf{A}$, we denote its operator ($L_2 \to L_2$) norm by $\|\mathbf{A}\| = \sup_{\mathbf{x} \in S^{n-1}} \|\mathbf{A}\mathbf{x}\|$ and its Frobenious (or Hilbert-Schmidt) norm by $\|\mathbf{A}\|_F = \sqrt{\text{trace}(\mathbf{A}^\top \mathbf{A})}$. Throughout, we denote by $\sigma_{\min}(\mathbf{A})$ and $\sigma_{\max}(\mathbf{A})$ the smallest and largest singular values of the matrix $\mathbf{A}$ (that is, eigenvalues of the matrix $\sqrt{\mathbf{A}^\top \mathbf{A}}$). Moreover, we always assume that the matrix $\mathbf{A}$ has full column rank, so that $\sigma_{\min}(\mathbf{A}) > 0$ and the convergence rate is nontrivial. We also denote the (scaled) condition number of the matrix as $\kappa(\mathbf{A}) = \|\mathbf{A}\|_F / \sigma_{\min}(\mathbf{A}) =$

$\|\mathbf{A}\|_F \|\mathbf{A}^{-1}\|$, where $\|\mathbf{A}^{-1}\|$ is defined to be $1/\sigma_{\min}(\mathbf{A})$.

Additionally, our work relies on several concepts that arise in high dimensional probability. We list all relevant definitions here, proper review of the concepts and their properties can be found in e.g., [Ver18]. If X is a real-valued random variable, then the sub-Gaussian norm of X is defined to be $\|X\|_{\Psi_2} = \inf\{t > 0 : \mathbb{E}\exp(X^2/t^2) \leq 2\}$. If $\mathbf{v}$ is a random vector in $\mathbb{R}^n$, then the the sub-Gaussian norm of $\mathbf{v}$ is defined to be $\|\mathbf{v}\|_{\Psi_2} = \sup_{\mathbf{x}\in S^{n-1}} \|\langle \mathbf{v}, \mathbf{x}\rangle\|_{\Psi_2}$. A random variable is said to be sub-Gaussian if it has finite sub-Gaussian norm. If $\mathbf{v}$ is a random vector in $\mathbb{R}^n$ then $\mathbf{v}$ is said to be isotropic if $\mathbb{E}(\mathbf{v}\mathbf{v}^\top) = I_n$ where I_n is the identity on $\mathbb{R}^n$.

Our convergence analyses will take expectation with regards to the random sample taken in each iteration. We denote expectation taken with regards to all iterative samples as $\mathbb{E}$. We denote by $\mathbb{E}_k$ the expectation with respect to the random sample selected in the kth iteration, conditioned on the results of the $k-1$ previous iterations of the method.

We use the following notations for the statistics of the corrupted and uncorrupted residual. We let $Q_q(\mathbf{x})$ denote the empirical q-quantile of the corrupted residual,

$$Q_q(\mathbf{x}) := q\text{- quantile}\{|\langle \mathbf{a}_i, \mathbf{x}\rangle - b_i| : i \in [m]\}. \tag{2.4}$$

For our purposes, the q-quantile of a multi-set S is defined to be the $\lfloor q|S| \rfloor^{\text{th}}$ smallest entry of S.

We let $\tilde{Q}_q(\mathbf{x})$ denote the empirical q-quantile of the uncorrupted residual,

$$\tilde{Q}_q(\mathbf{x}) := q\text{- quantile}\left\{|\langle \mathbf{x} - \mathbf{x}^*, \mathbf{a}_i\rangle| : i \in [m]\right\}. \tag{2.5}$$

We additionally define notation for the quantile statistics of sampled portions of the corrupted and uncorrupted residuals,

$$Q_q(\mathbf{x}, S) := q\text{- quantile}\{|\langle \mathbf{a}_i, \mathbf{x}\rangle - b_i| : i \in S\} \tag{2.6}$$

and

$$\tilde{Q}_q(\mathbf{x}, S) := q\text{- quantile}\left\{|\langle \mathbf{x} - \mathbf{x}^*, \mathbf{a}_i\rangle| : i \in S\right\} \tag{2.7}$$

where $S \subset [m]$ is the set of sampled indices. Note that only Q_q is available to us at run time since it makes use of the corrupted measurement vector $\mathbf{b}$; $\tilde{Q}_q$ is not available due to the use of unknown $\mathbf{x}^*$. We employ $\tilde{Q}_q$ in our theoretical results as it allows us to naturally relate Q_q and random matrix parameters. Finally, we let $M(\mathbf{x})$ denote the average magnitude of the entries of $\mathbf{Ax}$,

$$M(\mathbf{x}) := \frac{1}{m}\sum_{i=1}^{m} |\langle \mathbf{x}, \mathbf{a}_i\rangle| . \tag{2.8}$$

This quantity will be useful later when we wish to bound the quantiles computed by QuantileRK and QuantileSGD.

We will also make use of a Chernoff bound several times. As there are many variants, we give a statement for completeness.

Chernoff Bound. Let $X_1, \dots X_n$ be i.i.d random variables with expectation p taking values in $\{0, 1\}$. Let $X = X_1 + \ldots + X_n$, and let μ be the expected value of the sum. Then

$$\Pr\left(X \geq ((1+\delta)\mu)\right) \leq \exp\left(-\delta^2\mu/3\right).$$

2.3 Proposed Methods

In this section, we give formal descriptions of the proposed *QuantileRK(q)* and *QuantileSGD(q)* methods. Our methods use the q-quantile entry of the residual $|\mathbf{Ax} - \mathbf{b}|$ as a proxy to avoid large steps in the direction of corrupted equations. Namely, in both methods, in each iteration we sample not only an index for the RK update (which we will call the *RK-index*), but also t additional indices. We then access the entries of the residual associated to these indices and compute their empirical q-quantile, $Q_q(\mathbf{x}, \{i_l : l \in [t]\})$.

Then, the QuantileRK(q) method below takes the step (associated to the RK-index and governed by standard RK projection (2.2)) only if the entry of the residual associated to this index is less than or equal to $Q_q(\mathbf{x}_{j-1}, \{i_l : l \in [t]\})$; we say that a row $\mathbf{a}_i$ of A is *acceptable* on a given iteration if this is true. We assume that the rows of our system are normalized. If this is not the case, one could normalize the rows as they are sampled.

Algorithm 1 QuantileRK(q)

```
procedure QuantileRK($\mathbf{A}$, b, q, t, N)
    $\mathbf{x}_0 = \mathbf{0}$
    for j = 1, ..., N do
        sample $i_1, \dots i_t \sim$ Uniform$(1, \dots, m)$
        sample $k \sim$ Uniform$(1, \dots, m)$
        if $|\langle \mathbf{a}_k, \mathbf{x}_{j-1} \rangle - b_k| \leq Q_q(\mathbf{x}_{j-1}, \{i_l : l \in [t]\})$ then
            $\mathbf{x}_j = \mathbf{x}_{j-1} - (\langle \mathbf{x}_{j-1}, \mathbf{a}_k \rangle - b_k)\, \mathbf{a}_k$
        else
            $\mathbf{x}_j = \mathbf{x}_{j-1}$
        end if
    end for
     return $\mathbf{x}_N$
end procedure
```

The QuantileSGD(q) method, Method 2 uses the same quantile of the sampled residual $Q_q(\mathbf{x}_{j-1}, \{i_l : l \in [t]\})$ to define the step size. The method steps along the direction defined by the RK update (2.2) based on the RK-index with step size γ equal to $Q_q(\mathbf{x}_{j-1}, \{i_l : l \in [t]\})$.

Algorithm 2 QuantileSGD(q)

```
procedure QuantileSGD($\mathbf{A}$, b, q, t, N)
    $\mathbf{x}_0 = \mathbf{0}$
    for j = 1, ..., N do
        sample $i_1, \dots i_t \sim$ Uniform$(1, \dots, m)$
        sample $k \sim$ Uniform$(1, \dots, m)$
        $\gamma = Q_q(\mathbf{x}_{j-1}, \{i_l : l \in [t]\})$
        $\mathbf{x}_j = \mathbf{x}_{j-1} - \gamma \cdot \text{sign}\left(\langle \mathbf{x}_{j-1}, \mathbf{a}_k \rangle - b_k\right) \mathbf{a}_k$
    end for
     return $\mathbf{x}_N$
end procedure
```

Note that this pseudocode uses only the maximum number of iterations N as stopping criterion, but one could also run these methods for a specific amount of time, or implement any other stopping criterion.

Finally, we note that the behavior of both the QuantileRK and QuantileSGD depend heavily upon the input parameters. We clarify required constraints on these parameters in the theoretical results in Section 2.4. Additionally, we discuss the effect of these parameter choices on computation and other implementation considerations in Section 2.5.

2.4 Theoretical Results

Here we state and prove our theoretical results. We begin with foundational results in high-dimensional probability in Subsection 2.4.1 and then prove our main convergence results, Theorems 2.1 and 2.2, in Subsections 2.4.2 and 2.4.3. In our proof of convergence of QuantileSGD(q), Theorem 2.2, we propose an ideal method, OptSGD, and demonstrate that it is well approximated by QuantileSGD(q). We additionally prove convergence of QuantileSGD(q) in the simpler streaming setting in Subsection 2.4.3.

2.4.1 Theoretical Foundations

In this subsection, we prove several fundamental results which we apply in our convergence analyses for QuantileRK and QuantileSGD in Sections 2.4.2 and 2.4.3.

Auxiliary results – properties of random matrices

For the largest singular values of a random matrix with independent isotropic rows, we will be using the following standard bound (the proof can be found in e.g., [Ver18, Theorem 4.6.1]).

Theorem 2.4. *Let $\mathbf{A} \in \mathbb{R}^{m\times n}$ be a matrix whose rows are independent, mean zero, sub-Gaussian and isotropic with sub-Gaussian norm bounded by K. Then the largest singular value (operator norm) of $\mathbf{A}$ is bounded by*

$$\sqrt{m} + CK^2(\sqrt{n} + t)$$

with probability at least $1 - 2\exp(-t^2)$.

The smallest singular value of random matrices is sometimes called the "hard edge" as it is typically harder to quantify. This is the case in our application as well; we will prove Proposition 2.7 giving a uniform lower bound on the singular values of the submatrices of $\mathbf{A}$.

The first ingredient that we need for this (and it will be used in other places later in the text as well) is an *ϵ-net for the unit sphere*. We say that $\mathcal{N}$ is an ϵ-net of a set $S \subseteq \mathbb{R}^n$ if $\mathcal{N}$ is a subset of S and each point in S is within a Euclidean distance ϵ of some point in $\mathcal{N}$. The ϵ-covering number of S is the cardinality of the smallest ϵ-net for S. We will use the fact that the ϵ-covering number of S^{n-1} is bounded by $(3/\epsilon)^n$ [Ver18, Corollary 4.2.13].

We will also use the following direct corollary of Hoeffding's inequality (see, e.g., [Ver18, Theorem 2.6.2]) that subgaussian random variables concentrate as well as Gaussians under taking means.

Lemma 2.5. *Let $X_1, \dots, X_m$ be i.i.d. subgaussian random variables with subgaussian norm K. Then the subgaussian norm of the mean satisfies*

$$\left\| \frac{1}{m} \sum_{i=1}^{m} X_i \right\|_{\Psi_2} \leq C \frac{K}{\sqrt{m}}.$$

Next, the following anti-concentration lemma for random vectors with bounded density is a direct corollary of [RV15, Theorem 1.2].

Lemma 2.6. *Let $\mathbf{x}$ be a random vector in $\mathbb{R}^n$ such that the density function of each coordinate x_i is bounded by $D\sqrt{n}$, where $D > 0$ is an absolute constant. Then for any fixed $\mathbf{u} \in S^{n-1}$ we have*

$$\Pr\left(|\langle \mathbf{x}, \mathbf{u} \rangle| \leq \frac{\sqrt{t}}{\sqrt{n}} \right) \leq 2\sqrt{2} D \sqrt{t}.$$

We will use this anti-concentration result to prove a uniform lower bound for the smallest singular value over all $\alpha m \times n$ submatrices of a tall random matrix of the size $m \times n$. It is well known that for a single fixed (row-)submatrix $\mathbf{A}_T$ of that size, $\sigma_{\min}(\mathbf{A}_T) \gtrsim \sqrt{m}/\sqrt{n}$ (see e.g., [Ver18, Theorem 4.6.1]). However, naively taking a union bound over all $\binom{m}{\alpha m}$ αm-tall row submatrices results in a trivial probability bound. In Proposition 2.7, we provide a more delicate row-wise analysis by employing Chernoff's bound to provide a good uniform lower bound with probability exponentially close to one.

Proposition 2.7. *Let $\alpha \in (0, 1]$ and let random matrix $\mathbf{A} \in \mathbb{R}^{m\times n}$ satisfy Assumptions 1 and 2. Then there exist absolute constants $C_1, c_2 > 0$ so that if the matrix $\mathbf{A}$ is tall enough, namely,*

$$\frac{m}{n} > C_1 \frac{1}{\alpha} \log \frac{DK}{\alpha}, \tag{2.9}$$

then the following uniform lower bound holds for the smallest singular values of all its row submatrices that have at least αm rows.

$$\Pr\left(\inf_{\substack{T\subseteq[m]:\\ |T|\geq \alpha m}} \sigma_{\min}(\mathbf{A}_T) \geq \frac{\alpha^{3/2}}{24D}\sqrt{\frac{m}{n}} \right) \geq 1 - 3\exp(-c_2 \alpha m)$$

Proof. Let $\epsilon \in (0, 1]$ be a constant (chosen below in (2.11)). Recall that there is an ϵ-net $\mathcal{N}$ of S^{n-1} with cardinality $|\mathcal{N}| \leq \left(\frac{3}{\epsilon}\right)^n$. That is, for any $\mathbf{y} \in S^{n-1}$ there exists $\mathbf{x} \in \mathcal{N}$ such that $\|\mathbf{y} - \mathbf{x}\|_2 \leq \epsilon$. Taking the infimum over all unit norm vectors $\mathbf{x}$, we get that for any $T \subseteq [n]$, we have

$$\sigma_{\min}(\mathbf{A}_T) = \inf_{\mathbf{y}\in S^{n-1}} \|\mathbf{A}_T \mathbf{y}\| \geq \left(\inf_{\mathbf{x}\in\mathcal{N}} \|\mathbf{A}_T\, \mathbf{x}\| \right) - \epsilon \,\|\mathbf{A}_T\|_. \tag{2.10}$$

We will bound the two terms in the right hand side of (2.10) separately. First, for any subset $T \subset [n]$, we can bound $\|\mathbf{A}_T\|_{\text{op}} \leq \|\mathbf{A}\|_{\text{op}}$, and so by Theorem 2.4

$$\Pr\left(\|\mathbf{A}_T\| \leq (1 + CK^2)\sqrt{\frac{m}{n}} \right) \geq 1 - 2\exp(-cm)$$

for some absolute constants $C, c > 0$. Let us choose

$$\epsilon = \frac{\alpha^{3/2}}{24D(1+CK^2)}. \tag{2.11}$$

To bound the first term in the right-hand side of (2.10), first consider a fixed $\mathbf{x}$ in $\mathcal{N}$. For $i \in [m]$, let $\mathcal{E}_i^{\mathbf{x}}$ be the event

$$\mathcal{E}_i^{\mathbf{x}} := \left\{ |\langle \mathbf{a}_i, \mathbf{x} \rangle|^2 < \frac{\alpha^2}{64D^2} \cdot \frac{1}{n} \right\}.$$

By Lemma 2.6, $\Pr(\mathcal{E}_i^{\mathbf{x}}) \leq \alpha/4$ for each fixed $\mathbf{x} \in S^{n-1}$ and $i \in [m]$. A Chernoff bound then gives

$$\Pr\text{(events } \mathcal{E}_i^{\mathbf{x}} \text{ occur for at least } \alpha m/2 \text{ indices } i \in [m] \text{)} \leq \exp(-\alpha m/12).$$

Now, for any fixed $\mathbf{x}$, provided that $\mathcal{E}_i^{\mathbf{x}}$ occurs for at most $\alpha m/2$ indices $i \in [m]$, for all T with $|T| \geq \alpha m$ we have

$$\|\mathbf{A}_T\,\mathbf{x}\| \geq \sqrt{\left(\frac{\alpha}{2}m\right) \cdot \left(\frac{\alpha^2}{64D^2} \cdot \frac{1}{n}\right)} = \frac{\alpha^{3/2}}{12D}\sqrt{\frac{m}{n}}.$$

Finally, taking a union bound over $\mathbf{x} \in \mathcal{N}$, we have

$$\Pr\left(\inf_{\mathbf{x} \in \mathcal{N}} \|\mathbf{A}_T\,\mathbf{x}\| \leq C_\alpha \sqrt{\frac{m}{n}}\right) \leq \exp\left(n \log \frac{3}{\epsilon} - \frac{\alpha m}{12}\right) \leq \exp\left(-\frac{\alpha m}{24}\right),$$

where the last inequality holds due to the submatrix size assumption (2.9) and our choice of ϵ in (2.11).

Returning to the estimate (2.10), we can now conclude that with probability

$$1 - 2\exp(-cm) - \exp(-\alpha m/24) \geq 1 - 3\exp(-c_2 \alpha m),$$

for all T with $|T| \geq \alpha m$,

$$\sigma_{\min}(\mathbf{A}_T) \geq \frac{\alpha^{3/2}}{12D}\sqrt{\frac{m}{n}} - \epsilon \cdot (1 + CK^2)\sqrt{\frac{m}{n}} \geq \frac{\alpha^{3/2}}{24D}\sqrt{\frac{m}{n}}$$

due to our choice of ϵ in (2.11). This concludes the proof of Proposition 2.7. □

Remark 2.8. *Note that the bounded density assumption is crucial for Proposition 2.7. For instance, the rows of a normalized Bernoulli matrix violate the hypotheses of Lemma 2.6, and Proposition 2.7 does not apply. Unfortunately this cannot be overcome. Indeed, consider taking* $\mathbf{x}$ *to be the vector* $(1, -1, 0, \ldots, 0)$. *Then* $\langle \mathbf{a}_i, \mathbf{x} \rangle = 0$ *with probability* $1/2$. *So if* $\alpha < 1/2$ *in Proposition 2.7 then* $\mathbf{x}$ *will lie in the kernel of some* $\alpha m \times n$ *submatrix of* $\mathbf{A}$ *with high probability, violating the uniform lower bound on the smallest singular value of the submatrices.*

Auxiliary results – structure of the residual

Recall that $\mathbf{a}_i$ denotes a (normalized) row of the matrix $\mathbf{A}$. We recall the notations for the statistics of the corrupted and uncorrupted residuals; we denote the q-quantile of the corrupted residual as $Q_q(\mathbf{x})$, and the q-quantile of the uncorrupted residual as $\tilde{Q}_q(\mathbf{x})$. We additionally recall that the empirical mean of the entries of $\mathbf{Ax}$ is denoted $M(\mathbf{x})$.

The key observation is that for all uncorrupted indices i we have

$$\langle \mathbf{x}_k - \mathbf{x}^*, \mathbf{a}_i \rangle = \langle \mathbf{x}_k, \mathbf{a}_i \rangle - \langle \mathbf{x}^*, \mathbf{a}_i \rangle = \langle \mathbf{x}_k, \mathbf{a}_i \rangle - b_i.$$

Each of $\mathbf{x}_k, \mathbf{a}_i$, and b_i is available at runtime (unlike the exact solution $\mathbf{x}^*$), so this quantity may be computed directly. Then, due to the robustness to noise of the order statistics, we can use the quantiles of the corrupted residual, $Q_q(\mathbf{x}_k)$, to estimate quantiles of the uncorrupted residual, $\tilde{Q}_q(\mathbf{x}_k)$.

In particular, the following straightforward implication of the definition of quantiles is used in the proof of QuantileSGD convergence. We omit the proof.

Lemma 2.9. *With at most a β fraction of samples corrupted by an adversary, we have*

$$\tilde{Q}_{q-\beta}(\mathbf{x}_k) \leq Q_q(\mathbf{x}_k) \leq \tilde{Q}_{q+\beta}(\mathbf{x}_k).$$

We will estimate empirical uncorrupted quantiles $\tilde{Q}_q(\mathbf{x})$ instead of $Q_q(\mathbf{x})$ first. The rest of this section consists of two parts: upper bounds for $\tilde{Q}_q(\mathbf{x})$, and lower bounds for $\tilde{Q}_q(\mathbf{x})$. As in the previous subsection, the main challenge is to get uniform high-probability estimates over the unit sphere.

Concentration of $M(\mathbf{x})$ and upper bound for empirical quantiles

The next lemma shows that any fairly large collection of rows is reasonably incoherent. We will need this result in order to handle situations in which the locations of corruptions are chosen adversarially.

Lemma 2.10. *Let random matrix $\mathbf{A} \in \mathbb{R}^{m \times n}$ satisfy Assumption 1. With probability at least $1 - 2\exp(-cm)$ we have that for all unit vectors $\mathbf{x} \in \mathbb{R}^n$ and every $T \subseteq [m]$,*

$$\sum_{i \in T} |\langle \mathbf{x}, \mathbf{a}_i \rangle| \leq C_K \sqrt{\frac{m|T|}{n}}.$$

Proof. Consider a vector $\mathbf{s} = (s_i) \in \{-1, 0, 1\}^m$ defined by

$$s_i = \begin{cases} \text{sign}(\langle \mathbf{x}, \mathbf{a}_i \rangle), & \text{if } i \in T \\ 0, & \text{otherwise,} \end{cases}$$

for $i \in [m]$. Note that $\|\mathbf{s}\| \leq \sqrt{|T|}$.

The left hand side of the desired inequality can be written as

$$\sum_{i \in T} |\langle \mathbf{x}, \mathbf{a}_i \rangle| = \sum_{i=1}^{m} \langle \mathbf{x}, s_i \mathbf{a}_i \rangle = \left\langle \mathbf{x}, \sum_{i \in [m]} s_i \mathbf{a}_i \right\rangle \leq \left\| \sum_{i \in [m]} s_i \mathbf{a}_i \right\| = \left\| \mathbf{A}^\top \mathbf{s} \right\|.$$

Now the last norm can be estimated using the bound from Theorem 2.4 (since the $\sqrt{n}$-rescaled rows of $\mathbf{A}$ are isotropic and bounded) to get

$$\left\|\mathbf{A}^\top \mathbf{s}\right\| \leq \left\|\mathbf{A}^\top\right\| \|\mathbf{s}\| = \|\mathbf{A}\| \|\mathbf{s}\| \leq C_K \sqrt{\frac{m|T|}{n}}.$$

This conclude the proof of Lemma 2.10. □

The next corollary allows us to upper bound the quantiles computed by QuantileRK. We assume that αm “bad” indices from the next lemma are those that will be excluded by the quantile statistic.

Corollary 2.11. *Let $\alpha \in (0,1]$, let random matrix $\mathbf{A} \in \mathbb{R}^{m\times n}$ satisfy Assumption 1, and suppose that $\mathbf{A}\mathbf{x} = \mathbf{b}$ is corrupted by at most βm corruptions. Let $\mathbf{x}^* \in \mathbb{R}^n$ be the solution of the uncorrupted system. Assuming that $m \geq n$, there exists a constant $C_K > 0$ so that with probability at least $1 - 2\exp(-cm)$, for every $\mathbf{x} \in \mathbb{R}^n$ the bound*

$$|\langle \mathbf{a}_i, \mathbf{x}\rangle - b_i| \leq \frac{C_K}{\alpha\sqrt{n}} \|\mathbf{x} - \mathbf{x}^*\| \tag{2.12}$$

holds for all but at most $(\alpha + \beta)m$ indices i.

Proof. Note that by Lemma 2.10, with probability $1 - 2\exp(-cm)$, for all unit vectors $\mathbf{x}$, the set $I := \{i : |\langle \mathbf{x}, \mathbf{a}_i\rangle| \geq C_K/\alpha\sqrt{n}\}$ has cardinality at most αm. Indeed, this follows as we can lower bound the sum estimate in Lemma 2.10 as

$$\frac{|I|C_K}{\alpha\sqrt{n}} \leq \sum_{i\in T} |\langle \mathbf{x}, \mathbf{a}_i\rangle| \leq C_K \frac{m}{\sqrt{n}}.$$

For a unit vector $u = (\mathbf{x} - \mathbf{x}^*)/\|\mathbf{x} - \mathbf{x}^*\|$, this implies that

$$|\langle \mathbf{a}_i, \mathbf{x}\rangle - b_i| = |\langle \mathbf{a}_i, \mathbf{x} - \mathbf{x}^*\rangle| \leq \frac{C_K}{\alpha\sqrt{n}} \|\mathbf{x} - \mathbf{x}^*\|.$$

holds for all but αm uncorrupted indices $i \in [m]$. Thus, (2.12) holds for all but $(\alpha + \beta)m$ indices in total. □

Note that Lemma 2.10 establishes the high-probability upper estimate for the quantity $M(x)$ of the order $n^{-1/2}$ under the same Assumption 1 for the model. A complementary lower bound of the same order, used in the analysis of QuantileSGD, requires more sophisticated concentration of measure techniques employed in the proposition below.

Proposition 2.12. *Let* $\alpha \in (0,1]$ *and let random matrix* $\mathbf{A} \in \mathbb{R}^{m\times n}$ *with* $m \geq C_K n$ *satisfy Assumption 1. Then, with probability at least* $1 - 2\exp(-cm)$*, for every* $\mathbf{x} \in S^{n-1}$*, we have*

$$M(\mathbf{u}) := \frac{1}{m}\sum_{i=1}^{m} |\langle \mathbf{a}_i, \mathbf{u}\rangle| \geq \frac{c_K}{\sqrt{n}}.$$

Proof. First, we show that the expectation of $M(\mathbf{u})$ is lower bounded by $cn^{-1/2}$ with some positive constant c. To do that, note that $\mathbb{E}M(\mathbf{u}) = \mathbb{E}\,|X_n|$, where $X_n := \langle \mathbf{a}_1, \mathbf{u}\rangle$, and $\mathbf{u}$ is uniform over S^{n-1}. The expectation is taken over $\mathbf{u}$, and is independent of the $\mathbf{a}_i$'s by symmetry of $\mathbf{u}$. Then, by The Projective Central Limit Theorem (see for instance Remark 3.4.8 in [Ver18]), $\sqrt{n}X_n$ converges in distribution to a standard normal as $n \to \infty$. Moreover the random variables $\sqrt{n}X_n$ are uniformly integrable and so $\mathbb{E}(|\sqrt{n}X_n|) \to \mu$ where $\mu \approx 0.78$ is the mean of a standard half-normal random variable.[3] In particular $\mathbb{E}(|\sqrt{n}X_n|)$ is bounded below by a constant uniformly in n.

Then, with probability one $M(\mathbf{u})$ is bounded below by its expectation $cn^{-1/2}$ for some $\mathbf{u} \in S^{n-1}$.

Now, we will use a chaining argument to show that the averages $M(\mathbf{u})$ are concentrated uniformly over the sphere. For $\mathbf{u}, \mathbf{v} \in S^{n-1}$, we have

$$|M(\mathbf{u}) - M(\mathbf{v})| \leq \frac{1}{m}\sum_{i=1}^{m} |\langle \mathbf{a}_i, \mathbf{u} - \mathbf{v}\rangle|\,.$$

The terms in this sum are independent sub-Gaussian random variables with sub-Gaussian norm no

[3] i.e., the absolute value of a standard normal random variable

larger than $K \|\mathbf{u}-\mathbf{v}\| / \sqrt{n}$. Therefore by Lemma 2.5,

$$\|M(\mathbf{u}) - M(\mathbf{v})\|_{\psi_2} \leq \frac{C \cdot K \|\mathbf{u}-\mathbf{v}\|}{\sqrt{m}\sqrt{n}}.$$

By the tail bound version of Dudley's inequality ([Ver18, Theorem 8.1.6]) and the bound $(3/\epsilon)^n$ for the ϵ-covering number of the unit sphere, we then have with probability at least $1-2\exp(-t^2 m)$

$$\sup_{\mathbf{u},\mathbf{v}\in S^{n-1}} |M(\mathbf{u}) - M(\mathbf{v})| \leq \frac{C_1 K}{\sqrt{m}\sqrt{n}} \left(\sqrt{n} + \text{diam}\left(S^{n-1}\right) t\sqrt{m}\right) = K\left(\frac{C_1}{\sqrt{m}} + \frac{2C_1 t}{\sqrt{n}}\right). \quad (2.13)$$

Thus, for all $\mathbf{u} \in S^{n-1}$,

$$M(\mathbf{u}) \geq \frac{c}{\sqrt{n}} - K\left(\frac{c_1}{\sqrt{m}} + \frac{c_2 t}{\sqrt{n}}\right)$$

with probability at least $1 - 2\exp(-t^2 m)$. Provided that $m \geq C_K n$ and t is small enough, this bound reduces to the claim of Proposition 2.12. □

As a result of Lemma 2.10 and Proposition 2.12, we obtain the lower and upper uniform high-probability bounds on $M(\mathbf{u})$ of the form $c_1 n^{-1/2} \leq M(\mathbf{x}) \leq C_2 n^{-1/2}$ for some constants $c_1, C_2 > 0$. This is enough for our analysis of QuantileRK and QuantileSGD methods. However, in the next remark we discuss that these constants can be sharpened essentially for free.

Remark 2.13. *Note that the argument of Proposition 2.12 leads to more refined upper bound on* $M(\mathbf{u})$. *Namely, note that*

$$(\mathbb{E}M(\mathbf{u}))^2 = (\mathbb{E}\,|\langle \mathbf{u}, \mathbf{a}_1\rangle|)^2 \leq \mathbb{E}\,|\langle \mathbf{u}, \mathbf{a}_1\rangle|^2 = n^{-1}.$$

So, for some $\mathbf{u}$*,* $M(\mathbf{u})$ *is at most its expectation and hence* $M(\mathbf{u}) \leq n^{-1/2}$ *for some* $\mathbf{u} \in S^{n-1}$*. Then, from* (2.13)*, for* $t \geq 0$ *and with probability* $1 - 2\exp(-t^2 m)$*, for every* $\mathbf{u} \in S^{n-1}$ *we have* *the bound*

$$M(\mathbf{u}) \leq \frac{1}{\sqrt{n}} + K\left(\frac{c_1}{\sqrt{m}} + \frac{c_2 t}{\sqrt{n}}\right) \leq \frac{1+\epsilon}{\sqrt{n}} \tag{2.14}$$

for any small $\epsilon > 0$ if the matrix A is tall enough and parameter t is small enough.

If in addition, one assumes that n is sufficiently large (greater than come constant C_ϵ) so that $\mathbb{E}(|\sqrt{n}M(\mathbf{u})|)$ is near the mean of half-normal random variable μ, then one can have

$$M(\mathbf{x}) \geq \frac{\mu - \epsilon}{\sqrt{n}}$$

with probability at least $1 - 2\exp(-c_{K,\epsilon}m)$ provided that $m \geq C_{K,\epsilon}n$. The latter bound allows us to extend the guarantees for the QuantileSGD algorithm for a wider range of quantiles, under additional restrictions on the model (we will not carry out this analysis in detail, however see Remark 2.20).

Lower bound for empirical quantiles

We also use the following lower-bound variant of the above result when analyzing QuantileSGD.

Lemma 2.14. *Let $\mathbf{A} \in \mathbb{R}^{m\times n}$ be a random matrix satisfying Assumption 2 and let $\mathbf{x}^*$ and $\mathbf{b}$ be such that $\mathbf{A}\mathbf{x}^* = \mathbf{b}$ (that is, the linear system is consistent). If $m \geq \frac{C}{q} n \log\left(\frac{cnD}{q}\right)$ then*

$$\Pr\left\{Q_q(\mathbf{x}) \geq \frac{cq}{D\sqrt{n}} \|\mathbf{x} - \mathbf{x}^*\| \textit{ for all } \mathbf{x} \in \mathbb{R}^n\right\} \geq 1 - \exp(-cm).$$

Proof. We assume without loss of generality that $\mathbf{x}^* = \mathbf{0}$ and $\mathbf{b} = \mathbf{0}$. The general result follows in the same way as in Corollary 2.11 above. By scaling, it suffices to prove the result for $\mathbf{x} \in S^{n-1}$.

First consider a fixed $\mathbf{x}$ in S^{n-1}. By Lemma 2.6, we can choose $c_q = cq/D$ so that,

$$\Pr\left(|\langle \mathbf{x}, \mathbf{a}_i\rangle| \leq \frac{2c_q}{\sqrt{n}}\right) \leq \frac{q}{2} \tag{2.15}$$

for all i. By a Chernoff bound, $Q_q(\mathbf{x}) \geq 2c_q/\sqrt{n}$ with probability at least $1 - \exp(-qm/6)$.

Let $\mathcal{N}$ be a $c_q/\sqrt{n}$-net of S^{n-1} which we can take to have size

$$|\mathcal{N}| = \left(\frac{3}{c_q/\sqrt{n}}\right)^n = \exp(n \log(3\sqrt{n}/c_q)).$$

By a union bound, there are constants so that if

$$m \geq \frac{C}{q} n \log\left(\frac{cnD}{q}\right),$$

then the quantile bound (2.15) holds for all x in $\mathcal{N}$ with probability at least $1 - \exp(-cm)$.

In order to upgrade our bound on $\mathcal{N}$ to all of S^{n-1}, it remains to show that $Q_q(\mathbf{x})$ is stable under small perturbations of $\mathbf{x}$.

Suppose that $\mathbf{x}$ and $\mathbf{y}$ in $\mathbb{R}^n$ are arbitrary. Then for all i, we have the bound

$$|\langle \mathbf{x}, \mathbf{a}_i\rangle| - |\langle \mathbf{y}, \mathbf{a}_i\rangle| \leq |\langle \mathbf{x}, \mathbf{a}_i\rangle - \langle \mathbf{y}, \mathbf{a}_i\rangle| = |\langle \mathbf{x} - \mathbf{y}, \mathbf{a}_i\rangle| \leq \|\mathbf{x} - \mathbf{y}\|.$$

Therefore

$$|\langle \mathbf{x}, \mathbf{a}_i\rangle| - \|\mathbf{x} - \mathbf{y}\| \leq |\langle \mathbf{y}, \mathbf{a}_i\rangle| \leq |\langle \mathbf{x}, \mathbf{a}_i\rangle| + \|\mathbf{x} - \mathbf{y}\|.$$

By taking the q-quantiles over i and using monotonicity of quantiles, it follows that

$$|Q_q(\mathbf{x}) - Q_q(\mathbf{y})| \leq \|\mathbf{x} - \mathbf{y}\| \tag{2.16}$$

for all $\mathbf{x}, \mathbf{y}$.

Each point in S^{n-1} is within $c_q n^{-1/2}$ of some point in $\mathcal{N}$. Lemma 2.14 follows by combining (2.16) with our bound on $Q_q(\mathbf{x})$ over $\mathcal{N}$. □

Remark 2.15. *We require the aspect ratio of* $\mathbf{A}$ *to be at least order* $\log(n)$. *It is plausible that Lemma 2.14 can be improved to hold for constant aspect ratios as was the case of the upper bound in Corollary 2.11. We will not attempt to do so, and as a result we require QuantileSGD to have a* *slightly stronger condition on the aspect ratio of* $\mathbf{A}$ *than QuantileRK.*

2.4.2 Analysis of the QuantileRK method

In this section we provide a proof that the QuantileRK method converges.

Roadmap. The proof will proceed as follows. We condition on the sampling of a row that will be accepted by the QuantileRK iteration; recall a row is acceptable in a given iteration if the entry of the residual associated to this row is less than or equal to $Q_q(\mathbf{x}_{j-1}, \{i_l : l \in [t]\})$. We then show that the uncorrupted rows help substantially, while the corrupted rows do not overly affect the convergence. Conditioned on the current row being uncorrupted, we argue that an iteration of the QuantileRK method brings us closer in expectation to $\mathbf{x}^*$. To accomplish this, we show that the restriction of $\mathbf{A}$ to the acceptable uncorrupted rows is well-conditioned via Lemma 2.10. In that case, the current iteration of QuantileRK is equivalent to an iteration of the standard RK method on the restricted matrix. This allows us to apply a known per-iteration guarantee for RK.

To argue that corrupted rows do not significantly harm convergence, we consider a subset $J \subset [m]$, of row indices with $|J|/m \geq c$, and with J containing all corrupted indices as a subset. By making J sufficiently large, we ensure that the subset of the rows of $\mathbf{A}$ indexed by J inherits incoherence properties from the full matrix (uniformly over all such subsets, due to Proposition 2.7). Incoherence will ensure that the average projection of $\mathbf{x}$ onto a corrupted hyperplane moves the point in a direction nearly orthogonal to $\mathbf{x} - \mathbf{x}^*$. The length of such a step is bounded by $Cn^{-1/2}$ by Corollary 2.11, so a "bad" step is unlikely to move $\mathbf{x}$ much further from $\mathbf{x}^*$. In particular, a constant number of "good" steps will suffice to "cancel out" a bad step. If the fraction of bad rows is sufficiently small, then the QuantileRK method will enjoy linear convergence to $\mathbf{x}^*$.

Proof of Theorem 2.1. We will start by introducing some useful notation. Recall that each instance of an absolute constant may refer to a different constant value; however, we track the dependence of q on β explicitly.

Let $\mathcal{E}_{\text{Accept}}(k)$ denote the event that we sample an acceptable row at the k-th step of the method;

that is, if the if-statement in line 6 of the QuantileRK Method 1 evaluates to true for that row. Recall that an i-th row of $\mathbf{A}$ is acceptable at iteration k if $|\langle \mathbf{x}_k, \mathbf{a}_i \rangle - b_i| \leq Q_q(\mathbf{x}_k)$, where Q_q is defined as in (2.4). Clearly, $\Pr(\mathcal{E}_{\text{Accept}}(k)) = \lfloor qm \rfloor / m$ for any integer $k \geq 1$.

Further, we will consider three subsets of indices denoted as J, I_1 and I_2. Let J denote a collection of indices of size[4] $2\beta m$ which contains all corrupted indices and at least βm acceptable indices. We assume that $\beta < q$ so there exists that many acceptable indices (as there are exactly $\lfloor qm \rfloor$ acceptable indices total). Then, all acceptable indices are split into two types: those inside the set J (we denote them I_1, by construction, $|I_1| \geq \beta m$) and those outside of J (we denote them I_2). Finally, let $\mathcal{E}_L^k$ denote the event that k-th iteration of the QuantileRK method samples an index from an index subset $L \subset [n]$.

We first observe that

$$\mathbb{E}_k(\|\mathbf{e}_{k+1}\|^2) = (\lfloor qm \rfloor / m) \mathbb{E}_k(\|\mathbf{e}_{k+1}\|^2 \, | \mathcal{E}_{\text{Accept}}(k+1)) + (1 - \lfloor qm \rfloor / m) \|\mathbf{e}_k\|^2 , \qquad (2.17)$$

since QuantileRK(q) does not update x_k if a sampled row index was not acceptable.

Conditioned on choosing an acceptable row, we either pick an index from I_1 or from I_2, and the conditional probability p_J to choose an index in I_1 satisfies $p_J \leq 2\beta m / qm = 2\beta / q$ (the upper bound refers to the case when $I_1 = J$).

Now, given $\mathcal{E}_{I_2}^{k+1}$, the iterate $\mathbf{x}_{k+1}$ is obtained by applying an iteration of the Standard RK method for the matrix $\mathbf{A}_{I_2}$. Note that I_2 has size at least $(q - 2\beta)m - 1$, since at least $\lfloor qm \rfloor$ indices are acceptable, and at most $2\beta m$ of these are contained in I_1. Next we apply Proposition 2.7 with $\alpha = q - 2\beta - \frac{1}{m} > 0$. As long as β is at most a constant factor of q (e.g., $\beta \leq q/4$), then we have $\alpha \geq cq$. The proposition then gives $\|\mathbf{A}_{I_2}^{-1}\|_2 \leq C_{\beta,D}\sqrt{n/m}$ with probability $1 - 3\exp(-c_q m)$ provided that

$$\frac{m}{n} \geq C_{q,D} := C \frac{1}{\alpha} \log\left(\frac{DK}{\alpha}\right).$$

[4] We assume without loss of generality that βm is an integer. If this is not the case, consider β' such that $\beta' m = \lceil \beta m \rceil$ instead of β throughout the proof.

Since all the rows of $\mathbf{A}$ are normalized to have unit norm, we also know that $\|\mathbf{A}\|_F = \sqrt{m}$. Therefore, with high probability, we may bound the condition number of $\mathbf{A}_{I_2}$ as

$$\kappa(\mathbf{A}_{I_2}) \leq \|\mathbf{A}\|_F \|\mathbf{A}_{I_2}^{-1}\|_2 \leq \sqrt{m} C_{q,D} \sqrt{n/m} = C_{q,D}\sqrt{n}. \tag{2.18}$$

Note that Proposition 2.7 gives a uniform lower bound for the condition number for all index subsets of size at least αm. So in each iteration of the method, $\mathbf{A}_{I_2}$ will have a good condition number upper bounded by (2.18) with probability at least $1 - 3\exp(-c_q m)$. In particular, $\mathbf{A}_{I_2}$ has full rank since $\|\mathbf{A}_{I_2}^{-1}\|$ is finite. Then, by the analysis of the Standard RK method [SV09] given in (2.3), we have

$$\mathbb{E}_k(\|\mathbf{e}_{k+1}\|^2 \,|\mathcal{E}_{I_2}^{k+1}) \leq \left(1 - \frac{c_1}{n}\right) \|\mathbf{e}_k\|^2 .$$

Now, we consider two cases. In the corruptionless case when $\beta = 0$, we have that the set I_1 is empty and $p_J = 0$ by definition. So,

$$\mathbb{E}_k(\|\mathbf{e}_{k+1}\|^2) \leq q\left(1 - \frac{c_1}{n}\right) \|\mathbf{e}_k\|^2 + (1-q)\|\mathbf{e}_k\|^2 \leq \left(1 - \frac{qc_1}{n}\right) \|\mathbf{e}_k\|^2 . \tag{2.19}$$

In the other case, when $\beta > 0$, we need to consider the second possibility, if the next index was coming from I_1.

Conditioned on taking an acceptable row, we can choose h_i with $|h_i| \leq Q_q(\mathbf{x}_k)$, so that

$$\begin{aligned}
\mathbb{E}_k(\|\mathbf{e}_{k+1}\|^2 \,|\mathcal{E}_{I_1}^{k+1}) &= \mathbb{E}_k(\|\mathbf{e}_k - h_i\mathbf{a}_i\|^2 \,|\mathcal{E}_{I_1}^{k+1}) \\
&= \|\mathbf{e}_k\|^2 + h_i^2 - 2\mathbb{E}_k\left(h_i \langle \mathbf{e}_k, \mathbf{a}_i\rangle \,|\mathcal{E}_{I_1}^{k+1}\right) \\
&\leq \|\mathbf{e}_k\|^2 + Q_q(\mathbf{x}_k)^2 + 2Q_q(\mathbf{x}_k)\mathbb{E}_k(|\langle \mathbf{e}_k, \mathbf{a}_i\rangle| \,|i \sim \mathrm{Unif}(I_1)).
\end{aligned}$$

We would like to bound these last two terms. By Corollary 2.11, for $\alpha \leq 1 - q - \beta$,

$$\Pr\left(Q_q(\mathbf{x}_k) \leq \frac{C_K \|\mathbf{e}_k\|}{\alpha\sqrt{n}}\right) \geq 1 - 2\exp(-m).$$

As long as $q+\beta$ is bounded away from one, as in the case if the constants in the statement of the theorem are chosen appropriately small, this yields a bound of the form

$$\Pr\left(Q_q(\mathbf{x}_k) \leq \frac{C_K \|\mathbf{e}_k\|}{\sqrt{n}}\right) \geq 1 - 2\exp(-m).$$

Also, we apply Lemma 2.10 to the set I_1 (recall that $|I_1| \geq \beta m$) to get that with probability $1 - 2\exp(-cm)$,

$$\mathbb{E}_k\left(|\langle \mathbf{e}_k, \mathbf{a}_i\rangle| \; | i \sim \mathrm{Unif}(J)\right) = \frac{1}{|I_1|}\sum_{i \in I_1} |\langle \mathbf{e}_k, \mathbf{a}_i\rangle| \leq C\|\mathbf{e}_k\|\sqrt{\frac{m}{n|I_1|}} \leq \frac{C\|\mathbf{e}_k\|}{\sqrt{\beta n}}.$$

Thus,

$$\mathbb{E}_k(\|\mathbf{e}_{k+1}\|^2 \,|\mathcal{E}_{I_1}^{k+1}) \leq \left(1 + \frac{\sqrt{\beta}c_2 + c_3}{\sqrt{\beta n}}\right)\|\mathbf{e}_k\|^2. \tag{2.20}$$

So, in this case the norm of the error could increase, but not too much (as we will see below).

So, by the total expectation theorem, we have

$$\begin{aligned}
\mathbb{E}_k(\|\mathbf{e}_{k+1}\|^2 \,|\mathcal{E}_{\mathrm{Accept}}(k+1)) &= p_J \mathbb{E}_k(\|\mathbf{e}_{k+1}\|^2 \,|\mathcal{E}_{I_1}^{k+1}) + (1-p_J)\mathbb{E}_k(\|\mathbf{e}_{k+1}\|^2 \,|\mathcal{E}_{I_2}^{k+1}) \\
&\leq \left[p_J\left(1 + \frac{\sqrt{\beta}c_2 + c_3}{\sqrt{\beta n}}\right) + (1-p_J)(1 - \frac{c_1}{n})\right]\|\mathbf{e}_k\|^2 \\
&= \left[1 - \frac{c_1}{n} + p_J\left(\frac{(c_1+c_2)\sqrt{\beta} + c_3}{\sqrt{\beta n}}\right)\right]\|\mathbf{e}_k\|^2 \\
&\leq \left[1 - \frac{c_1}{n} + \frac{2\beta}{q}\cdot\frac{c_1+c_2+c_3}{\sqrt{\beta n}}\right]\|\mathbf{e}_k\|^2 \\
&\leq \left[1 - \frac{0.5c_1}{n}\right]\|\mathbf{e}_k\|^2,
\end{aligned}$$

where the last step holds if β a sufficiently small constant (we need $\sqrt{\beta} \leq cq =: C_q$). Finally, from (2.17) we obtain the per-iteration guarantee

$$\mathbb{E}_k(\|\mathbf{e}_{k+1}\|^2) \leq (\lfloor qm)\rfloor/m\left(1 - \frac{0.5c_1}{n}\right)\|\mathbf{e}_k\|^2 + (1 - \lfloor qm\rfloor/m)\|\mathbf{e}_k\|^2 \leq \left(1 - \frac{cq}{n}\right)\|\mathbf{e}_k\|^2. \tag{2.21}$$

Theorem 2.1 now follows from (2.19) or (2.21) by induction. □

Remark 2.16 (Condition on β)**.** *We need the fraction of corruptions β to be sufficiently small. Specifically, our proof of Theorem 2.1 requires $\sqrt{\beta} < cq$, where c is some small positive constant. Intuitively, this is required since the quantile bound (admissibility) is the only way to bound potential loss if the step is made using one of the corrupted equations (as we do not impose any restrictions on the size of corruptions). Moreover, the expected loss of progress, given the projection on the admissible corrupted equation, must be so small that it is compensated by the expected exponential convergence rate, given that one of the equations from the uncorrupted part was selected.*

Remark 2.17. *Although the bounded density assumption is crucial for Proposition 2.7 to hold (see Remark 2.8), one should not expect the failure of Proposition 2.7 to result in the QuantileRK method not converging. In the Bernoulli case, the per-iteration guarantee given likely no longer holds, however one expects it to fail for only a very small set of vectors $\mathbf{x}_k$. Provided that the QuantileRK method does not attract iterates to this set of bad vectors, one should still expect convergence from a randomly chosen $\mathbf{x}_0$. We leave such an analysis to future work. (However we empirically demonstrate convergence in Figure 2.4 (a).)*

2.4.3 Analysis of QuantileSGD method

In this section, we provide a proof that the QuantileSGD method converges. To do so, we first introduce an optimal SGD method in Section 2.4.3 and then prove that QuantileSGD approximates this optimal method in Section 2.4.3. We then give an improved analysis in the streaming setting in Section 2.4.3.

OptSGD

As a first step towards the analysis of the quantile-based SGD method, we introduce the OptSGD method taking the steps of the optimal size towards the solution.

Note that SGD iterates can be written in the form

$$\mathbf{x}_{k+1} = \mathbf{x}_k - \eta_k s_i(\mathbf{x}_k)\mathbf{a}_i, \qquad \text{where} \quad s_i(\mathbf{x}_k) := \text{sign}(\langle \mathbf{a}_i, \mathbf{x}_k \rangle - b_i); \tag{2.22}$$

that is, the vector $s_i(\mathbf{x}_k)\mathbf{a}_i$ is directed from the hyperplane defined by the i^{th} equation towards the half space that $\mathbf{x}_k$ lies on. We assume that SGD samples rows uniformly, so $i \sim \text{Unif}([m])$. The constant $\eta_k > 0$ defines the length of the step (recall that $\|\mathbf{a}_i\|_2 = 1$)). OptSGD chooses the step size η_k^* so that the expected distance to the solution $\mathbb{E}\,\|\mathbf{e}_{k+1}\|_2^2 = \mathbb{E}\,\|\mathbf{x}_{k+1} - \mathbf{x}^*\|_2^2$ is minimized.

Namely, we have

$$\begin{aligned}
\mathbb{E}\left(\|\mathbf{e}_{k+1}\|_2^2\right) &= \mathbb{E}\left(\|\mathbf{e}_k - s_i(\mathbf{x}_k)\eta_k\mathbf{a}_i\|_2^2\right) \\
&= \mathbb{E}\left(\|\mathbf{e}_k\|_2^2 - 2s_i(\mathbf{x}_k)\left\langle \mathbf{e}_k, \mathbf{a}_i\right\rangle \eta_k + s_i(\mathbf{x}_k)^2 \|\mathbf{a}_i\|_2^2\,\eta_k^2\right) \\
&= \|\mathbf{e}_k\|_2^2 - 2\mathbb{E}\left(s_i(\mathbf{x}_k)\left\langle \mathbf{e}_k, \mathbf{a}_i\right\rangle\right)\eta_k + \eta_k^2 \\
&= \left(\eta_k - \mathbb{E}(s_i(\mathbf{x}_k)\left\langle \mathbf{e}_k, \mathbf{a}_i\right\rangle)\right)^2 - \left(\mathbb{E}(s_i(\mathbf{x}_k)\left\langle \mathbf{e}_k, \mathbf{a}_i\right\rangle)\right)^2 + \|\mathbf{e}_k\|_2^2\,,
\end{aligned} \tag{2.23}$$

which is minimized by setting

$$\eta^*(\mathbf{x}_k) = \mathbb{E}\left(s_i(\mathbf{x}_k)\left\langle \mathbf{e}_k, \mathbf{a}_i\right\rangle\right) = \frac{1}{m}\sum_{i=1}^{m} s_i(\mathbf{x}_k)\left\langle \mathbf{e}_k, \mathbf{a}_i\right\rangle. \tag{2.24}$$

Quantile SGD

In the previous section, we derived a theoretically optimal step size for l_1 stochastic gradient descent. The formula for the step size (2.24) relied on $\mathbf{e}_k$ which is unknown during runtime. Actually, since $\langle \mathbf{e}_k, \mathbf{a}_i\rangle = \langle \mathbf{x}_k, \mathbf{a}_i\rangle - \langle \mathbf{x}^*, \mathbf{a}_i\rangle = \langle \mathbf{x}_k, \mathbf{a}_i\rangle - b_i$ for any uncorrupted equation, it is the presence of corruptions that makes $\eta^*(\mathbf{x}_k)$ unavailable at runtime. Here we show that order statistics can be applied to give an approximation to the optimal step size.

First, let us show that $\eta_k^*(\mathbf{x}_k)$ is well-approximated by $M(\mathbf{x}_k - \mathbf{x}^*)$. We notice that the sums defining $\eta_k^*(\mathbf{x}_k)$ and $M(\mathbf{x}_k - \mathbf{x}^*)$ respectively differ only in the terms corresponding to the indices of the corrupted equations. So, given that the fraction of corruptions is small enough, we can efficiently bound this difference.

Proposition 2.18. *Fix any $\delta \in (0,1)$. Let the system be defined by random matrix $\mathbf{A} \in \mathbb{R}^{m\times n}$ satisfying Assumptions 1 and 2 with $m \geq C_K n$, and $\beta = |\text{supp}(\mathbf{b}_C)|/m$ a small enough positive constant. Let $\eta^*(\mathbf{x})$ be optimal step size for SGD method defined as in* (2.24)*. Then, with probability at least $1 - c\exp(-c_K m)$ we have for any $\mathbf{x} \in \mathbb{R}^n$ that*

$$(1-\delta)\eta^*(\mathbf{x}) \leq M(\mathbf{x}-\mathbf{x}^*) \leq (1+\delta)\eta^*(\mathbf{x}). \tag{2.25}$$

Proof. Let S denote the set of indices corresponding to negative terms in the sum (2.24). Note that for all uncorrupted equations i, we have $s_i(\mathbf{x}_k) = \text{sign}(\langle \mathbf{e}_k, \mathbf{a}_i\rangle)$, so the i-th term in $\eta^*(\mathbf{x}_k)$ is non-negative, and $|S| \leq \beta m$. We then have

$$|\eta^*(\mathbf{x}) - M(\mathbf{x}-\mathbf{x}^*)| \leq \frac{2}{m}\sum_{i\in S} |\langle \mathbf{x}-\mathbf{x}^*, \mathbf{a}_i\rangle| .$$

Rescaling to normalize $\mathbf{x}-\mathbf{x}^*$ and applying Lemma 2.10 allows us to further bound

$$|\eta^*(\mathbf{x}) - M(\mathbf{x}-\mathbf{x}^*)| \leq \frac{2|S|}{m} C_K \frac{1}{\sqrt{|S|/m}\sqrt{n}} \|\mathbf{x}-\mathbf{x}^*\| \leq 2C_K \frac{\sqrt{\beta}}{\sqrt{n}} \|\mathbf{x}-\mathbf{x}^*\|$$

uniformly for all $\mathbf{x}$ with probability at least $1 - 2\exp(-cm)$.

Moreover, by Propostion 2.12,

$$M(\mathbf{x}-\mathbf{x}^*) \geq \frac{c}{\sqrt{n}} \|\mathbf{x}-\mathbf{x}^*\|$$

for all $\mathbf{x}$ with probability at least $1 - 2\exp(-c_K m)$. Thus by taking β to be a sufficiently small constant (so that the difference between $\eta^*(\mathbf{x})$ and $M(\mathbf{x}-\mathbf{x}^*)$ is negligible compared to the size of $M(\mathbf{x}-\mathbf{x}^*)$), we conclude the proof of Proposition 2.18. □

Although the empirical mean $M(\mathbf{x}-\mathbf{x}^*)$, as well as η_k^* is not available at runtime, the above proposition allows us to show that in order to obtain a near optimal convergence guarantee, it suffices to approximate η_k^* to within a constant factor.

Proposition 2.19. *Let the system be defined by random matrix $\mathbf{A} \in \mathbb{R}^{m\times n}$ satisfying Assumptions 1 and 2 with $m \geq C_k n$. Suppose we run an SGD method* (2.22) *with the stepsize η_k, satisfying $0 < c_1 \leq \eta_k/\eta^*(\mathbf{x}_k) \leq c_2 < 2$ at each iteration $k = 1, 2, 3, \ldots$, where $\eta^*(\mathbf{x}_k)$ is an optimal step size given by* (2.24)*. Then, for any $\beta = |\text{supp}(\mathbf{b}_C)|/m \in (0,1)$, there exists a constant $c = c(c_1, c_2) > 0$ such that*

$$\mathbb{E}(\|\mathbf{e}_{k+1}\|_2^2) \leq \left(1 - c\left(\frac{\eta^*(\mathbf{x}_k)}{\|\mathbf{e}_k\|_2}\right)^2\right)\|\mathbf{e}_k\|_2^2 . \tag{2.26}$$

Moreover, if the fraction of corrupted equations β is small enough, then with probability at least $1 - c\exp(-c_K m)$, $\mathbf{A}$ is sampled such that the rate of convergence is linear, namely, there exists a constant $C = C(c_1, c_2) > 0$ such that

$$\mathbb{E}(\|\mathbf{e}_{k+1}\|_2^2) \leq \left(1 - \frac{C}{n}\right)\|\mathbf{e}_k\|^2 . \tag{2.27}$$

Proof. Throughout the proof, we adopt a shorthand notation $\eta_k^* = \eta^*(\mathbf{x}_k)$.

Indeed, by the condition on η_k we have that

$$|\eta_k - \eta_k^*| \leq \eta_k^* \max\{c_2 - 1, c_1 - 1\}$$

and $c = 1 - (\max\{c_2 - 1, c_1 - 1\})^2 > 0$. So, by equation (2.23) and the definition of η_k^* (in (2.24)), we have that

$$\mathbb{E}\left(\|\mathbf{e}_{k+1}\|_2^2\right) = (\eta_k - \eta_k^*)^2 - (\eta_k^*)^2 + \|\mathbf{e}_k\|_2^2 \leq \|\mathbf{e}_k\|_2^2 - c(\eta_k^*)^2,$$

and so

$$\mathbb{E}(\|\mathbf{e}_{k+1}\|_2^2) \leq \left(1 - c\left(\frac{\eta_k^*}{\|\mathbf{e}_k\|_2}\right)^2\right)\|\mathbf{e}_k\|_2^2 .$$

To show that the convergence rate is linear, note that by applying Proposition 2.18 with $\delta = 1/3$, and Proposition 2.12, we have the bound

$$\eta^*(\mathbf{x}_k) \geq \frac{3}{4} M(\mathbf{x}_k - \mathbf{x}^*) \gtrsim \frac{1}{\sqrt{n}} \left\| \mathbf{x}_k - \mathbf{x}^* \right\| .$$

This concludes the proof of Proposition 2.19. □

Roadmap. We are now set to give a proof of Theorem 2.2. The general plan is as follows: we know that quantiles of the residual $Q_q(\mathbf{x}_k)$ are well-approximated by the empirical uncorrupted quantiles $\tilde{Q}_q(\mathbf{x}_k)$ (Lemma 2.9), then we show that empirical uncorrupted quantiles concentrate near the empirical mean $M(\mathbf{x} - \mathbf{x}^*)$, which is in turn close enough to the optimal step size $\eta^*(\mathbf{x})$ (Proposition 2.18). Finally, we invoke Proposition 2.19 to conclude the linear convergence rate of the QuantileSGD(q) method.

Proof of Theorem 2.2. We upper bound the q-quantile of the corrupted residual,

$$Q_{q-\beta}(\mathbf{x}_k) \leq \tilde{Q}_q(\mathbf{x}_k) \leq \frac{1}{1-q} M(\mathbf{x}_k - \mathbf{x}^*) \leq \frac{1+\delta}{1-q} \eta^*(\mathbf{x}_k) < 2\eta^*(\mathbf{x}_k), \tag{2.28}$$

where the first inequality follows from Lemma 2.9, the second from Markov's inequality, the third from Proposition 2.18 with probability at least $1 - 2\exp(-cm)$, and the fourth by choosing $\delta \in (0, 1-2q)$.

For the lower bound, we have that $Q_{q-\beta}(\mathbf{x}_k) \geq \tilde{Q}_{q-2\beta}(\mathbf{x}_k)$ by Lemma 2.9. Then,

$$\tilde{Q}_{q-2\beta}(\mathbf{x}) \geq \frac{c_1}{\sqrt{n}} \left\| \mathbf{x} - \mathbf{x}^* \right\|$$

for all $\mathbf{x}$ with probability at least $1 - \exp(-cm)$ by Lemma 2.14. Now,

$$M(\mathbf{x}) \leq \frac{c_2}{\sqrt{n}} \left\| \mathbf{x} - \mathbf{x}^* \right\| ,$$

so we may upper bound $\eta^*(\mathbf{x})$,

$$\eta^*(\mathbf{x}) \leq \frac{M(\mathbf{x})}{1-\delta} \leq \frac{c_2}{(1-\delta)\sqrt{n}} \|\mathbf{x} - \mathbf{x}^*\| \leq \frac{c_2}{(1-\delta)c_1} \tilde{Q}_{q-2\beta}(\mathbf{x}_k) \leq \frac{1}{c} Q_{q-\beta}(\mathbf{x}_k)$$

for some positive constant c.

Combining these upper and lower bounds on $Q_{q-\beta}(\mathbf{x})$ we find that there exists $c > 0$ so that for all $\mathbf{x} \in \mathbb{R}^n$,

$$0 < c < \frac{Q_{q-\beta}(\mathbf{x})}{\eta^*(\mathbf{x})} < \frac{1+\delta}{(1-q)\eta^*(\mathbf{x})} < 2.$$

We have shown that the hypothesis of Proposition 2.19 holds. Theorem 2.2 follows by induction.

□

Remark 2.20. *In some cases, for example, when $\mathbf{a}_i$ are independent vectors sampled uniformly from S^{n-1} we can show that a bigger range of quantiles for an SGD step guarantees exponential convergence of the QuantileSGD method. In particular, using Gaussian concentration instead of Markov's inequality in* (2.28)*, the statement of Theorem 2.2 holds for* QuantileSGD($q-\beta$) *for all* $q \in (0, 0.75)$. *Note that this justifies the optimal values for the quantile q obtained experimentally (see Figure 2.1 b).*

Streaming Setting

In the matrix setting we only prove convergence for a sufficiently small fraction of corruptions β. While one could in principle unwind the constants from the random matrix theorems that we have applied, it would be unlikely to result in new insights. Instead we note that the key complication in the matrix setting was handling "asymmetries" in the matrix $\mathbf{A}$. While the rows were sampled over S^{n-1} in a close-to-uniform way, there was no guarantee that the rows of $\mathbf{A}$ (representing only a sample from this distribution) were uniformly spread over the sphere.

Here we present a more optimized analysis in the streaming setting, which may be viewed as a model for extremely tall matrices where each row is likely to be sampled only once in the course of the method. In particular, it allows us to justify the QuantileSGD method when up to a 0.35 fraction of all equations are corrupted (note that in both Theorem 2.1 and Theorem 2.2 we formally asked for the fraction of corruptions β to be "small enough").

Instead of a matrix let us consider some distribution $\mathcal{D}$ over $\mathbb{R}^n$ and $\beta \in [0, 1]$. On each of many iterations, we receive a pair $(\mathbf{a}_k, b_k)$ (in a non-streaming setting this pair was a row of the matrix and a corresponding entry of the vector $\mathbf{b}$ respectively). The vector $\mathbf{a_k}$ is always sampled from $\mathcal{D}$. With probability $1 - \beta$, b was selected so that $b = \langle \mathbf{a}, \mathbf{x}^* \rangle$, and with probability β, b was chosen arbitrarily, and possibly adversarially. Our goal is to approximate $\mathbf{x}^*$.

For simplicity, we allow ourselves an arbitrary number of samples to estimate the quantiles of the residual $Q_q(\mathbf{x}_k)$, where $\mathbf{a_k}$ from Definition (2.4) are random gaussian vectors and respective b_i are given by the samples.

Theorem 2.21. *In the streaming setting with arbitrary corruptions and Gaussian samples (namely $\mathbf{a}_k$) are standard n-variate Gaussian random vectors), QuantileSGD(q) converges to $\mathbf{x}^*$ with $\beta = 0.35$ as long as the quantile q is chosen sufficiently small.*

Remark 2.22. *The model of the left hand side of the system is different from our earlier convention, in particular, $\mathbf{a}_k$'s do not have exactly unit norm. However this distinction is unimportant since (i) our methods are invariant under rescaling the $\mathbf{a}_k$s and (ii) the one-dimensional projections of the uniform distribution over $\sqrt{n}S^{n-1}$ converge in distribution to a Gaussian as $n \to \infty$.*

Proof. Recall that for QuantileSGD, we chose our step size $\eta_k = Q_q(\mathbf{x}_k)$. In the streaming setting with Gaussian samples, the value of $Q_q(\mathbf{x})$ only depends on $\|\mathbf{x} - \mathbf{x}^*\|$. This follows directly from the definition of $\tilde{Q}_q$ along with rotation-invariance of Gaussian vectors. Furthermore, Q_q respects dilations about $\mathbf{x}^*$ in the sense that

$$Q_q(\mathbf{x}^* + \lambda(\mathbf{x} - \mathbf{x}^*)) = \lambda Q_q(\mathbf{x})$$

for $\lambda \in \mathbb{R}$. Again this is a simple check from the definition of $\tilde{Q}$. The same properties hold for the optimal SGD step size η^* as per (2.24) for the same reasons.

These properties imply that $Q_q(\mathbf{x})/\eta^*(\mathbf{x})$ is constant over $\mathbb{R}^n \setminus \{\mathbf{x}^*\}$. We are going to show that for small q this quantity lies strictly between 0 and 2. In other words

$$0 < c < Q_q(\mathbf{x}_k)/\eta^*(\mathbf{x}_k) < C < 2 \tag{2.29}$$

for all iterates $\mathbf{x}_k$. (Of course this bound holds for all $\mathbf{x} \in \mathbb{R}^n$, but we emphasize that we apply this bound to the iterates.) This will allow us to apply Proposition 2.19 (in the form of (2.26)) to conclude that QuantileSGD(q) converges for q small enough.

The lower bound of (2.29) clearly holds for q positive, since $Q_q(\mathbf{x}_k)/\eta^*(\mathbf{x}_k)$ is nonzero as long as $\mathbf{x}_k \neq \mathbf{x}^*$ (and of course $\eta^*(\mathbf{x}_k) < \infty$).

Also recall that for all uncorrupted equations we have $\eta^*(\mathbf{x}_k) = \mathbb{E}|\langle \mathbf{x}_k - \mathbf{x}^*, \mathbf{a}_k\rangle|$. So, we can lower bound

$$\begin{aligned}\eta^*(\mathbf{x}_k) &\geq (1-\beta)\mathbb{E}(|\langle \mathbf{e}_k, \mathbf{a}_k\rangle|) + \beta\mathbb{E}(-|\langle \mathbf{e}_k, \mathbf{a}_k\rangle|)\\ &= (1-2\beta)\mathbb{E}(|\langle \mathbf{e}_k, \mathbf{a}_k\rangle|)\\ &= (1-2\beta)\sqrt{\frac{2}{\pi}}\,\|\mathbf{e}_k\|\,,\end{aligned}$$

where the last constant is the expectation of a standard half-normal random variable.

By Lemma 2.9, we also have

$$Q_q(\mathbf{x}_k) \leq \tilde{Q}_{q+\beta}(\mathbf{x}_k) = \|\mathbf{e}_k\|\,\Phi_{q+\beta},$$

where Φ_q denotes the q-quantile of the standard half-normal distribution, $|\mathcal{N}(0,1)|$. The upper bound in equation (2.29) is equivalent to the inequality

$$\|\mathbf{e}_k\|\,\Phi_{q+\beta} < C(1-2\beta)\sqrt{\frac{2}{\pi}}\,\|\mathbf{e}_k\|\,, \tag{2.30}$$

where C is allowed to be any constant smaller than 2 (e.g 1.99). This inequality is true for small positive q as long as

$$\tilde{Q}_\beta(|\mathcal{N}(0,1)|) < \sqrt{\frac{8}{\pi}}(1-2\beta).$$

One can verify numerically that the inequality holds for $\beta = 0.35$, and indeed for slightly larger values. This concludes the proof of Theorem 2.21. □

Remark 2.23. *One can find explicit pairs* q, β *that work by solving the inequality* (2.30) *numerically. For instance quantiles* $0.1, 0.3,$ *and* 0.5 *can handle corruption rates of roughly* $0.32, 0.25,$ *and* 0.18 *respectively.*

Remark 2.24. *An adversary generating corruptions at runtime can make the bounds in the proof of 2.21 as tight as desired. Thus one cannot expect convergence in general if* β *is much larger than* $0.35.$

Remark 2.25. *While the above analysis gives results that are on the same order of magnitude as experiments show, this setting is far more adversarial than what one would encounter in practice. Our experiments demonstrate that one can tolerate higher levels of corruptions than what our theory predicts in this setting. Extending the analysis to the setting of our experiments would require fixing a particular model for the corruptions. By considering adversarial corruptions generated at run-time, we handle any such model.*

2.5 Implementation Considerations

In this section, we discuss several important considerations regarding the implementation of QuantileRK and QuantileSGD. In particular, we touch on the streaming setting in which the rows of the measurement matrix are sampled from a distribution and provided in an online manner. We additionally discuss various considerations for constructing the sample of the residual, and the choice of quantile to apply in each method.

2.5.1 Streaming setting

First, we note that the streaming setting described in Section 2.4.3 provides a good model for many of our experiments. For example, in several of the experiments below, we sample 2000 rows (2000 iterations) from a 50000 row Gaussian matrix. We expect most rows to be sampled only once, which places us within the context of the streaming setting. For this reason, we expect that our methods can in practice handle a larger fraction of corruptions than is reflected in Theorems 2.1 and 2.2.

2.5.2 Sample size

Next, we mention several approaches for decreasing the computational burden of computing the residual in each iteration of QuantileRK and QuantileSGD. Note that both QuantileRK and QuantileSGD as written in Methods 1 and 2 use a sample of the residual of size t. This is much more efficient than constructing the entire residual in each iteration, with the cost scaling with tn instead of mn when constructing the entire residual.

The optimal sample size depends upon the quantile chosen, the fraction of corruptions, and the number of iterations employed. Given the fraction of corruptions, one should choose the sample size and quantile so that the number of corruptions in the sample is at most $(1-q)t$ with high probability (this could be calculated with a Chernoff bound). In particular, more aggressive methods with higher choice of quantile demand larger sample size to ensure that corruptions may be avoided with the quantile calculation.

2.5.3 Quantile selection

For QuantileRK, a larger quantile corresponds to a more aggressive method which is more likely to make the sampled projection. The quantile can be chosen quite close to one if very few corruptions are expected. Meanwhile, for QuantileSGD, the OptSGD theory demonstrates that the optimal quantile to select is the mean of the uncorrupted residual. In the case of Gaussian rows with no corruptions, the mean happens to coincide with the 0.58 quantile. So for QuantileSGD the quantile should be chosen near 0.5 if few corruptions are expected.

2.5.4 Sliding window

Now, as mentioned previously, constructing the sample of the residual requires $\mathcal{O}(tn)$ computation. We can decrease this per-iteration cost by reusing residual entries between iterations. This suggests using a 'sliding window' approach where the sample from which we compute the quantile consists of residual entries collected over multiple iterations. We implement this approach in the experiments below, using on the order of several hundred of the most recently computed residuals. One might expect that this causes significant loss in performance due to the varying scale of the residuals in each iteration, but empirically we see nearly identical performance for moderately sized windows (on the order of 100-500 iterations).

The sliding window approach raises the question of what to do in the initial iterations before the iteration number has reached the window size. One could populate the entire window in the first iteration by sampling as many residual entries as the window size, and then just replacing residual entries as new ones are sampled in the next iterations. Alternatively, one could simply use a partial window until the iteration number reaches the window size. However, this could significantly slow convergence if there are corruptions that are large relative to the initial error $\|\mathbf{x}_0 - \mathbf{x}^*\|$ that get sampled in these initial iterations.

2.6 Experimental Results

Each experiment is run using using Python version 3.6.9 on a single 24-core machine.

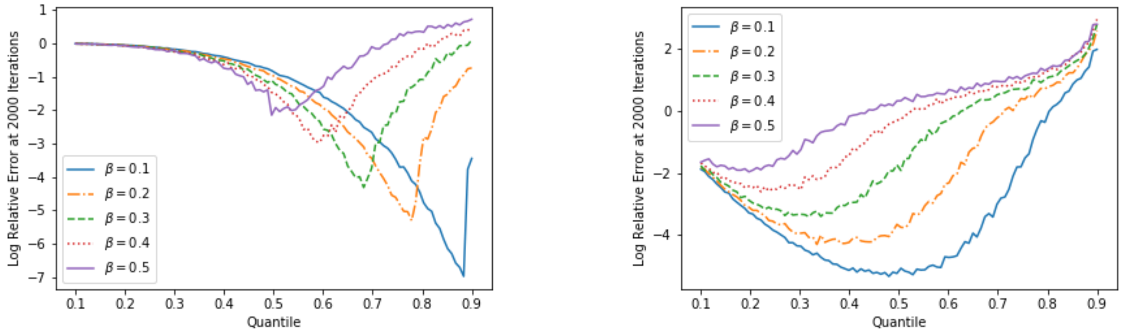
2.6.1 Comparing various quantiles

Our theoretical analysis does not provide specific guidance for choice of quantiles (besides rough relationships between q and β), so we investigate the problem of choosing quantiles empirically. Figure 2.1 shows the behaviors of QuantileRK and QuantileSGD for various corruption rates β and choices of quantile q. For each β, we plot the log relative error after 2000 iterations as a function of q (this is the quantity $\log(\|\mathbf{x}_{2000} - \mathbf{x}^*\| \,/\, \|\mathbf{x}_0 - \mathbf{x}^*\|)$. In order to de-noise the plots, each plotted point is the median over 10 trials. On each trial we generate a new 50000×100 Gaussian system with a β fraction of corrupted entries. Each corrupted entry of $\mathbf{b}$ is modified by adding a uniformly random value in $[-5, 5]$.

Figure 2.2 presents the same plot, however this time the corruptions are chosen to form a consistent subsystem. We seem similar behavior to when the corruptions are independent. However for corruption rates near 0.5 it becomes impossible to distinguish the corrupted subsystem from the uncorrupted subsystem, and so convergence is poor for all quantiles.

In the case of QuantileRK, we see that the optimal quantile tends to be just shy of $1 - \beta$. This aligns with the intuition that QuantileRK should be as aggressive as possible while avoiding projections onto badly corrupted hyperplanes. It is clear that QuantileRK cannot choose a quantile larger than β, otherwise we are likely to sample in the β fraction of corrupted rows, resulting in a threshold which is too large. In practice it is often best to choose a quantile which is somewhat smaller that what the graph suggests. As the quantile approaches $1 - \beta$ the risk of performing a bad projection becomes large enough that we observe bad projections within a few thousand iterations.

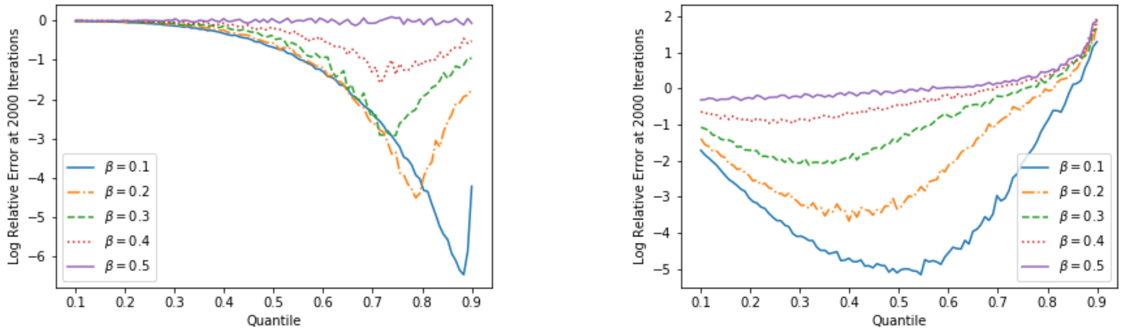
We see that QuantileSGD is much more robust to the choice of quantile. For instance when $\beta = 0.1$, the optimal quantile appears to be near 0.5. However we see near-optimal convergence behavior as long as β is between 0.3 and 0.7.



(a) QuantileRK

(b) QuantileSGD

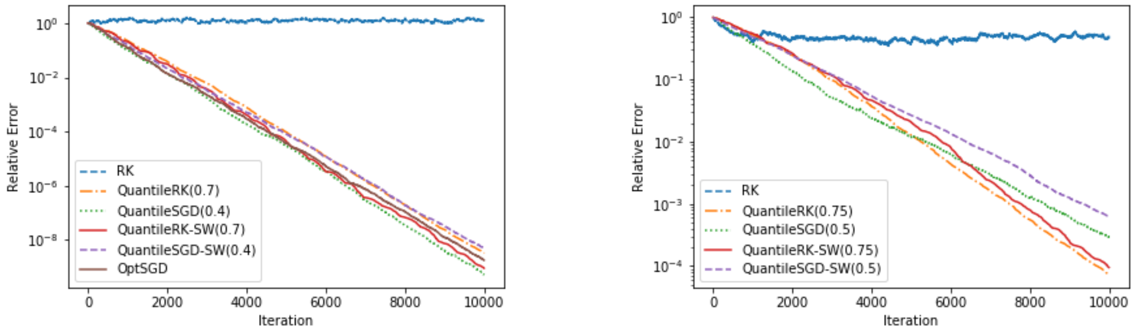
Figure 2.1: $\log(\|\mathbf{x}_{2000} - \mathbf{x}^*\| \,/\, \|\mathbf{x}_0 - \mathbf{x}^*\|)$ for (a) QuantileRK and (b) QuantileSGD run on 50000×100 Gaussian system, with various corruption rates β and quantile choices.



(a) QuantileRK

(b) QuantileSGD

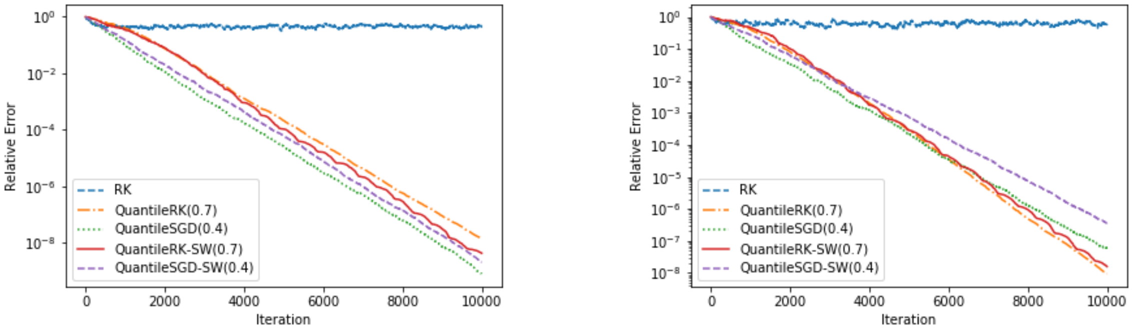
Figure 2.2: $\log(\|\mathbf{x}_{2000} - \mathbf{x}^*\| \,/\, \|\mathbf{x}_0 - \mathbf{x}^*\|)$ for (a) QuantileRK and (b) QuantileSGD run on 50000×100 system with consistent corruptions, for various corruption rates β and quantile choices.



(a) Gaussian model

(b) Coherent model

Figure 2.3: Relative error as a function of iteration count plotted for a 50000×100 Gaussian and coherent model with a 0.2 corruption rate. The coherent system was generated by sampling entries uniformly in $[0, 1)$ and then normalizing the rows of the resulting matrix.



(a) Bernoulli model

(b) Coherent subsystem model

Figure 2.4: Relative error as a function of iteration count plotted for a 50000×100 Bernoulli and adversarial model with a 0.2 corruption rate. Each entry of the Bernoulli matrix is generated to be -1 or 1 before normalizing rows. For the coherent subsystem model, a random subset of rows from the corresponding Gaussian system were selected and corrupted to yield a $0.2m$ sized consistent subsystem.

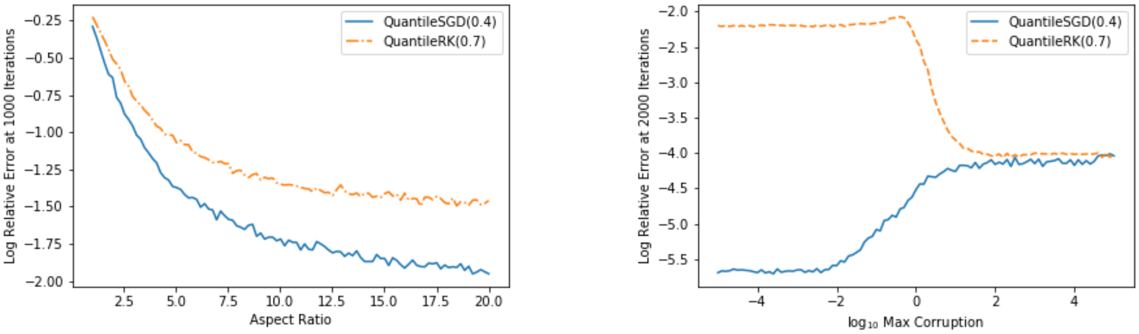
2.6.2 Convergence plots for the streaming model

In Figure 2.3 and Figure 2.4 we show the convergence behavior of our methods on a 50000×100 system with a $\beta = 0.2$ fraction of corruptions. In Figure 2.3 and Figure 2.4 (a) entries are corrupted by adding a uniformly random value in $[-5, 5]$.

The label "RK" signifies the standard Randomized Kaczmarz method without thresholding. The methods marked QuantileRK-SW and QuantileSGD-SW are the "sliding window" versions of QuantileRK and QuantileSGD. The methods marked QuantileRK and QuantileSGD are the sampled variants. We set our window size and sample size to 400 for these experiments. Finally, we include OptSGD only in Figure 2.3 (a).

In Figure 2.3 (a) we show a normalized Gaussian system (i.e., a system with rows sampled uniformly over S^{n-1}). We observe that all four of our quantile methods exhibit similar convergence behavior. Notably, these methods perform comparably to OptSGD, which chooses an optimal step size at each iteration. (Of course OptSGD cannot be run in practical settings, as it requires knowledge of $\mathbf{x}^*$.)

In Figure 2.3 (b) we consider a poorly-conditioned system with "coherent rows". This matrix



(a) Effect of aspect ratio

(b) Effect of corruption size

Figure 2.5: (a) Log relative error for QuantileSGD and QuantileRK after 1000 iterations on a $100a \times 100$ Gaussian system with a 0.2 corruption rate, where $a = m/n$ is the aspect ratio of the matrix. (b) Log relative error for QuantileSGD and QuantileRK after 2000 iterations, as a function of corruption size. We use a 50000×100 Gaussian system and corrupt our system by adding a uniform value in $[-10^x, 10^x]$.

is created by generating each entry i.i.d. uniformly in $[0, 1]$, and then normalizing the rows of the resulting matrix. We call the system coherent because pairs of rows typically have large inner product with one another. Such a matrix does not have isotropic rows, and is therefore not covered by our theoretical analysis. Nonetheless, we do observe convergence, albeit at a slower rate than for the Gaussian model.

In Figure 2.4 (a) we show a Bernoulli system. Here each entry of our matrix is sampled uniformly in $\{-1, 1\}$ and the rows are normalized. This matrix violates the “bounded density” assumption of our theoretical analysis. However we still see convergence behavior which is comparable to the Gaussian case.

Figure 2.4 (b) shows a Gaussian system which is corrupted as badly as possible, in the sense that the values of the corrupted entries are chosen to form a consistent subsystem. In this model, we choose a random collection of indices to corrupt, and then choose values such that the corrupted subsystem is consistent. In effect, we attempt to trick the solver by creating a phantom solution in addition to $\mathbf{x}^*$. Our theory does address this case, and here we see convergence is comparable to when the corrupted values are independently chosen.

2.6.3 Influence of the aspect ratio

Each of our experiments so far dealt with extremely tall 50000×100 matrices. Since we ran at most 10000 iterations we were unlikely to sample a given row many times. Thus our experiments have effectively been run in the streaming setting. A strength of our theory was providing convergence guarantees even for matrices which are not too tall. In Figure 2.5 (a) we show the convergence behavior of QuantileSGD and QuantileRK as a function of the aspect ratio. In this plot we consider random Gaussian matrices with a $\beta = 0.2$ fraction of corruptions which are $100a \times 100$, where a is the aspect ratio. Each data point is the median error taken over 100 separate trials.

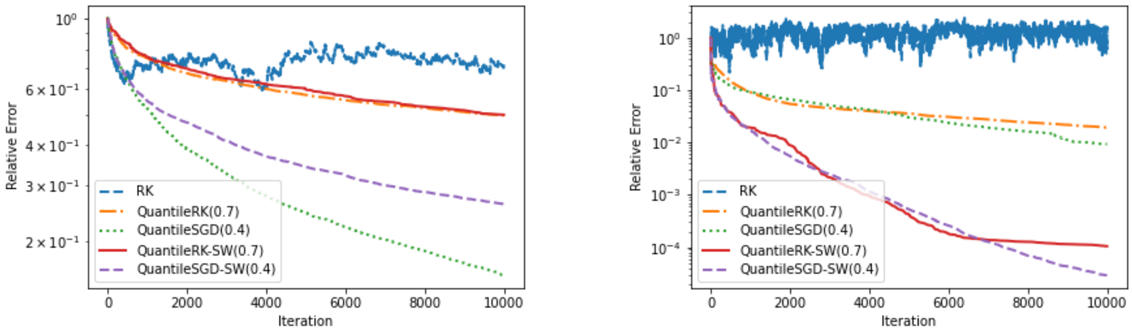
2.6.4 Effect of corruption size

In Figure 2.5 (b) we illustrate the behaviors of QuantileSGD and QuantileRK as the corruption sizes are varied. For each value on the x-axis, x, we corrupt the vector $\mathbf{b}$ by adding values sampled uniformly from $[-10^x, 10^x]$ to a $\beta = 0.2$ fraction of entries. As we see, both of our methods still converge well even when the corruption sizes are very large. Their behavior for very small errors is perhaps surprising.

In particular, QuantileRK seems to perform better when the corruptions are very large.[5] The reason for this is that when the corruptions are very small relative to $\|\mathbf{x}_k - \mathbf{x}^*\|$, the system behaves as though it is consistent. For a consistent system QuantileRK behaves too conservatively by rejecting 30 percent of the rows. When the size of corruptions becomes comparable to or larger than $\|\mathbf{x}_0 - \mathbf{x}^*\|$, this behavior disappears.

QuantileSGD on the other hand behaves better for consistent systems as rows are never rejected. The more consistent the system, the more likely a given step is to move the iterate closer to $\mathbf{x}^*$. We see this behavior for QuantileRK and QuantileSGD in Figure 2.1 as well.

[5] This type of behavior was noted in [HN18b], although for different reasons.



(a) Tomography system

(b) Wisconsin Breast Cancer dataset

Figure 2.6: (a) Relative error for each method run on a 1200×400 system designed for tomography. Corruptions were added to 100 uniformly random entries of $\mathbf{b}$. (b) Relative error for each method run on a 699×10 matrix obtained from the Wisconsin Breast Cancer dataset. Corruptions were added to 100 uniformly random entries of $\mathbf{b}$.

2.6.5 Real world data

Finally, in Figure 2.6 we illustrate our methods on two real world data sets. In Figure 2.6 (a), we experiment on a tomography problem generated using the Matlab Regularization Toolbox by P.C. Hansen (`http://www.imm.dtu.dk/~pcha/Regutools/`) [Han07]. We present a 2D tomography problem $\mathbf{A}\mathbf{x} = \mathbf{b}$ for an $m \times n$ matrix with $m = fN^2$ and $n = N^2$. Here $\mathbf{A}$ corresponds to the absorption along a random line through an $N \times N$ grid. In this experiment, we set $N = 20$ and the oversampling factor $f = 3$, which yields a matrix $\mathbf{A} \in \mathbb{R}^{1200\times400}$. As the resulting system was consistent, we randomly sampled 100 indices uniformly from among the rows of $\mathbf{A}$ and corrupted the right-hand side vector $\mathbf{b}$ in these entries by adding a uniformly random value in $[-5, 5]$.

In Figure 2.6 (b) we use a corrupted system generated from the Wisconsin (Diagnostic) Breast Cancer data set, which includes data points whose features are computed from a digitized image of a fine needle aspirate (FNA) of a breast mass and describe characteristics of the cell nuclei present in the image [Lic13]. This collection of data points forms our matrix $\mathbf{A} \in \mathbb{R}^{699\times10}$, we construct $\mathbf{b}$ to form a consistent system, and then corrupt a random selection of 100 entries of the right-hand side by adding a uniformly random value in $[-5, 5]$.

The label “RK” signifies the standard Randomized Kaczmarz method without thresholding. The methods marked QuantileRK-SW and QuantileSGD-SW are the “sliding window” versions of QuantileRK and QuantileSGD. The methods marked QuantileRK and QuantileSGD are the sampled variants. We set our window size and sample size to 100 for these experiments. Again, all four of our proposed methods converge; however, the difference in empirical convergence rate is clearly discernible on this data. It is notable that in Figure 2.6 (b) the sliding window variants of the method converge more quickly. Since the sliding window quantile estimate lags into the past where residual entries had larger magnitude, it will typically yield a larger quantile than resampling on each iteration. The effect is to allow for more aggressive projections and step sizes in Quantile-RK and Quantile-SGD.

2.7 Conclusion

In this work, we propose two new methods, QuantileRK and QuantileSGD, for solving large-scale systems of equations which are inconsistent due to sparse, arbitrarily large corruptions in the measurement vector. Such corrupted systems of equations arise in practice in many applications, but are especially abundant and challenging in areas such as distributed computing, internet of things, and other network problems facing potentially adversarial corruption.

The QuantileRK and QuantileSGD methods make use of a quantile statistic of a sample of the residual in each iteration. We prove that each method enjoys exponential convergence with mild assumptions on the distribution of the entries of the measurement matrix $\mathbf{A}$, the quantile parameter of the method q, and the fraction of corruptions β.

Our experiments support these theoretical results, as well as illustrate that the methods converge in many scenarios not captured by our theoretically required assumptions. In particular, these methods are able to handle fractions of corruption larger than those predicted theoretically, and converge for systems defined by structured and real measurement matrices which are far from the random matrices for which our theoretical results hold. We note that both theoretically and experimentally we see that the magnitude of the corruptions do not negatively impact convergence.

While our experiments show that QuantileRK and QuantileSGD yield good results on many types of corrupted systems, our theory is currently limited to near-Gaussian random matrix model. One could hope to extend the theory in several directions.

A concurrent work addressed the situation of noise in addition to large corruptions [JN21]. It would also be nice to show a convergence result in the Bernoulli random model. The main obstacle is that we can no longer hope for a per-iteration guarantee that holds over all potential iterates. One would need to show that the set of points at which the per-iteration guarantee fails is small, and that the dynamics of our algorithms are unlikely to be biased towards these "bad" points.

Second, one could also hope to give a non-random characterization of matrices for which our algorithms have good convergence properties. To handle adversarial corruptions it is probably necessary to assume some type of incoherence. Otherwise the corruptions could be structured to align in a particular direction which points away from x^*. Alternatively, is it possible to detect coherent row subsets of A in order to preempt the effect of structured corruptions?

As $\mathbf{x}_k$ approaches $\mathbf{x}^*$ the larger corruptions should become easier to identify. Can one design an algorithm which removes such rows, thereby speeding up convergence? This is similar in spirit to the iterative removal approach discussed in [HN18a]. Alternatively, might it be useful to reduce the value of the quantile q throughout the course of the algorithm in order to more aggressively reject corrupted rows when $\mathbf{x}_k$ is near $\mathbf{x}^*$?

Finally, one might also consider a non-random model for A, but where the corrupted entries of b are non-random. In this setting it seems reasonable that the theory should continue to hold for structured A. One could also attempt to generalize our results to systems of inequalities, and to partially-greedy row sampling schemes.

CHAPTER 3

Testing Positive Semidefiniteness with Linear Measurments

Given a symmetric matrix it is natural to ask whether it is positive semi-definite (PSD). For instance, one might have access to the Hessian of a function and then ask whether a particular critical point x_0 corresponds to a local minimum.

For extremely large matrices this may require a large amount of calculation, particularly if we wish to distinguish between PSD matrices and matrices with a very small negative eigenvalue. Following the property testing framework, we relax this problem slightly and aim to distinguish between matrices that are PSD and matrices with a large negative eigenvalue. Moreover we allow for randomized algorithms and only require that we distinguish between these two cases with a small constant probability of failure.

To quantify the difficulty of this problem we consider query complexity. That is, we allow ourselves extract information about the matrix by making some kind of linear measurement. For example we consider matrix-vector products as one such query model, as well queries to the bilinear form associated to the matrix.

We will aim to give tight bounds for the PSD-testing problem in these linear query models both for adaptive and non-adaptive queries. Our upper bounds rely on a novel sketch for detecting an extreme eigenvalue, as well as a new analysis of Oja's method for approximate the largest negative eigenvalue. To prove matching lower bounds we use a variety of techniques including reductions from communication complexity lower bounds.

In the following chapter we will build on some of the techniques from this chapter to address the more general spectral approximation problem.

3.1 Contributions

This section contains work presented in [NSW22] which is joint with Deanna Needell and David Woodruff. I proposed analyzing the PSD-testing problem in the matrix-vector and vector-matrix-vector query models, proposed the main algorithms, and proved the main upper bound results. David Woodruff contributed much of the insight for the lower bound results, proposed the spectral approximation algorithm given here, and was the first to observe a separation between one-sided and two-sided testers. All authors contributed to writing and editing the manuscript, and participated in many helpful discussions about this work.

3.2 Introduction

A real-valued matrix $A \in \mathbb{R}^{n \times n}$ is said to be Positive Semi-Definite (PSD) if it defines a non-negative quadratic form, namely, if $x^T A x \geq 0$ for all x. If A is symmetric, the setting on which we focus, this is equivalent to the eigenvalues of A being non-negative. Multiple works [KS03; Han+17; BCJ20] have studied the problem of testing whether a real matrix is PSD, or is far from being PSD, and this testing problem has numerous applications, including to faster algorithms for linear systems and linear algebra problems, detecting the existence of community structure, ascertaining local convexity, and differential equations; we refer the reader to [BCJ20] and the references therein.

We study two fundamental query models. In the matrix-vector model, one is given implicit access to a matrix A and may query A by choosing a vector v and receiving the vector Av. In the vector-matrix-vector model one chooses a pair of vectors (v, w) and queries the bilinear form associated to A. In other words the value of the query is $v^T A w$. In both models, multiple, adaptively-chosen queries can be made, and the goal is to minimize the number of queries to solve a certain task. These models are standard computational models in the numerical linear algebra community, see, e.g., [Han+17] where PSD testing was studied in the matrix-vector query model. These models were recently formalized in the theoretical computer science community in [Sun+19; RWZ20], though similar models have been studied in numerous fields, such as the number of measurements in compressed sensing, or the sketching dimension of a streaming algorithm. The matrix-vector query and vector-matrix-vector query models are particularly relevant when the input matrix A is not given explicitly.

A natural situation occurs when A is presented implicitly as a the Hessian of a function $f : \mathbb{R}^d \to \mathbb{R}^d$ at a point x_0, where f could be the loss function of a neural network for example. One might want to quickly distinguish between a proposed optimum of f truly being a minimum, or being a saddle point with a direction of steep downward curvature. Our query model is quite natural in this context. A Hessian-vector product is efficient to compute using automatic differentiation techniques. A vector-matrix-vector product corresponds to a single second derivative computation, $D^2f(v, w)$. This can be approximated using 4 function queries by the finite difference approximation $D^2f(v, w) \approx \frac{f(x_0+hv+hw)-f(x_0+hv)-f(x_0+hw)+f(x_0)}{h^2}$, where h is small.

While there are numerically stable methods for computing the spectrum of a symmetric matrix, and thus determining if it is PSD, these methods can be prohibitively slow for very large matrices, and require a large number of matrix-vector or vector-matrix-vector products. Our goal is to obtain significantly more efficient algorithms in these models, and we approach this problem from a property testing perspective. In particular, we focus on the following version of the PSD-testing problem. In what follows, $\|A\|_p = \left(\sum_{i=1}^n \sigma_i^p\right)^{1/p}$ is the Schatten-p norm of A, where the σ_i are the singular values of A.

Definition 3.1. *For $p \in [1, \infty]$, an (ϵ, ℓ_p)-tester is an algorithm that makes either matrix-vector or vector-matrix-vector queries to a real symmetric matrix A, and outputs True with at least $2/3$ probability if A is PSD, and outputs False with $2/3$ probability if A is $\epsilon \|A\|_p$-far in spectral dis-*

tance from the PSD cone, or equivalently, if the minimum eigenvalue $\lambda_{\min}(A) \leq -\epsilon \|A\|_p$. If the tester is guaranteed to output True on all PSD inputs (even if the input is generated by an adversary with access to the random coins of the tester), then the tester has one-sided error. Otherwise it has two-sided error. When ϵ is clear from the context we will often drop the ϵ and simply refer to an ℓ_p-tester.

Our work fits more broadly into the growing body of work on property testing for linear algebra problems, see, for example [Bal+19b; BCJ20; BMR21]. However, a key difference is that we focus on matrix-vector and vector-matrix-vector query models, which might be more appropriate than the model in the above works which charges a cost of 1 for reading a single entry. Indeed, such models need to make the assumption that the entries of the input are bounded by a constant or slow-growing function of n, as otherwise strong impossibility results hold. This can severely limit the applicability of such algorithms to real-life matrices that do not have bounded entries; indeed, even a graph Laplacian matrix with a single degree that is large would not fit into the above models. In contrast, we use the matrix-vector and vector-matrix-vector models, which are ideally suited for modern machines such as graphics processing units and when the input matrix cannot fit into RAM, and are standard models in scientific computing, see, e.g., [BFG96].

While we focus on vector-matrix-vector queries, our results shed light on several other natural settings. Many of our results are in fact tight for general linear measurements which vectorize the input matrix and apply adaptively chosen linear forms to it. For long enough streams the best known single or multi-pass algorithms *for any problem* in the turnstile streaming model form a sketch using general linear measurements, and with some additional restrictions, it can be shown that the optimal multi-pass streaming algorithm just adaptively chooses general linear measurements [Ai+16]. Therefore, it is quite plausible that many of our vector-matrix-vector algorithms give tight single pass streaming bounds, given that vector-matrix-vector queries are a special case of general linear measurements, and that many our lower bounds are tight even for general linear measurements.

Moreover our vector-matrix-vector algorithms lead to efficient communication protocols for deciding whether a distributed sum of matrices is PSD, provided that exact vector-matrix-vector products may be communicated. While we expect our methods to be stable under small perturbations (i.e. when the vector-matrix-vector products are slightly inexact), we leave the full communication complexity analysis to future work.

We note that our PSD-testing problem is closely related to that of approximating the largest eigenvalue of a PSD matrix. Indeed by appropriately negating and shifting the input matrix, it is essentially equivalent to estimate the largest eigenvalue of a PSD matrix A up to additive error $\epsilon\left(\sum_i |\lambda_{\max}(A) - \lambda_i(A)|^p\right)^{1/p}$. However this problem is much less natural as real-world matrices often have many small eigenvalues, but only a few large eigenvalues.

3.2.1 Our Contributions

We study PSD-testing in the matrix-vector and vector-matrix-vector models. In particular, given a real symmetric matrix A, and $p \in [1, \infty]$, we are interested in deciding between (i) A is PSD and (ii) A has an eigenvalue less than $-\epsilon \left\|A\right\|_p$, where $\left\|A\right\|_p$ is the Schatten p-norm of A.

Tight Bounds for One-sided Testers. We make particular note of the distinction between one-sided and two-sided testers. In some settings one is interested in a tester that produces one-sided error. When such a tester outputs False, it must be able to produce a proof that A is not PSD. The simplest such proof is a witness vector v such that $v^T A v < 0$, and indeed we observe that in the matrix-vector model, any one-sided tester can produce such a v when it outputs False. This may be a desirable feature if one wishes to apply these techniques to saddle point detection for example: given a point that is not a local minimum, it would be useful to produce a descent direction so that optimization may continue. In the vector-matrix-vector model the situation is somewhat more complicated in general, but all of our one-sided testers produce a witness vector whenever they output False.

We provide *optimal bounds* for one-sided testers for both matrix-vector and vector-matrix-

Vector-matrix-vector queries		
Adaptive, one-sided ℓ_p	$\widetilde{\Theta}(\frac{1}{\epsilon} d^{1-1/p})$	Corollary 3.3, Theorem 3.15
Non-adaptive, one-sided ℓ_p	$\widetilde{\Theta}(\frac{1}{\epsilon^2} d^{2-2/p})$	Corollary 3.42, Theorem 3.13
Adaptive, two-sided ℓ_2	$\widetilde{\Theta}(\frac{1}{\epsilon^2})^*$	Proposition 3.29, Corollary 3.33
Non-adaptive, two-sided ℓ_2	$\widetilde{\Theta}(\frac{1}{\epsilon^4})^*$	Theorem 3.28, Theorem 3.31
Adaptive, two-sided ℓ_p, $2 \leq p < \infty$	$\widetilde{\Theta}(\frac{1}{\epsilon^2} d^{1-2/p})^*$	Corollary 3.30, Corollary 3.33
Matrix-vector queries		
Adaptive one-sided ℓ_p	$\widetilde{O}((1/\epsilon)^{p/(2p+1)} \log d)$, $\Omega((1/\epsilon)^{p/(2p+1)})$	Theorem 3.17, Theorem 3.20
Adaptive one-sided ℓ_1	$\widetilde{\Theta}((1/\epsilon)^{1/3})$	Theorem 3.17, Theorem 3.20
Non-adaptive one-sided ℓ_p	$\Theta(\frac{1}{\epsilon} d^{1-1/p})$	Proposition 3.43, Corollary 3.46

Table 3.1: Our upper and lower bounds for the matrix-vector and vector-matrix-vector query models. * indicates that the lower bound holds for general linear measurements.

vector models. The bounds below are stated for constant probability algorithms. Here $\widetilde{O}(f) = f \cdot \text{poly}(\log f)$.

1. In the matrix-vector query model, we show that up to a factor of $\log d$, $\widetilde{\Theta}(1/\epsilon^{p/(2p+1)})$ queries are necessary and sufficient for an ℓ_p-tester for any $p \geq 1$. In the $p = 1$ case, we note that the $\log d$ factor may be removed.

2. In the vector-matrix-vector query model, we show that $\widetilde{\Theta}(d^{1-1/p}/\epsilon)$ queries are necessary and sufficient for an ℓ_p-tester for any $p \geq 1$. Note that when $p = 1$ we obtain a very efficient $\widetilde{O}(1/\epsilon)$-query algorithm. In particular, our tester for $p = 1$ has query complexity independent of the matrix dimensions, and we show a sharp phase transition for $p > 1$, showing in some sense that $p = 1$ is the largest value of p possible for one-sided queries.

The matrix-vector query complexity is very different than the vector-matrix-vector query complexity, as the query complexity is $\text{poly}(1/\epsilon)$ for any $p \geq 1$, which captures the fact that each matrix-vector query response reveals more information than that of a vector-matrix-vector query, though a priori it was not clear that such responses in the matrix-vector model could not be compressed using vector-matrix-vector queries.

An Optimal Bilinear Sketch for Two-Sided Testing. Our main technical contribution for two-sided testers is a bilinear sketch for PSD-testing with respect to the Frobenius norm, i.e. $p = 2$. We consider a Gaussian sketch $G^T AG$, where G has small dimension $\widetilde{O}(\frac{1}{\epsilon^2})$. By looking at the smallest eigenvalue of the sketch, we are able to distinguish between A being PSD and being ϵ-far from PSD. Notably this tester may reject even when $\lambda_{\min}(G^T AG) > 0$, which results in a two-sided error guarantee. This sketch allows us to obtain tight two-sided bounds in the vector-matrix-vector model for $p \geq 2$, both for adaptive and non-adaptive queries.

Separation Between One-Sided and Two-Sided Testers. Surprisingly, we show a separation between one-sided and two-sided testers in the vector-matrix-vector model. For the important case of the Frobenius norm, i.e., $p = 2$, we utilize our bilinear sketch to construct a $\widetilde{O}(1/\epsilon^2)$ query two-sided tester, whereas by our results above, any adaptive one-sided tester requires at least $\Omega(\sqrt{d}/\epsilon)$ queries.

We also show that for any $p > 2$, any possibly adaptive two-sided tester requires $d^{\Omega(1)}$ queries for constant ϵ, and thus in some sense, $p = 2$ is the largest value of p possible for two-sided queries.

On the Importance of Adaptivity. We also study the role of adaptivity in both matrix-vector and vector-matrix-vector models. In both the one-sided and two-sided vector-matrix-vector models we show a quadratic separation between adaptive and non-adaptive testers, which is the largest gap possible for any vector-matrix-vector problem [Sun+19].

In the matrix-vector model, each query reveals more information about A than in the vector-matrix-vector model, allowing for even better choices for future queries. Thus we have an even larger gap between adaptive and non-adaptive testers in this setting.

Spectrum Estimation. While the two-sided tester discussed above yields optimal bounds for PSD testing, it does not immediately give a way to estimate the negative eigenvalue when it exists. Via a different approach, we show how to give such an approximation with $\epsilon \left\|A\right\|_F$ additive error. In fact, we show how to approximate all of the top k eigenvalues of A using $O(k^2 \text{poly}\frac{1}{\epsilon})$ non-adaptive vector-matrix-vector queries, which may be of independent interest.

We note that this gives an $O(k^2 \text{poly}\frac{1}{\epsilon})$ space streaming algorithm for estimating the top k eigenvalues of A to within additive Frobenius error. Prior work yields a similar guarantee for the singular values [AN13], but cannot recover the signs of eigenvalues.

3.2.2 Our Techniques

Matrix-Vector Queries. For the case of adaptive matrix-vector queries, we show that Krylov iteration starting with a single random vector yields an optimal ℓ_p-tester for all p. Interestingly, our analysis is able to beat the usual Krylov matrix-vector query bound for approximating the top eigenvalue, as we modify the usual polynomial analyzed for eigenvalue estimation to implicitly implement a *deflation* step of all eigenvalues above a certain threshold. We do not need to explicitly know the values of the large eigenvalues in order to deflate them; rather, it suffices that there exists a low degree polynomial in the Krylov space that implements this deflation.

Further, we show that our technique is tight for all $p \geq 1$ by showing that any smaller number of matrix-vector products would violate a recent lower bound of [Bra+20] for approximating the smallest eigenvalue of a Wishart matrix. This lower bound applies even to two-sided testers.

Vector-Matrix-Vector Queries. We start by describing our result for $p = 1$. We give one of the first examples of an algorithm in the vector-matrix-vector query model that leverages adaptivity in an interesting way. Most known algorithms in this model work non-adaptively, either by applying a bilinear sketch to the matrix, or by making many independent queries in the case of Hutchinson's trace estimator [Hut89]. Indeed, the algorithm of [AN13] works by computing $G^T AG$ for a Gaussian matrix G with $1/\epsilon$ columns, and arguing that all eigenvalues that are at least $\epsilon\|A\|_1$ can be estimated from the sketch. The issue with this approach is that it uses $\Omega(1/\epsilon^2)$ queries and this bound is tight for non-adaptive testers! One could improve this by running our earlier matrix-vector algorithm on top of this sketch, without ever explicitly forming the $1/\epsilon \times 1/\epsilon$ matrix $G^T AG$; however, this would only give an $O(1/\epsilon^{4/3})$ query algorithm.

To achieve our optimal $\widetilde{O}(1/\epsilon)$ complexity, our algorithm instead performs a novel twist to Oja's algorithm [Oja82], the latter being a stochastic gradient descent (SGD) algorithm applied to optimizing the quadratic form $f(x) = x^T Ax$ over the sphere. In typical applications, the randomness of SGD arises via randomly sampling from a set of training data. In our setting, we instead artificially introduce randomness at each step, by computing the projection of the gradient onto a randomly chosen direction. This idea is implemented via the iteration

$$x^{(k+1)} = x^k - \eta(g^T Ax^k)g \text{ where } g \sim \mathcal{N}(0, I) \tag{3.1}$$

for a well-chosen step size η. If f ever becomes negative before reaching the maximum number of iterations, then the algorithm outputs False, otherwise it outputs True. For $p = 1$, we show that this scheme results in an optimal tester (up to logarithmic factors). Our proof uses a second moment analysis to analyze a random walk, that is similar in style to [Jai+16], though our analysis is quite different. Whereas [Jai+16] considers an arbitrary i.i.d. stream of unbiased estimators to A (with bounded variance), our estimators are simply $gg^T A$, which do not seem to have been considered before. We leverage this special structure to obtain a better variance bound on the iterates throughout the first $\widetilde{O}(1/\epsilon)$ iterations, where each iteration can be implemented with a single vector-matrix-vector query. Our algorithm and analysis gives a new method for the fundamental problem of approximating eigenvalues.

Our result for general $p > 1$ follows by relating the Schatten-p norm to the Schatten-1 norm and invoking the algorithm above with a different setting of ϵ. We show our method is optimal by proving an $\Omega(d^{2-2/p}/\epsilon^2)$ lower bound for non-adaptive one-sided testers, and then using a theorem in [RWZ20] which shows that adaptive one-sided testers can give at most a quadratic improvement. We note that one could instead use a recent streaming lower bound of [INW22] to prove this lower bound, though such a lower bound would depend on the bit complexity.

Two-Sided Testers. The key technical ingredient behind all of our two-sided testers is a bilinear sketch for PSD-testing. Specifically, we show that a sketch of the form $G^T AG$ with $G \in R^{d\times k}$ is sufficient for obtaining a two-sided tester for $p = 2$. In contrast to the $p = 1$ case, we do not simply output False when $\lambda_{\min} := \lambda_{\min}(G^T AG) < 0$ as such an algorithm would automatically be one-sided. Instead we require a criterion to detect when $\lambda_{\min}$ is suspiciously small. For this we require two results.

The first is a concentration inequality for $\lambda_{\min}(G^T AG)$ when A is PSD. We show that $\lambda_{\min} \geq \mathrm{Tr}(A) - \widetilde{O}(\sqrt{k})\, \|A\|_F$ with very good probability. This result is equivalent to bounding the smallest singular value of $A^{1/2}G$, which is a Gaussian matrix whose rows have different variances. Although many similar bounds for constant variances exist in the literature [Lit+05; Ver18], we were not able to find a bound for general covariances. In particular, most existing bounds do not seem to give the concentration around $\mathrm{Tr}(A)$ that we require.

When A has a negative eigenvalue of $-\epsilon$, we show that $\lambda_{\min} \leq \mathrm{Tr}(A) - \epsilon O(k)$. By combining these two results, we are able to take $k = \widetilde{O}(1/\epsilon^2)$, yielding a tight bound for non-adaptive testers in the vector-matrix-vector model. In fact this bound is even tight for general linear sketches, as we show by applying the results in [LW16].

We also utilize this bilinear sketch to give tight bounds for adaptive vector-matrix-vector queries, and indeed for general linear measurements. By first (implicitly) applying the sketch, and then shifting by an appropriate multiple of the identity we are able to reduce to the (ϵ^2, ℓ_1)-testing problem, which as described above may solved using $\widetilde{O}(1/\epsilon^2)$ queries.

Spectrum Estimation. A natural approach for approximating the eigenvalues of an $n \times n$ matrix A is to first compute a sketch $G^T AG$ or a sketch $G^T AH$ for Gaussian matrices G and H with a small number of columns. Both of these sketches appear in [AN13]. As noted above, $G^T AG$ is a useful non-adaptive sketch for spectrum approximation, but the error in approximating each eigenvalue is proportional to the Schatten-1 norm of A. One could instead try to make the error depend on the Frobenius norm $\|A\|_2$ of A by instead computing $G^T AH$ for independent Gaussian matrices G and H, but now $G^T AH$ is no longer symmetric and it is not clear how to extract the signs of the eigenvalues of A from $G^T AH$. Indeed, [AN13] are only able to show that the *singular values* of $G^T AH$ are approximately the same as those of A, up to additive $\epsilon\|A\|_2$ error. We thus need a new way to *preserve sign information of eigenvalues*.

To do this, we show how to use results for providing the best PSD low rank approximation to an input matrix A, where A need not be PSD and need not even be symmetric. In particular, in [CW17b] it was argued that if G is a Gaussian matrix with $O(k/\epsilon)$ columns, then if one sets up the optimization problem $\min_{\text{rank k PSD } Y} \|AGYG^T A^T - A\|_F^2$, then the cost will be at most $(1+\epsilon)\|A_{k,+} - A\|_F^2$, where $A_{k,+}$ is the best rank-k PSD approximation to A. By further sketching on the left and right with so-called *affine embeddings* S and T, which have poly(k/ϵ) rows and columns respectively, one can reduce this problem to $\min_{\text{rank k PSD } Y} \|SAGYG^TA^TT - SAT\|_F^2$, and now SAG, G^TA^TT and SAT are all poly$(k/\epsilon) \times$ poly(k/ϵ) matrices so can be computed with a poly(k/ϵ) number of vector-matrix-vector products. At this point the optimal Y can be found with no additional queries and its cost can be evaluated. By subtracting this cost from $\|A\|_F^2$, we approximate $\|A_{+,i}\|_F^2$, and $\|A_{-,i}\|_F^2$ for all $i \in [k]$, which in turn allows us to produce (signed) estimates for the eigenvalues of A.

When A is PSD, we note that Theorem 1.2 in [AN13] is able to reproduce our spectral approximation guarantee using sketching dimension $O(\frac{k^2}{\epsilon^8})$, compared to our sketch of dimension $O(\frac{k^2}{\epsilon^{12}})$. However as mentioned above, our guarantee is stronger in that it allows for the signs of the eigenvalues to be recovered, i.e. our guarantee holds even when A is not PSD. Additionally, we are able to achieve $O(\frac{k^2}{\epsilon^8})$ using just a single round of adaptivity.

Lower Bounds for One-sided Testers. To prove lower bounds for one-sided non-adaptive testers, we first show that a one-sided tester must be able to produce a witness whenever it outputs False. In the matrix-vector model, the witness is a vector v with $v^TAv < 0$, and in the vector-matrix-vector model, the witness is a PSD matrix M with $\langle M, A\rangle < 0$. In both cases we show that even for simplest non-PSD spectrum $(-\lambda, 1, \ldots, 1)$, that it takes many queries to produce a witness when λ is small. In the matrix-vector model, our approach is simply to show that the $-\lambda$ eigenvector is typically far from span of all queried vectors, when the number of queries is small. This will imply that A is non-negative on the queried subspace, which precludes the tester from producing a witness. In the vector-matrix-vector model our approach is similar, however now the queries take the form of inner products against rank one matrices $x_ix_i^T$. We therefore need to work within the space of symmetric matrices, and this requires a more delicate argument.

3.2.3 Additional Related Work

Numerous other works have considered matrix-vector queries and vector-matrix queries, see, e.g., [Mey+21; Bra+20; Sun+19; SER18; MM15; WWZ14]. We outline a few core areas here.

Oja's Algorithm. Several works have considered Oja's algorithm in the context of streaming PCA, [Sha16; Jai+16; AL17]. [Jai+16] gives a tight convergence rate for iteratively approximating the top eigenvector of a PSD matrix, given an eigengap, and [AL17] extends this to a gap free result for k-PCA.

PSD Testing. As mentioned above, PSD-testing has been investigated in the bounded entry model, where one assumes that the entries of A are bounded by 1 [BCJ20], and one is allowed to query the entries of A. This is a restriction of the vector-matrix-vector model that we consider where only coordinate vectors may be queried. However since we consider a more general query model, we are able to give a better adaptive tester – for us $\widetilde{O}(1/\epsilon)$ vector-matrix-vector queries suffice, beating the $\Omega(1/\epsilon^2)$ lower bound given in [BCJ20] for entry queries.

Another work on PSD-testing is that of [Han+17], who construct a PSD-tester in the matrix-vector model. They first show how to approximate a general trace function $\sum f(\lambda_i)$ for sufficiently smooth f, by using a Chebyshev polynomial construction to approximate f in the sup-norm over an interval. This allows them to construct an ℓ_∞-tester by taking f to be a smooth approximation of a shifted Heaviside function. Unfortunately this approach is limited to ℓ_∞-testers, and does not achieve the optimal bound; they require $\Omega((\log d)/\epsilon)$ matrix-vector queries compared to the $\widetilde{O}((\log d)/\sqrt{\epsilon})$ queries achieved by Krylov iteration.

Spectrum Estimation. The closely-related problem of spectrum estimation has been considered several times, in the context of sketching the largest k elements of the spectrum [AN13] discussed above, and approximating the entire spectrum from entry queries in the bounded entry model [Bha+21].

3.2.4 Notation

A symmetric matrix A is positive semi-definite (PSD) if all eigenvalues are non-negative. We use Δ_+^d to represent the PSD-cone, which is the subset of $d \times d$ symmetric matrices that are PSD.

For a matrix A we use $\|A\|_p$ to denote the Schatten p-norm, which is the ℓ_p norm of the vector of singular values of A. The Frobenius norm will play a special role in several places, so we sometimes use the notation $\|A\|_F$ to emphasize this. Additionally, $\|A\|$ without the subscript indicates operator norm (which is equivalent to $\|A\|_\infty$).

We always use d to indicate the dimension of the matrix being tested, and use $\epsilon < 1$ to indicate the parameter in Definition 3.1.

When applied to vectors, $\langle \cdot, \cdot \rangle$ indicates the standard inner product on $\mathbb{R}^n$. When applied to matrices, it indicates the Frobenius inner product $\langle X, Y \rangle := \mathrm{Tr}(X^T Y)$.

S^{d-1} indicates the set of all unit vectors in $\mathbb{R}^d$.

We use the notation $X^\dagger$ to indicate the Moore-Penrose pseudoinverse of X.

For a symmetric matrix $A \in \mathbb{R}^{d \times d}$ with eigenvalues $\lambda_1 \geq \lambda_2 \geq \ldots \geq \lambda_d$, we let A_k denote the matrix A with all but the top k eigenvalues zeroed out. Formally, if U is an orthogonal matrix diagonalizing A, then $A_k = U^T \operatorname{diag}(\lambda_1, \ldots, \lambda_k, 0, \ldots, 0) U$. We also let $A_{-k} = A - A_k$.

Throughout, we use c to indicate an absolute constant. The value of c may change between instances.

3.3 Vector-matrix-vector queries

3.3.1 An optimal one-sided tester.

To construct our vector-matrix-vector tester, we analyze the iteration

$$x^{(k+1)} = x^{(k)} - \eta \left((g^{(k)})^T A x^{(k)} \right) g^{(k)} = \left(I - \eta g^{(k)} (g^{(k)})^T A \right) x^{(k)}, \tag{3.2}$$

where $g^{(k)} \sim \mathcal{N}(0, I_d)$ and $x^{(0)} \sim \mathcal{N}(0, I_d)$.

Our algorithm is essentially to run this scheme for a fixed number of iterations with with well-chosen step size η. If the value of $(x^{(k)})^T A x(k)$ ever becomes negative, then we output False, otherwise we output True. Using this approach we prove the following.

Theorem 3.2. *There exists a one-sided adaptive ℓ_1-tester, that makes $O(\frac{1}{\epsilon}\log^3\frac{1}{\epsilon})$ vector-matrix-vector queries to A.*

As an immediate corollary we obtain a bound for ℓ_p-testers.

Corollary 3.3. *There is a one-sided adaptive ℓ_p-tester that makes $O(\frac{1}{\epsilon}d^{1-1/p}\log^3(\frac{1}{\epsilon}d^{1-1/p}))$ vector-matrix-vector queries.*

Proof. This follows from the previous result along with the bound $\|A\|_p \geq d^{1/p-1}\|A\|_1$. □

We now turn to the proof of Theorem 3.2. Since our iterative scheme is rotation-invariant, we assume without loss of generality that $A = \operatorname{diag}(\lambda_1, \ldots, \lambda_d)$. For now, we assume that $\|A\|_1 \leq 1$, and that the smallest eigenvalue of A is $\lambda_1 = -\epsilon$. We consider running the algorithm for N iterations. We will show that our iteration finds an x with $x^T Ax < 0$ in $N = \tilde{O}(1/\epsilon)$ iterations. We will use c to denote absolute constants that we don't track, and that may vary between uses.

Our key technical lemma is to show that the first coordinate (which is associated to the $-\epsilon$ eigenvalue) grows fairly quickly with good probability.

Lemma 3.4. *Suppose η and N satisfy the following list of assumptions: (1) $\eta \leq \frac{1}{4}$, (2) $\eta^2\epsilon N \leq \frac{1}{8}$, (3) $(1+\eta^2\epsilon^2)^N \leq \frac{5}{4}$, (4) $(1+\eta\epsilon)^N \geq \frac{10}{\epsilon^2}$. Then $x_1^{(N)} \geq \frac{1}{\epsilon^2}$ with probability at least 0.2.*

Proof. Following [Jai+16] we define the matrix $B_k = \prod_{i=1}^k \left(I - \eta g^{(i)}(g^{(i)})^T A\right)$, where the $g^{(i)}$ are independent $\mathcal{N}(0, I)$ gaussians. Note that $x^{(k)} = B_k x^{(0)}$. We will show that $B_k^T e_1$ has large norm with good probability (in fact we will show that $\left\langle B_k^T e_1, e_1\right\rangle$ is large). This will then imply that $\left\langle B_k x^{(0)}, e_1\right\rangle$ is large with high probability, where $x^{(0)} \sim \mathcal{N}(0, I)$.

Step 1: Deriving a recurrence for the second moments.

Let $y^{(k)} = B_k^T e_1$ and let $u_i^{(k)}$ be the second moment of the coordinate $y_i^{(k)}$. Note that $u_i^{(0)} = \delta_{1i}$ (where δ is the Dirac delta). To simplify the notation, we drop the superscript on the g. We compute $y_i^{(k+1)} = \left((I - \eta g g^T A)y^{(k)}\right)_i = y_i^{(k)} - \eta(Ag)_i(g_1 y_1^{(k)} + \ldots + g_d y_d^{(k)}) = y_i^{(k)} - \eta\lambda_i g_i(g_1 y_1^{(k)} + \ldots + g_d y_d^{(k)})$.

Next we observe that (after grouping terms) the coefficients of the $y_i^{(k)}$ terms are pairwise uncorrelated. Using this, along with the fact that the g_i's are independent of the $y_i^{(k)}$'s gives

$$u_i^{(k+1)} = \mathbb{E}(1-\eta\lambda_i g_i^2)^2 u_i^{(k)} + \eta^2\lambda_i^2 \sum_{j\neq i} u_j^{(k)} = (1-2\eta\lambda_i+3\eta^2\lambda_i^2)u_i^{(k)} + \eta^2\lambda_i^2\sum_{j\neq i} u_j^{(k)}$$
$$= (1-2\eta\lambda_i+2\eta^2\lambda_i^2)u_i^{(k)} + \eta^2\lambda_i^2\sum_{j=1}^{d} u_j^{(k)}.$$

Let $S^{(k)} = u_1^{(k)} + \ldots + u_d^{(k)}$, and $\gamma_i = 1-2\eta\lambda_i+2\eta^2\lambda_i^2$. Then we can write the recurrence as $u_i^{(k+1)} = \gamma_i u_i^{(k)} + \eta^2\lambda_i^2 S^{(k)}$. Iterating this recurrence gives

$$u_i^{(k)} = \delta_{1i}\gamma_i^k + \eta^2\lambda_i^2\left(\gamma_i^{k-1}S^{(0)} + \gamma_i^{k-2}S^{(1)} + \ldots + S^{(k-1)}\right). \tag{3.3}$$

Step 2: Bounding $S^{(k)}$.

Summing the above equation over i allows us to write a recurrence for the $S^{(k)}$'s: $S^{(k)} = \gamma_1^k + \alpha_{k-1}S^{(0)} + \alpha_{k-2}S^{(1)} + \ldots + \alpha_0 S^{(k-1)}$, where we define $\alpha_j := \sum_{i=1}^d \eta^2\lambda_i^2\gamma_i^j$.

We split α_j into two parts, α_j^+ and α_j^- corresponding to terms in the sum where λ_i is positive or negative respectively. We now use the recurrence to bound $S^{(k)}$. First by Holder's inequality, $S^{(k)} \leq \gamma_1^k + \max(S^{(0)},\ldots,S^{(k-1)})(\alpha_0^+ + \ldots + \alpha_{k-1}^+) + (\alpha_{k-1}^- S^{(0)} + \alpha_{k-2}^- S^{(1)} + \ldots + \alpha_0^- S^{(k-1)})$.

We calculate

$$\sum_{j=0}^{k-1}\alpha_j^+ = \sum_{j=0}^{k-1}\sum_{i:\lambda_i>0}\eta^2\lambda_i^2\gamma_i^j = \sum_{i:\lambda_i>0}\eta^2\lambda_i^2\sum_{j=0}^{k-1}\gamma_i^j = \sum_{i:\lambda_i>0}\eta^2\lambda_i^2\frac{1-\gamma_i^k}{1-\gamma_i}$$
$$= \sum_{i:\lambda_i>0}\eta^2\lambda_i^2\frac{1-\gamma_i^k}{2\eta\lambda_i-2\eta^2\lambda_i^2} = \sum_{i:\lambda_i>0}\eta\lambda_i\frac{1-\gamma_i^k}{2-2\eta\lambda_i} \leq \sum_{i:\lambda_i>0}\eta\lambda_i \leq \eta,$$

where we used that $\eta\lambda_i \leq 1/2$, (which is a consequence of Assumption 1), that $\gamma_i < 1$ (which holds since $\lambda_i > 0$) and that $\sum_{i:\lambda_i>0} \lambda_i \leq 1$. Since we assume that $-\epsilon$ is the smallest eigenvalue,

$$\alpha_j^- \leq \eta^2 \gamma_1^j \sum_{i:\lambda_i<0} \lambda_i^2 \leq \eta^2 \gamma_1^j \epsilon \sum_{i:\lambda_i<0} |\lambda_i| \leq \eta^2 \gamma_1^j \epsilon.$$

Let $\tilde{S}^{(k)} = \max(S^{(0)}, \ldots S^{(k)})$. Then combining our bounds gives

$$\tilde{S}^{(k)} \leq \max\left(\tilde{S}^{(k-1)}, \gamma_1^k + \eta \tilde{S}^{(k-1)} + \eta^2 \epsilon (\gamma_1^{k-1} \tilde{S}^{(0)} + \gamma_1^{k-2} \tilde{S}^{(1)} + \ldots + \tilde{S}^{(k-1)})\right).$$

The next step is to use this recurrence to bound $\tilde{S}^{(k)}$. For this, define $c^{(k)}$ such that $\tilde{S}^{(k)} = c^{(k)} \gamma_1^k$. Plugging in to the above and dividing through by γ_1^k, we get that $c^{(k)}$ satisfies

$$\begin{aligned} c^{(k)} &\leq \max\left(\frac{c^{(k-1)}}{\gamma_1}, 1 + \frac{\eta}{\gamma_1} c^{(k-1)} + \frac{\eta^2 \epsilon}{\gamma_1}(c^{(0)} + \ldots + c^{(k-1)})\right) \\ &\leq \max\left(c^{(k-1)}, 1 + \eta c^{(k-1)} + \eta^2 \epsilon (c^{(0)} + \ldots + c^{(k-1)})\right), \end{aligned}$$

where we used the fact that $\gamma_1 \geq 1$. Now set $\tilde{c}^{(k)} = \max(c^{(0)}, \ldots c^{(k)})$. By assumptions 1 and 2, $\eta + \eta^2 \epsilon k \leq 1/2$. This gives

$$\tilde{c}^{(k)} \leq \max\left(\tilde{c}^{(k-1)}, 1 + \eta \tilde{c}^{(k-1)} + \eta^2 \epsilon k \tilde{c}^{(k-1)}\right) \leq \max\left(\tilde{c}^{(k-1)}, 1 + \frac{1}{2} \tilde{c}^{(k-1)}\right).$$

Note that $c^{(0)} = S^{(0)} = 1$, so a straightforward induction using the above recurrence shows that $\tilde{c}^{(k)} \leq 2$ for all k. It follows that $S^{(k)} \leq 2\gamma_1^k$.

Step 3: Bounding the second moment. Plugging the bound above in to (3.3) gives

$$u_1^{(k)} \leq \gamma_1^k + 2k\eta^2 \epsilon^2 \gamma_1^{k-1} \leq \left(1 + 2k\eta^2 \epsilon^2\right) \gamma_1^k.$$

Step 4: Applying Chebyshev. We focus on the first coordinate, $y_1^{(k)}$. Note that $I - \eta A g g^T$ has expectation $I - \eta A$, so a straightforward induction shows that $\mathbb{E} y_1^{(k)} = (1 + \eta \epsilon)^k$.

Using the bound for the second moment of the first coordinate, we get $\frac{u_1^{(k)}}{\left(\mathbb{E}y_1^{(k)}\right)^2} \leq \frac{(1+2k\eta^2\epsilon^2)\gamma_1^k}{(1+\eta\epsilon)^{2k}} = (1+2k\eta^2\epsilon^2)\left(\frac{1+2\eta\epsilon+2\eta^2\epsilon^2}{1+2\eta\epsilon+\eta^2\epsilon^2}\right)^k = (1+2k\eta^2\epsilon^2)\left(1+\frac{\eta^2\epsilon^2}{1+2\eta\epsilon+\eta^2\epsilon^2}\right)^k \leq (1+2k\eta^2\epsilon^2)(1+\eta^2\epsilon^2)^k$.

By Assumptions 2 and 4, $N\eta^2\epsilon^2 \leq 1/8$ and $(1+\eta^2\epsilon^2)^N \leq 5/4$, so we get that $u_1^{(k)} \leq 25/16\left(\mathbb{E}u_1^{(k)}\right)^2$.

Thus by Chebyshev's inequality, $\mathbb{P}\left(\left|y_1^{(k)} - \mathbb{E}(y_1^{(k)})\right| \geq 0.9\mathbb{E}(y_1^{(k)})\right) \leq \frac{25}{36}$. So with probability at least 0.3, $y_1^{(N)} \geq \frac{1}{10}\mathbb{E}(y_1^{(N)}) = \frac{1}{10}(1+\eta\epsilon)^N$.

Under assumption 4, $(1+\eta\epsilon)^N \geq \frac{10}{\epsilon^2}$, which means that $y_1^{(N)} \geq \frac{1}{\epsilon^2}$ with at least 0.3 probability.

Step 5: Concluding the argument. We showed that $\left\langle B_N^T e_1, e_1\right\rangle \geq \frac{1}{\epsilon^2}$ with probability at least 0.3. In particular this implies that $\left\|B_N^T e_1\right\| \geq \frac{1}{\epsilon^2}$. Now since $x^{(0)}$ is distributed as $\mathcal{N}(0, I)$, $\left\langle B_N x^{(0)}, e_1\right\rangle = \left\langle x^{(0)}, B_N^T e_1\right\rangle \sim \mathcal{N}(0, \left\|B_N^T e_1\right\|^2)$, which is at least $\left\|B_N^T e_1\right\|$ in magnitude with 0.67 probability. It follows that $x_1^{(N)} \geq \frac{1}{\epsilon^2}$ with probability at least 0.2. □

Let $f(x) = x^T A x$. We next understand how the value of $f(x^{(k)})$ is updated on each iteration.

Proposition 3.5. *For $g \sim \mathcal{N}(0,1)$, we have $f(x^{(k)}) - f(x^{(k+1)}) = \eta(g^T A x^{(k)})^2(2 - \eta g^T A g)$.*

Proof. Plugging in the update rule and expanding gives

$$\begin{aligned} f(x^{(k+1)}) &= (x^{(k)})^T A x^{(k)} - 2\eta(g^T A x^{(k)})^2 + \eta^2(g^T A x^{(k)})^2 g^T A g \\ &= (x^{(k)})^T A x^{(k)} - \eta(g^T A x^{(k)})^2(2 - \eta g^T A g), \end{aligned}$$

from which the proposition follows. □

A consequence of this update is that the sequence $f(x^{(k)})$ is almost guaranteed to be decreasing as long as η is chosen small enough.

Proposition 3.6. *Assume that $\mathrm{Tr}(A) \leq 1$ and that $\eta < c$. After N iterations, $f(x^{(N)}) \leq f(x^{(0)})$ with probability at least $99/100$ provided that $\eta \leq \frac{c}{\log N+1}$.*

Proof. We show something stronger; namely that for the first N iterations, the sequence $f(x^{(k)})$ is decreasing. By Proposition 3.5, $f(x^{(k+1)}) \leq f(x^{(k)})$ as long as $g^T A g \leq \frac{2}{\eta}$. The probability that this does not occur is $\Pr\left(\sum \lambda_i g_i^2 \geq \frac{2}{\eta}\right) \leq \Pr\left(\sum \lambda_i (g_i^2 - 1) \geq \frac{2}{\eta} - 1\right)$.

The $g_i^2 - 1$ terms are independent subexponential random variables. So by Bernstein's inequality (see [Ver18] Theorem 2.8.2 for the version used here), this probability is bounded by $2\exp(-c/\eta)$ as long as η is a sufficiently small constant. Taking a union bound gives that $f(x^{(N)}) \leq f(x^{(0)})$ with probability at least $1 - 2N\exp(-c/\eta)$, which is at least $99/100$ under the conditions given. □

Theorem 3.7. *Suppose that $\|A\|_1 \leq 1$, $\epsilon < 1/2$, and that A has $-\epsilon$ as an eigenvalue. If we take η such that $c\epsilon^2 \leq \eta \leq \min\left(\frac{1}{32\log(10/\epsilon^2)}, \frac{c}{\log\frac{1}{\epsilon}}\right)$, then for some $N = \Theta\left(\frac{1}{\epsilon\eta}\log\frac{1}{\epsilon}\right)$ we have $f(x^{(N)}) < 0$ with at least $1/10$ probability.*

Proof. Given an η as in the statement of the theorem, choose $N = \left\lceil \frac{2}{\eta\epsilon}\log\frac{10}{\epsilon^2}\right\rceil$, which satisfies the assumptions of Lemma 3.4. Then $x_1^{(N)} \geq \frac{1}{\epsilon^2}$ with probability at least 0.2. By proposition 3.6, $f(x^{(N)}) \leq f(x^{(0)}) \leq 2$ with at least 0.99 probability, using the fact that $\eta \leq \frac{c}{\log\frac{1}{\epsilon}}$ for an appropriately chosen absolute constant c, such that the hypothesis of Proposition 3.6 holds.

If $f(x^{(N)}) < 0$, then the algorithm has already terminated. Otherwise conditioned on the events in the above paragraph, we have with at least 0.8 probability that $2 - \eta(g^{(N)})^T A g^{(N)} \geq \frac{1}{2}$ and

$$(g^{(N)})^T A x^{(N)} \sim \mathcal{N}\left(0, \|Ax^{(N)}\|^2\right) \geq \frac{1}{3}\left\|Ax^{(N)}\right\| \geq \frac{1}{3}\lambda_1 x_1^{(N)} \geq \frac{1}{3\epsilon^2}\lambda_1 \geq \frac{1}{3\epsilon}.$$

Then by Proposition 3.5 it follows that $f(x^{(N+1)}) \leq f(x^{(N)}) - \frac{\eta}{20\epsilon^2} \leq 2 - \frac{\eta}{20\epsilon^2} < 0$. □

We also observe that we can reduce the dimension of the problem by using a result of Andoni and Nguyen. This allows us to avoid a $\log d$ dependence.

Proposition 3.8. *Suppose that A satisfies $\lambda_{\min}(A) < -\alpha\|A\|_1$, and let $G \in \mathbb{R}^{d\times m}$ have independent $\mathcal{N}(0, \frac{1}{d})$ entries. Then we can choose $m = O(1/\alpha)$ such that $\lambda_{\min}(G^T A G) < -\alpha/2$ and $\left\|G^T A G\right\|_1 \leq 2\|A\|_1$.*

Proof. For the first claim, we simply apply Theorem 1.1 in [AN13] and (in their notation) set $\epsilon = O(1)$ and $k = O(1/\alpha)$.

To show that the Schatten 1-norm does not grow too much under the sketch, we first write $A = A_+ + A_-$ where the nonzero eigenvalues of A_+ are exactly the positive eigenvalues of A. Then using the usual analysis of Hutchinson's trace estimator (see [Mey+21] for example), we have

$$\begin{aligned} \left\|G^T A G\right\|_1 &\leq \left\|G^T A_+ G\right\|_1 + \left\|G^T A_- G\right\|_1 = \operatorname{Tr}(G^T A_+ G) + \operatorname{Tr}(G^T A_- G) \\ &= (1 \pm O(1/\sqrt{m}))\operatorname{Tr}(A_+) + (1 \pm O(1/\sqrt{m}))\operatorname{Tr}(A_-) \\ &\leq (1 \pm O(1/\sqrt{m})\left\|A\right\|_1 . \end{aligned}$$

□

We are now ready to give the proof of Theorem 3.2.

Proof. The above result applies after scaling the η given in Theorem 3.7 by $1/\left\|A\right\|_1$. So it suffices to choose η to be bounded above by

$$\frac{1}{\left\|A\right\|_1} \min\left(\frac{1}{32\log(10/\epsilon^2)}, \frac{c}{\log\frac{1}{\epsilon}}\right),$$

and within a constant factor of this value.

To choose an η, pick a standard normal g, and compute Ag using $1/\epsilon$ vector-matrix-vector queries. Then with constant probability, $\lambda_{\max}(A) \leq \|Ag\| \leq 2d\lambda_{\max}$. Given this, we have

$$d\left\|Ag\right\| \geq \left\|A\right\|_1 \geq \frac{\left\|Ag\right\|}{2d}, \tag{3.4}$$

which allows us to approximate $\left\|A\right\|_1$ to within a factor of d^2 with constant probability. Given this, one may simply try the above algorithm with an η at each of $O(\log(d^2)) = O(\log d)$ different scales, with the cost of an extra $\log d$ factor.

Finally, we may improve the $\log d$ factor to a $\log(1/\epsilon)$ factor by using Proposition 3.8 to sketch A, and then applying the above analysis to G^TAG. Note that the sketch may be used implicitly; once G is chosen, a vector-matrix-vector query to G^TAG can be simulated with a single vector-matrix-vector query to A. □

3.3.2 Lower bounds

We will show a bound for two-sided testers which will imply that the bound for ℓ_1-testers given in Theorem 3.2 is tight up to $\log$ factors. If we require the tester to have one-sided error, then we additionally show that the bound in Corollary 3.3 is tight for all p. Note that this distinction between one-sided and two-sided testers is necessary given Theorem 3.29.

In order to obtain these lower bounds for adaptive testers, we first show corresponding lower bounds for non-adaptive testers. A minor modification to Lemma 3.1 in [Sun+19] shows that an adaptive tester can have at most quadratic improvement over a non-adaptive tester. This will allow us to obtain our adaptive lower bounds as a consequence of the non-adaptive bounds.

Non-adaptive lower bounds

We first observe that a one-sided tester must always be able to produce a witness matrix X, that at least certifies that A is not positive definite.

Proposition 3.9. *If a one-sided tester makes a series of symmetric linear measurements $\langle M_i, A\rangle$ of A, and outputs False on a given instance, then there must exist nonzero $X \in \operatorname{span}(M_1, \dots, M_k)$ such that X is PSD and $\langle X, A\rangle \leq 0$.*

Proof. We work within the space of symmetric matrices. Let $W = \operatorname{span}(M_1, \dots, M_k)$, and let $\phi(X) = \langle A, X\rangle$ be the linear functional associated with A. Now suppose that ϕ is strictly positive for all nonzero $X \in W \cap \Delta_+^d$. We will construct $\widetilde{\phi}$ that agrees with ϕ on W and is non-negative on Δ_+^d.

Let $W' = \ker(\phi) \cap W$, and note that $W' \cap \Delta_+^d = \{0\}$. Now by convexity of Δ_+^d, there exists a hyperplane H and associated half-space H^+ such that (i)H contains W' (ii) $H \cap \Delta_+^d = \{0\}$, (iii) $H^+ \supseteq \Delta_+^d$ and (iv) ϕ is non-negative on $H^+ \cap W$. Moreover, since W' intersects Δ_+^d trivially, H can be chosen such that $H \cap W = W'$. Now let Π be a projection onto W that maps H to W', and choose $\widetilde{\phi} = \phi \circ \Pi_W$.

The linear functional $\widetilde{\phi}$ is represented by the inner product against some symmetric matrix B. By construction of $\widetilde{\phi}$, we have $\langle B, M_i \rangle = \langle A, M_i \rangle$ for all i, and also $\langle B, X \rangle \geq 0$ for all PSD X. So in particular $\left\langle B, xx^T \right\rangle = x^T Bx \geq 0$ for all x, which implies that B is PSD. Given the existence of the PSD matrix B consistent with all measurements, the one-sided tester must not reject.

□

We are now able to give an explicit non-PSD spectrum which is hard for any one-sided tester. Specifically, we show that it is hard for any vector-matrix-vector query algorithm to produce a witness X in the sense of the proposition above.

Theorem 3.10. *Let $\lambda > 0$ and suppose for all matrices A with spectrum $(-\lambda, 1, \ldots, 1)$ that a non-adaptive one-sided tester $\mathcal{T}$ outputs False with $2/3$ probability. Then $\mathcal{T}$ must make at least $\frac{1}{9}\left(\frac{d}{1+\lambda}\right)^2$ vector-matrix-vector queries.*

Proof. By the polarization identity,

$$x^T Ay = \frac{1}{2}\left((x+y)^T A(x+y) - y^T Ay - x^T Ax\right),$$

we may assume that all queries are of the form $x_i^T A x_i$, at the cost requiring at most a factor of three increase in the number of queries.

We set $A = I - (1+\lambda)vv^T$ where v is uniform over S^{d-1}, and let $W = \operatorname{span}(x_1x_1^T, \ldots x_kx_k^T)$. By Proposition 3.9, the tester may only reject if there is an X in $W \cap \Delta_+^d$ with $\|X\|_F = 1$ such that $\langle X, A\rangle \leq 0$. For such an X we have

$$\left\langle vv^T, X\right\rangle \geq \frac{\text{Tr}(X)}{1+\lambda} \geq \frac{1}{1+\lambda}. \tag{3.5}$$

But since vv^T and X both have unit norm and $X \in W$, this condition implies that $\left\|\Pi_W(vv^T)\right\|_F \geq \frac{1}{1+\lambda}$.

Now we turn to understanding $\mathbb{E}(\left\|\Pi_W(vv^T)\right\|_F^2)$. Indeed we have the following:

Lemma 3.11. *Let v be drawn uniformly from S^{d-1}, and let W be a k-dimensional subspace of the $d \times d$ symmetric matrices. Let $\alpha_4 = \mathbb{E}(v_1^4)$ and $\alpha_{22} = E(v_1^2 v_2^2)$. Then*

$$\mathbb{E}(\left\|\Pi_W(vv^T)\right\|_F^2) = (\alpha_4 - \alpha_{22})k + \alpha_{22}\left\|\Pi_W(I)\right\|_F^2,$$

where I is the identity matrix.

Proof. Let $M_1, \dots M_k$ be an orthonormal basis for W. By the Pythagorean theorem,

$$\mathbb{E}(\left\|\Pi_W(vv^T)\right\|_F^2) = \sum_{i=1}^{k} \mathbb{E}(\left\|\Pi_{M_i}(vv^T)\right\|_F^2). \tag{3.6}$$

For fixed M we have

$$\mathbb{E}(\left\|\Pi_M(vv^T)\right\|_F^2) = \mathbb{E}(\left\langle vv^T, M\right\rangle^2) = \mathbb{E}(\text{Tr}(vv^T M)^2) = \mathbb{E}((v^T M v)^2). \tag{3.7}$$

Since M is symmetric, we can diagonalize M to $D = \text{diag}(a_1, \dots, a_d)$ in some orthonormal basis. Since M has unit norm, $a_1^2 + \ldots + a_d^2 = 1$. Then we have

$$\begin{aligned}\mathbb{E}((v^T M v)^2) &= \mathbb{E}((v^T D v)^2)\\ &= \mathbb{E}((a_1 x_1^2 + \ldots + a_d x_d^2)^2)\\ &= \alpha_4(a_1^2 + \ldots + a_d^2) + 2\alpha_{22}\sum_{i<j} a_i a_j\\ &= \alpha_4 + 2\alpha_{22}\sum_{i<j} a_i a_j.\end{aligned}$$

Next observe that

$$\mathrm{Tr}(M)^2 = (a_1 + \ldots + a_d)^2 = a_1^2 + \ldots + a_d^2 + 2\sum_{i<j} a_i a_j = 1 + 2\sum_{i<j} a_i a_j,$$

so that

$$\mathbb{E}((v^T M v)^2) = \alpha_4 + \alpha_{22}(\mathrm{Tr}(M)^2 - 1).$$

Combining with (3.6) gives

$$\begin{aligned}\mathbb{E}(\left\|\Pi_W(vv^T)\right\|_F^2) &= \sum_{i=1}^{k}\left(\alpha_4 + \alpha_{22}(\mathrm{Tr}(M_i)^2 - 1)\right)\\ &= (\alpha_4 - \alpha_{22})k + \alpha_{22}\sum_{i=1}^{k}\mathrm{Tr}(M_i)^2.\end{aligned}$$

Finally, observe that $\sum_{i=1}^k \mathrm{Tr}(M_i)^2 = \sum_{i=1}^k \langle I, M_i\rangle^2 = \|\Pi_W(I)\|_F^2$, by the Pythagorean theorem, which finishes the proof. □

Remark 3.12. *While approximations would suffice, this result gives a quick way to compute α_4 and α_{22}. Set W to be the entire space of $n \times n$ symmetric matrices, and $k = d(d+1)/2$. The previous result gives*

$$1 = \frac{d(d+1)}{2}(\alpha_4 - \alpha_{22}) + d\alpha_{22}.$$

On the other hand, by expanding $1 = (v_1^2 + \ldots + v_d^2)^2$, *we have*

$$1 = d\alpha_4 + d(d-1)\alpha_{22}.$$

Solving the system yields $\alpha_4 = \frac{3}{d(d+2)}$ *and* $\alpha_{22} = \frac{1}{d(d+2)}$.

To finish the proof of the theorem, we recall that W is spanned by the matrices $x_1 x_1^T, \ldots, x_k x_k^T$, each of which has rank one. Therefore each matrix in W, and in particular $\Pi_W(I)$, has rank at most k.

We recall for a general matrix A, that $\operatorname{argmin}_{\mathrm{rk}(U) \leq k} \|A - U\|_F$ is gotten by truncating all but the largest k singular values of A. Applying this to the identity matrix, when $k \leq d$, we see that

$$\|\Pi_W(I)\|_F^2 = \|I\|_F^2 - \|\Pi_{W^\perp}(I)\|_F \leq d - (d-k) = k,$$

since $\|\Pi_{W^\perp}(I)\|_F^2 \geq \min_{\mathrm{rk}(U) \leq k} \|I - U\|_F^2 = d - k$. Since $\|I\|_F^2 = d$, we always have $\|\Pi_W(I)\|_F^2 \leq k$.

Combining this fact with Lemma 3.11 gives

$$\mathbb{E}(\left\|\Pi_W(vv^T)\right\|_F^2) \leq (\alpha_4 - \alpha_{22})k + \alpha_{22}k = k\alpha_4 = \frac{3k}{d(d+2)} \leq \frac{3k}{d^2},$$

and by Markov's inequality,

$$\mathbb{P}\left(\left\|\Pi_W(vv^T)\right\|_F^2 > \frac{9k}{d^2}\right) \leq \frac{1}{3}.$$

So with probability $2/3$, $\left\|\Pi_W(vv^T)\right\|_F^2 \leq \frac{9k}{d^2}$. But for $\mathcal{A}$ to be correct, we saw that we must have $\left\|\Pi_W(vv^T)\right\|_F \geq \frac{1}{1+\lambda}$ with probability $2/3$. It follows that

$$\left(\frac{1}{1+\lambda}\right)^2 \leq \frac{9k}{d^2},$$

which implies that

$$k \geq \frac{1}{9}\left(\frac{d}{1+\lambda}\right)^2.$$

□

In particular, this result implies that for non-adaptive one-sided testers, a poly$(1/\epsilon)$ ℓ_p-tester can only exist for $p = 1$.

Theorem 3.13. *A one-sided non-adaptive ℓ_p-tester must make at least $\Omega(\frac{1}{\epsilon^2}d^{2-2/p})$ vector-matrix-vector queries.*

Proof. This follows as a corollary of Theorem 3.10; simply apply that result to the spectrum $(\epsilon(d-1)^{1/p}, 1\ldots, 1)$ where there are $d-1$ 1's. □

Adaptive lower bounds

As remarked earlier, our adaptive lower bounds follow as a corollary of our non-adaptive bounds, and a slightly modified version of Lemma 3.1 in [Sun+19], which we give here.

Lemma 3.14. *Let $A = X\Sigma X^T$ be a random symmetric $d \times d$ real-valued matrix, with Σ diagonal, and where X is orthonormal and sampled from the rotationally invariant distribution. Any s adaptive vector-matrix-vector queries to A may be simulated by $O(s^2)$ non-adaptive vector-matrix-vector queries.*

Proof. (Sketch) First note that the adaptive protocol may be simulated by $3s$ adaptive quadratic form queries, of the form $x^T Ax$ by the polarization identity

$$x^T Ay = \frac{1}{2}\left((x+y)^T A(x+y) - x^T Ax - y^T Ay\right). \tag{3.8}$$

These queries in turn may be simulated by $9s^2$ non-adaptive queries by following exactly the same proof as Lemma 3.1 in [Sun+19] (but now with $u_i = v_i$ in their proof). □

As a direct consequence of this fact and our Theorem 3.13 we obtain the following.

Theorem 3.15. *An adaptive one-sided ℓ_p-tester must make at least $\Omega(\frac{1}{\epsilon}d^{1-1/p})$ vector-matrix-vector queries.*

3.4 Adaptive matrix-vector queries

We analyze random Krylov iteration. Namely we begin with a random $g \sim \mathcal{N}(0, I_d)$ and construct the sequence of iterates $g, Ag, A^2g, \dots A^kg$ using k adaptive matrix-vector queries. The span of these vectors is denoted $\mathcal{K}_k(g)$ and referred to as the k^{th} Krylov subspace.

Krylov iteration suggests a very simple algorithm. First compute $g, Ag, \dots, A^{k+1}g$. If $\mathcal{K}_k(g)$ contains a vector v such that $v^TAv < 0$ then output False, otherwise output True. (Note that one can compute Av and hence v^TAv for all such v, given the $k+1$ matrix-vector queries.) We show that this simple algorithm is in fact optimal.

As a point of implementation, we note that the above condition on $\mathcal{K}_k(g)$ can be checked algorthmically. One first uses Gram-Schmidt to compute the projection Π onto $\mathcal{K}_k(g)$. The existence of a $v \in \mathcal{K}_k(g)$ with $v^TAb < 0$ is equivalent to the condition $\lambda_{\min}(\Pi A \Pi) < 0$. When A is ϵ-far from PSD, the proof below will show that in fact $\lambda_{\min}(\Pi A \Pi) < -\Omega(\epsilon) \left\|A\right\|_p$, so it suffices to estimate $\lambda_{\min}(\Pi A \Pi)$ to within $O(\epsilon) \left\|A\right\|_p$ accuracy.

Proposition 3.16. *For $r > 0$, $\alpha > 0$ and $\delta > 0$ there exists a polynomial p of degree $O(\frac{\sqrt{r}}{\sqrt{\alpha}} \log \frac{1}{\delta})$, such that $p(-\alpha) = 1$ and $|p(x)| \leq \delta$ for all $x \in [0, r]$.*

Proof. Recall that the degree d Chebyshev polynomial T_d is bounded by 1 in absolute value on $[-1, 1]$ and satisfies

$$T_d(1 + \gamma \geq 2^{d\sqrt{\gamma}-1}).$$

(See [MM15] for example.) The proposition follows by shifting and scaling T_d. □

Theorem 3.17. *Suppose that A has an eigenvalue $\lambda_{\min}$ with $\lambda_{\min} \leq -\epsilon \left\|A\right\|_p$. When $p = 1$, the Krylov subspace $\mathcal{K}_k(g)$ contains a vector v with $v^TAv < 0$ for $k = O\left(\left(\frac{1}{\epsilon}\right)^{\frac{1}{3}} \log \frac{1}{\epsilon}\right)$. When $p \in (1, \infty]$, the same conclusion holds for $k = O\left(\left(\frac{1}{\epsilon}\right)^{\frac{p}{2p+1}} \log \frac{1}{\epsilon} \log d\right)$.*

Proof. Without loss of generality, assume that $\|A\|_p \leq 1$. Fix a value T to be determined later, effectively corresponding to the number of top eigenvalues that we deflate. By Proposition 3.16 we can construct a polynomial q, such that $q(\lambda_{\min}) = 1$ and $|q(x)| \leq \sqrt{\frac{\epsilon/10}{d^{1-1/p}}}$ for $x \in [0, T^{-1/p}]$ with

$$\deg(q) \leq C\frac{T^{-1/(2p)}}{\sqrt{\epsilon}} \log\left(\sqrt{\frac{d^{1-1/p}}{\epsilon/10}}\right), \tag{3.9}$$

where C is an absolute constant.

Now set

$$p(x) = q(x) \prod_{i:\lambda_i > T^{-1/p}} \frac{\lambda_i - x}{\lambda_i - \lambda_{\min}}. \tag{3.10}$$

Since we assume $\|A\|_p \leq 1$, there at most T terms in the product, so

$$\deg(p) \leq T + C\frac{T^{-1/(2p)}}{\sqrt{\epsilon}} \log\left(\sqrt{\frac{d^{1-1/p}}{\epsilon/10}}\right). \tag{3.11}$$

By setting $T = \epsilon^{-p/(2p+1)}$, we get

$$\deg(p) = \begin{cases} O\left(\left(\frac{1}{\epsilon}\right)^{\frac{p}{2p+1}} \log\frac{1}{\epsilon}\right) & \text{if } p = 1 \\ O\left(\left(\frac{1}{\epsilon}\right)^{\frac{p}{2p+1}} \log\frac{1}{\epsilon}\log d\right) & \text{if } p > 1 \end{cases} \tag{3.12}$$

As long as k is at least $\deg(p)$, then $v = p(A)g$ lies in $\mathcal{K}_k(g)$, and

$$v^T A v = g^T p(A)^2 A g. \tag{3.13}$$

By construction, $p(\lambda_{\min}) = 1$. Also for all x in $[0, T^{-1/p}]$, $|p(x)| \leq |q(x)| \leq \sqrt{\epsilon/10} d^{(1/p)-1}$.

Therefore the matrix $p(A)^2 A$ has at least one eigenvalue less than $-\epsilon$, and the positive eigenvalues sum to at most

$$\sum_{i:\lambda_i > 0} \frac{\epsilon}{10} d^{1/p-1} \lambda_i \leq \frac{\epsilon}{10}, \tag{3.14}$$

by using Holder's inequality along with the fact that $\|A\|_p \leq 1$. So with at least $2/3$ probability,

$g^T p(A)^2 Ag < 0$ as desired. $\square$

Remark 3.18. *For $1 < p < \infty$, the $\log d$ dependence can be removed by simply applying the $p = 1$ tester to $A^{\lceil p \rceil}$, as a matrix-vector query to $A^{\lceil p \rceil}$ may be simulated via $\lceil p \rceil$ matrix-vector queries to A. However this comes at the cost of a $\left(\frac{1}{\epsilon}\right)^{\lceil p \rceil/3}$ dependence, and is therefore only an improvement when d is extremely large.*

Remark 3.19. *While we observe that deflation of the top eigenvalues can be carried out implicitly within the Krylov space, this can also be done explicitly using block Krylov iteration, along with the guarantee given in Theorem 1 of [MM15].*

We showed above that we could improve upon the usual analysis of Krylov iteration in our context. We next establish a matching lower bound that shows our analysis is tight up to $\log$ factors. This is a corollary of the proof of Theorem 3.1 presented in [Bra+20].

Theorem 3.20. *A two-sided, adaptive ℓ_p-tester in the matrix-vector model must in general make at least $\Omega(\frac{1}{\epsilon^{p/(2p+1)}})$ queries.*

Proof. We make use of the proof of Theorem 3.1 given in [Bra+20]. We consider an algorithm $\mathcal{A}$ that receives a matrix W sampled from the Wishart distribution makes at most $(1 - \beta)d$ queries, and outputs either True or False, depending on whether $\lambda_{\min}(W)$ is greater or less than t (where $t = 1/(2d^2)$ is defined as in [Bra+20]). We say that $\mathcal{A}$ fails on a given instance if either (i) $\mathcal{A}$ outputs True and $t - \frac{1}{4d^2} \geq \lambda_{\min}(W)$ or (ii) $\mathcal{A}$ outputs False and $\lambda_{\min}(W) \geq t + \frac{1}{4d^2}$. Exactly the same proof given in [Bra+20] shows that $\mathcal{A}$ must fail with probability at least $c_{\text{wish}}\sqrt{\beta}$ where $c_{\text{wish}} > 0$ is an absolute constant, as long as d is chosen sufficiently large depending on β. Taking $\beta = 1/4$ say, means that any such algorithm fails with probability at least $c_{\text{wish}}/2$ as long as d is a large enough constant.

Now consider an ℓ_p-tester $\mathcal{T}$ with $d = 1/(4\epsilon^{p/(2p+1)})$, applied to the random matrix $W - tI$. While our definition allows $\mathcal{T}$ to fail with $1/3$ probability we can reduce this failure probability to $c_{\text{wish}}/2$ by running a constant number of independent instances and taking a majority vote. So from here on we assume that $\mathcal{T}$ fails on a given instance with probability at most $c_{\text{wish}}/2$.

First recall that $W \sim XX^T$ where each entry of X is i.i.d. $\mathcal{N}(0, 1/d)$. Then with high probability, the operator norm of X is bounded, say, by $\sqrt{2}$, and the eigenvalues of W are bounded by 2.

Therefore with high probability, $\left\|W\right\|_p \leq 2d^{1/p}$, and so $\left\|W - tI\right\|_p \leq 3d^{1/p}$. It follows that $1/(4d^2) = 4\epsilon(4d)^{1/p} \geq \epsilon \left\|W - tI\right\|_p$. This means that $\mathcal{T}$ can solve the problem above, and by correctness of the tester, fails with at most $c_{\text{wish}}/2$ probability. For ϵ sufficiently small, the above analysis implies that $\mathcal{T}$ must make at least $\Omega(d) = \Omega(1/\epsilon^{p/(2p+1)})$ queries.

□

3.5 An optimal bilinear sketch

In this section we analyze a bilinear sketch for PSD-testing which will also yield an optimal ℓ_2-tester in the vector-matrix-vector model.

Our sketch is very simple. We choose $G \in \mathbb{R}^{d \times k}$ to have independent $\mathcal{N}(0, 1)$ entries and take our sketch to be $G^T AG$. In parallel we construct estimates α and β for the trace and Frobenius norm of A respectively, such that β is accurate to within a multiplicative error of 2, and α is accurate to with $\left\|A\right\|_F$ additive error. (Note that this may be done at the cost of increasing the sketching dimension by $O(1)$.)

If $G^T AG$ is not PSD then we automatically reject. Otherwise, we then consider the quantity

$$\gamma := \frac{\alpha - \lambda_{\min}(G^T AG)}{\beta\sqrt{k}\log k} \tag{3.15}$$

If γ is at most c_{psd} for some absolute constant c_{psd}, then the tester outputs False, otherwise it outputs True.

We first show that a large negative eigenvalue of A results causes the smallest sketched eigenvalue to be at most $\mathrm{Tr}(A) - \Omega(\epsilon k)$. On the other hand, when A is PSD, we will show that $\lambda_{\min}(G^T AG)$ is substantially larger.

3.5.1 Upper bound on $\lambda_{\min}(G^T AG)$

We start with the following result on trace estimators which we will need below.

Proposition 3.21. *Let M be a symmetric matrix with eigenvalues $\lambda_1, \ldots, \lambda_d$, and let u be a random unit vector with respect to the spherical measure. Then*

$$\mathrm{Var}(u^T Mu) = \frac{2}{d+2}\left(\frac{\lambda_1^2 + \ldots + \lambda_d^2}{d} - \frac{(\lambda_1 + \ldots + \lambda_d)^2}{d^2}\right) := \frac{2}{d+2}\mathrm{Var}(\lambda_1, \ldots, \lambda_d).$$

Proof. By the spectral theorem, it suffices to prove the result when M is diagonal. Then

$$\mathrm{Var}\, u^T Mu = \mathbb{E}(\lambda_1 u_1^2 + \ldots + \lambda_d u_d^2)^2 - \left(\mathbb{E}(\lambda_1 u_1^2 + \ldots + \lambda_d u_d^2)\right)^2.$$

By Remark 3.12, we have $\mathbb{E}(u_i^2) = 1$, $\mathbb{E}(u_i^4) = \frac{3}{d(d+2)}$ and $\mathbb{E}(u_i^2 u_j^2) = \frac{1}{d(d+2)}$ for $i \neq j$. The result follows by expanding using linearity of expectation, and then applying these facts. □

The next two results will give an upper bound on the smallest eigenvalue of the Gaussian sketch. For the proof of Lemma 3.23 we will start with random orthogonal projections, from which the Gaussian result will quickly follow. We include a technical hypothesis that essentially enforces non-negativity of $\mathrm{Tr}(A)$. We write the hypothesis in the form below simply to streamline the argument.

Lemma 3.22. *Suppose that $\|A\|_F = 1$ and that v is an eigenvector of A with associated eigenvalue $-\epsilon$. Let $\Pi \in R^{d\times d}$ be a projection onto a random k dimensional subspace S of $\mathbb{R}^d$, sampled from the rotationally invariant measure. Also suppose that $x^T Ax \geq 0$ with probability 0.999 when x is* *a random unit vector. Then*

$$\frac{1}{\|\Pi v\|^2}(\Pi v)^T A(\Pi v) \leq \frac{1}{d}(-0.5\epsilon k + \mathrm{Tr}(A) + O(1))$$

with probability at least $0.99 - \exp(-ck)$.

Proof. Let $u = \frac{\Pi v}{\|\Pi v\|}$. The subspace S was chosen randomly, so with probability at least $1 - \exp(-ck)$,

$$\langle u, v\rangle^2 = \|\Pi v\|^2 \geq 0.5\frac{k}{d}.$$

Let u' be the projection of u onto the hyperplane $v^\perp$ orthogonal to v. Observe by symmetry that $u'/\|u'\|$ is distributed uniformly over the sphere in $v^\perp$.

Let $A' = A - \epsilon vv^T$ be the matrix A with the $-\epsilon$ eigenvalue zeroed out. Then

$$u^T Au \leq -0.5\frac{k}{d}\epsilon + (u')^T A'u' \leq -0.5\frac{k}{d}\epsilon + (u'/\|u'\|)^T A'(u'/\|u'\|),$$

as long as $(u')^T A'u' \geq 0$, which holds with probability at least 0.999 as a consequence of the similar hypothesis. The latter term is a trace estimator for $\frac{1}{d}A'$ with variance bounded by $\frac{2}{d^2}\|A'\|_F^2 \leq \frac{2}{d^2}$ (for example by Proposition 3.21). So with 0.999 probability

$$(u'/\|u'\|)^T A'(u'/\|u'\|) \leq \frac{1}{d}(\mathrm{Tr}(A') + O(1)) = \frac{1}{d}(\mathrm{Tr}(A) + \epsilon + O(1)) \leq \frac{1}{d}(\mathrm{Tr}(A) + O(1)),$$

and the result follows. □

In the following lemma, we introduce the technical assumption that $k < cd$. However this will be unimportant later, as any sketch with $k \geq cd$ might as well have sketching dimension d, at which point the testing problem is trivial.

Lemma 3.23. *Suppose that $A \in \mathbb{R}^{d\times d}$ has an eigenvalue of $-\epsilon$, $\|A\|_F = 1$, and $G \in \mathbb{R}^{d\times k}$ with $k < d$ has iid $\mathcal{N}(0,1)$ entries. Also suppose that $x^T Ax \geq 0$ with probability at least 0.999 for a* *random unit vector x, and that $k < cd$ for some absolute constant c. Then with probability at least $0.99 - 3\exp(-ck)$,*

$$\lambda_{\min}(G^T AG) \leq -0.4\epsilon k + O(1) + \mathrm{Tr}(A) + c\frac{\sqrt{k}}{\sqrt{d}} \left|\mathrm{Tr}(A)\right|.$$

Proof. Let $\Pi_G = GG^\dagger$ denote projection onto the image of G.

Let v be an eigenvector of A with associated eigenvalue smaller or equal to $-\epsilon$, and set $u = G^\dagger v / \left\|G^\dagger v\right\|$. We then have

$$\lambda_{\min}(G^T AG) \leq u^T (G^T AG) u = \frac{1}{\left\|G^\dagger v\right\|^2} (\Pi_G v)^T A (\Pi_G v).$$

We also have

$$\left\|\Pi_G v\right\| = \left\|GG^\dagger v\right\| \leq \left\|G\right\|_{\text{op}} \left\|G^\dagger v\right\|.$$

By Theorem 4.6.1 in [Ver18], $\left\|G\right\|_{\text{op}} \leq \sqrt{d} + c\sqrt{k}$ with probability at least $1 - 2\exp(-k)$. Conditional on this occurring,

$$\left\|G^\dagger v\right\| \geq \frac{\left\|\Pi_G v\right\|}{\sqrt{d} + c\sqrt{k}},$$

from which it follows that

$$\lambda_{\min}(G^T AG) \leq (d + c\sqrt{d}\sqrt{k}) \frac{1}{\left\|\Pi_G v\right\|^2} (\Pi_G v)^T A (\Pi_G v),$$

as long as the quantity on the right-hand side is non-negative. If this quantity is negative, then we similarly have

$$\lambda_{\min}(G^T AG) \leq (d - c\sqrt{d}\sqrt{k}) \frac{1}{\left\|\Pi_G v\right\|^2} (\Pi_G v)^T A (\Pi_G v),$$

using the analogous bound on the smallest singular value of G.

Since G is Gaussian, the image of G is distributed with the respect to the rotationally invariant measure on k-dimensional subspaces. Therefore Lemma 3.22 applies, and the result follows after collecting terms, and using the assumption that $k \leq cd$ in the negative case. □

3.5.2 Lower bound on $\lambda_{\min}(G^T AG)$

We follow a standard protocol for bounding the extreme eigenvalues of a random matrix. We first show that $u^T(G^T AG)u$ is reasonably large for a fixed vector u with high probability. Then by taking a union bound over an ϵ-net we upgrade this to a uniform bound over the sphere.

We require two additional tricks. Our lower bound on $u^T(G^T AG)u$ arises from Berstein's inequality, which is hampered by the existence of large eigenvalues of A. Therefore in order to get a guarantee that holds with high enough probability, we first prune the large eigenvalues of A.

Second, the mesh size of our ϵ-net needs to be inversely proportional to the Lipschitz constant of $x \mapsto x^T(G^T AG)x$ as x ranges over the sphere. A priori, the Lipschitz constant might be as bad as $\|A\|_{\text{op}}$ which is typically larger than $\sqrt{d}$. This would ultimately give rise to an additional $\log(d)$ factor in the final sketching dimension. However we show that the the Lipschitz constant is in fact bounded by $O(k)$ with good probability, avoiding the need for any d dependence in the sketching dimension.

Proposition 3.24. *Let Q be a symmetric matrix, and let x and y be unit vectors. Then*

$$\left|x^T Qx - y^T Qy\right| \leq 2(\lambda_{\max}(Q) - \lambda_{\min}(Q)) \|x - y\|$$

Proof. We first reduce to the 2-dimensional case. Let W be a 2-dimensional subspace passing through x and y. The largest and smallest eigenvalues of the restriction to W of the quadratic form associated to Q are bounded from above and below by $\lambda_{\max}(Q)$ and $\lambda_{\min}(Q)$ respectively. It therefore suffices to prove the result when Q has dimension 2.

Since the result we wish to show is invariant under shifting Q by multiples of the identity, it suffices to consider the case when $\lambda_2 = 0$. After these reductions, we have

$$x^T Qx - y^T Qy = \lambda_1(x_1^2 - y_1^2) = \lambda_1(x_1 + y_1)(x_1 - y_1).$$

Since x and y are unit vectors, $|x_1 + y_1| \leq 2$ and $|x_1 - y_1| \leq \|x - y\|$ and the result follows. □

Lemma 3.25. *Let $S = G^T AG$ where $G \in \mathbb{R}^{k\times d}$ has iid $\mathcal{N}(0, 1)$ entries and $\|A\|_F = 1$. Then*

$$\lambda_{\max}(S) - \lambda_{\min}(S) \leq t$$

with probability at least $1 - \frac{4k(k+2)}{t^2}$.

Proof. Consider the random quantity $\alpha = u^T G^T AGu$, where u is a random unit vector in $\mathbb{R}^k$, independent from G. Note that Gu is distributed as a standard Gaussian, so α is a trace estimator for A with variance 2 [AT11].

On the other hand, one can also study the variance of α conditional on G by using Proposition 3.21. Let E be the event that $\lambda_{\max}(S) - \lambda_{\min}(S) \geq t$. If E occurs too often, then $\mathrm{Var}(\alpha)$ would be too large. Specifically, in the notation of Proposition 3.21 when E occurs, we necessarily have $\mathrm{Var}(\lambda_1(S), \ldots, \lambda_k(S)) \geq \frac{1}{k}(t/2)^2$, so $\mathrm{Var}(\alpha|E) \geq \frac{t^2}{2k(k+2)}$. Thus we have

$$2 = \mathrm{Var}(\alpha) \geq \Pr(E)\,\mathrm{Var}(\alpha|E) \geq \Pr(E)\frac{t^2}{2k(k+2)}, \tag{3.16}$$

and so $\Pr(E) \leq 4k(k + 2)/t^2$ as desired. □

Lemma 3.26. *Suppose that A is PSD with $\|A\|_F = 1$, and that v consists of iid $\mathcal{N}(0, 1)$ entries. Then for t $\geq 2\sqrt{k}$ we have*

$$\Pr(v^T Av \leq \mathrm{Tr}(A) - t) \leq \exp(-c\sqrt{k}(t - \sqrt{k})).$$

Proof. We have

$$\Pr(v^T Av \leq \mathrm{Tr}(A) - t) \leq \Pr(v^T A_{-k} v \leq \mathrm{Tr}(A) - t) \tag{3.17}$$

$$= \Pr(v^T A_{-k} v \leq \mathrm{Tr}(A_{-k}) - (t - \mathrm{Tr}(A_k))) \tag{3.18}$$

Note that $v^T A_{-k} v$ has expectation $\mathrm{Tr}(A_{-k})$, so by Bernstein's inequality (or Hanson-Wright) [Ver18],

$$\Pr(v^T A v \leq \mathrm{Tr}(A) - t) \leq \exp\left(-c \min\left(\frac{(t - \mathrm{Tr}(A_k))^2}{\|A_{-k}\|_F^2}, \frac{t - \mathrm{Tr}(A_k)}{\lambda_{\max}(A_{-k})}\right)\right),$$

for $t \geq \mathrm{Tr}(A_k)$.

Now note that $\|A_{-k}\|_F \leq \|A\|_F \leq 1$, and that $\lambda_{\max}(A_{-k}) \leq \frac{1}{\sqrt{k}}$, since $\|A\|_F = 1$. Additionally, $\mathrm{Tr}(A_k) \leq \sqrt{k}\, \|A_k\|_F \leq \sqrt{k}$. These bounds imply that

$$\frac{(t - \mathrm{Tr}(A_k))^2}{\|A_{-k}\|_F^2} \geq (t - \sqrt{k})^2$$

and

$$\frac{t - \mathrm{Tr}(A_k)}{\lambda_{\max}(A_{-k})} \geq \sqrt{k}(t - \sqrt{k})$$

for $t \geq \mathrm{Tr}(A_k)$. When $t > 2\sqrt{k}$, the latter expression is smaller, and the conclusion follows.

□

Theorem 3.27. *Suppose that A is PSD with $\|A\|_F = 1$, and that $G \in \mathbb{R}^{d \times k}$ has iid $\mathcal{N}(0,1)$ entries and that $k \geq 5$. Then with at least 0.99 probability,*

$$\lambda_{\min}(G^T A G) \geq \mathrm{Tr}(A) - c\sqrt{k}\log(k)$$

for some absolute constant c.

Proof. For any fixed unit vector $u \in \mathbb{R}^k$, Gu is distributed as a standard Gaussian, and so Lemma 3.26 applies. Therefore for a choice of constant,

$$u^T(G^T A G)u \geq \mathrm{Tr}(A) - c\sqrt{k}\log(k) \tag{3.19}$$

with probability at least $1 - \exp(-10k\log k)$.

Let $\mathcal{N}$ be a net for the sphere in $\mathbb{R}^k$ with mesh size $1/k$, which can be taken to have at most $(3k)^k$ elements [Ver18]. By taking a union bound, equation 3.19 holds over $\mathcal{N}$ with probability at least

$$1 - (3k)^k \exp(-10k \log k) \geq 1 - \exp(-k)$$

for $k \geq 2$.

By choosing $t = 100k$ in Lemma 3.25, and applying Proposition 3.24, we get that

$$\left| x^T (G^T A G) x - y^T (G^T A G) y \right| \leq 100k \left\| x - y \right\|$$

with probability at least 0.999. Since $\mathcal{N}$ has mesh size $1/k$, we have that

$$u^T (G^T A G) u \geq \mathrm{Tr}(A) - c\sqrt{k} + O(1)$$

for all unit vectors u in $\mathbb{R}^k$ with probability at least $0.999 - \exp(-k)$.

□

As a consequence of Theorem 3.27 and Lemma 3.23 we obtain our main result.

Theorem 3.28. *There is a bilinear sketch $G^T A G$ with sketching dimension $k = O(\frac{1}{\epsilon^2} \log^2 \frac{1}{\epsilon})$ that yields a two-sided ℓ_2-tester that is correct with at least 0.9 probability.*

Proof. If $\lambda_{\min}(A) < 0$ then we automatically reject. Otherwise we first use $O(1)$ columns of the sketching matrices to estimate α of $\mathrm{Tr}(A)$ to within an additive error of $\left\| A \right\|_F$ with 0.01 failure probability. We then use another $O(1)$ columns of the sketching matrices to construct an approximation β of $\left\| A \right\|_F$ with $\frac{1}{2} \left\| A \right\|_F \leq \beta \leq 2 \left\| A \right\|_F$ with 0.01 failure probability (see for example [MSW19]).

Now consider the quantity

$$\gamma = \frac{\alpha - \lambda_{\min}(G^T A G)}{\beta \sqrt{k} \log k}.$$

If A is PSD, then by Theorem 3.27,

$$\lambda_{\min}(G^T AG) \geq \mathrm{Tr}(A) - c\sqrt{k}\log k \left\|A\right\|_F ,$$

which implies that

$$\gamma \leq c_{\text{psd}},$$

for some absolute constant c_{psd}.

On the other hand, if A has a negative eigenvalue less than or equal to $-\epsilon$, then by Theorem 3.23

$$\lambda_{\min}(G^T AG) \leq -0.4\epsilon k \left\|A\right\|_F + \left(1 + c\frac{\sqrt{k}}{\sqrt{d}}\right) \mathrm{Tr}(A) + O(1) \left\|A\right\|_F ,$$

which implies that

$$\gamma \geq c_{\text{far}}(\epsilon\sqrt{k} - c)/\log k,$$

for some absolute constant c_{far}.

Finally by taking $k = \Theta(\frac{1}{\epsilon^2}\log^2\frac{1}{\epsilon})$, we have $c_{\text{psd}} < c_{\text{far}}(\epsilon\sqrt{k} - c)/\log k$, which implies that the tester is correct if it outputs True precisely when $\gamma \leq c_{\text{psd}}$.

□

Note that this result immediately gives a non-adaptive vector-matrix-vector tester which makes $\widetilde{O}(1/\epsilon^4)$ queries.

3.5.3 Application to adaptive vector-matrix-vector queries

By combining our bilinear sketch with Theorem 3.2 we achieve tight bounds for adaptive queries.

Theorem 3.29. *There is a two-sided adaptive ℓ_2-tester in the vector-matrix-vector model, which makes $\widetilde{O}(1/\epsilon^2)$ queries.*

Proof. To handle the technical condition in Lemma 3.23, we first compute $x^T Ax$ for a constant number of independent Gaussian vectors x. If x^TAx is ever negative, then we automatically reject.

We showed in the proof of Theorem 3.28 that with 0.9 probability, $\gamma \leq c_{psd}$ if A is PSD, and $\gamma \geq c_{\text{far}}(\epsilon\sqrt{k}-c)/\log k := C_{\text{far}}(k)$ if A is ϵ-far from PSD. By choosing some $k = O(\frac{1}{\epsilon^2}\log^2\frac{1}{\epsilon})$ we can arrange for $C_{\text{far}}(k) - c_{\text{psd}} \geq 1$, and also for $C_{\text{far}}(k) = \Theta(1)$.

Next we compute estimates α and β of $\text{Tr}(A)$ and $\|A\|_F$ as above, and (implicitly) form the matrix

$$\Gamma = \frac{G^TAG - \alpha I_k}{\beta\sqrt{k}\log k} + C_{\text{far}}(k)I_k - I_k.$$

If A is PSD, then with very good probability,

$$\lambda_{\text{min}}(\Gamma) = -\gamma + C_{\text{far}}(k) - \frac{1}{\epsilon} \geq -c_{\text{psd}} + C_{\text{far}}(k) - 1 \geq 0.$$

Similarly, if A is ϵ-far from PSD, then

$$\lambda_{\text{min}}(\Gamma) = -\gamma + C_{\text{far}}(k) - \frac{1}{\epsilon} \leq -C_{\text{far}}(k) + C_{\text{far}}(k) - 1 = -1.$$

Thus it suffices to distinguish Γ being PSD from Γ having a negative eigenvalue less than or equal to -1.

For this we will utilize our adaptive ℓ_1-tester, so we must bound $\|\Gamma\|_1$. Note that G^TAG is a trace estimator for kA with variance $O(k)\|A\|_F$ [AN13]. Therefore $\text{Tr}(G^TAG) = k(\text{Tr}(A) \pm O(1)\|A\|_F)$. Define $M = \frac{1}{\beta}(G^TAG - \alpha I_k)$, so that $\text{Tr}(M) = O(k)$. The negative eigenvalues of M sum to at most $k\lambda_{\text{min}}(M)$ in magnitude, and so the bound on $\text{Tr}(M)$ implies that $\|M\|_1 \leq 2k\lambda_{\text{min}}(M) + O(k)$. Write $\Gamma = \frac{1}{\sqrt{k}\log k}M + C_{\text{far}}(k)I - I$, so that $\|\Gamma\|_1 \leq \frac{1}{\sqrt{k}\log k}\|M\|_1 + O(k)$. Note that $\lambda_{\text{min}}(M) \leq \sqrt{k}\log k\lambda_{\text{min}}(\Gamma) + O(\sqrt{k}\log k)$. Therefore $\|\Gamma\|_1 \leq 2k\lambda_{\text{min}}(\Gamma) + O(k)$. From this we have

$$\frac{1}{\lambda_{\text{min}}(\Gamma)}\|\Gamma\|_1 \leq 2k\left(\frac{\lambda_{\text{min}}(\Gamma) + O(1)}{\lambda_{\text{min}}(\Gamma)}\right) + \frac{O(k)}{\lambda_{\text{min}}(\Gamma)} \leq O(k)$$

as long as $|\lambda_{\min}(\Gamma)| \geq \Omega(1)$, which it is by assumption.

Therefore Theorem 3.2 gives an adaptive vector-matrix-vector tester for Γ which requires only $\widetilde{O}(k) = \widetilde{O}(\frac{1}{\epsilon^2})$ queries.

□

As a consequence we also obtain a two-sided p-tester for all $p \geq 2$.

Corollary 3.30. *For $p \geq 2$, there is a two-sided adaptive ℓ_p-tester in the vector-matrix-vector model, which make $\widetilde{O}(1/\epsilon^2)d^{1-1/p}$ queries.*

Proof. Apply Theorem 3.29 along with the bound $\|A\|_p \geq d^{\frac{1}{p}-\frac{1}{2}} \|A\|_F$. □

3.5.4 Lower bounds for two-sided testers

Our lower bounds for two-sided testers come from the spiked Gaussian model introduced in [LW16]. As before, our adaptive lower bounds will come as a consequence of the corresponding non-adaptive bounds.

Theorem 3.31. *A two-sided ℓ_p-tester that makes non-adaptive vector-matrix-vector queries requires at least*

- $\Omega(\frac{1}{\epsilon^{2p}})$ *queries for* $1 \leq p \leq 2$
- $\Omega(\frac{1}{\epsilon^4}d^{2-4/p})$ *queries for* $2 < p < \infty$ *as long as* d *can be taken to be* $\Omega(1/\epsilon^p)$.
- $\Omega(d^2)$ *queries for* $p = \infty$.

Proof. First, take G to be a $d \times d$ matrix with $\mathcal{N}(0,1)$ entries, where $d = 1/\epsilon$. Also let $\widetilde{G} = G + suv^T$ where u and v have $\mathcal{N}(0,1)$ entries, and s is to be chosen later. We will show that a PSD-tester can be used to distinguish G and $\widetilde{G}$, while this is hard for any algorithm that uses only a linear sketch.

Recall that G has spectral norm at most $c\sqrt{d}$ with probability at least $1-2e^{-d}$, where c is an absolute constant. (We will use c throughout to indicate absolute constants that we do not track – it may have different values between uses, even within the same equation.) Set

$$G_{\text{sym}} = \begin{bmatrix} 0 & G \\ G^T & 0, \end{bmatrix} \tag{3.20}$$

and define $\widetilde{G}_{\text{sym}}$ similarly. Note that the eigenvalues of G_{sym} are precisely $\{\pm\sigma_i\}$ where the σ_i are singular values of G. Therefore $G_{\text{sym}} + c\sqrt{d}I$ is PSD with high probability.

On the other hand, $\left\|uv^T\right\| \geq cd$ with high probability so

$$\left\|\widetilde{G}\right\| \geq s\left\|uv^T\right\| - \|G\| \geq csd - c\sqrt{d}, \tag{3.21}$$

which implies that $\widetilde{G}_{\text{sym}} + c\sqrt{d}I$ has a negative eigenvalue with magnitude at least $csd - c\sqrt{d} - c\sqrt{d} = csd - c\sqrt{d}$.

We also have that $\left\|\widetilde{G}_{\text{sym}}\right\|_p \leq c\sqrt{d}d^{1/p}$, since the operator norm of G is bounded by $c\sqrt{d}$ with high probability. Hence if

$$c\frac{sd-\sqrt{d}}{\sqrt{d}d^{1/p}} = c\frac{s\sqrt{d}-1}{d^{1/p}} \geq \epsilon, \tag{3.22}$$

then a two-sided PSD-tester can distinguish between G and $\widetilde{G}$ with constant probability of failure.

On the other hand, Theorem 4 in [LW16] implies that any sketch that distinguishes these distributions with constant probability, must have sketching dimension at least c/s^4.

It remains to choose values of s and d for which the inequality in equation (3.22) holds. When $1 \leq p \leq 2$, we take $d = \Theta(\epsilon^{-p})$ and $s = O(\epsilon^{p/2})$ giving a lower bound of $\Omega(1/\epsilon^{2p})$. When $2 < p < \infty$, we take $d = \Omega(1/\epsilon^p)$ and $s = O(\epsilon d^{1/p-1/2})$ giving a lower bound of $\Omega(\frac{1}{\epsilon^4}d^{2-4/p})$. Finally, when $p = \infty$ we take $s = O(1/\sqrt{d})$ giving a lower bound of $\Omega(d^2)$.

□

Remark 3.32. *The argument above applies equally well to arbitrary linear sketches, of which a* *series of non-adaptive vector-matrix-vector queries is a special case.*

Corollary 3.33. *A two-sided adaptive ℓ_p-tester in the vector-matrix-vector model requires at least*

- $\Omega(\frac{1}{\epsilon^p})$ *queries for* $1 \leq p \leq 2$
- $\Omega(\frac{1}{\epsilon^2} d^{1-2/p})$ *queries for* $2 < p < \infty$ *as long as* d *can be taken to be* $\Omega(1/\epsilon^p)$.
- $\Omega(d)$ *queries for* $p = \infty$.

Proof. Apply Theorem 3.31 along with Lemma 3.14. □

For adaptive measurements, we supply a second proof via communication complexity which has the advantage of applying to general linear measurements, albeit at the cost of an additional bit complexity term.

Proposition 3.34. *Let* $p \in [1, \infty)$, $\epsilon < \frac{1}{2}$ *and* $d \geq (p/\epsilon)^p$. *An adaptive two-sided ℓ_p-tester taking general linear measurements* $\langle M_i, A \rangle$ *of* A, *where each* M_i *has integer entries in* $(-2^{b-1}, 2^{b-1}]$, *must make at least* $\frac{c}{b + d \log \frac{1}{\epsilon}} \frac{1}{\epsilon^2} d^{1-2/p}$ *queries.*

Proof. We reduce from the multiplayer set disjointness problem [KPW21]. Let the k players have sets $S_1, \ldots, S_k \subseteq [d]$ which either are (i) all pairwise disjoint or (ii) all share precisely one common element. Distinguishing between (i) and (ii) with $2/3$ probability in the blackboard model of communication requires $\Omega(d/k)$ bits. We will choose $k = \lceil \max(4\epsilon d^{1/p}, 4p) \rceil = \lceil 4\epsilon d^{1/p} \rceil$.

For each i let χ_i be the characteristic vector of S_i and let $A_i = \operatorname{diag}(\chi_i)$. Consider the matrix $A := I_d - \sum_{i=1}^d A_i$.

In situation (i), A is PSD. In situation (ii), $\|A\|_p^p \leq k^p + d$ and $\lambda_{\min}(A) = -(k-1)$. We have

$$|\lambda_{\min}(A)|^p = (k-1)^p = k^p \left(1 - \frac{1}{k}\right)^p \geq k^p \left(1 - \frac{p}{k}\right) \geq \frac{3}{4} k^p$$

and

$$\epsilon^p \|A\|_p^p \leq \epsilon^p (k^p + d) = \epsilon^p k^p + \epsilon^p d \leq \frac{1}{2} k^p + \epsilon^p d \leq \frac{3}{4} k^p.$$

Given query access to A, an ℓ_p-tester can therefore distinguish between (i) and (ii) with $2/3$ probability. Note that a single linear measurement $\langle M, A\rangle$ may be simulated in the blackboard model using $O(k(\log b+\log d))$ bits; each player simply computes and communicates $\langle M, A_i\rangle$, and the players add the resulting measurements. The players therefore need at least $\Omega(\frac{1}{k(\log b+\log d)} \cdot \frac{d}{k})$ bits of communication to solve the PSD-testing problem.

□

3.6 Spectrum Estimation

We make use of the following result, which is Lemma 11 of [CW17b] specialized to our setting.

Lemma 3.35. *For a symmetric matrix $A \in \mathbb{R}^{d\times d}$, there is a distribution over an oblivious sketching matrix $R \in \mathbb{R}^{d\times m}$ with $m = O(\frac{k}{\epsilon})$ so that with at least 0.9 proability,*

$$\min_{Y^*\in \text{rank } k, \text{PSD}} \left\|(AR)Y^*(AR)^T - A\right\|_F^2 \leq (1+\epsilon)\left\|A_{k,+} - A\right\|_F^2, \tag{3.23}$$

where $A_{k,+}$ is the optimal rank-one PSD approximation to A in Frobenius norm.

Remark 3.36. *In our setting one can simply take R to be Gaussian since the guarantee above must hold when A is drawn from a rotationally invariant distribution. In many situations, structured or sparse matrices are useful, but we do not need this here.*

We also recall the notion of an affine embedding [CW17a].

Definition 3.37. *S is an affine embedding for matrices A and B if for all matrices X of the appropriate dimensions, we have*

$$\left\|S(AX-B)\right\|_F^2 = (1\pm\epsilon)\left\|AX-B\right\|_F^2. \tag{3.24}$$

We also recall that when A is promised to have rank at most r, there is a distribution over S

with $O(\epsilon^{-2}r)$ rows such that (3.24) holds with constant probability for any choice of A and B [CW17a].

Lemma 3.38. *There is an algorithm which makes $O(\frac{k^2}{\epsilon^6}\log\frac{1}{\delta})$ vector-matrix-vector queries to A and with at least $1-\delta$ probability outputs an approximation of $\|A\|_{k,+}$, accurate to within $\epsilon\,\|A\|_F^2$ additive error.*

Proof. We run two subroutines in parallel.

Subroutine 1. Approximate $\|A_{k,+}-A\|_F^2$ up to $O(\epsilon)$ multiplicative error.

Our algorithm first draws affine embedding matrices S_1 and S_2 for $r = k/\epsilon$, and with ϵ distortion, each with $O(\frac{k}{\epsilon^3})$ rows. We also draw a matrix R as in Lemma 3.35 with $m = O(\frac{k}{\epsilon})$ columns.

We then compute S_1AR and S_2AR, each requiring $\frac{k^2}{\epsilon^4}$ vector-matrix-vector queries, and compute $S_1AS_2^T$ requiring $\frac{k^2}{\epsilon^6}$ queries.

Let Y_k be arbitrary with the appropriate dimensions (later we will optimize Y_k over rank k PSD matrices). By using the affine embedding property along with the fact that R has rank at most $\frac{k}{\epsilon}$, we have

$$\begin{aligned}\left\|(S_1AR)Y_k(S_2AR)^T + S_1AS_2^T\right\|_F^2 &= (1\pm\epsilon)\left\|ARY_k(S_2AR)^T + AS_2^T\right\|_F^2\\ &= (1\pm\epsilon)\left\|S_2ARY_kR^TA + S_2A\right\|_F^2\\ &= (1\pm 3\epsilon)\left\|ARY_kR^TA + A\right\|_F^2.\end{aligned}$$

As a consequence of this, and the property held by R, we have

$$\min_{\mathrm{rk}(Y_k)\le k, Y_k\text{ PSD}} \left\|(S_1AR)Y_k(S_2AR)^T + S_1AS_2^T\right\|_F^2 = (1\pm 3\epsilon)\min_{Y_k}\left\|ARY_kR^TA + A\right\|_F^2 \tag{3.25}$$

$$= (1\pm 7\epsilon)\left\|A_{k,+} - A\right\|_F^2. \tag{3.26}$$

Thus by computing the quantity in the left-hand-side above, our algorithm computes an $O(\epsilon)$ multiplicative approximation using $O(k^2/\epsilon^6)$ vector-matrix-vector queries.

Subroutine 2. Approximate $\|A\|_F^2$ up to $O(\epsilon)$ multiplicative error.

We simply apply Theorem 2.2. of [MSW19], set $q = 2$, and note that the entries of the sketch correspond to vector-matrix-vector products. By their bound we require $O(\epsilon^{-2}\log(1/\epsilon))$ vector-matrix-vector queries.

Since $\|A_{k,+}\|_F^2 = \|A\|_F^2 - \|A_{k,+} - A\|_F^2$, we obtain an additive $O(\epsilon)\,\|A\|_F^2$ approximation to $\|A_{k,+}\|_F^2$ by running the two subroutines above and subtracting their results.

Finally, by repeating the above procedure $O(\log \frac{1}{\delta})$ times in parallel and taking the median of the trails, we obtain a failure probability of at most δ.

□

The matrices S_1AR and S_2AR in Subroutine 1 each have rank k/ϵ whereas the dimensions of $S_1AS_2^T$ are k/ϵ^3. The matrix $S_1AS_2^T$ therefore contains a large amount of data that will not play a role when optimizing over Y_k. If S_1AR and S_2AR were known ahead of time, then we could choose to compute only the portion of $S_1AS_2^T$ that is relevant to the optimization step, and simply estimate the Frobenius error incurred by the rest. This allows us to construct a slightly more efficient two-pass protocol.

Proposition 3.39. *By using a single round adaptivity, the guarantee of Lemma 3.38 may be achieved using $O(\frac{k^2}{\epsilon^4}\log\frac{1}{\delta})$ vector-matrix-vector queries.*

Proof. As described above, we modify Subroutine 1. Write M_i for S_iAR and Q for $S_1AS_2^T$. Instead of computing M_1, M_2, and Q at once, we instead compute M_1 and M_2 first using k^2/ϵ^4 vector-matrix-vector queries.

We wish to estimate $\min_{Y_k} \left\|M_1Y_kM_2^T - Q\right\|_F^2$, where the minimum is over PSD matrices Y_k of rank at most k. Let Π_i denote orthogonal projection onto the image of M_i, and set $\Pi_i^\perp = I - \Pi_i$.

Then for fixed Y, we use the Pythagorean theorem to write

$$
\begin{aligned}
\|M_1 Y M_2 - Q\|_F^2 &= \|\Pi_1 M_1 Y M_2 \Pi_2 - Q\|_F^2 && (3.27)\\
&= \left\|\Pi_1(M_1 Y M_2^T - Q)\Pi_2 + \Pi_1^{\perp} Q \Pi_2 + \Pi_1 Q \Pi_2^{\perp} + \Pi_1^{\perp} Q \Pi_2^{\perp}\right\|_F^2 && (3.28)\\
&= \left\|\Pi_1(M_1 Y M_2^T - Q)\Pi_2\right\|_F^2 + \left\|\Pi_1^{\perp} Q \Pi_2\right\|_F^2 + \left\|\Pi_1 Q \Pi_2^{\perp}\right\|_F^2 + \left\|\Pi_1^{\perp} Q \Pi_2^{\perp}\right\|_F^2 && (3.29)\\
&= \left\|M_1 Y M_2^T - \Pi_1 Q \Pi_2\right\|_F^2 + \left\|\Pi_1^{\perp} Q \Pi_2\right\|_F^2 + \left\|\Pi_1 Q \Pi_2^{\perp}\right\|_F^2 + \left\|\Pi_1^{\perp} Q \Pi_2^{\perp}\right\|_F^2 . && (3.30)
\end{aligned}
$$

Note that each of the last three terms can be estimated to within $O(\epsilon)$ multiplicative error using Subroutine 2, since a vector-matrix-vector query to one of these matrices may be simulated with a single query to A. Also since each M_i has rank $O(k/\epsilon)$, the Π_i's are projections onto $O(k/\epsilon)$ dimensional subspaces. Since the Π_i's are known to the algorithm, we may compute $\Pi_1 Q \Pi_2$ explicitly using k^2/ϵ^2 vector-matrix-vector queries, as it suffices to query $\Pi_1 Q \Pi_2$ over the Cartesian product of bases for the images of Π_1 and Π_2. By optimizing the first term over Y, we thus obtain an $O(\epsilon)$ multiplicative approximation to $\min_{Y_k} \left\|M_1 Y_k M_2^T - Q\right\|_F^2$ as desired. This gives a version of Subroutine 1 that makes $O(k^2/\epsilon^4)$ queries. □

We note that we immediately obtain a poly$(1/\epsilon)$ query ℓ_2-tester by applying Lemma 3.38 to approximate $A_{1,-}$. However this yields a worse ϵ dependence than Theorem 3.28. Perhaps more interestingly, these techniques also give a way to approximate the top k (in magnitude) eigenvalues of A while preserving their signs. We note a minor caveat. If λ_k and λ_{k+1} are very close in magnitude, but have opposite signs, then we cannot guarantee that we approximate λ_k. Therefore in the statement below, we only promise to approximate eigenvalues with magnitude at least $|\lambda_k| + 2\epsilon$.

Theorem 3.40. *Let $\lambda_1, \lambda_2, \dots$ be the (signed) eigenvalues of A sorted in decreasing order of magnitude.*

There is an algorithm that makes $O(\frac{k^2}{\epsilon^{12}} \log k)$ non-adaptive vector-matrix-vector queries to A, *and with probability at least* 0.9*, outputs* $\tilde{\lambda}_1, \ldots, \tilde{\lambda}_k$ *such that*

(i) There exists a permutation σ *on* $[k]$ *so that for all* i *with* $|\lambda_i| \geq |\lambda_k| + 2\epsilon$*,* $|\tilde{\lambda}_{\sigma(i)} - \lambda_i| \leq \epsilon \|A\|_F$

(ii) For all i*, there exists* j *with* $|\lambda_j| \geq |\lambda_k| - \epsilon$ *and* $|\tilde{\lambda}_i - \lambda_j| \leq \epsilon \|A\|_F$

With one additional round of adaptivity the number of measurements can be reduced to $O(\frac{k^2}{\epsilon^8} \log k)$*.*

Proof. We set $\delta = \frac{1}{20k}$ in Lemma 3.38 and use it to approximate $\|A_{1,+}\|_F^2, \ldots, \|A_{k,+}\|_F^2$, along with $\|A_{1,-}\|_F^2, \ldots, \|A_{k,-}\|_F^2$, each to within $(\epsilon^2/2)\|A\|_F^2$ additive error. Note that we may use the same sketching matrices for each of these $2k$ tasks, and then take a union bound to obtain a failure probability of at most 0.1. Thus we require only $O(\frac{k^2}{\epsilon^{12}} \log k)$ queries in total. With an additional round of adaptivity, Proposition 3.39 reduces this bound to $O(\frac{k^2}{\epsilon^8} \log k)$.

Let $\lambda_{i,+}$ be the i^{th} largest positive eigenvalue of A if it exists, and 0 otherwise. Define $\lambda_{i,-}$ similarly. Note that $\lambda_{i,+}^2 = \|A_{i,+}\|_F^2 - \|A_{i-1,+}\|_F^2$ for $i \geq 2$, and that $\lambda_{1,+}^2 = \|A_{1,+}\|_F^2$. This allows us to compute approximations $\tilde{\lambda}_{i,+} \geq 0$ such that $|\tilde{\lambda}_{i,+}^2 - \lambda_{i,+}^2| \leq \epsilon^2 \|A\|_F^2$, and similarly for the $\lambda_{i,-}$'s with $\tilde{\lambda}_{i,-} \leq 0$. Note that this bound implies $|\tilde{\lambda}_{i,+} - \lambda_{i,+}| \leq \epsilon \|A\|_F$.

Our algorithm then simply returns the k largest magnitude elements of $\{\tilde{\lambda}_{1,+}, \ldots, \tilde{\lambda}_{k,+}, \tilde{\lambda}_{1,-}, \ldots, \tilde{\lambda}_{k,-}\}$.

□

3.7 Non-adaptive testers

3.7.1 Non-adaptive vector-matrix-vector queries

We gave a lower bound for one-sided testers earlier in Theorem 3.13. Here we observe that the sketch of Andoni and Nguyen [AN13] provides a matching upper bound.

Proposition 3.41. *There is a one-sided non-adaptive* ℓ_1*-tester that makes* $O(1/\epsilon^2)$ *non-adaptive vector matrix-vector queries to* A*.*

Proof. We simply apply Proposition 3.8 Note that the sketch is of the form $G^T A G$, where $G \in \mathbb{R}^{m \times d}$ with $m = O(1/\epsilon)$ in our case. Each entry of $G^T A G$ of which there are m^2 can be computed with a single vector-matrix-vector query. □

Corollary 3.42. *There is a one-sided non-adaptive ℓ_p-tester that makes $O(\frac{1}{\epsilon^2}d^{2-2/p})$ non-adaptive vector matrix-vector queries to A.*

Proof. Apply the previous proposition along with the bound $\|A\|_p \geq d^{1/p-1}\|A\|_1$. □

3.7.2 Non-adaptive matrix-vector queries

As a simple corollary of the algorithm given by Corollary 3.42 we have the following.

Proposition 3.43. *There exists a one-sided non-adaptive tester making $O(\frac{1}{\epsilon}d^{1-1/p})$ matrix-vector queries.*

Proof. Simply note that a $k \times k$ bilinear sketch may be simulated with k matrix-vector queries. □

We next show that this bound is tight. While we consider the case where the tester queries the standard basis vectors, this is done essentially without loss of generality as any non-adaptive tester may be implemented by querying on an orthonormal set.

Proposition 3.44. *Suppose that a one sided matrix-vector tester queries on the standard basis vectors $e_1, \ldots, e_k$ and outputs False. Let U be the top $k \times k$ submatrix of $[Ae_1, \ldots, Ae_k]$. Then if U is non-singular, there must exist a "witness vector" $v \in \operatorname{span}(x_1, \ldots, x_k)$ such that $v^T Av < 0$.*

Proof. Let Q be the matrix with columns Ae_i, and decompose it as

$$Q = \begin{pmatrix} U \\ B \end{pmatrix}^T \tag{3.31}$$

where $U \in \mathbb{R}^{k\times k}$ and $B \in \mathbb{R}^{d\times(d-k)}$. Suppose that there does not exist a v as in the statement of the proposition. Note that this implies that U is PSD, and in fact positive definite by the assumption that U was non-singular. Now consider the block matrix

$$\widetilde{A}s = \begin{pmatrix} U & B^T \\ B & \lambda I \end{pmatrix} \tag{3.32}$$

for some choice of $\lambda > 0$. For arbitrary v and w of the appropriate dimensions, we have

$$\begin{pmatrix} v & w \end{pmatrix} \begin{pmatrix} U & B^T \\ B & \lambda I \end{pmatrix} \begin{pmatrix} v \\ w \end{pmatrix} = v^T U v + 2 v^T B w + \lambda w^T w \tag{3.33}$$

$$\geq \|v\|^2 \sigma_{\min}(U) + \lambda \|w\|^2 - 2 \|v\| \|w\| \sigma_{\max}(B). \tag{3.34}$$

Since $\sigma_{\min}(U) \neq 0$ this expression viewed as a quadratic form in $\|v\|$ and $\|w\|$ is positive definite for large enough λ. This implies that $\widetilde{A}$ is positive definite as well. Since $\widetilde{A} e_i = A e_i$ by construction, this shows that the queries are consistent with a PSD matrix. So a one-sided tester that cannot produce a witness vector in this case must not output False. □

Theorem 3.45. *Set $D = \operatorname{diag}(-\lambda, 1, \ldots, 1)$, let S be a random orthogonal matrix, and take $A = S^T D S$. In the matrix-vector model, a one-sided non-adaptive tester must make at least $\frac{1}{2} \frac{d}{1+\lambda}$ queries to be correct on this distribution with $2/3$ probability.*

Proof. Given this distribution we may assume without loss of generality that the tester queries on $e_1, \ldots e_k$, whose span we call E_k. Let u denote the $-\lambda$ eigen-direction of A, which is distributed uniformly over S^{d-1}. For unit vectors x, the quadratic form associated to A is negative exactly when $\langle x, u \rangle^2 \geq \frac{1}{1+\lambda}$. Also the U as in Proposition 3.44 is non-singular with probability 1. In this case, by Proposition 3.44 the tester can only succeed if $\|\Pi_{E_k} u\|^2 \geq \frac{1}{1+\lambda}$. On the other hand $\mathbb{E} \|\Pi_{E_k} u\|^2 = k/d$, so by Markov, $\|\Pi_{E_k} u\|^2 \leq 2k/d$ with probability at least $1/2$. Therefore a tester that succeeds with $2/3$ probability must have $2k/d \geq 1/(1+\lambda)$. □

Corollary 3.46. *In the matrix-vector model, a one-sided non-adaptive ℓ_p-tester must make at least $\Omega(\frac{1}{\epsilon} d^{1-1/p})$ queries.*

Proof. Apply Theorem 3.45 with $\lambda = \epsilon d^{1/p}$. □

3.8 Conclusion and Open Problems

We gave a series of tight bounds for PSD-testing in both the matrix-vector and vector-matrix-vector models. We provided tight bounds as well as a separation between one and two-sided testers in the latter model. There are a number of additional questions that may yield interesting future work.

- Our adaptive vector-matrix-vector algorithm for $p = 1$ uses $\Omega(1/\epsilon)$ rounds of adaptivity, but this may not always be desirable in practice, since the queries cannot be run in parallel. Are there good algorithms that use less adaptivity? What is the optimal trade-off between query complexity and the number of rounds of adaptivity?

- One could modify our testing model and consider testers which should output False whenever the ℓ_p norm of the negative eigenvalues is at least an ϵ fraction of the ℓ_p norm of positive eigenvalues. Is it possible to give tight bounds for this problem in the models that we considered?

- Is it possible to use the ideas behind our two-sided bilinear sketch to give better bounds for spectral estimation with additive Frobenius error?

CHAPTER 4

Optimal Eigenvalue Approximation via Sketching

In many applications, one is interested in computing spectral information about a symmetric matrix A. For example if A is a Hessian matrix for a function f, then the spectrum gives useful information about the local curvature. If A is an empirical covariance matrix, then the spectrum gives the singular values associated to the data matrix. If one is interested in applying PCA for example, it is useful to have an estimate for the singular values in order to truncate the SVD appropriately.

In the big data setting, matrices are often extremely large with dimensions in the tens of millions or higher. Directly computing eigenvalues in such a setting is often impractical. In this chapter we seek to estimate the spectrum of A via a technique known as sketching. We consider a particularly simple bilinear sketch of the from $G^T AG$ for a Gaussian matrix G.

Interestingly this sketch has been considered before in the context of eigenvalue estimation [AN13], however their algorithm for recovering the eigenvalues from the sketch is not sufficient to obtain the $\epsilon \left\|A\right\|_F$ additive approximation guarantees that we give here. In this chapter, we present a new recovery algorithm and introduce several new eigenvalue bounds in the analysis.

4.1 Contributions

This chapter presents joint work with David Woodruff [SW23]. I proposed the main algorithm as well as the lower bounds, and wrote the technical sections of the manuscript. David Woodruff suggested a technique for upper-bounding the eigenvalues and for speeding up the sketches. Both authors contributed to the writing.

4.2 Introduction

Estimating the eigenvalues of a real symmetric matrix has numerous applications in data analysis, engineering, optimization, spectral graph theory, and many other areas. As modern matrices may be very large, traditional algorithms based on the singular value decomposition (SVD), subspace iteration, or Krylov methods, may be be too slow. Therefore, a number of recent works have looked at the problem of creating a small summary, or sketch of the input matrix, so that from the sketch one can approximate each of the eigenvalues well. Indeed, in the realm of sublinear algorithms, this problem has been studied in the streaming model [AN13], the sampling and property testing models [Bal+19a; BCJ20; Bha+21; BKM22], and matrix-vector and vector-matrix-vector query models [AN13; LNW14; LNW19; NSW22]; the latter model also contains so-called bilinear sketches.

In this work we focus on designing linear sketches for eigenvalue estimation. Namely, we are interested in estimating the spectrum of a real symmetric matrix $A \in \mathbb{R}^{n \times n}$ up to $\epsilon \left\|A\right\|_F$ error via a bilinear sketch GAG^T with $G \in \mathbb{R}^{k \times n}$ is a matrix of i.i.d. $N(0, 1/k)$ random variables, i.e., Gaussian of mean zero and variance $1/k$. The algorithm should succeed with large constant probability in estimating the entire spectrum. This is a very natural sketch, and unsurprisingly has been used before both in [AN13] to estimate eigenvalues with an additive error of roughly $\epsilon \sum_{i=1}^n |\lambda_i(A)|$, where $\lambda_i(A)$ are the eigenvalues of A, as well as in [NSW22] for testing if a matrix is positive semidefinite (PSD). We note that the additive error of $\epsilon \|A\|_1 = \epsilon \sum_{i=1}^n |\lambda_i(A)|$ can be significantly weaker than our desired $\epsilon \left\|A\right\|_F$ error, as $\left\|A\right\|_F$ can be as small as $\frac{\|A\|_1}{\sqrt{d}}$. This is analogous to the ℓ_2 versus ℓ_1 guarantee for heavy hitters in the data stream model, see, e.g., [Woo16].

It may come as a surprise that GAG^T has any use at all for achieving additive error in terms of

$\epsilon\|A\|_F$! Indeed, the natural way to estimate the i-th eigenvalue of A is to output the i-th eigenvalue of GAG^T, and this is exactly what the algorithm of [AN13] does. However, by standard results for trace estimators, see, e.g., [Mey+21] and the references therein, the trace of GAG^T is about the trace of A, which can be a $\sqrt{d}$ factor larger than $\|A\|_F$, and thus the estimation error can be much larger than $\epsilon\|A\|_F$. This is precisely why [AN13] only achieves additive $\epsilon\|A\|_1$ error with this sketch. Moreover, the work of [NSW22] does use sketching for eigenvalue estimation, but uses a different, and much more involved sketch based on ideas for low rank approximation of PSD matrices [CW17b], and achieves a much worse $\tilde{O}(k^2/\epsilon^{12})$ number of measurements to estimate each of the top k eigenvalues, including their signs, up to additive error $\epsilon\|A\|_F$. Here we use $\tilde{O}()$ notation to suppress $\text{poly}(\log(n/\epsilon))$ factors. Note that for $k > 1/\epsilon^2$, one can output 0 as the estimate to λ_k, and thus the sketch size of [NSW22] is $\tilde{O}(1/\epsilon^{16})$.

To achieve error in terms of $\|A\|_F$, the work of [AN13] instead considers the sketch GAH^T, where $G, H \in \mathbb{R}^{k\times n}$ are independent Gaussian matrices. However, the major issue with this sketch is it inherently loses sign information of the eigenvalues. Indeed, their algorithm for reconstructing the eigenvalues uses only the sketched matrix, while forgetting G and H (more specifically they only use the singular values of this matrix). However the distributions of G and H are invariant under negation, so the sketch alone cannot even distinguish A from $-A$. In addition to this, even if one assumes the input A is PSD, so that the signs are all positive, their result for additive error $\epsilon\|A\|_F$ would give a suboptimal sketching dimension of $k = \tilde{O}(1/\epsilon^3)$; see further discussion below.

4.2.1 Our Contributions

Optimal Sketching Upper Bound. We obtain the first optimal bounds for eigenvalue estimation with the natural $\epsilon\|A\|_F$ error via sketching. We summarize our results compared to prior work in Table 4.1. We improve over [AN13; NSW22] in the following crucial ways.

Qualitatively, we drop the requirement that A is PSD. As mentioned, the eigenvalues of our sketch GAG^T may not be good approximations to the eigenvalues of A. In particular, we observe

Table 4.1: Our work and prior work on estimating each eigenvalue of an arbitrary symmetric matrix A up to additive $\epsilon\|A\|_F$ error.

Sketching dimension	Reference	Notes
$\tilde{O}(1/\epsilon^6)$	[AN13]	Loses sign information
$\tilde{O}(1/\epsilon^{16})$	[NSW22]	
$\Omega(1/\epsilon^4)$	[NSW22]	Lower bound
$O(1/\epsilon^4)$	**Our Work**	

that the sketched eigenvalues concentrate around $\frac{1}{k}\operatorname{Tr}(A)$, which could be quite large, on the order of $\frac{\sqrt{d}}{k}\|A\|_F$. By shifting the sketched eigenvalues by $-\frac{1}{k}\operatorname{Tr}(A)$ via an additional trace estimator we compute, this enables us to correct for this bias, and we are able to show that the resulting eigenvalues are good approximations to those of A. In order to perform this correction we in fact require the sketched eigenvalues to concentrate around $\frac{1}{k}\operatorname{Tr}(A)$. Obtaining this concentration is where we require Gaussianity in our argument[1]. We leave it as an open question to obtain similar concentration from common sketching primitives.

Comparison with existing work. Quantitatively, the analysis of [AN13] for the related GAH^T sketch works by splitting the spectrum into a "head" containing the large eigenvalues, and a "tail" containing the remaining eigenvalues. The authors then incur an additive loss from the operator norm of the tail portion of the sketch, and show that the head portion of the sketch approximates the corresponding eigenvalues to within a multiplicative error. Notably, their multiplicative constant is uniform over the large eigenvalues. This is a stronger guarantee than we need. For example, to approximate an eigenvalue of $1/2$ to within ϵ additive error, we need a $(1 \pm O(\epsilon))$ multiplicative guarantee. However to approximate an eigenvalue of 2ϵ to within ϵ additive error, a $(1 \pm O(1))$ multiplicative guarantee suffices. In other words, smaller eigenvalues require less stringent multiplicative guarantees to achieve the same additive guarantee. We leverage this observation in order to get a uniform *additive* guarantee for the large eigenvalues, while not relying on a uniform multiplicative guarantee. Thus, we improve the worst-case $k = O(1/\epsilon^3)$ bound of [AN13] to a

[1] However in the appendix we give a faster sketch for PSD matrices.

$k = O(1/\epsilon^2)$ bound for an $\epsilon\|A\|_F$ error guarantee.

Indeed, one can show if the eigenvalues of A are, in non-increasing order,

$$\frac{c_d}{\sqrt{1}}, \frac{c_d}{\sqrt{2}}, \frac{c_d}{\sqrt{3}}, \frac{c_d}{\sqrt{4}}, \ldots, \frac{c_d}{\sqrt{d}},$$

where $c_d = O(\log^{-1/2} d)$ so that $\|A\|_F = 1$, then $O(1/\epsilon^3)$ is the bound their Theorem 1.2 and corresponding Lemma 3.5 would give. To see this, their Lemma 3.5, which is a strengthening of their Theorem 1.2, states that for $i = 1 \ldots k$,

$$\left|\lambda_i^2(GAH^T) - \lambda_i^2(A)\right| \leq \alpha\lambda_i^2(A) + O\left(\lambda_k^2(A)\right) + O\left(\frac{\alpha^2}{k}\|A_{-k}\|_F^2\right), \tag{4.1}$$

with sketching dimension $O(k/\alpha^2)$ on each side (and hence $O(k^2/\alpha^4)$ total measurements). Suppose $\|A\|_F = O(1)$ and that we would like to use this bound to approximate $\lambda_\ell(A) > \alpha$ to within ϵ additive error. After adjusting for the squares, this is equivalent to bounding the left-hand side of (4.1) by $O(\epsilon\lambda_\ell)$ for $i = \ell$. Obtaining such a bound from (4.1) requires that the first two terms on the right-hand side are bounded by $O(\epsilon\lambda_\ell(A))$, i.e., that $\alpha \leq O(\epsilon/\lambda_\ell(A))$ and $\lambda_k^2(A) \leq O(\epsilon\lambda_\ell(A))$. For the spectrum above, we must therefore take $k \gtrsim c_d\frac{\sqrt{\ell}}{\epsilon}$, which results in a sketching dimension of

$$\frac{k}{\alpha^2} \approx \frac{c_d\sqrt{\ell}}{\epsilon} \cdot \frac{\lambda_\ell(A)^2}{\epsilon^2} = \frac{c_d^3}{\epsilon^3\sqrt{\ell}}$$

on each side.

Thus for this spectrum, [AN13] requires a sketching dimension of $O(1/\epsilon^3)$ (up to $\log d$ factors) to approximate the largest eigenvalues of A to ϵ additive error. Indeed this bound does not achieve $O(1/\epsilon^2)$ sketching dimension, unless $\ell \gtrsim 1/\epsilon^2$, at which point $\lambda_\ell(A) \leq O(\epsilon)$ and does not need to be approximated by our algorithm.

We note that while [NSW22] could also report the signs of the approximate eigenvalues, their $\tilde{O}(1/\epsilon^{16})$ sketch size makes it considerably worse for small values of ϵ.

In contrast, our sketching dimension k is optimal among all non-adaptive bilinear sketches, due to the proof of part 1 of Theorem 31 of [NSW22] applied with $p = 2$. Indeed, the proof of that theorem gives a pair of distributions on matrices A with $\|A\|_F = \Theta(1)$ for which in one distribution A is PSD, while in the other it has a negative eigenvalue of value $-\Theta(\epsilon)$. That theorem shows $\Omega(1/\epsilon^4)$ non-adaptive vector-matrix-vector queries are required to distinguish the two distributions, which implies in our setting that necessarily $k = \Omega(1/\epsilon^2)$.

Concentration of Singular Values with Arbitrary Covariance Matrices. Of independent technical interest, we give the first bounds on the singular values of GB for an $n \times n$ matrix B and a (normalized) Gaussian matrix G with k rows when $k \ll n$. When taken together, our upper and lower bounds on singular values show for any $1 \leq \ell$ and $k \geq \Omega(\ell)$, that

$$\sigma_\ell(GB)^2 = \sigma_\ell(B)^2 \pm O\left(\frac{1}{\sqrt{k}}\right) \|B\|_F^2 \,. \tag{4.2}$$

Although there is a large body of work on the singular values of GB, to the best of our knowledge there are no quantitative bounds of the form above known. There is work upper bounding $\|GB\|_2$ for a fixed matrix B [Ver11], and classical work (see, e.g., [Ver10]) which bounds all the singular values of G when B is the identity, but we are not aware of concrete bounds that prove concentration around $\|GB\|_F^2$ of the form in (4.2) for general matrices B that we need.

Optimal Adaptive Matrix-Vector Query Lower Bound. A natural question is whether adaptivity can further reduce our sketching dimension. We show that at least in the matrix-vector product model, where one receives a sequence of matrix-vector products $Av^1, Av^2, \ldots, Av^r$ for query vectors $v^1, v^2, \ldots, v^r$ that may be chosen adaptively as a function of previous matrix-vector products, that necessarily $r = \Omega(1/\epsilon^2)$.

Note that our non-adaptive sketch GAG^T gives an algorithm in the matrix-vector product model by computing AG^T, and so $r = k = O(1/\epsilon^2)$. This shows that adaptivity does not help for eigenvalue estimation, at least in the matrix-vector product model.

Our hard instance is distinguishing a Wishart matrix of rank r from a Wishart matrix of rank

$r+2$ (the choice of $r+2$ rather than $r+1$ is simply for convenience). We first argue that for our pair of distributions, adaptivity does not help. This uses rotational invariance properties of our Wishart distribution, even conditioned on the query responses we have seen so far. In fact, our argument shows that without loss of generality, the optimal tester is a non-adaptive tester which just observes the leading principle submatrix of the input matrix A. We then explicitly bound the variation distance between the distributions of a Wishart matrix of rank r and one of rank $r+2$. We also give an alternative, but related proof based on distinguishing a random r dimensional subspace from a random $r+2$ dimensional subspace, which may be of independent interest. As an example, we note that this lower bound immediately recovers the $\Omega(1/\epsilon)$ matrix-vector lower bound for estimating the trace of a PSD matrix to within $(1 \pm \epsilon)$ multiplicative error [Mey+21; Jia+21], as well as the $\Omega(1/\epsilon^p)$ lower bound given in [WZZ22] for approximating the trace of A to additive $\epsilon \left\|A\right\|_p$ error (however the bound in [WZZ22] is more refined as it captures the dependence on failure probability).

These results substantially broaden a previous lower bound for the rank-estimation problem [Sun+21]. Whereas the hard instance in [Sun+21] requires some non-zero eigenvalues to be extremely small, we show that the rank estimation problem remains hard even when all nonzero eigenvalues have comparable size (or in fact, even when they are all equal).

4.2.2 Additional Work on Sampling in the Bounded Entry Model

Recent work has considered the spectral estimation problem for entry queries to bounded-entry matrices. The work of [Bha+21] gives an $\widetilde{O}(1/\epsilon^6)$ query algorithm for approximating all eigenvalues of a symmetric matrix to within $\epsilon \left\|A\right\|_F$ additive error, given a row-norm sampling oracle. However it remains open whether this bound can be improved to $\widetilde{O}(1/\epsilon^4)$ even for principal submatrix queries.

Our result shows that $O(1/\epsilon^4)$ queries is at least attainable under the much less restrictive model of vector-matrix-vector queries. In contrast to [Bha+21], our algorithm does not simply return the eigenvalues of our sketch. Indeed no such algorithm can exist as it would violate the one-sided lower bound of [NSW22].

4.3 Sketching Algorithm and Proof Outline

Algorithm 3

```
Require: $A \in \mathbb{R}^{d\times d}$ real symmetric, $k \in \mathbb{N}$.
  procedure SPECTRUM_APPX($A$,$k$)
      Sample $G \in \mathbb{R}^{k\times k}$ with i.i.d. $\mathcal{N}(0, 1/k)$ entries.
      $S \leftarrow GAG^T$
      For $i = 1, \ldots, k$, let $\alpha_i = \lambda_i(S) - \frac{1}{k}\operatorname{Tr}(S)$
      For $i = k+1, \ldots, d$, let $\alpha_i = 0$
      return $\alpha_1, \ldots, \alpha_d$ sorted in decreasing order
  end procedure
```

Theorem 4.1. *Let $A \in \mathbb{R}^{d\times d}$ be symmetric (not necessarily PSD) with eigenvalues $\lambda_1 \geq \ldots \geq \lambda_d$. For $k \geq \Omega(1/\epsilon^2)$, Algorithm 3 produces a sequence $(\mu_1, \ldots, \mu_d)$ such that $|\mu_i - \lambda_i| < \epsilon \left\|A\right\|_F$ for all i with probability at least* $3/5$.

4.3.1 Proof Outline

A natural idea is to split the spectrum of A into two pieces, A_1 and A_2, where A_1 consists of the large eigenvalues of A which are at least $\epsilon \left\|A\right\|_F$ in magnitude, and where A_2 contains the remaining spectral tail. The eigenvalues of GA_2G^T will all concentrate around $\operatorname{Tr}(A)$ up to $O(\epsilon)$ additive error.

We are then left with showing that the eigenvalues of GA_1G^T are $O(\epsilon)$ additive approximations to the nonzero eigenvalues of A_1. In order to do this we prove upper and lower bounds on the eigenvalues of GA_1G^T. For the upper bound (or lower bound if $\lambda_\ell(A_1)$ is negative) we give a general upper bound on the operator norm of GMG^T for a PSD matrix M with $\left\|M\right\|_F \leq 1$. By applying this result to various deflations of A_1 we are able to give an upper bound on all eigenvalues of A_1 simultaneously.

For the lower bound, we first prove the analogous result in the PSD case where it is much simpler. We then upgrade to the general result. To get a lower bound on $\lambda_\ell(GDG^T)$ in the general case, we construct an ℓ dimensional subspace S_ℓ so that $u^T GDG^T u$ is large for all unit vectors u in S_ℓ. A natural choice would be to take S_ℓ to be the image of $GD_{+,\ell}G^T$, where $D_{+,\ell}$ refers to D with all but the top ℓ positive eigenvalues zeroed out. We would then like to argue that the quadratic form associated to GD_-G^T is small in magnitude uniformly over S_ℓ. Unfortunately it need not be as small as we require, due to the possible presence of large negative eigenvalues in D_-. We therefore restrict our choice of S_ℓ to lie in the orthogonal complement of the largest r negative eigenvectors of GD_-G^T. Since we restrict the choice of S_ℓ we incur a cost, which damages our lower bound on $\lambda_\ell(GD_+G^T)$ slightly. However by choosing r carefully, we achieve a lower bound on $\lambda_\ell(GDG^T)$ of $\lambda_\ell(D) - O(\epsilon)$.

4.4 Proof of Theorem 4.1

In this section and the next, we provide upper and lower bounds on the eigenvalues of a sketched $d \times d$ matrix. We emphasize the results below will later be applied only to the matrix A_1 which is rank $O(1/\epsilon^2)$. Hence we will use the results below for $d = O(1/\epsilon^2)$.

4.4.1 Upper bounds on the sketched eigenvalues

The following result is a consequence of Theorem 1 in [CNW15] along with the remark following it.

Theorem 4.2. *Let $G \in \mathbb{R}^{m \times n}$ have i.i.d. $\mathcal{N}(0, 1/m)$ entries, and let A and B be arbitrary matrices with compatible dimensions. With probability at least $1 - \delta$,*

$$\left\|A^T G^T G B - A^T B\right\| \le \epsilon \sqrt{\|A\|^2 + \frac{\|A\|_F^2}{k}} \sqrt{\|B\|^2 + \frac{\|B\|_F^2}{k}},$$

for $m = O(\frac{1}{\epsilon^2}(k + \log \frac{1}{\delta}))$.

Lemma 4.3. *Let $D \in \mathbb{R}^{d\times d}$ have eigenvalues $\lambda_1 \geq \ldots \geq \lambda_d \geq 0$ where $\|D\|_F \leq 1$. Let $G \in \mathbb{R}^{t\times d}$ have $\mathcal{N}(0, 1/t)$ entries. The bound*

$$\left\|GD^{1/2}\right\|^2 \leq \lambda_1 + O\left(\frac{1}{\sqrt{m}}\right)$$

holds with probability at least $1 - \frac{1}{20}2^{-\min(m,1/\lambda_1^2)}$, provided that $t \geq \Omega(m+d)$.

Proof. We first decompose D into two parts $D = D_1 + D_2$ where D_1 contains the eigenvalues of D larger than $\lambda_1/2$ and D_2 contains the eigenvalues which are at most $\lambda_1/2$. Let x be an arbitrary unit vector and partition its support according to D_1 and D_2 so that $x = x_1 + x_2$. This allows us to write

$$\begin{aligned} x^T D^{1/2} G^T G D^{1/2} x &= x_1^T D_1^{1/2} G^T G D_1^{1/2} x_1 + x_2^T D_2^{1/2} G^T G D_2^{1/2} x_2 \\ &\quad + 2 x_1^T D_1^{1/2} G^T G D_2^{1/2} x_2 \\ &\leq \|x_1\|^2 \left\|D_1^{1/2} G^T G D_1^{1/2}\right\| + \\ &\quad \|x_2\|^2 \left\|D_2^{1/2} G^T G D_2^{1/2}\right\| \\ &\quad + 2\|x_1\|\,\|x_2\| \left\|D_1^{1/2} G^T G D_2^{1/2}\right\|. \end{aligned}$$

We bound each of these operator norms in turn by using Theorem 4.2 above.

Note that D_1 has support of size at most $4/\lambda_1^2$ since $\|D_1\|_F^2 \leq 1$, and so $\mathrm{Tr}(D_1) \leq \frac{4}{\lambda_1}$. Taking

$k = \frac{1}{\lambda_1^2}$, $\epsilon = \frac{1}{\sqrt{m}\lambda_1}$, and $\delta = \frac{1}{60} 2^{-1/\lambda_1^2}$ in Theorem 4.2 and applying the triangle inequality, we get

$$\begin{aligned}
\left\| D_1^{1/2} G^T G D_1^{1/2} \right\| &\leq \lambda_1 + \epsilon \left(\left\| D_1^{1/2} \right\|^2 + \frac{\left\| D_1^{1/2} \right\|_F^2}{k} \right) \\
&\leq \lambda_1 + \epsilon \left(\lambda_1 + \frac{\operatorname{Tr}(D_1)}{k} \right) \\
&\leq \lambda_1 + \epsilon \left(\lambda_1 + \frac{4}{\lambda_1 k} \right) \\
&\leq \lambda_1 + \frac{5}{\sqrt{m}}
\end{aligned}$$

Similarly for the second term, we note that $\operatorname{Tr}(D_2) \leq \frac{\lambda_1}{2} n$, and apply Theorem 4.2 with $k = d$, $\epsilon = 1/4$, and $\delta = \frac{1}{60} 2^{-m}$ to get

$$\begin{aligned}
\left\| D_2^{1/2} G^T G D_2^{1/2} \right\| &\leq \frac{\lambda_1}{2} + \epsilon \left(\frac{\lambda_1}{2} + \frac{\operatorname{Tr}(D_2)}{k} \right) \\
&\leq \frac{\lambda_1}{2} + \frac{1}{4} \left(\frac{\lambda_1}{2} + \frac{\operatorname{Tr}(D_2)}{d} \right) \\
&\leq \frac{\lambda_1}{2} + \frac{1}{4} \left(\frac{\lambda_1}{2} + \frac{\lambda_1}{2} \right) \\
&= \frac{3}{4} \lambda_1.
\end{aligned}$$

For the third term we choose $k = \sqrt{d}/\lambda_1$, $\epsilon = 1/(\sqrt{\lambda_1} m^{1/4})$, and $\delta = \frac{1}{60} 2^{-\sqrt{m}/\lambda_1}$ which gives

$$\begin{aligned}
\left\| D_1^{1/2} G^T G D_2^{1/2} \right\| &\leq \epsilon \sqrt{\lambda_1 + \frac{\operatorname{Tr}(D_1)}{k}} \sqrt{\frac{\lambda_1}{2} + \frac{\operatorname{Tr}(D_2)}{k}} \\
&\leq \epsilon \sqrt{\lambda_1 + \frac{\sqrt{d}}{k}} \sqrt{\frac{\lambda_1}{2} + \frac{\sqrt{d}}{k}} \\
&\leq \epsilon \left(\lambda_1 + \frac{\sqrt{d}}{k} \right) \\
&\leq 2 \frac{\sqrt{\lambda_1}}{m^{1/4}}.
\end{aligned}$$

Note that each application of Theorem 4.2 above allows G to have have $\Theta(m)$ rows provided that $m \geq d$. Also note that each failure probability above is bounded by $\frac{1}{60}2^{-\min(m,1/\lambda_1^2)}$, since $\frac{\sqrt{m}}{\lambda_1} \geq \min(m, \frac{1}{\lambda_1^2})$.

Thus we conclude with probability at least $1 - \frac{1}{20}2^{-\min(m,1/\lambda_1^2)}$, that

$$x^T D^{1/2} G^T G D^{1/2} x \leq \left(\lambda_1 + \frac{5}{\sqrt{m}}\right) \|x_1\|^2 + \frac{3}{4}\lambda_1 \|x_2\|^2 + 4\frac{\sqrt{\lambda_1}}{m^{1/4}} \|x_1\| \|x_2\| .$$

We view the right-hand expression as a quadratic form applied to the unit vector $(\|x_1\|, \|x_2\|)$. So its value is bounded by the largest eigenvalue of the 2×2 matrix

$$M = \begin{pmatrix} \lambda_1 + \frac{5}{\sqrt{m}} & \frac{2\sqrt{\lambda_1}}{m^{1/4}} \\ \frac{2\sqrt{\lambda_1}}{m^{1/4}} & \frac{3}{4}\lambda_1 \end{pmatrix}.$$

Suppose that $\lambda_1 + \beta$ with $\beta \geq 0$ is an eigenvalue of M. Then plugging into the characteristic polynomial gives

$$\frac{4\lambda_1}{\sqrt{m}} = \left(\beta - \frac{5}{\sqrt{m}}\right)\left(\beta + \frac{\lambda_1}{4}\right) \geq \frac{\lambda_1}{4}\left(\beta - \frac{5}{\sqrt{m}}\right),$$

from which it follows that $\beta \leq O\left(\frac{1}{\sqrt{m}}\right)$ as desired.

□

Lemma 4.4. *Let $D \in \mathbb{R}^{d\times d}$ (not necessarily PSD) have $\|D\|_F \leq 1$, and suppose $\lambda_\ell(D) \geq 0$. Let $G \in \mathbb{R}^{k\times d}$ have i.i.d. $\mathcal{N}(0, 1/k)$ entries. Then with probability at least $1 - \frac{1}{20}2^{-\min(\ell,\epsilon^{-2})}$,*

$$\lambda_\ell(GDG^T) \leq \lambda_\ell(D) + O(\epsilon),$$

for $k \geq \Omega(d + \frac{1}{\epsilon^2})$.

First we have the following, where D_+ and D_- denote the positive and negative semi-definite parts of D:

$$\begin{aligned}\lambda_\ell(GDG^T) &= \lambda_\ell(GD_+G^T - GD_-G^T) \\ &\leq \lambda_\ell(GD_+G^T) \\ &= \lambda_\ell(D_+^{1/2}G^TGD_+^{1/2}).\end{aligned}$$

Let $S_{d-\ell+1}$ be the span of a set of eigenvectors of D corresponding to $\lambda_\ell(D), \ldots, \lambda_d(D)$. Then by Courant-Fischer[2],

$$\begin{aligned}\lambda_\ell(GDG^T) &\leq \max_{v \in S_{d-\ell+1}, \|v\|=1} v^T D_+^{1/2} G^T G D_+^{1/2} v \\ &= \max_{v \in S_{d-\ell+1}, \|v\|=1} \left\| G D_+^{1/2} v \right\|^2 \\ &= \left\| G D_{+,-(\ell-1)}^{1/2} \right\|^2,\end{aligned}$$

where $D_{+,-(\ell-1)}$ is D_+ with the top $\ell - 1$ eigenvalues zeroed out. Now Lemma 4.3 applies, and gives

$$\lambda_\ell(GDG^T) \leq \lambda_\ell(D_+) + O\left(\epsilon\right) = \lambda_\ell(D) + O\left(\epsilon\right),$$

with probability at least $1 - \frac{1}{20} 2^{-\min(1/\epsilon^2, 1/\lambda_\ell(D)^2)}$, for $k \geq \Omega(d + \frac{1}{\epsilon^2})$. Finally, note that $\lambda_\ell(D) \leq \frac{1}{\sqrt{\ell}}$, so

$$2^{-\min(1/\epsilon^2, 1/\lambda_\ell(D)^2)} \leq 2^{-\min(1/\epsilon^2, \ell)}.$$

4.4.2 Lower bounds on the sketched eigenvalues

Lemma 4.5. *Let $M \in \mathbb{R}^{d \times d}$ be a PSD matrix with $\|M\|_F \leq 1$. Let $G \in \mathbb{R}^{m \times d}$ have i.i.d. $\mathcal{N}(0, \frac{1}{m})$ entries, where $m \geq \Omega(d + \log(1/\delta))$. Also let S_ℓ denote an arbitrary ℓ dimensional subspace of*

[2] For example see [Ver18] for a statement of the Courant-Fischer minimax theorem.

$\mathbb{R}^m$. *Then with probability at least* $1-\delta$*, we have*

$$\max_{v\in S_\ell, \|v\|=1} v^T G M G^T v \le 3\frac{\ell}{m} \|M\| .$$

Proof. Let $\Pi \in \mathbb{R}^{m\times\ell}$ has columns forming an orthonormal basis of S_ℓ. Then we can write

$$\max_{v\in S_\ell, \|v\|=1} v^T G M G^T v = \left\|\Pi^T G M G^T \Pi\right\| .$$

Using rotational invariance of G we note that $\Pi^T G$ is distributed as $\sqrt{\frac{\ell}{m}}\tilde{G}$ where $\tilde{G} \in \mathbb{R}^{\ell\times d}$ has i.i.d. $\mathcal{N}(0, \frac{1}{\ell})$ entries. Then

$$\left\|\Pi^T G M G^T \Pi\right\| = \frac{\ell}{m}\left\|\tilde{G} M \tilde{G}^T\right\| = \frac{\ell}{m}\left\|M^{1/2}\tilde{G}^T\tilde{G}M^{1/2}\right\| ,$$

which by taking $(\epsilon, k) = (1, d)$ in Theorem 4.2 is bounded by

$$\begin{aligned}\frac{\ell}{m}\left(\|M\| + \left(\left\|M^{1/2}\right\|^2 + \frac{\left\|M^{1/2}\right\|_F^2}{d}\right)\right) &= \frac{\ell}{m}\left(\|M\| + \left(\|M\| + \frac{\operatorname{Tr}(M)}{d}\right)\right) \\ &\le 3\frac{\ell}{m}\|M\| ,\end{aligned}$$

with probability at least $1-\delta$. Note that we used the bound $\operatorname{Tr}(M) \le d\|M\|$ in the final step. □

Lemma 4.6. *Let* $M \in \mathbb{R}^{d\times d}$ *be PSD with* $\|M\|_F \le 1$*, and let* $G \in \mathbb{R}^{k\times d}$ *have i.i.d.* $\mathcal{N}(0, \frac{1}{k})$ *entries.*

By choosing $k = \Theta(d + \frac{1}{\epsilon^2})$ *the bound*

$$\lambda_\ell(GMG^T) \ge \lambda_\ell(M) - \epsilon$$

holds with probability at least $1 - \frac{1}{40}2^{-\ell}$.

Proof. Recall that the non-zero eigenvalues of GMG^T coincide with those of $M^{1/2}G^TGM^{1/2}$, so

$$\lambda_\ell(GMG^T) = \lambda_\ell(M^{1/2}G^TGM^{1/2}).$$

By the Courant-Fischer theorem, there exists an ℓ dimensional subspace S_ℓ of $\mathbb{R}^d$ such that $\left\|M^{1/2}x\right\|^2 = x^TMx \geq \lambda_\ell(M)$ for all $x \in S_\ell$.

Now suppose that G is an $(\frac{\epsilon}{\lambda_\ell}, \ell, \frac{1}{40}2^{-\ell})$-OSE[3], which can be achieved by taking

$$k = \Theta\left(\frac{\lambda_\ell^2}{\epsilon^2}\left(\ell + \log\frac{10}{2^{-\ell}}\right)\right).$$

Since $\|M\|_F \leq 1$, we have $\lambda_\ell^2 \leq \frac{1}{\ell}$, so in fact $k = O(1/\epsilon^2)$ above.

Then with probability at least $1 - \frac{1}{10}2^{-\ell}$, the bound

$$\begin{aligned}
\left\|GM^{1/2}x\right\|^2 &\geq \left(1 - \frac{\epsilon}{\lambda_\ell(M)}\right)\left\|M^{1/2}x\right\|^2 \\
&\geq \left(1 - \frac{\epsilon}{\lambda_\ell(M)}\right)\lambda_\ell(M) \\
&\geq \lambda_\ell(M) - \epsilon
\end{aligned}$$

holds for all $x \in S_\ell$. By the Courant-Fischer theorem, this implies that $\lambda_\ell(M^{1/2}G^TGM^{1/2}) \geq \lambda_\ell(M) - \epsilon$ as desired. □

Lemma 4.7. *Suppose that $D \in \mathbb{R}^{d\times d}$ is a (not necessarily PSD) matrix with $\|D\|_F \leq 1$ and that $G \in \mathbb{R}^{k\times d}$ has i.i.d. $\mathcal{N}(0, 1/k)$ entries. If $\lambda_\ell(D) \geq 0$, then with probability at least $\frac{1}{20}2^{-\ell}$,*

$$\lambda_\ell(GDG^T) \geq \lambda_\ell(D) - \epsilon,$$

for $k \geq \Omega(d + \frac{1}{\epsilon^2})$.

[3] An (ϵ, k, δ)-OSE refers to an oblivious embedding that has $1 \pm \epsilon$ distortion over any given k dimensional subspace with probability at least $1 - \delta$.

Throughout the course of this argument we will need the parameters k and r to satisfy various inequalities. To streamline the proof we will list these assumptions here and later verify that they are satisfied with appropriate choices. The assumptions we will need are as follows:

1. $k \geq c_1 d$, where $c_1 \geq 1$ is an absolute constant
2. $k - r \geq \frac{c_2}{\epsilon^2}$ where c_2 is an absolute constant
3. $\frac{r}{k\sqrt{\ell}} \leq \epsilon$
4. $\frac{\ell}{k\sqrt{r}} \leq \epsilon$

To produce a lower bound on $\lambda_\ell(GDG^T)$ we will find a subspace S such that $v^T GDG^T v$ is large for all unit vectors v in S.

First we write $D = D_+ - (D_{-,-r} + D_{-,+r})$ where D_+ is the positive semi-definite part of D, D_- is the negative semi-definite part of D, $D_{-,+r}$ denotes D_- with all but the top r eigenvalues zeroed out, and $D_{-,-r} = D_- - D_{-,+r}$ (recall that r is the parameter from above which is to be chosen later). We also write

$$\begin{aligned} GDG^T &= GD_+G^T - GD_{-,+r}G^T - GD_{-,-r}G^T \\ &= G_1 D_+ G_1^T - G_2 D_{-,+r} G_2^T - G_3 D_{-,-r} G_3^T \end{aligned}$$

where each component is PSD, and where G_1, G_2, G_3 consist of the columns of G corresponding to the nonzero entries of D_+ and $D_{-,+r}$ and $D_{-,-r}$ respectively. In particular note that this decomposition shows that these three random matrices are mutually independent.

Let $W_r \subseteq \mathbb{R}^k$ denote the image of $D_{-,+r}$ so that $W_r^\perp = \ker(D_{-,+r})$. Let $\Pi_{W_r^\perp} \in \mathbb{R}^{k\times(k-r)}$ have columns forming an orthonormal basis for $W_r^\perp$. By rotational invariance of G, $G^T \Pi_{W_r^\perp}$ has i.i.d. $\mathcal{N}(0, 1/k)$ entries. Thus it follows that

$$\Pi_{W_r^\perp}^T GD_+G^T \Pi_{W_r^\perp} \sim \frac{k-r}{k} \tilde{G} D_+ \tilde{G}^T \sim \left(1 - \frac{r}{k}\right) \tilde{G} D_+ \tilde{G}^T,$$

where $\tilde{G} \in \mathbb{R}^{(k-r)\times d}$ has i.i.d $\mathcal{N}(0, \frac{1}{k-r})$ entries.

Now by Lemma 4.6, along with our second assumption above, we have

$$\lambda_\ell(\tilde{G}D_+\tilde{G}^T) \geq \lambda_\ell(D_+) - \epsilon = \lambda_\ell(D) - \epsilon,$$

with probability at least $1 - \frac{1}{40}2^{-\ell}$. Thus with the same probability, we then have

$$\lambda_\ell(\Pi_{W_r^\perp}^T G D_+ G^T \Pi_{W_r^\perp}) \geq \left(1 - \frac{r}{k}\right)(\lambda_\ell(D) - \epsilon) \geq \lambda_\ell(D) - 2\epsilon,$$

where the last inequality follows from our third assumption above, along with the observation that $\lambda_\ell(D) \leq \frac{1}{\sqrt{\ell}}$ which comes from the assumption $\|D\|_F \leq 1$.

If the above holds, then by the Courant-Fischer theorem, there exists a subspace $S_\ell \subseteq W_r^\perp \subseteq \mathbb{R}^k$ such that

$$x^T G D_+ G^T x \geq \lambda_\ell(D) - 2\epsilon \tag{4.3}$$

for all $x \in S_\ell$. Note that the construction of S_ℓ was independent of $GD_{-,-r}G^T$ by the comment above. Thus we may apply Lemma 4.5, along with our first assumption, to conclude that with probability at least $1 - \frac{1}{40}2^{-d}$,

$$\max_{v \in S_\ell, \|v\|=1} v^T G D_{-,-r} G^T v \leq 3\frac{\ell}{k}\|D_{-,-r}\| \leq 3\frac{\ell}{k}\frac{1}{\sqrt{r}}. \tag{4.4}$$

The last inequality holds because $\|D_-\|_F = 1$, which implies that $\lambda_r(D_-) \leq \frac{1}{\sqrt{r}}$.

Now let $u \in S_\ell$ be an arbitrary unit vector. We write

$$uGDG^T u^T = u^T G D_+ G^T u - u^T G D_{-,-r} G^T u - u^T G D_{-,+r} G^T u.$$

The last term vanishes by design since $x \in W_r^\perp$. We then bound the first term using equation 4.3

and the second term using equation 4.4 to get

$$uGDG^T u^T \geq (\lambda_\ell(D) - 2\epsilon) - 3\frac{\ell}{k}\frac{1}{\sqrt{r}} \geq \lambda_\ell(D) - 5\epsilon,$$

where the second inequality is form the fourth assumption above.

Our total failure probability in the argument above is at most $\frac{1}{40}2^{-d} + \frac{1}{40}2^{-\ell} \leq \frac{1}{20}2^{-\ell}$ as desired. It remains to choose parameters so that our four assumptions are satisfied. For this we take

$$\begin{aligned} &k \geq \max\left(c_1 d, \frac{c_2}{\epsilon^2} + \lfloor 2\ell \rfloor, \frac{2\sqrt{\ell}}{\epsilon}\right) \\ &r = \lfloor 2\ell \rfloor. \end{aligned}$$

Assumptions 1 and 2 clearly hold with this choice. For assumption 3, we have

$$\epsilon k\sqrt{\ell} \geq \epsilon\frac{2\sqrt{\ell}}{\epsilon}\sqrt{\ell} = 2\ell \geq r,$$

and for assumption 4,

$$\epsilon k\sqrt{r} \geq \epsilon\frac{2\sqrt{\ell}}{\epsilon}\sqrt{2\ell - 1} = 2\sqrt{\ell}\sqrt{2\ell - 1} \geq \ell,$$

since $\ell \geq 1$. Finally, since $\ell \leq d$, this gives a bound of $k = O(d + \frac{1}{\epsilon^2})$ as desired (note the inequality $\frac{\sqrt{d}}{\epsilon} \leq \max(d, 1/\epsilon^2)$ for bounding the last term in the $\max$ defining k).

4.4.3 Controlling the Tail

In this section we use Hanson-Wright[4] to bound the effect of the tail eigenvalues of A on the sketch. Note that our application Hanson-Wright relies on Gaussianity of G in order for the entries of $G^T u$ to be independent.

[4] See [Ver18] for a precise statement of Hanson-Wright.

Lemma 4.8. *Let $Y \in \mathbb{R}^{d \times d}$ be symmetric (not necessarily PSD) with $\|Y\| \leq \epsilon$ and $\|Y\|_F \leq 1$. Let $G \in \mathbb{R}^{k \times n}$ have i.i.d. $\mathcal{N}(0, 1/k)$ entries. For $k \geq \Omega(1/\epsilon^2)$ we have*

$$\left\| GYG^T - \frac{1}{k}\operatorname{Tr}(Y)I \right\| \leq O(\epsilon),$$

with probability at least $29/30$.

Proof. Let $u \in \mathbb{R}^k$ be an arbitrary fixed unit vector. Note that $G^T u$ is distributed as $\mathcal{N}(0, \frac{1}{k}I_d)$ and so

$$\mathbb{E}(u^T GYG^T u) = \frac{1}{k}\operatorname{Tr}(Y).$$

Set $\tilde{Y} = GYG^T - \frac{\operatorname{Tr}(Y)}{k}I$. By Hanson-Wright,

$$\begin{aligned}\Pr\left(\left|u^T \tilde{Y} u\right| \geq 30\epsilon\right) &= \Pr\left(\left|u^T GYG^T u - \frac{1}{k}\operatorname{Tr}(Y)\right| \geq 30\epsilon\right) \\ &\leq 2\exp\left(-0.1\min\left(\frac{(30\epsilon)^2 k^2}{\|Y\|_F^2}, \frac{(30\epsilon)k}{\|Y\|_2}\right)\right) \\ &\leq 2\exp\left(-\min\left(90\epsilon^2 k^2, 3k\right)\right).\end{aligned}$$

Note that in the final bound above we used the fact that $\|Y\|_2 \leq \epsilon$.

Let $\mathcal{N}$ be a net for the sphere in $\mathbb{R}^k$ with mesh size $1/3$, which may be taken to have size 9^k. By 4.4.3 in [Ver18],

$$\left\| G\tilde{Y}G^T \right\|_2 \leq 3 \sup_{x \in \mathcal{N}} |x^T G\tilde{Y}G^T x|.$$

By taking a union bound over the net and setting $k \geq \Omega(1/\epsilon^2)$, we then have

$$\Pr\left(\left\|\tilde{Y}\right\|_2 \geq 93\epsilon\right) \leq 2\exp\left(-\min\left(90\epsilon^2 k^2, 3k\right)\right) 9^k \leq \frac{1}{30},$$

for $\epsilon < 1$. □

4.4.4 Proof of Theorem 4.1

Proof. By rescaling, it suffices to consider that case $\|A\|_F = 1$. We start by decomposing A into two pieces $A = A_1 + A_2$, where A_1 is A with all eigenvalues smaller than ϵ in magnitude zeroed out.

To handle the large eigenvalues, we apply Lemma 4.4 and Lemma 4.7. Suppose that A_1 has n nonzero eigenvalues. Then we note that the nonzero eigenvalues of GA_1G^T have the same distribution as the eigenvalues of $\tilde{G}\tilde{A}_1\tilde{G}^T$ where $\tilde{A}_1$ is a symmetric $n \times n$ matrix with eigenvalues the same as the nonzero eigenvalues of A_1 and where $\tilde{G} \in \mathbb{R}^{k \times n}$ has i.i.d. $\mathcal{N}(0, 1/k)$ entries. This effectively means that we may treat A_1 has having dimension n when applying Lemma 4.4 and Lemma 4.7.

By taking a union bound over the positive eigenvalues of A_1 and applying Lemma 4.4 we get the upper bound $\lambda_\ell(GA_1G^T) \leq \lambda_\ell(A_1) + O(\epsilon)$ uniformly for all ℓ such that $\lambda_\ell(A_1) > 0$, with failure probability at most

$$\sum_{i=1}^{n} \frac{1}{20} 2^{-\min(\ell, \epsilon^{-2})} \leq \frac{1}{20} \sum_{i=1}^{n} 2^{-\ell} \leq \frac{1}{20},$$

where the the first inequality follows from the fact that $\ell \leq n \leq 1/\epsilon^2$, which in turn holds since $\|A_1\|_F \leq 1$.

Similarly Lemma 4.7 gives the lower bound $\lambda_\ell(GA_1G^T) \leq \lambda_\ell(A_1) - \epsilon$ uniformly for all ℓ such that $\lambda_\ell(A_1) > 0$, with failure probability at most

$$\sum_{i=1}^{\ell} \frac{1}{20} 2^{-\ell} \leq \frac{1}{20}.$$

Thus with at least $9/10$ probability, $\left|\lambda_\ell(GA_1G^T) - \lambda_\ell(A_1)\right| \leq O(\epsilon)$ for all ℓ such that $\lambda_\ell(A_1) > 0$. By applying the above argument to $-A_1$ we get the same guarantee for the negative eigenvalues, i.e. $\left|\lambda_{k-\ell}(GA_1G^T) - \lambda_{k-\ell}(A_1)\right| \leq O(\epsilon)$ for all ℓ such that $\lambda_{k-\ell}(A_1) < 0$. By a union bound, the positive and negative guarantees hold together with failure probability at most $1/5$.

Next we apply the tail bound of Lemma 4.8 to control the perturbations resulting from the tail. By the triangle inequality,

$$\begin{aligned}\left\|GA_2G^T - \frac{1}{k}\operatorname{Tr}(GAG^T)I\right\| &\leq \left\|GA_2G^T - \frac{1}{k}\operatorname{Tr}(A_2)I\right\| \\ &\quad + \left\|\frac{1}{k}\operatorname{Tr}(A_2)I - \frac{1}{k}\operatorname{Tr}(GAG^T)I\right\| \\ &\leq \left\|GA_2G^T - \frac{1}{k}\operatorname{Tr}(A_2)I\right\| \\ &\quad + \frac{1}{k}\left|\operatorname{Tr}(A_2) - \operatorname{Tr}(GA_2G^T)\right| \\ &\quad + \frac{1}{k}\left|\operatorname{Tr}(GA_1G^T)\right|\end{aligned}$$

The first of these terms is bounded by $O(\epsilon)$ with failure probability at most $1/30$ by Lemma 4.8. The second term is easily bounded by $O(\epsilon)$ with failure probability at most $1/30$ since $\operatorname{Tr}(GA_2G^T)$ is a trace estimator for A_2 with variance at $O(\|A_2\|_F) = O(1)$ (in fact the variance is even smaller). For the third term, note that A_1 has at most $1/\epsilon^2$ nonzero eigenvalues, so $\operatorname{Tr}(A_1) \leq \frac{1}{\epsilon}\|A\|_F \leq \frac{1}{\epsilon}$. Thus since $\operatorname{Tr}(GA_1G^T)$ is a trace estimator for A_1, the third term is bounded by $O(\epsilon)$ with failure probability at most $1/30$. Thus we have the bound

$$\left\|GA_2G^T - \frac{1}{k}\operatorname{Tr}(GAG^T)I\right\| \leq O(\epsilon),$$

with failure probability at most $1/10$. This gives the bound

$$\begin{aligned}\lambda_\ell(GAG^T) &= \lambda_\ell(GA_1G^T + GA_2G^T)\\ &= \lambda_\ell\left(GA_1G^T + \frac{1}{k}\operatorname{Tr}(GAG^T)I + GA_2G^T - \frac{1}{k}\operatorname{Tr}(GAG^T)I\right)\\ &= \lambda_\ell\left(GA_1G^T + \frac{1}{k}\operatorname{Tr}(GAG^T)I\right)\\ &\quad \pm \left\|GA_2G^T - \frac{1}{k}\operatorname{Tr}(GAG^T)I\right\|_2\\ &= \lambda_\ell(GA_1G^T) + \frac{1}{k}\operatorname{Tr}(GAG^T) \pm O(\epsilon).\end{aligned}$$

Setting $\widehat{\lambda_\ell} = \lambda_\ell(GAG^T) - \frac{1}{k}\operatorname{Tr}(GAG^T)$, we therefore have $\widehat{\lambda_\ell} = \lambda_\ell(GA_1G^T) \pm O(\epsilon)$. Combining with the bounds above gives $\widehat{\lambda_\ell} = \lambda_\ell(A_1) \pm O(\epsilon)$ if $\lambda_\ell(A_1) > 0$ and $\widehat{\lambda_{k-\ell}} = \lambda_{k-\ell}(A_1) \pm O(\epsilon)$ if $\lambda_{k-\ell}(A_1) > 0$.

Thus there is a subset of n of the $\widehat{\lambda_\ell}$'s which provide an $O(\epsilon)$ additive approximation to the set of eigenvalues of A which are at least ϵ. The above bound shows that the remaining $\widehat{\lambda_\ell}$'s are bounded by $O(\epsilon)$ and the result follows. □

4.5 Lower bounds for eigenvalue estimation

We will use the Wishart distribution throughout this section which is defined as follows.

Definition 4.9. *The n dimensional Wishart distribution with r degrees of freedom $W(n, r)$ is the distribution of GG^T where $G \in \mathbb{R}^{n\times r}$ has i.i.d. standard normal entries.*

In this section we show that $\Omega(r)$ matrix-vector queries are necessary to determine the rank of a matrix with all nonzero entries $\Omega(1)$. Specifically we show that distinguishing between $W(n, r)$ and $W(n, r + 2)$ requires $\Omega(r)$ queries for $r \leq O(n)$. In Appendix 4.6.1 we sketch a proof of a similar lower bound for determining the rank of the orthogonal projection onto a random subspace.

For now we consider the following problem.

Problem 4.10. *Given a matrix A sampled from either $\mathcal{D}_1 = W(n,r)$ or $\mathcal{D}_2 = W(n, r+2)$ each with equal probability, decide between $\mathcal{D}_1$ and $\mathcal{D}_2$ with at least $2/3$ probability, using (possibly adaptive) matrix-vector queries to A.*

We first make note of the following result, which is effectively a version of Lemma 13 from [Bra+20], adapted to Wishart matrices $W(n,r)$ with n and r not necessarily equal. This will allow us to show that adaptivity is unhelpful, and hence reduce to studying the non-adaptive case.

Proposition 4.11. *Let $A \sim W(n,r)$, and let $k < r \leq n$. Then the conditional distribution $A|\{Ae_1 = x_1, \dots, Ae_k = x_k\}$ can be written as*

$$M_k + \operatorname{diag}(0_{k\times k}, W(n-k, r-k)),$$

where $M_k \in \mathbb{R}^{n\times n}$ has rank at most k and depends only on $x_1, \dots, x_k$. In particular M_k does not depend on r.

Proof. Write $A = GG^T$ where $G \in \mathbb{R}^{n\times r}$ has i.i.d. $\mathcal{N}(0,1)$ entries. Write $g_1, g_2, \dots$ for the rows of G. We first consider the conditional distribution $A|\{Ae_1 = x_1\}$. In other words, we are conditioning on the events $\langle g_1, g_i\rangle = x_{1i}$ for all i. By rotational invariance, we may additionally condition on $g_1 = \sqrt{x_{11}}e_1$ without changing the resulting distribution. Then for $i > 1$, the conditional distribution of g_i can be written as $\frac{x_{1i}}{\sqrt{x_{11}}}e_1 + h_i$ where h_i is distributed as $\mathcal{N}(0, I_{n-1})$ in the orthogonal complement of e_1. It follows from this that we can write

$$A|\{Ae_1 = x_1\} \sim \frac{1}{x_{11}}x_1x_1^T + \operatorname{diag}(0, W(n-1, r-1)). \tag{4.5}$$

So we have $M_1 = \frac{1}{x_{11}}x_1x_1^T$. Now we apply the above line inductively.

For $j < r$, let $W_j \sim \operatorname{diag}(0_{k\times k}, W(n-j, r-j))$, and write

$$\begin{aligned} A|\{Ae_1 = x_1, \dots Ae_{j+1} = x_j\} &\sim (A|\{Ae_1 = x_1, \dots Ae_j = x_j\})\,|\{Ae_{j+1} = x_{j+1}\} \\ &\sim (M_j + W_j)|\{(M_j + W_j)e_{j+1} = x_{j+1}\} \\ &\sim (M_j + W_j)|\{W_j e_{j+1} = x_{j+1} - M_j e_{j+1}\} \\ &\sim (M_j + W_j)|\{W_j e_{j+1} = v_{j+1}\} \\ &\sim M_j + (W_j|\{W_j e_{j+1} = v_{j+1}\}) \end{aligned}$$

where we set $v_{j+1} = x_{j+1} - M_j e_{j+1}$.

By applying 4.5,

$$\{W_j e_{j+1} = v_{j+1}\} = \frac{1}{v_{j+1,j+1}} v_{j+1} v_{j+1}^T + W_{j+1}.$$

Hence we can take

$$M_{j+1} = M_j + \frac{1}{v_{j+1,j+1}} v_{j+1} v_{j+1}^T,$$

and the induction is complete.

□

Proposition 4.12. *Of all (possibly adaptive) algorithms for Problem 4.10 which make $k \le r$ queries, there is an optimal such algorithm (in the sense of minimizing the failure probability), which queries on the standard basis vectors $e_1, \dots, e_k$.*

Proof. Let s be either r or $r + 2$ corresponding to which of $\mathcal{D}_1$ and $\mathcal{D}_2$ is sampled from. By rescaling, we assume that only unit vectors are queried.

We argue by induction. Since $\mathcal{D}_1$ and $\mathcal{D}_2$ are rotationally invariant, we may without loss of generality take the first query to be e_1.

Now suppose inductively that there is an optimal k query algorithm $\mathcal{A}$ whose first j queries are always $e_1, \dots, e_j$. Suppose on a fixed run, that $Ae_1 = x_1, \dots, Ae_j = x_j$. By Proposition 4.11, we may write the resulting conditional distribution as

$$A|\{Ae_1 = x_1, \ldots Ae_j = x_j\} = M_j + A_j,$$

where M_j depends deterministically on $x_1, \ldots, x_j$ (and not on s), and $A_j \sim \operatorname{diag}(0_{j\times j}, W(n - j, s - j))$.

Now since M_j is know to $\mathcal{A}$, we may assume that on iteration $j + 1$, $\mathcal{A}$ is given matrix-vector query access to A_j, rather than to A. Since the first j rows and columns of A_j are filled with zeros, we may assume that $\mathcal{A}$ queries on a vector in $\operatorname{span}\{e_{j+1}, \ldots, e_n\}$. Then by rotational invariance of $W(n - j, s - j)$, we may take $\mathcal{A}$ to query on e_j on iteration $j + 1$. This completes the induction, and the claim follows. □

In light of the previous result, only non-adaptive queries are necessary. In fact we can make an even stronger claim. Let E_k denote the matrix with columns $e_1, \ldots, e_k$. The previous proposition showed that an optimal tester only needs to observe AE_k, the first k columns of A. In fact, only $E_k^T AE_k$, the leading principal submatrix of A is relevant. We first state a simple fact that drives the argument.

Proposition 4.13. *Let $X \in k \times r_1$ and $Y \in k \times r_2$ be fixed matrices such that $XX^T = YY^T$. Let $v_1 \in \mathbb{R}^{r_1}$ and $v_2 \in \mathbb{R}^{r_2}$ have i.i.d. standard normal entries. Then Xv_1 and Yv_2 have the same distribution.*

Proof. Suppose without loss of generality that $r_2 \geq r_1$. Then since $XX^T = YY^T$, there is an orthogonal matrix $U \in \mathbb{R}^{r_2\times r_2}$ such that

$$YU = [X, 0_{k\times(r_1-r_2)}].$$

Now let $g \in \mathbb{R}^{r_2}$ have i.i.d. standard normal entries. By rotational invariance $Ug \in \mathbb{R}^{r_2}$ does as well. So YU has the same distribution as Yv_2. Also $[X, 0_{k\times(r_1-r_2)}]g$ is distributed as Xv_1, so Xv_1 and Yv_2 have the same distribution as desired. □

Proposition 4.14. *Suppose that $A_1 \sim W(n,r)$ and $A_2 \sim W(n, r+2)$. Then for $k \leq r$,*

$$\mathrm{TV}(A_1E_k, A_2E_k) = \mathrm{TV}(E_k^TA_1E_k, E_k^TA_2E_k).$$

Proof. Let $G_1 \in \mathbb{R}^{k\times r}$ and $H_1 \in \mathbb{R}^{(n-k)\times r}$ have i.i.d. standard normal entries. Similarly let $G_2 \in \mathbb{R}^{k\times(r+2)}$ and $H_2 \in \mathbb{R}^{(n-k)\times(r+2)}$ have i.i.d. standard normal entries.

By the definition of the Wishart distribution, the joint distribution of the entries of A_1E_k is precisely that of $(G_1G_1^T, H_1G_1^T)$ and similarly for A_2E_k. Hence,

$$\mathrm{TV}(A_1E_k, A_2E_k) = \mathrm{TV}\left((G_1G_1^T, H_1G_1^T), (G_2G_2^T, H_2G_2^T)\right).$$

For a fixed matrix M of the appropriate dimensions, we consider the conditional distribution $H_iG_i^T|\{G_iG_i^T = M\}$ for $i = 1, 2$. The rows of this random matrix are independent (since the rows of H_i are independent), and by Proposition 4.13 the distribution of each row is a function of M. Hence it follows that

$$H_1G_1^T|\{G_1G_1^T = M\} = H_2G_2^T|\{G_2G_2^T = M\}$$

for all M. Therefore,

$$\mathrm{TV}\left((G_1G_1^T, H_1G_1^T), (G_2G_2^T, H_2G_2^T)\right) = \mathrm{TV}(G_1G_1^T, G_2G_2^T).$$

Since $E_k^TA_iE_k$ has the same distribution as $G_iG_i^T$, the claim follows. □

Our problem is now reduced to that of determining the degrees of freedom of a Wishart from observing the top corner (which is itself Wishart). We will give a lower bound for this problem.

Our proof uses the following version of Theorem 5.1 in [Jon82].

Theorem 4.15. *Let $\alpha \in (0,1)$ be a constant, and let $n, r \to \infty$ simultaneously, with $n/r \to \alpha$.*

Then

$$\frac{\det(W(n,r))}{(r-1)(r-2)\dots(r-n)} \to e^{\mathcal{N}(0,-2\log(1-\alpha))},$$

where the convergence is in distribution.

Lemma 4.16. *Let $\alpha = 0.1$. There exists a constant c so that if $r \geq c$, then*

$$\mathrm{TV}\left(W(\lfloor \alpha r \rfloor, r), W(\lfloor \alpha r \rfloor, r+2)\right) \leq 0.2.$$

Proof. We write $n = \lfloor \alpha r \rfloor$ with the understanding that n is a function of r. Let $\mu_{n,r}$ be the measure on $\mathbb{R}^{n(n+1)/2}$ associated to $W(n,r)$, and let $f_{n,r}$ be the corresponding density function (with respect to the Lebesgue measure). Also let $\Delta_+ \subseteq \mathbb{R}^{n(n+1)/2}$ be the PSD cone. Then we have

$$\begin{aligned}\mathrm{TV}(W(n,r), W(n,r+2)) &= \int_{\Delta_+} \left(f_{n,r}(A) - f_{n,r+2}(A)\right)_+ d\lambda \\ &= \int_{\Delta_+} \left(1 - \frac{f_{n,r+2}(A)}{f_{n,r}(A)}\right)_+ d\mu_{n,r}\end{aligned}$$

We recall the following standard formula for the density of the Wishart distribution (see [And62] for example):

$$f_{n,r}(A) = \frac{(\det A)^{\frac{1}{2}(r-n-1)} e^{-\frac{1}{2}\operatorname{Tr}(A)}}{\sqrt{2}^{rn} \pi^{\frac{1}{4}n(n-1)} \prod_{i=1}^{n} \Gamma\left(\frac{1}{2}(r+1-i)\right)}.$$

Cancelling and applying the identity $\Gamma(x+1) = x\Gamma(x)$ gives

$$\begin{aligned}\frac{f_{n,r+2}(A)}{f_{n,r}(A)} &= \frac{\det A}{2^n}\prod_{i=1}^{n}\frac{\Gamma\left(\frac{1}{2}(r+1-i)\right)}{\Gamma\left(1+\frac{1}{2}(r+1-i)\right)}\\ &= \frac{\det A}{2^n}\prod_{i=1}^{n}\frac{1}{\frac{1}{2}(r+1-i)}\\ &= \frac{\det A}{r(r-1)\dots(r-n+1)}.\end{aligned}$$

This gives

$$\begin{aligned}\mathrm{TV}(W(n,r),&W(n,r+2)) =\\ &\int_{\Delta_+}\left(1-\frac{\det A}{r(r-1)\dots(r-n+1)}\right)_+ d\mu_{n,r}(A)\\ &= \mathbb{E}_{A\sim W(n,r)}\left(1-\frac{\det A}{r(r-1)\dots(r-n+1)}\right)_+.\end{aligned}$$

Therefore it suffices to bound this expectation.

Since $\frac{r-n}{r}\to(1-\alpha)$ as $r\to\infty$ we have from Theorem 4.15 that

$$\frac{\det W(n,r)}{r(r-1)\dots(r-n+1)}\to(1-\alpha)e^{\mathcal{N}(0,-2\log(1-\alpha))}.$$

Therefore

$$\mathrm{TV}(W(n,r),W(n,r+2))\to\mathbb{E}_{x\sim\mathcal{N}(0,-2\log(1-\alpha))}\left[1-(1-\alpha)e^x\right]_+,$$

where swapping the limit with the expectation was justified since the random variables in the limit were all bounded by 1. This last expectation may be computed numerically to be approximately 0.1815 and the claim follows. □

Theorem 4.17. *Suppose that $r\geq C_1$ and $d\geq C_2 r$ for absolute constants C_1 and C_2. Let $\mathcal{A}$ be an* *adaptive algorithm making k matrix-vector queries, which correctly decides between $\mathcal{D}_1$ and $\mathcal{D}_2$ with $2/3$ probability. Then $k \geq r/10$.*

Proof. Consider a protocol which makes k matrix-vector queries. By Proposition 4.12 and Proposition 4.14 it suffices to consider non-adaptive protocols which observe $E_k^T \Pi E_k$. Suppose that A is either drawn from $\mathcal{D}_1$ or $\mathcal{D}_2$ and hence distributed as $W(k, r)$ or $W(k, r+2)$. Lemma 4.16 now implies that distinguishing these distributions requires $k \geq r/10$ as desired.

□

Corollary 4.18. *An algorithm which estimates all eigenvalues of any matrix A up to $\epsilon \left\|A\right\|_F$ error, with $3/4$ probability must make at least $\Omega(1/\epsilon^2)$ matrix-vector queries.*

Proof. The nonzero eigenvalues of $W(n, r)$ are precisely the squared singular values of an $n \times r$ matrix with i.i.d. Gaussian entries. So by standard bounds (see [Ver18] for example), the nonzero eigenvalues of $W(n, r)$ and $W(n, r + 2)$ are bounded between $\frac{1}{2}n$ and $2n$ with high probability as long as $n \geq Cr$ for an absolute constant C. Since $W(n, r)$ has rank r, the Frobenius norm of $W(n, r)$ is bounded by $2n\sqrt{r}$, and similarly for $W(n, r+2)$. Thus setting $\alpha = \frac{1}{10\sqrt{r+2}}$, we see that an algorithm which estimates all eigenvalues of a matrix to $\alpha \left\|A\right\|_F$ additive error could distinguish $W(n, r)$ from $W(n, r+2)$, and hence by Theorem 4.17 must make at least $r/10$ queries. The result follows by setting $r = \Theta(1/\epsilon^2)$. □

4.6 Appendix

4.6.1 Rank estimation lower bound from random projections

In this section, we show a lower bound on determining the rank of a random orthogonal projection from matrix-vector queries. The key intuition is that running a power-method type algorithm is unhelpful since projections are idempotent. This suggests that adaptivity should be unhelpful, and indeed this is the case.

Throughout this section, we let $\mathcal{D}_1 = \mathcal{D}_1(d, r)$ be an orthogonal projection $\mathbb{R}^d \to \mathbb{R}^d$ onto a random r dimensional subspace (sampled from the rotationally invariant measure), and let $\mathcal{D}_2$ be an orthogonal projection onto a random $r + 2$ dimensional subspace. Let $\mathcal{D}$ be the distribution obtained by sampling from either $\mathcal{D}_1$ or $\mathcal{D}_2$ each with probability $1/2$.

We first show that adaptivity is unhelpful in distinguishing $\mathcal{D}_1$ from $\mathcal{D}_2$. To prove this, we first make a simple observation.

Observation 4.19. *Suppose that $\mathcal{P}_1$ and $\mathcal{P}_2$ are any distributions over matrices, and let U be an orthogonal matrix. Suppose that x_1 is an optimal first query to distinguish $\mathcal{P}_1$ and $\mathcal{P}_2$. Then Ux_1 is an optimal first query to distinguish $U\mathcal{P}_1U^T$ and $U\mathcal{P}_2U^T$.*

Lemma 4.20. *Suppose that there is a (possibly randomized) adaptive algorithm $\mathcal{A}$ which makes k matrix-vector queries to an orthogonal matrix $\Pi \sim \mathcal{D}$ and then decides whether Π was drawn from $\mathcal{D}_1$ or $\mathcal{D}_2$ with advantage β. Then there is a non-adaptive algorithm which queries on $e_1, \ldots, e_k$ and also achieves advantage β.*

Proof. By Yao's principle, it suffices to consider deterministic protocols, so we will restrict ourselves to deterministic protocols in what follow.

First, let us say that an adaptive protocol making queries $v_1, v_2, \ldots$ is *normalized* if for each i, v_{i+1} is in the orthogonal complement of $\mathrm{span}(v_1, v_2, \ldots, v_i, \Pi v_1, \ldots \Pi v_i)$, and $v_i \neq 0$. We will argue that all normalized protocols making k queries achieve the same advantage.

We first observe that all choices of v_1 are equivalent, which is a consequence of rotational invariance along with the observation above.

Suppose that a normalized algorithm makes queries $v_1, \ldots, v_j$ and receives values $y_1, \ldots, y_j$ in the first j rounds. We observe that the conditional distribution of Π under these observations is invariant under the group of orthogonal transformations stabilizing $x_1, \ldots, x_j, y_1, \ldots, y_j$. Applying the observation to this conditional distribution, again shows that all x_{j+1} are equivalent since the stabilizer of $x_1, \ldots, x_j, y_1, \ldots, y_j$ acts transitively on their orthogonal complement.

Finally we observe that a non-adaptive algorithm which queries on $e_1, \dots, e_k$ can almost surely simulate a normalized protocol. Indeed let P_j denote projection onto $\mathrm{span}(e_1, \dots, e_j, \Pi e_1, \dots \Pi e_j)$. Then $e_1, P_1 e_2, \dots, P_{k-1} e_k$ is almost surely a normalized protocol. Moreover $\Pi P_{j-1} e_j$ may be computed for each j, since the values of $\Pi e_1, \dots \Pi e_j, \Pi^2 e_1, \Pi^2 e_j$ are all known (this uses that Π is a projection and hence idempotent).

□

We are now able to turn our attention to non-adaptive algorithms. Let $E_k \in \mathbb{R}^{d \times k}$ denote the matrix $[e_1, \dots, e_k]$. As we saw above a general matrix-vector query algorithm might as well observe ΠE_k. As in our argument for Wishart matrices, our next observation is that only the top $k \times k$ corner is useful.

Lemma 4.21. *Suppose that* $\Pi_1 \sim \mathcal{D}_1$ *and* $\Pi_2 \sim \mathcal{D}_2$. *We have that*

$$\mathrm{TV}(\Pi_1 E_k, \Pi_2 E_k) = \mathrm{TV}(E_k^T \Pi_1 E_k, E_k^T \Pi_2 E_k).$$

Proof. Let $\Pi E_k = [M_1; M_2]$ where $M_1 \in \mathbb{R}^{k \times k}$ and $M_2 \in \mathbb{R}^{(d-k) \times k}$. Observe that since Π is a projection, $M_2^T M_2 = M_1 - M_1 M_1^T$.

Let the orthogonal group $SO(n)$ act on Π via conjugation. Let H be the stabilizer of M_1 under the action, i.e., the set of U such that $U^T \Pi U E_k = [M_1, M_2']$ for some M_2'. We claim that the orbit of M_2 under H is $\{X : X^T X = M_1 - M_1 M_1^T\}$. To see this, simply observe that H is contained in the stabilizer of $e_1, \dots, e_k$, which is isomorphic copy of $SO(n-k)$ acting on $\mathrm{span}(e_1, \dots, e_k)^\perp$. This latter group acts transitively on $\{X : X^T X = M_1 - M_1 M_1^T\}$ under left multiplication as desired.

This implies that the conditional distribution of M_2 on observing M_1 is uniform over $\{X : X^T X = M_1 - M_1 M_1^T\}$. Since the conditional distribution is independent of r, the result follows.

□

Next we leverage a known result showing that a small principal minor of a random rotation is indistinguishable from Gaussian. This allows to observe that $E_k^T \Pi E_k$ is nearly indistinguishable from a Wishart distribution when d is large.

Lemma 4.22. *Suppose that $r \geq C_1$ and $d \geq C_2 r^2$ for some absolute constants C_1, C_2, and let $\Pi \sim \mathcal{D}_1(d, r)$ with $k \leq r$. Then*

$$\mathrm{TV}(E_k^T \Pi E_k, W(k, r)) \leq 0.1.$$

Proof. Note that Π can be written as $(UE_r)(UE_r)^T$ where U is a random orthogonal matrix sampled according to the Haar measure. Let $G \in \mathbb{R}^{k \times r}$ be a matrix with i.i.d. $\mathcal{N}(0, \frac{1}{d})$ entries. Then we have

$$\begin{aligned}
\mathrm{TV}(E_k^T \Pi E_k, G^T G) &= \mathrm{TV}(E_k^T (UE_r)(UE_r)^T E_k, G^T G) \\
&= \mathrm{TV}((E_k^T U E_r)(E_k^T U E_r)^T, G^T G) \\
&\leq \mathrm{TV}(E_k^T U E_r, G^T),
\end{aligned}$$

where the last line follows from the data processing inequality.

Note that $E_k^T U E_r$ is simply the top $k \times r$ corner of a random orthogonal matrix, and G^T is a $k \times r$ matrix with i.i.d. $\mathcal{N}(0, \frac{1}{d})$ entries. The claim now follows from Theorem 1 of [Jia06]. □

Theorem 4.23. *Suppose that $r \geq C_1$ and $d \geq C_2 r^2$ for absolute constants C_1 and C_2. Let $\mathcal{A}$ be an adaptive algorithm making k matrix-vector queries to a sample from $\mathcal{D}$ which correctly decides between $\mathcal{D}_1$ and $\mathcal{D}_2$ with $3/4$ probability. Then $k \geq r/10$.*

Proof. Consider a protocol which makes k matrix-vector queries. By Lemma 4.20 and Lemma 4.21 it suffices to consider non-adaptive protocols which observe $E_k^T \Pi E_k$. Suppose that Π_1 and Π_2 are random projections drawn from $\mathcal{D}_1$ and $\mathcal{D}_2$ respectively. Then by Lemma 4.22, we have

$$\mathrm{TV}(E_k^T \Pi_1 E_k, W(k, r)) \leq 0.1$$

and

$$\mathrm{TV}(E_k^T \Pi_2 E_k, W(k, r+2)) \leq 0.1.$$

By the triangle inequality,

$$\mathrm{TV}(E_k^T \Pi_1 E_k, E_k^T \Pi_2 E_k) \leq 0.2 + \mathrm{TV}(W(k,r), W(k, r+2)),$$

which in turn is bounded by 0.4 by Lemma 4.16 for $k < r/10$. The result follows. □

4.6.2 Faster sketching

In this section, we make several observations, which allow for our sketch to be applied more efficiently.

Optimized runtime of dense sketches

We observe that known results for fast rectangular matrix multiplication allow for the sketch to be applied in near linear time, provided that d is sufficiently large relative to ϵ.

[GU18] shows that multiplication of a $d \times d^{\alpha}$ matrix and a $d^{\alpha} \times d$ matrix, may be carried out in $O(d^{2+\gamma})$ time for any $\gamma > 0,$ for $\alpha \geq 0.32$. Since this is known to require the same number of operations as multiplying a $d^{\alpha} \times d$ and a $d \times d$ matrix (see [Le 12] for example), our dense Gaussian sketch may be applied in time $O(d^{2+\gamma})$ as long as the sketching dimension k is bounded by $O(d^{.32})$. Since we take $k = O(1/\gamma^2)$, our sketch may be applied in near-linear time as long as $k = 1/\gamma^2 \leq O(d^{.32})$ or equivalently when $\gamma \gtrsim d^{-0.16}$.

Faster sketching for sparse PSD matrices

We observe that a variant of our sketch may be applied quickly to sparse matrices, at least when the input matrix is PSD.

Suppose without loss of generality that $\|A\|_F = 1$. Our first step is to apply the ℓ_2 heavy hitters sketch, SAT^T of [AN13]. While they choose S and T to be Gaussian, it can be verified that their analysis carries through as long as S and T are ϵ-distortion oblivious subspace embeddings on k dimensional subspaces. We choose to take S and T to be the sparse embedding matrices of [CNW15].

Since S and T are in particular $O(1)$ distortion Johnson-Lindenstrauss maps, $\left\|SAT^T\right\|_F \leq 2\left\|A\right\|_F$ with good probability. Now, by setting $k = \text{poly}(1/\epsilon)$ in theorem 1.2 of [AN13], we get that the singular values of SAT^T approximate the top $1/\epsilon^2$ eigenvalues of A to within ϵ additive error (the remaining eigenvalues of A are $O(\epsilon)$ and so may be estimated as 0).

Write $M = SAT^T$. It now suffices to estimate the singular values of M to $O(\epsilon)$ additive error. For this we first symmetrize M forming the matrix

$$M_{\text{sym}} = \begin{pmatrix} 0 & M \\ M^T & 0 \end{pmatrix}. \tag{4.6}$$

Note that the eigenvalues of M_{sym} are precisely the singular values of M. To approximate the eigenvalues of M_{sym} we use our dense Gaussian sketch, yielding the optimal sketching dimension of $O(1/\epsilon^2)$. Since M_{sym} has dimensions $\text{poly}(1/\epsilon)$, this last sketch may be carried out in $\text{poly}(1/\epsilon)$ time.

Since S and T were chosen to be sparse embedding matrices, the full sketch runs in $\text{poly}(\frac{1}{\epsilon})\text{nnz}(A)$ time. To summarize, our final sketching dimension is $O(1/\epsilon^2)$ on each side, and we approximate all eigenvalues to within $\epsilon\left\|A\right\|_F$ additive error.

CHAPTER 5

Linear Regression in the Row-partition Model

In many modern applications a large amount of data is spread across numerous servers. Communicating between the servers may be expensive, and so we often seek protocols that minimize the amount of communication required.

Here we consider linear regression in the so-called “row-partion model”. This means that there is a d-dimensional linear system $Ax = b$ whose rows are spread across s servers. We are primarily interested in the overdetermined setting where the goal is to find the least squares solution x^*. In fact, we will relax this requirement slightly and solve the *approximate* ℓ_2 *regression* problem. In other words, it suffices to find a $\tilde{x}$ that yields an ℓ_2 error that is within a small constant factor of the optimal such error. Several communication models are possible. We focus on the coordinator model, where are communication channels are between a central coordinator and the s remaining servers.

In this work, we ask “How much communication is required to solve the approximate regression problem in the coordinator model?” This was previously addressed in [VWW20] where an upper bound of $O(sd^2)$ and a lower bound of $\Omega(sd + d^2)$ was given. Surprisingly this gap has remained open. We show that in fact $O(sd + d^2)$ communication is achievable. A similar gap remained for ℓ_1 regression, which we resolve as well.

We give two algorithms for the ℓ_2 regression problem. One algorithm is based on a novel sketch-based algorithm for approximating the so-called *block leverage scores*. Our other algorithm is based on the recursive leverage sampling procedure of [Coh+15].

5.1 Contributions

The block leverage score sampling algorithm was my contribution. The algorithm based on recursive leverage sampling arose from a conversation with David Woodruff. I would particularly like to thank Deanna Needell for helpful discussions during this work as well.

5.2 Introduction

We consider a situation where s servers each have a subset of rows for a d dimensional linear system $Ax = b$. In typical situations this system is overdetermined and inconsistent, so we are interested in solving for the least squares solution x^* which minimizes $\|Ax - b\|_2^2$. More specifically, we are interested in finding an ϵ-approximation $\tilde{x}$ to the least squares solution, which satisfies

$$(1-\epsilon)\|Ax_* - b\|_2^2 \leq \|A\tilde{x} - b\|_2^2 \leq (1+\epsilon)\|Ax_* - b\|_2^2 .$$

This type of multiplicative guarantee is standard for sketch-based algorithms [Woo+14]. This type of guarantee is convenient for both high and low precision settings. When the system is consistent, we must recover x^* exactly. Otherwise we settle for a solution which nearly minimizes the mean squared loss, up to a small multiplicative error. In many practical situations where linear regression is applied, this is quite reasonable. For instance if the underlying data is fairly noisy, then it is unnecessary to get the optimal solution to extremely high accuracy (as might be the case for an additive error guarantee). Our goal is to find an $\tilde{x}$ satisfying the goal above by communicating as few bits as possible.

There are several common communication models in the literature. In the blackboard model

Problem	Total Communication
ℓ_2 Regression ([VWW20])	$O(sd^2)$ (Theorem 6.1)
ℓ_2 Regression (ours)	$\tilde{O}(sd + d^2)$ (Theorem 5.4 and Theorem 5.9)
ℓ_1 Regression ([VWW20])	$O(sd^2)$ (Theorem 7.1)
ℓ_1 Regression (ours)	$\tilde{O}(sd + d^2)$ (Theorem 5.8)

Table 5.1: Communication complexity results for linear regression, for constant ϵ

each server may broadcast a message to all of the other s servers. In the point-to-point model, servers may send messages to only one other server at a time. This model is more restrictive, as simulating a broadcast requires a server to send s messages. We focus on the coordinator model of communication, where the s servers can communicate only with a single central server. This model is essentially equivalent to the point-to-point model as any two servers can communicate by relaying a message through the coordinator. Communication bounds between these two models can therefore only differ by a factor of two.

This problem was initially considered in [VWW20] , where the authors gave tight bounds for deterministic algorithms in both models, as well as for randomized algorithms in the blackboard model. However for randomized algorithms in the coordinator model there remained a gap of $O(sd^2)$ versus $\Omega(sd + d^2)$ which is substantial when the number of servers is large. We close this gap and show that in fact $\widetilde{O}(sd + d^2)$ communication is achievable for constant ϵ.

5.2.1 Our results

Given a matrix A with rows partitioned among the servers, we show that the coordinator can learn an ϵ distortion subspace embedding for the column span of A using $\widetilde{O}(sd + d^2)$ communication.

We give two protocols that work for ℓ_2-regression. We also build on one of these protocols to give an algorithm for the analagous ℓ_1-regression problem where one wishes to find a $\tilde{x}$ satsifying

$$(1 - \epsilon) \left\|Ax_* - b\right\|_1 \leq \left\|A\tilde{x} - b\right\|_1 \leq (1 + \epsilon) \left\|Ax_* - b\right\|_1,$$

where x_* minimizes $\left\|Ax - b\right\|_1$.

Rather than directly solve the ℓ_2 and ℓ_1 regression problems, our algorithms work by constructing ℓ_1 and ℓ_2 subspace embeddings. We recall that for $p \in \{1, 2\}$, S is an ϵ-distortion ℓ_p subspace embedding for a matrix A if $\|SAx\|_p = (1 \pm \epsilon) \|Ax\|_p$ for all x. It is well-known (see [Woo+14] for example) that an algorithm which computes an ℓ_p subspace embedding is sufficient to solve the ℓ_p regression problem.

5.3 Our Techniques

The first of our algorithms uses the recursive leverage sampling algorithm given in [Coh+15]. Implementing this algorithm roughly requires that one can sample rows of A from a *relative leverage score* distribution with respect to a matrix M held by the coordinator. So the probability of sampling row A_i should be proportional to $A_i M^{-1}(A_i)^T$. We show how this can be achieved with an ℓ_2-sampling sketch. Unfortunately this is not quite sufficient as one actually needs to truncate the relative leverage scores that are much larger than one. For this we use a heavy-hitters sketch to first identify and remove outlying rows. We remark that it would be natural to consider having the coordinator send a Johnson-Lindenstrauss((JL) sketch of the form $M^{-1/2}(A_i)^T$ to the servers. Unfortunately when A is poorly conditioned, the coordinator may need to send the sketch with very high bit precision, resulting in high communication complexity.

Our second protocol is based on the notion of *block leverage scores*, which may be of independent interest. The block leverage score is simply the sum of the leverage scores of the rows in a block.

Our key technical result is that a simple sketch suffices to estimate the block leverage scores of $A = [A^{(1)}; \ldots ; A^{(s)}]$. Our approach is to sketch each block down to roughly an $O(k) \times d$ matrix using a Rademacher (or other) sketch $S^{(i)}$ for each block. The block leverage scores are then estimated to be the block leverage scores of $[S^{(1)}A^{(1)}; \ldots ; S^{(s)}A^{(s)}]$. Unfortunately this does not necessarily yield good estimates for all block leverage scores. Indeed the block leverage scores of the sketched matrix are all bounded by k, so we may underestimate the scores of outlying blocks. However, by applying a novel characterization of the block leverage scores as a block sensitivity, we show that we obtain good (over-)estimates for all block leverage scores smaller than Ck. Additionally we can detect the blocks for which we do not obtain a good estimate; their leverage score estimates are guaranteed to be larger than Ck.

This yields a simple iterative procedure for computing block leverage score estimates. In the first round, the coordinator requests a roughly $1 \times d$ sketch from each block, yielding good estimates for all blocks with leverage score at most 1. The coordinator now only needs to focus on the blocks with leverage score at least 1, of which there are at most $O(d)$ (since the block leverage scores sum to at most d). The coordinator then requests roughly a $2 \times d$ sketch from each server, yielding good estimates for blocks with leverage score at most 2, of which there are at most $d/2$. This procedure is repeated, doubling the number of rows requested in each round. In round r the server requests a sketch of size $2^r d$ from each of approximately $d/2^r$ servers, so the procedure requires $O(d)$ communication per round. After $O(\log d)$ rounds, we find good estimates for all the blocks, and hence use $\widetilde{O}(sd + d^2)$ communication.

Given estimates for the block leverage scores, a version of the standard leverage score sampling algorithm suffices to construct a subspace embedding for the column span of A. We simply sample blocks proportional to the estimated block leverage scores and receive a $1 \times d$ sketch from that block. By taking $\frac{1}{\epsilon^2} d \log d$ such samples we obtain our desired subspace embedding with distortion ϵ. We note that the entire algorithm (estimating the block leverage scores and sampling) can be implemented simultaneously by having each server send twice as many rows during the leverage score estimation algorithm. The extra rows can then be used later during the sampling phase. Hence the algorithm is nearly one-way in the sense that coordinator only needs to communicate $O(s+d)$ bits in total, and only for the purposes of notifying the servers that are active in that round.

5.4 Recursive Leverage Score Sampling

In this section we give procedures which allow the coordinator to construct ℓ_1 and ℓ_2 subspace embeddings for a matrix $A = [A^{(1)}; \ldots ; A^{(s)}]$ distributed among s servers, where each block has at most n rows. In particular this allows the coordinator to solve ℓ_1 and ℓ_2 regression problems with ϵ error, i.e. the coordinator recovers an $\hat{x}$ with $\left\|A\hat{x} - b\right\|_p \leq (1+\epsilon)\left\|Ax^* - b\right\|_p$ where x^* is the optimal solution to the regression problem and $p \in \{1, 2\}$.

We apply the recursive leverage score sampling algorithm of [Coh+15], which iteratively computes improved spectral approximations of A. In order to implement this algorithm we need a procedure to carry out leverage score sampling with respect to an intermediate spectral approximation. We will give a subroutine to solve the following slightly more general problem. In our application M will be taken to be the inverse of these spectral approximations to A^TA.

Problem 5.1. *Let M be a PSD matrix owned by the coordinator. Let*

$$\begin{aligned} u_i^{(j)} &= A_i^{(j)} M M^T \left(A_i^{(j)}\right)^T \\ v_i^{(j)} &= \min(u_i^{(j)}, 1) \\ T &= \sum_{i,j} v_i^{(j)}. \end{aligned}$$

Sample r rows of A from approximately the probability distribution gotten by normalizing the $v_i^{(j)}$'s. The probability of sampling row i_0 from block j_0 should be

$$(1 \pm c)\frac{v_{i_0}^{j_0}}{\sum v_i^{(j)}} + \frac{1}{poly(n)}.$$

In our protocol, we use the following sketch which comes directly from Theorem 2.7 of [Mah+20].

Lemma 5.2. *Given an $\mathbb{R}^{n\times d}$ matrix A and $P \in \mathbb{R}^{d\times d}$, there is linear sketch S, and a recovery* *algorithm which when given SA outputs row index i for A with probability* $(1\pm\frac{1}{2})\frac{\left\|A^{(i)}P\right\|^2}{\|AP\|_F^2}+\frac{1}{poly(n)}$.

The sketch SA uses $O(d\log^3 n\log\frac{1}{\delta})$ space, and the guarantee above holds with probability $1-\delta$. The poly(n) term can have any desired exponent by adjusting constants.

Lemma 5.3. *There is a protocol which solves Problem 5.1 with failure probability at most δ and $\widetilde{O}(\frac{1}{\delta}Td+sd+rd)$ communication.*

Note that we will later apply this result with $T, r = O(d)$.

Proof. Our first step is to produce an estimate of

$$B^{(j)} := \sum_i v_i^{(j)}$$

for all j. To do this, we first estimate

$$\widetilde{B}^{(j)} := \sum_i u_i^{(j)} = \operatorname{Tr}\left[A_i^{(j)}MM^T\left(A_i^{(j)}\right)^T\right] = \left\|A^{(j)}M\right\|_F^2.$$

This can accomplished by having each server send a JL sketch $S^{(j)}A^{(j)}$ to the coordinator. By choosing these sketches to have $O(\frac{\log s}{\delta})$ rows, the coordinator obtains constant factor approximations to all $\widetilde{B}^{(j)}$ with failure probability at most $O(\delta)$.

In order to handle truncation, we would next like to find all $u_i^{(j)}$ with a value greater than 1. Call the corresponding rows "outlying". To identify the outlying rows we use the ℓ_2 sampling sketch given in Lemma 5.2. Roughly, an ℓ_2 sampling sketch will find an outlying row with good probability, so the coupon collector problem will allow us to find all of them.

More precisely, to implement ℓ_2 sampling across the blocks, the coordinator first uses the values of $\widetilde{B}^{(j)}$ to choose a server from which to sample. That server sends an ℓ_2 sampling sketch, allowing the coordinator to perform ℓ_2 sampling from the rows of $A^{(j)}M$, up to a constant factor on the sampling probabilities, as well as an additive $1/\text{poly}(n)$ error where poly(n) can have as large an exponent as desired.

We would like to identify all outlying rows and (temporarily) remove them as we go. We sample rows via the ℓ_2 sampling method described above. Each such row is sent to the coordinator who then checks by direct computation whether it is outlying. If it is, then the server temporarily removes that row and sends a new JL sketch to the coordinator, so that the coordinator can update its Frobenius norm estimate for that server.

Suppose that there are k outlying rows remaining. Then the total mass of the corresponding $u_i^{(j)}$'s is at least k, and the total mass of the non-outlying $u_i^{(j)}$'s is at most T (from the statement of Problem 5.1). Hence the probability of sampling an outlying row is at least $\frac{k}{T+k} \geq \frac{k}{2T}$ since $k \leq T$. The the expected number of samples needed to encounter an outlying row is therefore at most $\frac{2T}{k}$. There are at most T outlying rows to start, so the expected number of samples needed to find all outlying rows is at most

$$\frac{2T}{T} + \frac{2T}{T-1} + \ldots + \frac{2T}{1} = 2T\left(\frac{1}{1} + \frac{1}{2} + \ldots + \frac{1}{T}\right) \leq O(T \log T).$$

Hence after $O(\frac{1}{\delta} T \log T)$ rounds of sampling we identify all outlying rows with failure probability at most δ.

Next, the coordinator then counts total number of outlying rows for each server, and adds this to a constant factor approximation of the Frobenius norm of the remaining rows (which may again be obtained via a JL sketch). This gives the coordinator constant factor approximations $\widehat{B^{(j)}}$ to $B^{(j)}$.

To sample from the $v_i^{(j)}$ distribution, the coordinator chooses a server with probability proportional to $\widehat{B^{(j)}}$. Then the coordinator decides whether to sample an outlying row, with probability proportional to the number of outlying rows in the block. If not, then the coordinator requests an ℓ_2 sampling sketch from server for all the non-outlying rows. The coordinator uses this sketch to sample a row index and then requests the corresponding row from the server. This is repeated r times and requires a total of $O(rd \log^3 n \log(d/\delta))$ space.

□

Theorem 5.4. *There is a protocol using $\widetilde{O}(sd + \frac{1}{\epsilon^2}d^2)$ communication that allows the coordinator to produce an ϵ distortion ℓ_2 subspace embedding for the column span of A, with failure probability at most δ. As a consequence, the coordinator can solve linear regression with the same complexity.*

Proof. We simply apply the Repeated Halving algorithm given in Section 7.1 of [Coh+15].

In a given recursive call to their algorithm, let $u_i^{(j)}$ and $v_i^{(j)}$ be defined as in Problem 5.1 for $M = ((\tilde{A}')^T\tilde{A})^{-1/2}$.

The algorithm in [Coh+15] samples from the distribution given by $(1 + \frac{1}{u_i^{(j)}})^{-1}$ up to scaling. Note that our $v_i^{(j)}$ distribution is equivalent up to constants, so it suffices to run the sampling procedure given in our Lemma 5.3.

This yields a constant factor approximation for A on the coordinator side. Then by applying our Lemma 5.3 to sample an additional $O(d/\epsilon^2)$ rows, we obtain an ϵ distortion ℓ_1 embedding for the column space of A. □

5.4.1 Extension to ℓ_1 regression

Our algorithm is very similar to the algorithm we gave for ℓ_2 regression, except that we use the recursive Lewis weight sampling algorithm given in Lemma 6.2 of [CP15]. In order to implement their algorithm we need to implement a sampling algorithm which solves the following problem, analogous to the problem for the ℓ_2 case above.

Problem 5.5. *Let M be a PSD matrix owned by the coordinator. Let*

$$\begin{aligned} u_i^{(j)} &= A_i^{(j)} M M^T \left(A_i^{(j)}\right)^T \\ v_i^{(j)} &= \min(u_i^{(j)}, 1) \\ T &= \sum_{i,j} \sqrt{v_i^{(j)}}. \end{aligned}$$

Sample r rows of A from approximately the probability distribution gotten by normalizing the

$\sqrt{v_i^{(j)}}$'s. The probability of sampling row i_0 from block j_0 should be

$$(1 \pm c)\frac{\sqrt{v_{i_0}^{j_0}}}{\sum \sqrt{v_i^{(j)}}} + \frac{1}{\mathit{poly}(n)}.$$

To do this we need an $L_{1,2}$ sampling sketch, which is given in Lemma A.4 of [Mah+20].

Lemma 5.6. *Given an $\mathbb{R}^{n \times d}$ matrix X and $P \in \mathbb{R}^{d \times d}$, there is linear sketch S, and a recovery algorithm which when given SX outputs row index i for X with probability $(1 \pm \frac{1}{2})\frac{\|X_i P\|^2}{\|XP\|_F^2} + \frac{1}{\mathit{poly}(n)}$.*

The sketch uses $O(d\,\mathit{polylog}(n))$ space, and succeeds with high probability.

Lemma 5.7. *There is a protocol which solves Problem 5.5 with failure probability at most δ and communication $\widetilde{O}(\frac{1}{\delta}Td + sd + rd)$.*

Proof. The algorithm is nearly identical to the ℓ_2 case from Lemma 5.3, so we describe how to modify the procedure.

Rather than using JL sketches to track the Frobenius norm of each block, we instead use sketches for the $L_{1,2}$ norm of each block. Such a sketch is given in [And+09], which uses $O(d\,\text{polylog}(n))$ space and succeeds with high probability.

Secondly, rather than an ℓ_2 sampling sketch, we use the $L_{1,2}$ sketch from Lemma 5.6.

Otherwise, the same argument from Lemma 5.3 applies. □

Theorem 5.8. *There is a protocol using $\widetilde{O}(sd + \frac{1}{\epsilon^2}d^2)$ communication that allows the coordinator to produce an ϵ distortion ℓ_1 subspace embedding for the column span of A, with failure probability at most δ. As a consequence, the coordinator can solve linear regression with the same complexity.*

Proof. We use the recursive Lewis weight sampling algorithm given in Lemma 6.2 of [CP15]. Notably, by their remark after its proof, we can remove the $d^{p/2}$ sampling dependence for $p = 1$. For their algorithm, it then suffices that we use sampling probabilities $\sqrt{v_i^{(j)}}$ and take $O(d \log d)$ samples in each recursive call. (Note that similar to the ℓ_2 case, we still have $\sum \sqrt{v_i^{(j)}} \leq O(d)$.) Hence it suffices to apply the subroutine from our Lemma 5.7.

This yields a constant factor approximation to the Lewis quadratic form on the coordinator side. Then by applying our Lemma 5.7 to sample an additional $O(d/\epsilon^2)$ rows, we obtain an ϵ distortion ℓ_1 embedding for the column space of A. □

5.5 An algorithm based on block leverage scores

In this section we give an algorithm for constructing an ℓ_2 subspace embedding. In contrast to our other algorithm, this algorithm is much simpler to implement and is almost one-way. The coordinator only needs to send $O(\log d)$ bits to the servers over the course of d rounds. We summarize the result here which will be proven in the following sections.

Theorem 5.9. *There is a protocol which for constant ϵ constructs an ℓ_2 subspace embedding for A which runs in $\log d$ rounds, and uses $\widetilde{O}(sd+d^2)$ communication. Moreover the servers collectively only receive a total of $\widetilde{O}(s)$ bits from the coordinator.*

Block Leverage Scores

We give the following definition of the block leverage scores. We note that this definition has appeared before. For example [Kyn+16] gives that definition that we present here.

Definition 5.10. *Let $A = [A^{(1)}; \ldots; A^{(s)}]$. We define the block leverage score of block $A^{(i)}$ to be*

$$\mathcal{L}_i(A) = \operatorname{Tr}\left(A^{(i)}(A^TA)^{-1}(A^{(i)})^T\right). \tag{5.1}$$

For use throughout, we list a few basic properties of the block leverage scores.

Proposition 5.11. *Let $A = [A_1; \ldots; A_s]$. The following properties hold:*

1. *If $A^{(i)} \in \mathbb{R}^{k\times d}$, and the rows of $A^{(i)}$ have leverage scores $\ell_{i1}, \ldots, \ell_{ik}$, (as rows of A) then $\mathcal{L}_i(A) = \sum_{j=1}^k \ell_{ij}$.*

2. *$\sum_{i=1}^s \mathcal{L}(A^{(i)}) = \operatorname{rk}(A)$.*

3. *Suppose* $\tilde{A} = [A^{(1)}; \ldots; A^{(s)}; A^{(s+1)}]$. *For all* $i \in [s]$, $\mathcal{L}_i(A) \geq \mathcal{L}_i(\tilde{A})$.

Proof. For property 1,

$$\mathcal{L}_i(A) = \mathrm{Tr}(A^{(i)}(A^T A)^{-1}(A^{(i)})^T) = \sum_j A_i^{(j)}(A^T A)^{-1}\left(A_i^{(j)}\right)^T = \sum_j \ell_{ij}.$$

In light of property 1, properties 2 and 3 follow from the corresponding facts for classical leverage scores.

□

We also give a characterization of the block leverage score as a block sensitivity.

Proposition 5.12. *Given a full column rank matrix* A *consisting of blocks* $A^{(1)}, \ldots, A^{(s)}$*, we have*

$$\mathcal{L}_i(A) = \sup_X \frac{\|A^{(i)}X\|_F^2}{\|AX\|_2^2},$$

where $\sup_X$ *is over all matrices with compatible dimensions to* A.

Proof. Let UDV^T be the singular value decomposition for A, and let U_j have a subset of the rows of U so that $U_j DV^T = A^{(j)}$. We are interested in maximizing $\|(U_j DV^T)X\|_F^2 = ||U_j(DV^T X)||_F^2$ subject to $\|UDV^T X\|_2 = 1$. Since U is orthonormal, the constraint becomes $\|DV^T X\|_2 = 1$. DV^T has full rank so the optimization problem is equivalent to maximizing $||U_j Y||_F^2$ s.t. $\|Y\|_2 \leq 1$. This is optimized for $Y = I$ and the objective is the sum of squares of row norms for U_j which is the sum of leverage scores of the rows of $A^{(j)}$. □

Sketching Block Leverage Scores

We use our sensitivity characterization of the block leverage scores to show that sketching a block does not cause its leverage score to drop too much.

Lemma 5.13. *Let* $G^{(1)}$ *be a sketching matrix which is an* $O(1)$ *distortion OSE for* k *dimensional* *subspaces with probability* $1-\delta$. *With probability* $1-\delta$ *we have that*

$$\mathcal{L}_1\left([G^{(1)}A^{(1)}, A^{(2)}, \ldots, A^{s}]\right) \geq C\min(k, \mathcal{L}_1(A)).$$

Proof. By Proposition 5.12 can choose an X with $\|A^{(1)}X\|_F^2/||AX||_2^2 = \mathcal{L}_1(A)$. Theorem 1 from [Coh+15] implies that

$$\|G^{(1)}A^{(1)}X\|_2^2 \lesssim ||A^{(1)}X||_2^2 + (1/k)||A^{(1)}X||_F^2,$$

and so

$$\frac{\|G^{(1)}A^{(1)}X\|_2^2}{||AX||_2^2} \lesssim 1 + \frac{1}{k}\frac{\|A^{(1)}X\|_F^2}{\|AX\|_2^2} = 1 + \frac{\mathcal{L}_1(A)}{k}.$$

Then

$$\|G^{(1)}AX\|_2 \lesssim ||G^{(1)}A^{(1)}X||_2 + ||AX||_2 \lesssim \left(1 + \frac{1}{k}\mathcal{L}_1(A)\right)||AX||_2.$$

Hence

$$\frac{\|G^{(1)}A^{(1)}X\|_F^2}{\|G^{(1)}AX\|_2^2} \gtrsim \frac{\|A^{(1)}X\|_F^2}{\|G^{(1)}AX\|_2^2} \gtrsim \min(k, \mathcal{L}_1(A)),$$

by Johnson-Lindenstrauss, and the previous bound. Hence X witnesses a sensitivity of at least $C\min(k, \mathcal{L}_1(A))$ for the first block of $G^{(1)}A$ as desired. □

Next we analyze the situation where all but one block is sketched. To streamline the argument we first lead with a couple simple claims.

Proposition 5.14. *Let* $X \in \mathbb{R}^{d\times d}$ *be PSD, let* $U \in \mathbb{R}^{d\times m}$*, and suppose that* $\mathrm{Tr}(U^TXU) \leq \mathrm{Tr}(U^TU)$. *Then* $\mathrm{Tr}(U^TX^{-1}U) \geq \mathrm{Tr}(U^TU)$.

Proof. Since X and X^{-1} are simultaneously diagonalizable, the Loewner order inequality $X + X^{-1} \geq 2I$ follows from the scalar inequality $x + 1/x \geq 2$ for $x \geq 0$. Thus $U^T(X + X^{-1})U \geq$

$2U^TU$ and so $\operatorname{Tr}(U^T(X+X^{-1})U) \geq 2\operatorname{Tr}(U^TU)$. Therefore

$$\operatorname{Tr}(U^TX^{-1}U) \geq 2\operatorname{Tr}(U^TU) - \operatorname{Tr}(U^TXU) \geq 2\operatorname{Tr}(U^TU) - \operatorname{Tr}(U^TU) = \operatorname{Tr}(U^TU). \qquad \square$$

Proposition 5.15. *Let X be a random $d \times d$ matrix which is a.s. PSD, and let A be fixed matrix which is PSD and non-singular.*

Suppose that for every U in $\mathbb{R}^{d\times m}$ it holds that

$$\mathbb{P}\left(\operatorname{Tr}(U^TXU) \leq \operatorname{Tr}(U^TAU)\right) \geq 1-\delta.$$

Then for every V in $\mathbb{R}^{d\times m}$ it also holds that

$$\mathbb{P}\left(\operatorname{Tr}(V^TX^{-1}V) \geq \operatorname{Tr}(V^TA^{-1}V)\right) \geq 1-\delta.$$

Proof. Plugging $A^{-1}V$ into the hypothesis gives that for all V,

$$\mathbb{P}(\operatorname{Tr}(V^TA^{-1}XA^{-1}V) \leq \operatorname{Tr}(V^TA^{-1}V)) \geq 1-\delta,$$

or equivalently

$$\mathbb{P}\left(\operatorname{Tr}((A^{-1/2}V)^TA^{-1/2}XA^{-1/2}(A^{-1/2}V)) \leq \operatorname{Tr}((A^{-1/2}V)^T(A^{-1/2}V))\right) \geq 1-\delta.$$

By Proposition 5.14, this gives that for all V,

$$\mathbb{P}\left(\operatorname{Tr}((A^{-1/2}V)^TA^{1/2}X^{-1}A^{1/2}(A^{-1/2}V)) \geq \operatorname{Tr}((A^{-1/2}V)^T(A^{-1/2}V))\right) \geq 1-\delta,$$

which simplifies to the desired conclusion. $\square$

Lemma 5.16. *Let $A = [A^{(1)}; \ldots; A^{(s)}]$ be non-singular and let $S^{(1)}, \ldots S^{(s)}$ be random sketching* *matrices of appropriate dimension so that the products $S^{(i)}A^{(i)}$ are defined. Assume that each $S^{(i)}$ satisfies the $(1, \delta, 2)$-JL-moment property. Let $V \in \mathbb{R}^{d\times m}$. Then with probability at least $1 - \delta$,*

$$\operatorname{Tr}\left(V^T\left(\sum_{i=1}^{s} {A^{(i)}}^T {S^{(i)}}^T S^{(i)} A^{(i)}\right)^{-1} V\right) \geq \frac{1}{2}\operatorname{Tr}\left(V^T(A^TA)^{-1}V\right).$$

Proof. We apply Proposition 5.15. Let $U \in \mathbb{R}^{d\times m}$ be an arbitrary fixed matrix. Then we have

$$\operatorname{Tr}\left(U^T\left(\sum_{i=1}^{s} {A^{(i)}}^T {S^{(i)}}^T S^{(i)} A^{(i)}\right) U\right) = \sum_{i=1}^{s} \left\|S^{(i)}A^{(i)}U\right\|_F^2.$$

The block matrix $S^{(1)} \oplus \cdots \oplus S^{(s)}$ also has the $(1, \delta, 2)$-JL-moment property (see for example Lemma 13 of [Ahl+20]). So with probability at least $1 - \delta$,

$$\sum_{i=1}^{s} \left\|S^{(i)}A^{(i)}U\right\|_F^2 \leq 2\left\|AU\right\|_F^2 = 2\operatorname{Tr}(U^TA^TAU).$$

The claim now follows by Proposition 5.15. □

Lemma 5.17. *Let $A = [A^{(1)}; \ldots, A^{(s)}]$, let $S^{(1)}, \ldots, S^{(s)}$ be sketching matrices satisfying the $(1, \delta, 2)$-JL-moment property, and let $\tilde{A} = [S^{(1)}A^{(1)}; \ldots; S^{(s)}A^{(s)}]$, where $S_k = I$ for a fixed k. Then with probability at least $1 - \delta$, $\mathcal{L}_k(\tilde{A}) \geq \frac{1}{2}\mathcal{L}_k(A)$.*

Proof. By Lemma 5.16 we have

$$\mathcal{L}_k(\tilde{A}) = \operatorname{Tr}\left(A^{(k)}\left(\sum_{i=1}^{s} {A^{(i)}}^T {S^{(i)}}^T {S^{(i)}}^T A^{(i)}\right)^{-1} A^{(k)}\right) \geq \frac{1}{2}\operatorname{Tr}\left(A^{(k)}\left(\sum_{i=1}^{s} {A^{(i)}}^T A^{(i)}\right)^{-1} A^{(k)}\right) = \frac{1}{2}\mathcal{L}_k(A).$$

□

Combining the two block sketching results gives the following.

Lemma 5.18. *Let $S^{(1)}, \ldots, S^{(s)}$, each with d columns, all be (normalized) Rademacher with*

$O(k \log(s/\delta))$ rows. Then for each i,

$$\mathcal{L}_i([S^{(1)}A^{(1)}; \ldots ; S^{(s)}A^{(s)}]) \geq C \min\left(k, \mathcal{L}_i([A^{(1)}; \ldots ; A^{(s)}])\right),$$

with probability at least $1 - \delta$.

Remark 5.19. *The sketches in the above result were only taken to be Rademacher for convenience. The same argument applies to sparse sketches for example.*

5.5.1 Block Leverage Score estimation

The sketch from the previous section shows that we can accurately (over-)estimate a given block leverage score by sketching down to dimension roughly k. Unfortunately the block leverage scores can be as large as d and we are unable to take a sketch of d rows from all servers. Fortunately, not many servers can have large block leverage score, so by iteratively pruning off the ones that don't, we can focus on the servers with the most information.

Theorem 5.20. *Algorithm 4 runs with $O(\log d)$ rounds of communication, and returns a list L satisfying*

(i) $L[i] \geq C\mathcal{L}_i([A^{(1)}; \ldots ; A^{(s)}])$ for all i

(ii) $\sum_{i=1}^{s} L[i] \leq O(d \log d)$

Moreover the servers collectively send at most $O(cs + cd \log d)$ vectors of length d to the coordinator.

Proof. We start by bounding the number of servers which are active in a given round. In round 0, $|\mathcal{S}_0| = s$. For $r \geq 1$, note that for every server i in $\mathcal{S}_r$, $\widehat{\mathcal{L}}_{r-1,i} \geq Ck_{r-1}$. On the other hand there cannot be many such servers, since by Proposition 5.11,

$$\sum_{i \in \mathcal{S}_r} \widehat{\mathcal{L}}_{r-1,i} \leq d,$$

Algorithm 4

```
procedure BLOCK_LEVERAGE_APPX($A$,$k$)
    Let $L = [\perp, \dots, \perp]$ of length $s$
    Let $\mathcal{S}_0 = \{1, \dots, s\}$
    for round $r = 0, 1, \dots, \lceil \log d \rceil$ do
        Let $k_r = 2^r$
        for $i$ in $\mathcal{S}_r$ do
            Server $i$ draws $S_{r,i}$ a constant distortion $k_r$-dimensional oblivious subspace embedding, and sends $S_{r,i}A^{(i)}$ to the coordinator
        end for
        Coordinate forms block matrix $A^{(r)}$ with blocks given by $S_{r,i}A^{(i)}$ for $i$ in $\mathcal{S}_r$, and where the blocks are indexed by $\mathcal{S}_r$
        For all $i \in \mathcal{S}_r$, coordinator computes $\widehat{\mathcal{L}}_{r,i} = \mathcal{L}_i(A^{(r)})$
        $\mathcal{S}_{r+1} = \{i \in \mathcal{S}_r : \widehat{\mathcal{L}}_{r,i} \geq Ck_r\}$
        for $i$ in $\mathcal{S}_r \setminus \mathcal{S}_{r+1}$ do
            $L[i] = \widehat{\mathcal{L}}_{r,i}$
        end for
    end for
    For all $i$, if $L[i] =\perp$, then set $L[i] = d$
    return $L$
end procedure
```

which implies that $|\mathcal{S}_r| \leq \frac{d}{Ck_{r-1}}$.

This immediately gives a bound on the communication cost. Summing the number of vectors transmitted in each round gives a total of

$$\sum_{i=0}^{\lceil \log d \rceil} ck_r|\mathcal{S}_r| \leq c\left(sk_0 + \sum_{i=1}^{\lceil \log d \rceil} k_r \frac{d}{Ck_{r-1}}\right) = c\left(s + \frac{2d}{C}\lceil \log d \rceil\right)$$

vectors sent to the coordinator.

Next we show that (i) holds. By the algorithm, note that either $L[i] = d$ or on round r we set $L[i] = \widehat{\mathcal{L}}_{r,i}$. In the first case (i) is trivial since all block leverage scores are at most d. In the latter case, $\widehat{\mathcal{L}}_{r,i} \leq Ck_r$. But $\widehat{\mathcal{L}}_{r,i} \geq C\min\left(k_r, \mathcal{L}_i(A_{\langle r \rangle})\right)$ which is at least $\mathcal{L}_i(A)$ by monotonicity. So we have $k_r \geq \mathcal{L}_i(A)$, which implies that

$$\widehat{\mathcal{L}}_{r,i} \geq C\min(k_r, \mathcal{L}_i(A)) = C\mathcal{L}_i(A).$$

Finally we show (ii). Since we have

$$\sum_{i \in \mathcal{S}_r} \widehat{\mathcal{L}}_{r,i} = \sum_{i \in \mathcal{S}_r} \mathcal{L}_i(A_{\langle r \rangle}) \leq d,$$

it follows that the entries of L which are set in round r sum to at most d. Hence the sum of the entries of L set in the outer for-loop is at most $(\lceil \log d \rceil + 1)d$. By the argument given above for the communication cost, there are at most $\frac{d}{Ck_r} \leq \frac{1}{C}$ entries of L which are not set after the loop. These entries are set to d, which gives

$$\sum_{i=1}^{s} L[i] \leq (\lceil \log d \rceil + 1)d + \frac{d}{C} \leq O(d \log d). \qquad \square$$

5.5.2 Block Leverage Sampling

Given the overestimates computed for the block leverage scores in the previous section, a straightforward concentration bound allows to get a spectral approximation via block leverage sampling. By combining with the algorithm for estimating the block leverage scores, this immediately yields an algorithm with $O(sd + \frac{1}{\epsilon^2} d^2)$ communication in the coordinator model, for computing an ϵ distortion subspace embedding for the columns of A.

Algorithm 5

```
procedure BLOCK_LEVERAGE_SAMPLING($p$,$N$)
    Coordinator set $\widehat{A} = 0 \in \mathbb{R}^{N \times d}$
    for $i = 1, \ldots, N$ do
        Sample server $j$ from the distribution $p$
        Server $j$ generates a Rademacher random vector $g \in \mathbb{R}^{m_j}$ and sends $g^T A^{(j)}$ to coordinator
        Coordinator sets row $\widehat{A}_i = \frac{1}{\sqrt{p_j N}} g^T A^{(j)}$
    end for
    return $\widehat{A}$
end procedure
```

As is standard for analyses of leverage score sampling, we rely the Matrix Chernoff bound (see

[Woo+14] for example). We state a version here which follows from [Tro12]. The version we use is slightly less general, but more convenient for our purposes.

Theorem 5.21. *Let $X_1, \dots, X_d \in \mathbb{R}^{d\times d}$ be random matrices which are independent and symmetric PSD, with $\mu_{\min} I \leq \mathbb{E}X_i \leq \mu_{\max} I$, and $\|X_i\| \leq R$ a.s. Let $\overline{X} = \frac{1}{N}\sum_{i=1}^N X_i$. Then for all $\delta \in [0,1)$,*

$$\mathbb{P}\left(\lambda_{\max}(\overline{X}) \geq (1+\delta)\mu_{\max}\right) \leq d\exp(-\delta^2 \frac{N\mu_{\max}}{3R}).$$

$$\mathbb{P}\left(\lambda_{\max}(\overline{X}) \leq (1-\delta)\mu_{\min}\right) \leq d\exp(-\delta^2 \frac{N\mu_{\min}}{2R}).$$

The quantities that we apply the matrix Chernoff bound to will have operator norm given by a Hutchinson trace estimator [Hut89]. This standard application of Matrix Chernoff is the core argument. The additional work simply fixes a technical issue.

Hutchinson's estimator may very occasionally be much larger than expected, which would require R in Theorem 5.21 to be undesirably large. Fortunately Hutchinson's estimator has exponential tail decay, and so these potential large values may be safely ignored with high probability. To make this precise, we will use the following technical fact, which is effectively a restatement of results in [DM21].

Proposition 5.22. *Let $A \in \mathbb{R}^{d\times d}$ be symmetric PSD, and let $g \in \mathbb{R}^d$ be a Rademacher random vector. Let $\mu = \mathbb{E}(g^T A g) = \mathrm{Tr}(A)$. Then*

$$\mathbb{P}(g^T A g \geq t) \leq c_1 e^{-c_2 t/\mu},$$

for all $t \geq c_3\mu$. The c_i's are positive absolute constants.

Proof. We set $\ell = 1$ in Claim A.3 of [DM21]. By bounding $\|A\|_2$ and $\|A\|_F$ each by $\mathrm{Tr}(A) = \mu$, we may take $\nu = c_4\mu$ and $\beta = c_5\mu$ in A.3. By properties of subexponential random variables given in [Wai15], it then follows that $\mathbb{P}(g^T A g \geq \mu + t) \leq 2e^{-c_6 t/\mu}$ for $t \geq \nu^2/\beta = c_7\mu$, which by adjusting constants rearranges to claim above. □

Theorem 5.23. *Suppose that the input to Algorithm 5 satisfies $p_i \geq \beta \frac{\mathcal{L}_i(A)}{d}$ for some $\beta \in (0,1]$, and with $N \geq \Omega\left(\frac{d}{\beta\epsilon^2}\log\left(\frac{d}{\beta\epsilon}\right)\log d\right)$, where $\epsilon < 1$. Then the output $\widehat{A}$ of Algorithm 5 satisfies*

$$(1-\epsilon)A^TA \leq \widehat{A}^T\widehat{A} \leq (1+\epsilon)A^TA.$$

Proof. Let X_k be distributed as $\frac{1}{p_j}A_j^Tgg^TA_j$ where the index j is drawn from p, and g is independently drawn as a Rademacher random vector. Then $\widehat{A}^T\widehat{A}$ is distributed as $\frac{1}{N}\sum_{k=1}^N X_k$, so we show concentration for this average.

As is standard in such arguments, we show that the following equivalent statement holds with the desired probability:

$$(1-\epsilon)I \leq (A^TA)^{-1/2}X(A^TA)^{-1/2} \leq (1+\epsilon)I.$$

First note that

$$\mathbb{E}(X_k) = \sum_{i=1}^s p_i \mathbb{E}_g(\frac{1}{p_i}(A^{(i)})^Tg^TgA^{(i)}) = \sum_{i=1}^s (A^{(i)})^T\mathbb{E}_g(g^Tg)A^{(i)} = \sum_{i=1}^s (A^{(i)})^TA^{(i)} = A^TA,$$

since $\mathbb{E}(g^Tg) = I$. Let $Y_k = (A^TA)^{-1/2}X_k(A^TA)^{-1/2}$, and note that by the above, $\mathbb{E}(Y_k) = I$.

Next we have

$$\|Y_k\| = \left\|(A^TA)^{-1/2}\left(\frac{1}{p_i}(A^{(i)})^Tgg^TA^{(i)}\right)(A^TA)^{-1/2}\right\| = g^T\left(\frac{1}{p_i}A^{(i)}(A^TA)^{-1}(A^{(i)})^T\right)g.$$

For fixed i, this latter expression is the classic Hutchinson's trace estimator for the matrix $\frac{1}{p_i}A^{(i)}(A^TA)^{-1}(A^{(i)})^T$, which has mean

$$\mathrm{Tr}\left(\frac{1}{p_i}A^{(i)}(A^TA)^{-1}(A^{(i)})^T\right) = \frac{1}{p_i}\mathcal{L}_i(A) \leq \frac{d}{\beta}.$$

So by Proposition 5.22,

$$\mathbb{P}(\|Y_k\| \geq t) \leq c_1 e^{-c_2 t/\mu}, \tag{5.2}$$

for $t \geq c_3\mu$, where we set $\mu = d/\beta$.

At this point we would like to apply Matrix Chernoff to the Y_k's. Unfortunately we cannot since the Y_k's are not bounded a.s. Therefore we let $\tilde{Y_k}$ be the random variable got by conditioning Y_k on the event that $\|Y_k\| \leq M$. We will show below that taking $M = c_5\mu \log \frac{\mu}{\epsilon}$ gives $\left\|\mathbb{E}Y_k - \mathbb{E}\tilde{Y_k}\right\| \leq \epsilon$. As a consequence this gives

$$(1-\epsilon)I \leq \mathbb{E}(\tilde{Y_k}) \leq (1+\epsilon)I.$$

Given this choice of M, we apply the Matrix Chernoff bound to the $\tilde{Y_k}$'s, which now satisfy $\left\|\tilde{Y_k}\right\| \leq M$. Setting $\overline{\tilde{Y}} = \frac{1}{N}\sum_{i=1}^{N} \tilde{Y_k}$, and plugging into Theorem 5.21 gives

$$\mathbb{P}(\left\|\overline{\tilde{Y}} - I\right\| \geq 3\epsilon) \leq d \exp\left(-c\epsilon^2 \frac{N}{M}\right),$$

which is bounded by 0.05 for $N \geq \Omega\left(\frac{d}{\beta\epsilon^2} \log\left(\frac{d}{\beta\epsilon}\right) \log d\right)$. (We replace ϵ with $\epsilon/3$ to recover the statement in the theorem.) Possibly by adjusting the constant in the definition of M, we can arrange so that with probability at least 0.95, all of the N samples Y_k are such that $\|Y_k\| \leq M$ (this follows from the exponential tail bound on the Y_k's), and hence indistinguishable from the $\tilde{Y_k}$'s. So with probability at least 0.9 the Y_k's enjoy the same concentration bound as the $\tilde{Y_k}$'s above, which then implies the conclusion of the theorem.

Finally we conclude the argument by showing that $\mathbb{E}\tilde{Y_k}$ is approximately $\mathbb{E}Y_k$. To simplify notation, let Y and $\tilde{Y}$ be distributed as Y_k and $\tilde{Y_k}$ respectively. We write

$$\mathbb{E}Y = \mathbb{E}(\tilde{Y})\mathbb{P}(\|Y\| \leq M) + \mathbb{E}(Y| \, \|Y\| \geq M)\mathbb{P}(\|Y\| \geq M).$$

Thus we have

$$\begin{aligned}\left\|\mathbb{E}Y - \mathbb{E}\tilde{Y}\right\| &\leq (1 - \mathbb{P}(\|Y\| \leq M)) \left\|\mathbb{E}\tilde{Y}\right\| + \mathbb{P}(\|Y\| > M) \left\|\mathbb{E}(Y| \, \|Y\| \geq M)\right\| \\ &= \mathbb{P}(\|Y\| > M) \left\|\mathbb{E}\tilde{Y}\right\| + \mathbb{P}(\|Y\| > M) \left\|\mathbb{E}(Y| \, \|Y\| \geq M)\right\| \\ &\leq \mathbb{P}(\|Y\| > M)\mathbb{E}\left\|\tilde{Y}\right\| + \mathbb{P}(\|Y\| > M)\mathbb{E}(\|Y\| \,|\, \|Y\| \geq M) \\ &\leq \mathbb{P}(\|Y\| > M)\mathbb{E}\left\|Y\right\| + \mathbb{P}(\|Y\| > M)\mathbb{E}(\|Y\| \,|\, \|Y\| \geq M),\end{aligned}$$

where in the last step we observed that $\mathbb{E}\left\|\tilde{Y}\right\| \leq \mathbb{E}\left\|Y\right\|$. We bound each of the relevant terms.

We will take $M \geq c_3\mu$. As shown above, $\mathbb{E}\left\|Y\right\| \leq \mu$. By 5.2, $\mathbb{P}(\|Y\| > M) \leq c_1 e^{-c_2 M/\mu}$. To handle the last term,

$$\mathbb{E}(\|Y\| \,|\, \|Y\| \geq M) = \int_0^\infty \mathbb{P}\left(\|Y\| \geq t| \, \|Y\| \geq M\right) \, dt = M + \int_M^\infty \mathbb{P}\left(\|Y\| \geq t| \, \|Y\| \geq M\right) \, dt.$$

So

$$\begin{aligned}\mathbb{P}(\|Y\| > M)\mathbb{E}(\|Y\| \,|\, \|Y\| \geq M) &= \mathbb{P}(\|Y\| > M)M + \int_M^\infty \mathbb{P}\left(\|Y\| \geq t\right) \, dt \\ &\leq c_1 e^{-c_2 M/\mu} M + \int_M^\infty c_1 e^{-c_2 t/\mu} \, dt \\ &= c_1 e^{-c_2 M/\mu} M + c_4 \mu e^{-c_2 M/\mu}.\end{aligned}$$

Putting the pieces together gives

$$\left\|\mathbb{E}Y - \mathbb{E}\tilde{Y}\right\| \leq c_1 \mu e^{-c_2 M/\mu} + c_1 M e^{-c_2 M/\mu} + c_4 \mu e^{-c_2 M/\mu},$$

which is bounded by ϵ for $M \geq c_5 \mu \log\left(\frac{\mu}{\epsilon}\right)$. □

CHAPTER 6

Conclusion

In this thesis we studied efficiently solving linear algebra problems from a variety of angles.

We first studied the random Kaczmarz method and asked whether it could adapted to handle a small fraction of corruptions. We showed that it could, and developed two algorithms that we called QuantileRK and QuantileSGD that are more robust while maintaining the low RAM requirements of Kaczmarz.

We then turned our attention to the problem of estimating the eigenvalues of a symmetric matrix. We began by considering a particularly simple variant of the problem – testing if a matrix is positive semi-definite. For the PSD-testing problem we gave tight query-complexity bounds in terms of both matrix-vector products and bilinear form queries. We considered both adaptive and non-adaptive methods which required separate techniques. For adaptive measurements we gave an optimal algorithm based on stochastic gradient descent, whereas for non-adaptive measurements we saw that sketching gives optimal bounds.

The PSD-testing sketch was only presented for approximating the smallest eigenvalue, however in the following chapter we showed that with more work it could be adapted into a sketch for approximating all eigenvalues. This led us to develop an optimal sketch for approximating eigenvalues of a matrix A to within additive $\epsilon \|A\|_F$ error.

Finally we returned our attention to the heart of linear algebra – solving linear systems. When studying Kaczmarz we were interested in space-efficiency. Here we instead considered communication-efficiency, and gave the first optimal regression algorithms in the coordinator model.

There are many interesting directions that this work leaves open. We list only a few here.

When studying Kaczmarz, we focused mainly on the situation where the true system is fixed. However what happens in the streaming setting where an attacker may intentionally choose to inject corrupt rows? Under what circumstances is it possible to guarantee convergence to the true solution?

It would also be interesting to improve our spectral approximation algorithm. In particular when approximating the spectrum of a sparse matrix, it should be possible to do so in time nearly proportional to the number of nonzero entries. Is it possible to obtain such a guarantee?

Finally, are there new situations where the ideas behind our regression protocols can be employed? Linear regression and subspace embeddings are useful primitives. Can our protocol be employed as a black-box in other situations to give improved algorithms?

BIBLIOGRAPHY

[Kac37] S. Kaczmarz. “Angenäherte Auflösung von Systemen linearer Gleichungen”. In: *Bull. Internat. Acad. Polon. Sci. Lettres A* (1937), pp. 335–357.

[RM51] Herbert Robbins and Sutton Monro. “A stochastic approximation method”. In: *Ann. Math. Stat.* (1951), pp. 400–407.

[Agm54] S Agmon. “The relaxation method for linear inequalities”. In: *Canadian J. Math.* (1954), pp. 382–392. ISSN: 0008-414X.

[MS54] Theodore S Motzkin and Isaac J Schoenberg. “The relaxation method for linear inequalities”. In: *Canadian J. Math.* 6 (1954), pp. 393–404.

[And62] Theodore Wilbur Anderson. *An introduction to multivariate statistical analysis*. Tech. rep. Wiley New York, 1962.

[GBH70] R. Gordon, R. Bender, and G. T. Herman. “Algebraic reconstruction techniques (ART) for three-dimensional electron microscopy and X-ray photography”. In: *J. Theoret. Biol.* 29 (1970), pp. 471–481.

[BR73] Ian Barrodale and Frank DK Roberts. “An improved algorithm for discrete ℓ_1 linear approximation”. In: *SIAM J. Numer. Anal.* 10.5 (1973), pp. 839–848.

[Sch73] EJ Schlossmacher. “An iterative technique for absolute deviations curve fitting”. In: *J. Am. Stat. Assoc.* 68.344 (1973), pp. 857–859.

[BS80] Peter Bloomfield and William Steiger. “Least absolute deviations curve-fitting”. In: *SIAM J. Sci. Stat. Comp.* 1.2 (1980), pp. 290–301.

[Elf80] Tommy Elfving. “Block-iterative methods for consistent and inconsistent linear equations”. In: *Numer. Math.* 35.1 (1980), pp. 1–12.

[Cen81] Y Censor. “Row-action methods for huge and sparse systems and their applications”. In: *SIAM Rev.* 23 (1981), pp. 444–466. ISSN: 0036-1445. DOI: 10.1137/1023097.

[EHL81] Paulus Petrus Bernardus Eggermont, Gabor T Herman, and Arnold Lent. "Iterative algorithms for large partitioned linear systems, with applications to image reconstruction". In: *Linear Algebra Appl.* 40 (1981), pp. 37–67.

[Wes81] GO Wesolowsky. "A new descent algorithm for the least absolute value regression problem: A new descent algorithm for the least absolute value". In: *Commun. Stat. Simulat.* 10.5 (1981), pp. 479–491.

[Jon82] Dag Jonsson. "Some limit theorems for the eigenvalues of a sample covariance matrix". In: *Journal of Multivariate Analysis* 12.1 (1982), pp. 1–38.

[Oja82] Erkki Oja. "Simplified neuron model as a principal component analyzer". In: *Journal of mathematical biology* 15.3 (1982), pp. 267–273.

[CEG83] Yair Censor, Paul P B Eggermont, and Dan Gordon. "Strong underrelaxation in Kaczmarz's method for inconsistent systems". In: *Numer. Math.* 41 (1983), pp. 83–92. ISSN: 0029599X. DOI: 10.1007/BF01396307.

[Nat86] Frank Natterer. *The mathematics of computerized tomography*. B. G. Teubner, Stuttgart; John Wiley & Sons, Ltd., Chichester, 1986, pp. x+222. ISBN: 3-519-02103-X.

[SS87] M Ibrahim Sezan and Henry Stark. "Incorporation of a priori moment information into signal recovery and synthesis problems". In: *J. Math. Anal. Apl.* 122.1 (1987), pp. 172–186.

[GSN88] James E Gentle, VA Sposito, and Subhash C Narula. "Algorithms for unconstrained L_1 simple linear regression". In: *J. Comp. Stat. Data Anal.* 6.4 (1988), pp. 335–339.

[Hut89] Michael F Hutchinson. "A stochastic estimator of the trace of the influence matrix for Laplacian smoothing splines". In: *Communications in Statistics-Simulation and Computation* 18.3 (1989), pp. 1059–1076.

[HN90] Martin Hanke and Wilhelm Niethammer. “On the acceleration of Kaczmarz’s method for inconsistent linear systems”. In: *Linear Algebra Appl.* 130 (1990), pp. 83–98. ISSN: 00243795. DOI: 10.1016/0024-3795(90)90207-S.

[Fei+92] H G Feichtinger et al. “New variants of the POCS method using affine subspaces of finite codimension with applications to irregular sampling”. In: *P. Soc. Photo-Opt. Ins.* Vol. 1818. International Society for Optics and Photonics. 1992, pp. 299–311.

[HM93] Gabor T Herman and Lorraine B Meyer. “Algebraic reconstruction techniques can be made computationally efficient (positron emission tomography application)”. In: *IEEE T. Med. Imaging* 12.3 (1993), pp. 600–609.

[AK95] Edoardo Amaldi and Viggo Kann. “The complexity and approximability of finding maximum feasible subsystems of linear relations”. In: *Theor. Comput. Sci.* 147.1-2 (1995), pp. 181–210.

[FS95] Hans G Feichtinger and Thomas Strohmer. “A Kaczmarz-based approach to nonperiodic sampling on unions of rectangular lattices”. In: *SampTA’95: 1995 Workshop on Sampling Theory and Applications*. 1995, pp. 32–37.

[BFG96] Zhaojun Bai, Gark Fahey, and Gene Golub. “Some large-scale matrix computation problems”. In: *Journal of Computational and Applied Mathematics* 74.1-2 (1996), pp. 71–89.

[Pop99] Constantin Popa. “Block-projections algorithms with blocks containing mutually orthogonal rows and columns”. In: *BIT* 39.2 (1999), pp. 323–338.

[Pop01] Constantin Popa. “A fast Kaczmarz-Kovarik algorithm for consistent least-squares problems”. In: *Korean J. Comp. App. Math.* 8.1 (2001), pp. 9–26.

[SHS01] Andreas Savvides, Chih-Chieh Han, and Mani B Strivastava. “Dynamic fine-grained localization in ad-hoc networks of sensors”. In: *Proc. Int. Conf. on Mobile computing and networking*. 2001, pp. 166–179.

[KS03] Robert Krauthgamer and Ori Sasson. “Property testing of data dimensionality”. In: *Proceedings of the Fourteenth Annual ACM-SIAM Symposium on Discrete Algorithms, January 12-14, 2003, Baltimore, Maryland, USA*. ACM/SIAM, 2003, pp. 18–27.

[BV04] Stephen Boyd and Lieven Vandenberghe. *Convex optimization*. Cambridge university press, 2004.

[LA04] Yinbo Li and Gonzalo R Arce. “A maximum likelihood approach to least absolute deviation regression”. In: *Eurasip. J. Adv. Sig. Pr.* 2004.12 (2004), p. 948982.

[ABH05] Edoardo Amaldi, Pietro Belotti, and Raphael Hauser. “Randomized relaxation methods for the maximum feasible subsystem problem”. In: *Integer programming and combinatorial optimization*. Vol. 3509. Lecture Notes in Comput. Sci. Springer, Berlin, 2005, pp. 249–264.

[CT05] Emmanuel Candes and Terence Tao. “Decoding by linear programming”. In: *arXiv preprint math/0502327* (2005).

[Can+05] Emmanuel Candes et al. “Error correction via linear programming”. In: *FOCS*. IEEE. 2005, pp. 668–681.

[Lit+05] Alexander E Litvak et al. “Smallest singular value of random matrices and geometry of random polytopes”. In: *Advances in Mathematics* 195.2 (2005), pp. 491–523.

[Jia06] Tiefeng Jiang. “How many entries of a typical orthogonal matrix can be approximated by independent normals?” In: *The Annals of Probability* 34.4 (2006), pp. 1497–1529.

[WGZ06] Li Wang, Michael D Gordon, and Ji Zhu. “Regularized least absolute deviations regression and an efficient algorithm for parameter tuning”. In: *Int. Conf. on Data Mining*. IEEE. 2006, pp. 690–700.

[Han07] Per Christian Hansen. “Regularization tools version 4.0 for Matlab 7.3”. In: *Numer. Algorithms* 46.2 (2007), pp. 189–194.

[And+09] Alexandr Andoni et al. “Efficient sketches for earth-mover distance, with applications”. In: *2009 50th Annual IEEE Symposium on Foundations of Computer Science*. IEEE. 2009, pp. 324–330.

[SV09] Thomas Strohmer and Roman Vershynin. “A randomized Kaczmarz algorithm with exponential convergence”. In: *J Fourier Anal. Appl.* 15.2 (2009), p. 262.

[Bot10] Léon Bottou. “Large-scale machine learning with stochastic gradient descent”. In: *Proc. of COMPSTAT*. Springer, 2010, pp. 177–186.

[Nee10] Deanna Needell. “Randomized Kaczmarz solver for noisy linear systems”. In: *BIT* 50.2 (2010), pp. 395–403.

[Ver10] Roman Vershynin. “Introduction to the non-asymptotic analysis of random matrices”. In: *arXiv preprint arXiv:1011.3027* (2010).

[WM10] John Wright and Yi Ma. “Dense Error Correction Via ℓ^1-Minimization”. In: *IEEE T. Infor. Theory* 56.7 (2010), pp. 3540–3560.

[AT11] Haim Avron and Sivan Toledo. “Randomized algorithms for estimating the trace of an implicit symmetric positive semi-definite matrix”. In: *Journal of the ACM (JACM)* 58.2 (2011), pp. 1–34.

[Ver11] Roman Vershynin. “Spectral norm of products of random and deterministic matrices”. In: *Probability theory and related fields* 150.3-4 (2011), pp. 471–509.

[CP12] X Chen and A Powell. “Almost Sure Convergence of the Kaczmarz Algorithm with Random Measurements”. In: *J.Fourier Anal. Appl.* (2012), pp. 1–20.

[Dek+12] Ofer Dekel et al. “Optimal distributed online prediction using mini-batches”. In: *J. Mach. Learn. Res.* 13 (2012), pp. 165–202.

[EK12] Yonina C Eldar and Gitta Kutyniok. *Compressed sensing: theory and applications*. Cambridge University Press, 2012.

[Le 12] François Le Gall. “Faster algorithms for rectangular matrix multiplication”. In: *2012 IEEE 53rd annual symposium on foundations of computer science*. IEEE. 2012, pp. 514–523.

[Tro12] Joel A Tropp. “User-friendly tail bounds for sums of random matrices”. In: *Foundations of computational mathematics* 12.4 (2012), pp. 389–434.

[AN13] Alexandr Andoni and Huy L Nguyn. “Eigenvalues of a matrix in the streaming model”. In: *Proceedings of the twenty-fourth annual ACM-SIAM symposium on Discrete algorithms*. SIAM. 2013, pp. 1729–1737.

[FR13] S. Foucart and H. Rauhut. *A Mathematical Introduction to Compressive Sensing*. In press. Springer, 2013.

[Lic13] M Lichman. *{UCI} Machine Learning Repository*. 2013.

[SZ13] Ohad Shamir and Tong Zhang. “Stochastic gradient descent for non-smooth optimization: Convergence results and optimal averaging schemes”. In: *Int. conf. on machine learning*. 2013, pp. 71–79.

[ZF13] Anastasios Zouzias and Nikolaos M. Freris. “Randomized extended Kaczmarz for solving least squares”. In: *SIAM J. Matrix Anal. A.* 34 (2013), pp. 773–793. ISSN: 08954798. DOI: 10.1137/120889897. arXiv: 1205.5770.

[LNW14] Yi Li, Huy L. Nguyen, and David P. Woodruff. “On Sketching Matrix Norms and the Top Singular Vector”. In: *Proceedings of the Twenty-Fifth Annual ACM-SIAM Symposium on Discrete Algorithms, SODA 2014, Portland, Oregon, USA, January 5-7, 2014*. Ed. by Chandra Chekuri. SIAM, 2014, pp. 1562–1581.

[NT14] Deanna Needell and Joel A Tropp. “Paved with good intentions: analysis of a randomized block Kaczmarz method”. In: *Linear Algebra Appl.* 441 (2014), pp. 199–221.

[WWZ14] Karl Wimmer, Yi Wu, and Peng Zhang. “Optimal query complexity for estimating the trace of a matrix”. In: *International Colloquium on Automata, Languages, and Programming*. Springer. 2014, pp. 1051–1062.

[Woo+14] David P Woodruff et al. “Sketching as a tool for numerical linear algebra”. In: *Foundations and Trends® in Theoretical Computer Science* 10.1–2 (2014), pp. 1–157.

[CNW15] Michael B Cohen, Jelani Nelson, and David P Woodruff. “Optimal approximate matrix product in terms of stable rank”. In: *arXiv preprint arXiv:1507.02268* (2015).

[CP15] Michael B Cohen and Richard Peng. “Lp row sampling by lewis weights”. In: *Proceedings of the forty-seventh annual ACM symposium on Theory of computing*. 2015, pp. 183–192.

[Coh+15] Michael B Cohen et al. “Uniform sampling for matrix approximation”. In: *Proceedings of the 2015 Conference on Innovations in Theoretical Computer Science*. 2015, pp. 181–190.

[JCC15] Noreen Jamil, Xuemei Chen, and Alexander Cloninger. “Hildreth’s algorithm with applications to soft constraints for user interface layout”. In: *J. Comput. Appl. Math.* 288 (2015), pp. 193–202.

[MM15] Cameron Musco and Christopher Musco. “Randomized block krylov methods for stronger and faster approximate singular value decomposition”. In: *arXiv preprint arXiv:1504.05477* (2015).

[PP15] Stefania Petra and Constantin Popa. “Single projection Kaczmarz extended algorithms”. In: *Numer. Algorithms* (2015), pp. 1–16. ISSN: 15729265. DOI: 10.1007/s11075-016-0118-7. arXiv: 1504.00231.

[RV15] Mark Rudelson and Roman Vershynin. “Small ball probabilities for linear images of high-dimensional distributions”. In: *Int. Math. Res. Notices* 2015.19 (2015), pp. 9594–9617.

[Wai15] Martin Wainwright. “Basic tail and concentration bounds”. In: *URl: https://www. stat. berkeley. edu/.../Chap2_TailBounds_Jan22_2015. pdf (visited on 12/31/2017)* (2015).

[Ai+16] Yuqing Ai et al. “New characterizations in turnstile streams with applications”. In: *31st Conference on Computational Complexity (CCC 2016)*. Schloss Dagstuhl-Leibniz-Zentrum fuer Informatik. 2016.

[Jai+16] Prateek Jain et al. “Streaming pca: Matching matrix bernstein and near-optimal finite sample guarantees for oja’s algorithm”. In: *Conference on learning theory*. PMLR. 2016, pp. 1147–1164.

[Kyn+16] Rasmus Kyng et al. “Sparsified cholesky and multigrid solvers for connection laplacians”. In: *Proceedings of the forty-eighth annual ACM symposium on Theory of Computing*. 2016, pp. 842–850.

[LW16] Yi Li and David P Woodruff. “Tight bounds for sketching the operator norm, Schatten norms, and subspace embeddings”. In: *Approximation, Randomization, and Combinatorial Optimization. Algorithms and Techniques (APPROX/RANDOM 2016)*. Schloss Dagstuhl-Leibniz-Zentrum fuer Informatik. 2016.

[NSW16] Deanna Needell, Nathan Srebro, and Rachel Ward. “Stochastic gradient descent, weighted sampling, and the randomized Kaczmarz algorithm”. In: *Math. Program.* 155.1-2 (2016), pp. 549–573.

[Nut+16] Julie Nutini et al. “Convergence Rates for Greedy Kaczmarz Algorithms”. In: *UAI* (2016).

[Sha16] Ohad Shamir. “Convergence of stochastic gradient descent for PCA”. In: *International Conference on Machine Learning*. PMLR. 2016, pp. 257–265.

[Woo16] David P. Woodruff. “New Algorithms for Heavy Hitters in Data Streams (Invited Talk)”. In: *19th International Conference on Database Theory, ICDT 2016, Bor-*

deaux, France, March 15-18, 2016. Ed. by Wim Martens and Thomas Zeume. Vol. 48. LIPIcs. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2016, 4:1–4:12.

[AL17] Zeyuan Allen-Zhu and Yuanzhi Li. "First efficient convergence for streaming k-pca: a global, gap-free, and near-optimal rate". In: *2017 IEEE 58th Annual Symposium on Foundations of Computer Science (FOCS)*. IEEE. 2017, pp. 487–492.

[CW17a] Kenneth L Clarkson and David P Woodruff. "Low-rank approximation and regression in input sparsity time". In: *Journal of the ACM (JACM)* 63.6 (2017), pp. 1–45.

[CW17b] Kenneth L Clarkson and David P Woodruff. "Low-rank PSD approximation in input-sparsity time". In: *Proceedings of the Twenty-Eighth Annual ACM-SIAM Symposium on Discrete Algorithms*. SIAM. 2017, pp. 2061–2072.

[DHN17] J. A. De Loera, J. Haddock, and D. Needell. "A Sampling Kaczmarz-Motzkin Algorithm for Linear Feasibility". In: *SIAM J. Sci. Comput.* 39.5 (2017), S66–S87.

[Han+17] Insu Han et al. "Approximating spectral sums of large-scale matrices using stochastic chebyshev approximations". In: *SIAM Journal on Scientific Computing* 39.4 (2017), A1558–A1585.

[BW18a] Zhong-Zhi Bai and Wen-Ting Wu. "On greedy randomized Kaczmarz method for solving large sparse linear systems". In: *SIAM J. Sci. Comput.* 40.1 (2018), A592–A606.

[BW18b] Zhong-Zhi Bai and Wen-Ting Wu. "On relaxed greedy randomized Kaczmarz methods for solving large sparse linear systems". In: *Appl. Math. Lett.* 83 (2018), pp. 21–26.

[BCN18] Léon Bottou, Frank E Curtis, and Jorge Nocedal. "Optimization methods for large-scale machine learning". In: *SIAM Rev.* 60.2 (2018), pp. 223–311.

[GU18] François Le Gall and Florent Urrutia. “Improved rectangular matrix multiplication using powers of the Coppersmith-Winograd tensor”. In: *Proceedings of the Twenty-Ninth Annual ACM-SIAM Symposium on Discrete Algorithms*. SIAM. 2018, pp. 1029–1046.

[HN18a] J. Haddock and D. Needell. “Randomized Projection Methods for Linear Systems with Arbitrarily Large Sparse Corruptions”. In: *SIAM J. Sci. Comput.* 41.5 (2018), S19–S36.

[HN18b] Jamie Haddock and Deanna Needell. “Randomized projections for corrupted linear systems”. In: *AIP Conf. Proc.* 1978 1. AIP Publishing. 2018, p. 470071.

[KS18] Ana Sović Kržić and Damir Seršić. “L1 minimization using recursive reduction of dimensionality”. In: *Signal Process.* 151 (2018), pp. 119–129.

[SER18] Max Simchowitz, Ahmed El Alaoui, and Benjamin Recht. “Tight query complexity lower bounds for PCA via finite sample deformed Wigner law”. In: *Proceedings of the 50th Annual ACM SIGACT Symposium on Theory of Computing*. 2018, pp. 1249–1259.

[Ver18] Roman Vershynin. *High-dimensional probability: An introduction with applications in data science*. Vol. 47. Cambridge university press, 2018.

[Bal+19a] Maria-Florina Balcan et al. “Testing Matrix Rank, Optimally”. In: *Proceedings of the Thirtieth Annual ACM-SIAM Symposium on Discrete Algorithms, SODA 2019, San Diego, California, USA, January 6-9, 2019*. Ed. by Timothy M. Chan. SIAM, 2019, pp. 727–746.

[Bal+19b] Maria-Florina Balcan et al. “Testing matrix rank, optimally”. In: *Proceedings of the Thirtieth Annual ACM-SIAM Symposium on Discrete Algorithms*. SIAM. 2019, pp. 727–746.

[Chi+19] Yuejie Chi et al. “Median-Truncated Gradient Descent: A Robust and Scalable Nonconvex Approach for Signal Estimation”. In: *Appl. Numer. Harmon. An.* Springer, 2019, pp. 237–261.

[HN19] J. Haddock and D. Needell. “On Motzkin’s Method for Inconsistent Linear Systems”. In: *BIT* 59.2 (2019), pp. 387–401.

[HM19] Jamie Haddock and Anna Ma. “Greed Works: An Improved Analysis of Sampling Kaczmarz-Motzkin”. In: *arXiv preprint arXiv:1912.03544* (2019).

[LNW19] Yi Li, Huy L. Nguyen, and David P. Woodruff. “On Approximating Matrix Norms in Data Streams”. In: *SIAM J. Comput.* 48.6 (2019), pp. 1643–1697.

[LR19] Nicolas Loizou and Peter Richtárik. “Revisiting Randomized Gossip Algorithms: General Framework, Convergence Rates and Novel Block and Accelerated Protocols”. In: *arXiv preprint arXiv:1905.08645* (2019).

[MSW19] Michela Meister, Tamas Sarlos, and David Woodruff. “Tight dimensionality reduction for sketching low degree polynomial kernels”. In: (2019).

[MIN19] M. S. Morshed, M. S. Islam, and M. Noor-E-Alam. “Accelerated sampling Kaczmarz Motzkin algorithm for the linear feasibility problem”. In: *J. Global Optim.* (2019), pp. 1–22.

[Sun+19] Xiaoming Sun et al. “Querying a matrix through matrix-vector products”. In: *arXiv preprint arXiv:1906.05736* (2019).

[Ahl+20] Thomas D Ahle et al. “Oblivious sketching of high-degree polynomial kernels”. In: *Proceedings of the Fourteenth Annual ACM-SIAM Symposium on Discrete Algorithms*. SIAM. 2020, pp. 141–160.

[BCJ20] Ainesh Bakshi, Nadiia Chepurko, and Rajesh Jayaram. “Testing positive semi-definiteness via random submatrices”. In: *2020 IEEE 61st Annual Symposium on Foundations of Computer Science (FOCS)*. IEEE. 2020, pp. 1191–1202.

[Bra+20] Mark Braverman et al. “The gradient complexity of linear regression”. In: *Conference on Learning Theory*. PMLR. 2020, pp. 627–647.

[DSS20] Kui Du, Wutao Si, and Xiaohui Sun. “Pseudoinverse-free randomized extended block Kaczmarz for solving least squares”. In: *arXiv preprint arXiv:2001.04179* (2020).

[Had+20] J. Haddock et al. “Stochastic Gradient Descent Methods for Corrupted Systems of Linear Equations”. In: *Proc. Conf. on Information Sciences and Systems*. 2020.

[KL20] Kenji Kawaguchi and Haihao Lu. “Ordered SGD: A New Stochastic Optimization Framework for Empirical Risk Minimization”. In: *Int. Conf. on Artificial Intelligence and Statistics*. 2020, pp. 669–679.

[Li+20] Yuanxin Li et al. “Non-convex low-rank matrix recovery with arbitrary outliers via median-truncated gradient descent”. In: *Information and Inference: A Journal of the IMA* 9.2 (2020), pp. 289–325.

[Mah+20] Sepideh Mahabadi et al. “Non-adaptive adaptive sampling on turnstile streams”. In: *Proceedings of the 52nd Annual ACM SIGACT Symposium on Theory of Computing*. 2020, pp. 1251–1264.

[MI+20] Md Sarowar Morshed, Md Saiful Islam, et al. “On Generalization and Acceleration of Randomized Projection Methods for Linear Feasibility Problems”. In: *arXiv preprint arXiv:2002.07321* (2020).

[RWZ20] Cyrus Rashtchian, David P Woodruff, and Hanlin Zhu. “Vector-matrix-vector queries for solving linear algebra, statistics, and graph problems”. In: *arXiv preprint arXiv:2006.14015* (2020).

[RN20] Elizaveta Rebrova and Deanna Needell. “On block Gaussian sketching for the Kaczmarz method”. In: *Numer. Algorithms* (2020), pp. 1–31.

[VWW20] Santosh S Vempala, Ruosong Wang, and David P Woodruff. “The communication complexity of optimization”. In: *Proceedings of the Fourteenth Annual ACM-SIAM Symposium on Discrete Algorithms*. SIAM. 2020, pp. 1733–1752.

[BMR21] Rajarshi Bhattacharjee, Cameron Musco, and Archan Ray. “Sublinear Time Eigenvalue Approximation via Random Sampling”. In: *CoRR* abs/2109.07647 (2021).

[Bha+21] Rajarshi Bhattacharjee et al. “Sublinear Time Eigenvalue Approximation via Random Sampling”. In: *CoRR* abs/2109.07647 (2021).

[DM21] Prathamesh Dharangutte and Christopher Musco. “Dynamic Trace Estimation”. In: *Advances in Neural Information Processing Systems* 34 (2021), pp. 30088–30099.

[JN21] Benjamin Jarman and Deanna Needell. “QuantileRK: Solving Large-Scale Linear Systems with Corrupted, Noisy Data”. In: *arXiv preprint arXiv:2108.02304* (2021).

[Jia+21] Shuli Jiang et al. “Optimal Sketching for Trace Estimation”. In: *Advances in Neural Information Processing Systems 34: Annual Conference on Neural Information Processing Systems 2021, NeurIPS 2021, December 6-14, 2021, virtual*. Ed. by Marc’Aurelio Ranzato et al. 2021, pp. 23741–23753.

[KPW21] Akshay Kamath, Eric Price, and David P Woodruff. “A simple proof of a new set disjointness with applications to data streams”. In: *arXiv preprint arXiv:2105.11338* (2021).

[Mey+21] Raphael A Meyer et al. “Hutch++: Optimal stochastic trace estimation”. In: *Symposium on Simplicity in Algorithms (SOSA)*. SIAM. 2021, pp. 142–155.

[Sun+21] Xiaoming Sun et al. “Querying a matrix through matrix-vector products”. In: *ACM Transactions on Algorithms (TALG)* 17.4 (2021), pp. 1–19.

[BKM22] Vladimir Braverman, Aditya Krishnan, and Christopher Musco. “Sublinear time spectral density estimation”. In: *STOC ’22: 54th Annual ACM SIGACT Symposium on

Theory of Computing, Rome, Italy, June 20 - 24, 2022. Ed. by Stefano Leonardi and Anupam Gupta. ACM, 2022, pp. 1144–1157.

[Had+22] Jamie Haddock et al. "Quantile-based iterative methods for corrupted systems of linear equations". In: *SIAM Journal on Matrix Analysis and Applications* 43.2 (2022), pp. 605–637.

[INW22] Piotr Indyk, Shyam Narayanan, and David P. Woodruff. "Frequency Estimation with One-Sided Error". In: *SODA*. 2022.

[NSW22] Deanna Needell, William Swartworth, and David P Woodruff. "Testing Positive Semidefiniteness Using Linear Measurements". In: *2022 IEEE 63rd Annual Symposium on Foundations of Computer Science (FOCS)*. IEEE. 2022, pp. 87–97.

[WZZ22] David P Woodruff, Fred Zhang, and Qiuyi Zhang. "Optimal Query Complexities for Dynamic Trace Estimation". In: *arXiv preprint arXiv:2209.15219* (2022).

[SW23] William Swartworth and David P Woodruff. "Optimal Eigenvalue Approximation via Sketching". In: *arXiv preprint arXiv:2304.09281* (2023).