Beyond the Worst-Case Analysis of Algorithms

Edited by
Tim Roughgarden

Contents

1
Robust High-Dimensional Statistics

Ilias Diakonikolas and Daniel M. Kane

Abstract

Learning in the presence of outliers is a fundamental problem in statistics. Until recently, all known efficient unsupervised learning algorithms were very sensitive to outliers in high dimensions. In particular, even for the task of robust mean estimation under natural distributional assumptions, no efficient algorithm was known. A recent line of work gave the first efficient robust estimators for a number of fundamental statistical tasks, including mean and covariance estimation. This chapter introduces the core ideas and techniques in the emerging area of algorithmic high-dimensional robust statistics with a focus on robust mean estimation.

1.1 Introduction

Consider the following basic statistical task: Given n independent samples from an unknown mean spherical Gaussian distribution $\mathcal{N}(\mu, I)$ on $\mathbf{R}^d$, estimate its mean vector μ within small ℓ_2-norm. It is not hard to see that the empirical mean has ℓ_2-error at most $O(\sqrt{d/n})$ from μ with high probability. Moreover, this error upper bound is best possible among all n-sample estimators.

The Achilles heel of the empirical estimator is that it crucially relies on the assumption that the observations were generated by a spherical Gaussian. The existence of even a *single* outlier can arbitrarily compromise this estimator's performance. However, the Gaussian assumption is only ever approximately valid, as real datasets are typically exposed to some source of contamination. Hence, any estimator that is to be used in practice must be *robust* in the presence of outliers.

Learning in the presence of outliers is an important goal in statistics and has been studied in the robust statistics community since the 1960s (Huber, 1964). Classical work in statistics pinned down the sample complexity of high-dimensional robust estimation in several settings of interest. In contrast, until very recently, even the most basic computational questions in this field were poorly understood. For example, the Tukey median (Tukey, 1975) is a sample-efficient robust mean estimator for spherical Gaussian distributions. However, it is NP-hard to compute in general (Johnson and Preparata, 1978) and the heuristics proposed to approximate it degrade in the quality of their approximation as the dimension scales.

Until recently, all known computationally efficient high-dimensional estimators could only tolerate a negligible fraction of outliers, even for the basic statistical task of mean estimation. Recent work by Diakonikolas, Kamath, Kane, Li, Moitra, and Stewart (Diakonikolas et al., 2016), and by Lai, Rao, and Vempala (Lai et al., 2016) gave the first efficient robust estimators for various high-dimensional unsupervised tasks, including mean and covariance estimation. Specifically, Diakonikolas et al. (2016) obtained the first robust estimators with *dimension-independent* error, i.e., with error scaling only with the fraction of corrupted samples and *not* with the dimensionality of the data. Since then, there has been significant research activity on designing computationally efficient robust estimators in a variety of settings.

Contamination Model. Throughout this chapter, we focus on the following model of robust estimation that generalizes several other existing models:

Definition 1.1 Given $0 < \epsilon < 1/2$ and a distribution family $\mathcal{D}$ on $\mathbf{R}^d$, the *adversary* operates as follows: The algorithm specifies a number of samples n, and n samples are drawn from some unknown $D \in \mathcal{D}$. The adversary is allowed to inspect the samples, remove up to ϵn of them and replace them with arbitrary points. This modified set of n points is then given to the algorithm. We say that a set of samples is *ϵ-corrupted* if it is generated by the above process.

The contamination model of Definition 1.1 is qualitatively similar to the semi-random models studied in Chapters 9 and 10 of this book: First, nature draws a set S of i.i.d. samples from a statistical model of interest, and then an adversary is allowed to change the set S in a bounded way to obtain an ϵ-corrupted set T. The parameter ϵ is the proportion of contamination and quantifies the power of the adversary. Intuitively, among our samples, a $(1 - \epsilon)$ fraction are generated from a distribution of interest and are called *inliers*, and the rest are called *outliers*.

One can consider less powerful adversaries, giving rise to weaker contamination models. An adversary may be (i) adaptive or oblivious to the inliers, (ii) only allowed to add corrupted points, or only allowed to remove existing points, or allowed to do both. For example, in Huber's contamination model (Huber, 1964), the adversary is oblivious to the inliers and is only allowed to add outliers.

In the context of robust mean estimation, given an ϵ-corrupted set of samples from a well-behaved distribution (e.g., $\mathcal{N}(\mu, I)$), we want to output a vector $\widehat{\mu}$ such that the ℓ_2-error is minimized. The goal here is to achieve *dimension-independent* error, i.e., error that scales only with the fraction of outliers ϵ.

Sample Efficient Robust Estimation. The problem of robust mean estimation seems so innocuous that one could naturally wonder why simple approaches do not work.

In the one-dimensional case, it is well-known that the median is a robust estimator of the mean in the Gaussian setting. It is easy to show (see Exercise 1.1) that several natural high-dimensional generalizations of the median (e.g., coordinate-wise median, geometric median, etc.) lead to ℓ_2-error of $\Omega(\epsilon\sqrt{d})$ in d dimensions.

It should also be noted that, in contrast to the uncorrupted i.i.d. setting, in the contaminated setting it is not possible to obtain consistent estimators — that is, estimators with error converging to zero in probability as the sample size increases indefinitely. Typically, there is an information-theoretic limit on the minimum error that depends on ϵ and structural properties of the underlying distribution family. In particular, for the one-dimensional Gaussian case, we have:

Fact 1.2 *Any robust estimator for the mean of $\mathcal{N}(\mu, 1)$, must have ℓ_2-error $\Omega(\epsilon)$, even in Huber's contamination model.*

To prove this fact, we proceed as follows: Given two distributions $\mathcal{N}(\mu_1, 1)$ and $\mathcal{N}(\mu_2, 1)$ with $|\mu_1 - \mu_2| = \Omega(\epsilon)$, the adversary constructs two noise distributions N_1, N_2 such that $(1-\epsilon)\mathcal{N}(\mu_1, 1) + \epsilon N_1 = (1-\epsilon)\mathcal{N}(\mu_2, 1) + \epsilon N_2$ (see Exercise 1.2).

Ignoring computational considerations, it is not difficult to obtain a sample-efficient robust estimator matching this error guarantee *in any dimension*:

Proposition 1.3 *There exists an (inefficient) algorithm that, on input an ϵ-corrupted set of samples from $\mathcal{N}(\mu, I)$ of size $\Omega((d + \log(1/\tau))/\epsilon^2)$, outputs $\widehat{\mu} \in \mathbf{R}^d$ such that with probability at least $1-\tau$, it holds that $\|\widehat{\mu} - \mu\|_2 = O(\epsilon)$.*

The algorithm underlying Proposition 1.3 relies on the following simple idea, which is the underlying idea in Tukey's median (Tukey, 1975): It is possible to reduce the high-dimensional robust mean estimation problem to a collection of (exponentially many) one-dimensional robust mean estimation problems. In more detail, the algorithm proceeds by using a one-dimensional robust mean estimator to estimate $v \cdot \mu$, for an appropriate net of $2^{O(d)}$ unit vectors $v \in \mathbf{R}^d$, and then combines these estimates to obtain an accurate estimate of μ (see Exercise 1.2). Tukey's median gives the same guarantee for a spherical Gaussian and can be shown to be robust for more general symmetric distributions. On the other hand, the aforementioned estimator is applicable to non-symmetric distributions as well, as long as there is an accurate robust mean estimator for each univariate projection.

Structure of this Chapter. In Section 1.2, we present efficient algorithms for robust mean estimation. Section 1.2 is the main technical section of this chapter and showcases a number of core ideas and techniques that can be applied to several high-dimensional robust estimation tasks. Section 1.3 provides a high-level overview of recent algorithmic progress for more general robust estimation tasks. Finally, in Section 1.4 we conclude with a few remarks on the relevant literature.

1.2 Robust Mean Estimation

In this section, we illustrate the main insights underlying recent algorithms for high-dimensional robust estimation by focusing on the problem of robust mean estimation. The objective of this section is to provide the intuition and background required to develop robust learning algorithms in an accessible way. As such, we will not attempt to optimize the sample or computational complexities of the algorithms presented, other than to show that they are polynomial in the relevant parameters.

In the problem of robust mean estimation, we are given an ϵ-corrupted set of samples from a distribution X on $\mathbf{R}^d$ and our goal is to approximate the mean of X, within small error in ℓ_2-norm (Euclidean distance). In order for such a goal to be information-theoretically possible, it is required that X belongs to a suitably well-behaved family of distributions. A typical assumption is that X belongs to a family whose moments are guaranteed to satisfy certain conditions, or equivalently, a family with appropriate concentration properties. In our initial discussion, we will use the running example of a spherical Gaussian, although the results presented here hold in greater generality. That is, the reader is encouraged to imagine that X is of the form $\mathcal{N}(\mu, I)$, for some unknown $\mu \in \mathbf{R}^d$.

Structure of this Section. In Section 1.2.1, we discuss the basic intuition underlying the presented approach. In Section 1.2.2, we will describe a stability condition that is necessary for the algorithms in this chapter to succeed. In the subsequent subsections, we present two related algorithmic techniques taking advantage of the stability condition in different ways. Specifically, in Section 1.2.3, we describe an algorithm that relies on convex programming. In Section 1.2.4, we describe an iterative outlier removal technique, which has been the method of choice in practice.

1.2.1 Key Difficulties and High-Level Intuition

Arguably the most natural idea to robustly estimate the mean of a distribution would be to identify the outliers and output the empirical mean of the remaining points. The key conceptual difficulty is the fact that, in high dimensions, the outliers cannot be identified at an individual level even when they move the mean significantly. In many cases, we can easily identify the “extreme outliers” — via a pruning procedure exploiting the concentration properties of the inliers. Alas, such naive approaches typically do not suffice to obtain non-trivial error guarantees.

The simplest example illustrating this difficulty is that of a high-dimensional spherical Gaussian. Typical samples will be at ℓ_2-distance approximately $\Theta(\sqrt{d})$ from the true mean. That is, we can certainly identify as outliers all points of our dataset at distance more than $\Omega(\sqrt{d})$ from the coordinate-wise median of the dataset. All other points cannot be removed via such a procedure, as this could result in removing many inliers as well. However, by placing an ϵ-fraction of outliers at distance $\sqrt{d}$ in the same direction from the unknown mean, an adversary can corrupt the sample mean by as much as $\Omega(\epsilon\sqrt{d})$.

This leaves the algorithm designer with a dilemma of sorts. On the one hand, potential outliers at distance $\Theta(\sqrt{d})$ from the unknown mean could lead to large ℓ_2-error, scaling polynomially with d. On the other hand, if the adversary places outliers at distance approximately $\Theta(\sqrt{d})$ from the true mean in *random directions*, it may be information-theoretically impossible to distinguish them from the inliers. The way out is the realization that *it is in fact not necessary to detect and remove all outliers.* It is only required that the algorithm can detect the "consequential outliers", i.e., the ones that can significantly impact our estimates of the mean.

Let us assume without loss of generality that there no extreme outliers (as these can be removed via pre-processing). Then *the only way that the empirical mean can be far from the true mean is if there is a "conspiracy" of many outliers, all producing errors in approximately the same direction.* Intuitively, if our corrupted points are at distance $O(\sqrt{d})$ from the true mean in random directions, their contributions will on average cancel out, leading to a small error in the sample mean. In conclusion, it suffices to be able to detect these kinds of conspiracies of outliers.

The next key insight is simple and powerful. Let T be an ϵ-corrupted set of points drawn from $\mathcal{N}(\mu, I)$. If such a conspiracy of outliers substantially moves the empirical mean $\widehat{\mu}$ of T, it must move $\widehat{\mu}$ in some direction. That is, there is a unit vector v such that these outliers cause $v \cdot (\widehat{\mu} - \mu)$ to be large. For this to happen, it must be the case that these outliers are on average far from μ in the v-direction. In particular, if an ϵ-fraction of corrupted points in T move the sample average of $v \cdot (X - \mu)$, where X is the uniform distribution on T, by more than δ (δ should be thought of as small, but substantially larger than ϵ), then on average these corrupted points x must have $v \cdot (x - \mu)$ at least δ/ϵ. This in turn means that these corrupted points will have a contribution of at least $\epsilon \cdot (\delta/\epsilon)^2 = \delta^2/\epsilon$ to the variance of $v \cdot X$. Fortunately, this condition can actually be algorithmically detected! In particular, by computing the top eigenvector of the sample covariance matrix, we can efficiently determine whether or not there is any direction v for which the sample variance of $v \cdot X$ is abnormally large.

The aforementioned discussion leads us to the overall structure of the algorithms we will describe in this chapter. Starting with an ϵ-corrupted set of points T (perhaps weighted in some way), we compute the sample covariance matrix and find the eigenvector v^* with largest eigenvalue λ^*. If λ^* is not much larger than what it should be (in the absence of outliers), by the above discussion, the empirical mean is close to the true mean, and we can return that as an answer. Otherwise, we have obtained a particular direction v^* for which we know that the outliers play an unusual role, i.e., behave significantly differently than the inliers. The distribution of the points projected in the v^*-direction can then be used to perform some sort of outlier removal. The outlier removal procedure can be quite subtle and crucially depends on our distributional assumptions about the clean data.

1.2.2 Good Sets and Stability

In this section, we give a deterministic condition on the uncorrupted data that is necessary for the algorithms in this chapter to succeed (Definition 1.4). We also provide an efficiently checkable condition under which the empirical mean is certifiably close to the true mean (Lemma 1.6).

Let S be a set of n i.i.d. samples drawn from X. We will typically call these sample points good. The adversary can select up to an ϵ-fraction of points in S and replace them with arbitrary points to obtain an ϵ-corrupted set T, which is given as input to the algorithm. To establish correctness of an algorithm, we need to show that with high probability over the choice of the set S, for any choices the adversary makes, the algorithm will output an accurate estimate of the target mean.

To carry out such an analysis, it is convenient to explicitly state a collection of sufficient deterministic conditions on the set S. Specifically, we will define a notion of a "good" or "stable" set, quantified by the proportion of contamination ϵ and the distribution X. The precise stability conditions vary considerably based on the underlying estimation task and the assumptions on the distribution family of the uncorrupted data. Roughly speaking, we require that the uniform distribution over a stable set S behaves similarly to the distribution X with respect to higher moments and, potentially, tail bounds. Importantly, we require that these conditions hold even after removing an arbitrary ϵ-fraction of points in S.

The notion of a stable set must have two critical properties: (1) A set of N i.i.d. samples from X is stable with high probability, when N is at least a sufficiently large polynomial in the relevant parameters; and (2) If S is a stable set and T is obtained from S by changing at most an ϵ-fraction of the points in S, then the algorithm when run on the set T will succeed.

The robust mean estimation algorithms that will be presented in this chapter crucially rely on considering sample means and covariances. The following stability condition is an important ingredient in the success criteria of these algorithms:

Definition 1.4 (Stability Condition) Fix $0 < \epsilon < 1/2$ and $\delta \geq \epsilon$. A finite set $S \subset \mathbf{R}^d$ is (ϵ, δ)-*stable* (with respect to a distribution X) if for every unit vector $v \in \mathbf{R}^d$ and every $S' \subseteq S$ with $|S'| \geq (1-\epsilon)|S|$, the following conditions hold:

1. $\left|\frac{1}{|S'|}\sum_{x \in S'} v \cdot (x - \mu_X)\right| \leq \delta$, and
2. $\left|\frac{1}{|S'|}\sum_{x \in S'} (v \cdot (x - \mu_X))^2 - 1\right| \leq \delta^2/\epsilon$.

The aforementioned stability condition or a variant thereof is used in almost every known robust mean estimation algorithm. Definition 1.4 requires that after restricting to a $(1-\epsilon)$-density subset S', the sample mean of S' is within δ of μ_X and the sample variance of S' is $1 \pm \delta^2/\epsilon$ in every direction. The fact that these conditions must hold *for every* large subset S' of S might make it unclear if they can hold with high probability. However, it is not difficult to show the following:

Proposition 1.5 *A set of i.i.d. samples from a spherical Gaussian of size $\Omega(d/\epsilon^2)$ is $(\epsilon, O(\epsilon\sqrt{\log(1/\epsilon)})$-stable with high probability.*

We sketch a proof of Proposition 1.5. The only property required for the proof is that the distribution of the uncorrupted data has identity covariance and subgaussian tails in each direction, i.e., the tail probability of each univariate projection is bounded from above by the Gaussian tail.

Fix a direction v. To show the first condition, we note that we can maximize $\frac{1}{|S'|}\sum_{x\in S'} v\cdot(x-\mu_X)$ by removing from S the ϵ-fraction of points x for which $v\cdot x$ is smallest. Since the empirical mean of S is close to μ_X with high probability, we need to understand how much this quantity is altered by removing the ϵ-tail in the v-direction. Given our assumptions on the distribution of the uncorrupted data, removing the ϵ-tail only changes the mean by $O(\epsilon\sqrt{\log(1/\epsilon)})$. Therefore, if the empirical distribution of $v\cdot x$, $x\in S$, behaves like a spherical Gaussian in this way, the first condition is satisfied.

The second condition follows via a similar analysis. We can minimize the relevant quantity by removing the ϵ-fraction of points $x\in S$ with $|v\cdot(x-\mu_X)|$ as large as possible. If $v\cdot x$ is distributed like a unit-variance Gaussian, the total mass of its square over the ϵ-tails is $O(\epsilon\log(1/\epsilon))$. We have thus established that both conditions hold with high probability for any fixed direction. Showing that the conditions hold with high probability for all directions simultaneously can be shown by an appropriate covering argument.

More generally, one can show quantitatively different stability conditions under various distributional assumptions. In particular, if the distribution of the uncorrupted data is only assumed to have covariance matrix bounded by the identity (in the Loewner order), then it can be shown that an $\tilde{\Omega}(d/\epsilon)$ sized sample is $(\epsilon, O(\sqrt{\epsilon}))$ stable with high probability. (See Exercise 1.3 for additional examples.)

The aforementioned notion of stability is powerful and suffices for robust mean estimation. For some of the algorithms that will be presented in this chapter, a good set will be identified with a stable set; while others require the good set to satisfy additional conditions beyond stability.

The main reason why stability suffices is quantified in the following lemma:

Lemma 1.6 (Certificate for Empirical Mean) *Let S be an (ϵ,δ)-stable set with respect to a distribution X, for some $\delta\geq\epsilon>0$. Let T be an ϵ-corrupted version of S. Let μ_T and Σ_T be the empirical mean and covariance of T. If the largest eigenvalue of Σ_T is at most $1+\lambda$, then $\|\mu_T-\mu_X\|_2\leq O(\delta+\sqrt{\epsilon\lambda})$.*

Roughly speaking, Lemma 1.6 states that if we consider an ϵ-corrupted version T of any stable set S such that the empirical covariance of T has no large eigenvalues, then the empirical mean of T closely approximates the true mean. This lemma, or a variant thereof, is a key result in all known robust mean estimation algorithms.

Proof of Lemma 1.6. Let $S' = S\cap T$ and $T' = T\setminus S'$. We can assume w.l.o.g.

that $|S'| = (1-\epsilon)|S|$ and $|T'| = \epsilon|S|$. Let $\mu_{S'}, \mu_{T'}, \Sigma_{S'}, \Sigma_{T'}$ represent the empirical means and covariance matrices of S' and T'. A simple calculation gives that

$$\Sigma_T = (1-\epsilon)\Sigma_{S'} + \epsilon\Sigma_{T'} + \epsilon(1-\epsilon)(\mu_{S'} - \mu_{T'})(\mu_{S'} - \mu_{T'})^T \ .$$

Let v be the unit vector in the direction of $\mu_{S'} - \mu_{T'}$. We have that

$$\begin{aligned} 1+\lambda \geq v^T\Sigma_T v &= (1-\epsilon)v^T\Sigma_{S'}v + \epsilon v^T\Sigma_{T'}v + \epsilon(1-\epsilon)v^T(\mu_{S'} - \mu_{T'})(\mu_{S'} - \mu_{T'})^T v \\ &\geq (1-\epsilon)(1-\delta^2/\epsilon) + \epsilon(1-\epsilon)\|\mu_{S'} - \mu_{T'}\|_2^2 \\ &\geq 1 - O(\delta^2/\epsilon) + (\epsilon/2)\|\mu_{S'} - \mu_{T'}\|_2^2 \ , \end{aligned}$$

where we used the variational characterization of eigenvalues, the fact that $\Sigma_{T'}$ is positive semidefinite, and the second stability condition for S'. By rearranging, we obtain that $\|\mu_{S'} - \mu_{T'}\|_2 = O(\delta/\epsilon + \sqrt{\lambda/\epsilon})$. Therefore, we can write

$$\begin{aligned} \|\mu_T - \mu_X\|_2 &= \|(1-\epsilon)\mu_{S'} + \epsilon\mu_{T'} - \mu_X\|_2 = \|\mu_{S'} - \mu_X + \epsilon(\mu_{T'} - \mu_{S'})\|_2 \\ &\leq \|\mu_{S'} - \mu_X\|_2 + \epsilon\|\mu_{S'} - \mu_{T'}\|_2 = O(\delta) + \epsilon \cdot O(\delta/\epsilon + \sqrt{\lambda/\epsilon}) \\ &= O(\delta + \sqrt{\lambda\epsilon}) \ , \end{aligned}$$

where we used the first stability condition for S' and our bound on $\|\mu_{S'} - \mu_{T'}\|_2$. □

Lemma 1.6 says that if our input set of points T is an ϵ-corrupted version of a stable set S and has bounded covariance, the sample mean of T must be close to the true mean. Unfortunately, we are not always guaranteed that the set T we are given has this property. In order to deal with this, we will want to find a subset of T with bounded covariance and large intersection with S. However, for some of the algorithms presented, it will be convenient to find a probability distribution over T rather than a subset. For this, we will need a slight generalization of Lemma 1.6.

Lemma 1.7 *Let S be an (ϵ, δ)-stable set with respect to a distribution X, for some $\delta \geq \epsilon > 0$ with $|S| > 1/\epsilon$. Let W be a probability distribution on S that differs from U_S, the uniform distribution over S, by at most ϵ in total variation distance. Let μ_W and Σ_W be the mean and covariance of W. If the largest eigenvalue of Σ_W is at most $1+\lambda$, then $\|\mu_W - \mu_X\|_2 \leq O(\delta + \sqrt{\epsilon\lambda})$.*

Note that this subsumes Lemma 1.6 by letting W be the uniform distribution over T. The proof is essentially identical to that of Lemma 1.6, except that we need to show that the mean and variance of the conditional distribution $W \mid S$ are approximately correct, whereas in Lemma 1.6 the bounds on the mean and variance of $S \cap T$ followed directly from stability.

Lemma 1.7 clarifies the goal of our outlier removal procedure. In particular, given our initial ϵ-corrupted set T, we will attempt to find a distribution W supported on T so that Σ_W has no large eigenvalues. The weight $W(x)$, $x \in T$, quantifies our belief whether point x is an inlier or an outlier. We will also need to ensure that any such W we choose is close to the uniform distribution over S.

More concretely, we now describe a framework that captures our robust mean estimation algorithms. We start with the following definition:

Definition 1.8 Let S be a $(3\epsilon, \delta)$-stable set with respect to X and let T be an ϵ-corrupted version of S. Let $\mathcal{C}$ be the set of all probability distributions W supported on T, where $W(x) \leq \frac{1}{|T|(1-\epsilon)}$, for all $x \in T$.

We note that *any* distribution in $\mathcal{C}$ differs from U_S, the uniform distribution on S, by at most 3ϵ. Indeed, for $\epsilon \leq 1/3$, we have that:

$$\begin{aligned}
d_{\mathrm{TV}}(U_S, W) &= \sum_{x \in T} \max\{W(x) - U_S(x), 0\} \\
&= \sum_{x \in S \cap T} \max\{W(x) - 1/|T|, 0\} + \sum_{x \in T \setminus S} W(x) \\
&\leq \sum_{x \in S \cap T} \frac{\epsilon}{|T|(1-\epsilon)} + \sum_{x \in T \setminus S} \frac{1}{|T|(1-\epsilon)} \\
&\leq |T| \left(\frac{\epsilon}{|T|(1-\epsilon)} \right) + \epsilon |T| \left(\frac{1}{|T|(1-\epsilon)} \right) \\
&= \frac{2\epsilon}{1-\epsilon} \leq 3\epsilon \;.
\end{aligned}$$

Therefore, if we find $W \in \mathcal{C}$ with Σ_W having no large eigenvalues, Lemma 1.7 implies that μ_W is a good approximation to μ_X. Fortunately, we know that such a W exists! In particular, if we take W to be W^*, the uniform distribution over $S \cap T$, the largest eigenvalue is at most $1 + \delta^2/\epsilon$, and thus we achieve ℓ_2-error $O(\delta)$.

At this point, we have an *inefficient* algorithm for approximating μ_X: Find *any* $W \in \mathcal{C}$ with bounded covariance. The remaining question is how we can efficiently find one. There are two basic algorithmic techniques to achieve this, that we present in the subsequent subsections.

The first algorithmic technique we will describe is based on convex programming. We will call this *the unknown convex programming method*. Note that $\mathcal{C}$ is a convex set and that finding a point in $\mathcal{C}$ that has bounded covariance is *almost* a convex program. It is not quite a convex program, because the variance of $v \cdot W$, for fixed v, is not a convex function of W. However, one can show that given a W with variance in some direction significantly larger than $1 + \delta^2/\epsilon$, we can efficiently construct a hyperplane separating W from W^* (recall that W^* is the uniform distribution over $S \cap T$) (Section 1.2.3). This method has the advantage of naturally working under only the stability assumption. On the other hand, as it relies on the ellipsoid algorithm, it is quite slow (although polynomial time).

Our second technique, which we will call *filtering*, is an iterative outlier removal method that is typically faster, as it relies on spectral techniques. The main idea of the method is the following: If Σ_W does not have large eigenvalues, then the empirical mean is close to the true mean. Otherwise, there is some unit vector v

such that $\mathbf{Var}(v \cdot W)$ is substantially larger than it should be. This can only be the case if W assigns substantial mass to elements of $T \setminus S$ that have values of $v \cdot x$ very far from the true mean of $v \cdot \mu$. This observation allows us to perform some kind of outlier removal, in particular by removing (or down-weighting) the points x that have $v \cdot x$ inappropriately large. An important conceptual property is that one cannot afford to remove only outliers, but it is possible to ensure that more outliers are removed than inliers. Given a W where Σ_W has a large eigenvalue, one filtering step gives a new distribution $W' \in \mathcal{C}$ with $d_{\mathrm{TV}}(W', W^*) < d_{\mathrm{TV}}(W, W^*)$. Repeating the process eventually gives a W with no large eigenvalues. The filtering method and its variations are discussed in Section 1.2.4.

1.2.3 The Unknown Convex Programming Method

By Lemma 1.7, it suffices to find a distribution $W \in \mathcal{C}$ with Σ_W having no large eigenvalues. We note that this condition *almost* defines a convex program. This is because $\mathcal{C}$ is a convex set of probability distributions and the bounded covariance condition says that $\mathbf{Var}(v \cdot W) \leq 1 + \lambda$ for all unit vectors v. Unfortunately, the variance $\mathbf{Var}(v \cdot W) = \mathbf{E}[|v \cdot (W - \mu_W)|^2]$ is not quite linear in W. (If we instead had $\mathbf{E}[|v \cdot (W - \nu)|^2]$, where ν is some fixed vector, this would be linear in W.) However, we will show that finding a unit vector v for which $\mathbf{Var}(v \cdot W)$ is too large, can be used to obtain a separation oracle, i.e., a linear function on W that is violated.

Suppose that we identify a unit vector v such that $\mathbf{Var}(v \cdot W) = 1 + \lambda$, where $\lambda > c(\delta^2/\epsilon)$ for a sufficiently large universal constant $c > 0$. Applying Lemma 1.7 to the one-dimensional projection $v \cdot W$, gives $|v \cdot (\mu_W - \mu_X)| \leq O(\delta + \sqrt{\epsilon\lambda}) = O(\sqrt{\epsilon\lambda})$.

Let $L(Y) := \mathbf{E}_X[|v \cdot (Y - \mu_W)|^2]$ and note that L is a linear function of the probability distribution Y with $L(W) = 1 + \lambda$. We can write

$$\begin{aligned} L(W^*) &= \mathbf{E}_{W^*}[|v \cdot (W^* - \mu_W)|^2] = \mathbf{Var}(v \cdot W^*) + |v \cdot (\mu_W - \mu_{W^*})|^2 \\ &\leq 1 + \delta^2/\epsilon + 2|v \cdot (\mu_W - \mu_X)|^2 + 2|v \cdot (\mu_{W^*} - \mu_X)|^2 \\ &\leq 1 + O(\delta^2/\epsilon + \epsilon\lambda) < 1 + \lambda = L(W) \ . \end{aligned}$$

In summary, we have an explicit convex set $\mathcal{C}$ of probability distributions from which we want to find one with eigenvalues bounded by $1 + O(\delta^2/\epsilon)$. Given any $W \in \mathcal{C}$ which does not satisfy this condition, we can produce a linear function L that separates W from W^*. Using the ellipsoid algorithm, we obtain the following general theorem:

Theorem 1.9 *Let S be a $(3\epsilon, \delta)$-stable set with respect to a distribution X and let T be an ϵ-corrupted version of S. There exists a polynomial time algorithm which given T returns $\widehat{\mu}$ such that $\|\widehat{\mu} - \mu_X\|_2 = O(\delta)$.*

1.2.4 The Filtering Method

As in the convex programming method, the goal of the filtering method is to find a distribution $W \in \mathcal{C}$ so that Σ_W has bounded eigenvalues. Given a $W \in \mathcal{C}$, Σ_W either has bounded eigenvalues (in which case the weighted empirical mean works) or there is a direction v in which $\mathbf{Var}(v \cdot W)$ is too large. In the latter case, the projections $v \cdot W$ must behave very differently from the projections $v \cdot S$ or $v \cdot X$. In particular, since an ϵ-fraction of outliers are causing a much larger increase in the standard deviation, this means that the distribution of $v \cdot W$ will have many "extreme points" — more than one would expect to find in $v \cdot S$. This fact allows us to identity a non-empty subset of extreme points the majority of which are outliers. These points can then be removed (or down-weighted) in order to "clean up" our sample. Formally, given a $W \in \mathcal{C}$ without bounded eigenvalues, we can efficiently find a $W' \in \mathcal{C}$ so that $d_{\mathrm{TV}}(W', W^*) \leq d_{\mathrm{TV}}(W, W^*) - \gamma$, where $\gamma > 0$ is bounded from below. Iterating this procedure eventually terminates giving a W with bounded eigenvalues.

We note that while it may be conceptually useful to consider the above scheme for general distributions W over points, in most cases it suffices to consider only W given as the uniform distribution over some set T of points. The filtering step in this case consists of replacing the set T by some subset $T' = T \setminus R$, where $R \subset T$. To guarantee progress towards W^* (the uniform distribution over $S \cap T$), it suffices to ensure that at most a third of the elements of R are also in S, or equivalently that at least two thirds of the removed points are outliers (perhaps in expectation). The algorithm will terminate when the current set of points T' has bounded empirical covariance, and the output will be the empirical mean of T'.

Before we proceed with a more detailed technical discussion, we note that there are several possible ways to implement the filtering step, and that the method used has a significant impact on the analysis. In general, a filtering step removes all points that are "far" from the sample mean in a large variance direction. However, the precise way that this is quantified can vary in important ways.

Basic Filtering

In this subsection, we present a filtering method that applies to identity covariance (or, more generally, known covariance) distributions whose univariate projections satisfy appropriate concentration bounds. For the purpose of this section, we will restrict ourselves to the Gaussian setting. We note that this method immediately extends to distributions with weaker concentration properties, e.g., sub-exponential or even inverse polynomial concentration, with appropriate modifications.

We note that the filtering method presented here requires an additional condition on our good set of samples, on top of the stability condition. This is quantified in the following definition:

Definition 1.10 A set $S \subset \mathbf{R}^d$ is *tail-bound-good (with respect to $X = \mathcal{N}(\mu_X, I)$)*

if for any unit vector v, and any $t > 0$, we have

$$\Pr_{x \sim_u S}(|v \cdot (x - \mu_X)| > 2t + 2) \leq e^{-t^2/2} \ . \tag{1.1}$$

Since any projection of X is distributed like a standard Gaussian, Equation (1.1) should hold if the uniform distribution over S were replaced by X. It can be shown that this condition holds with high probability if S consists of i.i.d. random samples from X of a sufficiently large polynomial size.

Intuitively, the additional tail condition of Definition 1.10 is needed to guarantee that the filter will remove more outliers than inliers. Formally, we have the following:

Lemma 1.11 *Let $\epsilon > 0$ be a sufficiently small constant. Let $S \subset \mathbf{R}^d$ be both $(2\epsilon, \delta)$-stable and tail-bound-good with respect to $X = \mathcal{N}(\mu_X, I)$, with $\delta = c\epsilon\sqrt{\log(1/\epsilon)}$, for $c > 0$ a sufficiently large constant. Let $T \subset \mathbf{R}^d$ be such that $|T \cap S| \geq (1-\epsilon)\min(|T|, |S|)$ and assume we are given a unit vector $v \in \mathbf{R}^d$ for which $\mathbf{Var}(v \cdot T) > 1 + 2\delta^2/\epsilon$. There exists a polynomial time algorithm that returns a subset $R \subset T$ satisfying $|R \cap S| < |R|/3$.*

Proof Let $\mathbf{Var}(v \cdot T) = 1 + \lambda$. By applying Lemma 1.6 to the set T, we get that $|v \cdot \mu_X - v \cdot \mu_T| \leq c\sqrt{\lambda\epsilon}$. By (1.1), it follows that $\Pr_{x \sim_u S}(|v \cdot (x - \mu_T)| > 2t + 2 + c\sqrt{\lambda\epsilon}) \leq e^{-t^2/2}$. We claim that there exists a threshold t_0 such that

$$\Pr_{x \sim_u T}(|v \cdot (x - \mu_T)| > 2t_0 + 2 + c\sqrt{\lambda\epsilon}) > 4e^{-t_0^2/2} \ , \tag{1.2}$$

where the constants have not been optimized. Given this claim, the set $R = \{x \in T : |v \cdot (x - \mu_T)| > 2t_0 + 2 + c\sqrt{\lambda\epsilon}\}$ will satisfy the conditions of the lemma.

To prove our claim, we analyze the variance of $v \cdot T$ and note that much of the excess must be due to points in $T \setminus S$. In particular, by our assumption on the variance in the v-direction, $\sum_{x \in T} |v \cdot (x - \mu_T)|^2 = |T|\mathbf{Var}(v \cdot T) = |T|(1 + \lambda)$, where $\lambda > 2\delta^2/\epsilon$. The contribution from the points $x \in S \cap T$ is at most

$$\begin{aligned}\sum_{x \in S} |v \cdot (x - \mu_T)|^2 &= |S|(\mathbf{Var}(v \cdot S) + |v \cdot (\mu_T - \mu_S)|^2) \leq |S|(1 + \delta^2/\epsilon + 2c^2\lambda\epsilon) \\ &\leq |T|(1 + 2c^2\lambda\epsilon + 3\lambda/5) \ ,\end{aligned}$$

where the first inequality uses the stability of S, and the last uses that $|T| \geq (1-\epsilon)|S|$. If ϵ is sufficiently small relative to c, it follows that $\sum_{x \in T \setminus S} |v \cdot (x - \mu_T)|^2 \geq |T|\lambda/3$. On the other hand, by definition we have:

$$\sum_{x \in T \setminus S} |v \cdot (x - \mu_T)|^2 = |T| \int_0^\infty 2t \Pr_{x \sim_u T}(|v \cdot (x - \mu_T)| > t, x \notin S) dt. \tag{1.3}$$

Assume for the sake of contradiction that there is no t_0 for which Equation (1.2) is satisfied. Then the RHS of (1.3) is at most

$$
\begin{aligned}
&|T|\left(\int_0^{2+c\sqrt{\lambda\epsilon}+10\sqrt{\log(1/\epsilon)}} 2t \Pr_{x\sim_u T}(x\notin S) + \int_{2+c\sqrt{\lambda\epsilon}+10\sqrt{\log(1/\epsilon)}}^{\infty} 2t \Pr_{x\sim_u T}(|v\cdot(x-\mu_T)|>t)dt\right)\\
\leq&|T|\left(\epsilon(2+c\sqrt{\lambda\epsilon}+10\sqrt{\log(1/\epsilon)})^2+\int_{5\sqrt{\log(1/\epsilon)}}^{\infty}16(2t+2+c\sqrt{\lambda\epsilon})e^{-t^2/2}dt\right)\\
\leq&|T|\left(O(c^2\lambda\epsilon^2+\epsilon\log(1/\epsilon))+O(\epsilon^2(\sqrt{\log(1/\epsilon)}+c\sqrt{\lambda\epsilon}))\right)\\
\leq&|T|O(c^2\lambda\epsilon^2+(\delta^2/\epsilon)/c)<|T|\lambda/3\ ,
\end{aligned}
$$

which is a contradiction. Therefore, the tail bounds and the concentration violation together imply the existence of such a t_0 (which can be efficiently computed). □

Randomized Filtering

The basic filtering method of the previous subsection is deterministic, relying on the violation of a concentration inequality satisfied by the inliers. In some settings, deterministic filtering seems to fail and we require the filtering procedure to be randomized. A concrete such setting is when the uncorrupted distribution is only assumed to have bounded covariance.

The main idea of randomized filtering is simple: Suppose we can identify a non-negative function $f(x)$, defined on the samples x, for which (under some high probability condition on the inliers) it holds that $\sum_T f(x) \geq 2\sum_S f(x)$, where T is an ϵ-corrupted set of samples and S is the corresponding set of inliers. Then we can create a randomized filter by removing each sample point $x \in T$ with probability proportional to $f(x)$. This ensures that the *expected* number of outliers removed is at least the *expected* number of inliers removed. The analysis of such a randomized filter is slightly more subtle, so we will discuss it in the following paragraphs.

The key property the above randomized filter ensures is that the sequence of random variables (# Inliers removed) − (# Outliers removed) (where "inliers" are points in S and "outliers" points in $T\backslash S$) across iterations is a super-martingale. Since the total number of outliers removed across all iterations accounts for at most an ϵ-fraction of the total samples, this means that with probability at least $2/3$, at no point does the algorithm remove more than a 2ϵ-fraction of the inliers. A formal statement follows:

Theorem 1.12 *Let $S \subset \mathbf{R}^d$ be a $(3\epsilon,\delta)$-stable set (with respect to X). Suppose that T is an ϵ-corrupted version of S. Suppose furthermore that given any $T' \subset T$ with $|T'\cap S| \geq (1-3\epsilon)|S|$ for which $\mathbf{Cov}(T')$ has an eigenvalue bigger than $1+\lambda$, there is an efficient algorithm that computes a non-zero function $f : T' \to \mathbf{R}_+$ such that $\sum_{x\in T'} f(x) \geq 2\sum_{x\in T'\cap S} f(x)$. Then there exists a polynomial time randomized algorithm that computes a vector $\widehat{\mu}$ that with probability at least $2/3$ satisfies $\|\widehat{\mu}-\mu\|_2 = O(\delta+\sqrt{\epsilon\lambda})$.*

The algorithm is described in pseudocode below:

```
Algorithm Randomized Filtering

1. Compute $\mathbf{Cov}(T)$ and its largest eigenvalue $\nu$.
2. If $\nu \leq 1 + \lambda$, return $\mu_T$.
3. Else
    • Compute $f$ as guaranteed in the theorem statement.
    • Remove each $x \in T$ with probability $f(x)/\max_{x \in T} f(x)$ and return to
      Step 1 with the new set $T$.
```

Proof of Theorem 1.12 First, it is easy to see that this algorithm runs in polynomial time. Indeed, as the point $x \in T$ attaining the maximum value of $f(x)$ is definitely removed in each filtering iteration, each iteration reduces $|T|$ by at least one. To establish correctness, we will show that, with probability at least 2/3, at each iteration of the algorithm it holds $|S \cap T| \geq (1 - 3\epsilon)|S|$. Assuming this claim, Lemma 1.6 implies that our final error will be as desired.

To prove the desired claim, we consider the sequence of random variables $d(T) = |S \setminus T| - |T \setminus S|$ across the iterations of the algorithm. We note that, initially, $d(T) = 0$ and that $d(T)$ cannot drop below $-\epsilon|S|$. Finally, we note that at each stage of the algorithm $d(T)$ increases by (# Inliers removed) − (# Outliers removed), and that the expectation of this quantity is

$$\sum_{x \in S \setminus T} f(x) - \sum_{x \in T \setminus S} f(x) = 2 \sum_{x \in S \cap T} f(x) - \sum_{x \in T} f(x) \leq 0.$$

This means that $d(T)$ is a super-martingale (at least until we reach a point where $|S \cap T| \leq (1 - 3\epsilon)|S|$). However, if we set a stopping time at the first occasion where this condition fails, we note that the expectation of $d(T)$ is at most 0. Since it is at least $-\epsilon|S|$, this means that with probability at least 2/3 it is never more than $2\epsilon|S|$, which would imply that $|S \cap T| \geq (1 - 3\epsilon)|S|$ throughout the algorithm. This completes the proof. □

Methods of Point Removal. The randomized filtering method described above only requires that each point x is removed with probability $f(x)/\max_{x \in T} f(x)$, without any assumption of independence. Therefore, given an f, there are several ways to implement this scheme. A few natural ones are given here:

- *Randomized Thresholding:* Perhaps the easiest method for implementing our randomized filter is generating a uniform random number $y \in [0, \max_{x \in T} f(x)]$ and removing all points $x \in T$ for which $f(x) \geq y$. This method is practically useful in many applications. Finding the set of such points is often fairly easy, as this condition may well correspond to a simple threshold.

- *Independent Removal:* Each $x \in T$ is removed independently with probability $f(x)/\max_{x \in T} f(x)$. This scheme has the advantage of leading to less variance in $d(T)$. A careful analysis of the random walk involved allows one to reduce the failure probability to $\exp(-\Omega(\epsilon|S|))$.
- *Deterministic Reweighting:* Instead of removing points, this scheme allows for weighted sets of points. In particular, each point will be assigned a weight in $[0,1]$ and we will consider weighted means and covariances. Instead of removing a point with probability proportional to $f(x)$, we can remove a fraction of x's weight proportional to $f(x)$. This ensures that the appropriate weighted version of $d(T)$ is definitely non-increasing, implying correctness of the algorithm.

Universal Filtering

In this subsection, we show how to use randomized filtering to construct a universal filter that works under only the stability condition (Lemma 1.4) — not requiring the tail-bound condition of the basic filter (Lemma 1.11). Formally, we show:

Proposition 1.13 *Let $S \subset \mathbf{R}^d$ be an (ϵ, δ)-stable set for $\epsilon, \delta > 0$ sufficiently small constants and δ at least a sufficiently large multiple of ϵ. Let T be an ϵ-corrupted version of S. Suppose that $\mathbf{Cov}(T)$ has largest eigenvalue $1 + \lambda > 1 + 8\delta^2/\epsilon$. Then there exists an algorithm that, on input ϵ, δ, T, computes a function $f : T \to \mathbf{R}_+$ satisfying $\sum_{x \in T} f(x) \geq 2 \sum_{x \in T \cap S} f(x)$.*

Proof The algorithm to construct f is the following: We start by computing the sample mean μ_T and the top (unit) eigenvector v of $\mathbf{Cov}(T)$. For $x \in T$, we let $g(x) = (v \cdot (x - \mu_T))^2$. Let L be the set of $\epsilon \cdot |T|$ elements of T on which $g(x)$ is largest. We define f to be $f(x) = 0$ for $x \notin L$, and $f(x) = g(x)$ for $x \in L$.

The basic plan of attack is as follows: First, we note that the sum of $g(x)$ over $x \in T$ (which is the variance of $v \cdot Z$, $Z \sim_u T$) is substantially larger than the sum of $g(x)$ over S (which is approximately the variance of $v \cdot Z$, $Z \sim_u S$). Therefore, the sum of $g(x)$ over the $\epsilon|S|$ elements of $T \setminus S$ must be quite large. In fact, using the stability condition, we can show that the latter quantity must be larger than the sum of the largest $\epsilon|S|$ values of $g(x)$ over $x \in S$. However, since $|T \setminus S| \leq |L|$, we have that $\sum_{x \in T} f(x) = \sum_{x \in L} g(x) \geq \sum_{x \in T \setminus S} g(x) \geq 2 \sum_{x \in S} f(x)$.

We now proceed with the detailed analysis. First, note that

$$\sum_{x \in T} g(x) = |T| \mathbf{Var}(v \cdot T) = |T|(1 + \lambda) \; .$$

Moreover, for any $S' \subseteq S$ with $|S'| \geq (1 - 2\epsilon)|S|$, we have that

$$\sum_{x \in S'} g(x) = |S'|(\mathbf{Var}(v \cdot S') + (v \cdot (\mu_T - \mu'_S))^2). \tag{1.4}$$

By the stability condition, we have that $|\mathbf{Var}(v \cdot S') - 1| \leq \delta^2/\epsilon$. Furthermore, the stability condition and Lemma 1.6 give

$$\|\mu_T - \mu'_S\|_2 \leq \|\mu_T - \mu\|_2 + \|\mu - \mu'_S\|_2 = O(\delta + \sqrt{\epsilon\lambda}) \ .$$

Since $\lambda \geq 8\delta^2/\epsilon$, this implies that $\sum_{x \in T \setminus S} g(x) \geq (2/3)|S|\lambda$. Moreover, since $|L| \geq |T \setminus S|$ and since g takes its largest values on points $x \in L$, we have that

$$\sum_{x \in T} f(x) = \sum_{x \in L} g(x) \geq \sum_{x \in T \setminus S} g(x) \geq (16/3)|S|\delta^2/\epsilon \ .$$

Comparing the results of Equation (1.4) with $S' = S$ and $S' = S \setminus L$, we find that

$$\begin{aligned}\sum_{x \in S \cap T} f(x) = \sum_{x \in S \cap L} g(x) &= \sum_{x \in S} g(x) - \sum_{x \in S \setminus L} g(x) \\ &= |S|(1 \pm \delta^2/\epsilon + O(\delta^2 + \epsilon\lambda)) - |S \backslash L|(1 \pm \delta^2/\epsilon + O(\delta^2 + \epsilon\lambda)) \\ &\leq 2|S|\delta^2/\epsilon + |S|O(\delta^2 + \epsilon\lambda).\end{aligned}$$

The latter quantity is at most $(1/2)\sum_{x \in T} f(x)$ when δ and ϵ/δ are sufficiently small constants. This completes the proof of Proposition 1.13. □

Practical Considerations. While the aforementioned point removal methods have similar theoretical guarantees, recent implementations (Diakonikolas et al., 2018c) suggest that they have different practical performance on real datasets. The deterministic reweighting method is somewhat slower in practice as its worst-case runtime and its typical runtime are comparable. In more detail, one can guarantee termination by setting the constant of proportionality so that at each step at least one of the non-zero weights is set to zero. However, in practical circumstances, we will not be able to do better. That is, the algorithm may well be forced to undergo $\epsilon|S|$ iterations. On the other hand, the randomized versions of the algorithm are likely to remove several points of T at each filtering step.

Another reason why the randomized versions may be preferable has to do with the quality of the results. The randomized algorithms only produce bad results when there is a chance that $d(T)$ ends up being very large. However, since $d(T)$ is a supermartingale, this will only ever be the case if there is a corresponding possibility that $d(T)$ will be exceptionally small. Thus, although the randomized algorithms may have a probability of giving worse results some of the time, this will only happen if a corresponding fraction of the time, they also give *better* results than the theory guarantees. This consideration suggests that the randomized thresholding procedure might have advantages over the independent removal procedure precisely because it has a higher probability of failure. This has been observed experimentally in (Diakonikolas et al., 2018c): In real datasets (poisoned with a constant fraction of adversarial outliers), the number of iterations of randomized filtering is typically bounded by a small constant.

1.3 Beyond Robust Mean Estimation

In this section, we provide a brief overview of the ideas behind recently developed robust estimators for more general statistical tasks.

Robust Stochastic Optimization. A simple and powerful idea is that efficient algorithms for robust mean estimation can be used in essentially a black-box manner to obtain robust learners for a range of stochastic optimization problems. Consider the following general stochastic optimization problem: There is some unknown true distribution p^* over (convex) functions $f : \mathcal{W} \to \mathbf{R}$, and the goal is to find an approximate minimizer of $F(w) = \mathbf{E}_{f \sim p^*}[f(w)]$. Here $\mathcal{W} \subseteq \mathbf{R}^d$ is a space of possible parameters. As an example, the problem of linear regression fits in this framework for $f(w) = (1/2)(w \cdot x - y^2)$ and $(x, y) \in \mathbf{R}^d \times \mathbf{R}$ is drawn from the data distribution.

On input a set of clean samples, i.e., i.i.d. set of functions $f_1, \ldots, f_n \sim p^*$, this problem can be efficiently solved by (stochastic) gradient descent. In the robust setting, we have access to an ϵ-corrupted training set of functions $f_1, \ldots, f_n$ drawn from p^*. Unfortunately, even a single corrupted sample can completely compromise standard gradient descent. Charikar et al. (2017) first studied the robust version of this problem in the presence of a majority of outliers. The vanilla outlier-robust setting, where $\epsilon < 1/2$, was studied in two concurrent works (Prasad et al., 2018; Diakonikolas et al., 2018c). The main intuition present in both these works is that robustly estimating the gradient of the objective function can be viewed as a robust mean estimation problem. Diakonikolas et al. (2018c) take this connection a step further: Instead of using a robust gradient estimator as a black-box, they apply a filtering step each time the vanilla SGD reaches an approximate critical point of the empirical risk. The correctness of this method relies on properties of the filtering algorithm. Importantly, it turns out that this method is more efficient in practice.

Robust Covariance Estimation. The robust estimation techniques described in this chapter can be generalized to robustly estimate the covariance of high-dimensional distributions. For concreteness, here we consider the Gaussian case, specifically we assume that the inliers are drawn from $G = \mathcal{N}(0, \Sigma)$. (Note that by considering the differences of independent samples we can reduce to the centered case, and that this reduction works in the robust setting as well.) The high-level idea is to filter based on the empirical fourth moment tensor. In more detail, let X be the random variable GG^T and note that $\mathbf{Cov}(G) = \mathbf{E}[X]$.

We can attempt to use the described robust mean estimation techniques on X. However, these techniques require a priori bounds on its covariance, $\mathbf{Cov}(X)$. To handle this issue, we leverage the fact that the covariance of X can be expressed as a function of the covariance of G. Although it might appear that we run into a chicken-and-egg problem, it is in fact possible to bootstrap better and better approximations to the covariance $\mathbf{Cov}(X)$.

In particular, any upper bound on the covariance of G will imply an upper bound on the covariance of X, which can in turn be used to robustly estimate the mean of X, providing a better estimate of $\mathbf{Cov}(G)$. Via a careful iterative refinement, one can show that is possible to learn the covariance $\mathbf{Cov}(G)$ within relative error $O(\epsilon \log(1/\epsilon))$ with respect to the Frobenius norm, which corresponds to robustly estimating G within error $O(\epsilon \log(1/\epsilon))$ in total variation distance.

List-Decodable Learning. In this chapter, we focused on the classical robust setting where the outliers constitute the minority of the dataset, quantified by the fraction of corruptions $\epsilon < 1/2$, and the goal is to obtain estimators with error scaling as a function of ϵ (and is independent of the dimension d). A related setting of interest focuses on the regime when the fraction α of real data is small – strictly smaller than 1/2. That is, we observe n samples, an α-fraction of which (for some $\alpha < 1/2$) are drawn from the distribution in question, but the rest are arbitrary.

This model was first studied in the context of mean estimation in Charikar et al. (2017). A first observation is that, in this regime, it is information-theoretically impossible to estimate the mean with a single hypothesis. Indeed, an adversary can produce $\Omega(1/\alpha)$ clusters of points each drawn from a good distribution with different mean. Even if the algorithm could learn the distribution of the samples exactly, it still would not be able to identify which of the clusters is the correct one. To circumvent this, the definition of learning must be somewhat relaxed. In particular, the algorithm should be allowed to return a small list of hypotheses with the guarantee that *at least one* of the hypotheses is close to the true mean. Moreover, as opposed to the small ϵ regime, it is often information-theoretically necessary for the error to increase as α goes to 0. In summary, given polynomially many samples, we would like to output $O(1/\alpha)$ many hypotheses, with the guarantee that with high probability at least one hypothesis is within $f(\alpha)$ of the true mean, where $f(\alpha)$ depends on the concentration properties of the distribution in question.

Charikar et al. (2017) used an SDP-based approach to solve this problem. We note that the techniques discussed in this chapter can be adapted to work in this setting. In particular, if the sample covariance matrix has no large eigenvalues, this certifies that the true mean and sample mean are not too far apart. However, if a large eigenvalue exists, the construction of a filter is more elaborate. To some extent, this is a necessary difficulty because the algorithm must return more than one hypotheses. To handle this issue, one needs to construct a *multi-filter*, which may return several subsets of the original sample set with the guarantee that at least one of them is cleaner than the original dataset. Such a multi-filter was introduced in Diakonikolas et al. (2018a).

Robust Sparse Estimation. The task of leveraging sparsity in high-dimensional parameter estimation is a well-studied problem in statistics. In the context of robust estimation, this problem was first considered in Balakrishnan et al. (2017), which adapted the unknown convex programming method of Diakonikolas et al. (2016) described in this chapter. Here we describe the filtering method in this setting for the problem of robust sparse mean estimation.

Formally, given ϵ-corrupted samples from $\mathcal{N}(\mu, I)$, where the mean μ is unknown and assumed to be k-sparse, i.e., supported on an unknown set of k coordinates, we would like to approximate μ, in ℓ_2-distance. Without corruptions, this problem is easy: We draw $O(k \log(d/k)/\epsilon^2)$ samples and output the empirical mean truncated in its largest magnitude k entries. The goal is to obtain similar sample complexity and error guarantees in the robust setting.

At a high level, we note that the truncated sample mean should be accurate as long as there is no k-sparse direction in which the error between the true mean and sample mean is large. This condition can be certified, as long as we know that the sample variance of $v \cdot X$ is close to 1 for all unit, k-sparse vectors v. This would in turn allow us to create a filter-based algorithm for k-sparse robust mean estimation that uses only $O(k \log(d/k)/\epsilon^2)$ samples. Unfortunately, the problem of determining whether or not there is a k-sparse direction with large variance is computationally hard. By considering a convex relaxation of this problem, one can obtain a polynomial time version of this algorithm that requires $O(k^2 \log(d/k)/\epsilon^2)$ samples. Moreover, there is evidence (Diakonikolas et al., 2017b), in the form of a lower bound in the Statistical Query model (a restricted but powerful computational model), that this increase in the sample complexity is necessary.

More recently, Diakonikolas et al. (2019) developed iterative spectral algorithms for robust sparse estimation tasks (including sparse mean estimation and sparse PCA). These algorithms achieve the same error guarantees as Balakrishnan et al. (2017), while being significantly faster.

Robust Estimation of High-Degree Moments. Suppose we are interested in robustly estimating the k-th order moments of a distribution X. In some sense, this problem is equivalent to estimating the mean of the random variable $Y = X^{\otimes k}$. Unfortunately, in order to estimate the mean of Y robustly, one needs concentration bounds on it, which are rarely directly available. Typically, concentration bounds on Y are implied by upper bounds on the higher moments of X. In particular, upper bounds on the k'-th central moments of X for some $k' > k$, imply concentration bounds on Y. Unfortunately, just knowing bounds on the central moments of X is often hard to leverage computationally. Given a set of points, even *determining* whether or not they have bounded central moments is a computationally intractable problem. Instead, known algorithmic approaches (Hopkins and Li, 2018; Kothari et al., 2018) generally require some kind of efficiently certifiable bounded moment conditions (e.g., via a sum of squares proof). This allows one to search for subsets of sample points whose central moments can be similarly certified as bounded, and these will allow us to approximate higher moments of X.

1.4 Notes

The convex programming and filtering methods described in this chapter appeared in (Diakonikolas et al., 2016, 2017a). The idea of removing outliers by projecting on the top eigenvector of the empirical covariance goes back to Klivans et al. (2009), who used it in the context of robustly learning linear separators. Klivans et al. (2009) use a "hard" filtering step which only removes outliers and consequently leads to errors that scale logarithmically with the dimension, even in Huber's model.

The work of Lai et al. (2016) developed a recursive dimension-halving technique for robust mean estimation. Their technique leads to error $O(\epsilon\sqrt{\log(1/\epsilon)}\sqrt{\log d})$ for Gaussian robust mean estimation in Huber's contamination model. Diakonikolas et al. (2016) and Lai et al. (2016) obtained robust estimators for various other statistical tasks, including robust covariance estimation, robust density estimation for mixtures of spherical Gaussians and product distributions, and independent component analysis.

The algorithmic approaches described in this chapter robustly estimate the mean of a spherical Gaussian within error $O(\epsilon\sqrt{\log(1/\epsilon)})$ in the strong contamination model of Definition 1.1. Diakonikolas et al. (2018b) developed a more sophisticated filtering technique that achieves the optimal error of $O(\epsilon)$ in the additive contamination model. For the strong contamination model, it was shown in Diakonikolas et al. (2017b) that any improvement on the $O(\epsilon\sqrt{\log(1/\epsilon)})$ error requires super-polynomial time in the Statistical Query model. Steinhardt et al. (2018) gave an efficient algorithm for robust mean estimation with respect to all ℓ_p-norms.

Finally, we note that ideas from Diakonikolas et al. (2016) have led to proof-of-concept improvements in the analysis of genetic data (Diakonikolas et al., 2017a) and in adversarial machine learning (Diakonikolas et al., 2018c).

References

Balakrishnan, S., Du, S. S., Li, J., and Singh, A. 2017. Computationally Efficient Robust Sparse Estimation in High Dimensions. Pages 169–212 of: *Proc. 30th Annual Conference on Learning Theory.*

Charikar, M., Steinhardt, J., and Valiant, G. 2017. Learning from untrusted data. Pages 47–60 of: *Proc. 49th Annual ACM Symposium on Theory of Computing.*

Diakonikolas, I., Kamath, G., Kane, D. M., Li, J., Moitra, A., and Stewart, A. 2016. Robust Estimators in High Dimensions without the Computational Intractability. Pages 655–664 of: *Proc. 57th IEEE Symposium on Foundations of Computer Science (FOCS).*

Diakonikolas, I., Kamath, G., Kane, D. M., Li, J., Moitra, A., and Stewart, A. 2017a. Being Robust (in High Dimensions) Can Be Practical. Pages 999–1008 of: *Proc. 34th International Conference on Machine Learning (ICML).*

Diakonikolas, I., Kane, D. M., and Stewart, A. 2017b. Statistical Query Lower Bounds for Robust Estimation of High-Dimensional Gaussians and Gaussian

Mixtures. Pages 73–84 of: *Proc. 58th IEEE Symposium on Foundations of Computer Science (FOCS).*

Diakonikolas, I., Kane, D. M., and Stewart, A. 2018a. List-Decodable Robust Mean Estimation and Learning Mixtures of Spherical Gaussians. Pages 1047–1060 of: *Proc. 50th Annual ACM Symposium on Theory of Computing (STOC).*

Diakonikolas, I., Kamath, G., Kane, D. M., Li, J., Moitra, A., and Stewart, A. 2018b. Robustly Learning a Gaussian: Getting Optimal Error, Efficiently. Pages 2683–2702 of: *Proc. 29th Annual Symposium on Discrete Algorithms.*

Diakonikolas, I., Kamath, G., Kane, D. M., Li, J., Steinhardt, J., and Stewart, A. 2018c. Sever: A Robust Meta-Algorithm for Stochastic Optimization. *CoRR*, **abs/1803.02815**. Conference version in ICML 2019.

Diakonikolas, I., Karmalkar, S., Kane, D., Price, E., and Stewart, A. 2019. Outlier-Robust High-Dimensional Sparse Estimation via Iterative Filtering. In: *Advances in Neural Information Processing Systems 33, NeurIPS 2019.*

Hopkins, S. B., and Li, J. 2018. Mixture models, robustness, and sum of squares proofs. Pages 1021–1034 of: *Proc. 50th Annual ACM Symposium on Theory of Computing (STOC).*

Huber, P. J. 1964. Robust Estimation of a Location Parameter. *Ann. Math. Statist.*, **35**(1), 73–101.

Johnson, D. S., and Preparata, F. P. 1978. The densest hemisphere problem. *Theoretical Computer Science*, **6**, 93–107.

Klivans, A., Long, P., and Servedio, R. 2009. Learning Halfspaces with Malicious Noise. *Journal of Machine Learning Research*, **10**, 2715–2740.

Kothari, P. K., Steinhardt, J., and Steurer, D. 2018. Robust moment estimation and improved clustering via sum of squares. Pages 1035–1046 of: *Proc. 50th Annual ACM Symposium on Theory of Computing (STOC).*

Lai, K. A., Rao, A. B., and Vempala, S. 2016. Agnostic Estimation of Mean and Covariance. Pages 665–674 of: *Proc. 57th IEEE Symposium on Foundations of Computer Science (FOCS).*

Prasad, A., Suggala, A. S., Balakrishnan, S., and Ravikumar, P. 2018. Robust Estimation via Robust Gradient Estimation. *arXiv preprint arXiv:1802.06485.*

Steinhardt, J., Charikar, M., and Valiant, G. 2018. Resilience: A Criterion for Learning in the Presence of Arbitrary Outliers. Pages 45:1–45:21 of: *Proc. 9th Innovations in Theoretical Computer Science Conference (ITCS).*

Tukey, J. W. 1975. Mathematics and picturing of data. Pages 523–531 of: *Proceedings of ICM*, vol. 6.

Exercises

1.1 Let S be an ϵ-corrupted set of samples from $\mathcal{N}(\mu, I)$ of sufficiently large size.

(a) Show that the geometric median of S has ℓ_2-distance $O(\epsilon\sqrt{d})$ from μ with high probability.

(b) Show that this upper bound is tight for a worst-case adversary.

1.2 (Sample complexity of Robust Mean Estimation)

(a) Prove Fact 1.2 and Proposition 1.3.

(b) How do Fact 1.2 and Proposition 1.3 change when the distribution of the uncorrupted data has bounded k-th moments, for even k?

1.3 For what values of (ϵ, δ) do the following distribution families satisfy the stability condition of Definition 1.4: bounded covariance ($\Sigma \preceq I$), bounded covariance and sub-gaussian tails in every direction, identity covariance and log-concave (i.e., the logarithm of probability density function is concave), identity covariance with bounded k-th central moments?

1.4 Prove Lemma 1.7.

1.5 (Diakonikolas et al. (2016)) Let S be a sufficiently large ϵ-corrupted set of samples from a binary product distribution on $\{\pm 1\}^d$. Modify the basic filter algorithm of Section 1.2.4 to obtain an estimate of the mean with ℓ_2-distance error $O(\epsilon\sqrt{\log(1/\epsilon)})$. [Hint: Use the modified empirical covariance with its diagonal zeroed out.]

1.6 (Robust Estimation of Heavy-Tailed Distributions) Let X be a product distribution on $\mathbf{R}^d$ that is centrally symmetric about a center m. Suppose that, for some constant $c > 0$, each marginal distribution has probability density function bounded below by c at all x within distance one of its median. Give a polynomial-time algorithm that estimates m to within ℓ_2 error $\tilde{O}(\epsilon)$ in the presence of an ϵ-fraction of corruptions. (The $\tilde{O}(\cdot)$ notation hides polylogarithmic factors in its argument.)

Remark: This algorithm applies to distributions that may not even have well-defined means, e.g., products of Cauchy distributions.)

[*Hint*: Reduce the problem to robust mean estimation of a binary product distribution and use the previous exercise.]

1.7 (Robust Estimation of a 2-Mixture of Spherical Gaussians) In this exercise, we will adapt the filtering method to robustly learn a 2-mixture of spherical Gaussians. Let $F = (1/2)\mathcal{N}(\mu_1, I) + (1/2)\mathcal{N}(\mu_2, I)$ be an unknown balanced mixture of two identity covariance Gaussians with unknown means. Let T be an ϵ-corrupted set of samples from F.

(a) Show that if the eigenvalue of the empirical covariance in a given direction is $1 + \delta$, then both means in this direction are accurate within $\tilde{O}(\sqrt{\epsilon + \delta})$.

(b) Show that if the empirical covariance has only one large eigenvalue, then there is a simple procedure to learn the means to small error.

(c) Show that if empirical covariance has at least two large eigenvalues, then we can construct a filter.

(d) Combine the above to give a polynomial-time algorithm that with high probability learns the means to error $\tilde{O}(\sqrt{\epsilon})$.

(Remark: This accuracy is essentially best possible information-theoretically. One can have have two mixtures $F^{(i)} = (1/2)\mathcal{N}(\mu_1^{(i)}, I) + (1/2)\mathcal{N}(\mu_2^{(i)}, I)$, $i = 1, 2$ that have $d_{\mathrm{TV}}(F^{(1)}, F^{(2)}) = \epsilon$, where $\mu_1^{(2)}$, $\mu_2^{(2)}$ are at distance $\Omega(\sqrt{\epsilon})$ from $\mu_1^{(1)}$, $\mu_2^{(1)}$.)