Robustly Learning a Gaussian: Getting Optimal Error, Efficiently

Robustly Learning a Gaussian: Getting Optimal Error, Efficiently

Ilias Diakonikolas* Gautam Kamath† Daniel M. Kane‡ Jerry Li §
Ankur Moitra¶ Alistair Stewart‖

Abstract

We study the fundamental problem of learning the parameters of a high-dimensional Gaussian in the presence of noise — where an ε-fraction of our samples were chosen by an adversary. We give robust estimators that achieve estimation error $O(\varepsilon)$ in the total variation distance, which is optimal up to a universal constant that is independent of the dimension.

In the case where just the mean is unknown, our robustness guarantee is optimal up to a factor of $\sqrt{2}$ and the running time is polynomial in d and $1/\varepsilon$. When both the mean and covariance are unknown, the running time is polynomial in d and quasipolynomial in $1/\varepsilon$. Moreover all of our algorithms require only a polynomial number of samples. Our work shows that the same sorts of error guarantees that were established over fifty years ago in the one-dimensional setting can also be achieved by efficient algorithms in high-dimensional settings.

1 Introduction

1.1 Background The most popular and widely used modeling assumption is that data is approximately Gaussian. This is a convenient simplification to make when modeling velocities of particles in an ideal gas [Goo15], measuring physical characteristics across a population (after controlling for gender), and even modeling fluctuations in a stock price on a logarithmic scale. However, real data is not actually Gaussian and is at best crudely approximated by a Gaussian (e.g., with heavier tails). What's worse is that estimators designed under this assumption can perform poorly in practice and be heavily biased by just a few errant samples that do not fit the model.

For over fifty years, the field of robust statistics [HR09, HRRS86, RL05] has studied exactly this phenomenon — the sensitivity or insensitivity of estimators to small deviations in the model. Unsurprisingly, one of the central questions that shaped its development was the problem of learning the parameters of a one-dimensional Gaussian distribution when a small fraction of the samples are arbitrarily corrupted. More precisely, in 1964, Huber [Hub64] introduced the following model:

Definition 1. *In* Huber's contamination model*, we are given samples from a distribution*

$$\mathcal{D} = (1-\varepsilon)\mathcal{N}(\mu, \sigma^2) + \varepsilon\mathcal{Z} \ ,$$

where $\mathcal{N}(\mu, \sigma^2)$ is a Gaussian of mean μ and variance σ^2, and $\mathcal{Z}$ is an arbitrary distribution chosen by an adversary.

Intuitively, among our samples, about a $(1-\varepsilon)$ fraction will have been generated from a Gaussian and are called *inliers*, and the rest are called *outliers* or *gross corruptions*. We will work with an even more challenging[1] model — called the *strong contamination model* (Definition 2) — where the adversary is allowed to look at the inliers and then decide on the outliers. The literature on robust statistics has given numerous explanations and empirical investigations [GCSR14, Ham01] into how such outliers might arise as the result of equipment failure, data being entered incorrectly, or even from a subpopulation that was not accounted for in a medical study. These types of errors are erratic and difficult to model, so instead our goal is to design a procedure that accurately estimates μ and σ^2 without making any assumptions about them.

*Supported by NSF CAREER Award CCF-1652862, a Sloan Research Fellowship, and a Google Faculty Research Award.

†Supported by NSF CCF-1551875, CCF-1617730, CCF-1650733, and ONR N00014-12-1-0999.

‡Supported by NSF CAREER Award CCF-1553288 and a Sloan Research Fellowship.

§Supported by NSF CAREER Award CCF-1453261, CCF-1565235, a Google Faculty Research Award, and an NSF Graduate Research Fellowship.

¶Supported by NSF CAREER Award CCF-1453261, CCF-1565235, a Packard Fellowship, a Sloan Research Fellowship, a grant from the MIT NEC Corporation, and a Google Faculty Research Award.

‖Supported by a USC startup grant.

[1] None of the results in our paper were previously known in Huber's contamination model either. The reason we work with this stronger model is because we can — nothing in our analysis relies on the inliers and outliers being independent.

In one dimension, the median and median absolute deviation are well-known robust estimators for the mean and variance respectively. In particular, given samples $X_1, X_2, \ldots, X_n$, we can compute

$$\widehat{\mu} = \text{median}(X_1, X_2, \ldots, X_n)$$

and

$$\widehat{\sigma} = \frac{\text{median}(|X_i - \widehat{\mu}|)}{\Phi^{-1}(3/4)},$$

where Φ is the cumulative distribution of the standard Gaussian. (This scaling constant is needed to ensure that $\widehat{\sigma}$ is an unbiased estimator when there is no noise.) If $n \geq C\frac{\log 1/\delta}{\varepsilon^2}$, then with probability at least $1-\delta$ we have that $d_{TV}(\mathcal{N}(\mu,\sigma^2),\mathcal{N}(\widehat{\mu},\widehat{\sigma}^2)) \leq C\varepsilon$. In Huber's contamination model, this is the strongest type of error guarantee we could hope for[2] and captures both the task of learning the underlying parameters μ and σ^2, and finding the approximately best fit to the observed distribution within the family of one-dimensional Gaussians. In fact there are plentifully many other estimators — such as the *trimmed mean*, *winsorized mean*, *Tukey's biweight function*, and the *interquartile range* — that achieve the same sorts of error guarantees, up to constant factors. The design of robust estimators for *location* (e.g., estimating μ) and *scale* (e.g., estimating σ^2) is guided by certain overarching principles, such as the notion of the influence curve [HRRS86] or the notion of breakdown point [RL05]. In some cases, it is even possible to design robust estimators that are minimax optimal [Hub64].

These days, much of modern data analysis revolves around high-dimensional data — for example, when we model documents [BNJ03], images [OF96], and genomes [NJB+08] as vectors in a very high-dimensional space. The need for robust estimators is even more pressing in these applications, since it is infeasible to remove obvious outliers by inspection. However, adapting robust statistics to high-dimensional settings is fraught with challenges. The principles that guided the design of robust estimators in one dimension seem to inherently lead to high-dimensional estimators that are hard to compute [Ber06, HM13].

In this paper, we focus on the central problem of learning the parameters of a multivariate Gaussian $\mathcal{N}(\mu,\Sigma)$ in the strong contamination model. The textbook estimators for the mean and covariance – such as the *Tukey median* [Tuk75] and *minimum volume enclosing ellipsoid* [Rou85] – essentially search for directions where the projection of $\mathcal{D}$ is suitably non-Gaussian. However, trying to find a direction where the projection is non-Gaussian can be like looking for a needle in an exponentially-large haystack – these statistics are not efficiently computable, in general. Furthermore, a random projection will look Gaussian with high probability [Kla07].

In this paper, our main result is an efficiently computable estimator for a high-dimensional Gaussian that achieves error

$$d_{TV}(\mathcal{N}(\mu,\Sigma),\mathcal{N}(\widehat{\mu},\widehat{\Sigma})) \leq C\varepsilon$$

in the strong contamination model, for a universal constant C that is independent of the dimension. For a Gaussian distribution, we consider estimation in terms of total variation distance, which is equivalent to estimating the parameters under the natural measures. Our main idea is to use various regularity conditions satisfied by the inliers to make the problem of searching for non-Gaussian projections easier. When just the mean μ is unknown, our algorithm runs in time polynomial in the dimension d and $1/\varepsilon$. When both the mean and covariance are unknown, our algorithm runs in time polynomial in d and quasi-polynomial in $1/\varepsilon$. All of our algorithms achieve polynomial sample complexity.

Prior to our work, the best known algorithm of Diakonikolas et al. [DKK+16] achieved estimation error $O(\varepsilon \log 1/\varepsilon)$ for this problem[3], again with respect to total variation distance. Concurrently, Lai, Rao and Vempala [LRV16] gave an algorithm which achieves estimation error roughly $O(\varepsilon^{1/2}\log^{1/2} d)$. In fact, the algorithm of Diakonikolas et al. [DKK+16] works in a stronger model than what we consider here, where an adversary gets to look at the samples and then decides on an ε-fraction to move arbitrarily. Such errors are both additive and subtractive (because inliers are removed). Interestingly, Diakonikolas, Kane and Stewart [DKS17] proved that any Statistical Query learning algorithm that works in such an additive and subtractive model and achieves an error guarantee asymptotically better than $O(\varepsilon \log^{1/2} 1/\varepsilon)$ must make a super-polynomial number of statistical queries. Our work shows a natural conclusion that in an additive only model it is possible to algorithmically achieve the same error guarantees as are possible in the one-dimensional case, up to a universal constant.

1.2 Our Results and Techniques In what follows, we will explain both our work as well as prior

[2] See lower bounds in the appendix of the full version.

[3] We note that, as stated, the results in [DKK+16] give estimation error $O(\varepsilon \log^{3/2} 1/\varepsilon)$. However, combining the techniques in [DKK+16] with the arguments in Section 7 of this paper gives the stated bound. This argument will be included in the full version of [DKK+16].

work through the following lens:

> *At the core of any robust estimator is some procedure to certify that the estimates have not been moved too far away from the true parameters by a small number of corruptions.*

First, we consider the subproblem where the covariance $\Sigma = I$ is known and only the mean μ is unknown. In the terminology of robust statistics, this is called *robust estimation of location*. If we could compute the Tukey median, we would have an estimate that satisfies $d_{TV}(\mathcal{N}(\mu, I), \mathcal{N}(\widehat{\mu}, I)) \leq C\varepsilon$. The way that the Tukey median guarantees that it is close to the true mean is that along every direction u it is close to the median of the projection of the samples. More precisely, at least a $\frac{1-\varepsilon}{2}$ fraction of the samples satisfy $u^T X_i \geq u^T \widehat{\mu}$, and at least a $\frac{1-\varepsilon}{2}$ fraction of the samples satisfy $u^T \widehat{\mu} \geq u^T X_i$. However, if we have a candidate $\widehat{\mu}$, finding a direction u that violates this condition is again like searching for a needle in an exponentially large haystack.

The approach of Diakonikolas et al. [DKK+16] was essentially a data-dependent way to search for appropriate directions u, by looking for directions where the empirical variance is larger than it should be (if there were no corruptions). However, because their approach considers only a single direction at a time, it naturally gets stuck at error $\Theta(\varepsilon \log^{1/2} 1/\varepsilon)$. This is because along the direction u, only when a point is $\Omega(\log^{1/2} 1/\varepsilon)$ away from most of the rest of the samples can we be relatively confident that it is an outlier. Thus, an adversary could safely place all the corruptions in the tails and move the mean by as much as $\Theta(\varepsilon \log^{1/2} 1/\varepsilon)$. This would not affect the Tukey median by as much, but would affect an estimate based on the empirical mean (because the algorithm could find no other outliers to remove) by considerably more.

Our approach is to consider logarithmically many directions at once. Even though an inlier can be logarithmically many standard deviations away from the mean along a single direction u with reasonable probability, it is unlikely to be that many standard deviations away simultaneously across many orthogonal directions. Essentially, this allows us to remove the influence of outliers on all but a logarithmic dimensional subspace. Combining this with an algorithm for robustly learning the mean in time exponential in the dimension (but polynomial in the number of samples), we obtain our first main result:

THEOREM 1.1. *Suppose we are given a set of* $n = \mathrm{poly}(d, 1/\varepsilon)$ *samples from the strong contamination model, where the underlying* d*-dimensional Gaussian is* $\mathcal{N}(\mu, I)$*. Let* $\varepsilon \leq \varepsilon_0$*, where* ε_0 *is a positive universal constant. For any* $\beta > 0$*, there is an algorithm to learn an estimate* $\mathcal{N}(\widehat{\mu}, I)$ *that with high probability satisfies*

$$d_{TV}(\mathcal{N}(\mu, I), \mathcal{N}(\widehat{\mu}, I)) \leq \left(\frac{1}{\sqrt{2}} + O\left(\frac{1}{\sqrt{\beta}} + \varepsilon^2\right)\right)\varepsilon\,.$$

Moreover, the algorithm runs in time $\mathrm{poly}(n, (1/\varepsilon)^{\beta})$.

We prove an almost matching lower bound of $\frac{\varepsilon}{2} + \Omega(\varepsilon^2)$ on the estimation error. Thus, our robustness guarantee is optimal up to a factor of $\sqrt{2}$, even among computationally inefficient robust estimators. Interestingly, our extra factor of $\sqrt{2}$ comes from the following geometric fact which we make crucial use of: Any convex body of diameter D in any dimension can be covered by a ball of radius $D/\sqrt{2}$, and moreover such a ball can be (approximately) found in time exponential in the dimension. Suppose that along some direction u we have an estimate p that is guaranteed to be within $\varepsilon/2$ of the projection of the true mean μ. We can now confine μ to a slab of width ε, and by taking the intersection of all such slabs we get a convex body that contains μ and has diameter of at most ε. By covering the body with a ball of radius $\varepsilon/\sqrt{2}$, we are guaranteed that the center of the ball is within $\varepsilon/\sqrt{2}$ of the true mean. This gives us a general way to combine one-dimensional robust estimates along a net of directions.

We note that, for general isotropic sub-Gaussian distributions, the bound of $O(\varepsilon \log^{1/2} 1/\varepsilon)$ of [DKK+17] is optimal for robust mean estimation, even in one dimension. See the full version of this paper for a proof of this fact. However, our results can be seen to hold more generally than stated above – indeed, the same arguments work for a class of symmetric isotropic sub-Gaussian distributions which are sufficiently smooth near their mean. More precisely, we require that along any univariate projection, the mean is robustly estimated by the median.

We next consider the subproblem where the mean $\mu = 0$ is known and only the covariance Σ is unknown. In the terminology of robust statistics, this is called *robust estimation of scale*. In this case, we want to compute an estimate $\widehat{\Sigma}$ that satisfies[4] $\|\Sigma - \widehat{\Sigma}\|_F \leq C\varepsilon$. When $\widehat{\Sigma}$ does not satisfy this condition, it can be

[4] More precisely, to obtain $O(\varepsilon)$ error guarantee with respect to the total variation distance, we need to robustly approximate Σ within $O(\varepsilon)$ in Mahalanobis distance, which is a stronger metric than the Frobenius norm. As part of our approach, we are able to efficiently reduce to the case that Σ is close to the identity matrix, in which case the Frobenius error suffices.

shown (in Section 6.2.3) that there is a degree-two polynomial $p(X)$, where

$$\mathop{\mathbf{E}}_{X\sim\mathcal{N}(0,\Sigma)}[p(X)] = 1 \text{ and } \mathop{\mathbf{E}}_{X\sim\mathcal{N}(0,\widehat{\Sigma})}[p(X)] = 1 + C'\varepsilon \ .$$

It turns out that, even given the polynomial $p(X)$, deciding whether or not the above conditions approximately hold is challenging. Given $p(X)$ and $\widehat{\Sigma}$, we can certainly compute $\mathbf{E}_{X\sim\mathcal{N}(0,\widehat{\Sigma})}[p(X)]$. But given only contaminated samples from $\mathcal{N}(0,\Sigma)$ and without knowing what Σ is, can we estimate $\mathbf{E}_{X\sim\mathcal{N}(0,\Sigma)}[p(X)]$?

Often, univariate robust estimation problems are considered easy, with a simple recipe: Construct an unbiased estimator for the statistic for which each sample point has low influence. However, in our setting, it is highly non-trivial to construct such an estimator. The naive attempt in this case would be the median – this immediately fails since the distribution of $p(X)$ is asymmetric. Even if there were no noise, that would not necessarily be an unbiased estimator. So how can we dampen the influence of outliers, if there is no natural symmetry in the distribution? We construct a robust estimator crucially using the fact that $p(X)$ is the weighted sum of chi-squared random variables when there is no noise. The key structural fact we exploit is the following: Given two sums of chi-squared random variables, if the random variables are far in total variation distance, most of their difference must lie close to their means. We use this fact to show how, given a weak estimate of the mean (i.e., one which is only accurate up $\omega(\varepsilon)$), one can improve the estimate by a constant factor. Our result follows by an iterative application of this technique.

However, there is still a major complication in utilizing our low-dimensional estimator to obtain a high-dimensional estimator. In the unknown mean case, we knew the higher-order moments (since we assumed that the covariance is the identity). Here, we do not have control over the higher-order moments of the unknown Gaussian. Overcoming this difficulty requires several new techniques, which are quite complicated, and we defer the full details to Section 6. Our second main result is:

THEOREM 1.2. *Suppose we are given a set of* $n = \mathrm{poly}(d, 1/\varepsilon)$ *samples from the strong contamination model, where the underlying* d*-dimensional Gaussian is* $\mathcal{N}(0,\Sigma)$*. There is an algorithm to learn an estimate* $\mathcal{N}(0,\widehat{\Sigma})$ *that runs in time* $\mathrm{poly}(n, (1/\varepsilon)^{O(\log^4 1/\varepsilon)})$ *and with high probability satisfies*

$$d_{TV}(\mathcal{N}(0,\Sigma),\mathcal{N}(0,\widehat{\Sigma})) \le C\varepsilon \ ,$$

for a universal constant C *that is independent of the dimension.*

A key technical problem arises when we attempt to combine estimates for the covariance restricted to a subspace and its orthogonal complement. We refer to this as a *stitching* problem, where if we write Σ as

$$\Sigma = \begin{bmatrix} \Sigma_V & A^T \\ A & \Sigma_{V^\perp} \end{bmatrix} \ ,$$

and have accurate estimates for Σ_V and $\Sigma_{V^\perp}$, we still need to accurately estimate A. Our algorithm utilizes an unexpected connection to the unknown mean case: We show that, under a carefully chosen projection scheme, we can simulate noisy samples from a Gaussian with identity covariance, where the mean of this distribution encodes the information needed to recover A. We defer the full details to Section 6.4.

It turns out that we can solve the general case when both μ and Σ are unknown, by directly reducing to the previous subproblems, exactly as was done in [DKK+16] (with some caveats, addressed in Section 4.4). Since all of our error guarantees are optimal up to constant factors, there is only a constant factor loss in this reduction. Finally, we obtain the following corollary:

COROLLARY 1.1. *Suppose we are given a set of* $n = \mathrm{poly}(d, 1/\varepsilon)$ *samples from the strong contamination model, where the underlying* d*-dimensional Gaussian is* $\mathcal{N}(\mu,\Sigma)$*. There is an algorithm to learn an estimate* $\mathcal{N}(\widehat{\mu},\widehat{\Sigma})$ *that runs in time* $\mathrm{poly}(n, (1/\varepsilon)^{O(\log^4 1/\varepsilon)})$ *and with high probability satisfies*

$$d_{TV}(\mathcal{N}(\mu,\Sigma),\mathcal{N}(\widehat{\mu},\widehat{\Sigma})) \le C\varepsilon \ ,$$

for a universal constant C *that is independent of the dimension.*

This essentially settles the complexity of robustly learning a high-dimensional Gaussian. The sample complexity of our algorithm depends polynomially on d and $1/\varepsilon$, and the running time depends polynomially on d and quasi-polynomially on $1/\varepsilon$. Up to a constant factor, ours is the first high-dimensional algorithm that achieves the same error guarantees as in the one-dimensional case, where results were known for more than fifty years! It is an interesting open problem to reduce the running time to polynomial in $1/\varepsilon$ (while still being polynomial in d). As we explain in Section 6.6, this seems to require fundamentally new ideas.

More Related Work In addition to the works mentioned above, there has been an exciting flurry of recent work on robust high-dimensional estimation. This includes studying graphical models in the presence of noise [DKS16], tolerating much more noise by allowing the algorithm to output a list of candidate hypotheses [CSV17], formulating general conditions under which robust estimation is possible [SCV18], developing robust algorithms under sparsity assumptions [Li17, DBS17, BDLS17] where the number of samples is sublinear in the dimension, and leveraging theoretical insights to give practical algorithms that can be applied to genomic data [DKK+17]. We note that, in comparison to all these other works, ours is the only to efficiently achieve the information-theoretically optimal error guarantee (up to constant factors). Despite all of this rapid progress, there are still many interesting theoretical and practical questions left to explore.

1.3 Organization In Section 2, we go over preliminaries and notation that we will use throughout the paper. In Section 3, we describe an algorithm for robustly estimating the mean of a Gaussian in low-dimensional settings, and crucially apply it in the design of an algorithm for mean-estimation in high dimensions, described in Section 4. Similarly, in Section 5, we give an algorithm for robustly estimating the mean of degree-two polynomials in certain settings, which is applied in the context of our covariance-estimation algorithm in Section 6. Finally, we put these tools together and describe our general algorithm for robustly estimating a Gaussian in Section 7.

2 Preliminaries

In this section, we give various definitions and lemmata we will require throughout the paper. First, given a distribution F, we let $\mathbf{E}_F[f(X)] = \mathbf{E}_{X\sim F}[f(X)]$ denote the expectation of $f(X)$ under F. If S is a finite set, we let $\mathbf{E}_S[f(X)] = \mathbf{E}_{X\sim \mathrm{unif}(S)}[f(X)]$ denote the expectation of $f(X)$ under the uniform distribution over points in S (i.e., the empirical mean of f under S). Given any subspace $V \subseteq \mathbb{R}^d$, we let $\Pi_V : \mathbb{R}^d \to \mathbb{R}^d$ be the projection operator onto V. If $V = \mathrm{span}(v)$ is 1-dimensional, we will denote this projection as Π_v.

2.1 The Strong Contamination Model Here we formally define the *strong contamination model*.

DEFINITION 2. *Fix $\varepsilon > 0$. We say a set of samples $X_1, \ldots, X_n$ was generated from the* strong contamination model *on a distribution F, if it was generated via the following process:*

1. *We produce $(1-\varepsilon)n$ i.i.d. samples G from F.*
2. *An adversary is allowed to observe these samples and* add *εn points E arbitrarily.*

We are then given the set of samples $G \cup E$ in random order. Also, we will say that the samples $X_1, \ldots, X_n$ are ε-corrupted. Moreover given an ε-corrupted set of samples S, we will write $S = (G, E)$ where G is the set of uncorrupted *points and E is the set of* corrupted *points. Moreover, given a subset $S' \subset S$, we will also write $S' = (G', E')$, where $G' = S' \cap G$ and $E' = S' \cap E$ denote the set of uncorrupted points and corrupted points remaining in S'. L will denote $G \setminus G'$, which is the set of "lost" uncorrupted points.*

Given a contaminated set $S' = (G', E')$ and a set G so that $G' \subseteq G$, define the following quantities

$$\phi(S', G) = \frac{|G \setminus G'|}{|S'|}, \quad \psi(S', G) = \frac{|E'|}{|S'|}$$

$$\Delta(S', G) = \psi(S', G) + \phi(S', G) \log \frac{1}{\phi(S', G)}. \tag{2.1}$$

In particular, observe that if $\Delta(S', G) < O(\varepsilon)$, then a simple calculation implies that $\phi(S', G) \leq O(\varepsilon / \log 1/\varepsilon)$. Equivalently, we have removed at most an $O(\varepsilon/\log 1/\varepsilon)$ fraction of good points from G. This is crucial, as if we throw out an ε-fraction of good points then we essentially put ourselves in the subtractive model, and there our guarantees no longer hold.

There are two differences between the strong contamination model and Huber's contamination model. First, the number of corrupted points is fixed to be εn instead of being a random variable. However, this difference is negligible. It follows from basic Chernoff bounds that n samples from Huber's contamination model with parameter ε (for n sufficiently large) can be simulated by a $(1+o(1))\varepsilon$-corrupted set of samples, except with negligible failure probability. Hence, we lose only an additive $o(\varepsilon)$ term when translating from Huber's contamination model to the strong contamination model, which will not change any of the guarantees in our paper. The second difference is that the adversary is allowed to inspect the uncorrupted points before deciding on the corrupted points. This makes the model genuinely stronger since the samples we are given are no longer completely independent of each other.

2.2 Deterministic Regularity Conditions In analyzing our algorithms, we only need certain deterministic regularity conditions to hold on the uncorrupted points. In this subsection, we formally state

what these conditions are. It follows from known concentration bounds that these conditions all hold with high probability given a polynomial number of samples. Now with these regularity conditions defined once and for all, we will be able to streamline our proofs in the sense that each step in the analysis will only ever use one of these fixed set of conditions and will not use the randomness in the sampling procedure. We remark that some subroutines in our algorithm only need a subset of these conditions to hold, so we could improve the sample complexity by changing the regularity conditions we need at each step. However, since we will not be concerned with optimizing the sample complexity beyond showing that it is polynomial, we choose not to complicate our proofs in this manner.

2.2.1 Regularity Conditions for Unknown Mean In the unknown mean case, we will require the following condition:

DEFINITION 3. *Let G be a multiset of points in $\mathbb{R}^d$ and $\eta, \delta > 0$. We say that G is (η, δ)-good with respect to $\mathcal{N}(\mu, I)$ if the following hold:*

- *For all $x \in G$ we have $\|x - \mu\|_2 \leq O(\sqrt{d \log(|G|/\delta)})$.*
- *For every affine function $L : \mathbb{R}^d \to \mathbb{R}$ we have $|\Pr_G(L(X) \geq 0) - \Pr_{\mathcal{N}(\mu,I)}(L(X) \geq 0)| \leq \eta/(d \log(d/\eta\delta))$.*
- *We have that $\|\mathbf{E}_G[X] - \mathbf{E}_{\mathcal{N}(\mu,I)}[X]\|_2 \leq \eta$.*
- *We have that $\|\mathrm{Cov}_G[X] - I\|_2 \leq \eta/d$.*
- *For any even degree-2 polynomial $p : \mathbb{R}^d \to \mathbb{R}$ we have that*

$$\left|\mathop{\mathbf{E}}_{G}[p(X)] - \mathop{\mathbf{E}}_{\mathcal{N}(\mu,I)}[p(X)]\right| \leq \eta \mathop{\mathbf{E}}_{\mathcal{N}(\mu,I)}[p^2(X)]^{1/2},$$
$$\left|\mathop{\mathbf{E}}_{G}[p^2(X)] - \mathop{\mathbf{E}}_{\mathcal{N}(\mu,I)}[p^2(X)]\right| \leq \eta \mathop{\mathbf{E}}_{\mathcal{N}(\mu,I)}[p^2(X)] ,$$
$$\Pr_{G}[p(X) \geq 0] \leq \Pr_{\mathcal{N}(\mu,I)}[p(X) \geq 0] + \frac{\eta}{d \log(|G|/\delta)} .$$

It is easy to show (see Lemma 4.2) that given enough samples from $\mathcal{N}(\mu, I)$, the empirical data set will satisfy these conditions with high probability.

2.2.2 Regularity Conditions for Unknown Covariance In the unknown covariance case, we will require the following condition:

DEFINITION 4. *Let G be a set of n points of $\mathbb{R}^d$, and $\eta, \delta > 0$. We say that G is (η, δ)-good with respect to $\mathcal{N}(0, \Sigma)$ if the following hold:*

- *For all $x \in G$ we have that $x^T \Sigma^{-1} x = O(d \log(|G|/\delta))$.*
- *For any even degree-2 polynomial $p : \mathbb{R}^d \to \mathbb{R}$ we have*

$$\left|\mathop{\mathbf{E}}_{G}[p(X)] - \mathop{\mathbf{E}}_{\mathcal{N}(0,\Sigma)}[p(X)]\right| \leq \eta \mathop{\mathbf{E}}_{\mathcal{N}(0,\Sigma)}[p^2(X)]^{1/2} ,$$
$$\left|\mathop{\mathbf{E}}_{G}[p^2(X)] - \mathop{\mathbf{E}}_{\mathcal{N}(0,\Sigma)}[p^2(X)]\right| \leq \eta \mathop{\mathbf{E}}_{\mathcal{N}(0,\Sigma)}[p^2(X)] ,$$
$$\Pr_{X\sim G}[p(X) \geq 0] \leq \Pr_{\mathcal{N}(0,\Sigma)}[p(X) \geq 0] + \frac{\eta^2}{d \log(|G|/\delta)} .$$

- *For any even degree-4 polynomial $p : \mathbb{R}^d \to \mathbb{R}$ we have*

$$\left|\mathop{\mathbf{E}}_{G}[p(X)] - \mathop{\mathbf{E}}_{\mathcal{N}(0,\Sigma)}[p(X)]\right| \leq \eta \mathop{\mathbf{Var}}_{\mathcal{N}(0,\Sigma)}[p(X)]^{1/2} ,$$
$$\Pr_{G}[p(X) \geq 0] \leq \Pr_{\mathcal{N}(\mu,I)}[p(X) \geq 0] + \frac{\eta^2}{2 \log(1/\varepsilon)(d \log(|G|/\delta))^2} .$$

As before, it is easy to show (see Lemma 6.2) that given enough samples from $\mathcal{N}(0, \Sigma)$, the empirical data set will satisfy these conditions with high probability.

2.3 Bounds on the Total Variation Distance We will require some simple bounds on the total variation distance between two Gaussians. These bounds are well-known. Roughly speaking, they say that the total variation distance between two Gaussians with identity covariance is governed by the ℓ_2 norm between their means, and the total variation distance between two Gaussians with mean zero is governed by the Frobenius norm between their covariance matrices, provided that the matrices are close to the identity.

LEMMA 2.1. *Let $\mu_1, \mu_2 \in \mathbb{R}^d$ be such that $\|\mu_1 - \mu_2\|_2 = \varepsilon$ for $\varepsilon < 1$. Then*

$$d_{\mathrm{TV}}\left(\mathcal{N}(\mu_1, I), \mathcal{N}(\mu_2, I)\right) = \left(\frac{1}{\sqrt{2\pi}} + o(1)\right)\varepsilon .$$

For clarity of exposition we defer this calculation to the Appendix.

We also need to bound the total variation distance between two Gaussians with zero mean and different covariance matrices. The natural norm to use is the Mahalanobis distance. But in our setting, we will be able to use the more convenient Frobenius norm instead (because we effectively reduce to the case that the covariance matrices will be close to the identity):

LEMMA 2.2. (COR. 2.14 IN [DKK+16]) *Let* $\Sigma, \widehat{\Sigma}$ *be such that* $\|\Sigma - I\|_F \le O(\varepsilon \log 1/\varepsilon)$, *and* $\|\Sigma - \widehat{\Sigma}\|_F \le C\varepsilon$. *Then* $d_{\mathrm{TV}}(\mathcal{N}(0,\Sigma), \mathcal{N}(0,\widehat{\Sigma})) \le O(\varepsilon)$.

These lemmata show that parameter estimation and approximation in total variation distance are essentially equivalent. Indeed, in this paper, we achieve both guarantees, but state our results in terms of total variation estimation.

3 Robustly Learning the Mean in Low Dimensions

This section is dedicated to the proof of the following theorem:

THEOREM 3.1. *Fix* $\mu \in \mathbb{R}^d$, *and let* $\varepsilon, \gamma, \delta > 0$. *Let* $S_0 = (G_0, E_0)$ *be such that* G_0 *is a* $(\gamma\varepsilon, \delta)$*-good set with respect to* $\mathcal{N}(\mu, I)$, *and* $|E_0|/|S_0| \le \varepsilon$. *Let* $S = (G, E)$ *be another set with* $\Delta(S, S_0) < \varepsilon$. *Let* $V \subseteq \mathbb{R}^d$ *be a subspace. For all* $0 < \rho < 1$, *the algorithm* LEARNMEANLOWD$(V, \gamma, \varepsilon, \delta, S, \rho)$ *runs in time* $\mathrm{poly}(d, |S|, (1/\rho)^{O(\dim(V))}, \log(\rho\varepsilon/(1-\rho)), \log(1/\rho))$ *and returns a* $\widetilde{\mu}$ *so that*

$$\|\Pi_V(\mu - \widetilde{\mu})\|_2 = \frac{1+2\rho}{1-\rho}\left(\sqrt{\pi} + O\left(\frac{\gamma}{d}\right)\right)\varepsilon\,.$$

In particular, as we let $\rho, \gamma \to 0$, the parameter estimation error approaches $\sqrt{\pi}\varepsilon$ (corresponding to a total variation approximation of $\varepsilon/\sqrt{2}$). In the full version we show that no algorithm can achieve parameter estimation error better than $\sqrt{\frac{\pi}{2}}\varepsilon$. Thus, we achieve a $\sqrt{2}$ approximation to the optimal error.

For simplicity, in the rest of this section, we will let $V = \mathbb{R}^d$, that is, we assume there is no projection. It should be clear that this can be done without loss of generality. Our algorithm proceeds as follows: First, we show that in one dimension, the median produces an estimate which is optimal, up to lower order terms, if the sample set is $(\gamma\varepsilon, \delta)$-good with respect to the underlying Gaussian. Then, we show that by using a net argument, we can produce a convex body in $\mathbb{R}^d$ with diameter at most $2R = 2(\sqrt{\frac{\pi}{2}} + o(1))\varepsilon$ which must contain the true mean. Finally, we use an old result of Jung [Jun01] that such a set can be circumscribed by a ball of radius $\sqrt{2}R$ (see [BW41] for an English language version of the result). We use the center of the ball as our estimate $\widetilde{\mu}$.

3.1 Robustness of the Median First we show that if we project onto one dimension, then the median of the corrupted data differs from the true mean by at most $\sqrt{\frac{\pi}{2}}\varepsilon + o(\varepsilon)$. Our proof will rely only on the notion of a $(\gamma\varepsilon, \delta)$-good set with respect to $\mathcal{N}(\mu, I)$ and thus it works even in the strong contamination model. By a fairly standard calculation, we show:

LEMMA 3.1. *Fix any* $v \in \mathbb{R}^d$. *Fix* $\mu \in \mathbb{R}^d$, *and let* $\delta > 0$. *Let* $S_0 = (G_0, E_0)$ *be so that* G_0 *be a* $(\gamma\varepsilon, \delta)$*-good set with respect to* $\mathcal{N}(\mu, I)$, *and* $|E_0|/|S_0| \le \varepsilon$. *Let* $S = (G, E)$ *be another set with* $\Delta(S, S_0) < \varepsilon$. *Let* b *be the median of* S *when projected onto* v. *Then,* $|b - \Pi_v \mu| \le \left(\sqrt{\frac{\pi}{2}} + O\left(\frac{\gamma}{d}\right)\right)\varepsilon$.

For conciseness we defer the proof of this to the full version.s

3.2 Finding a Minimum Radius Circumscribing Ball For any $x \in \mathbb{R}^d$ and $r > 0$, let $B(x, r) = \{y \in \mathbb{R}^d : \|x - y\|_2 \le r\}$ denote the closed ball of radius r centered at x. The following classical result of Gale gives a bound on the radius of the circumscribing ball of any convex set in terms of its diameter:

THEOREM 3.2. (SEE [JUN01, BW41]) *Fix* $R > 0$. *Let* $\mathcal{C} \subseteq \mathbb{R}^d$ *be a convex body so that for all* $x, y \in \mathcal{C}$, *we have* $\|x - y\|_2 \le 2R$. *Then* $\mathcal{C}$ *is contained within a ball of radius* $R\sqrt{2}$.

The bound is asymptotically achieved for the standard simplex as we increase its dimension. The goal of this subsection is to show that the (approximately) minimum radius circumscribing ball can be found efficiently. We will assume we are given an *approximate projection* oracle for the convex body that given a point $y \in \mathbb{R}^d$, outputs a point which is almost the closest point in $\mathcal{C}$ to x:

DEFINITION 5. *A* ρ-projection oracle *for a convex body* $\mathcal{C}$ *is a function* $\mathcal{O} : \mathbb{R}^d \to \mathbb{R}^d$, *which, given a point* $y \in \mathbb{R}^d$, *outputs a point* $x \in \mathcal{C}$ *so that* $\|x - y\|_2 \le \inf_{x' \in \mathcal{C}} \|x' - y\|_2 + \rho$.

Our first step is to use such an oracle to construct a net for $\mathcal{C}$. First, we need the following well-known bound on the size of the net.

CLAIM 1. *Fix* $r > 0$. *Then, for any* $\beta > 0$, *there is a* β*-net* $\mathcal{F}$ *for the sphere of radius* r *around* 0 *in* $\mathbb{R}^d$ *of size* $(r/\beta)^{O(d)}$. *Moreover, this net can be constructed in time* $\mathrm{poly}(d, |\mathcal{F}|)$.

With this, we can show:

LEMMA 3.2. *Fix* $R, \mathcal{C}$ *as in Theorem 3.2, and let* $1 > \rho > 0$. *Let* $x \in \mathcal{C}$ *be arbitrary. Let* $\mathcal{O}$ *be a* $(\rho R/3)$*-projection oracle for* $\mathcal{C}$. *Suppose a call to* $\mathcal{O}$ *runs in time* T. *Then, there is an algorithm* CIRCUMSCRIBENET$(R, \rho, \mathcal{O}, x)$ *which runs in time* $\mathrm{poly}((R/\rho)^{O(d)}, T)$ *and outputs a set* $\mathcal{X} \subseteq \mathbb{R}^d$ *so that* $\mathcal{X}$ *is a* (ρR)*-net for* $\mathcal{C}$, *and moreover,* $|\mathcal{X}| \le (R/\rho)^{O(d)}$.

The algorithm is fairly straightforward. First, we observe that $\mathcal{C}$ is contained within $B(x, 2R)$. We then form a $(\rho R)/3$-net of $B(x, 2R)$ using Claim 1. We then iterate over every element v of this net, and use our projection oracle to (approximately) find the closest point in $\mathcal{C}$ to v. If this point is too far away, we throw it out, otherwise, we add this projected point into the net. The formal pseudocode for CircumscribeNet is given in Algorithm 1.

Algorithm 1 Generating a net of $\mathcal{C}$

1: **function** CircumscribeNet$(R, \rho, \mathcal{O}, x)$
2: Form an $\rho/3$-net $\mathcal{F}'$ of the sphere of radius 2 of size $(1/\rho)^{O(d)}$ as in Claim 1.
3: Let $\mathcal{F} = R \cdot \mathcal{F}' + x$.
4: Let $\mathcal{X} \leftarrow \emptyset$
5: **for** each $v \in \mathcal{F}$ **do**
6: Let $u_v \leftarrow \mathcal{O}(v)$
7: **if** $\|v - u_v\|_2 \le 2\rho R/3$ **then**
8: Add u_v to $\mathcal{X}$
9: **end if**
10: **end for**
11: **return** $\mathcal{X}$
12: **end function**

Proof. The runtime bound follows from Claim 1. We now turn our attention to correctness. By Claim 1, and rescaling and shifting, the set $\mathcal{F}$ is clearly a $(\rho R)/3$-net for a ball B of radius $2R$ containing $\mathcal{C}$. We now claim that the set $\mathcal{X}$ is indeed a $(\rho R)/3$-net for $\mathcal{C}$. Fix $y \in \mathcal{C}$. Since $\mathcal{C} \subseteq B$, this implies there is some $v \in \mathcal{F}$ so that $\|y - v\|_2 \le \rho R/3$. Thus, in Line 7, when processing v, we must find some $u_v \in \mathcal{C}$ so that $\|u_v - v\|_2 \le 2\rho R/3$. The claim then follows from the triangle inequality.

With this, we obtain:

Corollary 3.1. *Fix $R, \mathcal{C}, \rho, \mathcal{O}, x$ as in Lemma 3.2. Suppose a call to $\mathcal{O}$ runs in time T. Then, there is an algorithm* CircumscribeNet$(R, \rho, \mathcal{O}, x)$ *which runs in time* $\mathrm{poly}((R/\rho)^{O(d)}, T)$ *and returns a point $\widehat{y}$ so that $\mathcal{C}$ is contained within a ball of radius $\sqrt{2}(1 + 2\rho)R$.*

The algorithm at this point is very simple. Using the output of CircumscribeNet, we iterate over all points in a net over $B(x, 2R)$, find an x in this net so that the distance to all points in the net is at most $\sqrt{2}(1 + \rho)R$, and output any such point. The formal pseudocode for Circumscribe is given in Algorithm 2.

Algorithm 2 Finding a circumscribing ball of small radius

1: **function** Circumscribe$(R, \rho, \mathcal{O}, x)$
2: Form an $\rho/3$-net $\mathcal{F}'$ of $B(0, 2)$ of size $(1/\rho)^{O(d)}$ as in Claim 1.
3: Let $\mathcal{F} = R \cdot \mathcal{F}' + x$.
4: Let $\mathcal{X} \leftarrow$ CircumscribeNet$(R, \rho, \mathcal{O}, x)$.
5: **for** each $v \in \mathcal{F}$ **do**
6: **if** for all $u \in \mathcal{X}$, we have $\|u - v\|_2 \le \sqrt{2}(1 + \rho)R$ **then**
7: **return** u
8: **end if**
9: **end for**
10: **end function**

Proof. The runtime bound is immediate. By Theorem 3.2, there is some $y \in B(x, 2R)$ so that $\mathcal{C} \subseteq B(y, R\sqrt{2})$. Thus, by the triangle inequality, there is some $y' \in \mathcal{F}$ so that $\mathcal{C} \subseteq B(y, \sqrt{2}(1 + \rho)R)$. Thus, the algorithm will output some point $y'' \in \mathcal{F}$. By an additional application of the triangle inequality, since $\mathcal{X}$ is a ρR-net for $\mathcal{C}$, this implies that $\mathcal{C} \subseteq B(y'', \sqrt{2}(1 + 2\rho)R)$, as claimed.

3.3 The Full Low-Dimensional Algorithm We now have all the tools to describe the full algorithm in low-dimensions. Let S be our corrupted dataset as in Theorem 3.1. Fix $\rho > 0$. We first produce a ρ-net $\mathcal{F}$ over the unit sphere in $\mathbb{R}^d$. By (a slight modification of) Claim 1, this net has size $(1/\rho)^{O(d)}$ and can be constructed in time $\mathrm{poly}(d, |\mathcal{F}|)$. For each $v \in \mathcal{F}$, we project all points in S onto v, and take the median of these points to produce b_v. We then construct the following set:

$$\mathcal{C} = \bigcap_{v \in \mathcal{F}} \{y \in \mathbb{R}^d : \langle v, y \rangle \in [b_v - \beta, b_v + \beta]\} , \tag{3.2}$$

where $\beta = \sqrt{\frac{\pi}{2}}\varepsilon + O\left(\frac{\gamma\varepsilon}{d}\right) + o(\varepsilon)$ is as in Lemma 3.1. We now show two properties of this set, which in conjunction with the machinery above, allows us to prove Theorem 3.1. The first shows that $\mathcal{C}$ has small diameter:

Claim 2. *For all $x, y \in \mathcal{C}$, we have $\|x - y\|_2 \le 2\beta/(1 - \rho)$.*

Proof. Fix any $x, y \in \mathcal{C}$. By definition of $\mathcal{C}$, it follows that for all $v \in \mathcal{F}$, we have $|\langle x - y, v \rangle| \le 2\beta$. For any u with $\|u\|_2 = 1$, there is some $v \in \mathcal{F}$ with $\|u - v\|_2 \le \varepsilon$, and so we have

$$\begin{aligned} |\langle x - y, u \rangle| &\le |\langle x - y, v \rangle| + |\langle x - y, u - v \rangle| \\ &\le 2\beta + \rho\|x - y\|_2 . \end{aligned}$$

Taking the supremum over all unit vectors u and simplifying yields that $\|x - y\|_2 \le 2\beta/(1 - \rho)$, as claimed.

The second property shows that we may find an α-projection oracle for $\mathcal{C}$ efficiently.

CLAIM 3. *Fix $\rho' > 0$. There is a ρ'-projection oracle* PROJORACLE$(y, \rho', \mathcal{C})$ *for $\mathcal{C}$ which runs in time* $\mathrm{poly}((1/\rho)^{O(d)}, \log(\gamma\varepsilon/(1-\rho)), \log(1/\rho'))$.

Proof. The projection problem may be stated as

$$\min \|x - y\|_2 \text{ s.t.} \langle v, y\rangle \in [b_v - \beta, b_v + \beta],\ \forall v \in \mathcal{F}\ .$$

This is a convex minimization problem with linear constraints. By the classical theory of optimization [GLS88], finding a ρ-approximate y can be done in $\mathrm{poly}(d, \log(\mathrm{vol}(\mathcal{C})/\rho'))$ queries to a separation oracle for $\mathcal{C}$. Since the separation oracle must only consider the constraints induced by $\mathcal{F}$, this can be done in time $(1/\rho)^{O(d)}$. Since by Claim 2 we have $\mathrm{vol}(\mathcal{C}) \le (2\beta/(1-\rho))^{O(d)}$, the desired runtime follows immediately.

We now finally describe LEARNMEANLOWD. Using convex optimization, we first find an arbitrary $x \in \mathcal{C}$. By Lemma 3.1 we know $\mu \in \mathcal{C}$ and so this step succeeds. After constructing $\mathcal{C}$, we run CIRCUMSCRIBE with appropriate parameters, and return the outputted point. The formal pseudocode for LEARNMEANLOWD is given in Algorithm 3.

Algorithm 3 Learning the mean robustly in low dimensions

function LEARNMEANLOWD$(\varepsilon, \delta, S, \rho)$
  Form a ρ-net $\mathcal{F}$ of $B(0,1)$ of size $(1/\rho)^{O(d)}$ as in Claim 1.
  for each $v \in \mathcal{F}$ **do**
    Let b_v be the median of S projected onto v.
  end for
  Form $\mathcal{C}$ as in Equation (3.2).
  Find an $x \in \mathcal{C}$ using convex optimization.
  Let $\beta = \sqrt{\frac{\pi}{2}}\varepsilon + O\left(\frac{\gamma\varepsilon}{d}\right) + o(\varepsilon)$
  Let $R = \beta/(1-\rho)$
  Let $\mathcal{O}(\cdot) =$ PROJORACLE$(\cdot, (\rho R)/3, \mathcal{C})$
  return CIRCUMSCRIBE$(R, \rho, \mathcal{O}, x)$
end function

Proof. The runtime claim follows from the runtime claims for CIRCUMSCRIBE and PROJORACLE. Thus, it suffices to prove correctness of this algorithm. By Lemma 3.1, we know that $\mu \in \mathcal{C}$. By Claim 2 and Corollary 3.1, the output y satisfies $B(y, \sqrt{2}\frac{1+2\rho}{1-\rho}\beta)$. Thus, we have $\|\mu - y\|_2 \le \sqrt{2}\frac{1+2\rho}{1-\rho}\beta$, as claimed.

4 Robustly Learning the Mean in High Dimensions

In this section, we prove the following theorem, which is our first main result:

THEOREM 4.1. *Fix $\varepsilon, \gamma, \delta > 0$, and let $X_1, \ldots, X_n$ be an ε-corrupted set of points from $\mathcal{N}(\mu, I)$, where $\|\mu\|_2 \le O(\varepsilon \log 1/\varepsilon)$, and where*

$$n = \Omega\left(\frac{(d\log(d/\gamma\varepsilon\delta))^6}{\gamma^2\varepsilon^2}\right)\ .$$

Then, for every $\alpha, \beta > 0$, there is an algorithm RECOVERMEAN$(X_1, \ldots, X_n, \varepsilon, \delta, \gamma, \alpha, \beta)$ *which runs in time* $\mathrm{poly}(d, 1/\gamma, 1/\varepsilon^\beta, 1/\alpha, \log 1/\delta)$ *and outputs a $\widehat{\mu}$ so that with probability $1-\delta$, we have $\|\widehat{\mu} - \mu\|_2 \le \left(\frac{\sqrt{\pi}+O(\gamma)}{1-\alpha} + \frac{1}{\sqrt{\beta}}\right)\varepsilon$.*

In particular, observe that Theorem 4.1, in conjunction with Lemma 2.1, gives us Theorem 1.1, if we set $\gamma = o(1)$. With this, we may state our primary algorithmic contribution:

THEOREM 4.2. *Fix $\varepsilon, \gamma, \alpha, \delta, \beta > 0$, and let $S_0 = (G_0, E_0)$ be an ε-corrupted set of samples of size n from $\mathcal{N}(\mu, I)$, where $\|\mu\|_2 \le O(\varepsilon \log 1/\varepsilon)$, and where $n = \mathrm{poly}(d, 1/(\gamma\varepsilon), \log 1/\delta)$. Suppose that G_0 is $(\gamma\varepsilon, \delta)$-good with respect to $\mathcal{N}(\mu, I)$. Let $S \subseteq S_0$ be a set so that $\Delta(S, G_0) \le \varepsilon$. Then, there exists an algorithm* FILTERMEANOPT *that given $S, \varepsilon, \gamma, \alpha, \beta$ outputs one of two possible outcomes:*

(i) A $\widehat{\mu}$, so that $\|\widehat{\mu} - \mu\|_2 \le \left(\frac{\sqrt{\pi}+O(\gamma)}{1-\alpha} + \frac{1}{\sqrt{\beta}}\right)\varepsilon$.

(ii) A set $S' \subset S$ so that $\Delta(S', G_0) < \Delta(S, G_0)$.

Moreover, FILTERMEANOPT *runs in time* $\mathrm{poly}(d, 1/\gamma, 1/\varepsilon^\beta, 1/\alpha, \log 1/\delta)$.

By first running the algorithm of [DKK+16] to obtain an estimate of the mean to error $O(\varepsilon\sqrt{\log 1/\varepsilon})$, then running FILTERMEANOPT at most polynomially many times, we clearly recover the guarantee in Theorem 4.1. Thus, the rest of the section is dedicated to the proof of Theorem 4.2.

At a high level, the structure of the argument is as follows: We first show that if there is a subspace of eigenvectors of dimension at least $O(\log 1/\varepsilon)$ of the empirical covariance matrix with large associated eigenvalues, then we can produce a filter using a degree-2 polynomial (Section 4.1). Otherwise, we know that there are at most $O(\log 1/\varepsilon)$ eigenvectors of the empirical covariance with a large eigenvalue. We can learn the mean in this small dimensional subspace using our learning algorithm from the previous

section, and then we can argue that the empirical mean on the remaining subspace is close to the true mean (Section 4.2).

This outline largely follows the structure of the filter arguments given in [DKK+16], however, the filtering algorithm we use here requires a couple of crucial new ideas. First, to produce the filter, instead of using a generic degree-2 polynomial over this subspace, we construct an explicit, structured, degree-2 polynomial which produces such a filter. Crucially, we can exploit the structure of this polynomial to obtain very tight tail bounds, e.g., via the Hanson-Wright inequality. This is critical to avoid a quasi-polynomial runtime. If instead we used arbitrary degree-2 polynomials in this subspace, it would need to be of dimension $O(\log^2 1/\varepsilon)$ and the low-dimensional algorithm in the second step would take quasi-polynomial time.

Second, we must be careful to throw out far fewer good points than corrupted points. In particular, by our definition of Δ (which gives an additional logarithmic penalty to discarding good points) and our guarantee that Δ decreases, our filter can only afford to throw out an $\varepsilon/\log(1/\varepsilon)$ fraction of good points in total, since Δ is initially ε. This is critical, as if we threw away an ε-fraction of good points, then proving that the problem remains efficiently solvable becomes problematic. In particular, if these points were thrown away arbitrarily, then this becomes the full additive and subtractive model, for which a statistical query lower bound prevents us from getting an $O(\varepsilon)$-approximate answer in polynomial time [DKS17]. To avoid discarding too many good points, we exploit tight exponential tail bounds of Gaussians, and observe that by slightly increasing the threshold at which we filter away points, we decrease the fraction of good points thrown away dramatically.

4.1 Making Progress with Many Large Eigenvalues We now give an algorithm for the case when there are many eigenvalues which are somewhat large. Formally, we show:

THEOREM 4.3. *Fix $\varepsilon, \gamma, \delta, \alpha, \beta > 0$, and let $S_0 = (G_0, E_0)$ be an ε-corrupted set of samples of size n from $\mathcal{N}(\mu, I)$, where $\|\mu\|_2 \le O(\varepsilon \log 1/\varepsilon)$, and where $n = \mathrm{poly}(d, 1/(\gamma\varepsilon), \log 1/\delta)$. Suppose that G_0 is $(\gamma\varepsilon, \delta)$-good with respect to $\mathcal{N}(\mu, I)$. Let $S \subseteq S_0$ be a set so that $\Delta(S, G_0) \le \varepsilon$. Let $\widehat{\Sigma}$ be the sample covariance of S, let $\widehat{\mu}$ be the sample mean of S, and let V be the subspace of all eigenvectors of $\widehat{\Sigma} - I$ with eigenvalue more than $\frac{1}{\beta}\varepsilon$. Then, there exists an algorithm* FILTERMEANMANYEIG *that given $S, \varepsilon, \gamma, \delta, \alpha, \beta$ outputs one of two possible outcomes:*

1. *If $\dim(V) \ge C_1 \beta \log(1/\varepsilon)$, then it outputs an S' so that $\Delta(S', G_0) < \Delta(S, G_0)$.*
2. *Otherwise, the algorithm outputs "OK", and outputs an orthonormal basis for V.*

Our algorithm works as follows: It finds all large eigenvalues of $\widehat{\Sigma} - I$, and if there are too many, produces an explicit degree-2 polynomial which, as we will argue, produces a valid filter. The formal pseudocode for our algorithm is in Algorithm 4.

Algorithm 4 Filter if there are many large eigenvalues of the covariance

function FILTERMEANMANYEIG$(S, \varepsilon, \gamma, \delta, \alpha, \beta)$
  Let $C_1, C_2, C_3 > 0$ be sufficiently large constants.
  Let $\widehat{\mu}$ and $\widehat{\Sigma}$ be the empirical mean and covariance of S, respectively.
  Let V be the subspace of $\mathbb{R}^d$ spanned by eigenvectors of $\widehat{\Sigma} - I$ with eigenvalue more than $\frac{1}{\beta}\varepsilon$.
  if $\dim(V) \ge C_1 \beta \log(1/\varepsilon)$ **then**
    Let V' be a subspace of V of dimension $C_1 \beta \log(1/\varepsilon)$.
    Let $\tilde{\mu}$ be an approximation to $\Pi_{V'}(\mu)$ with ℓ_2-error $\frac{\sqrt{\pi}+O(\gamma)}{1-\alpha}\varepsilon$, computed using LEARNMEANLOWD$(V, \gamma, \varepsilon, \delta, S, \gamma)$.
    Let $p(x)$ be the quadratic polynomial $p(x) = \|\Pi_{V'}(x) - \tilde{\mu}\|_2^2 - \dim(V')$.
    Find a value $T > 0$ so that either:
    (a) $T > C_2 d \log(|S|/\delta)$ and $p(x) > T$ for at least one $x \in S$, or
    (b) $T > 2C_3 \log(1/\varepsilon)/c_0$ and $\Pr_S(p(x) > T) > \exp(-c_0 T/(2C_3)) + \gamma\varepsilon/(d \log(|S|/\delta))$.
    return $S' = \{x \in S : p(x) \le T\}$
  else
    return an orthonormal basis for V.
  end if
end function

For clarity of exposition, we defer the proof of Theorem 4.3 to the full version.

4.2 Returning an Estimate When There are Few Large Eigenvalues At this point, we have run the filter of Algorithm 4 until there are few large eigenvalues. In the subspace with large eigenvalues, we again run the low dimensional algorithm to obtain an estimate for the mean in this subspace. Recall that Lemma 3.1 guarantees the accuracy of this estimator within this subspace. In the complement of this subspace, where the empirical covariance is very close

to the identity, Lemma 4.1 (stated below) shows that the empirical mean is close to the true mean. This leads to a simple algorithm which outputs an estimate for the mean, described in Algorithm 5.

Algorithm 5 Return a mean if there are few large eigenvalues of the covariance

1: **function** FILTERMEANFEWEIG($S, \varepsilon, \gamma, \delta, \alpha, \beta, V$)
2: Let $\tilde{\mu}_V$ be an approximation to $\Pi_V(\mu)$ with ℓ_2-error $\frac{\sqrt{\pi}+O(\gamma)}{1-\alpha}\varepsilon$, computed using LEARNMEANLOWD($V, \gamma, \varepsilon, \delta, S, \gamma$).
3: Let $\tilde{\mu}_{V^\perp}$ be the empirical mean on $V^\perp$, $\Pi_{V^\perp}\hat{\mu}$.
4: **return** $\tilde{\mu}_V + \tilde{\mu}_{V^\perp}$.
5: **end function**

LEMMA 4.1. *Let μ, η, G_0, S be as in Theorem 4.3. Let $\widehat{\mu}$ be the sample mean of S, and let v be a unit vector. Suppose that $\langle v, \mu - \widehat{\mu}\rangle > \frac{\varepsilon}{\beta^{1/2}}$. Then $\mathbf{Var}_S[\langle v, X\rangle] > 1 + \frac{\varepsilon}{\beta}$.*

For clarity of exposition, we defer the proof of Lemma 4.1 to the full version.

4.3 The Full High-Dimensional Algorithm

We now have almost all the pieces needed to prove the full result. The last ingredient is the fact that, given enough samples, the good set condition is satisfied by the samples from the true distribution. Formally,

LEMMA 4.2. *Fix $\eta, \delta > 0$. Let $X_1, \ldots, X_n$ be independent samples from $\mathcal{N}(\mu, I)$, where $n = \Omega((d\log(d/\eta\delta))^6/\eta^2)$. Then, $S = \{X_1, \ldots, X_n\}$ is (η, δ)-good with respect to $\mathcal{N}(\mu, I)$ with probability at least $1 - \delta$.*

Proof. This follows from Lemmas 8.3 and 8.16 of [DKK+16].

At this point, we conclude with the proof of Theorem 4.1. Within the subspace V, Lemma 3.1 guarantees that the mean is accurate up to ℓ_2-error $\frac{\sqrt{\pi}+O(\gamma)}{1-\alpha}\varepsilon$. Within the subspace $V^\perp$, the contrapositive of the statement of Lemma 4.1 guarantees the mean is accurate up to ℓ_2-error $\frac{\varepsilon}{\beta^{1/2}}$. The desired result follows from the Pythagorean theorem.

4.4 An Extension, with Small Spectral Noise

For learning of arbitrary Gaussians, we will need a simple extension that allows us to learn the mean even in the presence of some spectral norm error in the covariance matrix. Since the algorithms and proofs are almost identical to the techniques above, we omit them for conciseness. Formally, we require:

THEOREM 4.4. *Fix $\chi, \varepsilon, \delta > 0$, and let $X_1, \ldots, X_n$ be an ε-corrupted set of points from $\mathcal{N}(\mu, \Sigma)$, where $\|\Sigma - I\|_2 \le O(\chi)$, $\|\mu\|_2 \le O(\varepsilon \log 1/\varepsilon)$, and where $n = \mathrm{poly}(d, 1/\chi, 1/\varepsilon, \log 1/\delta)$. For any $\gamma > 0$, there is an algorithm* RECOVERMEANNOISY$(X_1, \ldots, X_n, \varepsilon, \delta, \gamma, \chi)$ *which runs in time $\mathrm{poly}(d, 1/\chi, 1/\varepsilon, \log 1/\delta)$ and outputs a $\widehat{\mu}$ so that with probability $1 - \delta$, we have $\|\widehat{\mu} - \mu\|_2 \le (C + \gamma)\varepsilon + O(\chi)$.*

This extension follows from two elementary observations:

1. For the learning in low dimensions, observe that the median is naturally robust to error in the covariance, and in general, by the same calculation we did, the error of the median becomes $O(\varepsilon + \alpha)$.
2. For the filter, observe that we only need concentration of squares of linear functions, and whatever error we have in this concentration goes directly into our error guarantee. Thus, by the same calculations that we had above, if we filtered for eigenvalues above $1+O(\varepsilon+\alpha)$, we would immediately get the desired bound.

5 Robustly Estimating the Mean of Degree Two Polynomials

In this section, we give robust estimates of $\mathbf{E}[p^2(X)]$ for degree-2 polynomials p in subspaces of small dimension, which is an important prerequisite to learning the covariance in high-dimensions. A crucial ingredient in our algorithm is the following improvement theorem (stated and proved in the next section) which shows how to take any weak high-dimensional estimate for the covariance and use it to get an even better robust estimate for $\mathbf{E}[p^2(X)]$.

5.1 Additional Preliminaries

Here we give some additional preliminaries we require for the low-dimensional learning algorithm we present here. We will need the following well-known tail bound for degree-2 polynomials:

LEMMA 5.1. (HANSON-WRIGHT [LM00, VER10]) *Let $X \sim \mathcal{N}(0, I) \in \mathbb{R}^d$ and A be a $d \times d$ matrix. Then for some absolute constant c_0, for every $t \ge 0$,*

$$\Pr\left(\left|X^T AX - \mathbf{E}[X^T AX]\right| > t\right) \le 2\exp\left(-c_0 \cdot \min\left(\frac{t^2}{\|A\|_F^2}, \frac{t}{\|A\|_2}\right)\right).$$

We will also require the following lemmata:

LEMMA 5.2. (HÖLDER'S FOR SCHATTEN NORMS) *Let A, B be matrices. Then, for all p, q so that $\frac{1}{p} + \frac{1}{q} = 1$, we have $\|AB\|_{S^1} \le \|A\|_{S^p}\|B\|_{S^q}$.*

This implies the following corollary, whose proof we defer to the full version:

COROLLARY 5.1. *Let $\Sigma, \widehat{\Sigma}, M$ be so that $\|\Sigma - \widehat{\Sigma}\|_F \leq O(\delta)$, and so that $\|M\|_F = 1$. Then, we have $\|\Sigma^{1/2} M \Sigma^{1/2} - \widehat{\Sigma}^{1/2} M \widehat{\Sigma}^{1/2}\|_{S^1} \leq 5\delta$.*

5.2 An Improvement Theorem Here we state and prove one of the main technical ingredients in our algorithm for robustly learning the covariance.

THEOREM 5.1. *Fix $\varepsilon, \delta, \tau > 0$. Let Σ be so that $\|\Sigma - I\|_F \leq O(\varepsilon \log 1/\varepsilon)$, and fix a $p \in \mathcal{P}_2$, where $\mathcal{P}_2$ denotes the set of even degree-2 polynomials in d variables. Let G_0 be an (ε, δ)-good set of samples from $\mathcal{N}(0, \Sigma)$, and let $S = \{X_1, \ldots, X_n\}$ be so that $\Delta(S, G_0) \leq \varepsilon$. Then, for any $C > 0$ there is an algorithm* LEARNMEANCHISQUARED *which, given $p, X_1, \ldots, X_n$, and ε, outputs a $\widehat{\mu}$ so that with probability $1 - \tau$ over the randomness of the algorithm,*

$$\left| \widehat{\mu} - \mathop{\mathbf{E}}_{X \sim \mathcal{N}(0,\Sigma)}[p(X)] \right| \leq \|\Sigma - I\|_F / C + O(\log(C)\varepsilon) \; .$$

Moreover, the algorithm runs in time $O(|S| + \log(1/\tau)/\varepsilon^2)$.

The way to think about how this result fits into the overall strategy is that robustly estimating the covariance is equivalent to robustly estimating the mean of every (normalized) degree-two polynomial p. The above theorem shows how a weak estimate in high-dimensions can be used to obtain stronger estimates in one dimension, which ultimately we will use to improve the high-dimensional estimate as well. The above theorem is the *workhorse* in our proof.

Our algorithm itself is simple, however, its correctness is quite non-trivial. We define some threshold T. Given our corrupted set of samples from $\mathcal{N}(0, \Sigma)$, we use our corrupted data set to estimate the mean of $p(X)$ conditioned on the event that $|p(X)| \leq T$. Then, to estimate the contribution of the mean from points X so that $|p(X)| > T$, we estimate this by $\mathbf{E}_{X \sim \mathcal{N}(0,I)}[p(X) 1_{|p(X)|>T}]$. In other words, we are replacing the contribution of the true tail by an estimate of the contribution of $p(X)$ when $X \sim \mathcal{N}(0, I)$ on this tail. The formal pseudocode is given in Algorithm 6.

Intuitively, this algorithm works because of two reasons. First, it is not hard to show that the influence of points $p(X)$ within the threshold T on the estimator are bounded by at most T. Hence, the adversary cannot add corrupted points within this threshold and cause our estimator to deviate too much. Secondly, because we know that $\|\Sigma - I\|_F$ is small, by carefully utilizing smoothness properties of sums of chi-squared random variables, we are able to show that our estimate for the contribution of the tail is not too large. At a high level, this is because "most" of the distance between two chi-squared random variables must remain close to the means, so the difference in the tails is much smaller. Proving that this holds in a formal sense is the majority of the technical work of this section.

Proof. We know the distribution of $p(X')$ for $X' \sim N(0, I)$ explicitly and wish to use this to get a better estimate for the mean of $p(X)$ for $X \sim N(0, \Sigma)$ than might be given by the mean of the ε-corrupted set of samples.

Algorithm 6 Approximating $\mathbf{E}[p(X)]$ for $X \sim N(0, \Sigma)$ with corrupted samples.

1: **function** LEARNMEANCHISQUARED$(X_1, \ldots, X_n, p(x), \varepsilon, \tau)$
2: Let $T = O(\log C)$.
3: Let $f(x) = \begin{cases} x - T, & \text{for } x \geq T \\ 0, & \text{for } |x| \leq T \\ x + T, & \text{for } x \leq -T \end{cases}$.
4: Compute $\alpha = \sum_{i=1}^n (p(X_i) - f(p(X_i)))/n$.
5: Simulate $m = O((\ln \tau)/\varepsilon^2)$ samples $X_1', \ldots, X_m'$ from $X' \sim N(0, I)$.
6: Return $\widehat{\mu} = \alpha + \sum_{i=1}^m f(p(X_i'))/n$.
7: **end function**

In the full version, we show that (ε, δ)-goodness implies that $|\mathbf{E}[Z - f(Z)] - \alpha| \leq 2T\varepsilon$.

Since $p \in \mathcal{P}_2$, we have $\mathbf{E}[p(X')] = 1$ for $X' \sim N(0, I)$. Thus, we have $\mathbf{Var}[f(p(X'))] \leq \mathbf{E}[f(p(X'))^2] \leq \mathbf{E}[p(X')^2] = 1$. It follows by standard concentration results that the empirical after taking $m = O(\ln(1 - \tau)/\varepsilon^2)$ samples has $|\sum_{i=1}^m f(p(X_i)')/n - \mathbf{E}[p(X')]| \leq \varepsilon$ with probability $1 - \tau$. When this holds, we have

$$\left| \widehat{\mu} - \mathop{\mathbf{E}}_{X \sim \mathcal{N}(0,\Sigma)}[p(X)] \right| \leq (2T + 1)\varepsilon + \left| \mathop{\mathbf{E}}_{X \sim \mathcal{N}(0,I)}[f(p(X))] - \mathop{\mathbf{E}}_{X \sim \mathcal{N}(0,\Sigma)}[f(p(X))] \right| \; .$$

To prove the correctness of the algorithm it remains to show that:

LEMMA 5.3. *For any constant $C > 0$, for $T = O(\log C)$, we obtain*

$$\left| \mathop{\mathbf{E}}_{X \sim \mathcal{N}(0,I)}[f(p(X))] - \mathop{\mathbf{E}}_{X' \sim \mathcal{N}(0,\Sigma)}[f(p(X'))] \right| \leq \frac{\|\Sigma - I\|_F}{C} \; .$$

The proof of this is quite involved and we defer it to the full version. At a high level, we observe that $p(X)$ and $p(X')$ can be viewed as two nonuniform chi-squared random variables, with weights nearly the same. By carefully bounding the change in the PDF value between $p(X)$ and $p(X')$ by changing the weights, we show that almost all the probability mass due to the difference must occur near the mean of the distribution. This in turn allows us to show that the means cannot differ too much.

5.3 Working in a Low-Dimensional Space of Degree-Two Polynomials We now show that via similar techniques as before, we can patch our estimates together to find a matrix which agrees with the ground truth on all degree-two polynomials in a fixed subspace of low dimension. Formally, we show:

THEOREM 5.2. *Fix $\varepsilon, \tau > 0$. Let Σ be so that $\|\Sigma - I\|_F \leq O(\varepsilon \log 1/\varepsilon)$. Let G_0 be an (ε, δ)-good set of samples from $\mathcal{N}(0, \Sigma)$, and let $S = \{X_1, \ldots, X_n\}$ be so that $\Delta(S, G_0) \leq \varepsilon$. Let W_1 be a subspace of degree-2 polynomials, and let W_2 be an orthogonal subspace of degree-2 polynomials, so that we have a $\widehat{\Sigma}$ so that $\left|\mathbf{E}_{X \sim \mathcal{N}(0,\Sigma)}[p(X)] - \mathbf{E}_{X \sim \mathcal{N}(0,\widehat{\Sigma})}[p(X)]\right| \leq \xi$ for all $p \in W_2$. Then there is an algorithm* LEARNMEANPOLYLOWD *which given $\varepsilon, S, W_1, W_2, \widehat{\Sigma}$ runs in time $\mathrm{poly}(d, |S|, 2^{O(\dim(W_1))}, \log 1/\tau)$, and returns a Σ' so that*

$$\left|\mathop{\mathbf{E}}_{\mathcal{N}(0,\Sigma)}[p(X)] - \mathop{\mathbf{E}}_{\mathcal{N}(0,\Sigma')}[p(X)]\right| \leq 4\left(\|\Sigma - I\|_F/C + O(\log(C)\varepsilon) + \xi\right),$$

for all $p \in \mathrm{span}(W_1 \cup W_2) \cap \mathcal{P}_2$, with probability $1 - \tau$.

In particular, this implies:

COROLLARY 5.2. *Fix $\varepsilon, \tau > 0$. Let Σ be so that $\|\Sigma - I\|_F \leq O(\varepsilon \log 1/\varepsilon)$. Let G_0 be an (ε, δ)-good set of samples from $\mathcal{N}(0, \Sigma)$. Let $S = \{X_1, \ldots, X_n\}$ be so that $\Delta(S, G_0) \leq \varepsilon$. Let V be a subspace of $\mathbb{R}^d$. Then there is an algorithm* LEARNCOVLOWDIM *which given $S, \varepsilon, \xi, \tau, V$ runs in time $\mathrm{poly}(|S|, 2^{O(\dim(V)^2)}, \log 1/\tau)$ and returns a Σ' so that*

$$\|\Pi_V(\Sigma - \Sigma')\Pi_V\|_2 \leq 4\left(\|\Sigma - I\|_F/C + O(\log(C)\varepsilon)\right),$$

with probability $1 - \tau$.

Proof. Observe that the dimension of the space of degree-2 polynomials W in V is $O(\dim(V)^2)$. Run the algorithm in Theorem 5.2 with the same parameters as before, with $W_1 = W$ and $W_2 = \emptyset$ (so that we may take $\xi = 0$), and then the guarantee of that algorithm, along with Lemma 6.1, gives our desired guarantee.

We now describe the algorithm for Theorem 5.2. Essentially, we do the same thing as we did for low-dimensional learning in the unknown mean case: we take a constant net over $V \cap \mathcal{P}_2$, l earn the mean over every polynomial in the net, and then find a Σ' which is close in each direction to the learned mean. Since we will not attempt to optimize the constant factor here, will will use a naive LP-based approach to find a point which is close to optimal. The formal pseudocode is given in Algorithm 7.

Algorithm 7 Filter if there are many large eigenvalues of the covariance

function LOWDIMCOVLEARNING$(S, \varepsilon, \xi, \tau, W_1, W_2)$
  Generate a 1/2-cover $\mathcal{C}$ for $W_1 \cap \mathcal{P}_2$.
  Let $\tau' = 2^{-|\mathcal{C}|}\tau$
  for $p \in \mathcal{C}$ **do**
    Compute $m_p =$ LEARNMEANCHISQUARED$(S, p, \varepsilon, \tau')$.
    Generate a linear constraint $c_p(\Sigma')$: $\left|\mathbf{E}_{\mathcal{N}(0,\Sigma')}[p(X)] - m_p\right| \leq \|\Sigma - I\|_F/C + O(\log(C))\varepsilon$.
  end for
  Generate the convex constraint that $\left|\mathbf{E}_{\mathcal{N}(0,\Sigma')}[p(X)] - \mathbf{E}_{\mathcal{N}(0,\Sigma')}[p(X)]\right| \leq \xi$ for all $p \in W_2$.
  Using a convex program, **return** any matrix Σ' which obeys $c_p(\Sigma')$ for all $p \in \mathcal{C}$.
end function

Observe that every constraint for each polynomial in W_1 is indeed linear in Σ', by Lemma 6.1. Moreover, the constraint for W_2 has an explicit separation oracle, since it induces a norm, and for any $p \in W_2$, we may explicitly compute $\mathbf{E}_{\mathcal{N}(0,\Sigma')}[p(X)] - \mathbf{E}_{\mathcal{N}(0,\Sigma')}[p(X)]$. Thus, we may use separating hyperplane techniques to solve this convex program in the claimed running time.

Proof. [Proof of Theorem 5.2] Let us condition on the event that LEARNMEANCHSQUARED succeeds for each $p \in \mathcal{C}$. By a union bound, this occurs with probability at least $1 - \tau$. Thus, in each $p \in \mathcal{C}$, we have that $|m_p - \mathbf{E}_{X \sim \mathcal{N}(0,\Sigma)}[p(X)]| \leq \beta$, where $\beta = \|\Sigma - I\|_F/C + O(\log(C))\varepsilon$. Let Σ' be the matrix we find. By the triangle inequality, we then have that for every $p \in \mathcal{C}$, that $|\mathbf{E}_{\mathcal{N}(0,\Sigma')}[p(X)] - \mathbf{E}_{\mathcal{N}(0,\Sigma)}[p(X)]| \leq 2\beta$. Hence, by the usual net arguments, we know that for every $p \in V \cap \mathcal{P}_2$,

$$|\mathop{\mathbf{E}}_{\mathcal{N}(0,\Sigma')}[p(X)] - \mathop{\mathbf{E}}_{\mathcal{N}(0,\Sigma)}[p(X)]| \leq 4\beta.$$

Moreover, by triangle inequality, for every $p \in W_2$, we

have $\left|\mathbf{E}_{\mathcal{N}(0,\Sigma')}[p(X)] - \mathbf{E}_{\mathcal{N}(0,\Sigma')}[p(X)]\right| \leq 2\xi$. The result then follows from the Pythagorean theorem.

6 Robustly Learning the Covariance in High-Dimensions

In this section, we show how to robustly estimate the covariance of a mean-zero Gaussian in high-dimensions up to error $O(\varepsilon)$. We use our low-dimensional learning algorithm from the previous section as a crucial subroutine in what follows.

Our main algorithmic contribution is as follows:

THEOREM 6.1. *Fix $\varepsilon, \delta > 0$, and let $S_0 = (G_0, E_0)$ be an ε-corrupted set of samples of size n from $\mathcal{N}(0, \Sigma)$, where $\|\Sigma - I\|_F \leq \xi$ where $\xi = O(\varepsilon \log 1/\varepsilon)$, and where $n = \mathrm{poly}(d, 1/\varepsilon, \log 1/\delta)$. Suppose that G_0 is (ε, δ)-good with respect to $\mathcal{N}(0, \Sigma)$. Let $S \subseteq S_0$ be a set so that $\Delta(S, G_0) \leq \varepsilon$. Then, there exists an algorithm* IMPROVECOV *that given S, ξ, ε, fails with probability at most $\mathrm{poly}(\varepsilon, 1/d, \delta)$, and otherwise outputs one of two possible outcomes:*

(i) A matrix $\widehat{\Sigma}$, so that $\|\widehat{\Sigma} - \Sigma\|_F \leq \|\Sigma - I\|_F/2$.

(ii) A set $S' \subset S$ so that $\Delta(S', G_0) < \Delta(S, G_0)$.

Moreover, IMPROVECOV *runs in time $\mathrm{poly}(d, (1/\varepsilon)^{O(\log^4 1/\varepsilon)}, \log 1/\delta)$.*

By first applying the algorithm in [DKK+16] to produce an initial estimate for Σ, and then iterating the above algorithm polynomially many times, this immediately yields:

COROLLARY 6.1. *Fix $\varepsilon, \delta > 0$, and let G_0 be a set of i.i.d. samples from $\mathcal{N}(0, \Sigma)$, where $n = \mathrm{poly}(d, 1/\varepsilon, \log 1/\delta)$. Let S be so that $\Delta(S, G_0) \leq \varepsilon$. There is an universal constant C and an algorithm which outputs a $\widehat{\Sigma}$ so that with probability $1 - \delta$, we have $\|\widehat{\Sigma}^{-1/2} \Sigma \widehat{\Sigma}^{-1/2} - I\|_F \leq C\varepsilon$. In particular, this implies that $d_{\mathrm{TV}}\left(\mathcal{N}(0, \Sigma), \mathcal{N}(0, \widehat{\Sigma})\right) \leq 2C\varepsilon$.*

6.1 Technical Overview Our strategy for obtaining a high-dimensional estimate for the covariance based on solving low-dimensional subproblems will be substantially more challenging than it was for the unknown mean case. The natural approach is to take the $\mathrm{poly}\log(1/\varepsilon)$-dimensional subspace of degree-2 polynomials of largest empirical variance and construct a filter. However, this fails because, unlike in the mean case, we do not know the variance of these degree-2 polynomials to small error. For the unknown mean case, because we assumed that we knew the covariance was the identity (or spectrally close to the identity), this was not an issue. Now, the variance of our polynomials depends on the (unknown) covariance of the true Gaussian, which may be more than $O(\varepsilon)$-far from our current estimate. Indeed, it is not difficult to come up with counterexamples where there are many large eigenvalues of the empirical covariance matrix, but no filter can make progress.

We overcome this hurdle in several steps. First, in Section 6.3, we show how to find a filter if there are many medium-sized eigenvalues of the empirical covariance matrix. This will proceed roughly in the same way that the filter for the unknown mean does. If no filter is created, then we know there are at most logarithmically large eigenvalues of the empirical covariance. In the subspace $V \subseteq \mathbb{R}^d$ spanned by their eigenvectors, we can then learn the covariance to high accuracy using our low-dimensional estimator.

Then, in Section 6.3, we show that if we restrict to the orthogonal subspace, i.e., the subspace where the empirical covariance matrix does not have large eigenvalues, we can indeed either produce a filter or improve our estimate of the covariance restricted to this subspace using our low-dimensional estimator. While the blueprint is similar to the filter for the unknown mean, the techniques are much more involved and subtle.

Supposing we have not yet created a filter, we have now estimated the covariance on a polylogarithmic dimensional subspace V, and on $V^\perp$. This does not in general imply that we have learned the covariance in Frobenius norm. In block form, if we write

$$\Sigma = \begin{bmatrix} \Sigma_V & A^T \\ A & \Sigma_{V^\perp} \end{bmatrix},$$

where here $\mathbb{R}^d$ is written as $V \oplus V^\perp$, this implies we have learned Σ_V and $\Sigma_{V^\perp}$ to high accuracy. Thus, it remains to estimate the cross term A.

In Section 6.4, we show, given a polylogarithmically sized subspace V, and a good estimate of the covariance matrix on V and $V^\perp$, how to fill in the entire covariance matrix. Roughly, we do this by randomly fixing directions in V, and performing rejection sampling based on the correlation in the direction in V, and showing that the problem reduces to one of robustly learning the mean of a Gaussian, which (conveniently) we have already solved. These steps together yield our overall algorithm IMPROVECOV. Finally, in Section 6.6 we explain why there is a natural barrier that makes reducing the running time from quasi-polynomial to polynomial (in $1/\varepsilon$) difficult.

6.2 Additional Preliminaries Here we give some additional preliminaries we will require in this

Section.

6.2.1 The Agnostic Tournament We also require the following classical result, which allows us to do agnostic hypothesis selection with corrupted samples (see e.g., [DL01, DDS12, DK14, DDS15]).

THEOREM 6.2. *Fix $\varepsilon, \delta > 0$. Let $D_1, \ldots, D_k, D$ be a set of distributions where $\min_i d_{\mathrm{TV}}(D_i, D) = \gamma$. Set $n = \Omega\left(\frac{\log k + \log 1/\delta}{\varepsilon^2}\right)$. There is an algorithm* TOURNAMENT *which given oracles for evaluating the pdfs of $D_1, \ldots, D_k$ along with n independent samples $X_1, \ldots X_n$ from D, outputs a D_i so that $d_{\mathrm{TV}}(D_i, D) \leq 3\gamma + \varepsilon$ with probability $1 - \delta$. Moreover, the running time and number of oracle calls needed is at most $O(n^2/\varepsilon^2)$.*

REMARK 1. *As a simple corollary of the agnostic tournament, observe that this allows us to do agnostic learning without knowing the precise error rate ε. Throughout the paper, we assume the algorithm knows ε. However, if the algorithm is not given this information, and instead given an η and asked to return something with error at most $O(\varepsilon + \eta)$, we may simply grid over $\{\eta, (1+\gamma)\eta, (1+\gamma)^2\eta, \ldots, 1\}$ (here γ is some arbitrary constant that governs a tradeoff between runtime and accuracy), run our algorithm with ε set to each element in this set, and perform hypothesis selection via* TOURNAMENT. *Then it is not hard to see that we are guaranteed to output something which has error at most $O(\varepsilon + (1+\gamma)\eta)$.*

6.2.2 The Fourth Moment Tensor of a Gaussian As in [DKK+16], it will be crucial for us to understand the behavior of the fourth moment tensor of a Gaussian. Let $\otimes$ denote the Kronecker product on matrices. We will make crucial use of the following definition:

DEFINITION 6. *For any matrix $M \in \mathbb{R}^{d \times d}$, let $M^\flat \in \mathbb{R}^{d^2}$ denote its canonical flattening into a vector in $\mathbb{R}^{d^2}$, and for any vector $v \in \mathbb{R}^{d^2}$, let $v^\sharp$ denote the unique matrix $M \in \mathbb{R}^{d \times d}$ so that $M^\flat = v$.*

We will also require the following definition:

DEFINITION 7. $\mathcal{S}_{\mathrm{sym}} = \{M^\flat \in \mathbb{R}^{d^2} : M \text{ symmetric}\}$.

The following result was proven in [DKK+16]:

THEOREM 6.3. (THEOREM 4.15 IN [DKK+16]) *Let $X \sim \mathcal{N}(0, \Sigma)$. Let M be the $d^2 \times d^2$ matrix given by $M = \mathbf{E}[(X \otimes X)(X \otimes X)^T]$. Then, as an operator on $\mathcal{S}_{\mathrm{sym}}$, we have $M = 2\Sigma^{\otimes 2} + \left(\Sigma^\flat\right)\left(\Sigma^\flat\right)^T$.*

6.2.3 Polynomials in Gaussian Space Here we review some basic facts about polynomials under Gaussian measure, which will be crucial for our algorithm for learning Gaussians with unknown covariance. We equip the set of polynomials over $\mathbb{R}^d$ with the Gaussian inner product, defined by $\langle f, g \rangle = \mathbf{E}_{X \sim \mathcal{N}(0,I)}[f(X)g(X)]$, and we let $\|f\|_2^2 = \langle f, f \rangle$.

For any symmetric M with $\|M\|_F = 1$, define the degree-2 polynomial $p(x) = \frac{1}{\sqrt{2}}(x^T M x - \mathrm{tr}(M))$. We call p the *polynomial associated to* M. Observe that p is even (i.e., has no degree-1 terms). We will use the following properties of such polynomials:

LEMMA 6.1. *Let M be symmetric, so that $\|M\|_F = 1$. Let p be its associated polynomial. Then, we have:*

(i) $\mathbf{E}_{X \sim \mathcal{N}(0,I)}[p(X)] = 0$.

(ii) More generally, for any positive definite matrix Σ, we have $\mathbf{E}_{X \sim \mathcal{N}(0,\Sigma)}[p(X)] = \langle M, \Sigma - I \rangle$.

(iii) $\mathbf{Var}_{X \sim \mathcal{N}(0,I)}[p(X)] = \mathbf{E}_{X \sim \mathcal{N}(0,I)}[p^2(X)] = \langle p, p \rangle = 1$.

(iv) More generally, for any positive definite matrix Σ, we have $\mathbf{E}_{X \sim \mathcal{N}(0,\Sigma)}[p^2(X)] = {M^\flat}^T \Sigma^{\otimes 2} M^\flat + \frac{1}{2}\left(\langle \Sigma - I, M^\flat \rangle\right)^2$.

The proof of this is standard and we defer it to the appendix. Observe that Lemma 6.1(iv) implies that if we take the top eigenvector of the $d^2 \times d^2$ matrix

$$\Sigma^{\otimes 2} + \frac{1}{2}\left(M^\flat\right)\left(M^\flat\right)^\top$$

on the linear subspace $\mathcal{S}_{\mathrm{sym}}$, then the associated polynomial maximizes $\mathbf{E}_{X \sim \mathcal{N}(0,\Sigma)}[p^2(X)]$, and so we can find these polynomials efficiently. More generally, if we take any linear subspace of degree two polynomials with associated matrix subspace V', so that $V' \subseteq \mathcal{S}_{\mathrm{sym}}$, then the top eigenvector of the same matrix restricted to V' allows us to find the polynomial in this subspace which maximizes $\mathbf{E}_{X \sim \mathcal{N}(0,\Sigma)}[p^2(X)]$ efficiently.

We have the following tail bound for degree-2 polynomials in Gaussian space: We will use $\Pi_V(x)$ and $\Pi_V(S)$ to denote projection to a subspace V, of a point x and a set of points S, respectively. We will also need the following hypercontractivity theorem for low-degree polynomials in Gaussian space, see i.e., [O'D14]:

THEOREM 6.4. *Let $p : \mathbb{R}^d \to \mathbb{R}$ be a degree m polynomial, and let $q \geq 2$ be even. Then $\mathbf{E}_{X \sim \mathcal{N}(0,I)}[p(X)^q]^{1/q} \leq (\sqrt{q-1})^m \|p\|_2$.*

We need the following definition:

DEFINITION 8. *Let $\mathcal{P}_k$ denote the set of even degree-k polynomials over d variables satisfying $\mathbf{Var}_{X\sim\mathcal{N}(0,1)}[p(X)] = 1$. Moreover, for any subspace $W \subseteq \mathbb{R}^d$, let $\mathcal{P}_k(W)$ denote the set of even polynomials over d variables which only depend on the coordinates in W.*

Then by the arguments above, we have that for any two matrices $\Sigma, \widehat{\Sigma}$, $\|\Sigma - \widehat{\Sigma}\|_F = \sup_{p\in\mathcal{P}_2}\left(\mathbf{E}_{X\sim\mathcal{N}(0,\Sigma)}[p(X)] - \mathbf{E}_{X\sim\mathcal{N}(0,\widehat{\Sigma})}[p(X)]\right)$. In particular, by Lemma 2.2, this implies that when $\|\Sigma - I\|_2$ is small, then learning a Gaussian with unknown covariance in total variation distance is equivalent to learning the expectation of every even degree-2 polynomial.

Theorem 6.4 implies the following concentration for degree-4 (more generally, low-degree) polynomials of Gaussians:

COROLLARY 6.2. *Let p be a degree-4 polynomial. Then there is some $A, C \geq 0$ so that for all $t \geq C$, we have $\Pr_{\mathcal{N}(0,I)}[|p(X) - \mathbf{E}_{\mathcal{N}(0,I)}[p(X)]| \geq t\|p\|_2] \leq \exp\left(-At^{1/2}\right)$.*

Proof. Hypercontractivity in particular implies the following moment bound: for all $q \geq 2$ even, we have $\mathbf{E}_{\mathcal{N}(0,I)}[(p(X) - \mathbf{E}_{\mathcal{N}(0,I)}[p(X)])^q] \leq (q-1)^{qm/2}\|p(X) - \mathbf{E}_{\mathcal{N}(0,I)}[p(X)]\|_2^q$. By a typical moment argument, and optimizing the choice of q, this gives the desired bound.

Hermite polynomials Hermite polynomials are what arise by Gram-Schmidt orthogonalization applied with respect to this inner product. For a vector of non-negative integers $a = (a_1, \ldots, a_d)$, we let $H_a(x) : \mathbb{R}^d \to \mathbb{R}$ be the Hermite polynomial associated with multi-index a. It is well-known that the degree of H_a is $|a| = \sum_{i=1}^d a_i$, and moreover, $\langle H_a, H_b\rangle = \delta_{a,b}$. In particular, for any $r \geq 1$, the Hermite polynomials of degree at most r form an orthonormal basis with respect to the Gaussian inner product for all polynomials with degree at most r.

Therefore, given any polynomial $p : \mathbb{R}^d \to \mathbb{R}$ with degree r, we may write it uniquely as $p(x) = \sum_{|a|\leq r} c_a(p)H_a(x)$, where $c_a(p) = \langle p, H_a\rangle$. We define the kth harmonic component of p to be $p^{[k]}(x) = \sum_{|a|=k} c_a(p)H_a(x)$, and we say p is *harmonic* of degree k if it equals its kth part.

6.3 Working with Many Large Eigenvalues of the Second and Fourth Moment

As in the unknown mean case, we will need a filter to detect if there are many directions of the empirical covariance which have too large an eigenvalue. Formally, we need:

THEOREM 6.5. *Fix $\varepsilon, \delta > 0$. Assume $\|\Sigma - I\|_F \leq \xi$, where $\xi = O(\varepsilon \log 1/\varepsilon)$. Suppose that G_0 is (ε,δ)-good with respect to $\mathcal{N}(0,\Sigma)$. Let S be a set so that $\Delta(S, G_0) \leq \varepsilon$. Let $\widehat{\Sigma} = \mathbf{E}_S[XX^T]$. Then there is an algorithm* FILTERCOVMANYDEG2EIG *and a universal constant C such that the following guarantee holds:*

1. *If $\widehat{\Sigma} - I$ has more than $O(\log 1/\varepsilon)$ eigenvalues larger than $C\xi$, then the algorithm outputs a S' so that $\Delta(S', G_0) < \Delta(S, G_0)$.*
2. *Otherwise, the algorithm outputs "OK", and outputs an orthonormal basis $v_1, \ldots, v_k$ for the subspace V of vectors spanned by all eigenvectors of $\widehat{\Sigma} - I$ with eigenvalue larger than $C\xi$.*

The filter developed here is almost identical to the one developed for unknown mean. Thus, for conciseness we describe and prove the theorem in the full version.

We will also need a subroutine to enforce the condition that not only does the fourth moment tensor have spectral norm which is at most $O(\varepsilon \log^2 1/\varepsilon)$ (restricted to a certain subspace of polynomials), but there can only be at most $O(\mathrm{poly}\log 1/\varepsilon)$ directions in which the eigenvalue is large. However, the techniques here are a bit more complicated, for a number of reasons. Intuitively, the main complication comes from the fact that we do not know what the fourth moment tensor looks like, whereas in the unknown mean case, we knew that the covariance was the identity by assumption. Our main result in this subsection is the following subroutine:

THEOREM 6.6. *Fix $\varepsilon, \delta > 0$. Assume $\|\Sigma - I\|_F \leq \xi$, where $\xi = O(\varepsilon \log 1/\varepsilon)$. Let C be the universal constant in* FILTERCOVMANYDEG2EIG. *Let $W \subseteq \mathbb{R}^d$ be a subspace, so that for all $v \in W$ with $\|v\|_2 = 1$, we have $v^T\mathbf{E}_S[XX^T]v \leq 1 + C\xi$. Suppose that G_0 is (ε,δ)-good with respect to $\mathcal{N}(0,\Sigma)$. Let S be a set so that $\Delta(S, G_0) \leq \varepsilon$. Let $k = O(\log^4 1/\varepsilon)$. Then there is an algorithm* FILTERCOVMANYDEG4EIG *and universal constants C_1, C_2 such that the following guarantee holds:*

1. *If there exist $p_1, \ldots, p_k \in \mathcal{P}_2(W)$ so that $\langle p_j, p_\ell\rangle = \delta_{j\ell}$ for all j, ℓ, and so that $\mathbf{E}_S[p_j^2(Y)] - 1 \geq C_1\varepsilon$ for all j, then the algorithm outputs an S' so that $\Delta(S', G_0) < \Delta(S, G_0)$.*
2. *Otherwise, the algorithm outputs "OK", and outputs an orthonormal basis $p_1, \ldots, p_{k'}$ for a subspace V of degree-2 polynomials in $\mathcal{P}_2(W)$ with $k' \leq k$ so that for all $p \in V^\perp \cap \mathcal{P}_2$, we have $\mathbf{E}_S[p^2(X)] - 1 \leq C_2\varepsilon$.*

Moreover, FILTERCOVMANYEIG *runs in time* $\mathrm{poly}(d, 1/\varepsilon, \log 1/\delta)$.

Roughly, we will show that if there are many polynomials with large empirical variance, this implies that there is a degree-four polynomial whose value is much larger than it could be if w were the set of uniform weights over the uncorrupted points. Moreover, we can explicitly construct this polynomial, and it has a certain low-rank structure which allows us to use the concentration bounds we have previously derived.

Algorithm 8 Filter if there are many large eigenvalues of the fourth moment tensor

function FilterCovManyEig($S, \varepsilon, \xi, \delta, W$)
  Let $\widehat{\Sigma} = \mathbf{E}_S[XX^T]$
  Let C_1, C_2, C_3 be some universal constants sufficiently large
  Let A be the constant in Corollary 6.2
  Let B be the constant in Claim 20 of the full version
  Let $m = 0$
  Let $k = O(\log^4 1/\varepsilon)$
  while there exists $p \in \mathcal{P}_2(W)$ so that $p \in V^\perp$ and $\mathbf{E}_S[p^2(X)] - 1 > C_1\xi$ **do**
    Let $V_{m+1} = \mathrm{span}(V_m \cup p)$
    Let $m \leftarrow m + 1$
  end while
  Let $p_1, \ldots, p_m$ be an orthonormal basis for V_m
  if $m \geq k$ **then**
    Let $q_i = (p_i^2)^{[4]}$ be the 4th harmonic component of p_i^2
    Let $r_i = p_i^2 - q_i$ be the degree-2 component of p_i^2
    Let $Q(x) = \sum_{i=1}^k q_i$
    Find a T so that either:

- $T > C_3 d^2 \sqrt{k} \log(|S|)$ and $p(X) > T$ for at least one $x \in S'$, OR
- $T > 4A^2 C_2 B \sqrt{k} \log^2(1/\varepsilon)$ and $\Pr_{X \in_u S}[Q(X) > T] > \exp(-A(T/4B\sqrt{k})^{1/2}) + \varepsilon^2/(d\log(|S|/\delta))^2$.

    return the set $S' = \{X \in S : Q(X) \leq T\}$
  else
    return "OK", and output $p_1, \ldots, p_m$
  end if
end function

6.4 Stitching Together Two Subspaces This section is dedicated to giving an algorithm which allows us to fully reconstruct the covariance matrix given that we know it up to small error on a low-dimensional subspace V and on $W = V^\perp$.

Theorem 6.7. *Let $1 > \xi > \eta > \varepsilon > 0$, and let $\tau > 0$. Let Σ so that $\|\Sigma - I\|_F \leq \xi$. Suppose that $\mathbb{R}^d$ is written as $V \oplus W$ for orthogonal subspaces V and W with $\dim(V) = O(\log(1/\varepsilon))$. Suppose furthermore that*

$$\Sigma = \begin{bmatrix} \Sigma_V & A^T \\ A & \Sigma_W \end{bmatrix},$$

with $\|\Sigma_V - I_V\|_F, \|\Sigma_W - I_W\|_F = O(\eta)$. Let $S_0 = (G_0, E_0)$ be an ε-corrupted set of samples from $\mathcal{N}(0, \Sigma)$, and let $S \subseteq S_0$ with $\Delta(S, G) \leq O(\varepsilon)$ of size $\mathrm{poly}(d, 1/\eta, \log 1/\delta)$.

Then, there exists a universal constant C_5 and an algorithm Stitching *that given $V, W, \xi, \eta, \varepsilon, \tau$ and S runs in polynomial time and with probability at least $1 - \tau$ returns a matrix Σ_0 with $\|\Sigma_0 - \Sigma\|_F = C_5\eta + O(\xi^2)$.*

In the latter, we will show the algorithm works when $\tau = 2/3$. As usual the probability of success can be boosted by repeating it independently.[5] The basic idea of the proof is as follows. Since we already know good approximations to Σ_V and Σ_W, it suffices to find an approximation to A. In order to do this, we note that if we take a sample x from G conditioned on its projection to V being some vector v, we find that the distribution over W is a Gaussian with mean approximately Av. Running our algorithm for approximating the mean of a noisy-Gaussian, we can then compute the mapping $v \to Av$, which will allow us to compute A.

There are three main technical obstacles to this approach. The first is that we cannot condition on x_V taking a particular value, as we will likely see no samples from X with exactly that projection. Instead, what we will do is given samples from X we will reject them with probabilities depending on their projections to V in such a way to approximate the conditioning we require. The second obstacle is that the errors in X may well be concentrated around some particular projection to V. Therefore, some of these conditional distributions may have a much larger percentage of errors than ε. To circumvent this, we will show that by carefully choosing how we do our conditioning and by carefully picking the correct distribution over vectors v, that on average these errors are only $O(\varepsilon)$. Finally, we need to be able to reconstruct A from a collection of noisy approximations to Av. We show that this can be done by computing these approximations at a suitably large random set of v's, and finding the matrix A

[5] Observe the only randomness at this point is in the random choices made by the algorithm. Thus, one can just run this algorithm $O(\log 1/\delta)$ times to obtain $\Sigma_0^{(1)}, \ldots, \Sigma_0^{(\ell)}$ and find any $\Sigma_0^{(j)}$ which is $O(\eta + \xi^2)$ close to at least a $2/3 + o(1)$ fraction of the other outputs.

that minimizes the average ℓ_2 error between Av and its approximation.

Our algorithm is given in Algorithm 9, and the proof of Theorem 6.7 is deferred to the full version.

Algorithm 9 Stitching the two subspaces together

function STITCHING($V, W, \delta, \varepsilon, \tau, S$)
  Given a vector x, let x_V and x_W be the projections onto V and W, respectively.
  Let C be a sufficiently large constant (where C may depend on the constants in the big-O terms in the guarantee that $\dim(V) = O(\log(1/\varepsilon))$).
  Generate a set $V = \{v_1, \ldots, v_m\}$ of $(n/\varepsilon)^C$ independent random samples from $\mathcal{N}(0, 2I_V)$.
  for $v \in V$ **do**
    For each sample $x \in S$, add x_W to a new set T independently with probability
$$\exp(-\|x_V - v\|^2/2) .$$
    Treat T as a collection of independent samples from a noisy Gaussian with covariance matrix $I_W + O(\eta)$.
    Set a_v equal to 0 if T did not contain enough samples for our algorithm or if $\|\tilde{\mu}\|_2 > C\log(1/\varepsilon)$.
    for $\varepsilon \in \{1, 1/2, 1/4, 1/8, \ldots, \eta\}$ **do**
      Let $\widetilde{\mu}$ be the output of RECOVERMEANNOISY$(T, \varepsilon, (\varepsilon/n)^{2C}, o(1), O(\eta))$.
    end for
    Run TOURNAMENT with the output hypotheses.
    Set $a_v = \tilde{\mu}$, where $\widetilde{\mu}$ is the winning hypothesis.
  end for
  Use linear programming to find the $\dim(W) \times \dim(V)$-matrix B that minimizes the convex function $\mathbf{E}_{v \in_u S}[|a_v - Bv|]$.
  return
$$\Sigma_0 = \begin{bmatrix} I_V & 2B^T \\ 2B & I_W \end{bmatrix} .$$
end function

6.5 The Full High-Dimensional Algorithm We now show how to prove Theorem 6.1, given the pieces we have. We first show that given enough samples from $\mathcal{N}(0, \Sigma)$, the empirical data set without corruptions satisfies the regularity conditions in Section 2.2.2 with high probability. For clarity of exposition, the proof of this lemma is deferred to the full version.

LEMMA 6.2. *Fix $\eta, \delta > 0$. Let $X_1, \ldots, X_n$ be independent samples from $\mathcal{N}(\mu, I)$, where $n = \mathrm{poly}(d, 1/\eta, \log 1/\delta)$. Then, $S = \{X_1, \ldots, X_n\}$ is (η, δ)-good with respect to $\mathcal{N}(\mu, I)$ with probability at least $1 - \delta$.*

Finally, we require the following guarantee, which states that if there is a degree-2 polynomial whose expectation under S and the truth differs by a lot (equivalently, if the empirical covariance differs from the true covariance in Frobenius norm substantially), then it must also have very large variance under S.

LEMMA 6.3. *Fix $\varepsilon, \delta > 0$. Assume $\|\Sigma - I\|_F \le \xi$, where $\xi = O(\varepsilon \log 1/\varepsilon)$. Suppose that G_0 is (ε, δ)-good with respect to $\mathcal{N}(0, \Sigma)$, and let $S \subseteq S_0$ be a set so that $\Delta(S, G_0) \le \varepsilon$. There is some absolute constant C_5 so that if $p \in \mathcal{P}_2$ is a polynomial so that $\left|\mathbf{E}_S[p(X)] - \mathbf{E}_{\mathcal{N}(0,\Sigma)}[p(X)]\right| > C_5\sqrt{\xi\varepsilon}$, then $\mathbf{E}_S[p^2(X)] - 1 > C_1\xi$.*

We defer the proof of this lemma to the Appendix.

We are now ready to present the full algorithm as Algorithm 10.

Proof. [Proof of Theorem 6.1] Condition on the events that neither LEARNCOVLOWDIM nor STITCHING fail. This happens with probability at least $\mathrm{poly}(\varepsilon, 1/d, \delta)$. Observe that if we pass the "if" statement in Line 5, then by the guarantee of FILTERCOVMANYDEG2EIG this is indeed an S' satisfying the desired properties. Otherwise, by the guarantees of FILTERCOVMANYDEG2EIG, we have that W satisfies the conditions needed by FILTERCOVMANYDEG4EIG. Hence, if we pass the "if" statement in Line 11, then the guarantee of FILTERCOVMANYDEG4EIG this is indeed a S' satisfying the desired properties. Otherwise, by Lemma 6.3, we know that for all polynomials $p \in \mathcal{P}_2$ over W orthogonal to U_1, we have $|\mathbf{E}_{\mathcal{N}(0,\Sigma)} - \mathbf{E}_S[XX^T]| \le C_5\sqrt{\xi\varepsilon}$. Thus, Σ_W satisfies the conditions needed by STITCHING.

By Corollary 5.2, we know that Σ_V satisfies the conditions for STITCHING, and so the correctness of the algorithm follows from Theorem 6.7.

6.6 The Barrier at Quasi-Polynomial Here we explain why improving the running time from quasi-polynomial to polynomial in $1/\varepsilon$ will likely be rather difficult. Recall that our strategy is to project the problem onto lower dimensional subproblems and stitch together the answer. We need the dimension of the subspace to be large enough that we can find a polynomial Q that is itself the sum of squares of k orthogonal degree two polynomials p_i so that the value of Q on the corrupted points is considerably larger than the value on the uncorrupted points.

Algorithm 10 Filter if there are many large eigenvalues of the covariance

function ImproveCov($S, \xi, \varepsilon, \delta$)
  Let C be the universal constant in FilterCovManyDeg2Eig
  Let $\tau = \mathrm{poly}(\varepsilon, 1/d, \delta)$.
  Run FilterCovManyDeg2Eig(S, ε, ξ)
  if FilterCovManyDeg2Eig outputs S' **then**
    return S'
  else
    Let V be the subspace returned by FilterCovManyDeg2Eig
    Let $W = V^{\perp}$.
    Run FilterCovManyDeg4Eig($S, \varepsilon, \xi, \delta, W$)
    if FilterCovManyDeg4Eig outputs S' **then**
      return S'
    else
      Let U_1 be the subspace of degree 2 polynomials over W it returns
      Let U_2 be the perpendicular subspace of degree 2 polynomials over W
      Let $\widehat{\Sigma} = \mathbf{E}_S[XX^T]$
      Let Σ_V = LearnCovLowDim($S, \varepsilon, \xi, \tau, V$)
      Let Σ_W = LearnMeanPolyLowD($S, \varepsilon, \xi, \tau, U_1, U_2, \widehat{\Sigma}$)
      Take $\mathrm{poly}(n, 1/\varepsilon)$ fresh ε-corrupted samples S'
      return Stitching($V, W, \Sigma_V, \Sigma_W, \xi, \varepsilon, S'$)
    end if
  end if
end function

More precisely, if we let $S = (G, E)$ denote our corrupted set of samples then we want $\mathbf{E}_E[Q(X)]$ to be larger than $Q(X)$ for all but a $\mathrm{poly}(\varepsilon)$ fraction of $X \in G$. We then remove all points $X \in S$ with large $Q(X)$ and by the properties of Q we are guaranteed that we throw out mostly corrupted points. It turns out that the most aggressive we could be is removing points where $Q(X)$ is more than $\sqrt{k}$ standard deviations away from its expectation under the true Gaussian. But since Q is a degree-four polynomial and we want $Q(X)$ to be smaller than our cutoff for all but a $\mathrm{poly}(\varepsilon)$ fraction of $X \in G$, we are forced to choose $\sqrt{k} = \Omega(\log 1/\varepsilon)$, which means that we need to reduce to $k = \Omega(\log^2 1/\varepsilon)$ dimensional subproblems. Thus, if we solve low-dimensional subproblems in time exponential in the dimension, we naturally arrive at a quasi-polynomial running time. It seems that any approach for reducing the running time to polynomial would require fundamentally new ideas.

7 The General Algorithm

We now have all the tools to robustly learn the mean and covariance of an arbitrary high-dimensional Gaussian. We first show how to reduce the problem of robustly learning the covariance of $\mathcal{N}(\mu, \Sigma)$ to learning the covariance of $\mathcal{N}(0, \Sigma)$, by at most doubling error, a trick previously used in [DKK+16] and [LRV16]. Given an ε-corrupted set of samples $X_1, \ldots, X_{2n}$ of size $2n$ from $\mathcal{N}(\mu, \Sigma)$, we may let $Y_i = (X_i - X_{n+i})/\sqrt{2}$. Then we see that if X_i and X_{n+i} are uncorrupted, then $Y_i \sim \mathcal{N}(0, \Sigma)$. Moreover, at most $2\varepsilon n$ of the Y_i can be corrupted, since there are at most $2\varepsilon n$ corrupted X_i. Therefore, by doubling the error rate, we may assume that $\mu = 0$. We may then apply the algorithm in Corollary 6.1 to obtain a $\widehat{\Sigma}$ so that with high probability, we have $\|\widehat{\Sigma}^{-1/2}\Sigma\widehat{\Sigma}^{-1/2} - I\|_F \le O(\varepsilon)$ with polynomially many samples, and in $\mathrm{poly}(d, (1/\varepsilon)^{O(\log^4 1/\varepsilon)})$ time.

We may then take an additional set of ε-corrupted samples $\{X_i', \ldots, X_n'\}$, and let $Y_i' = \widehat{\Sigma}^{-1/2} X_i'$. Then, by our guarantee on $\widehat{\Sigma}$, we have that if X_i' is uncorrupted, then $Y_i' \sim \mathcal{N}(0, \widetilde{\Sigma})$ where $\|\widetilde{\Sigma} - I\|_F \le O(\varepsilon)$. We then run RecoverMeanNoisy with the Y_i' to obtain a $\widehat{\mu}$ so that $\|\widehat{\mu} - \widehat{\Sigma}^{-1/2}\mu\|_2 \le O(\varepsilon)$. This guarantees that $d_{\mathrm{TV}}(\mathcal{N}(\widehat{\mu}, \widetilde{\Sigma}), \mathcal{N}(\widehat{\Sigma}^{-1/2}\mu, \widehat{\Sigma})) \le O(\varepsilon)$, which in turn implies that $d_{\mathrm{TV}}(\mathcal{N}(\widehat{\mu}, \widehat{\Sigma}), \mathcal{N}(\mu, \Sigma)) \le O(\varepsilon)$, as claimed.

Therefore, we have shown:

Theorem 7.1. *Fix $\varepsilon, \delta > 0$. Given an ε-corrupted set of samples S from $\mathcal{N}(\mu, \Sigma)$, where $n = \mathrm{poly}(d, 1/\varepsilon, \log 1/\delta)$, there is an algorithm* RecoverGaussian *which takes as input S, ε, δ, and outputs a $\widehat{\mu}, \widehat{\Sigma}$ so that*

$$d_{\mathrm{TV}}(\mathcal{N}(\mu, \Sigma), \mathcal{N}(\widehat{\mu}, \widehat{\Sigma})) \le O(\varepsilon) .$$

Moreover, the algorithm runs in time $\mathrm{poly}(d, (1/\varepsilon)^{O(\log^4 1/\varepsilon)}, \log 1/\delta)$.

References

[BDLS17] S. Balakrishnan, S. S. Du, J. Li, and A. Singh. Computationally efficient robust sparse estimation in high dimensions. In *Proceedings of the 30th Annual Conference on Learning Theory*, COLT '17, 2017.

[Ber06] T. Bernholt. Robust estimators are hard to compute. Technical report, University of Dortmund, Germany, 2006.

[BNJ03] D. M Blei, A. Y. Ng, and M. I. Jordan. Latent dirichlet allocation. *Journal of machine Learning research*, 3(Jan):993–1022, 2003.

[BW41] L. M. Blumenthal and G. E. Wahlin. On the spherical surface of smallest radius enclosing a bounded subset of n-dimensional euclidean space. *Bull. Amer. Math. Soc.*, 47:771–777, 1941.

[CSV17] M. Charikar, J. Steinhardt, and G. Valiant. Learning from untrusted data. In *Proc. 49th Annual ACM Symposium on Theory of Computing (STOC)*, pages 47–60. ACM Press, 2017.

[DBS17] S. S. Du, S. Balakrishnan, and A. Singh. Computationally efficient robust estimation of sparse functionals. *CoRR*, abs/1702.07709, 2017.

[DDS12] C. Daskalakis, I. Diakonikolas, and R.A. Servedio. Learning Poisson Binomial Distributions. In *Proceedings of the 44th Symposium on Theory of Computing*, pages 709–728, 2012.

[DDS15] A. De, I. Diakonikolas, and R. Servedio. Learning from satisfying assignments. In *Proceedings of the 26th Annual ACM-SIAM Symposium on Discrete Algorithms, SODA 2015*, pages 478–497, 2015.

[DK14] C. Daskalakis and G. Kamath. Faster and sample near-optimal algorithms for proper learning mixtures of gaussians. In *Proceedings of The 27th Conference on Learning Theory, COLT 2014*, pages 1183–1213, 2014.

[DKK+16] I. Diakonikolas, G. Kamath, D. M. Kane, J. Li, A. Moitra, and A. Stewart. Robust estimators in high dimensions without the computational intractability. In *Proceedings of FOCS'16*, 2016.

[DKK+17] I. Diakonikolas, G. Kamath, D. M. Kane, J. Li, A. Moitra, and A. Stewart. Being robust (in high dimensions) can be practical. In *Proceedings of the 34th International Conference on Machine Learning*, ICML '17, pages 999–1008. JMLR, Inc., 2017.

[DKS16] I. Diakonikolas, D. M. Kane, and A. Stewart. Robust learning of fixed-structure bayesian networks. *CoRR*, abs/1606.07384, 2016.

[DKS17] I. Diakonikolas, D. M. Kane, and A. Stewart. Statistical query lower bounds for robust estimation of high-dimensional gaussians and gaussian mixtures. In *Proceedings of the 58th Annual IEEE Symposium on Foundations of Computer Science*, FOCS '17, Washington, DC, USA, 2017. IEEE Computer Society.

[DL01] L. Devroye and G. Lugosi. *Combinatorial methods in density estimation*. Springer Series in Statistics, Springer, 2001.

[GCSR14] A. Gelman, J. B. Carlin, H. S. Stern, and D. B. Rubin. *Bayesian data analysis*, volume 2. Chapman & Hall/CRC Boca Raton, FL, USA, 2014.

[GLS88] M. Grötschel, L. Lovász, and A. Schrijver. *Geometric Algorithms and Combinatorial Optimization*, volume 2. Springer, 1988.

[Goo15] R. Goodell. Personal communication, 2015.

[Ham01] F. Hampel. Robust statistics: A brief introduction and overview. In *First International Symposium on Robust Statistics and Fuzzy Techniques in Geodesy and GIS. A. Carosio, H. Kutterer (editors), Swiss Federal Institute of Technology Zurich (ETH), Institute of Geodesy and Photogrammetry, IGP-Bericht*, number 295, pages 13–17, 2001.

[HM13] M. Hardt and A. Moitra. Algorithms and hardness for robust subspace recovery. In *COLT 2013*, pages 354–375, 2013.

[HR09] P.J. Huber and E. M. Ronchetti. *Robust statistics*. Wiley New York, 2009.

[HRRS86] F. R. Hampel, E. M. Ronchetti, P. J. Rousseeuw, and W. A. Stahel. *Robust statistics. The approach based on influence functions*. Wiley New York, 1986.

[Hub64] P. J. Huber. Robust estimation of a location parameter. *The Annals of Mathematical Statistics*, 35(1):73–101, 1964.

[Jun01] H. W. E. Jung. Über die kleinste kugel, die eine räumliche figur einschliesst. *J. Reine Angew. Math*, 123:241–257, 1901.

[Kla07] B. Klartag. A central limit theorem for convex sets. *Inventiones mathematicae*, 168(1):91–131, 2007.

[Li17] J. Li. Robust sparse estimation tasks in high dimensions. *CoRR*, abs/1702.05860, 2017.

[LM00] B. Laurent and P. Massart. Adaptive estimation of a quadratic functional by model selection. *Ann. Statist.*, 28(5):1302–1338, 2000.

[LRV16] K. A. Lai, A. B. Rao, and S. Vempala. Agnostic estimation of mean and covariance. In *Proceedings of FOCS'16*, 2016.

[NJB+08] J. Novembre, T. Johnson, K. Bryc, Z. Kutalik, A. R. Boyko, A. Auton, A. Indap, K. S. King, S. Bergmann, M. R. Nelson, et al. Genes mirror geography within europe. *Nature*, 456(7218):98–101, 2008.

[O'D14] R. O'Donnell. *Analysis of Boolean Functions*. Cambridge University Press, 2014.

[OF96] B. A. Olshausen and D. J. Field. Emergence of simple-cell receptive field properties by learning a sparse code for natural images. *Nature*, 381(6583):607, 1996.

[RL05] P. J. Rousseeuw and A. M. Leroy. *Robust regression and outlier detection*, volume 589. John Wiley & Sons, 2005.

[Rou85] P. Rousseeuw. Multivariate estimation with high breakdown point. *Mathematical Statistics and Applications*, pages 283–297, 1985.

[SCV18] J. Steinhardt, M. Charikar, and G. Valiant. Resilience: A criterion for learning in the presence of arbitrary outliers. In *Proc. of the 9th Conference on Innovations in Theoretical Computer Science*, 2018. to appear.

[Tuk75] J. W. Tukey. Mathematics and picturing of data. In *Proceedings of ICM*, volume 6, pages 523–531, 1975.

[Ver10] R. Vershynin. Introduction to the non-asymptotic analysis of random matrices, 2010.