Being Robust (in High Dimensions) Can Be Practical

Ilias Diakonikolas [*1] **Gautam Kamath** [*2] **Daniel M. Kane** [*3] **Jerry Li** [*2] **Ankur Moitra** [*2] **Alistair Stewart** [*1]

Abstract

Robust estimation is much more challenging in high dimensions than it is in one dimension: Most techniques either lead to intractable optimization problems or estimators that can tolerate only a tiny fraction of errors. Recent work in theoretical computer science has shown that, in appropriate distributional models, it is possible to robustly estimate the mean and covariance with polynomial time algorithms that can tolerate a constant fraction of corruptions, independent of the dimension. However, the sample and time complexity of these algorithms is prohibitively large for high-dimensional applications. In this work, we address both of these issues by establishing sample complexity bounds that are optimal, up to logarithmic factors, as well as giving various refinements that allow the algorithms to tolerate a much larger fraction of corruptions. Finally, we show on both synthetic and real data that our algorithms have state-of-the-art performance and suddenly make high-dimensional robust estimation a realistic possibility.

1. Introduction

Robust statistics was founded in the seminal works of (Tukey, 1960) and (Huber, 1964). The overarching motto is that any model (especially a parametric one) is only approximately valid, and that any estimator designed for a particular distribution that is to be used in practice must also be stable in the presence of model misspecification. The standard setup is to assume that the samples we are given come from a nice distribution, but that an adversary has the power to arbitrarily corrupt a constant fraction of the observed data. After several decades of work, the robust statistics community has discovered a myriad of estimators that are provably robust. An important feature of this line of work is that it can tolerate a constant fraction of corruptions *independent of the dimension* and that there are estimators for both the location (e.g., the mean) and scale (e.g., the covariance). See (Huber & Ronchetti, 2009) and (Hampel et al., 1986) for further background.

It turns out that there are vast gaps in our understanding of robustness, when computational considerations are taken into account. In one dimension, robustness and computational efficiency are in perfect harmony. The empirical mean and empirical variance are not robust, because a single corruption can arbitrarily bias these estimates, but alternatives such as the median and the interquartile range are straightforward to compute and are provably robust.

But in high dimensions, there is a striking tension between robustness and computational efficiency. Let us consider estimators for location. The Tukey median (Tukey, 1960) is a natural generalization of the one-dimensional median to high-dimensions. It is known that it behaves well (i.e., it needs few samples) when estimating the mean for various symmetric distributions (Donoho & Gasko, 1992; Chen et al., 2016). However, it is hard to compute in general (Johnson & Preparata, 1978; Amaldi & Kann, 1995) and the many heuristics for computing it degrade badly in the quality of their approximation as the dimension scales (Clarkson et al., 1993; Chan, 2004; Miller & Sheehy, 2010). The same issues plague estimators for scale. The minimum volume ellipsoid (Rousseeuw, 1985) is a natural generalization of the one-dimensional interquartile range and is provably robust in high-dimensions, but is also hard to compute. And once again, heuristics for computing it (Van Aelst & Rousseeuw, 2009; Rousseeuw & Struyf, 1998) work poorly in high dimensions.

The fact that robustness in high dimensions seems to come

[*]Equal contribution [1]University of Southern California, Los Angeles, California, USA [2]Massachusetts Institute of Technology, Cambridge, Massachusetts, USA [3]University of California, San Diego, La Jolla, California, USA. Correspondence to: Ilias Diakonikolas <diakonik@usc.edu>, Gautam Kamath <g@csail.mit.edu>, Daniel M. Kane <dakane@cs.ucsd.edu>, Jerry Li <jerryzli@mit.edu>, Ankur Moitra <moitra@mit.edu>, Alistair Stewart <alistais@usc.edu>.

Proceedings of the 34^{th} International Conference on Machine Learning, Sydney, Australia, PMLR 70, 2017.

Collectively, the authors were supported by NSF CCF-1652862, CCF-1551875, CCF-1617730, CCF-1650733, CCF-1553288, CCF-1453261, ONR N00014-12-1-0999, three Sloan Research Fellowships, two Google Faculty Research Awards, an NSF fellowship, the MIT NEC Corporation, and a USC startup grant.

at such a steep price has long been a point of consternation within robust statistics. In a 1997 retrospective on the development of robust statistics, Huber laments: "It is one thing to design a theoretical algorithm whose purpose is to prove [large fractions of corruptions can be tolerated] and quite another thing to design a practical version that can be used not merely on small, but also on medium sized regression problems, with a 2000 by 50 matrix or so. This last requirement would seem to exclude all of the recently proposed [techniques]."

The goal of this paper is to answer Huber's call to action and design estimators for both the mean and covariance that are highly practical, provably robust, and work in high-dimensions. Such estimators make the promise of robust statistics – estimators that work in high-dimensions and limit the error induced by outliers – much closer to a reality.

First, we make some remarks to dispel some common misconceptions. There has been a considerable amount of recent work on robust principal component analysis, much of it making use of semidefinite programming. Some of these works can tolerate a constant fraction of corruptions (Candès et al., 2011), however require that the locations of the corruptions are evenly spread throughout the dataset so that no individual sample is entirely corrupted. In contrast, the usual models in robust statistics are quite rigid in what they require and they do this for good reason. A common scenario that is used to motivate robust statistical methods is if two studies are mixed together, and one subpopulation does not fit the model. Then one wants estimators that work without assuming anything at all about these outliers.

There have also been semidefinite programming methods proposed for robust principal component analysis with outliers (Xu et al., 2010). These methods assume that the uncorrupted matrix is rank r and that the fraction of outliers is at most $1/r$, which again degrades badly as the rank of the matrix increases. Moreover, any method that uses semidefinite programming will have difficulty scaling to the sizes of the problems we consider here. For sake of comparison – even with state-of-the-art interior point methods – it is not currently feasible to solve the types of semidefinite programs that have been proposed when the matrices have dimension larger than a hundred.

1.1. Robustness in a Generative Model

Recent works in theoretical computer science have sought to circumvent the usual difficulties of designing efficient and robust algorithms by instead working in a generative model. The starting point for our paper is the work of Diakonikolas et al. (2016a) who gave an efficient algorithm for the problem of *agnostically learning a Gaussian*: Given a polynomial number of samples from a high-dimensional Gaussian $\mathcal{N}(\mu, \Sigma)$, where an adversary has arbitrarily corrupted an ε-fraction, find a set of parameters $\mathcal{N}'(\widehat{\mu}, \widehat{\Sigma})$ that satisfy $d_{TV}(\mathcal{N}, \mathcal{N}') \leq \widetilde{O}(\varepsilon)$[1].

Total variation distance is the natural metric to use to measure closeness of the parameters, since a $(1-\varepsilon)$-fraction of the observed samples came from a Gaussian. (Diakonikolas et al., 2016a) gave an algorithm for the above problem (note that the guarantees are dimension independent), whose running time and sample complexity are polynomial in the dimension d and $1/\varepsilon$. (Lai et al., 2016) independently gave an algorithm for the unknown mean case that achieves $d_{TV}(\mathcal{N}, \mathcal{N}') \leq \widetilde{O}(\varepsilon\sqrt{\log d})$, and in the unknown covariance case achieves guarantees in a weaker metric that is not affine invariant. A crucial feature is that both algorithms work even when the moments of the underlying distribution satisfy certain conditions, and thus are not necessarily brittle to the modeling assumption that the inliers come from a Gaussian distribution.

A more conceptual way to view such work is as a proof-of-concept that the Tukey median and minimum volume ellipsoid can be computed efficiently *in a natural family of distributional models*. This follows because not only would these be good estimates for the mean and covariance in the above model, but in fact any estimates that are good must also be close to them. Thus, these works fit into the emerging research direction of circumventing worst-case lower bounds by going *beyond worst-case analysis*.

Since the dissemination of the aforementioned works (Diakonikolas et al., 2016a; Lai et al., 2016), there has been a flurry of research activity on computationally efficient robust estimation in a variety of high-dimensional settings, including studying graphical models (Diakonikolas et al., 2016b), understanding the computation-robustness tradeoff for statistical query algorithms (Diakonikolas et al., 2016c), tolerating much more noise by allowing the algorithm to output a list of candidate hypotheses (Charikar et al., 2017), and developing robust algorithms under sparsity assumptions (Li, 2017; Du et al., 2017), and more (Diakonikolas et al., 2017; Steinhardt et al., 2017).

1.2. Our Results

Our goal in this work is to show that high-dimensional robust estimation can be highly practical. However, there are two major obstacles to achieving this. First, the sample complexity and running time of the algorithms in (Diakonikolas et al., 2016a) is prohibitively large for high-dimensional applications. We just would not be able to store as many samples as we would need, in order to com-

[1] We use the notation $\tilde{O}(\cdot)$ to hide factors which are polylogarithmic in the argument – in particular, we note that this bound does not depend on the dimension.

pute accurate estimates, in high-dimensional applications.

Our first main contribution is to show nearly-tight bounds on the sample complexity of the filtering-based algorithm of (Diakonikolas et al., 2016a). Roughly speaking, we accomplish this with a new definition of the *good set* which straightforwardly plugs into the existing analysis, showing that one can estimate the mean with $\widetilde{O}(d/\varepsilon^2)$ samples (when the covariance is known) and the covariance with $\widetilde{O}(d^2/\varepsilon^2)$ samples. Both of these bounds are information-theoretically optimal, up to logarithmic factors.

Our second main contribution is to vastly improve the fraction of adversarial corruptions that can be tolerated in applications. The fraction of errors that the algorithms of (Diakonikolas et al., 2016a) can tolerate is indeed a constant that is independent of the dimension, but it is very small both in theory and in practice – a naive implementation of the algorithm did not remove *any* outliers in many realistic scenarios. We avoid this by giving new ways to empirically tune the threshold for where to remove points from the sample set.

Finally, we show that the same bounds on the error guarantee continue to work even when the underlying distribution is sub-Gaussian. This theoretically confirms that the robustness guarantees of such algorithms are in fact not overly brittle to the distributional assumptions. In fact, the filtering algorithm of (Diakonikolas et al., 2016a) is easily shown to be robust under much weaker distributional assumptions, while retaining near-optimal sample and error guarantees. As an example, we show that it yields a near sample-optimal efficient estimator for robustly estimating the mean of a distribution, under the assumption that its covariance is bounded. Even in this regime, the filtering algorithm guarantees optimal error, up to a constant factor. Furthermore we empirically corroborate this finding by showing that the algorithm works well on real world data, as we describe below.

Now we come to the task of testing out our algorithms. To the best of our knowledge, there have been no experimental evaluations of the performance of the myriad of approaches to robust estimation. It remains mostly a mystery which ones perform well in high-dimensions, and which do not. To test out our algorithms, we design a synthetic experiment where a $(1-\varepsilon)$-fraction of the samples come from a Gaussian and the rest are noise and sampled from another distribution (in many cases, Bernoulli). This gives us a baseline to compare how well various algorithms recover μ and Σ, and how their performance degrades based on the dimension. Our plots show a predictable and yet striking phenomenon: All earlier approaches have error rates that scale polynomially with the dimension and ours is a constant that is almost indistinguishable from the error that comes from sample noise alone. Moreover, our algorithms are able to scale to hundreds of dimensions.

But are algorithms for agnostically learning a Gaussian unduly sensitive to the distributional assumptions they make? We are able to give an intriguing visual demonstration of our techniques on real data. The famous study of (Novembre et al., 2008) showed that performing principal component analysis on a matrix of genetic data recovers a map of Europe. More precisely, the top two singular vectors define a projection into the plane and when the groups of individuals are color-coded with where they are from, we recover familiar country boundaries that corresponds to the map of Europe. The conclusion from their study was that *genes mirror geography*. Given that one of the most important applications of robust estimation ought to be in exploratory data analysis, we ask: To what extent can we recover the map of Europe in the presence of noise? We show that when a small number of corrupted samples are added to the dataset, the picture becomes entirely distorted (and this continues to hold even for many other methods that have been proposed). In contrast, when we run our algorithm, we are able to once again recover the map of Europe. Thus, even when some fraction of the data has been corrupted (e.g., medical studies were pooled together even though the subpopulations studied were different), it is still possible to perform principal component analysis and recover qualitatively similar conclusions as if there were no noise at all!

2. Formal Framework

Notation. For a vector v, we will let $\|v\|_2$ denote its Euclidean norm. If M is a matrix, we will let $\|M\|_2$ denote its spectral norm and $\|M\|_F$ denote its Frobenius norm. We will write $X \in_u S$ to denote that X is drawn from the empirical distribution defined by S.

Robust Estimation. We consider the following powerful model of robust estimation that generalizes many other existing models, including Huber's contamination model:

Definition 2.1. *Given $\varepsilon > 0$ and a distribution family $\mathcal{D}$, the* adversary *operates as follows: The algorithm specifies some number of samples m. The adversary generates m samples $X_1, X_2, \ldots, X_m$ from some (unknown) $D \in \mathcal{D}$. It then draws m' from an appropriate distribution. This distribution is allowed to depend on $X_1, X_2, \ldots, X_m$, but when marginalized over the m samples satisfies $m' \sim$ Bin(ε, m). The adversary is allowed to inspect the samples, removes m' of them, and replaces them with arbitrary points. The set of m points is then given to the algorithm.*

In summary, the adversary is allowed to inspect the samples before corrupting them, both by adding corrupted points and deleting uncorrupted points. In contrast, in Huber's model the adversary is oblivious to the samples and is only allowed to add corrupted points.

We remark that there are no computational restrictions on the adversary. The goal is to return the parameters of a distribution $\widehat{D}$ in $\mathcal{D}$ that are close to the true parameters in an appropriate metric. For the case of the mean, our metric will be the Euclidean distance. For the covariance, we will use the Mahalanobis distance, i.e., $\|\Sigma^{-1/2}\widehat{\Sigma}\Sigma^{-1/2} - I\|_F$. This is a strong affine invariant distance that implies corresponding bounds in total variation distance.

We will use the following terminology:

Definition 2.2. *We say that a set of samples is ε-corrupted if it is generated by the process described in Definition 2.1.*

3. Nearly Sample-Optimal Efficient Robust Learning

In this section, we present near sample-optimal efficient robust estimators for the mean and the covariance of high-dimensional distributions under various structural assumptions of varying strength. Our estimators rely on the *filtering technique* introduced in (Diakonikolas et al., 2016a).

This paper gave two algorithmic techniques: the first one was a spectral technique to iteratively remove outliers from the dataset (filtering), and the second one was a soft-outlier removal method relying on convex programming. The filtering technique seemed amenable to practical implementation (as it only uses simple eigenvalue computations), but the corresponding sample complexity bounds given in (Diakonikolas et al., 2016a) are polynomially worse than the information-theoretic minimum. On the other hand, the convex programming technique of Diakonikolas et al. (2016a) achieved better sample complexity bounds (e.g., near sample-optimal for robust mean estimation), but relied on the ellipsoid method, which seemed to preclude a practically efficient implementation.

In this work, we achieve the best of both worlds: we give a better analysis of the filter, giving sample-optimal bounds (up to logarithmic factors) for both the mean and the covariance. Moreover, we show that the filtering technique easily extends to much weaker distributional assumptions (e.g., under bounded second moments). Roughly speaking, the filtering technique follows a general iterative recipe: (1) via spectral methods, find some univariate test which is violated by the corrupted points, (2) find some concrete tail bound violated by the corrupted points, and (3) discard all points which violate this tail bound.

We start with sub-gaussian distributions. Recall that if P is sub-gaussian on $\mathbb{R}^d$ with mean vector μ and parameter $\nu > 0$, then for any unit vector $v \in \mathbb{R}^d$ we have that $\Pr_{X\sim P}[|v \cdot (X - \mu)| \geq t] \leq \exp(-t^2/2\nu)$.

Theorem 3.1. *Let G be a sub-gaussian distribution on $\mathbb{R}^d$ with parameter $\nu = \Theta(1)$, mean μ^G, covariance matrix I, and $\varepsilon > 0$. Let S be an ε-corrupted set of samples from G of size $\Omega((d/\varepsilon^2)\operatorname{poly}\log(d/\varepsilon))$. There exists an efficient algorithm that, on input S and $\varepsilon > 0$, returns a mean vector $\widehat{\mu}$ so that with probability at least $9/10$ we have $\|\widehat{\mu} - \mu^G\|_2 = O(\varepsilon\sqrt{\log(1/\varepsilon)})$.*

Diakonikolas et al. (2016a) gave algorithms for robustly estimating the mean of a Gaussian distribution with known covariance and for robustly estimating the mean of a binary product distribution. The main motivation for considering these specific distribution families is that robustly estimating the mean within Euclidean distance immediately implies total variation distance bounds for these families. The above theorem establishes that these guarantees hold in a more general setting with near sample-optimal bounds. Under a bounded second moment assumption, we show:

Theorem 3.2. *Let P be a distribution on $\mathbb{R}^d$ with unknown mean vector μ^P and unknown covariance matrix $\Sigma_P \preceq \sigma^2 I$. Let S be an ε-corrupted set of samples from P of size $\Theta((d/\varepsilon)\log d)$. There exists an efficient algorithm that, on input S and $\varepsilon > 0$, with probability $9/10$ outputs $\widehat{\mu}$ with $\|\widehat{\mu} - \mu^P\|_2 \leq O(\sqrt{\varepsilon}\sigma)$.*

The sample size above is optimal, up to a logarithmic factor, and the error guarantee is easily seen to be the best possible up to a constant factor. The main difference between the filtering algorithm establishing the above theorem and the filtering algorithm for the sub-gaussian case is how we choose the threshold for the filter. Instead of looking for a violation of a concentration inequality, here we will choose a threshold *at random*. In this case, randomly choosing a threshold weighted towards higher thresholds suffices to throw out more corrupted samples than uncorrupted samples *in expectation*. Although it is possible to reject many good samples this way, we show that the algorithm still only rejects a total of $O(\varepsilon)$ samples with high probability.

Finally, estimating the covariance of a Gaussian:

Theorem 3.3. *Let $G \sim \mathcal{N}(0, \Sigma)$ be a Gaussian in d dimensions, and let $\varepsilon > 0$. Let S be an ε-corrupted set of samples from G of size $\Omega((d^2/\varepsilon^2)\operatorname{poly}\log(d/\varepsilon))$. There exists an efficient algorithm that, given S and ε, returns the parameters of a Gaussian distribution $G' \sim \mathcal{N}(0, \widehat{\Sigma})$ so that with probability at least $9/10$, it holds $\|I - \Sigma^{-1/2}\widehat{\Sigma}\Sigma^{-1/2}\|_F = O(\varepsilon\log(1/\varepsilon))$.*

We now provide a high-level description of the main ingredient which yields these improved sample complexity bounds. The initial analysis of Diakonikolas et al. (2016a) established sample complexity bounds which were sub-optimal by polynomial factors because it insisted that the set of good samples (i.e., before the corruption) satisfied very tight tail bounds. To some degree such bounds are necessary, as when we perform our filtering procedure, we need to ensure that not too many good samples are thrown

away. However, the old analysis required that fairly strong tail bounds hold *uniformly*. The idea for the improvement is as follows: If the errors are sufficient to cause the variance of some polynomial p (linear in the unknown mean case or quadratic in the unknown covariance case) to increase by more than ε, it must be the case that for some T, roughly an ε/T^2 fraction of samples are error points with $|p(x)| > T$. As long as we can ensure that less than an ε/T^2 fraction of our good sample points have $|p(x)| > T$, this will suffice for our filtering procedure to work. For small values of T, these are much weaker tail bounds than were needed previously and can be achieved with a smaller number of samples. For large values of T, these tail bounds are comparable to those used in previous work (Diakonikolas et al., 2016a) , but in such cases we can take advantage of the fact that $|p(G)| > T$ only with very small probability, again allowing us to reduce the sample complexity. The details are deferred to the supplementary material.

4. Filtering

We now describe the filtering technique more rigorously, as well as some additional practical heuristics.

4.1. Robust Mean Estimation

We first consider mean estimation. The algorithms which achieve Theorems 3.1 and 3.2 both follow the general recipe in Procedure 1. We must specify three parameter functions:

- $\mathrm{Thres}(\varepsilon)$ is a threshold function—we terminate if the covariance has spectral norm bounded by $\mathrm{Thres}(\varepsilon)$.
- $\mathrm{Tail}(T, d, \varepsilon, \delta, \tau)$ is an univariate tail bound, which would only be violated by a τ fraction of points if they were uncorrupted, but is violated by many more of the current set of points.
- $\delta(\varepsilon, s)$ is a slack function, which we require for technical reasons.

Given these objects, our filter is fairly easy to state: first, we compute the empirical covariance. Then, we check if the spectral norm of the empirical covariance exceeds $\mathrm{Thres}(\varepsilon)$. If it does not, we output the empirical mean with the current set of data points. Otherwise, we project onto the top eigenvector of the empirical covariance, and throw away all points which violate $\mathrm{Tail}(T, d, \varepsilon, \delta, \tau)$, for some choice of slack function δ.

Sub-gaussian case To instantiate this algorithm for the subgaussian case, we take $\mathrm{Thres}(\varepsilon) = O(\varepsilon \log 1/\varepsilon)$, $\delta(\varepsilon, s) = 3\sqrt{\varepsilon(s-1)}$, and $\mathrm{Tail}(T, d, \varepsilon, \delta, \tau) = 8\exp(-T^2/2\nu) + 8\frac{\varepsilon}{T^2 \log(d \log(d/\varepsilon\tau))}$, where ν is the subgaussian parameter. See the supplementary material for details.

Procedure 1 Filter-based algorithm template for robust mean estimation

Input: An ε-corrupted set of samples S, $\mathrm{Thres}(\varepsilon)$, $\mathrm{Tail}(T, d, \varepsilon, \delta, \tau)$, $\delta(\varepsilon, s)$
Compute the sample mean $\mu^{S'} = \mathbb{E}_{X \in_u S'}[X]$, covariance Σ, approximations for the largest absolute eigenvalue and eigenvector of Σ, $\lambda^* := \|\Sigma\|_2$, and v^*.
if $\|\Sigma\|_2 \leq \mathrm{Thres}(\varepsilon)$ **then**
  return $\mu^{S'}$.
end if
Let $\delta = \delta(\varepsilon, \|\Sigma\|_2)$.
Find $T > 0$ such that

$$\Pr_{X \in_u S'}\left[|v^* \cdot (X - \mu^{S'})| > T + \delta\right] > \mathrm{Tail}(T, d, \varepsilon, \delta, \tau).$$

return $\{x \in S' : |v^* \cdot (x - \mu^{S'})| \leq T + \delta\}$.

Second moment case To instantiate this algorithm for the second moment case, we take $\mathrm{Thres}(\varepsilon) = 9$, $\delta = 0$, and we take Tail to be a *random rescaling* of the largest deviation in the data set, in the direction v^*. See the supplementary material for details.

4.2. Robust Covariance Estimation

Our algorithm for robust covariance follows the exact recipe outlined above, with one key difference—we check for deviations in the empirical *fourth* moment tensor. Intuitively, just as in the robust mean setting, we used degree-2 information to detect outliers for the mean (the degree-1 moment), here we use degree-4 information to detect outliers for the covariance (the degree-2 moment).

This corresponds to finding a normalized degree-2 polynomial whose empirical variance is too large. Filtering along this polynomial with an appropriate choice of $\mathrm{Thres}(\varepsilon)$, $\delta(\varepsilon, s)$, and Tail gives the desired bounds. See the supplementary material for more details.

4.3. Better Univariate Tests

In the algorithms described above for robust mean estimation, after projecting onto one dimension, we center the points at the empirical mean along this direction. This is theoretically sufficient, however, introduces additional constant factors since the empirical mean along this direction may be corrupted. Instead, one can use a robust estimate for the mean in one direction. Namely, it is well known that the median is a provably robust estimator for the mean for symmetric distributions (Huber & Ronchetti, 2009; Hampel et al., 1986), and under certain models it is in fact optimal in terms of its resilience to noise (Dvoretzky et al., 1956; Massart, 1990; Chen, 1998; Daskalakis & Kamath, 2014; Diakonikolas et al., 2017). By centering the points

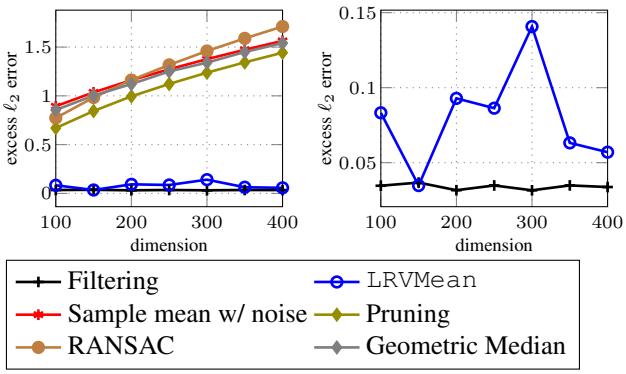

Figure 1. Experiments with synthetic data for robust mean estimation: excess ℓ_2 error is reported against dimension.

at the median instead of the mean, we are able to achieve better error in practice.

4.4. Adaptive Tail Bounding

In our empirical evaluation, we found that it was important to find an appropriate choice of Tail, to achieve good error rates, especially for robust covariance estimation. Concretely, in this setting, our tail bound is given by $\text{Tail}(T, d, \varepsilon, \delta, \tau) = C_1 \exp(-C_2 T) + \text{Tail}_2(T, d, \varepsilon, \delta, \tau)$, for some function Tail_2, and constants C_1, C_2. We found that for reasonable settings, the term that dominated was always the first term on the RHS, and that Tail_2 is less significant. Thus, we focused on optimizing the first term.

We found that depending on the setting, it was useful to change the constant C_2. In particular, in low dimensions, we could be more stringent, and enforce a stronger tail bound (which corresponds to a higher C_2), but in higher dimensions, we must be more lax with the tail bound. To do this in a principled manner, we introduced a heuristic we call *adaptive tail bounding*. Our goal is to find a choice of C_2 which throws away roughly an ε-fraction of points. The heuristic is fairly simple: we start with some initial guess for C_2. We then run our filter with this C_2. If we throw away too many data points, we increase our C_2, and retry. If we throw away too few, then we decrease our C_2 and retry. Since increasing C_2 strictly decreases the number of points thrown away, and vice versa, we binary search over our choice of C_2 until we reach something close to our target accuracy. In our current implementation, we stop when the fraction of points we throw away is between $\varepsilon/2$ and $3\varepsilon/2$, or if we've binary searched for too long. We found that this heuristic drastically improves our accuracy, and allows our algorithm to scale fairly smoothly from low to high dimension.

5. Experiments

We performed an empirical evaluation of the above algorithms on synthetic and real data sets with and without

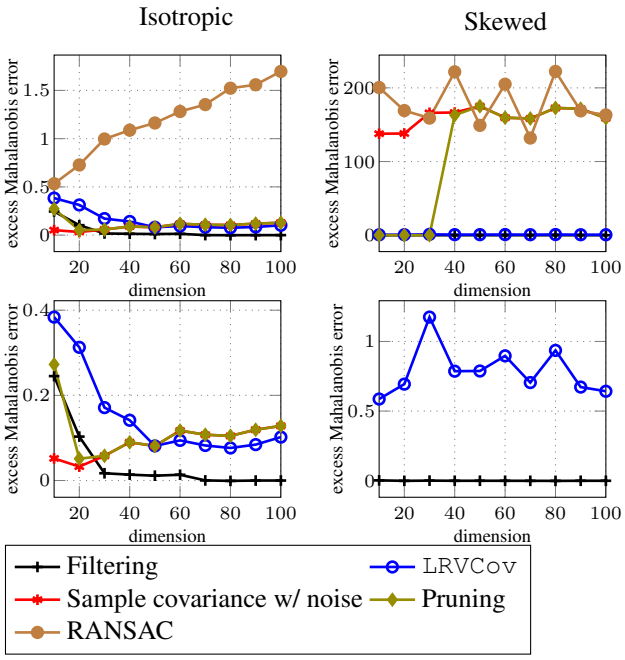

Figure 2. Experiments with synthetic data for robust covariance estimation: excess Mahalanobis error is reported against dimension.

synthetic noise. All experiments were done on a laptop computer with a 2.7 GHz Intel Core i5 CPU and 8 GB of RAM. The focus of this evaluation was on statistical accuracy, not time efficiency. In all synthetic trials, our algorithm consistently had the smallest error, sometimes orders of magnitude better than any other algorithms. In the semi-synthetic benchmark, our algorithm also (arguably) performs the best, though this is subjective. While we did not optimize our code for runtime, it is always comparable to (and often better than) the effective alternatives.

5.1. Synthetic Data

Experiments with synthetic data allow us to verify the error guarantees and the sample complexity rates proven in Section 3. In both cases, the experiments validate the accuracy and usefulness of our algorithm, almost exactly matching the best rate without noise.

Unknown mean The results of our synthetic mean experiment are shown in Figure 1. In the synthetic mean experiment, we set $\varepsilon = 0.1$, and for dimension $d = [100, 150, \ldots, 400]$, we generate $n = \frac{10d}{\varepsilon^2}$ samples, where a $(1-\varepsilon)$-fraction come from $\mathcal{N}(\mu, I)$, and an ε fraction come from a noise distribution. Our goal is to produce an estimator which minimizes the ℓ_2 error the estimator has to the truth. As a baseline, we compute the error that is achieved by only the uncorrupted sample points. This error will be used as the gold standard for comparison, since in the presence of error, this is roughly the best one could do

even if all the noise points were identified exactly.[2]

On this data, we compared the performance of our Filter algorithm to that of (1) the empirical mean of all the points, (2) a trivial pruning procedure, (3) the geometric median of the data, (4) a RANSAC-based mean estimation algorithm, and (5) a recently proposed robust estimator for the mean due to (Lai et al., 2016), which we will call `LRVMean`. For (5), we use the implementation available in their Github.[3] In Figure 1, the x-axis indicates the dimension of the experiment, and the y-axis measures the ℓ_2 error of our estimated mean minus the ℓ_2 error of the empirical mean of the true samples from the Gaussian, i.e., the excess error induced over the sampling error.

We tried various noise distributions, and found that the same qualitative pattern arose for all of them. In the reported experiment, our noise distribution was a mixture of two binary product distributions, where one had a couple of large coordinates (see the supplementary material for a detailed description). For all (nontrivial) error distributions we tried, we observed that indeed the empirical mean, pruning, geometric median, and RANSAC all have error which diverges as d grows, as the theory predicts. On the other hand, both our algorithm and `LRVMean` have markedly smaller error as a function of dimension. Indeed, our algorithm's error is almost identical to that of the empirical mean of the uncorrupted sample points.

Unknown covariance See Figure 2 for the results of our synthetic covariance experiment. Our setup is similar to that for the synthetic mean. Since both our algorithm and `LRVCov` require access to fourth moments, we ran into issues with limited memory on machines. This limitation prevented us from performing experiments at the same dimensionality as the unknown mean setting, and we could not use as many samples. We fix $\varepsilon = 0.05$. For dimension $d = [10, 20, \ldots, 100]$, we generate $\frac{0.5d}{\varepsilon^2}$ samples, where a $(1-\varepsilon)$-fraction come from $\mathcal{N}(0, \Sigma)$, and the rest come from a noise distribution. We measure distance in the natural affine invariant way, namely, the Mahalanobis distance induced by Σ to the identity: $\mathrm{err}(\widehat{\Sigma}) = \|\Sigma^{-1/2}\widehat{\Sigma}\Sigma^{-1/2} - I\|_F$. As before, we use the empirical error of only the uncorrupted data points as a benchmark.

On this corrupted data, we compared the performance of our Filter algorithm to that of (1) the empirical covariance of all the points, (2) a trivial pruning procedure, (3) a RANSAC-based minimal volume ellipsoid (MVE) algorithm, and (5) a recently proposed robust estimator for the covariance due to (Lai et al., 2016), which we will call `LRVCov`. For (5), we again obtained the implementation from their Github repository.

We tried various choices of Σ and noise distribution. Figure 2 shows two choices of Σ and noise. Again, the x-axis indicates the dimension of the experiment and the y-axis indicates the estimator's excess Mahalanobis error over the sampling error. In the left figure, we set $\Sigma = I$, and our noise points are simply all located at the all-zeros vector. In the right figure, we set $\Sigma = I + 10e_1e_1^T$, where e_1 is the first basis vector, and our noise distribution is a somewhat more complicated distribution, which is similarly spiked, but in a different, random, direction. We formally define this distribution in the supplementary material. For all choices of Σ and noise we tried, the qualitative behavior of our algorithm and `LRVCov` was unchanged. Namely, we seem to match the empirical error without noise up to a very small slack, for all dimensions. On the other hand, the performance of empirical mean, pruning, and RANSAC varies widely with the noise distribution. The performance of all these algorithms degrades substantially with dimension, and their error gets worse as we increase the skew of the underlying data. The performance of `LRVCov` is the most similar to ours, but again is worse by a large constant factor. In particular, our excess risk was on the order of 10^{-4} for large d, for both experiments, whereas the excess risk achieved by `LRVCov` was in all cases a constant between 0.1 and 2.

These experiments demonstrate that our statistical guarantees are in fact quite strong. As our excess error is almost zero (and orders of magnitude smaller than other approaches), this suggests that our sample complexity is indeed near-optimal, since we match the rate without noise, and that the constants and logarithmic factors in the theoretical recovery guarantee are often small or non-existent.

5.2. Semi-synthetic Data

To demonstrate the efficacy of our method on real data, we revisit the famous study of Novembre et al. (2008). In this study, the authors investigated data collected as part of the POPRES project. This dataset consists of the genotyping of thousands of individuals using the Affymetrix 500K single nucleotide polymorphism (SNP) chip. The authors pruned the dataset to obtain the genetic data of over 1387 European individuals, annotated by their country of origin. Using principal components analysis, they produce a two-dimensional summary of the genetic variation, which bears a striking resemblance to the map of Europe.

Our experimental setup is as follows. While the original dataset is very high dimensional, we use a 20 dimensional version of the dataset as found in the authors' GitHub[4]. We

[2] We note that it is possible that an estimator may achieve slightly better error than this baseline.

[3] https://github.com/kal2000/AgnosticMean\AndCovarianceCode

[4] https://github.com/NovembreLab/Novembre_etal_2008_misc

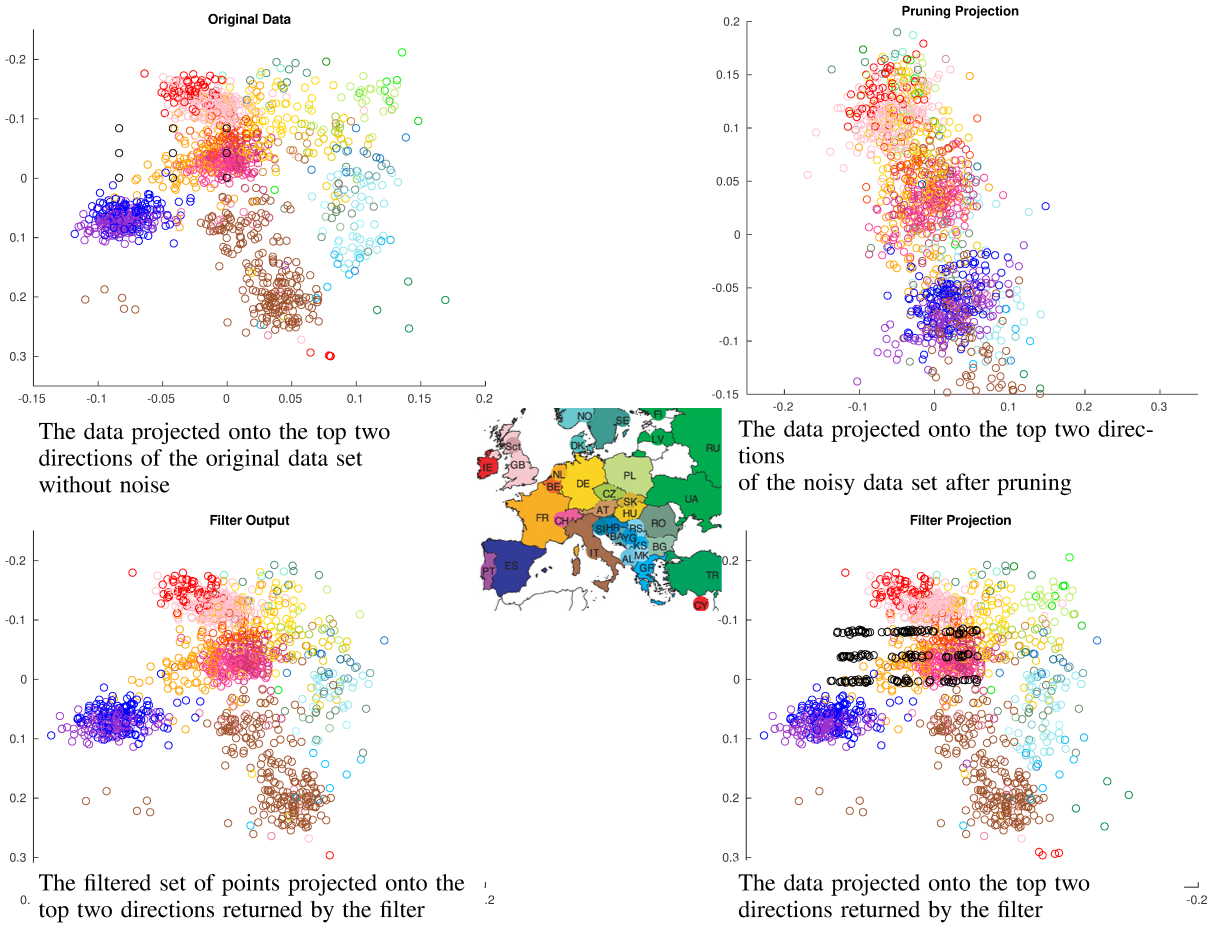

Figure 3. Given genetic data from (Novembre et al., 2008), projected down to 20-dimensions, with added noise. Colors indicate the individual's country of origin, and match the colors of the countries in the map of Europe. Black points are added noise. The top left plot is the original plot from (Novembre et al., 2008). We recover Europe in the presence of noise whereas naive methods do not.

first randomly rotate the data, as then 20 dimensional data was diagonalized, and the high dimensional data does not follow such structure. We then add an additional $\frac{\varepsilon}{1-\varepsilon}$ fraction of points (so that they make up an ε-fraction of the final points). These added points were discrete points, following a simple product distribution (described in the supplementary materials). We used a number of methods to obtain a covariance matrix for this dataset, and we projected the data onto the top two singular vectors of this matrix. In Figure 3, we compare our techniques to pruning. In particular, our output was able to more or less reproduce the map of Europe, whereas pruning fails to. In the supplementary material, we also compare our result with a number of other techniques, including those we tested against in the unknown covariance experiments, and other robust PCA techniques. The only alternative algorithm which was able to produce meaningful output was `LRVCov`, which produced output that was similar to ours, but which produced a map which was somewhat more skewed. We believe that our algorithm produces the best picture.

In Figure 3, we also display the actual points which were output by our algorithm's Filter. While it manages to remove most of the noise points, it also seems to remove some of the true data points, particularly those from Eastern Europe and Turkey. We attribute this to a lack of samples from these regions, and thus one could consider them as outliers to a dataset consisting of Western European individuals. For instance, Turkey had 4 data points, so it seems quite reasonable that any robust algorithm would naturally consider these points outliers.

We view our experiments as a proof of concept demonstration that our techniques can be useful in real world exploratory data analysis tasks, particularly those in high-dimensions. Our experiments reveal that a minimal amount of noise can completely disrupt a data analyst's ability to notice an interesting phenomenon, thus limiting us to only very well-curated data sets. But with robust methods, this noise does not interfere with scientific discovery, and we can still recover interesting patterns which otherwise would have been obscured by noise.

References

Amaldi, E. and Kann, V. The complexity and approximability of finding maximum feasible subsystems of linear relations. *Theoretical Computer Science*, 147:181–210, 1995.

Candès, E. J., Li, X., Ma, Y., and Wright, J. Robust principal component analysis? *J. ACM*, 58(3):11, 2011.

Chan, T. M. An optimal randomized algorithm for maximum tukey depth. In *Proceedings of the Fifteenth Annual ACM-SIAM Symposium on Discrete Algorithms (SODA)*, pp. 430–436, 2004.

Charikar, M., Steinhardt, J., and Valiant, G. Learning from untrusted data. In *Proceedings of STOC'17*, 2017.

Chen, M., Gao, C., and Ren, Z. A general decision theory for huber's ε-contamination model. *Electronic Journal of Statistics*, 10(2):3752–3774, 2016.

Chen, Z. A note on bias robustness of the median. *Statistics & probability letters*, 38(4):363–368, 1998.

Clarkson, K. L., Eppstein, D., Miller, G. L., Sturtivant, C., and Teng, S.-H. Approximating center points with iterated radon points. In *Proceedings of the Ninth Annual Symposium on Computational Geometry*, SCG '93, pp. 91–98, New York, NY, USA, 1993. ACM.

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

Diakonikolas, I., Kamath, G., Kane, D. M., Li, J., Moitra, A., and Stewart, A. Robust estimators in high dimensions without the computational intractability. In *Proceedings of FOCS'16*, 2016a. Full version available at https://arxiv.org/pdf/1604.06443.pdf.

Diakonikolas, I., Kane, D. M., and Stewart, A. Robust learning of fixed-structure bayesian networks. *CoRR*, abs/1606.07384, 2016b. URL https://arxiv.org/abs/1606.07384.

Diakonikolas, I., Kane, D. M., and Stewart, A. Statistical query lower bounds for robust estimation of high-dimensional gaussians and gaussian mixtures. *CoRR*, abs/1611.03473, 2016c. URL http://arxiv.org/abs/1611.03473.

Diakonikolas, I., Kamath, G., Kane, D. M., Li, J., Moitra, A., and Stewart, A. Robustly learning a gaussian: Getting optimal error, efficiently. *CoRR*, abs/1704.03866, 2017.

Donoho, D. L. and Gasko, M. Breakdown properties of location estimates based on halfspace depth and projected outlyingness. *Ann. Statist.*, 20(4):1803–1827, 12 1992.

Du, S. S., Balakrishnan, S., and Singh, A. Computationally efficient robust estimation of sparse functionals. In *Proceedings of COLT'17*, 2017.

Dvoretzky, A., Kiefer, J., and Wolfowitz, J. Asymptotic minimax character of the sample distribution function and of the classical multinomial estimator. *Ann. Mathematical Statistics*, 27(3):642–669, 1956.

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

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

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

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

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

Li, J. Robust sparse estimation tasks in high dimensions. In *Proceedings of COLT'17*, 2017.

Massart, P. The tight constant in the Dvoretzky-Kiefer-Wolfowitz inequality. *Annals of Probability*, 18(3): 1269–1283, 1990.

Miller, G.L. and Sheehy, D. Approximate centerpoints with proofs. *Comput. Geom.*, 43(8):647–654, 2010.

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

Rousseeuw, P. Multivariate estimation with high breakdown point. *Mathematical Statistics and Applications*, pp. 283–297, 1985.

Rousseeuw, P. J. and Struyf, A. Computing location depth and regression depth in higher dimensions. *Statistics and Computing*, 8(3):193–203, 1998.

Steinhardt, J., Charikar, M., and Valiant, G. Resilience: A criterion for learning in the presence of arbitrary outliers. *CoRR*, abs/1703.04940, 2017.

Tukey, J.W. A survey of sampling from contaminated distributions. *Contributions to probability and statistics*, 2: 448–485, 1960.

Van Aelst, S. and Rousseeuw, P. Minimum volume ellipsoid. *Wiley Interdisciplinary Reviews: Computational Statistics*, 1(1):71–82, 2009.

Xu, H., Caramanis, C., and Sanghavi, S. Robust pca via outlier pursuit. In *Advances in Neural Information Processing Systems*, pp. 2496–2504, 2010.