CS 224 Fall 2023 Scribes: Prashanth Amireddy, Costin-Andrei Oncescu
October 04, 2023

Lecture 9: Robust statistics I: Motivation, Iterative filtering

While the last unit on sum-of-squares algorithms provided strong mathematical guarantees for various estimation tasks, this unit on robust statistics brings us closer to more practical implementations along with the theoretical guarantees.

1 Introduction

The general set-up for robust statistics problems is as follows: We get "corrupted" samples from an unknown distribution q and the goal is to perform various statistical inference tasks with strong guarantees that are competitive with respect to the setting where no samples were corrupted (or a setting where we know which samples were corrupted). There are four broad ways of modelling corruptions based on how "powerful" the adversary is. There are two axes to determine this: *oblivous* vs *adaptive* and *additive* vs *non-additive*. An adversary is said to be oblivious if the bad points are independent of the good points and is said to be adaptive otherwise. It is said to be additive if the good points are i.i.d. draws from q and non-additive otherwise. These notions lead to the following four types of corruption models:

- **Huber contamination:** Adversary fixes a distribution p_{adv} and we get n i.i.d. samples from the mixture $(1-\eta)\cdot q + \eta \cdot p_{\text{adv}}$.
- **Additive-η contamination:** Nature draws $x_1^*, x_2^*, \ldots, x_{(1-\eta)n}^* \sim q$ i.i.d. Adversary inspects these, adds $\eta \cdot n$ arbitrary points to the dataset, and shuffles the dataset arbitrarily. We get the corrupted dataset.
- **TV contamination:** Adversary picks any distribution q' such that $TV(q, q') \leq \eta$ and we get n i.i.d. samples from q'. Here $TV(.,.)$ denotes the total variation distance between the two distributions.
- **Strong contamination:** Nature draws $x_1^*, x_2^*, \ldots, x_n^* \sim q$ i.i.d. Adversary inspects these points, picks a subset of size $\eta \cdot n$ and corrupts those points arbitrarily. We get the corrupted dataset.

Table 1 classifies which of the above models are oblivious and/or additive. Note that as we move to the right and/or to the bottom cell in the table, we are moving to a more general model of contamination.

	Oblivious	**Adaptive**
Additive	Huber	Additive-η
Non-additive	TV	Strong

Table 1: Contamination models

In this lecture, we consider the robust mean estimation problem as studied by Huber [Hub92], although we actually provide an algorithm for the strong contamination model.

Parameters: Unknown mean $\mu \in \mathbb{R}^d$ and (known) corruption fraction η.
Given: Corrupted samples $x_1, x_2, \ldots, x_n$ from $\mathcal{N}(\mu, \mathrm{Id})$ under the strong contamination model.
Goal: Estimate μ "as best as possible".

But what is "as best as possible"? Consider ν such that $\|\nu - \mu\|_2 = \eta$. Then one can show that $TV(\mathcal{N}(\mu, 1), \mathcal{N}(\nu, 1)) = \Theta(\eta)$, so it is impossible to distinguish whether our data is just i.i.d. samples from $\mathcal{N}(\nu, 1)$ or whether it was obtained by TV-contaminating samples from $\mathcal{N}(\mu, 1)$. Therefore, we cannot hope to estimate the mean to error less than $O(\eta)$.

When $d = 1$, there are various approaches to achieving error $O(\eta)$, e.g., median, trimmed mean and Huber M-estimation. However, when d is large, trimming-based and M-estimation based algorithms will suffer additional poly(d) factors in the error. The high-dimensional (Tukey) median does not, but it is computationally inefficient.

Recently, Diakonikolas et al. [DKK$^+$16] proved that there exists a polynomial time algorithm for robustly estimating the mean of a Gaussian , even under strong contamination.

Theorem 1 (see Theorem 1.2 of [DKK$^+$16] for a formal statement)**.** *There is a (practical) polynomial-time algorithm for robustly estimating the parameters of a Gaussian $\mathcal{N}(\mu, \Sigma)$, even under strong contamination, to error $\tilde{O}(\eta)$.*

We will prove the above theorem (although we only show a weaker bound on the error) based on a technique called *iterative filtering*.

2 Robust mean estimation

2.1 Setup

Suppose q is a distribution over $\mathbb{R}^d$ with with mean $\mu \in \mathbb{R}^d$ and covariance matrix $\Sigma \preceq \mathrm{Id}$. Let $x_1^*, x_2^*, \ldots, x_n^* \sim q$ be i.i.d. draws sampled by nature. Suppose an adversary corrupts an arbitrary η (small constant) fraction of these points and we are given the corrupted samples $\{x_1, x_2, \ldots, x_n\}$.

Now let

$$\{x_1, x_2, \ldots, x_n\} = S_g \cup S_b \setminus S_r,$$

where $|S_b| = |S_r| = \eta n$, $S_g = \{x_1^*, x_2^*, \ldots, x_n^*\}$ are the ("good") original draws from q, S_r are the points which were corrupted and S_b are the points they have been replaced with by the adversary. As shorthand notation, we use $\sum_{\text{clean } i} x_i$ to refer to $\sum_{x \in S_g \setminus S_r} x$ and $\sum_{\text{bad } i} x_i$ to refer to $\sum_{x \in S_b} x$.

Assume (by considering large enough n) that for $\mu_g \triangleq \frac{1}{|S_g|} \sum_{x \in S_g} x$ and $\Sigma_g \triangleq \frac{1}{|S_g|} \sum_{x \in S_g} (x - \mu_g)(x - \mu_g)^\top$, we have

$$\|\mu_g - \mu\|_2 \lesssim \sqrt{\epsilon} \quad \text{and} \tag{1}$$

$$\|\Sigma_g\|_{\text{op}} \lesssim 1. \tag{2}$$

2.2 Motivating the algorithm

The starting point of the algorithm is to maintain (and keep updating) weights $w = \{w_i\}_{i \in [n]}$ for the dataset that indicate how confident we are that x_i is clean for $i \in [n]$. That is, let

$$0 \le w_i \le \frac{1}{n}, \quad \text{for all } i \in [n]. \tag{3}$$

Note. Here w_i is like $\frac{1}{n} a_i$ where a_i is from the SoS analysis. But these are actual real numbers now and not SoS variables. Ideally we would want $w_i = \frac{1}{n} \cdot \mathbb{1}[i \in S_g]$.

Define the *weighted mean* and *weighted covariance* by

$$\mu_w \triangleq \frac{1}{\sum_i w_i} \sum_i w_i x_i \quad \text{and} \tag{4}$$

$$\Sigma_w \triangleq \frac{1}{\sum_i w_i} \sum_i w_i (x_i - \mu_w)(x_i - \mu_w)^\top. \tag{5}$$

In general, w_i's can be viewed as "soft" indicators for a subset. We want to pick out a large subset of clean points, so we care about w_i's that satisfy

$$\sum_i w_i \geq 1 - \eta. \tag{6}$$

Note that the set of $w \in \mathbb{R}^d$ satisfying (3) and 6 is a convex hull K_η of the set $\{\frac{1}{n} \cdot \mathbb{1}_S : S \subseteq [n]$ such that $|S| \geq (1-\eta)n\}$. We will maintain that $w \in K_\eta$ throughout the algorithm. [1]

We now state a lemma that will be important in arguing the correctness of the algorithm (which we defer to Algorithm 1). Informally, the above lemma states that if μ_w is a "wrong" estimate for the mean, then $\|\Sigma_w\|_{\text{op}}$ is "large".

Lemma 1 (Spectral signature lemma). *For any $w \in K_\eta$,*

$$\|\mu_g - \mu_w\|_2 \lesssim \sqrt{\eta} \cdot \left(1 + \sqrt{\|\Sigma_w\|_{\text{op}}}\right).$$

We defer the proof of the lemma for now, but instead look at how to use it to come up with a robust mean estimation algorithm. Given current weights w, we define *scores*

$$\tau_i \triangleq \langle v, x_i - \mu_w \rangle^2,$$

where v is the top eigenvector of Σ_w. Let $\tau_{\text{max}} \triangleq \max_{i:w_i>0} \tau_i$. Intuitively,

$$\text{higher score } \tau_i \iff \text{more likely that } x_i \text{ is a corrupted sample}$$

since the clean samples (projected to v) would be "close" to the mean.

The above intuition leads to the following algorithm (take note of the update rule for weights: we are decreasing the weights for samples with large τ_i).

2.3 The algorithm: Iterative filtering

Let $C \geq 1$ be a small constant that we are going to fix later.

[1] This ends up being $K_{2\eta}$ in the final proof.

```
Algorithm 1: ROBUST MEAN ESTIMATION ALGORITHM
$w_i \leftarrow \frac{1}{n}$
while $\|\Sigma_w\|_{\text{op}} \geq C$ do
    $v \leftarrow$ top eigenvector of $\Sigma_w$
    $\tau_i \leftarrow \langle v, x_i - \mu_w \rangle^2, \ \forall i \in [n]$
    $\tau_{\text{max}} \leftarrow \max_{i : w_i > 0} \tau_i$
    $w'_i \leftarrow w_i(1 - \tau_i / \tau_{\text{max}})$
end
Output $\mu_w$
```

To prove correctness and efficiency, we will first show that the update rule satisfies two useful properties: "safety condition" and "progress condition".

Lemma 2. *Consider the update rule in Algorithm 1:*

$$w'_i \leftarrow w_i(1 - \tau_i / \tau_{\text{max}}).$$

If $\sum_{\text{clean } i} w_i \tau_i < \sum_{\text{bad } i} w_i \tau_i$, then

- **(safety condition)** $\sum_{\text{clean } i}(w_i - w'_i) < \sum_{\text{bad } i}(w_i - w'_i)$ *i.e., the update removes more "bad mass" than "good mass", and*

- **(progress condition)** $\text{nnz}(w') < \text{nnz}(w)$, *where* $\text{nnz}(.)$ *denotes the number of non-zero entries.*

Proof. The progress condition is immediate from the definition of τ_{max}. Note that the safety condition follows by the following lemma (which we prove later):

Lemma 3. *Suppose $\|\Sigma_w\|_{\text{op}} \geq C$. Then $\sum_{\text{clean } i} w_i \tau_i < \sum_{\text{bad } i} w_i \tau_i$.*

This is because we have $w_i - w'_i = w_i \tau_i / \tau_{\text{max}}$ for all $i \in [n]$. □ (Lemma 2)

As the initial weights of the algorithm were all $1/n$, the safety condition implies the following invariant during the run of the algorithm:

$$\sum_{\text{clean } i} \left(\frac{1}{n} - w_i \right) < \sum_{\text{bad } i} \left(\frac{1}{n} - w_i \right) \tag{INV}$$

Note that it is apriori not clear how long the while-loop in Algorithm 1 runs. However, we can easily bound the number of iterations/updates:

Observation 1. *If* (INV) *is always maintained, then the (while-loop of) Algorithm 1 runs for at most $2\eta n$ iterations.*

Proof. Suppose that the algorithm runs for more than $2\eta n$ iterations. Then, we will show that (INV) gets violated in the $2\eta n$-th iteration: In each iteration, the total mass gets reduced by at least $\sum_i w_i \tau_i / \tau_{\text{max}} \geq 1/n$. (this is similar to the analysis in the proof of Lemma 3). Hence, we will have removed at least 2η mass in total. By (INV), this implies that at least η mass has been removed from bad points. Since there are only ηn bad points, the initial bad mass is η. Hence, $\sum_{\text{bad } i} = 0$ by the $2\eta n$-th iteration. Since (INV) should also be maintained at this point but there is no bad mass left, we conclude that the algorithm must terminate now, if it hasn't already. □

We now state two more observations.

Observation 2. *If* $\|\Sigma_w\|_{\text{op}} \lesssim C$*, it is safe to output* μ_w*, i.e.,* $\|\mu_g - \mu_w\| \lesssim O(\sqrt{\eta})$.

The above observation follows directly from the spectral signature lemma.

Observation 3. *If* (INV) *is always maintained, then* $w \in K_{2\eta}$.

Proof. We have

$$\sum_{\text{clean } i} \left(\frac{1}{n} - w_i \right) < \sum_{\text{bad } i} \left(\frac{1}{n} - w_i \right) \leq \eta,$$

so $\sum_{\text{all } i} w_i > 1 - 2\eta$. □

Note that Observation 2 immediately implies the desired error bound on the estimated mean of Algorithm 1. By Observation 1, we see that the number of iterations of the algorithm is also small i.e., $2\eta n$.

We will now finish with a proof of Lemma 3.

Proof of Lemma 3. Note

$$\begin{aligned}\sum_{\text{all } i} w_i \tau_i &= \sum_i w_i \langle v, x_i - \mu_w \rangle^2 \\ &= v^\top \Big(\sum_i w_i (x_i - \mu_w)(x_i - \mu_w)^\top \Big) v \\ &= v^\top \Sigma_w v = \|\Sigma_w\|_{\text{op}}.\end{aligned}$$

Hence, it suffices to show that $\sum_{\text{clean } i} w_i \tau_i < \frac{1}{2} \|\Sigma_w\|_{\text{op}}$.

We have

$$\begin{aligned}\sum_{\text{clean } i} w_i \tau_i &\leq \frac{1}{n} \sum_{x \in S_g} \langle v, x - \mu_w \rangle^2 && \text{(as each } w_i \leq 1/n\text{)} \\ &= \frac{1}{n} \sum_{x \in S_g} \Big(\langle v, x - \mu_g \rangle + \langle v, \mu_g - \mu_w \rangle \Big)^2 \\ &\leq \underbrace{\frac{2}{n} \sum_{x \in S_g} \langle v, x - \mu_g \rangle^2}_{A} + \underbrace{\frac{2}{n} \sum_{x \in S_g} \langle v, \mu_g - \mu_w \rangle^2}_{B}. && \text{(using } (a+b)^2 \leq 2(a^2+b^2)\text{)}\end{aligned}$$

Now we bound A and B separately:

$$A \leq 2v^\top \left(\frac{1}{|S_g|} \sum_{x \in S_g} (x - \mu_g)(x - \mu_g)^\top \right) v \lesssim 2, \qquad \text{(by assumption (2) from Setup)}$$

and

$$\begin{aligned}B &\leq 2\langle v, \mu_g - \mu_w \rangle^2 \\ &\leq 2\|\mu_g - \mu_w\|_2 \\ &\leq \sqrt{\eta} \cdot \left(1 + \sqrt{\|\Sigma_w\|_{\text{op}}} \right). && \text{(by the spectral signature lemma (Lemma 1))}\end{aligned}$$

Hence, provided η is a sufficiently small constant, if $\|\Sigma_w\|_{\text{op}} \geq C$ for a sufficiently large constant C, we get that

$$A + B \leq 2 + O\left(\sqrt{\eta} \cdot \left(1 + \sqrt{\|\Sigma_w\|_{\text{op}}} \right) \right) < \frac{1}{2} \|\Sigma_w\|_{\text{op}},$$

where the last inequality follows from the fact that the LHS is a linear function of $\sqrt{\|\Sigma_{\text{op}}\|}$, while the RHS is quadratic and hence dominates the LHS for sufficiently large C (note that the choice of C gets closer to 1 as η gets smaller).

Therefore, $\sum_{\text{clean } i} w_i \tau_i < \frac{1}{2} \|\Sigma_w\|_{\text{op}}$. □

This concludes the analysis of run-time and correctness of the robust mean estimation algorithm based on iterative filtering (assuming the spectral signature lemma).

References

[DKK+16] Ilias Diakonikolas, Gautam Kamath, Daniel M Kane, Jerry Li, K Moitra, and Alistair Stewart. Robust estimators in high dimensions without the computational intractability. foundations of computer science (focs). In *2016 IEEE 57th Annual Symposium*, 2016.

[Hub92] Peter J Huber. Robust estimation of a location parameter. In *Breakthroughs in statistics: Methodology and distribution*, pages 492–518. Springer, 1992.