High Dimensional Robust Sparse Regression

Liu Liu Yanyao Shen Tianyang Li Constantine Caramanis

The University of Texas at Austin

December, 2020

Introduction

- Large-scale statistical problems: both the dimension d and the sample size n may be large (possibly $n \ll d$).

- Low dimensional structures in the high dimensional setting.

Introduction

- Large-scale statistical problems: both the dimension d and the sample size n may be large (possibly $n \ll d$).

- Low dimensional structures in the high dimensional setting.

- Many examples of this:
  - Sparse regression.
  - Low rank matrix completion.
  - Low rank + sparse matrix decomposition.
  - etc...

Motivation

Well known that most state of the art approaches for these problems are fragile.

- Typically need very light tails.

- Data must be pristine: A single corrupted sample can arbitrarily corrupt the original maximum likelihood estimation.

Problem setup: robust estimation for sparse regression

Sparse regression model:

- dimensions: $n \ll d$.
- iid Gaussian X.
- $y_i = \boldsymbol{x}_i{}^T \boldsymbol{\beta}^* + \xi_i$.
- noise: $\xi_i \sim \mathcal{N}(0, \sigma^2)$.
- $\boldsymbol{\beta}^* \in \mathbb{R}^d$ is k-sparse.

Contamination model:

- we observe $z_i = (y_i, \boldsymbol{x}_i)$.
- $\{z_1, \cdots, z_n\} \sim (1-\epsilon)P + \epsilon Q$.
- P: sparse regression model .
- Q: *arbitrary* distribution.
- ϵ: *const fraction* of outliers.

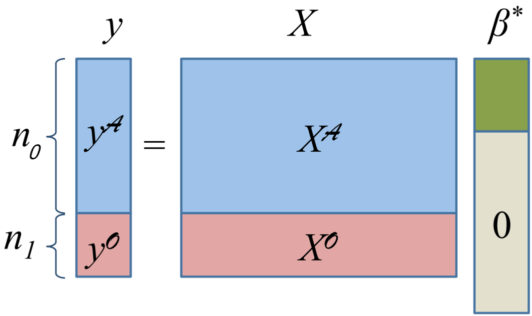
Problem setup: robust estimation for sparse regression

Sparse regression model:

- dimensions: $n \ll d$.
- iid Gaussian X.
- $y_i = \boldsymbol{x}_i{}^T \beta^* + \xi_i$.
- noise: $\xi_i \sim \mathcal{N}(0, \sigma^2)$.
- $\beta^* \in \mathbb{R}^d$ is k-sparse.

Contamination model:

- we observe $z_i = (y_i, \boldsymbol{x}_i)$.
- $\{z_1, \cdots, z_n\} \sim (1-\epsilon)P + \epsilon Q$.
- P: sparse regression model .
- Q: *arbitrary* distribution.
- ϵ: *const fraction* of outliers.



Related work

Robust regression

- [Li13][BJK15][DT19]: robust regression with corruptions only in y.
- [KKM18] [PSBR18] [DKK$^+$18] [DKS19]: low dimensional linear regression with corruptions in $\boldsymbol{x}$ and y, $n = \Omega(d)$ and $\epsilon = \text{const}$.
- [CCM13] [LLC19]: robust sparse regression resilient to corruptions in $\boldsymbol{x}$ and y, with $\epsilon = O(1/\sqrt{k})$.

Robust mean estimation

- [LRV16] [DKK$^+$16]: robust mean estimation with $\epsilon = \text{const}$, $n = \Omega(d)$.
- [BDLS17]: robust sparse mean estimation with $\epsilon = \text{const}$, $n = \Omega(k^2 \log(d))$. [a] This is based on the ellipsoid algorithm in [DKK$^+$16].

[a] [DKS16]: statistical query-based l.b. of $\Omega(k^2)$ on rob. sparse mean estimation.

Estimation tasks for robust sparse regression

Problem: ϵ-corrupted samples from robust sparse regression model, can we recover $\boldsymbol{\beta}^*$?

- [CCM13]: corruptions in $\boldsymbol{x}$ and y, but cannot deal with constant ϵ.
- [Gao17, LM16, LL$^+$20] show the minimax rate $O(\epsilon\sigma)$, but only provides exponential-time algorithm.
- [BDLS17] has sub-optimal rates depending on $\|\boldsymbol{\beta}^*\|_2$.
- [KKM18] [PSBR18] [DKK$^+$18] [DKS19]: recent advances in robust regression, but require at least $n = \Omega(d)$.
- [Li13][BJK15][DT19]: corruptions only in y.

Our approach

Algorithmic idea:

Iterative Hard Thresholding

+

Robust Sparse Mean Estimation on gradients

Our approach

Algorithmic idea:

Iterative Hard Thresholding

+

Robust Sparse Mean Estimation on gradients

Required ingredients:

- Robust Sparse Mean Estimation
- Stability of IHT

This work

- Meta-Theorem: stability of IHT. Given any Robust Sparse Mean Estimation sub-procedure, IHT has controlled error.
- We provide order-wise faster Robust Sparse Mean Estimation algorithm based on filtering, which is scalable and practical.
- With the ellipsoid algorithm, we have optimal rate of convergence.
- With the faster filtering algorithm, we can deal with unknown but sparse covariance matrix. Exact recovery when ϵ or σ goes to zero.

Iterative Hard Thresholding

We look at the gradient part of uncorrupted IHT:

$$\beta^{t+1} = P_k(\beta^t - \frac{1}{n}\sum_{i=1}^{n} \boldsymbol{g}_i^t),$$

where $\boldsymbol{g}_i^t = \boldsymbol{x}_i({\boldsymbol{x}_i}^T\beta^t - y_i)$ is gradient of the i^{th} sample $(y_i, \boldsymbol{x}_i)$.

Iterative Hard Thresholding

We look at the gradient part of uncorrupted IHT:

$$\beta^{t+1} = P_k(\beta^t - \frac{1}{n}\sum_{i=1}^{n} \boldsymbol{g}_i^t),$$

where $\boldsymbol{g}_i^t = \boldsymbol{x}_i(\boldsymbol{x}_i{}^T\beta^t - y_i)$ is gradient of the i^{th} sample $(y_i, \boldsymbol{x}_i)$.

If $\Sigma = I_d$, $\mathbb{E}(\boldsymbol{g}_i)$ is guaranteed to be $2k$-sparse:

$$\mathbb{E}_P(\boldsymbol{g}_i^t) = \mathbb{E}_P(\boldsymbol{x}_i\boldsymbol{x}_i{}^T(\beta^t - \beta^*)) = \beta^t - \beta^* = \boldsymbol{G}^t.$$

Iterative Hard Thresholding

We look at the gradient part of uncorrupted IHT:

$$\beta^{t+1} = P_k(\beta^t - \frac{1}{n}\sum_{i=1}^{n} \boldsymbol{g}_i^t),$$

where $\boldsymbol{g}_i^t = \boldsymbol{x}_i(\boldsymbol{x}_i{}^T\beta^t - y_i)$ is gradient of the i^{th} sample $(y_i, \boldsymbol{x}_i)$.

If $\Sigma = I_d$, $\mathbb{E}(\boldsymbol{g}_i)$ is guaranteed to be $2k$-sparse:

$$\mathbb{E}_P(\boldsymbol{g}_i^t) = \mathbb{E}_P(\boldsymbol{x}_i\boldsymbol{x}_i{}^T(\boldsymbol{\beta}^t - \boldsymbol{\beta}^*)) = \boldsymbol{\beta}^t - \boldsymbol{\beta}^* = \boldsymbol{G}^t.$$

When $\{y_i, \boldsymbol{x}_i\}_{i=1}^n$ come from $(1-\epsilon)P + \epsilon Q$, we can use robust sparse mean estimation on $\boldsymbol{G}^t$, and then use inexact IHT.

Robust Sparse Gradient Estimator (RSGE)

Definition 1 (RSGE)

We call $\widehat{\boldsymbol{G}}(\boldsymbol{\beta})$ a $\psi(\epsilon)$-RSGE, if given $\{\boldsymbol{g}_i\}_{i=1}^n$, $\widehat{\boldsymbol{G}}(\boldsymbol{\beta})$ guarantees

$$\left\|\widehat{\boldsymbol{G}}(\boldsymbol{\beta}) - \boldsymbol{G}(\boldsymbol{\beta})\right\|_2^2 \leq \alpha \left\|\boldsymbol{G}(\boldsymbol{\beta})\right\|_2^2 + \psi(\epsilon),$$

with high probability, where $\alpha \in (0, 0.1)$ is a constant.

Robust Sparse Gradient Estimator (RSGE)

Definition 1 (RSGE)

We call $\widehat{\boldsymbol{G}}(\boldsymbol{\beta})$ a $\psi(\epsilon)$-RSGE, if given $\{\boldsymbol{g}_i\}_{i=1}^n$, $\widehat{\boldsymbol{G}}(\boldsymbol{\beta})$ guarantees

$$\left\|\widehat{\boldsymbol{G}}(\boldsymbol{\beta}) - \boldsymbol{G}(\boldsymbol{\beta})\right\|_2^2 \leq \alpha \left\|\boldsymbol{G}(\boldsymbol{\beta})\right\|_2^2 + \psi(\epsilon),$$

with high probability, where $\alpha \in (0, 0.1)$ is a constant.

Theorem 1 (Thm. 2.1 in our paper)

IHT is stable. In particular, using an $\psi(\epsilon)$-RSGE as defined in Definition 1, IHT outputs $\widehat{\boldsymbol{\beta}}$, such that

$$\left\|\widehat{\boldsymbol{\beta}} - \boldsymbol{\beta}^*\right\|_2 = O\left(\sqrt{\psi(\epsilon)}\right),$$

with high probability.

Robust sparse regression with corrupted gradients

Algorithm 1: Robust sparse regression by RSGE

Input: Data samples $\{y_i, \boldsymbol{x}_i\}_{i=1}^{N}$, RSGE subroutine.
Output: The estimation $\widehat{\boldsymbol{\beta}}$
Split samples into T subsets of size n.
Initialize with $\boldsymbol{\beta}^0 = \mathbf{0}$.
for $t = 0$ to $T - 1$, **do**
    At current $\boldsymbol{\beta}^t$, calculate all gradients:

$$\boldsymbol{g}_i^t = \boldsymbol{x}_i \left(\boldsymbol{x}_i^\top \boldsymbol{\beta}^t - y_i \right), i \in [n].$$

    We use a RSGE to get $\widehat{\boldsymbol{G}}^t$.
    $\boldsymbol{\beta}^{t+1} = P_k \left(\boldsymbol{\beta}^t - \widehat{\boldsymbol{G}}^t \right).$
end for
Output the estimation $\widehat{\boldsymbol{\beta}} = \boldsymbol{\beta}^T$.

How to design RSGE?

Theorem 2 (RSGE by ellipsoid algorithm in [BDLS17], Cor. 3.1 in our paper)

With $n \geq \Omega(\frac{k^2 \log d}{\epsilon^2})$, we can guarantee

$$\|\widehat{\boldsymbol{G}}^t - \boldsymbol{G}^t\|_2^2 = O(\epsilon^2 \|\boldsymbol{G}^t\|_2^2 + \epsilon^2 \sigma^2).$$

Theorem 3

Combining Theorem 2 with Theorem 1, we have $\|\widehat{\beta} - \beta^*\|_2 = O(\epsilon\sigma)$.

How to design RSGE?

Theorem 2 (RSGE by ellipsoid algorithm in [BDLS17], Cor. 3.1 in our paper)

With $n \geq \Omega(\frac{k^2 \log d}{\epsilon^2})$, we can guarantee

$$\|\widehat{\boldsymbol{G}}^t - \boldsymbol{G}^t\|_2^2 = O(\epsilon^2 \|\boldsymbol{G}^t\|_2^2 + \epsilon^2 \sigma^2).$$

Theorem 3

Combining Theorem 2 with Theorem 1, we have $\|\widehat{\beta} - \beta^*\|_2 = O(\epsilon\sigma)$.

- This algorithm's time complexity is polynomial.
- However, it cannot handle unknown covariance.

How to design RSGE?

We provide a new, faster filtering algorithm.

Theorem 4 (RSGE by filtering algorithm, Cor. 4.1 in our paper)

With $n \geq \Omega(\frac{k^2 \log d}{\epsilon})$, we can guarantee

$$\|\widehat{\boldsymbol{G}}^t - \boldsymbol{G}^t\|_2^2 = O(\epsilon\|\boldsymbol{G}^t\|_2^2 + \epsilon\sigma^2).$$

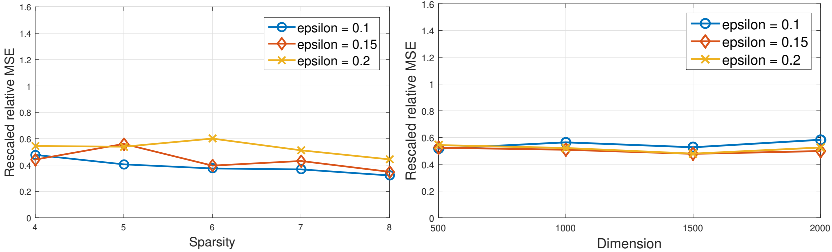
Theorem 5

Combining Theorem 1 and Theorem 4, we have $\|\widehat{\beta} - \beta^*\|_2 = O(\sqrt{\epsilon}\sigma)$.

- The new filtering algorithm is orderwise faster, at the expense of $\sqrt{\epsilon}$ rather than ϵ in the guarantee.
- This new filtering algorithm also works for unknown yet sparse covariance matrix.

Experimental results I: robust sparse mean estimation

We generate authentic samples through $\boldsymbol{g}_i = \boldsymbol{x}_i \boldsymbol{x}_i^\top \boldsymbol{G}$, where $\boldsymbol{G}$ is k-sparse. The rescaled relative MSE: $\|\widehat{\boldsymbol{G}} - \boldsymbol{G}\|_2^2 / (\epsilon \|\boldsymbol{G}\|_2^2)$ should be independent of the parameters $\{\epsilon, k, d\}$.

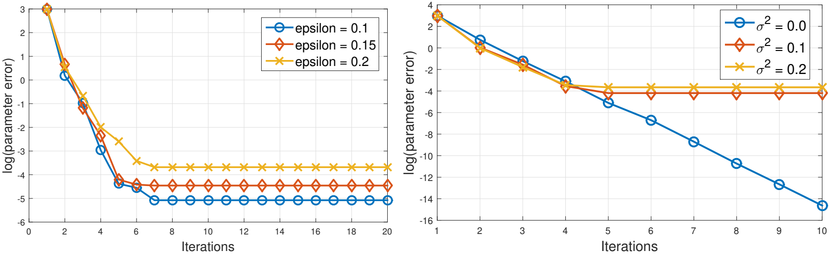


(a) Rescaled relative MSE vs. sparsity. (b) Rescaled relative MSE vs. dimension.

Figure: Sample complexity $n \propto k^2 \log(d)/\epsilon$. Different curves for $\epsilon \in \{0.1, 0.15, 0.2\}$ are the average of 15 trials.

Experimental results II: robust sparse regression

We use filtering algorithm as our RSGE, and generate authentic samples $y_i = \boldsymbol{x}_i^\top \boldsymbol{\beta}^* + \xi_i$. As expected, the convergence is linear, and flattens out at the level of the final error.



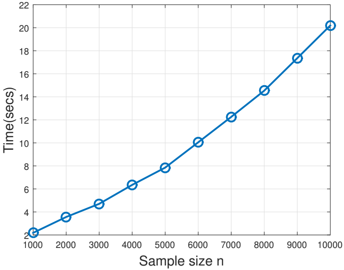
(a) $\log(\|\boldsymbol{\beta}^t - \boldsymbol{\beta}^*\|_2^2)$ vs. iterates.

(b) $\log(\|\boldsymbol{\beta}^t - \boldsymbol{\beta}^*\|_2^2)$ vs. iterates.

Figure: In all cases, we fix $k = 5$, $d = 500$, and choose the sample complexity $n \propto 1/\epsilon$. (2a) has fixed $\sigma^2 = 0.1$. (2b) has fixed $\epsilon = 0.1$.

Experimental results III: Large scale experiments

The wall clock time vs. the sample size or the dimensionality.



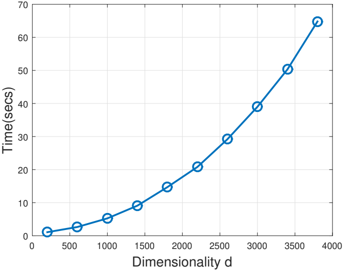


Figure: In both plots, we use $\epsilon = 0.1$. In the left plot, we fix $d = 500$ and in the right plot, we fix $n = 1000$.

Other Important Directions

- What if the gradients are not sparse? For example: for general (non-sparse, non-identity) covariance.

- Then RSGE cannot be used! It is too much to ask for

$$\left\| \widehat{\boldsymbol{G}}(\boldsymbol{\beta}) - \boldsymbol{G}(\boldsymbol{\beta}) \right\|_2^2$$

  to be small.

- Different tools/ideas are needed. For some results along these lines, see: `https://arxiv.org/abs/1901.08237`

Our contribution

- Sparse regression algorithm that is resilient to a constant fraction of arbitrary outliers. Our algorithm requires $n = \Omega(k^2 \log d)$ samples.

*[Gao17]: this error rate is minimax optimal under the ϵ-contamination model.

Our contribution

- Sparse regression algorithm that is resilient to a constant fraction of arbitrary outliers. Our algorithm requires $n = \Omega(k^2 \log d)$ samples.

- Meta-theorem which allows the use of any robust sparse mean estimation subroutine:
  - By ellipsoid algorithm in [BDLS17], we can recover β^* within additive error $O(\epsilon\sigma)$.*

*[Gao17]: this error rate is minimax optimal under the ϵ-contamination model.

Our contribution

- Sparse regression algorithm that is resilient to a constant fraction of arbitrary outliers. Our algorithm requires $n = \Omega(k^2 \log d)$ samples.

- Meta-theorem which allows the use of any robust sparse mean estimation subroutine:
  - By ellipsoid algorithm in [BDLS17], we can recover $\boldsymbol{\beta}^*$ within additive error $O(\epsilon\sigma)$.*

- Efficient filtering algorithm for robust sparse mean estimation.
  - By this algorithm, we can recover $\boldsymbol{\beta}^*$ within additive error $O(\sqrt{\epsilon}\sigma)$.
  - The filtering algorithm is practical and **faster by at least d^2**.
- In particular: **exact recovery as $\sigma \to 0$.**

*[Gao17]: this error rate is minimax optimal under the ϵ-contamination model.

For more information please refer to our paper

Liu Liu, Yanyao Shen, Tianyang Li, Constantine Caramanis.
High Dimensional Robust Sparse Regression.
`https://arxiv.org/abs/1805.11643`

References I

[BDLS17] Sivaraman Balakrishnan, Simon S. Du, Jerry Li, and Aarti Singh. Computationally efficient robust sparse estimation in high dimensions. In *Proceedings of the 2017 Conference on Learning Theory*, 2017.

[BJK15] Kush Bhatia, Prateek Jain, and Purushottam Kar. Robust regression via hard thresholding. In *Advances in Neural Information Processing Systems*, pages 721–729, 2015.

[CCM13] Yudong Chen, Constantine Caramanis, and Shie Mannor. Robust sparse regression under adversarial corruption. In *International Conference on Machine Learning*, pages 774–782, 2013.

[DKK$^+$16] Ilias Diakonikolas, Gautam Kamath, Daniel M Kane, Jerry Li, Ankur Moitra, and Alistair Stewart. Robust estimators in high dimensions without the computational intractability. In *Foundations of Computer Science (FOCS), 2016 IEEE 57th Annual Symposium on*, pages 655–664. IEEE, 2016.

[DKK$^+$18] Ilias Diakonikolas, Gautam Kamath, Daniel M Kane, Jerry Li, Jacob Steinhardt, and Alistair Stewart. Sever: A robust meta-algorithm for stochastic optimization. *arXiv preprint arXiv:1803.02815*, 2018.

[DKS16] Ilias Diakonikolas, Daniel M Kane, and Alistair Stewart. Statistical query lower bounds for robust estimation of high-dimensional gaussians and gaussian mixtures. *arXiv preprint arXiv:1611.03473*, 2016.

References II

[DKS19] Ilias Diakonikolas, Weihao Kong, and Alistair Stewart. Efficient algorithms and lower bounds for robust linear regression. In *Proceedings of the Thirtieth Annual ACM-SIAM Symposium on Discrete Algorithms*, pages 2745–2754. SIAM, 2019.

[DT19] Arnak Dalalyan and Philip Thompson. Outlier-robust estimation of a sparse linear model using ℓ_1-penalized huber's *m*-estimator. In *Advances in Neural Information Processing Systems*, pages 13188–13198, 2019.

[Gao17] Chao Gao. Robust regression via mutivariate regression depth. *arXiv preprint arXiv:1702.04656*, 2017.

[KKM18] Adam Klivans, Pravesh K Kothari, and Raghu Meka. Efficient algorithms for outlier-robust regression. *arXiv preprint arXiv:1803.03241*, 2018.

[Li13] Xiaodong Li. Compressed sensing and matrix completion with constant proportion of corruptions. *Constructive Approximation*, 37(1):73–99, 2013.

[LL$^+$20] Guillaume Lecué, Matthieu Lerasle, et al. Robust machine learning by median-of-means: theory and practice. *Annals of Statistics*, 48(2):906–931, 2020.

[LLC19] Liu Liu, Tianyang Li, and Constantine Caramanis. High dimensional robust m-estimation: Arbitrary corruption and heavy tails. *arXiv preprint arXiv:1901.08237*, 2019.

References III

[LM16] Gabor Lugosi and Shahar Mendelson. Risk minimization by median-of-means tournaments. *arXiv preprint arXiv:1608.00757*, 2016.

[LRV16] Kevin A Lai, Anup B Rao, and Santosh Vempala. Agnostic estimation of mean and covariance. In *Foundations of Computer Science (FOCS), 2016 IEEE 57th Annual Symposium on*, pages 665–674. IEEE, 2016.

[PSBR18] Adarsh Prasad, Arun Sai Suggala, Sivaraman Balakrishnan, and Pradeep Ravikumar. Robust estimation via robust gradient estimation. *arXiv preprint arXiv:1802.06485*, 2018.