# Robust sparse estimation: An overview

**Ankit Pensia**

IBM Research

Workshop on New Frontiers in Robust Statistics, 2024

# Overview

# Introducing structured robust estimation

- So far, we have seen *unstructured* parameter estimation

**Problem statement.** (Robust mean estimation)
Let $\mathcal{P}$ be an unknown nice distribution over $\mathbb{R}^d$ with mean $\mu$

| | |
|---|---|
| Input: | corrupted samples from $\mathcal{P}$ |
| Output: | $\widehat{\mu}$ such that $\|\widehat{\mu} - \mu\|_2$ is small w.h.p. |

# Introducing structured robust estimation

- So far, we have seen *unstructured* parameter estimation

**Problem statement.** (Robust mean estimation)
Let $\mathcal{P}$ be an unknown nice distribution over $\mathbb{R}^d$ with mean $\mu$

| | |
|---|---|
| Input: | corrupted samples from $\mathcal{P}$ |
| Output: | $\widehat{\mu}$ such that $\|\widehat{\mu} - \mu\|_2$ is small w.h.p. |

- Sample complexity: $\Theta(d)$

# Introducing structured robust estimation

- So far, we have seen *unstructured* parameter estimation

**Problem statement.** (Robust mean estimation)
Let $\mathcal{P}$ be an unknown nice distribution over $\mathbb{R}^d$ with mean $\mu$

| | |
|---|---|
| Input: | corrupted samples from $\mathcal{P}$ |
| Output: | $\widehat{\mu}$ such that $\|\widehat{\mu} - \mu\|_2$ is small w.h.p. |

- Sample complexity: $\Theta(d)$

Can we reduce the sample complexity if $\mu$ is **structured**?

# Introducing structured robust estimation

- So far, we have seen *unstructured* parameter estimation

**Problem statement.** (Robust mean estimation)
Let $\mathcal{P}$ be an unknown nice distribution over $\mathbb{R}^d$ with mean $\mu$

| | |
|---|---|
| Input: | corrupted samples from $\mathcal{P}$ |
| Output: | $\widehat{\mu}$ such that $\|\widehat{\mu} - \mu\|_2$ is small w.h.p. |

- Sample complexity: $\Theta(d)$

Can we reduce the sample complexity if $\mu$ is **structured**?

in this talk: sparsity

# Motivating sparsity

- Many data distributions are sparse
  - Images in wavelet basis
  - Bioinformatics

# Motivating sparsity

- Many data distributions are sparse
  - Images in wavelet basis
  - Bioinformatics


- A classical concept in statistics
  - Extra information about the true parameter
  - Allows us to get smaller error (alternatively, **lower sample complexity**)

# Motivating sparsity

- Many data distributions are sparse
  - Images in wavelet basis
  - Bioinformatics
- A classical concept in statistics
  - Extra information about the true parameter
  - Allows us to get smaller error (alternatively, lower sample complexity)

**This talk:** Utilizing the structure of sparsity **robustly**.

# Our question: efficient robust sparse estimation

**Problem statement.** (Robust sparse mean estimation)
Let $\mathcal{P}$ be an unknown nice distribution over $\mathbb{R}^d$ with a ***k*-sparse** mean $\mu$

Input: corrupted samples from $\mathcal{P}$

Output: $\widehat{\mu}$ such that $\|\widehat{\mu} - \mu\|_2$ is small w.h.p.

# Our question: efficient robust sparse estimation

**Problem statement.** (Robust sparse mean estimation)
Let $\mathcal{P}$ be an unknown nice distribution over $\mathbb{R}^d$ with a $k$-sparse mean $\mu$

| | |
|---|---|
| Input: | corrupted samples from $\mathcal{P}$ |
| Output: | $\widehat{\mu}$ such that $\|\widehat{\mu} - \mu\|_2$ is small w.h.p. |

- Sample complexity: $\Theta(k \log d)$
  - Huge reduction in sample complexity!

# Our question: efficient robust sparse estimation

**Problem statement.** (Robust sparse mean estimation)
Let $\mathcal{P}$ be an unknown nice distribution over $\mathbb{R}^d$ with a $k$-sparse mean $\mu$

| | |
|---|---|
| Input: | corrupted samples from $\mathcal{P}$ |
| Output: | $\widehat{\mu}$ such that $\|\widehat{\mu} - \mu\|_2$ is small w.h.p. |

- Sample complexity: $\Theta(k \log d)$
  - Huge reduction in sample complexity!
- Alas, achieving $\mathbf{o(k^2)}$ sample complexity is *computationally hard*

# Our question: efficient robust sparse estimation

**Problem statement.** (Robust sparse mean estimation)
Let $\mathcal{P}$ be an unknown nice distribution over $\mathbb{R}^d$ with a $k$-sparse mean $\mu$

Input: corrupted samples from $\mathcal{P}$

Output: $\widehat{\mu}$ such that $\|\widehat{\mu} - \mu\|_2$ is small w.h.p.

- Sample complexity: $\Theta(k \log d)$
  - Huge reduction in sample complexity!
- Alas, achieving $\mathbf{o(k^2)}$ sample complexity is *computationally hard*

**Relaxed goal:** Achieving $\mathrm{poly}(k, \log d)$ sample complexity, **efficiently**

# Overview

- **Background**
  - **Algorithmic framework**
- Polynomial-time algorithms
  - Some improvements
- Quadratic-time algorithms
- Subquadratic-time algorithms

# Prelude: a path towards robust dense estimation

# Prelude: a path towards robust dense estimation

- Suppose the inliers are sampled from $\mathcal{N}(\mu, I)$
- (Reducing to one-dimension) $\|\widehat{\mu} - \mu\|_2 = \sup_v \langle v, \widehat{\mu} - \mu \rangle$
  - Equivalent to ensuring accurate estimates in all directions $v$

# Prelude: a path towards robust dense estimation

- Suppose the inliers are sampled from $\mathcal{N}(\mu, I)$
- (Reducing to one-dimension) $\|\widehat{\mu} - \mu\|_2 = \sup_v \langle v, \widehat{\mu} - \mu \rangle$
  - Equivalent to ensuring accurate estimates in all directions $v$
- Key insight **[DKKLMS16; LRV16]**: For any direction $v$,
  - The sample mean is accurate **if** the sample variance is bounded
  - **Else if** the sample variance is large, we can filter the outliers

**Algorithmic template**: robust (dense) estimation

1. While there exists a direction $v$ with large variance:
   1.1 Filter each point $x$ using $v^\top x$
2. $\widehat{\mu} \leftarrow$ sample mean

[DKKLMS16] I. Diakonikolas, G. Kamath, D. Kane, J. Li, A. Moitra, A. Stewart. Robust estimators in high... *FOCS*. 2016
[LRV16] K. A. Lai, A. B. Rao, S. Vempala. Agnostic Estimation of Mean and Covariance. *FOCS*. 2016

# Prelude: a path towards robust dense estimation

- Suppose the inliers are sampled from $\mathcal{N}(\mu, I)$
- (Reducing to one-dimension) $\|\widehat{\mu} - \mu\|_2 = \sup_v \langle v, \widehat{\mu} - \mu \rangle$
  - Equivalent to ensuring accurate estimates in all directions $v$
- Key insight [DKKLMS16; LRV16]: For any direction $v$,
  - The sample mean is accurate **if** the sample variance is bounded
  - **Else if** the sample variance is large, we can filter the outliers

**Algorithmic template**: robust (dense) estimation

1. While there exists a direction $v$ with large variance:
   1.1 Filter each point $x$ using $v^\top x$
2. $\widehat{\mu} \leftarrow$ sample mean

- sample mean and covariance should be accurate for **clean data**

[DKKLMS16] I. Diakonikolas, G. Kamath, D. Kane, J. Li, A. Moitra, A. Stewart. Robust estimators in high... *FOCS*. 2016
[LRV16] K. A. Lai, A. B. Rao, S. Vempala. Agnostic Estimation of Mean and Covariance. *FOCS*. 2016

# Prelude: a path towards robust dense estimation

- Suppose the inliers are sampled from $\mathcal{N}(\mu, I)$
- (Reducing to one-dimension) $\|\widehat{\mu} - \mu\|_2 = \sup_v \langle v, \widehat{\mu} - \mu \rangle$
  - Equivalent to ensuring accurate estimates in all directions $v$
- Key insight [DKKLMS16; LRV16]: For any direction $v$,
  - The sample mean is accurate if the sample variance is bounded
  - Else if the sample variance is large, we can filter the outliers

**Algorithmic template**: robust (dense) estimation

1. While there exists a direction $v$ with large variance:
   1.1 Filter each point $x$ using $v^\top x$
2. $\widehat{\mu} \leftarrow$ sample mean

- sample mean and covariance should be accurate for **clean data** and **all large subsets**

[DKKLMS16] I. Diakonikolas, G. Kamath, D. Kane, J. Li, A. Moitra, A. Stewart. Robust estimators in high... *FOCS*. 2016
[LRV16] K. A. Lai, A. B. Rao, S. Vempala. Agnostic Estimation of Mean and Covariance. *FOCS*. 2016

# Prelude: a path towards robust dense estimation

- Suppose the inliers are sampled from $\mathcal{N}(\mu, I)$
- (Reducing to one-dimension) $\|\widehat{\mu} - \mu\|_2 = \sup_v \langle v, \widehat{\mu} - \mu \rangle$
  - Equivalent to ensuring accurate estimates in all directions $v$
- Key insight [DKKLMS16; LRV16]: For any direction $v$,
  - The sample mean is accurate if the sample variance is bounded
  - Else if the sample variance is large, we can filter the outliers

**Algorithmic template**: robust (dense) estimation

1. While there exists a direction $v$ with large variance:
   1.1 Filter each point $x$ using $v^\top x$
2. $\widehat{\mu} \leftarrow$ sample mean

- sample mean and covariance should be accurate for **clean data** and **all large subsets** (termed **stability**)

[DKKLMS16] I. Diakonikolas, G. Kamath, D. Kane, J. Li, A. Moitra, A. Stewart. Robust estimators in high... *FOCS*. 2016
[LRV16] K. A. Lai, A. B. Rao, S. Vempala. Agnostic Estimation of Mean and Covariance. *FOCS*. 2016

# Next, a path towards robust sparse estimation

- Suppose the inliers are sampled from $\mathcal{N}(\mu, I)$, where $\mu$ is **$k$-sparse**

# Next, a path towards robust sparse estimation

- Suppose the inliers are sampled from $\mathcal{N}(\mu, I)$, where $\mu$ is **$k$-sparse**
- (Projections) $\|\mathrm{HardThresh}(\widehat{\mu}) - \mu\|_2 \lesssim \sup_{v:\textbf{k-sparse}} \langle v, \widehat{\mu} - \mu \rangle$
  - Only the sparse directions matter

# Next, a path towards robust sparse estimation

- Suppose the inliers are sampled from $\mathcal{N}(\mu, I)$, where $\mu$ is ***k*-sparse**
- (Projections) $\|\mathrm{HardThresh}(\widehat{\mu}) - \mu\|_2 \lesssim \sup_{v:k\text{-sparse}} \langle v, \widehat{\mu} - \mu \rangle$
  - Only the sparse directions matter

**Algorithmic template**: robust **sparse** estimation

1. While there exists a **sparse** direction $v$ with large variance:
   1.1 Filter points after projecting onto $v$
2. Return $\mathrm{HardThresh}(\text{sample mean})$

# Next, a path towards robust sparse estimation

- Suppose the inliers are sampled from $\mathcal{N}(\mu, I)$, where $\mu$ is $k$-sparse
- (Projections) $\|\mathrm{HardThresh}(\widehat{\mu}) - \mu\|_2 \lesssim \sup_{v:k\text{-sparse}} \langle v, \widehat{\mu} - \mu \rangle$
  - Only the sparse directions matter

**Algorithmic template**: robust sparse estimation

1. While there exists a sparse direction $v$ with large variance:
   1.1 Filter points after projecting onto $v$
2. Return $\mathrm{HardThresh}(\text{sample mean})$

intractable!

# Next, a path towards robust sparse estimation

- Suppose the inliers are sampled from $\mathcal{N}(\mu, I)$, where $\mu$ is $k$-sparse
- (Projections) $\|\mathrm{HardThresh}(\widehat{\mu}) - \mu\|_2 \lesssim \sup_{v:k\text{-sparse}} \langle v, \widehat{\mu} - \mu \rangle$
  - Only the sparse directions matter

**Algorithmic template**: robust sparse estimation

1. While there exists a sparse direction $v$ with large variance:
   1.1 Filter points after projecting onto $v$
2. Return $\mathrm{HardThresh}(\text{sample mean})$

intractable!

**How to design an efficient subroutine?**

# Towards efficient estimation via relaxed certificates

- Sparse operator norm $\|\mathbf{A}\|_{\mathrm{op},k} := \max_{v:k\text{-sparse}} v^\top \mathbf{A} v$

# Towards efficient estimation via relaxed certificates

- Sparse operator norm $\|\mathbf{A}\|_{\mathrm{op},k} := \max_{v:k\text{-sparse}} v^\top \mathbf{A} v$
- Robustness requires ensuring that $\|\widehat{\boldsymbol{\Sigma}} - \mathbf{I}\|_{\mathrm{op},k}$ be small

# Towards efficient estimation via relaxed certificates

- Sparse operator norm $\|\mathbf{A}\|_{\text{op},k} := \max_{v:k\text{-sparse}} v^\top \mathbf{A} v$
- Robustness requires ensuring that $\|\widehat{\mathbf{\Sigma}} - \mathbf{I}\|_{\text{op},k}$ be small

**Instead, we design an efficient certificate $f(\cdot)$ such that:**

1. $\|\mathbf{A}\|_{\text{op},k} \leq f(\mathbf{A})$ and ...

# Towards efficient estimation via relaxed certificates

- Sparse operator norm $\|\mathbf{A}\|_{\mathrm{op},k} := \max_{v:k\text{-sparse}} v^\top \mathbf{A} v$
- Robustness requires ensuring that $\|\widehat{\mathbf{\Sigma}} - \mathbf{I}\|_{\mathrm{op},k}$ be small

**Instead, we design an efficient certificate $f(\cdot)$ such that:**

1. $\|\mathbf{A}\|_{\mathrm{op},k} \leq f(\mathbf{A})$ and ...
2. $f(\widehat{\mathbf{\Sigma}} - \mathbf{I})$ is small for **clean data**

# Towards efficient estimation via relaxed certificates

- Sparse operator norm $\|\mathbf{A}\|_{\text{op},k} := \max_{v:k\text{-sparse}} v^\top \mathbf{A} v$
- Robustness requires ensuring that $\|\widehat{\mathbf{\Sigma}} - \mathbf{I}\|_{\text{op},k}$ be small

**Instead, we design an efficient certificate $f(\cdot)$ such that:**

1. $\|\mathbf{A}\|_{\text{op},k} \leq f(\mathbf{A})$ and ...
2. $f(\widehat{\mathbf{\Sigma}} - \mathbf{I})$ is small for clean data and **its all large subsets** (**$f$-stability**)

# Towards efficient estimation via relaxed certificates

- Sparse operator norm $\|\mathbf{A}\|_{\mathrm{op},k} := \max_{v:k\text{-sparse}} v^\top \mathbf{A} v$
- Robustness requires ensuring that $\|\widehat{\boldsymbol{\Sigma}} - \mathbf{I}\|_{\mathrm{op},k}$ be small

**Instead, we design an efficient certificate $f(\cdot)$ such that:**

1. $\|\mathbf{A}\|_{\mathrm{op},k} \leq f(\mathbf{A})$ and ...
2. $f(\widehat{\boldsymbol{\Sigma}} - \mathbf{I})$ is small for clean data and its all large subsets ($f$-stability)

**Algorithmic template**: Robust sparse estimation, **efficiently**

1. While $f(\widehat{\boldsymbol{\Sigma}} - \mathbf{I})$ large:
   1.1 Filter points and update $\widehat{\boldsymbol{\Sigma}}$
2. Return $\mathrm{HardThresh}(\text{sample mean})$

# Towards efficient estimation via relaxed certificates

- Sparse operator norm $\|\mathbf{A}\|_{\mathrm{op},k} := \max_{v:k\text{-sparse}} v^\top \mathbf{A} v$
- Robustness requires ensuring that $\|\widehat{\boldsymbol{\Sigma}} - \mathbf{I}\|_{\mathrm{op},k}$ be small

**Instead, we design an efficient certificate $f(\cdot)$ such that:**

1. $\|\mathbf{A}\|_{\mathrm{op},k} \leq f(\mathbf{A})$ and ...
2. $f(\widehat{\boldsymbol{\Sigma}} - \mathbf{I})$ is small for clean data and its all large subsets ($f$-stability)

**Algorithmic template**: Robust sparse estimation, efficiently

1. While $f(\widehat{\boldsymbol{\Sigma}} - \mathbf{I})$ large:
   1.1 Filter points and update $\widehat{\boldsymbol{\Sigma}}$
2. Return $\mathrm{HardThresh}(\text{sample mean})$

**Better certificates $\implies$ better algorithms**

# Overview

- Background
  - Algorithmic framework
- **Polynomial-time algorithms**
  - Some improvements
- Quadratic-time algorithms
- Subquadratic-time algorithms

# An approach via semidefinite programs

- Efficient algorithms first developed in [BDLS17].

[BDLS17] S. Balakrishnan, S. S. Du, J. Li, A. Singh. Computationally Efficient Robust Sparse Estimation.. *COLT*. 2017

# An approach via semidefinite programs

- Efficient algorithms first developed in [BDLS17].
- Recall $\|\mathbf{A}\|_{\mathrm{op},k} := \max_{v:k\text{-sparse}} |\langle vv^\top, \mathbf{A}\rangle|$

[BDLS17] S. Balakrishnan, S. S. Du, J. Li, A. Singh. Computationally Efficient Robust Sparse Estimation.. *COLT*. 2017

# An approach via semidefinite programs

- Efficient algorithms first developed in [BDLS17].
- Recall $\|\mathbf{A}\|_{\text{op},k} := \max_{v:k\text{-sparse}} |\langle vv^\top, \mathbf{A}\rangle|$
- Inspired by [dGJL07] , they defined

$$\|\mathbf{A}\|_{\mathcal{X}_k} := \max_{\mathbf{M}\in\mathcal{X}_k} |\langle \mathbf{M}, \mathbf{A}\rangle|,$$

where

$$\mathcal{X}_k := \{\mathbf{M} \succeq 0 : \text{tr}(\mathbf{M}) = 1, \|\mathbf{M}\|_1 \leq k\}$$

[BDLS17] S. Balakrishnan, S. S. Du, J. Li, A. Singh. Computationally Efficient Robust Sparse Estimation.. *COLT*. 2017
[dGJL07] A. d'Aspremont, L. Ghaoui, M. Jordan, G. Lanckriet. A direct formulation for sparse pca using SDP. 2007

# An approach via semidefinite programs

- Efficient algorithms first developed in [BDLS17].
- Recall $\|\mathbf{A}\|_{\mathrm{op},k} := \max_{v:k\text{-sparse}} |\langle vv^\top, \mathbf{A}\rangle|$
- Inspired by [dGJL07] , they defined

$$\|\mathbf{A}\|_{\mathcal{X}_k} := \max_{\mathbf{M}\in\mathcal{X}_k} |\langle \mathbf{M}, \mathbf{A}\rangle|,$$

where

$$\mathcal{X}_k := \{\mathbf{M} \succeq 0 : \mathrm{tr}(\mathbf{M}) = 1, \|\mathbf{M}\|_1 \le k\}$$

- Valid relaxation: $vv^\top \in \mathcal{X}_k$ (dropped rank constraint; $\|v\|_1 \le \sqrt{k}$)

---

[BDLS17] S. Balakrishnan, S. S. Du, J. Li, A. Singh. Computationally Efficient Robust Sparse Estimation.. *COLT*. 2017
[dGJL07] A. d'Aspremont, L. Ghaoui, M. Jordan, G. Lanckriet. A direct formulation for sparse pca using SDP. 2007

# An approach via semidefinite programs

- Efficient algorithms first developed in [BDLS17].
- Recall $\|\mathbf{A}\|_{\mathrm{op},k} := \max_{v:k\text{-sparse}} |\langle vv^\top, \mathbf{A}\rangle|$
- Inspired by [dGJL07] , they defined

$$\|\mathbf{A}\|_{\mathcal{X}_k} := \max_{\mathbf{M}\in\mathcal{X}_k} |\langle \mathbf{M}, \mathbf{A}\rangle|,$$

where

$$\mathcal{X}_k := \{\mathbf{M} \succeq 0 : \mathrm{tr}(\mathbf{M}) = 1, \|\mathbf{M}\|_1 \leq k\}$$

- Valid relaxation: $vv^\top \in \mathcal{X}_k$ (dropped rank constraint; $\|v\|_1 \leq \sqrt{k}$)

**SDP-stability**. A set $S$ is SDP-stable w.r.t. $\mu$ if for all large $S' \subset S$

[BDLS17] S. Balakrishnan, S. S. Du, J. Li, A. Singh. Computationally Efficient Robust Sparse Estimation.. *COLT*. 2017
[dGJL07] A. d'Aspremont, L. Ghaoui, M. Jordan, G. Lanckriet. A direct formulation for sparse pca using SDP. 2007

# An approach via semidefinite programs

- Efficient algorithms first developed in [BDLS17].
- Recall $\|\mathbf{A}\|_{\mathrm{op},k} := \max_{v:k\text{-sparse}} |\langle vv^\top, \mathbf{A}\rangle|$
- Inspired by [dGJL07] , they defined

$$\|\mathbf{A}\|_{\mathcal{X}_k} := \max_{\mathbf{M}\in\mathcal{X}_k} |\langle \mathbf{M}, \mathbf{A}\rangle|,$$

where

$$\mathcal{X}_k := \{\mathbf{M} \succeq 0 : \mathrm{tr}(\mathbf{M}) = 1, \|\mathbf{M}\|_1 \leq k\}$$

- Valid relaxation: $vv^\top \in \mathcal{X}_k$ (dropped rank constraint; $\|v\|_1 \leq \sqrt{k}$)

**SDP-stability.** A set $S$ is SDP-stable w.r.t. $\mu$ if for all large $S' \subset S$

- (Mean) $\sup_{v:k\text{-sparse}} \langle v, \mu_{S'} - \mu\rangle$ is small

[BDLS17] S. Balakrishnan, S. S. Du, J. Li, A. Singh. Computationally Efficient Robust Sparse Estimation.. *COLT*. 2017
[dGJL07] A. d'Aspremont, L. Ghaoui, M. Jordan, G. Lanckriet. A direct formulation for sparse pca using SDP. 2007

# An approach via semidefinite programs

- Efficient algorithms first developed in [BDLS17].
- Recall $\|\mathbf{A}\|_{\mathrm{op},k} := \max_{v:k\text{-sparse}} |\langle vv^\top, \mathbf{A}\rangle|$
- Inspired by [dGJL07] , they defined

$$\|\mathbf{A}\|_{\mathcal{X}_k} := \max_{\mathbf{M}\in\mathcal{X}_k} |\langle \mathbf{M}, \mathbf{A}\rangle|,$$

where

$$\mathcal{X}_k := \{\mathbf{M} \succeq 0 : \mathrm{tr}(\mathbf{M}) = 1, \|\mathbf{M}\|_1 \le k\}$$

- Valid relaxation: $vv^\top \in \mathcal{X}_k$ (dropped rank constraint; $\|v\|_1 \le \sqrt{k}$)

**SDP-stability.** A set $S$ is SDP-stable w.r.t. $\mu$ if for all large $S' \subset S$

- (Mean) $\sup_{v:k\text{-sparse}}\langle v, \mu_{S'} - \mu\rangle$ is small
- (Covariance) $\|\mathbf{\Sigma}_{S'} - \mathbf{I}\|_{\mathcal{X}_k}$ is small

---

[BDLS17] S. Balakrishnan, S. S. Du, J. Li, A. Singh. Computationally Efficient Robust Sparse Estimation.. *COLT*. 2017
[dGJL07] A. d'Aspremont, L. Ghaoui, M. Jordan, G. Lanckriet. A direct formulation for sparse pca using SDP. 2007

# Robust sparse mean estimation in polynomial time [BDLS17]

# Robust sparse mean estimation in polynomial time [BDLS17]

**Theorem: (BDLS17)**

Given $\epsilon$-contaminated samples from an isotropic subgaussian distribution with $k$-sparse mean $\mu$, a **polynomial-time** algorithm to compute $\widehat{\mu}$:

---

[BDLS17] S. Balakrishnan, S. S. Du, J. Li, A. Singh. Computationally Efficient Robust Sparse Estimation.. *COLT*. 2017

# Robust sparse mean estimation in polynomial time [BDLS17]

**Theorem: (BDLS17)**

Given $\epsilon$-contaminated samples from an isotropic subgaussian distribution with $k$-sparse mean $\mu$, a **polynomial-time** algorithm to compute $\widehat{\mu}$:

- (sample complexity) $n = \widetilde{O}\left(k^2/\epsilon^2\right)$ samples
- (error) $\|\widehat{\mu} - \mu\|_2 = \widetilde{O}(\epsilon)$

- Near-optimal asymptotic error
- Near-optimal *computational* sample complexity

[BDLS17] S. Balakrishnan, S. S. Du, J. Li, A. Singh. Computationally Efficient Robust Sparse Estimation.. *COLT*. 2017

# Robust sparse mean estimation in polynomial time [BDLS17]

**Theorem: (BDLS17)**

Given $\epsilon$-contaminated samples from an isotropic subgaussian distribution with $k$-sparse mean $\mu$, a **polynomial-time** algorithm to compute $\widehat{\mu}$:

- (sample complexity) $n = \widetilde{O}\left(k^2/\epsilon^2\right)$ samples
- (error) $\|\widehat{\mu} - \mu\|_2 = \widetilde{O}(\epsilon)$

- Near-optimal asymptotic error
- Near-optimal *computational* sample complexity
- Runtime: polynomial but existing SDP solvers are **impractical**
  - Current bounds: $\Omega(d^4)$ time
  - **Open problem:** design faster solvers for this SDP

[BDLS17] S. Balakrishnan, S. S. Du, J. Li, A. Singh. Computationally Efficient Robust Sparse Estimation.. *COLT*. 2017

# Proof sketch: stability with a small number of samples

$\mathcal{X}_k := \{\mathbf{M} \succeq 0 : \mathrm{tr}(\mathbf{M}) = 1, \|\mathbf{M}\|_1 \leq k\}$

$\|\mathbf{A}\|_{\mathcal{X}_k} := \sup_{\mathbf{M} \in \mathcal{X}_k} |\langle \mathbf{M}, \mathbf{A} \rangle|$

- Algorithm: Filtering (with SDP relaxation)
- **SDP-Stability**: For all large subsets $S'$ of $S$:
  - (Mean) $\sup_{v:\text{sparse}} \langle v, \mu_{S'} - \mu \rangle$ is small
  - (Covariance) $\|\mathbf{\Sigma}_{S'} - \mathbf{I}\|_{\mathcal{X}_k}$ is small
- Goal: $k^2$ samples

# Proof sketch: stability with a small number of samples

$\mathcal{X}_k := \{\mathbf{M} \succeq 0 : \mathrm{tr}(\mathbf{M}) = 1, \|\mathbf{M}\|_1 \le k\}$

$\|\mathbf{A}\|_{\mathcal{X}_k} := \sup_{\mathbf{M} \in \mathcal{X}_k} |\langle \mathbf{M}, \mathbf{A} \rangle|$

- Algorithm: Filtering (with SDP relaxation)
- SDP-Stability: For all large subsets $S'$ of $S$:
  - (Mean) $\sup_{v:\text{sparse}} \langle v, \mu_{S'} - \mu \rangle$ is small
  - (Covariance) $\|\mathbf{\Sigma}_{S'} - \mathbf{I}\|_{\mathcal{X}_k}$ is small
- Goal: $k^2$ samples

**Proof sketch** (of a weaker bound)

# Proof sketch: stability with a small number of samples

$\mathcal{X}_k := \{\mathbf{M} \succeq 0 : \mathrm{tr}(\mathbf{M}) = 1, \|\mathbf{M}\|_1 \leq k\}$

$\|\mathbf{A}\|_{\mathcal{X}_k} := \sup_{\mathbf{M} \in \mathcal{X}_k} |\langle \mathbf{M}, \mathbf{A} \rangle|$

- Algorithm: Filtering (with SDP relaxation)
- SDP-Stability: For all large subsets $S'$ of $S$:
  - (Mean) $\sup_{v:\text{sparse}} \langle v, \mu_{S'} - \mu \rangle$ is small
  - (Covariance) $\|\mathbf{\Sigma}_{S'} - \mathbf{I}\|_{\mathcal{X}_k}$ is small
- Goal: $k^2$ samples

**Proof sketch** (of a weaker bound)

$$\max_{S' \subset S:\text{large}} \|\mathbf{\Sigma}_{S'}\|_{\mathcal{X}_k}$$

# Proof sketch: stability with a small number of samples

$\mathcal{X}_k := \{\mathbf{M} \succeq 0 : \mathrm{tr}(\mathbf{M}) = 1, \|\mathbf{M}\|_1 \leq k\}$

$\|\mathbf{A}\|_{\mathcal{X}_k} := \sup_{\mathbf{M} \in \mathcal{X}_k} |\langle \mathbf{M}, \mathbf{A} \rangle|$

- Algorithm: Filtering (with SDP relaxation)
- SDP-Stability: For all large subsets $S'$ of $S$:
  - (Mean) $\sup_{v:\text{sparse}} \langle v, \mu_{S'} - \mu \rangle$ is small
  - (Covariance) $\|\mathbf{\Sigma}_{S'} - \mathbf{I}\|_{\mathcal{X}_k}$ is small
- Goal: $k^2$ samples

**Proof sketch** (of a weaker bound)

$$\max_{S' \subset S:\text{large}} \|\mathbf{\Sigma}_{S'}\|_{\mathcal{X}_k} \lesssim \|\mathbf{\Sigma}_S\|_{\mathcal{X}_k}$$

$\mathbf{M}$ PSD and $0 \preceq \mathbf{\Sigma}_{S'} \preceq 2\mathbf{\Sigma}_S$

# Proof sketch: stability with a small number of samples

$\mathcal{X}_k := \{\mathbf{M} \succeq 0 : \mathrm{tr}(\mathbf{M}) = 1, \|\mathbf{M}\|_1 \leq k\}$

$\|\mathbf{A}\|_{\mathcal{X}_k} := \sup_{\mathbf{M} \in \mathcal{X}_k} |\langle \mathbf{M}, \mathbf{A} \rangle|$

- Algorithm: Filtering (with SDP relaxation)
- SDP-Stability: For all large subsets $S'$ of $S$:
  - (Mean) $\sup_{v:\text{sparse}} \langle v, \mu_{S'} - \mu \rangle$ is small
  - (Covariance) $\|\mathbf{\Sigma}_{S'} - \mathbf{I}\|_{\mathcal{X}_k}$ is small
- Goal: $k^2$ samples

**Proof sketch** (of a weaker bound)

$$\max_{S' \subset S:\text{large}} \|\mathbf{\Sigma}_{S'}\|_{\mathcal{X}_k} \lesssim \|\mathbf{\Sigma}_S\|_{\mathcal{X}_k} \leq 1 + \|\mathbf{\Sigma}_S - \mathbf{I}\|_{\mathcal{X}_k}$$

triangle inequality

# Proof sketch: stability with a small number of samples

$\mathcal{X}_k := \{\mathbf{M} \succeq 0 : \mathrm{tr}(\mathbf{M}) = 1, \|\mathbf{M}\|_1 \le k\}$

$\|\mathbf{A}\|_{\mathcal{X}_k} := \sup_{\mathbf{M} \in \mathcal{X}_k} |\langle \mathbf{M}, \mathbf{A} \rangle|$

- Algorithm: Filtering (with SDP relaxation)
- SDP-Stability: For all large subsets $S'$ of $S$:
  - (Mean) $\sup_{v:\text{sparse}} \langle v, \mu_{S'} - \mu \rangle$ is small
  - (Covariance) $\|\mathbf{\Sigma}_{S'} - \mathbf{I}\|_{\mathcal{X}_k}$ is small
- Goal: $k^2$ samples

**Proof sketch** (of a weaker bound)

$$\max_{S' \subset S:\text{large}} \|\mathbf{\Sigma}_{S'}\|_{\mathcal{X}_k} \lesssim \|\mathbf{\Sigma}_S\|_{\mathcal{X}_k} \le 1 + \|\mathbf{\Sigma}_S - \mathbf{I}\|_{\mathcal{X}_k}$$
$$\le k \, \|\mathbf{\Sigma}_S - \mathbf{I}\|_\infty$$

Hölder's inequality

# Proof sketch: stability with a small number of samples

$\mathcal{X}_k := \{\mathbf{M} \succeq 0 : \mathrm{tr}(\mathbf{M}) = 1, \|\mathbf{M}\|_1 \leq k\}$

$\|\mathbf{A}\|_{\mathcal{X}_k} := \sup_{\mathbf{M} \in \mathcal{X}_k} |\langle \mathbf{M}, \mathbf{A} \rangle|$

- Algorithm: Filtering (with SDP relaxation)
- SDP-Stability: For all large subsets $S'$ of $S$:
  - (Mean) $\sup_{v:\text{sparse}} \langle v, \mu_{S'} - \mu \rangle$ is small
  - (Covariance) $\|\mathbf{\Sigma}_{S'} - \mathbf{I}\|_{\mathcal{X}_k}$ is small
- Goal: $k^2$ samples

**Proof sketch** (of a weaker bound)

$$\max_{S' \subset S:\text{large}} \|\mathbf{\Sigma}_{S'}\|_{\mathcal{X}_k} \lesssim \|\mathbf{\Sigma}_S\|_{\mathcal{X}_k} \leq 1 + \|\mathbf{\Sigma}_S - \mathbf{I}\|_{\mathcal{X}_k}$$

$$\leq k \|\mathbf{\Sigma}_S - \mathbf{I}\|_\infty$$

$$\leq k \frac{\widetilde{O}(1)}{\sqrt{n}} \quad \blacksquare$$

Hoeffding's inequality and union bound

# Overview

# I. Heavy-tailed distributions

To apply [BDLS17] to heavy-tailed distributions, we need to ask:

Do heavy-tailed inliers satisfy SDP stability with **good** sample complexity?

# I. Heavy-tailed distributions

To apply [BDLS17] to heavy-tailed distributions, we need to ask:

Do heavy-tailed inliers satisfy SDP stability with **good** sample complexity?

- No! Samples might be aligned with coordinate axes

# I. Heavy-tailed distributions

To apply [BDLS17] to heavy-tailed distributions, we need to ask:

Do heavy-tailed inliers satisfy SDP stability with **good** sample complexity?

- No! Samples might be aligned with coordinate axes
- Fix: clip samples **coordinatewise** (i.e., $\|x\|_\infty \leq \nu$ for $\nu = \mathrm{poly}(k)$)
  - clipping-induced **bias** versus **tails**

# I. Heavy-tailed distributions

To apply [BDLS17] to heavy-tailed distributions, we need to ask:

Do heavy-tailed inliers satisfy SDP stability with **good** sample complexity?

- No! Samples might be aligned with coordinate axes
- Fix: clip samples coordinatewise (i.e., $\|x\|_\infty \leq \nu$ for $\nu = \mathrm{poly}(k)$)
  - clipping-induced bias versus tails
- Previous proof: $S$ is stable w.p. $1-\delta$, if $n \approx k^2 \cdot \nu^4 \cdot \log(1/\delta)$

# I. Heavy-tailed distributions

To apply [BDLS17] to heavy-tailed distributions, we need to ask:

> Do heavy-tailed inliers satisfy SDP stability with **good** sample complexity?

- No! Samples might be aligned with coordinate axes
- Fix: clip samples coordinatewise (i.e., $\|x\|_\infty \leq \nu$ for $\nu = \mathrm{poly}(k)$)
  - clipping-induced bias versus tails
- Previous proof: $S$ is stable w.p. $1 - \delta$, if $n \approx k^2 \cdot \nu^4 \cdot \log(1/\delta)$
- Two **shortcomings** of this result:
  - Dependence on $k$: **superquadratic** ($\nu$) instead of quadratic
  - Dependence on $\delta$: **multiplicative** instead of additive

# I. Heavy-tailed distributions

To apply [BDLS17] to heavy-tailed distributions, we need to ask:

Do heavy-tailed inliers satisfy SDP stability with **good** sample complexity?

- No! Samples might be aligned with coordinate axes
- Fix: clip samples coordinatewise (i.e., $\|x\|_\infty \leq \nu$ for $\nu = \mathrm{poly}(k)$)
  - clipping-induced bias versus tails
- Previous proof: $S$ is stable w.p. $1 - \delta$, if $n \approx k^2 \cdot \nu^4 \cdot \log(1/\delta)$
- Two **shortcomings** of this result:
  - Dependence on $k$: **superquadratic** ($\nu$) instead of quadratic
  - Dependence on $\delta$: **multiplicative** instead of additive

**Can we close this gap?**

# Heavy-tailed distributions: improved sample complexity

**Theorem: [DKLP22]**

$\mathcal{P}$: $k$-sparse mean $\mu$, bounded covariance, and degree-four* moments.
An efficient algorithm to output $\widehat{\mu}$ from $\epsilon$-contaminated data: w.p. $1-\delta$,

[DKLP22] I. Diakonikolas, D. Kane, J. Lee, A. Pensia. Outlier-Robust Sparse Estimation for Heavy-Tailed. *NeurIPS*. 2022

# Heavy-tailed distributions: improved sample complexity

**Theorem: [DKLP22]**

$\mathcal{P}$: $k$-sparse mean $\mu$, bounded covariance, and degree-four* moments.
An efficient algorithm to output $\widehat{\mu}$ from $\epsilon$-contaminated data: w.p. $1-\delta$,

- (sample complexity) $n = O\left(\frac{\boldsymbol{k^2} \log d + \log(1/\delta)}{\epsilon}\right)$ samples
- (error) $\|\widehat{\mu} - \mu\|_2 = O(\sqrt{\epsilon})$

[DKLP22] I. Diakonikolas, D. Kane, J. Lee, A. Pensia. Outlier-Robust Sparse Estimation for Heavy-Tailed. *NeurIPS*. 2022

# Heavy-tailed distributions: improved sample complexity

**Theorem: [DKLP22]**

$\mathcal{P}$: $k$-sparse mean $\mu$, bounded covariance, and degree-four* moments.
An efficient algorithm to output $\widehat{\mu}$ from $\epsilon$-contaminated data: w.p. $1-\delta$,

- (sample complexity) $n = O\left(\frac{k^2 \log d + \log(1/\delta)}{\epsilon}\right)$ samples
- (error) $\|\widehat{\mu} - \mu\|_2 = O(\sqrt{\epsilon})$

- Near-optimal asymptotic error*, *computational* sample complexity*

[DKLP22] I. Diakonikolas, D. Kane, J. Lee, A. Pensia. Outlier-Robust Sparse Estimation for Heavy-Tailed. *NeurIPS*. 2022

# Heavy-tailed distributions: improved sample complexity

**Theorem: [DKLP22]**

$\mathcal{P}$: $k$-sparse mean $\mu$, bounded covariance, and degree-four* moments.
An efficient algorithm to output $\widehat{\mu}$ from $\epsilon$-contaminated data: w.p. $1-\delta$,

- (sample complexity) $n = O\left(\frac{\boldsymbol{k^2} \log d + \log(1/\delta)}{\epsilon}\right)$ samples
- (error) $\|\widehat{\mu} - \mu\|_2 = O(\sqrt{\epsilon})$

- Near-optimal asymptotic error*, *computational* sample complexity*
- Algorithm: same SDP as [BDLS17]; with improved probabilistic analysis

[DKLP22] I. Diakonikolas, D. Kane, J. Lee, A. Pensia. Outlier-Robust Sparse Estimation for Heavy-Tailed. *NeurIPS*. 2022

# Heavy-tailed distributions: improved sample complexity

**Theorem: [DKLP22]**

$\mathcal{P}$: $k$-sparse mean $\mu$, bounded covariance, and degree-four* moments.
An efficient algorithm to output $\widehat{\mu}$ from $\epsilon$-contaminated data: w.p. $1-\delta$,

- (sample complexity) $n = O\left(\frac{\boldsymbol{k^2} \log d + \log(1/\delta)}{\epsilon}\right)$ samples
- (error) $\|\widehat{\mu} - \mu\|_2 = O(\sqrt{\epsilon})$

- Near-optimal asymptotic error*, *computational* sample complexity*
- Algorithm: same SDP as [BDLS17]; with improved probabilistic analysis
- **Open questions**
  - removing bounded fourth-moment* condition
  - faster runtime

---

[DKLP22] I. Diakonikolas, D. Kane, J. Lee, A. Pensia. Outlier-Robust Sparse Estimation for Heavy-Tailed. *NeurIPS*. 2022

# Proof sketch of improved sample complexity

- Algorithm works even if inliers contains a **large stable subset**
- Do heavy-tailed (clipped) inliers contain a large stable subset?

# Proof sketch of improved sample complexity

- Algorithm works even if inliers contains a large stable subset
- Do heavy-tailed (clipped) inliers contain a large stable subset?

**Equivalent to the following question:**

# Proof sketch of improved sample complexity

- Algorithm works even if inliers contains a large stable subset
- Do heavy-tailed (clipped) inliers contain a large stable subset?

**Equivalent to the following question:**

Let $S$ be a set of $n$ i.i.d. samples from $P$ (heavy-tailed, bdd. coordinates)

$\mathcal{X}_k := \{\mathbf{M} \succeq 0 : \operatorname{tr}(\mathbf{M}) = 1, \|\mathbf{M}\|_1 \leq k\}$

# Proof sketch of improved sample complexity

- Algorithm works even if inliers contains a large stable subset
- Do heavy-tailed (clipped) inliers contain a large stable subset?

**Equivalent to the following question:**

Let $S$ be a set of $n$ i.i.d. samples from $P$ (heavy-tailed, bdd. coordinates)
Does the following hold

$$\text{w.h.p.,} \qquad \forall\, \mathbf{M} \in \mathcal{X}_k : \mathbb{P}_{x \sim \mathbf{S}}\left(x^\top \mathbf{M} x \gg 1\right) \leq 0.1$$

$\mathcal{X}_k := \{\mathbf{M} \succeq 0 : \mathrm{tr}(\mathbf{M}) = 1, \|\mathbf{M}\|_1 \leq k\}$

# Proof sketch of improved sample complexity

- Algorithm works even if inliers contains a large stable subset
- Do heavy-tailed (clipped) inliers contain a large stable subset?

**Equivalent to the following question:**

Let $S$ be a set of $n$ i.i.d. samples from $P$ (heavy-tailed, bdd. coordinates)
Does the following hold

w.h.p., $$\forall\, \mathbf{M} \in \mathcal{X}_k : \mathbb{P}_{x \sim \mathbf{S}}\left(x^\top \mathbf{M} x \gg 1\right) \leq 0.1$$

holds at the population ($n \to \infty$) by Markov

$\mathcal{X}_k := \{\mathbf{M} \succeq 0 : \mathrm{tr}(\mathbf{M}) = 1, \|\mathbf{M}\|_1 \leq k\}$

# Proof sketch of improved sample complexity

- Algorithm works even if inliers contains a large stable subset
- Do heavy-tailed (clipped) inliers contain a large stable subset?

**Equivalent to the following question:**

Let $S$ be a set of $n$ i.i.d. samples from $P$ (heavy-tailed, bdd. coordinates)
Does the following hold **with $n \approx k^2$?**

$$\text{w.h.p.,} \qquad \forall\, \mathbf{M} \in \mathcal{X}_k : \mathbb{P}_{x \sim S}\left(x^\top \mathbf{M} x \gg 1\right) \leq 0.1$$

$\mathcal{X}_k := \{\mathbf{M} \succeq 0 : \mathrm{tr}(\mathbf{M}) = 1, \|\mathbf{M}\|_1 \leq k\}$

# Rounding analytically sparse PSD matrices to sparse matrices

Let $S$ be a set of $n$ i.i.d. samples from $P$ (heavy-tailed, bdd. coordinates)
Does the following hold with $n \approx k^2$?

w.h.p., $$\forall\ \mathbf{M} \in \mathcal{X}_k : \mathbb{P}_{x \sim S}\left(x^\top \mathbf{M} x \gg 1\right) \leq 0.1$$

$\mathcal{X}_k := \{\mathbf{M} \succeq 0 : \mathrm{tr}(\mathbf{M}) = 1, \|\mathbf{M}\|_1 \leq k\}$

# Rounding analytically sparse PSD matrices to sparse matrices

Let $S$ be a set of $n$ i.i.d. samples from $P$ (heavy-tailed, bdd. coordinates)
Does the following hold with $n \approx k^2$?

w.h.p., $$\forall\ \mathbf{M} \in \mathcal{X}_k : \mathbb{P}_{x \sim S}\left(x^\top \mathbf{M} x \gg 1\right) \leq 0.1$$

- **Challenge**: VC dimension of $\mathcal{X}_k \gg k^2$

$\mathcal{X}_k := \{\mathbf{M} \succeq 0 : \mathrm{tr}(\mathbf{M}) = 1, \|\mathbf{M}\|_1 \le k\}$

# Rounding analytically sparse PSD matrices to sparse matrices

Let $S$ be a set of $n$ i.i.d. samples from $P$ (heavy-tailed, bdd. coordinates)
Does the following hold with $n \approx k^2$?

$$\text{w.h.p.,} \qquad \forall\, \mathbf{M} \in \mathcal{X}_k : \mathbb{P}_{x\sim S}\left(x^\top \mathbf{M} x \gg 1\right) \le 0.1$$

- **Challenge**: VC dimension of $\mathcal{X}_k \gg k^2$
- Idea [Li18]: relate it to $\mathcal{A}_k := \{\mathbf{B} : \|\mathbf{B}\|_{\mathrm{Fr}} = 1, \|\mathbf{B}\|_0 \le k^2\}$

[Li18] J. Li. Principled Approaches to Robust Machine Learning and Beyond. PhD thesis. 2018

$\mathcal{X}_k := \{\mathbf{M} \succeq 0 : \mathrm{tr}(\mathbf{M}) = 1, \|\mathbf{M}\|_1 \leq k\}$

# Rounding analytically sparse PSD matrices to sparse matrices

Let $S$ be a set of $n$ i.i.d. samples from $P$ (heavy-tailed, bdd. coordinates)
Does the following hold with $n \approx k^2$?

$$\text{w.h.p.,} \qquad \forall\, \mathbf{M} \in \mathcal{X}_k : \mathbb{P}_{x \sim S}\left(x^\top \mathbf{M} x \gg 1\right) \leq 0.1$$

- **Challenge**: VC dimension of $\mathcal{X}_k \gg k^2$
- Idea [Li18]: relate it to $\mathcal{A}_k := \{\mathbf{B} : \|\mathbf{B}\|_{\mathrm{Fr}} = 1, \|\mathbf{B}\|_0 \leq k^2\}$
  - **(Good)** VC dimension of $\mathcal{A}_k$ is $k^2$ and $\|\cdot\|_{\mathcal{X}_k} \lesssim \|\cdot\|_{\mathcal{A}_k}$

[Li18] J. Li. Principled Approaches to Robust Machine Learning and Beyond. PhD thesis. 2018

$\mathcal{X}_k := \{\mathbf{M} \succeq 0 : \mathrm{tr}(\mathbf{M}) = 1, \|\mathbf{M}\|_1 \le k\}$

# Rounding analytically sparse PSD matrices to sparse matrices

Let $S$ be a set of $n$ i.i.d. samples from $P$ (heavy-tailed, bdd. coordinates)
Does the following hold with $n \approx k^2$?

w.h.p., $\forall\, \mathbf{M} \in \mathcal{X}_k : \mathbb{P}_{x\sim S}\left(x^\top \mathbf{M} x \gg 1\right) \le 0.1$

- **Challenge**: VC dimension of $\mathcal{X}_k \gg k^2$
- Idea [Li18]: relate it to $\mathcal{A}_k := \{\mathbf{B} : \|\mathbf{B}\|_{\mathrm{Fr}} = 1, \|\mathbf{B}\|_0 \le k^2\}$
  - **(Good)** VC dimension of $\mathcal{A}_k$ is $k^2$ and $\|\cdot\|_{\mathcal{X}_k} \lesssim \|\cdot\|_{\mathcal{A}_k}$
  - But $x^\top \mathbf{B} x$ might have large variance (dependent coordinates)

[Li18] J. Li. Principled Approaches to Robust Machine Learning and Beyond. PhD thesis. 2018

$\mathcal{X}_k := \{\mathbf{M} \succeq 0 : \mathrm{tr}(\mathbf{M}) = 1, \|\mathbf{M}\|_1 \leq k\}$

# Rounding analytically sparse PSD matrices to sparse matrices

Let $S$ be a set of $n$ i.i.d. samples from $P$ (heavy-tailed, bdd. coordinates)
Does the following hold with $n \approx k^2$?

$$\text{w.h.p.,} \qquad \forall\, \mathbf{M} \in \mathcal{X}_k : \mathbb{P}_{x\sim S}\left(x^\top \mathbf{M} x \gg 1\right) \leq 0.1$$

- **Challenge**: VC dimension of $\mathcal{X}_k \gg k^2$
- Idea [Li18]: relate it to $\mathcal{A}_k := \{\mathbf{B} : \|\mathbf{B}\|_{\mathrm{Fr}} = 1, \|\mathbf{B}\|_0 \leq k^2\}$
- Fix [DKLP22]: $\mathcal{A}_{k,P} := \{\mathbf{B} \in \mathcal{A}_k : \mathbb{P}_{x\sim P}(x^\top \mathbf{B} x \gg 1) \leq 0.1\}$

[Li18] J. Li. Principled Approaches to Robust Machine Learning and Beyond. PhD thesis. 2018

$\mathcal{X}_k := \{\mathbf{M} \succeq 0 : \mathrm{tr}(\mathbf{M}) = 1, \|\mathbf{M}\|_1 \leq k\}$

# Rounding analytically sparse PSD matrices to sparse matrices

Let $S$ be a set of $n$ i.i.d. samples from $P$ (heavy-tailed, bdd. coordinates)
Does the following hold with $n \approx k^2$?

$$\text{w.h.p.,} \qquad \forall\, \mathbf{M} \in \mathcal{X}_k : \mathbb{P}_{x\sim S}\left(x^\top \mathbf{M} x \gg 1\right) \leq 0.1$$

- **Challenge**: VC dimension of $\mathcal{X}_k \gg k^2$
- Idea [Li18]: relate it to $\mathcal{A}_k := \{\mathbf{B} : \|\mathbf{B}\|_{\mathrm{Fr}} = 1, \|\mathbf{B}\|_0 \leq k^2\}$
- Fix [DKLP22]: $\mathcal{A}_{k,P} := \{\mathbf{B} \in \mathcal{A}_k : \mathbb{P}_{x\sim P}(x^\top \mathbf{B} x \gg 1) \leq 0.1\}$

**Theorem:** Sparse rounding (worst-case) [DKLP22]

Given $\mathbf{M} \in \mathcal{X}_k$, there is a random matrix $\mathbf{Q}$

- w.h.p., $\mathbf{Q} \in \mathcal{A}_{k,P}$
- $x^\top \mathbf{M} x \gg 1$ for clipped $x$ implies $\mathbb{P}_{\mathbf{Q}}(x^\top \mathbf{Q} x \gg 1) \geq 0.4$

[Li18] J. Li. Principled Approaches to Robust Machine Learning and Beyond. PhD thesis. 2018

# II. Adapting to unknown covariance

Suppose the distribution has bounded $t$-th moments; $t \gg 1$

- Optimal asymptotic error: $O(\epsilon^{1-\frac{1}{t}})$
- However, for **unknown** covariance, [BDLS17] gets stuck at $\Omega(\sqrt{\epsilon})$

[BDLS17] S. Balakrishnan, S. S. Du, J. Li, A. Singh. Computationally Efficient Robust Sparse Estimation.. *COLT*. 2017

# II. Adapting to unknown covariance

Suppose the distribution has bounded $t$-th moments; $t \gg 1$

- Optimal asymptotic error: $O(\epsilon^{1-\frac{1}{t}})$
- However, for **unknown** covariance, [BDLS17] gets stuck at $\Omega(\sqrt{\epsilon})$

**Theorem: [DKKPP22]**

Given $\epsilon$-contaminated samples from a distribution $P$ on $\mathbb{R}^d$ with $k$-sparse mean $\mu$ and bounded $t$-th moments:

[BDLS17] S. Balakrishnan, S. S. Du, J. Li, A. Singh. Computationally Efficient Robust Sparse Estimation.. *COLT*. 2017
[DKKPP22] I. Diakonikolas, D. Kane, S. Karmalkar, A. Pensia, T. Pittas. Robust Sparse Estimation via SoS. *COLT*. 2022

# II. Adapting to unknown covariance

Suppose the distribution has bounded $t$-th moments; $t \gg 1$

- Optimal asymptotic error: $O(\epsilon^{1-\frac{1}{t}})$
- However, for **unknown** covariance, [BDLS17] gets stuck at $\Omega(\sqrt{\epsilon})$

**Theorem: [DKKPP22]**

Given $\epsilon$-contaminated samples from a distribution $P$ on $\mathbb{R}^d$ with $k$-sparse mean $\mu$ and bounded $t$-th moments:

- (**lower bound**) Efficient* algorithms need $\boldsymbol{n \gg k^{\Omega(t)}}$ samples for $O(\epsilon^{1-\frac{1}{t}})$ error

[BDLS17] S. Balakrishnan, S. S. Du, J. Li, A. Singh. Computationally Efficient Robust Sparse Estimation.. *COLT*. 2017
[DKKPP22] I. Diakonikolas, D. Kane, S. Karmalkar, A. Pensia, T. Pittas. Robust Sparse Estimation via SoS. *COLT*. 2022

# II. Adapting to unknown covariance

Suppose the distribution has bounded $t$-th moments; $t \gg 1$

- Optimal asymptotic error: $O(\epsilon^{1-\frac{1}{t}})$
- However, for **unknown** covariance, [BDLS17] gets stuck at $\Omega(\sqrt{\epsilon})$

**Theorem: [DKKPP22]**

Given $\epsilon$-contaminated samples from a distribution $P$ on $\mathbb{R}^d$ with $k$-sparse mean $\mu$ and bounded $t$-th moments:

- (**lower bound**) Efficient* algorithms need $n \gg k^{\Omega(t)}$ samples for $O(\epsilon^{1-\frac{1}{t}})$ error
- (**upper bound**) A polynomial-time algorithm using $n = k^{O(t)}/\epsilon^2$ samples with matching error

[BDLS17] S. Balakrishnan, S. S. Du, J. Li, A. Singh. Computationally Efficient Robust Sparse Estimation.. *COLT*. 2017
[DKKPP22] I. Diakonikolas, D. Kane, S. Karmalkar, A. Pensia, T. Pittas. Robust Sparse Estimation via SoS. *COLT*. 2022

# II. Adapting to unknown covariance

Suppose the distribution has bounded $t$-th moments; $t \gg 1$

- Optimal asymptotic error: $O(\epsilon^{1-\frac{1}{t}})$
- However, for **unknown** covariance, [BDLS17] gets stuck at $\Omega(\sqrt{\epsilon})$

**Theorem: [DKKPP22]**

Given $\epsilon$-contaminated samples from a distribution $P$ on $\mathbb{R}^d$ with $k$-sparse mean $\mu$ and bounded $t$-th moments:

- (**lower bound**) Efficient* algorithms need $n \gg k^{\Omega(t)}$ samples for $O(\epsilon^{1-\frac{1}{t}})$ error
- (**upper bound**) A polynomial-time algorithm using $n = k^{O(t)}/\epsilon^2$ samples with matching error *if moments are certifiably* bounded*

[BDLS17] S. Balakrishnan, S. S. Du, J. Li, A. Singh. Computationally Efficient Robust Sparse Estimation.. *COLT*. 2017
[DKKPP22] I. Diakonikolas, D. Kane, S. Karmalkar, A. Pensia, T. Pittas. Robust Sparse Estimation via SoS. *COLT*. 2022

# Overview

- Background
  - Algorithmic framework
- Polynomial-time algorithms
  - Some improvements
- **Quadratic-time algorithms**
- Subquadratic-time algorithms

# Towards practical algorithms

$\|\mathbf{A}\|_{\mathrm{op},k} := \sup_{v:\text{sparse}} |v^\top \mathbf{A} v|$

- More practical certificates for the sparse operator norm?
- We want a practical function $f(\cdot)$:
  - $\|\mathbf{A}\|_{\mathrm{op},k} \le f(\mathbf{A})$
  - $f(\mathbf{\Sigma} - \mathbf{I})$ is bounded for clean data **and all large subsets** (stability)

$\|\mathbf{A}\|_{\mathrm{op},k} := \sup_{v:\text{sparse}} |v^\top \mathbf{A} v|$

# Towards practical algorithms

- More practical certificates for the sparse operator norm?
- We want a practical function $f(\cdot)$:
  - $\|\mathbf{A}\|_{\mathrm{op},k} \leq f(\mathbf{A})$
  - $f(\mathbf{\Sigma} - \mathbf{I})$ is bounded for clean data **and all large subsets** (stability)

$\|\mathbf{A}\|_{\text{op},k} := \sup_{v:\text{sparse}} |v^\top \mathbf{A} v|$

# Towards practical algorithms

- More practical certificates for the sparse operator norm?
- We want a practical function $f(\cdot)$:
  - $\|\mathbf{A}\|_{\text{op},k} \leq f(\mathbf{A})$
  - $f(\mathbf{\Sigma} - \mathbf{I})$ is bounded for clean data **and all large subsets** (stability)
- Suppose $f(\mathbf{A}) = \sup_{\mathbf{B} \in \mathcal{B}} \langle \mathbf{B}, \mathbf{A} \rangle$.
- Desirable properties of $\mathcal{B}$:
  - sparsity-aware
  - practical to search for $\mathbf{B}^*$

$\|\mathbf{A}\|_{\mathrm{op},k} := \sup_{v:\text{sparse}} |v^\top \mathbf{A} v|$

# Towards practical algorithms

- More practical certificates for the sparse operator norm?
- We want a practical function $f(\cdot)$:
  - $\|\mathbf{A}\|_{\mathrm{op},k} \le f(\mathbf{A})$
  - $f(\mathbf{\Sigma} - \mathbf{I})$ is bounded for clean data **and all large subsets** (stability)
- Suppose $f(\mathbf{A}) = \sup_{\mathbf{B} \in \mathcal{B}} \langle \mathbf{B}, \mathbf{A} \rangle$.
- Desirable properties of $\mathcal{B}$:
  - sparsity-aware
  - practical to search for $\mathbf{B}^*$
  - (For stability) For all $\mathbf{B}$ in $\mathcal{B}$, $x^\top \mathbf{B} x$ has **bdd. variance**

# Towards practical algorithms

$\|\mathbf{A}\|_{\mathrm{op},k} := \sup_{v:\text{sparse}} |v^\top \mathbf{A} v|$

- More practical certificates for the sparse operator norm?
- We want a practical function $f(\cdot)$:
  - $\|\mathbf{A}\|_{\mathrm{op},k} \leq f(\mathbf{A})$
  - $f(\boldsymbol{\Sigma} - \mathbf{I})$ is bounded for clean data **and all large subsets** (stability)
- Suppose $f(\mathbf{A}) = \sup_{\mathbf{B} \in \mathcal{B}} \langle \mathbf{B}, \mathbf{A} \rangle$.
- Desirable properties of $\mathcal{B}$:
  - sparsity-aware
  - practical to search for $\mathbf{B}^*$
  - (For stability) For all $\mathbf{B}$ in $\mathcal{B}$, $x^\top \mathbf{B} x$ has **bdd. variance**
- **[DKKPS19]**: $\mathcal{B} := \{\mathbf{B} : \|\mathbf{B}\|_{\mathrm{Fr}} = 1, \|\mathbf{B}\|_0 \leq k^2\}$

---

[DKKPS19] I. Diakonikolas, D. Kane, S. Karmalkar, E. Price, A. Stewart. Outlier-Robust Sparse Estimation... *NeurIPS*. 2019

$$\|\mathbf{A}\|_{\text{op},k} := \sup_{v:\text{sparse}} |v^\top \mathbf{A} v|$$

# Towards practical algorithms

- More practical certificates for the sparse operator norm?
- We want a practical function $f(\cdot)$:
  - $\|\mathbf{A}\|_{\text{op},k} \leq f(\mathbf{A})$
  - $f(\mathbf{\Sigma} - \mathbf{I})$ is bounded for clean data **and all large subsets** (stability)
- Suppose $f(\mathbf{A}) = \sup_{\mathbf{B} \in \mathcal{B}} \langle \mathbf{B}, \mathbf{A} \rangle$.
- Desirable properties of $\mathcal{B}$:
  - sparsity-aware
  - practical to search for $\mathbf{B}^*$
  - (For stability) For all $\mathbf{B}$ in $\mathcal{B}$, $x^\top \mathbf{B} x$ has **bdd. variance**
- **[DKKPS19]**: $\mathcal{B} := \{\mathbf{B} : \|\mathbf{B}\|_{\text{Fr}} = 1, \|\mathbf{B}\|_0 \leq k^2\}$
  - $f(\mathbf{A})$ is a "sparse Frobenius norm": $\ell_2$ norm of the largest $k^2$ entries

---

[DKKPS19] I. Diakonikolas, D. Kane, S. Karmalkar, E. Price, A. Stewart. Outlier-Robust Sparse Estimation... *NeurIPS*. 2019

$\|\mathbf{A}\|_{\mathrm{op},k} := \sup_{v:\text{sparse}} |v^\top \mathbf{A} v|$

# Towards practical algorithms

- More practical certificates for the sparse operator norm?
- We want a practical function $f(\cdot)$:
  - $\|\mathbf{A}\|_{\mathrm{op},k} \leq f(\mathbf{A})$
  - $f(\mathbf{\Sigma} - \mathbf{I})$ is bounded for clean data **and all large subsets** (stability)
- Suppose $f(\mathbf{A}) = \sup_{\mathbf{B} \in \mathcal{B}} \langle \mathbf{B}, \mathbf{A} \rangle$.
- Desirable properties of $\mathcal{B}$:
  - sparsity-aware
  - practical to search for $\mathbf{B}^*$
  - (For stability) For all $\mathbf{B}$ in $\mathcal{B}$, $x^\top \mathbf{B} x$ has **bdd. variance**
- **[DKKPS19]**: $\mathcal{B} := \{\mathbf{B} : \|\mathbf{B}\|_{\mathrm{Fr}} = 1, \|\mathbf{B}\|_0 \leq k^2\}$
  - $f(\mathbf{A})$ is a "sparse Frobenius norm": $\ell_2$ norm of the largest $k^2$ entries (in $d^2$ time)

[DKKPS19] I. Diakonikolas, D. Kane, S. Karmalkar, E. Price, A. Stewart. Outlier-Robust Sparse Estimation... *NeurIPS*. 2019

$\|\mathbf{A}\|_{\mathrm{op},k} := \sup_{v:\text{sparse}} |v^\top \mathbf{A} v|$

# Towards practical algorithms

- More practical certificates for the sparse operator norm?
- We want a practical function $f(\cdot)$:
  - $\|\mathbf{A}\|_{\mathrm{op},k} \leq f(\mathbf{A})$
  - $f(\mathbf{\Sigma} - \mathbf{I})$ is bounded for clean data **and all large subsets** (stability)
- Suppose $f(\mathbf{A}) = \sup_{\mathbf{B} \in \mathcal{B}} \langle \mathbf{B}, \mathbf{A} \rangle$.
- Desirable properties of $\mathcal{B}$:
  - sparsity-aware ✓
  - practical to search for $\mathbf{B}^*$
  - (For stability) For all $\mathbf{B}$ in $\mathcal{B}$, $x^\top \mathbf{B} x$ has **bdd. variance**
- **[DKKPS19]**: $\mathcal{B} := \{\mathbf{B} : \|\mathbf{B}\|_{\mathrm{Fr}} = 1, \|\mathbf{B}\|_0 \leq k^2\}$
  - $f(\mathbf{A})$ is a "sparse Frobenius norm": $\ell_2$ norm of the largest $k^2$ entries

[DKKPS19] I. Diakonikolas, D. Kane, S. Karmalkar, E. Price, A. Stewart. Outlier-Robust Sparse Estimation... *NeurIPS*. 2019

$\|\mathbf{A}\|_{\mathrm{op},k} := \sup_{v:\text{sparse}} |v^\top \mathbf{A} v|$

# Towards practical algorithms

- More practical certificates for the sparse operator norm?
- We want a practical function $f(\cdot)$:
  - $\|\mathbf{A}\|_{\mathrm{op},k} \le f(\mathbf{A})$
  - $f(\mathbf{\Sigma} - \mathbf{I})$ is bounded for clean data **and all large subsets** (stability)
- Suppose $f(\mathbf{A}) = \sup_{\mathbf{B} \in \mathcal{B}} \langle \mathbf{B}, \mathbf{A} \rangle$.
- Desirable properties of $\mathcal{B}$:
  - sparsity-aware ✓
  - practical to search for $\mathbf{B}^*$ ✓
  - (For stability) For all $\mathbf{B}$ in $\mathcal{B}$, $x^\top \mathbf{B} x$ has **bdd. variance**
- **[DKKPS19]**: $\mathcal{B} := \{\mathbf{B} : \|\mathbf{B}\|_{\mathrm{Fr}} = 1, \|\mathbf{B}\|_0 \le k^2\}$
  - $f(\mathbf{A})$ is a "sparse Frobenius norm": $\ell_2$ norm of the largest $k^2$ entries

---

[DKKPS19] I. Diakonikolas, D. Kane, S. Karmalkar, E. Price, A. Stewart. Outlier-Robust Sparse Estimation... *NeurIPS*. 2019

$\|\mathbf{A}\|_{\mathrm{op},k} := \sup_{v:\text{sparse}} |v^\top \mathbf{A} v|$

# Towards practical algorithms

- More practical certificates for the sparse operator norm?
- We want a practical function $f(\cdot)$:
  - $\|\mathbf{A}\|_{\mathrm{op},k} \leq f(\mathbf{A})$
  - $f(\mathbf{\Sigma} - \mathbf{I})$ is bounded for clean data **and all large subsets** (stability)
- Suppose $f(\mathbf{A}) = \sup_{\mathbf{B} \in \mathcal{B}} \langle \mathbf{B}, \mathbf{A} \rangle$.
- Desirable properties of $\mathcal{B}$:
  - sparsity-aware ✓
  - practical to search for $\mathbf{B}^*$ ✓
  - (For stability) For all $\mathbf{B}$ in $\mathcal{B}$, $x^\top \mathbf{B} x$ has **bdd. variance** ✓ **for gaussians**
- **[DKKPS19]**: $\mathcal{B} := \{\mathbf{B} : \|\mathbf{B}\|_{\mathrm{Fr}} = 1, \|\mathbf{B}\|_0 \leq k^2\}$
  - $f(\mathbf{A})$ is a "sparse Frobenius norm": $\ell_2$ norm of the largest $k^2$ entries

---

[DKKPS19] I. Diakonikolas, D. Kane, S. Karmalkar, E. Price, A. Stewart. Outlier-Robust Sparse Estimation... *NeurIPS*. 2019

# A practical algorithm using sparse Frobenius norm

**Theorem: [DKKPS19]**

Given $n$ $\epsilon$-contaminated samples from $\mathcal{N}(\mu, I)$ with $k$-sparse mean $\mu$, a **practical** algorithm to compute $\widehat{\mu}$ such that w.h.p.,

[DKKPS19] I. Diakonikolas, D. Kane, S. Karmalkar, E. Price, A. Stewart. Outlier-Robust Sparse Estimation... *NeurIPS*. 2019

# A practical algorithm using sparse Frobenius norm

**Theorem: [DKKPS19]**

Given $n$ $\epsilon$-contaminated samples from $\mathcal{N}(\mu, I)$ with $k$-sparse mean $\mu$, a **practical** algorithm to compute $\widehat{\mu}$ such that w.h.p.,

- (sample complexity) $n = \widetilde{O}\left(\frac{k^2}{\epsilon^2}\right)$ samples
- (error) $\|\widehat{\mu} - \mu\|_2 = \widetilde{O}(\epsilon)$
- (**runtime**) $d^2 \cdot \mathrm{poly}(k, 1/\epsilon)$

- Near-optimal asymptotic error, computational sample complexity

[DKKPS19] I. Diakonikolas, D. Kane, S. Karmalkar, E. Price, A. Stewart. Outlier-Robust Sparse Estimation... *NeurIPS*. 2019

# A practical algorithm using sparse Frobenius norm

**Theorem: [DKKPS19]**

Given $n$ $\epsilon$-contaminated samples from $\mathcal{N}(\mu, I)$ with $k$-sparse mean $\mu$, a **practical** algorithm to compute $\widehat{\mu}$ such that w.h.p.,

- (sample complexity) $n = \widetilde{O}\left(\frac{k^2}{\epsilon^2}\right)$ samples
- (error) $\|\widehat{\mu} - \mu\|_2 = \widetilde{O}(\epsilon)$
- (**runtime**) $d^2 \cdot \mathrm{poly}(k, 1/\epsilon)$

- Near-optimal asymptotic error, computational sample complexity
- **Open questions**:
  - Beyond Gaussians? Even, all (isotropic) subgaussian distributions?

[DKKPS19] I. Diakonikolas, D. Kane, S. Karmalkar, E. Price, A. Stewart. Outlier-Robust Sparse Estimation... *NeurIPS*. 2019

# A practical algorithm using sparse Frobenius norm

**Theorem: [DKKPS19]**

Given $n$ $\epsilon$-contaminated samples from $\mathcal{N}(\mu, I)$ with $k$-sparse mean $\mu$, a **practical** algorithm to compute $\widehat{\mu}$ such that w.h.p.,

- (sample complexity) $n = \widetilde{O}\left(\frac{k^2}{\epsilon^2}\right)$ samples
- (error) $\|\widehat{\mu} - \mu\|_2 = \widetilde{O}(\epsilon)$
- (**runtime**) $d^2 \cdot \mathrm{poly}(k, 1/\epsilon)$

- Near-optimal asymptotic error, computational sample complexity
- **Open questions**:
  - Beyond Gaussians? Even, all (isotropic) subgaussian distributions?
  - Beyond isotropy? Say, unknown covariance Gaussians

[DKKPS19] I. Diakonikolas, D. Kane, S. Karmalkar, E. Price, A. Stewart. Outlier-Robust Sparse Estimation... *NeurIPS*. 2019

# Overview

- Background
  - Algorithmic framework
- Polynomial-time algorithms
  - Some improvements
- Quadratic-time algorithms
- **Subquadratic-time algorithms**

# Quest for faster algorithms

- Input size: $nd$, where $n$ is the sample complexity
  - Ideal runtime $\widetilde{O}\left(nd\right)$

# Quest for faster algorithms

- Input size: $nd$, where $n$ is the sample complexity
  - Ideal runtime $\widetilde{O}(nd)$
  - Possible for *dense* estimation: [CDG19; DL22; DHL19; CMY20; DKKLT22; DKPP22]

[CDG19] Y. Cheng, I. Diakonikolas, R. Ge. High-Dimensional Robust Mean Estimation in Nearly-Linear Time. *SODA*. 2019
[DL22] J. Depersin, G. Lecué. Robust Subgaussian Estimation of a Mean Vector in Nearly Linear Time. *Ann. Stats.* 2022
[DHL19] Y. Dong, S. Hopkins, J. Li. Quantum entropy scoring for fast robust mean estimation.. *NeurIPS*. 2019
[CMY20] Y. Cherapanamjeri, S. Mohanty, M. Yau. List decodable mean estimation in nearly linear time. *FOCS*. 2020
[DKKLT22] I. Diakonikolas, D. M. Kane, D. Kongsgaard, J. Li, K. Tian. Clustering Mixture Models in ..Linear.. *STOC*. 2022
[DKPP22] I. Diakonikolas, D. Kane, A. Pensia, T. Pittas. Streaming Algorithms for .. Robust Statistics.. *ICML*. 2022

# Quest for faster algorithms

- Input size: $nd$, where $n$ is the sample complexity
  - Ideal runtime $\widetilde{O}(nd)$
  - Possible for *dense* estimation: [CDG19; DL22; DHL19; CMY20; DKKLT22; DKPP22]
- (Slightly) relaxed goal: $d \cdot \mathrm{poly}(n)$ time and $n = \mathrm{poly}(k/\epsilon)$

[CDG19] Y. Cheng, I. Diakonikolas, R. Ge. High-Dimensional Robust Mean Estimation in Nearly-Linear Time. *SODA*. 2019
[DL22] J. Depersin, G. Lecué. Robust Subgaussian Estimation of a Mean Vector in Nearly Linear Time. *Ann. Stats*. 2022
[DHL19] Y. Dong, S. Hopkins, J. Li. Quantum entropy scoring for fast robust mean estimation.. *NeurIPS*. 2019
[CMY20] Y. Cherapanamjeri, S. Mohanty, M. Yau. List decodable mean estimation in nearly linear time. *FOCS*. 2020
[DKKLT22] I. Diakonikolas, D. M. Kane, D. Kongsgaard, J. Li, K. Tian. Clustering Mixture Models in ..Linear.. *STOC*. 2022
[DKPP22] I. Diakonikolas, D. Kane, A. Pensia, T. Pittas. Streaming Algorithms for .. Robust Statistics.. *ICML*. 2022

# Quest for faster algorithms

- Input size: $nd$, where $n$ is the sample complexity
  - Ideal runtime $\widetilde{O}(nd)$
  - Possible for *dense* estimation: [CDG19; DL22; DHL19; CMY20; DKKLT22; DKPP22]
- (Slightly) relaxed goal: $d \cdot \mathrm{poly}(n)$ time and $n = \mathrm{poly}(k/\epsilon)$

A **linear-time** algorithm for robust sparse estimation?

[CDG19] Y. Cheng, I. Diakonikolas, R. Ge. High-Dimensional Robust Mean Estimation in Nearly-Linear Time. *SODA*. 2019
[DL22] J. Depersin, G. Lecué. Robust Subgaussian Estimation of a Mean Vector in Nearly Linear Time. *Ann. Stats*. 2022
[DHL19] Y. Dong, S. Hopkins, J. Li. Quantum entropy scoring for fast robust mean estimation.. *NeurIPS*. 2019
[CMY20] Y. Cherapanamjeri, S. Mohanty, M. Yau. List decodable mean estimation in nearly linear time. *FOCS*. 2020
[DKKLT22] I. Diakonikolas, D. M. Kane, D. Kongsgaard, J. Li, K. Tian. Clustering Mixture Models in ..Linear.. *STOC*. 2022
[DKPP22] I. Diakonikolas, D. Kane, A. Pensia, T. Pittas. Streaming Algorithms for .. Robust Statistics.. *ICML*. 2022

# Quest for faster algorithms

- Input size: $nd$, where $n$ is the sample complexity
  - Ideal runtime $\widetilde{O}(nd)$
  - Possible for *dense* estimation: [CDG19; DL22; DHL19; CMY20; DKKLT22; DKPP22]
- (Slightly) relaxed goal: $d \cdot \mathrm{poly}(n)$ time and $n = \mathrm{poly}(k/\epsilon)$
- **Challenges**:

A **linear-time** algorithm for robust sparse estimation?

[CDG19] Y. Cheng, I. Diakonikolas, R. Ge. High-Dimensional Robust Mean Estimation in Nearly-Linear Time. *SODA*. 2019
[DL22] J. Depersin, G. Lecué. Robust Subgaussian Estimation of a Mean Vector in Nearly Linear Time. *Ann. Stats*. 2022
[DHL19] Y. Dong, S. Hopkins, J. Li. Quantum entropy scoring for fast robust mean estimation.. *NeurIPS*. 2019
[CMY20] Y. Cherapanamjeri, S. Mohanty, M. Yau. List decodable mean estimation in nearly linear time. *FOCS*. 2020
[DKKLT22] I. Diakonikolas, D. M. Kane, D. Kongsgaard, J. Li, K. Tian. Clustering Mixture Models in ..Linear.. *STOC*. 2022
[DKPP22] I. Diakonikolas, D. Kane, A. Pensia, T. Pittas. Streaming Algorithms for .. Robust Statistics.. *ICML*. 2022

# Quest for faster algorithms

- Input size: $nd$, where $n$ is the sample complexity
  - Ideal runtime $\widetilde{O}(nd)$
  - Possible for *dense* estimation: [CDG19; DL22; DHL19; CMY20; DKKLT22; DKPP22]
- (Slightly) relaxed goal: $d \cdot \mathrm{poly}(n)$ time and $n = \mathrm{poly}(k/\epsilon)$
- **Challenges**:
  - Analog of power iteration for sparse eigenvectors?
  - In fact, existing approaches need **explicit** $\Sigma$

A **linear-time** algorithm for robust sparse estimation?

[CDG19] Y. Cheng, I. Diakonikolas, R. Ge. High-Dimensional Robust Mean Estimation in Nearly-Linear Time. *SODA*. 2019
[DL22] J. Depersin, G. Lecué. Robust Subgaussian Estimation of a Mean Vector in Nearly Linear Time. *Ann. Stats*. 2022
[DHL19] Y. Dong, S. Hopkins, J. Li. Quantum entropy scoring for fast robust mean estimation.. *NeurIPS*. 2019
[CMY20] Y. Cherapanamjeri, S. Mohanty, M. Yau. List decodable mean estimation in nearly linear time. *FOCS*. 2020
[DKKLT22] I. Diakonikolas, D. M. Kane, D. Kongsgaard, J. Li, K. Tian. Clustering Mixture Models in ..Linear.. *STOC*. 2022
[DKPP22] I. Diakonikolas, D. Kane, A. Pensia, T. Pittas. Streaming Algorithms for .. Robust Statistics.. *ICML*. 2022

# Quest for faster algorithms

- Input size: $nd$, where $n$ is the sample complexity
  - Ideal runtime $\widetilde{O}(nd)$
  - Possible for *dense* estimation: [CDG19; DL22; DHL19; CMY20; DKKLT22; DKPP22]
- (Slightly) relaxed goal: $d \cdot \mathrm{poly}(n)$ time and $n = \mathrm{poly}(k/\epsilon)$
- **Challenges**:
  - Analog of power iteration for sparse eigenvectors?
  - In fact, existing approaches need **explicit** $\Sigma$

A **subquadratic-time** algorithm for robust sparse estimation?


[CDG19] Y. Cheng, I. Diakonikolas, R. Ge. High-Dimensional Robust Mean Estimation in Nearly-Linear Time. *SODA*. 2019
[DL22] J. Depersin, G. Lecué. Robust Subgaussian Estimation of a Mean Vector in Nearly Linear Time. *Ann. Stats*. 2022
[DHL19] Y. Dong, S. Hopkins, J. Li. Quantum entropy scoring for fast robust mean estimation.. *NeurIPS*. 2019
[CMY20] Y. Cherapanamjeri, S. Mohanty, M. Yau. List decodable mean estimation in nearly linear time. *FOCS*. 2020
[DKKLT22] I. Diakonikolas, D. M. Kane, D. Kongsgaard, J. Li, K. Tian. Clustering Mixture Models in ..Linear.. *STOC*. 2022
[DKPP22] I. Diakonikolas, D. Kane, A. Pensia, T. Pittas. Streaming Algorithms for .. Robust Statistics.. *ICML*. 2022

# A subquadratic-time algorithm

**Theorem: [P24]**

Given $\epsilon$-contaminated samples from $\mathcal{N}(\mu, I)$ on $\mathbb{R}^d$ with $k$-sparse $\mu$

[Pen24] A. Pensia. A Sub-Quadratic Time Algorithm for Robust Sparse Mean Estimation. *ICML*. 2024

# A subquadratic-time algorithm

**Theorem: [P24]**

Given $\epsilon$-contaminated samples from $\mathcal{N}(\mu, I)$ on $\mathbb{R}^d$ with $k$-sparse $\mu$ and a natural number **q**,

[Pen24] A. Pensia. A Sub-Quadratic Time Algorithm for Robust Sparse Mean Estimation. *ICML*. 2024

# A subquadratic-time algorithm

**Theorem: [P24]**

Given $\epsilon$-contaminated samples from $\mathcal{N}(\mu, I)$ on $\mathbb{R}^d$ with $k$-sparse $\mu$ and a natural number **q**, there is an algorithm to compute $\widehat{\mu}$:

---

[Pen24] A. Pensia. A Sub-Quadratic Time Algorithm for Robust Sparse Mean Estimation. *ICML*. 2024

# A subquadratic-time algorithm

**Theorem: [P24]**

Given $\epsilon$-contaminated samples from $\mathcal{N}(\mu, I)$ on $\mathbb{R}^d$ with $k$-sparse $\mu$ and a natural number **q**, there is an algorithm to compute $\widehat{\mu}$:

- (error) $\|\widehat{\mu} - \mu\|_2 = \widetilde{O}(\epsilon)$

- Near-optimal asymptotic error

[Pen24] A. Pensia. A Sub-Quadratic Time Algorithm for Robust Sparse Mean Estimation. *ICML*. 2024

# A subquadratic-time algorithm

**Theorem: [P24]**

Given $\epsilon$-contaminated samples from $\mathcal{N}(\mu, I)$ on $\mathbb{R}^d$ with $k$-sparse $\mu$ and a natural number **q**, there is an algorithm to compute $\widehat{\mu}$:

- (error) $\|\widehat{\mu} - \mu\|_2 = \widetilde{O}(\epsilon)$
- (**runtime**) $d^{1.6+\frac{1}{\mathsf{q}}} \cdot \mathrm{poly}(n)$

- Near-optimal asymptotic error
- Subquadratic for any $q \geq 3$

[Pen24] A. Pensia. A Sub-Quadratic Time Algorithm for Robust Sparse Mean Estimation. *ICML*. 2024

# A subquadratic-time algorithm

**Theorem: [P24]**

Given $\epsilon$-contaminated samples from $\mathcal{N}(\mu, I)$ on $\mathbb{R}^d$ with $k$-sparse $\mu$ and a natural number **q**, there is an algorithm to compute $\widehat{\mu}$:

- (error) $\|\widehat{\mu} - \mu\|_2 = \widetilde{O}(\epsilon)$
- (**runtime**) $d^{1.6+\frac{1}{\mathbf{q}}} \cdot \mathrm{poly}(n)$
- (sample complexity) $n = \mathrm{poly}(k^{\mathbf{q}}, 1/\epsilon^{\mathbf{q}}, \log d)$ samples

- Near-optimal asymptotic error
- Subquadratic for any $q \geq 3$

---

[Pen24] A. Pensia. A Sub-Quadratic Time Algorithm for Robust Sparse Mean Estimation. *ICML*. 2024

# A subquadratic-time algorithm

**Theorem: [P24]**

Given $\epsilon$-contaminated samples from $\mathcal{N}(\mu, I)$ on $\mathbb{R}^d$ with $k$-sparse $\mu$ and a natural number **q**, there is an algorithm to compute $\widehat{\mu}$:

- (error) $\|\widehat{\mu} - \mu\|_2 = \widetilde{O}(\epsilon)$
- (**runtime**) $d^{1.6+\frac{1}{\mathsf{q}}} \cdot \mathrm{poly}(n)$
- (sample complexity) $n = \mathrm{poly}(k^{\mathsf{q}}, 1/\epsilon^{\mathsf{q}}, \log d)$ samples

- Near-optimal asymptotic error
- Subquadratic for any $q \geq 3$
- **Open questions**:
  - $k^2$ sample complexity
  - linear time
  - a wider family of distributions (same as [DKKPS19])

[Pen24] A. Pensia. A Sub-Quadratic Time Algorithm for Robust Sparse Mean Estimation. *ICML*. 2024

$\|\mathbf{A}\|_{\mathrm{Fr},k^2} := \ell_2$ norm of largest $k^2$ entries of $\mathbf{A}$

# Proof idea: Algorithm blueprint

**Algorithmic template** from [DKKPS19].

1. While $\|\mathbf{\Sigma} - \mathbf{I}\|_{\mathrm{Fr},k^2}$ large:
   1.1 Filter points and update $\mathbf{\Sigma}$
2. Return $\mathrm{HardThresh}($sample mean$)$

[DKKPS19] I. Diakonikolas, D. Kane, S. Karmalkar, E. Price, A. Stewart. Outlier-Robust Sparse Estimation... *NeurIPS*. 2019

$\|\mathbf{A}\|_{\mathrm{Fr},k^2} := \ell_2$ norm of largest $k^2$ entries of $\mathbf{A}$

# Proof idea: Algorithm blueprint

**Algorithmic template** from [DKKPS19].

1. While $\|\mathbf{\Sigma} - \mathbf{I}\|_{\mathrm{Fr},k^2}$ large: ← Takes $d^2$ time
   1.1 Filter points and update $\mathbf{\Sigma}$
2. Return $\mathrm{HardThresh}($sample mean$)$

[DKKPS19] I. Diakonikolas, D. Kane, S. Karmalkar, E. Price, A. Stewart. Outlier-Robust Sparse Estimation... *NeurIPS*. 2019

$\|\mathbf{A}\|_{\mathrm{Fr},k^2} := \ell_2$ norm of largest $k^2$ entries of $\mathbf{A}$

# Proof idea: Algorithm blueprint

**Algorithmic template** from [DKKPS19].

1. While $\|\mathbf{\Sigma} - \mathbf{I}\|_{\mathrm{Fr},k^2}$ large:
   1.1 Filter points and update $\mathbf{\Sigma}$
2. Return $\mathrm{HardThresh}($sample mean$)$

- Key challenge: off-diagonal correlated coordinates

[DKKPS19] I. Diakonikolas, D. Kane, S. Karmalkar, E. Price, A. Stewart. Outlier-Robust Sparse Estimation... *NeurIPS*. 2019

$\|\mathbf{A}\|_{\mathrm{Fr},k^2} := \ell_2$ norm of largest $k^2$ entries of $\mathbf{A}$

# Proof idea: Algorithm blueprint

**Algorithmic template** from [DKKPS19].

1. While $\|\mathbf{\Sigma} - \mathbf{I}\|_{\mathrm{Fr},k^2}$ large:
   1.1 Filter points and update $\mathbf{\Sigma}$
2. Return $\mathrm{HardThresh}($sample mean$)$

- Key challenge: off-diagonal correlated coordinates
- $H := \{(i,j) : i \neq j \ , \ |\mathbf{\Sigma}_{i,j}| \gg 1/k\}$

Strongly correlated coordinates

[DKKPS19] I. Diakonikolas, D. Kane, S. Karmalkar, E. Price, A. Stewart. Outlier-Robust Sparse Estimation... *NeurIPS*. 2019

$\|\mathbf{A}\|_{\mathrm{Fr},k^2} := \ell_2$ norm of largest $k^2$ entries of $\mathbf{A}$

# Proof idea: Algorithm blueprint

**Algorithmic template** from [DKKPS19].

1. While $\|\mathbf{\Sigma} - \mathbf{I}\|_{\mathrm{Fr},k^2}$ large:
   1.1 Filter points and update $\mathbf{\Sigma}$
2. Return $\mathrm{HardThresh}($sample mean$)$

- Key challenge: off-diagonal correlated coordinates
- $H := \{(i,j) : i \neq j \ , \ |\mathbf{\Sigma}_{i,j}| \gg 1/k\}$
- First observation: Coordinates in $H^{\complement}$ are nice
  - $\|(\mathbf{\Sigma} - \mathbf{I})_{H^{\complement}}\|_{\mathrm{Fr},k^2} \leq \sqrt{k^2} \cdot \frac{1}{k} = O(1)$

[DKKPS19] I. Diakonikolas, D. Kane, S. Karmalkar, E. Price, A. Stewart. Outlier-Robust Sparse Estimation... *NeurIPS*. 2019

$\|\mathbf{A}\|_{\mathrm{Fr},k^2} := \ell_2$ norm of largest $k^2$ entries of $\mathbf{A}$

# Proof idea: Algorithm blueprint

**Algorithmic template** from [DKKPS19].

1. While $\|\mathbf{\Sigma} - \mathbf{I}\|_{\mathrm{Fr},k^2}$ large:
   1.1 Filter points and update $\mathbf{\Sigma}$
2. Return $\mathrm{HardThresh}($sample mean$)$

- Key challenge: off-diagonal correlated coordinates
- $H := \{(i,j) : i \neq j \;,\; |\mathbf{\Sigma}_{i,j}| \gg 1/k\}$
- First observation: Coordinates in $H^\complement$ are nice
  - $\|(\mathbf{\Sigma} - \mathbf{I})_{H^\complement}\|_{\mathrm{Fr},k^2} \leq \sqrt{k^2} \cdot \frac{1}{k} = O(1)$

**How to find $H$ in subquadratic time?**

[DKKPS19] I. Diakonikolas, D. Kane, S. Karmalkar, E. Price, A. Stewart. Outlier-Robust Sparse Estimation... *NeurIPS*. 2019

$H := \{(i,j) : i \neq j \, , \, |\mathbf{\Sigma}_{i,j}| \gg \rho\}$

# Connections to correlation detection

Definition: Two vectors $x, y \in \mathbb{R}^n$ are $\rho$-correlated if $\left| \left\langle \frac{x}{\|x\|_2}, \frac{y}{\|y\|_2} \right\rangle \right| \geq \rho$

$H := \{(i,j) : i \neq j\ ,\ |\mathbf{\Sigma}_{i,j}| \gg \rho\}$

# Connections to correlation detection

Definition: Two vectors $x, y \in \mathbb{R}^n$ are $\rho$-correlated if $\left|\left\langle \frac{x}{\|x\|_2}, \frac{y}{\|y\|_2} \right\rangle\right| \geq \rho$

**Problem statement.** Correlation detection

Input:

- vectors $y_1, \ldots, y_d \in \mathbb{R}^n$; $n \ll d$
- a threshold $\rho \in (0, 1)$

Output: all $\rho$-correlated pairs $(i, j) \in [d] \times [d]$

$H := \{(i,j) : i \neq j\ ,\ |\mathbf{\Sigma}_{i,j}| \gg \rho\}$

# Connections to correlation detection

Definition: Two vectors $x, y \in \mathbb{R}^n$ are $\rho$-correlated if $\left|\left\langle \frac{x}{\|x\|_2}, \frac{y}{\|y\|_2} \right\rangle\right| \geq \rho$

**Problem statement.** Correlation detection

Input:

- vectors $y_1, \ldots, y_d \in \mathbb{R}^n$; $\quad n \ll d$
- a threshold $\rho \in (0,1)$

Output: all $\rho$-correlated pairs $(i,j) \in [d] \times [d]$

- **Naïve algorithm**: try all possible pairs, runs in $d^2$ time

$H := \{(i,j) : i \neq j\,,\, |\mathbf{\Sigma}_{i,j}| \gg \rho\}$

# Connections to correlation detection

Definition: Two vectors $x, y \in \mathbb{R}^n$ are $\rho$-correlated if $\left|\left\langle \frac{x}{\|x\|_2}, \frac{y}{\|y\|_2} \right\rangle\right| \geq \rho$

**Problem statement.** Correlation detection

Input:

- vectors $y_1, \ldots, y_d \in \mathbb{R}^n$; $\quad n \ll d$
- a threshold $\rho \in (0,1)$

Output: all $\rho$-correlated pairs $(i,j) \in [d] \times [d]$

- **Naïve algorithm**: try all possible pairs, runs in $d^2$ time
  - Likely to be optimal

$H := \{(i,j) : i \neq j\,,\, |\boldsymbol{\Sigma}_{i,j}| \gg \rho\}$

# Connections to correlation detection

Definition: Two vectors $x, y \in \mathbb{R}^n$ are $\rho$-correlated if $\left|\left\langle \frac{x}{\|x\|_2}, \frac{y}{\|y\|_2} \right\rangle\right| \geq \rho$

**Problem statement.** Correlation detection **with margin**

Input:
- vectors $y_1, \ldots, y_d \in \mathbb{R}^n$; $\quad n \ll d$
- a threshold $\rho \in (0,1)$

Output: all $\rho$-correlated pairs $(i,j) \in [d] \times [d]$

- **Naïve algorithm**: try all possible pairs, runs in $d^2$ time

$H := \{(i,j) : i \neq j\ ,\ |\mathbf{\Sigma}_{i,j}| \gg \rho\}$

# Connections to correlation detection

Definition: Two vectors $x, y \in \mathbb{R}^n$ are $\rho$-correlated if $\left|\left\langle \frac{x}{\|x\|_2}, \frac{y}{\|y\|_2} \right\rangle\right| \geq \rho$

**Problem statement.** Correlation detection **with margin**

Input:

- vectors $y_1, \ldots, y_d \in \mathbb{R}^n$; $\quad n \ll d$
- a threshold $\rho \in (0,1)$, a threshold $\tau \ll \rho$

Output: all $\rho$-correlated pairs $(i,j) \in [d] \times [d]$

- **Naïve algorithm**: try all possible pairs, runs in $d^2$ time

$H := \{(i,j) : i \neq j \,,\, |\boldsymbol{\Sigma}_{i,j}| \gg \rho\}$

# Connections to correlation detection

Definition: Two vectors $x, y \in \mathbb{R}^n$ are $\rho$-correlated if $\left|\left\langle \frac{x}{\|x\|_2}, \frac{y}{\|y\|_2} \right\rangle\right| \geq \rho$

**Problem statement.** Correlation detection **with margin**

Input:

- vectors $y_1, \ldots, y_d \in \mathbb{R}^n$; $\quad n \ll d$
- a threshold $\rho \in (0,1)$, a threshold $\tau \ll \rho$
- **very few**, say $o(d)$ out of $d^2$ pairs, are $\tau$-correlated

Output: all $\rho$-correlated pairs $(i,j) \in [d] \times [d]$

- **Naïve algorithm**: try all possible pairs, runs in $d^2$ time

$H := \{(i,j) : i \neq j \,,\, |\mathbf{\Sigma}_{i,j}| \gg \rho\}$

# Connections to correlation detection

Definition: Two vectors $x, y \in \mathbb{R}^n$ are $\rho$-correlated if $\left|\left\langle \frac{x}{\|x\|_2}, \frac{y}{\|y\|_2} \right\rangle\right| \geq \rho$

**Problem statement.** Correlation detection **with margin**

Input:

- vectors $y_1, \ldots, y_d \in \mathbb{R}^n$; $n \ll d$
- a threshold $\rho \in (0,1)$, a threshold $\tau \ll \rho$
- **very few**, say $o(d)$ out of $d^2$ pairs, are $\tau$-correlated

Output: all $\rho$-correlated pairs $(i,j) \in [d] \times [d]$

- **Naïve algorithm**: try all possible pairs, runs in $d^2$ time
- **[Val15]** gives an $o(d^2)$ algorithm if $\tau \ll \rho$

[Val15] G. Valiant. Finding Correlations in Subquadratic Time,... *J. ACM* (2015)

$H := \{(i,j) : i \neq j\ ,\ |\mathbf{\Sigma}_{i,j}| \gg \rho\}$

# Connections to correlation detection

Definition: Two vectors $x, y \in \mathbb{R}^n$ are $\rho$-correlated if $\left|\left\langle \frac{x}{\|x\|_2}, \frac{y}{\|y\|_2} \right\rangle\right| \geq \rho$

**Problem statement.** Correlation detection **with margin**

Input:
- vectors $y_1, \ldots, y_d \in \mathbb{R}^n$; $\quad n \ll d$
- a threshold $\rho \in (0,1)$, a threshold $\tau \ll \rho$
- **very few**, say $o(d)$ out of $d^2$ pairs, are $\tau$-correlated

Output: all $\rho$-correlated pairs $(i,j) \in [d] \times [d]$

- **Naïve algorithm**: try all possible pairs, runs in $d^2$ time
- **[Val15]** gives an $o(d^2)$ algorithm if $\tau \ll \rho$
  - **runtime** $\approx d^{1.6+\frac{1}{q}}$ if $\tau = \mathrm{poly}(\rho^q)$

[Val15] G. Valiant. Finding Correlations in Subquadratic Time,... *J. ACM* (2015)

$\|\mathbf{A}\|_{\mathrm{Fr},k^2} := \ell_2$ norm of largest $k^2$ entries of $\mathbf{A}$

# Filtering using fast correlation detection

While $\|\mathbf{\Sigma} - \mathbf{I}\|_{\mathrm{Fr},k^2}$ large:

Filter outliers

**Algorithm outline.**

1. $H \leftarrow \{(i,j) : |\mathbf{\Sigma}_{i,j}| \geq \rho\}$ $\quad \rho = 1/k$
2. $J \leftarrow \{(i,j) : |\mathbf{\Sigma}_{i,j}| \geq \tau\}$ $\quad \tau = \rho^{100}$
3. **While** $|H| \gg \mathrm{poly}(k)$:
   - **If** $|J| = o(d)$:
     - Use **[Val15]** to find $H$ and filter
   - **Else**
     - ????

$\|\mathbf{A}\|_{\mathrm{Fr},k^2} := \ell_2$ norm of largest $k^2$ entries of $\mathbf{A}$

While $\|\mathbf{\Sigma} - \mathbf{I}\|_{\mathrm{Fr},k^2}$ large:

Filter outliers

# Filtering using fast correlation detection

**Algorithm outline.**

1. $H \leftarrow \{(i,j) : |\mathbf{\Sigma}_{i,j}| \geq \rho\}$
2. $J \leftarrow \{(i,j) : |\mathbf{\Sigma}_{i,j}| \geq \tau\}$
3. **While** $|H| \gg \mathrm{poly}(k)$:
   - **If** $|J| = o(d)$:
     - Use **[Val15]** to find $H$ and filter
   - **Else**
     - ????

$\rho = 1/k$

$\tau = \rho^{100}$

How to calculate size of $J$

$\|\mathbf{A}\|_{\mathrm{Fr},k^2} := \ell_2$ norm of largest $k^2$ entries of $\mathbf{A}$

While $\|\mathbf{\Sigma} - \mathbf{I}\|_{\mathrm{Fr},k^2}$ large:

Filter outliers

# Filtering using fast correlation detection

**Algorithm outline.**

1. $H \leftarrow \{(i,j) : |\mathbf{\Sigma}_{i,j}| \geq \rho\}$
2. $J \leftarrow \{(i,j) : |\mathbf{\Sigma}_{i,j}| \geq \tau\}$
3. **While** $|H| \gg \mathrm{poly}(k)$:
   - **If** $|J| = o(d)$: ← How to calculate size of $J$
     - Use [Val15] to find $H$ and filter
   - **Else**
     - ????

$\rho = 1/k$

$\tau = \rho^{100}$

**Size of $J$**: randomly sample $d^{1.5}$ many $\{(i,j)\}$ & count $\tau$-correlation

- whp, $\Omega(\sqrt{d})$ hits **iff** $|J| = \Omega(d)$

$\|\mathbf{A}\|_{\mathrm{Fr},k^2} := \ell_2$ norm of largest $k^2$ entries of $\mathbf{A}$

While $\|\mathbf{\Sigma} - \mathbf{I}\|_{\mathrm{Fr},k^2}$ large:

Filter outliers

# Filtering using fast correlation detection

**Algorithm outline.**

1. $H \leftarrow \{(i,j) : |\mathbf{\Sigma}_{i,j}| \geq \rho\}$
2. $J \leftarrow \{(i,j) : |\mathbf{\Sigma}_{i,j}| \geq \tau\}$
3. **While** $|H| \gg \mathrm{poly}(k)$:
   - **If** $|J| = o(d)$:
     - Use [Val15] to find $H$ and filter
   - **Else**
     - ????

$\rho = 1/k$

$\tau = \rho^{100}$

How to calculate size of $J$

**Size of $J$**: randomly sample $d^{1.5}$ many $\{(i,j)\}$ & count $\tau$-correlation

- whp, $\Omega(\sqrt{d})$ hits **iff** $|J| = \Omega(d)$

$\|\mathbf{A}\|_{\mathrm{Fr},k^2} := \ell_2$ norm of largest $k^2$ entries of $\mathbf{A}$

# Filtering using fast correlation detection

While $\|\mathbf{\Sigma} - \mathbf{I}\|_{\mathrm{Fr},k^2}$ large:

Filter outliers

**Algorithm outline.**

1. $H \leftarrow \{(i,j) : |\mathbf{\Sigma}_{i,j}| \geq \rho\}$  $\rho = 1/k$
2. $J \leftarrow \{(i,j) : |\mathbf{\Sigma}_{i,j}| \geq \tau\}$  $\tau = \rho^{100}$
3. **While** $|H| \gg \mathrm{poly}(k)$:
   - ▷ $R \leftarrow$ a set of $d^{1.5}$ randomly sampled $(i,j)$
   - ▷ $\widehat{J} \leftarrow \{\ (i,j) \in R : |\mathbf{\Sigma}_{i,j}| \geq \tau\ \}$
   - ▷ **If** $|\widehat{J}| = o(\sqrt{d})$ :
     - ▷ Use [Val15] to find $H$ and filter
   - ▷ **Else**
     - ▷ ????

**Size of $J$**: randomly sample $d^{1.5}$ many $\{(i,j)\}$ & count $\tau$-correlation

- whp, $\Omega(\sqrt{d})$ hits **iff** $|J| = \Omega(d)$

$\|\mathbf{A}\|_{\mathrm{Fr},k^2} := \ell_2$ norm of largest $k^2$ entries of $\mathbf{A}$

# Filtering using fast correlation detection

While $\|\mathbf{\Sigma} - \mathbf{I}\|_{\mathrm{Fr},k^2}$ large:

Filter outliers

**Algorithm outline.**

1. $H \leftarrow \{(i,j) : |\mathbf{\Sigma}_{i,j}| \geq \rho\}$ $\quad \rho = 1/k$
2. $J \leftarrow \{(i,j) : |\mathbf{\Sigma}_{i,j}| \geq \tau\}$ $\quad \tau = \rho^{100}$
3. **While** $|H| \gg \mathrm{poly}(k)$:
   - $R \leftarrow$ a set of $d^{1.5}$ randomly sampled $(i,j)$
   - $\widehat{J} \leftarrow \{\ (i,j) \in R : |\mathbf{\Sigma}_{i,j}| \geq \tau\ \}$
   - **If** $|\widehat{J}| = o(\sqrt{d})$ :
     - Use [Val15] to find $H$ and filter
   - **Else**
     - ????

$\|\mathbf{A}\|_{\mathrm{Fr},k^2} := \ell_2$ norm of largest $k^2$ entries of $\mathbf{A}$

While $\|\mathbf{\Sigma} - \mathbf{I}\|_{\mathrm{Fr},k^2}$ large:

Filter outliers

# Filtering using fast correlation detection

**Algorithm outline.**

1. $H \leftarrow \{(i,j) : |\mathbf{\Sigma}_{i,j}| \geq \rho\}$ $\qquad \rho = 1/k$
2. $J \leftarrow \{(i,j) : |\mathbf{\Sigma}_{i,j}| \geq \tau\}$ $\qquad \tau = \rho^{100}$
3. **While** $|H| \gg \mathrm{poly}(k)$:
   - ▷ $R \leftarrow$ a set of $d^{1.5}$ randomly sampled $(i,j)$
   - ▷ $\widehat{J} \leftarrow \{\ (i,j) \in R : |\mathbf{\Sigma}_{i,j}| \geq \tau\ \}$
   - ▷ **If** $|\widehat{J}| = o(\sqrt{d})$ :
     - ▷ Use [Val15] to find $H$ and filter
   - ▷ **Else**
     - ▷ **????**

How to make progress when $|J|$ large?

$\|\mathbf{A}\|_{\mathrm{Fr},k^2} := \ell_2$ norm of largest $k^2$ entries of $\mathbf{A}$

While $\|\mathbf{\Sigma} - \mathbf{I}\|_{\mathrm{Fr},k^2}$ large:

Filter outliers

# Filtering using fast correlation detection

**Algorithm outline.**

$\rho = 1/k$

$\tau = \rho^{100}$

1. $H \leftarrow \{(i,j) : |\mathbf{\Sigma}_{i,j}| \geq \rho\}$
2. $J \leftarrow \{(i,j) : |\mathbf{\Sigma}_{i,j}| \geq \tau\}$
3. **While** $|H| \gg \mathrm{poly}(k)$:
   - ▷ $R \leftarrow$ a set of $d^{1.5}$ randomly sampled $(i,j)$
   - ▷ $\widehat{J} \leftarrow \{ (i,j) \in R : |\mathbf{\Sigma}_{i,j}| \geq \tau \}$
   - ▷ **If** $|\widehat{J}| = o(\sqrt{d})$ :
     - ▷ Use [Val15] to find $H$ and filter
   - ▷ **Else**
     - ▷ ????

**Filter**: If many entries bigger than $\tau$, then $\|\mathbf{\Sigma} - I\|_{\mathrm{Fr},\mathrm{poly}(1/\tau)} \gg 1$

- Can filter **if** stability holds with $k' = \mathrm{poly}(1/\tau)$

$\|\mathbf{A}\|_{\mathrm{Fr},k^2} := \ell_2$ norm of largest $k^2$ entries of $\mathbf{A}$

# Filtering using fast correlation detection

While $\|\mathbf{\Sigma} - \mathbf{I}\|_{\mathrm{Fr},k^2}$ large:

Filter outliers

**Algorithm outline.**

1. $H \leftarrow \{(i,j) : |\mathbf{\Sigma}_{i,j}| \geq \rho\}$ — $\rho = 1/k$
2. $J \leftarrow \{(i,j) : |\mathbf{\Sigma}_{i,j}| \geq \tau\}$ — $\tau = \rho^{100}$
3. **While** $|H| \gg \mathrm{poly}(k)$:
   - ▷ $R \leftarrow$ a set of $d^{1.5}$ randomly sampled $(i,j)$
   - ▷ $\widehat{J} \leftarrow \{\ (i,j) \in R : |\mathbf{\Sigma}_{i,j}| \geq \tau\ \}$
   - ▷ **If** $|\widehat{J}| = o(\sqrt{d})$ :
     - ▷ Use [Val15] to find $H$ and filter
   - ▷ **Else**
     - ▷ Filter using $\mathrm{poly}(1/\tau)$ coordinates in $R$

**Filter**: If many entries bigger than $\tau$, then $\|\mathbf{\Sigma} - I\|_{\mathrm{Fr},\mathrm{poly}(1/\tau)} \gg 1$

- Can filter **if** stability holds with $k' = \mathrm{poly}(1/\tau)$

$\|\mathbf{A}\|_{\mathrm{Fr},k^2} := \ell_2$ norm of largest $k^2$ entries of $\mathbf{A}$

# Filtering using fast correlation detection

While $\|\mathbf{\Sigma} - \mathbf{I}\|_{\mathrm{Fr},k^2}$ large:

Filter outliers

**The complete algorithm.**

1. $H \leftarrow \{(i,j) : |\mathbf{\Sigma}_{i,j}| \geq \rho\}$ $\rho = 1/k$
2. $J \leftarrow \{(i,j) : |\mathbf{\Sigma}_{i,j}| \geq \tau\}$ $\tau = \rho^{100}$
3. **While** $|H| \gg \mathrm{poly}(k)$:
   - ▷ $R \leftarrow$ a set of $d^{1.5}$ randomly sampled $(i,j)$
   - ▷ $\widehat{J} \leftarrow \{\ (i,j) \in R : |\mathbf{\Sigma}_{i,j}| \geq \tau\ \}$
   - ▷ **If** $|\widehat{J}| = o(\sqrt{d})$ :
     - ▷ Use [Val15] to find $H$ and filter
   - ▷ **Else**
     - ▷ Filter using $\mathrm{poly}(1/\tau)$ coordinates in $R$

# Conclusion

- Today: robust sparse estimation through the lens of mean estimation
- What we didn't discuss?
  - Sparsity in other contexts: PCA, linear regression, covariance,. . .
  - Privacy
  - Information-computation tradeoffs

# Conclusion

- Today: robust sparse estimation through the lens of mean estimation
- What we didn't discuss?
  - Sparsity in other contexts: PCA, linear regression, covariance,. . .
  - Privacy
  - Information-computation tradeoffs
- Open questions:
  - Similar progress on sparse PCA, linear regression,
  - Custom SDP solvers for $\{M \succeq 0; \mathrm{tr}(M) = 1; \|M\|_1 \leq k\}$
  - Relaxing assumptions on data distributions
  - Linear-time/Practical algorithms

# Conclusion

- Today: robust sparse estimation through the lens of mean estimation
- What we didn't discuss?
  - Sparsity in other contexts: PCA, linear regression, covariance,. . .
  - Privacy
  - Information-computation tradeoffs
- Open questions:
  - Similar progress on sparse PCA, linear regression,
  - Custom SDP solvers for $\{M \succeq 0; \mathrm{tr}(M) = 1; \|M\|_1 \leq k\}$
  - Relaxing assumptions on data distributions
  - Linear-time/Practical algorithms

**Happy to chat more**

# Thank You


[BDLS17] S. Balakrishnan, S. S. Du, J. Li, A. Singh. Computationally Efficient Robust Sparse Estimation.. *COLT*. 2017.

[CDG19] Y. Cheng, I. Diakonikolas, R. Ge. High-Dimensional Robust Mean Estimation in Nearly-Linear Time. *SODA*. 20

[CMY20] Y. Cherapanamjeri, S. Mohanty, M. Yau. List decodable mean estimation in nearly linear time. *FOCS*. 2020.

[dGJL07] A. d'Aspremont, L. Ghaoui, M. Jordan, G. Lanckriet. A direct formulation for sparse pca using SDP. 2007.

[DHL19] Y. Dong, S. Hopkins, J. Li. Quantum entropy scoring for fast robust mean estimation.. *NeurIPS*. 2019.

[DKKLMS16] I. Diakonikolas, G. Kamath, D. Kane, J. Li, A. Moitra, A. Stewart. Robust estimators in high... *FOCS*. 2016.

[DKKLT22] I. Diakonikolas, D. M. Kane, D. Kongsgaard, J. Li, K. Tian. Clustering Mixture Models in ..Linear.. *STOC*. 2022

[DKKPP22] I. Diakonikolas, D. Kane, S. Karmalkar, A. Pensia, T. Pittas. Robust Sparse Estimation via SoS. *COLT*. 2022.

[DKKPS19] I. Diakonikolas, D. Kane, S. Karmalkar, E. Price, A. Stewart. Outlier-Robust Sparse Estimation... *NeurIPS*. 2

[DKLP22] I. Diakonikolas, D. Kane, J. Lee, A. Pensia. Outlier-Robust Sparse Estimation for Heavy-Tailed. *NeurIPS*. 202

[DKPP22] I. Diakonikolas, D. Kane, A. Pensia, T. Pittas. Streaming Algorithms for .. Robust Statistics.. *ICML*. 2022.

[DL22] J. Depersin, G. Lecué. Robust Subgaussian Estimation of a Mean Vector in Nearly Linear Time. *Ann. Stats*. 202

[Li18] J. Li. Principled Approaches to Robust Machine Learning and Beyond. PhD thesis. 2018.

[LRV16] K. A. Lai, A. B. Rao, S. Vempala. Agnostic Estimation of Mean and Covariance. *FOCS*. 2016.

[Pen24] A. Pensia. A Sub-Quadratic Time Algorithm for Robust Sparse Mean Estimation. *ICML*. 2024.

[Val15] G. Valiant. Finding Correlations in Subquadratic Time,... *J. ACM* (2015).