Differential Privacy Meets Robust Statistics

Sewoong Oh

Paul G. Allen School of Computer Science and Engineering
University of Washington

joint work with

Xiyang Liu  Weihao Kong  Sham Kakade
What can go wrong when training on shared data?

- Increasingly more models are being trained on shared data
- Sensitive information should not be revealed by the trained model
- **Membership inference attacks** can identify individual’s sensitive data used in the training

Potential defense: **Differentially Private Stochastic Gradient Descent**

when computing the average of the gradients in the mini-batch, use differentially private mean estimation

* [Carlini et al., 2020]
† [Chaudhuri, Monteleoni, Sarwate, 2011], [Abadi et al., 2016]
What can go wrong when training on shared data?

- When training on shared data, not all participants are trusted
- Malicious users can easily inject corrupted data
- **Data poisoning attacks** can create backdoors on the trained model such that any sample with the trigger will be predicted as ‘deer’

\[ y_i = \text{‘deer’} \]

- **Strong defense:** Robust estimation
- **Insight:** successful backdoor attacks leave a path of activations in the trained model that are triggered only by the corrupted samples

*Hayase, Kong, Somani, O., 2021, ICML* inspired by *Tran, Li, Madry, 2018*
Middle layer of a model trained with corrupted data

- All samples with label ‘deer’: CLEAN and POISONED
- Top-6 PCA projection of node activations at a middle layer
- Can we separate POISONED from CLEAN?
Middle layer of a model trained with corrupted data

- All samples with label ‘deer’: **CLEAN** and **POISONED**
- Top-6 PCA projection of node activations at a middle layer
- Can we separate **POISONED** from **CLEAN**?

After whitening with the covariance of **CLEAN**
Middle layer of a model trained with corrupted data

- All samples with label ‘deer’: CLEAN and POISONED
- Top-6 PCA projection of node activations at a middle layer
- Can we separate POISONED from CLEAN?

After whitening with estimated robust mean and covariance
SPECTRE: Defense against backdoor attacks

[Hayase, Somani, Kong, O., 2021, ICML]

‡

No defense

Retrained on filtered data

‡https://github.com/SewoongLab/backdoor-suite
We need privacy and robustness, simultaneously

- When learning from shared data
  - **Differential privacy** is crucial in defending against inference attacks
  - **Robust estimation** is crucial in defending against data poisoning attacks

- Critical components are mean/covariance estimation
  - DP-SGD relies on DP mean estimation
  - Backdoor defense relies on robust mean/covariance estimation

- We provide the first efficient estimators that are provably differentially private and robust against data corruption
This talk focuses on mean estimation. What is the extra cost (in the estimation error) we pay for {Robustness, Privacy, and Robustness+Privacy}?
Statistical estimation, robustly and privately

- **Statistics** $\Rightarrow$ Robust estimation

Data $S_{\text{good}}$  

Data poisoning

$P_\theta$  

$\hat{\theta}$

![Diagram](image-url)
Statistical estimation, robustly and privately

- Statistics $\Rightarrow$ Robust estimation $\Rightarrow$ Robust and private estimation

This talk focuses on mean estimation

Q. What is the extra cost (in the estimation error) we pay for 
{Robustness, Privacy, and Robustness+Privacy}
Mean estimation

- Estimate the mean $\mu$ from $n$ i.i.d. samples
- For this talk, we assume sub-Gaussian distribution with identity covariance matrix
- Minimax error rate:

$$\min_{\hat{\mu} \in \mathcal{F}_{S_n}} \max_{P_\mu} \mathbb{E}[\|\hat{\mu}(S_n) - \mu\|] \propto \sqrt{\frac{d}{n}}$$

$\mathcal{F}_{S_n}$ is set of all estimators over $n$ i.i.d. samples in $\mathbb{R}^d$ from $P_\mu$, $P_\mu$ is maximized over all sub-Gaussian distributions with identity covariance
- In this talk, I will ignore all constant and logarithmic factors
Robust mean estimation

- Threat model
  - Adversarial corruption model:
    \( \{ x_i \}_{i=1}^{n} \sim P_\mu \) is drawn, then adversary replaces \( \alpha \)-fraction arbitrarily

- Robust mean estimation:
  - Low dimensional:
    - [Tukey, 1960] [Huber, 1964]
  - Computationally intractable methods in high dimension:
    - [Donoho, Liu, 1988], [ChenGaoRen, 2015], [Zhu, Jiao, Steinhardt, 2019]
  - Breakthroughs in polynomial time algorithms:
    - [Lai, Rao, Vempala, 2016], [Diakonikolas, Kamath, Kane, Li, Moitra, Stewart, 2019]
  - Linear time algorithms:
    - [Cheng, Dianikolas, Ge, 2019], [Depersin, Lecué, 2019], [Dong, Hopkins, Li, 2019]
Robust mean estimation

- Threat model
  - Adversarial corruption model:
    \[ \{x_i\}_{i=1}^n \sim P_\mu \] is drawn, then adversary replaces \( \alpha \)-fraction arbitrarily
- Relatively easy to estimate mean robustly in low-dimensions

![Histogram of sub-Gaussian samples in 1D](image)
Robust mean estimation

- Threat model
  - Adversarial corruption model:
    \[ \{x_i\}_{i=1}^n \sim P_\mu \] is drawn, then adversary replaces \( \alpha \)-fraction arbitrarily
- Relatively easy to estimate mean robustly in low-dimensions

simple outlier detection achieves \( |\hat{\mu} - \mu| \leq \alpha \)
Robust mean estimation

- Threat model
  - Adversarial corruption model:
    \[ \{x_i\}_{i=1}^n \sim P_\mu \text{ is drawn, then adversary replaces } \alpha \text{-fraction arbitrarily} \]
- Mean estimation becomes challenging in high-dimensions

scatter plot of sub-Gaussian samples in high-dimension

each corrupted sample looks uncorrupted and still \[ \|\hat{\mu} - \mu\| \geq \alpha \sqrt{d} \]
Efficient algorithm: Filtering [Diakonikolas et al., 2017]

Geometric Lemma [Dong, Hopkins, Li, 2019]

Given $n$ i.i.d. samples from a sub-Gaussian distribution with identity covariance matrix, if at most $\alpha n$ samples are corrupted, then, w.h.p.

$$\|\mu_{\text{emp}}(S) - \mu\| \leq \sqrt{\frac{d}{n}} + \alpha + \sqrt{\alpha}\|\text{Cov}(S) - I\|$$

- While $\|\text{Cov}(S) - I\| > c\alpha$
  - $v \leftarrow \arg\max_{v: \|v\|=1} v^T \text{Cov}(S)v$
  - $S \leftarrow 1D\text{-Filter}(\{\langle v, x_i - \mu_{\text{emp}}(S)\rangle^2\}_{i \in S})$

- Each step guarantees that
  - at least one sample is removed
  - more corrupted samples removed than clean samples in expectation
Efficient algorithm: Filtering [Diakonikolas et al., 2017]

Geometric Lemma [Dong, Hopkins, Li, 2019]

Given $n$ i.i.d. samples from a sub-Gaussian distribution with identity covariance matrix, if at most $\alpha n$ samples are corrupted, then, w.h.p.

$$\|\mu_{\text{emp}}(S) - \mu\| \leq \sqrt{\frac{d}{n}} + \alpha + \sqrt{\alpha \|\text{Cov}(S) - I\|}$$

- While $\|\text{Cov}(S) - I\| > c\alpha$
  - $v \leftarrow \arg \max_{v: \|v\| = 1} v^T \text{Cov}(S)v$
  - $S \leftarrow 1D\text{-Filter}(\{\langle v, x_i - \mu_{\text{emp}}(S) \rangle^2 \}_{i \in S})$

- Each step guarantees that
  - at least one sample is removed
  - more corrupted samples removed than clean samples in expectation
Efficient algorithm: Filtering [Diakonikolas et al., 2017]

Geometric Lemma [Dong, Hopkins, Li, 2019]

Given $n$ i.i.d. samples from a sub-Gaussian distribution with identity covariance matrix, if at most $\alpha n$ samples are corrupted, then, w.h.p.

$$
\|\mu_{\text{emp}}(S) - \mu\| \leq \sqrt{\frac{d}{n}} + \alpha + \sqrt{\alpha} \|\text{Cov}(S) - I\|
$$

- While $\|\text{Cov}(S) - I\| > c \alpha$
  - $v \leftarrow \arg \max_{v: \|v\|=1} v^T \text{Cov}(S)v$
  - $S \leftarrow 1D\text{-Filter}(\{\langle v, x_i - \mu_{\text{emp}}(S)\rangle^2\}_{i \in S})$

Each step guarantees that
  - at least one sample is removed
  - more corrupted samples removed than clean samples in expectation
Efficient algorithm: Filtering [Diakonikolas et al., 2017]

Geometric Lemma [Dong, Hopkins, Li, 2019]

Given $n$ i.i.d. samples from a sub-Gaussian distribution with identity covariance matrix, if at most $\alpha n$ samples are corrupted, then, w.h.p.

$$\|\mu_{\text{emp}}(S) - \mu\| \leq \sqrt{\frac{d}{n}} + \alpha + \sqrt{\alpha}\|\text{Cov}(S) - I\|$$

- While $\|\text{Cov}(S) - I\| > c\alpha$
  - $v \leftarrow \arg\max_{v: \|v\|=1} v^T \text{Cov}(S)v$
  - $S \leftarrow 1D-\text{Filter}(\{\langle v, x_i - \mu_{\text{emp}}(S)\rangle^2\}_{i \in S})$

- Each step guarantees that
  - at least one sample is removed
  - more corrupted samples removed than clean samples in expectation
Robust mean estimation

- Minimax error rate under $\alpha$-corruption

\[
\min_{\hat{\mu}} \max_{P_\mu} \mathbb{E} \left[ \| \hat{\mu}(S_{n,\alpha}) - \mu \| \right] \propto \sqrt{\frac{d}{n}} + \alpha
\]

achieved by filtering algorithm of [Diakonikolas et al., 2017]
information-theoretic lower bound from [Chen, Gao, Ren, 2015]
Minimax error rate for mean estimation under sub-Gaussian distributions with identity covariance

|                     | Error $||\hat{\mu} - \mu||$ |
|---------------------|-------------------------------|
| no corruption or privacy | $\sqrt{\frac{d}{n}}$         |
| $\alpha$-corruption  | $\sqrt{\frac{d}{n}} + \alpha$ [Diakonikolas et al., 2017] |
| $(\varepsilon, \delta)$-DP |                                |
| $\alpha$-corruption and $(\varepsilon, \delta)$-DP |                                |
Differential Privacy provably ensures plausible deniability

- Goal: a strong adversary who knows all the other entries in the database except for yours, should not be able to identify whether you participated in that database or not

- Definition*: For two databases $S$ and $S'$ that differ by only one entry, a randomized output to a query is $(\varepsilon, \delta)$-differentially private if

  $$\mathbb{P}(query_output(S) \in A) \leq e^\varepsilon \mathbb{P}(query_output(S') \in A) + \delta$$

- smaller $\varepsilon, \delta \Rightarrow$ Testing $S$ or $S'$ fails $\Rightarrow$ inference attack fails

* [Dwork, McSherry, Nissim, Smith, 2006]
\((\varepsilon, \delta)\)-differentially private mean estimation

<table>
<thead>
<tr>
<th>(S)</th>
<th>0</th>
<th>1</th>
<th>0</th>
<th>0</th>
<th>0</th>
<th>1</th>
<th>\cdots</th>
<th>0</th>
</tr>
</thead>
</table>

\[
\hat{\mu}(S) = \mu(S) + N(0, (\Delta \sqrt{\log \frac{1}{\delta}} / \varepsilon)^2)
\]

The extra error due to \((\varepsilon, \delta)\)-DP is
\[
|\hat{\mu}(S) - \mu(S)| \approx \Delta \varepsilon = \frac{1}{n\varepsilon}
\]
$(\varepsilon, \delta)$-differentially private mean estimation

\[ \hat{\mu}(S) = \mu(S) + \mathcal{N}(0, (\Delta \sqrt{\log \frac{1}{\delta}})^2) \]

extra error due to $(\varepsilon, \delta)$-DP is

\[ \frac{1}{n} = \text{sensitivity } \Delta \]
\((\varepsilon, \delta)\)-differentially private mean estimation

\[
\hat{\mu}(S) = \mu(S) + \mathcal{N}\left(0, \left(\frac{\Delta \sqrt{\log 1/\delta}}{\varepsilon}\right)^2\right)
\]

- extra error due to \((\varepsilon, \delta)\)-DP is

\[
|\hat{\mu}(S) - \mu(S)| \simeq \frac{\Delta}{\varepsilon} = \frac{1}{n \varepsilon}
\]
(ε, δ)-differentially private mean estimation

step 1. privately find a bounding hypercube
step 2. add Gaussian noise:
\[ \hat{\mu}(S) = \mu(S) + N(0, (\Delta \sqrt{\log 1/\delta})^2 I_{d \times d}) \]

extra error due to (ε, δ)-DP is
\[ \| \hat{\mu}(S) - \mu(S) \| \approx \Delta \sqrt{d} = d^{n/2} \]

*[Karwa, Vadhan, 2017], [Kamath, Li, Singhal, Ullman, 2019]*
$(\varepsilon, \delta)$-differentially private mean estimation

- step 1. privately find a bounding hypercube

- extra error due to $(\varepsilon, \delta)$-DP is $\|\hat{\mu}(S) - \mu(S)\| \approx \Delta \varepsilon^{\frac{1}{d}} \approx \sqrt{d n \varepsilon}$

* [Karwa, Vadhan, 2017], [Kamath, Li, Singhal, Ullman, 2019]
$(\epsilon, \delta)$-differentially private mean estimation

- step 1. privately find a bounding hypercube

---

* [Karwa, Vadhan, 2017], [Kamath, Li, Singhal, Ullman, 2019]
(ε, δ)-differentially private mean estimation

- step 1. privately find a bounding hypercube
- step 2. add Gaussian noise: \( \hat{\mu}(S) = \mu(S) + N\left( 0, \left( \frac{\Delta \sqrt{\log 1/\delta}}{\varepsilon} \right)^2 I_{d \times d} \right) \)
- extra error due to (ε, δ)-DP is

\[
\| \hat{\mu}(S) - \mu(S) \| \simeq \frac{\Delta}{\varepsilon} \sqrt{d} = \frac{d}{n \varepsilon}
\]

\* [Karwa, Vadhan, 2017], [Kamath, Li, Singhal, Ullman, 2019]
Minimax error rate for mean estimation under sub-Gaussian distribution with identity covariance

| Error $||\hat{\mu} - \mu||$ | no corruption or privacy $\sqrt{\frac{d}{n}}$ | $\alpha$-corruption $\sqrt{\frac{d}{n} + \alpha}$ [Diakonikolas et al., 2017] | $(\varepsilon, \delta)$-DP $\sqrt{\frac{d}{n} + \frac{d}{\varepsilon n}}$ [Kamath, Li, Singhal, Ullman, 2019] | $\alpha$-corruption and $(\varepsilon, \delta)$-DP |
Two main challenges in making filtering algorithms private

Algorithm (non-private) robust mean estimation [Diakonikolas et al., 2017]

1: while $\|\text{Cov}(S) - I\| > c\alpha$ do
2: $v \leftarrow \arg \max_{v: \|v\|=1} v^T \text{Cov}(S)v$
3: $S \leftarrow \text{1D-Filter}(\{\langle v, x_i - \mu_{\text{emp}}(S) \rangle^2 \}_{i \in S})$

- First challenge:
  - in the worst case, the filter runs for $O(d)$ iterations
  - this happens if corrupted sample are spread out in orthogonal directions
  - because the filter only checks 1-dimensional subspace at a time

- This is particularly damaging for privacy, as more iterations mean more privacy leakage
Two main challenges in making filtering algorithms private

Algorithm Quantum robust mean estimation [Dong, Hopkins, Li, 2019]

1: while $||\text{Cov}(S) - I|| > c \alpha$ do
2: $V \leftarrow \frac{1}{\text{Trace}(\exp\{\beta \text{Cov}(S)\})} \exp\{\beta \text{Cov}(S)\}$
3: $S \leftarrow 1D\text{-Filter}(\{(x_i - \mu_{\text{emp}}(S))^T V (x_i - \mu_{\text{emp}}(S))\}_{i \in S})$
Two main challenges in making filtering algorithms private

Algorithm  Quantum robust mean estimation [Dong, Hopkins, Li, 2019]

1: while $\|\text{Cov}(S) - I\| > c\alpha$ do
2: $V \leftarrow \frac{1}{\text{Trace}(\exp\{\beta\text{Cov}(S)\})} \exp\{\beta\text{Cov}(S)\}$
3: $S \leftarrow 1\text{D-Filter}(\{(x_i - \mu_{\text{emp}}(S))^T V (x_i - \mu_{\text{emp}}(S))\}_{i \in S})$

- If $\beta = \infty$, this recovers top PCA and uses only one-dimensional subspace
- If $\beta = 0$, this filters on $\|x_i - \mu_{\text{emp}}(S)\|^2$ treating all directions equally
- For appropriate $\beta$, iterations reduce from $O(d)$ to $O((\log d)^2)$
Two main challenges in making filtering algorithms private

**Algorithm** Quantum robust mean estimation [Dong, Hopkins, Li, 2019]

1: **while** $\|\text{Cov}(S) - I\| > c\alpha$ **do**
2: \[ V \leftarrow \frac{1}{\text{Trace}(\exp\{\beta\text{Cov}(S)\})} \exp\{\beta\text{Cov}(S)\} \]
3: \[ S \leftarrow 1D-\text{Filter}(\{(x_i - \mu_{\text{emp}}(S))^T V (x_i - \mu_{\text{emp}}(S))\}_{i \in S}) \]

- **Second challenge:**
  - 1D-Filter has high sensitivity
  - each sample is independently filtered with probability proportional to
  \[ \tau_i \triangleq (x_i - \mu_{\text{emp}}(S))^T V (x_i - \mu_{\text{emp}}(S)) \]
Two main challenges in making filtering algorithms private

Algorithm Quantum robust mean estimation [Dong, Hopkins, Li, 2019]

1: while $\|\text{Cov}(S) - I\| > c \alpha$ do
2: $V \leftarrow \frac{1}{\text{Trace}(\exp\{\beta \text{Cov}(S)\})} \exp\{\beta \text{Cov}(S)\}$
3: $S \leftarrow 1D\text{-Filter}(\{(x_i - \mu_{\text{emp}}(S))^T V (x_i - \mu_{\text{emp}}(S))\}_{i \in S})$

- Second challenge:
  - 1D-Filter has high sensitivity
  - each sample is independently filtered with probability proportional to
    $\tau_i \triangleq (x_i - \mu_{\text{emp}}(S))^T V (x_i - \mu_{\text{emp}}(S))$
Two main challenges in making filtering algorithms private

<table>
<thead>
<tr>
<th>Algorithm</th>
<th>Quantum robust mean estimation [Dong, Hopkins, Li, 2019]</th>
</tr>
</thead>
<tbody>
<tr>
<td>1: while ( | \text{Cov}(S) - I | &gt; c \alpha ) do</td>
<td></td>
</tr>
<tr>
<td>2: ( V \leftarrow \frac{1}{\text{Trace} \left( \exp \left{ \beta \text{Cov}(S) \right} \right)} \exp \left{ \beta \text{Cov}(S) \right} )</td>
<td></td>
</tr>
<tr>
<td>3: ( S \leftarrow 1D-\text{Filter} \left( \left{ (x_i - \mu_{\text{emp}}(S))^T V (x_i - \mu_{\text{emp}}(S)) \right}_{i \in S} \right) )</td>
<td></td>
</tr>
</tbody>
</table>

- Second challenge:
  - 1D-Filter has high sensitivity
  - each sample is independently filtered with probability proportional to \( \tau_i \triangleq (x_i - \mu_{\text{emp}}(S))^T V (x_i - \mu_{\text{emp}}(S)) \)
Two main challenges in making filtering algorithms private

**Algorithm** Quantum robust mean estimation [Dong, Hopkins, Li, 2019]

1: while \( \| \text{Cov}(S) - I \| > c \alpha \) do

2: \( V \leftarrow \frac{1}{\text{Trace}(\exp\{\beta \text{Cov}(S)\})} \exp\{\beta \text{Cov}(S)\} \)

3: \( S \leftarrow 1D\text{-Filter}\{ (x_i - \mu_{\text{emp}}(S))^T V (x_i - \mu_{\text{emp}}(S)) \}_{i \in S} \)

- Second challenge:
  - 1D-Filter has high sensitivity
  - each sample is independently filtered with probability proportional to \( \tau_i \triangleq (x_i - \mu_{\text{emp}}(S))^T V (x_i - \mu_{\text{emp}}(S)) \)

Two datasets lead to independent filtering, and sensitivity blows up
Two main challenges in making filtering algorithms private

**Algorithm Quantum robust mean estimation [Dong,Hopkins,Li,2019]**

1: while $\|\text{Cov}(S) - I\| > c \alpha$ do
2: $V \leftarrow \frac{1}{\text{Trace}(\exp\{\beta \text{Cov}(S)\})} \exp\{\beta \text{Cov}(S)\}$
3: $S \leftarrow 1D\text{-Filter}(\{(x_i - \mu_{\text{emp}}(S))^T V (x_i - \mu_{\text{emp}}(S))\}_i \in S)$

- **Solution:**
  - Use a **single** random threshold $Z \sim \text{Uniform}[0, \rho]$, and filter samples above $Z$.
  - this preserves the sensitivity to be one

![Diagram showing filtering with threshold Z and two sets S and S']
Two main challenges in making filtering algorithms private

**Algorithm**  Quantum robust mean estimation [Dong, Hopkins, Li, 2019]

<table>
<thead>
<tr>
<th>Step</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.</td>
<td>[ \text{while } | \text{Cov}(S) - I | &gt; c \alpha \text{ do} ]</td>
</tr>
<tr>
<td>2.</td>
<td>[ V \leftarrow \frac{1}{\text{Trace}(\exp{ \beta \text{Cov}(S) })} \exp{ \beta \text{Cov}(S) } ]</td>
</tr>
<tr>
<td>3.</td>
<td>[ S \leftarrow 1\text{D-Filter}({(x_i - \mu_{\text{emp}}(S))^T V (x_i - \mu_{\text{emp}}(S))}_{i \in S}) ]</td>
</tr>
</tbody>
</table>

**Solution:**
- Use a *single* random threshold \( Z \sim \text{Uniform}[0, \rho] \), and filter samples above \( Z \)
- this preserves the sensitivity to be one

![Diagram showing filtering process](image)
PRIME: PRIvate and robust Mean Estimation

- Run private histogram to get a bounding hypercube
- While $\|\tilde{\Sigma} - I\| > c\alpha$
  - $\tilde{\mu} \leftarrow \mu_{\text{emp}}(S) + \mathcal{N}\left(0, \left(\frac{d^{1/2}\sqrt{\log(1/\delta)}}{n\varepsilon}\right)^2 I_{d \times d}\right)$
  - $\tilde{\Sigma} \leftarrow \text{Cov}(S) + \mathcal{N}\left(0, \left(\frac{d\sqrt{\log(1/\delta)}}{n\varepsilon}\right)^2 I_{d^2 \times d^2}\right)$
  - $V \leftarrow \frac{1}{\text{Trace}(\exp\{\beta\tilde{\Sigma}\})} \exp\{\beta\tilde{\Sigma}\}$
  - $\rho \leftarrow \text{DP-threshold}\left(\{(x_i - \tilde{\mu})^T V (x_i - \tilde{\mu})\}_{i \in S}\right)$
  - $Z \leftarrow \text{Uniform}[0, \rho]$  
  - $S \leftarrow 1\text{D-Filter}(\{(x_i - \tilde{\mu})^T V (x_i - \tilde{\mu})\}_{i \in S, Z})$

Theorem. [Liu,Kong,Kakade,O.,2021,NeurIPS]

PRIME is $(\varepsilon, \delta)$-differentially private. For an $\alpha$-corruption of $n$ i.i.d. samples from a sub-Gaussian distribution with identity covariance matrix, with high probability

$$\|\hat{\mu} - \mu\| \lesssim \sqrt{\frac{d}{n}} + \alpha + \frac{d^{3/2}}{\varepsilon n}.$$
Mean estimation under sub-Gaussian distributions with identity covariance

| Error $||\hat{\mu} - \mu||$ | no corruption or privacy $\sqrt{\frac{d}{n}}$ | $\alpha$-corruption $\sqrt{\frac{d}{n}} + \alpha$ [Diakonikolas et al., 2017] |
|-------------------------------|---------------------------------------------|-------------------------------------------------|
|                               |                                             | $(\varepsilon, \delta)$-DP $\sqrt{\frac{d}{n}} + \frac{d}{\varepsilon n}$ [KamathLiSinghalUllman., 2019] |
|                               |                                             | $\alpha$-corruption and $(\varepsilon, \delta)$-DP $\sqrt{\frac{d}{n}} + \alpha + \frac{d^{3/2}}{\varepsilon n}$ (SVD time) [LiuKongKakadeO., 2021] |

There is a $d^{1/2}$ gap between PRIME and lower bound!
Where does $\frac{d^{3/2}}{\varepsilon n}$ come from?

- Sample complexity bottleneck: we need to privately compute
  \[ \tilde{\Sigma} \leftarrow \text{Cov}(S) + W \]

- Best known algorithm adds i.i.d. entry Gaussian matrix $W \in \mathbb{R}^{d \times d}$
  with $\mathcal{N}(0, (\frac{d\sqrt{\log 1/\delta}}{\varepsilon n})^2)$ to the covariance matrix

- The spectral norm perturbation is $\|W\|_{\text{spectral}} = O\left(\frac{d^{3/2}}{\varepsilon n}\right)$

- In general, this cannot be improved as it matches a known lower bound [Dwork, Talwar, Thakurta, Zhang, 2014]
### Minimax optimal mean estimation

<table>
<thead>
<tr>
<th></th>
<th>Error $|\hat{\mu} - \mu|$</th>
</tr>
</thead>
<tbody>
<tr>
<td>no corruption or privacy</td>
<td>$\sqrt{\frac{d}{n}}$</td>
</tr>
<tr>
<td>$\alpha$-corruption</td>
<td>$\sqrt{\frac{d}{n}} + \alpha$ [Diakonikolas et al., 2017]</td>
</tr>
<tr>
<td>$(\varepsilon, \delta)$-DP</td>
<td>$\sqrt{\frac{d}{n}} + \frac{d}{\varepsilon n}$ [KamathLiSinghalUllman., 2019]</td>
</tr>
<tr>
<td>$\alpha$-corruption and $(\varepsilon, \delta)$-DP (SVD time)</td>
<td>$\sqrt{\frac{d}{n}} + \alpha + \frac{d^{3/2}}{\varepsilon n}$ [LiuKongKakadeO., 2021]</td>
</tr>
<tr>
<td></td>
<td>$\sqrt{\frac{d}{n}} + \alpha + \frac{d}{\varepsilon n}$ (exponential time)</td>
</tr>
</tbody>
</table>

There is no extra *statistical* cost in requiring robustness and privacy simultaneously.
High-dimensional Propose-Test-Release
What is the fundamental connection between robust estimators and DP estimators?
High-dimensional Propose-Test-Release

- General framework for solving (inefficiently) statistical estimation problems with $(\varepsilon, \delta)$-DP guarantee

- as a byproduct, we get robustness against $\alpha$-corruption for free

- gives optimal sample complexity for mean estimation, covariance estimation, linear regression, and principal component analysis
HPTR step 1: design the score function

- Problem instance:
  mean estimation with i.i.d. samples from a sub-Gaussian distribution with mean $\mu$ and covariance $\Sigma$ with error metric

$$\| \Sigma^{-1/2} (\hat{\mu} - \mu) \|$$
HPTTR step 1: design the score function

- Problem instance:
  mean estimation with i.i.d. samples from a sub-Gaussian distribution with mean $\mu$ and covariance $\Sigma$ with error metric
  $$\|\Sigma^{-1/2}(\hat{\mu} - \mu)\|$$

- Polynomial-time [Kamath, Mouzakis, Singhal, Steinke, Ullman, 2021]:
  if $n \geq d^{5/2}/\varepsilon$
  $$\|\Sigma^{-1/2}(\hat{\mu} - \mu)\| \leq \sqrt{\frac{d}{n}} + \frac{d}{\varepsilon n}$$

- Exponential-time [Brown, Gaboardi, Smith, Ullman, Zakynthinou, 2021]:
  $$\|\Sigma^{-1/2}(\hat{\mu} - \mu)\| \leq \sqrt{\frac{d}{n}} + \frac{d}{\varepsilon^2 n}$$

- Lower bound [Barber, Duchi, 2014]:
  $$\min_{\hat{\mu} \in \mathcal{F}_{\varepsilon, \delta}} \max_{P_{\mu, \Sigma}} \mathbb{E} \left[ \|\Sigma^{-1/2}(\hat{\mu} - \mu)\| \right] \geq \sqrt{\frac{d}{n}} + \frac{d}{\varepsilon n}$$
HPTTR step 1: design the score function

- Problem instance:
  mean estimation with i.i.d. samples from a sub-Gaussian distribution with mean $\mu$ and covariance $\Sigma$ with error metric

$$\|\Sigma^{-1/2}(\hat{\mu} - \mu)\|$$
HPTTR step 1: design the score function

- Problem instance:
  mean estimation with i.i.d. samples from a sub-Gaussian distribution with mean $\mu$ and covariance $\Sigma$ with error metric

\[
\|\Sigma^{-1/2}(\hat{\mu} - \mu)\| = \max_{\|v\|=1} v^T \Sigma^{-1/2}(\hat{\mu} - \mu)
\]

\[
= \max_{\|v\|=1} \sqrt{v^T \Sigma v} \sqrt{\frac{v^T \hat{\mu} - v^T \mu}{\sigma_v}}
\]
HPTR step 1: design the score function

- Problem instance:
  mean estimation with i.i.d. samples from a sub-Gaussian distribution
  with mean $\mu$ and covariance $\Sigma$ with error metric

\[
\|\Sigma^{-1/2} (\hat{\mu} - \mu)\| = \max_{\|v\|=1} v^T \Sigma^{-1/2} (\hat{\mu} - \mu)
\]

\[
= \max_{\|v\|=1} \frac{v^T \hat{\mu} - v^T \mu}{\sqrt{v^T \Sigma v}}
\]

- Design empirical loss function:

\[
D_S(\hat{\mu}) = \max_{\|v\|=1} \frac{v^T \hat{\mu} - \mu_{v}^{\text{robust}}}{\sigma_{v}^{\text{robust}}}
\]
H PTR step 2: sensitivity analysis

We want to minimize the loss function:

\[ D_S(\hat{\mu}) = \max_{\|v\|=1} \frac{v^T \hat{\mu} - \mu_v^{\text{robust}}}{\sigma_v^{\text{robust}}} \]

- To stochastically minimize this robust empirical loss, we want to sample from (exponential mechanism*)

\[ \hat{\mu} \sim \frac{1}{Z} \exp \left\{ -\frac{\varepsilon}{2\Delta} D_S(\hat{\mu}) \right\} \]

- If \( \Delta \) is the sensitivity, then this is \((\varepsilon, 0)\)-differentially private
- The sensitivity of \( D_S(\hat{\mu}) \) dramatically reduces if we use 1-d robust statistics
- Key ingredient is resilience property

* [McSherry, Talwar, 2007]
H PTR step 2: sensitivity analysis

- \( \mu_v^{\text{robust}} = \frac{1}{|\mathcal{M}|} \sum_{\mathcal{M}} v^T x_i \) has sensitivity \( \Delta = \frac{\sigma_v}{n} \)
HPTR step 2: sensitivity analysis

- \( \mu^\text{robust} = \frac{1}{|M|} \sum_M v^T x_i \) has sensitivity \( \Delta = \frac{\sigma_v}{n} \)

Resilience property for sub-Gaussian [Steinhardt, Charikar, Valiant, 2018]

Given \( n \) i.i.d. sub-Gaussian samples \( S \) with \( n \geq d/\alpha^2 \), for all \( S' \subset S \) of size at least \( \alpha n \),

\[
| v^T (\mu(S) - \mu(S')) | \leq \sigma_v
\]
High-dimensional Propose-Test-Release*

- **HPT(R(S))**
  - **Propose**: Propose $\Delta = O(1/n)$ based on the resilience of the distribution
  - **Test**: Privately test the sensitivity for all neighboring dataset $S'$
  - **Release**: If $S$ passes the test, release $\hat{\mu}$ sampled from

$$\hat{\mu} \sim \frac{1}{Z} \exp \left\{ - \frac{\epsilon}{2\Delta} D_S(\hat{\mu}) \right\}$$

*inspired by original PTR [Dwork, Lei, 2009] and a more advanced PTR [Brown, Gaboardi, Smith, Ullman, Zakynthinou, 2021]*
Generality of HPTR

- HPTR can be applied to any statistical estimation problem to achieve the near-optimal error rate under \((\varepsilon, \delta)\)-DP
  - sub-Gaussian mean estimation:
    \[
    \| \Sigma^{-1/2} (\hat{\mu} - \mu) \| = O\left( \sqrt{\frac{d}{n}} + \frac{d}{\varepsilon n} \right)
    \]
  - \(k\)-th moment bounded mean estimation:
    \[
    \| \Sigma^{-1/2} (\hat{\mu} - \mu) \| = O\left( \sqrt{\frac{d}{n}} + \left( \frac{d}{\varepsilon n} \right)^{1 - \frac{1}{k}} \right)
    \]
  - sub-Gaussian linear regression:
    \[
    \| \Sigma^{1/2} (\hat{\beta} - \beta) \| = O\left( \sqrt{\frac{d}{n}} + \frac{d}{\varepsilon n} \right)
    \]
  - Gaussian covariance estimation:
    \[
    \| \Sigma^{-1/2} \hat{\Sigma} \Sigma^{-1/2} - I \|_F = O\left( \sqrt{\frac{d^2}{n}} + \frac{d^2}{\varepsilon n} \right)
    \]
  - sub-Gaussian principal component analysis:
    \[
    1 - \frac{\hat{v}^\top \Sigma \hat{v}}{\| \Sigma \|} = O\left( \sqrt{\frac{d}{n}} + \frac{d}{\varepsilon n} \right)
    \]
Conclusion and open questions

- First half of the talk, we gave the first efficient algorithm that achieves both differential privacy and robustness:

\[
\| \hat{\mu} - \mu \| \leq \sqrt{\frac{d}{n}} + \alpha + \frac{d^{1.5}}{\varepsilon n}
\]

\[
\| \Sigma^{-1/2} \hat{\Sigma} \Sigma^{-1/2} - \mathbf{I} \|_F \leq \sqrt{\frac{d^2}{n}} + \alpha + \frac{d^3}{\varepsilon n}
\]

- Can we have an efficient algorithm that closes the \(d^{1/2}\) gap (for mean)?
- Can we use it to make DP-SGD robust?
- Can we use it to make defense against backdoor attacks (such as SPECTRE) also private?
- Can we design efficient algorithms for other problems:
  - Principal component analysis, linear regression, convex optimization
Conclusion and open questions

- Second half of the talk, we introduced HPTR that achieves optimal error rate on mean estimation, covariance estimation, linear regression, and PCA
  - Characterize fundamental tradeoffs in structured data (sparsity and low-rank)
  - Characterize fundamental tradeoffs in discrete or graph data

- arXiv:2102.09159, Xiyang Liu, Weihao Kong, Sham Kakade, Sewoong Oh “Robust and Differentially Private Mean Estimation”
- arxiv:2111.06578, Xiyang Liu, Weihao Kong, Sewoong Oh “Differential Privacy and Robust Statistics in High Dimensions”