Advances in Private Statistics

The Case for Covariance Estimation

Argyris Mouzakis (University of Waterloo)

December 22, 2021

Overview

Statistical Estimation - Basic Setting

- Let $\mathcal{D}$ be an unknown distribution in $\mathbb{R}^d$ and $\theta = \theta(\mathcal{D})$ be some quantity associated with it.
- Given $X_1, \ldots, X_n$ i.i.d. samples from $\mathcal{D}$, how can we design estimators $\hat{\theta} = \hat{\theta}(X_{1,\ldots,n})$ to approximate θ?
- Targets:
  - small error (denoted by α).
  - small probability of error exceeding α (denoted by β).
  - sample efficiency ($n = \tilde{\mathcal{O}}\left(poly\left(d, \frac{1}{\alpha}, \frac{1}{\beta}\right)\right)$ samples should suffice).
  - computational efficiency (time complexity should be $\tilde{\mathcal{O}}(poly(n))$).

Covariance Estimation

- Today's focus: $\theta = \Sigma = \mathop{\mathbb{E}}_{X \sim \mathcal{D}} \left[(X - \mu)(X - \mu)^\intercal \right]$.
- The problem has been studied extensively by the statistics and tcs communities.
- The standard solution involves computing the *sample covariance*:

$$\widehat{\Sigma} = \frac{1}{n} \sum_{i=1}^{n} (X_i - \mu)(X_i - \mu)^\intercal .$$

- Why is this a good solution?
    - For many distributions, the above is the *MLE*, which boasts a number of desirable properties (*asymptotic unbiasedness, consistency, asymptotic minimization of MSE* etc) and, in this case, it's easy to compute (computational efficiency).
    - Optimality results are known for various distributions (e.g., Gaussians), though stronger tools are required other distributions (e.g., heavy-tailed).
    - What if privacy is an additional concern? 👀

Differential Privacy (DP)

- Privacy is a fundamental notion in the crypto/security community.
- DP is the main notion of privacy in statistical inference, where sensitive data may be involved.

Definition (Differential Privacy - see [4])

A randomized algorithm $M : \mathcal{X}^n \to \mathcal{Y}$ satisfies (ϵ, δ)-*DP* if for every pair of neighboring datasets[1] $X, X' \in \mathcal{X}^n$:

$$\forall Y \subseteq \mathcal{Y} : \mathbb{P}\left[M(X) \in Y\right] \leqslant e^{\epsilon}\mathbb{P}\left[M\left(X'\right) \in Y\right] + \delta.$$

- Depending on whether $\delta = 0$ or > 0, we say that M satisfies *pure DP* or *approx DP*, respectively.
- A related notion is that of Concentrated DP (CDP), which is known to be intermediate to the previous two.

[1] If X and X' are neighboring, they differ only on a single element.

Remarks on the Definition

- The following lemma formalizes the connection among the variants of DP claimed previously.

Lemma (see [3])

For every $\epsilon \geqslant 0$:

1. *If M satisfies $(\epsilon, 0)$-DP, then M is $\frac{\epsilon^2}{2}$-zCDP.*
2. *If M satisfies $\frac{\epsilon^2}{2}$-zCDP, then M satisfies $\left(\frac{\epsilon^2}{2} + \epsilon\sqrt{2\log\left(\frac{1}{\delta}\right)}, \delta\right)$-DP for every $\delta > 0$.*

- ϵ should be thought of as a small constant (e.g., between 0.1 and 5).
- δ should be thought of as cryptographically small (eg $\delta = \frac{1}{\omega(n)}$).

Properties of the Definition

Differential Privacy enjoys a number of very useful properties.

- Composition -> running multiple (potentially adaptively chosen) private mechanisms over a dataset does not violate privacy guarantee (only weakens it gradually).
- Closure under post-processing -> if the output of an algorithm is guaranteed to be private, it can be used without privacy being compromised.
- Group privacy -> datasets with Hamming distance greater than 1 still lead to roughly similar outputs.

Privatization

- How do we obtain DP algorithms from non-private ones?
- The main technique is by adding noise proportional to the *sensitivity* Δ_f of a non-private estimator $f\colon \mathcal{X}^n \to \mathcal{Y}$:

$$\Delta_f = \sup_{X \sim_h X'} \left\| f(X) - f\left(X'\right) \right\|,$$

where $\|\cdot\|$ is an appropriately chosen norm and $X \sim_h X'$ implies that X, X' have Hamming distance 1 (neighboring datasets).

The Laplace Mechanism

- The Laplace mechanism is the main tool for pure DP.

Theorem

Let $f : \mathcal{X}^n \to \mathbb{R}^d$ be a function with ℓ_1-sensitivity Δ_f. Then the Laplace mechanism[2]:

$$M_f(X) = f(X) + \mathrm{Lap}\left(\frac{\Delta_f}{\epsilon}\right)^{\otimes d},$$

satisfies $\epsilon-$DP.

[2] The Laplace distribution in one dimension $\mathrm{Lap}\,(b)$ has density $g\,(x) = \frac{1}{2b} e^{-\frac{|x|}{b}}$.

The Gaussian Mechanism

- The Gaussian mechanism is the main tool for cDP.

Theorem

Let $f : \mathcal{X}^n \to \mathbb{R}^d$ be a function with ℓ_2-sensitivity Δ_f. Then the Gaussian mechanism:

$$M_f(X) = f(X) + \mathcal{N}\left(0, \left(\frac{\Delta_f}{\sqrt{2\rho}}\right)^2 \cdot \mathbb{I}\right),$$

satisfies ρ-zCDP.

Problem Formulation

Problem

Let $\mathcal{D}$ be a distribution over $\mathbb{R}^d$ with $\underset{X\sim\mathcal{D}}{\mathbb{E}}[X] = 0$ and unknown covariance $\Sigma = \underset{X\sim\mathcal{D}}{\mathbb{E}}\left[XX^\top\right]$. Give a DP estimator $\widehat{\Sigma}$ such that:

$$\mathbb{P}\left[\left\|\widehat{\Sigma} - \Sigma\right\|_\Sigma > \alpha\right] \leqslant \beta,$$

with as few samples as possible.

- For some of our results, we will assume that $\mathbb{I} \preceq \Sigma \preceq u\mathbb{I}, u > 0$.
- The above is necessary to get pure DP and CDP guarantees (by lower bounds).
- Observe that the above formulation prioritizes sample efficiency.
- Some of our estimators will be *sample near-optimal but not time efficient* and others will be *time efficient but statistically sub-optimal*.

Assumptions on $\mathcal{D}$

- For the previous problem to be solvable, it is necessary to have some kind of assumptions about the behavior of the data-generating distribution $\mathcal{D}$.
- For some of our results, we will assume that $\mathcal{D}$ is a Gaussian distribution.
- This may be too restrictive, since it assumes that the distribution has a Moment Generating Function (aka all moments exist and are bounded).

Bounded Moments

Definition (Bounded Moments)

Let $\mathcal{D}$ be a distribution over $\mathbb{R}^d$ with mean μ and covariance Σ. We say that $X \sim \mathcal{D}$ has bounded moments of $2k$-th order for some $k \geqslant 2$ if there exists an absolute constant $C_{2k} \geqslant 1$ such that, for every unit vector v, we have:

$$\mathbb{E}\left[\langle v, X-\mu\rangle^{2k}\right] \leqslant C_{2k}\,\mathbb{E}\left[\langle v, X-\mu\rangle^{2}\right]^{k} = \left(v^{\mathsf{T}}\Sigma v\right)^{k}.$$

- The distributions satisfying this moment bound are known as $(C_{2k}, 2k)$-hypercontractive distributions.
- The above definition implies that, given $X, X' \sim \mathcal{D}$, the distribution of $\frac{X-X'}{\sqrt{2}}$ also satisfies it. Thus, we may assume that $\mu = 0$.
- We will assume that $C_{2k} = \mathcal{O}(1)$.

Related Prior Work (pre-2021)

- Karwa and Vadhan in [12] perform mean and variance estimation in the 1$-$D setting with (ϵ, δ) $-$DP.
- Kamath, Li, Singhal and Ullman [6] and Biswas, Dong, Kamath and Ullman [2] perform covariance estimation for $d-$dimensional sub-Gaussian distributions with CDP.
- Kamath, Singhal and Ullman [9] perform mean estimation for d-dimensional distributions with a finite number of bounded moments under CDP and pure DP.

Today's Results

Table 1: Sample Complexity Bounds for Covariance Estimation

Privacy Guarantee	**Gaussians**	**Bounded Moments**
CDP	-	$\tilde{\mathcal{O}}\left(\frac{d^2}{\alpha^2} + \frac{d^{2+\frac{1}{2(k-1)}}}{\sqrt{\rho}\alpha^{\frac{k}{k-1}}} + \frac{d^{\frac{3}{2}}\, poly(\log u)}{\sqrt{\rho}}\right)$ [7]
Approx DP	$\tilde{\mathcal{O}}\left(\frac{d^2}{\alpha^2} + \frac{d^2}{\alpha\epsilon} + \frac{d^{2.5}}{\epsilon}\right)$ [8]	-

- Results for Gaussians under CDP were given in prior work.
- We believe our result for Gaussians under approx dp can also be generalized to other classes of distributions, provided we have sufficiently strong concentration properties.
- We also give a *sample near-optimal but computationally inefficient* algorithm in [7] for pure DP under bounded moments, which we will not present today in full detail, but may sketch at the end (time permitting).

Other Recent Results (1)

- Liu, Kong and Oh in [11] define a general framework based on Propose-Test-Release (PTR) to obtain *sample-optimal but computationally inefficient* (ϵ, δ)-DP algorithms for a multitude of tasks (but not covariance estimation) for data that comes from a hypercontractive distribution. Their algorithms are robust to adversarial corruptions.
- Ashtiani and Liaw [1] define a general framework to reduce estimation under (ϵ, δ)-DP to its non-private counterpart, again based on PTR. They obtain a *computationally efficient and statistically near-optimal* algorithm for covariance estimation for Gaussians that is also robust to adversarial corruptions.

Other Recent Results (2)

- Hopkins, Kamath and Majid [5] give the *first computationally efficient and statistically near optimal* pure DP algorithm for mean estimation under bounded moments using the Sum-of-Squares proofs to algorithms framework. Their algorithm is also robust to adversarial corruptions.
- Kothari, Manurangsi and Velingker [10] define a general framework again based on SoS to obtain *computationally efficient but statistically sub-optimal* (ϵ, δ)-DP algorithms for a multitude of problems (including covariance estimation) for sub-Gaussian distributions. Their algorithms are also robust to adversarial corruptions.

The Naive Approach for CDP

- Since we assume the mean to be 0, the sample covariance is:

$$\widehat{\Sigma} = \frac{1}{n}\sum_{i=1}^{n} X_i X_i^{\mathsf{T}}.$$

- This is unbounded, so we must truncate our data within a ball centered at the origin.
- Since the distribution may be heavy-tailed, we can't pick a truncation radius such that all the dataset will be within the ball whp (as is the case with sub-gaussian data).
- We end up having to consider 3 types of error. These are: bias error due to truncation, noise error due to the DP requirement and sampling error (the only inherent of the 3).

The Naive Algorithm and its Analysis

Algorithm 1 Naïve Heavy Tailed Private Covariance Estimation

Input: $X = (X_1, \dots, X_n) \sim \mathcal{D}^{\otimes n}$. Parameters $\rho, \gamma > 0$.
Output: A noised covariance matrix M.

procedure $\textsc{NaïveHTPCE}_{\rho,\gamma,\beta}(X)$
  for $i \in [n]$ **do**
    Let $X_{i,\mathrm{tr},\gamma} = \mathbb{1}\{X_i \in B_\gamma(0)\} X_i$. ▷ Truncate the samples.
  end for
  Let $\sigma = \Theta\left(\frac{\gamma^2}{n\sqrt{\rho}}\right)$.
  Let $M' = \frac{1}{n}\sum_{i\in[n]} X_{i,\mathrm{tr},\gamma} X_{i,\mathrm{tr},\gamma}^\top + N$. ▷ $N \sim \mathrm{GOE}(\sigma^2)$.[3]
  Let M be the Euclidean projection of M' onto the PSD cone.
  return M.
end procedure

Theorem 3.1. *For every $\rho, \gamma > 0$, Algorithm 1 satisfies ρ-zCDP. Also, if $X_1, \dots, X_n \sim_{\text{i.i.d.}} \mathcal{D}$ with $\mathop{\mathbb{E}}_{X\sim\mathcal{D}}[X] = 0$, $\frac{1}{u}\mathbb{I} \preceq \Sigma \preceq \mathbb{I}$ that satisfies Definition 2.1 for some $1 \leqslant C_{2k} = \mathcal{O}(1)$ and $k \geqslant 2$, we have the following guarantees:*

1. *Setting $\gamma = C_{2k}^{\frac{1}{2(k-1)}} \cdot \frac{\sqrt{d}}{\left(\frac{a}{2}\right)^{\frac{1}{2(k-1)}}}$, it suffices to have $n = \mathcal{O}\left(\frac{d\log d}{a^2\beta^2} + \frac{d^{\frac{3}{2}}\log\left(\frac{1}{\beta}\right)}{a^{\frac{k}{k-1}}\rho^{\frac{1}{2}}}\right)$ samples so that $\|\Sigma - M\|_2 \leqslant a$ with probability at least $1-\beta$.*
2. *Setting $\gamma = C_{2k}^{\frac{1}{2(k-1)}} \cdot \frac{d^{\frac{2k-1}{4(k-1)}}}{\left(\frac{a}{2}\right)^{\frac{1}{2(k-1)}}}$, it suffices to have $n = \mathcal{O}\left(\frac{d^2}{a^2\beta} + \frac{ud^{2+\frac{1}{2(k-1)}}\log^{\frac{1}{2}}\left(\frac{1}{\beta}\right)}{\rho^{\frac{1}{2}}a^{\frac{k}{k-1}}}\right)$ samples so that $\|\Sigma - M\|_\Sigma \leqslant a$ with probability at least $1-\beta$.*

- Why is the dependence on u prohibitive for Mahalanobis estimation?
- What is going on in the exponent of d for Mahalanobis estimation?

Positive Definite Matrices as Ellipsoids

- There is a $1-1$ correspondence between ellipsoids and positive definite matrices.
- Let Σ be a positive definite matrix. Consider the set:

$$S = \left\{ x \in \mathbb{R}^d : \left\| \Sigma^{-\frac{1}{2}} x \right\|_2 = 1 \right\} = \left\{ x \in \mathbb{R}^d : x^\mathsf{T} \Sigma^{-1} x = 1 \right\}.$$

- If the matrix is diagonal, the equation is equivalent to $\sum_{i=1}^d \frac{x_i^2}{\lambda_i} = 1$, which corresponds to an axis-aligned ellipsoid.
- For Σ non-diagonal, we have $\Sigma = U \Lambda U^\mathsf{T} \implies \sum_{i=1}^d \frac{z_i^2}{\lambda_i} = 1$, where $z = U^\mathsf{T} x$. Thus, we also have an ellipsoid, but one aligned based on the eigenvectors of Σ.

Why the Bad Dependence on u?

- The truncation radius and, as a consequence, the intensity of the gaussian noise we added, were calibrated based on the largest eigenvalue (the "longest" principal direction of the ellipsoid).
- Thus, the "short" directions had a very small signal-to-noise ratio.
- Accounting for the loss along those directions leads to a blow-up in the sample complexity!

Intuition for the Solution

- What if, instead of going for a guarantee wrt the Mahalanobis norm, we focused on the spectral norm instead?
- The spectral norm only takes into account the error along the longest direction, so we don't have to worry about the effect of the noise on the other directions.
- Intuition: use this to obtain coarse estimates of the inverse of the covariance matrix and use them to rescale the data and perform *constant factor progress* in reducing the upper bound u on the condition number of Σ (*preconditioning*).

Preconditioning via Confidence Ellipsoids

- We use an approach inspired by [2] to perform the preconditioning step.
- The preconditioning process is:

Algorithm 2 One Step Heavy-Tailed Private Preconditioning via Confidence Ellipsoids

Input: $X = (X_1, \ldots, X_n) \sim \mathcal{D}^{\otimes n}$. Matrices $A, L > 0$ and $C_{2k} \geqslant 1, a, \rho_t, \beta_t > 0$.
Output: Matrices L' (lower bound), A' (symmetric) and M (noised covariance).

```
procedure OneStepHTPPCE_{A,L,C_{2k},a,ρ_t,β_t}(X)
    for i ∈ [n] do
        Let W_i = AX_i.                                   ▷ Σ_{W_i} = AΣA, L ⪯ Σ_{W_i} ⪯ 𝕀.
    end for
    Let γ = C_{2k}^{1/(2(k−1))} · √d / (a/2)^{1/(2(k−1))}.
    Let W = (W_1, ..., W_n).
    Let M = NaïveHTPCE_{ρ_t,γ,β_t}(W).
    Let S diag{λ_1, ..., λ_d} S^⊤ be the eigendecomposition of M.      ▷ SS^⊤ = 𝕀.
    Let η, ν be as defined in (4) and (5), respectively.
    Let s = a/2 + η + ν.                                  ▷ s : spectral error.
    Let I = [λ_min(L) + s, 1 − s].                        ▷ Assuming s ⩽ ½(1 − λ_min(L)).[4]
    for i ∈ [d] do
        Let λ'_i be the projection of λ_i into interval I.
    end for
    Let M_1 = S diag{λ'_1, ..., λ'_d} S^⊤.                ▷ (λ_min(L) + s)𝕀 ⪯ M_1 ⪯ (1 − s)𝕀.
    Let Ũ = M_1 + s𝕀.
    Let L' = Ũ^{−1/2}(M_1 − s𝕀)Ũ^{−1/2} and A' = Ũ^{−1/2}A.
    return L', A', M_1.
end procedure
```

Theorem 3.2. *For every $\rho_t > 0$ and every possible input, Algorithm 2 satisfies ρ_t-zCDP. Additionally, assume that $A, L \in \mathbb{R}^{d \times d}$ are symmetric PD matrices and $X = (X_1, \ldots, X_n) \sim \mathcal{D}^{\otimes n}$ with $\mathbb{E}_{X \sim \mathcal{D}}[X] = 0, L \preceq A\Sigma A \preceq \mathbb{I}$ that satisfies Definition 2.1 for some $1 \leqslant C_{2k} = \mathcal{O}(1)$ and $k \geqslant 2$. Then, if $\lambda_{\min}(L) < \frac{1}{4}$, a call to* $\textsc{OneStepHTPPCE}_{A,L,C_{2k},a,\rho_t,\beta_t}(X)$ *with $a = \frac{1}{16}$ and $n = \mathcal{O}\left(\frac{d \log d}{\beta_t^2} + \frac{d^{\frac{3}{2}} \log\left(\frac{1}{\beta_t}\right)}{\rho_t^{\frac{1}{2}}}\right)$ yields symmetric PD matrices A', L' such that $L' \preceq A'\Sigma A' \preceq \mathbb{I}$ and $\lambda_{\min}(L') \geqslant 2\lambda_{\min}(L)$ with probability at least $1 - \beta_t$.*

Intuition Behind the Preconditioning Step

- Assume we have a quantity χ we wish to estimate, such that $\frac{1}{u} \leqslant \chi \leqslant 1$ and we want rescale the data in a fashion that will narrow the range where the resulting value may lie by increasing the lower bound $\frac{1}{u}$ and maintaining 1 as the upper bound.
- Suppose we obtain an estimate x of χ such that $x - \epsilon \leqslant \chi \leqslant x + \epsilon$ where $\epsilon = \epsilon(n)$ that is a decreasing function of n with $\epsilon(n) \to 0$.
- We have $\frac{x-\epsilon}{x+\epsilon} \leqslant \frac{\chi}{x+\epsilon} \leqslant 1$.
- Observe that, if $\epsilon << x$, we have $\frac{x-\epsilon}{x+\epsilon} = \frac{1-\frac{\epsilon}{x}}{1+\frac{\epsilon}{x}} \approx 1$.
- Thus, if $\frac{1}{u}$ is not very close to 1, assuming we have a non-trivial lower bound on x ($x = \Omega(1)$ instead of just $x > 0$), we can pick n to be large enough so that $\frac{x-\epsilon}{x+\epsilon} \geqslant \frac{2}{u}$.
- The previous algorithm is the multidimensional analogue to this, where s plays the role of ϵ, M plays the role of x and the construction of M_1 ensures we have the aforementioned non-trivial lower bound.

The Overall Algorithm

Algorithm 3 Heavy Tailed Private Covariance Estimation

Input: $(X_1, \dots, X_n) \sim \mathcal{D}^{\otimes n}$, $u > 0$ with $\mathbb{I} \preceq \Sigma \preceq u\mathbb{I}$, $t \in \mathbb{N}^+, C_{2k} \geqslant 1, \rho_1, \dots, \rho_t, \delta > 0$.
Output: A $\left(\sum_{i=1}^{t} \rho_i\right)$-zCDP estimate $\hat{\Sigma}$ of Σ.

procedure $\text{HT\_PCE}_{u, C_{2k}, t, \rho_{1,\dots,t}, \delta}(X_{1,\dots,n})$
  Let $A_0 = \frac{1}{\sqrt{u}}\mathbb{I}, L_0 = \frac{1}{u}\mathbb{I}$.
  for $i \in [t-1]$ **do**
    $(A_i, L_i, M_i) = \text{ONE\_STEP\_HT\_PPCE}_{A_{i-1}, L_{i-1}, C_{2k}, \frac{1}{16}, \rho_{i-1}, \frac{\delta}{2(t-1)}}(X_{1,\dots,n})$.
  end for
  for $i \in [n]$ **do**
    $W_i = A_{t-1} X_i$.
  end for
  Let $\gamma = C_{2k}^{\frac{1}{2(k-1)}} \cdot \frac{d^{\frac{2k-1}{4(k-1)}}}{\left(\frac{\alpha}{2}\right)^{\frac{1}{2(k-1)}}}$.
  $M_t = \text{NAIVE\_HT\_PCE}_{\rho_t, \gamma, \frac{\delta}{2}}(W_{1,\dots,n})$.
  return $A_{t-1}^{-1} M_t A_{t-1}^{-1}$.
end procedure

- The final sample complexity is $\tilde{\mathcal{O}}\left(\frac{d^2}{\alpha^2} + \frac{d^{2+\frac{1}{2(k-1)}}}{\sqrt{\rho}\alpha^{\frac{k}{k-1}}} + \frac{d^{\frac{3}{2}} poly(\log u)}{\sqrt{\rho}}\right)$.
- The $2 + \frac{1}{2(k-1)}$ term in the exponent of d is because of the truncation radius we are forced to use to get dimension-independent bias error.

Covariance Estimation Under Approx DP

- Unlike pure DP and CDP, having a priori bounds on the parameters of the distribution is not necessary for approx DP estimation.
- To estimate the mean with known covariance (e.g., $\Sigma = \mathbb{I}$), it suffices to run the Karwa and Vadhan algorithm over each component.
- For unknown covariance, the problem is non-trivial, due to the need of identifying the principal components of the covariance matrix.

Estimating the Eigenvalues

- Our first step is to output estimates of the eigenvalues, without outputting estimates of the eigenvectors.
- This will help us identify multiplicative gaps between eigenvalues $\lambda_1 \geqslant \cdots \geqslant \lambda_d \geqslant 0$ in order to decide whether preconditioning is necessary.
- We use stability-based histograms, which do not satisfy CDP, but only approx DP.

Algorithm 1: Differentially Private EigenvalueEstimator$_{\varepsilon,\delta,\beta}(X)$

Input: Samples $X_1, \ldots, X_n \in \mathbb{R}^d$. Parameters $\varepsilon, \delta, \beta > 0$.
Output: Noisy eigenvalues of X: $(\hat{\lambda}_1, \ldots, \hat{\lambda}_d) \in \mathbb{R}^d$.

Set parameters: $t \leftarrow \frac{C_1 \log(d/\delta\beta)}{\varepsilon}$ $\quad m \leftarrow \lfloor n/t \rfloor$

Split X into t datasets of size m: $X^1, \ldots, X^t$.

```
// Estimate the eigenvalues via DP Histograms.
```

For $i \leftarrow 1, \ldots, d$
  For $j \leftarrow 1, \ldots, t$
    Let λ_i^j be the i-th eigenvalue of $\frac{1}{m} \cdot X^{j\top} X^j$.
  Divide $[0, \infty)$ into $\Omega \leftarrow \{\ldots, [1/\sqrt{2}, 1/2^{1/4})[1/2^{1/4}, 1)[1, 2^{1/4}), [2^{1/4}, \sqrt{2}), \ldots\} \cup \{[0,0]\}$.
  Run $\left(\frac{\varepsilon}{\sqrt{6d\log(1/\delta)}}, \frac{\delta}{d+1}\right)$-DP histogram on all λ_i^j over Ω.
  If *no bucket is returned*
    Return $\perp$.
  Let $[l, r]$ be a non-empty bucket returned.
  Set $\bar{\lambda}_i \leftarrow l$.

Sort $(\bar{\lambda}_1, \ldots, \bar{\lambda}_d)$ to get $\hat{\lambda}_1, \ldots, \hat{\lambda}_d$.

Return $(\hat{\lambda}_1, \ldots, \hat{\lambda}_d)$

Theorem 3.1. *For every $\varepsilon, \delta, \beta > 0$, there exists an (ε, δ)-DP algorithm, that takes*

$$n = O\left(\frac{d^{3/2} \cdot \operatorname{polylog}(d, 1/\delta, 1/\varepsilon, 1/\beta)}{\varepsilon}\right)$$

samples from $\mathcal{N}(0, \Sigma)$, for an arbitrary symmetric, positive-semidefinite $\Sigma \in \mathbb{R}^{d \times d}$, and outputs $\hat{\lambda}_1 \geq \cdots \geq \hat{\lambda}_d$, such that with probability at least $1 - O(\beta)$, $\hat{\lambda}_i \in \left[\frac{\lambda_i(\Sigma)}{\sqrt{2}}, \sqrt{2}\lambda_i(\Sigma)\right]$ for all i.

Preconditioning for Unbounded Gaussians

- For convenience, assume for the the time being that $\lambda_d > 0$.
- If there are no multiplicative gaps between eigenvalues (e.g., $\frac{\lambda_1}{\lambda_d} \leqslant 1000 = \mathcal{O}(1)$), preconditioning is not necessary at all.
- If there are gaps, we use an approach that involves iterating over all d eigenvalues that is inspired by dynamic programming.

- At the beginning of the k-th iteration, assume that $\frac{\lambda_1}{\lambda_k} = \mathcal{O}(1)$, but $\frac{\lambda_k}{\lambda_{k+1}}$ is large (we have no prior bound on how large).
- We need a preconditioner that will help us “close” the gap $\frac{\lambda_k}{\lambda_{k+1}}$, regardless of how large that may be.
- Truncate-and-noise doesn’t work!
- We use an algorithm that will help us identify the projection matrix of the subspace spanned by the eigenvectors corresponding to the eigenvalues $\lambda_1, \ldots, \lambda_k$.

The Subspace Algorithm

Algorithm 2: DP Subspace Estimator $\text{SubspaceRecovery}_{\varepsilon,\delta,\alpha,\gamma,k}(X)$

Input: Samples $X_1, \dots, X_n \in \mathbb{R}^d$. Parameters $\varepsilon, \delta, \alpha, \gamma, k > 0$.
Output: Projection matrix $\widehat{\Pi} \in \mathbb{R}^{d \times d}$ of rank k.

Set parameters: $t \leftarrow \frac{C_0\sqrt{dk}\cdot\text{polylog}(d,k,\frac{1}{\varepsilon},\frac{1}{\delta})}{\varepsilon}$ $\quad m \leftarrow \lfloor n/t \rfloor \quad q \leftarrow C_1 k$

$r \leftarrow \frac{C_2\gamma\sqrt{d}(\sqrt{k}+\sqrt{\ln(kt)})}{\sqrt{m}}$

Sample reference points $p_1, \dots, p_q$ from $\mathcal{N}(\vec{0}, \mathbb{I})$ independently.

`// Subsample from X, and form projection matrices.`

For $j \in 1, \dots, t$
  Let $X^j = (X_{(j-1)m+1}, \dots, X_{jm}) \in \mathbb{R}^{d \times m}$.
  Let $\Pi_j \in \mathbb{R}^{d \times d}$ be the projection matrix onto the subspace spanned by the eigenvectors of $X^j(X^j)^\top \in \mathbb{R}^{d \times d}$ corresponding to the largest k eigenvalues.
  For $i \in 1, \dots, q$
    $p_i^j \leftarrow \Pi_j p_i$

`// Aggregate using a ball-finding algorithm.`

For $i \in [q]$
  Let $P_i \in \mathbb{R}^{d \times t}$ be the dataset, where column j is p_i^j.
  Set $c_i \leftarrow \text{GoodCenter}_{\frac{\varepsilon}{\sqrt{q\ln(1/\delta)}},\frac{\delta}{q},r}(P_i)$.

Set $R \leftarrow C_3 r\sqrt{\log(t)}$

`// Return the subspace.`

Let $\sigma \leftarrow \frac{4R\sqrt{q}\ln(q/\delta)}{\varepsilon t}$.

For *each* $i \in [q]$
  Truncate all p_i^j's to within $B_R(c_i)$.
  Let $\widehat{p}_i \leftarrow \sum_{j=1}^{t} p_i^j + \mathcal{N}(0, \sigma^2\mathbb{I}_{d\times d})$.

Let $\widehat{P} \leftarrow (\widehat{p}_i, \dots, \widehat{p}_q)$.
Let $\widehat{\Pi}$ be the projection matrix of the top-k subspace of $\widehat{P}$.
Return $\widehat{\Pi}$.

- The algorithm requires $\tilde{\mathcal{O}}\left(d^{1.5}\right)$ samples independently of the gap $\frac{\lambda_k}{\lambda_{k+1}}$!

The Coarse Preconditioner

Algorithm 3: Differentially Private CoarsePreconditioner$_{\varepsilon,\delta,\beta,k,\hat{\gamma}}(X)$

Input: Samples $X_1, \ldots, X_n \in \mathbb{R}^d$. Parameters $\varepsilon, \delta, \beta, k > 0, \hat{\gamma} \geq 0$.
Output: Matrix $A \in \mathbb{R}^{d \times d}$.

Set $1 - \eta \leftarrow \hat{\gamma}$.
Set $\hat{\Pi}_{1:k} \leftarrow \text{SubspaceRecovery}_{\varepsilon,\delta,\beta,k,\hat{\gamma}}(X)$ and $\hat{\Pi}_{k+1:d} \leftarrow \mathbb{I} - \hat{\Pi}_{1:k}$.
Set $A \leftarrow (1-\eta)\hat{\Pi}_{1:k} + \hat{\Pi}_{k+1:d}$.

Return A.

Theorem 5.1 (Coarse Preconditioner). *Let $0 < \overline{\gamma} \leq 1$ and $0 < \hat{\gamma} < 1$ be arbitrary parameters. Then for all $\varepsilon, \delta, \beta > 0$ and*

$$n \geq O\left(\frac{d^2 \cdot \text{polylog}(d, \frac{1}{\varepsilon}, \frac{1}{\delta}, \frac{1}{\beta})}{\varepsilon \overline{\gamma}^4}\right),$$

there exists an (ε, δ)-DP algorithm, such that the following holds. Let $X = (X_1, \ldots, X_n)$ be i.i.d. samples from $\mathcal{N}(0, \Sigma)$, where, for some $1 \leq k < d$, $\frac{\lambda_k(\Sigma)}{\lambda_1(\Sigma)} \geq \overline{\gamma}^2$, and $\gamma^2 := \frac{\lambda_{k+1}(\Sigma)}{\lambda_k(\Sigma)} \in \left[\frac{\hat{\gamma}^2}{4}, 4\hat{\gamma}^2\right]$. Then with probability at least $1 - O(\beta)$, the algorithm takes X and $\hat{\gamma}$ as input, and outputs $A \in \mathbb{R}^{d \times d}$ that satisfies $\frac{\lambda_{k+1}(A\Sigma A)}{\lambda_1(A\Sigma A)} \geq \frac{\overline{\gamma}^2}{40}$.

Cumulative Gaps - Fine Preconditioning

- Assume now that, at the k-th iteration, we have that $\frac{\lambda_1}{\lambda_{k+1}}$ is large (e.g., larger than 1000) but we have upper bounds for $\frac{\lambda_1}{\lambda_k}$ and $\frac{\lambda_k}{\lambda_{k+1}}$.
- Then, we have an upper bound on $\frac{\lambda_1}{\lambda_{k+1}}$, so truncate-and-noise works!

Algorithm 4: Differentially Private FinePreconditioner$_{\varepsilon,\delta,\beta,k,\overline{\gamma},\kappa}(X)$

Input: Samples $X_1, \ldots, X_n \in \mathbb{R}^d$. Parameters $\varepsilon, \delta, \beta, k, \overline{\gamma}, \kappa > 0$.
Output: Matrix $A \in \mathbb{R}^{d \times d}$.

Set $Z \leftarrow$ NaiveEstimator$_{\varepsilon,\delta,\beta,\kappa}(X)$.
Let $S \leftarrow \{i : \lambda_i(Z) \geq \frac{\lambda_{k+1}(Z)}{16\overline{\gamma}^2}\}$.
Let $g_i \leftarrow \sqrt{\frac{\lambda_i(Z)}{\lambda_{k+1}(Z)}}$.
Let v_i be the i-th eigenvector of Z.
Set $\hat{\Pi}_S \leftarrow \sum_{i \in S} \frac{v_i v_i^\top}{4 g_i \overline{\gamma}}$ and $\hat{\Pi}_{\overline{S}} \leftarrow \sum_{i \notin S} v_i v_i^\top$.
Set $A \leftarrow \hat{\Pi}_S + \hat{\Pi}_{\overline{S}}$.
Return A.

Theorem 5.2 (Fine Preconditioner). *Let $X = (X_1, \ldots, X_n)$ be i.i.d. samples from $\mathcal{N}(0, \Sigma)$, such that for some $1 \leq k < d$, $\frac{\lambda_{k+1}(\Sigma)}{\lambda_1(\Sigma)} \geq \tau^2 \overline{\gamma}^2$ for $\overline{\gamma} \leq 1$. Then for all $\varepsilon, \delta > 0$, there exists an (ε, δ)-DP algorithm, such that if*

$$n \geq O\left(\frac{d^{3/2} \cdot \mathrm{polylog}(d, \frac{1}{\varepsilon}, \frac{1}{\delta}, \frac{1}{\beta})}{\varepsilon \tau^2 \overline{\gamma}^2}\right),$$

then with probability at least $1 - O(\beta)$, it takes X as input, and outputs a matrix A that satisfies $\frac{\lambda_{k+1}(A\Sigma A)}{\lambda_1(A\Sigma A)} \geq \overline{\gamma}^2$.

The Overall Algorithm

- Iterate over the eigenvalues and apply the coarse and fine preconditioners in succession whenever a gap of either type is identified.
- If we have a zero eigenvalue, first identify the subspace corresponding to the unknown covariance.
- The final sample complexity is $\tilde{\mathcal{O}}\left(\frac{d^2}{\alpha^2} + \frac{d^2}{\alpha\epsilon} + \frac{d^{2.5}}{\epsilon}\right)$.

Recap

- We gave an algorithm that performs covariance estimation for heavy-tailed data under CDP.
- We gave an algorithm that performs covariance estimation for Gaussian data under approx DP with no dependence on u.
- We omitted an algorithm that performs covariance estimation under pure DP for heavy-tailed data (but can discuss now, if time permits :)).

Directions for Future Work

- Private heavy tailed estimation with sub-gaussian rates.
- Make covariance estimation under pure DP computationally efficient.

Questions?

* Your question here. *

Thank You!

Ashtiani H. and Liaw C. *Private and polynomial time algorithms for learning Gaussians and beyond*.

Biswas S., Dong Y., Kamath G. and Ullman J. *CoinPress: Practical Private Mean and Covariance Estimation*.

Bun M. and Steinke T. *Concentrated differential privacy: Simplifications, extensions, and lower bounds*.

Dwork C, McSherry F., Nissim K. and Smith A. *Calibrating Noise to Sensitivity in Private Data Analysis*.

Hopkins S., Kamath G. and Majid M. *Efficient Mean Estimation with Pure Differential Privacy via a Sum-of-Squares Exponential Mechanism*.

- Kamath G., Li J., Singhal V. and Ullman J. *Privately Learning High Dimensional Distributions*.
- Kamath G. and Mouzakis A. *Private Covariance Estimation of Heavy-Tailed Distributions*.
- Kamath G., Mouzakis A., Singhal V., Steinke T. and Ullman, J. *A Private and Computationally-Efficient Estimator for Unbounded Gaussians*.
- Kamath G., Singhal V. and Ullman J. *Private Mean Estimation of Heavy-Tailed Distributions*.
- Kothari P., Manurangsi P. and Velingker A. *Private Robust Estimation by Stabilizing Convex Relaxations*.
- Liu X., Kong W. and Oh S. *Differential privacy and robust statistics in high dimensions*.

Karwa V. and Vadhan S. *Finite Sample Differentially Private Confidence Intervals*.