Robust and Differentially Private Mean Estimation

Xiyang Liu [1] **Weihao Kong** [1] **Sham Kakade** [1] **Sewoong Oh** [1]

Abstract

In statistical learning and analysis from shared data, which is increasingly widely adopted in platforms such as federated learning and meta-learning, there are two major concerns: privacy and robustness. Each participating individual should be able to contribute without the fear of leaking one's sensitive information. At the same time, the system should be robust in the presence of malicious participants inserting corrupted data. Recent algorithmic advances in learning from shared data focus on either one of these threats, leaving the system vulnerable to the other. We bridge this gap for the canonical problem of estimating the mean from i.i.d. samples. We introduce PRIME, which is the first efficient algorithm that achieves both privacy and robustness for a wide range of distributions. We further complement this result with a novel exponential time algorithm that improves the sample complexity of PRIME, achieving a near-optimal guarantee and matching a known lower bound for (non-robust) private mean estimation. This proves that there is no extra statistical cost to simultaneously guarantee privacy and robustness.

1. Introduction

When releasing database statistics on a collection of entries from individuals, we would ideally like to make it impossible to reverse-engineer each individual's potentially sensitive information. Privacy-preserving techniques add just enough randomness tailored to the statistical task to guarantee protection. At the same time, it is becoming increasingly common to apply such techniques to databases collected from multiple sources, not all of which can be trusted. Emerging data access frameworks, such as federated analyses across users' devices or data silos (Kairouz et al., 2019), make it easier to temper with such collected datasets, leaving private statistical analyses vulnerable to a malicious corruption of a fraction of the data.

Differential privacy has emerged as a widely accepted de facto measure of privacy, which is now a standard in releasing the statistics of the U.S. Census data (Abowd, 2018) statistics and also deployed in real-world commercial systems (Tang et al., 2017; Erlingsson et al., 2014; Fanti et al., 2016). A statistical analysis is said to be *differentially private* (DP) if the likelihood of the (randomized) outcome does not change significantly when a single arbitrary entry is added/removed (formally defined in §1.2). This provides a strong privacy guarantee: even a powerful adversary who knows all the other entries in the database cannot confidently identify whether a particular individual is participating in the database based on the outcome of the analysis. This ensures *plausible deniability*, central to protecting an individual's privacy.

In this paper, we focus on one of the most canonical problems in statistics: estimating the mean of a distribution from i.i.d. samples. For distributions with unbounded support, such as sub-Gaussian and heavy-tailed distributions, fundamental trade-offs between accuracy, sample size, and privacy have only recently been identified (Karwa & Vadhan, 2017; Kamath et al., 2019; 2020b; Aden-Ali et al., 2020) and efficient private estimators proposed. However, these approaches are brittle when a fraction of the data is corrupted, posing a real threat, referred to as *data poisoning* attacks (Chen et al., 2017; Xiao et al., 2015). In defense of such attacks, robust (but not necessarily private) statistics has emerged as a popular setting of recent algorithmic and mathematical breakthroughs (Steinhardt et al., 2018; Diakonikolas et al., 2017).

One might be misled into thinking that privacy ensures robustness since DP guarantees that a single outlier cannot change the estimation too much. This intuition is true only in a low dimension; each sample has to be an obvious outlier to significantly change the mean. However, in a high

[1]Paul G. Allen School of Computer Science and Engineering, University of Washington, Seattle, USA. Correspondence to: Xiyang Liu <xiyangl@cs.washington.edu>, Weihao Kong <whkong@cs.washington.edu>, Sham Kakade <sham@cs.washington.edu>, Sewoong Oh <sewoong@cs.washington.edu>.

This work was presented at the International Workshop on Federated Learning for User Privacy and Data Confidentiality in Conjunction with ICML 2021 (FL-ICML'21). This workshop does not have official proceedings and this paper is non-archival.

dimension, each corrupted data point can look perfectly uncorrupted but still shift the mean significant when colluding together (e.g., see Fig. 1). Focusing on the canonical problem of mean estimation, we introduce novel algorithms that achieve robustness and privacy simultaneously even when a fraction of data is corrupted arbitrarily. For such algorithms, there is a fundamental question of interest: do we need more samples to make private mean estimation also robust against adversarial corruption?

Sub-Gaussian distributions. If we can afford exponential run-time in the dimension, robustness can be achieved without extra cost in sample complexity. We introduce a novel estimator that (i) satisfies (ε, δ)-DP, (ii) achieves near-optimal robustness under α-fraction of corrupted data, achieving accuracy of $O(\alpha\sqrt{\log(1/\alpha)})$ nearly matching the fundamental lower bound of $\Omega(\alpha)$ that holds even for a (non-private) robust mean estimation with *infinite* samples, and (iii) achieves near-optimal sample complexity matching that of a fundamental lower bound for a (non-robust) private mean estimation as shown in Table 1.

Theorem 1 (Informal Theorem 9, exponential time)**.** *Algorithm 3 is (ε, δ)-DP. When α fraction of the data is arbitrarily corrupted from n samples from a d-dimensional sub-Gaussian distribution with mean μ and an identity sub-Gaussian parameter, if $n = \widetilde{\Omega}(d/\alpha^2 + (d + d^{1/2}\log(1/\delta))/(\alpha\varepsilon))$ then Algorithm 3 achieves $\|\hat{\mu} - \mu\|_2 = O(\alpha\sqrt{\log(1/\alpha)})$ w.h.p.*

We introduce PRIME (PRIvate and robust Mean Estimation) in §2.3 with details in Algorithm 9 in Appendix F.1, to achieve computational efficiency. It requires a run-time of only $\widetilde{O}(d^3 + nd^2)$, but at the cost of requiring extra $d^{1/2}$ factor larger number of samples. This cannot be improved upon with current techniques since efficient robust estimators rely on the top PCA directions of the covariance matrix to detect outliers. (Wei et al., 2016) showed that $\widetilde{\Omega}(d^{3/2})$ samples are necessary to compute PCA directions while preserving (ε, δ)-DP when $\|x_i\|_2 = O(\sqrt{d})$. It remains an open question if this $\widetilde{\Omega}(d^{3/2}/(\alpha\varepsilon))$ bottleneck is fundamental; no matching lower bound is currently known.

Theorem 2 (Informal Theorem 6, polynomial time)**.** *PRIME is (ε, δ)-DP and under the assumption of Thm.1 if $n = \widetilde{\Omega}(d/\alpha^2 + (d^{3/2}\log(1/\delta))/(\alpha\varepsilon))$ achieves $\|\hat{\mu} - \mu\|_2 = O(\alpha\sqrt{\log(1/\alpha)})$ w.h.p.*

Heavy-tailed distributions. When samples are drawn from a distribution with a bounded covariance, parameters of Algorithm 3 can be modified to nearly match the optimal sample complexity of (non-robust) private mean estimation in Table 2. This algorithm also matches the fundamental limit on the accuracy of (non-private) robust estimation, which in this case is $\Omega(\alpha^{1/2})$.

Theorem 3 (Informal Theorem 7, exponential time)**.** *From a distribution with mean $\mu \in \mathbb{R}^d$ and covariance $\Sigma \preceq \mathbf{I}$, n samples are drawn and α-fraction is corrupted. Algorithm 3 is (ε, δ)-DP and if $n = \widetilde{\Omega}((d + d^{1/2}\log(1/\delta))/(\alpha\varepsilon) + d^{1/2}\log^{3/2}(1/\delta)/\varepsilon)$ achieves $\|\hat{\mu} - \mu\|_2 = O(\alpha^{1/2})$ w.h.p.*

The proposed PRIME-HT for covariance bounded distributions achieve computational efficiency at the cost of an extra factor of $d^{1/2}$ in sample size. This bottleneck is also due to DP PCA, and it remains open whether this gap can be closed by an efficient estimator.

Theorem 4 (Informal Theorem 8, polynomial time)**.** PRIME-HT *is (ε, δ)-DP and if $n = \widetilde{\Omega}((d^{3/2}\log(1/\delta))/(\alpha\varepsilon))$ achieves $\|\hat{\mu} - \mu\|_2 = O(\alpha^{1/2})$ w.h.p. under the assumptions of Thm. 3.*

1.1. Technical contributions

We introduce PRIME which simultaneously achieves (ε, δ)-DP and robustness against α-fraction of corruption. A major challenge in making a standard filter-based robust estimation algorithm (e.g., (Diakonikolas et al., 2017)) private is the high sensitivity of the filtered set that we pass from one iteration to the next. We propose a new framework which makes private only the statistics of the set, hence significantly reducing the sensitivity. Our major innovation is a tight analysis of the end-to-end sensitivity of this multiple interactive accesses to the database. This is critical in achieving robustness while preserving privacy and is also of independent interest in making general iterative filtering algorithms private.

The classical filter approach (see, e.g. (Diakonikolas et al., 2017)) needs to access the database $O(d)$ times, which brings an extra $O(\sqrt{d})$ factor in the sample complexity due to DP composition. In order to reduce the iteration complexity, following the approach in (Dong et al., 2019), we propose filtering multiple directions simultaneously using a new score based on the matrix multiplicative weights (MMW). In order to privatize the MMW filter, our major innovation is a novel adaptive filtering algorithm DPTHRESHOLD$(\cdot)$ that outputs a *single private threshold* which guarantees sufficient progress at every iteration. This brings the number of database accesses from $O(d)$ to $O((\log d)^2)$.

One downside of PRIME is that it requires an extra $d^{1/2}$ factor in the sample complexity, compared to known lower bounds for (non-robust) DP mean estimation. To investigate whether this is also necessary, we propose a *sample optimal* exponential time robust mean estimation algorithm in §C and prove that there is no extra statistical cost to jointly requiring privacy and robustness. Our major technical innovations is in using *resilience property of the dataset* to not only find robust mean (which is the typical use case of resilience) but also bound sensitivity of that robust mean.

	Upper bound (poly-time)	Upper bound (exp-time)	Lower bound
(ε, δ)-DP (Kamath et al., 2019)	$\widetilde{O}(\frac{d}{\alpha^2} + \frac{d\log^{1/2}(1/\delta)}{\alpha\varepsilon})$	$\widetilde{O}(\frac{d}{\alpha^2} + \frac{d}{\alpha\varepsilon})^{\clubsuit}$	$\widetilde{\Omega}(\frac{d}{\alpha^2} + \frac{d}{\alpha\varepsilon})^{\spadesuit}$
α-corruption (Dong et al., 2019)	$\widetilde{O}(\frac{d}{\alpha^2})$	$\widetilde{O}(\frac{d}{\alpha^2})$	$\Omega(\frac{d}{\alpha^2})$
α-corruption and (ε, δ)-DP (this paper)	$\widetilde{O}(\frac{d}{\alpha^2} + \frac{d^{3/2}\log(1/\delta)}{\alpha\varepsilon})$ [Theorem 6]	$\widetilde{O}(\frac{d}{\alpha^2} + \frac{d+d^{1/2}\log(1/\delta)}{\alpha\varepsilon})$ [Theorem 9]	$\widetilde{\Omega}(\frac{d}{\alpha^2} + \frac{d}{\alpha\varepsilon})^{\spadesuit}$ (Kamath et al., 2019)

Table 1. For estimating the mean $\mu \in [-R, R]^d$ of a *sub-Gaussian* distribution with a known covariance, we list the sufficient or necessary conditions on the sample sizes to achieve an error $\|\hat{\mu} - \mu\|_2 = \widetilde{O}(\alpha)$ under (ε, δ)-DP, corruption of an α-fraction of samples, and both. $^{\clubsuit}$ requires the distribution to be a Gaussian (Bun et al., 2019) and $^{\spadesuit}$ requires $\delta \leq \sqrt{d}/n$.

	Upper bound (poly-time)	Upper bound (exp-time)	Lower bound
(ε, δ)-DP (Kamath et al., 2020b)	$\widetilde{O}(\frac{d\log^{1/2}(1/\delta)}{\alpha\varepsilon})$	$\widetilde{O}(\frac{d\log^{1/2}(1/\delta)}{\alpha\varepsilon})$	$\Omega(\frac{d}{\alpha\varepsilon})$
α-corruption (Dong et al., 2019)	$\widetilde{O}(\frac{d}{\alpha})$	$\widetilde{O}(\frac{d}{\alpha})$	$\Omega(\frac{d}{\alpha})$
α-corruption and (ε, δ)-DP (this paper)	$\widetilde{O}(\frac{d^{3/2}\log(1/\delta)}{\alpha\varepsilon})$ [Theorem 8]	$\widetilde{O}(\frac{d+d^{1/2}\log^{3/2}(1/\delta)}{\alpha\varepsilon})$ [Theorem 7]	$\Omega(\frac{d}{\alpha\varepsilon})$ ((Kamath et al., 2020b))

Table 2. For estimating the mean $\mu \in [-R, R]^d$ of a covariance bounded distribution, we list the sufficient or necessary conditions on the sample size to achieve an error $\|\hat{\mu} - \mu\|_2 = O(\alpha^{1/2})$ under (ε, δ)-DP, corruption of an α-fraction of samples, and both.

1.2. Preliminary on differential privacy (DP)

DP is a formal metric for measuring privacy leakage when a dataset is accessed with a query (Dwork et al., 2006).

Definition 1.1. *Given two datasets $S = \{x_i\}_{i=1}^n$ and $S' = \{x_i'\}_{i=1}^{n'}$, we say S and S' are* neighboring *if $d_\triangle(S, S') \leq 1$ where $d_\triangle(S, S') \triangleq \max\{|S \setminus S'|, |S' \setminus S|\}$, which is denoted by $S \sim S'$. For an output of a stochastic query q on a database, we say q satisfies* (ε, δ)-differential privacy *for some $\varepsilon > 0$ and $\delta \in (0, 1)$ if $\mathbb{P}(q(S) \in A) \leq e^\varepsilon \mathbb{P}(q(S') \in A) + \delta$ for all $S \sim S'$ and all subset A.*

Let $z \sim \mathrm{Lap}(b)$ be a random vector with entries i.i.d. sampled from Laplace distribution with pdf $(1/2b)e^{-|z|/b}$. Let $z \sim \mathcal{N}(\mu, \Sigma)$ denote a Gaussian random vector with mean μ and covariance Σ.

Definition 1.2. *The* sensitivity *of a query $f(S) \in \mathbb{R}^k$ is defined as $\Delta_p = \sup_{S \sim S'} \|f(S) - f(S')\|_p$ for a norm $\|x\|_p = (\sum_{i\in[k]} |x_i|^p)^{1/p}$. For $p = 1$, the Laplace mechanism outputs $f(S) + \mathrm{Lap}(\Delta_1/\varepsilon)$ and achieves $(\varepsilon, 0)$-DP (Dwork et al., 2006). For $p = 2$, the Gaussian mechanism outputs $f(S) + \mathcal{N}(0, (\Delta_2(\sqrt{2\log(1.25/\delta)})/\varepsilon)^2\mathbf{I})$ and achieves (ε, δ)-DP (Dwork & Roth, 2014).*

We use these output perturbation mechanisms along with the exponential mechanism (McSherry & Talwar, 2007) as building blocks. Appendix A provides detailed survey of privacy and robust estimation.

1.3. Problem formulation

We are given n samples from a sub-Gaussian distribution with a known covariance but unknown mean, and α fraction of the samples are corrupted by an adversary. Our goal is to estimate the unknown mean. We follow the standard definition of adversary in (Diakonikolas et al., 2017), which can adaptively choose which samples to corrupt and arbitrarily replace them with any points.

Assumption 1. *An uncorrupted dataset S_{good} consists of n i.i.d. samples from a d-dimensional sub-Gaussian distribution with mean $\mu \in [-R, R]^d$ and covariance $\mathbb{E}[xx^\top] = \mathbf{I}_d$, which is 1-sub-Gaussian, i.e., $\mathbb{E}[\exp(v^\top x)] \leq \exp(\|v\|_2^2/2)$. For some $\alpha \in (0, 1/2)$, we are given a corrupted dataset $S = \{x_i \in \mathbb{R}^d\}_{i=1}^n$ where an adversary adaptively inspects all the samples in S_{good}, removes αn of them, and replaces them with S_{bad} which are αn arbitrary points in $\mathbb{R}^d$.*

Similarly, we consider the same problem for heavy-tailed distributions with a bounded covariance. We present the assumption and main results for covariance bounded distributions in Appendix B.

Outline. We present PRIME for sub-Gaussian distribution in §2, and present theoretical analysis in §3. We then introduce an exponential time algorithm with near optimal guarantee in §C (due to space constraints). Analogous results for heavy-tailed distributions are presented in Appendix B (due to space constraints).

2. PRIME: efficient algorithm for robust and DP mean estimation

In order to describe the proposed algorithm PRIME, we need to first describe a standard (non-private) iterative filtering algorithm for robust mean estimation.

2.1. Background on (non-private) iterative filtering for robust mean estimation

Non-private robust mean estimation approaches recursively apply the following *filter*, whose framework is first proposed in (Diakonikolas et al., 2019a). Given a dataset $S = \{x_i\}_{i=1}^n$, the current set $S_0 \subseteq [n]$ of data points is updated starting with $S_1 = [n]$. At each step, the following filter (Algorithm 1 in (Li, 2019)) attempts to detect the corrupted data points and remove them.

1. Compute the top eigenvector $v_t \leftarrow \arg\max_{v:\|v\|_2=1} v^\top \mathrm{Cov}(S_{t-1})v$ of the covariance of the current data set $\{x_i\}_{i\in S_{t-1}}$;
2. Compute scores for all data points $j \in S_{t-1}$: $\tau_j \leftarrow \left(v_t^\top (x_j - \mathrm{Mean}(S_{t-1}))\right)^2$;
3. Draw a random threshold: $Z_t \leftarrow \mathrm{Unif}([0,1])$;
4. Remove outliers from S_{t-1} defined as $\{i \in S_{t-1} : \tau_i \text{ is in the largest } 2\alpha\text{-tail of } \{\tau_j\}_{j\in S_{t-1}} \text{ and } \tau_i \geq Z_t\, \tau_{\max}\}$, where $\tau_{\max} = \max_{j\in S_{t-1}} \tau_j$

This is repeated until the empirical covariance is sufficiently small and the empirical mean $\hat{\mu}$ is output. At a high level, the correctness of this algorithm relies on the key observation that the α-fraction of adversarial corruption can not significantly change the mean of the dataset without introducing large eigenvalues in the empirical covariance. Therefore, the algorithm finds top eigenvector of the empirical covariance in step 1, and tries to correct the empirical covariance by removing corrupted data points. Each data point is assigned a score in step 2 which indicates the "badness" of the data points, and a threshold Z_t in step 3 is carefully designed such that step 4 guarantees to remove more corrupted data points than good data points (in expectation). This guarantees the following bound achieving the near-optimal sample complexity shown in the second row of Table 1. A formal description of this algorithm is in Algorithm 4 in Appendix D.

Proposition 2.1 (Corollary of (Li, 2019, Theorem 2.1))**.** *Under assumption 1, the above filtering algorithm achieves accuracy* $\|\hat{\mu} - \mu\|_2 \leq O(\alpha\sqrt{\log(1/\alpha)})$ *w.p.* 0.9 *if* $n \geq \widetilde{\Omega}(d/\alpha^2)$.

Challenges in making robust mean estimation private. To get a DP and robust mean, a naive attempt is to apply a standard output perturbation mechanism to $\hat{\mu}$. However, this is obviously challenging since the end-to-end sensitivity is intractable. The standard recipe to circumvent this is to make the current "state" S_t private at every iteration. Once S_{t-1} is private (hence, public knowledge), making the next "state" S_t private is simpler. We only need to analyze the sensitivity of a single step and apply some output perturbation mechanism with $(\varepsilon_t, \delta_t)$. End-to-end privacy is guaranteed by accounting for all these $(\varepsilon_t, \delta_t)$'s using the advanced composition (Kairouz et al., 2015). This recipe has been quite successful, for example, in training neural networks with (stochastic) gradient descent (Abadi et al., 2016), where the current state can be the optimization variable $\mathbf{x}_t$. However, for the above (non-private) filtering algorithm, this standard recipe fails, since the state S_t is a set and has large sensitivity. Changing a single data point in S_t can significantly alter which (and how many) samples are filtered out.

2.2. A new framework for *private* iterative filtering

Instead of making the (highly sensitive) S_t itself private, we propose a new framework which makes private only the statistics of S_t: the mean μ_t and the top principal direction v_t. There are two versions of this algorithm, which output the exactly same $\hat{\mu}$ with the exactly same privacy guarantees, but are written from two different perspectives. We present here the *interactive* version from the perspective of an analyst accessing the dataset via DP queries ($q_{\mathrm{range}}, q_{\mathrm{size}}, q_{\mathrm{mean}}, q_{\mathrm{norm}}$ and q_{PCA}), because this version makes clear the inner operations of each private mechanisms, hence making (i) the sensitivity analysis transparent, (ii) checking the correctness of privacy guarantees easy, and (iii) tracking privacy accountant simple. In practice, one should implement the *centralized* version (Algorithm 7 in Appendix E), which is significantly more efficient.

Algorithm 1: Private iterative filtering (interactive version)

Input: range $[-R, R]^d$, $\alpha \in (0, 1/2)$, probability $\zeta \in (0,1)$, # of iterations $T = \Theta(d)$, (ε, δ)

$(\bar{x}, B) \leftarrow q_{\mathrm{range}}(R, 0.01\varepsilon, 0.01\delta)$

$\varepsilon_1 \leftarrow \min\{0.99\varepsilon, 0.9\}/(4\sqrt{2T\log(2/\delta)})$, $\delta_1 \leftarrow 0.99\delta/(8T)$

if $n < (4/\varepsilon_1)\log(1/(2\delta_1))$ **then Output:** $\emptyset$

for $t = 1, \ldots, T$ **do**

  $n_t \leftarrow q_{\mathrm{size}}(\{(\mu_\ell, v_\ell, Z_\ell)\}_{\ell\in[t-1]}, \varepsilon_1, \bar{x}, B)$, **if** $n_t < 3n/4$ **then Output:** $\emptyset$

  $\mu_t \leftarrow q_{\mathrm{mean}}(\{(\mu_\ell, v_\ell, Z_\ell)\}_{\ell\in[t-1]}, \varepsilon_1, \bar{x}, B)$

  $\lambda_t \leftarrow q_{\mathrm{norm}}(\{(\mu_\ell, v_\ell, Z_\ell)\}_{\ell\in[t-1]}, \mu_t, \varepsilon_1, \bar{x}, B)$

  if $\lambda_t \leq (C - 0.01)\alpha \log 1/\alpha$ **then Output:** μ_t

  $v_t \leftarrow q_{\mathrm{PCA}}(\{(\mu_\ell, v_\ell, Z_\ell)\}_{\ell\in[t-1]}, \mu_t, \varepsilon_1, \delta_1, \bar{x}, B))$

  $Z_t \leftarrow \mathrm{Unif}([0,1])$

Output: μ_t

We give a high-level explanation of each step of Algorithm 1 here and give the formal definitions of all the queries in Appendix E. First, q_{range} returns (the parameters of) a hypercube $\bar{x} + [-B/2, B/2]^d$ that is guaranteed to include all uncorrupted samples while preserving privacy. This is achieved by running d coordinate-wise private histograms and selecting $\bar{x}_j$ as the center of the

largest bin for the j-th coordinate. Since covariance is $\mathbf{I}$, q_{range} returns a fixed $B = 8\sigma\sqrt{\log(dn/\zeta)}$. Such an adaptive estimate of the support is critical in tightly bounding the sensitivity of all subsequent queries, which operate on the clipped dataset; all data points are projected as $\mathcal{P}_{\bar{x}+[-B/2,B/2]^d}(x) = \arg\min_{y\in\bar{x}+[-B/2,B/2]^d} \|y - x\|_2$ in all the queries that follow. With clipping, a single data point can now change at most by $B\sqrt{d}$.

The subsequent steps perform the non-private filtering algorithm of §2.1, but with private statistics μ_t and v_t. As the set S_t changes over time, we lower bound its size (which we choose to be $|S_t| > n/2$) to upper bound the sensitivity of other queries $q_{\text{mean}}, q_{\text{norm}}$ and q_{PCA}.

At the t-th iterations, every time a query is called the data curator (i) uses $(\bar{x}, B)$ to clip the data, (ii) computes S_t by running $t-1$ steps of the non-private filtering algorithm of §2.1 but with a given *fixed* set of parameters $\{(\mu_\ell, v_\ell)\}_{\ell\in[t-1]}$ (and the given randomness $\{Z_\ell\}_{\ell\in[t-1]}$), and (iii) computes the queried private statistics of S_t. If the private spectral norm of the covariance of S_t (i.e., λ_t) is sufficiently small, we output the private and robust mean $\hat{\mu} = \mu_t$ (line 8). Otherwise, we compute the private top PCA direction v_t and draw an randomness Z_t to be used in the next step of filtering, as in the non-private filtering algorithm. We emphasize that $\{S_\ell\}$ are not private, and hence never returned to the analyst. We also note that this interactive version is redundant as every query is re-computing S_t. In our setting, the analyst has the dataset and there is no need to separate them. This leads to a *centralized* version we provide in Algorithm 7 in the appendix, which avoids redundant computations and hence is significantly more efficient.

The main challenge in this framework is the privacy analysis. Because $\{S_\ell\}_{\ell\in[t-1]}$ is not private, each query runs $t-1$ steps of filtering whose end-to-end sensitivity could blow-up. Algorithmically, (i) we start with a specific choice of a non-private iterative filtering algorithm (among several variations that are equivalent in non-private setting but widely differ in its sensitivity), and (ii) make appropriate changes in the private queries (Algorithm 1) to keep the sensitivity small. Analytically, the following key technical lemma allows a sharp analysis of the end-to-end sensitivity of iterative filtering.

Lemma 2.2. *Let $S_t(\mathcal{S})$ denote the resulting subset of samples after t iterations of the filtering in the queries (q_{size}, q_{mean}, q_{norm}, and q_{PCA}) are applied to a dataset $\mathcal{S}$ using* fixed *parameters $\{(\mu_\ell, v_\ell, Z_\ell)\}_{\ell=1}^t$. Then, we have $d_\triangle(S_t(\mathcal{S}), S_t(\mathcal{S}')) \le d_\triangle(\mathcal{S}, \mathcal{S}')$, where $d_\triangle(\mathcal{S}, \mathcal{S}') \triangleq \max\{|\mathcal{S} \setminus \mathcal{S}'|, |\mathcal{S}' \setminus \mathcal{S}|\}$.*

Recall that two datasets are neighboring, i.e., $\mathcal{S} \sim \mathcal{S}'$, iff $d_\triangle(\mathcal{S}, \mathcal{S}') \le 1$. This lemma implies that if two datasets are neighboring, then they are still neighboring after filtering with the same parameters, no matter how many times we filter them. Hence, this lemma allows us to use the standard output-perturbation mechanisms with $(\varepsilon_1, \delta_1)$-DP. Advanced composition ensures that end-to-end guarantee of $4T$ such queries is $(0.99\varepsilon, 0.99\delta)$-DP. Together with $(0.01\varepsilon, 0.01\delta)$-DP budget used in q_{range}, this satisfied the target privacy. Analyzing the utility of this algorithm, we get the following guarantee.

Theorem 5. *Algorithm 1 is (ε, δ)-DP. Under Assumption 1, there exists a universal constant $c \in (0, 0.1)$ such that if $\alpha \le c$ and $n = \widetilde{\Omega}\left((d/\alpha^2) + d^2(\log(1/\delta))^{3/2}/(\varepsilon\alpha)\right)$ then Algorithm 1 achieves $\|\hat{\mu} - \mu\|_2 \le O(\alpha\sqrt{\log(1/\alpha)})$ with probability 0.9.*

The first term $O(d/\alpha^2)$ in the sample complexity is optimal (cf. Table 1), but there is a factor of d gap in the second term. This is due to the fact that we need to run $O(d)$ iterations in the worst-case. Such numerous accesses to the database result in large noise to be added at each iteration, requiring large sample size to combat that extra noise. We introduce PRIME to reduce the number of iterations to $O((\log d)^2)$ and significantly reduce the sample complexity.

2.3. PRIME: novel robust and private mean estimator

Algorithm 1 (specifically Filter$(\cdot)$ in Algorithm 1) accesses the database $O(d)$ times. This is necessary for two reasons. First, the filter checks only one direction v_t at each iteration. In the worst case, the corrupted samples can be scattered in $\Omega(d)$ orthogonal directions such that the filter needs to be repeated $O(d)$ times. Secondly, even if the corrupted samples are clustered together in one direction, the filter still needs to be repeated $O(d)$ times. This is because we had to use a large (random) threshold of $dB^2 Z_t = O(d)$ to make the threshold data-independent so that we can keep the sensitivity of Filter$(\cdot)$ low, which results in slow progress. We propose filtering multiple directions simultaneously using a new score $\{\tau_i\}$ based on the matrix multiplicative weights. Central to this approach is a novel adaptive filtering algorithm DPTHRESHOLD$(\cdot)$ that guarantees sufficient decrease in the total score at every iteration.

2.3.1. Matrix Multiplicative Weight (MMW) scoring

The MMW-based approach, pioneered in (Dong et al., 2019) for non-private robust mean estimation, filters out multiple directions simultaneously. It runs over $O(\log d)$ epochs and every epoch consists of $O(\log d)$ iterations. At every epoch s and iteration t, step 2 of the iterative filtering in §2.1 is replaced by a new score $\tau_i = (x_i - \text{Mean}(S_t^{(s)}))^T U_t^{(s)} (x_i - \text{Mean}(S_t^{(s)}))$ where $U_t^{(s)}$ now accounts for all directions in $\mathbb{R}^d$ but appropriately weighted. Precisely, it is defined via

the matrix multiplicative update:

$$U_t^{(s)} = \frac{\exp\left(\alpha^{(s)} \sum_{r\in[t]} (\mathrm{Cov}(S_t^{(s)}) - \mathbf{I})\right)}{\mathrm{Tr}\left(\exp(\alpha^{(s)} \sum_{r\in[t]} (\mathrm{Cov}(S_t^{(s)}) - \mathbf{I}))\right)},$$

for some choice of $\alpha^{(s)} > 0$. If we set the number of iterations to one, a choice of $\alpha^{(s)} = \infty$ recovers the previous score that relied on the top singular vector from §2.1 and a choice of $\alpha^{(s)} = 0$ gives a simple norm based score $\tau_i = \|x_i\|_2^2$. An appropriate choice of $\alpha^{(s)}$ smoothly interpolates between these two extremes, which ensures that $O(\log d)$ iterations are sufficient for the spectral norm of the covariance to decrease strictly by a constant factor. This guarantees that after $O(\log d)$ epochs, we sufficiently decrease the covariance to ensure that the empirical mean is accurate enough. Critical in achieving this gain is our carefully designed filtering algorithm DPTHRESHOLD that uses the privately computed MMW-based scores using Gaussian mechanism on the covariance matrices as shown in Algorithm 11 in Appendix F.

2.3.2. Adaptive filtering with DPThreshold

Novelty. The corresponding non-private filtering of (Dong et al., 2019) for robust mean estimation takes advantage of an *adaptive threshold*, but filters out each sample independently resulting in a prohibitively large sensitivity; the coupling between each sample and the randomness used to filter it can change widely between two neighboring datasets. On the other hand, Algorithm 1 (i.e., Filter(·) in Algorithm 6) takes advantage of jointly filtering all points above a *single threshold* $B^2 dZ_t$ with a single randomness $Z_t \sim \mathrm{Unif}[0,1]$, but the non-adaptive (and hence large) choice of the range $B^2 d$ results in a large number of iterations because each filtering only decrease the score by little. To sufficiently reduce the total score while maintaining a small sensitivity, we introduce a filter with a single and adaptive threshold.

Algorithm. Our goal here is to privately find a single scalar ρ such that when a randomized filter is applied on the scores $\{\tau_i\}$ with a (random) threshold ρZ (with Z drawn uniform in $[0,1]$), we filter out enough samples to make progress in each iteration while ensuring that we do not remove too many uncorrupted samples. This is a slight generalization of the non-private algorithm in Section 2.1, which simply set $\rho = \max_{j\in S_t} \tau_j$. While this guarantees the filter removes more corrupted samples than good samples, it does not make sufficient progress in reducing the total score of the samples.

Ideally, we want the thresholding to decrease the total score by a constant multiplicative factor, which will in the end allow the algorithm to terminate within logarithmic iterations. To this end, we propose a new scheme of using the largest ρ such that the following inequality holds:

$$\sum_{\tau_i > \rho} (\tau_i - \rho) \geq 0.31 \sum_{\tau_i \in S_t} (\tau_i - 1)\,. \tag{1}$$

We use a private histogram of the scores to approximate this threshold. Similar to (Kaplan et al., 2020; Karwa & Vadhan, 2017), we use geometrically increasing bin sizes such that we use only $O(\log B^2 d)$ bins while achieving a preferred *multiplicative* error in our quantization. At each epoch s and iteration t, we run DPTHRESHOLD sketched in the following to approximate ρ followed by a random filter. Step 3 replaces the non-private condition in Eq. (1). A complete description is provided in Algorithm 11.

1. Privately compute scores for all data points $i \in S_t^{(s)}$: $\tau_i \leftarrow (x_i - \mu_t)^\top U_t^{(s)} (x_i - \mu_t)$;
2. Compute a private histogram $\{\tilde{h}_j\}_{j=1}^{2+\log(B^2 d)}$ of the scores over geometrically sized bins $I_1 = [1/4, 1/2)$, $I_2 = [1/2, 1), \ldots, I_{2+\log(B^2 d)} = [2^{\log(B^2 d)-1}, 2^{\log(B^2 d)}]$;
3. Privately find the largest ℓ satisfying $\sum_{j\geq \ell} (2^j - 2^\ell)\, \tilde{h}_j \geq 0.31 \sum_{i\in S_t^{(s)}} (\tau_i - 1)$;
4. Output $\rho = 2^\ell$.

3. Analyses of PRIME

Building on the framework of Algorithm 1, PRIME (Algorithm 9) replaces the score with the MMW-based score presented in §2.3.1 and the filter with the adaptive DPTHRESHOLD. This reduces the number of iterations to $T = O((\log d)^2)$ achieving the following bound.

Theorem 6. *PRIME is (ε, δ)-differentially private. Under Assumption 1 there exists a universal constant $c \in (0, 0.1)$ such that if $\alpha \leq c$ and $n = \widetilde{\Omega}((d/\alpha^2) + (d^{3/2}/(\varepsilon\alpha)) \log(1/\delta))$, then PRIME achieves $\|\hat{\mu} - \mu\|_2 = O(\alpha\sqrt{\log(1/\alpha)})$ with probability 0.9.*

A proof is provided in Appendix G. The notation $\widetilde{\Omega}(\cdot)$ hides logarithmic terms in d, R, and $1/\alpha$. To achieve an error of $O(\alpha\sqrt{\log(1/\alpha)})$, the first term $\widetilde{\Omega}(d/\alpha^2 \log(1/\alpha))$ is necessary even if there is no corruption. The accuracy of $O(\alpha\sqrt{\log(1/\alpha)})$ matches the lower bound shown in (Diakonikolas et al., 2017) for any polynomial time statistical query algorithm, and it nearly matches the information theoretical lower bound on robust estimation of $\Omega(\alpha)$. On the other hand, the second term of $\widetilde{\Omega}(d^{3/2}/(\varepsilon\alpha \log(1/\alpha)))$ has an extra factor of $d^{1/2}$ compared to the optimal one achieved by exponential time Algorithm 3. It is an open question if this gap can be closed by a polynomial time algorithm.

The bottleneck is the private matrix multiplicative weights. Such spectral analyses are crucial in filter-based robust estimators. Even for a special case of privately computing the

top principal component, the best polynomial time algorithm requires $O(d^{3/2})$ samples (Dwork et al., 2014; Chaudhuri et al., 2013; Wei et al., 2016), and this sample complexity is also necessary as shown in Dwork et al. (2014, Corollary 25).

To boost the success probability to $1-\zeta$ for some small $\zeta > 0$, we need an extra $\log(1/\zeta)$ factor in the sample complexity to make sure the dataset satisfies the regularity condition with probability $\zeta/2$. Then we can run PRIME $\log(1/\zeta)$ times and choose the output of a run that satisfies $n^{(s)} > n(1-10\alpha)$ and $\lambda^{(s)} \le C\alpha\log(1/\alpha)$ at termination.

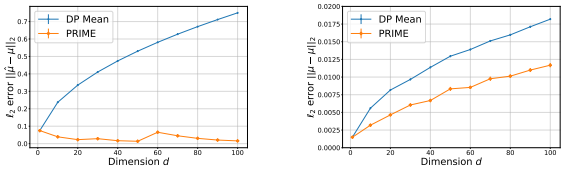

Figure 1. Private mean estimators (e.g., DP mean (Kamath et al., 2019)) are vulnerable to adversarial corruption especially in high dimensions, while the proposed PRIME achieves robustness (and privacy) regardless of the dimension of the samples.

Numerical experiments support our theoretical claims. The left figure with $(\alpha, \varepsilon, \delta, n) = (0.05, 20, 0.01, 10^6)$ is in the large α regime where the DP Mean error is dominates by $\alpha\sqrt{d}$ and PRIME error by $\alpha\sqrt{\log(1/\alpha)}$. Hence, PRIME error is constant whereas DP Mean error increases with the dimension d. The second figure with $(\alpha, \varepsilon, \delta, n) = (0.001, 20, 0.01, 10^6)$ is in the small α regime when DP Mean error consists of $\alpha\sqrt{d} + \sqrt{d/n}$ and PRIME is dominated by $\sqrt{d/n}$. Both increase with the dimension d, and the gap can be made large by increasing α. Details of the experiments are in Appendix K.

4. Conclusion

Differentially private mean estimation is brittle against a small fraction of the samples being corrupted by an adversary. We show that robustness can be achieves without any increase in the sample complexity by introducing a novel DP mean estimator, which requires run-time exponential in the dimension of the samples. The technical contribution is in leveraging the resilience property of well-behaved distributions in an innovative way to not only find robust mean (which is the typical use case of resilience) but also bound sensitivity for optimal privacy guarantee. To cope with the computational challenge, we propose an efficient algorithm, which we call PRIME, that achieves the optimal target accuracy at the cost of an increased sample complexity. The technical contributions are (i) a novel framework for private iterative filtering and its tight analysis of the end-to-end sensitivity and (ii) novel filtering algorithm of DPTHRESHOLD which is critical in privately running matrix multiplicative weights and hence significantly reducing the number of accesses to the database. With appropriately chosen parameters, we show that our exponential time approach achieves near-optimal guarantees for both sub-Gaussian and covariance bounded distributions and PRIME achieves the same accuracy efficiently but at the cost of an increased sample complexity by a $d^{1/2}$ factor.

References

Abadi, M., Chu, A., Goodfellow, I., McMahan, H. B., Mironov, I., Talwar, K., and Zhang, L. Deep learning with differential privacy. In *Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security*, pp. 308–318, 2016.

Abowd, J. M. The us census bureau adopts differential privacy. In *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pp. 2867–2867, 2018.

Aden-Ali, I., Ashtiani, H., and Kamath, G. On the sample complexity of privately learning unbounded high-dimensional gaussians. *arXiv preprint arXiv:2010.09929*, 2020.

Anscombe, F. J. Rejection of outliers. *Technometrics*, 2(2): 123–146, 1960.

Bakshi, A. and Kothari, P. List-decodable subspace recovery via sum-of-squares. *arXiv preprint arXiv:2002.05139*, 2020.

Balakrishnan, S., Du, S. S., Li, J., and Singh, A. Computationally efficient robust sparse estimation in high dimensions. In *Proceedings of the 30th Conference on Learning Theory, COLT 2017*, pp. 169–212, 2017.

Bhatia, K., Jain, P., and Kar, P. Robust regression via hard thresholding. In *Advances in Neural Information Processing Systems*, pp. 721–729, 2015.

Bhatia, K., Jain, P., Kamalaruban, P., and Kar, P. Consistent robust regression. In *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017*, pp. 2107–2116, 2017.

Biswas, S., Dong, Y., Kamath, G., and Ullman, J. Coinpress: Practical private mean and covariance estimation. *arXiv preprint arXiv:2006.06618*, 2020.

Blum, A., Dwork, C., McSherry, F., and Nissim, K. Practical privacy: the sulq framework. In *Proceedings of the twenty-fourth ACM SIGMOD-SIGACT-SIGART symposium on Principles of database systems*, pp. 128–138, 2005.

Bun, M., Nissim, K., and Stemmer, U. Simultaneous private learning of multiple concepts. In *ITCS*, pp. 369–380, 2016.

Bun, M., Kamath, G., Steinke, T., and Wu, S. Z. Private hypothesis selection. In *Advances in Neural Information Processing Systems*, pp. 156–167, 2019.

Cai, T. T., Wang, Y., and Zhang, L. The cost of privacy: Optimal rates of convergence for parameter estimation with differential privacy. *arXiv preprint arXiv:1902.04495*, 2019.

Canonne, C. L., Kamath, G., McMillan, A., Ullman, J., and Zakynthinou, L. Private identity testing for high-dimensional distributions. *arXiv preprint arXiv:1905.11947*, 2019.

Charikar, M., Steinhardt, J., and Valiant, G. Learning from untrusted data. In *Proceedings of the 49th Annual ACM SIGACT Symposium on Theory of Computing*, pp. 47–60, 2017.

Chaudhuri, K., Sarwate, A. D., and Sinha, K. A near-optimal algorithm for differentially-private principal components. *The Journal of Machine Learning Research*, 14(1):2905–2943, 2013.

Chen, X., Liu, C., Li, B., Lu, K., and Song, D. Targeted backdoor attacks on deep learning systems using data poisoning. *arXiv preprint arXiv:1712.05526*, 2017.

Cheng, Y., Diakonikolas, I., and Ge, R. High-dimensional robust mean estimation in nearly-linear time. In *Proceedings of the Thirtieth Annual ACM-SIAM Symposium on Discrete Algorithms*, pp. 2755–2771. SIAM, 2019a.

Cheng, Y., Diakonikolas, I., Ge, R., and Woodruff, D. P. Faster algorithms for high-dimensional robust covariance estimation. In *Conference on Learning Theory*, pp. 727–757. PMLR, 2019b.

Cherapanamjeri, Y., Mohanty, S., and Yau, M. List decodable mean estimation in nearly linear time. *arXiv preprint arXiv:2005.09796*, 2020.

Dalalyan, A. and Thompson, P. Outlier-robust estimation of a sparse linear model using ℓ_1-penalized huber's m-estimator. In *Advances in Neural Information Processing Systems*, pp. 13188–13198, 2019.

Depersin, J. and Lecué, G. Robust subgaussian estimation of a mean vector in nearly linear time. *arXiv preprint arXiv:1906.03058*, 2019.

Dhar, A. and Huang, J. Designing differentially private estimators in high dimensions. *arXiv preprint arXiv:2006.01944*, 2020.

Diakonikolas, I. and Kane, D. M. Recent advances in algorithmic high-dimensional robust statistics. *arXiv preprint arXiv:1911.05911*, 2019.

Diakonikolas, I., Kamath, G., Kane, D. M., Li, J., Moitra, A., and Stewart, A. Being Robust (in High Dimensions) Can Be Practical. *arXiv e-prints*, art. arXiv:1703.00893, March 2017.

Diakonikolas, I., Kane, D. M., and Stewart, A. Statistical query lower bounds for robust estimation of high-dimensional gaussians and gaussian mixtures. In *2017 IEEE 58th Annual Symposium on Foundations of Computer Science (FOCS)*, pp. 73–84. IEEE, 2017.

Diakonikolas, I., Kamath, G., Kane, D. M., Li, J., Moitra, A., and Stewart, A. Robustly learning a gaussian: Getting optimal error, efficiently. In *Proceedings of the Twenty-Ninth Annual ACM-SIAM Symposium on Discrete Algorithms*, pp. 2683–2702. SIAM, 2018a.

Diakonikolas, I., Kane, D. M., and Stewart, A. List-decodable robust mean estimation and learning mixtures of spherical gaussians. In *Proceedings of the 50th Annual ACM SIGACT Symposium on Theory of Computing*, pp. 1047–1060, 2018b.

Diakonikolas, I., Kamath, G., Kane, D., Li, J., Moitra, A., and Stewart, A. Robust estimators in high-dimensions without the computational intractability. *SIAM Journal on Computing*, 48(2):742–864, 2019a.

Diakonikolas, I., Kamath, G., Kane, D., Li, J., Steinhardt, J., and Stewart, A. Sever: A robust meta-algorithm for stochastic optimization. In *International Conference on Machine Learning*, pp. 1596–1606, 2019b.

Diakonikolas, I., Kong, W., and Stewart, A. Efficient algorithms and lower bounds for robust linear regression. In *Proceedings of the Thirtieth Annual ACM-SIAM Symposium on Discrete Algorithms*, pp. 2745–2754. SIAM, 2019c.

Diakonikolas, I., Hopkins, S. B., Kane, D., and Karmalkar, S. Robustly learning any clusterable mixture of gaussians. *arXiv preprint arXiv:2005.06417*, 2020.

Dong, Y., Hopkins, S., and Li, J. Quantum entropy scoring for fast robust mean estimation and improved outlier detection. In *Advances in Neural Information Processing Systems*, pp. 6067–6077, 2019.

Dwork, C. and Roth, A. The algorithmic foundations of differential privacy. *Foundations and Trends in Theoretical Computer Science*, 9(3-4):211–407, 2014.

Dwork, C., McSherry, F., Nissim, K., and Smith, A. Calibrating noise to sensitivity in private data analysis. In *Theory of cryptography conference*, pp. 265–284. Springer, 2006.

Dwork, C., Talwar, K., Thakurta, A., and Zhang, L. Analyze gauss: optimal bounds for privacy-preserving principal component analysis. In *Proceedings of the forty-sixth annual ACM symposium on Theory of computing*, pp. 11–20, 2014.

Erlingsson, Ú., Pihur, V., and Korolova, A. Rappor: Randomized aggregatable privacy-preserving ordinal response. In *Proceedings of the 2014 ACM SIGSAC conference on computer and communications security*, pp. 1054–1067, 2014.

Fanti, G., Pihur, V., and Erlingsson, Ú. Building a rappor with the unknown: Privacy-preserving learning of associations and data dictionaries. *Proceedings on Privacy Enhancing Technologies*, 2016(3):41–61, 2016.

Gao, C. et al. Robust regression via mutivariate regression depth. *Bernoulli*, 26(2):1139–1170, 2020.

Hopkins, S., Li, J., and Zhang, F. Robust and heavy-tailed mean estimation made simple, via regret minimization. *Advances in Neural Information Processing Systems*, 33, 2020.

Hopkins, S. B. Mean estimation with sub-gaussian rates in polynomial time. *Annals of Statistics*, 48(2):1193–1213, 2020.

Hopkins, S. B. and Li, J. Mixture models, robustness, and sum of squares proofs. In *Proceedings of the 50th Annual ACM SIGACT Symposium on Theory of Computing*, pp. 1021–1034, 2018.

Hopkins, S. B. and Li, J. How hard is robust mean estimation? In *Conference on Learning Theory*, pp. 1649–1682. PMLR, 2019.

Huber, P. J. Robust Estimation of a Location Parameter. *The Annals of Mathematical Statistics*, 35(1):73 – 101, 1964.

Jambulapati, A., Li, J., and Tian, K. Robust sub-gaussian principal component analysis and width-independent schatten packing. *Advances in Neural Information Processing Systems*, 33, 2020.

Jia, H. and Vempala, S. Robustly clustering a mixture of gaussians. *arXiv preprint arXiv:1911.11838*, 2019.

Kairouz, P., Oh, S., and Viswanath, P. The composition theorem for differential privacy. In *International conference on machine learning*, pp. 1376–1385, 2015.

Kairouz, P., McMahan, H. B., Avent, B., Bellet, A., Bennis, M., Bhagoji, A. N., Bonawitz, K., Charles, Z., Cormode, G., Cummings, R., et al. Advances and open problems in federated learning. *arXiv preprint arXiv:1912.04977*, 2019.

Kamath, G., Li, J., Singhal, V., and Ullman, J. Privately learning high-dimensional distributions. In *Conference on Learning Theory*, pp. 1853–1902, 2019.

Kamath, G., Sheffet, O., Singhal, V., and Ullman, J. Differentially private algorithms for learning mixtures of separated gaussians. In *2020 Information Theory and Applications Workshop (ITA)*, pp. 1–62. IEEE, 2020a.

Kamath, G., Singhal, V., and Ullman, J. Private mean estimation of heavy-tailed distributions. *arXiv preprint arXiv:2002.09464*, 2020b.

Kaplan, H., Ligett, K., Mansour, Y., Naor, M., and Stemmer, U. Privately learning thresholds: Closing the exponential gap. In *Conference on Learning Theory*, pp. 2263–2285. PMLR, 2020.

Karmalkar, S. and Price, E. Compressed sensing with adversarial sparse noise via l1 regression. In *2nd Symposium on Simplicity in Algorithms*, 2019.

Karmalkar, S., Klivans, A., and Kothari, P. List-decodable linear regression. In *Advances in Neural Information Processing Systems*, pp. 7423–7432, 2019.

Karwa, V. and Vadhan, S. Finite sample differentially private confidence intervals. *arXiv preprint arXiv:1711.03908*, 2017.

Klivans, A., Kothari, P. K., and Meka, R. Efficient algorithms for outlier-robust regression. In *Conference On Learning Theory*, pp. 1420–1430, 2018.

Kong, W., Somani, R., Kakade, S., and Oh, S. Robust meta-learning for mixed linear regression with small batches. *Advances in Neural Information Processing Systems*, 33, 2020.

Kothari, P. K., Steinhardt, J., and Steurer, D. Robust moment estimation and improved clustering via sum of squares. In *Proceedings of the 50th Annual ACM SIGACT Symposium on Theory of Computing*, pp. 1035–1046, 2018.

Lai, K. A., Rao, A. B., and Vempala, S. Agnostic estimation of mean and covariance. In *2016 IEEE 57th Annual Symposium on Foundations of Computer Science (FOCS)*, pp. 665–674. IEEE, 2016.

Li, J. CSE 599-M, Lecture Notes: Robustness in Machine Learning , 2019. URL: https://jerryzli.github.io/robust-ml-fall19/lec7.pdf.

Li, J. and Ye, G. Robust gaussian covariance estimation in nearly-matrix multiplication time. *Advances in Neural Information Processing Systems*, 33, 2020.

Liu, L., Shen, Y., Li, T., and Caramanis, C. High dimensional robust sparse regression. *arXiv preprint arXiv:1805.11643*, 2018.

Lugosi, G., Mendelson, S., et al. Sub-gaussian estimators of the mean of a random vector. *Annals of Statistics*, 47 (2):783–794, 2019.

McSherry, F. and Talwar, K. Mechanism design via differential privacy. In *48th Annual IEEE Symposium on Foundations of Computer Science (FOCS'07)*, pp. 94–103. IEEE, 2007.

Mukhoty, B., Gopakumar, G., Jain, P., and Kar, P. Globally-convergent iteratively reweighted least squares for robust regression problems. In *The 22nd International Conference on Artificial Intelligence and Statistics*, pp. 313–322, 2019.

Prasad, A., Suggala, A. S., Balakrishnan, S., and Ravikumar, P. Robust estimation via robust gradient estimation. *arXiv preprint arXiv:1802.06485*, 2018.

Raghavendra, P. and Yau, M. List decodable learning via sum of squares. In *Proceedings of the Fourteenth Annual ACM-SIAM Symposium on Discrete Algorithms*, pp. 161–180. SIAM, 2020.

Steinhardt, J., Charikar, M., and Valiant, G. Resilience: A criterion for learning in the presence of arbitrary outliers. In *9th Innovations in Theoretical Computer Science Conference (ITCS 2018)*. Schloss Dagstuhl-Leibniz-Zentrum fuer Informatik, 2018.

Tang, J., Korolova, A., Bai, X., Wang, X., and Wang, X. Privacy loss in apple's implementation of differential privacy on macos 10.12. *arXiv preprint arXiv:1709.02753*, 2017.

Tukey, J. W. A survey of sampling from contaminated distributions. *Contributions to probability and statistics*, pp. 448–485, 1960.

Vadhan, S. The complexity of differential privacy. In *Tutorials on the Foundations of Cryptography*, pp. 347–450. Springer, 2017.

Wainwright, M. J. *High-dimensional statistics: A non-asymptotic viewpoint*, volume 48. Cambridge University Press, 2019.

Wei, L., Sarwate, A. D., Corander, J., Hero, A., and Tarokh, V. Analysis of a privacy-preserving pca algorithm using random matrix theory. In *2016 IEEE Global Conference on Signal and Information Processing (GlobalSIP)*, pp. 1335–1339. IEEE, 2016.

Xiao, H., Biggio, B., Brown, G., Fumera, G., Eckert, C., and Roli, F. Is feature selection secure against training data poisoning? In *International Conference on Machine Learning*, pp. 1689–1698. PMLR, 2015.

Zhang, H., Kamath, G., Kulkarni, J., and Wu, Z. S. Privately learning markov random fields. *arXiv preprint arXiv:2002.09463*, 2020.

Zhu, B., Jiao, J., and Steinhardt, J. Generalized resilience and robust statistics. *arXiv preprint arXiv:1909.08755*, 2019.

We provide related work in Appendix A, the main results for heavy-tailed distributions in Appendix B, the non-private robust mean estimation in Appendix D, a new framework for private iterative filtering in Appendix E, description of PRIME in Appendix F, analysis of PRIME in Appendix G, technical lemmas in Appendix H, analysis of the exponential time Algorithm 3 in Appendix I, the algorithm and analysis of PRIME-HT in Appendix J, and the experimental details in Appendix K.

A. Related work

Private statistical analysis. Traditional private data analyses require bounded support of the samples to leverage the resulting bounded sensitivity. For example, each entry is constrained to have finite ℓ_2 norm in standard private principal component analysis (Chaudhuri et al., 2013), which does not apply to Gaussian samples. Fundamentally departing from these approaches, (Karwa & Vadhan, 2017) first established an optimal mean estimation of Gaussian samples with *unbounded* support. The breakthrough is in first adaptively estimating the range of the data using a private histogram, thus bounding the support and the resulting sensitivity. This spurred the design of private algorithms for high-dimensional mean and covariance estimation (Kamath et al., 2019; Biswas et al., 2020), heavy-tailed mean estimation (Kamath et al., 2020b), learning mixture of Gaussian (Kamath et al., 2020a), learning Markov random fields (Zhang et al., 2020), and statistical testing (Canonne et al., 2019). Under the Gaussian distribution with no adversary, (Aden-Ali et al., 2020) achieves an accuracy of $\|\hat{\mu} - \mu\|_2 \leq \tilde{\alpha}$ with the best known sample complexity of $n = \widetilde{O}((d/\tilde{\alpha}^2) + (d/\tilde{\alpha}\varepsilon) + (1/\varepsilon)\log(1/\delta))$ while guaranteeing (ε, δ)-differential privacy. This nearly matches the known lower bounds of $\Omega(d/\tilde{\alpha}^2)$ for non-private finite sample complexity, $\widetilde{\Omega}((1/\varepsilon)\min\{\log(1/\delta), \log(R)\})$ for privately learning one-dimensional unit variance Gaussian (Karwa & Vadhan, 2017), and $\widetilde{\Omega}(d/\tilde{\alpha}\varepsilon)$ for multi-dimensional Gaussian estimation (Kamath et al., 2019). However, this does not generalize to sub-Gaussian distributions and (Aden-Ali et al., 2020) does not provide a tractable algorithm. A polynomial time algorithm is proposed in (Kamath et al., 2019) that achieves a slightly worse sample complexity of $\widetilde{O}((d/\tilde{\alpha}^2) + (d\log^{1/2}(1/\delta)/\tilde{\alpha}\varepsilon))$, which can also seamlessly generalized to sub-Gaussian distributions.

(Cai et al., 2019) takes a different approach of deviating from standard definition of sub-Gaussianity to provide a larger lower bound on the sample complexity scaling as $n = \Omega(d\sqrt{\log(1/\delta)}/(\alpha\varepsilon))$ for mean estimation with a known covariance. Concretely, they consider distributions satisfying $\mathbb{E}_{x\sim P}[e^{\lambda\langle x-\mu, e_k\rangle}] \leq e^{\lambda^2\sigma^2}$ for all $k \in [d]$ where e_k is the k-th standard basis vector. Notice that this condition only requires sub-Gaussianity when projected onto standard bases. Standard definition of high-dimensional sub-Gaussianity (which is assumed in this paper) requires sub-Gaussianity in all directions. Therefore, their lower bound is not comparable with our achievable upper bounds. Further, the example they construct to show the lower bound does not satisfy our sub-Gaussianity assumptions.

In an attempt to design efficient algorithms for robust and private mean estimation, (Dhar & Huang, 2020) proposed an algorithm with a mis-calculated sensitivity, which can result in violating the privacy guarantee. This can be corrected by pre-processing with our approach of checking the resilience (as in Algorithm 3), but this requires a run-time exponential in the dimension.

For estimating the mean of a *covariance bounded* distributions up to an error of $\|\hat{\mu} - \mu\|_2 = O(\tilde{\alpha}^{1/2})$, (Kamath et al., 2020b) shows that $\Omega(d/(\tilde{\alpha}\varepsilon))$ samples are necessary and provides an efficient algorithm matching this up to a factor of $\log^{1/2}(1/\delta)$. For a more general family of distributions with bounded k-moment, (Kamath et al., 2020b) shows that an error of $\|\hat{\mu} - \mu\|_2 = O(\tilde{\alpha}^{(k-1)/k})$ can be achieved with $n = \widetilde{O}((d/\tilde{\alpha}^{2(k-1)/k}) + (d\log^{1/2}(1/\delta)/(\varepsilon\tilde{\alpha})))$ samples.

However, under α-corruption, (Hopkins & Li, 2019) shows that achieving an error better than $O(\alpha^{1/2})$ under k-th moment bound is as computationally hard as the small-set expansion problem, even without requiring DP. Hence, under the assumption of P $\neq$ NP, no polynomial-time algorithm exists that can outperform our PRIME-HT even if we have stronger assumptions of k-th moment bound. On the other hand, there exists an exponential time algorithm for non-private robust mean estimation that achieves $\|\mu - \hat{\mu}\|_2 = O(\alpha^{(k-1)/k})$ (Zhu et al., 2019). Combining it with the bound of (Hopkins & Li, 2019), an interesting open question is whether there is an (exponential time) algorithm that achieves $\|\mu - \hat{\mu}\|_2 = O(\alpha^{(k-1)/k})$ with sample complexity $n = \widetilde{O}((d/\alpha^{2(k-1)/k}) + (d\log^{1/2}(1/\delta)/(\varepsilon\alpha)))$ under α-corruption and (ε, δ)-DP.

Robust estimation. Designing robust estimators under the presence of outliers has been considered by statistics community since 1960s (Tukey, 1960; Anscombe, 1960; Huber, 1964). Recently, (Diakonikolas et al., 2019a; Lai et al., 2016) give the first polynomial time algorithm for mean and covariance estimation with no (or very weak) dependency on the dimensionality in the estimation error. Since then, there has been a flurry of research on robust estimation problems, including mean

estimation (Diakonikolas et al., 2017; Dong et al., 2019; Hopkins et al., 2020; Hopkins, 2020; Diakonikolas et al., 2018a), covariance estimation (Cheng et al., 2019b; Li & Ye, 2020), linear regression and sparse regression (Bhatia et al., 2015; 2017; Balakrishnan et al., 2017; Gao et al., 2020; Prasad et al., 2018; Klivans et al., 2018; Diakonikolas et al., 2019b; Liu et al., 2018; Karmalkar & Price, 2019; Dalalyan & Thompson, 2019; Mukhoty et al., 2019; Diakonikolas et al., 2019c; Karmalkar et al., 2019), principal component analysis (Kong et al., 2020; Jambulapati et al., 2020), mixture models (Diakonikolas et al., 2020; Jia & Vempala, 2019; Kothari et al., 2018; Hopkins & Li, 2018) and list-decodable learning (Diakonikolas et al., 2018b; Raghavendra & Yau, 2020; Charikar et al., 2017; Bakshi & Kothari, 2020; Cherapanamjeri et al., 2020). See (Diakonikolas & Kane, 2019) for a survey of recent work.

One line of work that is particularly related to our algorithm PRIME is (Cheng et al., 2019a; Dong et al., 2019; Depersin & Lecué, 2019; Cheng et al., 2019b; Cherapanamjeri et al., 2020), which leverage the ideas from matrix multiplicative weight and fast SDP solver to achieve faster, sometimes nearly linear time, algorithms for mean and covariance estimation. In PRIME, we use a matrix multiplicative weight approach similar to (Dong et al., 2019) to reduce the iteration complexity to logarithmic, which enables us to achieve the $d^{3/2}$ dependency in the sample complexity.

The concept of *resilience* is introduced in (Steinhardt et al., 2018) as a sufficient condition such that learning in the presence of adversarial corruption is information-theoretically possible. The idea of resilience is later generalized in (Zhu et al., 2019) for a wider range of adversarial corruption models. While there exists simple exponential time robust estimation algorithm under resilience condition, it is challenging to achieve differential privacy due to high sensitivity. We propose a novel approach to leverage the resilience property in our exponential time algorithm for sub-gaussian and heavy-tailed distributions.

B. Main results under heavy-tailed distributions

We consider distributions with bounded covariance as defined as follows.

Assumption 2. *An uncorrupted dataset $S_{\rm good}$ consists of n i.i.d. samples from a distribution with mean $\mu \in [-R, R]^d$ and covariance $\Sigma \preceq \mathbf{I}$. For some $\alpha \in (0, 1/2)$, we are given a corrupted dataset $S = \{x_i\}_{i=1}^n$ where an adversary adaptively inspects all samples in $S_{\rm good}$, removes αn of them and replaces them with $S_{\rm bad}$ that are αn arbitrary points in $\mathbb{R}^d$.*

Under these assumptions, Algorithm 3 achieves near optimal guarantees but takes exponential time. The dominant term in the sample complexity $\widetilde{\Omega}(d/(\varepsilon\alpha))$ cannot be improved as it matches that of the optimal non-robust private estimation (Kamath et al., 2020b). The accuracy $O(\sqrt{\alpha})$ cannot be improved as it matches that of the optimal non-private robust estimation (Dong et al., 2019). We provide a proof in Appendix I.1.

Theorem 7 (Exponential time algorithm for covariance bounded distributions)**.** *Algorithm 3 is (ε, δ)-differentially private. Under Assumption 2, if*

$$n = \Omega\Big(\frac{d\log(dR/\alpha) + d^{1/2}\log(1/\delta)}{\varepsilon\alpha} + \frac{d^{1/2}\log^{3/2}(1/\delta)\min\{\log(dR), \log(d/\delta)\}}{\varepsilon} \Big) ,$$

this algorithm achieves $\|\hat{\mu} - \mu\|_2 = O(\sqrt{\alpha})$ with probability 0.9.

We propose an efficient algorithm PRIME-HT and show that it achieves the same optimal accuracy but at the cost of increased sample complexity of $O(d^{3/2}\log(1/\delta)/(\varepsilon\alpha))$. In the first step, we need increase the radius of the ball to $O(\sqrt{d/\alpha})$ to include a $1-\alpha$ fraction of the clean samples, where $q_{\rm range-ht}$ returns $B = O(1/\sqrt{\alpha})$ and $\mathcal{B}_{\sqrt{d}B/2}(\bar{x})$ is a ℓ_2-ball of radius $\sqrt{d}B/2$ centered at $\bar{x}$. This is followed by a matrix multiplicative weight filter similar to DPMMWFILTERR but the parameter choices are tailored for covariance bounded distributions. We provide a proof in Appendix J.2.

Theorem 8 (Efficient algorithm for covariance bounded distributions)**.** PRIME-HT *is (ε, δ)-differentially private. Under Assumption 2 there exists a universal constant $c \in (0, 0.1)$ such that if $\alpha \le c$, and $n = \widetilde{\Omega}((d^{3/2}/(\varepsilon\alpha))\log(1/\delta))$, then* PRIME-HT *achieves $\|\hat{\mu} - \mu\|_2 = O(\alpha^{1/2})$ with probability 0.9. The notation $\widetilde{\Omega}(\cdot)$ hides logarithmic terms in d, R, and $1/\alpha$.*

Remark 1. To boost the success probability to $1-\zeta$ for some small $\zeta > 0$, we will randomly split the data into $O(\log(1/\zeta))$ subsets of equal sizes, and run Algorithm 2 to obtain a mean estimation from each of the subset. Then we can apply multivariate "mean-of-means" type estimator (Lugosi et al., 2019) to get $\|\hat{\mu} - \mu\|_2 = O(\alpha^{1/2})$ with probability $1-\zeta$. This is efficient as we only have $O(\log 1/\zeta)$ trials and run-time of mean-of-means is dominated by the time it takes to find all

pairwise distances, which is only $O(d\,(\log(1/\zeta))^2)$. There are $(\log(1/\zeta))^2$ pairs, and for each pair we compute the distance between means in d operations.

Algorithm 2: PRIvate and robust Mean Estimation for covariance bounded distributions (PRIME-HT)

Input: $S = \{x_i \in \mathbb{R}^d\}_{i=1}^n$, range $[-R, R]^d$, adversarial fraction $\alpha \in (0, 1/2)$, number of iterations $T_1 = O(\log(d/\alpha)), T_2 = O(\log d)$, target privacy (ε, δ)

$(\bar{x}, B) \leftarrow q_{\mathrm{range-ht}}(R, 0.01\varepsilon, 0.01\delta)$ [Algorithm 14 in Appendix J]

Project the data onto the ball: $\tilde{x}_i \leftarrow \mathcal{P}_{\mathcal{B}_{\sqrt{d}B/2}(\bar{x})}(x_i)$, for all $i \in [n]$

$\hat{\mu} \leftarrow$ DPMMWFILTER-HT$(\{\tilde{x}_i\}_{i=1}^n, \alpha, T_1, T_2, 0.99\varepsilon, 0.99\delta)$ [Algorithm 15 in Appendix J]

Output: $\hat{\mu}$

C. Exponential time algorithm with near-optimal sample complexity

Novelty. An existing exponential time algorithm for robust and private mean estimation in (Bun et al., 2019) strictly requires the uncorrupted samples to be drawn from a Gaussian distribution. We introduce a novel estimator that achieves near-optimal guarantees for more general sub-Gaussian distributions (and also covariance bounded distributions) but takes an *exponential* run-time. Its innovation is in leveraging on the *resilience* property of well-behaved distributions not only to estimate the mean robustly (which is the standard use of the property) but also to adaptively bound the sensitivity of the estimator, thus achieving optimal privacy-accuracy tradeoff.

Definition C.1 (Resilience from Definition 1 in (Steinhardt et al., 2018))**.** *A set of points $\{x_i\}_{i\in S}$ lying in $\mathbb{R}^d$ is (σ,α)-resilient around a point μ if $\|(1/|T|)\sum_{i\in T}(x_i-\mu)\|_2 \le \sigma$ for all subsets $T\subset S$ of size $(1-\alpha)|S|$.*

Algorithm. As data is corrupted, we define $R(S)$ as a surrogate for resilience of the uncorrupted part of the set. If S indeed consists of a $1-\alpha$ fraction of independent samples from the promised class of distributions, the goodness score $R(S)$ will be close to the resilience property of the good data.

Definition C.2 (Goodness of a set)**.** *For $\mu(S)=(1/|S|)\sum_{i\in S}x_i$, let us define*

$$R(S) \triangleq \min_{S'\subset S,|S'|=(1-2\alpha)|S|.} \ \max_{T\subset S',|T|=(1-\alpha)|S'|.} \|\mu(T)-\mu(S')\|_2 \ .$$

Algorithm 3 first checks if the resilience matches that of the promised distribution. The data is pre-processed with q_{range} to ensure we can check $R(S)$ privately. Once resilience is cleared, we can safely use the exponential mechanism based on the score function $d(\hat{\mu},S)$ in Definition C.3 to select an approximate robust mean $\hat{\mu}$ privately. The choice of the sensitivity critically relies on the fact that resilient datasets have small sensitivity of $O((1/n)\sqrt{\log(1/\alpha)})$. Without the resilience check, the sensitivity is $O(d^{1/2}/n)$ resulting in an extra factor of $\sqrt{d}$ in the sample complexity.

Algorithm 3: Exponential-time private and robust mean estimation

Input: $S=\{x_i\}_{i\in[n]}$, $\alpha\in(0,1/2)$, R, (ε,δ)

if $n < cd^{1/2}\log(1/\delta)/\left(\varepsilon\alpha\sqrt{\log(1/\alpha)}\right)$ **then Output:** $\emptyset$ [$cd^{1/2}\log(1/\delta)/(\varepsilon\alpha)$ for hevay-tail]

$(\bar{x},B)\leftarrow q_{\text{range}}(R,(1/3)\varepsilon,(1/3)\delta)$ [$q_{\text{range-ht}}(\cdot)$ for hevay-tail]

Project the data points onto the ball: $x_i\leftarrow \mathcal{P}_{\mathcal{B}_{\sqrt{d}B/2}(\bar{x})}(x_i)$, for all $i\in[n]$

$\widehat{R}(S)\leftarrow R(S)+\text{Lap}(3Bd^{1/2}/(n\varepsilon))$

if $\widehat{R}(S) > 2\alpha\sqrt{\log(1/\alpha)}$ **then Output:** $\emptyset$ [$\widehat{R}(S) > 2c_\zeta\sqrt{\alpha}$ for hevay-tail]

else Output: a randomly drawn point $\hat{\mu}\in[-2R,2R]^d$ sampled from a density

$r(\hat{\mu}) \propto e^{-(1/(24\sqrt{\log(1/\alpha)}))\varepsilon\, n\, d(\hat{\mu},S)}$ [$e^{-(\varepsilon n\sqrt{\alpha}/(24c_\zeta))d(\hat{\mu},S)}$ for heavy-tail]

We propose the score function $d(\hat{\mu},S)$ in the following definition, which is a robust estimator of the distance between the mean and the candidate $\hat{\mu}$.

Definition C.3. *For a set of data $\{x_i\}_{i\in S}$ lying in $\mathbb{R}^d$, for any $v\in\mathbb{S}^{d-1}$, define $\mathcal{T}^v$ to be the $3\alpha|S|$ points with the largest $v^\top x_i$ value, $\mathcal{B}^v$ to be the $3\alpha|S|$ points with the smallest $v^\top x_i$ value, and $\mathcal{M}^v = S\setminus(\mathcal{T}^v\cup\mathcal{B}^v)$. Define $d(\hat{\mu},S) \triangleq \max_{v\in\mathbb{S}^{d-1}}\left|v^\top(\mu(\mathcal{M}^v)-\hat{\mu})\right|$.*

Analysis. For any direction v, the truncated mean estimator $\mu(\mathcal{M}^v)$ provides a robust estimation of the true mean along the direction v, thus the distance can be simply defined by taking the maximum over all directions v. We show the sensitivity of this simple estimator is bounded by the resilience property σ divided by n, which is $O((1/n)\sqrt{\log(1/\alpha)})$ once the resilience check is passed. This leads to the following near-optimal sample complexity. We provide a proof in Appendix I.2.

Theorem 9 (Exponential time algorithm for sub-Gaussian distributions)**.** *Algorithm 3 is (ε,δ)-DP. Under Assumption 1, this algorithm achieves $\|\hat{\mu}-\mu\|_2 = O(\alpha\sqrt{\log(1/\alpha)})$ with probability $1-\zeta$ if*

$$n=\widetilde{\Omega}\Big(\frac{d+\log\frac{1}{\zeta}}{\alpha^2\log\frac{1}{\alpha}}+\frac{d\log\frac{dR}{\alpha}+d^{1/2}\log\frac{1}{\delta}+\log\frac{1}{\zeta}}{\varepsilon\alpha}+\frac{\sqrt{d\log\frac{1}{\delta}}\min\{\log\frac{dR}{\zeta},\log\frac{d}{\zeta\delta}\}}{\varepsilon}\Big) \ .$$

Run-time. Computing $R(S)$ exactly can take $O(de^{\Theta(n)})$ operations. The exponential mechanism implemented with α-covering for $\hat{\mu}$ and a constant covering for v can take $O(nd(R/\alpha)^d)$ operations.

D. Background on (non-private) robust mean estimation

The following tie-breaking rule is not essential for robust estimation, but is critical for proving differential privacy, as shown later in Appendix G.1.

Definition D.1 (Subset of the largest α fraction)**.** *Given a set of scalar values $\{\tau_i = \langle V, (x_i - \mu)(x_i - \mu)^\top \rangle\}_{i \in S'}$ for a subset $S' \subseteq [n]$, define the sorted list π of S' such that $\tau_{\pi(i)} \geq \tau_{\pi(i+1)}$ for all $i \in [|S'| - 1]$. When there is a tie such that $\tau_i = \tau_j$, it is broken by $\pi^{-1}(i) \leq \pi^{-1}(j) \Leftrightarrow x_{i,1} \geq x_{j,1}$. Further ties are broken by comparing the remaining entries of x_i and x_j, in an increasing order of the coordinate. If $x_i = x_j$,then the tie is broken arbitrarily. We define $\mathcal{T}_\alpha = \{\pi(1), \ldots, \pi(\lceil n\alpha \rceil)\}$ to be the set of largest $\lceil n\alpha \rceil$ valued samples.*

With this definition of α-tail, we can now provide a complete description of the robust mean estimation that achieves the guarantee provided in Proposition 2.1.

Algorithm 4: Non-private robust mean estimation (Li, 2019)

Input: $S = \{x_i\}_{i=1}^n$, $\alpha \in (0, 1)$, $S_0 = [n]$

for $t = 1, \ldots$ **do**

  if $\| \sum_{i \in S_{t-1}} (x_i - \mu_{t-1})(x_i - \mu_{t-1})^\top - \mathbf{I} \|_2 < C\alpha \log(1/\alpha)$ **then**

    Output: $\hat{\mu} = \sum_{i \in S_{t-1}} x_i$

  else

    $\mu_t \leftarrow (1/|S_{t-1}|) \sum_{i \in S_{t-1}} x_i$

    $v_t \leftarrow$ 1st principal direction of $(\{(x_i - \mu_t)\}_{i \in S_{t-1}})$

    $Z_t \leftarrow \mathrm{Unif}([0, 1])$

    $S_t \leftarrow S_{t-1} \setminus \{i \,|\, i \in \mathcal{T}_{2\alpha}$ for $\{\tau_j = (v_t^\top (x_j - \mu_t))^2\}_{j \in S_{t-1}}$ and $\tau_i \geq Z_t \max_{j \in S_{t-1}} (v_t^\top (x_j - \mu_t))^2\}$, where $\mathcal{T}_{2\alpha}$ is defined in Definition D.1.

E. A new framework for *private* iterative filtering

We provide complete descriptions of all algorithms used in private iterative filtering. We present the *interactive* version first, followed by the *centralized* version.

E.1. Interactive version of the algorithm

Adaptive estimation of the range of the dataset is essential in computing private statistics of data. We use the following algorithm proposed in (Karwa & Vadhan, 2017). It computes a private histogram of a set of 1-dimensional points and select the largest bin as the one potentially containing the mean of the data. Note that B does not need not be chosen adaptively to include all the uncorrupted data with a high probability.

Algorithm 5: Differentially private range estimation (q_{range}) (**?**)Algorithm 1]KV17

Input: $\mathcal{D}_n = \{x_i\}_{i=1}^n$, R, ε, δ, $\sigma = 1$

for $j \leftarrow 1$ **to** d **do**

  Run the histogram learner of Lemma E.1 with privacy parameters $\left(\min\{\varepsilon, 0.9\}/2\sqrt{2d \log(2/\delta)}, \delta/(2d)\right)$ and bins $B_l = (2\sigma\ell, 2\sigma(\ell + 1)]$ for all $\ell \in \{-\lceil R/2\sigma \rceil - 1, \ldots, \lceil R/2\sigma \rceil\}$ on input $\mathcal{D}_n$ to obtain noisy estimates $\{\tilde{h}_{j,l}\}_{l=-\lceil R/2\sigma \rceil - 1}^{\lceil R/2\sigma \rceil}$

  $\bar{x}_j \leftarrow 2\sigma \cdot \arg\max_{\ell \in \{-\lceil R/2\sigma \rceil - 1, \ldots, \lceil R/2\sigma \rceil\}} \tilde{h}_{j,\ell}$

Output: $(\bar{x}, B = 8\sigma\sqrt{\log(dn/\zeta)})$

The following guarantee (and the algorithm description) is used in the analysis (and the implementation) of the query q_{range}.

Lemma E.1 (Histogram Learner, Lemma 2.3 in (Karwa & Vadhan, 2017))**.** *For every $K \in \mathbb{N} \cup \infty$, domain Ω, for every collection of disjoint bins $B_1, \ldots, B_K$ defined on Ω, $n \in \mathbb{N}$, $\varepsilon, \delta \in (0, 1/n)$, $\beta > 0$ and $\alpha \in (0, 1)$ there exists an (ε, δ)-differentially private algorithm $M : \Omega^n \to \mathbb{R}^K$ such that for any set of data $X_1, \ldots, X_n \in \Omega^n$*

1. $\hat{p}_k = \frac{1}{n}\sum_{X_i \in B_k} 1$

2. $(\tilde{p}_1, \ldots, \tilde{p}_K) \leftarrow M(X_1, \ldots, X_n)$, *and*

3.
$$n \geq \min\left\{\frac{8}{\varepsilon\beta}\log(2K/\alpha), \frac{8}{\varepsilon\beta}\log(4/\alpha\delta)\right\}$$

then,

$$\mathbb{P}(|\tilde{p}_k - \hat{p}_k| \leq \beta) \geq 1 - \alpha$$

Proof. This is an intermediate result in the proof of Lemma 2.3 in (Karwa & Vadhan, 2017). □

The rest of the queries (q_{size}, q_{mean}, q_{PCA}, and q_{norm}) are provided below. The most innovative part is the repeated application of filtering that is run every time one of the queries is called. In the Filter query below, because we choose (i) to use the sampling version of robust mean estimation as opposed to weighting version which assigned a weight on each sample between zero and one measuring how good (i.e., score one) or bad (i.e., score zero) each sample point is, and (ii) we switched the threshold to be dB^2Z_ℓ, we can show that this filtering with fixed parameters $\{\mu_\ell, v_\ell, Z_\ell\}_{\ell\in[t-1]}$ preserves sensitivity in Lemma 2.2. This justifies the choice of noise in each output perturbation mechanism, satisfying the desired level of (ε, δ)-DP. We provide the complete privacy analysis in Appendix E.3 and also the analysis of the utility of the algorithm as measure by the accuracy.

Algorithm 6: Interactive private queries used in Algorithm 1

Filter $(\{(\mu_\ell, v_\ell, Z_\ell)\}_{\ell\in[t-1]}, \bar{x}, B)$**:**
  $S_0 \leftarrow [n]$
  Clip the data points: $x_i \leftarrow \mathcal{P}_{\bar{x}+[-B/2,B/2]^d}(x_i)$, for all $i \in [n]$
  for $\ell = 1, \ldots, t-1$ **do**
    $S_\ell \leftarrow S_{\ell-1} \setminus \{i \in S_{\ell-1} : i \in \mathcal{T}_{2\alpha}$ for $\{\tau_j = (v_\ell^\top(x_j - \mu_\ell))^2\}_{j\in S_{\ell-1}}$ and $\tau_i \geq d\,B^2\,Z_\ell\}$

$q_{\text{mean}}(\{(\mu_\ell, v_\ell, Z_\ell)\}_{\ell\in[t-1]}, \varepsilon, \bar{x}, B)$**:**
  Filter$(\{(\mu_\ell, v_\ell, Z_\ell)\}_{\ell\in[t-1]}, \bar{x}, B)$
  return $\mu_t \leftarrow (1/|S_{t-1}|)\big(\sum_{i\in S_{t-1}} x_i\big) + \text{Lap}(2B/(n\varepsilon))$

$q_{\text{PCA}}(\{(\mu_\ell, v_\ell, Z_\ell)\}_{\ell\in[t-1]}, \mu_t, \varepsilon, \delta, \bar{x}, B)$**:**
  Filter$(\{(\mu_\ell, v_\ell, Z_\ell)\}_{\ell\in[t-1]}, \bar{x}, B)$
  return $v_t \leftarrow$ top singular vector of $\Sigma_{t-1} =$
  $(1/n)\sum_{i\in S_{t-1}}(x_i - \mu_t)(x_i - \mu_t)^\top + \mathcal{N}(0, (B^2 d\sqrt{2\log(1.25/\delta)}/(n\varepsilon))^2 \mathbf{I}_{d^2\times d^2})$

$q_{\text{norm}}(\{(\mu_\ell, v_\ell, Z_\ell)\}_{\ell\in[t-1]}, \mu_t, \varepsilon, \bar{x}, B)$**:**
  Filter$(\{(\mu_\ell, v_\ell, Z_\ell)\}_{\ell\in[t-1]}, \bar{x}, B)$
  return $\lambda_t \leftarrow \|(1/n)\sum_{i\in S_{t-1}}(x_i - \mu_t)(x_i - \mu_t)^\top\|_2 + \text{Lap}(2B^2 d/(n\varepsilon))$

$q_{\text{size}}(\{(\mu_\ell, v_\ell, Z_\ell)\}_{\ell\in[t-1]}, \varepsilon, \bar{x}, B)$**:**
  Filter$(\{(\mu_\ell, v_\ell, Z_\ell)\}_{\ell\in[t-1]}, \bar{x}, B)$
  return $n_t \leftarrow |S_{t-1}| + \text{Lap}(1/\varepsilon)$

E.2. Centralized version of the algorithm

In practice, one should run the centralized version of the private iterative filtering, in order to avoid multiple redundant computations of the interactive version. The main difference is that the redundant filtering repeated every time a query is called in the interactive version is now merged into a single run. The resulting estimation and the privacy loss are exactly the

same.

Algorithm 7: Private iterative filtering (centralized version)

Input: $S = \{x_i \in \mathbb{R}^d\}_{i=1}^n$, range $[-R, R]^d$, adversarial fraction $\alpha \in (0, 1)$, target probability $\eta \in (0, 1)$, number of iterations $T = \widetilde{\Theta}(d)$, target privacy (ε, δ)

1 $(\bar{x}, B) \leftarrow q_{\text{range}}(R, 0.01\varepsilon, 0.01\delta)$ [Algorithm 5]

2 Clip the data points: $\tilde{x}_i \leftarrow \mathcal{P}_{\bar{x}+[-B/2,B/2]^d}(x_i)$, for all $i \in [n]$

3 $\hat{\mu} \leftarrow$ DPFILTER$(\{\tilde{x}_i\}_{i=1}^n, \alpha, T, 0.99\varepsilon, 0.99\delta)$ [Algorithm 8]

Output: $\hat{\mu}$

First, q_{range} introduced in (Karwa & Vadhan, 2017), returns a hypercube $\bar{x} + [-B, B]^d$ that is guaranteed to include all uncorrupted samples, while preserving privacy. It is followed by a private filtering DPFILTER in Algorithm 8.

Algorithm 8: Differentially private filtering (DPFILTER)

Input: $S = \{x_i \in \bar{x} + [-B/2, B/2]^d\}_{i=1}^n$, $\alpha \in (0, 1/2)$, $T = \widetilde{O}(dB^2 \log(dB^2/(\alpha \log(1/\alpha)))), (\varepsilon, \delta)$

1 $S_0 \leftarrow [n]$, $\varepsilon_1 \leftarrow \min\{\varepsilon, 0.9\}/(4\sqrt{2T \log(2/\delta)})$, $\delta_1 \leftarrow \delta/(8T)$

2 **if** $n < (4/\varepsilon_1) \log(1/(2\delta_1))$ **then Output:** $\emptyset$

3 **for** $t = 1, \ldots, T$ **do**

4 $\quad n_t \leftarrow |S_{t-1}| + \text{Lap}(1/\varepsilon_1)$

5 $\quad$ **if** $n_t < 3n/4$ **then**

6 $\quad\quad$ **Output:** $\emptyset$

7 $\quad \mu_t \leftarrow (1/|S_{t-1}|) \sum_{i \in S_{t-1}} x_i + \text{Lap}(2B/(n\,\varepsilon_1))$

8 $\quad \lambda_t \leftarrow \|(1/n) \sum_{i \in S_{t-1}} (x_i - \mu_t)(x_i - \mu_t)^\top - \mathbf{I}\|_2 + \text{Lap}(2B^2 d/(n\varepsilon_1))$

9 $\quad$ **if** $\lambda_t \le (C - 0.01)\alpha \log(1/\alpha)$ **then**

$\quad\quad$ **Output:** μ_t

10 $\quad v_t \leftarrow$ top singular vector of $\Sigma_{t-1} \triangleq$

$\quad\quad \frac{1}{n} \sum_{i \in S_{t-1}} (x_i - \mu_t)(x_i - \mu_t)^\top + \mathcal{N}(0, (B^2 d \sqrt{2 \log(1.25/\delta)}/(n\varepsilon_1))^2 \mathbf{I}_{d^2 \times d^2})$

11 $\quad Z_t \leftarrow \text{Unif}([0, 1])$

12 $\quad S_t \leftarrow S_{t-1} \setminus \{i \mid i \in \mathcal{T}_{2\alpha}$ for $\{\tau_j = (v_t^\top (x_j - \mu_t))^2\}_{j \in S_{t-1}}$ and $\tau_i \ge d\,B^2\,Z_t\}$, where $\mathcal{T}_{2\alpha}$ is defined in Definition D.1.

E.3. The analysis of private iterative filtering (Algorithms 1 and 7) and a proof of Theorem 5

q_{range}, introduced in (Karwa & Vadhan, 2017), returns a hypercube $\bar{x} + [-B, B]^d$ that is guaranteed to include all uncorrupted samples, while preserving privacy. In the following lemma, we show that q_{range} is also *robust* to adversarial corruption. Such adaptive bounding of the support is critical in privacy analysis of the subsequent steps. We clip all data points by projecting all the points with $\mathcal{P}_{\bar{x}+[-B/2,B/2]^d}(x) = \arg\min_{y \in \bar{x}+[-B/2,B/2]^d} \|y - x\|_2$ to lie inside the hypercube and pass them to DPFILTER for filtering. The algorithm and a proof are provided in §E.3.1. Perhaps surprisingly, there is no dependence in R for $R > 1/\delta$, which is achieved by utilizing the private histogram mechanism from (Vadhan, 2017; Bun et al., 2016).

Lemma E.2. *$q_{\text{range}}(S, R, \varepsilon, \delta)$ (Algorithm 5) is (ε, δ)-differentially private. Under Assumption 1, $q_{\text{range}}(S, R, \varepsilon, \delta)$ returns $(\bar{x}, B)$ such that if $n = \Omega\left((\sqrt{d \log(1/\delta)}/\varepsilon) \min\left(\log(dR/\zeta), \log(d/\zeta\delta)\right)\right)$ and $\alpha < 0.1$, then all uncorrupted samples in S are in $\bar{x} + [-B, B]^d$ with probability $1 - \zeta$.*

In DPFILTER, we make only the mean μ_t and the top principal direction v_t private to decrease sensitivity. The analysis is now more challenging since (μ_t, v_t) depends on all past iterates $\{(\mu_j, v_j)\}_{j=1}^{t-1}$ and internal randomness $\{Z_j\}_{j=1}^{t-1}$. To decrease the sensitivity, we modify the filter in line 12 to use the maximum support dB^2 (which is data independent) instead of the maximum contribution $\max_i (v_t^\top (x_i - \mu_t))^2$ (which is data dependent and sensitive). While one data point can significantly change $\max_i (v_t^\top (x_i - \mu_t))^2$ and the output of one step of the filter in Algorithm 4, the sensitivity of the proposed filter is bounded conditioned on all past $\{(\mu_j, v_j)\}_{j=1}^{t-1}$, as we show in the following lemma. This follows from the fact that conditioned on (μ_j, v_j), the proposed filter is a contraction. We provide a proof in Appendix E.3.3 and Appendix E.3.4. Putting together Lemmas E.2 and E.3, we get the desired result in Theorem 5.

Lemma E.3. DPFILTER$(S, \alpha, T, \varepsilon, \delta)$ *is* (ε, δ)*-differentially private. Under the hypotheses of Theorem 5,* DPFILTER$(S, \alpha, T = \widetilde{\Theta}(B^2 d), \varepsilon, \delta)$ *achieves* $\|\hat{\mu} - \mu\|_2 = O(\alpha\sqrt{\log(1/\alpha)})$ *with probability* 0.9*, if* $n = \widetilde{\Omega}(d/\alpha^2 + B^3 d^2 \log(1/\delta)/(\varepsilon\alpha))$ *and* B *is large enough such that the original uncorrupted samples are inside the hypercube* $\bar{x} + [-B/2, B/2]^d$.

Differential privacy guarantee. To achieve $(\varepsilon_0, \delta_0)$ end-to-end target privacy guarantee, Algorithm 7 separates the privacy budget into two. The $(0.01\varepsilon_0, 0.01\delta_0)$-DP guarantee of q_{range} follows from Lemma E.2. The $(0.99\varepsilon_0, 0.99\delta_0)$-DP guarantee of DPFILTER follows from Lemma E.3.

Accuracy. From Lemma E.2 q_{range} is guaranteed to return a hypercube that includes all clean data in the dataset. It follows from Lemma E.3 that when $n = \widetilde{\Omega}(d/\alpha^2 + d^2\log(1/\delta)/(\varepsilon\alpha))$, we have $\|\mu - \hat{\mu}\|_2 = O(\alpha\sqrt{\log(1/\alpha)})$.

E.3.1. Proof of Lemma E.2 and the analysis of q_{range} in Algorithm 5

Assuming the distribution is σ^2 sub-Gaussian, we use $\mathcal{P}$ to denote the sub-Gaussian distribution. Denote $I_l = [2\sigma l, 2\sigma(l+1)]$ as the interval of the l'th bin. Denote the population probability in the l'th bin $h_{j,l} = \mathbb{P}_{x\sim\mathcal{P}}[x_j \in I_l]$, empirical probability in the l'th bin $\tilde{h}_{j,l} = \frac{1}{n}\sum_{x_i\in\mathcal{D}} \mathbf{1}\{x_{i,j} \in I_l\}$, and the noisy version $\hat{h}_{j,l}$ computed by the histogram learner of Lemma E.1. Notice that Lemma E.1 with d compositions (Lemma H.13) immediately implies that our algorithm is (ε, δ)-differentially private.

For the utility of the algorithm, we will first show that for all dimension $j \in [d]$, the output $|\bar{x}_j - \mu_j| = O(\sigma)$. Note that by the definition of σ^2-subgaussian, it holds that for all $i \in [d]$, $\mathbb{P}[|x_i - \mu_i| \geq z] \leq 2\exp(-z^2/\sigma^2)$ where x is drawn from distribution $\mathcal{P}$. This implies that $\mathbb{P}[|x_i - \mu_i| \geq 2\sigma] \leq 2\exp(-4) \leq 0.04$. Suppose the k'th bin contains μ_j, namely $\mu_j \in I_k$. Then it is clear that $[\mu_j - 2\sigma, \mu_j + 2\sigma] \subset (I_{k-1} \cup I_k \cup I_{k+1})$. This implies $h_{j,k-1} + h_{j,k} + h_{j,k+1} \geq 1 - 0.04 = 0.96$, hence $\min(h_{j,k-1}, h_{j,k}, h_{j,k+1}) \geq 0.32$.

Recall that $\mathcal{G}$ is the set of clean data drawn from distribution P. By Dvoretzky-Kiefer-Wolfowitz inequality and an union bound over $j \in [d]$, we have that with probability $1-\zeta$, $\max_{j,l}(|h_{j,l} - \frac{1}{n}\sum_{x\in G} x_j|) \leq \sqrt{\frac{\log(d/\zeta)}{n}}$. The deviation due to corruption is at most α on each bin, hence we have $\max_{j,l}(|h_{j,l} - \hat{h}_{j,l}) \leq \sqrt{\frac{\log(d/\zeta)}{n}} + \alpha$. Lemma E.1 and a union bound over $j \in [d]$ implies that with probability $1-\zeta$, $\max_{j,l}(|\tilde{h}_{j,l} - \hat{h}_{j,l}|) \leq \beta$ when $n \geq \Omega(\min\left\{\frac{\sqrt{d\log(1/\delta)}}{\varepsilon\beta}\log(dR/\zeta), \frac{\sqrt{d\log(1/\delta)}}{\varepsilon\beta}\log(d/\zeta\delta)\right\})$.

Assuming that $n = \Omega\left(\frac{\sqrt{d\log(1/\delta)}}{\varepsilon}\min\{\log(dR/\zeta), \log(d/\zeta\delta)\}\right)$, we have that with probability $1-\zeta$, $\max_{j,l}(|h_{j,l} - \hat{h}_{j,l}|) \leq 0.01 + \alpha$. Using the assumption that $\alpha \leq 0.1$, since $\min(h_{j,k-1}, h_{j,k}, h_{j,k+1}) - 0.11 \geq 0.31 \geq 0.04 + 0.11 \geq \max_{l\neq k-1,k,k+1} h_{j,l} + 0.11$. This implies that with probability $1-\zeta$, the algorithm choose the bin from $k-1, k, k+1$, which means the estimate $|\bar{x}_j - \mu| \leq 4\sigma$. By the tail bound of sub-Gaussian distribution and a union bound over n, d, we have that with probability $1-\zeta$, for all $x_i \in \mathcal{D}$ and $j \in [d]$, $x_{i,j} \in [\bar{x}_j - 8\sigma\sqrt{\log(nd/\zeta)}, \bar{x}_j + 8\sigma\sqrt{\log(nd/\zeta)}]$.

E.3.2. Proofs of the sensitivity of the filtering in Lemma 2.2 and Lemma G.1

Proof of Lemma 2.2. We only need to show that one step of the proposed filter is a contraction. To this end, we only need to show contraction for two datasets at distance 1, i.e., $d_\triangle(\mathcal{D}, \mathcal{D}') = 1$. For fixed (μ, v) and Z, we apply filter to set of scalars $(v^\top(\mathcal{D} - \mu))^2$ and $(v^\top(\mathcal{D}' - \mu))^2$, whose distance is also one. If the entries that are different (say $a \in \mathcal{D}$ and $a' \in \mathcal{D}'$) are both below the subset of the top $2n\alpha$ points (as in Definition D.1), then the same set of points will be removed for both and the distance is preserved $d_\triangle(S(\mathcal{D}), S(\mathcal{D}')) = 1$. If they are both above the top $2n\alpha$ subset, then either both are removed, one of them is removed, or both remain. The rest of the points that are removed coincide in both sets. Hence, $d_\triangle(S(\mathcal{D}), S(\mathcal{D}')) \leq 1$. If a is below and a' is above the top $2n\alpha$ subset of respective datasets, then either a' is not removed (in which case $d_\triangle(S(\mathcal{D}), S(\mathcal{D}')) = 1$) or a' is removed (in which case $S(\mathcal{D}) = S(\mathcal{D}') \cup \{a\}$ and the distance remains one).

Note that when there are ties, it is critical to resolve them in a consistent manner in both datasets $\mathcal{D}$ and $\mathcal{D}'$. The tie breaking rule of Definition D.1 is critical in sorting those samples with the same score τ_i's in a consistent manner.

Proof of Lemma G.1. The analysis of contraction of the filtering step in DPMMWFILTER is analogous to that of private

iterative filtering in Lemma 2.2.

E.3.3. Proof of part 1 of Lemma E.3 on differential privacy of DPFilter

We explicitly write out how many times we access the database and how much privacy is lost each time in an interactive version of DPFilter in Algorithm 1, which performs the same operations as DPFilter. In order to apply Lemma H.13, we cap ε at 0.9 in initializing ε_1. We call q_{mean}, q_{PCA}, q_{norm} and q_{size} T times, each with $(\varepsilon_1, \delta_1)$ guarantee. In total this accounts for (ε, δ) privacy loss, using Lemma H.13 and our choice of ε_1 and δ_1.

This proof is analogous to the proof of DP for DPMMWFilter in Appendix G.1, and we omit the details here. We will assume for now that $|S_r| \geq n/2$ for all $r \in [t]$ and prove privacy. This happens with probability larger than $1 - \delta_1$, hence ensuring the privacy guarantee. In all sub-routines, we run Filter$(\cdot)$ in Algorithm 1 to simulate the filtering process so far and get the current set of samples S_t. Lemma 2.2 allows us to prove privacy of all interactive mechanisms. This shows that the two data datasets S_t and S_t' are neighboring, if they are resulting from the identical filtering but starting from two neighboring datasets $\mathcal{D}_n$ and $\mathcal{D}_n'$. As all four sub-routines are output perturbation mechanisms with appropriately chosen sensitivities, they satisfy the desired $(\varepsilon_1, \delta_1)$-DP guarantees. Further, the probability that $n_t > 3/4n$ and $|S_t| \leq n/2$ is less than δ_1 for $n = \tilde{\Omega}((1/\varepsilon_1)\log(1/\delta_1))$.

E.3.4. Proof of part 2 of Lemma E.3 on accuracy of DPFilter

The following theorem analyzing DPFilter implies the desired Lemma E.3 when the good set is α-subgaussian good, which follows from H.3 and the assumption that $n = \widetilde{\Omega}(d/\alpha^2)$.

Theorem 10 (Anlaysis of DPFilter)**.** *Let S be an α-corrupted sub-Gaussian dataset under Assumption 1, where $\alpha \leq c$ for some universal constant $c \in (0, 1/2)$. Let S_{good} be α-subgaussian good with respect to $\mu \in \mathbb{R}^d$. Suppose $\mathcal{D} = \{x_i \in \bar{x} + [-B/2, B/2]^d\}_{i=1}^n$ be the projected dataset where all of the uncorrupted samples are contained in $\bar{x} + [-B/2, B/2]^d$. If $n = \widetilde{\Omega}\left(d^2 B^3 \log(1/\delta)/(\varepsilon\alpha)\right)$, then* DPFilter *terminates after at most $O\left(dB^2\right)$ iterations and outputs S_t such that with probability 0.9, we have $|S_t \cap S_{\text{good}}| \geq (1 - 10\alpha)n$ and*

$$\|\mu(S_t) - \mu\|_2 \lesssim \alpha\sqrt{\log 1/\alpha}\,.$$

To prove this theorem, we use the following lemma to first show that we do not remove too many uncorrupted samples. The upper bound on the accuracy follows immediately from Lemma H.7 and the stopping criteria of the algorithm.

Lemma E.4. *If $n \gtrsim \frac{B^2 d^{3/2}}{\varepsilon_1 \alpha \log 1/\alpha}\log(1/\delta)$, $\lambda_t \geq (C - 0.01)\cdot \alpha \log 1/\alpha$ and $|S_t \cap S_{\text{good}}| \geq (1 - 10\alpha)n$, then there exists constant $C > 0$ such that for each iteration t, with probability $1 - O(1/d)$, we have Eq. (4) holds. If this condition holds, we have*

$$\mathbb{E}\,|(S_t \setminus S_{t+1}) \cap S_{\text{good}}| \leq \mathbb{E}\,|S_t \setminus S_{t+1} \cap S_{\text{bad}}|\,.$$

We measure the progress by by summing the number of clean samples removed up to iteration t and the number of remaining corrupted samples, defined as $d_t \triangleq |(S_{\text{good}} \cap S) \setminus S_t| + |S_t \setminus (S_{\text{good}} \cap S)|$. Note that $d_1 = \alpha n$, and $d_t \geq 0$. At each iteration, we have

$$\mathbb{E}[d_{t+1} - d_t | d_1, d_2, \cdots, d_t] \quad = \quad \mathbb{E}\left[|S_{\text{good}} \cap (S_t \setminus S_{t+1})| - |S_{\text{bad}} \cap (S_t \setminus S_{t+1})|\right] \;\leq\; 0,$$

from the Lemma E.4. Hence, d_t is a non-negative super-martingale. By optional stopping theorem, at stopping time, we have $\mathbb{E}[d_t] \leq d_1 = \alpha n$. By Markov inequality, d_t is less than $10\alpha n$ with probability 0.9, i.e. $|S_t \cap S_{\text{good}}| \geq (1 - 10\alpha)n$. The desired bound follows from induction and Lemma H.7.

Now we bound the number of iterations under the conditions of Lemma E.5. Let $W_t = |S_t \setminus S_{t-1}|/n$. Since Eq. (5), we have

$$\mathbb{E}[W_t] \geq \frac{1}{n}\sum_{i \in \mathcal{T}_{2\alpha}} \frac{\tau_i}{dB^2} \geq \frac{0.7\|M(S_{t-1}) - \mathbf{I}\|_2}{\alpha d B^2} \geq \frac{0.7C\alpha\log(1/\alpha)}{dB^2}\,.$$

Let T be the stopping time. We know $\sum_{t=1}^T W_t \leq 10\alpha$. By Wald's equation, we have

$$\mathbb{E}[\sum_{t=1}^T W_t] = \mathbb{E}[\sum_{t=1}^T \mathbb{E}[W_t]] \geq \mathbb{E}[T]\frac{0.7C\alpha\log(1/\alpha)}{dB^2}\,.$$

This means $\mathbb{E}[T] \leq (15dB^2)/(C\log(1/\alpha))$. By Markov inequality we know with probability 0.9, we have $T = O(dB^2/\log(1/\alpha))$.

E.3.5. Proof of Lemma E.4

The expected number of removed good points and bad points are proportional to the $\sum_{i\in S_{\rm good}\cap\mathcal{T}_{2\alpha}}\tau_i$ and $\sum_{i\in S_{\rm bad}\cap\mathcal{T}_{2\alpha}}\tau_i$. It suffices to show

$$\sum_{i\in S_{\rm good}\cap\mathcal{T}_{2\alpha}}\tau_i \;\leq\; \sum_{i\in S_{\rm bad}\cap\mathcal{T}_{2\alpha}}\tau_i\,.$$

Assuming we have $\|M(S_{t-1})-I\|_2 \geq C\alpha\log 1/\alpha$ for some $C>0$ sufficiently large, it suffices to show

$$\frac{1}{n}\sum_{i\in S_{\rm bad}\cap\mathcal{T}_{2\alpha}}\tau_i \geq \frac{1}{1000}\|M(S_{t-1})-\mathbf{I}\|_2\,.$$

First of all, we have

$$\begin{aligned}\frac{1}{n}\sum_{i\in S_{t-1}}\tau_i - 1 &= v_t^\top M(S_{t-1})v_t - 1\\ &= v_t^\top\left(M(S_{t-1})-\mathbf{I}\right)v_t\end{aligned}$$

Lemma H.6 shows that the magnitude of the largest eigenvalue of $M(S_{t-1})-\mathbf{I}$ is positive since the magnitudes negative eigenvalues are all less than $c\alpha\log 1/\alpha$. So we have

$$\begin{aligned}\frac{1}{n}\sum_{i\in S_{t-1}}\tau_i - 1 &\geq \|M(S_{t-1})-\mathbf{I}\|_2 - O(\alpha\log 1/\alpha) &(2)\\ &\geq 0.9\|M(S_{t-1})-\mathbf{I}\|_2\,, &(3)\end{aligned}$$

where the first inequality follows from Lemma E.6, and the second inequality follows from our choice of large constant C. The next lemma regularity conditions for τ_i's for each iteration is satisfied.

Lemma E.5. *If $n \gtrsim \frac{B^2d^{3/2}}{\varepsilon_1\alpha\log 1/\alpha}\log(1/\delta)$, then there exists a large constant $C>0$ such that, with probability $1-O(1/d)$, we have*

1. $$\frac{1}{n}\sum_{i\in S_{\rm good}\cap\mathcal{T}_{2\alpha}\cap S_{t-1}}\tau_i \leq \frac{1}{1000}\|M(S_{t-1})-\mathbf{I}\|_2\,. \tag{4}$$

2. *For all $i\notin\mathcal{T}_{2\alpha}$,*
$$\alpha\tau_i \leq \frac{1}{1000}\|M(S_{t-1})-\mathbf{I}\|_2\,.$$

3. $$\frac{1}{n}\sum_{i\in S_{\rm good}\cap S_{t-1}}(\tau_i-1) \leq \frac{1}{1000}\|M(S_{t-1})-\mathbf{I}\|_2\,.$$

Thus, by combining with Lemma E.5, we have

$$\frac{1}{n}\sum_{i\in S_{t-1}\cap S_{\rm bad}}\tau_i \;\geq\; 0.8\|M(S_{t-1})-\mathbf{I}\|_2\,.$$

We now have

$$\begin{aligned}\frac{1}{n}\sum_{i\in S_{\rm bad}\cap\mathcal{T}_{2\alpha}}\tau_i &\geq 0.8\|M(S_{t-1})-\mathbf{I}\|_2-\sum_{i\in S_{\rm bad}\cap S_{t-1}\setminus\mathcal{T}_{2\alpha}}\tau_i\\ &\geq 0.8\|M(S_{t-1})-\mathbf{I}\|_2-\max_{i\in S_{\rm bad}\cap S_{t-1}\setminus\mathcal{T}_{2\alpha}}\alpha\tau_i\\ &\geq 0.8\|M(S_{t-1})-\mathbf{I}\|_2-\frac{1}{1000}\|M(S_{t-1})-\mathbf{I}\|_2 \qquad (5)\\ &\geq \frac{1}{n}\sum_{i\in S_{\rm good}\cap\mathcal{T}_{2\alpha}}\tau_i\,,\end{aligned}$$

which completes the proof.

E.3.6. Proof of Lemma E.5

By our choice of sample complexity n, with probability $1-O(1/dB^2)$, we have $\|\mu(S_{t-1})-\mu_t\|_2^2\lesssim\alpha\log 1/\alpha$, $v_t^\top(M(S_{t-1})-\mathbf{I})v_t\gtrsim\|M(S_{t-1})-\mathbf{I}\|_2-\alpha\log 1/\alpha$ (Lemma E.6), and $\|M(S_{t-1})-\mathbf{I}\|_2\geq C\alpha\log 1/\alpha$ simultaneously hold before stopping.

Lemma E.6. *If*

$$n\gtrsim\frac{d^{3/2}B^2}{\eta\varepsilon_1}\sqrt{2\ln\frac{1.25}{\delta}}\log\frac{1}{\zeta}\,,$$

then with probability $1-\zeta$*, we have*

$$v_t^\top(M(S_{t-1})-\mathbf{I})v_t\geq\|M(S_{t-1})-\mathbf{I}\|_2-2\eta-\frac{2|S_{t-1}|}{n}\|\mu_t-\mu(S_{t-1})\|_2^2$$

We first consider the upper bound of the good points.

$$\begin{aligned}&\frac{1}{n}\sum_{i\in S_{\rm good}\cap\mathcal{T}_{2\alpha}\cap S_{t-1}}\tau_i=\frac{1}{n}\sum_{i\in S_{\rm good}\cap\mathcal{T}_{2\alpha}\cap S_{t-1}}\langle x_i-\mu_t,v_t\rangle^2\\ &\overset{(a)}{\leq}\frac{2}{n}\sum_{i\in S_{\rm good}\cap\mathcal{T}_{2\alpha}\cap S_{t-1}}\langle x_i-\mu,v_t\rangle^2+\frac{2}{n}|S_{\rm good}\cap\mathcal{T}_{2\alpha}\cap S_{t-1}|\langle\mu-\mu_t,v_t\rangle^2\\ &\leq O(\alpha\log 1/\alpha)+\alpha\left(\|\mu-\mu(S_{t-1})\|_2+\|\mu_t-\mu(S_{t-1})\|_2\right)^2\\ &\overset{(b)}{\leq}O(\alpha\log 1/\alpha)+\alpha\left(O(\alpha\sqrt{\log 1/\alpha})+\sqrt{\alpha\left(\|M(S_{t-1})-\mathbf{I}\|_2+O(\alpha\log 1/\alpha)\right)}+O(\sqrt{\alpha\log 1/\alpha})\right)^2\\ &\leq O(\alpha\log 1/\alpha)+\alpha^2\|M(S_{t-1})-\mathbf{I}\|_2\\ &\overset{(c)}{\leq}\frac{1}{1000}\|M(S_{t-1})-I\|_2\end{aligned}$$

where the (a) is implied by the fact that for any vector x,y,z, we have $(x-y)(x-y)^\top\preceq 2(x-z)(x-z)^\top+2(y-z)(y-z)^\top$, (b) follows from Lemma H.7 and c follows from our choice of large constant C.

Since $|S_{\rm bad}\cap\mathcal{T}_{2\alpha}|\leq\alpha n$, we know $|S_{\rm good}\cap\mathcal{T}_{2\alpha}|\geq\alpha n$, so we have for $i\notin\mathcal{T}_{2\alpha}$,

$$\alpha\tau_i\leq\frac{\alpha}{|S_{\rm good}\cap\mathcal{T}_{2\alpha}\cap S_{t-1}|}\sum_{i\in S_{\rm good}\cap\mathcal{T}_{2\alpha}\cap S_{t-1}}\tau_i\leq\frac{1}{1000}\|M(S_{t-1})-\mathbf{I}\|_2\,.$$

Since $|S_{\text{good}} \cap S_{t-1}| \geq (1-10\alpha)n$, we have

$$\frac{1}{n}\sum_{i\in S_{\text{good}}\cap S_{t-1}} \tau_i = \frac{1}{n}\sum_{i\in S_{\text{good}}\cap S_{t-1}} \langle x_i - \mu(S_{t-1}), v_t\rangle^2 \tag{6}$$

$$= \frac{1}{n}\sum_{i\in S_{\text{good}}\cap S_{t-1}} \langle x_i - \mu(S_{\text{good}}\cap S_{t-1}), v_t\rangle^2 + \frac{|S_{\text{good}}\cap S_{t-1}|}{n}\langle \mu(S_{\text{good}}\cap S_{t-1}) - \mu(S_{t-1}), v_t\rangle^2 \tag{7}$$

$$\overset{(a)}{\leq} c\alpha\log 1/\alpha + 1 + \|\mu(S_{\text{good}}\cap S_{t-1}) - \mu(S_{t-1})\|_2^2 \tag{8}$$

$$\leq c\alpha\log 1/\alpha + 1 + (\|\mu(S_{\text{good}}\cap S_{t-1}) - \mu\|_2 + \|\mu - \mu(S_{t-1})\|_2)^2 \tag{9}$$

$$\overset{(b)}{\leq} c\alpha\log 1/\alpha + 1 + \alpha\|M(S_{t-1}) - \mathbf{I}\|_2 + O(\alpha\log 1/\alpha) \tag{10}$$

$$\overset{(c)}{\leq} \frac{1}{1000}\|M(S_{t-1}) - \mathbf{I}\|_2\ , \tag{11}$$

where (a) follows from Lemma H.6, and (b) follows from Lemma H.7, and (c) follows from our choice of large constant C.

E.3.7. Proof of Lemma E.6

Proof. We have following identity.

$$\begin{aligned} &\frac{1}{n}\sum_{i\in S_{t-1}}(x_i-\mu_t)(x_i-\mu_t)^\top \\ =\ &\frac{1}{n}\sum_{i\in S_{t-1}}(x_i-\mu(S_{t-1}))(x_i-\mu(S_{t-1}))^\top + \frac{|S_{t-1}|}{n}(\mu(S_{t-1})-\mu_t)(\mu(S_{t-1})-\mu_t)^\top\ . \end{aligned}$$

So we have,

$$\begin{aligned} &v_t^\top\left(M(S_{t-1})-\mathbf{I}\right)v_t \\ \geq\ &v_t^\top\left(\frac{1}{n}\sum_{i\in S_{t-1}}(x_i-\mu_t)(x_i-\mu_t)^\top - \mathbf{I}\right)v_t - \frac{|S_{t-1}|}{n}\|\mu_t-\mu(S_{t-1})\|_2^2 \\ \geq\ &\|M(S_{t-1})-\mathbf{I}\|_2 - 2\eta - \frac{2|S_{t-1}|}{n}\|\mu_t-\mu(S_{t-1})\|_2^2 \end{aligned}$$

where the last inequality follows from Lemma H.6, which shows that the magnitude of the largest eigenvalue of $M(S_{t-1})-\mathbf{I}$ must be positive. □

F. PRIME: efficient algorithm for private and robust mean estimation

We provide our main algorithms, Algorithm 9 and Algorithm 10, in Appendix F.1 and the corresponding proof in Appendix G. We provide our novel DPTHRESHOLD and its anlysis in Appendix F.2.

We define S_{good} as the original set of n clean samples (as defined in Assumption 1 and 2) and S_{bad} as the set of corrupted samples that replace αn of the clean samples. The (rescaled) covariance is denoted by $M(S^{(s)}) \triangleq (1/n)\sum_{i\in S^{(s)}}(x_i - \mu(S^{(s)}))(x_i-\mu(S^{(s)}))^\top$, where $\mu(S^{(s)}) \triangleq (1/|S^{(s)}|)\sum_{i\in S^{(s)}} x_i$ denotes the mean.

F.1. PRIvate and robust Mean Estimation (PRIME)

Algorithm 9: PRIvate and robust Mean Estimation (PRIME)

Input: $S = \{x_i \in \mathbb{R}^d\}_{i=1}^n$, range $[-R, R]^d$, adversarial fraction $\alpha \in (0, 1/2)$, number of iterations $T_1 = O(\log d), T_2 = O(\log d)$, target privacy (ε, δ)

$(\bar{x}, B) \leftarrow q_{\text{range}}(\{x_i\}_{i=1}^n, R, 0.01\varepsilon, 0.01\delta)$ [Algorithm 5 in Appendix E.3.1]

Clip the data points: $\tilde{x}_i \leftarrow \mathcal{P}_{\bar{x}+[-B/2,B/2]^d}(x_i)$, for all $i \in [n]$

$\hat{\mu} \leftarrow$ DPMMWFILTER$(\{\tilde{x}_i\}_{i=1}^n, \alpha, T_1, T_2, 0.99\varepsilon, 0.99\delta)$ [Algorithm 10]

Output: $\hat{\mu}$

Algorithm 10: Differentially private filtering with matrix multiplicative weights (DPMMWFILTER)

Input: $S = \{x_i \in \bar{x} + [-B/2, B/2]^d\}_{i=1}^n$, $\alpha \in (0, 1/2)$, $T_1 = O(\log(B\sqrt{d})), T_2 = O(\log d)$, privacy (ε, δ)

Initialize $S^{(1)} \leftarrow [n]$, $\varepsilon_1 \leftarrow \varepsilon/(4T_1)$, $\delta_1 \leftarrow \delta/(4T_1)$, $\varepsilon_2 \leftarrow \min\{0.9, \varepsilon\}/(4\sqrt{10T_1T_2\log(4/\delta)})$, $\delta_2 \leftarrow \delta/(20T_1T_2)$, a large enough constant $C > 0$

if $n < (4/\varepsilon_1)\log(1/(2\delta_1))$ **then Output:** $\emptyset$

for epoch $s = 1, 2, \ldots, T_1$ **do**

  $\lambda^{(s)} \leftarrow \|M(S^{(s)}) - \mathbf{I}\|_2 + \text{Lap}(2B^2d/(n\varepsilon_1))$

  $n^{(s)} \leftarrow |S^{(s)}| + \text{Lap}(1/\varepsilon_1)$

  if $n^{(s)} \le 3n/4$ **then Output:** $\emptyset$

  if $\lambda^{(s)} \le C\,\alpha\log(1/\alpha)$ **then**

    Output: $\mu^{(s)} \leftarrow (1/|S^{(s)}|)\big(\sum_{i\in S^{(s)}} x_i\big) + \mathcal{N}(0, (2B\sqrt{2d\log(1.25/\delta_1)}/(n\,\varepsilon_1))^2\mathbf{I}_{d\times d})$

  $\alpha^{(s)} \leftarrow 1/(100(0.1/C + 1.01)\lambda^{(s)})$

  $S_1^{(s)} \leftarrow S^{(s)}$

  for $t = 1, 2, \ldots, T_2$ **do**

    $\lambda_t^{(s)} \leftarrow \|M(S_t^{(s)}) - \mathbf{I}\|_2 + \text{Lap}(2B^2d/(n\varepsilon_2))$

    if $\lambda_t^{(s)} \le 0.5\lambda_0^{(s)}$ **then**

      terminate epoch

    else

      $\Sigma_t^{(s)} \leftarrow M(S_t^{(s)}) + \mathcal{N}(0, (4B^2d\sqrt{2\log(1.25/\delta_2)}/(n\varepsilon_2))^2\mathbf{I}_{d^2\times d^2})$

      $U_t^{(s)} \leftarrow (1/\operatorname{Tr}(\exp(\alpha^{(s)}\sum_{r=1}^t(\Sigma_r^{(s)} - \mathbf{I})))\exp(\alpha^{(s)}\sum_{r=1}^t(\Sigma_r^{(s)} - \mathbf{I}))$

      $\psi_t^{(s)} \leftarrow \left\langle M(S_t^{(s)}) - \mathbf{I}, U_t{}^{(s)}\right\rangle + \text{Lap}(2B^2d/(n\varepsilon_2))$

      if $\psi_t^{(s)} \le (1/5.5)\lambda_t^{(s)}$ **then**

        $S_{t+1}^{(s)} \leftarrow S_t^{(s)}$

      else

        $Z_t^{(s)} \leftarrow \text{Unif}([0, 1])$

        $\mu_t^{(s)} \leftarrow (1/|S_t^{(s)}|)\big(\sum_{i\in S_t} x_i\big) + \mathcal{N}(0, (2B\sqrt{2d\log(1.25/\delta_2)}/(n\,\varepsilon_2)\mathbf{I}_{d\times d})^2)$

        $\rho_t^{(s)} \leftarrow$ DPTHRESHOLD$(\mu_t^{(s)}, U_t^{(s)}, \alpha, \varepsilon_2, \delta_2, S_t^{(s)})$ [Algorithm 11]

        $S_{t+1}^{(s)} \leftarrow S_t^{(s)} \setminus \{i \mid i \in \mathcal{T}_{2\alpha}$ for $\{\tau_j = (x_j - \mu_t^{(s)})^\top U_t^{(s)}(x_j - \mu_t^{(s)})\}_{j\in S_t^{(s)}}$ and $\tau_i \ge \rho_t^{(s)}\,Z_t^{(s)}\}$, where $\mathcal{T}_{2\alpha}$ is defined in Definition D.1.

  $S^{(s+1)} \leftarrow S_t^{(s)}$

Output: $\mu^{(T_1)}$

F.2. Algorithm and analysis of DPTHRESHOLD

Algorithm 11: Differentially private estimation of the threshold (DPTHRESHOLD)

Input: $\mu, U, \alpha \in (0, 1/2)$, target privacy (ε, δ), $S = \{x_i \in \bar{x} + [-B/2, B/2]^d\}$

Set $\tau_i \leftarrow (x_i - \mu)^\top U (x_i - \mu)$ for all $i \in S$

Set $\tilde{\psi} \leftarrow (1/n) \sum_{i \in S} (\tau_i - 1) + \mathrm{Lap}(2B^2 d/n\varepsilon))$

Compute a histogram over geometrically sized bins

$I_1 = [1/4, 1/2), I_2 = [1/2, 1), \ldots, I_{2+\log(B^2 d)} = [2^{\log(B^2 d)-1}, 2^{\log(B^2 d)}]$

$$h_j \leftarrow \frac{1}{n} \cdot |\{i \in S \,|\, \tau_i \in [2^{-3+j}, 2^{-2+j})\}| \,, \qquad \text{for all } j = 1, \ldots, 2 + \log(B^2 d)$$

Compute a privatized histogram $\tilde{h}_j \leftarrow h_j + \mathcal{N}(0, (4\sqrt{2\log(1.25/\delta)}/(n\varepsilon))^2)$, for all $j \in [2 + \log(B^2 d)]$

Set $\tilde{\tau}_j \leftarrow 2^{-3+j}$, for all $j \in [2 + \log(B^2 d)]$

Find the largest $\ell \in [2 + \log(B^2 d)]$ satisfying $\sum_{j \geq \ell} (\tilde{\tau}_j - \tilde{\tau}_\ell)\, \tilde{h}_j \geq 0.31\tilde{\psi}$

Output: $\rho = \tilde{\tau}_\ell$

Lemma F.1 (DPTHRESHOLD: picking threshold privately)**.** *Algorithm* DPTHRESHOLD*$(\mu, U, \alpha, \varepsilon, \delta, S)$ running on a dataset $\{\tau_i = (x_i - \mu)^\top U (x_i - \mu)\}_{i \in S}$ is (ε, δ)-DP. Define $\psi \triangleq \frac{1}{n} \sum_{i \in S} (\tau_i - 1)$. If τ_i's satisfy*

$$\begin{aligned} \frac{1}{n} \sum_{i \in S_{\mathrm{good}} \cap \mathcal{T}_{2\alpha} \cap S} \tau_i &\leq \psi/1000 \\ \frac{1}{n} \sum_{i \in S_{\mathrm{good}} \cap S} (\tau_i - 1) &\leq \psi/1000 \,, \end{aligned}$$

and $n \geq \widetilde{\Omega}\left(\frac{B^2 d \sqrt{\log(1/\delta)}}{\varepsilon\alpha}\right)$, then DPTHRESHOLD *outputs a threshold ρ such that with probability $1 - O(1/\log^3 d)$,*

$$\frac{1}{n} \sum_{\tau_i < \rho} (\tau_i - 1) \;\leq\; 0.75\psi \;\; \textit{and} \tag{12}$$

$$2\Big(\sum_{i \in S_{\mathrm{good}} \cap \mathcal{T}_{2\alpha}} \mathbf{1}\{\tau_i \leq \rho\} \frac{\tau_i}{\rho} + \mathbf{1}\{\tau_i > \rho\}\Big) \leq \sum_{i \in S_{\mathrm{bad}} \cap \mathcal{T}_{2\alpha}} \mathbf{1}\{\tau_i \leq \rho\} \frac{\tau_i}{\rho} + \mathbf{1}\{\tau_i > \rho\} \,. \tag{13}$$

F.3. Proof of Lemma F.1

1. Threshold ρ sufficiently reduces the total score.

Let ρ be the threshold picked by the algorithm. Let $\hat{\tau}_i$ denote the minimum value of the interval of the bin that τ_i belongs to. It holds that

$$\begin{aligned} & \frac{1}{n} \sum_{\tau_i \geq \rho, i \in [n]} (\tau_i - \rho) \geq \frac{1}{n} \sum_{\hat{\tau}_i \geq \rho, i \in [n]} (\hat{\tau}_i - \rho) \\ = \;& \sum_{\tilde{\tau}_j \geq \rho, j \in [2+\log(B^2 d)]} (\tilde{\tau}_j - \rho) h_j \\ \overset{(a)}{\geq} \;& \sum_{\tilde{\tau}_j \geq \rho, j \in [2+\log(B^2 d)]} (\tilde{\tau}_j - \rho) \tilde{h}_j - O\left(\log(B^2 d) \cdot B^2 d \cdot \frac{\sqrt{\log(\log(B^2 d) \log d) \log(1/\delta)}}{\varepsilon n}\right) \\ \overset{(b)}{\geq} \;& 0.31\tilde{\psi} - \tilde{O}(\frac{B^2 d}{\varepsilon n}) \\ \overset{(c)}{\geq} \;& 0.3\psi - \tilde{O}(\frac{B^2 d}{\varepsilon n}) \,, \end{aligned}$$

where (a) holds due to the accuracy of the private histogram (Lemma H.12), (b) holds by the definition of ρ in our algorithm, and (c) holds due to the accuracy of $\tilde{\psi}$. This implies if $\rho < 1$, then $\frac{1}{n}\sum_{\tau_i<\rho}(\tau_i - 1)$ is negative and if $\rho \geq 1$, then

$$\frac{1}{n}\sum_{\tau_i<\rho}(\tau_i - 1) = \psi - \frac{1}{n}\sum_{\tau_i\geq\rho}(\tau_i - 1) \leq \psi - \frac{1}{n}\sum_{\tau_i\geq\rho}(\tau_i - \rho) \leq 0.7\psi + \tilde{O}(B^2 d/\varepsilon n).$$

By Lemma F.2, it holds that

$$\begin{aligned}\frac{1}{n}\sum_{i\in S\setminus\mathcal{T}_{2\alpha}}(\tau_i - 1) &= \psi - \frac{1}{n}\sum_{i\in S_{\text{good}}\cap\mathcal{T}_{2\alpha}}(\tau_i - 1) - \frac{1}{n}\sum_{i\in S_{\text{bad}}\cap\mathcal{T}_{2\alpha}}(\tau_i - 1)\\ &\leq \psi - \frac{1}{n}\sum_{i\in S_{\text{bad}}\cap\mathcal{T}_{2\alpha}}(\tau_i - 1)\\ &\leq (2/1000)\psi\end{aligned}$$

And we conclude that

$$\frac{1}{n}\sum_{\tau_i<\rho \text{ or } i\notin\mathcal{T}_{2\alpha}}(\tau_i - 1) \leq 0.71\psi + \tilde{O}(B^2 d/\varepsilon n) \leq 0.75\psi$$

2. Threshold ρ removes more bad data points than good data points.

Define C_2 to be the threshold such that $\frac{1}{n}\sum_{\tau_i>C_2}(\tau_i - C_2) = (2/3)\psi$. Suppose $2^b \leq C_2 \leq 2^{b+1}$, $\frac{1}{n}\sum_{\hat{\tau}_i\geq 2^{b-1}}(\hat{\tau}_i - 2^{b-1}) \geq (1/3)\psi$ because $\forall\tau_i \geq C_2$, $(\hat{\tau}_i - 2^{b-1}) \geq \frac{1}{2}(\tau_i - C_2)$. Trivially $C_2 \geq 1$ due to the fact that $\frac{1}{n}\sum_{\tau_i\geq 1}\tau_i - 1 \geq \psi$. Then we have the threshold picked by the algorithm $\rho \geq 2^{b-1}$, which implies $\rho \geq \frac{1}{4}C_2$. Suppose $\rho < C_2$, since $\rho \geq \frac{1}{4}C_2$, we have

$$\begin{aligned}\Big(\sum_{i\in S_{\text{bad}}\cap\mathcal{T}_{2\alpha},\tau_i<\rho}\tau_i + \sum_{i\in S_{\text{bad}}\cap\mathcal{T}_{2\alpha},\tau_i\geq\rho}\rho\Big) &\geq \frac{1}{4}\Big(\sum_{i\in S_{\text{bad}}\cap\mathcal{T}_{2\alpha},\tau_i<C_2}\tau_i + \sum_{i\in S_{\text{bad}}\cap\mathcal{T}_{2\alpha},\tau_i\geq C_2}C_2\Big)\\ &\overset{(a)}{\geq} \frac{10}{4}\Big(\sum_{i\in S_{\text{good}}\cap\mathcal{T}_{2\alpha},\tau_i<C_2}\tau_i + \sum_{i\in S_{\text{good}}\cap\mathcal{T}_{2\alpha},\tau_i\geq C_2}C_2\Big)\\ &\overset{(b)}{\geq} \frac{10}{4}\Big(\sum_{i\in S_{\text{good}}\cap\mathcal{T}_{2\alpha},\tau_i<\rho}\tau_i + \sum_{i\in S_{\text{good}}\cap\mathcal{T}_{2\alpha},\tau_i>=\rho}\rho\Big),\end{aligned}$$

where (a) holds by Lemma F.3, and (b) holds since $\rho \leq C_2$. If $\rho \geq C_2$, the statement of the Lemma F.3 directly implies Equation (13).

Lemma F.2. *[Conditions for τ_i's] Suppose*

$$\frac{1}{n}\sum_{i\in S_{\text{good}}\cap S}(\tau_i - 1) \leq \psi/1000$$
$$\frac{1}{n}\sum_{i\in S_{\text{good}}\cap\mathcal{T}_{2\alpha}}\tau_i \leq \psi/1000$$

then, we have

$$\begin{aligned}\alpha\tau_{2\alpha n} &\leq \psi/1000\\ \frac{1}{n}\sum_{i\in S_{\text{bad}}\cap\mathcal{T}_{2\alpha}}(\tau_i - 1) &\geq (998/1000)\psi\end{aligned}$$

Proof. Since $|S_{\text{good}}\cap\mathcal{T}_{2\alpha}| \geq \alpha n$, it holds

$$\alpha\tau_{2\alpha n} \leq \psi/1000.$$

$$\begin{aligned}\frac{1}{n}\sum_{i\in S_{\rm bad}\cap\mathcal{T}_{2\alpha}}(\tau_i-1) &= \frac{1}{n}\sum_{i\in S_{\rm bad}\cap S}(\tau_i-1)-\frac{1}{n}\sum_{i\in S_{\rm bad}\cap S\setminus\mathcal{T}_{2\alpha}}(\tau_i-1)\\ &\geq (999/1000)\psi-\frac{1}{n}\sum_{i\in S_{\rm bad}\cap S\setminus\mathcal{T}_{2\alpha}}(\tau_i-1)\\ &\geq (999/1000)\psi-(1/1000)\psi\\ &= (998/1000)\psi\end{aligned}$$

□

Lemma F.3. *Assuming that the conditions in Lemma F.2 holds, and for any C such that*

$$\frac{1}{n}\sum_{i\in S,\tau_i<C}(\tau_i-1)+\frac{1}{n}\sum_{i\in S,\tau_i\geq C}(C-1)\geq(1/3)\psi\,,$$

we have

$$\sum_{i\in S_{\rm bad}\cap\mathcal{T}_{2\alpha},\tau_i<C}\tau_i+\sum_{i\in S_{\rm bad}\cap\mathcal{T}_{2\alpha},\tau_i\geq C}C\geq 10(\sum_{i\in S_{\rm good}\cap\mathcal{T}_{2\alpha},\tau_i<C}\tau_i+\sum_{i\in S_{\rm good}\cap\mathcal{T}_{2\alpha},\tau_i\geq C}C)$$

Proof. First we show an upper bound on $S_{\rm good}\cap\mathcal{T}_{2\alpha}$:

$$\frac{1}{n}\sum_{i\in S_{\rm good}\cap\mathcal{T}_{2\alpha},\tau_i<C}\tau_i+\frac{1}{n}\sum_{i\in S_{\rm good}\cap\mathcal{T}_{2\alpha},\tau_i\geq C}C\leq\frac{1}{n}\sum_{i\in S_{\rm good}\cap\mathcal{T}_{2\alpha}}\tau_i\leq\psi/1000.$$

Then we show an lower bound on $S_{\rm bad}\cap\mathcal{T}_{2\alpha}$:

$$\begin{aligned}&\frac{1}{n}\sum_{i\in S_{\rm bad}\cap S,\tau_i<C}(\tau_i-1)+\frac{1}{n}\sum_{i\in S_{\rm bad}\cap S,\tau_i>C}(C-1)\\ =\;&\frac{1}{n}\sum_{i\in S,\tau_i<C}(\tau_i-1)+\frac{1}{n}\sum_{i\in S,\tau_i\geq C}(C-1)\\ &-(\frac{1}{n}\sum_{i\in S_{\rm good}\cap S,\tau_i<C}(\tau_i-1)+\frac{1}{n}\sum_{i\in S_{\rm good}\cap S,\tau_i\geq C}(C-1))\\ \geq\;&(1/3-1/1000)\psi\,.\end{aligned}$$

We have

$$\begin{aligned}&\frac{1}{n}\sum_{i\in S_{\rm bad}\cap\mathcal{T}_{2\alpha},\tau_i<C}\tau_i+\frac{1}{n}\sum_{i\in S_{\rm bad}\cap\mathcal{T}_{2\alpha},\tau_i>C}C\geq\frac{1}{n}\sum_{i\in S_{\rm bad}\cap\mathcal{T}_{2\alpha},\tau_i<C}(\tau_i-1)+\frac{1}{n}\sum_{i\in S_{\rm bad}\cap\mathcal{T}_{2\alpha},\tau_i>C}(C-1)\\ &=\frac{1}{n}\sum_{i\in S_{\rm bad}\cap S,\tau_i<\rho}(\tau_i-1)+\frac{1}{n}\sum_{i\in S_{\rm bad}\cap S,\tau_i>C}(C-1)\\ &-\left(\frac{1}{n}\sum_{i\in S_{\rm bad}\cap S\setminus\mathcal{T}_{2\alpha},\tau_i<C}(\tau_i-1)+\frac{1}{n}\sum_{i\in S_{\rm bad}\cap S\setminus\mathcal{T}_{2\alpha},\tau_i>C}(C-1)\right)\\ &\geq(1/3-1/1000)\psi-\alpha\tau_{2\alpha n}\\ &\geq(1/3-2/1000)\psi\end{aligned}$$

Combing the lower bound and the upper bound yields the desired statement □

G. The analysis of PRIME and the proof of Theorem 6

G.1. Proof of part 1 of Theorem 6 on differential privacy

Let $(\varepsilon_0, \delta_0)$ be the end-to-end target privacy guarantee. The $(0.01\varepsilon_0, 0.01\delta_0)$-DP guarantee of q_{range} follows from Lemma E.2. We are left to show that DPMMWFILTER in Algorithm 10 satisfy $(0.99\varepsilon_0, 0.99\delta_0)$-DP. To this end, we explicitly write out how many times we access the database and how much privacy is lost each time in an interactive version of DPMMWFILTER in Algorithm 13, which performs the same operations as DPMMWFILTER.

In order to apply Lemma H.13, we cap ε at 0.9 in initializing ε_2. We call q_{spectral} and q_{size} T_1 times, each with $(\varepsilon_1, \delta_1)$ guarantee. In total this accounts for $(0.5\varepsilon, 0.5\delta)$ privacy loss. The rest of the mechanisms are called $5T_1T_2$ times ($q_{\text{spectral}}(\cdot)$ and $q_{\text{MMW}}(\cdot)$ each call two DP mechanisms internally), each with $(\varepsilon_2, \delta_2)$ guarantee. In total this accounts for $(0.5\varepsilon, 0.5\delta)$ privacy loss. Altogether, this is within the privacy budget of $(\varepsilon = 0.99\varepsilon_0, \delta = 0.99\delta_0)$.

We are left to show privacy of q_{spectral}, q_{MMW}, and q_{1Dfilter}, and q_{size} in Algorithm 12. We will assume for now that $|S_r^{(\ell)}| \geq n/2$ for all $\ell \in [T_1]$ and $r \in [T_2]$ and prove privacy. We show in the end that this happens with probability larger than $1 - \delta_1$. In all sub-routines, we run Filter$(\cdot)$ in Algorithm 12 to simulate the filtering process so far and get the current set of samples $S_{t_s}^{(s)}$. The following main technical lemma allows us to prove privacy of all interactive mechanisms. This is a counterpart of Lemma 2.2 used for DPFILTER. We provide a proof in Appendix E.3.2.

Lemma G.1. *Let $S(\mathcal{D}_n) \subseteq \mathcal{D}_n$ denote the output of the simulated filtering process* Filter$(\cdot)$ *on $\mathcal{D}_n$ for a given set of parameters $(\{\{\Psi_r^{(\ell)}\}_{r\in[t_\ell]}\}_{\ell\in[s]}, \{(\mu^{(\ell)}, \lambda^{(\ell)})\}_{\ell\in[s]})$ in Algorithm 12. Then we have $d_\triangle(S(\mathcal{D}_n), S(\mathcal{D}')_n) \leq d_\triangle(\mathcal{D}_n, \mathcal{D}'_n)$, where $d_\triangle(\mathcal{D}, \mathcal{D}') \triangleq \max\{|\mathcal{D} \setminus \mathcal{D}'|, |\mathcal{D}' \setminus \mathcal{D}|\}$.*

This is a powerful tool for designing private mechanisms, as it guarantees that we can safely simulate the filtering process with privatized parameters and preserve the neighborhood of the dataset; if $\mathcal{D}_n \sim \mathcal{D}'_n$ are neighboring (i.e., $d_\Delta(\mathcal{D}_n, \mathcal{D}'_n) \leq 1$) then so are the filtered pair $S(\mathcal{D}_n)$ and $S(\mathcal{D}'_n)$ (i.e., $d_\Delta(S(\mathcal{D}_n), S(\mathcal{D}'_n)) \leq 1$). Note that in all the interactive mechanisms in Algorithm 12, the noise we need to add is proportional to the set sensitivity of Filter$(\cdot)$ defined as $\Delta_{\text{set}} \triangleq \max_{\mathcal{D}_n \sim \mathcal{D}'_n} d_\Delta(S(\mathcal{D}_n), S(\mathcal{D}'_n))$. If the repeated application of the Filter$(\cdot)$ is not a contraction in $d_\Delta(\cdot, \cdot)$, this results in a sensitivity blow-up. Fortunately, the above lemma ensures contraction of the filtering, proving that $\Delta_{\text{set}} = 1$. Hence, it is sufficient for us to prove privacy for two neighboring filtered sets $S \sim S'$ (as opposed to proving privacy for two neighboring original datasets before filtering $\mathcal{D}_n \sim \mathcal{D}'_n$).

In q_{spectral}, λ satisfy $(\varepsilon, 0)$-DP as the L_1 sensitivity is $\Delta_1 = (1/n)B^2 d$ (Definition 1.2) and we add $\text{Lap}(\Delta_1/\varepsilon)$. The release of μ also satisfy (ε, δ)-DP as the L_2 sensitivity is $\Delta_2 = 2B\sqrt{d}/n$, assuming $|S| \geq n/2$ as ensured by the stopping criteria, and we add $\mathcal{N}(0, \Delta_2(2\log(1.25/\delta))/\varepsilon)^2\mathbf{I})$. Note that in the outer loop call of q_{spectral}, we only release μ once in the end, and hence we count q_{spectral} as one access. On the other hand, in the inner loop, we use both μ and λ from q_{spectral} so we count it as two accesses.

In q_{size}, the returned set size $(\varepsilon, 0)$-DP as the L_1 sensitivity is $\Delta_1 = 1$ and we add $\text{Lap}(\Delta_1/\varepsilon)$. One caveat is that we need to ensure that the stopping criteria of checking $n^{(s)} > 3n/4$ ensures that $|S_t^{(s)}| > n/2$ with probability at least $1 - \delta_1$. This guarantees that the rest of the private mechanisms can assume $|S_t^{(s)}| > n/2$ in analyzing the sensitivity. Since Laplace distribution follows $f(z) = (\varepsilon/2)e^{-\varepsilon|z|}$, we have $\mathbb{P}(n^{(s)} > 3n/4$ and $|S_t^{(s)}| < n/2) \leq (1/2)e^{-n\varepsilon/4}$. Hence, the desired privacy is ensured for $(1/2)e^{-n\varepsilon/4} \leq \delta_1$ (i.e., $n \geq (4/\varepsilon_1)\log(1/(2\delta_1))$).

In q_{MMW}, Σ is (ε, δ)-DP as the L_2 sensitivity is $\Delta_2 = B^2 d/n$, and we add $\mathcal{N}(0, \Delta_2(2\log(1.25/\delta))/\varepsilon)^2\mathbf{I})$. ψ is $(\varepsilon, 0)$-DP as the L_1 sensitivity is $\Delta_1 = 2B^2 d/n$ and we add $\text{Lap}(\Delta_1/\varepsilon)$. This is made formal in the following theorem with a proof. in Appendix G.1.1. This algorithm is identical to the MOD-SULQ algorithm introduced in (Blum et al., 2005) and analyzed in (**?**)Theorem 5]PPCA, up to the choice of the noise variance. But a tighter analysis improves over the MOD-SULQ analysis from (Chaudhuri et al., 2013) by a factor of d in the variance of added Gaussian noise as noted in (Dwork et al., 2014).

Lemma G.2 (Differentially Private PCA)**.** *Consider a dataset $\{x_i \in \mathbb{R}^d\}_{i=1}^n$. If $\|x_i\|_2 \leq 1$ for all $i \in [n]$, the following privatized second moment matrix satisfies (ε, δ)-differential privacy:*

$$\frac{1}{n}\sum_{i=1}^{n} x_i x_i^\top + Z \,,$$

with $Z_{i,j} \sim \mathcal{N}(0, (\,(1/(n\varepsilon))\sqrt{2\log(1.25/\delta)}\,)^2)$ for $i \geq j$ and $Z_{i,j} = Z_{j,i}$ for $i < j$.

In q_{1Dfilter}, the (ε, δ) differential privacy follows from that of DPTHRESHOLD proved in Lemma F.1.

G.1.1. PROOF OF LEMMA G.2

Consider neighboring two databases $\mathcal{D} = \{x_i\}_{i=1}^n$ and $\tilde{\mathcal{D}} = \mathcal{D} \cup \{\tilde{x}_n\} \setminus \{x_n\}$, and let $A = (1/n)\sum_{x_i \in \mathcal{D}} x_i x_i^\top$ and $\tilde{A} = (1/n)\sum_{x_i \in \tilde{\mathcal{D}}} x_i x_i^\top$. Let B and $\tilde{B}$ be the Gaussian noise matrix with β^2 as variance. Let $G = A + B$ and $\tilde{G} = \tilde{A} + \tilde{B}$. At point H, we have

$$\begin{aligned}\ell_{D,\tilde{D}} \;=\; \log \frac{f_G(H)}{f_{\tilde{G}}(H)} \quad &= \quad \sum_{1\le i\le j\le d} \left(-\frac{1}{2\beta^2}\left(H_{ij} - A_{ij}\right)^2 + \frac{1}{2\beta^2}\left(H_{ij} - \hat{A}_{ij}\right)^2 \right) \\ &= \quad \frac{1}{2\beta^2} \sum_{1\le i\le j\le d} \left(\frac{2}{n}\left(H_{ij} - A_{ij}\right)\left(x_{n,i}x_{n,j} - \hat{x}_{n,i}\hat{x}_{n,j}\right) + \frac{1}{n^2}\left(\hat{x}_{n,i}\hat{x}_{n,j} - x_{n,i}x_{n,j}\right)^2 \right) .\end{aligned}$$

Since $\|x_n\|_2 \le 1$ and $\|\tilde{x}_n\|_2 \le 1$, we have $\sum_{1\le i\le j\le d}\left(\hat{x}_{n,i}\hat{x}_{n,j} - x_{n,i}x_{n,j}\right)^2 = 1/2\|\tilde{x}_n\tilde{x}_n^\top - x_n x_n^\top\|_F^2 \le 2$.

Now we bound the first term,

$$\begin{aligned}2\sum_{1\le i\le j\le d}\left(H_{ij} - A_{ij}\right)\left(x_{n,i}x_{n,j} - \hat{x}_{n,i}\hat{x}_{n,j}\right) \quad &= \quad \left\langle H - A, x_n x_n^\top - \tilde{x}_n\tilde{x}_n^\top \right\rangle \\ &= \quad x_n^\top B x_n - \tilde{x}_n^\top B \tilde{x}_n \\ &\le \quad 2\|B\|_2 .\end{aligned}$$

So we have $|\ell_{D,\tilde{D}}| \le \varepsilon$ whenever $\|B\|_2 \le n\varepsilon\beta^2 - 1/n$.

For any fixed unit vector $\|v\|_2 = 1$, we have

$$v^\top B v = 2\sum_{1\le i\le j\le d} B_{ij}v_i v_j \sim \mathcal{N}(0, 2\sum_{1\le i\le j\le d} v_i^2 v_j^2) \;=\; \mathcal{N}(0,1) .$$

Then we have

$$\begin{aligned}\mathbb{P}\left(|\ell_{D,\tilde{D}}| \ge \varepsilon\right) \quad &\le \quad \mathbb{P}\left(\|B\|_2 \ge n\varepsilon\beta^2 - 1/n\right) \\ &= \quad \mathbb{P}\left(\mathcal{N}(0,1) \ge n\varepsilon\beta^2 - \frac{1}{n}\right) \\ &= \quad \Phi\left(\frac{1}{n} - n\varepsilon\beta^2\right) ,\end{aligned}$$

where Φ is CDF of standard Gaussian. According to Gaussian mechanism, if $\beta = (1/(n\varepsilon))\sqrt{2\log(1.25/\delta)}$, we have $\Phi\left(\frac{1}{n} - n\varepsilon\beta^2\right) \le \delta$.

G.2. Proof of part 2 of Theorem 6 on accuracy

The accuracy of PRIME follows from the fact that q_{range} returns a hypercube that contains all the clean data with high probability (Lemma E.2) and that DPMMWFILTER achieves the desired accuracy (Theorem 11) if the original uncorrupted dataset S_{good} is α-subgaussian good. S_{good} is α-subgaussian good if we have $n = \widetilde{\Omega}(d/\alpha^2)$ as shown in Lemma H.3. We present the proof of Theorem 11 below.

Theorem 11 (Analysis of accuracy of DPMMWFILTER)**.** *Let S be an α-corrupted sub-Gaussian dataset, where $\alpha \le c$ for some universal constant $c \in (0, 1/2)$. Let S_{good} be α-subgaussian good with respect to $\mu \in \mathbb{R}^d$. Suppose $\mathcal{D} = \{x_i \in \bar{x} + [-B/2, B/2]^d\}_{i=1}^n$ be the projected dataset. If $n \ge \widetilde{\Omega}\left(\frac{d^{3/2}B^2\log(2/\delta)}{\varepsilon\alpha\log 1/\alpha}\right)$, then* DPMMWFILTER *terminates after at most $O(\log dB^2)$ epochs and outputs $S^{(s)}$ such that with probability 0.9, we have $|S_t^{(s)} \cap S_{\text{good}}| \ge (1 - 10\alpha)n$ and*

$$\|\mu(S^{(s)}) - \mu\|_2 \lesssim \alpha\sqrt{\log 1/\alpha} .$$

Moreover, each epoch runs for at most $O(\log d)$ iterations.

Algorithm 12: Interactive differentially private mechanisms for DPMMWFILTER

$q_{\text{spectral}}(\{\{\Psi_r^{(\ell)}\}_{r\in[t_\ell]}\}_{\ell\in[s]}, \{(\mu^{(\ell)}, \lambda^{(\ell)})\}_{\ell\in[s]}, \varepsilon, \delta)$**:**
  $S \leftarrow \text{Filter}(\{\{\Psi_r^{(\ell)}\}_{r\in[t_\ell]}\}_{\ell\in[s]}, \{(\mu^{(\ell)}, \lambda^{(\ell)})\}_{\ell\in[s]}, \varepsilon, \delta)$
  $\mu \leftarrow (1/|S|)\big(\sum_{i\in S} x_i\big) + \mathcal{N}(0, (2B\sqrt{2d\log(1.25/\delta)}/(n\varepsilon))^2\mathbf{I})$
  $\lambda \leftarrow \|M(S) - \mathbf{I}\|_2 + \text{Lap}(2B^2 d/(n\varepsilon))$
  return (μ, λ)

$q_{\text{size}}(\{\{\Psi_r^{(\ell)}\}_{r\in[t_\ell]}\}_{\ell\in[s]}, \{(\mu^{(\ell)}, \lambda^{(\ell)})\}_{\ell\in[s]}, \varepsilon, \delta)$**:**
  $S \leftarrow \text{Filter}(\{\{\Psi_r^{(\ell)}\}_{r\in[t_\ell]}\}_{\ell\in[s]}, \{(\mu^{(\ell)}, \lambda^{(\ell)})\}_{\ell\in[s]}, \varepsilon, \delta)$
  return $|S| + \text{Lap}(1/\varepsilon)$

$q_{\text{MMW}}(\{\{\Psi_r^{(\ell)}\}_{r\in[t_\ell]}\}_{\ell\in[s]}, \{(\mu^{(\ell)}, \lambda^{(\ell)})\}_{\ell\in[s]}, \alpha^{(s)}, \mu_t^{(s)}, \varepsilon, \delta)$**:**
  $S \leftarrow \text{Filter}(\{\{\Psi_r^{(\ell)}\}_{r\in[t_\ell]}\}_{\ell\in[s]}, \{(\mu^{(\ell)}, \lambda^{(\ell)})\}_{\ell\in[s]}, \varepsilon, \delta)$
  $\Sigma_{t_s+1}^{(s)} \leftarrow M(S) + \mathcal{N}(0, (4B^2 d\sqrt{2\log(1.25/\delta)}/(n\varepsilon))^2\mathbf{I})$
  $U \leftarrow (1/\text{Tr}(\exp(\alpha^{(s)}\sum_{r=1}^{t_s+1}(\Sigma_r^{(s)} - \mathbf{I})))) \exp(\alpha^{(s)}\sum_{r=1}^{t_s+1}(\Sigma_r^{(s)} - \mathbf{I}))$
  $\psi \leftarrow \langle M(S) - \mathbf{I}, U\rangle + \text{Lap}(2B^2 d/(n\varepsilon))$
  return $(\Sigma_{t_s+1}^{(s)}, U, \psi)$

$q_{\text{1Dfilter}}(\{\{\Psi_r^{(\ell)}\}_{r\in[t_\ell]}\}_{\ell\in[s]}, \{(\mu^{(\ell)}, \lambda^{(\ell)})\}_{\ell\in[s]}, \mu, U, \alpha, \varepsilon, \delta)$**:**
  $S \leftarrow \text{Filter}(\{\{\Psi_r^{(\ell)}\}_{r\in[t_\ell]}\}_{\ell\in[s]}, \{(\mu^{(\ell)}, \lambda^{(\ell)})\}_{\ell\in[s]}, \varepsilon, \delta)$
  return $\rho \leftarrow \text{DPTHRESHOLD}(\mu, U, \alpha, \varepsilon, \delta, S)$

$\text{Filter}(\{\{\Psi_r^{(\ell)}\}_{r\in[t_\ell]}\}_{\ell\in[s]}, \{(\mu^{(\ell)}, \lambda^{(\ell)})\}_{\ell\in[s]})$**:**
  $S^{(1)} \leftarrow [n]$
  for epoch $\ell = 1, \ldots, s$ **do**
    $\alpha^{(\ell)} \leftarrow 1/(100(0.1/C + 1.01)\lambda^{(\ell)})$
    $S_1^{(\ell)} \leftarrow S^{(\ell)}$
    for $r = 1, \ldots, t_s$ **do**
      $S_{r+1}^{(\ell)} \leftarrow S_r^{(\ell)} \setminus \{i \,|\, i \in \mathcal{T}_{2\alpha} \text{ for } \{\tau_j = (x_j - \mu_r^{(\ell)})^\top U_r^{(\ell)}(x_j - \mu_r^{(\ell)})\}_{j\in S_r^{(\ell)}} \text{ and } \tau_i \geq \rho_r^{(\ell)}\, Z_r^{(\ell)}\}$, where $\mathcal{T}_{2\alpha}$ is defined in Definition D.1.
  Output: $S_{t_s}^{(s)}$

Proof. In $s = O(\log_{0.98}((C\alpha\log(1/\alpha))/\|M(S^{(1)}) - \mathbf{I}\|_2))$ epochs, following Lemma G.3 guarantees that we find a candidate set $S^{(s)}$ of samples with $\|M(S^{(s)} - \mathbf{I}\|_2 \leq C\alpha\log(1/\alpha)$. We provide proof of Lemma G.3 in the Appendix G.3.

Lemma G.3. *Let S be an α-corrupted sub-Gaussian dataset under Assumption 1. For an epoch $s \in [T_1]$ and an iteration $t \in [T_2]$, under the hypotheses of Lemma G.4, if S_{good} is α-subgaussian good with respect to $\mu \in \mathbb{R}^d$ as in Definition H.2, $n = \widetilde{\Omega}(d^{3/2}\log(1/\delta)/(\varepsilon\alpha))$, and $|S_t^{(s)} \cap S_{\text{good}}| \geq (1 - 10\alpha)n$ then with probability $1 - O(1/\log^3 d)$ the conditions in Eqs. (14) and (15) hold. When these two conditions hold, more corrupted samples are removed in expectation than the uncorrupted samples, i.e., $\mathbb{E}|(S_t^{(s)} \setminus S_{t+1}^{(s)}) \cap S_{\text{good}}| \leq \mathbb{E}|(S_t^{(s)} \setminus S_{t+1}^{(s)}) \cap S_{\text{bad}}|$. Further, for an epoch $s \in [T_1]$ there exists a constant $C > 0$ such that if $\|M(S^{(s)}) - \mathbf{I}\|_2 \geq C\,\alpha\log(1/\alpha)$, then with probability $1 - O(1/\log^2 d)$, the s-th epoch terminates after $O(\log d)$ iterations and outputs $S^{(s+1)}$ such that $\|M(S^{(s+1)}) - \mathbf{I}\|_2 \leq 0.98\|M(S^{(s)}) - \mathbf{I}\|_2$.*

Lemma H.7 ensures that we get the desired bound of $\|\mu(S^{(s)}) - \mu\|_2 = O(\alpha\sqrt{\log(1/\alpha)})$ as long as $S^{(s)}$ has enough clean data, i.e., $|S^{(s)} \cap S_{\text{good}}| \geq n(1 - \alpha)$. Since Lemma G.3 gets invoked at most $O((\log d)^2)$ times, we can take a union bound, and the following argument conditions on the good events in Lemma G.3 holding, which happens with probability at least 0.99. To turn the average case guarantee of Lemma G.3 into a constant probability guarantee, we apply the optional stopping theorem. Recall that the s-th epoch starts with a set $S^{(s)}$ and outputs a filtered set $S_t^{(s)}$ at the t-th inner iteration. We measure the progress by by summing the number of clean samples removed up to epoch s and iteration t and the number of remaining corrupted samples, defined as $d_t^{(s)} \triangleq |(S_{\text{good}} \cap S^{(1)}) \setminus S_t^{(s)}| + |S_t^{(s)} \setminus (S_{\text{good}} \cap S^{(1)})|$. Note that $d_1^{(1)} = \alpha n$,

Algorithm 13: Interactive version of DPMMWFILTER

Input: $\alpha \in (0,1), T_1, T_2, \varepsilon_1 = \varepsilon/(4T_1)$, $\delta_1 = \delta/(4T_1)$, $\varepsilon_2 = \min\{0.9, \varepsilon\}/(4\sqrt{10T_1T_2\log(4/\delta)})$, $\delta_2 = \delta/(20T_1T_2)$

if $n < (4/\varepsilon_1)\log(1/(2\delta_1))$ **then Output:** $\emptyset$

for epoch $s = 1, 2, \ldots, T_1$ **do**

- $(\mu^{(s)}, \lambda^{(s)}) \leftarrow q_{\text{spectral}}(\{\{\Psi_r^{(\ell)}\}_{r\in[t_\ell]}\}_{\ell\in[s-1]}, \{(\mu^{(\ell)}, \lambda^{(\ell)})\}_{\ell\in[s-1]}, \varepsilon_1, \delta_1)$
- $n^{(s)} \leftarrow q_{\text{size}}(\{\{\Psi_r^{(\ell)}\}_{r\in[t_\ell]}\}_{\ell\in[s-1]}, \{(\mu^{(\ell)}, \lambda^{(\ell)})\}_{\ell\in[s-1]}, \varepsilon_1, \delta_1)$
- **if** $n^{(s)} \le 3n/4$ **then** terminate
- **if** $\lambda^{(s)} \le C\alpha\log(1/\alpha)$ **then**
  - **Output:** $\mu^{(s)}$
- $\alpha^{(s)} \leftarrow 1/(100(0.1/C + 1.01)\lambda^{(s)})$
- $t_s \leftarrow 0$
- **for** $t = 1, 2, \ldots, T_2$ **do**
  - $(\mu_t^{(s)}, \lambda_t^{(s)}) \leftarrow q_{\text{spectral}}(\{\{\Psi_r^{(\ell)}\}_{r\in[t_\ell]}\}_{\ell\in[s]}, \{(\mu^{(\ell)}, \lambda^{(\ell)})\}_{\ell\in[s]}, \varepsilon_2, \delta_2)$
  - **if** $\lambda_t^{(s)} \le 0.5\lambda^{(s)}$ **then**
    - terminate epoch
  - **else**
    - $(\Sigma_t^{(s)}, U_t^{(s)}, \psi_t^{(s)}) \leftarrow q_{\text{PMMW}}(\{\{\Psi_r^{(\ell)}\}_{r\in[t_\ell]}\}_{\ell\in[s]}, \{(\mu^{(\ell)}, \lambda^{(\ell)})\}_{\ell\in[s]}, \alpha^{(s)}, \mu_t^{(s)}, \varepsilon_2, \delta_2)$
    - **if** $\psi_t^{(s)} \le (1/5.5)\lambda_t^{(s)}$ **then**
      - $\alpha_t^{(s)} \leftarrow 0$
    - **else**
      - $Z_t^{(s)} \leftarrow \text{Unif}([0,1])$
      - $\rho_t^{(s)} \leftarrow q_{\text{1Dfilter}}(\{\{\Psi_r^{(\ell)}\}_{r\in[t_\ell]}\}_{\ell\in[s]}, \{(\mu^{(\ell)}, \lambda^{(\ell)})\}_{\ell\in[s]}, \mu_t^{(s)}, U_t^{(s)}, \alpha, \varepsilon_2, \delta_2)$
      - $\alpha_t^{(s)} \leftarrow \alpha$
  - $\Psi_t^{(s)} \leftarrow (\mu_t^{(s)}, \lambda_t^{(s)}, \Sigma_t^{(s)}, U_t^{(s)}, \psi_t^{(s)}, Z_t^{(s)}, \rho_t^{(s)}, \alpha_t^{(s)})$
  - $t_s \leftarrow t$

Output: $\mu_{t_{T_1}}^{(T_1)}$

and $d_t^{(s)} \ge 0$. At each epoch and iteration, we have

$$\mathbb{E}[d_{t+1}^{(s)} - d_t^{(s)} | d_1^{(1)}, d_2^{(1)}, \cdots, d_t^{(s)}] \;=\; \mathbb{E}\left[|S_{\text{good}} \cap (S_t^{(s)} \setminus S_{t+1}^{(s)})| - |S_{\text{bad}} \cap (S_t^{(s)} \setminus S_{t+1}^{(s)})|\right] \;\le\; 0,$$

from part 1 of Lemma G.3. Hence, $d_t^{(s)}$ is a non-negative super-martingale. By the optional stopping theorem, at stopping time, we have $\mathbb{E}[d_t^{(s)}] \le d_1^{(1)} = \alpha n$. By the Markov inequality, $d_t^{(s)}$ is less than $10\alpha n$ with probability 0.9, i.e., $|S_t^{(s)} \cap S_{\text{good}}| \ge (1 - 10\alpha)n$. The desired bound in Theorem 11 follows from Lemma H.7.

□

G.3. Proof of Lemma G.3

Lemma G.3 is a combination of Lemma G.4 and Lemma G.5. We state the technical lemmas and subsequently provide the proofs.

Lemma G.4. *For an epoch s and an iteration t such that $\lambda^{(s)} > C\alpha\log(1/\alpha)$, $\lambda_t^{(s)} > 0.5\lambda_0^{(s)}$, and $n^{(s)} > 3n/4$, if $n \gtrsim \frac{B^2(\log B)d^{3/2}\log(1/\delta)}{\varepsilon\alpha}$ and $|S_t^{(s)} \cap S_{\text{good}}| \ge (1 - 10\alpha)n$ then with probability $1 - O(1/\log^3 d)$, the conditions in Eqs. (14) and (15) hold. When these two conditions hold we have $\mathbb{E}|S_t^{(s)} \setminus S_{t+1}^{(s)} \cap S_{\text{good}}| \le \mathbb{E}|S_t^{(s)} \setminus S_{t+1}^{(s)} \cap S_{\text{bad}}|$. If $n \gtrsim \frac{B^2(\log B)d^{3/2}\log(1/\delta)}{\varepsilon\alpha}$, $\psi_t^{(s)} > \frac{1}{5.5}\lambda_t^{(s)}$, and $n^{(s)} > 3n/4$, then we have with probability $1 - O(1/\log^3 d)$,*

$\left\langle M(S_{t+1}^{(s)}) - \mathbf{I}, U_t^{(s)} \right\rangle \le 0.76 \left\langle M(S_t^{(s)}) - \mathbf{I}, U_t^{(s)} \right\rangle$.

Lemma G.5. *For an epoch s and for all $t = 0, 1, \cdots, T_2 = O(\log d)$ if Lemma G.4 holds, $n^{(s)} > 3n/4$, and $n \gtrsim \frac{B^2(\log B)d^{3/2}\log(1/\delta)}{\varepsilon\alpha}$, then we have $\|M(S^{(s+1)}) - \mathbf{I}\|_2 \le 0.98\|M(S^{(s)}) - \mathbf{I}\|_2$ with probability $1 - O(1/\log^2 d)$.*

G.3.1. Proof of Lemma G.4

Proof of Lemma G.4. To prove that we make progress for each iteration, we first show our dataset satisfies regularity conditions in Eqs. (14) and (15) that we need for DPTHRESHOLD. Following Lemma G.6 implies with probability $1 - 1/(\log^3 d)$, our scores satisfies the regularity conditions needed in Lemma F.1.

Lemma G.6. *For each epoch s and iteration t, under the hypotheses of Lemma G.4, with probability $1 - O(1/\log^3 d)$, we have*

$$\frac{1}{n}\sum_{i\in S_{\text{good}}\cap\mathcal{T}_{2\alpha}} \tau_i \le \psi/1000 \tag{14}$$

$$\frac{1}{n}\sum_{i\in S_{\text{good}}\cap S_t^{(s)}} (\tau_i - 1) \le \psi/1000\,, \tag{15}$$

where $\psi \triangleq \frac{1}{n}\sum_{i\in S_t^{(s)}}(\tau_i - 1)$.

Then by Lemma F.1 our DPTHRESHOLD gives us a threshold ρ such that

$$\sum_{i\in S_{\text{good}}\cap\mathcal{T}_{2\alpha}} \mathbf{1}\{\tau_i \le \rho\}\frac{\tau_i}{\rho} + \mathbf{1}\{\tau_i > \rho\} \le \sum_{i\in S_{\text{bad}}\cap\mathcal{T}_{2\alpha}} \mathbf{1}\{\tau_i \le \rho\}\frac{\tau_i}{\rho} + \mathbf{1}\{\tau_i > \rho\}\,.$$

Conditioned on the hypotheses and the claims of Lemma F.1, according to our filter rule from Algorithm 10, we have

$$\mathbb{E}|(S_t^{(s)} \setminus S_{t+1}^{(s)}) \cap S_{\text{good}}| = \sum_{i\in S_{\text{good}}\cap\mathcal{T}_{2\alpha}} \mathbf{1}\{\tau_i \le \rho\}\frac{\tau_i}{\rho} + \mathbf{1}\{\tau_i > \rho\}$$

and

$$\mathbb{E}|(S_t^{(s)} \setminus S_{t+1}^{(s)}) \cap S_{\text{bad}}| = \sum_{i\in S_{\text{bad}}\cap\mathcal{T}_{2\alpha}} \mathbf{1}\{\tau_i \le \rho\}\frac{\tau_i}{\rho} + \mathbf{1}\{\tau_i > \rho\}\,.$$

This implies $\mathbb{E}|(S_t^{(s)} \setminus S_{t+1}^{(s)}) \cap S_{\text{good}}| \le \mathbb{E}|(S_t^{(s)} \setminus S_{t+1}^{(s)}) \cap S_{\text{bad}}|$. At the same time, Lemma F.1 gives us a ρ such that with probability $1 - O(\log^3 d)$

$$\frac{1}{n}\sum_{i\in S_{t+1}^{(s)}} (\tau_i - 1) - 2\alpha \le \frac{1}{n}\sum_{\tau_i\le\rho}(\tau_i - 1) \le \frac{3}{4}\cdot\frac{1}{n}\sum_{i\in S_t^{(s)}} (\tau_i - 1)\,.$$

Hence, we have

$$\begin{aligned}
\left\langle M(S_t^{(s)}) - \mathbf{I}, U_t^{(s)} \right\rangle - \left\langle M(S_{t+1}^{(s)}) - \mathbf{I}, U_t^{(s)} \right\rangle &= \frac{1}{n}\sum_{i\in S_t^{(s)}\setminus S_{t+1}^{(s)}} (\tau_i - 1) \\
&\ge \frac{1}{4n}\sum_{i\in S_t^{(s)}} (\tau_i - 1) - 2\alpha \\
&\overset{(a)}{\ge} \frac{1}{4}\cdot\frac{998}{1000}\left\langle M(S_t^{(s)}) - \mathbf{I}, U_t^{(s)} \right\rangle\,,
\end{aligned}$$

where (a) follows from our assumption on λ_t and stopping criteria. Rearranging the terms completes the proof. □

G.3.2. PROOF OF LEMMA G.6

Proof of Lemma G.6. First of all, Lemma H.9, Lemma H.10 and Lemma H.11 gives us following Lemma G.7, which basically shows with enough samples, we can make sure the noises added for privacy guarantees are small enough with probability $1 - O(1/\log^3 d)$.

Lemma G.7. *For $\alpha \in (0, 0.5)$, if $n \gtrsim \frac{B^2(\log B)d^{3/2}\log(1/\delta)}{\varepsilon\alpha}$ and $n^{(s)} > 3n/4$ then we have with probability $1-O(1/\log^3 d)$, following conditions simultaneously hold:*

1. $\|\mu_t^{(s)} - \mu(S_t^{(s)})\|_2^2 \le 0.001\alpha\log 1/\alpha$
2. $|\psi_t^{(s)} - \left\langle M(S_t^{(s)}) - \mathbf{I}, U_t^{(s)}\right\rangle| \le 0.001\alpha\log 1/\alpha$
3. $\left|\lambda_t^{(s)} - \|M(S_t^{(s)}) - \mathbf{I}\|_2\right| \le 0.001\alpha\log 1/\alpha$
4. $\left|\lambda^{(s)} - \|M(S^{(s)}) - \mathbf{I}\|_2\right| \le 0.001\alpha\log 1/\alpha$
5. $\left\|M(S_{t+1}^{(s)}) - \Sigma_t^{(s)}\right\|_2 \le 0.001\alpha\log 1/\alpha$
6. $\|\mu^{(s)} - \mu(S^{(s)})\|_2^2 \le 0.001\alpha\log 1/\alpha$

Now under above conditions, since $\lambda_1^{(s)} > C\alpha\log 1/\alpha$, we have $\|M(S_t^{(s)}) - \mathbf{I}\|_2 > 0.5(C - 0.002)\alpha\log 1/\alpha$. Using the fact that $\mu(S_t^{(s)}) = (1/n)\sum_{i\in S_t^{(s)}} x_i$, we also have

$$\begin{aligned}
&\frac{1}{n}\sum_{i\in S_t^{(s)}}(\tau_i - 1)\\
=\ &\frac{1}{n}\sum_{i\in S_t^{(s)}}\left\langle\left(x_i - \mu_t^{(s)}\right)\left(x_i - \mu_t^{(s)}\right)^\top - \mathbf{I}, U_t^{(s)}\right\rangle\\
=\ &\frac{1}{n}\sum_{i\in S_t^{(s)}}\left\langle\left(x_i - \mu(S_t^{(s)})\right)\left(x_i - \mu(S_t^{(s)})\right)^\top - \mathbf{I}, U_t^{(s)}\right\rangle\\
&+\ \frac{|S_t^{(s)}|}{n}\left\langle\left(\mu(S_t^{(s)}) - \mu_t^{(s)}\right)\left(\mu(S_t^{(s)}) - \mu_t^{(s)}\right)^\top, U_t^{(s)}\right\rangle\\
=\ &\left\langle M(S_t^{(s)}) - \mathbf{I}, U_t^{(s)}\right\rangle + \frac{|S_t^{(s)}|}{n}\left\langle\left(\mu(S_t^{(s)}) - \mu_t^{(s)}\right)\left(\mu(S_t^{(s)}) - \mu_t^{(s)}\right)^\top, U_t^{(s)}\right\rangle .
\end{aligned}$$

Thus, from the first and the second claims in Lemma G.7, we have

$$|\psi - \psi_t^{(s)}| \le 0.002\ \alpha\log 1/\alpha\ . \tag{16}$$

For an epoch s and an iteration t, since $\alpha n \le S_{\rm good} \cap \mathcal{T}_{2\alpha} \cap S_t^{(s)} \le 2\alpha n$, we have

$$
\begin{aligned}
\frac{1}{n} \sum_{i \in S_{\rm good} \cap \mathcal{T}_{2\alpha} \cap S_t^{(s)}} \tau_i \;&=\; \frac{1}{n} \sum_{i \in S_{\rm good} \cap \mathcal{T}_{2\alpha} \cap S_t^{(s)}} \left\langle (x_i - \mu_t^{(s)})(x_i - \mu_t^{(s)})^\top, U_t^{(s)} \right\rangle \\
&\overset{(a)}{\le}\; \frac{2}{n} \sum_{i \in S_{\rm good} \cap \mathcal{T}_{2\alpha} \cap S_t^{(s)}} \left\langle (x_i - \mu)(x_i - \mu)^\top, U_t^{(s)} \right\rangle + \frac{2|S_{\rm good} \cap \mathcal{T}_{2\alpha} \cap S_t^{(s)}|}{n} \left\langle (\mu - \mu_t^{(s)})(\mu - \mu_t^{(s)})^\top, U_t^{(s)} \right\rangle \\
&\overset{(b)}{\le}\; O(\alpha \log 1/\alpha) + 4\alpha \left\langle (\mu - \mu_t^{(s)})(\mu - \mu_t^{(s)})^\top, U_t^{(s)} \right\rangle \\
&\le\; O(\alpha \log 1/\alpha) + 4\alpha \|\mu_t^{(s)} - \mu\|_2^2 \\
&\le\; O(\alpha \log 1/\alpha) + 4\alpha \left(\|\mu - \mu(S_t^{(s)})\|_2 + \|\mu(S_t^{(s)}) - \mu_t^{(s)}\|_2 \right)^2 \\
&\overset{(c)}{\le}\; O\left(\alpha \log 1/\alpha\right) + 4\alpha \left(O\left(\alpha \sqrt{\log 1/\alpha}\right) + \sqrt{\alpha \left(O\left(\alpha \log 1/\alpha\right) + \|M(S_t^{(s)}) - \mathbf{I}\|_2 \right)} + \|\mu(S_t^{(s)}) - \mu_t^{(s)}\|_2 \right)^2 \\
&\le\; O(\alpha \log 1/\alpha) + 8\alpha^2 \left(\|M(S_t^{(s)}) - \mathbf{I}\|_2 + O\left(\alpha \log 1/\alpha\right) \right) + O(8\alpha^3 \log 1/\alpha) + 8\alpha^2 \log 1/\alpha \\
&\overset{(d)}{\le}\; \frac{1}{1000} \left(\frac{\|M(S_t^{(s)}) - \mathbf{I}\|_2 - 0.001\, \alpha \log 1/\alpha}{5.5} - 0.002\, \alpha \log 1/\alpha \right) \\
&\le\; \frac{\psi_t^{(s)} - 0.002\, \alpha \log 1/\alpha}{1000} \\
&\le\; \frac{\psi}{1000}\,,
\end{aligned}
$$

where (a) follows from the fact that for any vector x, y, z, we have $(x-y)(x-y)^\top \preceq 2(x-z)(x-z)^\top + 2(y-z)(y-z)^\top$, (b) follows from Lemma H.4, (c) follows from Lemma H.7, (d) follows from our choice of large constant C, and in the last inequality we used Eq. (16).

Similarly we have

$$
\begin{aligned}
& \frac{1}{n} \sum_{i \in S_{\text{good}} \cap S_t^{(s)}} (\tau_i - 1) \\
= \; & \frac{1}{n} \sum_{i \in S_{\text{good}} \cap S_t^{(s)}} \left\langle (x_i - \mu_t^{(s)})(x_i - \mu_t^{(s)})^\top - \mathbf{I}, U_t^{(s)} \right\rangle \\
= \; & \frac{1}{n} \sum_{i \in S_{\text{good}} \cap S_t^{(s)}} \left\langle \left(x_i - \mu(S_{\text{good}} \cap S_t^{(s)})\right) \left(x_i - \mu(S_{\text{good}} \cap S_t^{(s)})\right)^\top - \mathbf{I}, U_t^{(s)} \right\rangle \\
& + \frac{|S_{\text{good}} \cap S_t^{(s)}|}{n} \left\langle \left(\mu(S_{\text{good}} \cap S_t^{(s)}) - \mu_t^{(s)}\right) \left(\mu(S_{\text{good}} \cap S_t^{(s)}) - \mu_t^{(s)}\right)^\top, U_t^{(s)} \right\rangle \\
\overset{(a)}{\leq} \; & O\left(\alpha \log 1/\alpha\right) + \left\| \mu(S_{\text{good}} \cap S_t^{(s)}) - \mu_t^{(s)} \right\|_2^2 \\
\leq \; & O\left(\alpha \log 1/\alpha\right) + \left(\left\| \mu(S_{\text{good}} \cap S_t^{(s)}) - \mu \right\|_2 + \left\| \mu - \mu(S_t^{(s)}) \right\|_2 \right)^2 + 0.001\, \alpha \log 1/\alpha \\
\overset{(b)}{\leq} \; & O\left(\alpha \log 1/\alpha\right) + \left(O(\alpha \sqrt{\log 1/\alpha}) + \sqrt{\alpha(\|M(S_t^{(s)}) - \mathbf{I}\|_2 + O(\alpha \log 1/\alpha))} \right)^2 + 0.001\, \alpha \log 1/\alpha \\
\leq \; & O\left(\alpha \log 1/\alpha\right) + \alpha \left(\|M(S_t^{(s)}) - \mathbf{I}\|_2 + O\left(\alpha \log 1/\alpha\right) \right) + O(\alpha^2 \log 1/\alpha) + +0.001\, \alpha \log 1/\alpha \\
\overset{(c)}{\leq} \; & \frac{1}{1000} \left(\frac{\|M(S_t^{(s)}) - \mathbf{I}\|_2 - 0.001\, \alpha \log 1/\alpha}{5.5} - 0.002\, \alpha \log 1/\alpha \right) \\
\leq \; & \frac{\psi_t^{(s)} - 0.002\, \alpha \log 1/\alpha}{1000} \\
\leq \; & \frac{\psi}{1000}\,,
\end{aligned}
$$

where (a) follows from Lemma H.4, (b) follows from Lemma H.5 and Lemma H.7 and (c) follows from our choice of large constant C.

□

G.3.3. Proof of Lemma G.5

Proof of Lemma G.5. Under the conditions of Lemma G.7, we have picked n large enough such that with probability $1 - O(1/\log^3 d)$, we have

$$
\|\Sigma_t^{(s)} - \mathbf{I}\|_2 \approx_{0.01} \|M(S_t^{(s)}) - \mathbf{I}\|_2\,.
$$

By Lemma G.4, we now have

$$
\begin{aligned}
\left\langle M(S_t^{(s)}) - \mathbf{I}, U_t^{(s)} \right\rangle & \leq 0.76 \left\langle M(S_{t-1}^{(s)}) - \mathbf{I}, U_t^{(s)} \right\rangle \\
& \leq 0.76 \left\langle M(S_1^{(s)}) - \mathbf{I}, U_t^{(s)} \right\rangle \\
& \leq 0.76 \|M(S_1^{(s)}) - \mathbf{I}\|_2\,. \qquad (17)
\end{aligned}
$$

Since $\lambda_1^{(s)} > C\alpha \log 1/\alpha$, we have $\|M(S_t^{(s)}) - \mathbf{I}\|_2 > 0.5(C - 0.002)\alpha \log 1/\alpha$. Combining the above inequality and the fifth claim of Lemma G.7 together, we have

$$
\left\langle \Sigma_t^{(s)} - \mathbf{I}, U_t^{(s)} \right\rangle \leq \left\langle M(S_t^{(s)}) - \mathbf{I}, U_t^{(s)} \right\rangle + \|\Sigma_t^{(s)} - M(S_t^{(s)})\|_2 \leq 0.77 \|M(S_1^{(s)}) - \mathbf{I}\|_2\,.
$$

By Lemma H.1, we have $M(S_t^{(s)}) - \mathbf{I} \preceq M(S_1^{(s)}) - \mathbf{I}$. by our choice of $\alpha^{(s)}$, we have $\alpha^{(s)} \left(M(S_{t+1}^{(s)}) - \mathbf{I}\right) \preceq \frac{1}{100}\mathbf{I}$ and $\alpha^{(s)} \left(\Sigma_t^{(s)} - \mathbf{I}\right) \preceq \frac{1}{100}\mathbf{I}$. Therefore, by Lemma H.14, we have

$$\begin{aligned}
&\left\|\sum_{i=1}^{T_2} \Sigma_t^{(s)} - \mathbf{I}\right\|_2 \\
\leq\ & \sum_{t=1}^{T_2} \left\langle \Sigma_t^{(s)} - \mathbf{I}, U_t^{(s)} \right\rangle + \alpha^{(s)} \sum_{t=0}^{T_2} \left\langle U_t^{(s)}, \left|\Sigma_t^{(s)} - \mathbf{I}\right| \right\rangle \|\Sigma_t^{(s)} - \mathbf{I}\|_2 + \frac{\log(d)}{\alpha^{(s)}} \\
\overset{(a)}{\leq}\ & \sum_{t=1}^{T_2} \left\langle \Sigma_t^{(s)} - \mathbf{I}, U_t^{(s)} \right\rangle + \frac{1}{100} \sum_{t=1}^{T_2} \left\langle U_t^{(s)}, \left|\Sigma_t^{(s)} - \mathbf{I}\right| \right\rangle + 200\log(d)\|M(S_1^{(s)}) - \mathbf{I}\|_2
\end{aligned}$$

where (a) follows from our choice of $\alpha^{(s)}$ and C. By Lemma H.6, $M(S_t^{(s)}) - \mathbf{I} \succeq -c_1\alpha\log 1/\alpha \cdot I$ for $t = 1, 2, \cdots, T_2$, we have

$$|M(S_t^{(s)}) - \mathbf{I}| \preceq M(S_t^{(s)}) - \mathbf{I} + 2c_1\alpha\log 1/\alpha\ \mathbf{I},$$

and hence

$$\left\langle U_t^{(s)}, \left|M(S_t^{(s)}) - \mathbf{I}\right| \right\rangle \leq \left\langle U_t^{(s)}, M(S_t^{(s)}) - \mathbf{I} \right\rangle + 2c_1\alpha\log 1/\alpha$$

Meanwhile, we have

$$M(S_{t+1}^{(s)}) - \mathbf{I} - \|\Sigma_t^{(s)} - M(S_t^{(s)})\|_2\ \mathbf{I} \preceq \Sigma_t^{(s)} - \mathbf{I} \preceq M(S_{t+1}^{(s)}) - \mathbf{I} + \|\Sigma_t^{(s)} - M(S_t^{(s)})\|_2\ \mathbf{I}\ .$$

Hence,

$$|\Sigma_t^{(s)} - \mathbf{I}| \preceq M(S_t^{(s)}) - \mathbf{I} + (3\|\Sigma_t^{(s)} - M(S_t^{(s)})\|_2 + 2c_1\alpha\log 1/\alpha)\ \mathbf{I}$$

Together with Eq. (17), we have

$$\begin{aligned}
&\left\langle U_t^{(s)}, \left|\Sigma_t^{(s)} - \mathbf{I}\right| \right\rangle \\
\leq\ & \left\langle U_t^{(s)}, M(S_t^{(s)}) - \mathbf{I} \right\rangle + 3\|\Sigma_t^{(s)} - M(S_t^{(s)})\|_2 + 2c_1\alpha\log 1/\alpha \\
\leq\ & 0.79\ \left\|M(S_1^{(s)}) - \mathbf{I}\right\|_2 + 2c_1\alpha\log 1/\alpha\ .
\end{aligned}$$

By Lemma H.6, we have $M(S_t^{(s)}) - \mathbf{I} \succeq -c_1\alpha\log 1/\alpha\ \mathbf{I}$. Also, we know $M(S_t^{(s)}) - \mathbf{I} \preceq M(S_1^{(s)}) - \mathbf{I}$. Then we have

$$\begin{aligned}
&\left\|M(S_{T_2}^{(s)}) - \mathbf{I}\right\|_2 \\
\leq\ & \frac{1}{T_2}\left\|\sum_{i=1}^{T_2} M(S_t^{(s)}) - \mathbf{I}\right\|_2 \\
\leq\ & \frac{1}{T_2}\left\|\sum_{i=1}^{T_2} \Sigma_t^{(s)} - \mathbf{I}\right\|_2 + 0.001\ \alpha\log 1/\alpha \\
\leq\ & \frac{1}{T_2}\left(\sum_{t=1}^{T_2} \left\langle \Sigma_t^{(s)} - \mathbf{I}, U_t^{(s)} \right\rangle + \frac{1}{100}\sum_{t=1}^{T_2} \left\langle U_t^{(s)}, \left|\Sigma_t^{(s)} - \mathbf{I}\right| \right\rangle + 200\log(d)\|M(S_1^{(s)}) - \mathbf{I}\|_2\right) + 0.001\ \alpha\log 1/\alpha \\
\leq\ & 0.79\|M(S_1^{(s)}) - \mathbf{I}\|_2 + 2c_1\alpha\log 1/\alpha + \frac{200\log(d)}{T_2}\|M(S_1^{(s)}) - \mathbf{I}\|_2 + 0.001\ \alpha\log 1/\alpha \\
\leq\ & 0.98\ \|M(S_1^{(s)}) - \mathbf{I}\|_2\ ,
\end{aligned}$$

where the last inequality follows from our assumption that $\lambda_0^{(s)} > C\alpha \log 1/\alpha$, and conditions of Lemma G.7 hold and we have $\|M(S_t^{(s)}) - \mathbf{I}\|_2 > 0.5(C - 0.002)\alpha \log 1/\alpha$. □

H. Technical lemmas

H.1. Lemmata for sub-Gaussian regularity from (Dong et al., 2019)

Lemma H.1 ((?)Lemma 3.4]dong2019quantum). *If $S' \subset S$, then $M(S') \preceq M(S)$.*

Definition H.2 ((?)Definition 4.1]dong2019quantum). *Let D be a distribution with mean $\mu \in \mathbb{R}^d$ and covariance $\mathbf{I}$. For $0 < \alpha < 1/2$, we say a set of points $S = \{X_1, X_2, \cdots, X_n\}$ is α-subgaussian good with respect to $\mu \in \mathbb{R}^d$ if following inequalities are satisfied:*

- $\|\mu(S) - \mu\|_2 \lesssim \alpha\sqrt{\log 1/\alpha}$ *and* $\left\|\frac{1}{|S|}\sum_{i\in S}(X_i - \mu(S))(X_i - \mu(S))^\top - \mathbf{I}\right\|_2 \lesssim \alpha \log 1/\alpha$.
- *for any subset $T \subset S$ so that $|T| = 2\alpha|S|$, we have*

$$\left\|\frac{1}{|T|}\sum_{i\in T} X_i - \mu\right\|_2 \lesssim \sqrt{\log 1/\alpha} \;\; \text{and} \;\; \left\|\frac{1}{|T|}\sum_{i\in T}(X_i - \mu(S))(X_i - \mu(S))^\top - \mathbf{I}\right\|_2 \lesssim \log 1/\alpha \; .$$

Lemma H.3 ((?)Lemma 4.1]dong2019quantum). *A set of i.i.d. samples from an identity covariance sub-Gaussian distribution of size $n = \Omega\left(\frac{d+\log 1/\delta}{\alpha^2 \log 1/\alpha}\right)$ is α-subgaussian good with respect to μ with probability $1 - \delta$.*

Lemma H.4 ((?)Fact 4.2]dong2019quantum). *Let S be an α-corrupted sub-Gaussian dataset under Assumption 1. If S_{good} is α-subgaussian good with respect to $\mu \in \mathbb{R}^d$, then for any $T \subset S$ such that $|T| \le 2\alpha|S|$, we have for any unit vector $v \in \mathbb{R}^d$*

$$\frac{1}{|S|}\sum_{X_i\in T}\langle (X_i - \mu), v\rangle^2 \lesssim \alpha \log 1/\alpha \; .$$

For any subset $T \subset S$ such that $|T| \ge (1 - 2\alpha)|S|$, we have

$$\left\|\frac{1}{|S|}\sum_{i\in T}(x_i - \mu)(x_i - \mu)^\top - \mathbf{I}\right\|_2 \lesssim \alpha \log 1/\alpha \;\; \text{and} \;\; ,$$
$$\left\|\frac{1}{|S|}\sum_{i\in T}(x_i - \mu(T))(x_i - \mu(T))^\top - \mathbf{I}\right\|_2 \lesssim \alpha \log 1/\alpha$$

Lemma H.5 ((?)Corollary 4.3]dong2019quantum). *Let S be an α-corrupted sub-Gaussian dataset under Assumption 1. If S_{good} is α-subgaussian good with respect to $\mu \in \mathbb{R}^d$, then for any $T \subset S$ such that $|T| \le 2\alpha|S|$, we have*

$$\left\|\frac{1}{|S|}\sum_{X_i\in T}(X_i - \mu)\right\|_2 \lesssim \alpha\sqrt{\log 1/\alpha} \; .$$

For any subset $T \subset S$ such that $|T| \ge (1 - 2\alpha)|S|$, we have

$$\|\mu(T) - \mu\|_2 \lesssim \alpha\sqrt{\log 1/\alpha} \; .$$

Lemma H.6 ((?)Lemma 4.5]dong2019quantum). *Let S be an α-corrupted sub-Gaussian dataset under Assumption 1. If S_{good} is α-subgaussian good with respect to $\mu \in \mathbb{R}^d$, then for any $T \subset S$ such that $|T \cap S_{\text{good}}| \ge (1 - 2\alpha)|S|$, then there is some universal constant c_1 such that*

$$\frac{1}{|S|}\sum_{i\in T}(x_i - \mu(T))(x_i - \mu(T))^\top \succeq (1 - c_1\alpha \log 1/\alpha)\mathbf{I} \; .$$

Lemma H.7 ((Dong et al., 2019) Lemma 4.6). *Let S be an α-corrupted sub-Gaussian dataset under Assumption 1. If S_{good} is α-subgaussian good with respect to $\mu \in \mathbb{R}^d$, then for any $T \subset S$ such that $|T \cap S_{\text{good}}| \ge (1 - 2\alpha)|S|$, we have*

$$\|\mu(T) - \mu\|_2 \le \frac{1}{1-\alpha} \cdot \left(\sqrt{\alpha\left(\|M(T) - \mathbf{I}\|_2 + O\left(\alpha \log 1/\alpha\right)\right)} + O\left(\alpha\sqrt{\log 1/\alpha}\right)\right) \; .$$

H.2. Auxiliary Lemmas on Laplace and Gaussian mechanism

Lemma H.8 (Theorem A.1 in (Dwork & Roth, 2014)). *Let $\varepsilon \in (0,1)$ be arbitrary. For $c^2 \geq 2\ln(1.25/\delta)$, the Gaussian Mechanism with parameter $\sigma^2 \geq c^2\Delta_2 f/\varepsilon$ is (ε,δ)-differentially private.*

Lemma H.9. *Let $Y \sim \mathrm{Lap}(b)$. Then for all $h > 0$, we have $\mathbb{P}(|Y| \geq hb) = e^{-h}$.*

Lemma H.10 (Tail bound of χ-square distribution (Wainwright, 2019)). *Let $x_i \sim \mathcal{N}(0,\sigma^2)$ for $i = 1,2,\cdots,d$. Then for all $\zeta \in (0,1)$, we have $\mathbb{P}(\|X\|_2 \geq \sigma\sqrt{d\log(1/\zeta)}) \leq \zeta$.*

Lemma H.11 ((**?**)Corollary 2.3.6]tao2012topics). *Let $Z \in \mathbb{R}^{d\times d}$ be a matrix such that $Z_{i,j} \sim \mathcal{N}(0,\sigma^2)$ for $i \geq j$ and $Z_{i,j} = Z_{j,i}$ for $i < j$. For $\forall \zeta \in (0,1)$, then with probability $1-\zeta$ we have $\|Z\|_2 \leq \sigma\sqrt{d}\log(1/\zeta)$.*

Lemma H.12 (Accuracy of the histogram using Gaussian Mechanism). *Let $f : \mathcal{X}^n \to \mathbb{R}^{\mathcal{S}}$ be a histogram over K bins. For any dataset $D \in \mathcal{X}^n$ and ε, Gaussian Mechanism is an (ε,δ)-differentially private algorithm $M(D)$ such that given with probability $1-\zeta$ we have*

$$\|M(D) - f(D)\|_\infty \leq O(\frac{\sqrt{\log(K/\zeta)\log(1/\delta)}}{\varepsilon n}) .$$

Proof. First notice that the ℓ_2 sensitivity of histogram function f is $\sqrt{2}/n$. Thus, by Lemma H.8, by adding noise $\mathcal{N}(0,(\frac{2\sqrt{2\log(1.25/\delta)}}{n\varepsilon})^2)$ to each entry of f, we have a (ε,δ) differentially private algorithm. Since Gaussian tail bound implies that $\mathbb{P}_{x\sim\mathcal{N}(0,\sigma^2)}[x \geq \Omega(\sqrt{\log(K/\eta)}\sigma)] \leq \eta/K$, we have that with probability $1-\eta$, the ℓ_∞ norm of the added noise is bounded by $O(\frac{\sqrt{\log(1/\delta)\log(K/\eta)}}{n\varepsilon})$. This concludes the proof. □

Lemma H.13 (Composition theorem of (**?**)Theorem 3.4]composition). *For $\varepsilon \leq 0.9$, an end-to-end guarantee of (ε,δ)-differential privacy is satisfied if a dataset is accessed k times, each with a $(\varepsilon/2\sqrt{2k\log(2/\delta)}, \delta/2k)$-differential private mechanism.*

H.3. Analysis of $\|M(S_t^{(s)}) - \mathbf{I}\|_2$ shrinking

For any symmetric matrix $A = \sum_{i=1}^d \lambda_i v_i v_i^\top$, we let $|A|$ denote $|A| = \sum_{i=1}^d |\lambda_i| v_i v_i^\top$.

Lemma H.14 (Regret bound, Special case of (**?**)Theorem 3.1]allen2015spectral). *Let*

$$U_t \;=\; \frac{\exp(\alpha\sum_{k=1}^{t-1}(\Sigma_k - \mathbf{I}))}{\mathrm{Tr}(\exp(\alpha\sum_{k=1}^{t-1}(\Sigma_k - \mathbf{I})))} ,$$

and α satisfies $\alpha(\Sigma_t - \mathbf{I}) \preceq I$ for all $k \in [T]$, then for all $U \succeq 0$, $\mathrm{Tr}(U) = 1$, it holds that

$$\sum_{t=1}^T \langle(\Sigma_t - \mathbf{I}), U - U_t\rangle \leq \alpha\sum_{t=1}^T \langle|(\Sigma_t - \mathbf{I}), U_t|\rangle \cdot \|(\Sigma_t - \mathbf{I})\|_2 + \frac{\log d}{\alpha}.$$

Rearranging terms, and taking a supremum over U, we obtain that

$$\|\sum_{t=1}^T (\Sigma_t - \mathbf{I})\|_2 \leq \sum_{t=1}^T \langle U_t, (\Sigma_t - \mathbf{I})\rangle + \alpha\sum_{t=1}^T \langle|(\Sigma_t - \mathbf{I}), U_t|\rangle \cdot \|(\Sigma_t - \mathbf{I})\|_2 + \frac{\log d}{\alpha}.$$

I. Exponential time DP robust mean estimation of sub-Gaussian and heavy tailed distributions (Algorithm 3)

In this section, we give a self-contained proof of the privacy and utility of our exponential time robust mean estimation algorithm for sub-Gaussian and heavy tailed distributions. The proof relies on the resilience property of the uncorrupted data as shown in the following lemmas.

Lemma I.1 (Lemma 10 in (Steinhardt et al., 2018)). *If a set of points $\{x_i\}_{i\in S}$ lying in $\mathbb{R}^d$ is (σ,α)-resilient around a point μ, then*

$$\|\frac{1}{|T'|}\sum_{i\in T'}(x_i-\mu)\|_2 \le \frac{2-\alpha}{\alpha}\sigma.$$

for all sets T' of size at least $\alpha|S|$.

Lemma I.2 (Finite sample resilience of sub-Gaussian distributions (**?**)Theorem G.1]zhu2019generalized). *Let S_{good} be a set of i.i.d. points from a sub-Gaussian distribution $\mathcal{D}$ with a parameter $\mathbf{I}_d$. Given that $|S_{\text{good}}| = \Omega((d+\log(1/\zeta))/(\alpha^2\log 1/\alpha))$, S_{good} is $(\alpha\sqrt{\log(1/\alpha)},\alpha)$-resilient around its mean μ with probability $1-\zeta$.*

Lemma I.3 (Finite sample resilience of heavy-tailed distributions (**?**)Theorem G.2]zhu2019generalized). *Let S_{good} be a set of i.i.d. samples drawn from distribution $\mathcal{D}$ whose mean and covariance are μ,Σ respectively, and that $\Sigma \preceq I$. Given that $|S| = \Omega(d/(\zeta\alpha))$, there exists a constant c_ζ that only depends on ζ such that S_{good} is $(c_\zeta\sqrt{\alpha},\alpha)$-resilient around μ with probability $1-\zeta$.*

I.1. Case of heavy-tailed distributions and a proof of Theorem 7

Lemma J.1 ensures that $q_{\text{range}-\text{ht}}$ returns samples in a bounded support of Euclidean distance $\sqrt{d}B/2$ with $B = 50/\sqrt{\alpha}$ where $(1-2\alpha)n$ samples are uncorrupted (αn is corrupted by adversary and αn can be corrupted by the pre-processing step). For a $(c_\zeta\sqrt{3\alpha},3\alpha)$-resilient dataset, we first show that $R(S)$ is robust against corruption.

Lemma I.4 (α-corrupted data has small $R(S)$). *Let S be the set of 2α-corrupted data. Given that $n = \Omega(d/(\zeta\alpha))$, with probability $1-\zeta$, $R(S) \le c_\zeta\sqrt{3\alpha}$.*

This follows immediately by selecting S' to be the uncorrupted $(1-2\alpha)$ fraction of the dataset and applying $(c_\zeta\sqrt{3\alpha},3\alpha)$-resilience. After pre-processing, we have that $\|x_i-\bar{x}\|_2 \le B\sqrt{d}/2$, and then clearly $R(\cdot)$ has sensitivity $\Delta_R \le B\sqrt{d}/n$.

Lemma I.5 (Sensitivity and Privacy of $\hat{R}(S)$). *Given that $\hat{R}(S) = R(S) + \text{Lap}(\frac{3B\sqrt{d}}{n\varepsilon})$, $\hat{R}(S)$ is $(\varepsilon/3,0)$-differentially private. Further, with probability $1-\delta/3$, $|\hat{R}(S)-R(S)| \le \frac{3B\sqrt{d}\log(3/\delta)}{n\varepsilon}$.*

In the algorithm, we first compute $\hat{R}(S)$. If $\hat{R}(S) \ge 2c_\zeta\sqrt{\alpha}$, we stop and output $\emptyset$. Otherwise, we use exponential mechanism with score function $d(\hat{\mu},S)$ to find an estimate $\hat{\mu}$. We prove the privacy guarantee of our algorithm as follows.

Lemma I.6 (Privacy). *Algorithm 3 is (ε,δ)-differentially private if $n \ge 6B\sqrt{d}\log(3/\delta)/(c_\zeta\varepsilon\sqrt{\alpha})$.*

Proof. We consider neighboring datasets S, S' under the following two scenario

1. $R(S) > 3c_\zeta\sqrt{\alpha}$

   In this case, given that $n \ge \frac{6B\sqrt{d}\log(3/\delta)}{c_\zeta\sqrt{\alpha}\varepsilon}$, we have $\widehat{R}(S) > 2c_\zeta\sqrt{\alpha}$ and the output of the algorithm $\mathcal{A}(S) = \emptyset$ with probability at least $1-\delta/3$, and $\mathcal{A}(S') = \emptyset$ with probability at least $1-\delta/3$. Thus, for any set Q, $\mathbb{P}[\mathcal{A}(S)\in Q] \le \mathbb{P}[\mathcal{A}(S')\in Q] + \delta/3$.

2. $R(S) \le 3c_\zeta\sqrt{\alpha}$

   Lemma I.7 (Sensitivity of $d(\hat{\mu},S)$). *Given that $R(S) \le 3c_\zeta\sqrt{\alpha}$, for any neighboring dataset S', $|d(\hat{\mu},S)-d(\hat{\mu},S')| \le 12c_\zeta/(n\sqrt{\alpha})$.*

   In this case, the privacy guarantee of $\hat{R}(S)$ yields that $\mathbb{P}[\hat{R}(S)\in Q] \le \exp(\varepsilon/3)\cdot\mathbb{P}[\hat{R}(S')\in Q]$. Lemma I.7 yields that $\mathbb{P}[\hat{\mu}(S)\in Q] \le \exp(\varepsilon)\cdot\mathbb{P}[\hat{\mu}(S')\in Q]$. A simple composition of the privacy guarantee with $q_{\text{range}-\text{ht}}(\cdot)$ and the exponential mechanism gives that

$$\mathbb{P}[(\hat{R}(S),\hat{\mu}(S))\in Q] \le \exp(\varepsilon)\cdot\mathbb{P}[(\hat{R}(S'),\hat{\mu}(S'))\in Q] + \delta/3$$

This implies that $\mathbb{P}[\mathcal{A}(S) \in Q] \leq \exp(\varepsilon) \cdot \mathbb{P}[\mathcal{A}(S') \in Q] + \delta/3$.

□

Lemma I.8 (Utility of the algorithm)**.** *For an 2α-corrupted dataset S, Algorithm 3 achieves $\|\hat{\mu} - \mu^*\|_2 \leq c_\zeta\sqrt{\alpha}$ with probability $1-\zeta$, if $n = \Omega(d/(\alpha\zeta) + (d\log(dR/\alpha) + \log(1/\zeta)/(\varepsilon\alpha))$.*

Proof of Lemma I.8. We use the following lemma showing that $d(\hat{\mu}, S)$ is a good approximation of $\|\hat{\mu} - \mu^*\|_2$.

Lemma I.9 ($d(\mu, S)$ approximates $\|\mu - \mu^*\|$)**.** *Let S be the set of 2α-corrupted data. Given that $n = \Omega(d/(\zeta\alpha))$, with probability $1-\zeta$,*

$$\left| d(\hat{\mu}, S) - \|\hat{\mu} - \mu^*\|_2 \right| \;\leq\; 7c_\zeta\sqrt{\alpha}\,.$$

This implies that the exponential mechanism achieves the following bounds.

$$\begin{aligned}
\mathbb{P}(\|\hat{\mu} - \mu^*\| \leq c_\zeta\sqrt{\alpha}) &\geq \frac{1}{A}e^{-\frac{\varepsilon\alpha n}{3}}\,\mathrm{Vol}(c_\zeta\sqrt{\alpha}, d), \text{ and} \\
\mathbb{P}(\|\hat{\mu} - \mu^*\| \geq 22c_\zeta\sqrt{\alpha}) &\leq \frac{1}{A}e^{-\frac{5\varepsilon\alpha n}{8}}\,(4R)^d\,,
\end{aligned}$$

where A denotes the normalizing factor for the exponential mechanism and $\mathrm{Vol}(r, d)$ is the volume of a ball of radius r in d dimensions. It follows that

$$\begin{aligned}
\log\Big(\frac{\mathbb{P}(\|\hat{\mu} - \mu^*\|_2 \leq c_\zeta\sqrt{\alpha})}{\mathbb{P}(\|\hat{\mu} - \mu^*\|_2 \geq 22c_\zeta\sqrt{\alpha})}\Big) &\geq \frac{7}{24}\varepsilon\alpha n - C\,d\log(dR/\alpha) \\
&\geq \log(1/\zeta)\,,
\end{aligned}$$

for $n = \Omega((d\log(dR/\alpha) + \log(1/\zeta))/(\varepsilon\alpha))$.

□

I.1.1. Proof of Lemma I.7

Since $R(S) \leq 3c_\zeta\sqrt{\alpha}$, define S_{good} as the minimizing subset in Definition C.2 such that

$$R(S) = \max_{T \subset S_{\mathrm{good}}, |T| = (1-\alpha)|S_{\mathrm{good}}|} \|\mu(T) - \mu(S_{\mathrm{good}})\|_2\,.$$

By this definition of S_{good} and Lemma I.1,

$$|v^\top(\mu(S_{\mathrm{good}} \cap \mathcal{T}^v) - \mu(S_{\mathrm{good}}))| \leq 6c_\zeta\sqrt{1/\alpha}, \text{ and}$$

$$|v^\top(\mu(S_{\mathrm{good}} \cap \mathcal{B}^v) - \mu(S_{\mathrm{good}}))| \leq 6c_\zeta\sqrt{1/\alpha}.$$

Therefore,

$$\min_{i \in S_{\mathrm{good}} \cap \mathcal{T}^v} |v^\top(x_i - \mu(S_{\mathrm{good}}))| \leq |v^\top(\mu(S_{\mathrm{good}} \cap \mathcal{T}^v) - \mu(S_{\mathrm{good}}))| \leq 6c_\zeta\sqrt{1/\alpha},$$

and similarly

$$\min_{i \in S_{\mathrm{good}} \cap \mathcal{B}^v} |v^\top(x_i - \mu(S_{\mathrm{good}}))| \leq |v^\top(\mu(S_{\mathrm{good}} \cap \mathcal{B}^v) - \mu(S_{\mathrm{good}}))| \leq 6c_\zeta\sqrt{1/\alpha}$$

This implies

$$\min_{i \in S_{\mathrm{good}} \cap \mathcal{T}^v} v^\top x_i - \max_{i \in S_{\mathrm{good}} \cap \mathcal{B}^v} v^\top x_i \;\leq\; 12c_\zeta\sqrt{1/\alpha}\,. \tag{18}$$

This implies that distribution of one-dimensional points $S_{(v)} = \{v^\top x_i\}$ is dense at the boundary of top and bottom α quantiles, and hence cannot be changed much by changing one entry. Formally, consider a neighboring dataset S' (and the corresponding $S'_{(v)}$) where one point x_i in $\mathcal{M}^{(v)}(S)$ is replaced by another point $\tilde{x}_i$. If $v^\top\tilde{x}_i \in$

$[\max_{i\in S_{\rm good}\cap\mathcal{B}^v} v^\top x_i\,,\ \min_{i\in S_{\rm good}\cap\mathcal{T}^v} v^\top x_i\,]$, then Eq. (18) implies that this only changes the mean by $6c_\zeta/(\sqrt{\alpha}n)$. Otherwise, $\mathcal{M}^v(S')$ will have x_i replaced by either $\arg\min_{i\in S_{\rm good}\cap\mathcal{T}^v} v^\top x_i$ or $\arg\max_{i\in S_{\rm good}\cap\mathcal{B}^v} v^\top x_i$. In both cases, Eq. (18) implies that this only changes the mean by $12c_\zeta/(\sqrt{\alpha}n)$. The other case of when the replaced sample $x_i\in S$ is not in $\mathcal{M}^v(S)$ follows similarly. From this, we upper bounds the maximum difference between S and S' when projected on v, that is

$$\left|v^\top\left(\mu(\mathcal{M}^v(S))-\mu(\mathcal{M}^v(S'))\right)\right|\le\frac{12c_\zeta}{\sqrt{\alpha}n}\,.$$

This implies the sensitivity of $d(\mu,S)$ is bounded by $6c_\zeta/(\sqrt{\alpha}n)$:

$$\begin{aligned}|d(\mu,S)-d(\mu,S')| &= \Big|\max_{v\in\mathbb{S}^{d-1}} v^\top\mu(M^v(S))-\max_{\tilde v\in\mathbb{S}^{d-1}}\tilde v^\top\mu(M^v(S'))\Big|\\ &\le \max_{v\in\mathbb{S}^{d-1}}\big|\,v^\top(\mu(M^v(S))-\mu(M^v(S'))\,)\big| \;\le\; \frac{12c_\zeta}{\sqrt{\alpha}\,n}\end{aligned}$$

I.1.2. Proof of Lemma I.9

First we show $|v^\top(\mu(\mathcal{M}^v)-\mu^*)|\le 7c_\zeta\sqrt{\alpha}$. Notice that $|S_{\rm good}\cap\mathcal{T}^v|\le 3\alpha|S|$, and $|S_{\rm good}\cap\mathcal{B}^v|\le 3\alpha|S|$. By the $(c_\zeta\sqrt{3\alpha},3\alpha)$-resilience property, we have $|v^\top(\mu(S_{\rm good}\cap\mathcal{T}^v)-\mu^*)|\le c_\zeta\sqrt{3/\alpha}$, and $|v^\top(\mu(S_{\rm good}\cap\mathcal{B}^v)-\mu^*)|\le c_\zeta\sqrt{3/\alpha}$. Since $|S_{\rm good}\cap\mathcal{M}^v|\ge(1-8\alpha)|S_{\rm good}|$, by the $(c_\zeta\sqrt{8\alpha},8\alpha)$-resilience property,

$$|v^\top(\mu(S_{\rm good}\cap\mathcal{M}^v)-\mu^*)| \;\le\; c_\zeta\sqrt{8\alpha}\,.$$

Since $\mathcal{T}^v$, $\mathcal{B}^v$ are the largest and smallest $3\alpha n$ points respectively and $|S_{\rm bad}|\le 2\alpha n$, we get

$$|v^\top(\mu(S_{bad}\cap\mathcal{M}^v)-\mu^*)| \;\le\; 2c_\zeta\sqrt{3/\alpha}.$$

Combining $S_{\rm good}\cap\mathcal{M}^v$ and $S_{\rm bad}\cap\mathcal{M}^v$ we get

$$\begin{aligned}&|v^\top(\mu(\mathcal{M}^v)-\mu^*)|\\ &\le\frac{|S_{bad}\cap\mathcal{M}^v|}{|\mathcal{M}^v|}|v^\top(\mu(S_{bad}\cap\mathcal{M}^v)-\mu^*)|+\frac{|\mu(S_{\rm good}\cap\mathcal{M}^v|}{|\mathcal{M}^v|}|v^\top(\mu(S_{\rm good}\cap\mathcal{M}^v)-\mu^*)|\\ &\le 7c_\zeta\sqrt{\alpha}.\end{aligned}$$

Finally we get that

$$\begin{aligned}\left|\,d(\hat\mu,S)-\|\hat\mu-\mu^*\|_2\,\right| &\overset{(a)}{=}\left|\max_{v\in\mathbb{S}^{d-1}}\left|v^\top\left(\mu(\mathcal{M}^{(v)})-\hat\mu\right)\right|-\max_{v\in\mathbb{S}^{d-1}}|v^\top(\hat\mu-\mu^*)|\right|\\ &\overset{(b)}{\le}\max_{v\in\mathbb{S}^{d-1}}\left|v^\top\left(\mu(\mathcal{M}^{(v)})-\mu^*\right)\right|\\ &\le 7c_\zeta\sqrt{\alpha},\end{aligned}$$

where (a) holds by the definition of the distance :

$$\|\mu-\mu^*\|_2 \;=\; \max_{v\in\mathbb{S}^{d-1}}|v^\top(\mu-\mu^*)|,$$

and (b) holds by triangle inequality.

I.2. Case of sub-Gaussian distributions and a proof of Theorem 9

Th proof is analogous to the previous section, we only state the lemmas that differ. $q_{\rm range}$ returns a hypercube $\bar x+[-B/2,B/2]^d$ that includes all uncorrupted data points with a high probability.

Lemma I.10 (α-corrupted data has small $R(S)$)**.** *Let S be the set of α-corrupted data. Given that $n=\Omega(\frac{d+\log(1/\zeta)}{\alpha^2\log 1/\alpha})$, with probability $1-\zeta$, $R(S)\le 3\,\alpha\sqrt{\log(1/3\alpha)}$.*

Lemma I.11 (Privacy). *Algorithm 3 is (ε, δ)-differentially private if $n \geq 3B\sqrt{d}\log(3/\delta)/(\varepsilon\alpha\sqrt{\log(1/\alpha)})$.*

This follows from the following lemma.

Lemma I.12 (Sensitivity of $d(\hat{\mu}, S)$). *Given that $R(S) \leq 3\alpha\sqrt{\log(1/\alpha)}$, for any neighboring dataset S', $|d(\hat{\mu}, S) - d(\hat{\mu}, S')| \leq 12\sqrt{\log 1/\alpha}/n$.*

Lemma I.13 ($d(\hat{\mu}, S)$ approximates $\|\hat{\mu} - \mu^*\|$). *Let S be the set of α-corrupted data. Given that $n = \Omega(\frac{d+\log(1/\zeta)}{\alpha^2 \log 1/\alpha})$, with probability $1 - \zeta$,*

$$\left| d(\hat{\mu}, S) - \|\hat{\mu} - \mu^*\|_2 \right| \quad \leq \quad 14\,\alpha\,\sqrt{\log 1/\alpha}\,.$$

This implies the following utility bound.

Lemma I.14 (Utility of the algorithm). *For an α-corrupted dataset S, Algorithm 3 achieves $\|\hat{\mu} - \mu^*\|_2 \leq \alpha\sqrt{\log 1/\alpha}$ with probability $1 - \zeta$, if $n = \Omega((d + \log(1/\zeta))/(\alpha^2 \log(1/\alpha)) + (d\log(dR/\alpha) + \log(1/\zeta)/(\varepsilon\alpha))$.*

J. The algorithmic details and the analysis of PRIME-HT for covariance bounded distributions

We provide the algorithm and the analysis for the range estimation query $q_{\mathrm{range-ht}}$, and then prove the result on analyzing PRIME-HT.

J.1. Range estimation with $q_{\mathrm{range-ht}}$

Algorithm 14: Differentially private range estimation for covariance bounded distributions ($q_{\mathrm{range-ht}}$) (**?**)Algorithm 2]kamath2020private

Input: $S = \{x_i\}_{i=1}^n$, R, ε, δ, ζ

Randomly partition the dataset $S = \cup_{\ell\in[m]} S^{(\ell)}$ with $m = 200\log(2/\zeta)$

$\bar{x}^{(\ell)} \leftarrow q_{\mathrm{range}}(S^{(\ell)}, R, \varepsilon/m, \delta/m, \sigma = 40)$ for all $\ell \in [m]$

$\hat{x}_j \leftarrow \mathrm{median}(\{\bar{x}_j^{(\ell)}\}_{\ell\in[m]})$ for all $j \in [d]$

Output: $(\hat{x}, B = 50/\sqrt{\alpha})$

Lemma J.1. *$q_{\mathrm{range-ht}}$ is (ε,δ)-differentially private. Under Assumption 2 and for $\alpha \in (0, 0.01)$, if $n = \Omega((1/\alpha)\log(1/\zeta) + (\sqrt{d\log(1/\delta)}\log(1/\zeta)/\varepsilon)\min\{\log(dR), \log(d/\delta)\})$, $q_{\mathrm{range-ht}}$ returns a ball $\mathcal{B}_{\sqrt{d}B/2}(\bar{x})$ of radius $\sqrt{d}B/2$ centered at $\bar{x}$ that includes $(1-2\alpha)n$ uncorrupted samples where $B = 50/\sqrt{\alpha}$ with probability $1-\zeta$.*

We first show that applying the private histogram to each coordinate provides a robust estimate of the range, but with a constant probability 0.9.

Lemma J.2 (Robustness of a single private histogram)**.** *Under the α-corruption model of Assumption 2, if $n = \Omega((\sqrt{d\log(1/\delta)}/\varepsilon)\min\{\log(dR), \log(d/\delta)\})$, for $\alpha \in (0, 0.01)$, q_{range} in Algorithm 5 with a choice of $\sigma = 40$ and $B = 120$ returns intervals $\{I_j\}_{j=1}^d$ of size $|I_j| = 240$ such that $\mu_j \in I_j$ with probability 0.9 for each $j \in [d]$.*

Proof of Lemma J.2. The proof is analogous to Appendix E.1 and we only highlight the differences here. By Lemma E.1 we know that $|\tilde{p}_k - \hat{p}_k| \le 0.01$ with the assumption on n. The corruption can change the normalized count in each bin by $\alpha \le 0.01$ by assumption. It follows from Chebyshev inequality that $\mathbb{P}(|x_{i,j} - \mu_j|^2 > \sigma^2) \le 1/\sigma^2$. It follows from (e.g. (**?**)Lemma A.3]kamath2020private) that $\mathbb{P}(|\{i : x_{i,j} \notin [\mu-\sigma, \mu+\sigma]\}| > (100/\sigma^2)n) < 0.05$. Hence the maximum bin has $\tilde{p}_k \ge 0.5(1 - 100/\sigma^2) - 0.02$ and the true mean is in the maximum bin or in an adjacent bin. The largest non-adjacent bucket is at most $100/\sigma^2 + 0.02$. Hence, the choice of $\sigma = 40$ ensures that we find the μ within $3\sigma = 120$.

□

Following (**?**)Algorithm 2]kamath2020private, we partition the dataset into $m = 200\log(2/\zeta)$ subsets of an equal size n/m and apply the median-of-means approach. Applying Lemma J.2, it is ensured (e.g., by (**?**)Lemma A.4]kamath2020private) that more than half of the partitions satisfy that the center of the interval is within 240 away from μ, with probability $1-\zeta$. Therefore the median of those m centers is within 240 from the true mean in each coordinate. This requires the total sample size larger only by a factor of $\log(d/\zeta)$.

To choose a radius $\sqrt{d}B/2$ ball around this estimated mean that includes $1-\alpha$ fraction of the points, we choose $B = 25/\sqrt{\alpha}$. Since $\|\hat{\mu} - \mu\|_2 \le 120\sqrt{d} \ll \sqrt{d}B/2$ for $\alpha \le 0.01$, this implies that we can choose $\sqrt{d}B/2$-ball around the estimated mean with $B = 50/\sqrt{\alpha}$.

Let $z_i = \mathbb{I}(\|x_i - \mu\|_2 > \sqrt{d}B/2)$. We know that $\mathbb{E}[z_i] = \mathbb{P}[(\|x_i-\mu\|_2 > \sqrt{d}B/2)] \le \mathbb{E}[\|x_i-\mu\|_2^2(2/dB^2)] = (1/1250)\alpha$. Applying multiplicative Chernoff bound (e.g., in (**?**)Lemma A.3]kamath2020private), we get $|\{i : \|x_i-\mu\|_2 \le \sqrt{d}B/2\}| \ge 1 - (3/2500)\alpha$ with probability $1-\zeta$, if $n = \Omega((1/\alpha)\log(1/\zeta))$. This ensures that with high probability, $(1-\alpha)$ fraction of the original uncorrupted points are included in the ball. Since the adversary can corrupt αn samples, at least $(1-2\alpha)n$ of the remaining good points will be inside the ball.

J.2. Proof of Theorem 8

The proof of the privacy guarantee of Algorithm 15 follows analogously from the proof of the privacy of PRIME and is omitted here. The accuracy guarantee follows form the following theorem and Lemma J.1.

Theorem 12 (Analysis of accuracy of DPMMWFILTER-HT)**.** *Let S be an α-corrupted covariance bounded dataset under Assumption 2, where $\alpha \le c$ for some universal constant $c \in (0, 1/2)$. Let S_{good} be α-good with respect to $\mu \in \mathbb{R}^d$. Suppose $\mathcal{D} = \{x_i \in \mathcal{B}_{\sqrt{d}B/2}(\bar{x})\}_{i=1}^n$ be the projected dataset. If $n \ge \widetilde{\Omega}\left(\frac{d^{3/2}B^2\log(1/\delta)}{\varepsilon}\right)$, then* DPMMWFILTER-HT *terminates after at most $O(\log dB^2)$ epochs and outputs $S^{(s)}$ such that with probability 0.9, we have $|S_t^{(s)} \cap S_{\mathrm{good}}| \ge (1-10\alpha)n$ and*

$$\|\mu(S^{(s)}) - \mu\|_2 \lesssim \sqrt{\alpha}\,.$$

Moreover, each epoch runs for at most $O(\log d)$ iterations.

Algorithm 15: Differentially private filtering with matrix multiplicative weights (DPMMWFILTER-HT) for distributions with bounded covariance

Input: $S = \{x_i \in \mathcal{B}_{\sqrt{d}B/2}(\bar{x})\}_{i=1}^n$, $\alpha \in (0,1)$, $T_1 = O(\log B\sqrt{d})$, $T_2 = O(\log d)$, $B \in \mathbb{R}_+$, (ε, δ)

if $n < (4/\varepsilon_1)\log(1/(2\delta_1))$ **then Output:** $\emptyset$

Initialize $S^{(1)} \leftarrow [n]$, $\varepsilon_1 \leftarrow \varepsilon/(4T_1)$, $\delta_1 \leftarrow \delta/(4T_1)$, $\varepsilon_2 \leftarrow \min\{0.9, \varepsilon\}/(4\sqrt{10T_1T_2\log(4/\delta)})$, $\delta_2 \leftarrow \delta/(20T_1T_2)$, a large enough constant $C > 0$

for epoch $s = 1, 2, \ldots, T_1$ **do**

- $\lambda^{(s)} \leftarrow \|M(S^{(s)})\|_2 + \mathrm{Lap}(2B^2d/(n\varepsilon_1))$
- $n^{(s)} \leftarrow |S^{(s)}| + \mathrm{Lap}(1/\varepsilon_1)$
- **if** $n^{(s)} \le 3n/4$ **then** terminate
- **if** $\lambda^{(s)} \le C$ **then**
  - **Output:** $\mu^{(s)} \leftarrow (1/|S^{(s)}|)\big(\sum_{i\in S^{(s)}} x_i\big) + \mathcal{N}(0, (2B\sqrt{2d\log(1.25/\delta_1)}/(n\,\varepsilon_1))^2\mathbf{I}_{d\times d})$
- $\alpha^{(s)} \leftarrow 1/(100(0.1/C + 1.05)\lambda^{(s)})$
- $S_1^{(s)} \leftarrow S^{(s)}$
- **for** $t = 1, 2, \ldots, T_2$ **do**
  - $\lambda_t^{(s)} \leftarrow \|M(S_t^{(s)})\|_2 + \mathrm{Lap}(2B^2d/(n\varepsilon_2))$
  - **if** $\lambda_t^{(s)} \le 2/3\lambda_0^{(s)}$ **then**
    - terminate epoch
  - **else**
    - $\Sigma_t^{(s)} \leftarrow M(S_t^{(s)}) + \mathcal{N}(0, (4B^2d\sqrt{2\log(1.25/\delta_2)}/(n\varepsilon_2))^2\mathbf{I}_{d^2\times d^2})$
    - $U_t^{(s)} \leftarrow (1/\operatorname{Tr}(\exp(\alpha^{(s)}\sum_{r=1}^t(\Sigma_r^{(s)})))\exp(\alpha^{(s)}\sum_{r=1}^t(\Sigma_r^{(s)}))$
    - $\psi_t^{(s)} \leftarrow \left\langle M(S_t^{(s)}), U_t^{(s)}\right\rangle + \mathrm{Lap}(2B^2d/(n\varepsilon_2))$
    - **if** $\psi_t^{(s)} \le (1/5.5)\lambda_t^{(s)}$ **then**
      - $S_{t+1}^{(s)} \leftarrow S_t^{(s)}$
    - **else**
      - $Z_t^{(s)} \leftarrow \mathrm{Unif}([0,1])$
      - $\mu_t^{(s)} \leftarrow (1/|S_t^{(s)}|)\big(\sum_{i\in S_t} x_i\big) + \mathcal{N}(0, (2B\sqrt{2d\log(1.25/\delta_2)}/(n\,\varepsilon_2)\mathbf{I}_{d\times d})^2)$
      - $\rho_t^{(s)} \leftarrow$ DPTHRESHOLD-HT$(\mu_t^{(s)}, U_t^{(s)}, \alpha, \varepsilon_2, \delta_2, S_t^{(s)})$ [Algorithm 16]
      - $S_{t+1}^{(s)} \leftarrow S_t^{(s)} \setminus \{i \mid \{\tau_j = (x_j - \mu_t^{(s)})^\top U_t^{(s)}(x_j - \mu_t^{(s)})\}_{j\in S_t^{(s)}}$ and $\tau_i \ge \rho_t^{(s)}\, Z_t^{(s)}\}$.
- $S^{(s+1)} \leftarrow S_t^{(s)}$

Output: $\mu^{(T_1)}$

J.2.1. ANALYSIS OF DPMMWFILTER-HT AND A PROOF OF THEOREM 12

Algorithm 15 is a similar matrix multiplicative weights based filter algorithm for distributions with bounded covariance. Similarly, we first state following Lemma J.3 and prove Theorem 12 given Lemma J.3

Lemma J.3. *Let S be an α-corrupted bounded covariance dataset under Assumption 2. For an epoch s and an iteration t such that $\lambda^{(s)} > C$, $\lambda_t^{(s)} > 2/3\lambda_0^{(s)}$, and $n^{(s)} > 3n/4$, if $n \gtrsim \frac{B^2(\log B)d^{3/2}\log(1/\delta)}{\varepsilon}$ and $|S_t^{(s)} \cap S_{\mathrm{good}}| \ge (1-10\alpha)n$,*

Algorithm 16: Differentially private estimation of the threshold for bounded covariance DPTHRESHOLD-HT

Input: $\mu, U, \alpha \in (0,1)$, target privacy (ε, δ), $S = \{x_i \in \mathcal{B}_{B\sqrt{d}/2}(\bar{x})\}$

Set $\tau_i \leftarrow (x_i - \mu)^\top U (x_i - \mu)$ for all $i \in S$

Set $\tilde{\psi} \leftarrow (1/n)\sum_{i\in S}\tau_i + \mathrm{Lap}(2B^2 d/n\varepsilon))$

Compute a histogram over geometrically sized bins

$I_1 = [1/4, 1/2), I_2 = [1/2, 1), \ldots, I_{2+\log(B^2 d)} = [2^{\log(B^2 d)-1}, 2^{\log(B^2 d)}]$

$$h_j \leftarrow \frac{1}{n}\cdot |\{i \in S \,|\, \tau_i \in [2^{-3+j}, 2^{-2+j})\}|\,, \qquad \text{for all } j = 1, \cdots, 2+\log(B^2 d)$$

Compute a privatized histogram $\tilde{h}_j \leftarrow h_j + \mathcal{N}(0, (4\sqrt{2d\log(1.25/\delta)}/(n\varepsilon))^2)$, for all $j \in [2+\log(B^2 d)]$

Set $\tilde{\tau}_j \leftarrow 2^{-3+j}$, for all $j \in [2+\log(B^2 d)]$

Find the largest $\ell \in [2+\log(B^2 d)]$ satisfying $\sum_{j\ge\ell}(\tilde{\tau}_j - \tilde{\tau}_\ell)\,\tilde{h}_j \ge 0.31\tilde{\psi}$

Output: $\rho = \tilde{\tau}_\ell$

then with probability $1 - O(1/\log(d)^3)$*, we have the condition in Eq.* (19) *holds. When this condition holds, we have more corrupted samples are removed in expectation than the uncorrupted samples, i.e.,* $\mathbb{E}|(S_t^{(s)} \setminus S_{t+1}^{(s)}) \cap S_{\mathrm{good}}| \le \mathbb{E}|(S_t^{(s)} \setminus S_{t+1}^{(s)}) \cap S_{\mathrm{bad}}|$*. Further, for an epoch* $s \in [T_1]$ *there exists a constant* $C > 0$ *such that if* $\|M(S^{(s)})\|_2 \ge C$*, then with probability* $1 - O(1/\log^2 d)$*, the* s*-th epoch terminates after* $O(\log d)$ *iterations and outputs* $S^{(s+1)}$ *such that* $\|M(S^{(s+1)})\|_2 \le 0.98\|M(S^{(s)})\|_2$.

Now we define $d_t^{(s)} \triangleq |(S_{\mathrm{good}} \cap S^{(1)}) \setminus S_t^{(s)}| + |S_t^{(s)} \setminus (S_{\mathrm{good}} \cap S^{(1)})|$. Note that $d_1^{(1)} = \alpha n$, and $d_t^{(s)} \ge 0$. At each epoch and iteration, we have

$$\mathbb{E}[d_{t+1}^{(s)} - d_t^{(s)} | d_1^{(1)}, d_2^{(1)}, \cdots, d_t^{(s)}] \;=\; \mathbb{E}\left[|S_{\mathrm{good}} \cap (S_t^{(s)} \setminus S_{t+1}^{(s)})| - |S_{\mathrm{bad}} \cap (S_t^{(s)} \setminus S_{t+1}^{(s)})|\right] \;\le\; 0,$$

from the part 1 of Lemma J.3. Hence, $d_t^{(s)}$ is a non-negative super-martingale. By optional stopping theorem, at stopping time, we have $\mathbb{E}[d_t^{(s)}] \le d_1^{(1)} = \alpha n$. By Markov inequality, $d_t^{(s)}$ is less than $10\alpha n$ with probability 0.9, i.e. $|S_t^{(s)} \cap S_{\mathrm{good}}| \ge (1 - 10\alpha)n$. The desired bound in Theorem 12 follows from Lemma J.11.

J.2.2. PROOF OF LEMMA J.3

Lemma J.3 is a combination of Lemma J.4, Lemma J.5 and Lemma J.6. We state the technical lemmas and subsequently provide the proofs.

Lemma J.4. *For each epoch* s *and iteration* t*, under the hypotheses of Lemma J.3 then with probability* $1 - O(1/\log^3 d)$*, we have*

$$\frac{1}{n}\sum_{i \in S_{\mathrm{good}} \cap S_t^{(s)}} \tau_i \;\le\; \psi/1000\,, \tag{19}$$

where $\psi \triangleq \frac{1}{n}\sum_{i\in S_t^{(s)}}\tau_i$.

Lemma J.5. *For each epoch* s *and iteration* t*, under the hypotheses of Lemma J.3, if condition Eq.* (19) *holds, then we have* $\mathbb{E}|S_t^{(s)} \setminus S_{t+1}^{(s)} \cap S_{\mathrm{good}}| \le \mathbb{E}|S_t^{(s)} \setminus S_{t+1}^{(s)} \cap S_{\mathrm{bad}}|$ *and with probability* $1 - O(1/\log^3 d)$*, and* $\left\langle M(S_{t+1}^{(s)}), U_t^{(s)} \right\rangle \le 0.76 \left\langle M(S_t^{(s)}), U_t^{(s)} \right\rangle$.

Lemma J.6. *For epoch* s*, suppose for* $t = 0, 1, \cdots, T_2$ *where* $T_2 = O(\log d)$*, if Lemma J.5 holds,* $n \gtrsim \frac{B^2(\log B) d^{3/2}\log(1/\delta)}{\varepsilon\alpha}$*, and* $n^{(s)} > 3n/4$*, then we have* $\|M(S^{(s+1)})\|_2 \le 0.98\|M(S^{(s)})\|_2$ *with probability* $1 - O(1/\log^2 d)$.

J.2.3. PROOF OF LEMMA J.4

Proof. By Lemma H.9, Lemma H.10 and Lemma H.11, we can pick $n = \widetilde{\Omega}\left(\frac{B^2 d^{3/2}\log}{\varepsilon}\right)$ such that with probability $1 - O(1/\log^3 d)$, following conditions simultaneously hold:

1. $\|\mu_t^{(s)} - \mu(S_t^{(s)})\|_2^2 \le 0.001$
2. $|\psi_t^{(s)} - \left\langle M(S_t^{(s)}), U_t^{(s)}\right\rangle| \le 0.001$
3. $\left|\lambda_t^{(s)} - \|M(S_t^{(s)})\|_2\right| \le 0.001$
4. $\left|\lambda^{(s)} - \|M(S^{(s)})\|_2\right| \le 0.001$
5. $\left\|M(S_{t+1}^{(s)}) - \Sigma_t^{(s)}\right\|_2 \le 0.001$
6. $\|\mu^{(s)} - \mu(S^{(s)})\|_2^2 \le 0.001$.

Then we have

$$\begin{aligned}
\frac{1}{n}\sum_{i\in S_{\text{good}}\cap S_t^{(s)}}\tau_i &= \frac{1}{n}\sum_{i\in S_{\text{good}}\cap S_t^{(s)}}\left\langle (x_i-\mu_t^{(s)})(x_i-\mu_t^{(s)})^\top, U_t^{(s)}\right\rangle \\
&\overset{(a)}{\le} \frac{2}{n}\sum_{i\in S_{\text{good}}\cap S_t^{(s)}}\left\langle (x_i-\mu(S_{\text{good}}\cap S_t^{(s)}))(x_i-\mu(S_{\text{good}}\cap S_t^{(s)}))^\top, U_t^{(s)}\right\rangle \\
&\quad + \frac{2|S_{\text{good}}\cap S_t^{(s)}|}{n}\left\langle (\mu(S_{\text{good}}\cap S_t^{(s)})-\mu_t^{(s)})(\mu(S_{\text{good}}\cap S_t^{(s)})-\mu_t^{(s)})^\top, U_t^{(s)}\right\rangle \\
&\le 2\left\langle M((S_{\text{good}}\cap S_t^{(s)}), U_t^{(s)}\right\rangle + 2\|\mu_t^{(s)} - \mu(S_{\text{good}}\cap S_t^{(s)})\|_2^2 \\
&\overset{(b)}{\le} 2 + 2\left(\|\mu_t^{(s)}-\mu\|_2 + \|\mu(S_{\text{good}}\cap S_t^{(s)})-\mu\|_2\right)^2 \\
&\overset{(c)}{\le} 2 + 2\left(0.01 + 2\sqrt{\alpha\|M(S_t^{(s)})\|_2} + 3\sqrt{\alpha}\right)^2 \\
&\le 3 + 8\alpha\|M(S_t^{(s)})\|_2 + 32\alpha \\
&\overset{(d)}{\le} \frac{\psi_t^{(s)} - 0.002}{1000} \\
&\le \frac{\psi}{1000}\,,
\end{aligned}$$

where (a) follows from the fact that for any vector x, y, z, we have $(x-y)(x-y)^\top \preceq 2(x-z)(x-z)^\top + 2(y-z)(y-z)^\top$, (b) follows from α-goodness of S_{good}, (c) follows from Lemma J.11 and (d) follows from our choice of large constant C and sample complexity n.

□

J.2.4. PROOF OF LEMMA J.5

Proof. Lemma J.4 implies with probability $1 - O(1/\log^3 d)$, our scores satisfies the condition in Eq. (19). Then by Lemma J.7 our DPTHRESHOLD-HT gives us a threshold ρ such that

$$\sum_{i\in S_{\text{good}}\cap S_t^{(s)}} \mathbf{1}\{\tau_i \le \rho\}\frac{\tau_i}{\rho} + \mathbf{1}\{\tau_i > \rho\} \le \sum_{i\in S_{\text{bad}}\cap S_t^{(s)}} \mathbf{1}\{\tau_i \le \rho\}\frac{\tau_i}{\rho} + \mathbf{1}\{\tau_i > \rho\}\,.$$

According to our filter rule from Algorithm 16, we have

$$\mathbb{E}|(S_t^{(s)} \setminus S_{t+1}^{(s)}) \cap S_{\mathrm{good}}| = \sum_{i \in S_{\mathrm{good}} \cap S_t^{(s)}} \mathbf{1}\{\tau_i \le \rho\}\frac{\tau_i}{\rho} + \mathbf{1}\{\tau_i > \rho\}$$

and

$$\mathbb{E}|(S_t^{(s)} \setminus S_{t+1}^{(s)}) \cap S_{\mathrm{bad}}| = \sum_{i \in S_{\mathrm{bad}} \cap S_t^{(s)}} \mathbf{1}\{\tau_i \le \rho\}\frac{\tau_i}{\rho} + \mathbf{1}\{\tau_i > \rho\}\ .$$

This implies $\mathbb{E}|(S_t^{(s)} \setminus S_{t+1}^{(s)}) \cap S_{\mathrm{good}}| \le \mathbb{E}|(S_t^{(s)} \setminus S_{t+1}^{(s)}) \cap S_{\mathrm{bad}}|$.

At the same time, Lemma J.7 gives us a ρ such that with probability $1 - O(\log^3 d)$, we have

$$\frac{1}{n}\sum_{i \in S_{t+1}^{(s)}} \tau_i \le \frac{1}{n}\sum_{\tau_i \le \rho, i \in S_t^{(s)}} \tau_i \le \frac{3}{4}\cdot\frac{1}{n}\sum_{i \in S_t^{(s)}} \tau_i\ .$$

Hence, we have

$$\begin{aligned}
\left\langle M(S_{t+1}^{(s)}), U_t^{(s)} \right\rangle &= \left\langle \frac{1}{n}\sum_{i \in S_{t+1}^{(s)}} (x_i - \mu(S_{t+1}^{(s)}))(x_i - \mu(S_{t+1}^{(s)}))^\top, U_t^{(s)} \right\rangle \\
&\le \left\langle \frac{1}{n}\sum_{i \in S_{t+1}^{(s)}} (x_i - \mu(S_t^{(s)}))(x_i - \mu(S_t^{(s)}))^\top, U_t^{(s)} \right\rangle \\
&\le \frac{1}{n}\sum_{i \in S_{t+1}^{(s)}} \tau_i + \|\mu_t^{(s)} - \mu(S_t^{(s)})\|_2^2 \\
&\le \frac{3}{4n}\sum_{i \in S_t^{(s)}} \tau_i + 0.01 \\
&\overset{(a)}{\le} 0.76\left\langle M(S_t^{(s)}), U_t^{(s)} \right\rangle\ ,
\end{aligned}$$

where (a) follows from our assumption that $\psi_t^{(s)} > \frac{1}{5.5}\lambda_t^{(s)} > \frac{2}{16.5}C$.

□

J.2.5. Proof of Lemma J.6

Proof. If Lemma J.5 holds, we have

$$\begin{aligned}
\left\langle M(S_t^{(s)}), U_t^{(s)} \right\rangle &\le 0.76\left\langle M(S_{t-1}^{(s)}), U_t^{(s)} \right\rangle \\
&\le 0.76\left\langle M(S_1^{(s)}), U_t^{(s)} \right\rangle \\
&\le 0.76\|M(S_1^{(s)})\|_2
\end{aligned}$$

We pick n large enough such that with probability $1 - O(\log^3 d)$,

$$\|\Sigma_t^{(s)}\|_2 \approx_{0.05} \|M(S_t^{(s)})\|_2\ .$$

Thus, we have

$$\left\langle \Sigma_t^{(s)}, U_t^{(s)} \right\rangle \le 0.81\|M(S_1^{(s)})\|_2\ .$$

By Lemma H.1, we have $M(S_t^{(s)}) \preceq M(S_1^{(s)})$. by our choice of $\alpha^{(s)}$, we have $\alpha^{(s)} M(S_{t+1}^{(s)}) \preceq \frac{1}{100}\mathbf{I}$ and $\alpha^{(s)}\Sigma_t^{(s)} \preceq \frac{1}{100}\mathbf{I}$. Therefore, by Lemma H.14 we have

$$\begin{aligned}
&\left\|\sum_{i=1}^{T_2}\Sigma_t^{(s)}\right\|_2 \\
\leq\ & \sum_{t=1}^{T_2}\left\langle \Sigma_t^{(s)}, U_t^{(s)}\right\rangle + \alpha^{(s)}\sum_{t=0}^{T_2}\left\langle U_t^{(s)}, \left|\Sigma_t^{(s)}\right|\right\rangle \|\Sigma_t^{(s)}\|_2 + \frac{\log(d)}{\alpha^{(s)}} \\
\overset{(a)}{\leq}\ & \sum_{t=1}^{T_2}\left\langle \Sigma_t^{(s)}, U_t^{(s)}\right\rangle + \frac{1}{100}\sum_{t=1}^{T_2}\left\langle U_t^{(s)}, \left|\Sigma_t^{(s)}\right|\right\rangle + 200\log(d)\|M(S_1^{(s)})\|_2
\end{aligned}$$

where (a) follows from our choice of $\alpha^{(s)}$, C, and n.

Meanwhile, we have

$$|\Sigma_t^{(s)}| \preceq M(S_t^{(s)}) + 0.15\,\mathbf{I}\,.$$

Thus we have

$$\left\langle U_t^{(s)}, \left|\Sigma_t^{(s)}\right|\right\rangle \leq 0.91\ \left\|M(S_1^{(s)})\right\|_2$$

Then we have

$$\begin{aligned}
\left\|M(S_{T_2}^{(s)})\right\|_2 &\leq \frac{1}{T_2}\left\|\sum_{i=1}^{T_2} M(S_t^{(s)})\right\|_2 \\
&\leq \frac{1}{T_2}\left\|\sum_{i=1}^{T_2}\Sigma_t^{(s)}\right\|_2 + 0.05\ \|M(S_1^{(s)})\|_2 \\
&\leq \frac{1}{T_2}\left(\sum_{t=1}^{T_2}\left\langle \Sigma_t^{(s)}, U_t^{(s)}\right\rangle + \frac{1}{100}\sum_{t=1}^{T_2}\left\langle U_t^{(s)}, \left|\Sigma_t^{(s)}\right|\right\rangle + 200\log(d)\|M(S_1^{(s)})\|_2\right) + 0.05\ \|M(S_1^{(s)})\|_2 \\
&\leq 0.91\|M(S_1^{(s)})\|_2 + \frac{200\log(d)}{T_2}\|M(S_1^{(s)})\|_2 + 0.05\ \|M(S_1^{(s)})\|_2 \\
&\leq 0.98\ \|M(S_1^{(s)})\|_2
\end{aligned}$$

□

J.2.6. Proof of DPThreshold-HT for distributions with bounded covariance

Lemma J.7 (DPThreshold-HT: picking threshold privately for distributions with bounded covariance)**.** *Algorithm* DPThreshold-HT$(\mu, U, \alpha, \varepsilon, \delta, S)$ *running on a dataset* $\{\tau_i = (x_i-\mu)^\top U(x_i-\mu)\}_{i\in S}$ *is* (ε,δ)*-DP. Define* $\psi \triangleq \frac{1}{n}\sum_{i\in S}\tau_i$*. If* τ_i*'s satisfy*

$$\frac{1}{n}\sum_{i\in S_{\text{good}}\cap S}\tau_i \quad\leq\quad \psi/1000\,,$$

and $n \geq \tilde{\Omega}\left(\frac{B^2 d}{\varepsilon}\right)$ *then* DPThreshold-HT *outputs a threshold* ρ *such that*

$$2\Big(\sum_{i\in S_{\text{good}}\cap S}\mathbf{1}\{\tau_i\leq\rho\}\frac{\tau_i}{\rho} + \mathbf{1}\{\tau_i>\rho\}\Big) \leq \sum_{i\in S_{\text{bad}}\cap S}\mathbf{1}\{\tau_i\leq\rho\}\frac{\tau_i}{\rho} + \mathbf{1}\{\tau_i>\rho\}\,, \tag{20}$$

and with probability $1 - O(1/\log^3 d)$*,*

$$\frac{1}{n}\sum_{\tau_i<\rho}\tau_i \quad\leq\quad 0.75\psi\,.$$

Proof. **1. ρ cuts enough**

Let ρ be the threshold picked by the algorithm. Let $\hat{\tau}_i$ denote the minimum value of the interval of the bin that τ_i belongs to. It holds that

$$\begin{aligned}
\frac{1}{n}\sum_{\tau_i\geq\rho, i\in[n]}(\tau_i-\rho) &\geq \frac{1}{n}\sum_{\hat{\tau}_i\geq\rho, i\in[n]}(\hat{\tau}_i-\rho)\\
&= \sum_{\tilde{\tau}_j\geq\rho, j\in[2+\log(B^2d)]}(\tilde{\tau}_j-\rho)h_j\\
&\overset{(a)}{\geq} \sum_{\tilde{\tau}_j\geq\rho, j\in[2+\log(B^2d)]}(\tilde{\tau}_j-\rho)\tilde{h}_j - O\left(\log(B^2d)\cdot B^2d\cdot\frac{\sqrt{\log(\log(B^2d)\log d)\log(1/\delta)}}{\varepsilon n}\right)\\
&\overset{(b)}{\geq} 0.31\tilde{\psi} - \tilde{O}(\frac{B^2d}{\varepsilon n})\\
&\overset{(c)}{\geq} 0.3\psi - \tilde{O}(\frac{B^2d}{\varepsilon n}),
\end{aligned}$$

where (a) holds due to the accuracy of the private histogram (Lemma H.12), (b) holds by the definition of ρ in our algorithm, and (c) holds due to the accuracy of $\tilde{\psi}$. This implies

$$\frac{1}{n}\sum_{\tau_i<\rho}\tau_i \leq \psi - \frac{1}{n}\sum_{\tau_i\geq\rho}(\tau_i-\rho) \leq 0.7\psi + \tilde{O}(B^2d/\varepsilon n).$$

2. ρ doesn't cut too much

Define C_2 to be the threshold such that $\frac{1}{n}\sum_{\tau_i>C_2}(\tau_i-C_2)=(2/3)\psi$. Suppose $2^b\leq C_2\leq 2^{b+1}$, we have $\sum_{\hat{\tau}_i\geq 2^{b-1}}(\hat{\tau}_i - 2^{b-1})\geq(1/3)\psi$ because $\forall\tau_i\geq C_2$, $(\hat{\tau}_i-2^{b-1})\geq\frac{1}{2}(\tau_i-C_2)$. Then the threshold picked by the algorithm $\rho\geq 2^{b-1}$, which implies $\rho\geq\frac{1}{4}C_2$. Suppose $\rho<C_2$, since $\rho\geq\frac{1}{4}C_2$

$$\begin{aligned}
\sum_{i\in S_{\text{bad}}\cap S,\tau_i<\rho}\tau_i + \sum_{i\in S_{\text{bad}}\cap S,\tau_i\geq\rho}\rho &\geq \frac{1}{4}(\sum_{i\in S_{\text{bad}}\cap S,\tau_i<C_2}\tau_i + \sum_{i\in S_{\text{bad}}\cap S,\tau_i\geq C_2}C_2)\\
&\overset{(a)}{\geq} \frac{10}{4}(\sum_{i\in S_{\text{good}}\cap S,\tau_i<C_2}\tau_i + \sum_{i\in S_{\text{good}}\cap S,\tau_i\geq C_2}C_2)\\
&\overset{(b)}{\geq} \frac{10}{4}(\sum_{i\in S_{\text{good}}\cap S,\tau_i<\rho}\tau_i + \sum_{i\in S_{\text{good}}\cap S,\tau_i\geq\rho}\rho),
\end{aligned}$$

where (a) holds by Lemma J.8, and (b) holds since $\rho\leq C_2$. If $\rho\geq C_2$, the statement of the Lemma J.8 directly implies Equation (20).

Lemma J.8. *Assuming that the condition in Eq.(19) holds, then for any C such that*

$$\frac{1}{n}\sum_{i\in S,\tau_i<C}\tau_i + \frac{1}{n}\sum_{i\in S,\tau_i\geq C}C \geq (1/3)\psi,$$

we have

$$\sum_{i\in S_{\text{bad}}\cap S,\tau_i<C}\tau_i + \sum_{i\in S_{\text{bad}}\cap S,\tau_i\geq C}C \geq 10(\sum_{i\in S_{\text{good}}\cap S,\tau_i<C}\tau_i + \sum_{i\in S_{\text{good}}\cap S,\tau_i\geq C}C)$$

Proof. First we show an upper bound on S_{good}:

$$\frac{1}{n}\sum_{i\in S_{\text{good}}\cap S,\tau_i<C}\tau_i + \frac{1}{n}\sum_{i\in S_{\text{good}}\cap S,\tau_i\geq C}C \leq \frac{1}{n}\sum_{i\in S_{\text{good}}\cap S}\tau_i \leq \psi/1000.$$

Then we show an lower bound on S_{bad}:

$$\begin{aligned} & \frac{1}{n}\sum_{i\in S_{\mathrm{bad}}\cap S,\tau_i<C}\tau_i+\frac{1}{n}\sum_{i\in S_{\mathrm{bad}}\cap S,\tau_i>C}C \\ = \quad & \frac{1}{n}\sum_{i\in S,\tau_i<C}\tau_i+\frac{1}{n}\sum_{i\in S,\tau_i\geq C}C \\ & -(\frac{1}{n}\sum_{i\in S_{\mathrm{good}}\cap S,\tau_i<C}\tau_i+\frac{1}{n}\sum_{i\in S_{\mathrm{good}}\cap S,\tau_i\geq C}C) \\ \geq \quad & (1/3-1/1000)\psi\ . \end{aligned}$$

Combing the lower bound and the upper bound yields the desired statement □

□

J.2.7. Regularity lemmas for distributions with bounded covariance

Definition J.9 ((**?**)Definition 3.1]dong2019quantum). *Let D be a distribution with mean $\mu\in\mathbb{R}^d$ and covariance $\Sigma\preceq\mathbf{I}$. For $0<\alpha<1/2$, we say a set of points $S=\{X_1,X_2,\cdots,X_n\}$ is α-good with respect to $\mu\in\mathbb{R}^d$ if following inequalities are satisfied:*

- $\|\mu(S)-\mu\|_2\leq\sqrt{\alpha}$
- $\left\|\frac{1}{|S|}\sum_{i\in S}\left(X_i-\mu(S)\right)\left(X_i-\mu(S)\right)^\top\right\|_2\leq 1.$

Lemma J.10 ((**?**)Lemma 3.1]dong2019quantum). *Let D be a distribution with mean $\mu\in\mathbb{R}^d$ and covariance $\Sigma\preceq\mathbf{I}$. Let $S=\{X_1,X_2,\cdots,X_n\}$ be a set of i.i.d. samples of D. If $n=\Omega(d\log(d)/\alpha)$, then with probability $1-O(1)$, there exists a set $S_{\mathrm{good}}\subseteq S$ such that S_{good} is α-good with respect to μ and $|S_{\mathrm{good}}|\geq(1-\alpha)n$.*

Lemma J.11 ((**?**)Lemma 3.2]dong2019quantum). *Let S be an α-corrupted bounded covariance dataset under Assumption 2. If S_{good} is α-good with respect to μ, then for any $T\subset S$ such that $|T\cap S_{\mathrm{good}}|\geq(1-\alpha)|S|$, we have*

$$\|\mu(T)-\mu\|_2\leq\frac{1}{1-2\alpha}\cdot\left(2\sqrt{\alpha\left\|M(T)\right\|_2}+3\sqrt{\alpha}\right)\ .$$

K. Experiments

We evaluate PRIME and compare with a DP mean estimator of (Kamath et al., 2019) on synthetic dataset in Figure 1 and Figure 2, which consists of samples from $(1-\alpha)\mathcal{N}(0, \mathbf{I}) + \alpha\mathcal{N}(\mu_{\text{bad}}, \mathbf{I})$. The main focus of this evaluation is to compare the estimation error and demonstrate the robustness of PRIME under differential privacy guarantees. Our choice of experimental settings and hyper parameters are as follows: $1 \le d \le 100$, $\mu_{\text{bad}} = (1.5, 1.5, \cdots, 1.5)_d$, $0.001 \le \varepsilon \le 100$, $0.01 \le \alpha \le 0.1$, $R = 10$, $C = 1$. When the algorithm returns $\emptyset$, we simply return the boundary vector i.e. $(R, \cdots, R)_d$.

Figure 2 shows additional experiments including the regime where we do not have enough number of samples. When $n \le cd^{1.5}/\alpha\varepsilon$, the utility guarantee (Theorem 5) does not hold. The noise we add on the final output becomes large as n decreases and dominates the estimation error. The DP Mean (Kamath et al., 2019) has lower error compared to PRIME when n is small because PRIME spends some privacy budget to perform operations other than those in DP Mean in the Algorithm 10. In practice, we can check whether there are enough number of samples based on known parameters $(\varepsilon, \delta, n, \alpha)$, and choose to use DP Mean (or adjust how the privacy budget is distributed in PRIME).

The right figure with $(\alpha, \delta, d, n) = (0.1, 0.01, 10, 10^6)$ is when DP Mean error is dominated by $\alpha\sqrt{d}$ and PRIME by $\alpha\sqrt{\log(1/\alpha)}$ when $\varepsilon > cd^{1.5}/(\alpha n)$. Below this threshold, which happens in this example around $\varepsilon = 0.05$, the added noise in the private mechanism starts to dominate with decreasing ε. Both algorithms have respective thresholds below which the error increases with decreasing ε. This threshold is larger for PRIME because it uses the privacy budget to perform multiple operations and hence the noise added to the final output is larger compared to DP Mean. Below this threshold, which can be easily determined based on the known parameters $(\varepsilon, \delta, n, \alpha)$, we should either collect more data (which will decrease the threshold) or give up filtering and spend all privacy budget on q_{range} and the empirical mean (which will reduce the error).

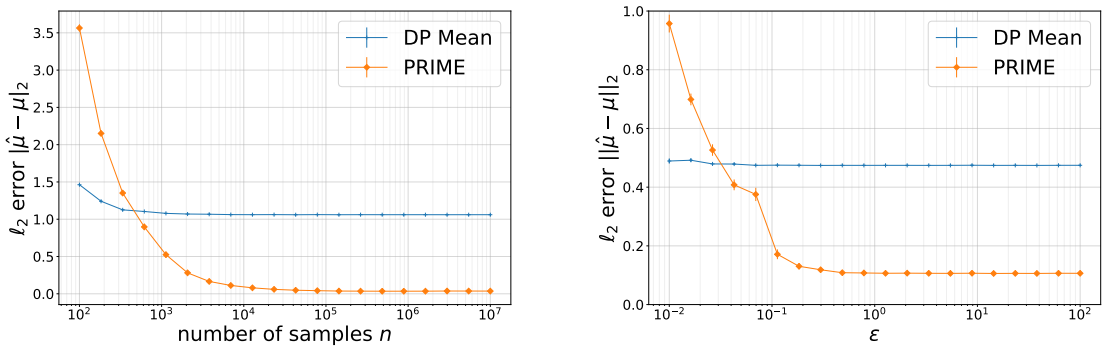

Figure 2. Estimation error achieved by PRIME significantly improves upon that of DP Mean in the large sample regime where our theoretical guarantees apply. In the small sample regime, the noise from the DP mechanisms dominate the error, which increases with decreasing n. We choose $(\alpha, \varepsilon, \delta, d) = (0.1, 100, 0.01, 50)$. Each data point is repeated 50 runs and standard error is shown in the error bar.

Our implementation is based on Python with basic Numpy library. We run on a 2018 Macbook Pro machine. For each choice of d in our settings, it takes less than 2 minutes and PRIME stops after at most 3 epochs. We have attached our code as supplementary materials.