We study the canonical statistical task of computing the principal component from $n$ i.i.d. data in $d$ dimensions under $(\varepsilon, \delta)$-differential privacy. Although extensively studied in literature, existing solutions fall short on two key aspects: (i) even for Gaussian data, existing private algorithms require the number of samples $n$ to scale super-linearly with $d$, i.e., $n = \Omega(d^{3/2})$, to obtain non-trivial results while non-private PCA requires only $n = O(d)$, and (ii) existing techniques suffer from a non-vanishing error even when the randomness in each data point is arbitrarily small. We propose DP-PCA, which is a single-pass algorithm that overcomes both limitations. It is based on a private minibatch gradient ascent method that relies on private mean estimation to add minimal noise required to ensure privacy by adapting to the geometry of a given minibatch of gradients. For sub-Gaussian data, we provide nearly optimal statistical error rates even for $n = \tilde{O}(d)$. Furthermore, we provide a lower bound showing that sub-Gaussian style assumption is necessary in obtaining the optimal error rate.
For non-private PCA with $n$ i.i.d. samples in $d$ dimensions, the popular Oja’s algorithm (provided in Algorithm 1) achieves the optimal error of $\sin(\hat{\vartheta}, v_1) = \Theta(\sqrt{d/n})$, where the error is measured by the sine function of the angle between the estimate, $\hat{\vartheta}$, and the principal component, $v_1$. [45] For differentially private PCA, there is a natural fundamental question: what is the extra cost we pay in the error rate for ensuring $(\epsilon, \delta)$-DP?

We introduce a novel approach we call DP-PCA (Algorithm 3) and show that it achieves an error bounded by $\sin(\hat{\vartheta}, v) = \tilde{O}(\sqrt{d/n + d/\epsilon n})$ for sub-Gaussian-like data defined in Assumption 1, which is a broad class of light-tailed distributions that includes Gaussian data as a special case. The second term characterizes the cost of privacy and this is tight; we prove a nearly matching information theoretic lower bound showing that no $(\epsilon, \delta)$-DP algorithm can achieve a smaller error. This significantly improves upon a long line of existing private algorithms for PCA, e.g., [14, 9, 38, 36, 23]. These existing algorithms are analyzed for fixed and non-stochastic data and achieve sub-optimal error rates of $O(\sqrt{d/n + d^{3/2}/(\epsilon n)})$ even in the stochastic setting we consider.

A remaining question is whether the sub-Gaussian-like assumption, namely Assumption A.4, is necessary or if it is an artifact of our analysis and our algorithm. It turns out that such an assumption on the lightness of the tail is critical; we prove an algorithmic independent and information theoretic lower bound (Theorem 5.4) to show that, without such an assumption, the cost of privacy is lower bounded by $\Omega(\sqrt{d/(\epsilon n)})$. This proves a separation of the error depending on the lightness of the tail.

We start with the formal description of the stochastic setting in Section 2 and present Oja’s algorithm for non-private PCA. Our first attempt in making this algorithm private in Section 3 already achieves near-optimal error, if the data is strictly from a Gaussian distribution. However, there are two remaining challenges that we describe in detail in Section 4 (i) the excessive number of iterations of Private Oja’s Algorithm (Algorithm 2) prevents using typical values of $\epsilon$ used in practice, and (ii) for general sub-Gaussian-like distributions, the error does not vanish even when the noise in the data (as measured by a certain fourth moment of a function of the data) vanishes. The first challenge is due to the analysis that requires amplification by shuffling [24] that is restrictive. The second is due to its reliance on gradient norm clipping [11] that does not adapt to the geometry of the current gradients. This drives the design of DP-PCA in Section 5, that critically relies on two techniques to overcome each challenge, respectively. First, minibatch SGD (instead of single sample SGD) significantly reduces the number of iterations, thus obviating the need for amplification by shuffling. Next, private mean estimation (instead of gradient norm clipping and noise adding) adapts to the geometry of the problem and adds the minimal noise necessary to achieve privacy. The main idea of this geometry adaptive stochastic gradient update is explained in detail in Section 6 with a sketch of a proof.

**Notations.** For a vector $x \in \mathbb{R}^d$, we use $||x||$ to denote the Euclidean norm. For a matrix $X \in \mathbb{R}^{d \times d}$, we use $\|X\|_2 = \max_{\|v\|=1} \|Xv\|_2$ to denote the spectral norm. We use $I_d$ to denote the $d \times d$ identity matrix. For $n \in \mathbb{Z}^+$, let $[n] := \{1, 2, \ldots, n\}$. Let $\mathbb{S}_{d^{-1}}^2$ denote the unit $d$-sphere of $\ell_2$, i.e., $\mathbb{S}_{d^{-1}}^2 := \{x \in \mathbb{R}^d : \|x\| = 1\}$. $O()$ hides logarithmic factors in $n, d$, and the failure probability $\zeta$.

## 2 Problem formulation and background on DP

Typical PCA assumes i.i.d. data $\{x_i \in \mathbb{R}^d\}$ from a distribution and finds the first eigenvector of $\Sigma = \mathbb{E}[(x_i - \mathbb{E}[x_i])(x_i - \mathbb{E}[x_i])^\top] \in \mathbb{R}^{d \times d}$. Our approach allows for a more general class of data $\{A_i \in \mathbb{R}^{d \times d}\}$ that recovers the standard case when $A_i = (x_i - \mathbb{E}[x_i])^\top(x_i - \mathbb{E}[x_i])$.

**Assumption 1** $(\Sigma, \{\lambda_i\}_{i=1}^d, M, V, \kappa, a, \gamma^2)$-model. Let $A_1, A_2, \ldots, A_n \in \mathbb{R}^{d \times d}$ be a sequence of (not necessarily symmetric) matrices sampled independently from the same distribution that satisfy the following with PSD matrices $\Sigma \in \mathbb{R}^{d \times d}$ and $H_n \in \mathbb{R}^{d \times d}$, and positive scalar parameters $M, V, \kappa, a, \gamma^2$:

**A.1.** Let $\Sigma := \mathbb{E}[A_1]$, for a symmetric positive semidefinite (PSD) matrix $\Sigma \in \mathbb{R}^{d \times d}$, $\lambda_i$ denote the $i$-th largest eigenvalue of $\Sigma$, and $\kappa := \lambda_1 / (\lambda_1 - \lambda_2)$,

**A.2.** $\|A_1 - \Sigma\|_2 \leq \lambda_1 M$ almost surely,

**A.3.** $\max \{ \|\mathbb{E}[(A_1 - \Sigma)(A_1 - \Sigma)^\top]\|_2, \|\mathbb{E}[(A_1 - \Sigma)^\top(A_1 - \Sigma)]\|_2 \} \leq \lambda_2^2 V.$
The first three assumptions are required for PCA even if privacy is not needed. The last assumption provides a Gaussian-like tail bound that determines how much noise we need to introduce in the algorithm for $(\varepsilon, \delta)$-DP. The following lemma is useful in the analyses.

**Lemma 2.1.** Under Assumptions [A.1] and [A.4] in Assumption [7] for any unit vector $u$, $v$, with probability $1 - \xi$,

$$\|u^\top (A_1 - \Sigma) v\|^2 \leq K^2 \lambda_1^2 \|H_u\|_2 \log(2/\xi).$$

(1)

### 2.1 Oja’s algorithm

In a non-private setting, the following streaming algorithm introduced in [69] achieves optimal sample complexity as analyzed in [45]. It is a projected stochastic gradient ascent on the objective defined on the empirical covariance: $\max_{\|w\|=1} (1/n) \sum_{t=1}^n w^\top A_tw$.

**Algorithm 1:** (Non-private) Oja’s Algorithm

1. Choose $w_0$ uniformly at random from the unit sphere
2. for $t = 1, 2, \ldots, T$ do
   $w_t' \leftarrow w_{t-1} + \eta_t A_tw_{t-1}$, $w_t \leftarrow w_t'/\|w_t'\|$
3. Return $w_T$

Central to our analysis is the following error bound on Oja’s Algorithm from [45].

**Theorem 2.2** ([45] Theorem 4.1]). Under Assumptions [A.1]/[A.3] suppose the step size $\eta_t = \frac{\alpha}{(\lambda_1 - \lambda_2)(\xi + t)}$ for some $\alpha > 1/2$ and $\xi := \max(\kappa, H_u, \kappa^2(V + 1)\alpha^2/\log(1 + (\xi/100)))$. If $T > \xi$ then there exists a constant $C > 0$ such that Algorithm [7] outputs $w_T$ achieving w.p. $1 - \xi$,

$$\sin^2(w_T, v_1) \leq C \log(1/\xi) \left( \frac{\alpha^2 \kappa^2 V}{(2\alpha - 1)T} + d \left( \frac{\xi}{T} \right)^{2\alpha} \right).$$

(2)

### 2.2 Background on Differential Privacy

Differential privacy (DP), introduced in [21], is a de facto mathematical measure for privacy leakage of a database accessed via queries. It ensures that even an adversary who knows all other entries cannot identify with a high confidence whether a person of interest participated in a database or not.

**Definition 2.3** (Differential privacy [21]). Given two multisets $S$ and $S'$, we say the pair $(S, S')$ is neighboring if $|S \setminus S'| + |S' \setminus S| \leq 1$. We say a stochastic query $q$ over a dataset $S$ satisfies $(\varepsilon, \delta)$-differential privacy for some $\varepsilon > 0$ and $\delta \in (0, 1)$ if $\Pr(q(S) \in A) \leq e^{\varepsilon}\Pr(q(S') \in A) + \delta$ for all neighboring $(S, S')$ and all subset $A$ of the range of $q$.

Small values of $\varepsilon$ and $\delta$ ensures that the adversary cannot identify any single data point with high confidence, thus providing plausible deniability. We provide useful DP lemmas in Appendix B. Within our stochastic gradient descent approach to PCA, we rely on the Gaussian mechanism to privatize each update. The sensitivity of a query $q$ is defined as $\Delta_q := \sup_{\text{neighboring } (S, S')} \|q(S) - q(S')\|$.

**Lemma 2.4** (Gaussian mechanism [22]). For a query $q$ with sensitivity $\Delta_q$, $\varepsilon \in (0, 1)$, and $\delta \in (0, 1)$, the Gaussian mechanism outputs $q(S) = N(0, \Delta_q^2 \sqrt{2\log(1.25/\delta)/\varepsilon^2} \mathbf{I}_d)$ and achieves $(\varepsilon, \delta)$-DP.

This is a special case of a family of output perturbation mechanisms which includes the Laplace mechanism [21] and stair-case mechanisms [34]. The latter is shown to be optimal in one-dimension [35] and for hypothesis testing under local DP [48]. Another mechanism we frequently use is the private histogram learner of [56], whose analysis is provide in Appendix B along with various composition theorems to provide end-to-end guarantees.

### 2.3 Comparisons with existing results in private PCA

We briefly discuss the most closely related work and provide more previous work in Appendix A. Most existing results assume a fixed data under a deterministic setting where each sample has a bounded
Algorithm 2: Private Oja’s Algorithm

\begin{algorithm}
\caption{Private Oja’s Algorithm}
\end{algorithm}

\begin{algorithmic}[1]
\State \textbf{Input:} $S = \{A_t \in \mathbb{R}^{d \times d}\}_{t=1}^n$, privacy $(\varepsilon, \delta)$, learning rates $\{\eta_t\}_{t=1}^n$
\State Randomly permute $S$ and choose $w_0$ uniformly at random from the unit sphere
\State Set DP noise multiplier: $\alpha \leftarrow \mathcal{N}(0, I_d)$
\State Set clipping threshold: $\beta \leftarrow C\lambda_1 \sqrt{d}(K\gamma \log^a(nd/\zeta) + 1)$
\For{$t = 1, 2, \ldots, n$}
\State Sample $z_t \sim \mathcal{N}(0, I_d)$
\State $w'_t \leftarrow w_{t-1} + \eta_t \text{clip}_\beta (A_t w_{t-1}) + 2\eta_t \alpha z_t$ where $\text{clip}_\beta (x) = x \cdot \min\{1, \beta \|x\|_2\}$
\State $w_t \leftarrow w'_t / \|w'_t\|$
\EndFor
\State Return $w_n$
\end{algorithmic}

norm, $\|x_i\| \leq \beta$, and the goal is to find the top eigenvector of $\hat{\Sigma} := (1/n) \sum_{i=1}^n (x_i - \hat{\mu})(x_i - \hat{\mu})^T$ for the empirical mean $\hat{\mu}$. For the purpose of comparisons, consider Gaussian $x_i \sim \mathcal{N}(0, \Sigma)$ with $\|x_i\| \leq \beta = O(\sqrt{\lambda_1 d \log(n/\zeta)})$ for all $i \in [n]$ with probability $1 - \zeta$. The first line of approaches in \cite{9,14,23} is a Gaussian mechanism that outputs $\text{PCA}(\hat{\Sigma} + Z)$, where $Z$ is a symmetric matrix with i.i.d. Gaussian entries with a variance $((\beta^2/n\varepsilon)\sqrt{2\log(1.25/\delta)})^2$ to ensure $(\varepsilon, \delta)$-DP. The tightest result in \cite{23} Theorem 7 achieves

$$\sin(\hat{v}, v_1) = \hat{O}\left(\frac{d}{\sqrt{n}} + \frac{d^{3/2} \sqrt{\log(1/\delta)}}{\varepsilon n}\right),$$

(3)
with high probability, under a strong assumption that the spectral gap is very large: $\lambda_1 - \lambda_2 = \omega(d^{3/2} \sqrt{\log(1/\delta)}/(\varepsilon n))$. In a typical scenario with $\lambda_1 = O(1)$, this requires a large sample size of $n = \omega(d^{3/2}/\varepsilon)$. Since this Gaussian mechanism does not exploit the statistical properties of i.i.d. samples, the second term in this upper bound is larger by a factor of $d^{1/2}$ compared to the proposed DP-PCA (Corollary 5.2). The error rate of Eq. (3) is also achieved in \cite{38,36} by adding Gaussian noise to the standard power method for computing the principal components. When the spectral gap, $\lambda_1 - \lambda_2$, is smaller, it is possible to trade-off the dependence in $\kappa$ and the sampling ratio $d/n$, which we do not address in this work but is surveyed in Appendix A.

3 First attempt: making Oja’s Algorithm private

Following the standard recipe in training with DP-SGD, e.g., \cite{1} and a recent work \cite{75}, we introduce Private Oja’s Algorithm in Algorithm 2. At each gradient update, we first apply gradient norm clipping to limit the contribution of a single data point and next add an appropriately chosen Gaussian noise from Lemma 2.4 to achieve $(\varepsilon, \delta)$-DP, end-to-end. The choice of clipping threshold $\beta$ ensures that, with high probability under our assumption, we do not clip any gradients. The choice of noise multiplier $\alpha$ ensures $(\varepsilon, \delta)$-DP.

One caveat in streaming algorithms is that we access data $n$ times, each with a private mechanism, but accessing only a single data point at a time. To prevent excessive privacy loss due to such a large number of data accesses, we apply a random shuffling in line 1 Algorithm 2 in order to benefit from a standard amplification by shuffling \cite{24,29}. This gives an amplified privacy guarantee that allows us to add a small noise proportional to $\alpha = O(\log(n/\delta)/(\varepsilon \sqrt{n}))$. Without the shuffle amplification, we will instead need a larger noise scaling as $\alpha = O(\log(n/\delta)/\varepsilon)$, resulting in a suboptimal utility guarantee. However, this comes with a restriction that the amplification holds only for small values of $\varepsilon = O(\sqrt{\log(n/\delta)/n})$. Our first contribution in the proposed DP-PCA Algorithm 3 is to expand this range to $\varepsilon = O(1)$, which includes the practical regime of interest $\varepsilon \in [1/2, 5]$.

Lemma 3.1 (Privacy). If $\varepsilon = O(\sqrt{\log(n/\delta)/n})$ and the noise multiplier is chosen to be $\alpha = \Omega(\log(n/\delta)/(\varepsilon \sqrt{n}))$, then Algorithm 2 is $(\varepsilon, \delta)$-DP.

Under Assumption 1, we select gradient norm clipping threshold $\beta$ such that no gradient exceeds $\beta$.

Lemma 3.2 (Gradient clipping). Let $\beta = C\lambda_1 \sqrt{d}(K\gamma \log^a(nd/\zeta) + 1)$ for some constant $C > 0$. Then with probability $1 - \zeta$, $\|A_t w_{t-1}\| \leq \beta$ for any fixed $w_{t-1}$ independent of $A_t$, for all $t \in [n]$.

We provide proofs of both lemmas and the next theorem in Appendix D. When no clipping is applied, we can use the standard analysis of Oja’s Algorithm from \cite{45} to prove the following utility guarantee.
This construction decomposes the signal and the noise. For $A$ we explain the two remaining challenges in Private Oja’s Algorithm and propose techniques to achieve with probability 0.99,

$$
\sin^2(w_n, v_1) = \tilde{O}(\kappa^2 \left( \frac{V}{n} + \frac{(\gamma + 1) \log(1/\delta)}{\varepsilon^2 n^2} \right)),
$$

where $\tilde{O}()$ hides poly-logarithmic factors in $n, d, 1/\varepsilon$, and $\log(1/\delta)$ and polynomial factors in $K$.

The first term in Eq. (5) matches the non-private error rate for Oja’s algorithm in Eq. (2) with $\alpha = O(\log n)$ and $T = n$, and the second term is the price we pay for ensuring $(\varepsilon, \delta)$-DP.

**Remark 3.4.** For a canonical setting of a Gaussian data with $A_i = x_i v_i^\top$ and $x_i \sim \mathcal{N}(0, \Sigma)$, we have, for example from [70, Lemma 1.12], that $M = O(d \log(n))$, $V = O(d)$, $K = 4$, $a = 1$, and $\gamma^2 = O(1)$. Theorem 3.3 implies the following error rate:

$$
\sin^2(w_n, v_1) = \tilde{O}(\kappa^2 \left( \frac{d}{n} + \frac{d^2 \log(1/\delta)}{\varepsilon^2 n^2} \right)).
$$

Comparing to a lower bound in Theorem 3.3, this is already near optimal. However, for general distributions satisfying Assumption 4, Algorithm 2 (in particular the second term in Eq. (5)) can be significantly sub-optimal. We explain this second weakness of Private Oja’s Algorithm in the following section (the first weakness is the restriction on $\varepsilon = O(\sqrt{\log(n/\delta)/n})$).

## 4 Two remaining challenges

We explain the two remaining challenges in Private Oja’s Algorithm and propose techniques to overcome these challenges that lead to the design of DP-PCA (Algorithm 3).

**First challenge:** restricted range of $\varepsilon = O(\sqrt{\log(n/\delta)/n})$. This is due to the large number, $n$, of iterations that necessitates the use of the amplification by shuffling in Theorem 4.1. We reduce the number of iterations with minibatch SGD. For $T = O(\log^2 n)$ and $t = 1, 2, \ldots, T$, we repeat

$$
w_t' \leftarrow w_{t-1} + \frac{\eta}{B} \sum_{i=1}^{B(t-1)} \text{clip}_p(A_i w_{t-1}) + \frac{\eta \beta \alpha}{B} z_t, \quad \text{and} \quad w_t \leftarrow w_t'/||w_t'||,
$$

where $z_t \sim \mathcal{N}(0, I_d)$ and the minibatch size is $B = \lceil n/T \rceil$. Since the dataset is accessed only $T = O(\log^2 n)$ times, the end-to-end privacy is analyzed with the serial composition (Lemma B.3) instead of the amplification by shuffling. This ensures $(\varepsilon, \delta)$-DP for any $\varepsilon = O(1)$, resolving the first challenge, and still achieves the utility guarantee of Eq. (5).

**Second challenge:** excessive noise for privacy. This is best explained with an example.

**Example 4.1 (Signal and noise separation).** Consider a setting with $A_i = x_i v_i^\top$ and $x_i = s_i + n_i$ where $s_i = v$ with probability half and $s_i = -v$ otherwise for a unit norm vector $v$ and $n_i \sim \mathcal{N}(0, \sigma^2 I)$. We want to find the principal component of $\Sigma = \mathbb{E}[x_i x_i^\top] = vv^\top + \sigma^2 I$, which is $v$.

This construction decomposes the signal and the noise. For $A_i = vv^\top + s_i n_i^\top + n_i s_i^\top + n_i n_i^\top$, the signal component is determined by $vv^\top$ that is deterministic due to the sign cancelling. The noise component is $s_i n_i^\top + n_i s_i^\top + n_i n_i^\top$, which is random. We can control the Signal-to-Noise Ratio (SNR), $1/\sigma^2$, by changing $\sigma^2$, and we are particularly interested in the regime where $\sigma^2$ is small. As we are interested in $\sigma^2 \approx 1$, this satisfies Assumption 4 with $\lambda_1 = 1 + \sigma^2$, $\lambda_2 = \sigma^2$, $V = O(da^2)$, $K = O(1)$, $a = 1$, and $\gamma^2 = \sigma^2$. Substituting this into Eq. (5), Private Oja’s Algorithm achieves

$$
\sin^2(w_n, v_1) = \tilde{O}(\frac{\sigma^2 d}{n} + \frac{d^2 \log(1/\delta)}{\varepsilon^2 n^2}),
$$

where we are interested in $\sigma^2 < 1$. 


We show an upper bound on the error achieved by DP-PCA under an appropriate choice of the learning rate $\eta$ with a large enough constant and a choice of learning rate $\eta_t$ that depends on $(t, M, V, K, \kappa, a, \gamma^2)$. We provide a complete proof in Appendix E.1 that includes the explicit choice of the learning rate $\eta_t$ in Eq. (67), and a proof sketch is provided in Section 6.1.

**Theorem 5.1.** For $\varepsilon \in (0, 0.9)$, DP-PCA guarantees $(\varepsilon, \delta)$-DP for all $S$, $B$, $\zeta$, and $\delta$. Given $n$ i.i.d. samples $\{A_i \in \mathbb{R}^{d \times d}\}_{i=1}^n$ satisfying Assumption 4 with parameters $(\Sigma, M, V, K, \kappa, a, \gamma^2)$, if

$$n = \tilde{O}\left(\varepsilon\kappa^2 + \frac{d^{1/2}(\log(1/\delta))^{3/2}}{\varepsilon} + \kappa M + \kappa^2 V + \frac{d\kappa \gamma (\log(1/\delta))^{1/2}}{\varepsilon} + \frac{d \log(1/\delta)}{\varepsilon}\right),$$

with a large enough constant and $\delta \leq 1/n$, then there exists a positive universal constant $c_1$ and a choice of learning rate $\eta_t$ that depends on $(t, M, V, K, a, \lambda_1, \lambda_1 - \lambda_2, n, d, \varepsilon, \delta)$ such that $T = \lfloor n/B \rfloor$ steps of DP-PCA in Algorithm 3 with choices of $\varepsilon = 0.01$ and $B = c_1 n/(\log n)^2$.
outputs $w_T$ such that with probability $0.99$,
\[
\sin (w_T, v_1) = \tilde{O} \left( \kappa \left( \sqrt{\frac{d}{n}} + \frac{\gamma d \sqrt{\log(1/\delta)}}{\varepsilon n} \right) \right),
\]
where $\tilde{O}(\cdot)$ hides poly-logarithmic factors in $n$, $d$, $1/\varepsilon$, and $\log(1/\delta)$ and polynomial factors in $K$.

We further interpret this analysis and show that (i) DP-PCA is nearly optimal when the data is from a Gaussian distribution by comparing against a lower bound (Theorem 5.3); and (ii) DP-PCA significantly improves upon the private Oja’s algorithm under Example 4.1. We discuss the necessity of some of the assumptions at the end of this section, including how to agnostically find the appropriate learning rate scheduling.

**Near-optimality of DP-PCA under Gaussian distributions.** Consider the case of i.i.d. samples $\{x_i\}_{i=1}^n$ from a Gaussian distribution from Remark 3.4.

**Corollary 5.2 (Upper bound; Gaussian distribution).** Under the hypotheses of Theorem 5.1 and $\{A_i = x_i x_i^\top\}_{i=1}^n$ with Gaussian random vectors $x_i$’s, after $T = n/B$ steps, DP-PCA outputs $w_T$ that achieves, with probability $0.99$,
\[
\sin(w_T, v_1) = \tilde{O} \left( \kappa \left( \sqrt{\frac{d}{n}} + \frac{d \sqrt{\log(1/\delta)}}{\varepsilon n} \right) \right).
\]

We prove a nearly matching lower bound, up to factors of $\sqrt{\lambda_1/\lambda_2}$ and $\sqrt{\log(1/\delta)}$. One caveat is that the lower bound assumes pure-DP with $\delta = 0$. We do not yet have a lower bound technique for approximate DP that is tight, and all known approximate DP lower bounds have gaps to achievable upper bounds in its dependence in $\log(1/\delta)$, e.g., [5, 62]. We provide a proof in Appendix C.1.

**Theorem 5.3 (Lower bound; Gaussian distribution).** Let $\mathcal{M}_\varepsilon$ be a class of $(\varepsilon, 0)$-DP estimators that map $n$ i.i.d. samples to an estimate $\hat{v} \in \mathbb{R}^d$. A set of Gaussian distributions with $(\lambda_1, \lambda_2)$, as the first and second eigenvalues of the covariance matrix is denoted by $\mathcal{P}(\lambda_1, \lambda_2)$. For $d > c$ where $c > 0$ is some absolute constant, there exists a universal constant $C > 0$ such that
\[
\inf_{v \in \mathcal{M}_\varepsilon} \sup_{p \in \mathcal{P}(\lambda_1, \lambda_2)} \mathbb{E}_{S \sim P_n} [\sin(\hat{v}(S), v_1)] \geq C \min \left( \kappa \left( \sqrt{\frac{d}{n}} + \frac{d}{\varepsilon n} \right) \sqrt{\frac{\lambda_2}{\lambda_1}} \right) .
\]

**Comparisons with private Oja’s algorithm.** We demonstrate that DP-PCA can significantly improve upon Private Oja’s Algorithm with Example 4.1 where DP-PCA achieves an error bound of $\sin(w_T, v_1) = \tilde{O}(\sigma \sqrt{d/n} + \sigma d \sqrt{\log(1/\delta)}/(\varepsilon n))$. As the noise power $\sigma^2$ decreases DP-PCA achieves a vanishing error, whereas Private Oja’s Algorithm has a non-vanishing error in Eq. (8). This follows from the fact that the second term in the error bound in Eq. (10) scales as $\gamma$, which can be made arbitrarily smaller than the second term in Eq. (5) that scales as $(\gamma + 1)$. Further, the error bound for DP-PCA holds for any $\varepsilon = O(1)$, whereas Private Oja’s Algorithm requires significantly smaller $\varepsilon = O(\sqrt{\log(n/\delta)/n})$.

**Remarks on the assumptions of Theorem 5.1.** We have an exponential dependence of the sample complexity in the spectral gap, $n \geq \exp(\kappa^2)$. This ensures we have a large enough $T = \lfloor n/B \rfloor$ to reduce the non-dominant second term in Eq. (2), in balancing the learning rate $\eta_t$ and $T$ (which is explicitly shown in Eqs. (69) and (70) in the Appendix). It is possible to get rid of this exponential dependence at the cost of an extra term of $\tilde{O}(\kappa^4 \gamma^2 d^2 \log(1/\delta)/(\varepsilon n)^2)$ in the error rate in Eq. (10), by selecting a slightly larger $T = \kappa^2 \log^2 n$. A Gaussian-like tail bound in Assumption A.4 is necessary to get the desired upper bound scaling as $\tilde{O}(d \sqrt{\log(1/\delta)}/(\varepsilon n))$ in Eq. (11) for example. The next lower bound shows that without such assumptions on the tail, the error due to privacy scales as $\Omega(\sqrt{d \log(1/\delta)}/(\varepsilon n))$. We believe that the dependence in $\delta$ is loose, and it might be possible to get a tighter lower bound using [52]. We provide a proof and other lower bounds in Appendix C.

**Theorem 5.4 (Lower bound without Assumption A.4).** Let $\mathcal{M}_\varepsilon$ be a class of $(\varepsilon, \delta)$-DP estimators that map $n$ i.i.d. samples to an estimate $\hat{v} \in \mathbb{R}^d$. A set of distributions satisfying Assumptions A.7-A.3...
with \( M = \tilde{O}(d + \sqrt{n\epsilon/d}) \), \( V = O(d) \) and \( \gamma = O(1) \) is denoted by \( \tilde{P} \). For \( d \geq 2 \), there exists a universal constant \( C > 0 \) such that

\[
\inf_{\hat{\nu} \in \mathcal{M}_s} \sup_{p \in \mathbb{P}} \mathbb{E}_{S \sim P^n} \left[ \sin(\hat{\nu}(S), v_1) \right] \geq C \kappa \min \left( \sqrt{\frac{d \log \left( (1 - e^{-\epsilon}) / \delta \right)}{\epsilon n}}, 1 \right). \tag{13}
\]

Currently, DP-PCA requires choices of the learning rates, \( \eta_t \), that depend on possibly unknown quantities. Since we can privately evaluate the quality of our solution, one can instead run multiple instances of DP-PCA with varying \( \eta_t = c_1/(c_2 + t) \) and find the best choice of \( c_1 > 0 \) and \( c_2 > 0 \). Let \( w_T(c_1, c_2) \) denote the resulting solution for one instance of \( \{\eta_t = c_1/(c_2 + t)\}^T_{t=1} \). We first set a target error \( \zeta \). For each round \( i = 1, \ldots, K \), we will run algorithm for \( (c_1, c_2) = [2^{-i-1}, 2^{-i+1}] \times [2^{-i+2}, 2^{-i-1}] \) and \( (c_1, c_2) = [2^{i-1}, 2^{-i+1}] \times [2^{-i-1}, 2^{i-1}] \), and compute each \( \sin(w_T(c_1, c_2), v_1) \) privately, each with privacy budget \( \epsilon_i = \frac{\delta}{2^{i+1}(2i-1)}, \delta_i = \frac{\delta}{2^{i+1}(2i+1)} \). We terminate the algorithm once there there is a \( w_T(c_1, c_2) \) satisfies \( \sin(w_T(c_1, c_2), v_1) \) \( \leq \zeta \). It is clear that this search meta-algorithm terminate in logarithmic round, and the total sample complexity only blows up by a poly-log factor.

6 Private mean estimation for the minibatch stochastic gradients

DP-PCA critically relies on private mean estimation to reduce variance of the noise required to achieve \((\epsilon, \delta)\)-DP. We follow a common recipe from [56, 50, 54, 8, 15]. First, we privately find an approximate range of the gradients in the minibatch (Alg. 4). Next, we apply the Gaussian mechanism designed for mean estimation with unknown covariance from [53], we propose to use a private preconditioning of [53] that estimates all eigenvalues and requires \( O(d/\epsilon^2 \log(1/\delta)) \) samples, we only require the top eigenvalue and hence the next theorem shows that we only need \( n = O(d \log(1/\delta)/\epsilon) \) samples.

**Theorem 6.1.** Algorithm 4 is \((\epsilon, \delta)\)-DP. Let \( g_t = A_t u \) for some fixed vector \( u \), where \( A_t \) satisfies A.4 and A.4 in Assumption 2 such that the mean is \( \mathbb{E}[g_t] = \Sigma u \) and the covariance is \( \mathbb{E}[(g_t - \Sigma u)^2] = \lambda^2_H u \). With a large enough sample size scaling as

\[
B = O \left( \frac{K^2 d \log(d \log(1/(\delta \zeta)))/(\zeta \epsilon)}{\epsilon} \right)^{1/2},
\]

Algorithm 4 outputs \( \hat{\lambda} \) achieving \( \hat{\lambda} \in [1/(\sqrt{2})\lambda^2_H u, 1/(\sqrt{2})\lambda^2_H u] \) with probability \( 1 - \zeta \), where the pair \( (K > 0, a > 0) \) parametrizes the tail of the distribution in A.4 and \( O(\cdot) \) hides logarithmic factors in \( B, d, 1/\zeta, \log(1/\delta), \) and \( \epsilon \).

We provide a proof in Appendix E.2. There are other ways to privately estimate the range. Some approaches require known bounds such as \( \sigma_{min}^2 \leq \lambda^2_H u \leq \sigma_{max}^2 \) for all \( t \in [d] \) [56], and other agnostic approaches are more involved such as instance optimal universal estimators of [17].

**Step 2: Gaussian mechanism for mean estimation.** Once we have a good estimate of the top eigenvalue from the previous section, we use it to select the bin size of the private histogram and compute the truncated empirical mean. Since truncated empirical mean has a bounded sensitivity, we can use Gaussian mechanism to achieve DP. The algorithm is now standard in DP mean estimation, e.g. [56, 50]. However, the analysis is slightly different since our assumptions on \( g_t \)‘s are different. For completeness, we provide the Algorithm 5 in Appendix E.3.

The next lemma shows that the Private Mean Estimation is \((\epsilon, \delta)\)-DP, and with high probability clipping does not apply to any of the gradients. The returned private mean, therefore, is distributed as a spherical Gaussian centered at the empirical mean of the gradients. This result requires that we have a good estimate of the top eigenvalue from Alg. 4 such that \( \hat{\lambda} \simeq \lambda^2_H u \). This analysis implies
that we get an unbiased estimate of the gradient mean (which is critical in the analysis) with noise scaling as $O(\lambda_1 \gamma \sqrt{d \log(1/\delta) / (\varepsilon B)})$, where $\gamma^2 = \max_{u: \|u\|=1} \| H_u \|^2$ (which is critical in getting the tight sample complexity in the second term of the final utility guarantee in Eq. (10)). We provide a proof in Appendix E.3.

**Lemma 6.2.** For $\varepsilon \in (0, 0.9)$ and any $\delta \in (0, 1)$, Algorithm 5 is $(\varepsilon, \delta)$-DP. Let $g_t = A_t u$ for some fixed vector $u$, where $A_t$ satisfies Assumption A.4 in Assumption 4 such that the mean is $E[g_t] = \Sigma u$ and the covariance is $E[(g_t - \Sigma u)(g_t - \Sigma u)^\top] = \lambda_1^2 H_u$. If $\Lambda \in [\lambda_1^2\|H_u\|^2 / \sqrt{2}, \sqrt{2}\lambda_1^2\|H_u\|^2]$, $\delta \leq 1/B$, and $B = \Omega((\sqrt{d \log(1/\delta) / \varepsilon}) \log(d / (\zeta \delta)))$ then, with probability $1 - \zeta$, $g_t \in \tilde{g} + \left[ -3K \sqrt{\lambda} \log^a(Bd / \zeta), 3K \sqrt{\lambda} \log^a(Bd / \zeta) \right]$ for all $i \in [B]$.

**6.1 Proof sketch of Theorem 5.1**

We choose $B = O(n / \log^2 n)$ such that we access the dataset only $T = O(\log^2 n)$ times. Hence we do not need to rely on amplification by shuffling. To add Gaussian noise that scales as the standard deviation of the gradients in each minibatch (as opposed to potentially excessively large mean of the gradients), DP-PCA adopts techniques from recent advances in private mean estimation. Namely, we first get a private and accurate estimate of the range from Theorem 6.1. Using this estimate, $\hat{\Lambda}$, Private Mean Estimation returns an unbiased estimate of the empirical mean of the gradients, as long as no truncation has been applied as ensured by Lemma 6.2. This gives

$$w_t' \leftarrow w_{t-1} + \eta_t \left( \frac{1}{B} \sum_{i=1}^{B} A_{B(t-1)+i} w_{t-1} + \beta_t z_t \right),$$

for $z_t \sim \mathcal{N}(0, I)$ and $\beta_t = \frac{8K \sqrt{2\lambda_1 \log^a(Bd / \zeta) / \sqrt{2d \log(2.5 / \delta)}}}{B}$. Using rotation invariance of spherical Gaussian random vectors and the fact that $\|w_{t-1}\| = 1$, we can reformulate it as

$$w_t' \leftarrow w_{t-1} + \eta_t \left( \frac{1}{B} \sum_{i=1}^{B} A_{B(t-1)+i} \beta_t G_t \right) w_{t-1}.$$

This process can be analyzed with Theorem 2.2 with $\hat{A}_t$ substituting $A_t$.

**7 Discussion**

Under the canonical task of computing the principal component from i.i.d. samples, we show the first result achieving an optimal error rate. This critically relies on two ideas: minbatch SGD and private mean estimation. In particular, private mean estimation plays a critical role in the case when the range of the gradients is significantly smaller than the norm; we achieve an optimal error rate that cannot be achieved with the standard recipe of gradient clipping.

Assumption A.4 can be relaxed to heavy-tail bounds with bounded $k$-th moment on $A_t$, in which case we expect the second term in Eq. (10) to scale as $O(d(\sqrt{\log(1/\delta)} / \varepsilon n)^{1-1/k})$, drawing analogy from a similar trend in a computationally inefficient DP-PCA without spectral gap [62 Corollary 6.10]. When a fraction of data is corrupted, recent advances in [34, 58, 45] provide optimal algorithms for PCA. However, existing approach of [62] for robust and private PCA is computationally intractable. Borrowing ideas from robust and private mean estimation in [61], one can design an efficient algorithm, but at the cost of sub-optimal sample complexity. It is an interesting direction to design an optimal and robust version of DP-PCA. Our lower bounds are loose in its dependence in $\log(1/\delta)$. Recently, a promising lower bound technique has been introduced in [22] that might close this gap.

There are two ways to extend our framework to general rank-$r$ PCA, whose analyses are promising research directions. First, applying Hotelling’s deflation method [40], we can iteratively find the PCA components one by one, by alternating our DP-PCA and deflation. For example, in one step of the iteration, we only update the current iterate vector in the directions orthogonal to all the previously found PCA components. Repeating this steps gives the estimates of the top principal components. Secondly, we can directly apply Oja’s algorithm. We keep track of a $r$-dimensional subspace in the Oja’s update rule for PCA, and perform QR decomposition to keep the iterates on the Grassmannian manifold. It might be possible to extend the analysis of [42] to analyze the private version.
Acknowledgments and Disclosure of Funding

This work is supported in part by Google faculty research award and NSF grants CCF-1705007, CNS-2002664, IIS-1929955, DMS-2134012, CCF-2019844 as a part of NSF Institute for Foundations of Machine Learning (IFML), CNS-2112471 as a part of NSF AI Institute for Future Edge Networks and Distributed Intelligence (AI-EDGE).

References


Checklist

1. For all authors...
   (a) Do the main claims made in the abstract and introduction accurately reflect the paper’s contributions and scope? [Yes]
   (b) Did you describe the limitations of your work? [Yes], see Section 7
   (c) Did you discuss any potential negative societal impacts of your work? [N/A] Our work is theoretical and does not have a direct negative societal impact.
   (d) Have you read the ethics review guidelines and ensured that your paper conforms to them? [Yes]

2. If you are including theoretical results...
   (a) Did you state the full set of assumptions of all theoretical results? [Yes]
   (b) Did you include complete proofs of all theoretical results? [Yes] See Appendix.

3. If you ran experiments...
   (a) Did you include the code, data, and instructions needed to reproduce the main experimental results (either in the supplemental material or as a URL)? [N/A] We don’t have experiments
   (b) Did you specify all the training details (e.g., data splits, hyperparameters, how they were chosen)? [N/A]
   (c) Did you report error bars (e.g., with respect to the random seed after running experiments multiple times)? [N/A]
   (d) Did you include the total amount of compute and the type of resources used (e.g., type of GPUs, internal cluster, or cloud provider)? [N/A]

4. If you are using existing assets (e.g., code, data, models) or curating/releasing new assets...
   (a) If your work uses existing assets, did you cite the creators? [N/A] We didn’t use existing assets.
   (b) Did you mention the license of the assets? [N/A]
   (c) Did you include any new assets either in the supplemental material or as a URL? [N/A]
   (d) Did you discuss whether and how consent was obtained from people whose data you’re using/curating? [N/A]
   (e) Did you discuss whether the data you are using/curating contains personally identifiable information or offensive content? [N/A]

5. If you used crowdsourcing or conducted research with human subjects...
   (a) Did you include the full text of instructions given to participants and screenshots, if applicable? [N/A]
   (b) Did you describe any potential participant risks, with links to Institutional Review Board (IRB) approvals, if applicable? [N/A]
   (c) Did you include the estimated hourly wage paid to participants and the total amount spent on participant compensation? [N/A]