Analyze Gauss: Optimal Bounds for Privacy-Preserving Principal Component Analysis

Cynthia Dwork
Microsoft Research

Kunal Talwar
Microsoft Research

Abhradeep Thakurta*
Stanford University
Microsoft Research

Li Zhang
Microsoft Research

Abstract

We consider the problem of privately releasing a low dimensional approximation to a set of data records, represented as a matrix A in which each row corresponds to an individual and each column to an attribute. Our goal is to compute a subspace that captures the covariance of A as much as possible, classically known as principal component analysis (PCA). We assume that each row of A has ℓ_2 norm bounded by one, and the privacy guarantee is defined with respect to addition or removal of any single row. We show that the well-known, but misnamed, randomized response algorithm, with properly tuned parameters, provides nearly optimal additive quality gap compared to the best possible singular subspace of A. We further show that when $A^T A$ has a large eigenvalue gap – a reason often cited for PCA – the quality improves significantly. Optimality (up to logarithmic factors) is proved using techniques inspired by the recent work of Bun, Ullman, and Vadhan on applying Tardos's fingerprinting codes to the construction of hard instances for private mechanisms for 1-way marginal queries. Along the way we define a *list culling game* which may be of independent interest.

By combining the randomized response mechanism with the well-known *following the perturbed leader* algorithm of Kalai and Vempala we obtain a private online algorithm with nearly optimal regret. The regret of our algorithm even outperforms all the previously known online *non-private* follow the perturbed leader type of algorithms. We achieve this better bound by, satisfyingly, borrowing insights and tools from differential privacy!

1 Introduction

In areas as diverse as machine learning, statistics, information retrieval, earth sciences, archaeology, and image processing, given a data set represented by a matrix $A \in \mathbb{R}^{m \times n}$, it is often desirable to find a good approximation to A that has low rank. Working with low-rank approximations improves space and time efficiency. Other benefits include removal of noise and extraction of correlations, useful, for example, in (approximate) matrix completion from a small set of observations – an impossible task if A is arbitrary but potentially feasible if A enjoys a good low rank approximation. The problem of low-rank approximation has also received substantial attention in the differential privacy literature [4, 15, 31, 26, 10, 21, 22]. If we think of the matrix $A \in \mathbb{R}^{m \times n}$ as containing information about n attributes of m individuals, the goal is to learn "about" A (we intentionally remain vague, for now) without compromising the privacy of any individual. That is, the literature focuses on being able to do, in a differentially private way, whatever is achieved by low-rank approximation in the non-private literature. Our work continues this line of research.

Existing differentially private algorithms can have errors with an unfortunate dependence on the ambient dimension n of the data. This bad dependence may sometimes be due to the suboptimality of our algorithms,

*Supported in part by the Sloan Foundation

sometimes due to the inherent difficulty of the problem. A driving motivation for our work is to extract better performance from these algorithms when the inherent dimensionality of the input is much lower than the ambient dimension. For example, the data may be generated according to a low dimensional model and the measurements may be noisy.

The standard method of the principal component analysis (PCA) for low rank approximation is to compute a best low-dimensional eigen-subspace B of the matrix $A^T A = \sum_{i=1}^{m} a_i^T a_i$ (recall that the a_i are row vectors). The underlying intuition is that the projection onto B preserves the important features of the data rows while projecting away the noise. We will focus on a private mechanism for computing B. By (1) privately finding a low-rank subspace B capturing most of the variance in A, and then (2) running the existing differentially private algorithm on the projection of A onto B, the hope is that poor dependence on the dimension in the second step is mitigated by the dimension reduction obtained in the first.

Because it was found in a privacy-preserving fashion, B can safely be made public. A key point is that the two-step procedure just described does not require publication of the projection. This, then, will be our approach: the *projector* (Π_B) will be public, the *projection* ($\Pi_B(A)$) will not be released. [1].

The literature sometimes focuses on the case of $m \gg n$, and at other times assumes $m \ll n$. In the first case, the rows of the data matrix are often assumed to be normalized to have norm at most 1, as is done here; when $m \ll n$ the row norms may be unbounded [21, 22]. The literature also varies in terms of granularity of the privacy guarantee, protecting, variously, the privacy of each row *in its entirety* [4, 15, 26, 10], which is what we do here, or individual entries [31, 22], or norm 1 changes to any row [21]. Finally, the literature varies on the nature of differential privacy offered: so-called *pure*, or $(\epsilon, 0)$-differential privacy [15, 26, 10] and *approximate*, or (ϵ, δ), differential privacy [4, 31, 21, 22], which is the notion used in our work.

Refined Randomization: Blum *et al.* were the first to suggest privately releasing $A^T A$ by adding independent noise to each of the n^2 entries of this matrix [4]. The data analyst is then free to compute best rank k approximations to the privacy preserving, noisy, $\widehat{A^T A}$ for any and all k. This naïve noising approach, which has somewhat erroneously become known as *randomized response*, was refined in [15] to add less noise; our main algorithmic result is a careful analysis of a version of this refinement. Specifically, we will use the Gaussian mechanism [13], which adds independently chosen Gaussian noise to each entry of $A^T A$. When there is a gap in the singular values of A, or even a gap between singular values whose indices are not adjacent (formally $\sigma_k^2 - \sigma_{k'}^2 \in \omega(\sqrt{n}/(k + k'))$), we see a clear improvement, in captured variance, over previously published results. In this case, the analysis further shows, the space spanned by the top k right singular vectors of the (refined) noisy version of $A^T A$ is very close to the space spanned by the top k right singular vectors of A, with the spectral norm of the difference in projectors being independent of k.

When there is no gap the algorithm performs no worse than the best in the literature; when $m \gg n$ we do expect such a gap: the more data, the better the algorithm's utility. The algorithm approaches the correct subspace of $A^T A$ at a rate faster than $1/m$, meaning that as we increase the number of samples the total error decreases.

Optimality: Our version of the refined noisy release of $A^T A$ is, up to logarithmic factors, optimal for approximate differential privacy. Pursuing a connection between differentially private algorithms and cryptographic traitor-tracing schemes [17], Bun, Ullman, and Vadhan [6] established lower bounds on errors for approximately differentially private release of a class of counting queries that are tight to within logarithmic factors. Their query class is based on a class of *fingerprinting codes* [5] due to Tardos [43]. We show that

[1] This was exploited by McSherry and Mironov in their work on differentially private recommendation systems [31]: in many non-private recommendation systems, recommendations made to individual i depend only on the item covariance information and the individual's own item ratings. In our terms, the recommendations to user i depend only on row i of the input matrix A and on $A^T A$. It makes no sense to hide the user's own ratings from himself, so it is sufficient that $A^T A$ be approximated in a privacy-protective fashion.

their result translates fairly easily to a lower bound for private approximation of the top singular vector. We also extend this to obtain lower bounds for rank k subspace estimation even for $k \in \Omega(n)$, a much more challenging task. Intuitively, for $k > 1$, we construct k "clusters" of fingerprinting codes. We have to overcome some difficulties to show that these clusters do not interfere much and to identify a "privacy-violating" vector hidden in a subspace. For the first we prove a stronger property of Tardos's codes, and for the second we introduce a game, called the list culling game, in which one player, using "planted questions", has to identify a good answer promised in a large set of answers provided by the other player. We propose a strategy for discovering the good answer with high success probability and apply it to constructing the privacy lower bound. Both results might be of independent interest.

Online Algorithms: Our third contribution merges two lines of research: differentially private regret minimization in online algorithms [16, 38] inspired by the Follow the Perturbed Leader (FPL) algorithm of Kalai and Vempala [25], and non-private online algorithms for principal components analysis [44]. A folk theorem says that differential privacy provides stability and hence reduces generalization error. We make this connection explicit in the online setting.

In the online model, computation proceeds in steps. At each time step t a rank k subspace V_t is output, a single data row A_t of A is received, and a reward is earned equal to $||A_t V_t||_2^2$. Regret is the difference between the sum of the earned rewards and the corresponding quantity for the best rank k matrix V chosen in hindsight (call it OPT). It is known, thanks to the pioneering work of [28], that the stability of an online algorithm is useful for achieving the low regret bound[2]. In [25], the FPL algorithm achieves stability by the addition of Laplace noise and is shown to have low regret. This technique has been successfully applied to several online algorithms. Indeed, for the online PCA problem, the previously best known FPL algorithm [44, 23] achieves a regret bound of $\widetilde{O}(\sqrt{kn\mathsf{OPT}})$. Our main observation is that a differentially private algorithm achieves similar stability to that of the FPL algorithm. With this insight, and borrowing tools from differential privacy, we show that, rather than adding Laplace noise, which might be unnecessarily large, one can instead add Gaussian noise, leading to an improved regret bound of only $\widetilde{O}(\sqrt{k\mathsf{OPT}}n^{1/4})$. In addition, by adding carefully correlated noise as in [16], we can make the entire algorithm private by incurring only a polylogarithmic factor in regret.

Granularity of Privacy: Two works of Hardt and Roth aim to exploit *low coherence* of the data matrix, a phenomenon of substantial interest in the (non-private) compressed sensing and matrix completion literature [7, 8, 36, 41, 34], to (privately) obtain good low rank approximations to the data matrix [21, 22]. There are several definitions of matrix coherence; roughly speaking coherence measures the extent to which the singular vectors are correlated with the standard basis. In the case of matrix completion, where the samples are intimately tied to the basis in which the data matrix is naturally represented, low coherence says that information is holographically embedded throughout the rows. The two definitions in [21] deal with row norms, either of the data matrix A or of U, when expressing $A = U\Sigma V^T$ in its singular value decomposition. There is an interplay between the granularity of the privacy guarantee and the specific coherence measure. The algorithms in [21], which are interesting when $n \geq m$, protect the rows in A up to any perturbation of Euclidean norm at most one. In this case the coherence conditions and the privacy granularity are rotationally invariant. In contrast, in [22] the coherence notion deals with the maximum entries of U and V, and the privacy granularity is for changes of magnitude at most one to a single entry of the data matrix. In this case neither the coherence condition nor the privacy granularity is rotationally invariant.

In our privacy definition, we protect the privacy against any individual row change. This is a natural choice for us as in many applications of PCA, each row corresponds to an individual. But for such a strong privacy notion (compared to single entry change or change of bounded norm), it is also more challenging to provide

[2]Roughly speaking, in this context stability means that the output of the online algorithm does not change significantly between adjacent steps.

good utility. Indeed, we cannot achieve meaningful utility if we allow arbitrary A, for instance if one row has arbitrarily large norm. But in practice, allowing such "overpowering" individuals often goes against the purpose of PCA for discovering the global structure of many data records, and row normalization is often recommended before applying PCA. For example, in face recognition each individual image (a row in A) is typically normalized to have unit variance [2, 45]. Motivated by such practical considerations, we assume each row to have at most unit ℓ_2 norm[3].

For detailed comparison to previous work, see Section 2.5.

2 Preliminaries

2.1 Notations and definitions

We treat vectors as column vectors (unless explicitly mentioned). For a given matrix $A \in \Re^{m \times n}$, we denote the i-th row of A by A_i, which in this case is a row vector. For a vector $x \in \Re^n$, $\|x\|$ denotes the ℓ_2 norm. For a matrix $A \in \Re^{m \times n}$, the spectral norm is defined as $\|A\|_2 = \max_{x \in \Re^n, \|x\|_2 = 1} \|Ax\|_2$; the Frobenius norm is defined as $\|A\|_F = \sqrt{\sum_{i \in [m], j \in [n]} a_{ij}^2}$, where a_{ij} are the entries of the matrix A. For a square matrix, the trace $\mathsf{tr}(\cdot)$ is defined as the sum of its diagonal elements. So $\|A\|_F^2 = \mathsf{tr}(A^T A) = \mathsf{tr}(AA^T)$. Slightly abusing terminology we will refer to $A^T A$ as the covariance matrix of A.

For a matrix A, the singular value decomposition of A is defined as $A = U \Sigma V^T$, where $U \in \Re^{m \times m}$ and $V \in \Re^{n \times n}$ are unitary matrices and called the *left* and *right* singular subspaces, respectively. The matrix $\Sigma \in \Re^{m \times n}$ is a diagonal matrix with non-negative entries $\sigma_1, \ldots, \sigma_{\min(m,n)}$ along the diagonal, called the singular values. In this paper, we assume they are ordered decreasingly, i.e $\sigma_1 \geq \sigma_2 \geq \ldots$. Suppose that $V = (v_1, \ldots, v_n)$. We define $V_k = (v_1, \ldots, v_k)$ and call it the principal (or top) k right singular subspace. It is well known that $\|A\|_2 = \sigma_1$, $\|A\|_F^2 = \sum_i \sigma_i^2$, and $\|AV_k\|_F^2 = \sum_{i=1}^k \sigma_i^2 = \max_{P \in \mathbb{P}_k} \|AP\|_F^2$.

Each row $a_i \in \Re^n$, $1 \leq i \leq m$, of the data matrix $A \in \Re^{m \times n}$ represents the attributes of a single user. As discussed above, we assume each row has at most unit ℓ_2 norm. The set of all such matrices is denoted $\mathcal{A}$.

Given the data matrix A, our objective is to output a subspace that preserves privacy and captures the variance of A as much as possible. To define privacy, we call two matrices $A, A' \in \mathcal{A}$ *neighbors* if they differ in exactly one row, as each row in A corresponds to an individual user. We will ensure (ϵ, δ)-differential privacy.

Definition 1 (Differential privacy [15, 13])**.** *A randomized mechanism $\mathcal{M}$ is (ϵ, δ)-differentially private if for every two neighboring matrices $A, A' \in \mathcal{A}$ and for all events $\mathcal{O} \subseteq Range(\mathcal{M})$, $\Pr[\mathcal{M}(A) \in \mathcal{O}] \leq e^\epsilon \Pr[\mathcal{M}(A') \in \mathcal{O}] + \delta$.*

Let $f : \mathcal{A} \to \Re^p$ be a vector-valued function operating on databases. The ℓ_2-*sensitivity* of f, denoted Δf, is the maximum over all pairs A, A' of neighboring datasets of $||f(A) - f(A')||_2$. The *Gaussian mechanism* adds independent noise drawn from a Gaussian with mean zero and standard deviation slightly greater than $(\Delta f) \ln(1/\delta)/\epsilon$ to each element of its output.

Theorem 2 (Gaussian Mechanism [13, 12])**.** *Let $f : \mathcal{A} \to \Re^p$ be a vector-valued function. Let $\tau = \Delta f \sqrt{2 \ln(1.25/\delta)}/\epsilon$. The Gaussian mechanism, which adds independently drawn random noise distributed as $\mathcal{N}(0, \tau^2)$ to each output of $f(A)$, ensures (ϵ, δ)-differential privacy.*

[3]To enforce this condition, an offending row can be divided by its own norm; this will not affect privacy.

We are interested in the function $f(A) = A^T A$, which may be viewed as an n^2-dimensional vector. Because we ensure that $||a_i||_2 \leq 1$, the sensitivity of f is at most one.

2.2 Background on matrix analysis and subspace estimation

In this paper we are primarily interested in privately recovering the top k right singular vectors of a given matrix $A \in \Re^{m \times n}$, which equivalently translates to recovering the top k eigenvectors in $A^T A$. More formally, any real matrix A can be decomposed into $U \Sigma V^T$, where U is an $m \times m$ unitary matrix, Σ is an $m \times n$ rectangular diagonal matrix with non-negative entries in the diagonals, and V is an $n \times n$ unitary matrix. The columns of U are called the left singular vectors of A, the diagonal entries of Σ are called the singular values of A, and the columns of V are called the right singular vectors of A. Using this definition of singular vectors, one can easily show that $A^T A = V \Lambda V^T$, where $\Lambda \in \Re^{n \times n}$ is a diagonal matrix with non-zero entries corresponding to the squares of the entries in Σ, also called the eigenvalues of $A^T A$. We denote V_k to be an $\Re^{n \times k}$ matrix, whose columns correspond to the top k right singular vectors of A. For various algebraic manipulations throughout this paper, we will use the following two useful inequalities. In the following useful result from matrix perturbation theory (see, for example [3]) we will denote $\lambda_1(X) \geq \lambda_2(X) \cdots$ as the eigenvalues of a symmetric matrix X.

Theorem 3 (Weyl's inequality)**.** *Let $X, Y \in \Re^{n \times n}$ be two real symmetric matrices. For each $i \in [n]$,*

$$\lambda_i(X) + \lambda_n(Y) \leq \lambda_i(X + Y) \leq \lambda_i(X) + \lambda_1(Y).$$

Along with recovering an approximation to V_k, we will be often interested in obtaining a rank k approximation to the matrix A or $A^T A$. Let A_k be the matrix formed by picking the top k right and left singular vectors of A and their corresponding singular values. It is well known that A_k is the matrix that minimizes $\|A - B\|_2$ and $\|A - B\|_F$ for any matrix B of rank at most k. This result is also called the *matrix approximation lemma* [18].

2.3 Basic tools in differential privacy

In this section we will discuss some the basic tools commonly used in the design of differentially private algorithms. Since we will use these tools as building blocks for designing various algorithms in this paper, we state them briefly here.

2.3.1 Noise mechanisms

Laplace mechanism [15]. Given a data set D from some arbitrary domain $\mathcal{D}$ and a function $f : \mathcal{D} \to \Re^n$, the objective is to design an algorithm $\mathcal{M}$ that outputs an approximation to $f(D)$ while preserving differential privacy. One quantity that becomes very useful in the design of $\mathcal{M}$ is the L_1-sensitivity of the function f which is defined as $\mathsf{Sensitivity}_1(f) = \max_{D \text{ and } D' \text{ being neighbors}} \|f(D) - f(D')\|_1$. Let $\mathsf{Lap}(\lambda)$ be the standard Laplace distribution with the scaling parameter λ, i.e., the density function is given by $\frac{1}{2\lambda} e^{-\frac{|x|}{\lambda}}$. [15] showed that adding independent Laplace noise sampled i.i.d. from $\mathsf{Lap}\left(\frac{\mathsf{Sensitivity}_1(f)}{\epsilon}\right)$ to $f(D)$ satisfies $(\epsilon, 0)$-differential privacy (or simply ϵ-differential privacy).

Gaussian mechanism [13]. Gaussian mechanism is very similar to Laplace mechanism, except the noise model is the Gaussian distribution. Let the L_2-sensitivity of the function f is defined as $\mathsf{Sensitivity}_2(f) =$

$\max_{D \text{ and } D' \text{ being neighbors}} \|f(D) - f(D')\|_2$. Let $\mathcal{N}(0, \sigma^2)$ be the Gaussian distribution with mean zero and standard deviation σ, i.e., the density function is given by $\frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{x^2}{2\sigma^2}}$. [13] showed that adding i.i.d. noise sampled from $\mathcal{N}\left(0, \left(\frac{1+\sqrt{2\log(1/\delta)}}{\epsilon}\right)^2 \mathsf{Sensitivity}_2(\mathsf{f})^2\right)$ to $f(D)$ satisfies (ϵ, δ)-differential privacy.

2.3.2 Differentially private tree based aggregation

In this section for completeness purposes, we state the differentially private tree based aggregation algorithm. It is important to mention here is that a variant of this algorithm has appeared in many different papers. The initial version as proposed by [17, 9]. We apply the same algorithm in the matrix setting, and with symmetric noise matrices, where each entry is drawn from a Gaussian distribution.

For the ease of exposition, assume that m is a power of two. Given a set of symmetric matrices $W_1, \cdots, W_m$, with $\|W_t\|_2 \leq 1$, the objective is to output $S = \langle W_1, W_1 + W_2, \cdots, \sum_{\tau=1}^{m} W_\tau \rangle$ while preserving (ϵ, δ)-differential privacy. First notice to ensure each of the partial sums are individually (ϵ, δ)-differentially private, it suffices to add a noise matrix $E \in \Re^{n \times n}$ which is a symmetric matrix whose upper triangle is i.i.d. samples from $\mathcal{N}\left(0, \frac{50\log(1/\delta)}{\epsilon^2}\right)$. By the standard sequential composition of differential privacy [15, 14] it follows that if one sets $\epsilon = \epsilon/m$ and $\delta = \delta/m$, then the sequence of partial sums is (ϵ, δ)-differentially private. In the following we provide a scheme by which one can add much lesser noise in computing the partial sums.

Consider a complete binary tree T with m leaves. The leaves in the tree corresponds to $W_1, \cdots, W_m$, and each internal node in the tree corresponds to the partial sums of the leaves in its subtree. It is not hard to see that one can reconstruct the sequence S using only $\log m$ number of nodes in the tree. Also see that if we consider T to be a vector of size $2m - 1$, then changing one of the W_t's only affects $\log m$, number of entries in T. So instead of outputting a private version of the sequence S directly, the idea is to output a noisy version of the binary tree T and then reconstruct the private version of S from it. Hence to ensure (ϵ, δ)-differential privacy, it suffices to add i.i.d. symmetric noise matrices to each node of T, whose each entry is distributed i.i.d. $\mathcal{N}\left(0, \frac{50\log^3(m/\delta)}{\epsilon^2}\right)$. The privacy guarantee follows from a direct application of the sequential composition property of differential privacy [15, 14].

2.4 Summary of main results

For the purposes of brevity, throughout the paper, we use $\widetilde{O}(\cdot), \widetilde{\Omega}(\cdot)$ to hide factors of $1/\epsilon$ and polynomial dependence on $\log(1/\delta), \log m$, and $\log n$, and "with high probability" means with probability $1 - 1/n^{\Omega(1)}$, under the internal randomness of the mechanism. Our first result (Main Result 1) is that the Gaussian mechanism is nearly optimal in the worst case[4]. We further show (Main Result 2) that, under natural assumptions on the data matrix A, this mechanism has even stronger utility guarantees.

Main Result 1 (Theorems 4 and 27 informal version)**.**

1. *For any $\epsilon, \delta > 0$ and $1 \leq k \leq n$, the Gaussian mechanism described in Thoerem 2 ensures that for any $A \in \mathcal{A}$, with high probability over the coin tosses of the mechanism, $\|A\mathcal{M}(A)\|_F^2 \geq \|AV_k(A)\|_F^2 - \widetilde{O}(k\sqrt{n})$.*

[4]We will tweak the mechanism slightly by ensuring that the matrix of noise values added to $A^T A$ is symmetric. We abuse notation by referring to this symmetric version simply as the Gaussian mechanism.

2. *The Gaussian mechanism is nearly optimal: for any $1 \leq k \leq n$ and any $(\epsilon, 1/n^2)$-differentially private mechanism $\mathcal{M}$, there exists $A \in \mathcal{A}$ such that $\|A\mathcal{M}(A)\|_F^2 \leq \|AV_k(A)\|_F^2 - \widetilde{\Omega}(k\sqrt{n})$.*

Main Result 2 (Theorems 5 and 7 informal version)**.** *Let $\sigma_1 \geq \cdots \geq \sigma_n$ be the singular values of $A \in \mathcal{A}$. Assuming $\sigma_k^2 - \sigma_{k'+1}^2 = \omega(\sqrt{n})$, with high probability the Gaussian mechanism $\mathcal{M}$ satifies*

$$\|A\mathcal{M}(A)\|_F^2 \geq \|AV_k\|_F^2 - \widetilde{O}\left(\frac{k'n}{\sigma_k^2 - \sigma_{k'+1}^2}\right).$$

Additionally, when $k' = k$,

$$\|\mathcal{M}(A)\mathcal{M}(A)^T - V_k V_k^T\|_2 = \widetilde{O}\left(\frac{\sqrt{n}}{\sigma_k^2 - \sigma_{k+1}^2}\right).$$

Finally, we consider the online version in which a_t arrives in a stream for $t = 1, \cdots, m$, and the mechanism $\mathcal{M}$ is required to compute a k-dimensional subspace $\mathcal{M}_t = \mathcal{M}(a_1, \cdots, a_{t-1})$ before seeing a_t. Define $\mathsf{OPT} = \max_{P \in \mathbb{P}_k} \sum_{t=1}^m \|P^T a_t\|_2^2$. The regret of $\mathcal{M}$ is defined as $\mathsf{Regret}(\mathcal{M}) = \mathsf{OPT} - \sum_{t=1}^m \|\mathcal{M}_t^T a_t\|_2^2$. We show (in Main Result 3) that by adding carefully calibrated noise, the Following the Perturbed Leader algorithm in [25] can be made both private and with low regret. And the regret bound is nearly optimal for any online private PCA algorithm.

Main Result 3 (Theorem 18 informal version)**.** *When $\mathsf{OPT} = \widetilde{\Omega}(k\sqrt{n}/\epsilon^2)$, we can obtain an (ϵ, δ)-differentially private online mechanism $\mathcal{M}$ such that $\mathbb{E}[\mathsf{Regret}(\mathcal{M})] = \widetilde{O}(\sqrt{k\mathsf{OPT}}n^{1/4})$. This bound is nearly optimal for $\mathsf{OPT} = \widetilde{O}(k\sqrt{n})$.*

2.5 Detailed comparison to prior work

There have been a series of recent works in this area of private singular subspace computation and private low-rank approximation [4, 21, 10, 26, 22]. Roughly speaking, these results can be categorized into three classes: i) *spectral perturbation methods* [4, 21, 10], ii) *exponential sampling method* [10, 26] and iii) *iterative power method* [22]. In this section, we provide comparison our approach to each of these methods. For the purposes of brevity, all the bounds below ignore parameters in $1/\epsilon$ and $\log(1/\delta)$.

Spectral perturbation methods [4, 21, 10]. The earliest work this area is that of [4] followed up by [10]. For a given matrix $A \in \Re^{m \times n}$ (with rows being individual data records), adding i.i.d. scaled Gaussian noise (with standard deviation roughly $O(n)$) to the covariance matrix $A^T A$ preserves (ϵ, δ)-differential privacy. In our work we make two modifications to this basic algorithm: i) we show that adding noise with standard deviation $O(1)$ is sufficient, and ii) we use a symmetric noise matrix. These modifications alone give us bound on the error in variance of $O(k\sqrt{n})$. In comparison, [4, 10] would result in an error of $O(kn)$. Furthermore, we show that the error bound can be improved to $O(kn/(\sigma_k^2 - \sigma_{k+1}^2))$ when there is a separation between σ_k^2 and σ_{k+1}^2. (In fact similar result hold even when the adjacent singular values for k and $k+1$ are not well seperated, as long as there is a reasonbale degradation of singular values and one is willing to output a slightly higher rank subspace.) It is important to mention here that the improved privacy analysis was also provided in [15], but no formal connection was provided to singular subspace computation. Using this same algorithm, we can obtain subspace closeness guarantees too, which to the best of our knowledge is the first of its kind.

[21] gave an algorithm for private low-rank approximation using random projection methods from [19]. Their algorithm outputs a rank k approximation $\hat{A}_k$ to the original matrix such that $\|A - \hat{A}_k\|_F \leq \|A -$

$A_k\|_F + O(k\sqrt{m} + \text{additional terms})$, where A_k is the best non-private rank k approximation of A. In our setting where $m \geq n$ and each row of A has a bounded L_2 norm, their results is not meaningful as $\|A\|_F$ is at most $\sqrt{m}$. However, we can show that projecting A onto the private rank k subspace $\widehat{V}_k$ computed by our algorithm is a very good rank k approximation in the *squared* Frobenius norm, i.e., $\|A - A(\widehat{V}_k\widehat{V}_k^T)\|_F^2 \leq \|A - A_k\|_F^2 + O(\min\{k\sqrt{n}, \frac{kn}{\sigma_k^2 - \sigma_{k+1}^2}\})$. One important distinction from [21] is that guarantee that $\hat{A}_k$ is differentially private, while we just ensure the projector is private. In most machine learning scenarios $\widehat{V}_k$ seems to be the object of interest, since it is used for pre-processing of the data (like dimension reduction).

One can trivially modify our algorithm to get a differentially private low-rank approximation to A^TA (see Section 3.3), and in that case our error guarantees are no worse than [21].

Exponential sampling methods [10, 26]. At the very outset it is important to mention that the results of [10] and [26] are for pure ϵ-differential privacy, which is a stronger privacy guarantee than ours. These two independent works select a good rank k subspace based on the exponential mechanism [33]. Since the error guarantees in [26] is strictly more general than [10], we compare our results to [26]. With a (mild) loss of generality we translate the results of [26] into our setting. When $k = 1$, the variance error in [26] is $O(n)$, which they show is optimal for pure DP. This is strictly worse than the variance error (stated earlier) of our algorithm. In the rank k case, they provide error guarantees for private rank k approximation as in [21] but under spectral norm. Using the same notation as above, they output a rank k matrix B such that $\|A^TA - B\|_2 \leq \|A^TA - (A^TA)_k\|_2 + O(nk^3)$. Our algorithm will incur an error of $O(\sqrt{n})$. Notice the rank independence of the error bound.

Iterative power method [22, 20]. In this work the authors used the well-known *power method* to get variance error bound for the top private right singular vector of A, and rank k approximation to A with the error being measured in spectral norm (as in [26]). First point to notice is that the privacy model of [22] is incomparable to ours, namely, they allow change of one entry by a bounded value between two neighboring matrices A an A' . Whereas we allow addition or deletion of one complete row (of bounded L_2 norm) from the matrix. Translating their result to our privacy model, their variance error for the top singular vector is $O(\sqrt{n})$. Using their algorithm for rank k estimation of A^TA (with the error measured in $\|\cdot\|_2$ norm), the error is $O(k^2\sqrt{n})$.

Recent independent work of Hardt [20] shows that subspace iteration method is robust to perturbations, so that the error due to adding noise at each step to give privacy can be bounded. This allows him to bypass the peeling approach used in [22], leading to improved bounds for rank-k approximation in the same privacy model as [22] as well as get better bounds under an incoherence assumption. Moreover, the approach can also be used to give privacy under spectral norm 1 perturbations to a matrix. Applying their bounds on the matrix A^TA, one get a bound of the form $\|A^TA - \mathcal{M}(A^TA)\|_2 \leq \sigma_{k+1}^2 + \tilde{O}(\frac{\sigma_1^2}{\sigma_k^2\gamma^{1.5}}\sqrt{kn})$, where $\gamma = (\frac{\sigma_k^2}{\sigma_{k+1}^2} - 1)$ is a measure of the gap between σ_k and σ_{k+1}. These bounds are incomparable to our results on spectral norm error in Theorem 9 as they give privacy under a larger class of perturbations, whereas their error bound can be larger than our bound of $\tilde{O}(\sqrt{n})$. Whereas the $\sin\Theta$ theorem also makes an appearance in [20], we remark that the privacy model, the algorithm and analysis are all different from our work.

Comparison to (non-private) online singular subspace computation. To our knowledge, [44] provided the first algorithm for online private singular subspace computation, followed up by a series of results improving various other aspects of the problem such as computational efficiency [24, 23, 35]. They used a generalization of the *multiplicative weights algorithm* for experts problem to obtain a regret bound of $\tilde{O}(\sqrt{k\mathsf{OPT}})$, where OPT is the maximum variance captured by the offline algorithm. A direct adaptation of their algorithm to the private setting will result in a regret of $\tilde{O}(\sqrt{kmn^{5/4}})$, where m is the total number of rows and n is the dimensionality of the problem. Instead, we use an adaptation of the *follow the perturbed leader* (FPL) algorithm of [25] to obtain a regret guarantee of $\tilde{O}(\sqrt{k\mathsf{OPT}}n^{1/4})$ while satisfying differential privacy. To our knowledge, this is the best FPL algorithm in the matrix setting, improving on [23] which achieved a

regret of $\tilde{O}(\sqrt{kn\mathsf{OPT}})$. Moreover, we believe that this algorithm can be used in other online learning problems like *learning rotations* [23] to obtain tighter regret guarantees even without privacy requirements.

3 Private Singular Subspace Computation via The Gaussian Mechanism

The Gaussian mechanism (with symmetric noise matrix) is straightforward: just release $\widehat{C} = A^T A + E$ where E is a symmetric noise matrix, with each (upper-triangle) entry drawn i.i.d. from Gaussian distribution with sufficiently high variance. Algorithm 1 describes such a mechanism, a variation of those in [4, 15] that enjoys smaller noise, and in which the noise matrix is symmetric. Set $\Delta_{\epsilon,\delta} = \sqrt{2\ln(1.25/\delta)}/\epsilon$.

Algorithm 1 The Gaussian Mechanism: releasing the covariance matrix privately

Input: matrix $A \in \Re^{m\times n}$, and privacy parameters $\epsilon, \delta > 0$.

1: $E \in \Re^{n\times n}$ be a symmetric matrix where the upper triangle (including the diagonal) is i.i.d. samples from $\mathcal{N}\left(0, {\Delta_{\epsilon,\delta}}^2\right)$, and each lower triangle entry is copied from its upper triangle counterpart.

2: Output $\widehat{C} \leftarrow A^T A + E$.

Differential privacy is closed under post-processing, so the data analyst can run any post-processing algorithm on $\widehat{C}$ with no further erosion of privacy. In particular, the analyst can compute the singular decomposition of $\widehat{C}$ to obtain any k-dimensional principal singular subspace $\widehat{V}_k$ of $\widehat{C}$. But how useful is such a $\widehat{V}_k$? In this section, we will show that $\widehat{V}_k$ can actually be a quite good approximation to the principal rank-k right singular subspace V_k of A (or equivalently the principal singular subspace of $A^T A$.) In particular, we consider three measures: *1) How well does $\widehat{V}_k$ capture the variance of A compared to V_k? 2) How close is $\widehat{V}_k$ to V_k? and 3) How well does the best rank-k approximation of $\widehat{C}$ approximate $A^T A$?*

Our analyses come in two flavors. One is on the worst case guarantee, where no assumption is made on A. Most of these results follow relatively easily from random matrix theory. As we will show later by our lower bound, one cannot expect to outperform these bounds in the worst case. The other set of results depend on the spectrum of $A^T A$. We show, by using tools from matrix perturbation theory, that when the spectrum of $A^T A$ has large drop in its eigenvalues, $\widehat{V}_{k'}$ can be a much better approximation to V_k when $k' \geq k$. For example, when the data are drawn from a distribution with an eigengap, the error will go to 0 as the number of samples $m \to \infty$! Since the presence of such drop is one of the rationales for principal components analysis, these results are probably more interesting in practice. We emphasize that this improved data dependent bound holds for the same algorithm; the gain comes entirely from the analysis.

3.1 Variance guarantee

We now consider how well $\widehat{V}_k$ captures the variance of A. We first provide a worst case bound.

Theorem 4 (Worst case utility guarantee)**.** *Let V_k be the principal rank-k right singular subspace of A and let $\widehat{V}_k$ be the principal rank-k subspace of the matrix $\widehat{C}$ (output by Algorithm 1). Then with high probability,*

$$\|A\widehat{V}_k\|_F^2 \geq \|AV_k\|_F^2 - O\left(k\sqrt{n}\Delta_{\epsilon,\delta}\right) .$$

Proof. We have the following with the noise matrix E in Algorithm 1.

$$\begin{aligned}\mathsf{tr}(V_k V_k^T(A^TA+E)) &= \mathsf{tr}(V_kV_k^T(A^TA)) + \sum_{i=1}^{k} v_i E v_i^T \\ &\geq \sum_{i=1}^{k}\sigma_i^2 - k\|E\|_2. \end{aligned} \tag{1}$$

By definition, the highest singular subspace captures the maximum variance. Therefore,

$$\mathsf{tr}(\widehat{V}_k^T(A^TA+E)\widehat{V}_k) \geq \mathsf{tr}(V_k^T(A^TA+E)V_k). \tag{2}$$

Combining (1) and (2), we get the following.

$$\begin{aligned}&\mathsf{tr}(\widehat{V}_k^T(A^TA+E)\widehat{V}_k) \geq \sum_{i=1}^{k}\sigma_i^2 - k\|E\|_2 \\ \Leftrightarrow\ &\mathsf{tr}(\widehat{V}_k^T(A^TA)\widehat{V}_k) \geq \sum_{i=1}^{k}\sigma_i^2 - k\|E\|_2 - \mathsf{tr}(\widehat{V}_k^T E\widehat{V}_k) \\ \Rightarrow\ &\mathsf{tr}(\widehat{V}_k^T(A^TA)\widehat{V}_k) \geq \sum_{i=1}^{k}\sigma_i^2 - 2k\|E\|_2 \end{aligned} \tag{3}$$

Since E is a symmetric Gaussian ensemble, by Corollary 2.3.6 from [42], with probability at least $1 - negl(n)$, $\|E\|_2 = O\left(\sqrt{n}\Delta_{\epsilon,\delta}\right)$. This completes the proof. □

As we will see in Section 5, the above bound is nearly tight in the worst case. Now, suppose there is a large eigengap, so that $\sigma_k - \sigma_{k+1} \in \omega(\sqrt{n})$. In this case we will see that $\hat{V}_k$ can provide utility that beats the worst-case lower bound. Moreover, an analogous claim holds even if there is not a precipitous drop between adjacent eigenvalues but a gap holds for non adjacent eigenvalues.

Theorem 5 (Spectrum separation guarantee)**.** *Let $\sigma_1 \geq \cdots \geq \sigma_n$ be the singular values of the data matrix A. Let V_k be the principal rank-k right singular subspace of A. Let $\widehat{V}_{k'}$ be the principal $k' \geq k$-dimensional subspace of the matrix $\widehat{C}$ (output by Algorithm 1). Assuming $\sigma_k^2 - \sigma_{k'+1}^2 = \omega(\sqrt{n}\Delta_{\epsilon,\delta})$, with high probability,*

$$\|A\widehat{V}_{k'}\|_F^2 \geq \|AV_k\|_F^2 - O\left(\frac{k' n {\Delta_{\epsilon,\delta}}^2}{\sigma_k^2 - \sigma_{k'+1}^2}\right).$$

Proof. The basic tool in our analysis is a sin-θ theorem, which is a generalization of the classic *Davis-Kahan sin-θ* theorem [11]. By the optimality of $\widehat{V}_{k'}$ and using $k' \geq k$, we have,

$$\begin{aligned}\mathsf{tr}(\widehat{V}_{k'}^T(A^TA)\widehat{V}_{k'}) &\geq \mathsf{tr}(V_k^T(A^TA)V_k) + \mathsf{tr}(V_k^T E V_k) - \mathsf{tr}(\widehat{V}_{k'}^T E\widehat{V}_{k'}) \\ &= \mathsf{tr}(V_k^T(A^TA)V_k) + \mathsf{tr}\left(\left(V_kV_k^T - \widehat{V}_{k'}\widehat{V}_{k'}^T\right)E\right).\end{aligned}$$

For the ease of notation, let $\Pi = V_kV_k^T$ and $\widehat{\Pi} = \widehat{V}_{k'}\widehat{V}_{k'}^T$. To bound $\mathsf{tr}\left(\left(\Pi - \widehat{\Pi}\right)E\right)$, we use Von Neumann's trace inequality: For two matrices $X \in \Re^{n\times n}$ and $Y \in \Re^{n\times n}$, let $\sigma_i(X), \sigma_i(Y)$ be the decreasingly ordered singular values of X, Y, respectively. Then $|\mathsf{tr}(XY)| \leq \sum_{i=1}^{n}\sigma_i(X)\sigma_i(Y)$. Hence, we have

$$|\mathsf{tr}\left(\left(\Pi - \widehat{\Pi}\right)E\right)| \leq \sum_{i=1}^{n}\sigma_i\left(\Pi - \widehat{\Pi}\right)\cdot\sigma_i(E). \tag{4}$$

Since $\left(\Pi - \widehat{\Pi}\right)$ is of rank at most $k + k' \leq 2k'$, at most $2k'$ of the σ_i are non-zero. So we further have

$$\begin{aligned} |\mathsf{tr}\left(\left(\Pi - \widehat{\Pi}\right) E\right)| &\leq \|E\|_2 \sum_{i=1}^{2k'} \sigma_i \left(\Pi - \widehat{\Pi}\right) \\ &\leq \sqrt{2k'}\|E\|_2 \left\|\Pi - \widehat{\Pi}\right\|_F \end{aligned} \tag{5}$$

We now have the following.

$$\Pi - \widehat{\Pi} = \Pi(\mathbb{I} - \widehat{\Pi}) - (\mathbb{I} - \Pi)\widehat{\Pi} = \Pi\widehat{\Pi}^{\perp} - \Pi^{\perp}\widehat{\Pi}\,. \tag{6}$$

Plugging (6) in (5), we have the following.

$$\begin{aligned} |\mathsf{tr}\left(\left(\Pi - \widehat{\Pi}\right) E\right)| &\leq \sqrt{2k'}\|E\|_2 \left\|\Pi\widehat{\Pi}^{\perp} - \Pi^{\perp}\widehat{\Pi}\right\|_F \\ &\leq \sqrt{2k'}\|E\|_2 \left(\left\|\Pi\widehat{\Pi}^{\perp}\right\|_F + \left\|\Pi^{\perp}\widehat{\Pi}\right\|_F\right) \\ &= \sqrt{2k'}\|E\|_2 \left(\left\|\Pi\widehat{\Pi}^{\perp}\right\|_F + \left\|\widehat{\Pi}\Pi^{\perp}\right\|_F\right) \qquad (7) \\ &\leq \sqrt{2}k'\|E\|_2 \left(\|\Pi\widehat{\Pi}^{\perp}\|_2 + \|\widehat{\Pi}\Pi^{\perp}\|_2\right), \qquad (8) \end{aligned}$$

where (7) follows because $\Pi^{\perp}, \widehat{\Pi}$ are symmetric matrices (since they are projectors), and for symmetric E, F, $\|EF\|_F = \|FE\|_F$.

Let $X, Y \in \Re^{n\times n}$ be two symmetric matrices, and let $\lambda_1(X) \geq \cdots$ and $\lambda_1(Y) \geq \cdots$ be the corresponding eigenvalues of X and Y. Let $\Pi_X^{(i)}$ be the projector to the subspace spanned by the top i singular vectors of X, where $i \leq n$. To bound $\|\Pi\widehat{\Pi}^{\perp}\|_2$ and $\|\widehat{\Pi}\Pi^{\perp}\|_2$, we will use the following result from matrix perturbation theory, which generalizes [11]:

Theorem 6 (Sin-Θ theorem [32] (Corollary 8))**.** *For any* $1 \leq i, j \leq n$*,*

$$(\lambda_i(X) - \lambda_{j+1}(Y))\|\Pi_X^{(i)}(\mathbb{I} - \Pi_Y^{(j)})\|_2 \leq \|X - Y\|_2\,.$$

Now to bound $\|\Pi\widehat{\Pi}^{\perp}\|_2$ in (8), we use Theorem 6 with $X = A^TA$ and $Y = A^TA + E$. Notice that $\|Y - X\|_2 = \|E\|_2$, and since E is a symmetric Gaussian ensemble, by Corollary 2.3.6 from [42], with high probability, $\|E\|_2 = O\left(\sqrt{n}\Delta_{\epsilon,\delta}\right)$. Also by Weyl's inequality (Theorem 3) it follows that $\lambda_{j+1}(Y) \leq \lambda_{j+1}(X) + \|E\|_2$. Plugging these bounds in Theorem 6 and recalling that $\sigma_k^2 - \sigma_{k'+1}^2 = \omega(\sqrt{n}\Delta_{\epsilon,\delta})$ (by assumption), we get $\|\Pi\widehat{\Pi}^{\perp}\|_2 = O\left(\frac{\sqrt{n}\Delta_{\epsilon,\delta}}{\sigma_k^2 - \sigma_{k'+1}^2}\right)$.

Using the same argument as above, and selecting $X = A^TA + E$ and $Y = A^TA$ in Theorem 6, we get $\|\widehat{\Pi}\Pi^{\perp}\|_2 = O\left(\frac{\sqrt{n}\Delta_{\epsilon,\delta}}{\sigma_k^2 - \sigma_{k'+1}^2}\right)$. Theorem 5 follows now follows from the bounds on $\|\Pi\widehat{\Pi}^{\perp}\|_2$ and $\|\widehat{\Pi}\Pi^{\perp}\|_2$. □

While the bound in Theorem 4 may not be useful when $\sigma_k^2 - \sigma_{k'+1}^2$ is small, in many cases (even for $k' = k$) the gap is quite large, especially when the number of samples m is large. Here we give two examples. In the first example, suppose that a_i's are drawn i.i.d. from some distribution with a specturm gap, say α, between σ_k^2 and σ_{k+1}^2. Then by the matrix concentration bound, it is easy to see that when $m \gg \sqrt{n\log n}/\alpha$, the gap is $\Omega(\alpha m)$ with high probability. In this case, Theorem 5 provides a better bound than Theorem 4. In the second example the a_i's are random Gaussian vectors, where there is no eigengap (in this case the usefulness of PCA is problematic but we use it as an illustration). For m random samples, the gap between two consecutive eigenvalues is expected to be $\Omega(\sqrt{m}/n^2)$, so in this case, Theorem 5 provides a better upper bound whenever $m = \Omega(n^5)$. In both cases, the error gap of Algorithm 1 goes to 0 when $m \to \infty$!

Bounds on residual variance. We observe that by Pythagorean theorem, $\|A - A(V_k V_k^T)\|_F^2 = \|A\|_F^2 - \|AV_k\|_F^2$. Since the bounds in Theorem 4 and 5 are additive, the same error guarantees hold if we are to minimize the total variance projected in the residual space.

3.2 Closeness to the right singular subspace

Another consequence of Theorem 5 is that when there is a spectrum gap in $A^T A$, $\widehat{V}_k$ not only captures large amount of variance, but is also close to the top k right singular subspace V_k of A. In Theorem 7, we provide the closeness between them, measured by the $\|\cdot\|_2$ norm. We note that the spectrum gap is necessary for such a bound as otherwise the top k-singular space is not uniquely defined. (In Section 3.4 we provide another technique (using subspace perturbation) to achieve similar subspace closeness guarantee.)

Theorem 7 (Subspace closeness)**.** *Let $\sigma_1 \geq \cdots \geq \sigma_n$ be the singular values of the data matrix A. Assuming $\sigma_k^2 - \sigma_{k+1}^2 = \omega(\sqrt{n}\Delta_{\epsilon,\delta})$, then with high probability,*

$$\left\| V_k V_k^T - \widehat{V}_k \widehat{V}_k^T \right\|_2 = O\left(\frac{\sqrt{n}\Delta_{\epsilon,\delta}}{\sigma_k^2 - \sigma_{k+1}^2} \right).$$

Proof. In order to prove the convergence in spectral norm, we need the following theorem.

Theorem 8 (Theorem I.5.5 in [40])**.** *Let $\mathcal{X}$ and $\mathcal{Y}$ be two k-dimensional subspaces of R^n, with orthogonal projectors Π and $\widehat{\Pi}$, respectively. If $s_1 \geq \cdots s_k \geq 0, \cdots$ are the singular values of $\Pi\widehat{\Pi}^{\perp}$, then the singular values of $\Pi - \widehat{\Pi}$ are*

$$s_1, s_1, s_2, s_2, \cdots, s_k, s_k, 0, \cdots.$$

Following the notation in the proof of Theorem 5, let $\Pi = V_k V_k^T$ and $\hat{\Pi} = \widehat{V}_k \widehat{V}_k^T$ be the orthogonal projectors. In the proof of Theorem 5, we already showed that with high probability, $\|\Pi\widehat{\Pi}^{\perp}\|_2 = O\left(\frac{\sqrt{n}\Delta_{\epsilon,\delta}}{\sigma_k^2 - \sigma_{k+1}^2}\right)$. Now, directly by Theorem 8, we have the required convergence in the spectral norm. □

We note that the above bound implies an upper bound in terms of the Frobenius norm

$$\left\| V_k V_k^T - \widehat{V}_k \widehat{V}_k^T \right\|_F = O\left(\frac{\sqrt{kn}\Delta_{\epsilon,\delta}}{\sigma_k^2 - \sigma_{k+1}^2} \right).$$

3.3 Low-rank approximation to the covariance matrix

$\widehat{C}$ also provides a good low rank approximation to $A^T A$ in the settings considered in seveal earlier works [21, 26, 22] (see Section 1 for comparison).

Theorem 9 (Low rank approximation)**.** *Let $A \in \Re^{m \times n}$ be the input data matrix and let C_k be the best rank-k approximation to $A^T A$. Let $\widehat{C}_k$ be the rank-k approximation to $\widehat{C}$ (output by Algorithm 1). Then with high probability,*

- $\|A^T A - \widehat{C}_k\|_F \leq \|A^T A - C_k\|_F + O\left(\Delta_{\epsilon,\delta} k\sqrt{n}\right)$.
- $\|A^T A - \widehat{C}_k\|_2 \leq \|A^T A - C_k\|_2 + O\left(\Delta_{\epsilon,\delta}\sqrt{n}\right)$.

In the above, the spectral norm bound can be derived immediately from [1] (Lemma 1.1). For the Frobenius norm bound, compared to the bound there, we need to prove a strengthened version with a better dependence on the spectrum of $C = A^T A$.

Proof. Write $C = A^T A$. Denote by $\Pi = V_k V_k^T$ and $\widehat{\Pi} = \widehat{V}_k \widehat{V}_k^T$ the projection to the rank-k principal subspace V_k of C and $\widehat{V}_k$ of $\widehat{C}$, respectively. We will use the following facts:

$$\mathsf{tr}(X) \le \mathrm{rank}(X)\|X\|_2 \quad \text{and} \tag{9}$$

$$\|X\|_F = \sqrt{\mathsf{tr}(X^T X)} \le \sqrt{\mathrm{rank}(X)}\|X\|_2 \tag{10}$$

First notice

$$\begin{aligned}\|C - \widehat{C}_k\|_F &= \|C - \widehat{C}\widehat{\Pi}\|_F = \|C - (C+E)\widehat{\Pi}\|_F \\ &\le \|C - C\widehat{\Pi}\|_F + \|E\widehat{\Pi}\|_F \quad \text{by triangle inequality,} \\ &\le \|C - C\widehat{\Pi}\|_F + \sqrt{k}\|E\|_2 \quad \text{by (10).}\end{aligned} \tag{11}$$

For $\|C - C\widehat{\Pi}\|_F$, we have the following bound.

Lemma 10. $\|C - C\widehat{\Pi}\|_F^2 \le \|C - C\Pi\|_F^2 + 8k\sigma_{k+1}(C)\|E\|_2 + 10k\|E\|_2^2$.

Proof. We note that in the following $C, E, \widehat{C}, \Pi, \widehat{\Pi}$ are symmetric matrices, and for any projection matrix P, $P^2 = P$.

$$\begin{aligned}\|C - C\widehat{\Pi}\|_F^2 &= \|C\widehat{\Pi}^\perp\|_F^2 = \mathsf{tr}(C^2\widehat{\Pi}^\perp) \\ &= \mathsf{tr}((\widehat{C}^2 - CE - EC - E^2)\widehat{\Pi}^\perp) \\ &= \mathsf{tr}(\widehat{C}^2\widehat{\Pi}^\perp) - \mathsf{tr}((CE+EC)\widehat{\Pi}^\perp) - \mathsf{tr}(E^2\widehat{\Pi}^\perp)\end{aligned} \tag{12}$$

Since $\widehat{\Pi}$ is the projection to the rank-k principal subspace of $\widehat{C}$, we have

$$\begin{aligned}\mathsf{tr}(\widehat{C}^2\widehat{\Pi}^\perp) &\le \mathsf{tr}(\widehat{C}^2\Pi^\perp) = \mathsf{tr}((C+E)^2\Pi^\perp) \\ &= \mathsf{tr}(C^2\Pi^\perp) + \mathsf{tr}((CE+EC)\Pi^\perp) + \mathsf{tr}(E^2\Pi^\perp) \\ &= \|C - C\Pi\|_F^2 + \mathsf{tr}((CE+EC)\Pi^\perp) + \mathsf{tr}(E^2\Pi^\perp)\,.\end{aligned} \tag{13}$$

Combining (12) and (13), we have

$$\begin{aligned}\|C - C\widehat{\Pi}\|_F^2 &\le \|C - C\Pi\|_F^2 + \mathsf{tr}((CE+EC)(\Pi^\perp - \widehat{\Pi}^\perp)) + \mathsf{tr}(E^2(\Pi^\perp - \widehat{\Pi}^\perp)) \\ &= \|C - C\Pi\|_F^2 + \mathsf{tr}((CE+EC)(\widehat{\Pi} - \Pi)) + \mathsf{tr}(E^2(\widehat{\Pi} - \Pi)) \\ &\qquad \text{by } \mathrm{rank}(\widehat{\Pi} - \Pi) \le 2k \text{ and (9)} \\ &\le \|C - C\Pi\|_F^2 + 2k(\|EC(\widehat{\Pi} - \Pi)\|_2 + \|E(\widehat{\Pi} - \Pi)C\|_2) + 2k\|E^2\|_2 \\ &\le \|C - C\Pi\|_F^2 + 4k\|E\|_2\|C(\widehat{\Pi} - \Pi)\|_2 + 2k\|E\|_2^2\,.\end{aligned} \tag{14}$$

To bound $\|C(\widehat{\Pi} - \Pi)\|_2$, we observe that

$$\begin{aligned}\|C(\widehat{\Pi} - \Pi)\|_2 &= \|C(\Pi^\perp - \widehat{\Pi}^\perp)\|_2 \le \|C\Pi^\perp\|_2 + \|C\widehat{\Pi}^\perp\|_2 \\ &\le \|C\Pi^\perp\|_2 + \|\widehat{C}\widehat{\Pi}^\perp\|_2 + \|E\widehat{\Pi}^\perp\|_2\,.\end{aligned}$$

By the definition of Π and $\widehat{\Pi}$, we have $\|C\Pi^\perp\|_2 = \sigma_{k+1}(C)$ and $\|\widehat{C}\widehat{\Pi}^\perp\|_2 = \sigma_{k+1}(\widehat{C}) \le \sigma_{k+1}(C) + \|E\|_2$ (by Weyl's inequality). So $\|C(\widehat{\Pi} - \Pi)\|_2 \le 2(\sigma_{k+1}(C) + \|E\|_2)$. Plugging it into (14) completes the proof. □

Now write $X = \|C - C\widehat{\Pi}\|_F$ and $Y = \|C - C\Pi\|_F$ for brevity. Notice that $Y^2 \geq \|C - C\Pi\|_2^2 = \sigma_{k+1}(C)^2$. From Lemma 10 above we have

$$\begin{aligned} X^2 &\leq Y^2 + 8kY\|E\|_2 + 10k\|E\|_2^2 \\ &= (Y + 4k\|E\|_2)^2 - 16k^2\|E\|_2^2 + 10k\|E\|_2^2 \\ &\leq (Y + 4k\|E\|_2)^2 \ . \end{aligned}$$

Hence, $\|C - C\widehat{\Pi}\|_F \leq \|C - C\Pi\|_F + 4k\|E\|_2$. Combining this bound with (11) and using the fact that with probability at least $1 - \text{negl}(n)$, $\|E\|_2 = O(\Delta_{\epsilon,\delta}\sqrt{n})$ (Corollary 2.3.6 from [42]) complete the proof.

To prove the second part, recall that $\hat{C} = A^T A + E$, where E is a symmetric matrix with entries drawn i.i.d. from $\mathcal{N}(0, {\Delta_{\epsilon,\delta}}^2 \mathbb{I})$. Let $\hat{\sigma}_1 \geq \cdots \geq \hat{\sigma}_n$ be the eigenvalues of $\hat{C}$ and $\sigma_1^2 \geq \cdots \geq \sigma_n^2$ be the singular values of $A^T A$. We have the following bound.

$$\begin{aligned} \|A^T A - \hat{C}_k\|_2 &\leq \|(A^T A + E) - \hat{C}_k\|_2 + \|E\|_2 \\ &= \hat{\sigma}_{k+1}^2 + \|E\|_2 \\ &\leq \sigma_{k+1}^2 + 2\|E\|_2 \end{aligned}$$

The last inequality follows from Weyl's inequality (Theorem 3). Now using the fact that with probability at least $1 - \text{negl}(n)$, $\|E\|_2 = O(\Delta_{\epsilon,\delta}\sqrt{n})$, the proof for the second part is complete.

□

3.4 Private subspace recovery via subspace perturbation

Here we show a different randomized response algorithm, in which we apply SVD first to A and then release a noisy version of the top-k singular subspace of A. We show that we can achieve similar bounds to that in Theorem 13. One benefit of this algorithm compared to Algorithm 1 can be in the case of running time, when the matrix $A^T A$ sparse. The key observation is that when there is a large eigen gap, according to sin-Θ theorem, the principal singular space is quite "stable" against a unit norm rank one update, and hence we can perform the SVD to A and then add smaller noise to the top k-singular space to guarantee privacy. But we need to be careful not to reveal the gap by employing a variant of the *propose-test-and-release* (PTR) framework of [14].

Algorithm 2 Private Subspace Recovery

Input: matrix: $A \in \Re^{m \times n}$, rank parameter: k, and privacy parameters: $\epsilon, \delta > 0$.

1: $V\Sigma V^T \leftarrow$ Eigenvalue decomposition of $A^T A$. Let $|\lambda_1| \geq \cdots \geq |\lambda_n|$ be the eigenvalues.
2: $\hat{d} \leftarrow (|\lambda_k| - |\lambda_{k+1}|) + \text{Lap}\left(\frac{2}{\epsilon}\right)$.
3: $V_k \leftarrow$ Top k eigenvectors of $A^T A$ (as a column matrix).
4: $\hat{W} \leftarrow V_k V_k^T + E$, where $E \in \Re^{n \times n}$ is a symmetric matrix where the upper triangle is i.i.d. samples from $\mathcal{N}\left(0, \frac{{\Delta_{\epsilon,\delta}}^2}{(\hat{d} - 2(1+\log(1/\delta)/\epsilon))^2}\right)$, where $\Delta_{\epsilon,\delta} = \frac{1+\sqrt{2\log(1/\delta)}}{\epsilon}$.
5: Let $\widehat{V}\widehat{\Sigma}\widehat{V}^T$ be the eigenvalue decomposition of $\hat{W}$ and let $\widehat{V}_k$ be the top k eigenvectors of $\widehat{V}$ (as a row matrix). Output $\widehat{V}_k \widehat{V}_k^T$.

3.4.1 Privacy analysis of subspace recovery via subspace perturbation

Theorem 11 (Privacy guarantee)**.** *Algorithm 2 is* $(2\epsilon, 2\delta)$*-differentially private.*

Proof. We prove the privacy guarantee in three stages. In the first stage, we will show that Step 2 in Algorithm 2 is ϵ-differentially private. In the second stage, by the property of Laplace distribution we show that with probability at least $1-\delta$, in Step 2 the random variable $\hat{d} - 2\log(1/\delta)/\epsilon \leq |\lambda_k| - |\lambda_{k+1}|$. In the last stage, by the use of sin-Θ theorem, we show that it suffices to add Gaussian noise with standard deviation $\tilde{O}(1/(\alpha\epsilon))$ to the projector $V_k V_k^T$ to obtain the final privacy guarantee.

Stage 1. For any matrix A' which differ in one row from A, let $G = A^T A - A'^T A'$. By the L_2-bound on the rows of A and A', the highest absolute value for the eigenvalue of G is at most two. By Weyl's eigenvalue perturbation guarantee (Theorem 3), it follows that for any $i \in [p]$, $\left|\lambda_i(A^T A) - \lambda_i(A'^T A')\right| \leq 1$. Hence, by standard Laplace mechanism argument from [15], it follows that Step 2 is ϵ-differentially private.

Stage 2. By the tail bound on Laplace distribution it follows that that with probability at least $1-\delta$, in Step 2 the random variable $\hat{d} - 2\log(1/\delta)/\epsilon \leq |\lambda_k(A^T A) - \lambda_{k+1}(A^T A)|$.

Stage 3. We need the matrix perturbation theorem to complete the proof. This version of the Sin-Θ theorem bounds the Frobenius norm rather than the spectral norm in Theorem 6 in Section 3.1.

Theorem 12 (Sin-Θ theorem [11, 30])**.** *Let* $A^T A = \begin{pmatrix} V_1^T & V_2^T \end{pmatrix} \begin{pmatrix} \Sigma_1 & 0 \\ 0 & \Sigma_2 \end{pmatrix} \begin{pmatrix} V_1 \\ V_2 \end{pmatrix}$ *be a real valued matrix and let* G *be a real valued symmetric perturbation matrix. Let* $\tilde{\Sigma}_1$ *and* $\tilde{\Sigma}_2$ *be the corresponding singular values for the perturbed matrix* $A^T A + G$*. If* $\left|\sigma_{\min}(\tilde{\Sigma}_1) - \sigma_{\max}(\Sigma_2)\right| \geq \alpha$ *and* $\sigma_{\min}(\tilde{\Sigma}_1) \geq \alpha$*, then*

$$\left\|\Pi_{A^T A} - \Pi_{A^T A + G}\right\|_F \leq \frac{\|G\|_F}{\alpha}.$$

Here $\Pi_{A^T A}$ *refers to the orthogonal projectors onto the subspace spanned by* $A^T A$*.*

If $\left|\lambda_k(A^T A) - \lambda_{k+1}(A^T A)\right| \geq \alpha$, then by Weyl's inequality, it follows that $\left|\lambda_k(A'^T A') - \lambda_{k+1}(A^T A)\right| \geq (\alpha - 2)$. Hence by Theorem 12, we have $\|V_k' V_k'^T - V_k V_k^T\|_F \leq \frac{\|G\|_F}{\alpha-2} = \frac{2}{\alpha-2}$, where V_k' corresponds to the top k eigenvectors of $A'^T A'$. Now by standard differential privacy argument for *Gaussian mechanism*, the proof is complete. □

3.4.2 Utility analysis of subspace recovery via subspace perturbation

In this section we provide the utility guarantee for Algorithm 2 by bounding the spectral norm of the difference between $\widehat{V}_k \widehat{V}_k^T$ and $V_k V_k^T$.

Theorem 13 (Subspace convergence in spectral norm)**.** *Let* $\sigma_1 \geq \cdots \geq \sigma_n$ *be the singular values of the data matrix* A*. Under the randomness of the algorithm, with probability at least* $1 - 2\delta$*, Algorithm 2 outputs a* k*-dimensional subspace* $\widehat{V}_k$ *such that*

$$\left\|V_k V_k^T - \widehat{V}_k \widehat{V}_k^T\right\|_2 = O\left(\frac{\Delta_{\epsilon,\delta}\sqrt{n}}{\sigma_k^2 - \sigma_{k+1}^2 - \log(1/\delta)/\epsilon}\right).$$

Proof. From the spectral norm bound on symmetric random Gaussian matrices (see Corollary 2.3.6 from [42]), we know that with high probability, $\|E\|_2 = O\left(\frac{\Delta_{\epsilon,\delta}\sqrt{n}}{\hat{d} - (1+\log(1/\delta)/\epsilon)}\right)$, where E is the error matrix in Algorithm 2 and $\hat{d}$ is as defined in Step 2 of Algorithm 2. Notice that by the tail property of Laplace distribution, with probability at least $1-\delta$, $\hat{d} \geq \sigma_k^2 - \sigma_{k+1}^2 - 2\log(1/\delta)/\epsilon$. Thus with probability at least $1-2\delta$, $\|E\|_2 = O\left(\frac{\Delta_{\epsilon,\delta}\sqrt{n}}{\sigma_k^2 - \sigma_{k+1}^2 - \log(1/\delta)/\epsilon}\right)$.

By the definition of $\hat{W}$ in Algorithm 2, $\|\hat{W} - V_k V_k^T\|_2 = \|E\|_2$. Also, from Weyl's inequality (Theorem 3) we have $\left|\sigma_i(\hat{W}) - \sigma_i(V_k V_k^T)\right| \leq \|E\|_2$ for all $i \in [n]$. This in particular implies that, $\sigma_{k+1}(W)$ is at most $\|E\|_2$. Let $X = \hat{W} - \widehat{V}_k^T \widehat{V}_k$. By the above argument, $\|X\|_2 \leq \|E\|_2$. Hence, we have the following.

$$\begin{aligned}\|\widehat{V}_k \widehat{V}_k^T - V_k V_k^T\|_2 &= \|(\hat{W} - V_k V_k^T) - X\|_2 \\ &\leq \|\hat{W} - V_k V_k^T\|_2 + \|X\|_2 \\ &\leq 2\|E\|_2\end{aligned}$$

Plugging in the bound for $\|E\|_2$ computed earlier completes the proof the first part of the theorem. □

Note that the above bound immediately implies an error bound on the Frobenius norm as for any matrix B of rank k, $\|B\|_F \leq \sqrt{k}\|B\|_2$, and $\widehat{V}_k \widehat{V}_k^T - V_k V_k^T$ has rank at most $2k$. The bound holds as long as the gap is at least $\log(1/\delta)/\epsilon$.

4 Private Online Singular Subspace Computation

The design of the private online algorithm turns out to be closedly related to the class of *follow the perturbed leader* (FPL) algorithms of [25]. Such algorithms add regularization noise to the problem to reduce generalization error and hence the regret. This noise also reduces the dependence of the algorithm outcome on individual data items and therefore is aligned with the goal of providing privacy. Recall that $\mathbb{P}_k$ denotes the set of k-dimensional orthogonal projectors in $\Re^n$. Define $\mathsf{OPT} = \max_{P \in \mathbb{P}_k} \sum_{t=1}^m \|Pa_t\|^2$. We show that by using the Gaussian noise for the regularization noise, we can achieve the regret bound (defined in Section 2.4) of $\widetilde{O}(\sqrt{k\mathsf{OPT}}n^{1/4})$. As can be shown from the lower bound for the offline problem, the $\widetilde{O}(\sqrt{k\mathsf{OPT}}n^{1/4})$ bound is nearly optimal for $\mathsf{OPT} = O(k\sqrt{n})$. Therefore the $n^{1/4}$ gap between the regret of the non-private and private algorithms is essentially tight.

We now provide the details of our online algorithm. We will first present a FPL algorithm that gives regret of $\widetilde{O}(\sqrt{k\mathsf{OPT}}n^{1/4})$. The analysis of the algorithm borrows techniques from differential privacy but the algorithm itself is not private. We then show, by using the tree based aggregation technique, we can obtain a private online mechanism with similar regret bound with only extra $\log(m/\delta)$ factors.

4.1 An FPL algorithm for online singular subspace computation

For us the most interesting feature of the algorithm and its corresponding regret analysis is that differential privacy acts as a tool in providing a low regret guarantee. The key ingredient is the exploitation of the intuition that differential privacy ensures *robustness* [14, 39], which in turn guarantees *low* regret. More concretely, following [25], the analysis is done in two steps. First we design our FPL algorithm to be differentially private, i.e. producing similar results under any single data item change. In particular, the output is similar compared to the *be the perturbed leader* (BPL) algorithm in which we include the next data item. Secondly, we show that the BPL algorithm has small regret as the noise we added to the FPL algorithm is small. Notice that the privacy property is crucial in guaranteeing the regret bound.

Algorithm 3 is the formal description of our algorithm. At a high-level Algorithm 3 is similar to Algorithm 1, except it repeatedly executes Algorithm 1 at each time step $t \in [m]$ on all the data seen so far.

Theorem 14 (Regret guarantee)**.** *For $\epsilon < 1, \delta < 1/m$, the regret guarantee for Algorithm 3 is the following.*

$$\mathbb{E}[\mathsf{Regret}] = O\left(\frac{k\sqrt{n}\log m}{\epsilon} + \epsilon\mathsf{OPT} + 1\right).$$

Algorithm 3 Online singular subspace computation.

Input: Vectors $a_1, \ldots, a_m \in \Re^n$ where $\|a_t\| \leq 1$, rank parameter: k, regularization parameter: ϵ, δ.
Output: k-dimensional subspaces $\widehat{V}_1, \ldots, \widehat{V}_m$.

1: Choose an arbitrary rank k subspace $\widehat{V}_1$.
2: **for** $t \leftarrow 1$ to m **do**
3: Get a *reward* $R_t = \|\widehat{V}_t^T a_t\|_2^2 = \mathsf{tr}(a_t^T \widehat{V}_t \widehat{V}_t^T a_t)$ and receive input a_t.
4: Compute $C_t = \sum_{\tau=1}^{t} a_\tau a_\tau^T$
5: Compute $\widehat{C}_t = C_t + E_t$, where E_t is sampled as in Algorithm 1 using the parameters ϵ, δ.
6: Compute $\widehat{V}_{t+1}$ as the top k singular subspace of $\widehat{C}_t$.
7: **end for**

Write $R = \sum_{t=1}^m R_t = \sum_{t=1}^m \|\widehat{V}_t^T a_t\|_2^2$. We define $R' = \sum_{t=1}^m \|\widehat{V}_{t+1}^T a_t\|_2^2$ as the reward for the *be the perturbed leader algorithm* (BPL). Notice that in the definition of R', we project a_t to $\widehat{V}_{t+1}$, hence the name of BPL. The proof of Theorem 14 consists of two steps.

Lemma 15 (FPL is close to BPL)**.** $\mathbb{E}[R'] \leq e^{2\epsilon}\mathbb{E}[R] + O(\delta m/\epsilon)$.

Lemma 16 (BPL has low regret)**.** $\mathsf{OPT} \leq \mathbb{E}[R'] + O(k\sqrt{n}\Delta_{\epsilon,\delta})$.

From the above two lemmas we have that

$$\mathbb{E}[\mathsf{Regret}] = \mathsf{OPT} - \mathbb{E}[R] \leq \mathsf{OPT} - e^{-2\epsilon}\mathbb{E}[R'] + O(\delta m/\epsilon) = O(\epsilon \mathsf{OPT} + k\sqrt{n}\Delta_{\epsilon,\delta} + \delta m/\epsilon)\,.$$

Recall that $\Delta_{\epsilon,\delta} = \frac{1+\sqrt{2\log(1/\delta)}}{\epsilon}$, and by $\epsilon < 1$ and $\delta = 1/m$, Theorem 14 follows. We now prove Lemma 15 and 16.

Proof. (Lemma 15) By Theorem 2, $\widehat{C}_t$ is (ϵ, δ) private with respect to any change aa^T ($\|a\|_2 \leq 1$) to C_t. Now define

$$S = \{V \,:\, \mathsf{Prob}[\widehat{V}_{t+1} = V] \geq e^{2\epsilon}\mathsf{Prob}[\widehat{V}_t = V]\}\,.$$

Since $C_{t+1} = C_t + a_{t+1}a_{t+1}^T$, by the property of (ϵ, δ)-differential privacy [27], $\mathsf{Prob}[\widehat{V}_{t+1} \in S] = O(\delta/\epsilon)$. Since for any projection V, $\|V^T a_t\|_2 \leq 1$, $\mathbb{E}[R'] \leq e^{2\epsilon}\mathbb{E}[R] + O(\delta m/\epsilon)$, where the first term accounts for the case of $V \notin S$, and second for $V \in S$. □

Proof. (Lemma 16) Observe that $\|V^T a\|_2 = \mathsf{tr}(VV^T aa^T)$ is a linear function in VV^T and aa^T. In addition, all the E_t's are drawn from the same distribution, denoted by $\mathcal{E}$, according to Algorithm 1. We can apply the same argument in [25] and arrive at:

$$\mathbb{E}[R'] \geq \mathsf{OPT} - \mathbb{E}_{E\sim\mathcal{E}}[\max_{V_1,V_2\in\mathbb{P}_k} \mathsf{tr}((V_1 - V_2)E)]\,. \tag{15}$$

In the above, we used the observation that in expectation the following two strategies in Line 5 (of Algorithm 3) are equivalent: i) sample the noise matrix E_1 once and set all the rest $E_t = E_1$, and ii) sample E_t's i.i.d. Furthermore,

$$\mathbb{E}_{E\sim\mathcal{E}}[\max_{V_1,V_2\in\mathbb{P}_k} \mathsf{tr}((V_1 - V_2)E)] \leq 2k\mathbb{E}[\max_{E\sim\mathcal{E}} \|E\|_2] = O(k\sqrt{n}\Delta_{\epsilon,\delta})\,. \tag{16}$$

The last step in (16) follows from $\|E\|_2 = O(\sqrt{n}\Delta_{\epsilon,\delta})$ with high probability. Plugging (16) in (15), we arrive at our conclusion. □

By setting $\epsilon = \sqrt{\frac{k\sqrt{n}\log m}{\mathsf{OPT}}}$ and $\delta = 1/m$, we have the following regret bound.

Corollary 17. *Algorithm 3 has regret* $O\left(\sqrt{k\mathsf{OPT}}n^{1/4}\sqrt{\log m}\right)$.

The improved regret bound of Algorithm 3 is due to that we allow a relaxed (ϵ, δ) privacy on $\widehat{C}_t$. If one insists on pure ϵ privacy, by sampling from Laplace distribution for example, then per [26], the best possible gap would be $O(n)$, i.e. (16) would have a factor of n, instead of $\sqrt{n}$. This would translate to $\sqrt{n}$ factor in the final regret bound.

4.2 Private online singular subspace computation

While each step in Algorithm 3 is (ϵ, δ) private, overall it is not. One can of course make each step $O(\epsilon/\sqrt{m}, \delta')$ private and apply composition theorem to obtain an (ϵ, δ) private algorithm. But this would induce a large factor in the regret bound. We now apply the tree-based aggregation scheme (Section 2.3.2) to generate the noise to obtain an (ϵ, δ)-private algorithm which is only $O(\log^{O(1)} m)$ factor worse. This is by observing it sufficient to obtain a private mechanism for $C_t = \sum_{\tau=1}^{t} a_\tau a_\tau^T$, for $t = 1, \cdots, m$. By the tree-based aggregation scheme, we can obtain, privately, $\widehat{C}_t = C_t + E'_t$ where each entry of E'_t is i.i.d. with variance $\log^3(m/\delta)\Delta_{\epsilon,\delta}{}^2$. Plugging it into Algorithm 3, we obtain the bound claimed in Theorem 14.

4.3 Choice of Privacy Parameter ϵ for Optimal Regret

In the above, we assume OPT is known so we can tune ϵ according to OPT to obtain the optimal regret bound. When OPT is not known, by that $\mathsf{OPT} \leq m$, we can obtain the bound by replacing OPT by m in the above bounds. Or we can apply the standard *doubling trick* (see Section 2.3.1. in [37]) to obtain a bound with an extra $\log m$ factor. We provide the details in Algorithm 4.

Let (ϵ, δ) be the be the required privacy parameters. The main idea in using the doubling trick is as follows: First, start with an initial guess about OPT, denote it by $\mathsf{OPT}_{\mathsf{approx}}$. Choose $\epsilon_0 = \sqrt{\frac{k\sqrt{n}\log^2(m/\delta)}{\mathsf{OPT}_{\mathsf{approx}}}}$ while making sure that the initial guess $\mathsf{OPT}_{\mathsf{approx}}$ is such that $\epsilon_0 \leq \epsilon$ is satisfied. Second, run the iterative steps of the FPL algorithm until the sum of the squares of the top k singular values of the data matrix so far (call it X) exceeds $\mathsf{OPT}_{\mathsf{approx}}$. Since the algorithm only has differentially private access to the data matrix, the condition is set to $X > \mathsf{OPT}_{\mathsf{approx}} - \frac{10k\sqrt{n}\log^2(m/\delta)}{\epsilon}$. If the condition is satisfied, then increase $\mathsf{OPT}_{\mathsf{approx}}$ by a factor of two and restart the execution.

Theorem 18 (Regret guarantee)**.** *If* $\delta < 1/m^2$, $\epsilon < 0.1$, $m = O(\mathsf{poly}\, n)$ *and* $\mathsf{OPT} > \frac{k\sqrt{n}\log^2(m/\delta)}{\epsilon^2}$, *then the regret guarantee for Algorithm 4 is the following.*

$$\mathbb{E}[\mathsf{Regret}] = O\left(\sqrt{k\mathsf{OPT}\sqrt{n}\log^5(m/\delta)}\right).$$

Proof. Let $\pi_1, \pi_2, \cdots$ be the time epochs at which Line 10 (of Algorithm 4) gets executed. Let Z_{π_i} be the total variance captured by the top k right singular values for $(a_{\pi_i+1}, \cdots, a_{\pi_{i+1}})$. First notice that if $m = O(\mathsf{poly}\, n)$, then with probability at least $1 - \mathrm{negl}(m)$, at any time step t, the value of $|X - Z_{\pi_i}|$ in Line 8 is at most $\frac{10k\sqrt{n}\log^2(m/\delta)}{\epsilon}$. This bound follows from Theorem 4. Therefore, within the time window $\pi_i + 1$ and π_{i+1}, $Z_{\pi_{i-1}} \leq \mathsf{OPT}_{\mathsf{approx}} \leq Z_{\pi_{i-1}} + \frac{10k\sqrt{n}\log^2(m/\delta)}{\epsilon}$. Moreover, there are most $\lceil \log \mathsf{OPT} \rceil \leq \lceil \log m \rceil$ executions of Line 10.

Algorithm 4 Private online singular vector computation without variance bound

Input: matrix: $A \in \Re^{m \times n}$ (available online with rows $a_1 \in \Re^n \cdots a_m \in \Re^n$) , rank parameter: k, privacy parameters: $\epsilon, \delta > 0$.

1: $\mathsf{OPT}_{\mathsf{approx}} \leftarrow \frac{k\sqrt{n}\log^2(m/\delta)}{\epsilon^2}$.

2: Choose an arbitrary rank k subspace V_1. Initialize the tree based aggregation in Section 2.3.2 with privacy parameters (ϵ_0, δ), where $\epsilon_0 = \sqrt{\frac{k\sqrt{n}\log^2(m/\delta)}{\mathsf{OPT}_{\mathsf{approx}}}}$.

3: **repeat**

4: Get a *reward* of $\|V_t^T a_t\|_2^2 = \mathsf{tr}(V_t^T a_t a_t^T V_t)$ and receive input a_t.

5: Compute $Q_t = \sum_{\tau=1}^{t} a_\tau a_\tau^T$ via tree based aggregation in Section 2.3.2.

6: $\alpha \leftarrow \#$ of ones in the binary representation of t. $Z_t \leftarrow Q_t + \sum_{j=1}^{\lceil \log m \rceil - \alpha} E_j$, where E_j is a symmetric $n \times n$ matrix with each entry drawn i.i.d. from $\mathcal{N}\left(0, \frac{50\log^3(m/\delta)}{\epsilon_0^2}\right)$.

7: $V_{t+1} \leftarrow$ Top k right singular vectors of Z_t.

8: $X \leftarrow$ sum of top k eigenvalues of Q_t.

9: **until** $X > \mathsf{OPT}_{\mathsf{approx}} - \frac{10k\sqrt{n}\log^2(m/\delta)}{\epsilon}$

10: $\mathsf{OPT}_{\mathsf{approx}} \leftarrow \mathsf{OPT}_{\mathsf{approx}} \times 2$. Re-initialize time counter $t \leftarrow 1$. Execute from Line 2 onwards.

By Theorem 14, and the choice of privacy parameter in Line 2 in each of these time windows $[\pi_i + 1, \pi_{i+1}]$, the regret for the sequence $(a_{\pi_i+1}, \cdots, a_{\pi_{i+1}})$ is $O\left(\sqrt{k\mathsf{OPT}\sqrt{n}\log^2(m/\delta)}\right)$. Since Line 10 (of Algorithm 4) executes $\log m$ times, therefore, the final regret is $O\left(\sqrt{k\mathsf{OPT}\sqrt{n}\log^5(m/\delta)}\right)$. □

5 Lower Bounds

Bun, Ullman and Vadhan [6] recently showed that the existence of fingerprinting codes can be used to prove lower bounds on the error of (ϵ, δ)-differentially private mechanisms. We next show that using some of their tools, with some extra effort, one can derive a lower bound for private subspace estimation that nearly matches our upper bounds.

Fingerprinting codes were introduced by Boneh and Shaw [5] for watermarking. Informally, a fingerprinting code is a (distribution over) collection of codewords, one to each agent which has the property that no coalition of agents with access only to its own codewords will be able to produce a valid-looking codeword without at least one coalition member being identified. Formally, we have a pair of (randomized) algorithms Gen and $Trace$. Gen outputs a codebook C consisting of t vector $c_1, \ldots c_t \in \{-1, 1\}^n$ with c_i representing the codeword given to user i. Given a subset $S \subseteq [t]$ of agents, let $c_S \in \{-1, 0, 1\}^n$ be defined as

$$c_{Sj} = \begin{cases} +1 & \text{if } c_{ij} = +1 \forall i \in S \\ -1 & \text{if } c_{ij} = -1 \forall i \in S \\ 0 & \text{otherwise} \end{cases}$$

Let $F_+(S) = \{j \in [n] : c_{Sj} = 1\}$ and similarly $F_-(S) = \{j \in [n] : c_{Sj} = -1\}$. Let $F(S) = F_+(S) \cup F_-(S)$ denote the set of unanimous coordinates in S where all codewords in S agree. We say that a vector $c' \in \{-1, 1\}^n$ is β-valid for S if c'_j agrees with c_{Sj} in at least a $(1 - \beta)|F(S)|$ of the locations in $F(S)$. In other words, $\Pr_{j \sim F(S)}[c'_j = c_{Sj}] \geq 1 - \beta$. (Robust) Fingerprinting codes have the property that

given a c' that is β-valid for a coalition S, the tracing algorithm $Trace$ outputs a member of the coalition with high probability. We use the following definition (essentially) from [6].

Definition 19 (Weakly Robust Fingerprinting Codes)**.** *Let t, n, f be integers and let $\xi, \beta \in [0,1]$. A pair of algorithms (Gen,$Trace$) is a (t, n, f, β, ξ)-fingerprinting code if Gen outputs a codebook $C = \{c_1, \ldots, c_t\} \subseteq \{-1,1\}^n$ and for every possible (possibly randomized) adversary A_{pirate}, and for every coalition $S \subseteq [t]$,*

1. $\Pr[Trace(C, c') \in S \mid c' \text{ is } \beta\text{-valid for } S] \geq 1 - \xi$.
2. $\Pr[Trace(C, c') \in [t] \setminus S] \leq \xi$.
3. $\Pr[|F(S)| \geq f] \geq 1 - \xi$.

where $c' = A_{pirate}(c_i : i \in S)$ and the probability is taken over the coins of Gen,$Trace$ and A_{pirate}.

Bun et al. [6] show that the fingerprinting codes construction of Tardos [43] is weakly robust.

Theorem 20. *For every $n \in \mathbb{N}$ and $\xi \in [0,1]$, the construction of [43] gives an $(t, n, f, \frac{1}{20}, \xi)$ fingerprinting code such that*

$$t = \Omega(\sqrt{n/\log(n/\xi)}) \qquad f = \Omega(t^{\frac{3}{2}})$$

Finally, the following theorem, essentially from [6] shows how fingerprinting codes lead to lower bounds for differentially private mechanisms (by setting $\xi, \delta = O(1/n^2)$).

Theorem 21. *Let $\mathcal{M} : D^m \to D$ be an (ϵ, δ)-DP mechanism with $D = \{-1,1\}^n$. If $(m+1, n, f, \beta, \xi)$-weakly robust fingerprinting codes exist with security $\xi \leq \frac{1}{2}$, then*

$$\Pr[\mathcal{M}(C|_S) \text{ is } \beta\text{-valid for } S] \leq m(2\xi \exp(\epsilon) + \delta).$$

Proof. Let $\mathcal{M}'(C|_S) = Trace(C, \mathcal{M}(C|_S))$, and let p denote $\Pr[\mathcal{M}(C|_S)$ is β-valid for S]. Then by the first property of fingerprinting codes, $\Pr[\mathcal{M}'(C|_{[m]}) \in [m]] \geq p(1-\xi)$. Thus there exists an $i \in [m]$ such that $\Pr[\mathcal{M}'(C|_{[m]}) = i] \geq \frac{p(1-\xi)}{m}$.

Let $S' = [m+1] \setminus \{i\}$. Then by the second property of fingerprinting codes, $\Pr[\mathcal{M}'(C|_{S'}) = i] \leq \xi]$. Since $\mathcal{M}'$ satisfies (ϵ, δ)-DP, it follows that

$$\frac{p(1-\xi)}{m} \leq \exp(\epsilon)\xi + \delta.$$

Rearranging gives the result. □

Because Differential Privacy is closed under post-processing, this says that a differentially private mechanism $\mathcal{M}$ cannot even produce a vector in $\Re^n$ whose sign agree with $C|_S$ in a $(1-\beta)$ fraction of the locations in $F(S)$ (or else we could round this vector and contradict the theorem).

5.1 Lower bound for eigenvector computation

We say a unit vector v is an α-useful eigenvector for a matrix A if $\|Av\|_2^2 \geq \|Av'\|_2^2 - \alpha$ for every unit vector v'. The main result of this section says that no differentially private mechanism can output a v that is $o(m)$-useful on any $m \times n$ matrix, if (m, n, f, β, ξ)-fingerprinting codes exist for appropriate f, β, ξ. At a high level, we construct a hard matrix by taking a fingerprinting codes matrix, padding it with many 1s, and suitably scaling to make rows norm 1. For the top eigenvector v_1 of this matrix, either v_1 or $-v_1$ must

agree with $C|_S$ in sign on all the consensus locations, and we can use the padding bits to pick between v_1 and $-v_1$. Lemma 23 is a robust version of this statement. The padding also ensures a large gap between the first and the second eigenvalue (Lemma 24), so that any $o(m)$-good vector must be very close to v_1. Thus we can use any $o(m)$-good vector to construct a β-valid vector for appropriate β. We next give the details.

Theorem 22. *There is a universal constant K such that the following holds. Suppose there is an (ϵ, δ)-DP mechanism that for any matrix $A \in \Re^{m \times 16n}$ with each row having norm at most 1 outputs an γm-useful eigenvector of A with probability p. Then there is an (ϵ, δ)-DP mechanism that on input $S = \{c_1, \ldots, c_m\}$ from a $(m+1, n, f, \beta_0, \xi)$-fingerprinting code outputs a c' that is $K\gamma$-valid for S with probability $p - \xi - \exp(-\Omega(\gamma^2 f))$.*

Algorithm 5 Pirate algorithm A_{pirate}

Input: Set of codewords $S = \{c_1, \ldots, c_m\}$ with $c_i \in \{-1, 1\}^n$. Oracle access to Mechanism $\mathcal{M}$ for privately computing top right singular vector.

Let $pad \leftarrow 1^{15n}$.
for $i = 1, \ldots, m$ **do**
  Let $c_i^{(1)} \leftarrow c_i \circ pad$.
  Let $c_i^{(2)} \leftarrow c_i^{(1)} / \sqrt{16n}$.
end for
Let P be a random permutation matrix. Replace each 1 in P by a -1 with probability $\frac{1}{2}$.
Let A be the $m \times n$ matrix with the transposes of $c_i^{(2)}$'s as its rows.
Let $A' \leftarrow AP$.
Let $v \leftarrow \mathcal{M}(A')$ be the γm-useful right singular vector output by $\mathcal{M}$.
Let $w \leftarrow Pv$.
if $\sum_{j=n+1}^{15n} w_j \leq 0$ **then**
  $w \leftarrow -w$.
end if
for $j = 1 \ldots n$ **do**
  $c'_j = sgn(w_j)$.
end for
return c'

Proof. Let $\mathcal{M}$ be a differentially private mechanism that outputs a γm-useful eigenvector for any input matrix A. We will use it as a subroutine to construct a differentially private mechanism $\mathcal{M}'$ that outputs a β-valid codeword for an appropriate β.

The mechanism $\mathcal{M}'$ works as follows. Let $c_1, \ldots, c_m$ be the input vectors to $\mathcal{M}'$. We first set pad to the vector 1^{15n} append it to each of the c_i's to get $c_i^{(1)} \in \Re^{16n}$. We then scale each $c_i^{(1)}$ to get a unit vector, by setting $c_i^{(2)} = c_i^{(1)} / \sqrt{16n}$. Let A be the matrix with rows (transpose of) $c_i^{(2)}$. Finally, we pick a random permutation matrix P and replace each 1 by -1 with probability $\frac{1}{2}$. We set $A' = AP$. Thus A' is obtained by randomly permuting the columns of A and randomizing the sign of each column. We run the mechanism $\mathcal{M}$ on A', to get a γm-useful vector v.

We then postprocess v as follows: we undo the signed permutation P and without loss of generality, assume that sum of entries of Pv on the pad locations is non-negative (if not, replace v by $-v$). We then strip off the padding and set $c'_j = sgn((Pv)_j)$ for each $j \in [n]$. This defines the output of $\mathcal{M}'$. The privacy of $\mathcal{M}'$ follows immediately from the post-processing property of differential privacy and the fact that pad and P did not depend on the data c_i's. We next argue that, conditioned on v being γ-useful, c' is β-valid.

We first establish two useful properties of the eigen-decomposition of A'. The permutation P does not change the eigen-spectrum so it suffices to prove the results for A. Let $\widehat{F}$ denote the unanimous locations in $c_i^{(1)}$ (i.e. the non-zero locations in c_S along with the padding bits). Slightly abusing notation, we extend c_S to be a vector in $\{-1,0,1\}^{16n}$ with $c_{Sj}=1$ for $j \geq n$ as all $c_i^{(1)}$'s have a 1 in the padding locations. The first lemma says the the top eigenvector must agree with c_S in sign on $\widehat{F}$, and moreover must be non-negligible on these coordinates.

Lemma 23. *Let v_1 be the top right singular vector of A such that $\sum_{j=n+1}^{16n} v_{1j} \geq 0$. Then for any $j \in \widehat{F}$, $sgn(v_{1j}) = c_{Sj}$ and $|v_{1j}| \geq \frac{1}{40\sqrt{n}}$.*

Proof. Let $a_i = c_i^{(2)} \in \Re^{16n}$. Since $\sum_i \langle a_i, v_1\rangle^2 \geq \frac{15m}{16}$, it follows that at least for one i, it is the case that $\langle a_i, v_1\rangle^2 \geq \frac{15}{16}$. Since $a_i|_{[n]}$ has norm $\frac{1}{4}$, it follows that the contribution to the dot product from the pad bits is at least $\frac{\sqrt{15}-1}{4}$. This in turn implies that for all i, $\langle a_i, v_1\rangle \geq \frac{\sqrt{15}-2}{4} \geq \frac{1}{4}$.

Let $j \in \widehat{F}$ with $c_{Sj}=1$ and suppose that $v_{1j} \leq \frac{1}{40\sqrt{n}}$. Let $e_j \in \Re^{16n}$ be a vector with one only in the j-th coordinate. We will argue that if v_1 is nearly orthogonal to e_j, then rotating v_1 slightly in the e_j direction gives a better Raleigh quotient, contradicting the optimaility of v_1. Indeed let $e'_j = e_j/100\sqrt{n}$. Thus $\langle e'_j, v_1\rangle \leq \frac{1}{4000n}$, which implies that $\|v_1+e'_j\|_2^2 \leq \|v_1\|_2^2 + \|e'_j\|_2^2 + 2\langle e'_j, v_1\rangle \leq 1 + \frac{1}{10000n} + \frac{2}{4000n} \leq 1 + \frac{6}{10000n}$. On the other hand, $\langle a_i, e'_j\rangle \geq \frac{1}{400n}$ for each i, so that $(\langle a_i, (v_1+e'_j)\rangle^2 - \langle a_i, v_1\rangle^2) \geq \frac{1}{400n}\cdot\frac{1}{4} \geq \frac{1}{1600n}$. In other words $\|A(v_1+e'_j)\|_2^2 \geq \|Av_1\|_2^2(1+\frac{1}{1600n})$, contradicting the optimality of v_1. The case of $c_{sj}=-1$ is identical. □

Lemma 24. *For the matrix A as defined, $\sigma_1^2 \geq \frac{15m}{16}$. Thus $\sigma_1^2 - \sigma_2^2 \geq \frac{7m}{8}$.*

Proof. The vector v_{pad} that is zero of the first n coordinates, and equals $pad/\sqrt{16n}$ on the remaining coordinates has norm less than 1 and gives $\|Av_{pad}\|_2^2 = \frac{15nm}{16n}$. This implies the first part of the lemma. The second part follows from noting that the sum of all σ_i^2 is m. □

Let v be a γ-useful vector output by the Algorithm $\mathcal{M}$ and let $w = Pv$. Let v_1 be the top right singular vector of A. From Lemma 24, it follows that, $\langle w, v_1\rangle^2 \geq (1-4\gamma/3)$ so that $\|w-v_1\|_2^2 \leq 8\gamma/3$. By Lemma 23, every coordinate in $\widehat{F}$ such that $sgn(w)_j$ is different from c_{Sj} contributes $\frac{1}{1600n}$ to the squared distance $\|w-v_1\|^2$. It follows that the sign is wrong on at most $(1600n)(8\gamma/3) = 12800\gamma n/3$ of the $\widehat{F} \geq 15n$ coordinates.

The permutation P being random and unknown to the mechanism $\mathcal{M}$ ensures that the fraction of mistakes on F is not too different from that on $\widehat{F}$. Formally, call a co-ordinate in $\widehat{F}$ bad if $sgn(w)_j \neq c_{Sj}$. Recall that P randomizes both the location and the sign of the bits in c_{Sj}. Thus from the point of view of $\mathcal{M}$, F is a random subset of $\widehat{F}$ of size $|F|$. Thus the number of bad co-ordinates in F is expected to be at most $(\frac{12800\gamma n}{3})(|F|/15n)$. Except with probability ξ, $|F| \geq f$. Moreover by concentration bounds for the hypergeometric distribution, the probability that the number of bad coordinates in F exceeds twice its expectation is at most $\exp(-\frac{1}{2}(\frac{12800\gamma}{45})^2 f)$. The claim follows. □

Combining with Theorems 20 and 21, we get

Corollary 25. *There is a universal constant γ such that the following holds for $m = \gamma\sqrt{n/\log n}$. Let $\mathcal{M}$ be a $(1, 1/n^2)$-DP mechanism that takes as input an $m \times 16n$ matrix A with each row having norm at most 1, and outputs a unit vector v. Then the probability that $\mathcal{M}(A)$ is γm-useful is at most $\frac{1}{n}$.*

Proof. Let $\mathcal{M}$ be an (ϵ,δ)-DP mechanism that on input an $m \times n$ matrix A outputs an approximate right singular vector. Applying Theorem 20 with $\xi = \frac{1}{n^3}$, we get fingerprinting codes with $m = t-1 = c\sqrt{n/\log n}$ and $f \geq \sqrt{n}$. Let p be the probabilty that $\mathcal{M}$ outputs a $\gamma_1\sqrt{m}$-useful vector v for $\gamma_1 = 1/20K$

where K is the constant in Theorem 22. Applying Theorem 22 on $\mathcal{M}$, we get a private mechanism that outputs a $\frac{1}{20}$ valid c' with probability $p' \geq p - \xi - \exp(-\Omega(f^2))$. Plugging into Theorem 21, and choosing γ appropriately, the theorem follows. □

5.2 Interlude: The List Culling Game

To help understand the proof for the lower bound for the subspace estimation, we introduce the *List Culling Game*. In this game, Dave has a vector $v \in \{-1, 1\}^n$. Alice has a version $v' \in \{-1, 1, \star\}^n$ of v where f of the bits chosen at random have been replaced by $\star$; we will be interested in the setting where f is $o(n)$. Dave, without knowing which bits are erased, sends Alice a list $L = \{w_1, \ldots, w_{|L|}\}$ of $\{-1, 1\}^n$ vectors with the promise that at least one of the w_i's has Hamming distance at most βn from v for a small constant $\beta < 1/20$. Alice wins if she can fill in the $\star$'s with error rate smaller than $\frac{1}{3}$, else Dave wins. Clearly if L is allowed to be size 2^n, then Dave can send the list of all binary vector, thus leaking no information and making it very unlikely that Alice can win. We will be interested in the question: For what values of L can Alice win?

The most natural strategy for Alice is the *most-agreement-strategy*: find a w_i that has the largest agreement on the non-$\star$ locations of v' and fill in the $\star$'s using it. We next argue that this strategy fails for lists size $\binom{n}{f}$. Indeed consider the list containing all vectors at Hamming distance exactly f from v. This list contains the vector w that agrees with v' on all non-$\star$ locations, and hence will be the one picked by the most-agreement-strategy. However, this vector w is wrong everywhere on the $\star$ locations!

This most-agreement-strategy for Alice thus fails badly once $L \geq \binom{n}{f}$. One may conjecture that beyond this threshold, Alice cannot win and instead Dave has a strategy that wins with non-negligible probability. We show that this conjecture is false: there is a strategy for Alice that wins with high probability even when the list size L is $\exp(cn)$ for some constant c.

The somewhat counter-intuitive strategy for Alice is as follows: she picks a random half of the non-$\star$ locations and finds a w_i that maximizes the agreement on this subset. This *most-agreement-on-random-half* strategy thus uses only half the information that Alice has about v! Consider a specific w_i that has Hamming distance more than $2\beta n$ from v, and let w^* be the promised vector in L that has Hamming distance at most βn from v. Alice tests w_i and w^* on a random $\frac{n-f}{2}$ subset, and the probability that w_i has larger agreement than w^* on this random subset is at most $\exp(-\Omega(\beta^2 n))$. Thus the probability that any w_i with Hamming distance larger than $2\beta n$ is chosen by the most-agreement-on-random-half strategy is $L \exp(-\Omega(\beta^2 n))$. Finally, if the chosen w_i has Hamming distance less than $2\beta n$ from v, the probability that it has disagreement more than $5\beta f$ on the $\star$ locations is at most $\exp(-\Omega(\beta^2 f))$. Thus for list size up to $\exp(cn)$ for a constant c, Alice wins with high probability. We have thus argued that

Theorem 26. *There is an absolute constant $\gamma > 0$ such that for any f and large enough n, the following holds. There is a strategy for Alice in the list culling game such that for any valid list L of size $\exp(\gamma n)$, Alice wins with probability at least $1 - \exp(-\gamma f)$.*

5.3 Lower bound for subspace estimation

We say a k-dimensional projection matrix Π_k v is an α-useful rank-k subspace for a matrix A if $\|\Pi_k A^T\|_F^2 \geq \|\Pi_k' A^T\|_F^2 - \alpha$ for any rank-k projection matrix Π_k'. The main result of this section is analagous to the result for private eigenvectors. To get this result, we combine k of the $m \times 16n$ matrices from the previous section into one $km \times 16n$ matrix. When k is small (at most n/m) we can rotate these k matrices so that their spans are all orthogonal and they do not interfere with each other and the "loss" of about m from each of them results in a total loss of km. For larger k, some interference is unavoidable, but rotating them in random

directions suffices to make them nearly orthogonal; this is the content of Lemma 30. Additionally, the eigenvalue separation result of the previous section is not sufficient any more as we output a k-dimensional subspace instead of a vector. We end up needing tighter control on the second (and thus smalller) eigenvalue of A, which we obtain in Lemma 28 by using the specific construction of Tardos and results from random matrix theory. A bigger difficulty comes from the fact that the output is now a k-dimensional subspace rather than a vector, and we need to extract a vector in this subspace that we will round to a β-valid vector for S. In the vector case, we used the padding bits to pick between w and $-w$; now we use them to pick amongst an $\exp(O(k))$-sized net of the subspace. This is where the List Culling Game is useful: the usefuleness of the subspace guarantees that one of these net points, appropriately rounded is β-valid. Using half of the padding bits to pick out the correct one allows us to complete the proof. Full details follow.

Theorem 27. *There are universal constants K, K' such that the following holds for any $k \leq n/K$. Suppose there is an (ϵ, δ)-DP mechanism that for any matrix $A \in \Re^{m \times 16n}$ with with each row having norm at most 1 outputs a γkm-useful rank-k projection matrix Π_k A with probability p. Then there is an (ϵ, δ)-DP mechanism that on a sample $S = \{c_1, \ldots, c_m\}$ from an $(m+1, n, f, \beta_0, \xi)$-fingerprinting code outputs a c' that is $K\gamma$-valid for S with probability $K'\gamma p - \xi - \exp(-\Omega(\gamma^2 f))$.*

Algorithm 6 Pirate algorithm A_{pirate}

Input: Set of codewords $S = \{c_1, \ldots, c_m\}$ with $c_i \in \{-1, 1\}^n$. Oracle access to Mechanism $\mathcal{M}$ for privately computing top k subspace of a matrix. Sampling access to distribution $\mathcal{D}$ from which S is sampled.

for $i = 1 \ldots k$ **do**
  Sample $S_i = \{c_{i1}, \ldots, c_{im}\}$ from $\mathcal{D}$.
end for
Pick r uniformly at random from $[k]$ and set $S_r \leftarrow S$.
Let $pad \leftarrow 1^{15n}$.
for $i = 1 \ldots k$ **do**
  for $j = 1 \ldots m$ **do**
    Let $c_{ij}^{(1)} \leftarrow c_{ij} \circ pad$.
    Let $c_{ij}^{(2)} \leftarrow c_{ij}^{(1)} / \sqrt{16n}$.
  end for
  Let $P^{(i)}$ be a random permutation matrix. Replace each 1 in P by a -1 with probability $\frac{1}{2}$.
  Let $A^{(i)}$ be the $m \times n$ matrix with the transposes of $c_i^{(2)}$ as it's rows.
  Let $R^{(i)}$ be a random $n \times n$ rotation matrix.
  Let $B^{(i)} \leftarrow A^{(i)} P^{(i)} R^{(i)}$.
end for
Let B be formed by vertically concatenating $B^{(i)}$'s for $i = 1, \ldots, k$ in random order.
Let $\Pi_k \leftarrow \mathcal{M}(B)$ be the γmk-useful rank-k projection matrix output by $\mathcal{M}$.
Let $\Pi_k^{(r)} = \Pi_k (R^{(r)})^T (P^{(r)})^T$.
Let $\theta \leftarrow \frac{1}{80\sqrt{n}}$.
Let w be the vector in $Span(\Pi_k^r)$ such that $\sum_{j=8n+1}^{16n} \mathbb{1}(|w_j| \geq \theta)$ is maximized.
for $j = 1 \ldots n$ **do**
  $c'_j = sgn(w_j)$.
end for
return c'

Proof. Let $\mathcal{D}$ be the distribution of the fingerprinting code and let $S_1, \ldots, S_{k-1}$ be $k-1$ fresh independent samples from $\mathcal{D}$ and let $S_k = S$. Thus the S_i's are identically and independently distributed. We randomly

permute the indices so that S is indistinguishable from any other sample S_j. We will show a mechanism that outputs a $K\gamma$-valid codeword for S with non-trivial probability.

Towards that goal, we transform each $S_i = \{c_{i1}, \ldots, c_{im}\}$ to a matrix $A^{(i)}$ in a manner similar to the proof of Theorem 22. We first set $pad = 1^{15n}$ and append it to each of the c_{ij}'s to get $c^{(1)}_{ij}$. We then scale each vector to get a unit vector, thus setting $c^{(2)}_{ij} = c^{(1)}_{ij}/\sqrt{16n}$. Let $A^{(i)}$ be the matrix with rows $c^{(2)}_{ij}$. Next, we pick a random permutation matrix $P^{(i)}$ with a random sign on each entry, and a random rotation matrix $R^{(i)}$ and set $B^{(i)} = A^{(i)}P^{(i)}R^{(i)}$. Thus $B^{(i)}$ is obtained by randomly permuting the columns of $A^{(i)}$, randomly flipping the sign of each column, and then randomly rotating the rows of the resulting matrix[5]. Finally, we set B to the $km \times 16n$ matrix formed by vertically concatenating the $B^{(i)}$'s.

Let Π_k be the γkm-useful rank-k projection matrix returned by our private mechanism on input B. We will postprocess Π_k to construct a valid pirate codeword. Let $S = S_r$ and let $\Pi^{(r)}_k = \Pi_k(R^{(r)})^T(P^{(r)})^T$. For a vector v, a parameter θ, and a location j, we say that v θ-agrees with c_S in location j if $c_{Sj}v_j \geq \theta$. Let $H = \{8n+1, \ldots, 16n\}$ be the second half of the indices, all corresponding to padding bits. Let w be a unit vector in $Span(\Pi^{(r)}_k)$ such that w $\frac{1}{80\sqrt{n}}$-agrees with c_S in the maximum number of indices in H. We strip off the padding from w and set $c'_j = sgn(v_j)$. We note that this process did not use S except through the differentially private output Π_k, and hence the mechanism that outputs c' is differentially private. We now argue that c' is $K\gamma$-valid with non-trivial probability, for a suitable constant K.

Let $v^{(i)}_1$ denote the top right singular vector of $B^{(i)}$ and recall from Lemma 24 that $\sigma_1(B^{(i)})^2 = \|B^{(i)}v^{(i)}_1\|^2_2 \geq \frac{15m}{16}$. Thus the projection matrix $\tilde{\Pi}_k$ that projects to the span of $\{v^{(i)}_1\}^k_{i=1}$ satisfies $\sum_i \|\tilde{\Pi}_k(B^{(i)})^T\|^2_F \geq \sum_i \sigma_1(B^{(i)})^2 \geq \frac{15mk}{16}$. Let $loss_i = \sigma_1(B^{(i)})^2 - \|\Pi_k(B^{(i)})^T\|^2_F$. Then the γkm-usefuleness of Π_k implies that $\mathbb{E}_i[loss_i] \leq \gamma m$. Each $loss_i \in [-m/16, m]$ and so it is easy to check, using arguments similar to Markov's inequality, that $\Pr_i[loss_i \geq 4\gamma m] \leq 1 - \gamma$. Indeed if this is not the case then

$$\begin{aligned}
\mathbb{E}_i[loss_i] &= \Pr[loss_i \leq 4\gamma m] \cdot \mathbb{E}[loss_i \mid loss_i \leq 4\gamma m] \\
&\quad + \Pr[loss_i \geq 4\gamma m] \cdot \mathbb{E}[loss_i \mid loss_i \geq 4\gamma m] \\
&\geq \Pr[loss_i \leq 4\gamma m] \cdot (-m/16) + \Pr[loss_i \geq 4\gamma m] \cdot 4\gamma m \\
&= (-m/16) + \Pr[loss_i \geq 4\gamma m] \cdot (m/16 + 4\gamma m) \\
&\geq (-m/16) + (1-\gamma)(m/16 + 4\gamma m) \\
&\geq 2\gamma m\,,
\end{aligned}$$

which contradicts our assumption.

We will in fact need a stronger version of Lemma 24 to bound $\sigma_2(B^{(i)}) = O(1)$. The proof uses the particular construction of fingerprinting codes by Tardos [43] and standard results in random matrix theory. For a proof of Lemma 28, see Appendix A.

Lemma 28. *Let $B^{(i)}$ be constructed as above, starting with an $S_i = \{c_{i1}, \ldots, c_{im}\}$ drawn from the fingerprinting code ensemble of [43]. Then there are universal constants K_1, K_2 such that for all $s \geq C_1$, $\Pr[\sigma_2(B^{(i)})^2 \geq s^2] \leq K_1 \exp(-K_2 sn)$.*

Corollary 29. *There is a universal constant K_1 such that except w.p. $\exp(-\Omega(n))$, all i's satisfy $\sigma^2_2(B^{(i)}) \leq K_1$.*

Let us condition on the event that for all i, $\sigma^2_2(B^{(i)}) \leq K_1$; by Corollary 29, this event happens except with negligible probability. The following lemma says that on average over i, the span of $B^{(i)}$ has a small projection on Π_k; in fact it says that the average projection is small for *any* k-dimensional subspace. The

[5]While this distribution is identical to that obtained by just applying R, it will be convenient in our proof to separate out the randomness in this fashion.

proof uses the fact that the rotations $R^{(i)}$ are random and independent, and standard tail bounds along with a net argument. We defer the proof to Appendix A.

Lemma 30. *For $i = 1, \ldots, k$, let $\{v_{ij}\}_{j=1}^{m}$ be a collection of orthogonal unit vectors in $\Re^n$ and let $R^{(i)}$'s be a independent random rotation matrices. Then for a universal constant K,*

$$\Pr[\exists \Pi_k : \sum_{i=1}^{k}\sum_{j=1}^{m} \|\Pi_k R^{(i)} v_{ij}\|_2^2 \geq Kk(1+(km/n))] \leq \exp(-\Omega(n)).$$

Let $x_i = \|\Pi_k v_1^{(i)}\|_2^2$. Let y_i be the total squared projection of the remaining $(m-1)$ right singular vectors of $B^{(i)}$ onto Π_k, i.e. $y_i = \sum_{j=2}^{m} \|\Pi_k v_j^{(i)}\|^2$. Thus $\|\Pi_k (B^{(i)})^T\|_F^2 \leq \sigma_1^2 x_i + \sigma_2^2 y_i$.

Using lemma 30 with v_{ij}'s being the eigenvectors of $A^{(i)}$, we conclude that $\mathbb{E}_i[x_i + y_i] = K(1+(km/n))$. Thus by Markov's inequality, at least a $(1-\gamma/2)$ fraction of the i's satisfy $x_i + y_i \leq (2K/\gamma)(1+(km/n))$. It follows that for at least a $\gamma/2$ fraction of the i's,

1. $loss_i \leq 4\gamma m$, and
2. $x_i + y_i \leq (2K/\gamma)(1+(km/n))$.

Since $S_r = S$ has the same distribution as every other S_i, it follows that this property holds for r with probability at least $\gamma/2$. Let us condition on this event. For the rest of the proof, we will use $A, B, \sigma_1, \sigma_2, x, y, loss$, etc. to denote $A^{(r)}, B^{(r)}, \sigma_1(B^{(r)}), \sigma_2(B^{(r)}), x_r, y_r, loss_r$, etc. Thus as long as $k \leq \gamma^2 n/16KK_1$, and m is at least some absolute constant,

$$\begin{aligned}
4\gamma m &\leq loss \\
&= \sigma_1^2 - \|\Pi_k B^T\|_F^2 \\
&\geq \sigma_1^2 - \sigma_1^2 x - \sigma_2^2 y \\
&\geq (1-x)(15m/16) - K_1(2K/\gamma)(1+(km/n)) \\
&\geq (1-x)(15m/16) - \gamma m/8.
\end{aligned}$$

It follows that $(1-x) \leq 5\gamma$ and thus there exists a unit vector $\tilde{v}_1 \in Span(\Pi_k)$ such that $\|v_1 - \tilde{v}_1\|_2^2 \leq 10\gamma$.

We call a vector $v \in Span(\Pi_k^{(r)})$ (θ, β)-good for a set of indices I if v θ-agrees with c_S in a $(1-\beta)$-fraction of the indices in I.

Claim 31. *If v (θ, β)-agrees with c_S on I but v' does not $(\theta - \theta', \beta + \beta')$-agree with c_S on I. Then $\|v - v'\|^2 \geq \theta'^2 \beta' |I|$.*

Proof. By definition, there are at least $\beta'|I|$ coordinates in which $|v - v'| \geq \theta'$. Each contributes at least θ'^2 to the squared distance, implying the result. □

Using Lemma 23, it follows that v_1 $(\frac{1}{40\sqrt{n}}, 0)$-agrees with c_{Sj} on H. Let $\beta = 2000^2\gamma$. The vector w found by our mechanism must therefore be $(\frac{1}{80\sqrt{n}}, \beta)$-good for H.

Let $\widehat{F}$ denote the unanimous locations in c_S (i.e. the unanimous locations $F(S)$ in $c_1, \ldots, c_m$ along with the padding bits). Recall that $H = \{8n+1, \ldots, 16n\}$ is the second half of the locations. From the point of view of the algorithm, the locations in H are indistinguishable from those in $\widehat{F} \setminus H$ and this will allow us to use arguments analagous to the list culling game. We know that w is $(\frac{1}{80\sqrt{n}}, \beta)$-good for H and we would like to argue that except with negligible probability, it is $(\frac{1}{320\sqrt{n}}, 5\beta)$-good for $\widehat{F} \setminus H$. Let N be a γ-net of

the set of unit vectors in $Span(\Pi_k^{(r)})$. For any net point that is $(\frac{1}{160\sqrt{n}}, 3\beta)$-bad for $\widehat{F}$, the probability (taken over the randomness in P) that it is $(\frac{1}{160\sqrt{n}}, 2\beta)$-good for H is no larger than $\exp(-\Omega(\beta^2 n))$. Taking a union bound over a $\exp(O(k\log(1/\gamma)))$ points in N, we conclude that except with negligible probability, every $v \in N$ that is $(\frac{1}{160\sqrt{n}}, 2\beta)$-good for H is also $(\frac{1}{160\sqrt{n}}, 4\beta)$-good for $\widehat{F} \setminus H$.

Since w is $(\frac{1}{80\sqrt{n}}, \beta)$-good for H, then its nearest net point w' is $(\frac{1}{160\sqrt{n}}, 2\beta)$-good for H. Thus w' is $(\frac{1}{160\sqrt{n}}, 4\beta)$-good for $\widehat{F} \setminus H$, which in turn implies that w itself is $(\frac{1}{320\sqrt{n}}, 5\beta)$-good for $\widehat{F} \setminus H$.

Finally, since F is itself a random subset of $\widehat{F} \setminus H$, w is $(\frac{1}{320\sqrt{n}}, 6\beta)$-good for F except with probability $\exp(-\Omega(\beta^2 f))$. This then implies that the output of the mechanism is 6β-valid with probability $\Omega(\gamma) - \exp(-\Omega(f)) - \exp(-\Omega(n))$, completing the proof of Theorem 27.[6] □

Corollary 32. *There is a universal constant γ such that the following holds for any $k \leq \gamma n$ and for $m = \gamma\sqrt{n/\log n}$. Let $\mathcal{M}$ be a $(1, 1/n^2)$-DP mechanism that takes as input an $km \times 16n$ matrix A with each row having norm at most 1, and outputs a rank k projection matrix. Then the probability that $\mathcal{M}(A)$ is γkm-useful is at most $\frac{1}{n}$.*

Acknowledgements

We are grateful to many colleagues for the generosity in sharing their insights and time: Moritz Hardt, Prateek Jain, Ravi Kannan, Frank McSherry, Sasho Nikolov, Adam Smith, and Jonathan Ullman. We would also like to thank especially the authors of [6] for sharing their manuscript.

References

[1] Dimitris Achlioptas and Frank Mcsherry. Fast computation of low-rank matrix approximations. *Journal of the ACM (JACM)*, 54, 2007.

[2] Peter N. Belhumeur, João P Hespanha, and David Kriegman. Eigenfaces vs. fisherfaces: Recognition using class specific linear projection. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 19(7):711–720, 1997.

[3] Rajendra Bhatia. *Matrix analysis*. Springer, 1997.

[4] Avrim Blum, Cynthia Dwork, Frank McSherry, and Kobbi Nissim. Practical privacy: the SuLQ framework. In *PODS*, 2005.

[5] Dan Boneh and James Shaw. Collusion-Secure Fingerprinting for Digital Data. *IEEE Transactions on Information Theory*, 44:1897–1905, 1998.

[6] Mark Bun, Jonathan Ullman, and Salil Vadhan. Fingerprinting codes and the price of approximate differential privacy. In *STOC*, 2014.

[7] Emmanuel J Candès and Benjamin Recht. Exact matrix completion via convex optimization. *Foundations of Computational mathematics*, 2009.

[8] Emmanuel J Candès and Terence Tao. The power of convex relaxation: Near-optimal matrix completion. *IEEE Transactions on Information Theory*, 2010.

[9] TH Hubert Chan, Elaine Shi, and Dawn Song. Private and continual release of statistics. In *ICALP*. 2010.

[6] The mechanism can be easily modified so that it either outputs a c', or FAIL and is correct except with negligible probability when it does not output FAIL. This can be done by testing (privately) whether the conditions 1 and 2 above hold approximately for r, so that it outputs FAIL with probability at most $(1 - \gamma)$.

[10] Kamalika Chaudhuri, Anand D. Sarwate, and Kaushik Sinha. Near-optimal differentially private principal components. In *NIPS*, 2012.

[11] Chandler Davis and William Morton Kahan. The rotation of eigenvectors by a perturbation. III. *SIAM Journal on Numerical Analysis*, 1970.

[12] C. Dwork and A. Roth. Algorithmic foundations of differential privacy, 2014. Monograph in preparation.

[13] Cynthia Dwork, Krishnaram Kenthapadi, Frank McSherry, Ilya Mironov, and Moni Naor. Our data, ourselves: Privacy via distributed noise generation. In *EUROCRYPT*, pages 486–503, 2006.

[14] Cynthia Dwork and Jing Lei. Differential privacy and robust statistics. In *Symp. Theory of Computing (STOC)*, pages 371–380, 2009.

[15] Cynthia Dwork, Frank McSherry, Kobbi Nissim, and Adam Smith. Calibrating noise to sensitivity in private data analysis. In *Theory of Cryptography Conference*, pages 265–284. Springer, 2006.

[16] Cynthia Dwork, Moni Naor, Toniann Pitassi, and Guy N Rothblum. Differential privacy under continual observation. In *Proceedings of the 42nd ACM symposium on Theory of computing*, pages 715–724. ACM, 2010.

[17] Cynthia Dwork, Moni Naor, Omer Reingold, Guy Rothblum, and Salil Vadhan. On the complexity of differentially private data release: efficient algorithms and hardness results. In *STOC*, pages 381–390, 2009.

[18] Carl Eckart and Gale Young. The approximation of one matrix by another of lower rank. *Psychometrika*, 1936.

[19] Nathan Halko, Per-Gunnar Martinsson, and Joel A Tropp. Finding structure with randomness: Probabilistic algorithms for constructing approximate matrix decompositions. *SIAM review*, 2011.

[20] Moritz Hardt. Robust subspace iteration, incoherence, and privacy-preserving spectral analysis. *Personal Communication*, 2013.

[21] Moritz Hardt and Aaron Roth. Beating randomized response on incoherent matrices. In *STOC*, 2012.

[22] Moritz Hardt and Aaron Roth. Beyond worst-case analysis in private singular vector computation. In *STOC*, 2013.

[23] Elad Hazan, Satyen Kale, and Manfred K Warmuth. Corrigendum to "learning rotations with little regret" september 7, 2010. 2010.

[24] Elad Hazan, Satyen Kale, and Manfred K Warmuth. On-line variance minimization in o (n2) per trial? In *COLT*, 2010.

[25] Adam Kalai and Santosh Vempala. Efficient algorithms for online decision problems. *Journal of Computer and System Sciences*, 2005.

[26] Michael Kapralov and Kunal Talwar. On differentially private low rank approximation. In *SODA*, 2013.

[27] Shiva Prasad Kasiviswanathan and Adam Smith. A note on differential privacy: Defining resistance to arbitrary side information. *CoRR*, abs/0803.3946, 2008.

[28] Jyrki Kivinen and Manfred K Warmuth. Exponentiated gradient versus gradient descent for linear predictors. *Information and Computation*, 132(1):1–63, 1997.

[29] B. Laurent and P. Massart. Adaptive estimation of a quadratic functional by model selection. *Annals of Statistics*, 28(5):1302–1338, 2000.

[30] Ren-Cang Li. Relative perturbation theory: Ii. eigenspace and singular subspace variations. *SIAM Journal on Matrix Analysis and Applications*, 1998.

[31] F. McSherry and I. Mironov. Differentially private recommender systems: building privacy into the net. In *Symp. Knowledge Discovery and Datamining (KDD)*, pages 627–636. ACM New York, NY, USA, 2009.

[32] Frank McSherry. Spectral methods for data analysis. 2004.

[33] Frank McSherry and Kunal Talwar. Mechanism design via differential privacy. In *FOCS*, pages 94–103. IEEE, 2007.

[34] Mehryar Mohri and Ameet Talwalkar. Can matrix coherence be efficiently and accurately estimated? In *International Conference on Artificial Intelligence and Statistics*, 2011.

[35] Jiazhong Nie, Wojciech Kotlowski, and Manfred K Warmuth. On-line pca with optimal regrets. *arXiv preprint arXiv:1306.3895*, 2013.

[36] Benjamin Recht. A simpler approach to matrix completion. *The Journal of Machine Learning Research*, 2011.

[37] Shai Shalev-Shwartz. Online learning and online convex optimization. *Foundations and Trends® in Machine Learning*, 2011.

[38] Adam Smith and Abhradeep Thakurta. *Personal Communication*, 2013.

[39] Adam Smith and Abhradeep Thakurta. Nearly optimal algorithms for private online learning in full-information and bandit settings. In *NIPS*, 2013.

[40] G.W. Stewart and J. Sun. *Matrix Perturbation Theory*. Academic Press, 1990.

[41] Ameet Talwalkar and Afshin Rostamizadeh. Matrix coherence and the Nystrom method. *arXiv preprint arXiv:1004.2008*, 2010.

[42] Terence Tao. *Topics in random matrix theory*, volume 132. AMS Bookstore, 2012.

[43] Gábor Tardos. Optimal probabilistic fingerprint codes. *J. ACM*, 55(2), 2008.

[44] Manfred K Warmuth and Dima Kuzmin. Randomized online PCA algorithms with regret bounds that are logarithmic in the dimension. *Journal of Machine Learning Research*, 2008.

[45] John Wright, Allen Y Yang, Arvind Ganesh, Shankar S Sastry, and Yi Ma. Robust face recognition via sparse representation. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 31(2):210–227, 2009.

A Missing Proofs from Section 5

Proof of Lemma 28. Since the rotation and the permutation matrix do not change the singular values, it suffices to prove the result for the matrix $A^{(i)}$. The fingerprinting code construction of Tardos has the following form: it samples values q_j's from a carefully chosen distribution, and then each c_{lj} is independently set to either 1 or -1 such that the expectation is q_j. Our padding bits are deterministic, corresponding to the q_j's being 1. Thus $\sqrt{16n}A^{(i)}$ is an $m \times 16n$ matrix where each entry is independently chosen from $-1, 1$ where $E[A^{(i)}_{lj}] = q_j$. Let $q \in \Re^{16n}$ be the vector with entries q_j and Q be the $m \times 16n$ matrix where each row is q^T. Thus $N \stackrel{def}{=} A^{(i)} - Q$ is a matrix each entry of which is an independent random variable with mean zero and variance at most 1. It is easy to see that $\sigma_2(A^{(i)})$ can be upper bounded by $\max_{y \in \Re^{16n}:\langle y,q\rangle=0} \frac{\|Ay\|_2}{\|y\|_2}$. This in turn is upper bounded by the operator norm of N. Standard results in random matrix theory (see e.g. Corollary 2.3.5 in Tao [42]) then imply the claimed bound. □

Proof of Lemma 30. We will in fact argue that with high probability, for every vector v, $\sum_{i=1}^{k}\sum_{j=1}^{m}\langle v, v_{ij}\rangle^2 \leq K(1 + (km/n))$. This then implies the lemma.

Indeed for any fixed v, the variable $X_{ij} = \langle v, v_{ij}\rangle$ is distributed as $N(0, 1/n)$ and since v_{ij}'s are orthogonal, the r.v.'s $\{X_{ij}\}_j$ are independent. Since each rotation is chosen independently, the variables $\{X_{ij}\}_{i,j}$ are all independent Gaussians. Thus n times the sum of their squares is a χ^2 with mk degrees of freedom. Standard tail bounds for χ^2 (see e.g. Lemma 1 on pg 1325 of [29]) then say that for any $K_1 \geq 1$,

$$\Pr[n\sum_i\sum_j X_{ij}^2 > mk + 2K_1 n + \sqrt{2K_1 mkn}] \leq \exp(-K_1 n)$$

Applying a standard net argument (see e.g. Lemma 2.3.2 in [42]), and choosing K appropriately gives the claimed bound. □