Minimax-Optimal Privacy-Preserving Sparse PCA in Distributed Systems

Jason Ge
Princeton University

Zhaoran Wang
Northwestern University

Mengdi Wang
Princeton University

Han Liu
Tencent AI Lab

Abstract

This paper proposes a distributed privacy-preserving sparse PCA (DPS-PCA) algorithm that generates a minimax-optimal sparse PCA estimator under differential privacy constraints. In a distributed optimization framework, data providers can use this algorithm to collaboratively analyze the union of their data sets while limiting the disclosure of their private information. DPS-PCA can recover the leading eigenspace of the population covariance at a geometric convergence rate, and simultaneously achieves the optimal minimax statistical error for high-dimensional data. Our algorithm provides fine-tuned control over the tradeoff between estimation accuracy and privacy preservation. Numerical simulations demonstrate that DPS-PCA significantly outperforms other privacy-preserving PCA methods in terms of estimation accuracy and computational efficiency.

1 Introduction

Principal component analysis (PCA) is one of the most widely used tools for data analysis and dimension reduction. Let $\mathbf{x}_1, \mathbf{x}_2, \cdots, \mathbf{x}_n \in \mathbb{R}^d$ be samples drawn from some underlying distribution with covariance matrix $\Sigma \in \mathbb{R}^{d\times d}$, where d is the dimension and n is the data size. Principal component analysis estimates the leading k ($k \ll d$) eigenvectors of Σ based on the given samples. Projection of the samples into the k-dimensional principal subspace spanned by the eigenvectors provides a low-dimensional representation of the high-dimensional data.

We focus on a distributed setting in which multiple data providers interact with one another. Suppose that the data providers, who may come from different interest groups, aim to calculate the principal components of the *union of their data sets*. For example, the data providers can be healthcare providers holding medical records related to certain types of drugs or procedures, or research institutes with genome-wide association (GWA) genetic study results for certain diseases. Even if the data providers may wish to collaborate on data analysis, they are often unwilling to directly share data with others because inappropriate data sharing may lead to severe privacy breach [22, 32].

In practical contexts where privacy matters, there has been a surging demand for *collaborative principal component analysis* applications, including clinical inference [13] and genome population structure modeling [12]. This motivates us to redesign principal component analysis algorithms, in order to allow data providers to collaboratively analyze the data sets without physically merging them. It is desired that the new algorithm achieves strong privacy guarantees, making it nearly impossible to infer an individual's exact state in the data set from the algorithm's output. We adopt the notion of *differential privacy* introduced by [10], a cryptographically inspired guarantee that is resistant to many forms of privacy attacks (see [14]). Intuitively, differential privacy of an algorithm means that a small change in the input has no significant impact on the output of the algorithm (a rigorous definition to be provided later).

High dimensionality poses another critical challenge to modern data applications such as population genome study and brain voxel estimation. It is desired that the new algorithm produces estimates achieving the optimal or nearly optimal statistical accuracy. However, in the typical high-dimensional setting where $d \gg n$, singular value decomposition fails to work by producing inconsistent estimates of principal components [23, 31]. To overcome this problem, we adopt the sparsity regularization for high dimensional PCA, which has been extensively studied in [1, 3, 5, 7, 8, 23, 24, 27, 29, 31, 36, 37, 40, 41, 43, 45, 49].

The goal of this paper is to provide an algorithmic solution to distributed principal component analysis, which leverages the sparsity of high-dimensional data as well as achieves sufficient privacy preservation. In

Proceedings of the 21^{st} International Conference on Artificial Intelligence and Statistics (AISTATS) 2018, Lanzarote, Spain. PMLR: Volume 84.

what follows, we briefly review some earlier works on privacy-preserving PCA, sparsity regularization and distributed optimization, and discuss their relevance with the current paper.

Related works. One line of existing works is on differentially private algorithms for *model-free PCA*, such as the Sub Linear Queries (SuLQ) method by [4], the Private PCA (PPCA) and modified SuLQ (MOD-SuLQ) method by [6] and Analyze Gauss in [11]. MOD-SuLQ and Analyze Gauss both add noise to the sample covariance matrix before applying the singular value decomposition. PPCA applies the exponential mechanism [30] to perform private selection of the leading eigenspace. However, model-free methods cannot be applied successfully on high dimensional data. It is demonstrated in [6] that the sample size has to be on the same order of dimension $n \approx d$ so that a differentially private algorithm can achieve a targeted level of estimation accuracy. The utility gap in Analyze Gauss [11] (Theorem 3) is on the order of $O(\sqrt{d})$. This is quite discouraging for high dimensional scenarios.

Another line of related work is on the locally differential privacy of PCA. In this setting, each data sample is obscured by some random transformation to ensure privacy. One of the earliest among these works is [48], which proposed to apply a random linear transformation to each sample. Other works such as [9, 42, 44] studied the theoretical tradeoff between the minimax statistical rate and the local privacy level imposed on the initial data. In contrast to these works, we consider the interactive setting in which each data provider releases information by answering a series of statistical queries with designed protocol, as opposed to simply obscuring all the samples once and for all. For this reason, our approach is more task-specific and very different from the locally differential privacy studied earlier.

A third line of related works is on high-dimensional PCA, e.g., [16–18], which proved the dimension-free statistical results under incoherent assumption of the sample covariance matrix. Power method for sparse PCA has been studied in [43], which is shown to achieve the statistical minimax rate in polynomial time. In this paper, we follow their technique of imposing sparsity by adding a thresholding step in the power iteration. A similar technique was also used in the truncated power method in [47]. We also note that distributed versions of PCA have been studied by [25, 28, 34]. The focus there is the tradeoff between communication efficiency and approximation accuracy. In contrast, our current focus is the privacy preservation feature of distributed PCA instead of its communication efficiency. Distributed privacy-preserving analysis has also been studied in works such as [46] and [19]. [19] focused Bayesian inference and is not directly comparable. [46] assumed that a non-distributed version of the privacy-preserving procedure is already available and proposed a multiparty computation protocol. In the context of sparse PCA estimation, [46] is not applicable because even the non-distributed version of privacy-preserving sparse PCA estimation algorithm has not been studied before in the literature.

Our Contributions. In this paper, we aim to study the distributed PCA with a focus on the privacy preservation of high dimensional data. From the algorithmic perspective, we are interested in designing efficient algorithms with fast convergence rate and reasonable complexity. From the theoretical perspective, we are interested in the tradeoff between the estimation accuracy and the level of privacy preservation, especially in the sparse high-dimensional setting. The main contributions of this paper are summarized as follows:

(i) We show that efficient estimation of high dimensional sparse principal subspace can be achieved under privacy preservation constraints.

(ii) We propose a privacy-preserving algorithmic framework which obtains an efficient estimate converging in geometric rate to the minimax-optimal statistical error.

(iii) Numerical simulations show that our method significantly outperforms current state-of-the-art privacy-preserving PCA methods such as MOD-SuLQ, PPCA [6] and Analyze Gauss [11] in terms of estimation accuracy and computational efficiency.

To the best knowledge of the authors, this is the first work on privacy-preserving method for high-dimensional sparse PCA in a distributed optimization framework. This is also the first work showing that the minimax statistical rate of sparse PCA can be achieved under differential privacy constraints in polynomial time.

2 Collaborative PCA in Distributed Systems

Let us consider the distributed computing setting with a number of data providers. The data providers aim to collaborate with one another, in order to compute the principal components of the union of all their local data sets. Each data set may contain sensitive information that can not be released to the public. In this section, we introduce a distributed collaborative framework that enables privacy-preserving PCA, and describe the modeling assumptions for high-dimensional data.

We propose a distributed algorithmic framework for privacy-preserving PCA, as illustrated in Figure 1. Let S_i $(i = 1, 2, \cdots, N)$ be the sample set held by the ith data provider. A central server is used to coordinate the distributed computation process but it is not trusted by the data providers. Each data provider is able to communicate with the central server. This framework

allows an iterative algorithm in which the central server repeatedly updates the global estimates while communicating with all the data providers.

In each iteration of the algorithm, the central server sends the current global estimate $Q^{(t)}$ to all the participating providers, where $Q^{(t)} \in \mathbb{R}^{d\times k}$ is an orthogonal basis matrix of the estimated leading eigenspace of the covariance matrix in the underlying statistical model. Upon receiving the current estimate $Q^{(t)}$, each data provider computes a local intermediate result $H_i(S_i, Q^{(t)}) \in \mathbb{R}^{d\times k}$ which is a $d \times k$ matrix contaminated with *intentional* noise. Then each data provider sends back $H_i(S_i, Q^{(t)})$ to the central server, which later combines all the messages $H_1, H_2, \cdots, H_N$ to update the global estimate.

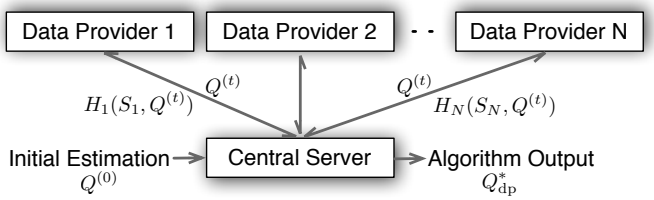


Figure 1: Distributed Private Sparse PCA Algorithm Structure

The proposed framework allows individual data providers to add noise to outgoing messages at their own discretion. This provides significant flexibility to the collaborative analysis. Note that due to the existence of noise, any algorithm under this framework is a randomized algorithm. We use the following notion of differential privacy for randomized algorithms, which has been studied in [26] and [18]. It says that an algorithm is differentially private if the output is insensitive to small input perturbation.

Definition 2.1 (Differential Privacy of Randomized Algorithms)**.** Let $\mathcal{S} = \{\mathbf{x}_1, \ldots, \mathbf{x}_j, \ldots, \mathbf{x}_n\}$ be the set of all samples. A randomized algorithm $\mathcal{M} : \Omega \to \mathcal{R}$ is (ϵ, δ)-differentially private $(\epsilon, \delta > 0)$ if for all pairs of neighboring data sets $\mathcal{S}, \mathcal{S}'$ differing in any individual data item: $\mathcal{S}' = \{\mathbf{x}_1, \ldots, \mathbf{x}'_j, \ldots, \mathbf{x}_n\}$, and for all $R \subseteq \mathcal{R}$, the algorithm satisfies

$$\Pr(\mathcal{M}(S') \in R) \leq e^{\epsilon} \cdot \Pr(\mathcal{M}(S') \in R) + \delta.$$

What privacy guarantee can we promise to the data providers under this definition? From the perspective of the i-th data provider, first of all, the central server cannot determine (to some extent) the presence or absence of any individual data item in the sample set S_i, given the information $H_i(S_i, Q^{(t)})$ $(t = 1, 2, \ldots, T)$ collected during the computation. Secondly, the change of any individual data item in sample set S_i does not have an significant impact on the output of the algorithm Q^*_{dp}. The extent of privacy is determined by $\epsilon > 0$ and $\delta > 0$, with smaller ϵ and δ implying less disclosure of private information related to an individual data item. In order to provide sharp estimates given high-dimensional data, we adopt the following notion of *subspace sparsity*.

Definition 2.2 (Subspace Sparsity)**.** A linear subspace is s-sparse if the number of non-zero diagonal entries of the projection matrix onto the subspace is equal to s.

The notion of subspace sparsity has been introduced in recent literature on sparse PCA such as [40, 41]. Being invariant to rotation, the sparsity level of the subspace spanned by columns of Q^* can be defined as the number of non-zero entries in the diagonal of the projection matrix $\Pi^* = Q^* Q^{*T}$. Note that $\Pi^*_{i,i} = \sum_{j=1}^{k} (Q^*_{i,j})^2$, the sparsity level $s^* = |\mathrm{supp}[\mathrm{diag}(\Pi^*)]|$ is actually the number of non-zero rows of Q^*. And thus we have

$$\begin{aligned} s^* &= |\mathrm{supp}[\mathrm{diag}(\Pi^*)]| = ||Q^*||_{2,0} \\ &= \big\|(||Q^*_{1,*}||_2, ||Q^*_{2,*}||_2, \ldots, ||Q^*_{d,*}||_2)\big\|_0. \end{aligned}$$

Now let us describe the modeling assumptions about the data.

Assumption 2.3. The data samples held by the data providers are independent and identically distributed with a bounded random variable $\boldsymbol{X}$ in a model class $\mathcal{M}(s^*, k)$ for integers $k, s^*,\ 0 < k < s^* < d$. For any $\boldsymbol{X} \in \mathcal{M}(s^*, k)$, the following three assumptions are made.

1. The random variable $\boldsymbol{X} = \Sigma^{1/2}\boldsymbol{Z}$, in which $\boldsymbol{Z} \in \mathbb{R}^d$ is a zero-mean bounded random variable with variance proxy less than one and identity covariance matrix. The matrix $\Sigma \in \mathbb{R}^{d\times d}$ is positive semidefinite.
2. The k-dimensional principal eigenspace of Σ is s^*-sparse.
3. The k-th eigengap is strictly positive, i.e., $\lambda_k - \lambda_{k+1} > 0$ and λ_k is the kth largest eigenvalue.

Let $\widehat{\Sigma}$ be the sample covariance matrix for the union of all the samples $S_1 \cup S_2 \ldots \cup S_N$. Our mathematical problem is to approximate the k-dimensional leading principal subspace estimator under sparsity and differential privacy constraints

$$\begin{aligned} \widehat{Q}^* = &\underset{Q\in\mathbb{R}^{d\times k}}{\mathrm{argmin}} - \mathrm{tr}\left(Q^\top \widehat{\Sigma} Q\right), \\ &\text{subject to } \ Q^\top Q = \mathrm{I}_k\ , ||Q||_{2,0} \leq s^*, \end{aligned} \tag{2.1}$$

and the algorithm has to be differentially private.

According to our framework of distributed algorithms, the overall covariance matrix $\widehat{\Sigma}$ is not known to neither the central server or any data provider. In what follows, we present such an algorithm that iteratively computes an approximate solution to (2.1) based on locally held sample sets $S_1, S_2, \ldots, S_N$.

3 Distributed Privacy-Preserving Sparse PCA Algorithm

In this section, we propose details of the privacay preservation mechanism under the distributed framework of collaborative PCA, see Algorithm 1. The algorithm can be viewed as a noisy truncated power iteration, which is a variant of the classical power iteration for eigenvector computation [15]. Specifically, the matrix-matrix multiplications in the power step are corrupted by i.i.d. Gaussian noise matrices. For the estimated principal subspace basis matrix in each iteration, thresholding procedure is used to set its rows to zeros whose ℓ_2-norm ranking (ranked in decreasing order among all the rows) is larger than $\widehat{s}$. Sparsity constraint is explicitly imposed by thresholding.

Algorithm 1 Distributed Privacy-Preserving Sparse PCA (DPS-PCA)

Input: Sample sets $S_i, i = 1, 2, \ldots, N$ held by the N data providers respectively; Number of samples held by the ith provider is $n_i = |S_i|$; Initialization $Q^{(0)} \in \mathbb{R}^{d\times k}$

Parameters: Sparsity parameter $\widehat{s}$; Maximum number of Iterations T; Differential privacy parameters ϵ, δ with $0 < \epsilon < 1$ and $0 < \delta < 1/2$

Output: $Q^*_{dp} \leftarrow$ DPS-PCA$(\widehat{\Sigma}, Q^{(0)})$

$\widetilde{Q}^{(0)} \leftarrow$ Threshold$(Q^{(0)}, \widehat{s})$
$Q^{(1)}, R \leftarrow$ Thin_QR$(\widetilde{Q}^{(0)})$
Let $\sigma(\epsilon, \delta) \leftarrow 2T\epsilon^{-1}\sqrt{2(2T/\delta)}$
for $i = 1, 2, \ldots, N$ **do**
    $\widehat{\Sigma}_i \leftarrow \sum_{\mathbf{x}_j \in S_i} \mathbf{x}_j \mathbf{x}_j^\top / n_i$
end for
for $t = 1, 2, \ldots, T$ **do**
    Generate entrywise i.i.d. gaussian matrix $G^{(t)} \sim N(0, \sigma^2)^{d\times k}$
    for $i = 1, 2, \ldots, N$ **do**
        $H_i^{(t)} \leftarrow \widehat{\Sigma}_i Q^{(t)} + G^{(t)}/n_i$ ▷ Privacy Preservation Step
    end for
    $K^{(t)} \leftarrow \sum_{i=1}^N n_i H_i^{(t)} / \sum_{i=1}^N n_i$
    $V^{(t)}, R_1 \leftarrow$ Thin_QR$(K^{(t)})$
    $\widetilde{Q}^{(t)} \leftarrow$ Threshold$(V^{(t)}, \widehat{s})$
    $Q^{(t+1)}, R_2 \leftarrow$ Thin_QR$(\widetilde{Q}^{(t)})$
end for
$Q^*_{dp} \leftarrow Q^{(T+1)}$

The Thin_QR step in Algorithm 1 is the thin QR decomposition in [15]. For any matrix $A \in \mathbb{R}^{m\times n}$ with full column rank, the Thin_QR factorization

$$A = QR$$

is unique where $Q \in \mathbb{R}^{m\times n}$ has orthonormal columns and $R \in \mathbb{R}^{n\times n}$ is upper triangular with positive diagonal entries. Effectively, the Thin_QR step orthogonalizes the k basis vectors of the k leading eigenspace in each iteration of the algorithm.

Algorithm 2 Thresholding procedure in Algorithm 1

Input: Matrix $V \in \mathbb{R}^{d\times k}$, Sparsity Parameter $\widehat{s}$

Output: $\widetilde{V} \leftarrow$ Threshold$(V, \widehat{s})$

1: Sort rows in Euclidean norm and let $\mathcal{S}$ to be the set of row indices corresponding to the top $\widehat{s}$ largest rows of V in Euclidean norm
2: $\widetilde{V}$ takes zeros on the rows not indexed by $\mathcal{S}$; The rows of $\widetilde{V}$ indexed by $\mathcal{S}$ take the same value as the entries in V

Algorithm 3 Thin_QR decomposition in Algorithm 1

Input: Matrix $A \in \mathbb{R}^{d\times k}$

Output: $Q, R \leftarrow$ Thin_QR(A), where $Q \in \mathbb{R}^{d\times k}$ has orthonormal columns and upper triangular matrix $R \in \mathbb{R}^{k\times k}$ has positive diagonal entries, and $A = QR$; Here we use the Householder QR method described in [15]

Algorithm 1 takes $O(k \cdot d^2)$ time per iteration with a naive implementation. Note that the privacy preservation step has $O(k \cdot d^2)$ operations in the matrix multiplication. The Householder procedure for Thin_QR decomposition has time complexity $O(d \cdot k^2)$. Besides, the thresholding procedure has time complexity $O(d \log d)$ in the sorting part and $O(d \cdot k)$ in the looping part. If we store $Q^{(t)}$ as a row sparse matrix, the matrix multiplication should take much fewer operations and Algorithm 1 can be as efficient as $O(k \cdot \widehat{s} \cdot d)$ time complexity for each iteration.

4 Analysis of Accuracy-Privacy Tradeoff

In this section we state the differential privacy guarantee of Algorithm 1 and then analyze the minimax-optimal estimation error obtained by Algorithm 1 in Theorem 4.3. Some technical proofs are deferred to the Appendix.

Theorem 4.1 (Privacy Preservation of Algorithm 1)**.** If $0 < \epsilon < 1, 0 < \delta < 1/2$, the principal subspace estimator Q^*_{dp} of the Algorithm 1 is (ϵ, δ)-differentially private. And the information collected by the central server related to the ith data provider

$$\big(H_i(S_i, Q^{(1)}), H_i(S_i, Q^{(2)}), \ldots, H_i(S_i, Q^{(T)})\big)$$

is also (ϵ, δ)-differentially private.

Proof. See Appendix §A for a detailed proof. □

Remind that the samples are generated by a sub-Gaussian random variable with covariance matrix $\Sigma \in \mathbb{R}^{d\times k}$ and we want to recover the top k leading eigenspace of the Σ. The eigenvalues are sorted in decreasing order as $\lambda_1 \geq \lambda_2 \geq \ldots \lambda_k > \lambda_{k+1} \geq \ldots \geq$

$\lambda_d \geq 0$. We use the scalar γ as a metric of the eigengap between λ_k and λ_{k+1},

$$\gamma = \frac{3\lambda_{k+1} + \lambda_k}{3\lambda_k + \lambda_{k+1}} < 1.$$

Smaller γ implies larger eigengap λ_k/λ_{k+1} and it is easier to recover the top k eigenspace from the observed samples.

After we fix the noise level $\sigma(\epsilon, \delta)$, k, sample number n, the number of data providers N, λ_k and the thresholding parameter $\widehat{s}$, assume that there exists parameters $\alpha > 0$ and $\tau > 0$ such that

$$\frac{2k\big(\sqrt{\widehat{s}} + \sqrt{k} + \tau\big)}{\alpha} \leq \frac{n\lambda_k}{N\sigma(\epsilon, \delta)}.$$

Note that when we have sufficiently large average sample number n/N or signal-to-noise ratio $\lambda_k/\sigma(\epsilon, \delta)$, the existence of $\tau > 0$ and $\alpha > 0$ can be easily justified. Besides, the existence of smaller α implies larger signal-to-noise ratio $\lambda_k/\sigma(\epsilon, \delta)$, and as we will show later, the parameter τ is related to the tradeoff between privacy and success probability of the algorithm. We denote

$$\rho = \frac{\gamma}{1 - \alpha}$$

as the effective eigengap which characterizes the rate of convergence in our analysis.

The tradeoff analysis is presented under reasonable assumptions on the parameters α, τ, sample size n, sample dimension d, number of data providers N, the thresholding parameter $\widehat{s}$, the true sparsity level of the k leading eigenspace of the covariance matrix Σ and the quality of initialization $Q^{(0)}$. Please refer to Appendix §B.2 for the assumptions and their justifications.

The estimation error is analyzed in terms of *subspace distance*, which is defined as follows.

Definition 4.2. Let the matrices with orthogonal columns $U \in \mathbb{R}^{d\times k}$ and $V \in \mathbb{R}^{d\times k}$ be the basis matrices for two k dimensional subspaces $\mathcal{U}$ and $\mathcal{V}$ in d dimensional space. Let $U^\perp \in \mathbb{R}^{d\times(d-k)}$ and $V^\perp \in \mathbb{R}^{d\times(d-k)}$ be matrices whose orthogonal columns span the subspaces $\mathcal{U}^\perp$ and $\mathcal{V}^\perp$ which are perpendicular to $\mathcal{U}$ and $\mathcal{V}$ respectively. We define the *subspace distance* between $\mathcal{U}$ and $\mathcal{V}$ as

$$\mathcal{D}[\mathcal{U}, \mathcal{V}] = \big\|U^\top V^\perp\big\|_{\mathrm{F}} = \big\|V^\top U^\perp\big\|_{\mathrm{F}}.$$

Such definition is related to the canonical angles between subspaces. Let $\Pi_u \in \mathbb{R}^{d\times d}$ and $\Pi_v \in \mathbb{R}^{d\times d}$ be orthonormal projection matrices for $\mathcal{U}$ and $\mathcal{V}$ respectively. It will be explained later in Appendix §B.4 that the singular values of $\Pi_u\Pi_v^\perp$ are

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

Canonical angles between $\mathcal{U}$ and $\mathcal{V}$ are defined as

$$\theta_i(U, V) = \arcsin(s_i),\ i = 1, 2, \ldots, k.$$

Let $\Theta(U, V) = \mathrm{diag}(\theta_1, \theta_2, \ldots, \theta_k)$. The distance between $\mathcal{U}$ and $\mathcal{V}$ can be characterized as

$$\mathcal{D}[\mathcal{U}, \mathcal{V}] = \|\sin\Theta(U, V)\|_{\mathrm{F}} = \big\|U^\top V^\perp\big\|_{\mathrm{F}} = \big\|V^\top U^\perp\big\|_{\mathrm{F}}.$$

In the special case of $k = 1$, the subspace distance between two unit-norm eigenvectors $\mathbf{u} \in \mathbb{R}^d$ and $\mathbf{v} \in \mathbb{R}^d$ is

$$D[\mathcal{U}, \mathcal{V}] = \sin\theta(\mathbf{u}, \mathbf{v}),\ \theta(\mathbf{u}, \mathbf{v}) = \arccos\langle\mathbf{u}, \mathbf{v}\rangle$$

$$||\mathbf{u}||_2 = 1,\ ||\mathbf{v}||_2 = 1.$$

Please see Lemma B.5 in Appendix §B.4 for a detailed discussion of Definition 4.2.

Theorem 4.3 (Convergence rate and estimation error of Algorithm 1)**.** Let Assumption B.1 hold. The sequence $\{Q^{(t)}\}_{t\geq 1}^\top$ generated by Algorithm 1 satisfies

$$\|Q^{(t)\top}Q^{*\perp}\|_{\mathrm{F}} \leq \xi_{\mathrm{stats\_err}} + \xi_{\mathrm{priv\_err}} + \xi_{\mathrm{opt\_err}}(t),$$

for all $t = 1, 2, \ldots, T$ with probability at least

$$1 - 2Te^{-\tau^2/2} - \frac{4}{n-1} - \frac{1}{d} - \frac{6\log n}{n} - \frac{1}{n}, \qquad (4.1)$$

where T is the total number of iterations. Here the near optimal minimax optimal statistical error is

$$\xi_{\mathrm{stats\_err}} = \frac{C_1\sqrt{k}}{1 - \rho^{1/4}}\frac{\sqrt{\lambda_1\lambda_{k+1}}}{\lambda_k - \lambda_{k+1}}\sqrt{\frac{s^*(k + \log d)}{n}},$$
$$C_1 > 0 \text{ is a constant.} \qquad (4.2)$$

and the error induced by privacy constraint is

$$\xi_{\mathrm{priv\_err}} = \frac{1}{\sqrt{3}(1 - \rho^{1/4})}\left(1 + 2\sqrt{\frac{ks^*}{\widehat{s}}}\right)\frac{\alpha}{1 - \alpha}, \quad (4.3)$$

and the optimization error decays at geometric rate

$$\xi_{\mathrm{opt\_err}}(t) = \rho^{(t-1)/4} \cdot \min\left\{\frac{\rho^{1/2}\sqrt{(1 - \rho^{1/2})}}{2}, \frac{1}{4}\right\}. \qquad (4.4)$$

Proof. Please see Appendix §B for a detailed proof. □

Theorem 4.3 precisely characterizes the accuracy and privacy tradeoff of our algorithm.

- Privacy for free. The statistical and privacy errors can be written as

$$\xi_{\mathrm{stat\_err}} = C'\sqrt{\log d}, \quad \text{with } d \to \infty,$$
$$\xi_{\mathrm{priv\_err}} < C'(\rho)\frac{\alpha}{1 - \alpha}, \quad \text{with } \alpha \to 0,$$

where $C', C'(\rho) > 0$ are $\alpha < 1$ are dimension-free parameters determined by eigenvalues, sparsity parameter and differential privacy parameters. When the parameter α is sufficiently small and dimension d is very large, the error induced by privacy constraints can be dominated by the statistical error $\xi_{\text{priv\_err}} < \xi_{\text{stat\_err}}$. In other words, we can enjoy the differential privacy guarantee with induced error even smaller than the inherent statistical error, as is shown in Figure 2.

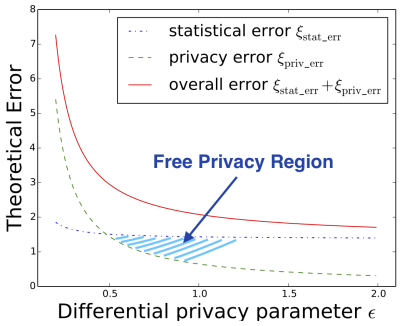


Figure 2: Illustration of theoretical privacy-accuracy tradeoff. We fix the differential parameter δ as 0.01 and let ϵ range from 0.2 to 2. With smaller ϵ, more privacy is preserved at the cost of less estimation accuracy.

- Privacy and success probability. The claims of Theorem 4.3 hold with probability at least

$$1 - 2Te^{-\tau^2/2} - \frac{4}{n-1} - \frac{1}{d} - \frac{6\log n}{n} - \frac{1}{n},$$

where in the second term, the parameter $\tau > 0$ also depends on the differential privacy parameters in the following way

$$\tau \leq \frac{n}{2kN}\frac{\lambda_k}{\sigma(\epsilon,\delta)} - \sqrt{s} - \sqrt{k}.$$

If σ is large due to stringent privacy constraint or sample size is very small, in the above equation τ has to be relatively small and thus the term $-2Te^{-\tau^2/2}$ could undermine the success probability of the algorithm.
- Minimax optimal statistical error. The statistical error $\xi_{\text{stat\_err}}$ obtained by our estimator actually achieves the lowest error (up to a constant) among all the possible estimators under worst-case generating distribution

$$\xi_{\text{stat\_err}} = C'' \inf_{\widetilde{Q}} \sup_{\mathbb{P}\in\mathcal{M}} \mathbb{E}_{\mathbb{P}}\big[\big\|\widetilde{Q}^\top Q^{*\perp}\big\|_{\text{F}}\big],$$

where $\mathbb{P}$ is any generating distribution in the model class $\mathcal{M} = \mathcal{M}(s^*, k)$ as described in Assumption 2.3, Q^* spans the eigenspace of the covariance matrix of $\mathbb{P}$ and $\widetilde{Q}$ is any estimator of Q^* based on n samples from $\mathbb{P}$. We further assume that the eigengap of population covariance matrix Σ satisfies $\lambda_k - \lambda_{k+1} > \kappa\lambda_{k+1}$ for some constant $\kappa > 0$. Under this model class, the minimax lower bound is

$$\begin{aligned}&\inf_{\widetilde{Q}} \sup_{\mathbb{P}\in\mathcal{M}} \mathbb{E}_{\mathbb{P}}\big[\big\|\widetilde{Q}^\top Q^{*\perp}\big\|_{\text{F}}\big]\\ =&C_2\frac{\sqrt{\lambda_1\lambda_{k+1}}}{\lambda_k-\lambda_{k+1}}\sqrt{\frac{s^*(k+1/4\cdot\log d)}{n}},\end{aligned}$$

where C_2 is a positive constant [41]. With eigengap condition and fixed α, we have

$$\rho = \frac{1}{3(1-\alpha)}\Big(1+\frac{4}{\kappa}\Big),$$

which can be treated as a constant for a fixed α. Hence the statistical error

$$\xi_{\text{stat\_err}} = \frac{C_1\sqrt{k}}{1-\rho^{1/4}}\frac{\sqrt{\lambda_1\lambda_{k+1}}}{\lambda_k-\lambda_{k+1}}\sqrt{\frac{s^*(k+\log d)}{n}}$$

matches the minimax optimal lower bound up to constants.
- Privacy and rate of convergence. The optimization error $\xi_{\text{opt\_err}}$ decays at the geometric rate $\rho = \gamma/(1-\alpha) < 1$. More stringent privacy requirements lead to smaller parameter α and thus much slower rate of convergence.

5 Numerical Results

In this section we present numerical synthetic experiments to validate our theory of accuracy and privacy tradeoff. Comparisons with existing state-of-the-art private PCA methods are also conducted. For any targeted privacy level, our algorithm is superior in both estimation accuracy and computation efficiency.

5.1 Empirical Privacy-Accuracy Tradeoff

We choose $d = 10^3$, $n = 10^5$, $k = 5$ and $s^* = 10$ in the experiment, and assume that there is only one data owner in the system for fair comparisons with other non-distributed private PCA method. Data samples are generated independently from a multivariate Gaussian distribution with population covariance Σ and mean 0, where $\Sigma = \sum_{j=1}^{d}\lambda_j u_j^*(u_j^*)^\top = U\Lambda U^\top$, and $\Lambda = \text{diag}\{\lambda_1, \lambda_2, \ldots, \lambda_k, \ldots, \lambda_d\}$ with

$$\begin{aligned}\lambda_i &= 100,\ \ i = 1, 2, \ldots k,\\ \lambda_i &\sim \text{Uniform}[0, 10],\ \ i = k+1, k+2, \ldots d.\end{aligned}$$

We generate an $s^* \times k$ matrix of i.i.d. standard normal Gaussian entries and orthogonalize it into a matrix L with orthogonal columns. After concatenated with a $(d - s^*) \times k$ all zero matrix, it becomes a $d \times k$ matrix $Q^* = [L^\top\ 0]^\top$ whose columns span the k leading

eigenspace. The rest of the eigenvectors $Q^{*\perp}$ are generated as follows. We project a $d \times (d-k)$ matrix R with i.i.d. standard normal entries into the linear space orthogonal to the space spanned by Q^*

$$R_\perp = (\mathrm{I}_{d\times d} - Q^* Q^{*\top})R,\ R \sim \mathcal{N}(0, \mathrm{I}_{d\times(d-k)}).$$

The columns of $R_\perp$ are then orthogonalized into matrix $Q^{*\perp}$. Finally, by concatenating the columns of Q^* and $Q^{*\perp}$ we obtain $U = [Q^*\ Q^{*\perp}]$ and thus $\Sigma = U\Lambda U^\top$. The parameters in Algorithm 1 are chosen as $T = 10$ and $\widehat{s} = 50$.

Fixing the differential parameter δ as 0.3, we plot the trajectory of estimation error $\|Q^{(t)\top} Q^{*\perp}\|_{\mathrm{F}}$ for $t = 1, 2, \ldots, T$ under different parameter ϵ in Figure 3 (a). Remind that $\|Q^{(t)\top} Q^{*\perp}\|_{\mathrm{F}}$ is the subspace distance between the subspace spanned by $Q^{(t)}$ and true eigenspace of Σ spanned by Q^*.

We also fix ϵ to be 1.0 and plot trajectory of the error for $t = 1, 2, \ldots, T$ under different parameter δ and obtain Figure 3 (b). It is clear in Figure 3 that more stringent privacy constraints can cause slower rate of convergence and larger approximation error of the output $Q^{(T)}$.

5.2 Comparisons With Existing Methods

Firstly we compare the accuracy of estimated eigenspace for fixed privacy level across three different methods: MOD-SULQ, PPCA and our DPS-PCA method described in Algorithm 1. The SuLQ method is proposed in [4] and later modified in [6] as MOD-SULQ. SuLQ contaminates the sample covariance matrix by a random matrix with independent zero mean Gaussian entries. The variance of the Gaussian noise is adjusted according to the differential parameter ϵ and δ. The PPCA method [6] privately selects leading eigenvectors by drawing from a matrix Bingham distribution by Gibbs sampling procedure [21]. Privacy is preserved thanks to the randomized noise in the Gibbs sampling. There are several other methods such as [11, 16, 18], but they follow different definition of differential privacy in terms of the granularity of privacy, which prevents direct comparisons.

5.2.1 Comparisons of Estimation Accuracy

We run simulations under the same high dimensional model described in §5.1 where dimension $d = 10^3$ and sample number $n = 10^5$. The differential parameter δ is fixed as 0.3. It is clear from Figure 4 (a) that MOD-SULQ cannot produce meaningful estimation of leading eigenvectors in high dimensional setting because the standard deviation of Gaussian noise used by MOD-SULQ is on the order of $O(d \log d)$, which becomes so large in high dimensional setting that the estimation accuracy is severely undermined. We take 1000 iterations of Gibbs sampling for the PPCA method but it is hard to determine whether or not the sampling process has reached the stationary distribution. Since the Gibbs sampling only approximates the matrix Bingham distribution, the ϵ-differential privacy is not guaranteed by PPCA in practice.

5.2.2 Comparisons of Scalability

In low dimensional settings, our DPS-PCA method also outperforms other methods. If we fix the sample size to be $n = 10^5$ and change the dimension from 40 to 140, it can be observed in Figure 4 (b) that MOD-SULQ and PPCA can perform reasonably well in low dimensions. The differential privacy parameters are set as $\delta = 0.1$ and $\epsilon = 1.0$. The simulations are carried out on the same statistical model as §5.1 but with parameters changed to $k = 3$ and $s^* = 10$.

5.2.3 Comparisons of Computation Efficiency

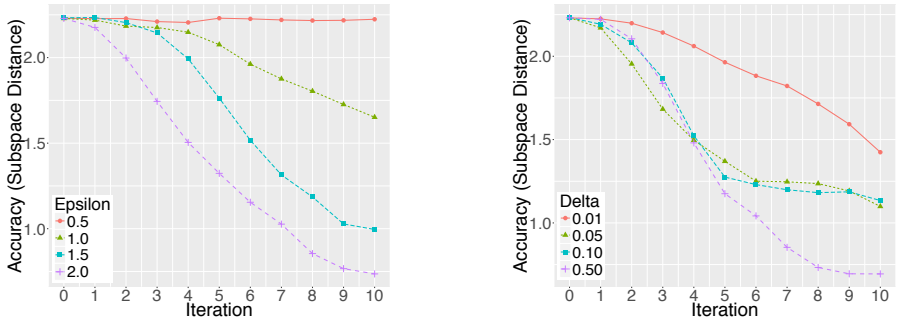
Our DPS-PCA method excels in terms of computation time thanks to the geometric convergence rate and the $O(d^2)$ time complexity in each iteration. In contrast, the Gibbs sampling procedure [21] used in PPCA takes $O(d^3)$ time per iteration and the number of iterations (burn-in time) could be on the order of $10^3 - 10^4$ before the sampling reaches the stationary distribution [6]. For ease of comparison, we only take $T = 100$ Gibbs sampling iterations for PPCA method, and use the R package rstiefel [20] as suggested in [6] for Gibbs sampling. The number of iterations for DPS-PCA is set to $T = 10$. Calculation of the leading k eigenvectors in MOD-SULQ uses the R package rARPACK [33]. We carry out simulations on the same statistical model described in §5.1 with the dimension d changed in each case. All simulations are conducted on an Intel Xeon 3.4GHz CPU.

Table 1: CPU time of privacy-preserving PCA methods under different dimensions. DPS-PCA has shown significant advantages in terms of computation complexity because it converges in geometric rate and takes only $O(d^2)$ time per iteration. PPCA takes $O(d^3)$ time for each iteration and the number of Gibbs sampling iterations can be as high as $10^4 - 10^5$ as in [6]. MOD-SULQ scales well computationally with dimension but its estimation accuracy quickly degenerates when d becomes large.

Method	CPU Time		
	$d = 200$	$d = 400$	$d = 800$
DPS-PCA	**<1ms**	**<1ms**	**<1ms**
MOD-SULQ	8ms	21ms	97ms
PPCA	9.82s	37.46s	173.78s

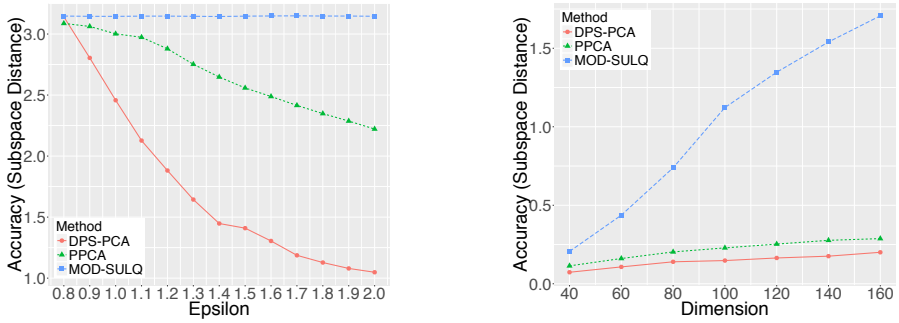
6 Conclusion

Our paper proposes an effective distributed optimization algorithm that produces a minimax-optimal sparse PCA estimator subject to differential privacy con-



(a) (b)

Figure 3: Geometric convergence of estimation accuracy (subspace distance) $\|Q^{(t)\top}Q^{*\perp}\|_{\mathrm{F}}$ in the first $T = 10$ iterations. We fix $\delta = 0.3$ and change ϵ in (a) and fix $\epsilon = 2.0$ and change δ in (b). More privacy is preserved for smaller differential parameters ϵ or δ at the cost of compromising convergence rate and the final estimation accuracy.



(a) (b)

Figure 4: Comparisons of privacy-preserving PCA methods. We plot the estimation accuracy of several methods under different differential privacy constraints in (a) where sample dimension d and sample number n are fixed as $d = 10^3$ and $n = 10^5$. In (b), the differential privacy parameters are set as $\epsilon = 1.0$ and $\delta = 0.1$ and the sample number is fixed as $n = 10^5$. We let sample dimension range from 40 to 160 and show that the performance of MOD-SULQ quickly deteriorates when d becomes large. Our method DPS-PCA outperforms in all scenarios.

straints. We analyze the tradeoff between privacy constraints and estimation accuracy and validate our theory and empirical performance of the algorithm by simulation results.

References

[1] Arash Amini and Martin J Wainwright. High-dimensional analysis of semidefinite relaxations for sparse principal components. *Annals of Statistics*, 37:2877–2921, 2009.

[2] Rajendra Bhatia. *Matrix analysis*, volume 169. Springer Science & Business Media, 2013.

[3] Aharon Birnbaum, Iain M Johnstone, Boaz Nadler, and Debashis Paul. Minimax bounds for sparse PCA with noisy high-dimensional data. *Annals of Statistics*, 41(3):1055, 2013.

[4] Avrim Blum, Cynthia Dwork, Frank McSherry, and Kobbi Nissim. Practical privacy: the SuLQ framework. In *Proceedings of ACM Symposium on Principles of Database Systems*, pages 128–138. ACM, 2005.

[5] T Tony Cai, Zongming Ma, and Yihong Wu. Sparse PCA: Optimal rates and adaptive estimation. *Annals of Statistics*, 41(6):3074–3110, 2013.

[6] Kamalika Chaudhuri, Anand Sarwate, and Kaushik Sinha. Near-optimal differentially private principal components. In *Advances in Neural Information Processing Systems*, pages 989–997, 2012.

[7] Alexandre d'Aspremont, Francis Bach, and Laurent El Ghaoui. Optimal solutions for sparse principal component analysis. *Journal of Machine Learning Research*, 9:1269–1294, 2008.

[8] Alexandre d'Aspremont, Laurent El Ghaoui, Michael I Jordan, and Gert RG Lanckriet. A direct formulation for sparse PCA using semidefinite programming. *SIAM review*, 49(3):434–448, 2007.

[9] John C Duchi, Michael Jordan, and Martin J Wainwright. Local privacy and statistical minimax rates. In *IEEE Annual Symposium on Foundations of Computer Science*, pages 429–438. IEEE, 2013.

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

[11] Cynthia Dwork, Kunal Talwar, Abhradeep Thakurta, and Li Zhang. Analyze Gauss: optimal bounds for privacy-preserving principal component analysis. In *Proceedings of ACM Symposium on Theory of Computing*, pages 11–20. ACM, 2014.

[12] Barbara E Engelhardt and Matthew Stephens. Analysis of Population Structure: A Unifying Framework and Novel Methods Based on Sparse Factor Analysis. *PLoS genetics*, 6(9):e1001117, 2010.

[13] PA Federolf, KA Boyer, and TP Andriacchi. Application of principal component analysis in clinical gait research: Identification of systematic differences between healthy and medial knee-osteoarthritic gait. *Journal of biomechanics*, 46(13):2173–2178, 2013.

[14] Srivatsava Ranjit Ganta, Shiva Prasad Kasiviswanathan, and Adam Smith. Composition attacks and auxiliary information in data privacy. In *Proceedings of the ACM SIGKDD international Conference on Knowledge Discovery and Data Mining*, pages 265–273. ACM, 2008.

[15] Gene H Golub and Charles F Van Loan. *Matrix computations*, volume 3. JHU Press, 2012.

[16] Moritz Hardt and Aaron Roth. Beating randomized response on incoherent matrices. In *Proceedings of the forty-fourth annual ACM symposium on Theory of computing*, pages 1255–1268. ACM, 2012.

[17] Moritz Hardt and Aaron Roth. Beyond worst-case analysis in private singular vector computation. In *Proceedings of Annual ACM Symposium on Theory of Computing*, pages 331–340. ACM, 2013.

[18] Hardt, Moritz and Price, Eric. The noisy power method: A meta algorithm with applications. *Advances in Neural Information Processing Systems*, pages 2861–2869, 2014.

[19] Mikko Heikkilä, Yusuke Okimoto, Samuel Kaski, Kana Shimizu, and Antti Honkela. Differentially private bayesian learning on distributed data. *arXiv preprint arXiv:1703.01106*, 2017.

[20] Peter Hoff. *rstiefel: Random orthonormal matrix generation on the Stiefel manifold*, 2014. R package version 0.10.

[21] Peter D Hoff. Simulation of the matrix Bingham-von Mises-Fisher distribution, with applications to multivariate and relational data. *Journal of Computational and Graphical Statistics*, 18(2), 2009.

[22] Nils Homer, Szabolcs Szelinger, Margot Redman, David Duggan, Waibhav Tembe, Jill Muehling, John V Pearson, Dietrich A Stephan, Stanley F Nelson, and David W Craig. Resolving individuals contributing trace amounts of dna to highly complex mixtures using high-density snp genotyping microarrays. *PLoS Genetics*, 4(8):e1000167, 2008.

[23] Iain M Johnstone and Arthur Yu Lu. On consistency and sparsity for principal components analysis in high dimensions. *Journal of the American Statistical Association*, 104(486), 2009.

[24] Ian T Jolliffe, Nickolay T Trendafilov, and Mudassir Uddin. A modified principal component technique based on the lasso. *Journal of Computational and Graphical Statistics*, 12(3):531–547, 2003.

[25] Ravi Kannan, Santosh Vempala, and David P Woodruff. Principal component analysis and higher correlations for distributed data. *International Conference on Machine Learning*, pages 1040–1057, 2014.

[26] Michael Kapralov and Kunal Talwar. On differentially private low rank approximation. In *Proceedings of the Annual ACM-SIAM Symposium on Discrete Algorithms*, pages 1395–1414. SIAM, 2013.

[27] Robert Krauthgamer, Boaz Nadler, and Dan Vilenchik. Do semidefinite relaxations really solve sparse PCA? *arXiv preprint arXiv:1306.3690*, 2013.

[28] Yingyu Liang, Maria-Florina Balcan, Vandana Kanchanapally, and David P Woodruff. Improved distributed principal component analysis. *Advances in Neural Information Processing Systems*, pages 3113–3121, 2014.

[29] Karim Lounici. Sparse principal component analysis with missing observations. In *High Dimensional Probability VI*, pages 327–356. Springer, 2013.

[30] Frank McSherry and Kunal Talwar. Mechanism Design via Differential Privacy. In *Proceedings of the Annual IEEE Symposium on Foundations of Computer Science*, pages 94–103. IEEE Computer Society, 2007.

[31] Boaz Nadler. Finite sample approximation results for principal component analysis: A matrix perturbation approach. *Annals of Statistics*, 41(2):2791–2817, 2008.

[32] Arvind Narayanan and Vitaly Shmatikov. Robust deanonymization of large sparse datasets. *IEEE Symposium on Security and Privacy*, pages 111–125, 2008.

[33] Yixuan Qiu, Jiali Mei, and authors of the ARPACK library. See file AUTHORS for details. *rARPACK: R wrapper of ARPACK for large scale eigenvalue/vector problems, on both dense and sparse matrices*, 2014. R package version 0.7-0.

[34] Yongming Qu, George Ostrouchov, Nagiza Samatova, and Al Geist. Principal component analysis for dimension reduction in massive distributed data sets. In *Proceedings of IEEE International Conference on Data Mining (ICDM)*, 2002.

[35] Mark Rudelson and Roman Vershynin. Non-asymptotic theory of random matrices: extreme singular values. *arXiv.org*, March 2010.

[36] Dan Shen, Haipeng Shen, and JS Marron. Consistency of sparse PCA in high dimension, low sample size contexts. *Journal of Multivariate Analysis*, 115:317–333, 2013.

[37] Haipeng Shen and Jianhua Z Huang. Sparse principal component analysis via regularized low rank matrix approximation. *Journal of Multivariate Analysis*, 99(6):1015–1034, 2008.

[38] Gilbert W Stewart. *Perturbation Theory for the Singular Value Decomposition*. Elsevier, 1990.

[39] Gilbert W Stewart and Ji-guang Sun. Matrix Perturbation Theory, 1990.

[40] Vincent Q Vu and Jing Lei. Minimax Rates of Estimation for Sparse PCA in High Dimensions. *International Conference on Artificial Intelligence and Statistics*, pages 1278–1286, 2012.

[41] Vincent Q Vu and Jing Lei. Minimax sparse principal subspace estimation in high dimensions. *Annals of Statistics*, 41(6):2905–2947, 2013.

[42] Martin J Wainwright, Michael I Jordan, and John C Duchi. Privacy aware learning. In *Advances in Neural Information Processing Systems*, pages 1430–1438, 2012.

[43] Zhaoran Wang, Huanran Lu, and Han Liu. Tighten after Relax: Minimax-Optimal Sparse PCA in Polynomial Time. *Advances in Neural Information Processing Systems*, pages 3383–3391, 2014.

[44] Larry Wasserman and Shuheng Zhou. A statistical framework for differential privacy. *Journal of the American Statistical Association*, 105(489):375–389, 2010.

[45] Daniela M Witten, Robert Tibshirani, and Trevor Hastie. A penalized matrix decomposition, with applications to sparse principal components and canonical correlation analysis. *Biostatistics*, page kxp008, 2009.

[46] Genqiang Wu, Yeping He, Jingzheng Wu, and Xianyao Xia. Inherit differential privacy in distributed setting: Multiparty randomized function computation. *arXiv preprint arXiv:1604.03001*, 2016.

[47] Xiao-Tong Yuan and Tong Zhang. Truncated power method for sparse eigenvalue problems. *Journal of Machine Learning Research*, 14(1):899–925, 2013.

[48] Shuheng Zhou, Katrina Ligett, and Larry Wasserman. Differential privacy with compression. In *IEEE International Symposium on Information Theory*, pages 2718–2722. IEEE, 2009.

[49] Hui Zou. The adaptive lasso and its oracle properties. *Journal of the American Statistical Association*, 101(476):1418–1429, 2006.