Near-optimal Differentially Private Principal Components

Kamalika Chaudhuri
UC San Diego
kchaudhuri@ucsd.edu

Anand D. Sarwate
TTI-Chicago
asarwate@ttic.edu

Kaushik Sinha
UC San Diego
ksinha@cs.ucsd.edu

Abstract

Principal components analysis (PCA) is a standard tool for identifying good low-dimensional approximations to data sets in high dimension. Many current data sets of interest contain private or sensitive information about individuals. Algorithms which operate on such data should be sensitive to the privacy risks in publishing their outputs. Differential privacy is a framework for developing tradeoffs between privacy and the utility of these outputs. In this paper we investigate the theory and empirical performance of differentially private approximations to PCA and propose a new method which explicitly optimizes the utility of the output. We demonstrate that on real data, there is a large performance gap between the existing method and our method. We show that the sample complexity for the two procedures differs in the scaling with the data dimension, and that our method is nearly optimal in terms of this scaling.

1 Introduction

Dimensionality reduction is a fundamental tool for understanding complex data sets that arise in contemporary machine learning and data mining applications. Even though a single data point can be represented by hundreds or even thousands of features, the phenomena of interest are often intrinsically low-dimensional. By reducing the "extrinsic" dimension of the data to its "intrinsic" dimension, analysts can discover important structural relationships between features, more efficiently use the transformed data for learning tasks such as classification or regression, and greatly reduce the space required to store the data. One of the oldest and most classical methods for dimensionality reduction is principal components analysis (PCA), which computes a low-rank approximation to the second moment matrix of a set of points in $\mathbb{R}^d$. The rank k of the approximation is chosen to be the intrinsic dimension of the data. We view this procedure as specifying a k-dimensional subspace of $\mathbb{R}^d$.

Much of today's machine-learning is performed on the vast amounts of personal information collected by private companies and government agencies about individuals, such as customers, users, and subjects. These datasets contain sensitive information about individuals and typically involve a large number of features. It is therefore important to design machine-learning algorithms which discover important structural relationships in the data while taking into account its sensitive nature. We study approximations to PCA which guarantee differential privacy, a cryptographically motivated definition of privacy [9] that has gained significant attention over the past few years in the machine-learning and data-mining communities [19, 21, 20, 10, 23]. Differential privacy measures privacy risk by a parameter α that bounds the log-likelihood ratio of output of a (private) algorithm under two databases differing in a single individual.

There are many general tools for providing differential privacy. The sensitivity method [9] computes the desired algorithm (PCA) on the data and then adds noise proportional to the maximum change than can be induced by changing a single point in the data set. The PCA algorithm is very sensitive

in this sense because the top eigenvector can change by 90° by changing one point in the data set. Relaxations such as smoothed sensitivity [24] are difficult to compute in this setting as well. The SULQ method of Blum et al. [2] adds noise to the second moment matrix and then runs PCA on the noisy matrix. As our experiments show, the amount of noise required is often quite severe and SULQ seems impractical for data sets of moderate size.

The general SULQ method does not take into account the quality of approximation to the non-private PCA output. We address this by proposing a new method, PPCA, that is an instance of the exponential mechanism of McSherry and Talwar [22]. For any $k < d$, this differentially private method outputs a k-dimensional subspace; the output is biased towards subspaces which are close to the output of PCA. In our case, the method corresponds to sampling from the matrix Bingham distribution. We implement this method using a Markov Chain Monte Carlo (MCMC) procedure due to Hoff [15] and show that it achieves significantly better empirical performance.

In order to understand the performance gap, we prove sample complexity bounds in case of $k = 1$ for SULQ and PPCA, as well as a general lower bound on the sample complexity for any differentially private algorithm. We show that (up to log factors) the sample complexity scales as $\Omega(d^{3/2}\sqrt{d})$ for SULQ and as $O(d)$ for PPCA. Furthermore, any differentially private algorithm requires $\Omega(d)$ samples, showing that PPCA is nearly optimal in terms of sample complexity as a function of data dimension. These theoretical results suggest that our experiments exhibit the limit of how well α-differentially private algorithms can perform, and our experiments show that this gap should persist for general k.

There are several interesting open questions suggested by this work. One set of issues is computational. Differentially privacy is a mathematical definition, but algorithms must be implemented using finite precision machines. Privacy and computation interact in many places, including pseudorandomness, numerical stability, optimization, and in the MCMC procedure we use to implement PPCA; investigating the impact of approximate sampling is an avenue for future work. A second set of issues is theoretical – while the privacy guarantees of PPCA hold for all k, our theoretical analysis of sample complexity applies only to $k = 1$ in which the distance and angles between vectors are related. An interesting direction is to develop theoretical bounds for general k; challenges here are providing the right notion of approximation of PCA, and extending the theory using packings of Grassman or Stiefel manifolds.

2 Preliminaries

The data given to our algorithm is a set of n vectors $\mathcal{D} = \{x_1, x_2, \ldots, x_n\}$ where each x_i corresponds to the private value of one individual, $x_i \in \mathbb{R}^d$, and $\|x_i\| \leq 1$ for all i. Let $X = [x_1, \ldots, x_n]$ be the matrix whose columns are the data vectors $\{x_i\}$. Let $A = \frac{1}{n} X X^T$ denote the $d \times d$ second moment matrix of the data. The matrix A is positive semidefinite, and has Frobenius norm at most 1.

The problem of dimensionality reduction is to find a "good" low-rank approximation to A. A popular solution is to compute a rank-k matrix $\hat{A}$ which minimizes the norm $\|A - \hat{A}\|_{\mathrm{F}}$, where k is much lower than the data dimension d. The Schmidt approximation theorem [25] shows that the minimizer is given by the singular value decomposition, also known as the PCA algorithm in some areas of computer science.

Definition 1. *Suppose A is a positive semidefinite matrix whose first k eigenvalues are distinct. Let the eigenvalues of A be $\lambda_1(A) \geq \lambda_2(A) \geq \cdots \geq \lambda_d(A) \geq 0$ and let Λ be a diagonal matrix with $\Lambda_{ii} = \lambda_i(A)$. The matrix A decomposes as*

$$A = V \Lambda V^T, \tag{1}$$

where V is an orthonormal matrix of eigenvectors. The top-k subspace of A is the matrix

$$V_k(A) = [v_1 \; v_2 \; \cdots \; v_k], \tag{2}$$

where v_i is the i-th column of V in (1)*.*

Given the top-k subspace and the eigenvalue matrix Λ, we can form an approximation $A^{(k)} = V_k(A) \Lambda_k V_k(A)^T$ to A, where Λ_k contains the k largest eigenvalues in Λ. In the special case $k = 1$

we have $A^{(1)} = \lambda_1(A) v_1 v_1^T$, where v_1 is the eigenvector corresponding to $\lambda_1(A)$. We refer to v_1 as the *top eigenvector* of the data. For a $d \times k$ matrix $\hat{V}$ with orthonormal columns, the quality of $\hat{V}$ in approximating A can be measured by

$$q_{\mathrm{F}}(\hat{V}) = \mathrm{tr}\left(\hat{V}^T A \hat{V}\right). \tag{3}$$

The $\hat{V}$ which maximizes $q(\hat{V})$ has columns equal to $\{v_i : i \in [k]\}$, corresponding to the top k eigenvectors of A.

Our theoretical results apply to the special case $k = 1$. For these results, we measure the inner product between the output vector $\hat{v}_1$ and the true top eigenvector v_1:

$$q_{\mathrm{A}}(\hat{v}_1) = |\langle \hat{v}_1, v_1 \rangle|. \tag{4}$$

This is related to (3). If we write $\hat{v}_1$ in the basis spanned by $\{v_i\}$, then

$$q_{\mathrm{F}}(\hat{v}_1) = \lambda_1 q_{\mathrm{A}}(\hat{v}_1)^2 + \sum_{i=2}^{d} \lambda_i \langle \hat{v}_1, v_i \rangle^2.$$

Our proof techniques use the geometric properties of $q_{\mathrm{A}}(\cdot)$.

Definition 2. *A randomized algorithm $\mathcal{A}(\cdot)$ is an (ρ, η)-close approximation to the top eigenvector if for all data sets $\mathcal{D}$ of n points,*

$$\mathbb{P}\left(q_{\mathrm{A}}(\mathcal{A}(\mathcal{D})) \geq \rho\right) \geq 1 - \eta, \tag{5}$$

where the probability is taken over $\mathcal{A}(\cdot)$.

We study approximations to $\hat{A}$ that preserve the privacy of the underlying data. The notion of privacy that we use is differential privacy, which quantifies the privacy guaranteed by a randomized algorithm $\mathcal{P}$ applied to a data set $\mathcal{D}$.

Definition 3. *An algorithm $\mathcal{A}(\mathcal{B})$ taking values in a set $\mathcal{T}$ provides α-differential privacy if*

$$\sup_{\mathcal{S}} \sup_{\mathcal{D}, \mathcal{D}'} \frac{\mu\left(\mathcal{S} \mid \mathcal{B} = \mathcal{D}\right)}{\mu\left(\mathcal{S} \mid \mathcal{B} = \mathcal{D}'\right)} \leq e^{\alpha}, \tag{6}$$

where the first supremum is over all measurable $\mathcal{S} \subseteq \mathcal{T}$, the second is over all data sets $\mathcal{D}$ and $\mathcal{D}'$ differing in a single entry, and $\mu(\cdot|\mathcal{B})$ is the conditional distribution (measure) on $\mathcal{T}$ induced by the output $\mathcal{A}(\mathcal{B})$ given a data set $\mathcal{B}$. The ratio is interpreted to be 1 whenever the numerator and denominator are both 0.

Definition 4. *An algorithm $\mathcal{A}(\mathcal{B})$ taking values in a set $\mathcal{T}$ provides (α, δ)-differential privacy if*

$$\mathbb{P}\left(\mathcal{A}(\mathcal{D}) \in \mathcal{S}\right) \leq e^{\alpha} \mathbb{P}\left(\mathcal{A}(\mathcal{D}') \in \mathcal{S}\right) + \delta, \tag{7}$$

for all all measurable $\mathcal{S} \subseteq \mathcal{T}$ and all data sets $\mathcal{D}$ and $\mathcal{D}'$ differing in a single entry.

Here α and δ are privacy parameters, where low α and δ ensure more privacy. For more details about these definitions, see [9, 26, 8]. The second privacy guarantee is weaker; the parameter δ bounds the probability of failure, and is typically chosen to be quite small.

In this paper we are interested in proving results on the sample complexity of differentially private algorithms that approximate PCA. That is, for a given α and ρ, how large must the number of individuals n in the data set be such that it is α-differentially private and also a (ρ, η)-close approximation to PCA? It is well known that as the number of individuals n grows, it is easier to guarantee the same level of privacy with relatively less noise or perturbation, and therefore the utility of the approximation also improves. Our results characterize how privacy and utility scale with n and the tradeoff between them for fixed n.

Related Work Differential privacy was proposed by Dwork et al. [9], and has spawned an extensive literature of general methods and applications [1, 21, 27, 6, 24, 3, 22, 10] Differential privacy has been shown to have strong *semantic* guarantees [9, 17] and is resistant to many attacks [12] that succeed against some other definitions of privacy. There are several standard approaches for designing differentially-private data-mining algorithms, including input perturbation [2], output perturbation [9], the exponential mechanism [22], and objective perturbation [6]. To our knowledge, other

than SULQ method [2], which provides a general differentially-private input perturbation algorithm, this is the first work on differentially-private PCA. Independently, [14] consider the problem of differentially-private low-rank matrix reconstruction for applications to sparse matrices; provided certain coherence conditions hold, they provide an algorithm for constructing a rank $2k$ approximation B to a matrix A such that $\|A - B\|_{\mathrm{F}}$ is $O(\|A - A_k\|)$ plus some additional terms which depend on d, k and n; here A_k is the best rank k approximation to A. Because of their additional assumptions, their bounds are generally incomparable to ours, and our bounds are superior for dense matrices.

The data-mining community has also considered many different models for privacy-preserving computation – see Fung et al. for a survey with more references [11]. Many of the models used have been shown to be susceptible to composition attacks, when the adversary has some amount of prior knowledge [12]. An alternative line of privacy-preserving data-mining work [28] is in the Secure Multiparty Computation setting; one work [13] studies privacy-preserving singular value decomposition in this model. Finally, dimension reduction through random projection has been considered as a technique for sanitizing data prior to publication [18]; our work differs from this line of work in that we offer differential privacy guarantees, and we only release the PCA subspace, not actual data. Independently, Kapralov and Talwar [16] have proposed a dynamic programming algorithm for differentially private low rank matrix approximation which involves sampling from a distribution induced by the exponential mechanism. The running time of their algorithm is $O(d^6)$, where d is the data dimension.

3 Algorithms and results

In this section we describe differentially private techniques for approximating (2). The first is a modified version of the SULQ method [2]. Our new algorithm for differentially-private PCA, PPCA, is an instantiation of the exponential mechanism due to McSherry and Talwar [22]. Both procedures provide differentially private approximations to the top-k subspace: SULQ provides (α, δ)-differential privacy and PPCA provides α-differential privacy.

Input perturbation. The only differentially-private approximation to PCA prior to this work is the SULQ method [2]. The SULQ method perturbs each entry of the empirical second moment matrix A to ensure differential privacy and releases the top k eigenvectors of this perturbed matrix. In particular, SULQ recommends adding a matrix N of i.i.d. Gaussian noise of variance $\frac{8d^2 \log^2(d/\delta)}{n^2\alpha^2}$ and applies the PCA algorithm to $A + N$. This guarantees a weaker privacy definition known as (α, δ)-differential privacy. One problem with this approach is that with probability 1 the matrix $A + N$ is not symmetric, so the largest eigenvalue may not be real and the entries of the corresponding eigenvector may be complex. Thus the SULQ algorithm is not a good candidate for practical privacy-preserving dimensionality reduction.

However, a simple modification to the basic SULQ approach does guarantee (α, δ) differential privacy. Instead of adding a asymmetric Gaussian matrix, the algorithm can add the a symmetric matrix with i.i.d. Gaussian entries N. That is, for $1 \leq i \leq j \leq d$, the variable N_{ij} is an independent Gaussian random variable with variance β^2. Note that this matrix is symmetric but not necessarily positive semidefinite, so some eigenvalues may be negative but the eigenvectors are all real. A derivation for the noise variance is given in Theorem 1.

Algorithm 1: Algorithm MOD-SULQ (input pertubation)

inputs: $d \times n$ data matrix X, privacy parameter α, parameter δ

outputs: $d \times k$ matrix $\hat{V}_k = [\hat{v}_1\ \hat{v}_2\ \cdots\ \hat{v}_k]$ with orthonormal columns

1 Set $A = \frac{1}{n} X X^T$.;

2 Set $\beta = \frac{d+1}{n\alpha}\sqrt{2\log\left(\frac{d^2+d}{\delta 2\sqrt{2\pi}}\right)} + \frac{1}{\sqrt{\alpha}n}$. Generate a $d \times d$ symmetric random matrix N whose entries are i.i.d. drawn from $\mathcal{N}\left(0, \beta^2\right)$. ;

3 Compute $\hat{V}_k = V_k(A + N)$ according to (2). ;

Exponential mechanism. Our new method, PPCA, randomly samples a k-dimensional subspace from a distribution that ensures differential privacy and is biased towards high utility. The distribution from which our released subspace is sampled is known in the statistics literature as the matrix Bingham distribution [7], which we denote by $\mathsf{BMF}_k(B)$. The algorithm is in terms of general $k < d$ but our theoretical results focus on the special case $k = 1$ where we wish to release a one-dimensional approximation to the data covariance matrix. The matrix Bingham distribution takes values on the set of all k-dimensional subspaces of $\mathbb{R}^d$ and has a density equal to

$$f(V) = \frac{1}{{}_1F_1\left(\frac{1}{2}k, \frac{1}{2}d, B\right)} \exp(\mathrm{tr}(V^T B V)), \tag{8}$$

where V is a $d \times k$ matrix whose columns are orthonormal and ${}_1F_1\left(\frac{1}{2}k, \frac{1}{2}d, B\right)$ is a confluent hypergeometric function [7, p.33].

Algorithm 2: Algorithm PPCA (exponential mechanism)

inputs: $d \times n$ data matrix X, privacy parameter α, dimension k

outputs: $d \times k$ matrix $\hat{V}_k = [\hat{v}_1\ \hat{v}_2\ \cdots\ \hat{v}_k]$ with orthonormal columns

1 Set $A = \frac{1}{n} X X^T$;

2 Sample $\hat{V}_k = \mathsf{BMF}\left(n\frac{\alpha}{2}A\right)$;

By combining results on the exponential mechanism [22] along with properties of PCA algorithm, we can show that this procedure is differentially private. In many cases, sampling from the distribution specified by the exponential mechanism distribution may be difficult computationally, especially for continuous-valued outputs. We implement PPCA using a recently-proposed Gibbs sampler due to Hoff [15]. Gibbs sampling is a popular Markov Chain Monte Carlo (MCMC) technique in which samples are generated according to a Markov chain whose stationary distribution is the density in (8). Assessing the "burn-in time" and other factors for this procedure is an interesting question in its own right; further details are in Section E.3.

Other approaches. There are other general algorithmic strategies for guaranteeing differential privacy. The sensitivity method [9] adds noise proportional to the maximum change that can be induced by changing a single point in the data set. Consider a data set $\mathcal{D}$ with $m+1$ copies of a unit vector u and m copies of a unit vector u' with $u \perp u'$ and let $\mathcal{D}'$ have m copies of u and $m+1$ copies of u'. Then $v_1(\mathcal{D}) = u$ but $v_1(\mathcal{D}') = u'$, so $\|v_1(\mathcal{D}) - v_1(\mathcal{D}')\| = \sqrt{2}$. Thus the global sensitivity does not scale with the number of data points, so as n increases the variance of the noise required by the Laplace mechanism [9] will not decrease. An alternative to global sensitivity is smooth sensitivity [24]; except for special cases, such as the sample median, smooth sensitivity is difficult to compute for general functions. A third method for computing private, approximate solutions to high-dimensional optimization problems is objective perturbation [6]; to apply this method, we require the optimization problems to have certain properties (namely, strong convexity and bounded norms of gradients), which do not apply to PCA.

Main results. Our theoretical results are sample complexity bounds for PPCA and MOD-SULQ as well as a general lower bound on the sample complexity for any α-differentially private algorithm. These results show that the PPCA is nearly optimal in terms the scaling of the sample complexity with respect to the data dimension d, privacy parameter α, and eigengap Δ. We further show that MOD-SULQ requires more samples as a function of d, despite having a slightly weaker privacy guarantee. Proofs are deferred to the supplementary material.

Even though both algorithms can output the top-k PCA subspace for general $k \leq d$, we prove results for the case $k = 1$. Finding the scaling behavior of the sample complexity with k is an interesting open problem that we leave for future work; challenges here are finding the right notion of approximation of the PCA, and extending the theory using packings of Grassman or Stiefel manifolds.

Theorem 1. *For the β in Algorithm 1, the MOD-SULQ algorithm is (α, δ) differentially private.*

Theorem 2. *Algorithm PPCA is α-differentially private.*

The fact that these two algorithms are differentially private follows from some simple calculations. Our first sample complexity result provides an upper bound on the number of samples required by

PPCA to guarantee a certain level of privacy and accuracy. The sample complexity of PPCA n grows linearly with the dimension d, inversely with α, and inversely with the correlation gap $(1-\rho)$ and eigenvalue gap $\lambda_1(A) - \lambda_2(A)$.

Theorem 3 (Sample complexity of PPCA). *If* $n > \frac{d}{\alpha(1-\rho)(\lambda_1-\lambda_2)}\left(\frac{\log(1/\eta)}{d} + \log\frac{4\lambda_1}{(1-\rho^2)(\lambda_1-\lambda_2)}\right)$, *then PPCA is a* (ρ, η)*-close approximation to PCA.*

Our second result shows a lower bound on the number of samples required by *any* α-differentially-private algorithm to guarantee a certain level of accuracy for a large class of datasets, and uses proof techniques in [4, 5].

Theorem 4 (Sample complexity lower bound). *Fix* d, α, $\Delta \leq \frac{1}{2}$ *and let* $1 - \phi = \exp\left(-2 \cdot \frac{\ln 8 + \ln(1+\exp(d))}{d-2}\right)$. *For any* $\rho \geq 1 - \frac{1-\phi}{16}$, *no* α*-differentially private algorithm* $\mathcal{A}$ *can approximate PCA with expected utility greater than* ρ *on all databases with* n *points in dimension* d *having eigenvalue gap* Δ, *where* $n < \max\left\{\frac{d}{\Delta\alpha}, \sqrt{\frac{1-\phi}{80}} \cdot \frac{d}{\Delta\alpha\sqrt{1-\rho}}\right\}$.

Theorem 3 shows that if n scales like $\frac{d}{\alpha\Delta(1-\rho)} \log \frac{1}{1-\rho^2}$ then PPCA produces an approximation $\hat{v}_1$ that has correlation ρ with v_1, whereas Theorem 4 shows that n must scale like $\frac{d}{\alpha\Delta\sqrt{(1-\rho)}}$ for any α-differentially private algorithm. In terms of scaling with d, α and Δ, the upper and lower bounds match, and they also match up to square-root factors with respect to the correlation. By contrast, the following lower bound on the number of samples required by MOD-SULQ to ensure a certain level of accuracy shows that MOD-SULQ has a less favorable scaling with dimension.

Theorem 5 (Sample complexity lower bound for MOD-SULQ). *There are constants* c *and* c' *such that if* $n < c\frac{d^{3/2}\sqrt{\log(d/\delta)}}{\alpha}(1 - c'(1-\rho))$, *then there is a dataset of size* n *in dimension* d *such that the top PCA direction* v *and the output* $\hat{v}$ *of MOD-SULQ satisfy* $\mathbf{E}[|\langle \hat{v}_1, v_1 \rangle|] \leq \rho$.

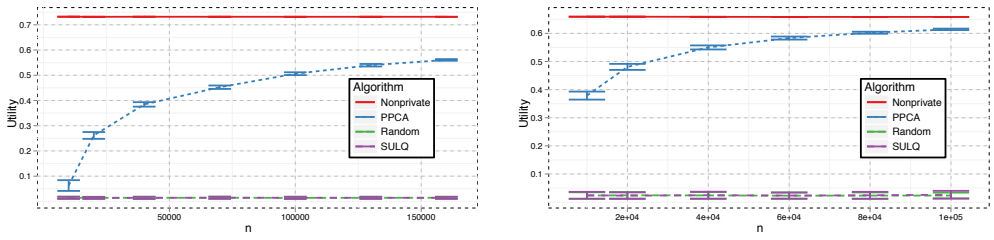
Notice that the dependence on n grows as $d^{3/2}$ in SULQ as opposed to d in PPCA. Dimensionality reduction via PCA is often used in applications where the data points occupy a low dimensional space but are presented in high dimensions. These bounds suggest that PPCA is better suited to such applications than MOD-SULQ. We next turn to validating this intuition on real data.

4 Experiments

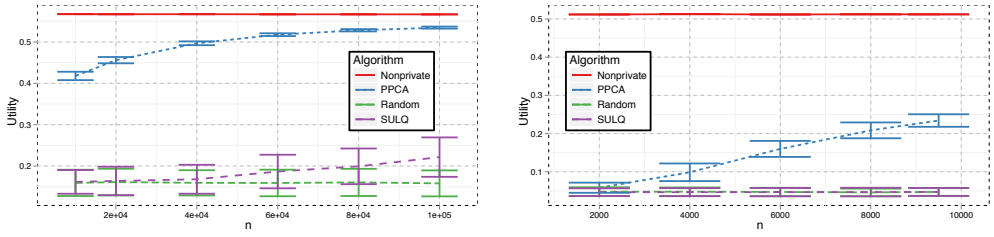
We chose four datasets from four different domains – `kddcup99`, which includes features of 494,021 network connections, `census`, a demographic data set on 199,523 individuals, `localization`, a medical dataset with 164,860 instances of sensor readings on individuals engaged in different activities, and `insurance`, a dataset on product usage and demographics of 9,822 individuals. After preprocessing, the dimensions of these datasets are 116, 513, 44 and 150 respectively. We chose k to be 4, 8, 10, and 11 such that the top-k PCA subspace had $q_{\mathrm{F}}(V_k)$ at least 80% of $\|A\|_{\mathrm{F}}$. More details are in Appendix E in the supplementary material.

We ran three algorithms on these data sets : standard (non-private) PCA, MOD-SULQ with $\alpha = 0.1$ and $\delta = 0.01$, and PPCA with $\alpha = 0.1$. As a sanity check, we also tried a uniformly generated random projection – since this projection is data-independent we would expect it to have low utility. Standard PCA is non-private; changing a single data point will change the output, and hence violate differential privacy. We measured the utility $q_{\mathrm{F}}(U)$, where U is the k-dimensional subspace output by the algorithm; $\|U\|$ is maximized when U is the top-k PCA subspace, and thus this reflects how close the output subspace is to the true PCA subspace in terms of representing the data. Although our theoretical results hold for $q_{\mathrm{A}}(\cdot)$, the "energy" $q_{\mathrm{F}}(\cdot)$ is more relevant in practice for larger k.

Figures 1(a), 1(b), 1(c), and 1(d) show $q_{\mathrm{F}}(U)$ as a function of sample size for the k-dimensional subspace output by PPCA, MOD-SULQ, non-private PCA, and random projections. Each value in the figure is an average over 5 random permutations of the data, as well as 10 random starting points of the Gibbs sampler per permutation (for PPCA), and 100 random runs per permutation (for MOD-SULQ and random projections).



(a) `census` (b) `kddcup`



(c) `localization` (d) `insurance`

Figure 1: Utility $q_{\mathrm{F}}(U)$ for the four data sets

	Non-private PCA	PPCA	MOD-SULQ	Random projections
KDDCUP	98.97 ± 0.05	98.95 ± 0.05	98.18 ± 0.65	98.23 ± 0.49
LOCALIZATION	100 ± 0	100 ± 0	97.06 ± 2.17	96.28 ± 2.34

Table 1: Classification accuracy in the k-dimensional subspaces for `kddcup99`($k = 4$), and `localization`($k = 10$) in the k-dimensional subspaces reported by the different algorithms.

The plots show that PPCA always outperforms MOD-SULQ, and approaches the performance of non-private PCA with increasing sample size. By contrast, for most of the problems and sample sizes considered by our experiments, MOD-SULQ does not perform much better than random projections. The only exception is `localization`, which has much lower dimension (44). This confirms that MOD-SULQ does not scale very well with the data dimension d. The performance of both MOD-SULQ and PPCA improve as the sample size increases; the improvement is faster for PPCA than for MOD-SULQ. However, to be fair, MOD-SULQ is simpler and hence runs faster than PPCA. At the sample sizes in our experiments, the performance of non-private PCA does not improve much with a further increase in samples. Our theoretical results suggest that the performance of differentially private PCA cannot be significantly improved over these experiments.

Effect of privacy on classification. A common use of a dimension reduction algorithm is as a precursor to classification or clustering; to evaluate the effectiveness of the different algorithms, we projected the data onto the subspace output by the algorithms, and measured the classification accuracy using the projected data. The classification results are summarized in Table 4. We chose the *normal* vs. all classification task in `kddcup99`, and the *falling* vs. all classification task in `localization`. [1] We used a linear SVM for all classification experiments.

For the classification experiments, we used half of the data as a holdout set for computing a projection subspace. We projected the classification data onto the subspace computed based on the holdout set; 10% of this data was used for training and parameter-tuning, and the rest for testing. We repeated the classification process 5 times for 5 different (random) projections for each algorithm, and then ran the entire procedure over 5 random permutations of the data. Each value in the figure is thus an average over $5 \times 5 = 25$ rounds of classification.

[1] For the other two datasets, `census` and `insurance`, the classification accuracy of linear SVM after (non-private) PCAs is as low as always predicting the majority label.

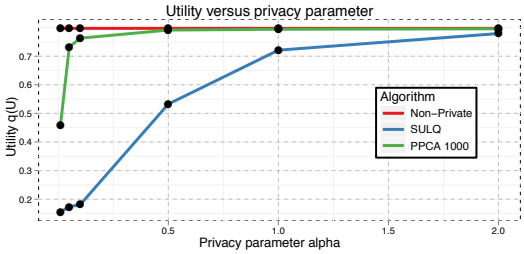


Figure 2: Plot of $q_{\mathrm{F}}(U)$ versus α for a synthetic data set with $n = 5{,}000$, $d = 10$, and $k = 2$.

The classification results show that our algorithm performs almost as well as non-private PCA for classification in the top k PCA subspace, while the performance of MOD-SULQ and random projections are a little worse. The classification accuracy while using MOD-SULQ and random projections also appears to have higher variance compared to our algorithm and non-private PCA; this can be explained by the fact that these projections tend to be farther from the PCA subspace, in which the data has higher classification accuracy.

Effect of the privacy requirement. To check the effect of the privacy requirement, we generated a synthetic data set of $n = 5{,}000$ points drawn from a Gaussian distribution in $d = 10$ with mean $\mathbf{0}$ and whose covariance matrix had eigenvalues $\{0.5, 0.30, 0.04, 0.03, 0.02, 0.01, 0.004, 0.003, 0.001, 0.001\}$. In this case the space spanned by the top two eigenvectors has most of the energy, so we chose $k = 2$ and plotted the utility $q_{\mathrm{F}}(\cdot)$ for non-private PCA, MOD-SULQ with $\delta = 0.05$, and PPCA. We drew 100 samples from each privacy-preserving algorithm and the plot of the average utility versus α is shown in Figure 2. As α increases, the privacy requirement is relaxed and both MOD-SULQ and PPCA approach the utility of PCA without privacy constraints. However, for moderate α the PPCA still captures most of the utility, whereas the gap between MOD-SULQ and PPCA becomes quite large.

5 Conclusion

In this paper we investigated the theoretical and empirical performance of differentially private approximations to PCA. Empirically, we showed that MOD-SULQ and PPCA differ markedly in how well they approximate the top-k subspace of the data. The reason for this, theoretically, is that the sample complexity of MOD-SULQ scales with $d^{3/2}\sqrt{\log d}$ whereas PPCA scales with d. Because PPCA uses the exponential mechanism with $q_{\mathrm{F}}(\cdot)$ as the utility function, it is not surprising that it performs well. However, MOD-SULQ often had a performance comparable to random projections, indicating that the real data sets we used were too small for it to be effective. We furthermore showed that PPCA is nearly optimal, in that any differentially private approximation to PCA must use $\Omega(d)$ samples.

Our investigation brought up many interesting issues to consider for future work. The description of differentially private algorithms assume an ideal model of computation : real systems require additional security assumptions that have to be verified. The difference between truly random noise and pseudorandomness and the effects of finite precision can lead to a gap between the theoretical ideal and practice. Numerical optimization methods used in objective perturbation [6] can only produce approximate solutions, and have complex termination conditions unaccounted for in the theoretical analysis. Our MCMC sampling has this flavor : we cannot sample exactly from the Bingham distribution because we must determine the Gibbs sampler's convergence empirically. Accounting for these effects is an interesting avenue for future work that can bring theory and practice together.

Finally, more germane to the work on PCA here is to prove sample complexity results for general k rather than the case $k = 1$ here. For $k = 1$ the utility functions $q_{\mathrm{F}}(\cdot)$ and $q_{\mathrm{A}}(\cdot)$ are related, but for general k it is not immediately clear what metric best captures the idea of "approximating" PCA. Developing a framework for such approximations is of interest more generally in machine learning.

References

[1] BARAK, B., CHAUDHURI, K., DWORK, C., KALE, S., MCSHERRY, F., AND TALWAR, K. Privacy, accuracy, and consistency too: a holistic solution to contingency table release. In *PODS* (2007), pp. 273–282.

[2] BLUM, A., DWORK, C., MCSHERRY, F., AND NISSIM, K. Practical privacy: the SuLQ framework. In *PODS* (2005), pp. 128–138.

[3] BLUM, A., LIGETT, K., AND ROTH, A. A learning theory approach to non-interactive database privacy. In *STOC* (2008), R. E. Ladner and C. Dwork, Eds., ACM, pp. 609–618.

[4] CHAUDHURI, K., AND HSU, D. Sample complexity bounds for differentially private learning. In *COLT* (2011).

[5] CHAUDHURI, K., AND HSU, D. Convergence rates for differentially private statistical estimation. In *ICML* (2012).

[6] CHAUDHURI, K., MONTELEONI, C., AND SARWATE, A. D. Differentially private empirical risk minimization. *Journal of Machine Learning Research 12* (March 2011), 1069–1109.

[7] CHIKUSE, Y. *Statistics on Special Manifolds*. No. 174 in Lecture Notes in Statistics. Springer, New York, 2003.

[8] DWORK, C., KENTHAPADI, K., MCSHERRY, F., MIRONOV, I., AND NAOR, M. Our data, ourselves: Privacy via distributed noise generation. In *EUROCRYPT* (2006), vol. 4004, pp. 486–503.

[9] DWORK, C., MCSHERRY, F., NISSIM, K., AND SMITH, A. Calibrating noise to sensitivity in private data analysis. In *3rd IACR Theory of Cryptography Conference,* (2006), pp. 265–284.

[10] FRIEDMAN, A., AND SCHUSTER, A. Data mining with differential privacy. In *KDD* (2010), pp. 493–502.

[11] FUNG, B. C. M., WANG, K., CHEN, R., AND YU, P. S. Privacy-preserving data publishing: A survey of recent developments. *ACM Comput. Surv. 42*, 4 (June 2010), 53 pages.

[12] GANTA, S. R., KASIVISWANATHAN, S. P., AND SMITH, A. Composition attacks and auxiliary information in data privacy. In *KDD* (2008), pp. 265–273.

[13] HAN, S., NG, W. K., AND YU, P. Privacy-preserving singular value decomposition. In *ICDE* (29 2009-april 2 2009), pp. 1267 –1270.

[14] HARDT, M., AND ROTH, A. Beating randomized response on incoherent matrices. In *STOC* (2012).

[15] HOFF, P. D. Simulation of the matrix Bingham-von Mises-Fisher distribution, with applications to multivariate and relational data. *J. Comp. Graph. Stat. 18*, 2 (2009), 438–456.

[16] KAPRALOV, M., AND TALWAR, K. On differentially private low rank approximation. In *Proc. of SODA* (2013).

[17] KASIVISWANATHAN, S. P., AND SMITH, A. A note on differential privacy: Defining resistance to arbitrary side information. *CoRR abs/0803.3946* (2008).

[18] LIU, K., KARGUPTA, H., AND RYAN, J. Random projection-based multiplicative data perturbation for privacy preserving distributed data mining. *IEEE Trans. Knowl. Data Eng. 18*, 1 (2006), 92–106.

[19] MACHANAVAJJHALA, A., KIFER, D., ABOWD, J. M., GEHRKE, J., AND VILHUBER, L. Privacy: Theory meets practice on the map. In *ICDE* (2008), pp. 277–286.

[20] MCSHERRY, F. Privacy integrated queries: an extensible platform for privacy-preserving data analysis. In *SIGMOD Conference* (2009), pp. 19–30.

[21] MCSHERRY, F., AND MIRONOV, I. Differentially private recommender systems: Building privacy into the netflix prize contenders. In *KDD* (2009), pp. 627–636.

[22] MCSHERRY, F., AND TALWAR, K. Mechanism design via differential privacy. In *FOCS* (2007), pp. 94–103.

[23] MOHAMMED, N., CHEN, R., FUNG, B. C. M., AND YU, P. S. Differentially private data release for data mining. In *KDD* (2011), pp. 493–501.

[24] NISSIM, K., RASKHODNIKOVA, S., AND SMITH, A. Smooth sensitivity and sampling in private data analysis. In *STOC* (2007), D. S. Johnson and U. Feige, Eds., ACM, pp. 75–84.

[25] STEWART, G. On the early history of the singular value decomposition. *SIAM Review 35*, 4 (1993), 551–566.

[26] WASSERMAN, L., AND ZHOU, S. A statistical framework for differential privacy. *JASA 105*, 489 (2010).

[27] WILLIAMS, O., AND MCSHERRY, F. Probabilistic inference and differential privacy. In *NIPS* (2010).

[28] ZHAN, J. Z., AND MATWIN, S. Privacy-preserving support vector machine classification. *IJIIDS 1*, 3/4 (2007), 356–385.