IMPROVED ALGORITHMS FOR DIFFERENTIALLY PRIVATE ORTHOGONAL TENSOR DECOMPOSITION

Hafiz Imtiaz and Anand D. Sarwate

Department of Electrical and Computer Engineering, Rutgers University

ABSTRACT

Tensor decompositions have applications in many areas including signal processing, machine learning, computer vision and neuroscience. In this paper, we propose two new differentially private algorithms for orthogonal decomposition of symmetric tensors from private or sensitive data; these arise in applications such as latent variable models. Differential privacy is a formal privacy framework that guarantees protections against adversarial inference. We investigate the performance of these algorithms with varying privacy and database parameters and compare against another recently proposed privacy-preserving algorithm. Our experiments show that the proposed algorithms provide very good utility even while preserving strict privacy guarantees.

***Index Terms*—** Differential privacy, orthogonal tensor decomposition, latent variable model

1. INTRODUCTION

Tensor decomposition has recently emerged as a powerful tool for inference algorithms because it can be used to infer dependencies beyond second-moment methods such as Singular Value Decomposition (SVD) or Principal Component Analysis (PCA) [1–4]. SVD or PCA operate only on sample second-moment or covariance matrices, whereas tensor decomposition exploits higher order dependencies; these are particularly useful for learning latent variable models [1].

Related Works. Some of the most well-known tensor decomposition algorithms are Tucker decomposition [5] and Canonical Polyadic Decomposition (CANDECOMP) or Parallel Factors (PARAFAC). The later two are sometimes referred together as the CP decomposition [6, 7]. These decompositions can be considered to be higher-order generalizations of the matrix SVD and PCA. Although decomposing arbitrary tensors is computationally intractable, efficient algorithms exists for finding decompositions of structured tensors. For example, the tensors that appear in several latent variable models can be efficiently decomposed [1] utilizing a variety of approaches such as generalizations of the power iteration [8].

Many signal processing and machine learning algorithms involve analyzing private or sensitive data. These algorithms may potentially leak information that could allow harmful inferences about individuals in the data. Differential privacy (DP) [9] is a strong and cryptographically-motivated framework for protecting algorithms against such inferences. In this paper, we propose two algorithms that approximate orthogonal decomposition of symmetric tensors while satisfying differential privacy. Wang and Anandkumar [10] recently proposed an algorithm for DP tensor decomposition using a noisy version of the tensor power iteration [1, 8]. We empirically show that, due to the large amount of noise introduced in their method, it requires large sample sizes to perform well.

To address this, we propose two new algorithms, AGN and AVN, for DP orthogonal tensor decomposition (OTD), both of which offer (ϵ, δ)-DP. These algorithms are inspired by input perturbation methods for differentially private PCA [11, 12]. We use the Analyze Gauss (AG) algorithm [12] at an intermediate stage of the OTD procedure. Our methods add symmetric noise to the third-order symmetric tensor, which we can obtain from the empirical third-order moment. We compare our proposed algorithms with the one proposed in [10] on a synthetic dataset and show significant improvements.

2. PROBLEM FORMULATION

Preliminaries and Notation. We refer the reader to Kolda and Bader [13] for detailed definitions of tensor terminology. We denote the set $\{1, \ldots, N\}$ by $[N]$, tensors with calligraphic scripts ($\mathcal{X}$), fibers and vectors with bold lower case letters ($\mathbf{x}$), matrices as bold upper case letters ($\mathbf{X}$), scalars with unbolded letters (M) and indices with smaller case letters. An M-way tensor $\mathcal{X} \in \mathbb{R}^{D_1 \times \ldots \times D_M}$ is rank-1 if it can be written as the outer product of M vectors: $\mathcal{X} = \mathbf{x}_1 \otimes \ldots \otimes \mathbf{x}_M$, where $\mathbf{x}_m \in \mathbb{R}^{D_m}$ and $\otimes$ denotes the outer product. A tensor $\mathcal{X}$ can be considered [1] to be a multilinear map: for a set of matrices $\{\mathbf{V}_m \in \mathbb{R}^{D_m \times K_m} : m \in [M]\}$, the $(k_1, \ldots, k_M)$-th entry in the M-way tensor representation of $\mathcal{Z} = \mathcal{X}(\mathbf{V}_1, \ldots, \mathbf{V}_M) \in \mathbb{R}^{K_1 \times \ldots \times K_M}$ is

$$\mathcal{Z}_{k_1,\ldots,k_M} = \sum_{d_1 \ldots d_M} [\mathcal{X}]_{d_1 \ldots d_M} [\mathbf{V}_1]_{d_1,k_1} \cdots [\mathbf{V}_M]_{d_M,k_M} . \quad (1)$$

The *vectorization* of the tensor $\mathcal{X}$ is defined as [14, 15]

$$\mathrm{vec}\mathcal{X} = \sum_{d_1=1}^{D_1} \cdots \sum_{d_M=1}^{D_M} [\mathcal{X}]_{d_1,\ldots,d_M} \mathbf{e}_{d_1}^{D_1} \circ \cdots \circ \mathbf{e}_{d_M}^{D_M},$$

where $\circ$ denotes the Kronecker product [13] and $\mathbf{e}^{D_m}$ denotes the D_m-dimensional elementary (or unit basis) vector. We note that $\mathrm{vec}\mathcal{X}$ is a $(\prod_{m=1}^{M} D_m)$-dimensional vector. A tensor is *symmetric* if the entries do not change under any permutation of the indices. The *rank* of a tensor $\mathcal{X}$ is the smallest number of rank-1 tensors that sums to the original tensor [4]. The *norm* $\|\mathcal{X}\|$ of a tensor $\mathcal{X}$ is the Frobenius norm, which is equal to the $\mathcal{L}_2$-norm $\|\mathrm{vec}\mathcal{X}\|_2$. We also observe that for a vector $\mathbf{x} \in \mathbb{R}^D$, if $\|\mathbf{x}\|_2 = 1$ then $\|\mathbf{x} \otimes \mathbf{x} \otimes \cdots \otimes \mathbf{x}\| = 1$ since $[\mathbf{x} \otimes \cdots \otimes \mathbf{x}]_{d_1,\ldots,d_M} = [\mathbf{x}]_{d_1} \cdots [\mathbf{x}]_{d_M}$.

The work of the authors was supported by the NSF under award CCF-1453432, by the NIH under award 1R01DA040487-01A1, and by DARPA and SSC Pacific under contract No. N66001-15-C-4070.

Orthogonal Tensor Decomposition (OTD). Let $\mathcal{X}$ be an M-way D dimensional symmetric tensor. Given real valued vectors $\mathbf{v}_k \in \mathbb{R}^D$, Comon et al. [16] showed that there exists a decomposition of the form $\mathcal{X} = \sum_{k=1}^{K} \lambda_k \mathbf{v}_k \otimes \cdots \otimes \mathbf{v}_k$. Without loss of generality, we can assume that $\|\mathbf{v}_k\|_2 = 1 \; \forall k$. If we can discover a $D \times K$ matrix $\mathbf{V} = [\mathbf{v}_1 \dots \mathbf{v}_K]$ with orthogonal columns, then we say that $\mathcal{X}$ has an *orthogonal symmetric tensor decomposition* [4]. For simplicity, we focus on $M = 3$. A unit vector $\mathbf{u} \in \mathbb{R}^D$ is an *eigenvector* of $\mathcal{X}$ with *eigenvalue* λ if $\mathcal{X}(\mathbf{I}, \mathbf{u}, \mathbf{u}) = \lambda \mathbf{u}$, where $\mathbf{I}$ is the $D \times D$ identity matrix. Now, the orthogonal tensor decomposition proposed in [1] is based on the mapping (tensor power method)

$$\mathbf{u} \mapsto \frac{\mathcal{X}(\mathbf{I}, \mathbf{u}, \mathbf{u})}{\|\mathcal{X}(\mathbf{I}, \mathbf{u}, \mathbf{u})\|_2}, \tag{2}$$

which can be considered as the tensor equivalent of the well-known matrix power method. Obviously, all tensors are not orthogonally decomposable. As the tensor power method requires the eigenvectors $\{\mathbf{v}_k\}$ to be orthonormal, we need to project the tensor on a subspace such that the eigenvectors become mutually orthogonal (*whitening*).

Applications. OTD is important for effective inference in models such as the single topic model (STM) of documents, and the mixture of Gaussians (MOG) model [1]. In this work, we consider STM model (due to page limits), where we measure the relative co-occurrence of tuples of words in a corpus of documents: this yields a 2-way-tensor ($\mathbf{M}_2 = \sum_{k=1}^{K} w_k \mathbf{a}_k \otimes \mathbf{a}_k$) for the second moment matrix of the data, and the 3-way-tensor ($\mathcal{M}_3 = \sum_{k=1}^{K} w_k \mathbf{a}_k \otimes \mathbf{a}_k \otimes \mathbf{a}_k$) for co-occurrences of 3-tuples of words. Our goal is to recover $\{w_k\}$ and $\{\mathbf{a}_k\}$ from the empirical moments: $\mathbf{M}_2 = \mathbb{E}[\mathbf{t}_1 \otimes \mathbf{t}_2]$ and $\mathcal{M}_3 = \mathbb{E}[\mathbf{t}_1 \otimes \mathbf{t}_2 \otimes \mathbf{t}_3]$, where $\mathbf{t}_l$ is the L-th word in a document. We recall that $\{w_k\}$ are the probabilities of selecting the k-th topic and $\{\mathbf{a}_k\}$ are the word probabilities given the k-th topic. We intend to estimate $\{w_k\}$ and $\{\mathbf{a}_k\}$ by performing orthogonal tensor decomposition on $\mathcal{M}_3$.

For both STM and MOG, decomposing $\mathcal{M}_3$ using the tensor power method in (2) requires the $\mathbf{a}_k$'s to be orthogonal to each other. But in general, they are not. To address this, we can project the tensor onto some subspace $\mathbf{W} \in \mathbb{R}^{D \times K}$ such that $\mathcal{M}_3(\mathbf{W}, \mathbf{W}, \mathbf{W})$ is orthogonally decomposable. From (1), we have

$$\mathcal{M}_3(\mathbf{W}, \mathbf{W}, \mathbf{W}) = \sum_{k=1}^{K} w_k (\mathbf{W}^\top \mathbf{a}_k) \otimes (\mathbf{W}^\top \mathbf{a}_k) \otimes (\mathbf{W}^\top \mathbf{a}_k).$$

To find $\mathbf{W}$, we can compute the SVD(K) of the $D \times D$ second-order moment $\mathbf{M}_2$ as $\mathbf{M}_2 = \mathbf{U}\mathbf{D}\mathbf{U}^\top$, where $\mathbf{U} \in \mathbb{R}^{D \times K}$ and $\mathbf{D} \in \mathbb{R}^{K \times K}$. Define $\mathbf{W} = \mathbf{U}\mathbf{D}^{-\frac{1}{2}}$ and then compute the projection $\tilde{\mathcal{M}}_3 = \mathcal{M}_3(\mathbf{W}, \mathbf{W}, \mathbf{W})$. The tensor $\tilde{\mathcal{M}}_3 \in \mathbb{R}^{K \times K \times K}$ is now orthogonally decomposable as the vectors $\{\mathbf{W}^\top \mathbf{a}_k\}$ are orthonormal to each other. We can utilize the tensor power method on $\tilde{\mathcal{M}}_3$ to recover the weights $\{w_k\}$ and the component vectors $\{\mathbf{a}_k\}$. The detail of the tensor power method is available in Anandkumar et al. [1].

Differentially-private OTD. In this paper, we study algorithms that approximate orthogonal decomposition of symmetric tensors, while preserving differential privacy [9]. An algorithm $\mathcal{A}(\mathbb{D})$ taking values in a set $\mathbb{T}$ provides (ϵ, δ)-differential privacy if

$$\Pr[\mathcal{A}(\mathbb{D}) \in \mathbb{S}] \leq \exp(\epsilon) \Pr[\mathcal{A}(\mathbb{D}') \in \mathbb{S}] + \delta, \tag{3}$$

for all measurable $\mathbb{S} \subseteq \mathbb{T}$ and all datasets $\mathbb{D}$ and $\mathbb{D}'$ differing in a single entry. This definition essentially states that the probability of the output of an algorithm is not changed significantly if the corresponding database input is changed by just one entry. Here, ϵ and δ are privacy parameters, where low ϵ and δ ensure more privacy. The parameter δ can be interpreted as the probability that the algorithm fails. For more details, see the recent survey [17] or monograph [18].

We note that the key step in OTD is the tensor power method mapping shown in (2). In order to ensure differential privacy for the decomposition, we may either add noise scaled to the $\mathcal{L}_2$ sensitivity [12] of the mapping operation at each iteration step or we can add noise to the tensor $\mathcal{X}$ itself just once. Adding noise in each iteration step might result in a poor accuracy of the recovered eigenvectors and eigenvalues. Therefore, we add noise to the tensor itself prior to employing the tensor power method. Recall that in the STM setup, we observe and record N documents. Let us consider two sets of documents differing in only one sample (e.g., the last one). Let the empirical second-order moment matrices be $\mathbf{M}_2$ and $\mathbf{M}_2'$ and the third-order moment tensors be $\mathcal{M}_3$ and $\mathcal{M}_3'$, respectively, for these two sets. We consider the two tensors, $\mathcal{M}_3$ and $\mathcal{M}_3'$, as *neighboring*. We observe that

$$\mathbf{M}_2 = \frac{1}{N} \sum_{n=1}^{N-1} \mathbf{t}_{1,n} \mathbf{t}_{2,n}^\top + \frac{1}{N} \mathbf{t}_{1,N} \mathbf{t}_{2,N}^\top$$

$$\mathbf{M}_2' = \frac{1}{N} \sum_{n=1}^{N-1} \mathbf{t}_{1,n} \mathbf{t}_{2,n}^\top + \frac{1}{N} \mathbf{t}'_{1,N} \mathbf{t}'^\top_{2,N},$$

where $\mathbf{t}_{l,n}$ denotes the l-th word of the n-th document. Similarly, we observe

$$\mathcal{M}_3 = \frac{1}{N} \sum_{n=1}^{N-1} \mathbf{t}_{1,n} \otimes \mathbf{t}_{2,n} \otimes \mathbf{t}_{3,n} + \frac{1}{N} \mathbf{t}_{1,N} \otimes \mathbf{t}_{2,N} \otimes \mathbf{t}_{3,N}$$

$$\mathcal{M}_3' = \frac{1}{N} \sum_{n=1}^{N-1} \mathbf{t}_{1,n} \otimes \mathbf{t}_{2,n} \otimes \mathbf{t}_{3,n} + \frac{1}{N} \mathbf{t}'_{1,N} \otimes \mathbf{t}'_{2,N} \otimes \mathbf{t}'_{3,N}.$$

We first perform SVD on $\mathbf{M}_2$ to compute $\mathbf{W}$. We use the AG algorithm [12] to make this operation differentially private. We look at the sensitivity:

$$\|\mathbf{M}_2 - \mathbf{M}_2'\|_2 = \frac{1}{N} \|\mathbf{t}_{1,N} \mathbf{t}_{2,N}^\top - \mathbf{t}'_{1,N} \mathbf{t}'^\top_{2,N}\|_2 \leq \frac{\sqrt{2}}{N} \triangleq \Delta_2.$$

The inequality follows from the encoding $\mathbf{t}_{l,n} = \mathbf{e}_d$ and recalling that at-most two entries in the difference term can be non-zero. We can therefore add i.i.d. Gaussian noise with variance scaled to Δ_2 to $\mathbf{M}_2$ to make the computation of $\mathbf{W}$ satisfy (ϵ_1, δ_1) differential privacy. Now, we need to project $\mathcal{M}_3$ on $\mathbf{W}$ before using the tensor power method. We can choose between making the projection operation differentially private, or we can make the $\mathcal{M}_3$ itself differentially private before projection. We found that making the projection differentially private involves addition of a large amount of noise and more importantly, the variance of the noise depends on the top-K singular values of $\mathbf{M}_2$. Therefore, we choose to make the tensor itself differentially private. To find the sensitivity of the tensor valued function $f(\mathcal{M}_3) = \mathcal{M}_3$, we observe:

$$\|\mathcal{M}_3 - \mathcal{M}_3'\| = \frac{1}{N} \|\mathbf{t}_{1,N} \otimes \mathbf{t}_{2,N} \otimes \mathbf{t}_{3,N} - \mathbf{t}'_{1,N} \otimes \mathbf{t}'_{2,N} \otimes \mathbf{t}'_{3,N}\|$$
$$\leq \frac{\sqrt{2}}{N} \triangleq \Delta_3.$$

Again, the inequality follows from the encoding $\mathbf{t}_{l,n} = \mathbf{e}_d$ and realizing that only two entries in the difference term can be non-zero, each with value at most 1.

Algorithm 1: AGN / AVN Algorithm

Input : Sample second-order moment matrix $\mathbf{M}_2 \in \mathbb{R}^{D\times D}$ and third-order moment tensor $\mathcal{M}_3 \in \mathbb{R}^{D\times D\times D}$, privacy parameters $\epsilon_1, \epsilon_2, \delta_1, \delta_2$

1 Generate $D \times D$ symmetric matrix $\mathbf{E}$ where $\{E_{ij} : i \in [D], j \leq i\}$ drawn i.i.d. from $\mathcal{N}(0, \tau_1^2)$, where

$$E_{ij} = E_{ji} \text{ and } \tau_1 = \begin{cases} \frac{\Delta_2}{\epsilon_1}\sqrt{2\log\left(\frac{1.25}{\delta_1}\right)}, \text{ for AGN} \\ \frac{\Delta_2}{\epsilon_1}\sqrt{2\log\left(\frac{1.25}{\delta_1+\delta_2}\right)}, \text{ for AVN} \end{cases}$$

2 Compute $\hat{\mathbf{M}}_2 \leftarrow \mathbf{M}_2 + \mathbf{E}$

3 Compute SVD(K) on $\hat{\mathbf{M}}_2 = \mathbf{U}\mathbf{D}\mathbf{U}^\top$

4 Compute $\mathbf{W} = \mathbf{U}\mathbf{D}^{-\frac{1}{2}}$

5 Draw a vector $\mathbf{b} \in \mathbb{R}^{D_{\text{sym}}}$:

$$\mathbf{b} \sim \begin{cases} \mathcal{N}(0, \tau_2^2\mathbf{I}),\ \tau_2 = \frac{\Delta_3}{\epsilon_2}\sqrt{2\log\left(\frac{1.25}{\delta_2}\right)} \text{ for AGN} \\ f_b(\mathbf{b}) = \frac{1}{\alpha}\exp\left(-\beta\|\mathbf{b}\|_2\right),\ \beta = \frac{\epsilon_2}{\Delta_3} \text{ for AVN} \end{cases}$$

6 Generate symmetric $\mathcal{E} \in \mathbb{R}^{D\times D\times D}$ from entries of $\mathbf{b}$

7 Compute $\hat{\mathcal{M}}_3 \leftarrow \mathcal{M}_3 + \mathcal{E}$

8 Compute $\tilde{\mathcal{M}}_3 \leftarrow \hat{\mathcal{M}}_3(\mathbf{W}, \mathbf{W}, \mathbf{W})$

Output: Private orthogonally decomposable tensor $\tilde{\mathcal{M}}_3$, projection subspace $\mathbf{W}$

3. ALGORITHMS

In this section, we describe two new algorithms for (ϵ, δ) differentially private orthogonal tensor decomposition: AGN and AVN. Both of the algorithms first utilize the AG algorithm [12] to compute a differentially private approximate to $\mathbf{M}_2$. The $\mathcal{L}_2$ sensitivity of $\mathbf{M}_2$ is given by Δ_2. We generate a $D \times D$ symmetric matrix $\mathbf{E}$ (Step 1 of Algorithm 1). By computing the SVD(K) of $\hat{\mathbf{M}}_2$ (the DP approximate to $\mathbf{M}_2$), we find the subspace $\mathbf{W}$ required for whitening and also for recovering $\{\mathbf{a}_k\}$. We note that this step is (ϵ_1, δ_1)-differentially private for AGN and $(\epsilon_1, \delta_1+\delta_2)$-differentially private for AVN. Next, we draw a $D_{\text{sym}} = \binom{D+2}{3}$-dimensional vector $\mathbf{b}$ (Step 5 of Algorithm 1). The proposed algorithms differ in this step, i.e., they differ in the distribution from which $\mathbf{b}$ is sampled. To preserve the symmetry of $\mathcal{M}_3$ upon noise addition, we form a symmetric tensor $\mathcal{E} \in \mathbb{R}^{D\times D\times D}$ from the entries of $\mathbf{b}$. Then we compute $\hat{\mathcal{M}}_3 = \mathcal{M}_3 + \mathcal{E}$. This is the (ϵ_2, δ_2)-differentially private approximate to $\mathcal{M}_3$ for AGN (and $(\epsilon_2, 0)$ for AVN). Finally, we project $\hat{\mathcal{M}}_3$ on the subspace $\mathbf{W}$ to get the orthogonally decomposable tensor $\tilde{\mathcal{M}}_3$. The overall procedure is (ϵ, δ)-differentially private, where $\epsilon = \epsilon_1 + \epsilon_2$ and $\delta = \delta_1 + \delta_2$. The detailed procedure is shown in Algorithm 1. In the density $f_b(\mathbf{b})$, α is a normalizing constant and $\beta = \frac{\epsilon_2}{\Delta_3}$ is a parameter of the density. We do not need to specify α because sampling from $f_b(\mathbf{b})$ can be performed without any knowledge of α. The sampling procedure is omitted due to page limits. Note that although the proposed algorithms differ in one step (Step 5), the implications are further-reaching. With AVN, the computation of $\hat{\mathcal{M}}_3$ is pure ϵ_2-DP. Therefore, if one uses an ϵ_1-DP algorithm for Step 3, or if the tensor is already orthogonally decomposable (i.e., no need for whitening), then the AVN algorithm would provide a pure ϵ-DP algorithm for OTD.

Theorem 1 (Privacy of AGN and AVN Algorithms)**.** *Algorithm 1 computes the orthogonally decomposable tensor $\tilde{\mathcal{M}}_3$ with $(\epsilon_1 + \epsilon_2, \delta_1 + \delta_2)$-differential privacy for both* AGN *and* AVN.

Proof sketch. Proof of Theorem 1 for AGN follows from using the Gaussian mechanism [9] and the sensitivities of $\mathbf{M}_2$ and $\mathcal{M}_3$ and considering that a D-dimensional M-mode symmetric tensor is fully determined by $D_{\text{sym}} = \binom{D+M-1}{M}$ entries [16]. For the privacy of AVN, consider the algorithm $\mathcal{Y} = \mathcal{M}_3 + \mathcal{E}$, where $\mathcal{E}$ is a symmetric tensor. $\mathcal{E}$ consists $D_{\text{sym}} = \binom{D+2}{3}$ number of unique entries. We draw a vector $\mathbf{b} \in \mathbb{R}^{D_{\text{sym}}}$ according to the density defined by $f_b(\mathbf{b})$ [19] and then form $\mathcal{E}$ from the entries of $\mathbf{b}$. The probability of the event of drawing a particular sample from $f_b(\mathbf{b})$ is the same as drawing a symmetric tensor with the same unique entries as the aforementioned vector from some equivalent density on symmetric tensors. Now, we look at the ratio of the densities:

$$\frac{f(\mathcal{Y}|\mathcal{M}_3)}{f(\mathcal{Y}|\mathcal{M}_3')} = \frac{\exp\left(-\beta\|\text{vec}\mathcal{Y} - \text{vec}\mathcal{M}_3\|_2\right)}{\exp\left(-\beta\|\text{vec}\mathcal{Y} - \text{vec}\mathcal{M}_3'\|_2\right)} \leq \exp\left(\beta\|\mathcal{M}_3' - \mathcal{M}_3\|\right) \leq \exp\left(\beta\Delta_3\right),$$

where the inequality is introduced using the triangle inequality of norms. Therefore, the algorithm $\mathcal{Y} = \mathcal{M}_3 + \mathcal{E}$ is $(\epsilon_2, 0)$-differentially private by setting $\beta = \frac{\epsilon_2}{\Delta_3}$. □

Theoretical Performance Guarantee. Although we are adding symmetric noise to the third-order moment tensor, an orthogonal decomposition need not to exist for the perturbed tensor, even though the perturbed tensor is symmetric [1, 4]. Anandkumar et al. [1] provided a bound on the error of the recovered decomposition in terms of the operator norm of the tensor perturbation. For our proposed algorithms, the perturbation includes the sampling error as well as the differential-privacy noise. Even without accounting for the sampling error, for both of our proposed algorithms, the operator norm of the added noise $\|\mathcal{E}\|_{\text{op}}$ is a random quantity, and requires new measure concentration results to analyze. Relating these bounds to the error in estimating $\{\mathbf{a}_k\}$ and $\{w_k\}$ is nontrivial and we defer this for future work.

Prior Work: Tensor Power Iteration. To the best of our knowledge, only one algorithm is proposed for differentially private OTD [10] with (ϵ, δ)-differential privacy. It requires that the input to the tensor power method be orthogonally decomposable and adds Gaussian noise at each step of the tensor power iteration and also while computing the eigenvalues. We demonstrate next that our methods significantly outperform this approach.

4. EXPERIMENTAL RESULTS

Because the algorithms have a large parameter space, we focus on measuring how well the outputs of these algorithms approximate the true components $\{\mathbf{a}_k\}$ and $\{w_k\}$. Let the recovered component vectors be $\{\hat{\mathbf{a}}_k\}$. To capture the disparity between $\{\mathbf{a}_k\}$ and $\{\hat{\mathbf{a}}_k\}$, we define an error metric: $e_{\text{comp}} = \frac{1}{K}\sum_{k=1}^{K}\gamma_{\min}^k$, where $\gamma_{\min}^k = \min_{k'\in[K]}\|\hat{\mathbf{a}}_k - \mathbf{a}_{k'}\|_2$. A similar measure is used in the dictionary learning literature [20]. For comparison, we show the error resulting from the $\hat{\mathbf{a}}_k$'s achieved from the two proposed methods, the DP-TPM [10] and the non-private method [1]. We also show the error considering random vectors as $\{\hat{\mathbf{a}}_k\}$ because this error corresponds to the worst possible results (i.e., not considering any information from data). As recovering $\{\hat{\mathbf{a}}_k\}$ is closely related with recovering $\{w_k\}$ (Section 4.3.1 in [1]), we only show the error associated with recovering $\{\hat{\mathbf{a}}_k\}$. In all cases we show the average performance over 10 runs of each algorithm. For both the AGN and AVN algorithms, there are two stages where we add noise to ensure

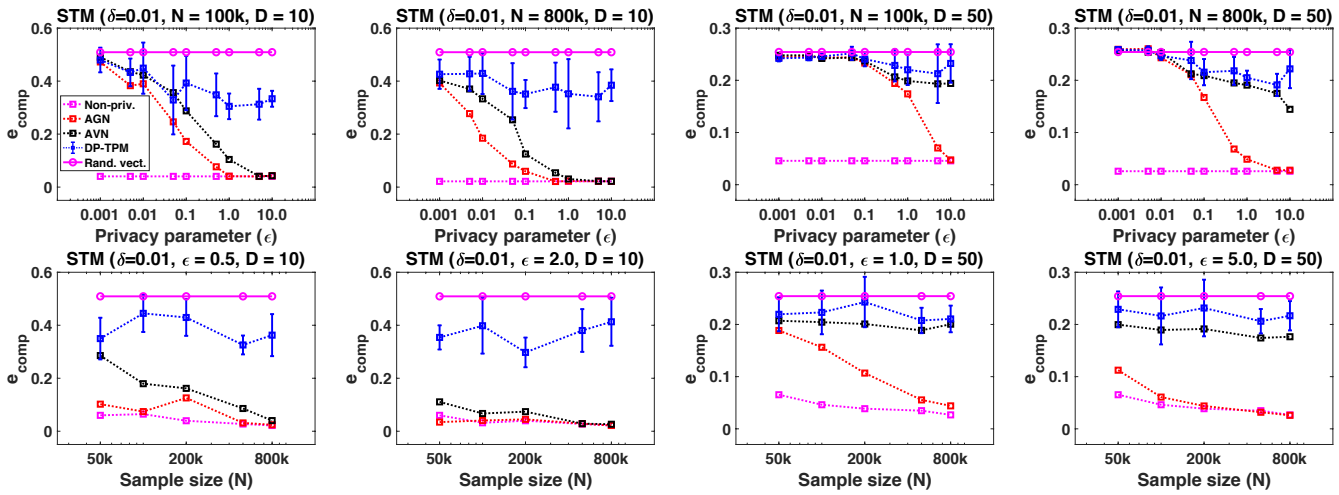

Fig. 1. Variation of error for the STM application using synthetic data. Top-row: with ϵ. Bottom-row: with N

differential-privacy. We equally divided ϵ to set ϵ_1 and ϵ_2 for the two stages. For the AGN, we equally divided δ to set δ_1 and δ_2. However, for the AVN, only the first stage requires δ. Optimal allocation of ϵ and δ in multi-stage algorithms is still an open question. We note that the neighborhood definition of $\mathcal{M}_3$ is a bit different in [10]. While implementing the DP-TPM [10], we modified the algorithm slightly (specifically, the assignment of ν in Algorithm 3 of [10]). In our definition $\Delta_2 f_1 \leq 2\|\mathbf{u}\|_\infty^2$ and $\Delta_2 f_2 \leq 2\|\mathbf{u}\|_\infty^3$. Therefore, we set $\nu = \frac{2}{\varepsilon'}\sqrt{2\log\left(\frac{1.25}{\delta'}\right)}$.

Performance Variation in the STM Setup. We performed experiments on two *synthetic* datasets of different feature dimensions ($D = 10$, $K = 5$ and $D = 50$, $K = 10$) generated with predetermined $\mathbf{w}$ and $\{\mathbf{a}_k\}$. It should be noted here that the recovery of $\{\mathbf{a}_k\}$ is difficult, because the recovered word probabilities from the tensor decomposition, whether private or non-private, may not always be valid probability vectors (i.e., no negative entries and sum to 1). Therefore, prior to computing the e_{comp}, we ran a post-processing step (0-out negative entries and then normalize by summation) to ensure that the recovered vectors are valid probability vectors. This process is non-linear and potentially makes the recovery error worse. However, for practical STM, D is not likely to be 10 or 50, rather it may be of the order of thousands, simulating which is a huge computational burden. In general, if we want the same privacy level for higher dimensional data, we need to increase the sample size. We refer the reader to some efficient (but non-differentially private) implementations [21].

Performance Variation with ϵ. We first explore the *privacy-utility tradeoff* between ϵ and e_{comp}. In the top-row of Figure 1, we show the variation of e_{comp} with ϵ for a fixed δ for two different feature dimensions. For both of the feature dimensions, we observe that as ϵ increases (higher privacy risk), the errors decrease. The proposed methods outperform the DP-TPM [10] and match the performance of the non-private method for large enough ϵ. The proposed AGN algorithm outperforms all others in all settings. The AVN algorithm performs slightly worse than the AGN for $D = 10$, but still much better than the DP-TPM. We observe that the performance gap between the proposed AVN algorithm and the DP-TPM algorithm is smaller for $D = 50$ than for $D = 10$. This can be explained in the following way: as D increases, the length of the vector $\mathbf{b} \in \mathbb{R}^{D_{\text{sym}}}$ increases non-linearly. This D_{sym} is in fact the *shape parameter* of the Erlang random variable $\|\mathbf{b}\|_2$. The variance of the Erlang random variable increases linearly with the shape parameter. Therefore, as D increases (with a fixed ϵ and N), the variance of the noise added according to $f_b(\mathbf{b})$ increases and thereby deteriorates the performance. However, increasing N makes the proposed algorithms perform better. We show the error bars for the DP-TPM to demonstrate the instability of the performance. We believe this is because the amount of added noise at each step for the DP-TPM is too high for a stable output even for larger ϵ.

Performance Variation with N. The bottom-row of Figure 1 shows how the errors vary as a function of N for two different feature dimensions, while keeping ϵ and δ fixed. The variation with N reiterates the results seen earlier. The proposed algorithms outperform the DP-TPM [10] by a large margin for $D = 10$. We observe that for $D = 50$, the AVN algorithm, performs only slightly better than the DP-TPM, which is unlike the situation for $D = 10$. This can again be attributed to the fact that as D increases, the length of the vector $\mathbf{b} \in \mathbb{R}^{D_{\text{sym}}}$ increases, which increases the variance of the added noise. Between the proposed algorithms, the AGN performs better than the AVN algorithm, just as before. For larger N, it achieves almost the same utility as the non-private algorithm. Even for smaller ϵ with a proper sample size, the error is very low. For the $D = 10$ case, the AGN always performs very closely with the non-private algorithm. For larger ϵ and smaller D, the AVN performs similarly as AGN.

5. CONCLUSION

In this paper, we proposed two new algorithms for differentially private orthogonal tensor decomposition. We empirically compared the performance of the proposed algorithms with that of the recently proposed differentially private orthogonal tensor decomposition algorithm on synthetic datasets, while varying relevant dataset and algorithm parameters. In general, the AGN and the AVN algorithms demonstrated better performance than the DP-TPM [10]. The proposed algorithms offered very good utility even for strong privacy guarantees and matched the utility of non-private orthogonal tensor decomposition for some parameter choices. Our initial results suggest that the asymptotic guarantees for differentially private algorithms, such as the one proposed by Wang and Anandkumar [10], may not always reflect their empirical performance.

6. REFERENCES

[1] A. Anandkumar, R. Ge, D. Hsu, S. M. Kakade, and M. Telgarsky, "Tensor Decompositions for Learning Latent Variable Models," *J. Mach. Learn. Res.*, vol. 15, no. 1, pp. 2773–2832, Jan. 2014.

[2] D. J. Hsu and S. M. Kakade, "Learning Mixtures of Spherical Gaussians: Moment Methods and Spectral Decompositions," *Proceedings of the 4th Conference on Innovations in Theoretical Computer Science*, pp. 11–20, 2013.

[3] D. Hsu, S. M. Kakade, and T. Zhang, "A Spectral Algorithm for Learning Hidden Markov Models," *Journal of Computer and System Sciences*, vol. 78, no. 5, pp. 1460 – 1480, 2012.

[4] T. G. Kolda, "Symmetric Orthogonal Tensor Decomposition is Trivial," in *eprint arXiv:1503.01375*, 2015.

[5] L. R. Tucker, "Some Mathematical Notes on Three-mode Factor Analysis," *Psychometrika*, vol. 31, no. 3, pp. 279–311, 1966.

[6] J. D. Carroll and Jih-Jie Chang, "Analysis of Individual Differences in Multidimensional Scaling via an n-way Generalization of "Eckart-Young" Decomposition," *Psychometrika*, vol. 35, no. 3, pp. 283–319, 1970.

[7] R. A. Harshman, "Foundations of the PARAFAC Procedure: Models and Conditions for an 'explanatory' Multi-modal Factor Analysis," *UCLA Working Papers in Phonetics*, vol. 16, no. 1, 1970.

[8] L.D. Lathauwer, B. D. Moor, and J. Vandewalle, "On the Best Rank-1 and Rank-(R1,R2,. . .,RN) Approximation of Higher-Order Tensors," *SIAM J. Matrix Anal. Appl.*, vol. 21, no. 4, pp. 1324–1342, Mar. 2000.

[9] C. Dwork, F. McSherry, K. Nissim, and A. Smith, "Calibrating Noise to Sensitivity in Private Data Analysis," in *Proceedings of the Third Conference on Theory of Cryptography*, 2006, pp. 265–284.

[10] Y. Wang and A. Anandkumar, "Online and Differentially-Private Tensor Decomposition," *ArXiv e-prints*, June 2016.

[11] A. Blum, C. Dwork, F. McSherry, and K. Nissim, "Practical Privacy: The SuLQ Framework," in *Proceedings of the Twenty-fourth ACM SIGMOD-SIGACT-SIGART Symposium on Principles of Database Systems*, 2005, pp. 128–138.

[12] C. Dwork, K. Talwar, A. Thakurta, and L. Zhang, "Analyze Gauss: Optimal Bounds for Privacy-preserving Principal Component Analysis," in *Proceedings of the 46th Annual ACM Symposium on Theory of Computing*, 2014, pp. 11–20.

[13] T. G. Kolda and B. W. Bader, "Tensor Decompositions and Applications," *SIAM REVIEW*, vol. 51, no. 3, pp. 455–500, 2009.

[14] M. Singulla, M. R. Ahmad, and D. von Rosen, "More on the Kronecker Structured Covariance Matrix," *Communications in Statistics - Theory and Methods*, vol. 41, no. 13-14, pp. 2512–2523, 2012.

[15] M. Ohlson, M. R. Ahmad, and D. von Rosen, "The Multilinear Normal Distribution: Introduction and Some Basic Properties," *Journal of Multivariate Analysis*, vol. 113, pp. 37 – 47, 2013.

[16] P. Comon, G. Golub, L.-H. Lim, and B. Mourrain, "Symmetric Tensors and Symmetric Tensor Rank," *SIAM Journal on Matrix Analysis and Applications*, vol. 30, no. 3, pp. 12541279, 2008.

[17] A. D. Sarwate and K. Chaudhuri, "Signal Processing and Machine Learning with Differential Privacy: Theory, Algorithms, and Challenges," *IEEE Signal Processing Magazine*, vol. 30, no. 5, pp. 86–94, September 2013.

[18] C. Dwork and A. Roth, "The Algorithmic Foundations of Differential Privacy," *Foundations and Trends in Theoretical Computer Science*, vol. 9, no. 3-4, pp. 211–407, 2013.

[19] K. Chaudhuri, C. Monteleoni, and A. D. Sarwate, "Differentially Private Empirical Risk Minimization," *J. Mach. Learn. Res.*, vol. 12, pp. 1069–1109, July 2011.

[20] H. V. Nguyen, V. M. Patel, N. M. Nasrabadi, and R. Chellappa, "Sparse Embedding: A Framework for Sparsity Promoting Dimensionality Reduction," in *Proceedings of the 12th European Conference on Computer Vision - Volume Part VI*. 2012, ECCV'12, pp. 414–427, Springer-Verlag.

[21] F. Huang, S. Matusevych, A. Anandkumar, N. Karampatziakis, and P. Mineiro, "Distributed Latent Dirichlet Allocation via Tensor Factorization," in *NIPS Optimization Workshop*, 2014.