Improved Algorithms for Differentially-private Orthogonal Tensor Decomposition

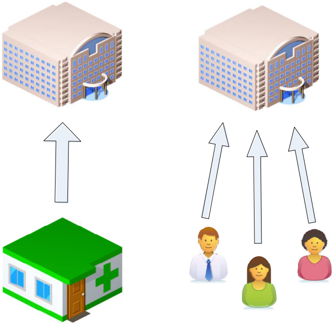
Hafiz Imtiaz & Anand D. Sarwate

Department of Electrical and Computer Engineering
Rutgers, the State University of New Jersey

April 17, 2018

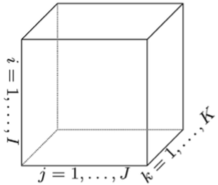
Outline

Why learn from private data?



- Much of private/sensitive data is being digitized
- Want to learn about population – using/reusing data
- Free and open sharing – ethical, legal, and technological obstacles

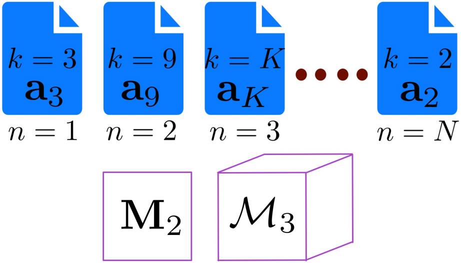
Why use tensors?



- Can infer dependencies beyond second-moment methods (e.g. PCA)
- Some parameter estimation problems can be posed as tensor decomposition problems
- More suited for learning latent variable models[AGHKT14]

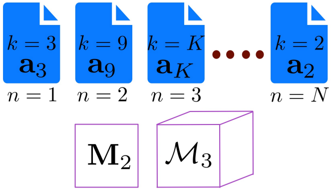
[1] Figure from [KB09]

Single topic model (STM)



- Hidden variable k – specifying the sole topic of a document
- k can take K distinct values with probability $\mathbb{P}[h = k] = w_k$
- Observe N documents, each with $L \geq 3$ words
- Given k, words are drawn independently $\sim \mathbf{a}_k \in \mathbb{R}^D$
- D is the alphabet size
- Words $\mathbf{t}_{l,n} \in \mathbb{R}^D$ represented using one-hot encoding

Single topic model



The way we record what we observe is: we form an $D \times D \times D$ tensor whose (d_1, d_2, d_3)-th entry is the proportion of times we see a document with first word d_1, second word d_2 and third word d_3.

$$\mathbf{M}_2 = \frac{1}{N} \sum_{n=1}^{N} \mathbf{t}_{1,n} \otimes \mathbf{t}_{2,n}, \quad \mathcal{M}_3 = \frac{1}{N} \sum_{n=1}^{N} \mathbf{t}_{1,n} \otimes \mathbf{t}_{2,n} \otimes \mathbf{t}_{3,n}$$

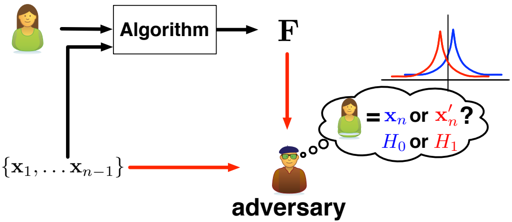
Single topic model

Define two population moments in terms of $\mathbf{a}_k$ and $\{w_k\}$

$$\mathbf{M}_2 = \sum_{k=1}^{K} w_k \mathbf{a}_k \otimes \mathbf{a}_k, \; \mathcal{M}_3 = \sum_{k=1}^{K} w_k \mathbf{a}_k \otimes \mathbf{a}_k \otimes \mathbf{a}_k.$$

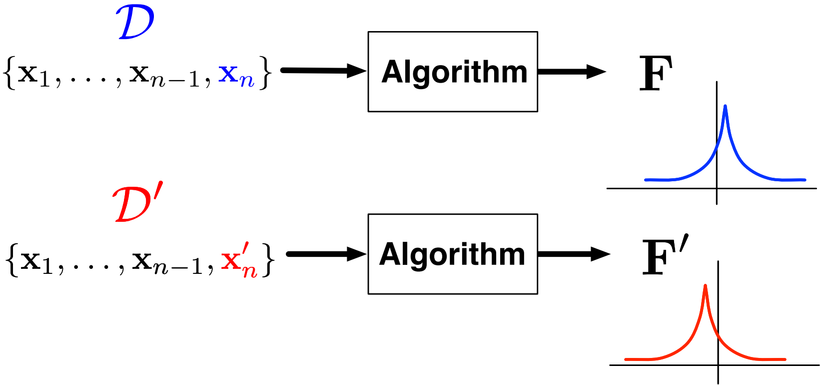
These can be estimated from the samples.

Goal: recover $\{w_k\}$ and $\{\mathbf{a}_k\}$



Differential Privacy

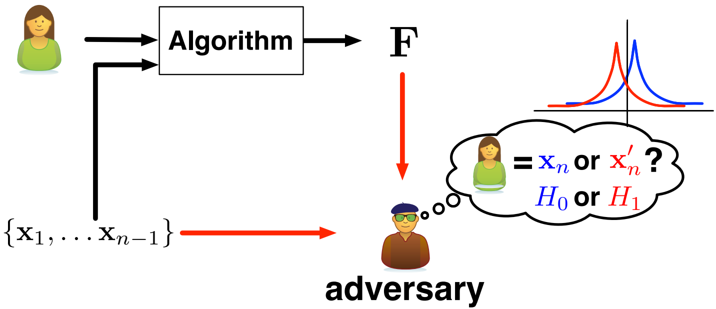
Differential privacy: a definition



[Dwork et al. 2006] An algorithm $\mathcal{A}$ is (ε, δ)-differentially private if for any set of outputs $\mathcal{F}$, and all $(\mathcal{D}, \mathcal{D}')$ differing in a single point,

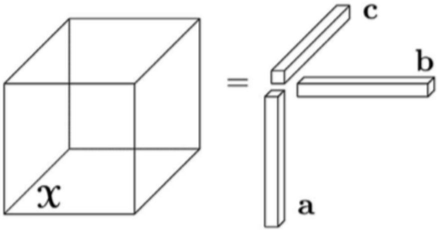
$$\mathbb{P}\left(\mathcal{A}(\mathcal{D}) \in \mathcal{F}\right) \leq \exp(\varepsilon) \cdot \mathbb{P}\left(\mathcal{A}(\mathcal{D}') \in \mathcal{F}\right) + \delta$$

Differential privacy: hypothesis testing



$$\log \frac{\mathbb{P}\left(\mathcal{A}(\mathcal{D}) \in \mathcal{F}\right)}{\mathbb{P}\left(\mathcal{A}(\mathcal{D}') \in \mathcal{F}\right)} \leq \varepsilon$$

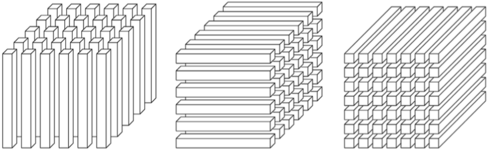
We want to design algorithms that satisfy differential privacy



Tensor Basics

[1] Figure from [KB09]

Modes and fibers



Definition

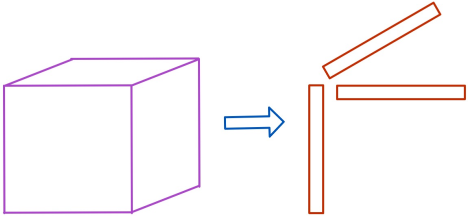
An M-th order tensor is an element of the tensor product of M vector spaces.

Definition

Fiber is higher order analog of row/column and is defined by fixing every index but one.

[1]Figure from [KB09]

Outer product and rank



- Consider a vector: $\mathbf{x}_m \in \mathbb{R}^{D_m}$. Then the M-way outer product is:

$$[\mathbf{x}_1 \otimes \mathbf{x}_2 \otimes \cdots \otimes \mathbf{x}_M]_{d_1,d_2,\ldots,d_M} = [\mathbf{x}_1]_{d_1}[\mathbf{x}_2]_{d_2}\cdots[\mathbf{x}_M]_{d_M}$$

- An M-way tensor $\mathcal{X} \in \mathbb{R}^{D_1 \times D_2 \times \ldots \times D_M}$ is rank-1 if:

$$\mathcal{X} = \mathbf{x}_1 \otimes \mathbf{x}_2 \otimes \ldots \otimes \mathbf{x}_M$$

Projecting tensors on matrices

- Consider the M-mode tensor: $\mathcal{X} \in \mathbb{R}^{D_1 \times D_2 \times \ldots \times D_M}$
- And a set of matrices $\{\mathbf{V}_m \in \mathbb{R}^{D_m \times K_m} : m = 1, 2, \ldots, M\}$
- We can project each mode of $\mathcal{X}$ on corresponding $\mathbf{V}_m$ as to get $\mathcal{X}\left(\mathbf{V}_1, \mathbf{V}_2, \ldots, \mathbf{V}_M\right) \in \mathbb{R}^{K_1 \times K_2 \times \ldots \times K_M}$:

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

This is the multilinear mapping [AGHKT14].

Orthogonal Decomposition of Tensors

Symmetric tensors

Definition

A tensor is **symmetric** if the entries do not change under any permutation of the indices.

Orthogonal Decomposition of Symmetric Tensors

- $\mathcal{X} \rightarrow M$-way D dimensional symmetric tensor
- There exists a decomposition [CGLM08]:

$$\mathcal{X} = \sum_{k=1}^{K} \lambda_k \mathbf{v}_k \otimes \mathbf{v}_k \otimes \cdots \otimes \mathbf{v}_k$$

- WLOG, assume that $\mathbf{v}_k \in \mathbb{R}^D$ have $\mathcal{L}_2$ norm at-most 1
- $\mathcal{X}$ is ODECO if we can find $\mathbf{V}$ with orthogonal columns [K15]: $\mathbf{V} = [\mathbf{v}_1 \;\; \mathbf{v}_2 \;\; \ldots \mathbf{v}_K] \in \mathbb{R}^{D \times K}$

Eigenvectors of ODECO tensors

Definition

A unit vector $\mathbf{u} \in \mathbb{R}^D$ is an **eigenvector** of $\mathcal{X}$ with corresponding **eigenvalue** λ if

$$\mathcal{X}(\mathbf{I}, \mathbf{u}, \mathbf{u}) = \lambda \mathbf{u}$$

- $\mathcal{X}$ is ODECO $\implies$ $\mathbf{v}_k$'s are orthogonal to each other
- So, $\mathcal{X}(\mathbf{I}, \mathbf{v}_k, \mathbf{v}_k) = \lambda_k \mathbf{v}_k$ for all $k = 1, 2, \ldots, K$

Tensor power method

$\mathcal{X}$ is ODECO $\implies$ we can find its eigenvectors and eigenvalues using:

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

- Not all tensors are ODECO – even if they are symmetric
- We need to perform **whitening** – project the tensor on a subspace such that the eigenvectors become orthogonal to each other

Recall the STM problem

$$\mathbf{M}_2 = \sum_{k=1}^{K} w_k \mathbf{a}_k \otimes \mathbf{a}_k \approx \frac{1}{N} \sum_{n=1}^{N} \mathbf{t}_{1,n} \otimes \mathbf{t}_{2,n} \mathcal{M}_3 = \sum_{k=1}^{K} w_k \mathbf{a}_k \otimes \mathbf{a}_k \otimes \mathbf{a}_k \approx \frac{1}{N}$$

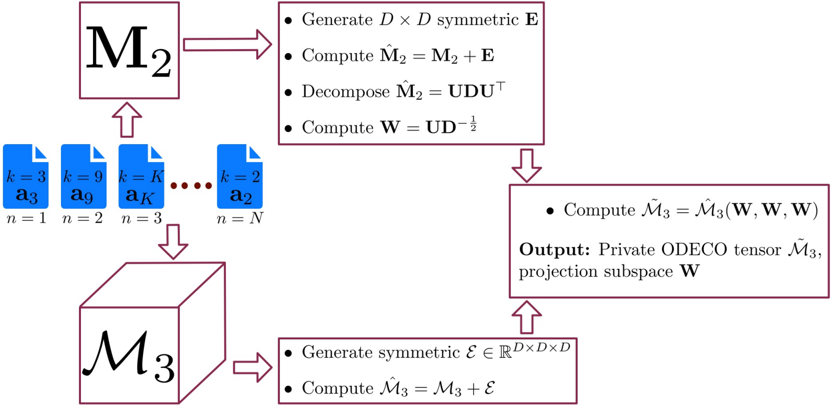
- Have sample estimates of $\mathbf{M}_2$ and $\mathcal{M}_3$
- Want to recover $\{w_k\}$ and $\{\mathbf{a}_k\}$
- Problem: $\mathcal{M}_3$ not ODECO in general
- Idea: use $\mathbf{M}_2$ to find a *good* projection subspace

Finding a subspace

- Goal: find $\mathbf{W} \in \mathbb{R}^{D \times K}$ to ensure $\mathcal{M}_3\left(\mathbf{W}, \mathbf{W}, \mathbf{W}\right)$ is ODECO $\implies$ $\mathbf{W}^\top \mathbf{a}_k$'s are orthogonal to each other
- How? Perform SVD on $\mathbf{M}_2$: $\mathbf{M}_2 = \mathbf{U}\mathbf{D}\mathbf{U}^\top$
- $\mathbf{U} \in \mathbb{R}^{D \times K}$ and $\mathbf{D} \in \mathbb{R}^{K \times K}$
- $\mathbf{W} = \mathbf{U}\mathbf{D}^{-\frac{1}{2}} \in \mathbb{R}^{D \times K}$
- Compute:

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

$\tilde{\mathcal{M}}_3 \in \mathbb{R}^{K \times K \times K}$ is ODECO $\implies$ so we can recover $\{w_k\}$ and $\{\mathbf{a}_k\}$



Proposed Algorithm

Differentially-private OTD (AGN / AVN)

Input: $\mathbf{M}_2 \in \mathbb{R}^{D\times D}$, $\mathcal{M}_3 \in \mathbb{R}^{D\times D\times D}$; parameters ϵ_1, ϵ_2, δ_1, δ_2

- Generate $D \times D$ symmetric $\mathbf{E}$ with $\{E_{ij} : i \in [D], j \leq i\}$ drawn i.i.d. from $\mathcal{N}(0, \tau_1^2)$ and $\tau_1 = \begin{cases} \frac{\sqrt{2}}{N\epsilon_1}\sqrt{2\log\left(\frac{1.25}{\delta_1}\right)}, & \text{for AGN} \\ \frac{\sqrt{2}}{N\epsilon_1}\sqrt{2\log\left(\frac{1.25}{\delta_1+\delta_2}\right)}, & \text{for AVN} \end{cases}$
- Compute $\mathbf{W} = \mathbf{U}\mathbf{D}^{-\frac{1}{2}}$, where $\mathbf{U}\mathbf{D}\mathbf{U}^\top = \mathbf{M}_2 + \mathbf{E}$
- Draw a vector $\mathbf{b} \in \mathbb{R}^{D_{\text{sym}}}$ and generate symmetric $\mathcal{E} \in \mathbb{R}^{D\times D\times D}$ from $\mathbf{b}$: $\mathbf{b} \sim \begin{cases} \mathcal{N}(0, \tau_2^2\mathbf{I}), \ \tau_2 = \frac{\sqrt{2}}{N\epsilon_2}\sqrt{2\log\left(\frac{1.25}{\delta_2}\right)} & \text{for AGN} \\ f_b(\mathbf{b}) = \frac{1}{\alpha}\exp\left(-\beta\|\mathbf{b}\|_2\right), \ \beta = \frac{N\epsilon_2}{\sqrt{2}} & \text{for AVN} \end{cases}$
- Compute $\tilde{\mathcal{M}}_3 \leftarrow (\mathcal{M}_3 + \mathcal{E})(\mathbf{W}, \mathbf{W}, \mathbf{W})$

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

A closer look at AGN / AVN

- Two quantities involve data – $\mathbf{W}$ and $\mathcal{M}_3$
- $\mathbf{W}$ needs to satisfy privacy – required for projection and computing $\{\mathbf{a}_k\}$
- Modifying the projection $(\mathcal{M}_3 + \mathcal{E})(\mathbf{W}, \mathbf{W}, \mathbf{W})$ to satisfy privacy is hard – large sensitivity
- AGN and AVN differs in the distribution $\mathbf{b}$ is sampled from
- However, the implications are further reaching – "pure" ϵ-DP mechanisms

Privacy guarantee of AGN and AVN Algorithms

Theorem (Privacy of AGN and AVN Algorithms)

Both AGN *and* AVN *algorithms compute the orthogonally decomposable tensor* $\tilde{\mathcal{M}}_3$ *with* $(\epsilon_1 + \epsilon_2, \delta_1 + \delta_2)$.

- $\mathcal{L}_2$ sensitivities of both $\mathbf{M}_2$ and $\mathcal{M}_3$ are $\frac{\sqrt{2}}{N}$
- By AG [Dwork et al. 2014] algorithm: computation of $\mathbf{M}_2 + \mathbf{E}$ is differentially private
- For AGN: Gaussian mechanism [Dwork et al. 2013] ensures the computation of $\mathcal{M}_3 + \mathcal{E}$ is DP
- For AVN: using the density $f_b(\mathbf{b})$ in the definition of DP shows the computation of $\mathcal{M}_3 + \mathcal{E}$ is DP
- Differential-privacy is invariant to post-processing: computation of $(\mathcal{M}_3 + \mathcal{E})(\mathbf{W}, \mathbf{W}, \mathbf{W})$ satisfies (ϵ, δ) differential privacy

Experimental Results

Dataset and performance measure

Datasets

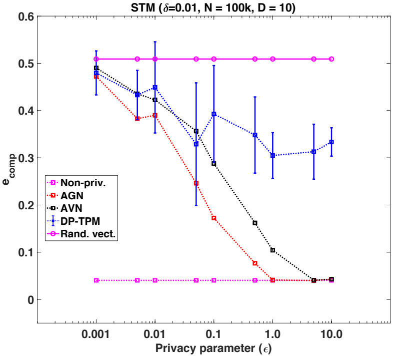
- Synthetic dataset1: $(D = 10,\ K = 5)$
- Synthetic dataset2: $(D = 50,\ K = 10)$

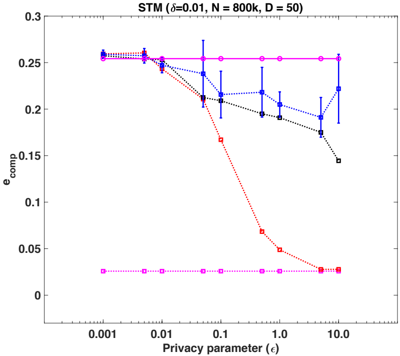
Performance measure

- True components: $\{\mathbf{a}_k\}$; recovered components: $\{\hat{\mathbf{a}}_k\}$
- Error metric: $e_{\text{comp}} = \frac{1}{K} \sum_{k=1}^{K} \gamma_{\min}^{k}$
- $\gamma_{\min}^{k} = \min_{k' \in [K]} \|\hat{\mathbf{a}}_k - \mathbf{a}_{k'}\|_2$

Performance variation

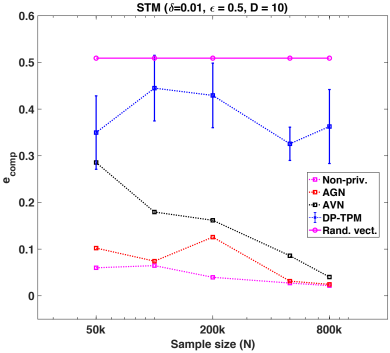
vs ϵ

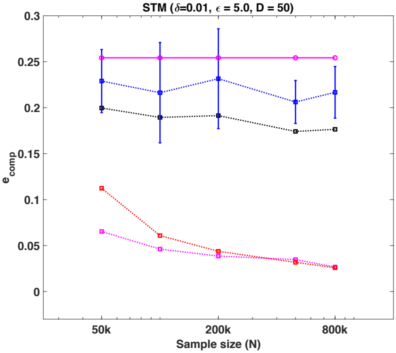




Performance variation

vs N





Concluding Remarks

Concluding remarks

- There are two stages where we add noise to ensure differential-privacy – optimal allocation of ϵ and δ is an open question
- The proposed methods outperform the DP-TPM [WA2016] and match the performance of the non-private method for large enough ϵ or N
- The AVN algorithm performs slightly worse than the AGN, but still much better than the DP-TPM
- The performance gap between AVN and DP-TPM is smaller for $D = 50$ than for $D = 10$

Questions

Thank you