Differentially Private Sparse Inverse Covariance Estimation

Di Wang, Mengdi Huai and Jinhui Xu
State University of New York at Buffalo

November 20, 2018

Outline

1. Motivations
   - Differential Privacy
   - Sparse Inverse Covariance Matrix Estimation

2. Methods

Outline

1. Motivations
   - Differential Privacy
   - Sparse Inverse Covariance Matrix Estimation

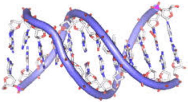
2. Methods

Why Estimating Privately

- Learning algorithms or Statistical inference always perform on sensitive dataset.

Why Estimating Privately

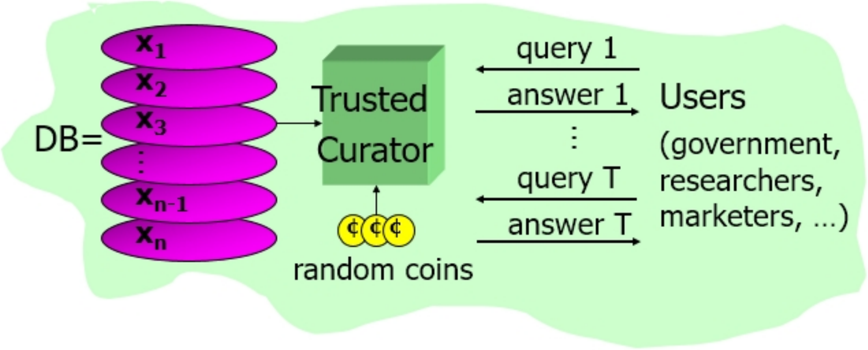
- Learning algorithms or Statistical inference always perform on sensitive dataset.



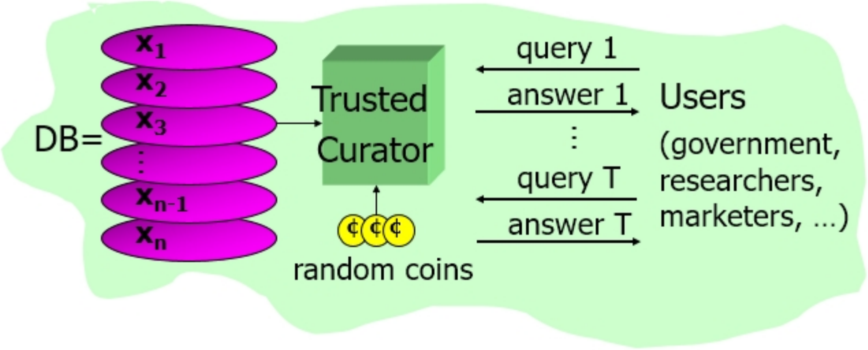
Why Estimating Privately

- Learning algorithms or Statistical inference always perform on sensitive dataset.
- Most learning algorithms are not private, which may be caused privacy breach and an adversary could infer data record! Even they are complex. [Calandrino et.al., 2011]

Privacy in Statistical Databases

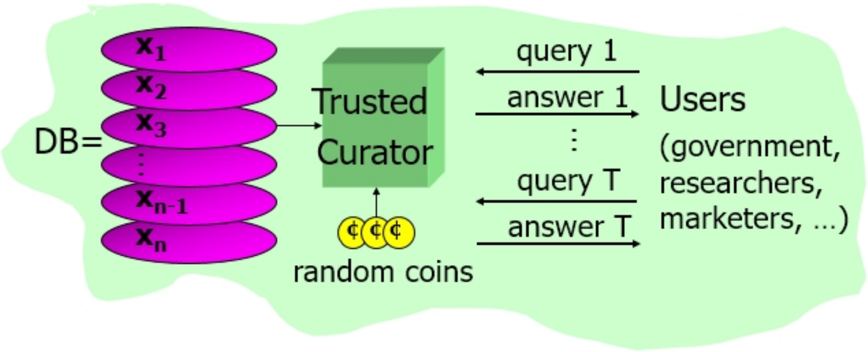


Privacy in Statistical Databases



- Two Conflict Goal: Privacy v.s Utility

Privacy in Statistical Databases



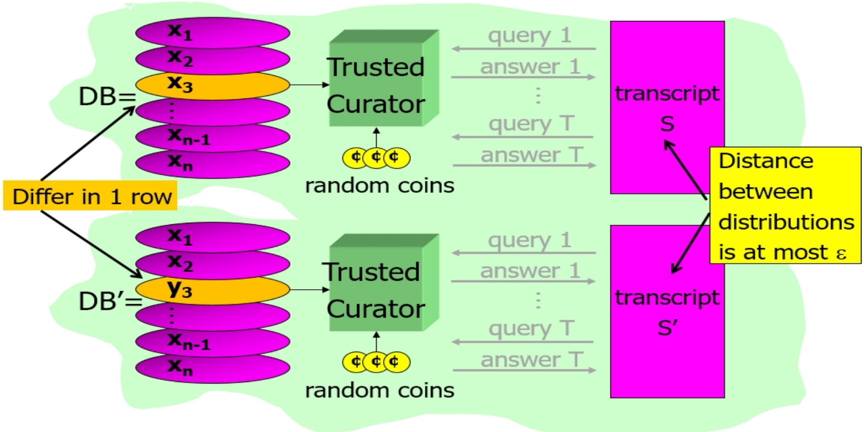
- Two Conflict Goal: Privacy v.s Utility
- Anonymization is unreliable [Narayanan-Shmatikov08],[Korolova11]...

Differential Privacy

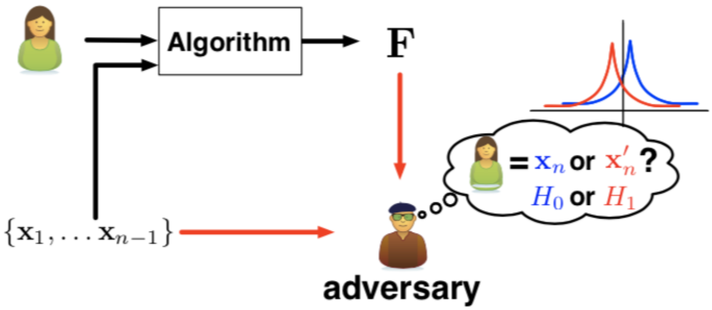
- Differential Privacy guarantees that the outcome distribution of the computation does not change significantly when a single record changes its data.

Differential Privacy

- Differential Privacy guarantees that the outcome distribution of the computation does not change significantly when a single record changes its data.



Differential Privacy



Differential Privacy

Definition of Differential Privacy

Definition (Differentially Private)

A randomized algorithm $\mathcal{A}$ is (ϵ, δ)-differentially private if for all neighboring datasets $D, D' \in \mathcal{X}^n$ and for all events S in the output space of $\mathcal{A}$, the following holds

$$Pr(\mathcal{A}(D) \in S) \leq e^{\epsilon} Pr(\mathcal{A}(D') \in S) + \delta.$$

when $\delta = 0$, $\mathcal{A}$ is ϵ-differentially private.

Outline

1. Motivations
   - Differential Privacy
   - Sparse Inverse Covariance Matrix Estimation

2. Methods

Problem Definition

- $\{x_1, x_2, \cdots, x_n\} \sim \mathcal{N}(0, \Sigma)$, where $\Sigma \in \mathbb{R}^{d \times d}$.

Problem Definition

- $\{x_1, x_2, \cdots, x_n\} \sim \mathcal{N}(0, \Sigma)$, where $\Sigma \in \mathbb{R}^{d\times d}$.
- If $n \geq d$, one can optimize

$$\Theta^* = S^{-1} = \arg\min_{\Theta\in\mathcal{S}^d_{++}} -\log\det\Theta + \langle S, \Theta\rangle,$$

where $S = \frac{1}{n}\sum_{i=1}^n x_i x_i^T$ is the empirical covariance.

Problem Definition

- $\{x_1, x_2, \cdots, x_n\} \sim \mathcal{N}(0, \Sigma)$, where $\Sigma \in \mathbb{R}^{d \times d}$.
- If $n \geq d$, one can optimize

$$\Theta^* = S^{-1} = \arg \min_{\Theta \in \mathcal{S}^d_{++}} - \log \det \Theta + \langle S, \Theta \rangle,$$

  where $S = \frac{1}{n} \sum_{i=1}^n x_i x_i^T$ is the empirical covariance.
- However, this will be ill-posed in the high dimensional case $p \geq n$.

Problem Definition

- $\{x_1, x_2, \cdots, x_n\} \sim \mathcal{N}(0, \Sigma)$, where $\Sigma \in \mathbb{R}^{d\times d}$.
- If $n \geq d$, one can optimize

$$\Theta^* = S^{-1} = \arg\min_{\Theta \in \mathcal{S}^d_{++}} -\log\det\Theta + \langle S, \Theta\rangle,$$

  where $S = \frac{1}{n}\sum_{i=1}^n x_i x_i^T$ is the empirical covariance.
- However, this will be ill-posed in the high dimensional case $p \geq n$.
- We borrow an idea in LASSO and use an ℓ_1 norm regularization in the objective function, which assumes that Θ^* is sparse.

Problem Definition

- $\{x_1, x_2, \cdots, x_n\} \sim \mathcal{N}(0, \Sigma)$, where $\Sigma \in \mathbb{R}^{d\times d}$.
- If $n \geq d$, one can optimize

$$\Theta^* = S^{-1} = \arg\min_{\Theta \in \mathcal{S}^d_{++}} -\log\det\Theta + \langle S, \Theta\rangle,$$

  where $S = \frac{1}{n}\sum_{i=1}^n x_i x_i^T$ is the empirical covariance.
- However, this will be ill-posed in the high dimensional case $p \geq n$.
- We borrow an idea in LASSO and use an ℓ_1 norm regularization in the objective function, which assumes that Θ^* is sparse.
- the objective function becomes the following:

$$\Theta^*_\rho = \arg\min_{\Theta \in \mathcal{S}^d_{++}} \{-\log\det\Theta + \langle S, \Theta\rangle + \rho\|\Theta\|_1\}, \tag{1}$$

  where $\rho > 0$ is the penalty parameter, $\langle S, \Theta\rangle = \mathrm{tr}(S\Theta^T)$, and $\|\Theta\|_1 = \sum_{i,j} |\Theta_{i,j}|$.

Problem Definition (Contin.)

- The problem has many applications in machine learning, signal processing and compotational biology.

Problem Definition (Contin.)

- The problem has many applications in machine learning, signal processing and compotational biology.
- Also a natural way for parameterizing the Gaussian graphical model

Problem Definition (Contin.)

- The problem has many applications in machine learning, signal processing and compotational biology.
- Also a natural way for parameterizing the Gaussian graphical model
- Thus, our goal is to get a private matrix Θ^{priv} which is close to the underlying sparse inverse covariance.

Problem Definition (Contin.)

- The problem has many applications in machine learning, signal processing and compotational biology.
- Also a natural way for parameterizing the Gaussian graphical model
- Thus, our goal is to get a private matrix Θ^{priv} which is close to the underlying sparse inverse covariance.
- make the error $\|\Theta^{\text{priv}} - \Theta^{*}\|_F$ as small as possible.

Method 1: Output Perturbation

- The first method is inspired by the redsensitivity of the optimization problem.

Method 1: Output Perturbation

- The first method is inspired by the redsensitivity of the optimization problem.
- The sensitivity of an algorithm $\mathcal{A}$ (under F-norm) is definied as

$$\max_{D\sim D'} \|\mathcal{A}(D) - \mathcal{A}(D')\|_F.$$

Method 1: Output Perturbation

- The first method is inspired by the redsensitivity of the optimization problem.
- The sensitivity of an algorithm $\mathcal{A}$ (under F-norm) is defined as

$$\max_{D \sim D'} \|\mathcal{A}(D) - \mathcal{A}(D')\|_F.$$

- Based on sensitivity, one can add random noise to ensure ϵ or (ϵ, δ) differential privacy.

Method 1: Output Perturbation

- The first method is inspired by the redsensitivity of the optimization problem.
- The sensitivity of an algorithm $\mathcal{A}$ (under F-norm) is definied as

$$\max_{D \sim D'} \|\mathcal{A}(D) - \mathcal{A}(D')\|_F.$$

- Based on sensitivity, one can add random noise to ensure ϵ or (ϵ, δ) differential privacy.
- We consider the case of adding Symmetric Laplacian Matrix and Wishard Matrix.

Output Perturbation (Contin.)

- What is Wishart distribution?

Output Perturbation (Contin.)

- What is Wishart distribution?
- Suppose $x_1, x_2, \cdots, x_m \in \mathbb{R}^d \sim \mathcal{N}(0, C)$. Then we call $S = \sum_{i=1}^m x_i x_i^T \sim \mathcal{W}_d(m, C)$.

Output Perturbation (Contin.)

- What is Wishart distribution?
- Suppose $x_1, x_2, \cdots, x_m \in \mathbb{R}^d \sim \mathcal{N}(0, C)$. Then we call $S = \sum_{i=1}^{m} x_i x_i^T \sim \mathcal{W}_d(m, C)$.

Algorithm 1 Output Perturbation

Input: $D = \{x_i\}_{i=1}^n$, $S = \frac{1}{n}\sum_{i=1}^n x_i x_i^T$, where the ℓ_2-norm of each row x_i is bounded by 1, $\rho > 0$.

1: Compute $\Theta_\rho^* = \arg\min_{\Theta \in \mathcal{S}_{++}^d} \{-\log\det\Theta + \langle S, \Theta\rangle + \rho\|\Theta\|_1\}$,

2: **return** $\tilde{\Theta}_\rho^* = \Theta_\rho^* + N$, where $N \sim \mathcal{W}_d(d+1, C), C = \frac{d^{\frac{5}{2}}}{n\epsilon\rho^2} I_d$.

Results

- Algorithm 1 is ϵ-differenctially private.

Results

- Algorithm 1 is ϵ-differenctially private.
- We have the following upper bound of error

$$\|\tilde{\Theta}^*_\rho - \Theta^*_\rho\|_F \leq O(\frac{\log \frac{d}{\delta} d^4}{n\epsilon\rho^2}),$$

where Θ^*_ρ is the optimal solution of the original problem.

Results

- Algorithm 1 is ϵ-differenctially private.
- We have the following upper bound of error

$$\|\tilde{\Theta}^*_\rho - \Theta^*_\rho\|_F \leq O(\frac{\log \frac{d}{\delta} d^4}{n\epsilon\rho^2}),$$

  where Θ^*_ρ is the optimal solution of the original problem.
- Quite large in the high dimensional case. Can we furtherly reduce the error?

Method 2: Covariance Perturbation

- Recall the non-private black box algorithm.

Method 2: Covariance Perturbation

- Recall the non-private black box algorithm.
- Our second method is perturbating covariance directly.

Algorithm 2 Covariance Perturbation

Input: $D = \{x_i\}_{i=1}^n$, where the ℓ_2-norm of each row x_i is bounded by 1, $\rho > 0$. $\epsilon, \delta \geq 0$ are the privacy parameters.

1: Let $S = \frac{1}{n}\sum_{i=1}^n x_i x_i^T$; sample a symmetric matrix $N \in \mathbb{R}^{d\times d} \sim \mathcal{P}$, which makes $S + N$ ϵ- or (ϵ, δ)-differentially private. Let $\tilde{S} = S + N$.

2: Return $\hat{\Theta}^*_\rho = \arg\min_{\Theta \in \mathcal{S}^d_{++}} \{-\log\det\Theta + \langle \tilde{S}, \Theta\rangle + \rho\|\Theta\|_1\}$.

Covariance Perturbation (Contin.)

- The key point is how to choose the random matrix N.

Covariance Perturbation (Contin.)

- The key point is how to choose the random matrix N.
- ϵ-differential privacy: Symmetric Laplace Matrix/ Wishart Matrix.

Covariance Perturbation (Contin.)

- The key point is how to choose the random matrix N.
- ϵ-differential privacy: Symmetric Laplace Matrix/ Wishart Matrix.
- (ϵ, δ)-differential privacy: Symmetric Gaussian Matrix/ Wishart Matrix.

Results: ϵ-DP

Theorem

For any $\epsilon > 0$, if N is a symmetric Laplacian matrix N whose entries are i.i.d drawn from $Lap(0, \frac{2d}{n\epsilon})$, then it is ϵ-differentially private. Moreover, the following holds

$$\frac{\|\hat{\Theta}^*_\rho - \Theta^*_\rho\|_F}{\max\{\|\Theta^*_\rho\|_2^2, \|\hat{\Theta}^*_\rho\|_2^2\}} \leq O(\frac{d^2}{n\epsilon}).$$

Results: ϵ-DP

Theorem

For any $\epsilon > 0$, if N is a symmetric Laplacian matrix N whose entries are i.i.d drawn from $Lap(0, \frac{2d}{n\epsilon})$, then it is ϵ-differentially private. Moreover, the following holds

$$\frac{\|\hat{\Theta}^*_\rho - \Theta^*_\rho\|_F}{\max\{\|\Theta^*_\rho\|_2^2, \|\hat{\Theta}^*_\rho\|_2^2\}} \leq O(\frac{d^2}{n\epsilon}).$$

Theorem

In Algorithm 2, for any $\epsilon > 0$, if choose $\mathcal{P} = \mathcal{W}_d(d+1, C)$ with $C = \frac{3}{2\epsilon n} I_d$, it is ϵ-differentially private. Moreover, the following holds

$$\frac{\|\hat{\Theta}^*_\rho - \Theta^*_\rho\|_F}{\max\{\|\Theta^*_\rho\|_2^2, \|\hat{\Theta}^*_\rho\|_2^2\}} \leq O(\frac{\log \frac{d}{\delta'} d^{\frac{3}{2}}}{n\epsilon}).$$

Results: (ϵ, δ)-DP

Theorem

If we choose $\mathcal{P} = \mathcal{W}_d(m, C)$ with $C = \frac{1}{n} I_d$ and $m = d + \frac{14}{\epsilon^2} \ln(\frac{4}{\delta})$ in Algorithm 2, it is (ϵ, δ)-differentially private. Moreover, we have

$$\frac{\|\hat{\Theta}^*_\rho - \Theta^*_\rho\|_F}{\max\{\|\Theta^*_\rho\|_2^2, \|\hat{\Theta}^*_\rho\|_2^2\}} \leq O(\frac{\ln(1/\delta)\ln(1/\delta')d^{\frac{3}{2}}}{n\epsilon^2}).$$

Results: (ϵ, δ)-DP

Theorem

If we choose $\mathcal{P} = \mathcal{W}_d(m, C)$ with $C = \frac{1}{n} I_d$ and $m = d + \frac{14}{\epsilon^2}\ln(\frac{4}{\delta})$ in Algorithm 2, it is (ϵ, δ)-differentially private. Moreover, we have

$$\frac{\|\hat{\Theta}^*_\rho - \Theta^*_\rho\|_F}{\max\{\|\Theta^*_\rho\|_2^2, \|\hat{\Theta}^*_\rho\|_2^2\}} \leq O(\frac{\ln(1/\delta)\ln(1/\delta')d^{\frac{3}{2}}}{n\epsilon^2}).$$

Theorem

If N is a symmetric Gaussian matrix N whose entries are i.i.d drawn from $\mathcal{N}(0, \beta^2)$, where $\beta = \frac{\sqrt{2\ln(\frac{1.25}{\delta})}}{n\epsilon}$, then it is (ϵ, δ)-differentially private.

$$\frac{\|\hat{\Theta}^*_\rho - \Theta^*_\rho\|_F}{\max\{\|\Theta^*_\rho\|_2^2, \|\hat{\Theta}^*_\rho\|_2^2\}} \leq O(\frac{d\sqrt{\ln(\frac{1}{\delta})}}{\epsilon n}).$$

Conclusion

- For ϵ-DP, covariance perturbation is better than output perturbation.
- For ϵ-DP, adding Wishart matrix is better than adding symmetric Laplacian matrices.
- The error bound of the (ϵ, δ)-differentially private algorithm with covariance perturbation strategy is lower than it under ϵ-differential privacy.
- Adding symmetric Gaussian noise will achieve the lowest error.

Experimental Results

Table 1: Performance comparisons of the ϵ-differentially private algorithms on both synthetic and real-world datasets.

		Synthetic Datasets			Real-world Datasets	
ϵ	Methods	$r = 0.5$	$r = 1.0$	$r = 1.5$	Colon	Parkinson's
0.5	**Wishart**	**0.993**	**0.9918**	**0.9914**	**0.995**	**0.9140**
	Output	NA	NA	NA	NA	NA
	Laplace	101.4	52.85	35.42	190.57	9.950
1.0	**Wishart**	**0.986**	**0.9863**	**0.9856**	**0.993**	**0.8899**
	Output	NA	NA	NA	NA	NA
	Laplace	49.44	25.41	16.83	95.01	4.690
1.5	**Wishart**	**0.9817**	**0.9815**	**0.9806**	**0.9907**	**0.8796**
	Output	NA	NA	NA	NA	NA
	Laplace	32.30	16.41	10.76	63.67	3.913

Experimental Results

Table 2: Performance comparisons of the (ϵ, δ)-differentially private algorithms on both synthetic and real-world datasets.

		Synthetic Datasets			Real-world Datasets	
ϵ	Methods	$r = 0.5$	$r = 1.0$	$r = 1.5$	Colon	Parkinson's
0.5	Wishart	0.9999	0.9997	0.9993	1.636	1.00
	SQLU	NA	NA	NA	NA	0.7419
	Gaussian	**0.1285**	**0.1607**	**0.1759**	**0.3039**	**0.1527**
1.0	Wishart	0.9982	0.9947	0.9906	1.1155	0.990
	SQLU	NA	NA	NA	NA	0.7318
	Gaussian	**0.1254**	**0.1605**	**0.1737**	**0.1081**	**0.1514**
1.5	Wishart	0.9954	0.9895	0.9837	1.0474	0.9992
	SQLU	NA	NA	NA	NA	0.7065
	Gaussian	**0.1242**	**0.1585**	**0.1701**	**0.0833**	**0.1474**

Thank you!

Questions?