Per-instance Differential Privacy (on graphs)

Yu-Xiang Wang

UC Santa Barbara

Outline

- Per-instance DP
- An example with linear regression
- pDP on Graphs

*** I prepared the slides in a rush… sorry for the missing references.**

How do we choose ε?

- No standard/guidelines.

- Need ε < 1: Quote Frank McSherry
  - *"Anything much bigger than one is not a very reassuring guarantee. Using an epsilon value of 14 per day strikes me as relatively pointless."*

- It's typical to use a larger ε in applications
  - Including some deployed DP systems

- A reasonable sentiment: DP is a worst-case guarantee
  - the actual privacy guarantee could be substantially better.

Recall the definition of DP

- Differential Privacy:

$$\sup_{Z,Z':d(Z,Z')\leq 1} \boxed{\sup_{h\in\mathcal{H}}} \log \frac{p_{h\sim\mathcal{A}(Z)}(h)}{p_{h\sim\mathcal{A}(Z')}(h)} \leq \epsilon$$

“I also get to choose any outcome.”

Approx DP, CDP, Renyi DP and so on.
Privacy r.v.: ε(output)

“I get to choose the worst pair of adjacent data sets.”

Per-instance DP: ε(Dataset, Individual)

- **Definition**: A is ε-pDP on (Z,z) if

$$\xcancel{\sup_{Z,Z': d(Z,Z')\leq 1}} \sup_{h\in\mathcal{H}} \log \frac{p_{h\sim\mathcal{A}(Z)}(h)}{p_{h\sim\mathcal{A}(Z')}(h)} \leq \epsilon$$

  - a strict generalization
  - Measures the privacy loss a specific person z suffers from running A on a specific data set Z.

"I can observe the data but cannot change it."

Per-instance sensitivity

- The per instance sensitivity of function f

$$\Delta_{\|\cdot\|_*}(f, Z, z) = \|f(Z) - f([Z, z])\|_*$$

- Global sensitivity : max over (Z,z)

- Local sensitivity: fix Z, max over z

Example: Linear regression

- Data matrix

$$X = [x_1^T, x_2^T, ..., x_n^T]^T$$

- Response vector

$$y = [y_1, ..., y_n]^T$$

- How do we release:

$$\theta = (X^T X)^{-1} X^T y$$

- Unbounded global sensitivity!

- Let's do ridge regression

$$\theta_\lambda = (X^T X + \lambda I)^{-1} X^T y$$

- And add noise to the output.

Per-instance sensitivity of linear regression coefficients

- per-instance sensitivity in A-norm is

$$|y - x\hat{\theta}'|\sqrt{x^T(X^TX)^{-1}A(X^TX)^{-1}x}$$

Residual/prediction error

Statistical leverage score, when A ≈ X^TX

- Multivariate Gaussian noise adding for pDP.

*Can be calculated very efficiently using the Woodbury Identity.

What can I do with pDP?

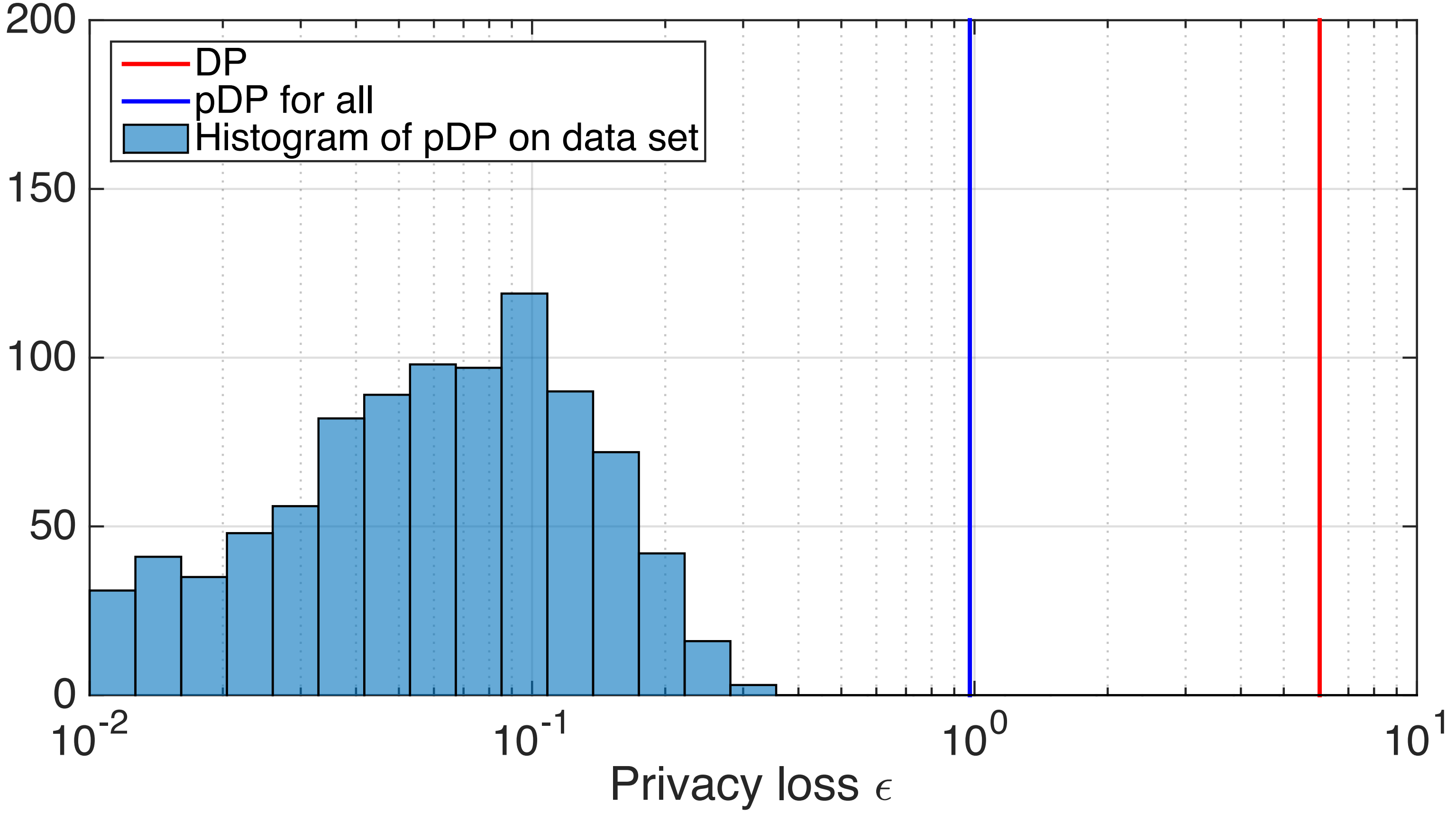
- Generate comprehensive privacy summary.
  - What is the privacy loss incurred to users in my data set?
  - How is Bob’s privacy loss comparing to Mary?

- As an analytical tool for data-dependent DP algorithm design
  - pDP to DP conversion
  - Complement smooth sensitivity (Nissim et al., 2007) and propose-test-release (Dwork and Lei, 2009).

pDP-based comprehensive privacy summary

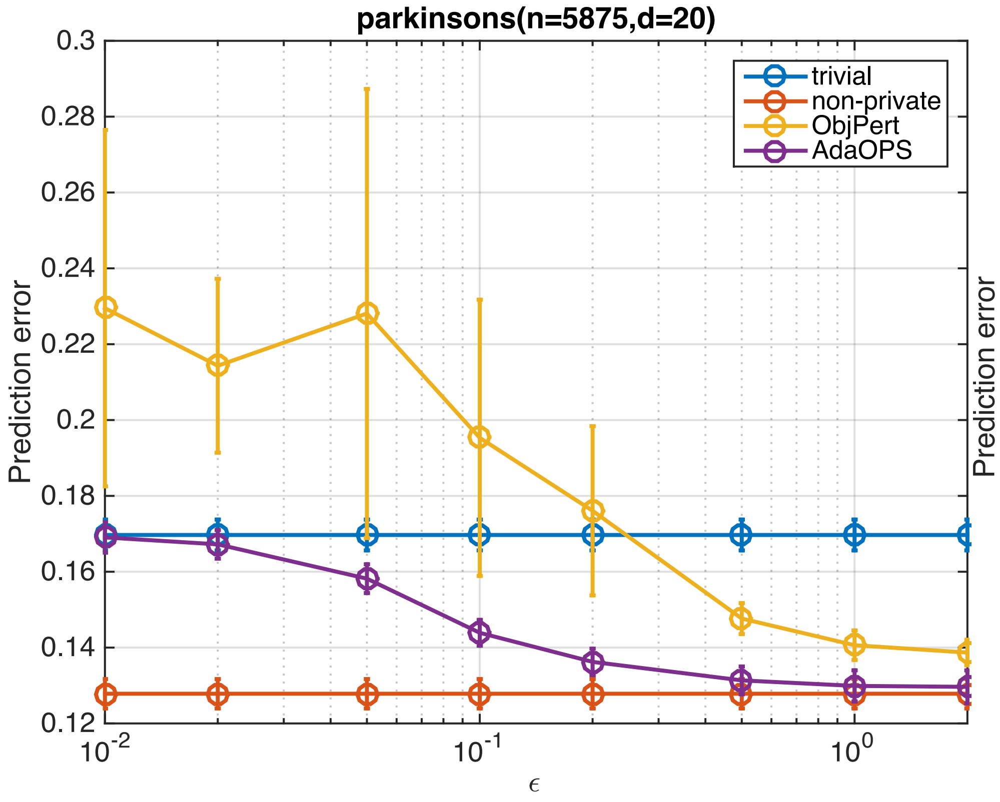
Generate data set by linear Gaussian model. Fix the algorithm below.

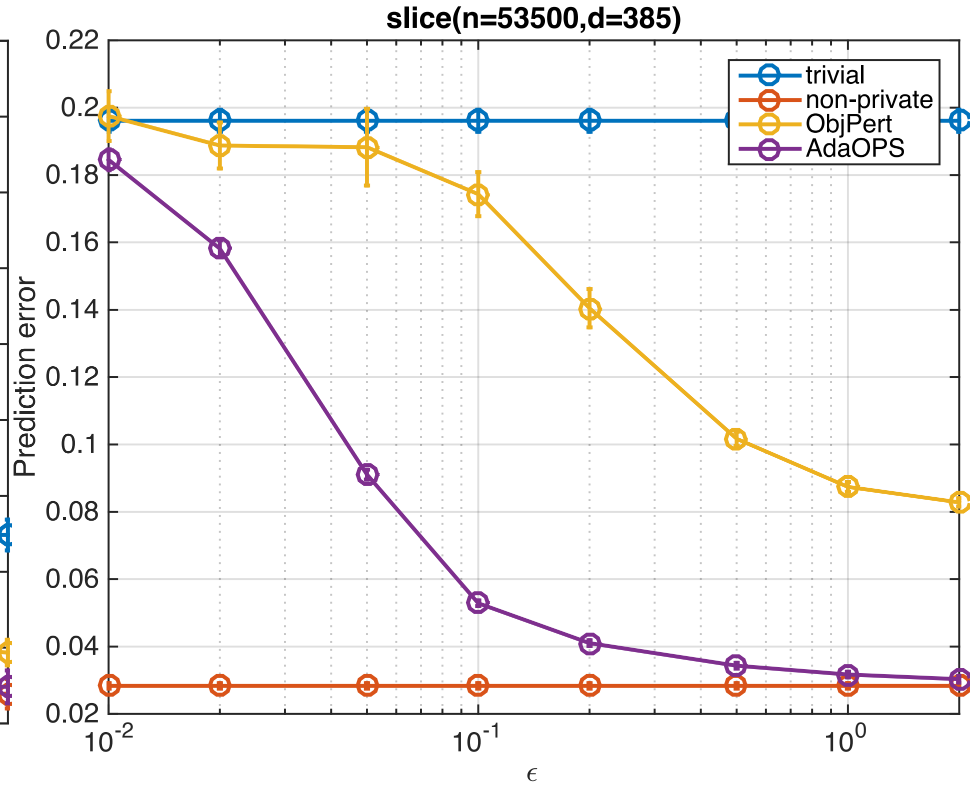
$$\tilde{\theta} \sim N((X^T X + I)^{-1} X\mathbf{y}, \sigma^2 I), \quad \sigma = 4$$



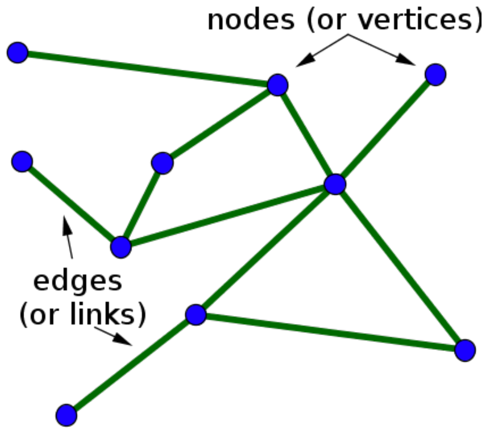
Results of a pDP analysis for the posterior sampling algorithm for linear regression

- AdaOPS --- Sample from posterior distribution with an data-driven choice of prior / regularization weight





pDP for Graphs?



- Data matrix

$$X = [x_1^T, x_2^T, \ldots, x_n^T]^T$$

- Gram matrix (covariance)

$$G = \sum_{\ell} x_\ell x_\ell^T = X^T X$$

- Edge incidence matrix

$$D_\ell = (0, \ldots \underset{\substack{\uparrow \\ i}}{-1}, \ldots \underset{\substack{\uparrow \\ j}}{1}, \ldots 0)$$

- Graph Laplacian

$$L = D^T D = \sum_{\ell} D_\ell D_\ell^T$$

Edge / nodal pDP

- A node is just a collection of edges

$$D_\ell = (0, \ldots \underset{\overset{\uparrow}{i}}{-1}, \ldots \underset{\overset{\uparrow}{j}}{1}, \ldots 0)$$

- A Justin Bieber node has a large privacy loss.
- But 99.9% of typical twitter users have could have ε = 0.1.

- pDP of any edge / node are efficiently computable!

Immediate applications

- Releasing Graph Laplacian
  - AnalyzeGauss, Johnson-Lindenstrauss
  - Can we use the same to releasing Graph Laplacian?
  - How about using graph sparsification?

- Will normalized Laplacian be more tractable?

- Private Laplacian smoothing over a graph?

$$x = \arg\min_x \|y - x\|^2 + x^T L x$$

Summary

- pDP as an analytical tool
- more interpretable/relevant privacy loss.

- Future work:
    - pDP analysis for more algorithms (graph mining algorithms?)
    - private release of pDP summaries.
    - Economic view of pDP in data collection process.

Thank you for your attention!

Yu-Xiang Wang, "Per-Instance Differential Privacy", Journal of Privacy and Confidentiality.
Yu-Xiang Wang, "Revisiting differentially private linear regression: optimal and adaptive prediction & estimation in unbounded domain", UAI'18

Disclaimer

- pDP is not a replacement of DP.
  - It is an analytical tool to represent more refined privacy footprint of a randomized algorithm.
- We should not calibrate the noise of an algorithm to achieve a particular pDP level for an individual.
- pDP is a data-dependent quantity. Cannot be naively revealed.

Stability of stationary points

- Let f be an optimization query:
  - Find me a stationary point of the loss function

$$f(Z) \in \{\theta | \nabla \mathcal{L}_Z(\theta) = 0\}$$

Lemma: Critical points of $\mathcal{L}_Z$ and $\mathcal{L}_{[Z,z]} = \mathcal{L}_Z + \ell_z$ obey that

$$\hat{\theta}' - \hat{\theta} = \left[\int_{\hat{\theta}}^{\hat{\theta}'} \nabla^2 \mathcal{L}_Z(t) dt\right]^{-1} \nabla \ell_z(\hat{\theta}')$$

AdaOPS for Linear Regression

1. DP-release of $\bar{\lambda} > \lambda_{\min}(XX^T)$ **1-Stable by Weyl's lemma**

2. DP-release of $\bar{B} > \|\theta^*\|_2$ **1-Stable after log(1+ .) transform**

3. Choose γ, λ appropriately using the remaining balance of ϵ, δ

Regularization plays a more important role than noise

1. Output: $\tilde{\theta} \sim N(\theta^*, \gamma^{-1}(XX^T + \lambda I)^{-1})$

Which ``A'' to use for Multivariate Gaussian noise adding?

- Standard choice:
  - A ∝ Identity ⇔ Output Pert. [CMS-2013]
- Democratic choice:
  - A ∝ (X^TX)^2 ⇔ Obj Pert. [CMS-2013]
- ``Fisher'' choice:
  - A ∝ X^TX ⇔ OPS

Refined statistical analysis of OPS for linear regression

- Previous analysis [W. Fienberg, Smola, 2015]
  - (1 + 4B/ε)-efficiency and ε-DP
  - Restrict domain s.t. loss function < B

- Direct analysis using pDP:

$$1 + O\left(\frac{d\log(1/\delta)}{n\epsilon^2}\right)$$ and (ε,δ)-pDP for all unit x

No domain restriction needed!

Faster rate, better dimension-dependence than [Smith, 2008] and [Dwork & Smith, 2009], who first obtain such 1+ o(1) statistical efficiency.

Regret of OPS in agnostic setting

- Let $$F(\theta) = 0.5\|\mathbf{y} - X\theta\|^2$$
- OPS on regularized objective $$F(\theta) + \frac{\lambda}{2}\|\theta\|_2^2$$

$$F(\tilde{\theta}) - F(\theta^*) \leq \frac{d \log(d/\delta) \log(2/\delta)}{[\lambda + \lambda_{\min}(X^T X)]\epsilon^2} + \lambda\|\theta^*\|_2^2$$

With probability 1-δ

Matches both lower bounds in [Bassily et. al., 14].

High probability bound. Run time does not depend on ε.
Works in unbounded domain. highly practical.

Data-dependent analysis

- Traditional DP algorithm design:
  - The algorithm receives a privacy budget ε
  - Calibrate noise to global sensitivity to achieve ε-DP
  - Calibrate noise to a data-dependent sensitivity to achieve ε-DP

Different noise level on different data set.

- Post-hoc DP analysis:
  - Fix my randomized algorithm A
  - Analyze the resulting ε-DP from running A on any data set
  - Analyze the resulting ε-DP from running A on my data set Z

Same noise level, different ε.

Is εpsilon a privacy budget or a privacy loss?

A priori declaration of privacy budget

- DP algorithm design.
- Calibrating noise to global sensitivity.
- Privacy budget ε is a hard constraint to be met.

Post-hoc calculation of privacy loss

- Privacy loss as a random variable: ε(output)
- Advanced composition
- CDP, Renyi DP.
- Privacy amplification by subsampling

Is εpsilon a privacy budget or a privacy loss?

- Traditional DP algorithm design:
  - The algorithm receives a privacy budget ε
  - Calibrate noise to sensitivity to achieve ε-DP

- Post-hoc DP analysis:
  - Fix my randomized algorithm A
  - Analyze the resulting ε-DP from running A

Post hoc privacy loss is not new

- Privacy loss as a random variable: ε(output)

- Essentially what's driving much of the recent breakthroughs:
  - Advanced composition
  - Privacy amplification
  - CDP / RDP
  - And many more

Recall the definition of DP

- Differential Privacy:

Max-Divergence

$$\sup_{Z,Z':d(Z,Z')\leq 1} \sup_{h\in\mathcal{H}} \log \frac{p_{h\sim\mathcal{A}(Z)}(h)}{p_{h\sim\mathcal{A}(Z')}(h)} \leq \epsilon$$

Approx DP, CDP, RDP and so on