

# Task-aware Privacy Preservation for Multi-dimensional Data

Jiangnan Cheng, Ao Tang  
*Cornell University*

Sandeep Chinchali  
*The University of Texas at Austin*

July, 2022

## Background and Motivation

- **Differential Privacy (DP)**: SOTA technique for data privacy.

## Background and Motivation

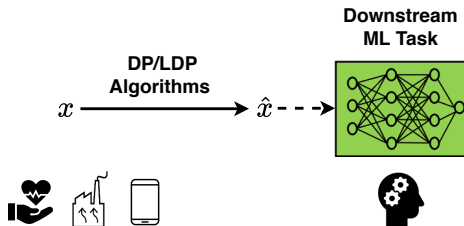
- **Differential Privacy (DP)**: SOTA technique for data privacy.
- **Local Differential Privacy (LDP)**: A variant that provides stronger privacy guarantee for individual users.

## Background and Motivation

- **Differential Privacy (DP)**: SOTA technique for data privacy.
  - **Local Differential Privacy (LDP)**: A variant that provides stronger privacy guarantee for individual users.
- **Status Quo**: used for basic frequency estimation tasks.

# Background and Motivation

- **Differential Privacy (DP)**: SOTA technique for data privacy.
  - **Local Differential Privacy (LDP)**: A variant that provides stronger privacy guarantee for individual users.
- **Status Quo**: used for basic frequency estimation tasks.



**Figure 1:** LDP has the promising potential to be adopted in complex scenarios (e.g., health care, IOT) that feature rich user data attributes.

- Task-aware privacy preservation problem: How to improve the **ultimate task performance** with **multi-dimensional user data**?

## Preliminary: $\epsilon$ -LDP and Laplace mechanism

- Let  $x \in \mathbb{R}^n$  be an individual data sample, and  $\mathcal{X}$  be the domain of  $x$ , which is assumed to be a compact subset of  $\mathbb{R}^n$ .

## Preliminary: $\epsilon$ -LDP and Laplace mechanism

- Let  $x \in \mathbb{R}^n$  be an individual data sample, and  $\mathcal{X}$  be the domain of  $x$ , which is assumed to be a compact subset of  $\mathbb{R}^n$ .
- A randomized algorithm  $\mathcal{M}$  is said to satisfy  $\epsilon$ -**LDP**, if

$$\Pr[\mathcal{M}(x) \in \mathcal{S}] \leq e^\epsilon \Pr[\mathcal{M}(x') \in \mathcal{S}], \quad \forall x, x' \in \mathcal{X}, \mathcal{S} \subseteq \text{im } \mathcal{M}. \quad (1)$$

## Preliminary: $\epsilon$ -LDP and Laplace mechanism

- Let  $x \in \mathbb{R}^n$  be an individual data sample, and  $\mathcal{X}$  be the domain of  $x$ , which is assumed to be a compact subset of  $\mathbb{R}^n$ .
- A randomized algorithm  $\mathcal{M}$  is said to satisfy  $\epsilon$ -**LDP**, if

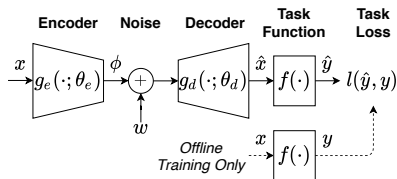
$$\Pr[\mathcal{M}(x) \in \mathcal{S}] \leq e^\epsilon \Pr[\mathcal{M}(x') \in \mathcal{S}], \quad \forall x, x' \in \mathcal{X}, \mathcal{S} \subseteq \text{im } \mathcal{M}. \quad (1)$$

- To release a function  $g : \mathcal{X} \mapsto \mathbb{R}^Z$  s.t.  $\epsilon$ -LDP of  $x$  is preserved, **Laplace mechanism** is a widely used **noise-adding mechanism**:

$$\mathcal{M}_{\text{Lap}}(x, g, \epsilon) = g(x) + \text{Lap}^Z(\mu = 0, b = \frac{\Delta_1 g}{\epsilon}), \quad (2)$$

where  $\Delta_1 g = \max_{x, x' \in \mathcal{X}} \|g(x) - g(x')\|_1$ .

# Problem Formulation



$$\min_{Z, \theta_e, \theta_d} \mathcal{L} = \mathbb{E}_{x, w} [l(\hat{y}, y)], \quad (3)$$

$$\text{s.t. } y = f(x), \quad \hat{y} = f(g_d(g_e(x; \theta_e) + w; \theta_d)), \quad (4)$$

$$x \sim \mathcal{D}_x, \quad w \sim \text{Lap}^Z\left(0, \frac{\Delta_1 g_e}{\epsilon}\right). \quad (5)$$

## Analysis: Linear Model and MSE Task Loss

- **Mild assumption.** Covariance matrix has Cholesky decomposition

$$\Sigma_{xx} \triangleq \mathbb{E}[(x - \mu_x)(x - \mu_x)^\top] = LL^\top. \quad (6)$$

For convenience we consider  $h = L^{-1}(x - \mu_x)$ , which has  $\Sigma_{hh} = I$ .

## Analysis: Linear Model and MSE Task Loss

- **Mild assumption.** Covariance matrix has Cholesky decomposition

$$\Sigma_{xx} \triangleq \mathbb{E}[(x - \mu_x)(x - \mu_x)^\top] = LL^\top. \quad (6)$$

For convenience we consider  $h = L^{-1}(x - \mu_x)$ , which has  $\Sigma_{hh} = I$ .

- The problem with linear model and MSE task loss:

$$\min_{Z,D,E} \mathcal{L} = \mathbb{E}_{x,w} [\|f(h) - f(\hat{h})\|_2^2] \quad (7)$$

$$\text{s.t. } \hat{h} = D(Eh + w), D \in \mathbb{R}^{n \times Z}, E \in \mathbb{R}^{Z \times n} \quad (8)$$

$$h \sim \mathcal{D}_h, w \sim \text{Lap}^Z(0, \frac{\Delta_1 Eh}{\epsilon}). \quad (9)$$

where  $f(h) = Ph \in \mathbb{R}^m$  with task matrix  $P \in \mathbb{R}^{m \times n}$ .

## Boundary and Optimal $\mathcal{L}^*$

- Use  $\mathcal{S}$  to denote the convex hull of domain  $\mathcal{H}$ .
- When the boundary  $\partial\mathcal{S}$  is a centered hypersphere with radius  $r \geq 0$ :  $\{h \in \mathbb{R}^n \mid \|h\|_2^2 = r^2\}$ , the optimal  $\mathcal{L}^*$  is

$$\mathcal{L}(r; \lambda_{1:n}, \epsilon) = \frac{8r^2/\epsilon^2}{1 + Z' \cdot 8r^2/\epsilon^2} \left( \sum_{i=1}^{Z'} \sqrt{\lambda_i} \right)^2 + \sum_{i=Z'+1}^n \lambda_i, \quad (10)$$

where  $\lambda_{1:n}$  are eigen-values of  $P^\top P$  in descending order, and  $Z' \leq n$  is the largest integer such that

$$\frac{\sqrt{\lambda_{Z'}}}{\sum_{i=1}^{Z'} \sqrt{\lambda_i}} \left( 1 + Z' \cdot \frac{8r^2}{\epsilon^2} \right) - \frac{8r^2}{\epsilon^2} > 0. \quad (11)$$

## Boundary and Optimal $\mathcal{L}^*$

- Use  $\mathcal{S}$  to denote the convex hull of domain  $\mathcal{H}$ .
- When the boundary  $\partial\mathcal{S}$  is a centered hypersphere with radius  $r \geq 0$ :  $\{h \in \mathbb{R}^n \mid \|h\|_2^2 = r^2\}$ , the optimal  $\mathcal{L}^*$  is

$$\mathcal{L}(r; \lambda_{1:n}, \epsilon) = \frac{8r^2/\epsilon^2}{1 + Z' \cdot 8r^2/\epsilon^2} \left( \sum_{i=1}^{Z'} \sqrt{\lambda_i} \right)^2 + \sum_{i=Z'+1}^n \lambda_i, \quad (10)$$

where  $\lambda_{1:n}$  are eigen-values of  $P^\top P$  in descending order, and  $Z' \leq n$  is the largest integer such that

$$\frac{\sqrt{\lambda_{Z'}}}{\sum_{i=1}^{Z'} \sqrt{\lambda_i}} \left( 1 + Z' \cdot \frac{8r^2}{\epsilon^2} \right) - \frac{8r^2}{\epsilon^2} > 0. \quad (11)$$

- Suppose  $\partial\mathcal{S} \subset \{h \in \mathbb{R}^n : r_{\min}^2 \leq \|h\|_2^2 \leq r_{\max}^2\}$ . Then

$$\mathcal{L}(r_{\min}; \lambda_{1:n}, \epsilon) \leq \mathcal{L}^* \leq \mathcal{L}(r_{\max}; \lambda_{1:n}, \epsilon). \quad (12)$$

# Gradient-based Algorithm for General Settings

Encoding function  $g_e$ , decoding function  $g_d$  and task function  $f$  are parameterized by **differentiable neural networks**.

---

**Require:** Privacy budget  $\epsilon$  and  $Z$

- 1: Initialize encoder/decoder parameters  $\theta_e, \theta_d$  and noise vector  $w$
  - 2: **for**  $\tau \in \{0, 1, \dots, N_{\text{epochs}} - 1\}$  **do**
  - 3: Update  $\theta_e$  and  $\theta_d$  with  $-\nabla(\mathcal{L} + \eta \|\theta_e\|_F^2) / \nabla \theta_e$  and  $-\nabla \mathcal{L} / \nabla \theta_d$ , respectively, by one or multiple steps ( $w$  fixed)
  - 4: Recompute  $\Delta_1 g_e$ , and re-sample  $w$  from  $\text{Lap}^Z(0, \Delta_1 g_e / \epsilon)$
  - 5: Return  $\theta_e, \theta_d$
- 

Penalizing  $\|\theta_e\|_F^2$ : avoid the growth of the scale of  $\phi$ .

# Evaluations

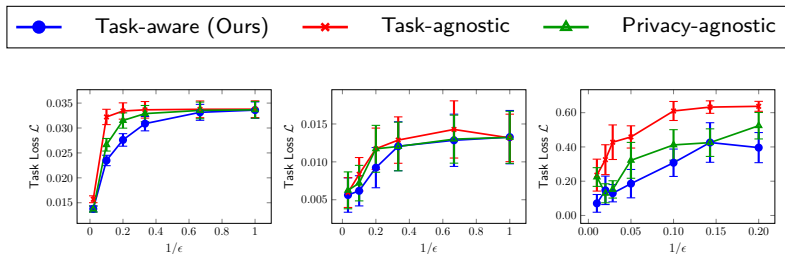


Figure 2: Evaluation results for hourly household power consumption (left), real estate valuation (middle), and breast cancer detection (right).