PCKV: Locally Differentially Private Correlated Key-Value Data Collection with Optimized Utility

Xiaolan Gu*, Ming Li*, Yueqiang Cheng**, Li Xiong# and Yang Cao†

* University of Arizona    ** Baidu Security    # Emory University    † Kyoto University

USENIX Security Symposium, August 2020
Overview

- Background of LDP
- Problem Statement and Existing Mechanism
- Our Framework: PCKV
- Experiments
- Conclusion
Background

• Companies are collecting our private data to provide better services (Google, Facebook, Apple, Yahoo, Uber, …)

• However, privacy concerns arise

• Possible solution: locally private data collection model

  • Yahoo: massive data breaches impacted 3 billion user account, 2013
  • Facebook: 267 million users’ data has reportedly been leaked, 2019
  • …
Local Differential Privacy (LDP) [Duchi et al, FOCS’ 13]

A mechanism \( M \) satisfies \( \epsilon \)-LDP if and only if for any pair of inputs \( x, x' \) and any output \( y \)

\[
\frac{\Pr(M(x) = y)}{\Pr(M(x') = y)} \leq e^\epsilon
\]

- \( x, x' \): the possible input (raw) data (generated by the user)
- \( y \): the output (perturbed) data (public and known by adversary)
- \( \epsilon \): privacy budget (a smaller \( \epsilon \) indicates stronger privacy)

An adversary cannot infer whether the input is \( x \) or \( x' \) with high confidence (controlled by \( \epsilon \))
Applications of LDP

Enabling developers and organizations to use differential privacy
Thursday, September 5, 2019

Posted by Miguel Guevara, Product Manager, Privacy and Data Protection Office

Source:

Source:

Apple: discovering popular Emojis under LDP
LDP Protocol: Randomized Response

- Randomized Response (RR) [Warner, 1965]: reports the truth with some probability (for binary answer: yes-or-no)

- Example: Is your annual income more than 100k?

<table>
<thead>
<tr>
<th>Truth $x$</th>
<th>Response $y$</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>w.p. $p$</td>
</tr>
<tr>
<td>0</td>
<td>w.p. $1-p$</td>
</tr>
<tr>
<td>1</td>
<td>w.p. $1-p$</td>
</tr>
<tr>
<td>0</td>
<td>w.p. $p$</td>
</tr>
</tbody>
</table>

To satisfy $\epsilon$-LDP: $p = \frac{e^\epsilon}{e^\epsilon + 1}$ (since $\frac{p}{1-p} = e^\epsilon$)

Frequency estimation: $\hat{f} = \frac{f - (1-p)}{2p - 1}$

Unbiasedness: $\mathbb{E}[\hat{f}] = f^*$

True frequency:

$$\mathbb{E}[f] = f^*p + (1-f^*)(1-p) = (2p - 1)f^* + (1-p)$$
Extend RR for General Cases

• Assume the domain size is \( d \) (taking \( d = 5 \) for example)

**Optimized Unary Encoding (OUE)**

[Wang et al, USENIX Security' 17]

<table>
<thead>
<tr>
<th>Input</th>
<th>Bit vector</th>
<th>Encode</th>
<th>Perturb</th>
</tr>
</thead>
<tbody>
<tr>
<td>3</td>
<td>0 0 1 0 0</td>
<td></td>
<td>Flip 1 w.p. 0.5</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Flip 0 w.p. 1 ( - p )</td>
</tr>
</tbody>
</table>

Output 1 0 1 1 0

To satisfy \( \epsilon \)-LDP: \( p = \frac{e^\epsilon}{e^\epsilon + 1} \)

**Staircase or Generalized RR (GRR)**

[Kairouz et al, NeuIPS' 16]

<table>
<thead>
<tr>
<th>Input</th>
<th>Directly perturb</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>2</td>
</tr>
<tr>
<td>2</td>
<td>3</td>
</tr>
<tr>
<td>3</td>
<td>3</td>
</tr>
<tr>
<td>4</td>
<td>4</td>
</tr>
<tr>
<td>5</td>
<td>5</td>
</tr>
</tbody>
</table>

W.p. \( p \)

W.p. \( \frac{1 - p}{4} \)

To satisfy \( \epsilon \)-LDP: \( p = \frac{e^\epsilon}{e^\epsilon + (d - 1)} \)

RR, OUE and GRR are building block mechanisms for frequency aggregation
### A motivating example (movie rating system)

**Data Type:** each user has different number of key-value pairs

**Data Domain:** key in \( \{1, 2, \ldots, d\} \), value in \([-1, 1]\)

**Task:** frequency and mean estimation

**Threat Model:** honest-but-curious server

**Objectives:** good privacy-utility tradeoff

**Challenges**

1. Each user has different number of key-value pairs.

2. If a fake key is reported, how to report the corresponding value?

3. How to design an optimal mechanism with the best privacy-utility tradeoff?

**Key-Value Data Collection**

<table>
<thead>
<tr>
<th>Key</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Spider-Man, 3.0</td>
<td></td>
</tr>
<tr>
<td>The Godfather, 4.0</td>
<td></td>
</tr>
<tr>
<td>Man in Black, 4.5</td>
<td></td>
</tr>
<tr>
<td>Spider-Man, 3.5</td>
<td></td>
</tr>
<tr>
<td>Man in Black, 3.5</td>
<td></td>
</tr>
<tr>
<td>The Godfather, 5.0</td>
<td></td>
</tr>
</tbody>
</table>

Ratings are in the range [1, 5]
Existing Mechanism: PrivKVM [Ye et al, S&P’19]

Step 1. Convert key-value pairs into a vector

In each round, each user 1) randomly samples an index $j$ from $\{1, \cdots, d\}$; 2) privately reports the $j$-th pair (if a fake key is reported, then the value will be perturbed from the estimated mean by the server)

Step 2. Iteratively update the mean of each key (use sequential composition)

Limitations of PrivKVM

- The multiple rounds require all users to be always online and the privacy budget in each round is very small (thus large error).
- The naive sampling protocol may not work well for a large domain.
- No improved privacy budget composition (although key and value are perturbed with some correlation).

Our Mechanism

- Only one round
- Advanced sampling protocol
- Tight privacy budget composition (and optimized budget allocation)
Outline

• Background of LDP
• Problem Statement and Existing Mechanism
• Our Framework: PCKV
• Experiments
• Conclusion
Overview of PCKV

1. **Privacy Budget Allocation and Perturbation**
   - **Probability Computation**
     - $\epsilon$: the total privacy budget
     - PCKV-UE: $\epsilon \rightarrow \{\epsilon_1, \epsilon_2\} \rightarrow \{a, b, p\}$
     - PCKV-GRR: $\epsilon \rightarrow \{\epsilon_1, \epsilon_2\} \rightarrow \{a, p\}$
   - $\epsilon_1$: budget for key perturbation
   - $\epsilon_2$: budget for value perturbation
   - $a, b, p$: perturbation probabilities

2. **Sampling**
   - $S \rightarrow x = \{k, v\}$

3. **Perturbation**
   - PCKV-UE: $x \rightarrow y$ (vector)
   - PCKV-GRR: $x \rightarrow y' = \{k', v'\}$

4. **Aggregation**
   - PCKV-UE: $y[k] \rightarrow \{n_1, n_2\}$
   - PCKV-GRR: $y' \rightarrow \{n_1, n_2\}$

5. **Estimation**
   - $\{n_1, n_2\} \rightarrow \{\hat{f}_k, \hat{m}_k\}$

---

- **Advanced sampling protocol**: each user pads her keys into a uniform length $\ell$ by some dummy keys

- **Joint privacy analysis**: in an end-to-end way (instead of directly using sequential composition)

- **Optimized allocation** of $\epsilon_1$ and $\epsilon_2$: by minimizing MSE of estimation under tight budget composition

---

We use Padding-and-Sampling [S&P’ 18] to improve sampling efficiency

We theoretically evaluate the utility by MSE of estimation

Joint perturbation and privacy analysis can improve privacy-utility tradeoff (due to tight privacy budget composition)
**Perturbation and Privacy Analysis**

### Joint/Correlated Perturbation

- **Unbiased map to 1 and -1**
- **With privacy budget $\epsilon_1$**
- **Key Perturbation**
- **Value Discretization**
- **If a fake key is reported?**
  - Yes
    - Report value as 1 and -1 w.p. 0.5
  - No
    - Value Perturbation

### Joint Privacy Analysis

- The final privacy budget is less than $\epsilon_1 + \epsilon_2$
- $\epsilon = \max\{\epsilon_2, \epsilon_1 + \ln[2/(1 + e^{-\epsilon_2})]\} \leq \epsilon_1 + \epsilon_2$
  (because $\epsilon_1 \geq 0$ and $\frac{2}{1 + e^{-\epsilon_2}} \leq e^{\epsilon_2}$)

### Observations

- **PCKV-UE** has tighter privacy budget composition than directly using sequential composition
- **PCKV-GRR** has similar tight composition and additional privacy benefit from sampling.
- **PrivKVM** does not have tight composition (because the fake value is reported with two different probabilities).
Aggregation and Estimation

- The server computes the supporting numbers of value 1 and -1 for the $k$-th key.

- Estimated frequency $\hat{f}_k$ : multiplied by $\ell$ due to sampling, where $\mathbb{E}[\hat{f}_k] = f_k^*$

- Estimated mean $\hat{m}_k = \frac{\text{calibrated sum}}{\text{calibrated counts}}$, where $\mathbb{E}[\hat{m}_k] \rightarrow m_k^*$ when $n \rightarrow \infty$

- The Mean Squared Errors (MSEs) of $\hat{f}_k$ and $\hat{m}_k$ depend on how to balance $\epsilon_1$ and $\epsilon_2$ under a fixed total privacy budget $\epsilon$
Optimized Privacy Budget Allocation

Relationship among $\epsilon_1$, $\epsilon_2$ and $\epsilon$

Tight Composition + $\min \text{MSE}$ $\rightarrow$ Optimized Allocation

Final Perturbation (after sampling)

Summary of PCKV

- Step 1. Choose the advanced sampling protocol
- Step 2. Jointly perturb key-value and jointly analyze the privacy (which provides tight privacy budget composition)
- Step 3. Optimally put all things together (i.e., optimized privacy budget allocation under a fixed total budget)
Experiments

- Our mechanisms outperform existing ones on both frequency and mean estimation.
- The theoretical results (dashed lines) close to the empirical results (solid lines).

Improvements of PCKV
- Advanced sampling protocol
- Tight budget composition
- Optimized budget allocation
Experiments

Benefit from each improvement

- Tight Budget Composition v.s. Sequential Composition
  - More Accurate
- Optimized Budget Allocation v.s. Non-optimized
  - More Accurate

Success of top frequent keys identification (varying domain size)

- PCKV mechanisms outperform other ones
  - More Accurate
- PCKV-UE gets small impact from large domain size
  - More Accurate
Real-world Data

Amazon Dataset

# ratings: 2M
# users: 1M
# keys: 249K

Data source: https://www.kaggle.com/skillsmuggler/amazon-ratings

Movie Dataset

# ratings: 20M
# users: 138K
# keys: 26K

Data source: https://www.kaggle.com/ashukr/movie-rating-data
Conclusion

- The **advanced sampling protocol** can improve the sampling efficiency and the utility.

- **Joint/correlated perturbations** of key and value (rather than independent ones) can provide more options for mechanism design and the chance to choose the **optimized one**.

- **Joint privacy analysis** can lead to better privacy-utility tradeoff (because it results in tighter privacy budget composition than sequential composition)

Future work

- Study the optimized strategy of choosing $\ell$ in Padding-and-Sampling protocol.

- Extend the correlated perturbation and tight composition analysis to other general types of multi-dimensional data.
Thanks for your attention!

Q&A

Contact Information: xiaolang@email.arizona.edu