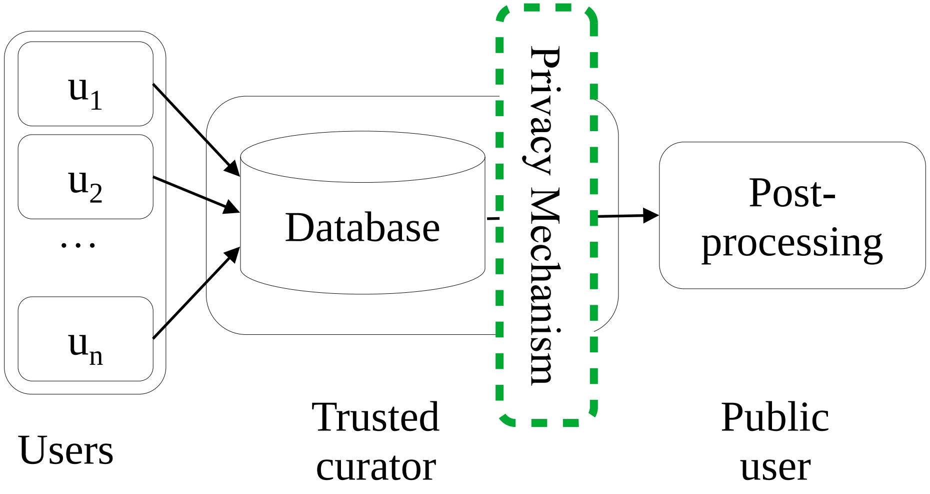
Multi-Freq-LDPy: Multiple Frequency Estimation Under Local Differential Privacy in Python

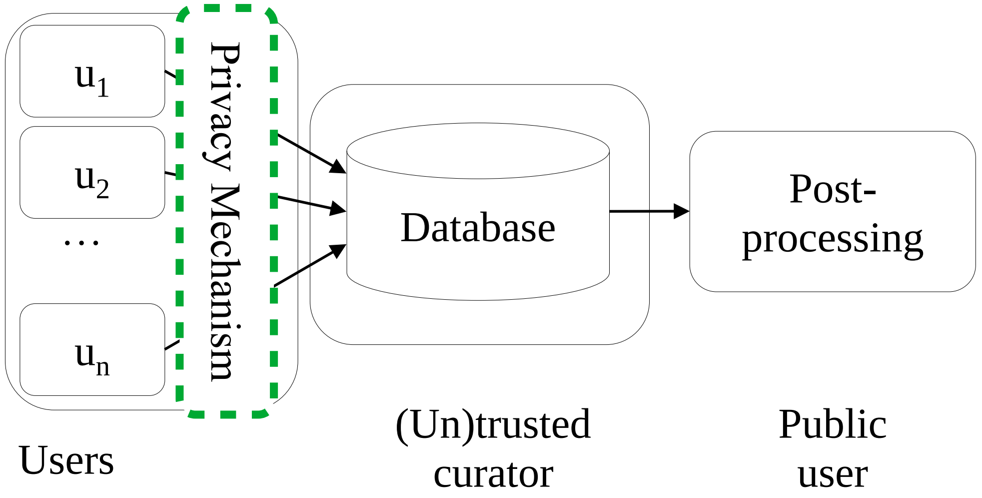
Héber H. Arcolezi, Jean-François Couchot, Sébastien Gambs, Catuscia Palamidessi, Majid Zolfaghari

Introduction

The Trust Model: Centralized *vs* Local



Centralized setting



Local setting

Differential Privacy (DP)*: DP → Local DP

A randomized algorithm $\mathcal{A}$ satisfies ϵ-DP, if for **any two neighbouring databases D and D'** and for any output O of $\mathcal{A}$:

Privacy loss

Intuitively: Any output should be about as likely regardless of whether I am in the database or not.

$$\Pr[\mathcal{A}(D) = O] \le e^{\epsilon} \cdot \Pr[\mathcal{A}(D') = O]$$

Run by a trusted server

A randomized algorithm $\mathcal{A}$ satisfies ϵ-local-differential-privacy (ϵ-LDP), if for **any two inputs x and x'** and for any output y of $\mathcal{A}$:

Privacy loss

Intuitively: Any output should be about as likely regardless of my secret.

$$\Pr[\mathcal{A}(x) = y] \le e^{\epsilon} \cdot \Pr[\mathcal{A}(x') = y]$$

Run by each user

* Dwork, C., Roth, A. The algorithmic foundations of differential privacy. In: Foundations and Trends in Theoretical Computer Science (2014).

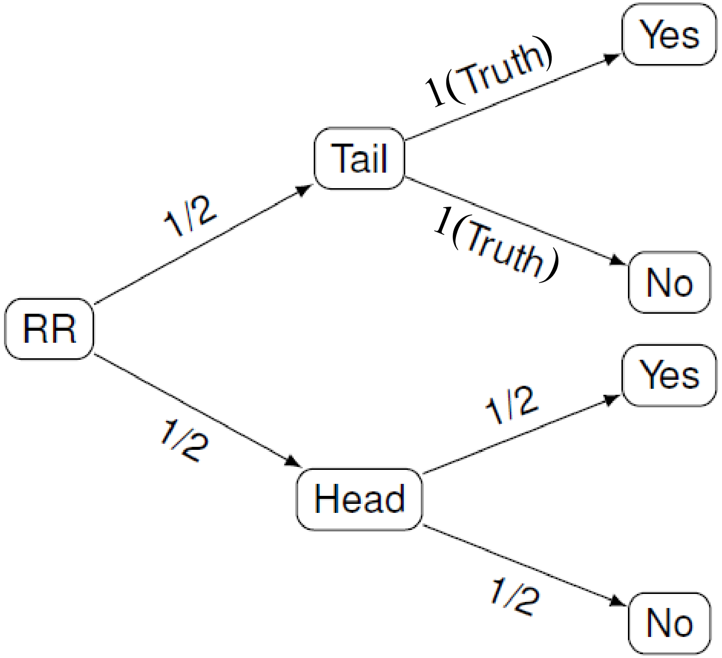
LDP: Ex. of Randomized Response (RR)*

- Motivated by surveying people on sensitive/embarrassing topics.
- Main idea → Providing **deniability** to users' answer (yes/no → binary).
- Ask: "Did you test positive for HIV (human immunodeficiency virus)?"
- Each person:
  - Throw a secret unbiased coin:
    - If tail, throw the coin again (ignoring the outcome) and answer the question honestly.
    - If head, then throw the coin again and answer "Yes" if head, "No" if tail.

RR: Seeing answer, still not certain about the secret.

* Warner, S.L. Randomized response: A survey technique for eliminating evasive answer bias. In: Journal of the American Statistical Association (1965).

Frequency Estimation and ϵ Study of RR



$$p = \Pr[RR(Yes) = Yes] = \Pr[RR(No) = No] = 0.75$$

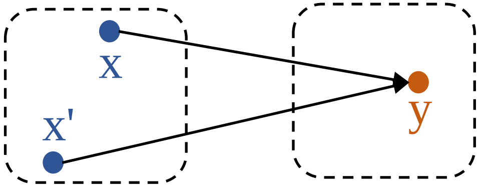
$$q = \Pr[RR(No) = Yes] = \Pr[RR(Yes) = No] = 0.25$$

- $f(v_Y) \rightarrow$ frequency of *true Yes (or No –* v_N*)*
- $\approx \hat{f}(v_i) = \frac{N_i - nq}{(p-q)}, \forall_{i \in \{Y,N\}}$ $\dashrightarrow$ Estimated frequency
- Satisfies ϵ-LDP w/:

$$\frac{\Pr(y|x)}{\Pr(y|x')} \leq e^{\epsilon} \Rightarrow e^{\epsilon} = \frac{0.75}{0.25}, \epsilon = \ln(3)$$

prob. p of ‘being honest’

prob. q of ‘lying’



Input set　　Output set

LDP Implem. of Big Tech Companies

RAPPOR: Randomized Aggregatable Privacy-Preserving Ordinal Response

Úlfar Erlingsson
Google, Inc.
ulfar@google.com

Vasyl Pihur
Google, Inc.
vpihur@google.com

Aleksandra Korolova
University of Southern California
korolova@usc.edu

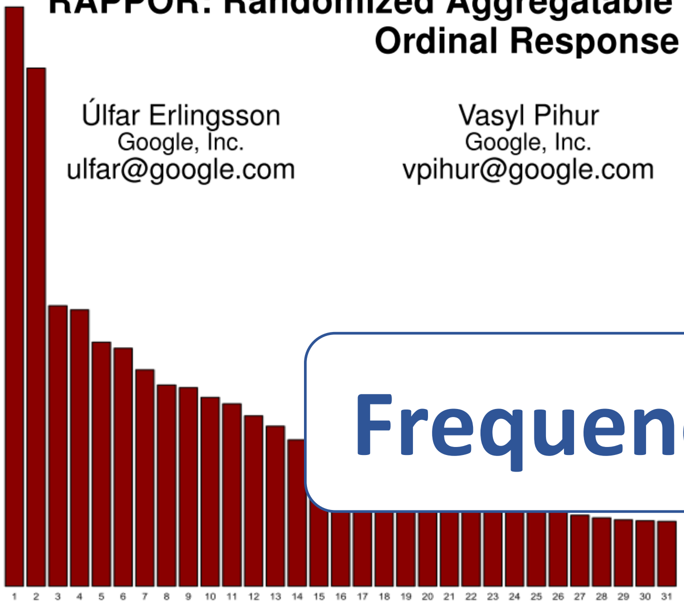
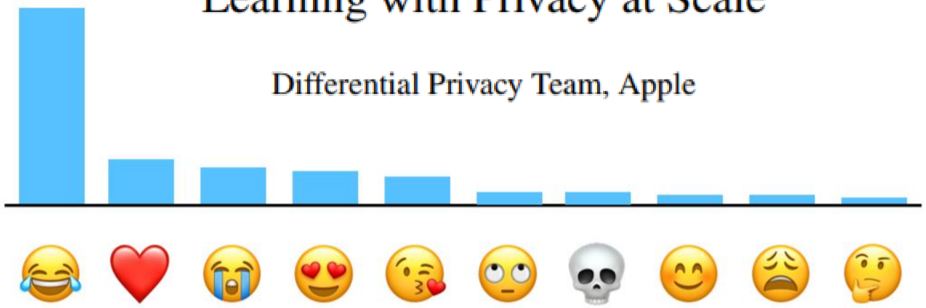


Figure 6: Relative frequencies of the top 31 unexpected Chrome homepage domains found by analyzing ~14 million RAPPOR reports, excluding expected domains (the homepage "google.com", etc.).

Learning with Privacy at Scale

Differential Privacy Team, Apple



Frequency (histogram) estimation

…most popular emoji to help
…i for US English speakers

Collecting Telemetry Data Privately

Bolin Ding, Janardhan Kulkarni, Sergey Yekhanin
Microsoft Research
{bolind, jakul, yekhanin}@microsoft.com

Windows Insiders in Windows 10 Fall Creators Update to protect users' privacy while collecting application usage statistics.

Outline

LDP Protocols for Frequency Estimation

- **Generalized RR (GRR)**[*]: Extends RR to the case of $k_j \geq 2$.

$$\forall_y \in A_j \ \Pr[\mathcal{A}_{GRR(\epsilon)}(v) = y] = \begin{cases} p = \frac{e^{\epsilon}}{e^{\epsilon}+k_j-1}, if\ y = v \\ q = \frac{1}{e^{\epsilon}+k_j-1}, otherwise \end{cases} \quad \epsilon = \ln\left(\frac{p}{q}\right)$$

- **Unary Encoding (UE)**[**]: Encode as a bit-vector B and perturb each bit independently into a new bit-vector B'. More specifically:

$$\Pr[B'_i = 1] = \begin{cases} p,\ if\ B_i = 1 \\ q,\ if\ B_i = 0 \end{cases} \qquad \epsilon = \ln\left(\frac{p(1-q)}{q(1-p)}\right)$$

Symmetric UE (SUE): $p = \frac{e^{\epsilon/2}}{e^{\epsilon/2}+1},\ q = \frac{1}{e^{\epsilon/2}+1},$ **Optimized UE (OUE)**[***]: $p = \frac{1}{2},\ q = \frac{1}{e^{\epsilon}+1}$

[*] Kairouz, P., Oh, S., Viswanath, P. Extremal mechanisms for local differential privacy. In: NeurIPS (2014).
[**] Erlingsson, Ú., Pihur, V. and Korolova, A. RAPPOR: Randomized Aggregatable Privacy-Preserving Ordinal Response. In: SIGSAC (2014).
[***] Wang, T., Blocki, J., Li, N. and Jha, S. Locally differentially private protocols for frequency estimation. In: USENIX Security Symposium (2017).

LDP Protocols for Frequency Estimation

- **Local Hashing**[*, **]: Deals with large domain size k_j by hashing input value to $g_j \ll k_j$ and then using GRR to the hashed value. Users report $\langle GRR\big(H(v_i)\big), H\rangle$.

$$\forall_y \in A_j \ \Pr\left[\mathcal{A}_{GRR(\epsilon)}(H(v)) = y\right] = \begin{cases} p = \frac{e^{\epsilon}}{e^{\epsilon}+\boldsymbol{g_j}-1}, if\ y = H(v) \\ q = \frac{1}{e^{\epsilon}+\boldsymbol{g_j}-1}, otherwise \end{cases} \qquad \epsilon = \ln\left(\frac{p}{q}\right)$$

Binary LH (BLH): $g_j = 2$ **Optimized LH (OLH)**[**]: $g_j = e^{\epsilon} + 1$

[*] Bassily, R., and Adam, S. Local, private, efficient protocols for succinct histograms. In: Proceedings of the forty-seventh annual ACM symposium on Theory of computing (2015).
[**] Wang, T., Blocki, J., Li, N. and Jha, S. Locally differentially private protocols for frequency estimation. In: USENIX Security Symposium (2017).

LDP Protocols for Frequency Estimation

- Server-Side (a.k.a. the aggregator): Unbiased[*] normalized frequency estimation $f(v_i)$ for $v_i \in A_j$:

$$\hat{f}(v_i) = \frac{N_i - nq}{\boldsymbol{n}(p-q)}$$

N_i = number of times the value v_i or bit i has been reported.

[*] Wang, T., Blocki, J., Li, N. and Jha, S. Locally differentially private protocols for frequency estimation. In: USENIX Security Symposium (2017).

Multi-freq-ldpy for Single Freq. Est.

multi-freq-ldpy is a function-based package that simulates the LDP data collection pipeline of users and the server. For each functionality, there is always a **Client** and an **Aggregator** function.

Multi-Freq-LDPy covers the following tasks:

1. **Single Frequency Estimation** -- The best-performing frequency oracles from Locally Differentially Private Protocols for Frequency Estimation, namely:
   - Generalized Randomized Response (GRR): `multi_freq_ldpy.pure_frequency_oracles.GRR`
   - Symmetric/Optimized Unary Encoding (UE): `multi_freq_ldpy.pure_frequency_oracles.UE`
   - Binary/Optimized Local Hashing (LH): `multi_freq_ldpy.pure_frequency_oracles.LH`
   - Adaptive (ADP) protocol, i.e., GRR or Optimized UE depending on variance value: `multi_freq_ldpy.pure_frequency_oracles.ADP`

PyPi Page: https://pypi.org/project/multi-freq-ldpy/

Pratical Demonstration: Colab Link or GitHub Link (tutorial 1)

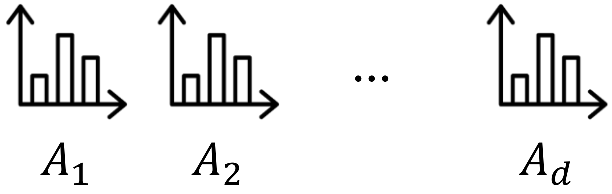
Outline

1. Introduction
2. Single Frequency Estimation
3. **Frequency Estimation of Multiple Attributes**
4. Single Longitudinal Frequency Estimation
5. Longitudinal Frequency Estimation of Multiple Attributes
6. Conclusion & Perspectives

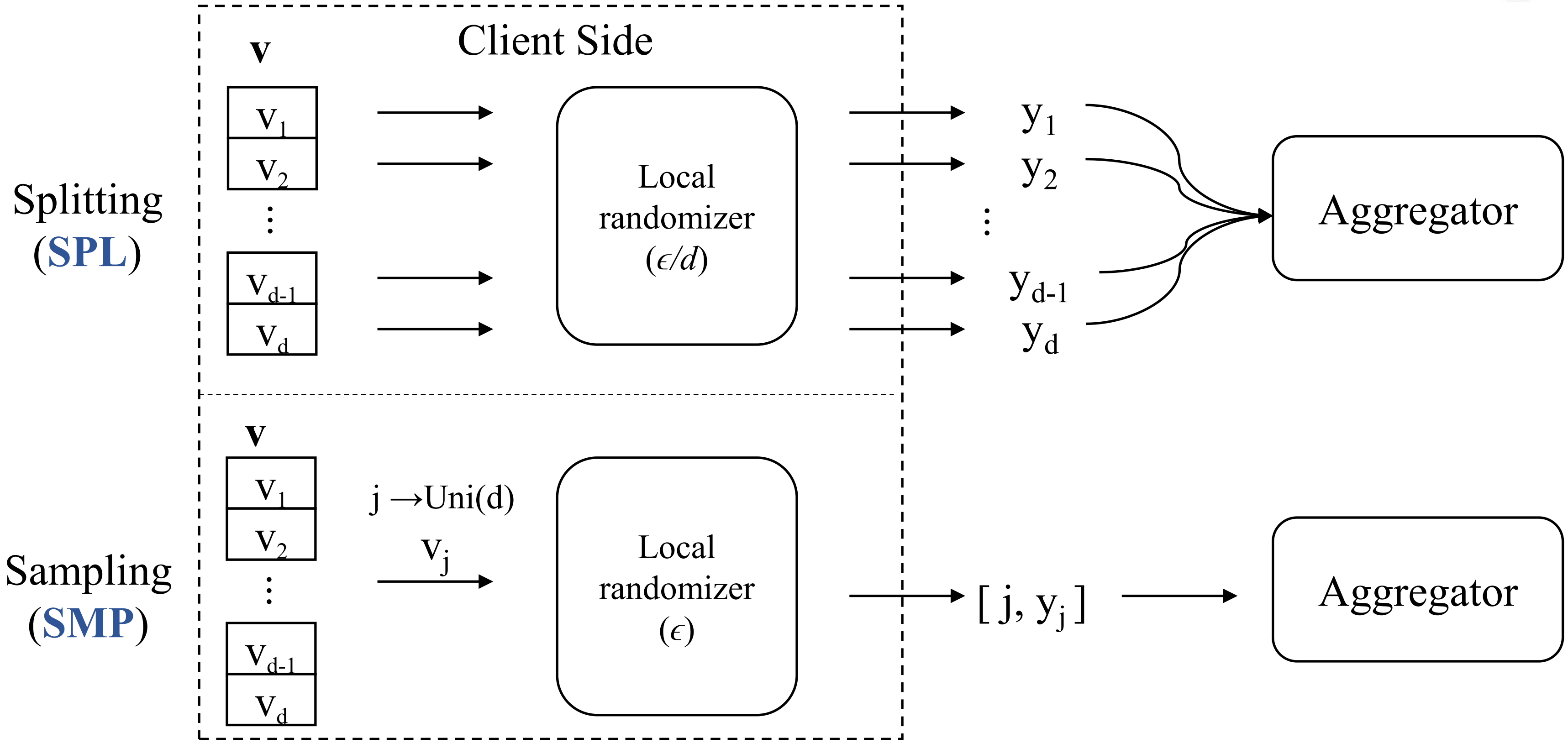
Multidimensional Frequency Estimation

- **Tackled Issue:** Collecting *multidimensional* data under *ϵ-LDP* for *frequency estimation*.

- **More formally (notation):**
  - d attributes $A = \{A_1, A_2, \dots, A_d\}$; → **Multiple attributes**
  - Each attribute A_j has a discrete domain of size $|A_j| = k_j$;
  - Each user u_i for $1 \leq i \leq n$ has a tuple $\mathbf{v}^i = (v_1^i, v_2^i, \dots, v_d^i)$;
  - **Analyzer:** estimate a k_j-bins histogram for each attribute $j \in [1, d]$.



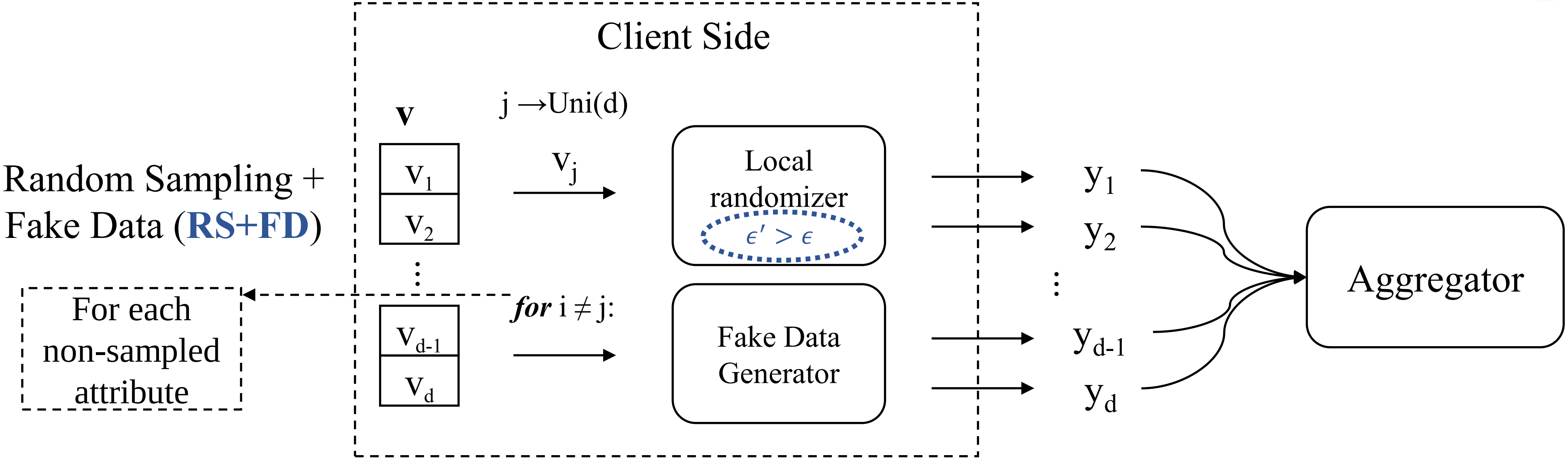
Solutions for Multiple Attributes*, **



* Nguyên, T.T., Xiao, X., Yang, Y., Hui, S.C., Shin, H., Shin, J. Collecting and analyzing data from smart device users with local differential privacy. In: arXiv:1606.05053 (2016).
** Wang, N., Xiao, X., Yang, Y., Zhao, J., Hui, S.C., Shin, H., Shin, J., Yu, G. Collecting and analyzing multidimensional data with local differential privacy. In: ICDE (2019).

Solutions for Multiple Attributes*

Random Sampling + Fake Data (**RS+FD**)



Intuition:

- **Sampling result is not disclosed**, privacy is amplified**.

* Arcolezi, H.H., Couchot, J.F., Al Bouna, B., and Xiao, X. Random Sampling Plus Fake Data: Multidimensional Frequency Estimates With Local Differential Privacy. In: ACM CIKM (2021).

** Li, N., Qardaji, W., Su, D. On sampling, anonymization, and differential privacy or, k-anonymization meets differential privacy. In: ASIACCS'12 (2012).

Multi-freq-ldpy for Multid. Freq. Est.

2. **Multidimensional Frequency Estimation** -- Three solutions for frequency estimation of multiple attributes from Random Sampling Plus Fake Data: Multidimensional Frequency Estimates With Local Differential Privacy with their respective frequency oracles (GRR, UE-based, and ADP), namely:
   - Splitting (SPL) the privacy budget: `multi_freq_ldpy.mdim_freq_est.SPL_solution`
   - Random Sampling (SMP) a single attribute: `multi_freq_ldpy.mdim_freq_est.SMP_solution`
   - Random Sampling + Fake Data (RS+FD) that samples a single attribute but also generates fake data for each non-sampled attribute: `multi_freq_ldpy.mdim_freq_est.RSpFD_solution`

PyPi Page: https://pypi.org/project/multi-freq-ldpy/

Pratical Demonstration: Colab Link or GitHub Link (tutorial 2)

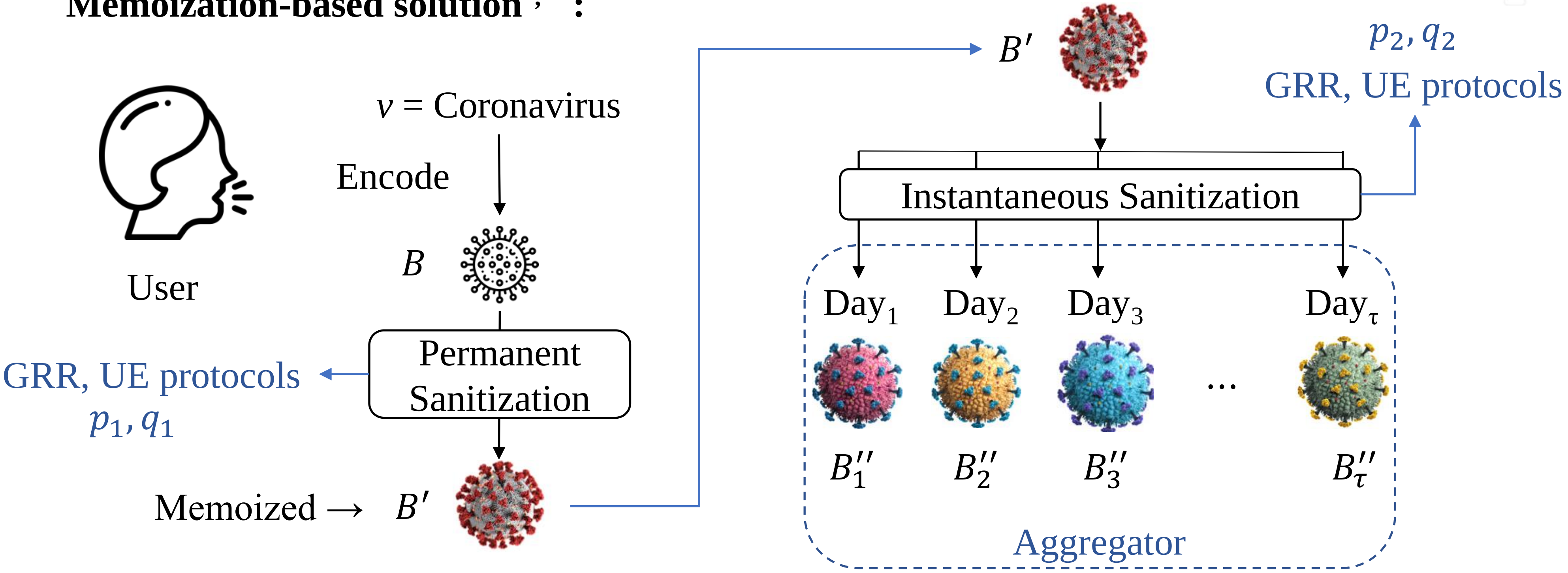
Inria femto-st SCIENCES & TECHNOLOGIES UQÀM Université du Québec à Montréal

Outline

1. Introduction
2. Single Frequency Estimation
3. Frequency Estimation of Multiple Attributes
4. **Single Longitudinal Frequency Estimation**
5. Longitudinal Frequency Estimation of Multiple Attributes
6. Conclusion & Perspectives

Longitudinal Frequency Estimation

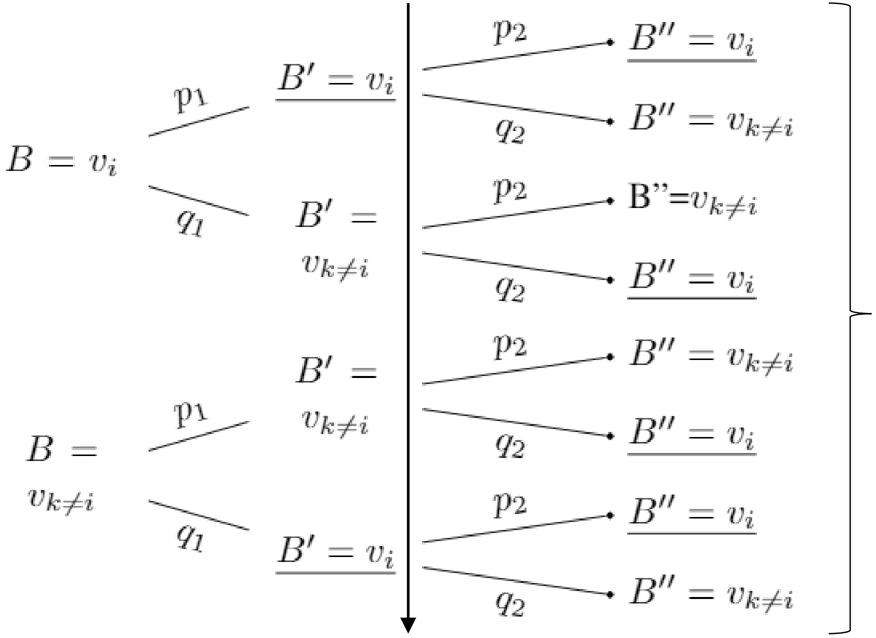
Memoization-based solution*, **:



* Erlingsson, Ú., Pihur, V., Korolova, A. RAPPOR: Randomized Aggregatable Privacy-Preserving Ordinal Response. In: ACM SIGSAC (2014).

** Ding, B., Kulkarni, J., Yekhanin, S. Collecting telemetry data privately. In: NeurIPS (2017).

Longitudinal GRR*: ϵ study



$$\Pr[B''|B] = \begin{cases} \Pr[B'' = v_i | B = v_i] = p_1p_2 + q_1q_2 \\ \Pr[B'' = v_{k\neq i} | B = v_i] = p_1q_2 + q_1p_2 \\ \Pr[B'' = v_i | B = v_{k\neq i}] = p_1q_2 + q_1p_2 \\ \Pr[B'' = v_{k\neq i} | B = v_{k\neq i}] = p_1p_2 + q_1q_2 \end{cases}$$

First report: $\epsilon_1 = \ln\left(\frac{p_1p_2 + q_1q_2}{p_1q_2 + q_1p_2}\right)$

Infinity reports:

$$\epsilon_\infty = \ln\left(\frac{p_1}{q_1}\right)$$

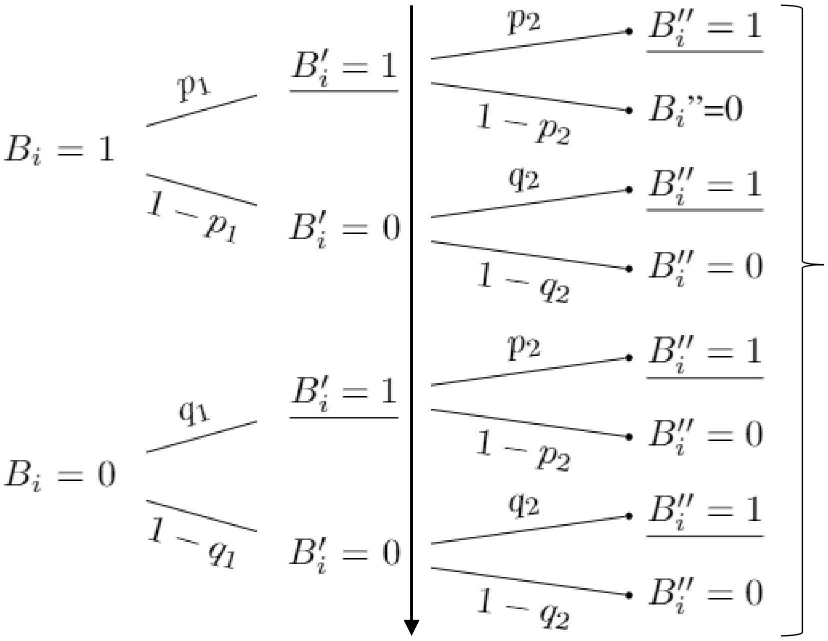
Given ϵ_∞ and ϵ_1:

$$p_1 = \frac{e^{\epsilon_\infty}}{e^{\epsilon_\infty} + k_j - 1}, q_1 = \frac{1 - p_1}{k_j - 1}$$

$$p_2 = \frac{e^{\epsilon_1 + \epsilon_\infty} - 1}{-k_j e^{\epsilon_1} + (k_j - 1)e^{\epsilon_\infty} + e^{\epsilon_1} + e^{\epsilon_\infty + \epsilon_1} - 1}, q_2 = \frac{1 - p_2}{k_j - 1}$$

* Arcolezi, H.H., Couchot, J.F., Al Bouna, B., and Xiao, X. Improving the Utility of Locally Differentially Private Protocols for Longitudinal and Multidimensional Frequency Estimates. arXiv:2111.04636 (2021).

Longitudinal UE*: ϵ study



$$\Pr[B_i''|B_i] = \begin{cases} \Pr[B_i'' = 1|B_i = 1] = p_1p_2 + (1-p_1)q_2 \\ \Pr[B_i'' = 0|B_i = 1] = p_1(1-p_2) + (1-p_1)(1-q_2) \\ \Pr[B_i'' = 1|B_i = 0] = q_1p_2 + (1-q_1)q_2 \\ \Pr[B_i'' = 0|B_i = 0] = q_1(1-p_2) + (1-q_1)(1-q_2) \end{cases}$$

First report:

$$\epsilon_1 = \ln\left(\frac{\left(p_1p_2 - q_2\left(p_1 - 1\right)\right)\left(p_2q_1 - q_2\left(q_1 - 1\right) - 1\right)}{\left(p_2q_1 - q_2\left(q_1 - 1\right)\right)\left(p_1p_2 - q_2\left(p_1 - 1\right) - 1\right)}\right)$$

Infinity reports:

$$\epsilon_\infty = \ln\left(\frac{p_1(1-q_1)}{(1-p_1)q_1}\right)$$

Given SUE and OUE:

- Apply OUE twice (L-OUE);
- Apply SUE twice (L-SUE);
- OUE then SUE (L-OSUE);
- SUE then OUE (L-SOUE).

* Arcolezi, H.H., Couchot, J.F., Al Bouna, B., and Xiao, X. Improving the Utility of Locally Differentially Private Protocols for Longitudinal and Multidimensional Frequency Estimates. arXiv:2111.04636 (2021).

Multi-freq-ldpy for Long. Freq. Est.

3. **Longitudinal Single Frequency Estimation** -- All longitudinal LDP protocols from Improving the Utility of Locally Differentially Private Protocols for Longitudinal and Multidimensional Frequency Estimates following the memoization-based framework from RAPPOR, namely:
   - Longitudinal GRR (L-GRR): `multi_freq_ldpy.long_freq_est.L_GRR`
   - Longitudinal OUE (L-OUE): `multi_freq_ldpy.long_freq_est.L_OUE`
   - Longitudinal OUE-SUE (L-OSUE): `multi_freq_ldpy.long_freq_est.L_OSUE`
   - Longitudinal SUE (L-SUE): `multi_freq_ldpy.long_freq_est.L_SUE`
   - Longitudinal SUE-OUE (L-SOUE): `multi_freq_ldpy.long_freq_est.L_SOUE`
   - Longitudinal ADP (L-ADP), i.e., L-GRR or L-OSUE: `multi_freq_ldpy.long_freq_est.L_ADP`

PyPi Page: https://pypi.org/project/multi-freq-ldpy/
Pratical Demonstration: Colab Link or GitHub Link (tutorial 3)

Inria femto-st SCIENCES & TECHNOLOGIES UQÀM Université du Québec à Montréal

Outline

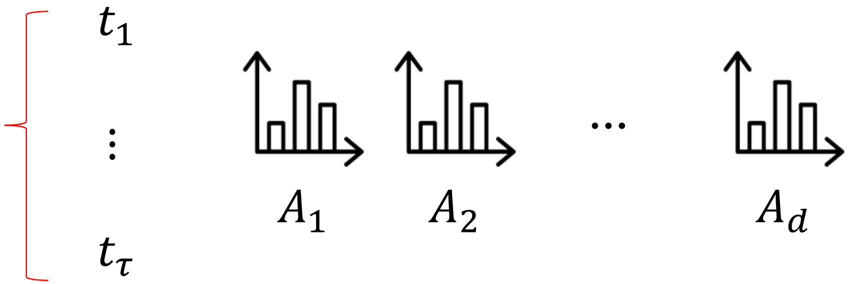
1. Introduction
2. Single Frequency Estimation
3. Frequency Estimation of Multiple Attributes
4. Single Longitudinal Frequency Estimation
5. **Longitudinal Frequency Estimation of Multiple Attributes**
6. Conclusion & Perspectives

Long. and Multid. Frequency Estimation

- **Tackled Issue:** Collecting *multidimensional* data under ϵ-*LDP* throughout time (i.e., *longitudinal study*) for *frequency estimation*.

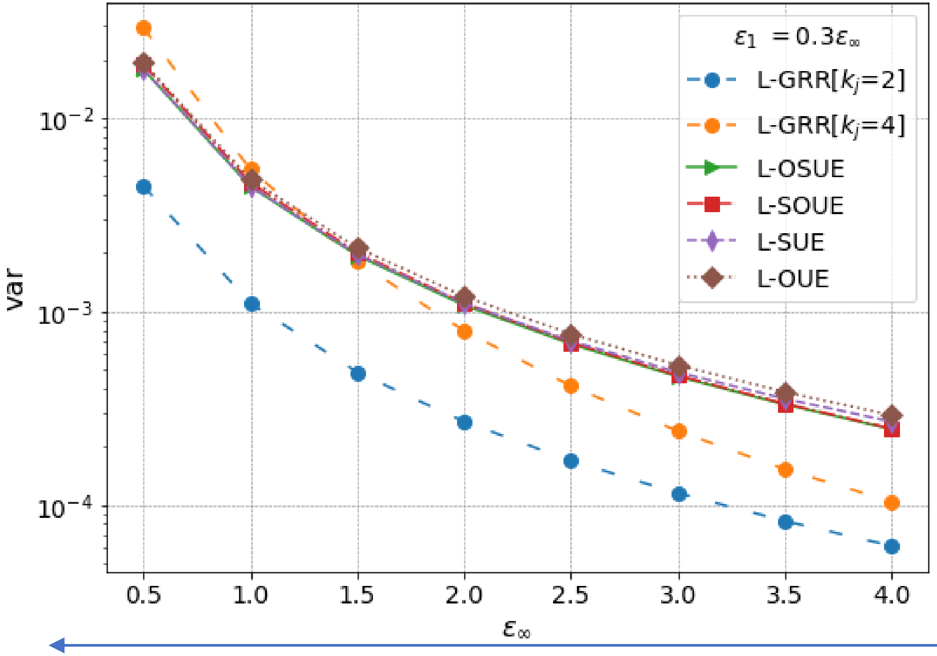
- **More formally (notation):**
  - d attributes $A = \{A_1, A_2, \dots, A_d\}$; → **Multiple attributes**
  - Each attribute A_j has a discrete domain of size $|A_j| = k_j$;
  - Each user u_i for $1 \leq i \leq n$ has a tuple $\mathbf{v}^i = (v_1^i, v_2^i, \dots, v_d^i)$;
  - **Analyzer:** estimate a k_j-bins histogram for each attribute $j \in [1, d]$.

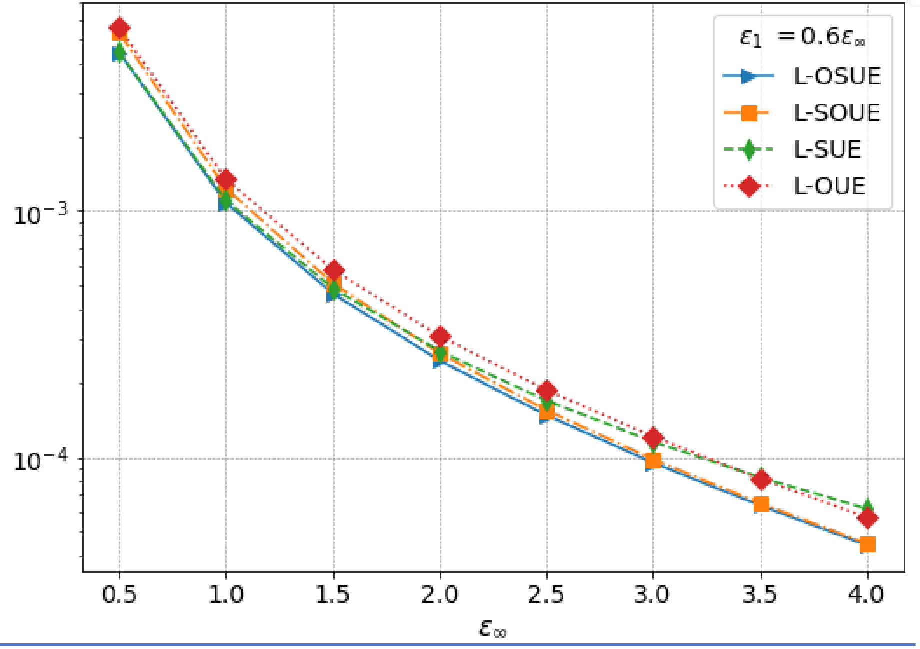
Multiple collection



Num. Eval. of L-GRR and L-UE Variances

More accurate





More private

Adaptive protocol (ADP)*: $\min\left(Var^*\left[\hat{f}_{L_{(L-GRR)}}\right], Var^*\left[\hat{f}_{L_{(L-OSUE)}}\right]\right)$

Inria femto-st SCIENCES & TECHNOLOGIES UQÀM Université du Québec à Montréal

* Arcolezi, H.H., Couchot, J.F., Al Bouna, B., and Xiao, X. Improving the Utility of Locally Differentially Private Protocols for Longitudinal and Multidimensional Frequency Estimates. arXiv:2111.04636 (2021).

Multi-freq-ldpy for Long./Multid. Freq. Est.

4. **Longitudinal Multidimensional Frequency Estimation** -- Both SPL and SMP solutions with all longitudinal protocols from previous point 3, namely:
   - Longitudinal SPL (L_SPL): `multi_freq_ldpy.long_mdim_freq_est.L_SPL`
   - Longitudinal SMP (L_SMP): `multi_freq_ldpy.long_mdim_freq_est.L_SMP`

PyPi Page: https://pypi.org/project/multi-freq-ldpy/

Pratical Demonstration: Colab Link or GitHub Link (tutorial 4)

Outline

Conclusion & Perspectives

General Conclusion:

- Multi-Freq-LDPy has been developed with ease of use and fast execution in mind;
- The package is accessible through PyPI under an MIT license;
- This package features separate and combined multidimensional and longitudinal frequency estimation.

Perspectives:

- Extend and integrate RS+FD solution with LH-based protocols;
- Extend and integrate longitudinal LH-based protocols;
- Add two more longitudinal protocols (original RAPPOR* and dBitFlip**);

* Erlingsson, Ú., Pihur, V., Korolova, A. RAPPOR: Randomized Aggregatable Privacy-Preserving Ordinal Response. In: ACM SIGSAC (2014).
** Ding, B., Kulkarni, J., Yekhanin, S. Collecting telemetry data privately. In: NeurIPS (2017).

Thank you for your attention!

Multi-Freq-LDPy: https://github.com/hharcolezi/multi-freq-ldpy

Please star ⋆ our GitHub repository, fork it, and contribute with us through pull requests.

Your feedback will be most welcome!

Contact: Héber H. Arcolezi (heber.hwang-arcolezi [at] inria.fr)