Utility-Optimized Local Differential Privacy Mechanisms for Distribution Estimation

Takao Murakami and Yusuke Kawamoto, *AIST*

https://www.usenix.org/conference/usenixsecurity19/presentation/murakami

This paper is included in the Proceedings of the 28th USENIX Security Symposium.

August 14–16, 2019 • Santa Clara, CA, USA

978-1-939133-06-9

Utility-Optimized Local Differential Privacy Mechanisms for Distribution Estimation [*]

Takao Murakami
AIST

Yusuke Kawamoto
AIST

Abstract

LDP (Local Differential Privacy) has been widely studied to estimate statistics of personal data (e.g., distribution underlying the data) while protecting users' privacy. Although LDP does not require a trusted third party, it regards all personal data equally sensitive, which causes excessive obfuscation hence the loss of utility. In this paper, we introduce the notion of *ULDP (Utility-optimized LDP)*, which provides a privacy guarantee equivalent to LDP only for sensitive data. We first consider the setting where all users use the same obfuscation mechanism, and propose two mechanisms providing ULDP: *utility-optimized randomized response* and *utility-optimized RAPPOR*. We then consider the setting where the distinction between sensitive and non-sensitive data can be different from user to user. For this setting, we propose a *personalized ULDP mechanism with semantic tags* to estimate the distribution of personal data with high utility while keeping secret what is sensitive for each user. We show theoretically and experimentally that our mechanisms provide much higher utility than the existing LDP mechanisms when there are a lot of non-sensitive data. We also show that when most of the data are non-sensitive, our mechanisms even provide almost the same utility as non-private mechanisms in the low privacy regime.

1 Introduction

DP (Differential Privacy) [21, 22] is becoming a gold standard for data privacy; it enables big data analysis while protecting users' privacy against adversaries with arbitrary background knowledge. According to the underlying architecture, DP can be categorized into the one in the *centralized model* and the one in the *local model* [22]. In the centralized model, a "trusted" database administrator, who can access to all users' personal data, obfuscates the data (e.g., by adding noise, generalization) before providing them to a (possibly malicious) data analyst. Although DP was extensively studied for the centralized model at the beginning, the original personal data in this model can be leaked from the database by illegal access or internal fraud. This issue is critical in recent years, because the number of data breach incidents is increasing [15].

The local model does not require a "trusted" administrator, and therefore does not suffer from the data leakage issue explained above. In this model, each user obfuscates her personal data by herself, and sends the obfuscated data to a data collector (or data analyst). Based on the obfuscated data, the data collector can estimate some statistics (e.g., histogram, heavy hitters [45]) of the personal data. DP in the local model, which is called *LDP* (*Local Differential Privacy*) [19], has recently attracted much attention in the academic field [5, 12, 24, 29, 30, 39, 43, 45, 46, 50, 56], and has also been adopted by industry [16, 23, 49].

However, LDP mechanisms regard all personal data as equally sensitive, and leave a lot of room for increasing data utility. For example, consider questionnaires such as: "Have you ever cheated in an exam?" and "Were you with a prostitute in the last month?" [11]. Obviously, "Yes" is a sensitive response to these questionnaires, whereas "No" is not sensitive. A RR (Randomized Response) method proposed by Mangat [37] utilizes this fact. Specifically, it reports "Yes" or "No" as follows: if the true answer is "Yes", always report "Yes"; otherwise, report "Yes" and "No" with probability p and $1-p$, respectively. Since the reported answer "Yes" may come from both the true answers "Yes" and "No", the confidentiality of the user reporting "Yes" is not violated. Moreover, since the reported answer "No" is always come from the true answer "No", the data collector can estimate a distribution of true answers with higher accuracy than Warner's RR [52], which simply flips "Yes" and "No" with probability p. However, Mangat's RR does not provide LDP, since LDP regards both "Yes" and "No" as equally sensitive.

There are a lot of "non-sensitive" data for other types of data. For example, locations such as hospitals and home can be sensitive, whereas visited sightseeing places, restaurants, and coffee shops are non-sensitive for many users. Divorced people may want to keep their divorce secret, while the oth-

[*] This study was supported by JSPS KAKENHI JP19H04113, JP17K12667, and by Inria under the project LOGIS.

ers may not care about their marital status. The distinction between sensitive and non-sensitive data can also be different from user to user (e.g., home address is different from user to user; some people might want to keep secret even the sightseeing places). To explain more about this issue, we briefly review related work on LDP and variants of DP.

Related work. Since Dwork [21] introduced DP, a number of its variants have been studied to provide different types of privacy guarantees; e.g., LDP [19], d-privacy [8], Pufferfish privacy [32], dependent DP [36], Bayesian DP [53], mutual-information DP [14], Rényi DP [38], and distribution privacy [31]. In particular, LDP [19] has been widely studied in the literature. For example, Erlingsson *et al.* [23] proposed the RAPPOR as an obfuscation mechanism providing LDP, and implemented it in Google Chrome browser. Kairouz *et al.* [29] showed that under the l_1 and l_2 losses, the randomized response (generalized to multiple alphabets) and RAPPOR are order optimal among all LDP mechanisms in the low and high privacy regimes, respectively. Wang *et al.* [51] generalized the RAPPOR and a random projection-based method [6], and found parameters that minimize the variance of the estimate.

Some studies also attempted to address the non-uniformity of privacy requirements among records (rows) or among items (columns) in the centralized DP: Personalized DP [28], Heterogeneous DP [3], and One-sided DP [17]. However, obfuscation mechanisms that address the non-uniformity among input values in the "local" DP have not been studied, to our knowledge. In this paper, we show that data utility can be significantly increased by designing such local mechanisms.

Our contributions. The goal of this paper is to design obfuscation mechanisms in the local model that achieve high data utility while providing DP for sensitive data. To achieve this, we introduce the notion of *ULDP (Utility-optimized LDP)*, which provides a privacy guarantee equivalent to LDP only for sensitive data, and obfuscation mechanisms providing ULDP. As a task for the data collector, we consider *discrete distribution estimation* [2, 23, 24, 27, 29, 39, 46, 56], where personal data take discrete values. Our contributions are as follows:

- We first consider the setting in which all users use the same obfuscation mechanism, and propose two ULDP mechanisms: *utility-optimized RR* and *utility-optimized RAPPOR*. We prove that when there are a lot of non-sensitive data, our mechanisms provide much higher utility than two state-of-the-art LDP mechanisms: the RR (for multiple alphabets) [29, 30] and RAPPOR [23]. We also prove that when most of the data are non-sensitive, our mechanisms even provide almost the same utility as a non-private mechanism that does not obfuscate the personal data in the low privacy regime where the privacy budget is $\varepsilon = \ln|\mathcal{X}|$ for a set $\mathcal{X}$ of personal data.

- We then consider the setting in which the distinction between sensitive and non-sensitive data can be different from user to user, and propose a *PUM (Personalized ULDP Mechanism) with semantic tags*. The PUM keeps secret what is sensitive for each user, while enabling the data collector to estimate a distribution using some background knowledge about the distribution conditioned on each tag (e.g., geographic distributions of homes). We also theoretically analyze the data utility of the PUM.

- We finally show that our mechanisms are very promising in terms of utility using two large-scale datasets.

The proofs of all statements in the paper are given in the extended version of the paper [40].

Cautions and limitations. Although ULDP is meant to protect sensitive data, there are some cautions and limitations.

First, we assume that each user sends a single datum and that each user's personal data is independent (see Section 2.1). This is reasonable for a variety of personal data (e.g., locations, age, sex, marital status), where each user's data is irrelevant to most others' one. However, for some types of personal data (e.g., flu status [48]), each user can be highly influenced by others. There might also be a correlation between sensitive data and non-sensitive data when a user sends multiple data (on a related note, non-sensitive attributes may lead to re-identification of a record [41]). A possible solution to these problems would be to incorporate ULDP with *Pufferfish privacy* [32, 48], which is used to protect correlated data. We leave this as future work (see Section 7 for discussions on the case of multiple data per user and the correlation issue).

We focus on a scenario in which it is easy for users to decide what is sensitive (e.g., cheating experience, location of home). However, there is also a scenario in which users do not know what is sensitive. For the latter scenario, we cannot use ULDP but can simply apply LDP.

Apart from the sensitive/non-sensitive data issue, there are scenarios in which ULDP does not cover. For example, ULDP does not protect users who have a sensitivity about "information disclosure" itself (i.e., those who will not disclose any information). We assume that users have consented to information disclosure. To collect as much data as possible, we can provide an incentive for the information disclosure; e.g., provide a reward or point-of-interest (POI) information nearby a reported location. We also assume that the data collector obtains a consensus from users before providing reported data to third parties. Note that these cautions are common to LDP.

There might also be a risk of discrimination; e.g., the data collector might discriminate against all users that provide a yes-answer, and have no qualms about small false positives. False positives decrease with increase in ε. We note that LDP also suffer from this attack; the false positive probability is the same for both ULDP and LDP with the same ε.

In summary, ULDP provides a privacy guarantee equivalent to LDP for sensitive data under the assumption of the data independence. We consider our work as a building-block of broader DP approaches or the basis for further development.

2 Preliminaries

2.1 Notations

Let $\mathbb{R}_{\geq 0}$ be the set of non-negative real numbers. Let n be the number of users, $[n] = \{1, 2, \dots, n\}$, $\mathcal{X}$ (resp. $\mathcal{Y}$) be a finite set of personal (resp. obfuscated) data. We assume continuous data are discretized into bins in advance (e.g., a location map is divided into some regions). We use the superscript "(i)" to represent the i-th user. Let $X^{(i)}$ (resp. $Y^{(i)}$) be a random variable representing personal (resp. obfuscated) data of the i-th user. The i-th user obfuscates her personal data $X^{(i)}$ via her obfuscation mechanism $\mathbf{Q}^{(i)}$, which maps $x \in \mathcal{X}$ to $y \in \mathcal{Y}$ with probability $\mathbf{Q}^{(i)}(y|x)$, and sends the obfuscated data $Y^{(i)}$ to a data collector. Here we assume that each user sends a single datum. We discuss the case of multiple data in Section 7.

We divide personal data into two types: *sensitive data* and *non-sensitive data*. Let $\mathcal{X}_S \subseteq \mathcal{X}$ be a set of sensitive data common to all users, and $\mathcal{X}_N = \mathcal{X} \setminus \mathcal{X}_S$ be the remaining personal data. Examples of such "common" sensitive data $x \in \mathcal{X}_S$ are the regions including public sensitive locations (e.g., hospitals) and obviously sensitive responses to questionnaires described in Section 1[1].

Furthermore, let $\mathcal{X}_S^{(i)} \subseteq \mathcal{X}_N$ ($i \in [n]$) be a set of sensitive data specific to the i-th user (here we do not include $\mathcal{X}_S$ into $\mathcal{X}_S^{(i)}$ because $\mathcal{X}_S$ is protected for all users in our mechanisms). $\mathcal{X}_S^{(i)}$ is a set of personal data that is possibly non-sensitive for many users but sensitive for the i-th user. Examples of such "user-specific" sensitive data $x \in \mathcal{X}_S^{(i)}$ are the regions including private locations such as their home and workplace. (Note that the majority of working population can be uniquely identified from their home/workplace location pairs [25].)

In Sections 3 and 4, we consider the case where all users divide $\mathcal{X}$ into the same sets of sensitive data and of non-sensitive data, i.e., $\mathcal{X}_S^{(1)} = \dots = \mathcal{X}_S^{(n)} = \emptyset$, and use the same obfuscation mechanism $\mathbf{Q}$ (i.e., $\mathbf{Q} = \mathbf{Q}^{(1)} = \dots = \mathbf{Q}^{(n)}$). In Section 5, we consider a general setting that can deal with the user-specific sensitive data $\mathcal{X}_S^{(i)}$ and user-specific mechanisms $\mathbf{Q}^{(i)}$. We call the former case a *common-mechanism scenario* and the latter a *personalized-mechanism scenario*.

We assume that each user's personal data $X^{(i)}$ is independently and identically distributed (i.i.d.) with a probability distribution $\mathbf{p}$, which generates $x \in \mathcal{X}$ with probability $\mathbf{p}(x)$. Let $\mathbf{X} = (X^{(1)}, \cdots, X^{(n)})$ and $\mathbf{Y} = (Y^{(1)}, \cdots, Y^{(n)})$ be tuples of all personal data and all obfuscated data, respectively. The data collector estimates $\mathbf{p}$ from $\mathbf{Y}$ by a method described in Section 2.5. We denote by $\hat{\mathbf{p}}$ the estimate of $\mathbf{p}$. We further denote by $\mathcal{C}$ the probability simplex; i.e., $\mathcal{C} = \{\mathbf{p} | \sum_{x \in \mathcal{X}} \mathbf{p}(x) = 1, \mathbf{p}(x) \geq 0 \text{ for any } x \in \mathcal{X}\}$.

2.2 Privacy Measures

LDP (Local Differential Privacy) [19] is defined as follows:

Definition 1 (ε-LDP). *Let $\varepsilon \in \mathbb{R}_{\geq 0}$. An obfuscation mechanism $\mathbf{Q}$ from $\mathcal{X}$ to $\mathcal{Y}$ provides* ε-LDP *if for any $x, x' \in \mathcal{X}$ and any $y \in \mathcal{Y}$,*

$$\mathbf{Q}(y|x) \leq e^{\varepsilon} \mathbf{Q}(y|x'). \tag{1}$$

LDP guarantees that an adversary who has observed y cannot determine, for any pair of x and x', whether it is come from x or x' with a certain degree of confidence. As the privacy budget ε approaches to 0, all of the data in $\mathcal{X}$ become almost equally likely. Thus, a user's privacy is strongly protected when ε is small.

2.3 Utility Measures

In this paper, we use the l_1 loss (i.e., absolute error) and the l_2 loss (i.e., squared error) as utility measures. Let l_1 (resp. l_2^2) be the l_1 (resp. l_2) loss function, which maps the estimate $\hat{\mathbf{p}}$ and the true distribution $\mathbf{p}$ to the loss; i.e., $l_1(\hat{\mathbf{p}}, \mathbf{p}) = \sum_{x \in \mathcal{X}} |\hat{\mathbf{p}}(x) - \mathbf{p}(x)|$, $l_2^2(\hat{\mathbf{p}}, \mathbf{p}) = \sum_{x \in \mathcal{X}} (\hat{\mathbf{p}}(x) - \mathbf{p}(x))^2$. It should be noted that $\mathbf{X}$ is generated from $\mathbf{p}$ and $\mathbf{Y}$ is generated from $\mathbf{X}$ using $\mathbf{Q}^{(1)}, \cdots, \mathbf{Q}^{(n)}$. Since $\hat{\mathbf{p}}$ is computed from $\mathbf{Y}$, both the l_1 and l_2 losses depend on $\mathbf{Y}$.

In our theoretical analysis in Sections 4 and 5, we take the expectation of the l_1 loss over all possible realizations of $\mathbf{Y}$. In our experiments in Section 6, we replace the expectation of the l_1 loss with the sample mean over multiple realizations of $\mathbf{Y}$ and divide it by 2 to evaluate the TV (Total Variation). In Appendix C, we also show that the l_2 loss has similar results to the ones in Sections 4 and 6 by evaluating the expectation of the l_2 loss and the MSE (Mean Squared Error), respectively.

2.4 Obfuscation Mechanisms

We describe the RR (Randomized Response) [29, 30] and a generalized version of the RAPPOR [51] as follows.

Randomized response. The RR for $|\mathcal{X}|$-ary alphabets was studied in [29, 30]. Its output range is identical to the input domain; i.e., $\mathcal{X} = \mathcal{Y}$.

Formally, given $\varepsilon \in \mathbb{R}_{\geq 0}$, the ε-*RR* is an obfuscation mechanism that maps x to y with the probability:

$$\mathbf{Q}_{RR}(y|x) = \begin{cases} \frac{e^{\varepsilon}}{|\mathcal{X}| + e^{\varepsilon} - 1} & (\text{if } y = x) \\ \frac{1}{|\mathcal{X}| + e^{\varepsilon} - 1} & (\text{otherwise}). \end{cases} \tag{2}$$

It is easy to check by (1) and (2) that $\mathbf{Q}_{RR}$ provides ε-LDP.

Generalized RAPPOR. The RAPPOR (Randomized Aggregatable Privacy-Preserving Ordinal Response) [23] is an obfuscation mechanism implemented in Google Chrome browser. Wang *et al.* [51] extended its simplest configuration called the basic one-time RAPPOR by generalizing two

[1] Note that these data might be sensitive for many/most users but not for all in practice (e.g., some people might not care about their cheating experience). However, we can regard these data as sensitive for all users (i.e., be on the safe side) by allowing a small loss of data utility.

probabilities in perturbation. Here we call it the *generalized RAPPOR* and describe its algorithm in detail.

The generalized RAPPOR is an obfuscation mechanism with the input alphabet $\mathcal{X} = \{x_1, x_2, \cdots, x_{|\mathcal{X}|}\}$ and the output alphabet $\mathcal{Y} = \{0,1\}^{|\mathcal{X}|}$. It first deterministically maps $x_i \in \mathcal{X}$ to $e_i \in \{0,1\}^{|\mathcal{X}|}$, where e_i is the i-th standard basis vector. It then probabilistically flips each bit of e_i to obtain obfuscated data $y = (y_1, y_2, \cdots, y_{|\mathcal{X}|}) \in \{0,1\}^{|\mathcal{X}|}$, where $y_i \in \{0,1\}$ is the i-th element of y. Wang *et al.* [51] compute ε from two parameters $\theta \in [0,1]$ (representing the probability of keeping 1 unchanged) and $\psi \in [0,1]$ (representing the probability of flipping 0 into 1). In this paper, we compute ψ from two parameters θ and ε.

Specifically, given $\theta \in [0,1]$ and $\varepsilon \in \mathbb{R}_{\geq 0}$, the (θ, ε)-*generalized RAPPOR* maps x_i to y with the probability:

$$\mathbf{Q}_{RAP}(y|x_i) = \textstyle\prod_{1 \leq j \leq |\mathcal{X}|} \Pr(y_j|x_i),$$

where $\Pr(y_j|x_i) = \theta$ if $i = j$ and $y_j = 1$, and $\Pr(y_j|x_i) = 1 - \theta$ if $i = j$ and $y_j = 0$, and $\Pr(y_j|x_i) = \psi = \frac{\theta}{(1-\theta)e^\varepsilon + \theta}$ if $i \neq j$ and $y_j = 1$, and $\Pr(y_j|x_i) = 1 - \psi$ otherwise. The basic one-time RAPPOR [23] is a special case of the generalized RAPPOR where $\theta = \frac{e^{\varepsilon/2}}{e^{\varepsilon/2}+1}$. $\mathbf{Q}_{RAP}$ also provides ε-LDP.

2.5 Distribution Estimation Methods

Here we explain the empirical estimation method [2, 27, 29] and the EM reconstruction method [1, 2]. Both of them assume that the data collector knows the obfuscation mechanism $\mathbf{Q}$ used to generate $\mathbf{Y}$ from $\mathbf{X}$.

Empirical estimation method. The empirical estimation method [2, 27, 29] computes an empirical estimate $\hat{\mathbf{p}}$ of $\mathbf{p}$ using an empirical distribution $\hat{\mathbf{m}}$ of the obfuscated data $\mathbf{Y}$. Note that $\hat{\mathbf{p}}$, $\hat{\mathbf{m}}$, and $\mathbf{Q}$ can be represented as an $|\mathcal{X}|$-dimensional vector, $|\mathcal{Y}|$-dimensional vector, and $|\mathcal{X}| \times |\mathcal{Y}|$ matrix, respectively. They have the following equation:

$$\hat{\mathbf{p}}\mathbf{Q} = \hat{\mathbf{m}}. \tag{3}$$

The empirical estimation method computes $\hat{\mathbf{p}}$ by solving (3).

Let $\mathbf{m}$ be the true distribution of obfuscated data; i.e., $\mathbf{m} = \mathbf{p}\mathbf{Q}$. As the number of users n increases, the empirical distribution $\hat{\mathbf{m}}$ converges to $\mathbf{m}$. Therefore, the empirical estimate $\hat{\mathbf{p}}$ also converges to $\mathbf{p}$. However, when the number of users n is small, many elements in $\hat{\mathbf{p}}$ can be negative. To address this issue, the studies in [23, 51] kept only estimates above a significance threshold determined via Bonferroni correction, and discarded the remaining estimates.

EM reconstruction method. The EM (Expectation-Maximization) reconstruction method [1, 2] (also called the iterative Bayesian technique [2]) regards $\mathbf{X}$ as a hidden variable and estimates $\mathbf{p}$ from $\mathbf{Y}$ using the EM algorithm [26] (for details of the algorithm, see [1, 2]). Let $\hat{\mathbf{p}}_{EM}$ be an estimate of $\mathbf{p}$ by the EM reconstruction method. The feature of this algorithm is that $\hat{\mathbf{p}}_{EM}$ is equal to the maximum likelihood estimate in the probability simplex $\mathcal{C}$ (see [1] for the proof). Since this property holds irrespective of the number of users n, the elements in $\hat{\mathbf{p}}_{EM}$ are always non-negative.

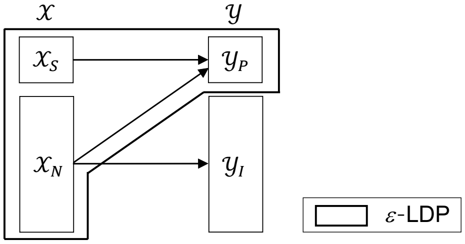

Figure 1: Overview of ULDP. It has no transitions from $\mathcal{X}_S$ to $\mathcal{Y}_I$, and every output in $\mathcal{Y}_I$ reveals the corresponding input in $\mathcal{X}_N$. It also provides ε-LDP for $\mathcal{Y}_P$.

In this paper, our theoretical analysis uses the empirical estimation method for simplicity, while our experiments use the empirical estimation method, the one with the significance threshold, and the EM reconstruction method.

3 Utility-Optimized LDP (ULDP)

In this section, we focus on the common-mechanism scenario (outlined in Section 2.1) and introduce ULDP (Utility-optimized Local Differential Privacy), which provides a privacy guarantee equivalent to ε-LDP only for sensitive data. Section 3.1 provides the definition of ULDP. Section 3.2 shows some theoretical properties of ULDP.

3.1 Definition

Figure 1 shows an overview of ULDP. An obfuscation mechanism providing ULDP, which we call the utility-optimized mechanism, divides obfuscated data into *protected data* and *invertible data*. Let $\mathcal{Y}_P$ be a set of protected data, and $\mathcal{Y}_I = \mathcal{Y} \setminus \mathcal{Y}_P$ be a set of invertible data.

The feature of the utility-optimized mechanism is that it maps sensitive data $x \in \mathcal{X}_S$ to only protected data $y \in \mathcal{Y}_P$. In other words, *it restricts the output set, given the input $x \in \mathcal{X}_S$, to $\mathcal{Y}_P$*. Then it provides ε-LDP for $\mathcal{Y}_P$; i.e., $\mathbf{Q}(y|x) \leq e^\varepsilon \mathbf{Q}(y|x')$ for any $x, x' \in \mathcal{X}$ and any $y \in \mathcal{Y}_P$. By this property, a privacy guarantee equivalent to ε-LDP is provided for any sensitive data $x \in \mathcal{X}_S$, since the output set corresponding to $\mathcal{X}_S$ is restricted to $\mathcal{Y}_P$. In addition, every output in $\mathcal{Y}_I$ reveals the corresponding input in $\mathcal{X}_N$ (as in Mangat's randomized response [37]) to optimize the estimation accuracy.

We now formally define ULDP and the utility-optimized mechanism:

Definition 2 ($(\mathcal{X}_S, \mathcal{Y}_P, \varepsilon)$-ULDP)**.** *Given $\mathcal{X}_S \subseteq \mathcal{X}$, $\mathcal{Y}_P \subseteq \mathcal{Y}$, and $\varepsilon \in \mathbb{R}_{\geq 0}$, an obfuscation mechanism $\mathbf{Q}$ from $\mathcal{X}$ to $\mathcal{Y}$ provides $(\mathcal{X}_S, \mathcal{Y}_P, \varepsilon)$-ULDP if it satisfies the following properties:*

1. *For any $y \in \mathcal{Y}_I$, there exists an $x \in \mathcal{X}_N$ such that*

$$\mathbf{Q}(y|x) > 0 \text{ and } \mathbf{Q}(y|x') = 0 \text{ for any } x' \neq x. \quad (4)$$

2. *For any $x, x' \in \mathcal{X}$ and any $y \in \mathcal{Y}_P$,*

$$\mathbf{Q}(y|x) \leq e^{\varepsilon}\mathbf{Q}(y|x'). \quad (5)$$

We refer to an obfuscation mechanism $\mathbf{Q}$ *providing $(\mathcal{X}_S, \mathcal{Y}_P, \varepsilon)$-ULDP as the $(\mathcal{X}_S, \mathcal{Y}_P, \varepsilon)$-utility-optimized mechanism.*

Example. For an intuitive understanding of Definition 2, we show that Mangat's randomized response [37] provides $(\mathcal{X}_S, \mathcal{Y}_P, \varepsilon)$-ULDP. As described in Section 1, this mechanism considers binary alphabets (i.e., $\mathcal{X} = \mathcal{Y} = \{0, 1\}$), and regards the value 1 as sensitive (i.e., $\mathcal{X}_S = \mathcal{Y}_P = \{1\}$). If the input value is 1, it always reports 1 as output. Otherwise, it reports 1 and 0 with probability p and $1 - p$, respectively. Obviously, this mechanism does not provide ε-LDP for any $\varepsilon \in [0, \infty)$. However, it provides $(\mathcal{X}_S, \mathcal{Y}_P, \ln \frac{1}{p})$-ULDP.

$(\mathcal{X}_S, \mathcal{Y}_P, \varepsilon)$-ULDP provides a privacy guarantee equivalent to ε-LDP for any sensitive data $x \in \mathcal{X}_S$, as explained above. On the other hand, no privacy guarantees are provided for non-sensitive data $x \in \mathcal{X}_N$ because every output in $\mathcal{Y}_I$ reveals the corresponding input in $\mathcal{X}_N$. However, it does not matter since non-sensitive data need not be protected. Protecting only minimum necessary data is the key to achieving locally private distribution estimation with high data utility.

We can apply any ε-LDP mechanism to the sensitive data in $\mathcal{X}_S$ to provide $(\mathcal{X}_S, \mathcal{Y}_P, \varepsilon)$-ULDP as a whole. In Sections 4.1 and 4.2, we propose a utility-optimized RR (Randomized Response) and utility-optimized RAPPOR, which apply the ε-RR and ε-RAPPOR, respectively, to the sensitive data $\mathcal{X}_S$.

In Appendix B, we also consider OSLDP (One-sided LDP), a local model version of OSDP introduced in a preprint [17], and explain the reason for using ULDP in this paper.

It might be better to generalize ULDP so that different levels of ε can be assigned to different sensitive data. We leave introducing such granularity as future work.

Remark. It should also be noted that the data collector needs to know $\mathbf{Q}$ to estimate $\mathbf{p}$ from $\mathbf{Y}$ (as described in Section 2.5), and that the $(\mathcal{X}_S, \mathcal{Y}_P, \varepsilon)$-utility-optimized mechanism $\mathbf{Q}$ itself includes the information on *what is sensitive for users* (i.e., the data collector learns whether each $x \in \mathcal{X}$ belongs to $\mathcal{X}_S$ or not by checking the values of $\mathbf{Q}(y|x)$ for all $y \in \mathcal{Y}$). This does not matter in the common-mechanism scenario, since the set $\mathcal{X}_S$ of sensitive data is common to all users (e.g., public hospitals). However, in the personalized-mechanism scenario, the $(\mathcal{X}_S \cup \mathcal{X}_S^{(i)}, \mathcal{Y}_P, \varepsilon)$-utility-optimized mechanism $\mathbf{Q}^{(i)}$, which expands the set $\mathcal{X}_S$ of personal data to $\mathcal{X}_S \cup \mathcal{X}_S^{(i)}$, includes the information on *what is sensitive for the i-th user*. Therefore, the data collector learns whether each $x \in \mathcal{X}_N$ belongs to $\mathcal{X}_S^{(i)}$ or not by checking the values of $\mathbf{Q}^{(i)}(y|x)$ for all $y \in \mathcal{Y}$, despite the fact that the i-th user wants to hide her user-specific sensitive data $\mathcal{X}_S^{(i)}$ (e.g., home, workplace). We address this issue in Section 5.

3.2 Basic Properties of ULDP

Previous work showed some basic properties of differential privacy (or its variant), such as compositionality [22] and immunity to post-processing [22]. We briefly explain theoretical properties of ULDP including the ones above.

Sequential composition. ULDP is preserved under adaptive sequential composition when the composed obfuscation mechanism maps sensitive data to pairs of protected data. Specifically, consider two mechanisms $\mathbf{Q}_0$ from $\mathcal{X}$ to $\mathcal{Y}_0$ and $\mathbf{Q}_1$ from $\mathcal{X}$ to $\mathcal{Y}_1$ such that $\mathbf{Q}_0$ (resp. $\mathbf{Q}_1$) maps sensitive data $x \in \mathcal{X}_S$ to protected data $y_0 \in \mathcal{Y}_{0P}$ (resp. $y_1 \in \mathcal{Y}_{1P}$). Then the sequential composition of $\mathbf{Q}_0$ and $\mathbf{Q}_1$ maps sensitive data $x \in \mathcal{X}_S$ to pairs (y_0, y_1) of protected data ranging over:

$$(\mathcal{Y}_0 \times \mathcal{Y}_1)_P = \{(y_0, y_1) \in \mathcal{Y}_0 \times \mathcal{Y}_1 \mid y_0 \in \mathcal{Y}_{0P} \text{ and } y_1 \in \mathcal{Y}_{1P}\}.$$

Then we obtain the following compositionality.

Proposition 1 (Sequential composition). *Let $\varepsilon_0, \varepsilon_1 \geq 0$. If* $\mathbf{Q}_0$ *provides $(\mathcal{X}_S, \mathcal{Y}_{0P}, \varepsilon_0)$-ULDP and* $\mathbf{Q}_1(y_0)$ *provides $(\mathcal{X}_S, \mathcal{Y}_{1P}, \varepsilon_1)$-ULDP for each $y_0 \in \mathcal{Y}_0$, then the sequential composition of* $\mathbf{Q}_0$ *and* $\mathbf{Q}_1$ *provides $(\mathcal{X}_S, (\mathcal{Y}_0 \times \mathcal{Y}_1)_P, \varepsilon_0 + \varepsilon_1)$-ULDP.*

For example, if we apply an obfuscation mechanism providing $(\mathcal{X}_S, \mathcal{Y}_P, \varepsilon)$-ULDP for t times, then we obtain $(\mathcal{X}_S, (\mathcal{Y}_P)^t, \varepsilon t)$-ULDP in total (this is derived by repeatedly using Proposition 1).

Post-processing. ULDP is immune to the post-processing by a randomized algorithm that *preserves data types*: protected data or invertible data. Specifically, if a mechanism $\mathbf{Q}_0$ provides $(\mathcal{X}_S, \mathcal{Y}_P, \varepsilon)$-ULDP and a randomized algorithm $\mathbf{Q}_1$ maps protected data over $\mathcal{Y}_P$ (resp. invertible data) to protected data over $\mathcal{Z}_P$ (resp. invertible data), then the composite function $\mathbf{Q}_1 \circ \mathbf{Q}_0$ provides $(\mathcal{X}_S, \mathcal{Z}_P, \varepsilon)$-ULDP.

Note that $\mathbf{Q}_1$ needs to preserve data types for utility; i.e., to make all $y \in \mathcal{Y}_I$ invertible (as in Definition 2) after post-processing. The DP guarantee for $y \in \mathcal{Y}_P$ is preserved by any post-processing algorithm. See Appendix A.1 for details.

Compatibility with LDP. Assume that data collectors A and B adopt a mechanism providing ULDP and a mechanism providing LDP, respectively. In this case, all protected data in the data collector A can be combined with all obfuscated data in the data collector B (i.e., data integration) to perform data analysis under LDP. See Appendix A.2 for details.

Lower bounds on the l_1 and l_2 losses. We present lower bounds on the l_1 and l_2 losses of any ULDP mechanism by using the fact that ULDP provides (5) for any $x, x' \in \mathcal{X}_S$ and any $y \in \mathcal{Y}_P$. Specifically, Duchi *et al.* [20] showed that for $\varepsilon \in [0, 1]$, the lower bounds on the l_1 and l_2 losses (minimax rates) of any ε-LDP mechanism can be expressed as $\Theta(\frac{|\mathcal{X}|}{\sqrt{n\varepsilon^2}})$

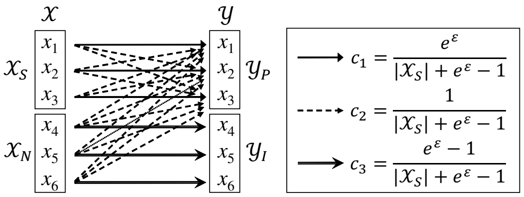
$\mathcal{X}$
$\mathcal{Y}$
$\mathcal{X}_S$
$\mathcal{X}_N$
x_1
x_2
x_3
x_4
x_5
x_6
$\mathcal{Y}_P$
$\mathcal{Y}_I$
$c_1 = \frac{e^\varepsilon}{|\mathcal{X}_S| + e^\varepsilon - 1}$
$c_2 = \frac{1}{|\mathcal{X}_S| + e^\varepsilon - 1}$
$c_3 = \frac{e^\varepsilon - 1}{|\mathcal{X}_S| + e^\varepsilon - 1}$

Figure 2: Utility-optimized RR in the case where $\mathcal{X}_S = \mathcal{Y}_P = \{x_1, x_2, x_3\}$ and $\mathcal{X}_N = \mathcal{Y}_I = \{x_4, x_5, x_6\}$.

and $\Theta(\frac{|\mathcal{X}|}{n\varepsilon^2})$, respectively. By directly applying these bounds to $\mathcal{X}_S$ and $\mathcal{Y}_P$, the lower bounds on the l_1 and l_2 losses of any $(\mathcal{X}_S, \mathcal{Y}_P, \varepsilon)$-ULDP mechanisms for $\varepsilon \in [0,1]$ can be expressed as $\Theta(\frac{|\mathcal{X}_S|}{\sqrt{n\varepsilon^2}})$ and $\Theta(\frac{|\mathcal{X}_S|}{n\varepsilon^2})$, respectively. In Section 4.3, we show that our utility-optimized RAPPOR achieves these lower bounds when ε is close to 0 (i.e., high privacy regime).

4 Utility-Optimized Mechanisms

In this section, we focus on the common-mechanism scenario and propose the *utility-optimized RR (Randomized Response)* and *utility-optimized RAPPOR* (Sections 4.1 and 4.2). We then analyze the data utility of these mechanisms (Section 4.3).

4.1 Utility-Optimized Randomized Response

We propose the utility-optimized RR, which is a generalization of Mangat's randomized response [37] to $|\mathcal{X}|$-ary alphabets with $|\mathcal{X}_S|$ sensitive symbols. As with the RR, the output range of the utility-optimized RR is identical to the input domain; i.e., $\mathcal{X} = \mathcal{Y}$. In addition, we divide the output set in the same way as the input set; i.e., $\mathcal{X}_S = \mathcal{Y}_P$, $\mathcal{X}_N = \mathcal{Y}_I$.

Figure 2 shows the utility-optimized RR with $\mathcal{X}_S = \mathcal{Y}_P = \{x_1, x_2, x_3\}$ and $\mathcal{X}_N = \mathcal{Y}_I = \{x_4, x_5, x_6\}$. The utility-optimized RR applies the ε-RR to $\mathcal{X}_S$. It maps $x \in \mathcal{X}_N$ to $y \in \mathcal{Y}_P$ $(= \mathcal{X}_S)$ with the probability $\mathbf{Q}(y|x)$ so that (5) is satisfied, and maps $x \in \mathcal{X}_N$ to itself with the remaining probability. Formally, we define the utility-optimized RR (uRR) as follows:

Definition 3 (($\mathcal{X}_S, \varepsilon$)-utility-optimized RR)**.** *Let* $\mathcal{X}_S \subseteq \mathcal{X}$ *and* $\varepsilon \in \mathbb{R}_{\geq 0}$. *Let* $c_1 = \frac{e^\varepsilon}{|\mathcal{X}_S| + e^\varepsilon - 1}$, $c_2 = \frac{1}{|\mathcal{X}_S| + e^\varepsilon - 1}$, *and* $c_3 = 1 - |\mathcal{X}_S| c_2 = \frac{e^\varepsilon - 1}{|\mathcal{X}_S| + e^\varepsilon - 1}$. *Then the* $(\mathcal{X}_S, \varepsilon)$*-utility-optimized RR (uRR) is an obfuscation mechanism that maps* $x \in \mathcal{X}$ *to* $y \in \mathcal{Y}$ $(= \mathcal{X})$ *with the probability* $\mathbf{Q}_{uRR}(y|x)$ *defined as follows:*

$$\mathbf{Q}_{uRR}(y|x) = \begin{cases} c_1 & (\textit{if } x \in \mathcal{X}_S,\ y = x) \\ c_2 & (\textit{if } x \in \mathcal{X}_S,\ y \in \mathcal{X}_S \setminus \{x\}) \\ c_2 & (\textit{if } x \in \mathcal{X}_N,\ y \in \mathcal{X}_S) \\ c_3 & (\textit{if } x \in \mathcal{X}_N,\ y = x) \\ 0 & (\textit{otherwise}). \end{cases} \quad (6)$$

Proposition 2. *The* $(\mathcal{X}_S, \varepsilon)$*-uRR provides* $(\mathcal{X}_S, \mathcal{X}_S, \varepsilon)$*-ULDP.*

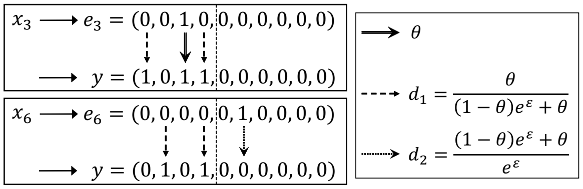
$x_3 \longrightarrow e_3 = (0,0,1,0,0,0,0,0,0,0)$
$\longrightarrow y = (1,0,1,1,0,0,0,0,0,0)$
$x_6 \longrightarrow e_6 = (0,0,0,0,0,1,0,0,0,0)$
$\longrightarrow y = (0,1,0,1,0,0,0,0,0,0)$
θ
$d_1 = \frac{\theta}{(1-\theta)e^\varepsilon + \theta}$
$d_2 = \frac{(1-\theta)e^\varepsilon + \theta}{e^\varepsilon}$

Figure 3: Utility-optimized RAPPOR in the case where $\mathcal{X}_S = \{x_1, \cdots, x_4\}$ and $\mathcal{X}_N = \{x_5, \cdots, x_{10}\}$.

4.2 Utility-Optimized RAPPOR

Next, we propose the utility-optimized RAPPOR with the input alphabet $\mathcal{X} = \{x_1, x_2, \cdots, x_{|\mathcal{X}|}\}$ and the output alphabet $\mathcal{Y} = \{0,1\}^{|\mathcal{X}|}$. Without loss of generality, we assume that $x_1, \cdots, x_{|\mathcal{X}_S|}$ are sensitive and $x_{|\mathcal{X}_S|+1}, \cdots, x_{|\mathcal{X}|}$ are non-sensitive; i.e., $\mathcal{X}_S = \{x_1, \cdots, x_{|\mathcal{X}_S|}\}$, $\mathcal{X}_N = \{x_{|\mathcal{X}_S|+1}, \cdots, x_{|\mathcal{X}|}\}$.

Figure 3 shows the utility-optimized RAPPOR with $\mathcal{X}_S = \{x_1, \cdots, x_4\}$ and $\mathcal{X}_N = \{x_5, \cdots, x_{10}\}$. The utility-optimized RAPPOR first deterministically maps $x_i \in \mathcal{X}$ to the i-th standard basis vector e_i. It should be noted that if x_i is sensitive data (i.e., $x_i \in \mathcal{X}_S$), then the last $|\mathcal{X}_N|$ elements in e_i are always zero (as shown in the upper-left panel of Figure 3). Based on this fact, the utility-optimized RAPPOR regards obfuscated data $y = (y_1, y_2, \ldots, y_{|\mathcal{X}|}) \in \{0,1\}^{|\mathcal{X}|}$ such that $y_{|\mathcal{X}_S|+1} = \cdots = y_{|\mathcal{X}|} = 0$ as protected data; i.e.,

$$\mathcal{Y}_P = \{(y_1, \ldots, y_{|\mathcal{X}_S|}, 0, \cdots, 0) | y_1, \ldots, y_{|\mathcal{X}_S|} \in \{0,1\}\}. \quad (7)$$

Then it applies the (θ, ε)-generalized RAPPOR to $\mathcal{X}_S$, and maps $x \in \mathcal{X}_N$ to $y \in \mathcal{Y}_P$ (as shown in the lower-left panel of Figure 3) with the probability $\mathbf{Q}(y|x)$ so that (5) is satisfied. We formally define the utility-optimized RAPPOR (uRAP):

Definition 4 (($\mathcal{X}_S, \theta, \varepsilon$)-utility-optimized RAPPOR)**.** *Let* $\mathcal{X}_S \subseteq \mathcal{X}$, $\theta \in [0,1]$, *and* $\varepsilon \in \mathbb{R}_{\geq 0}$. *Let* $d_1 = \frac{\theta}{(1-\theta)e^\varepsilon + \theta}$, $d_2 = \frac{(1-\theta)e^\varepsilon + \theta}{e^\varepsilon}$. *Then the* $(\mathcal{X}_S, \theta, \varepsilon)$*-utility-optimized RAPPOR (uRAP) is an obfuscation mechanism that maps* $x_i \in \mathcal{X}$ *to* $y \in \mathcal{Y} = \{0,1\}^{|\mathcal{X}|}$ *with the probability* $\mathbf{Q}_{uRAP}(y|x)$ *given by:*

$$\mathbf{Q}_{uRAP}(y|x_i) = \textstyle\prod_{1 \leq j \leq |\mathcal{X}|} \Pr(y_j | x_i), \quad (8)$$

where $\Pr(y_j|x_i)$ *is written as follows:*

(i) if $1 \leq j \leq |\mathcal{X}_S|$*:*

$$\Pr(y_j|x_i) = \begin{cases} 1 - \theta & (\textit{if } i = j,\ y_j = 0) \\ \theta & (\textit{if } i = j,\ y_j = 1) \\ 1 - d_1 & (\textit{if } i \neq j,\ y_j = 0) \\ d_1 & (\textit{if } i \neq j,\ y_j = 1). \end{cases} \quad (9)$$

(ii) if $|\mathcal{X}_S| + 1 \leq j \leq |\mathcal{X}|$*:*

$$\Pr(y_j|x_i) = \begin{cases} d_2 & (\textit{if } i = j,\ y_j = 0) \\ 1 - d_2 & (\textit{if } i = j,\ y_j = 1) \\ 1 & (\textit{if } i \neq j,\ y_j = 0) \\ 0 & (\textit{if } i \neq j,\ y_j = 1). \end{cases} \quad (10)$$

Proposition 3. *The* $(\mathcal{X}_S, \theta, \varepsilon)$*-uRAP provides* $(\mathcal{X}_S, \mathcal{Y}_P, \varepsilon)$*-ULDP, where* $\mathcal{Y}_P$ *is given by (7).*

Although we used the generalized RAPPOR in $\mathcal{X}_S$ and $\mathcal{Y}_P$ in Definition 4, hereinafter we set $\theta = \frac{e^{\varepsilon/2}}{e^{\varepsilon/2}+1}$ in the same way as the original RAPPOR [23]. There are two reasons for this. First, it achieves "order" optimal data utility among all $(\mathcal{X}_S, \mathcal{Y}_P, \varepsilon)$-ULDP mechanisms in the high privacy regime, as shown in Section 4.3. Second, it maps $x_i \in \mathcal{X}_N$ to $y \in \mathcal{Y}_I$ with probability $1 - d_2 = 1 - e^{-\varepsilon/2}$, which is close to 1 when ε is large (i.e., low privacy regime). Wang *et al.* [51] showed that the generalized RAPPOR with parameter $\theta = \frac{1}{2}$ minimizes the variance of the estimate. However, our uRAP with parameter $\theta = \frac{1}{2}$ maps $x_i \in \mathcal{X}_N$ to $y \in \mathcal{Y}_I$ with probability $1 - d_2 = \frac{e^{\varepsilon}-1}{2e^{\varepsilon}}$ which is less than $1 - e^{-\varepsilon/2}$ for any $\varepsilon > 0$ and is less than $\frac{1}{2}$ even when ε goes to infinity. Thus, our uRAP with $\theta = \frac{e^{\varepsilon/2}}{e^{\varepsilon/2}+1}$ maps $x_i \in \mathcal{X}_N$ to $y \in \mathcal{Y}_I$ with higher probability, and therefore achieves a smaller estimation error over all non-sensitive data. We also consider that an optimal θ for our uRAP is different from the optimal θ $(= \frac{1}{2})$ for the generalized RAPPOR. We leave finding the optimal θ for our uRAP (with respect to the estimation error over all personal data) as future work.

We refer to the $(\mathcal{X}_S, \theta, \varepsilon)$-uRAP with $\theta = \frac{e^{\varepsilon/2}}{e^{\varepsilon/2}+1}$ in shorthand as the $(\mathcal{X}_S, \varepsilon)$*-uRAP*.

4.3 Utility Analysis

We evaluate the l_1 loss of the uRR and uRAP when the empirical estimation method is used for distribution estimation[2]. In particular, we evaluate the l_1 loss when ε is close to 0 (i.e., high privacy regime) and $\varepsilon = \ln |\mathcal{X}|$ (i.e., low privacy regime). Note that ULDP provides a natural interpretation of the latter value of ε. Specifically, it follows from (5) that if $\varepsilon = \ln |\mathcal{X}|$, then for any $x \in \mathcal{X}$, the likelihood that the input data is x is almost equal to the sum of the likelihood that the input data is $x' \neq x$. This is consistent with the fact that the ε-RR with parameter $\varepsilon = \ln |\mathcal{X}|$ sends true data (i.e., $y = x$ in (2)) with probability about 0.5 and false data (i.e., $y \neq x$) with probability about 0.5, and hence provides plausible deniability [29].

uRR in the general case. We begin with the uRR:

Proposition 4 (l_1 loss of the uRR)**.** *Let* $\varepsilon \in \mathbb{R}_{\geq 0}$*,* $u = |\mathcal{X}_S| + e^{\varepsilon} - 1$*,* $u' = e^{\varepsilon} - 1$*, and* $v = \frac{u}{u'}$*. Then the expected* l_1 *loss of the* $(\mathcal{X}_S, \varepsilon)$*-uRR mechanism is given by:*

$$\mathbb{E}\left[l_1(\hat{\mathbf{p}}, \mathbf{p})\right] \approx \sqrt{\frac{2}{n\pi}} \Bigg(\sum_{x \in \mathcal{X}_S} \sqrt{\left(\mathbf{p}(x) + 1/u'\right)\left(v - \mathbf{p}(x) - 1/u'\right)} + \sum_{x \in \mathcal{X}_N} \sqrt{\mathbf{p}(x)\left(v - \mathbf{p}(x)\right)} \Bigg), \tag{11}$$

where $f(n) \approx g(n)$ *represents* $\lim_{n \to \infty} f(n)/g(n) = 1$.

Let $\mathbf{p}_{U_N}$ be the uniform distribution over $\mathcal{X}_N$; i.e., for any $x \in \mathcal{X}_S$, $\mathbf{p}_{U_N}(x) = 0$, and for any $x \in \mathcal{X}_N$, $\mathbf{p}_{U_N}(x) = \frac{1}{|\mathcal{X}_N|}$. Symmetrically, let $\mathbf{p}_{U_S}$ be the uniform distribution over $\mathcal{X}_S$.

For $0 < \varepsilon < \ln(|\mathcal{X}_N| + 1)$, the l_1 loss is maximized by $\mathbf{p}_{U_N}$:

Proposition 5. *For any* $0 < \varepsilon < \ln(|\mathcal{X}_N| + 1)$ *and* $|\mathcal{X}_S| \leq |\mathcal{X}_N|$*, (11) is maximized by* $\mathbf{p}_{U_N}$*:*

$$\begin{aligned} &\mathbb{E}\left[l_1(\hat{\mathbf{p}}, \mathbf{p})\right] \lesssim \mathbb{E}\left[l_1(\hat{\mathbf{p}}, \mathbf{p}_{U_N})\right] \\ &= \sqrt{\frac{2}{n\pi}} \left(\frac{|\mathcal{X}_S|\sqrt{|\mathcal{X}_S| + e^{\varepsilon} - 2}}{e^{\varepsilon} - 1} + \sqrt{\frac{|\mathcal{X}_S||\mathcal{X}_N|}{e^{\varepsilon} - 1} + |\mathcal{X}_N| - 1} \right), \end{aligned} \tag{12}$$

where $f(n) \lesssim g(n)$ *represents* $\lim_{n \to \infty} f(n)/g(n) \leq 1$.

For $\varepsilon \geq \ln(|\mathcal{X}_N| + 1)$, the l_1 loss is maximized by a mixture distribution of $\mathbf{p}_{U_N}$ and $\mathbf{p}_{U_S}$:

Proposition 6. *Let* $\mathbf{p}^*$ *be a distribution over* $\mathcal{X}$ *defined by:*

$$\mathbf{p}^*(x) = \begin{cases} \frac{1 - |\mathcal{X}_N|/(e^{\varepsilon} - 1)}{|\mathcal{X}_S| + |\mathcal{X}_N|} & \textit{(if } x \in \mathcal{X}_S\textit{)} \\ \frac{1 + |\mathcal{X}_S|/(e^{\varepsilon} - 1)}{|\mathcal{X}_S| + |\mathcal{X}_N|} & \textit{(otherwise)} \end{cases} \tag{13}$$

Then for any $\varepsilon \geq \ln(|\mathcal{X}_N| + 1)$*, (11) is maximized by* $\mathbf{p}^*$*:*

$$\mathbb{E}\left[l_1(\hat{\mathbf{p}}, \mathbf{p})\right] \lesssim \mathbb{E}\left[l_1(\hat{\mathbf{p}}, \mathbf{p}^*)\right] = \sqrt{\frac{2(|\mathcal{X}| - 1)}{n\pi}} \cdot \frac{|\mathcal{X}_S| + e^{\varepsilon} - 1}{e^{\varepsilon} - 1}, \tag{14}$$

where $f(n) \lesssim g(n)$ *represents* $\lim_{n \to \infty} f(n)/g(n) \leq 1$.

Next, we instantiate the l_1 loss in the high and low privacy regimes based on these propositions.

uRR in the high privacy regime. When ε is close to 0, we have $e^{\varepsilon} - 1 \approx \varepsilon$. Thus, the right-hand side of (12) in Proposition 5 can be simplified as follows:

$$\mathbb{E}\left[l_1(\hat{\mathbf{p}}, \mathbf{p}_{U_N})\right] \approx \sqrt{\frac{2}{n\pi}} \cdot \frac{|\mathcal{X}_S|\sqrt{|\mathcal{X}_S| - 1}}{\varepsilon}. \tag{15}$$

It was shown in [29] that the expected l_1 loss of the ε-RR is at most $\sqrt{\frac{2}{n\pi}} \frac{|\mathcal{X}|\sqrt{|\mathcal{X}| - 1}}{\varepsilon}$ when $\varepsilon \approx 0$. The right-hand side of (15) is much smaller than this when $|\mathcal{X}_S| \ll |\mathcal{X}|$. Although both of them are "upper-bounds" of the expected l_1 losses, we show that the total variation of the $(\mathcal{X}_S, \varepsilon)$-uRR is also much smaller than that of the ε-RR when $|\mathcal{X}_S| \ll |\mathcal{X}|$ in Section 6.

uRR in the low privacy regime. When $\varepsilon = \ln |\mathcal{X}|$ and $|\mathcal{X}_S| \ll |\mathcal{X}|$, the right-hand side of (14) in Proposition 6 can be simplified by using $|\mathcal{X}_S|/|\mathcal{X}| \approx 0$:

$$\mathbb{E}\left[l_1(\hat{\mathbf{p}}, \mathbf{p}^*)\right] \approx \sqrt{\frac{2(|\mathcal{X}| - 1)}{n\pi}}.$$

[2] We note that we use the empirical estimation method in the same way as [29], and that it might be possible that other mechanisms have better utility with a different estimation method. However, we emphasize that even with the empirical estimation method, the uRAP achieves the lower bounds on the l_1 and l_2 losses of any ULDP mechanisms when $\varepsilon \approx 0$, and the uRR and uRAP achieve almost the same utility as a non-private mechanism when $\varepsilon = \ln |\mathcal{X}|$ and most of the data are non-sensitive.

It should be noted that the expected l_1 loss of the non-private mechanism, which does not obfuscate the personal data at all, is at most $\sqrt{\frac{2(|\mathcal{X}|-1)}{n\pi}}$ [29]. Thus, when $\varepsilon = \ln|\mathcal{X}|$ and $|\mathcal{X}_S| \ll |\mathcal{X}|$, the $(\mathcal{X}_S,\varepsilon)$-uRR achieves almost the same data utility as the non-private mechanism, whereas the expected l_1 loss of the ε-RR is twice larger than that of the non-private mechanism [29].

uRAP in the general case. We then analyze the uRAP:

Proposition 7 (l_1 loss of the uRAP). *Let $\varepsilon \in \mathbb{R}_{\geq 0}$, $u' = e^{\varepsilon/2} - 1$, and $v_N = \frac{e^{\varepsilon/2}}{e^{\varepsilon/2}-1}$. The expected l_1-loss of the $(\mathcal{X}_S,\varepsilon)$-uRAP mechanism is:*

$$\mathbb{E}\left[l_1(\hat{\mathbf{p}},\mathbf{p})\right] \approx \sqrt{\frac{2}{n\pi}}\left(\sum_{j=1}^{|\mathcal{X}_S|}\sqrt{\left(\mathbf{p}(x_j)+1/u'\right)\left(v_N-\mathbf{p}(x_j)\right)} + \sum_{j=|\mathcal{X}_S|+1}^{|\mathcal{X}|}\sqrt{\mathbf{p}(x_j)\left(v_N-\mathbf{p}(x_j)\right)}\right), \quad (16)$$

where $f(n) \approx g(n)$ represents $\lim_{n\to\infty} f(n)/g(n) = 1$.

When $0 < \varepsilon < 2\ln(\frac{|\mathcal{X}_N|}{2}+1)$, the l_1 loss is maximized by the uniform distribution $\mathbf{p}_{U_N}$ over $\mathcal{X}_N$:

Proposition 8. *For any $0 < \varepsilon < 2\ln(\frac{|\mathcal{X}_N|}{2}+1)$ and $|\mathcal{X}_S| \leq |\mathcal{X}_N|$, (16) is maximized when $\mathbf{p} = \mathbf{p}_{U_N}$:*

$$\mathbb{E}\left[l_1(\hat{\mathbf{p}},\mathbf{p})\right] \lesssim \mathbb{E}\left[l_1(\hat{\mathbf{p}},\mathbf{p}_{U_N})\right] = \sqrt{\frac{2}{n\pi}}\left(\frac{e^{\varepsilon/4}|\mathcal{X}_S|}{e^{\varepsilon/2}-1} + \sqrt{\frac{e^{\varepsilon/2}|\mathcal{X}_N|}{e^{\varepsilon/2}-1} - 1}\right), \quad (17)$$

where $f(n) \lesssim g(n)$ represents $\lim_{n\to\infty} f(n)/g(n) \leq 1$.

Note that this proposition covers a wide range of ε. For example, when $|\mathcal{X}_S| \leq |\mathcal{X}_N|$, it covers both the high privacy regime ($\varepsilon \approx 0$) and low privacy regime ($\varepsilon = \ln|\mathcal{X}|$), since $\ln|\mathcal{X}| < 2\ln(\frac{|\mathcal{X}_N|}{2}+1)$. Below we instantiate the l_1 loss in the high and low privacy regimes based on this proposition.

uRAP in the high privacy regime. If ε is close to 0, we have $e^{\varepsilon/2} - 1 \approx \varepsilon/2$. Thus, the right-hand side of (17) in Proposition 8 can be simplified as follows:

$$\mathbb{E}\left[l_1(\hat{\mathbf{p}},\mathbf{p}_{U_N})\right] \approx \sqrt{\frac{2}{n\pi}} \cdot \frac{2|\mathcal{X}_S|}{\varepsilon}. \quad (18)$$

It is shown in [29] that the expected l_1 loss of the ε-RAPPOR is at most $\sqrt{\frac{2}{n\pi}} \cdot \frac{2|\mathcal{X}|}{\varepsilon}$ when $\varepsilon \approx 0$. Thus, by (18), the expected l_1 loss of the $(\mathcal{X}_S,\varepsilon)$-uRAP is much smaller than that of the ε-RAPPOR when $|\mathcal{X}_S| \ll |\mathcal{X}|$.

Moreover, by (18), the expected l_1 loss of the $(\mathcal{X}_S,\varepsilon)$-uRAP in the worst case is expressed as $\Theta(\frac{|\mathcal{X}_S|}{\sqrt{n\varepsilon^2}})$ in the high privacy regime. As described in Section 3.2, this is "order" optimal among all $(\mathcal{X}_S,\mathcal{Y}_P,\varepsilon)$-ULDP mechanisms (in Appendix C.1, we also show that the expected l_2 of the $(\mathcal{X}_S,\varepsilon)$-uRAP is expressed as $\Theta(\frac{|\mathcal{X}_S|}{n\varepsilon^2})$).

uRAP in the low privacy regime. If $\varepsilon = \ln|\mathcal{X}|$ and $|\mathcal{X}_S| \ll |\mathcal{X}|^{\frac{3}{4}}$, the right-hand side of (17) can be simplified, using $|\mathcal{X}_S|/|\mathcal{X}|^{\frac{3}{4}} \approx 0$, as follows:

$$\mathbb{E}\left[l_1(\hat{\mathbf{p}},\mathbf{p}_{U_N})\right] \approx \sqrt{\frac{2(|\mathcal{X}|-1)}{n\pi}}.$$

Thus, when $\varepsilon = \ln|\mathcal{X}|$ and $|\mathcal{X}_S| \ll |\mathcal{X}|^{\frac{3}{4}}$, the $(\mathcal{X}_S,\varepsilon)$-uRAP also achieves almost the same data utility as the non-private mechanism, whereas the expected l_1 loss of the ε-RAPPOR is $\sqrt{|\mathcal{X}|}$ times larger than that of the non-private mechanism [29].

Summary. In summary, the uRR and uRAP provide much higher utility than the RR and RAPPOR when $|\mathcal{X}_S| \ll |\mathcal{X}|$. Moreover, when $\varepsilon = \ln|\mathcal{X}|$ and $|\mathcal{X}_S| \ll |\mathcal{X}|$ (resp. $|\mathcal{X}_S| \ll |\mathcal{X}|^{\frac{3}{4}}$), the uRR (resp. uRAP) achieves almost the same utility as a non-private mechanism.

5 Personalized ULDP Mechanisms

We now consider the personalized-mechanism scenario (outlined in Section 2.1), and propose a *PUM (Personalized ULDP Mechanism)* to keep secret what is sensitive for each user while enabling the data collector to estimate a distribution.

Sections 5.1 describes the PUM. Section 5.2 explains its privacy properties. Section 5.3 proposes a method to estimate the distribution $\mathbf{p}$ from $\mathbf{Y}$ obfuscated using the PUM. Section 5.4 analyzes the data utility of the PUM.

5.1 PUM with κ Semantic Tags

Figure 4 shows the overview of the PUM $\mathbf{Q}^{(i)}$ for the i-th user ($i = 1,2,\ldots,n$). It first deterministically maps personal data $x \in \mathcal{X}$ to *intermediate data* using a *pre-processor* $f_{pre}^{(i)}$, and then maps the intermediate data to obfuscated data $y \in \mathcal{Y}$ using a utility-optimized mechanism $\mathbf{Q}_{cmn}$ common to all users. The pre-processor $f_{pre}^{(i)}$ maps user-specific sensitive data $x \in \mathcal{X}_S^{(i)}$ to one of κ bots: $\perp_1, \perp_2, \cdots$, or $\perp_\kappa$. The κ bots represent user-specific sensitive data, and each of them is associated with a *semantic tag* such as "home" or "workplace". The κ semantic tags are the same for all users, and are useful when the data collector has some background knowledge about $\mathbf{p}$ conditioned on each tag. For example, a distribution of POIs tagged as "home" or "workplace" can be easily obtained via the Fousquare venue API [54]. Although this is not a user distribution but a "POI distribution", it can be used to roughly approximate the distribution of users tagged as "home" or "workplace", as shown in Section 6. We define a set $\mathcal{Z}$ of intermediate data by $\mathcal{Z} = \mathcal{X} \cup \{\perp_1, \cdots, \perp_\kappa\}$, and a set $\mathcal{Z}_S$ of sensitive intermediate data by $\mathcal{Z}_S = \mathcal{X}_S \cup \{\perp_1, \cdots, \perp_\kappa\}$.

Formally, the PUM $\mathbf{Q}^{(i)}$ first maps personal data $x \in \mathcal{X}$ to intermediate data $z \in \mathcal{Z}$ using a pre-processor $f_{pre}^{(i)}: \mathcal{X} \to \mathcal{Z}$ specific to each user. The pre-processor $f_{pre}^{(i)}$ maps sensitive

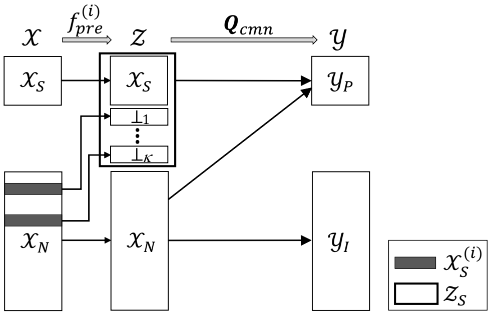

Figure 4: Overview of the PUM $\mathbf{Q}^{(i)}$ $(= \mathbf{Q}_{cmn} \circ f_{pre}^{(i)})$.

data $x \in \mathcal{X}_S^{(i)}$ associated with the k-th tag ($1 \leq k \leq \kappa$) to the corresponding bot $\perp_k$, and maps other data to themselves. Let $\mathcal{X}_{S,k}^{(i)}$ be a set of the i-th user's sensitive data associated with the k-th tag (e.g., set of regions including her primary home and second home). Then, $\mathcal{X}_S^{(i)}$ is expressed as $\mathcal{X}_S^{(i)} = \bigcup_{1 \leq k \leq \kappa} \mathcal{X}_{S,k}^{(i)}$, and $f_{pre}^{(i)}$ is given by:

$$f_{pre}^{(i)}(x) = \begin{cases} \perp_k & (\text{if } x \in \mathcal{X}_{S,k}^{(i)}) \\ x & (\text{otherwise}). \end{cases} \tag{19}$$

After mapping personal data $x \in \mathcal{X}$ to intermediate data $z \in \mathcal{Z}$, the $(\mathcal{Z}_S, \mathcal{Y}_P, \varepsilon)$-utility-optimized mechanism $\mathbf{Q}_{cmn}$ maps z to obfuscated data $y \in \mathcal{Y}$. Examples of $\mathbf{Q}_{cmn}$ include the $(\mathcal{Z}_S, \varepsilon)$-uRR (in Definition 3) and $(\mathcal{Z}_S, \varepsilon)$-uRAP (in Definition 4). As a whole, the PUM $\mathbf{Q}^{(i)}$ can be expressed as: $\mathbf{Q}^{(i)} = \mathbf{Q}_{cmn} \circ f_{pre}^{(i)}$. The i-th user stores $f_{pre}^{(i)}$ and $\mathbf{Q}_{cmn}$ in a device that obfuscates her personal data (e.g., mobile phone, personal computer). Note that if $f_{pre}^{(i)}$ is leaked, $x \in \mathcal{X}_N$ corresponding to each bot (e.g., home, workplace) is leaked. Thus, the user keeps $f_{pre}^{(i)}$ secret. To strongly prevent the leakage of $f_{pre}^{(i)}$, the user may deal with $f_{pre}^{(i)}$ using a tamper-resistant hardware/software. On the other hand, the utility-optimized mechanism $\mathbf{Q}_{cmn}$, which is common to all users, is available to the data collector.

The feature of the proposed PUM $\mathbf{Q}^{(i)}$ is two-fold: (i) the secrecy of the pre-processor $f_{pre}^{(i)}$ and (ii) the κ semantic tags. By the first feature, the i-th user can keep $\mathcal{X}_S^{(i)}$ (i.e., what is sensitive for her) secret, as shown in Section 5.2. The second feature enables the data collector to estimate a distribution $\mathbf{p}$ with high accuracy. Specifically, she estimates $\mathbf{p}$ from obfuscated data $\mathbf{Y}$ using $\mathbf{Q}_{cmn}$ and some background knowledge about $\mathbf{p}$ conditioned on each tag, as shown in Section 5.3.

In practice, it may happen that a user has her specific sensitive data $x \in \mathcal{X}_S^{(i)}$ that is not associated with any semantic tags. For example, if we prepare only tags named "home" and "workplace", then sightseeing places, restaurants, and any other places are not associated with these tags. One way to deal with such data is to create another bot associated with a tag named "others" (e.g., if $\perp_1$ and $\perp_2$ are associated with "home" and "workplace", respectively, we create $\perp_3$ associated with "others"), and map x to this bot. It would be difficult for the data collector to obtain background knowledge about $\mathbf{p}$ conditioned on such a tag. In Section 5.3, we will explain how to estimate $\mathbf{p}$ in this case.

5.2 Privacy Properties

We analyze the privacy properties of the PUM $\mathbf{Q}^{(i)}$. First, we show that it provides ULDP.

Proposition 9. *The PUM $\mathbf{Q}^{(i)}$ $(= \mathbf{Q}_{cmn} \circ f_{pre}^{(i)})$ provides $(\mathcal{X}_S \cup \mathcal{X}_S^{(i)}, \mathcal{Y}_P, \varepsilon)$-ULDP.*

We also show that our PUM provides DP in that an adversary who has observed $y \in \mathcal{Y}_P$ cannot determine, for any $i, j \in [n]$, whether it is obfuscated using $\mathbf{Q}^{(i)}$ or $\mathbf{Q}^{(j)}$, which means that $y \in \mathcal{Y}_P$ reveals almost no information about $\mathcal{X}_S^{(i)}$:

Proposition 10. *For any $i, j \in [n]$, any $x \in \mathcal{X}$, and any $y \in \mathcal{Y}_P$,*

$$\mathbf{Q}^{(i)}(y|x) \leq e^{\varepsilon} \mathbf{Q}^{(j)}(y|x).$$

We then analyze the secrecy of $\mathcal{X}_S^{(i)}$. The data collector, who knows the common-mechanism $\mathbf{Q}_{cmn}$, cannot obtain any information about $\mathcal{X}_S^{(i)}$ from $\mathbf{Q}_{cmn}$ and $y \in \mathcal{Y}_P$. Specifically, the data collector knows, for each $z \in \mathcal{Z}$, whether $z \in \mathcal{Z}_S$ or not by viewing $\mathbf{Q}_{cmn}$. However, she cannot obtain any information about $\mathcal{X}_S^{(i)}$ from $\mathcal{Z}_S$, because she does not know the mapping between $\mathcal{X}_S^{(i)}$ and $\{\perp_1, \cdots, \perp_\kappa\}$ (i.e., $f_{pre}^{(i)}$). In addition, Propositions 9 and 10 guarantee that $y \in \mathcal{Y}_P$ reveals almost no information about both input data and $\mathcal{X}_S^{(i)}$.

For example, assume that the i-th user obfuscates her home $x \in \mathcal{X}_S \cup \mathcal{X}_S^{(i)}$ using the PUM $\mathbf{Q}^{(i)}$, and sends $y \in \mathcal{Y}_P$ to the data collector. The data collector cannot infer either $x \in \mathcal{X}_S \cup \mathcal{X}_S^{(i)}$ or $z \in \mathcal{Z}_S$ from $y \in \mathcal{Y}_P$, since both $\mathbf{Q}_{cmn}$ and $\mathbf{Q}^{(i)}$ provide ULDP. This means that the data collector cannot infer *the fact that she was at home* from y. Furthermore, the data collector cannot infer *where her home is*, since $\mathcal{X}_S^{(i)}$ cannot be inferred from $\mathbf{Q}_{cmn}$ and $y \in \mathcal{Y}_P$ as explained above.

We need to take a little care when the i-th user obfuscates non-sensitive data $x \in \mathcal{X}_N \setminus \mathcal{X}_S^{(i)}$ using $\mathbf{Q}^{(i)}$ and sends $y \in \mathcal{Y}_I$ to the data collector. In this case, the data collector learns x from y, and therefore learns that x is not sensitive (i.e., $x \notin \mathcal{X}_S^{(i)}$). Thus, the data collector, who knows that the user wants to hide her home, would reduce the number of possible candidates for her home from $\mathcal{X}$ to $\mathcal{X} \setminus \{x\}$. However, if $|\mathcal{X}|$ is large (e.g., $|\mathcal{X}| = 625$ in our experiments using location data), the number $|\mathcal{X}| - 1$ of candidates is still large. Since the data collector cannot further reduce the number of candidates using $\mathbf{Q}_{cmn}$, her home is still kept strongly secret. In Section 7, we also explain that the secrecy of $\mathcal{X}_S^{(i)}$ is achieved under reasonable assumptions even when she sends multiple data.

5.3 Distribution Estimation

We now explain how to estimate a distribution $\mathbf{p}$ from data $\mathbf{Y}$ obfuscated using the PUM. Let $\mathbf{r}^{(i)}$ be a distribution of intermediate data for the i-th user:

$$\mathbf{r}^{(i)}(z) = \begin{cases} \sum_{x\in\mathcal{X}_{S,k}^{(i)}} \mathbf{p}(x) & (\text{if } z = \perp_k \text{ for some } k = 1,\dots,\kappa) \\ 0 & (\text{if } z \in \mathcal{X}_S^{(i)}) \\ \mathbf{p}(z) & (\text{otherwise}). \end{cases}$$

and $\mathbf{r}$ be the average of $\mathbf{r}^{(i)}$ over n users; i.e., $\mathbf{r}(z) = \frac{1}{n}\sum_{i=1}^{n}\mathbf{r}^{(i)}(z)$ for any $z \in \mathcal{Z}$. Note that $\sum_{x\in\mathcal{X}}\mathbf{p}(x) = 1$ and $\sum_{z\in\mathcal{Z}}\mathbf{r}(z) = 1$. Furthermore, let π_k be a distribution of personal data $x \in \mathcal{X}$ conditioned on $\perp_k$ defined by:

$$\pi_k(x) = \frac{\sum_{i=1}^{n}\mathbf{p}_k^{(i)}(x)}{\sum_{x'\in\mathcal{X}}\sum_{i=1}^{n}\mathbf{p}_k^{(i)}(x')}, \tag{20}$$

$$\mathbf{p}_k^{(i)}(x) = \begin{cases} \mathbf{p}(x) & (\text{if } f_{pre}^{(i)}(x) = \perp_k) \\ 0 & (\text{otherwise}). \end{cases}$$

$\pi_k(x)$ in (20) is a normalized sum of the probability $\mathbf{p}(x)$ of personal data x whose corresponding intermediate data is $\perp_k$. Note that although $x \in \mathcal{X}$ is deterministically mapped to $z \in \mathcal{Z}$ for each user, we can consider the probability distribution π_k for n users. For example, if $\perp_k$ is tagged as "home", then π_k is a distribution of users at home.

We propose a method to estimate a distribution $\mathbf{p}$ from obfuscated data $\mathbf{Y}$ using some background knowledge about π_k as an estimate $\hat{\pi}_k$ of π_k (we explain the case where we have no background knowledge later). Our estimation method first estimates a distribution $\mathbf{r}$ of intermediate data from obfuscated data $\mathbf{Y}$ using $\mathbf{Q}_{cmn}$. This can be performed in the same way as the common-mechanism scenario. Let $\hat{\mathbf{r}}$ be the estimate of $\mathbf{r}$.

After computing $\hat{\mathbf{r}}$, our method estimates $\mathbf{p}$ using the estimate $\hat{\pi}_k$ (i.e., background knowledge about π_k) as follows:

$$\hat{\mathbf{p}}(x) = \hat{\mathbf{r}}(x) + \sum_{k=1}^{\kappa}\hat{\mathbf{r}}(\perp_k)\hat{\pi}_k(x), \ \ \forall x \in \mathcal{X}. \tag{21}$$

Note that $\hat{\mathbf{p}}$ in (21) can be regarded as an empirical estimate of $\mathbf{p}$. Moreover, if both $\hat{\mathbf{r}}$ and $\hat{\pi}_k$ are in the probability simplex $\mathcal{C}$, then $\hat{\mathbf{p}}$ in (21) is always in $\mathcal{C}$.

If we do not have estimates $\hat{\pi}_k$ for some bots (like the one tagged as "others" in Section 5.1), then we set $\hat{\pi}_k(x)$ in proportion to $\hat{\mathbf{r}}(x)$ over $x \in \mathcal{X}_N$ (i.e., $\hat{\pi}_k(x) = \frac{\hat{\mathbf{r}}(x)}{\sum_{x'\in\mathcal{X}_N}\hat{\mathbf{r}}(x')}$) for such bots. When we do not have any background knowledge $\hat{\pi}_1, \cdots, \hat{\pi}_\kappa$ for all bots, it amounts to simply discarding the estimates $\hat{\mathbf{r}}(\perp_1), \cdots, \hat{\mathbf{r}}(\perp_\kappa)$ for κ bots and normalizing $\hat{\mathbf{r}}(x)$ over $x \in \mathcal{X}_N$ so that the sum is one.

5.4 Utility Analysis

We now theoretically analyze the data utility of our PUM. Recall that $\hat{\mathbf{p}}$, $\hat{\mathbf{r}}$, and $\hat{\pi}_k$ are the estimate of the distribution of personal data, intermediate data, and personal data conditioned on $\perp_k$, respectively. In the following, we show that the l_1 loss of $\hat{\mathbf{p}}$ can be upper-bounded as follows:

Theorem 1 (l_1 loss of the PUM)**.**

$$l_1(\hat{\mathbf{p}}, \mathbf{p}) \leq l_1(\hat{\mathbf{r}}, \mathbf{r}) + \sum_{k=1}^{\kappa}\hat{\mathbf{r}}(\perp_k)l_1(\hat{\pi}_k, \pi_k). \tag{22}$$

This means the upper-bound on the l_1 loss of $\hat{\mathbf{p}}$ can be decomposed into the l_1 loss of $\hat{\mathbf{r}}$ and of $\hat{\pi}_k$ weighted by $\hat{\mathbf{r}}(\perp_k)$.

The first term in (22) is the l_1 loss of $\hat{\mathbf{r}}$, which depends on $\mathbf{Q}_{cmn}$. For example, if we use the uRR or uRAP as $\mathbf{Q}_{cmn}$, the expectation of $l_1(\hat{\mathbf{r}}, \mathbf{r})$ is given by Propositions 4 and 7, respectively. In Section 6, we show they are very small.

The second term in (22) is the summation of the l_1 loss of $\hat{\pi}_k$ weighted by $\hat{\mathbf{r}}(\perp_k)$. If we accurately estimate π_k, the second term is very small. In other words, if we have enough background knowledge about π_k, we can accurately estimate $\mathbf{p}$ in the personalized-mechanism scenario.

It should be noted that when the probability $\hat{\mathbf{r}}(\perp_k)$ is small, the second term in (22) is small *even if we have no background knowledge about* π_k. For example, when only a small number of users map $x \in \mathcal{X}_S^{(i)}$ to a tag named "others", they hardly affect the accuracy of $\hat{\mathbf{p}}$. Moreover, the second term in (22) is upper-bounded by $2\sum_{k=1}^{\kappa}\hat{\mathbf{r}}(\perp_k)$, since the l_1 loss is at most 2. Thus, after computing $\hat{\mathbf{r}}$, the data collector can easily compute the worst-case value of the second term in (22) to know the effect of the estimation error of $\hat{\pi}_k$ on the accuracy of $\hat{\mathbf{p}}$.

Last but not least, the second term in (22) does not depend on ε (while the first term depends on ε). Thus, the effect of the second term is relatively small when ε is small (i.e., high privacy regime), as shown in Section 6.

Remark. Note that different privacy preferences might skew the distribution π_k. For example, doctors might not consider hospitals as sensitive as compared to patients. Consequently, the distribution π_k conditioned on "hospital" might be a distribution of patients (not doctors) in hospitals. This kind of systematic bias can increase the estimation error of $\hat{\pi}_k$. Theorem 1 and the above discussions are also valid in this case.

6 Experimental Evaluation

6.1 Experimental Set-up

We conducted experiments using two large-scale datasets:

Foursquare dataset. The Foursquare dataset (global-scale check-in dataset) [54] is one of the largest location datasets among publicly available datasets (e.g., see [10], [44], [55], [57]); it contains 33278683 check-ins all over the world, each of which is associated with a POI ID and venue category (e.g., restaurant, shop, hotel, hospital, home, workplace).

We used 359054 check-ins in Manhattan, assuming that each check-in is from a different user. Then we divided Manhattan into 25 × 25 regions at regular intervals and used them as input alphabets; i.e., $|\mathcal{X}| = 625$. The size of each region is about 400m (horizontal) × 450m (vertical). We assumed a region that includes a hospital visited by at least ten users as a sensitive region common to all users. The number of such regions was $|\mathcal{X}_S| = 15$. In addition, we assumed a region in $\mathcal{X}_N$ that includes a user's home or workplace as her user-specific sensitive region. The number of users at home and workplace was 5040 and 19532, respectively.

US Census dataset. The US Census (1990) dataset [35] was collected as part of the 1990 U.S. census. It contains responses from 2458285 people (each person provides one response), each of which contains 68 attributes.

We used the responses from all people, and used age, income, marital status, and sex as attributes. Each attribute has 8, 5, 5, and 2 categories, respectively. (See [35] for details about the value of each category ID.) We regarded a tuple of the category IDs as a total category ID, and used it as an input alphabet; i.e., $|\mathcal{X}| = 400$ $(= 8 \times 5 \times 5 \times 2)$. We considered the fact that "divorce" and "unemployment" might be sensitive for many users [34], and regarded such categories as sensitive for all users (to be on the safe side, as described in Section 2.1). Note that people might be students until their twenties and might retire in their fifties or sixties. Children of age twelve and under cannot get married. We excluded such categories from sensitive ones. The number of sensitive categories was $|\mathcal{X}_S| = 76$.

We used a frequency distribution of all people as a true distribution $\mathbf{p}$, and randomly chose a half of all people as users who provide their obfuscated data; i.e., $n = 179527$ and 1229143 in the Foursquare and US Census datasets, respectively. Here we did not use all people, because we would like to evaluate the non-private mechanism that does not obfuscate the personal data; i.e., the non-private mechanism has an estimation error in our experiments due to the random sampling from the population.

As utility, we evaluated the TV (Total Variation) by computing the sample mean over a hundred realizations of $\mathbf{Y}$.

6.2 Experimental Results

Common-mechanism scenario. We first focused on the common-mechanism scenario, and evaluated the RR, RAPPOR, uRR, and uRAP. As distribution estimation methods, we used empirical estimation, empirical estimation with the significance threshold, and EM reconstruction (denoted by "emp", "emp+thr", and "EM", respectively). In "emp+thr", we set the significance level α to be $\alpha = 0.05$, and uniformly assigned the remaining probability to each of the estimates below the significance threshold in the same way as [51].

Figure 5 shows the results in the case where ε is changed from 0.1 to 10. "no privacy" represents the non-private mechanism. It can be seen that our mechanisms outperform the existing mechanisms by one or two orders of magnitude. Our mechanisms are effective especially in the Foursquare dataset, since the proportion of sensitive regions is very small $(15/625 = 0.024)$. Moreover, the uRR provides almost the same performance as the non-private mechanism when $\varepsilon = \ln|\mathcal{X}|$, as described in Section 4.3. It can also be seen that "emp+thr" and "EM" significantly outperform "emp", since the estimates in "emp+thr" and "EM" are always non-negative. Although "EM" outperforms "emp+thr" for the RAPPOR and uRAP when ε was large, the two estimation methods provide very close performance as a whole.

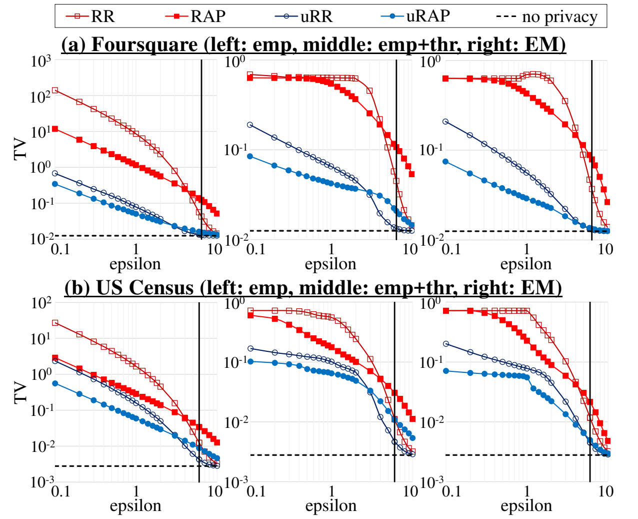

Figure 5: ε vs. TV (common-mechanism). A bold line parallel to the y-axis represents $\varepsilon = \ln|\mathcal{X}|$.

We then evaluated the relationship between the number of sensitive regions/categories and the TV. To this end, we randomly chose $\mathcal{X}_S$ from $\mathcal{X}$, and increased $|\mathcal{X}_S|$ from 1 to $|\mathcal{X}|$ (only in this experiment). We attempted one hundred cases for randomly choosing $\mathcal{X}_S$ from $\mathcal{X}$, and evaluated the TV by computing the sample mean over one hundred cases.

Figure 6 shows the results for $\varepsilon = 0.1$ (high privacy regime) or $\ln|\mathcal{X}|$ (low privacy regime). Here we omit the performance of "emp+thr", since it is very close to that of "EM" in the same way as in Figure 5. The uRAP and uRR provide the best performance when $\varepsilon = 0.1$ and $\ln|\mathcal{X}|$, respectively. In addition, the uRR provides the performance close to the non-private mechanism when $\varepsilon = \ln|\mathcal{X}|$ and the number $|\mathcal{X}_S|$ of sensitive regions/categories is less than 100. The performance of the uRAP is also close to that of the non-private mechanism when $|\mathcal{X}_S|$ is less than 20 (note that $|\mathcal{X}|^{\frac{3}{4}} = 125$ and 89 in the Foursquare and US Census datasets, respectively). However, it rapidly increases with increase in $|\mathcal{X}_S|$. Overall, our theoretical results in Section 4.3 hold for the two real datasets.

We also evaluated the performance when the number of attributes was increased from 4 to 9 in the US Census dataset. We added, one by one, five attributes as to whether or not a user has served in the military during five periods ("Sept80", "May75880", "Vietnam", "Feb55", and "Korean" in [18]; we added them in this order). We assumed that these attributes

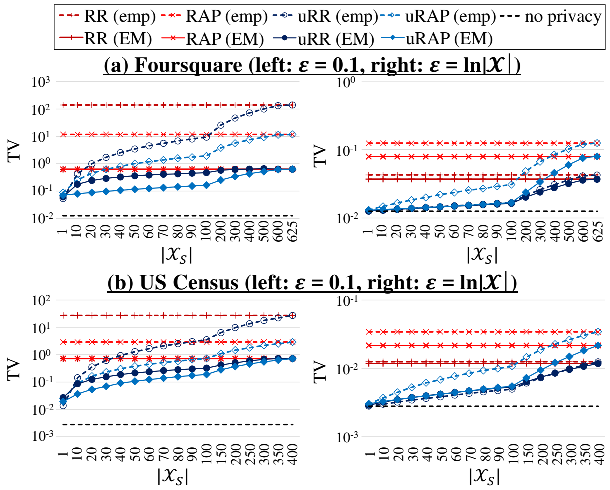

Figure 6: $|\mathcal{X}_S|$ vs. TV when $\varepsilon = 0.1$ or $\ln |\mathcal{X}|$.

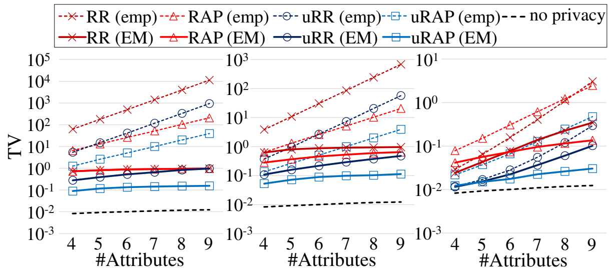

Figure 7: Number of attributes vs. TV (US Census dataset; left: $\varepsilon = 0.1$, middle: $\varepsilon = 1.0$, right: $\varepsilon = 6.0$).

are non-sensitive. Since each of the five attributes had two categories (1: yes, 0: no), $|\mathcal{X}|$ (resp. $|\mathcal{X}_S|$) was changed from 400 to 12800 (resp. from 76 to 2432). We randomly chose $n = 240000$ people as users who provide obfuscated data, and evaluated the TV by computing the sample mean over ten realizations of $\mathbf{Y}$ (only in this experiment).

Figure 7 shows the results in the case where $\varepsilon = 0.1$, 1.0, or 6.0 (=ln 400). Here we omit the performance of "emp+thr" in the same way as Figure 6. Although the TV increases with an increase in the number of attributes, overall our utility-optimized mechanisms remain effective, compared to the existing mechanisms. One exception is the case where $\varepsilon = 0.1$ and the number of attributes is 9; the TV of the RR (EM), RAPPOR (EM), and uRR (EM) is almost 1. Note that when we use the EM reconstruction method, the worst value of the TV is 1. Thus, as with the RR and RAPPOR, the uRR fails to estimate a distribution in this case. On the other hand, the TV of the uRAP (EM) is much smaller than 1 even in this case, which is consistent with the fact that the uRAP is order optimal in the high privacy regime. Overall, the uRAP is robust to the increase of the attributes at the same value of ε (note that for large $|\mathcal{X}|$, $\varepsilon = 1.0$ or 6.0 is a medium privacy regime where $0 \ll \varepsilon \ll \ln |\mathcal{X}|$).

We also measured the running time (i.e., time to estimate $\mathbf{p}$ from $\mathbf{Y}$) of "EM" (which sets the estimate by "emp+thr" as an initial value of $\hat{\mathbf{p}}$) on an Intel Xeon CPU E5-2620 v3 (2.40 GHz, 6 cores, 12 logical processors) with 32 GB RAM. We found that the running time increases roughly linearly with the number of attributes. For example, when $\varepsilon = 6.0$ and the number of attributes is 9, the running time of "EM" required 3121, 1258, 5225, and 1073 seconds for "RR", "uRR", "RAP", and "uRAP", respectively. We also measured the running time of 'emp" and "emp+thr", and found that they required less than one second even when the number of attributes is 9. Thus, if "EM" requires too much time for a large number of attributes, "emp+thr" would be a good alternative to "EM".

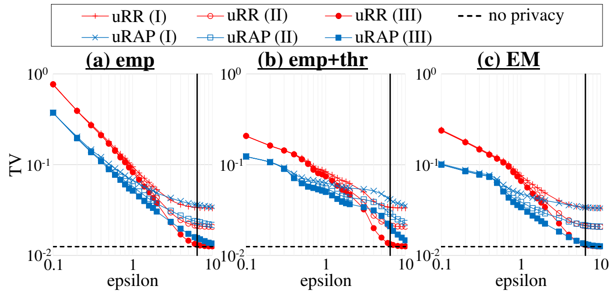

Figure 8: ε vs. TV (personalized-mechanism) ((I): w/o knowledge, (II): POI distribution, (III): true distribution).

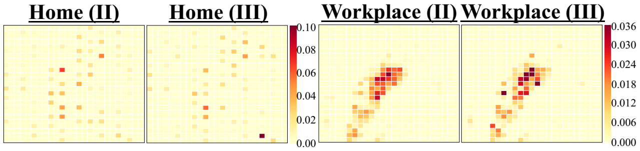

Figure 9: Visualization of the distributions ((II): POI distribution, (III): true distribution).

Personalized-mechanism scenario. We then focused on the personalized-mechanism scenario, and evaluated our utility-optimized mechanisms using the Foursquare dataset. We used the PUM with $\kappa = 2$ semantic tags (described in Section 5.1), which maps "home" and 'workplace" to bots $\perp_1$ and $\perp_2$, respectively. As the background knowledge about the bot distribution π_k $(1 \leq k \leq 2)$, we considered three cases: (I) we do not have any background knowledge; (II) we use a distribution of POIs tagged as "home" (resp. "workplace"), which is computed from the POI data in [54], as an estimate of the bot probability $\hat{\pi}_1$ (resp. $\hat{\pi}_2$); (III) we use the true distributions (i.e., $\hat{\pi}_k = \pi_k$). Regarding (II), we emphasize again that it is not a user distribution but a "POI distribution", and can be easily obtained via the Foursquare venue API [54].

Figure 8 shows the results. We also show the POI and true distributions in Figure 9. It can be seen that the performance of (II) lies in between that of (I) and (III), which shows that the estimate $\hat{\pi}_k$ of the bot distribution affects utility. However, when ε is smaller than 1, all of (I), (II), and (III) provide almost the same performance, since the effect of the estimation error of $\hat{\pi}_k$ does not depend on ε, as described in Section 5.4.

We also computed the l_1 loss $l_1(\hat{\mathbf{p}}, \mathbf{p})$ and the first and second terms in the right-hand side of (22) to investigate

Table 1: l_1 loss $l_1(\hat{\mathbf{p}}, \mathbf{p})$ and the first and second terms in the right-hand side of (22) in the case where $\varepsilon = \ln|\mathcal{X}|$ and the EM reconstruction method is used.

Method	$l_1(\hat{\mathbf{p}}, \mathbf{p})$	first term	second term
uRR (**I**)	6.73×10^{-2}	2.70×10^{-2}	7.34×10^{-2}
uRR (**II**)	4.24×10^{-2}	2.70×10^{-2}	2.96×10^{-2}
uRR (**III**)	2.62×10^{-2}	2.70×10^{-2}	0
uRAP (**I**)	6.77×10^{-2}	2.76×10^{-2}	7.35×10^{-2}
uRAP (**II**)	4.28×10^{-2}	2.76×10^{-2}	2.96×10^{-2}
uRAP (**III**)	2.67×10^{-2}	2.76×10^{-2}	0

whether Theorem 1 holds. Table 1 shows the results (we averaged the values over one hundred realizations of $\mathbf{Y}$). It can be seen that $l_1(\hat{\mathbf{p}}, \mathbf{p})$ is smaller than the summation of the first and second terms in all of the methods, which shows that Theorem 1 holds in our experiments.

From these experimental results, we conclude that our proposed methods are very effective in both the common-mechanism and personalized-mechanism scenarios. In Appendix C.2, we show the MSE has similar results to the TV.

7 Discussions

On the case of multiple data per user. We have so far assumed that each user sends only a single datum. Now we discuss the case where each user sends multiple data based on the compositionality of ULDP described in Section 3.2. Specifically, when a user sends t (> 1) data, we obtain $(\mathcal{X}_S, (\mathcal{Y}_P)^t, \varepsilon)$-ULDP in total by obfuscating each data using the $(\mathcal{X}_S, \mathcal{Y}_P, \varepsilon/t)$-utility-optimized mechanism. Note, however, that the amount of noise added to each data increases with increase in t. Consequently, for $\varepsilon \in [0, t]$, the lower bound on the l_1 (resp. l_2) loss (described in Section 3.2) can be expressed as $\Theta(\frac{\sqrt{t}|\mathcal{X}_S|}{\sqrt{n\varepsilon^2}})$ (resp. $\Theta(\frac{t|\mathcal{X}_S|}{n\varepsilon^2})$), which increases with increase in t. Thus, t cannot be large for distribution estimation in practice. This is also common to all LDP mechanisms.

Next we discuss the secrecy of $\mathcal{X}_S^{(i)}$. Assume that the i-th user obfuscates t data using different seeds, and sends t_P protected data in $\mathcal{Y}_P$ and t_I invertible data in $\mathcal{Y}_I$, where $t = t_P + t_I > 1$ (she can also use the same seed for the same data to reduce t_I as in [23]). If all the t_I data in $\mathcal{Y}_I$ are different from each other, the data collector learns t_I original data in $\mathcal{X}_N$. However, t_I $(\leq t)$ cannot be large in practice, as explained above. In addition, in many applications, a user's personal data is highly non-uniform and sparse. In locations data, for example, a user often visits only a small number of regions in the whole map $\mathcal{X}$. Let $\mathcal{T}^{(i)} \subseteq \mathcal{X}_N$ be a set of possible input values for the i-th user in $\mathcal{X}_N$. Then, even if t_I is large, the data collector cannot learn more than $|\mathcal{T}^{(i)}|$ data in $\mathcal{X}_N$.

Moreover, the t_P data in $\mathcal{Y}_P$ reveal almost no information about $\mathcal{X}_S^{(i)}$, since $\mathbf{Q}^{(i)}$ provides $(\mathcal{X}_S, (\mathcal{Y}_P)^t, \varepsilon)$-ULDP. $\mathbf{Q}_{cmn}$ provides no information about $\mathcal{X}_S^{(i)}$, since $f_{pre}^{(i)}$ is kept secret. Thus, the data collector, who knows that the user wants to hide her home, cannot reduce the number of candidates for her home from $\max\{|\mathcal{X}| - t_I, |\mathcal{X}| - |\mathcal{T}^{(i)}|\}$ using the t_P data and $\mathbf{Q}_{cmn}$. If either t_I or $|\mathcal{T}^{(i)}|$ is much smaller than $|\mathcal{X}|$, her home is kept strongly secret.

Note that $\mathbf{p}$ can be estimated even if $\mathcal{X}_S^{(i)}$ changes over time. $\mathcal{X}_S^{(i)}$ is also kept strongly secret if t_I or $|\mathcal{T}^{(i)}|$ is small.

On the correlation between $\mathcal{X}_S$ and $\mathcal{X}_N$. It should also be noted that there might be a correlation between sensitive data $\mathcal{X}_S$ and non-sensitive data $\mathcal{X}_N$. For example, if a user discloses a non-sensitive region close to a sensitive region including her home, the adversary might infer approximate information about the original location (e.g., the fact that the user lives in Paris). However, we emphasize that if the size of each region is large, the adversary cannot infer the exact location such as the exact home address. Similar approaches can be seen in a state-of-the-art location privacy measure called *geo-indistinguishability* [4, 7, 42, 47]. Andrés *et al.* [4] considered privacy protection within a radius of 200m from the original location, whereas the size of each region in our experiments was about 400m × 450m (as described in Section 6.1). We can protect the exact location by setting the size of each region to be large enough, or setting all regions close to a user's sensitive location to be sensitive.

There might also be a correlation between two attributes (e.g., income and marital status) in the US Census dataset. However, we combined the four category IDs into a total category ID for each user as described in Section 6.1. Thus, there is only "one" category ID for each user. Assuming that each user's data is independent, there is no correlation between data. Therefore, we conclude that the sensitive data are strongly protected in both the Foursquare and US Census datasets in our experiments.

It should be noted, however, that the number of total category IDs increases exponentially with the number of attributes. Thus, when there are many attributes as in Figure 7, the estimation accuracy might be increased by obfuscating each attribute independently (rather than obfuscating a total ID) while considering the correlation among attributes. We also need to consider a correlation among "users" for some types of personal data (e.g., flu status). For rigorously protecting such correlated data, we should incorporate Pufferfish privacy [32, 48] into ULDP, as described in Section 1.

8 Conclusion

In this paper, we introduced the notion of ULDP that guarantees privacy equivalent to LDP for only sensitive data. We proposed ULDP mechanisms in both the common and personalized mechanism scenarios. We evaluated the utility of our mechanisms theoretically and demonstrated the effectiveness of our mechanisms through experiments.

References

[1] D. Agrawal and C. C. Aggarwal. On the design and quantification of privacy preserving data mining algorithms. In *Proc. PODS*, pages 247–255, 2001.

[2] R. Agrawal, R. Srikant, and D. Thomas. Privacy preserving OLAP. In *Proc. SIGMOD*, pages 251–262, 2005.

[3] M. Alaggan, S. Gambs, and A.-M. Kermarrec. Heterogeneous differential privacy. *Journal of Privacy and Confidentiality*, 7(2):127–158, 2017.

[4] M. E. Andrés, N. E. Bordenabe, K. Chatzikokolakis, and C. Palamidessi. Geo-indistinguishability: Differential privacy for location-based systems. In *Proc. CCS*, pages 901–914, 2013.

[5] B. Avent, A. Korolova, D. Zeber, T. Hovden, and B. Livshits. BLENDER: Enabling local search with a hybrid differential privacy model. In *Proc. USENIX Security*, pages 747–764, 2017.

[6] R. Bassily and A. Smith. Local, private, efficient protocols for succinct histograms. In *Proc. STOC*, pages 127–135, 2015.

[7] N. E. Bordenabe, K. Chatzikokolakis, and C. Palamidessi. Optimal geo-indistinguishable mechanisms for location privacy. In *Proc. CCS*, pages 251–262, 2014.

[8] K. Chatzikokolakis, M. E. André, N. E. Bordenabe, and C. Palamidessi. Broadening the scope of differential privacy using metrics. In *Proc. PETS*, pages 82–102, 2013.

[9] X. Chen, A. Guntuboyina, and Y. Zhang. On Bayes risk lower bounds. *J. Mach. Learn. Res.*, 17(219):1–58, 2016.

[10] E. Cho, S. A. Myers, and J. Leskovec. Friendship and mobility: User movement in location-based social networks. In *Proc. KDD*, pages 1082–1090, 2011.

[11] J. E. Cohen. Statistical concepts relevant to AIDS. In *Proc. Symposium on Statistics in Science, Industry, and Public Policy*, pages 43–51, 1989.

[12] G. Cormode, T. Kulkarni, and D. Srivastava. Marginal release under local differential privacy. In *Proc. SIGMOD*, pages 131–146, 2018.

[13] T. M. Cover and J. A. Thomas. *Elements of Information Theory, Second Edition*. Wiley-Interscience, 2006.

[14] P. Cuff and L. Yu. Differential privacy as a mutual information constraint. In *Proc. CCS*, pages 43–54, 2016.

[15] Data Breaches Increase 40 Percent in 2016, Finds New Report from Identity Theft Resource Center and CyberScout. http://www.idtheftcenter.org/2016databreaches.html, 2017.

[16] B. Ding, J. Kulkarni, and S. Yekhanin. Collecting telemetry data privately. In *Proc. NIPS*, pages 3574–3583, 2017.

[17] S. Doudalis, I. Kotsoginannis, S. Haney, A. Machanavajjhala, and S. Mehrotra. One-sided differential privacy. *CoRR*, abs/1712.05888, 2017.

[18] D. Dua and E. K. Taniskidou. UCI machine learning repository. http://archive.ics.uci.edu/ml, 2017.

[19] J. C. Duchi, M. I. Jordan, and M. J. Wainwright. Local privacy and statistical minimax rates. In *Proc. FOCS*, pages 429–438, 2013.

[20] J. C. Duchi, M. I. Jordan, and M. J. Wainwright. Local privacy, data processing inequalities, and minimax rates. *CoRR*, abs/1302.3203, 2013.

[21] C. Dwork, F. Mcsherry, K. Nissim, and A. Smith. Calibrating noise to sensitivity in private data analysis. In *Proc. TCC*, pages 265–284, 2006.

[22] C. Dwork and A. Roth. *The Algorithmic Foundations of Differential Privacy*. Now Publishers, 2014.

[23] U. Erlingsson, V. Pihur, and A. Korolova. RAPPOR: Randomized aggregatable privacy-preserving ordinal response. In *Proc. CCS*, pages 1054–1067, 2014.

[24] G. Fanti, V. Pihur, and U. Erlingsson. Building a RAPPOR with the unknown: Privacy-preserving learning of associations and data dictionaries. *PoPETs*, 2016(3):1–21, 2016.

[25] P. Golle and K. Partridge. On the anonymity of home/work location pairs. In *Proc. Pervasive*, pages 390–397, 2009.

[26] T. Hastie, R. Tibshirani, and J. Friedman. *The Elements of Statistical Learning*. Spinger, 2nd edition, 2009.

[27] Z. Huang and W. Du. OptRR: Optimizing randomized response schemes for privacy-preserving data mining. In *Proc. ICDE*, pages 705–714, 2008.

[28] Z. Jorgensen, T. Yu, and G. Cormode. Conservative or liberal? Personalized differential privacy. In *Proc. ICDE*, pages 1023–1034, 2015.

[29] P. Kairouz, K. Bonawitz, and D. Ramage. Discrete distribution estimation under local privacy. In *Proc. ICML*, pages 2436–2444, 2016.

[30] P. Kairouz, S. Oh, and P. Viswanath. Extremal mechanisms for local differential privacy. *J. Mach. Learn. Res.*, 17(1):492–542, 2016.

[31] Y. Kawamoto and T. Murakami. Differentially private obfuscation mechanisms for hiding probability distributions. *CoRR*, abs/1812.00939, 2018.

[32] D. Kifer and A. Machanavajjhala. Pufferfish: A framework for mathematical privacy definitions. *ACM Trans. Database Syst.*, 39(1):1–36, 2014.

[33] S. Krishnan, J. Wang, M. J. Franklin, K. Goldberg, and T. Kraska. PrivateClean: Data cleaning and differential privacy. In *Proc. SIGMOD*, pages 937–951, 2016.

[34] R. L. Leahy. Feeling ashamed of being unemployed - am I afraid of telling people that I am out of work? https://www.psychologytoday.com/us/blog/anxiety-files/201310/feeling-ashamed-being-unemployed, 2013.

[35] M. Lichman. UCI machine learning repository, 2013.

[36] C. Liu, S. Chakraborty, and P. Mittal. Dependence makes you vulnerable: Differential privacy under dependent tuples. In *Proc. NDSS*, 2016.

[37] N. S. Mangat. An improved ranomized response strategy. *J. Royal Stat. Soc. Series B (Methodological)*, 56(1):93–95, 1994.

[38] I. Mironov. Rényi differential privacy. In *Proc. CSF*, pages 263–275, 2017.

[39] T. Murakami, H. Hino, and J. Sakuma. Toward distribution estimation under local differential privacy with small samples. *PoPETs*, 3:84–104, 2017.

[40] T. Murakami and Y. Kawamoto. Utility-optimized local differential privacy mechanisms for distribution estimation. *CoRR*, abs/1807.11317, 2019.

[41] A. Narayanan and V. Shmatikov. Myths and fallacies of "personally identifiable information". *Commun. ACM*, 53(6):24–26, 2010.

[42] S. Oya, C. Troncoso, and F. Pérez-González. Back to the drawing board: Revisiting the design of optimal location privacy-preserving mechanisms. In *Proc. CCS*, pages 1959–1972, 2017.

[43] A. Pastore and M. Gastpar. Locally differentially-private distribution estimation. In *Proc. ISIT*, pages 2694–2698, 2016.

[44] M. Piorkowski, N. Sarafijanovic-Djukic, and M. Grossglauser. CRAWDAD dataset epfl/mobility (v. 2009-02-24). http://crawdad.org/epfl/mobility/20090224, 2009.

[45] Z. Qin, Y. Yang, T. Yu, I. Khalil, X. Xiao, and K. Ren. Heavy hitter estimation over set-valued data with local differential privacy. In *Proc. CCS*, pages 192–203, 2016.

[46] Y. Sei and A. Ohusuga. Differential private data collection and analysis based on randomized multiple dummies for untrusted mobile crowdsensing. *IEEE Trans. Inf. Forensics Secur.*, 12(4):926–939, 2017.

[47] R. Shokri. Privacy games: Optimal user-centric data obfuscation. *PoPETs*, 2015(2):299–315, 2015.

[48] S. Song, Y. Wang, and K. Chaudhuri. Pufferfish privacy mechanisms for correlated data. In *Proc. SIGMOD*, pages 1291–1306, 2017.

[49] A. G. Thakurta, A. H. Vyrros, U. S. Vaishampayan, G. Kapoor, J. Freudiger, V. R. Sridhar, and D. Davidson. Learning New Words, US Patent 9,594,741, Mar. 14 2017.

[50] N. Wang, X. Xiao, T. D. Hoang, H. Shin, J. Shin, and G. Yu. PrivTrie: Effective frequent term discovery under local differential privacy. In *Proc. ICDE*, 2018.

[51] T. Wang, J. Blocki, N. Li, and S. Jha. Locally differentially private protocols for frequency estimation. In *Proc. USENIX Security*, pages 729–745, 2017.

[52] S. L. Warner. Randomized response: A survey technique for eliminating evasive answer bias. *J. Am. Stat. Assoc.*, 60(309):63–69, 1965.

[53] B. Yang, I. Sato, and H. Nakagawa. Bayesian differential privacy on correlated data. In *Proc. SIGMOD*, pages 747–762, 2015.

[54] D. Yang, D. Zhang, and B. Qu. Participatory cultural mapping based on collective behavior data in location based social network. *ACM Trans. Intell. Syst. Technol.*, 7(3):30:1–30:23, 2016.

[55] D. Yang, D. Zhang, V. W. Zheng, and Z. Yu. Modeling user activity preference by leveraging user spatial temporal characteristics in LBSNs. *IEEE Trans. Syst., Man, Cybern., Syst.*, 45(1):129–142, 2015.

[56] M. Ye and A. Barg. Optimal schemes for discrete distribution estimation under local differential privacy. In *Proc. ISIT*, pages 759–763, 2017.

[57] Y. Zheng, X. Xie, and W.-Y. Ma. GeoLife: A collaborative social networking service among user, location and trajectory. *IEEE Data Eng. Bull.*, 32(2):32–40, 2010.

A Properties of ULDP

In this section, we describe the properties of ULDP (the immunity to post-processing and the compatibility with LDP) in more details.

A.1 Post-processing

We first define a class of post-processing randomized algorithms that preserve data types:

Definition 5 (Preservation of data types)**.** Let $\mathcal{Y}_P$ and $\mathcal{Z}_P$ be sets of protected data, and $\mathcal{Y}_I$ and $\mathcal{Z}_I$ be sets of invertible data. Given a randomized algorithm $\mathbf{Q}_1$ from $\mathcal{Y}_P \cup \mathcal{Y}_I$ to $\mathcal{Z}_P \cup \mathcal{Z}_I$, we say that $\mathbf{Q}_1$ *preserves data types* if it satisfies:

- for any $z \in \mathcal{Z}_P$ and any $y \in \mathcal{Y}_I$, $\mathbf{Q}_1(z|y) = 0$, and
- for any $z \in \mathcal{Z}_I$, there exists a $y \in \mathcal{Y}_I$ such that $\mathbf{Q}_1(z|y) > 0$ and $\mathbf{Q}_1(z|y') = 0$ for any $y' \neq y$.

Then we show that ULDP is immune to the post-processing by this class of randomized algorithms.

Proposition 11 (Post-processing)**.** *Let* $\varepsilon \geq 0$*. Let* $\mathcal{Z}_P$ *and* $\mathcal{Z}_I$ *be sets of protected and invertible data respectively, and* $\mathcal{Z} = \mathcal{Z}_P \cup \mathcal{Z}_I$*. Let* $\mathbf{Q}_1$ *be a randomized algorithm from* $\mathcal{Y}$ *to* $\mathcal{Z}$ *that preserves data types. If an obfuscation mechanism* $\mathbf{Q}_0$ *from* $\mathcal{X}$ *to* $\mathcal{Y}$ *provides* $(\mathcal{X}_S, \mathcal{Y}_P, \varepsilon)$*-ULDP then the composite function* $\mathbf{Q}_1 \circ \mathbf{Q}_0$ *provides* $(\mathcal{X}_S, \mathcal{Z}_P, \varepsilon)$*-ULDP.*

For example, ULDP is immune to data cleaning operations (e.g., transforming values, merging disparate values) [33] as long as they are represented as $\mathbf{Q}_1$ explained above.

Note that $\mathbf{Q}_1$ needs to preserve data types for utility (i.e., to make all $y \in \mathcal{Y}_I$ invertible, as in Definition 2, after post-processing), and the DP guarantee for $y \in \mathcal{Y}_P$ is preserved by any post-processing algorithm. Specifically, by (5), for any randomized post-processing algorithm $\mathbf{Q}_1^*$, any obfuscated data $z \in \mathcal{Z}$ obtained from $y \in \mathcal{Y}_P$ via $\mathbf{Q}_1^*$, and any $x, x' \in \mathcal{X}$, we have: $\Pr(z|x) \leq e^{\varepsilon} \Pr(z|x')$.

A.2 Compatibility with LDP

Assume that data collectors A and B adopt a mechanism $\mathbf{Q}_A$ providing $(\mathcal{X}_S, \mathcal{Y}_P, \varepsilon_A)$-ULDP and a mechanism $\mathbf{Q}_B$ providing ε_B-LDP, respectively. In this case, all protected data in the data collector A can be combined with all obfuscated data in the data collector B (i.e., data integration) to perform data analysis under LDP. More specifically, assume that Alice transforms her sensitive personal data in $\mathcal{X}_S$ into $y_A \in \mathcal{Y}_P$ (resp. $y_B \in \mathcal{Y}$) using $\mathbf{Q}_A$ (resp. $\mathbf{Q}_B$), and sends y_A (resp. y_B) to the data collector A (resp. B) to request two different services (e.g., location check-in for A and point-of-interest search for B). Then, the composition $(\mathbf{Q}_A, \mathbf{Q}_B)$ in parallel has the following property:

Proposition 12 (Compatibility with LDP)**.** *If* $\mathbf{Q}_A$ *and* $\mathbf{Q}_B$ *respectively provide* $(\mathcal{X}_S, \mathcal{Y}_P, \varepsilon_A)$*-ULDP and* ε_B*-LDP, then for any* $x, x' \in \mathcal{X}$*,* $y_A \in \mathcal{Y}_P$*, and* $y_B \in \mathcal{Y}$*, we have:*

$$(\mathbf{Q}_A, \mathbf{Q}_B)(y_A, y_B|x) \leq e^{\varepsilon_A + \varepsilon_B} (\mathbf{Q}_A, \mathbf{Q}_B)(y_A, y_B|x').$$

Proposition 12 implies that Alice's sensitive personal data in $\mathcal{X}_S$ is protected by $(\varepsilon_A + \varepsilon_B)$-LDP after the data integration.

B Relationship between LDP, ULDP and OSLDP

In this section, we introduce the notion of OSLDP (One-sided LDP), a local model version of OSDP (One-sided DP) proposed in a preprint [17]:

Definition 6 ($(\mathcal{X}_S, \varepsilon)$-OSLDP)**.** *Given* $\mathcal{X}_S \subseteq \mathcal{X}$ *and* $\varepsilon \in \mathbb{R}_{\geq 0}$*, an obfuscation mechanism* $\mathbf{Q}$ *from* $\mathcal{X}$ *to* $\mathcal{Y}$ *provides* $(\mathcal{X}_S, \varepsilon)$*-OSLDP if for any* $x \in \mathcal{X}_S$*, any* $x' \in \mathcal{X}$ *and any* $y \in \mathcal{Y}$*, we have*

$$\mathbf{Q}(y|x) \leq e^{\varepsilon} \mathbf{Q}(y|x'). \tag{23}$$

OSLDP is a special case of OSDP [17] that takes as input personal data of a single user. Unlike ULDP, OSLDP allows the transition probability $\mathbf{Q}(y|x')$ from non-sensitive data $x' \in \mathcal{X}_N$ to be very large for any $y \in \mathcal{Y}$, and hence does not provide ε-LDP for $\mathcal{Y}$ (whereas ULDP provides ε-LDP for $\mathcal{Y}_P$). Thus, OSLDP can be regarded as a "relaxation" of ULDP. In fact, the following proposition holds:

Proposition 13. *If an obfuscation mechanism* $\mathbf{Q}$ *provides* $(\mathcal{X}_S, \mathcal{Y}_P, \varepsilon)$*-ULDP, then it also provides* $(\mathcal{X}_S, \varepsilon)$*-OSLDP.*

It should be noted that if an obfuscation mechanism provides ε-LDP, then it obviously provides $(\mathcal{X}_S, \mathcal{Y}_P, \varepsilon)$-ULDP, where $\mathcal{Y}_P = \mathcal{Y}$. Therefore, $(\mathcal{X}_S, \mathcal{Y}_P, \varepsilon)$-ULDP is a privacy measure that lies between ε-LDP and $(\mathcal{X}_S, \varepsilon)$-OSLDP.

We use ULDP instead of OSLDP for the following two reasons. The first reason is that ULDP is compatible with LDP, and makes it possible to perform data integration and data analysis under LDP (Proposition 12). OSLDP does not have this property in general, since it allows the transition probability $\mathbf{Q}(y|x')$ from non-sensitive data $x' \in \mathcal{X}_N$ to be very large for any $y \in \mathcal{Y}$, as explained above.

The second reason, which is more important, is that *the utility of OSLDP is not better than that of ULDP*. Intuitively, it can be explained as follows. First, although $\mathcal{Y}_P$ is not explicitly defined in OSLDP, we can define $\mathcal{Y}_P$ in OSLDP as the *image of* $\mathcal{X}_S$, and $\mathcal{Y}_I$ as $\mathcal{Y}_I = \mathcal{Y} \setminus \mathcal{Y}_P$, analogously to ULDP. Then, OSLDP differs from ULDP in the following two points: (i) it allows the transition probability $\mathbf{Q}(y|x')$ from $x' \in \mathcal{X}_N$ to $y \in \mathcal{Y}_P$ to be very large (i.e., (5) may not satisfied); (ii) it allows $y \in \mathcal{Y}_I$ to be non-invertible. (i.e., (4) may not satisfied). Regarding (i), it is important to note that the transition probability from $x' \in \mathcal{X}_N$ to $\mathcal{Y}_I$ decreases with increase in the transition probability from x' to $\mathcal{Y}_P$. Thus, (i) and (ii) only allow us to mix non-sensitive data with sensitive data or other non-sensitive data, and reduce the amount of output data $y \in \mathcal{Y}_I$ that can be inverted to $x \in \mathcal{X}_N$.

Then, each OSLDP mechanism can be decomposed into a ULDP mechanism and a randomized post-processing that mixes non-sensitive data with sensitive data or other non-sensitive data. Note that this post-processing does not preserve data types (in Definition 5), and hence OSLDP does not have a compatibility with LDP as explained above. In addition, although the post-processing might improve privacy for non-sensitive data, we would like to protect sensitive data in this paper and ULDP is sufficient for this purpose; i.e., it guarantees ε-LDP for sensitive data.

Since the information is generally lost (never gained) by mixing data via the randomized post-processing, the utility of OSLDP is not better than that of ULDP (this holds for the information-theoretic utility such as mutual information and f-divergences [30] because of the data processing inequality [9, 13]; we also show this for the expected l_1 and l_2 losses at the end of Appendix B). Thus, it suffices to consider ULDP for our goal of designing obfuscation mechanisms that achieve high utility while providing LDP for sensitive data (as tdescribed in Section 1).

We now formalize our claim as follows:

Proposition 14. *Let $\mathcal{M}_O$ be the class of all mechanisms from $\mathcal{X}$ to $\mathcal{Y}$ providing $(\mathcal{X}_S, \varepsilon)$-OSLDP. For any $\mathbf{Q}_O \in \mathcal{M}_O$, there exist two sets $\mathcal{Z}$ and $\mathcal{Z}_P$, a $(\mathcal{X}_S, \mathcal{Z}_P, \varepsilon)$-ULDP mechanism $\mathbf{Q}_U$ from $\mathcal{X}$ to $\mathcal{Z}$, and a randomized algorithm $\mathbf{Q}_R$ from $\mathcal{Z}$ to $\mathcal{Y}$ such that:*

$$\mathbf{Q}_O = \mathbf{Q}_R \circ \mathbf{Q}_U. \tag{24}$$

From Proposition 14, we show that the expected l_1 and l_2 losses of OSLDP are not better than those of ULDP as follows. For any OSLDP mechanism $\mathbf{Q}_O \in \mathcal{M}_O$ and any estimation method λ_O from data in $\mathcal{Y}$, we can construct a ULDP mechanism $\mathbf{Q}_U$ in (24) and an estimation method λ_U that perturbs data in $\mathcal{Z}$ via $\mathbf{Q}_R$ and then estimates a distribution from data in $\mathcal{Y}$ via λ_O. $\mathbf{Q}_U$ and λ_U provide the same expected l_1 and l_2 losses as $\mathbf{Q}_O$ and λ_O, and there might also exist ULDP mechanisms and estimation methods from data in $\mathcal{Z}$ that provide smaller expected l_1 and l_2 losses. Thus, the expected l_1 and l_2 losses of OSLDP are not better than those of ULDP.

C L2 loss of the utility-optimized Mechanisms

C.1 Utility Analysis

uRR in the general case. We first present the l_2 loss of the $(\mathcal{X}_S, \varepsilon)$-uRR.

Proposition 15 (l_2 loss of the uRR)**.** *The expected l_2 loss of the $(\mathcal{X}_S, \varepsilon)$-uRR mechanism is given by:*

$$\mathbb{E}[l_2^2(\hat{\mathbf{p}}, \mathbf{p})] = \frac{2(e^\varepsilon - 1)(|\mathcal{X}_S| - \mathbf{p}(\mathcal{X}_S)) + |\mathcal{X}_S|(|\mathcal{X}_S| - 1)}{n(e^\varepsilon - 1)^2} + \frac{1}{n}\big(1 - \sum_{x \in \mathcal{X}} \mathbf{p}(x)^2\big). \tag{25}$$

When $0 < \varepsilon < \ln(|\mathcal{X}_N| + 1)$, the l_2 loss is maximized by the uniform distribution $\mathbf{p}_{U_N}$ over $\mathcal{X}_N$.

Proposition 16. *For any $0 < \varepsilon < \ln(|\mathcal{X}_N| + 1)$, (25) is maximized by $\mathbf{p}_{U_N}$:*

$$\begin{aligned} \mathbb{E}\left[l_2^2(\hat{\mathbf{p}}, \mathbf{p})\right] &\leq \mathbb{E}\left[l_2^2(\hat{\mathbf{p}}, \mathbf{p}_{U_N})\right] \\ &= \tfrac{|\mathcal{X}_S|(|\mathcal{X}_S| + 2e^\varepsilon - 3)}{n(e^\varepsilon - 1)^2} + \tfrac{1}{n}\big(1 - \tfrac{1}{|\mathcal{X}_N|}\big). \end{aligned} \tag{26}$$

When $\varepsilon \geq \ln(|\mathcal{X}_N| + 1)$, the l_2 loss is maximized by a mixture of the uniform distribution $\mathbf{p}_{U_S}$ over $\mathcal{X}_S$ and the uniform distribution $\mathbf{p}_{U_N}$ over $\mathcal{X}_N$.

Proposition 17. *For any $\varepsilon \geq \ln(|\mathcal{X}_N| + 1)$, (25) is maximized by $\mathbf{p}^*$ in (13):*

$$\mathbb{E}\left[l_2^2(\hat{\mathbf{p}}, \mathbf{p})\right] \leq \mathbb{E}\left[l_2^2(\hat{\mathbf{p}}, \mathbf{p}^*)\right] = \frac{(|\mathcal{X}_S| + e^\varepsilon - 1)^2}{n(e^\varepsilon - 1)^2}\Big(1 - \frac{1}{|\mathcal{X}|}\Big).$$

uRR in the high privacy regime. Consider the high privacy regime where $\varepsilon \approx 0$. In this case, $e^\varepsilon - 1 \approx \varepsilon$. By using this approximation, the right-hand side of (26) in Proposition 16 can be simplified as follows:

$$\mathbb{E}\left[l_2^2(\hat{\mathbf{p}}, \mathbf{p})\right] \leq \mathbb{E}\left[l_2^2(\hat{\mathbf{p}}, \mathbf{p}_{U_N})\right] \approx \tfrac{|\mathcal{X}_S|(|\mathcal{X}_S| - 1)}{n\varepsilon^2}.$$

It is shown in [29] that the expected l_2 loss of the ε-RR is at most $\frac{|\mathcal{X}|(|\mathcal{X}| - 1)}{n\varepsilon^2}$ when $\varepsilon \approx 0$. Thus, the expected l_2 loss of the $(\mathcal{X}_S, \varepsilon)$-uRR is much smaller than that of the ε-RR when $|\mathcal{X}_S| \ll |\mathcal{X}|$.

uRR in the low privacy regime. Consider the low privacy regime where $\varepsilon = \ln |\mathcal{X}|$ and $|\mathcal{X}_S| \ll |\mathcal{X}|$. By Proposition 17, the expected l_2^2 loss of the $(\mathcal{X}_S, \varepsilon)$-uRR is given by:

$$\mathbb{E}\left[l_2^2(\hat{\mathbf{p}}, \mathbf{p})\right] \leq \mathbb{E}\left[l_2^2(\hat{\mathbf{p}}, \mathbf{p}^*)\right] \approx \frac{1}{n}.$$

It should be noted that the expected l_2 loss of the non-private mechanism is at most $\frac{1}{n}(1 - \frac{1}{|\mathcal{X}|})$ [29], and that $\frac{1}{n}(1 - \frac{1}{|\mathcal{X}|}) \approx \frac{1}{n}$ when $|\mathcal{X}| \gg 1$. Thus, when $\varepsilon = \ln |\mathcal{X}|$ and $|\mathcal{X}_S| \ll |\mathcal{X}|$, the $(\mathcal{X}_S, \varepsilon)$-uRR achieves almost the same data utility as the non-private mechanism, whereas the expected l_1 loss of the ε-RR is four times larger than that of the non-private mechanism [29].

Utility-optimized RAPPOR in the general case. We then present the l_2 loss of the $(\mathcal{X}_S, \varepsilon)$-uRAP.

Proposition 18 (l_2 loss of the uRAP)**.** *Then the expected l_2-loss of the $(\mathcal{X}_S, \varepsilon)$-uRAP mechanism is given by:*

$$\begin{aligned} &\mathbb{E}\left[l_2^2(\hat{\mathbf{p}}, \mathbf{p})\right] \\ &= \frac{1}{n}\Big(1 + \tfrac{(|\mathcal{X}_S| + 1)e^{\varepsilon/2} - 1}{(e^{\varepsilon/2} - 1)^2} - \tfrac{1}{e^{\varepsilon/2} - 1}\mathbf{p}(\mathcal{X}_S) - \sum_{j=1}^{|\mathcal{X}|} \mathbf{p}(x_j)^2\Big). \end{aligned} \tag{27}$$

For any $0 < \varepsilon < 2\ln(\frac{|\mathcal{X}_N|}{2} + 1)$, the l_2 loss is maximized by the uniform distribution $\mathbf{p}_{U_N}$ over $\mathcal{X}_N$.

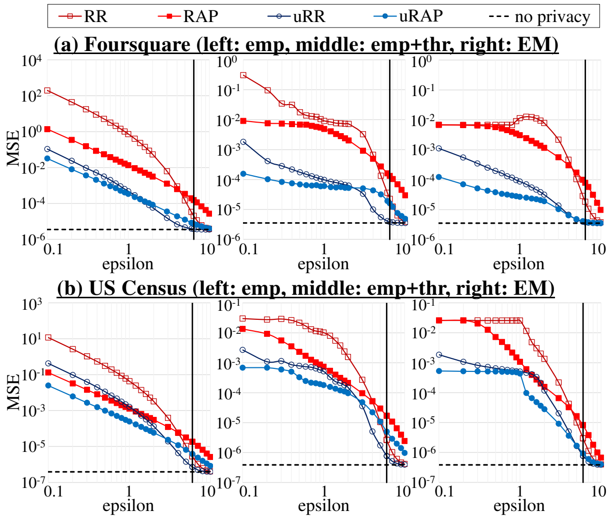

Figure 10: ε vs. MSE (common-mechanism). A bold line parallel to the y-axis represents $\varepsilon = \ln|\mathcal{X}|$.

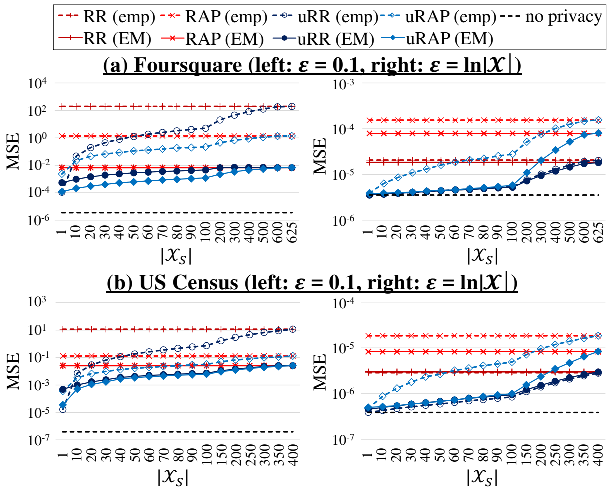

Figure 11: $|\mathcal{X}_S|$ vs. MSE when $\varepsilon = 0.1$ or $\ln|\mathcal{X}|$.

Proposition 19. *For any* $0 < \varepsilon < 2\ln(\frac{|\mathcal{X}_N|}{2}+1)$*, the* l_2*-loss* $\mathbb{E}\left[l_2^2(\hat{\mathbf{p}},\mathbf{p})\right]$ *is maximized when* $\mathbf{p} = \mathbf{p}_{U_N}$*:*

$$\mathbb{E}\left[l_2^2(\hat{\mathbf{p}},\mathbf{p})\right] \le \mathbb{E}\left[l_2^2(\hat{\mathbf{p}},\mathbf{p}_{U_N})\right] = \frac{1}{n}\left(1+\frac{(|\mathcal{X}_S|+1)e^{\varepsilon/2}-1}{(e^{\varepsilon/2}-1)^2}-\frac{1}{|\mathcal{X}_N|}\right). \tag{28}$$

uRAP in the high privacy regime. Consider the high privacy regime where $\varepsilon \approx 0$. In this case, $e^{\varepsilon/2}-1 \approx \varepsilon/2$. By using this approximation, the right-hand side of (28) in Proposition 19 can be simplified as:

$$\mathbb{E}\left[l_2^2(\hat{\mathbf{p}},\mathbf{p})\right] \le \mathbb{E}\left[l_2^2(\hat{\mathbf{p}},\mathbf{p}_{U_N})\right] \approx \frac{4|\mathcal{X}_S|}{n\varepsilon^2}.$$

Thus, the expected l_2 loss of the uRAP is at most $\frac{4|\mathcal{X}_S|}{n\varepsilon^2}$ in the high privacy regime. It is shown in [29] that the expected l_2 loss of the ε-RAPPOR is at most $\frac{4|\mathcal{X}|}{n\varepsilon^2}(1-\frac{1}{|\mathcal{X}|})$ when $\varepsilon \approx 0$. Thus, the expected l_2 loss of the $(\mathcal{X}_S,\varepsilon)$-uRAP is much smaller than that of the ε-RAPPOR when $|\mathcal{X}_S| \ll |\mathcal{X}|$.

Note that the expected l_2 loss of the uRAP in the worst case can also be expressed as $\Theta(\frac{|\mathcal{X}_S|}{n\varepsilon^2})$ in this case. As described in Section 3.2, this is "order" optimal among all ULDP mechanisms.

uRAP in the low privacy regime. If $\varepsilon = \ln|\mathcal{X}|$ and $|\mathcal{X}_S| \ll \sqrt{|\mathcal{X}|}$, the right-hand side of (28) in Proposition 19 can be simplified as follows:

$$\mathbb{E}\left[l_2^2(\hat{\mathbf{p}},\mathbf{p})\right] \le \mathbb{E}\left[l_2^2(\hat{\mathbf{p}},\mathbf{p}_{U_N})\right] \approx \frac{1}{n}. \tag{29}$$

Note that the expected l_2 loss of the non-private mechanism is at most $\frac{1}{n}(1-\frac{1}{|\mathcal{X}|})$ [29], and that $\frac{1}{n}(1-\frac{1}{|\mathcal{X}|}) \approx \frac{1}{n}$ when $|\mathcal{X}| \gg 1$. Thus, when $\varepsilon = \ln|\mathcal{X}|$ and $|\mathcal{X}_S| \ll \sqrt{|\mathcal{X}|}$, the $(\mathcal{X}_S,\varepsilon)$-uRAP achieves almost the same data utility as the non-private mechanism, whereas the expected l_2 loss of the ε-RAPPOR is $\sqrt{|\mathcal{X}|}$ times larger than that of the non-private mechanism [29].

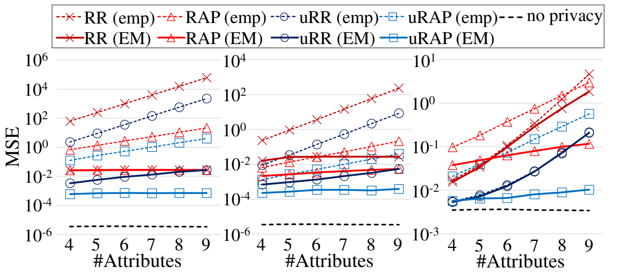

Figure 12: Number of attributes vs. MSE (US Census dataset; left: $\varepsilon = 0.1$, middle: $\varepsilon = 1.0$, right: $\varepsilon = 6.0$).

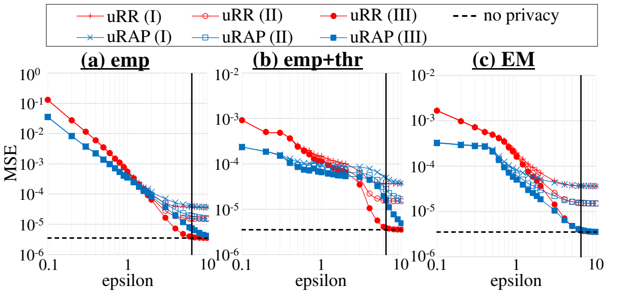

Figure 13: ε vs. MSE (personalized-mechanism) ((I): w/o knowledge, (II) POI distribution, (III) true distribution).

C.2 Experimental Results of the MSE

Figures 10, 11, 12, and 13 show the results of the MSE corresponding to Figures 5, 6, 7, and 8, respectively. It can be seen that a tendency similar to the results of the TV is obtained for the results of the MSE, meaning that our proposed methods are effective in terms of both the l_1 and l_2 losses.