Set-valued Data Publication with Local Privacy: Tight Error Bounds and Efficient Mechanisms

Shaowei Wang,Yuqiu Qian*, Jiachun Du
Tencent Games
{seawellwang,yuqiuqian,kevinjcdu}@tencent.com

Wei Yang, Liusheng Huang, Hongli Xu†
University of Science and Technology of China
{lshuang,qubit,xuhongli}@ustc.edu.cn

ABSTRACT

Most user-generated data in online services are presented as set-valued data, e.g., visited website URLs, recently used Apps by a person, and etc. These data are of great value to service providers, but also bring privacy concerns if collected and analyzed directly. To tackle potential privacy threatens, local differential privacy (LDP) attracts increasing attention nowadays. However, existing approaches only provide sub-optimal error bound for set-valued data distribution estimation with LDP. Besides, it is computational expensive and communication expensive to use for high dimensional set-valued data, considering large domains in real scenarios. Thus, existing approaches are unpractical to use on resource-constrained user-side devices (e.g., smartphones and wearable devices). In this paper, we propose a utility-optimal and efficient set-valued data publication method (i.e., *wheel mechanism*). On the user side, each user contributes only one numerical value to represent their privatized data. The computational complexity is $O(\min\{m \log m, me^\epsilon\})$ and communication cost is $O(\log(me^\epsilon))$ bits, while existing approaches usually depend on $O(d)$ or $O(\log d)$, where m is the number of items in the set-valued data ($m \equiv 1$ for categorical data), d is the domain size (usually $d \gg m$) and ϵ is the privacy budget. On the server side, the estimator takes numerical values from users as input and derives an unbiased distribution estimation. Theoretical results show that estimation error bounds are improved from previously known $\Theta(\frac{m^2 d}{n\epsilon^2})$ to the optimal rate $\Theta(\frac{md}{n\epsilon^2})$. Results on extensive experiments demonstrate that our proposed wheel mechanism is 3-100x faster than existing approaches, meanwhile has optimal statistical efficiency.

*Corresponding author.

†Supported in part by the National Science Foundation of China (NSFC) under Grants U1709217, 61936015, 61822210 and Anhui Initiative in Quantum Information Technologies under No.AHY150300.

Proceedings of the VLDB Endowment, Vol. 13, No. 8
ISSN 2150-8097.
DOI: https://doi.org/10.14778/3389133.3389140

PVLDB Reference Format:
Shaowei Wang, Yuqiu Qian, Jiachun Du, Wei Yang, Liusheng Huang and Hongli Xu. Set-valued Data Publication with Local Privacy: Tight Error Bounds and Efficient Mechanisms. *PVLDB*, 13(8): 1234-1247, 2020.
DOI: https://doi.org/10.14778/3389133.3389140

1. INTRODUCTION

User data (e.g., personal information, location data, and service usage data) is the key to the success of many online services and applications. When represented in bit-vector forms, these data can be usually deemed as high dimensional set-valued data (i.e., set-valued data with large size of data domain), such as a list of website URLs one user clicked [20], a list of Apps recently used by a user [9], multi-dimensional personal information [19, 6, 4], etc.

With the help of statistical analyzing and mining methods, service providers can provide better service to users. However, it brings privacy concerns at the same time, such as deriving individual identities from users' personal information, mining social statuses or activities from clicked websites of users [30]. Therefore, several regulations and laws (e.g., the GDPR regulation [32] in the Europe Union, and the CCPA act [17] in the state of California) have been enacted or activated in recent years. To protect user data's privacy and assist in complying with these privacy-related regulations, it's an urgent need to study privacy-preserving data analyzing methods for the benefits of both users and service providers.

Local differential privacy (LDP) [10] emerges as the *de facto* standard for data privacy preserving in computer networking systems, which sanitizes user data on their local machines or devices without trusting any parties (e.g., service providers or edge servers). It originates from centralized differential privacy (DP) [12] for databases, which needs a trustable data curator to collect and manage raw data from users (e.g., in [3, 24, 40, 5]). Compared to classic generalization or perturbation methods for privacy preserving (e.g., k-anonymity [28] and ℓ-diversity [26] in [16, 31, 21]), LDP approaches are more robust to adversaries with background knowledge. Besides, if compared to privacy preserving methods based on secure multi-party computation (e.g., in [8, 7, 22]), the computation and interaction costs of LDP are usually lower [25]. Therefore, data analyses with LDP have been widely studied in the community (e.g., mean estimation for numerical data [10], distribution estimation on categorical data [10, 2], location data [35] and set-valued data [27]) and deployed in popular services (e.g., Google Chrome browser [13, 14] and Apple iOS/Mac OS systems [18, 29]).

Table 1: Comparison of ϵ-LDP approaches for categorical data distribution estimation when $e^\epsilon \ll d$.

approaches	user	user-server	server	MSE
RR[23]	$O(1)$	$O(\log d)$	$O(n+d)$	$\Theta(\frac{d^2}{n(e^\epsilon-1)^2})$
d-RAPPOR[13]	$O(d)$	$O(\frac{d\log(e^{\frac{\epsilon}{2}}+1)}{e^{\frac{\epsilon}{2}}+1})$	$O(n\cdot d)$	$\Theta(\frac{e^\epsilon d}{n(e^{\epsilon/2}-1)^2})$
1-bit[2]	$O(1)$	$O(d)+O(1)$	$O(n\cdot d)$	$\Theta(\frac{e^{2\epsilon}d}{n(e^\epsilon-1)^2})$
k-Subset[34, 38]	$O(d)$	$O(\frac{d\log(e^\epsilon+1)}{e^\epsilon+1})$	$O(n\cdot d)$	$\Theta(\frac{e^\epsilon d}{n(e^\epsilon-1)^2})$
Hadamard[1]	$O(d)$	$O(\log d)$	$O(n+d)$	$\Theta(\frac{e^\epsilon d}{n(e^\epsilon-1)^2})$
this work	$O(1)$	$O(\log(e^\epsilon+1))$	$O(n\cdot d)$	$\Theta(\frac{e^\epsilon d}{n(e^\epsilon-1)^2})$

1.1 Existing Approaches

As to high dimensional set-valued data and categorical data (i.e., one special case of set-valued data), existing ϵ-LDP approaches suffer from high computation or high communication costs, which is intolerable for user-side devices. We here summarize the complexities of representative ϵ-LDP distribution estimation approaches for both categorical data (shown in Table 1) and set-valued data (shown in Table 2). The parameter ϵ is called the privacy budget or privacy level and is usually chosen in $(0.01, 3.0]$. The *user/server* columns show computational complexities on the user/server side correspondingly, the *user-server* column shows the communication complexities between a user and the server, and the *MSE* column shows their mean squared error bounds for distribution estimation as function of the privacy budget ϵ: $\sup_\theta \mathbb{E}[|\hat{\theta}-\theta|_2^2]$, where θ and $\hat{\theta}$ represent the true and estimated item distribution respectively.

Table 2: Comparison of ϵ-LDP approaches for set-valued data distribution estimation when $\epsilon = O(1)$.

approaches	user	user-server	server	MSE
d-RAPPOR[13]	$O(d)$	$O(d)$	$O(nd)$	$\Theta(\frac{m^2 d}{n\epsilon^2})$
sampling RR[27]	$O(m)$	$O(\log d)$	$O(n+d)$	$\Theta(\frac{m^2 d}{n\epsilon^2})$
sampling 1-bit[27]	$O(m)$	$O(d)+O(1)$	$O(d)$	$\Theta(\frac{m^2 d}{n\epsilon^2})$
PrivSet[33]	$O(d)$	$O(d)$	$O(nd)$	-
this work	$O(m)$	$O(\log m)$	$O(nd)$	$\Theta(\frac{md}{n\epsilon^2})$

Specifically, ϵ-LDP distribution estimation approaches for *categorical data* based on randomized response (e.g., in [10, 14, 34, 38, 1, 36]) usually require $O(d)$ computation complexity on the user side, and sending $O(d)$ bits (e.g. in [10, 13, 14, 36]) or $O(\log d)$ bits (e.g., in [34, 23, 38, 1]) from each user to the server. [2] reduces the communication cost to only one bit, accompanied by firstly sending $\Theta(d)$ bits from the server to the user. However, the estimation error in [2] is not optimal for a relatively large privacy budget, leaves a multiplicative $\exp(\epsilon) \geq 1$ gap comparing to the optimal minimax rate of $\Theta(\frac{e^\epsilon d}{n(e^\epsilon-1)^2})$ [38].

The RAPPOR mechanism [13] can also be employed for distribution estimation with *set-valued data*. However, its computational/communication costs on the user side become both $O(d)$, and its error bound is $\Theta(\frac{m^2\cdot d}{n\epsilon^2})$. Later, [27] proposes to decompose set-valued data into categorical data by random item sampling, following ordinary categorical ϵ-LDP approaches (e.g., the randomized response [23] or the 1-bit [2] mechanism). Though the random-sampling based approaches are almost efficient as ones for categorical data, their estimation error bounds depend on a factor of m^2 due to the sampling. A recent work [33] called PrivSet mechanism randomly responses with a subset for set-valued data, which needs $O(d)$ computation and $O(d)$ communication costs on the user side. It shows remarkable empirical results but lacks theoretical error guarantees. Actually, one of the most fundamental questions on set-valued data distribution estimation with ϵ-LDP: *where is the minimax lower error bound?* is still not explored in the literature.

1.2 Our Contributions

In summary, existing ϵ-LDP approaches for set-valued (and categorical) data distribution estimation suffer from following drawbacks:

I. **Domain dependence**: the computational / communication overheads on the user side and the storage costs on the server side depend on the item domain size d, which is inefficient for high-dimensional set-valued data;

II. **Statistical inefficiency**: the distribution estimator suffers from sub-optimal error bounds, and thus more samples from user population (i.e. larger n) are needed to achieve desirable estimation accuracy.

In this paper, we propose an efficient and optimal ϵ-LDP mechanism (i.e., *wheel mechanism*) for set-valued/ categorical data distribution estimation. It is domain agnostic (i.e., independent of dimension size d) and utility optimal. Specifically, the wheel mechanism maps the set-valued data to one or multiple points in a circled wheel, and then designs a calibrated probability distribution that satisfies ϵ-LDP based on these points. After that, it samples one numerical value as the output from the wheel according to the probability distribution. The mechanism needs only $O(\log_2(e^\epsilon+1))$ bits communication between a user and the server; Since the mapping procedure is done by a user-specific hash function, its computation costs are $O(1)$ for categorical data and $O(m)$ for set-valued data. By carefully designing the randomization distribution, the server could derive unbiased item distribution over the data domain with the optimal error bound.

Our contributions can be summarized as follows:

- We give tight minimax lower bounds of the ϵ-LDP set-valued distribution estimation problem, showing that the mean squared error $\Theta(\frac{md}{n\epsilon^2})$ is optimal.
- For categorical data (set-valued data that $m \equiv 1$) distribution estimation under ϵ-LDP, we propose the wheel mechanism. It only needs $O(1)$ computation and $O(\log_2(e^\epsilon+1))$ communication costs on the user side. We theoretically prove its optimality in terms of distribution estimation error.
- We extend the proposed wheel mechanism to set-valued data, which costs $\min\{O(me^\epsilon), O(m\log m)\}$

computation and $O(\log_2(e^\epsilon + 1))$ communication resources. The theoretical error bounds for item distribution estimation is improved to the optimal risk of $\Theta(\frac{md}{n\epsilon^2})$ from previously best $\Theta(\frac{m^2 d}{n\epsilon^2})$.

- Through extensive experiments, we validate the design of and analyses on the wheel mechanism, demonstrate its efficiency and effectiveness, and show around 3-100x running speed boost on the user side.

The remainder of the paper is organized as follows. Section 2 formally introduces categorical/set-valued data, local differential privacy and the distribution estimation framework. Section 3 gives tight lower bounds for set-valued data under the minimax risk framework. The wheel mechanism for categorical data along with its theoretical analysis is shown in section 4. The wheel mechanism is extended to set-valued data in section 5 with theoretical analysis. Section 6 evaluates the wheel mechanism. Finally, section 7 concludes the paper.

2. PRELIMINARIES

We here briefly introduce categorical/set-valued data, local differential privacy (ϵ-LDP) and the framework of ϵ-LDP data distribution estimation. Notations across the paper are summarized in Table 3.

Table 3: List of notations.

Notation	Description
$\mathcal{X}$	The item domain of user data
d	The size of the item domain
i	The index of items
m	The cardinality of set-valued data
$\mathcal{X}^m$	The set of subsets that $\mathbf{x} \subseteq \mathcal{X}$ with size m
n	The number of users or participants
j	The index of users
$\mathbf{x}^j$	The set-valued data of user j
$\mathbf{z}^j$	The private view from user j
θ	User data's distribution over item domain
$\hat{\theta}$	The estimation of θ
ϵ	The privacy budget (privacy level)
v_i	The hashed value of item X_i in $[0.0, 1.0)$
C_{v_i}	The coverage area of item X_i
p	The length parameter of a coverage area
P_t	True coverage probability of each item
P_f	False coverage probability of each item
l	The total length of the union coverage area

2.1 Categorical/Set-valued Data

Let $\mathcal{X} = \{X_1, X_2, ..., X_d\}$ denote the item domain, the categorical data $x \in \mathcal{X}$ is an item from the domain, and the set-valued data $\mathbf{x} \subseteq \mathcal{X}$ is a subset of the domain. In many practical applications, the size of the item domain $d = |\mathcal{X}|$ is large (e.g., $d > 100$), such as website URLs, English words, all possible Apps, and possible values of multi-dimensional data. Such set-valued is called a high dimensional one.

Let m denote the number of items that appeared in set-valued data $\mathbf{x}$. For the simplicity of analyses, we assume that the size of each set-valued data (i.e., m) is the same across all users. Note that categorical data is one special case of set-valued data, such that $m = 1$.

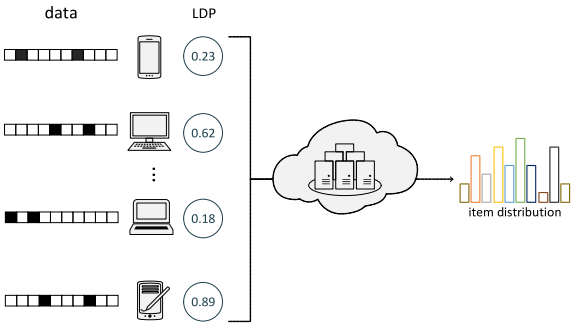


Figure 1: The model of ϵ-LDP set-valued data publication and distribution estimation with the wheel mechanism.

2.2 Local Differential Privacy (ϵ-LDP)

The ϵ-LDP ensures that privacy attackers can only gain a limited multiplicative factor over their prior knowledge after observing a privatized output z. Formally, let K denote a randomization mechanism that intends to provide ϵ-LDP. Assume that each user's true value $\mathbf{x}$ belongs to the input data domain $\mathcal{X}^m = \{\mathbf{x} \mid \mathbf{x} \subseteq \mathcal{X} \text{ and } |\mathbf{x}| = m\}$, the definition of (non-interactive) ϵ-LDP is given in the Definition 1.

Definition 1 ((NON-INTERACTIVE) ϵ-LDP [10]). *Let $\mathcal{D}_K$ denote the output domain, a randomized mechanism K satisfies local ϵ-differential privacy iff for any data pair $s, s' \in \mathcal{X}^m$, and any output $t \in \mathcal{D}_K$,*

$$\mathbb{P}[K(s) = t] \leq \exp(\epsilon) \cdot \mathbb{P}[K(s') = t]$$

stands, where ϵ is called the privacy level or privacy budget.

Practical values for ϵ falls in $(0.01, 3.0]$ so that the privacy attacker's knowledge cannot grow too much.

If user j adopts a randomization mechanism K^j according to former outputs $\{\mathbf{z}^1, ..., \mathbf{z}^{j-1}\}$ instead of a fixed universal K, we call such randomization mechanism provides interactive ϵ-LDP (in Definition 2).

Definition 2 (INTERACTIVE ϵ-LDP [10]). *Let $\mathcal{D}_{K^j}$ denote the output domain of a randomized K^j ($j \in [1, n]$), the random variable $\mathbf{z}^j$ is an ϵ-LDP view of $\mathbf{x}^j$. If*

$$\sup_{t \in \mathcal{D}_{K^j}} \frac{\mathbb{P}[K^j(s) = t | \mathbf{x}^j = s, \mathbf{z}^1 = z^1, ..., \mathbf{z}^{j-1} = z^{j-1}]}{\mathbb{P}[K^j(s') = t | \mathbf{x}^j = s', \mathbf{z}^1 = z^1, ..., \mathbf{z}^{j-1} = z^{j-1}]} \leq \exp(\epsilon)$$

holds for all $z^1 \in \mathcal{D}_{K^1}, ..., z^{j-1} \in \mathcal{D}_{K^{j-1}}$, any data pair $s, s' \in \mathcal{X}^m$, and any output $t \in \mathcal{D}_{K^j}$, we call such mechanism $\mathbf{K} = \{K^1, ..., K^n\}$ satisfies ϵ-LDP in an interactive setting.

Note that the non-interactive version of ϵ-LDP (in Definition 1) is a special case when $K^j \equiv K$.

2.3 Distribution Estimation Framework

Distribution estimation is a fundamental building block for many statistical analyses (e.g., histogram estimation, hypothesis testing, causal inference, and building machine

learning models). In the data distribution estimation framework (shown in Figure 1), every user u^j independently randomizes their true value $\mathbf{x}^j$ to a publishable view $\mathbf{z}^j$ through a ϵ-LDP mechanism and sends it to the service. The server side then estimates the item distribution over $\mathcal{X}$ of all users based on all collected $\mathbf{z}^j$.

Specifically, we denote the true item distribution as $\theta = \{\theta_1, \theta_2, ..., \theta_d\}$, where

$$\theta_i = \frac{1}{n} \cdot \#\{\mathbf{x}^j \mid j \in [1, n] \text{ and } X_i \in \mathbf{x}^j\}, \tag{1}$$

and $\hat{\theta}$ as the estimated item distribution.

A good ϵ-LDP mechanism should have low computation and communication burden on the user side, and minimum estimation error that matches minimax lower bounds for the service provider.

3. TIGHT MINIMAX LOWER BOUNDS

As an extension to lower bounds on ϵ-LDP distribution estimation problem on categorical data [10, 38], we here present lower bounds for ϵ-LDP (non-sparse or sparse) set-valued data distribution estimation. The intuition behind is that by decomposing the set-valued data into multiple categorical data, we can follow a similar procedure for categorical data in [10, 38].

3.1 Classic minimax risks

Let $\mathcal{P}$ denote all possible probability distributions over the data universe $\mathcal{X}^m$. For each distribution $P \in \mathcal{P}$, denote function $\theta(P) \in \Psi$ is the true estimate of interests. Suppose $\{\mathbf{x}^1, \mathbf{x}^2, ..., \mathbf{x}^n\}$ are n i.i.d. observations that are drawn according to some distribution $P \in \mathcal{P}$, and $\hat{\theta} : \mathcal{X}^m \mapsto \Psi$ is the estimation of $\theta(P)$. The minimax risk $\mathcal{M}_n$ under metric $\Phi \circ \rho$ can be defined as the following saddle point problem:

$$\mathcal{M}_n(\theta(\mathcal{P}), \Phi \circ \rho) := \inf_{\hat{\theta}} \sup_{P \in \mathcal{P}} \mathbb{E}_P[\Phi(\rho(\hat{\theta}(\mathbf{x}^1, \mathbf{x}^2, ..., \mathbf{x}^n), \theta(P)))],$$

where $\rho : \Psi \times \Psi \mapsto \mathbb{R}^+$ is a semi-metric function, and $\Phi : \mathbb{R}_+ \mapsto \mathbb{R}_+$ is a non-decreasing function with $\Phi(0) = 0$.

For simplicity, our analysis will focus on the squared error case when ρ is the absolute operator and Φ is the squared summation operator, that is, $\Phi \circ \rho = ||\cdot||_2^2$.

3.2 Local private minimax risks

We now move to the definition of local private minimax risk, which measures the fundamental hardness of an estimation problem under ϵ-LDP.

Given the privacy budget ϵ, we denote $\mathcal{K}_\epsilon$ as the set of all possible mechanisms $\mathbf{K} = \{K^1, ..., K^n\}$ that satisfy interactive ϵ-LDP. Takes as input samples $\{\mathbf{x}^j\}_{j=1}^n$, some mechanism $\mathbf{K} \in \mathcal{K}_\epsilon$ can produce a sequence of private observations $\{\mathbf{z}^j\}_{j=1}^n$. If the estimator $\hat{\theta} = \hat{\theta}(\mathbf{z}^1, ..., \mathbf{z}^n)$ only depends on these private views $\{\mathbf{z}^j\}_{j=1}^n$ while having no access to input samples $\{\mathbf{x}^j\}_{j=1}^n$, the minimax risk as a function of privacy budget ϵ can be defined as:

$$\mathcal{M}_n(\theta(\mathcal{P}), \Phi \circ \rho, \epsilon) := \inf_{\mathbf{K} \in \mathcal{K}_\epsilon} \inf_{\hat{\theta}} \sup_{P \in \mathcal{P}} \mathbb{E}_{p, \mathbf{K}}[\Phi(\rho(\hat{\theta}(\mathbf{z}^1, ..., \mathbf{z}^n), \theta(P)))].$$

In the following sections, we first theoretically give minimax lower bounds for ϵ-LDP distribution estimation on set-valued data (in Section 3.3), and then design attainable mechanisms $\mathbf{K}$(i.e., the wheel mechanism, shown in Section 4 and 5) with efficiency considerations.

3.3 Lower bounds for ϵ-LDP set-valued data

Motivated by the bounding procedure for ϵ-LDP categorical distribution estimation in [10], we utilize the Assouad's method [39] and its local private form (Lemma 1, [11]) to derive sharp lower bound of minimax risks for set-valued data distribution estimation under the interactive ϵ-LDP.

Assouard's method [39] transforms an estimation problem into multiple binary hypothesis testing problems. For some $d \in \mathbb{N}$, it defines a hypercube $\mathcal{V} = \{-1, 1\}^d$ and a family of distributions $\{P_\nu\}_{\nu \in \mathcal{V}}$ indexed by the hypercube. Assouard's method says that the distribution family induces a 2δ-Hamming separation for the loss $\Phi \circ \rho$ if there exists a vertex mapping (a function $\kappa : \theta(\mathcal{P}) \mapsto \{-1, 1\}^d$) satisfying:

$$\Phi(\rho(\theta, \theta(P_\nu))) \geq 2\delta \sum_{j=1}^{d} \mathbf{1}\{[\kappa(\theta)]_j \neq \nu_j\}.$$

Considering the nature first uniform randomly chooses a vector $V \in \{-1, 1\}^d$, after which the samples $\{\mathbf{x}^1, ..., \mathbf{x}^n\}$ are drawn from the distribution p_ν conditioned on $V = \nu$. These samples are later taken as input into any interactive ϵ-LDP mechanisms $\mathbf{K}$.

The work [11] gives the following private version of Assouad's method for lower bounding ϵ-LDP estimation problems.

Lemma 1 (Private Assouad bound [11]). *Let* $P_{+j} = \frac{1}{2^{d-1}} \sum_{\nu : \nu_j = 1} P_\nu$ *and* $P_{-j} = \frac{1}{2^{d-1}} \sum_{\nu : \nu_j = -1} P_\nu$, *we have*

$$\mathcal{M}_n(\theta(\mathcal{P}), \Phi \circ \rho) \geq d \cdot \delta [1 - (\frac{n(e^\epsilon - 1)^2}{2d} F_{\mathbb{B}_\infty(\mathcal{X}^m), \mathcal{P}})^{\frac{1}{2}}],$$

where $\mathbb{B}_\infty(\mathcal{X}^m)$ *denote the collection of function* γ *with supremum norm bounded by 1 as:*

$$\mathbb{B}_\infty(\mathcal{X}^m) := \{\gamma : \mathcal{X}^m \mapsto \mathbb{R} \mid ||\gamma||_\infty \leq 1\},$$

and maximum possible discrepancy $F_{\mathbb{B}_\infty(\mathcal{X}^m), \mathcal{P}}$ *is defined as:*

$$\sup_{\gamma \in \mathbb{B}_\infty(\mathcal{X}^m)} \sum_{i=1}^{d} (\int_{\mathcal{X}^m} \gamma(x)(dP_{+j}(x) - dP_{-j}(x)))^2.$$

Rely on Lemma 1, we can construct a class of $\frac{2\delta^2 m^2}{d^2}$-hamming separated distributions and bound the maximum possible marginal discrepancy $F_{\mathbb{B}_\infty(\mathcal{X}^m), \mathcal{P}}$ under $\frac{8\delta^2 m}{d}$, hence give lower bounds (in Theorem 1) for the problem of local private set-valued data distribution estimation.

Theorem 1. *For the set-valued data distribution estimation problem where* $m \leq \frac{d}{2}$, *for any non-interactive or interactive* ϵ*-LDP mechanism, there exists a universal constant* $c > 0$ *such that for all* $\epsilon \in [0, 1]$,

$$\mathcal{M}_n(\theta(\mathcal{P}), ||\cdot||_2^2, \epsilon) \geq c \cdot \min\{\frac{m^2}{d}, \frac{dm}{n\epsilon^2}\}$$

Proof. According to Lemma 1, in order to derive a good lower bound, we need to construct a well hamming-separated class of distributions, and simultaneously bound the maximum possible marginal discrepancy to be small. Our proof contains following 4 steps:

1. Constrain set-valued data to decomposable cases. Following analyzing procedures will utilize this decomposability in set-valued data to simplify analyses to multiple categorical cases. Specifically, we assume that

the domain size d is divisible by the cardinality m, that is, we can define an integer value $l := \frac{d}{m}$. We also assume that l is even ($l \geq 2$). We can then separate d items to m buckets, and the a-th bucket is $B_a = \{\mathbf{x}_{a\cdot l+1}, ..., \mathbf{x}_{a\cdot l+l}\}$ $(0 \leq a \leq m-1)$. As a special form of set-valued data, we consider cases when each bucket B_a has exactly 1 item and call such set-valued data a decomposable one. Such decomposability, along with independence among buckets, allows us to simplify the proof.

2. Construct 2δ-Hamming separation distributions. Follow standard procedure of Assuoad method, we set $\mathcal{V}_a = \{-1, 1\}^{\frac{l}{2}}$ for each bucket B_a ($a \in [0, m-1]$) and define hypercube as $\mathcal{V} = \Pi_{a=1}^{m} \mathcal{V}_a$. Fixing $\delta \in [0, 1]$, for $\nu \in \mathcal{V}$, separately consider each bucket, we define θ_{ν_a} be the multinomial distribution for bucket B_a as :

$$\theta_{\nu_a} := \frac{m}{d}\mathbf{1} + \delta\frac{m}{d}\begin{bmatrix}\mathcal{V}_a \\ -\mathcal{V}_a\end{bmatrix}.$$

Assuming independence among buckets, we then define the probability distribution over the universe $\mathcal{X}^m$ as a product distribution $\Pi_{a=1}^{m}\theta_{\nu_a}$. The item distribution is hence $\theta_\nu = [\theta_{\nu_1}\ \theta_{\nu_2}\ ...\ \theta_{\nu_m}]$. For any estimator $\hat{\theta} = [\hat{\theta}_{\nu_1}\ \hat{\theta}_{\nu_2}\ ...\ \hat{\theta}_{\nu_m}]$, by defining $\hat{\nu}_a = sign(\hat{\theta}_{\nu_a} - \frac{m}{d}\mathbf{1})$ for $a \in [0, m-1]$, we have lower bound on separation:

$$||\hat{\theta} - \theta_\nu||_2^2 \geq \frac{\delta^2 m^2}{d^2}\sum_{j=1}^{l/2}\sum_{a=1}^{m}\mathbf{1}\{\hat{\nu}_{a_j} \neq \nu_{a_j}\}.$$

3. Bound maximum discrepancy of induced marginal distributions. We now turn to bounding sums of integrals $\int_{\mathcal{X}^m}\gamma(x)(dP_{+j}(x) - dP_{-j}(x))$, and claim following inequality:

$$\sup_{\gamma\in\mathbb{B}_\infty(\mathcal{X}^m)}\sum_{i=1}^{d}(\int_{\mathcal{X}^m}\gamma(x)(dP_{+j}(x) - dP_{-j}(x)))^2 \leq \frac{8\delta^2 m}{d}.$$

Actually, by construction P_{+j} as a joint distribution $\Pi_{a=1}^{m}[\frac{m}{d}\mathbf{1} + \frac{m\delta}{d}[e_{j\%l}^\top - e_{j\%l}^\top]^\top\{\lfloor j/l \rfloor = a\}] \in \Delta_l$ and similarly for P_{-j}, where $e_j \in \{0, 1\}^{l/2}$ denote the j-th standard basis vector. Due to the interleaving structure of the m-dimensional distribution P_{+j} and P_{+j}, for any $\gamma \in [-1, 1]^d$, we have:

$$\sum_{a=1}^{m}\sum_{j=1}^{l/2}(\int_{\mathcal{X}^m}\gamma(x)(dP_{+j}(x) - dP_{-j}(x)))^2 \leq \frac{8\delta^2 m}{d},$$

that is, assigning $\gamma \in [-1, 1]^d$ according to one of the dimension maximizes the overall integral.

4. Bound minimax risks. Applying Lemma 1 and substitute the hamming separation parameter δ as $\frac{\delta^2 m^2}{d^2}$, we have:

$$\max_{\nu\in\mathcal{V}}\mathbb{E}_{P_\nu}[||\hat{\theta} - \theta_\nu||_2^2] \geq \frac{\delta^2 m^2}{d}[1 - (4n(e^\epsilon - 1)^2\delta^2 m/d)^{\frac{1}{2}}].$$

By choosing the parameter δ^2 at $\min\{1, d^2/(16n(e^\epsilon - 1)^2 m)\}$ or $\min\{1/m, d^2/(16n(e^\epsilon - 1)^2 m^2)\}$, we have the lower bound of:

$$\mathcal{M}_n(\theta(\mathcal{P}), ||\cdot||_2^2, \epsilon) \geq \min\{\frac{m^2}{4d}, \frac{dm}{64n(e^\epsilon - 1)^2}\}.$$

□

For understanding the minimax rate, let us consider the non-private error rate of set-valued data distribution estimation, where we have:

$$\mathbb{E}||\hat{\theta} - \theta||_2^2 \leq \sum_{i=1}^{d}\mathbb{E}||\hat{\theta}_i - \theta_i||_2^2 \leq \frac{m}{n}.$$

Hence for estimation with ℓ_2-norm error, the enforcement of ϵ-LDP causes the effective sample size decreases from n to $n\epsilon^2/d$. Compared with the ϵ-LDP lower bound $O(\frac{d}{n\epsilon^2})$ for categorical data with domain size d, the mean squared error lower bound of set-valued data with cardinality m is scaled by a factor of m. This is hinted by the fact that under the same domain size, the squared ℓ_2-norm of an m-sized set-valued data is m times of a 1-sized set-valued data's.

4. CATEGORICAL DATA

We present an attainable mechanism (i.e., the wheel mechanism) with optimal error bounds. It is highly efficient in terms of computation/communication costs on the user side. Here, we first elaborate it on categorical data (i.e., a special case of set-valued data that $m \equiv 1$).

4.1 User-side randomization

From the view of Shannon entropy, existing ϵ-LDP mechanisms that output one item [23] or one item set [34, 38] need at least $O(\log(Poly(d)))$ bits when the privacy budget is small (i.e., ϵ approaches 0), because the output approximates uniform random. However, intuitively, the output is allowed to contain less information about the input as the privacy budget decreases, and hence we should be able to convey/transmit the information about the input with fewer bits. Therefore, we propose a novel wheel mechanism, which replaces some randomness in the output with pseudo-randomness by a user-specific hash function and hence reduces bits for communication. Besides, the wheel mechanism uses a coverage parameter $p \in (0.0, 0.5)$ to tweak true/false coverage probability (in Definition 3 and 4) of an item, which is a continuous analogy to the standard discrete true/false positive rate tuning method [34, 38].

Specifically, the wheel mechanism for categorical data (in Algorithm 1) proceeds with the following three steps, where the coverage parameter $p \in (0.0, 0.5)$ is the length of coverage area to control true/false coverage probabilities.

1. Use the user id or a random-generated number as the seed, the user-specific hash function maps user's item x to a numerical value v in the range $[0.0, 1.0)$.

2. To satisfy ϵ-LDP requirement, the numerical value v is then randomized with a calibrated probability distribution Q over $[0.0, 1.0)$. The definition of Q with coverage parameter p is given as follows ($0.0 \leq y, v < 1.0$):

$$Q[y|v] = \begin{cases}\frac{e^\epsilon}{p\cdot e^\epsilon + (1-p)}, & \text{for } v \leq y < v + p \\ & \text{or } 0 \leq y < v + p - 1; \\ \frac{1}{p\cdot e^\epsilon + (1-p)}. & \text{for } y \geq v + p \\ & \text{or } y < v.\end{cases} \quad (2)$$

Figure 2 demonstrates the probability distribution on a wheel, the heavy-weight part of the distribution lies in the clockwise area begins at v with length p.

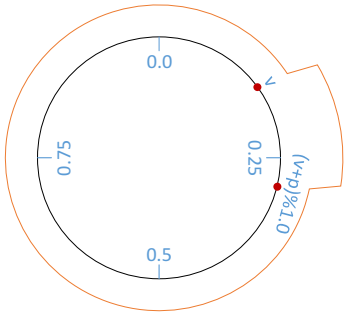


Figure 2: Probability distribution of the private view y demonstrated over a circled wheel. The coverage area begins at v clock-wisely with the length of p. The probability mass in the coverage area is $\frac{e^\epsilon}{p\cdot e^\epsilon+(1-p)}$, while the probability mass of the rest of the area is $\frac{1}{p\cdot e^\epsilon+(1-p)}$.

3. Sample a value $z \in [0.0, 1.0)$ from distribution $Q[y|v]$, then send it to the server (along with the seed if necessary).

Algorithm 1 Categorical Data Randomization

Input: Categorical data $\{x\} \in \mathcal{X}^1 = \{c \mid c \subseteq \mathcal{X} \text{ and } |c| = 1\}$, hash function $h : \mathcal{X} \to [0.0, 1.0)$ with random seed s, privacy level ϵ, the coverage parameter $p = \frac{1}{e^\epsilon+1}$.
Output: A private view $z \in [0.0, 1.0)$ that satisfies ϵ-LDP.

1: ▷ User-specific hash mapping
2: $v = h_s(x)$
3: ▷ Randomization according to probability distribution (2)
4: r = UniformRandom$(0.0, 1.0)$
5: **if** $r < \frac{p\cdot e^\epsilon}{p\cdot e^\epsilon+(1-p)}$ **then**
6: $\quad z = v + r \cdot \frac{p\cdot e^\epsilon+(1-p)}{e^\epsilon}$
7: **else**
8: $\quad z = v + p + r \cdot (p \cdot e^\epsilon + (1-p)) - p \cdot e^\epsilon$
9: **end if**
10: $z = z \bmod 1.0$
11: **return** (s, z)

The ϵ-LDP guarantee of the wheel mechanism is given in Theorem 2.

Theorem 2. *The wheel mechanism in Algorithm 1 satisfies ϵ-LDP.*

Proof. To prove ϵ-LDP constraints hold for any pair of items $\{x\}, \{x'\} \in \mathcal{X}^1$, it's enough to show that $Q(z|v) \le Q(z|v') \cdot e^\epsilon$ holds for any pair of numerical values $v, v' \in [0.0, 1.0)$ and for any output $z \in [0.0, 1.0)$. According to the design of probability distribution of $Q(z|v)$, both $Q(z|v)$ and $Q(z|v')$ are either $\frac{e^\epsilon}{p\cdot e^\epsilon+(1-p)}$ or $\frac{1}{p\cdot e^\epsilon+(1-p)}$, hence $Q(z|v) \le Q(z|v')\cdot e^\epsilon$ holds and the wheel mechanism satisfies ϵ-LDP. □

The computation complexity of Algorithm 1 is $O(1)$. The communication costs between a user and the server of Algorithm 1 is $O(\log(1/p)) = O(\log(e^\epsilon + 1))$, hence one 32-bit or 64-bit float number or even fewer bits are capable of storing the value of z with satisfactory precision when $\epsilon < +\infty$.

4.2 Server-side distribution estimation

It can be observed from Figure 2: when the mapped value of categorical X_i is v, z appears in the coverage area $[v, v+p)$ or $[0, v+p-1)$ with a relatively high probability; when the true category is not X_i, the probability that z appears in the coverage area $[v, v+p)$ or $[0, v+p-1)$ is relatively low. This implies the server side could estimate the frequency/distribution of each item according to the value of z.

Formally, if we denote the mapped value of a category X_i as v_i, and the coverage area $[v_i, v_i+p)$ or $[0, v_i+p-1)$ as C_{v_i}, we can define the true/false coverage probability in Definitions 3 and 4 respectively. The true coverage probability P_t is always greater than the false coverage probability P_f when $p < 0.5$, which further confirms previous observations.

Definition 3 (True Coverage Probability). *When the true category is X_i, the probability that the private view $z \in [0.0, 1.0)$ of X_i shows in the coverage area C_{v_i} is:*

$$\mathrm{P}_t = \mathbb{P}[z \in C_{v_i} | x = \{X_i\}] = \frac{p \cdot e^\epsilon}{p \cdot e^\epsilon + (1-p)}.$$

Definition 4 (False Coverage Probability). *When the true category is $X_{i'}$ ($i' \neq i$), assume the hash function h_s is perfect, namely $h(X_{i'})$ is uniformly random and is independent from $h(X_i)$, then the expected probability that the private view $z \in [0.0, 1.0)$ of $X_{i'}$ shows in the coverage area of X_i is:*

$$\mathrm{P}_f = \mathbb{E}[\mathbb{P}[z \in C_{v_i} | x = \{X_{i'}\}]] = p.$$

Based on the true/false coverage probabilities, let $\mathbb{P}[z \in C_{v_i}]$ denote the probability that z shows in the coverage area of X_i, we have:

$$\mathbb{P}[z \in C_{v_i}] = \theta_i \cdot \mathrm{P}_t + (1.0 - \theta_i) \cdot \mathrm{P}_f. \tag{3}$$

Consequently, given that the frequency $\frac{F_i}{n} = \frac{\sum_{j=1}^n \{z^j \in C_{v_i}\}}{n}$ is an unbiased estimation of $\mathbb{P}[z \in C_{v_i}]$, a linear transformation of $\frac{F_i}{n}$ provide an estimator of item distribution (shown in Algorithm 2). In the algorithm, lines 1 to 10 records the number of z falling in the coverage area of item X_i for each item, while lines 12 to 14 recovers unbiased estimation of item distribution θ_i from recorded numbers according to the formula (3). The computation complexity of Algorithm 2 is $O(d)$.

We theoretically prove that the estimator (i.e., Algorithm 2) is unbiased according to Lemma 2.

Lemma 2. *The estimator $\hat{\theta}$ in Algorithm 2 is an unbiased estimation of the true item distribution θ, where the expectation is taken over the (pseudo) randomness of perfect hash functions and the randomness of sampling in Algorithm 2.*

Proof. Proving $\hat{\theta}$ is an unbiased estimation of θ is equivalent to prove that $\hat{\theta}_i$ is an unbiased estimation of θ_i. Because there are $n \cdot \theta_i$ users hold category X_i and the other $n - n \cdot \theta_i$ users hold a category that is not X_i, the observed frequency F_i is the summation of $n \cdot \theta_i$ Bernoulli variables with success probability P_t and $n - n \cdot \theta_i$ Bernoulli variables

Algorithm 2 Categorical Data Distribution Estimation

Input: Private views and random seeds $(S, Z) = [\ (s^1, z^1), (s^2, z^2), ..., (s^n, z^n)\ ]$ from n users.
Output: An unbiased estimator $\hat{\theta}$ of the true item distribution $\theta = \{\theta_1, \theta_2, ..., \theta_d\}$.

$F = \{0\}^d$
▷ Record frequencies of items covered by z.
for $(s^j, z^j) \in (S, Z)$ **do**
  for $X_i \in \mathcal{X}$ **do**
    $v_i^j = h_{s^j}(X_i)$
    if $z^j - q < v_i^j \leq z^j$ or $z^j - q + 1 < v_i^j < 1$ **then**
      $F_i = F_i + 1$
    end if
  end for
end for
▷ Estimate item distribution from observed frequencies.
for $i = 1$ to d **do**
  $\hat{\theta}_i = \frac{1}{n} \cdot \frac{F_i - n \cdot \mathrm{P}_f}{\mathrm{P}_t - \mathrm{P}_f}$
end for
return $\hat{\theta}$

that their expected success probability is P_f. Consequently, we have:

$$\begin{aligned}
&\mathbb{E}[\frac{F_i - n \cdot \mathrm{P}_f}{\mathrm{P}_t - \mathrm{P}_f}] \\
=&\frac{\sum_{j \in [1,n]} [\mathbf{x}^j = \{X_i\}] \cdot P_t + [\mathbf{x}^j \neq \{X_i\}] \cdot P_f - n \cdot \mathrm{P}_f}{\mathrm{P}_t - \mathrm{P}_f} \\
=&\frac{n \cdot \theta_i \cdot \mathrm{P}_t + (n - n \cdot \theta_i) \cdot \mathrm{P}_f - n \cdot \mathrm{P}_f}{\mathrm{P}_t - \mathrm{P}_f}, \\
=&\frac{n \cdot \theta_i \cdot \mathrm{P}_t - n \cdot \theta_i \cdot \mathrm{P}_f}{\mathrm{P}_t - \mathrm{P}_f}, \\
=&n \cdot \theta_i,
\end{aligned}$$

which concludes that $\frac{1}{n} \cdot \frac{F_i - n \cdot \mathrm{P}_f}{\mathrm{P}_t - \mathrm{P}_f}$ is an unbiased estimation of the item distribution's bucket θ_i. □

It's worth noting that the wheel mechanism's randomization procedure doesn't need any information about the item domain $\mathcal{X}$ or the number of items d, hence it is domain-independent. Therefore, the server could derive the distribution of target items on demand, without knowing the details about the whole domain. It also works even when the domain changes over time.

4.3 Optimal utility guarantees

The asymptotic maximum absolute error of the distribution estimator given in Algorithm 2 is $O(\sqrt{\frac{\log(d/\beta)}{\epsilon^2 n}})$ with high probability (See Theorem 3 for detail), which matches the lower error bound of ϵ-LDP distribution estimation of categorical data [2].

Theorem 3 (Maximum Absolute Error). *With privacy budget $\epsilon = O(1)$, for any categorical data domain $\mathcal{X}$ that $|\mathcal{X}| = d$, and any $\beta > 0$, the error due to Algorithm 2 is bounded by*

$$\max_{i \in [1,d]} |\hat{\theta}_i - \theta_i| = O(\sqrt{\frac{\log(d/\beta)}{\epsilon^2 n}})$$

with probability $1 - \beta$ over the randomness of the user-specific hash functions and the randomization in Algorithm 1.

Proof. Consider the i-th element θ_i of the distribution, according to Lemma 2, the expectation of $\hat{\theta}_i - \theta_i$ is hence 0. Furthermore $\hat{\theta}_i - \theta_i$ is the summation of n independent random variables, every of which lies in the range of:

$$[\min\{\frac{0 - \mathrm{P}_f}{\mathrm{P}_t - \mathrm{P}_f}, \frac{0 - \mathrm{P}_t}{\mathrm{P}_t - \mathrm{P}_f}\}, \max\{\frac{1 - \mathrm{P}_f}{\mathrm{P}_t - \mathrm{P}_f}, \frac{1 - \mathrm{P}_t}{\mathrm{P}_t - \mathrm{P}_f}\}].$$

Apply the coverage parameter p at $\frac{1}{e^\epsilon + 1}$, the range is then:

$$[\min\{\frac{-2}{e^\epsilon - 1}, \frac{-1 - e^\epsilon}{e^\epsilon - 1}, \max\{\frac{2e^\epsilon}{e^\epsilon - 1}, \frac{1 + e^\epsilon}{e^\epsilon - 1}\}]$$

and could be written as $[O(\frac{1}{\epsilon}), O(\frac{1}{\epsilon})]$. Therefore, according to the Hoeffding's Inequality on strictly bounded random variables, we have the probability $\mathrm{P}[|\hat{\theta}_i - \theta_i| \leq \rho] \geq 1 - \exp(-\frac{2n^2\rho^2}{n \cdot O(1/\epsilon^2)})$. Replace $1 - \exp(-\frac{2n^2\rho^2}{n \cdot O(1/\epsilon^2)})$ with $1 - \beta$, we have $|\hat{\theta}_i - \theta_i| \leq \rho \leq O(\sqrt{\frac{\log(1/\beta)}{2n\epsilon^2}})$. Further replacing β with β/d, then with probability $1 - \beta/d$, the absolute error of each bucket $|\hat{\theta}_i - \theta_i|$ is bounded by $O(\sqrt{\frac{\log(d/\beta)}{\epsilon^2 n}})$. Finally apply the union bound of probabilities to the maximum absolute error $\max_{i \in [1,d]} |\hat{\theta}_i - \theta_i|$, we have the theorem proved. □

To further guarantee the performance of the wheel mechanism, we here give exact mean squared error bound on distribution estimation in Theorem 4. When the privacy budget is $\epsilon = O(1)$, the error bound matches the lower bound in Theorem 1 with $m = 1$. More generally, for any privacy budget ϵ in the practical regime (e.g., when $e^\epsilon \ll d$), the error bound matches the estimation lower bound $\Theta(\frac{e^\epsilon d}{n(e^\epsilon - 1)^2})$ given by [38]. Thus we can conclude that the wheel mechanism is optimal for categorical data distribution estimation.

Theorem 4 (Mean Squared Error Bound). *For any categorical data domain $\mathcal{X}$ that $|\mathcal{X}| = d$, the error due to Algorithm 2 is bounded by*

$$\sum_{i \in [1,d]} \mathbb{E}[|\hat{\theta}_i - \theta_i|^2] \leq \frac{1}{n} + \frac{4e^\epsilon d}{n(e^\epsilon - 1)^2},$$

the expectation is taken over the randomness of user-specific hash functions and the randomization in Algorithm 1.

Proof. Firstly we write the mean squared error of distribution estimation as a formulation of P_t and P_f. Recall that the observed frequency F_i in Algorithm 2 is the summation of $n \cdot \theta_i$ Bernoulli variables with success probability P_t and $n - n \cdot \theta_i$ Bernoulli variables that their expected success probability is P_f. Further apply Lemma 2 and $\sum_{i \in [1,d]} \theta_i = 1$,

then we have:

$$
\begin{aligned}
&\sum_{i\in[1,d]} \mathbb{E}[|\hat{\theta}_i - \theta_i|^2] \\
=&\sum_{i\in[1,d]} \mathbb{E}[|\hat{\theta}_i - \mathbb{E}[\hat{\theta}_i]|^2] = \mathrm{Var}(\hat{\theta}_i) \\
=&\sum_{i\in[1,d]} \mathrm{Var}(\frac{1}{n}\cdot\frac{F_i - n\cdot \mathrm{P}_f}{\mathrm{P}_t - \mathrm{P}_f}) \\
=&\frac{1}{n^2\cdot(\mathrm{P}_t-\mathrm{P}_f)^2}\sum_{i\in[1,d]}\mathrm{Var}(F_i) \\
=&\sum_{i\in[1,d]}\frac{n\cdot\theta_i\mathrm{P}_t(1-\mathrm{P}_t)+(n-n\cdot\theta_i)\mathrm{P}_f(1-\mathrm{P}_f)}{n^2\cdot(\mathrm{P}_t-\mathrm{P}_f)^2}. \\
=&\frac{(\sum_{i\in[1,d]}\theta_i)\mathrm{P}_t(1-\mathrm{P}_t)+(d-\sum_{i\in[1,d]}\theta_i)\mathrm{P}_f(1-\mathrm{P}_f)}{n\cdot(\mathrm{P}_t-\mathrm{P}_f)^2}. \\
=&\frac{\mathrm{P}_t(1-\mathrm{P}_t)+(d-1)\mathrm{P}_f(1-\mathrm{P}_f)}{n(\mathrm{P}_t-\mathrm{P}_f)^2}.
\end{aligned}
$$

Apply the coverage parameter $p = \frac{1}{1+e^\epsilon}$, we have $\mathrm{P}_t = \frac{1}{2}$ and $\mathrm{P}_f = p$, hence the total variance error is then:

$$
\begin{aligned}
\sum_{i\in[1,d]}\mathbb{E}[|\hat{\theta}_i-\theta_i|^2] &= \frac{e^{2\epsilon}+e^\epsilon(4d-2)+1}{(e^\epsilon-1)^2} \\
&= \frac{1}{n}+\frac{4e^\epsilon d}{n(e^\epsilon-1)^2},
\end{aligned}
$$

which gives the final error bound.

Actually, a slightly better choice of p could be deduced by directly minimize the former equation, the differential function of which in terms of p is quadratic. Though we can derive closed-formed optimal p^*, its form is quite complicated. Here $p = \frac{1}{1+e^\epsilon}$ is near to the p^* when privacy budget is not too large. □

When considering the Bloom filter size l as a dynamic parameter in the RAPPOR [13], a.k.a. the O-RAPPOR mechanism for open alphabet in the work of [23], the inverse of the Bloom filter size: $1/l$ is an analogy to the coverage parameter p in the wheel mechanism. However, the works of [13, 23] provide no theoretical guidance/performance guarantee on choosing l. Essentially, the wheel mechanism is also like a continuous analogy of the k-Subset mechanism in [34, 38].

5. SET-VALUED DATA

We here extend the design of the wheel mechanism to set-valued data with m items ($m > 1$) and prove its optimality for item distribution given the minimax lower bounds in Theorem 1. The mechanism is similar to the one for categorical data, except that extra post-processing is needed when items' coverage areas are overlapping. Hence, it still has the advantage of being domain-independent and highly efficient on the user side.

5.1 User-side randomization

Suppose each user u^j holds set-valued data $\mathbf{x}^j = \{X_1, ..., X_m\}$, the wheel mechanism for $\mathbf{x}^j$ follows similar steps as for categorical data, except using a user-specific hash function $h : \mathcal{X} \to [0.0, 1.0)$ to map every item $X_i \in \mathbf{x}^j$ to v_i.

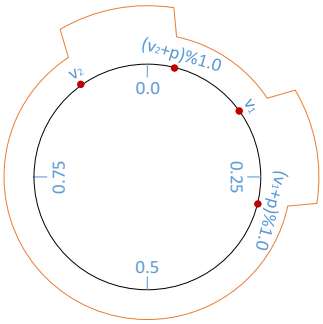


Figure 3: Probability distribution of the private view y demonstrated over a circled wheel when coverage areas are disjoint. The union coverage area begins at every v_i with the clockwise length of p. The probability mass in the coverage area is $\frac{e^\epsilon}{m\cdot p\cdot e^\epsilon+(1-m\cdot p)}$, while the probability mass of the rest area is $\frac{1}{m\cdot p\cdot e^\epsilon+(1-m\cdot p)}$.

The randomization distribution $Q[y|\mathbf{v} = \{v_1, .., v_m\}]$ is now defined as follows:

$$
Q[y|\mathbf{v}] = \begin{cases} \frac{e^\epsilon}{\Omega}, & \text{if } y \in [v_i, v_i+p) \text{ or } [0, v_i+p-1) \\ & \text{for any } i \in [1,m]; \\ \frac{\Omega - l\cdot e^\epsilon}{(1-l)\Omega}, & \text{otherwise.} \end{cases} \tag{4}
$$

where $\Omega := m\cdot p\cdot e^\epsilon + (1-m\cdot p)$ is called normalization factor, and l is the length (measure) of the union coverage area $C_\mathbf{v}$ ($y \in [0, 1.0)$):

$$
C_\mathbf{v} = \{y \mid \text{if } y \in [v_i, v_i+p) \text{ or } [0, v_i+p-1) \text{ for any } i \in [1,m]\}.
$$

Figure 3 and Figure 4 show examples that the probability distribution over the wheel with disjoint/overlapping coverage areas respectively.

Practically, by sorting neighboring areas [15] firstly, the union of coverage areas $C_\mathbf{v}$ can be merged in $\Theta(m \log m)$ time. Subsequent computing of the length l and drawing a uniform sample from the disjoint coverage ranges will further cost $O(m)$ time. Hence the total computation complexity of the naive wheel mechanism implementation for set-valued data is $\Theta(m \log m)$.

Alternatively, considering that the coverage range of each C_{v_i} is of the same length p on the wheel, we can thus reduce the complexity to $\Theta(\lceil\frac{1}{p}\rceil) = \Theta(me^\epsilon)$ by dividing $[0.0, 1.0)$ into $\lceil\frac{1}{p}\rceil$ buckets (as shown in Algorithm 3). For the b-th bucket, we record the current start/end point of coverage areas in the bucket as $left_b/right_b$ respectively. At first, the start point is the maximum value in the bucket and the end point is the minimum value in the bucket, which implies the bucket is empty (with no coverage area). When a new coverage area C_{v_i} comes, the start/end point of two corresponding buckets will be updated. Finally we merge continuous areas based on start/end points of neighboring buckets.

The whole algorithm has following four steps:

1. Divide $[0.0, 1.0)$ to $\lceil\frac{1}{p}\rceil$ buckets, each of which holds one part with length of p except the last one. Use $left_b/right_b$ to denote the start/end point of the b-th bucket respectively.

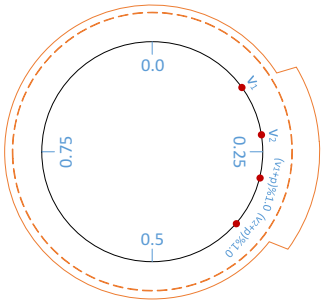


Figure 4: Probability distribution of the private view y demonstrated over a circled wheel when coverage areas are overlapping. The probability mass in the coverage area is $\frac{e^\epsilon}{m\cdot p\cdot e^\epsilon+(1-m\cdot p)}$, while the probability mass of the rest area is $\frac{\Omega-l\cdot e^\epsilon}{(1-l)\Omega}$. The dashed circle represents the mass of $\frac{1}{m\cdot p\cdot e^\epsilon+(1-m\cdot p)}$, which is lower than $\frac{\Omega-l\cdot e^\epsilon}{(1-l)\Omega}$.

2. For each p-length coverage range C_{v_i}, record the start/end points of corresponding neighboring buckets (see lines 6 to 14).
3. For each bucket, merge ranges when the end point is lower than the start point, and then split the covered areas into disjoint ranges (see lines 15 to 21).
4. Given disjoint covered ranges and their relative weights against non-covered ranges (computed using the union coverage length l), draw a random sample as the output.

The ϵ-LDP guarantee of the randomization process for set-valued data is given in Theorem 5.

Theorem 5. *The wheel mechanism for set-valued data in Algorithm 3 satisfies ϵ-LDP.*

Proof. The output of Algorithm 3 follows the probability distribution in Equation (4), hence to prove its ϵ-LDP guarantee, it's enough to show that $\frac{1}{\Omega} \geq \frac{\Omega-l\cdot e^\epsilon}{(1-l)\Omega} \geq \frac{e^{2\epsilon}}{\Omega}$. Given that $l \leq m\cdot p$, for any $\epsilon \geq 0.0$, we have $\frac{\Omega-l\cdot e^\epsilon}{(1-l)\Omega} \geq \frac{e^\epsilon}{\Omega}$, and obviously $1-l+l\cdot e^\epsilon \leq 1-m\cdot p+m\cdot p\cdot e^\epsilon$, which further implies $\frac{1}{\Omega} \geq \frac{\Omega-l\cdot e^\epsilon}{(1-l)\Omega}$. □

5.2 Server-side distribution estimation

Similar to the categorical data case, we now define true/false coverage probability in Definitions 5 and 6 respectively for the wheel mechanism of set-valued data.

Definition 5 (True Coverage Probability). *When the category X_i in the set-valued data $\mathbf{x}$, the probability that the private view $z \in [0.0, 1.0)$ of X_i shows in the coverage area C_{v_i} is:*

$$\mathrm{P}_t = Q(z \in C_{v_i} | X_i \in \mathbf{x}) = \frac{p\cdot e^\epsilon}{m\cdot p\cdot e^\epsilon + (1-m\cdot p)}.$$

Definition 6 (False Coverage Probability). *When the X_i is truly included in the set-valued data. When the true category is not X_i, assume the hash function h_s is perfect, the expected probability that the private view $z \in [0.0, 1.0)$ of $X_{i'}$ $(i' \neq i)$ shows in the coverage area of X_i is:*

$$\mathrm{P}_f = \mathbb{E}[Q(z \in C_{v_i} | X_{i'} \notin x)] = p.$$

Given the true/false coverage probabilities of items, the estimator of the item distribution for set-valued data is the same as Algorithm 2. The estimator is unbiased according to Theorem 2, and its computation complexity is $O(n\cdot d)$.

5.3 Optimal utility guarantees

We here give mean squared error bounds on distribution estimation of the wheel mechanism (shown in Theorem 6). Note that the error bound proved here is $O(\frac{md}{n\epsilon^2})$, which significantly improves state-of-art results (i.e., $\Theta(\frac{m^2 d}{n\epsilon^2})$ in [27]) by a factor of $\Theta(m)$.

Theorem 6 (Mean Squared Error Bound). *With privacy budget $\epsilon = O(1)$, for any set-valued data domain $\mathcal{X}^m$ that $|\mathcal{X}| = d$, the distribution estimation error due to Algorithms 3 and 2 is bounded by:*

$$\sum_{i\in[1,d]} \mathbb{E}[|\hat{\theta}_i - \theta_i|^2] = O(\frac{md}{n\epsilon^2}),$$

where the expectation is taken over the randomness of user-specific hash functions and randomization in Algorithm 3.

Proof. We firstly give mean squared error bounds as a formulation of the true/false coverage probabilities. Recall that the observed frequency F_i in Algorithm 2 is the summation of $n\cdot\theta_i$ Bernoulli variables with success probability P_t and $n-n\cdot\theta_i$ Bernoulli variables that their expected success probability is P_f. Because $\hat{\theta}_i$ is an unbiased estimation of θ_i, and we have $\sum_{i\in[1,d]}\theta_i = m$ for set-valued data, then the total variance error is as follows:

$$\begin{aligned}
&\sum_{i\in[1,d]} \mathbb{E}[|\hat{\theta}_i - \theta_i|^2] \\
=&\mathrm{Var}(\hat{\theta}_i) = \sum_{i\in[1,d]} \mathrm{Var}(\frac{1}{n}\cdot\frac{F_i - n\cdot\mathrm{P}_f}{\mathrm{P}_t-\mathrm{P}_f}) \\
=&\frac{1}{n^2\cdot(\mathrm{P}_t-\mathrm{P}_f)^2}\sum_{i\in[1,d]}\mathrm{Var}(F_i) \\
=&\sum_{i\in[1,d]}\frac{n\cdot\theta_i\mathrm{P}_t(1-\mathrm{P}_t)+(n-n\cdot\theta_i)\mathrm{P}_f(1-\mathrm{P}_f)}{n^2\cdot(\mathrm{P}_t-\mathrm{P}_f)^2}. \\
=&\frac{(\sum_{i\in[1,d]}\theta_i)\mathrm{P}_t(1-\mathrm{P}_t)+(d-\sum_{i\in[1,d]}\theta_i)\mathrm{P}_f(1-\mathrm{P}_f)}{n\cdot(\mathrm{P}_t-\mathrm{P}_f)^2}. \\
=&\frac{m\cdot\mathrm{P}_t(1-\mathrm{P}_t)+(d-m)\mathrm{P}_f(1-\mathrm{P}_f)}{n(\mathrm{P}_t-\mathrm{P}_f)^2}.
\end{aligned}$$

Apply the coverage parameter $p = \frac{1}{2m-1+me^\epsilon}$, we have $\mathrm{P}_t = \frac{e^\epsilon}{m-1+2me^\epsilon}$ and $\mathrm{P}_f = p$, hence the total variance error is then:

$$\sum_{i\in[1,d]} \mathbb{E}[|\hat{\theta}_i - \theta_i|^2] = \Theta(\frac{d(3m-1)^2(3m-2)}{(e^\epsilon-1)^2(me^\epsilon+m-1)^2}).$$

Given that $e^\epsilon \approx \epsilon+1$ when $\epsilon = O(1)$, we finally have $\sum_{i\in[1,d]}\mathbb{E}[|\hat{\theta}_i-\theta_i|^2] = O(\frac{md}{n\epsilon^2})$. □

The error bound mentioned above is data-independent. It holds for any possible set-valued samples with any true item distribution θ. Given the claim in [10] that the population minimax rate (shown in Theorem 1) lower bounds conditional sample risk (shown in Theorem 6) up to some multiplicative constant, we can conclude that the wheel mechanism is minimax optimal. Furthermore, combining Equation (3.3) and above bounds, we conclude that the minimax error bounds in Theorem 1 are tight.

Algorithm 3 Set-valued Data Randomization

Input: Set-valued data $\mathbf{x} \in \mathcal{X}^m = \{c \mid c \subseteq \mathcal{X} \text{ and } |c| = m\}$, hash function $h : \mathcal{X} \to [0.0, 1.0)$ with random seed s, privacy level ϵ, the coverage parameter $p = \frac{1}{2m-1+me^\epsilon}$.
Output: A private view $z \in [0.0, 1.0)$ that satisfies ϵ-LDP.

$\triangleright$ User-specific hash mapping
$\mathbf{v} = \{v_1, .., v_m\} = h_s(\mathbf{x})$
$\triangleright$ Merge coverage ranges
$\text{left} = \{\min\{b \cdot p, 1.0\} \mid b \in [1, \lceil \frac{1}{p} \rceil]\}$
$\text{right} = \{(b-1) \cdot p \mid b \in [1, \lceil \frac{1}{p} \rceil]\}$
for $v_i \in \mathbf{v}$ **do**
  $b = \lceil \frac{v_i}{p} \rceil$
  $\text{left}_b = \min\{v_i, \text{left}_b\}$
  if $b < \lceil \frac{1}{p} \rceil$ **then**
    $\text{right}_{b+1} = \max\{v_i + p, \text{right}_{b+1}\}$
  else
    $\text{right}_1 = \max\{v_i + p - 1, \text{right}_1\}$
  end if
end for
$b = \lceil \frac{1}{p} \rceil$
$\text{left}_b = \max\{\text{left}_b, \text{right}_b\}$
$\text{right}_b = \text{right}_1 + 1$
for $b \in [1, \lceil \frac{1}{p} \rceil - 1]$ **do**
  $\text{left}_b = \max\{\text{left}_b, \text{right}_b\}$
  $\text{right}_b = \text{right}_{b+1}$
end for
$\triangleright$ Randomization according to coverage ranges
$l = \sum_{b \in [1, \lceil \frac{1}{p} \rceil]} (\text{right}_b - \text{left}_b)$
$r = \text{UniformRandom}(0.0, 1.0)$
$a = 0.0$
for $b \in [1, \lceil \frac{1}{p} \rceil]$ **do**
  $a = a + \frac{e^\epsilon (\text{right}_b - \text{left}_b)}{\Omega}$
  if $a > r$ **then**
    $z = \text{right}_b - \frac{(a-r)\Omega}{e^\epsilon}$
    break
  end if
  $a = a + \frac{(\Omega - l \cdot e^\epsilon)(\text{left}_{b\%\lceil \frac{1}{p} \rceil + 1} + \lfloor b \cdot p \rfloor - \text{right}_b)}{(1-l)\Omega}$
  if $a > r$ **then**
    $z = \text{left}_{b\%\lceil \frac{1}{p} \rceil + 1} - \frac{(a-r)(1-l)\Omega}{\Omega - l \cdot e^\epsilon}$
    break
  end if
end for
$z = z \bmod 1.0$
return (s, z)

Table 4: Parameters for the experiment.

Parameter	Enumerated values
domain size d	$256, 512, 1024, 2048$
cardinality m	$1, 2, 4, 16$
number of users n	$10000, 100000, 1000000$
privacy budget ϵ	$0.001, 0.01, 0.1, 0.2, 0.8, 1.0, 1.5, 2.0, 3.0$

The wheel mechanism could be deemed as a continuous analogy of the PrivSet mechanism. Through the continuous design, it enjoys a theoretical-guaranteed optimal utility, meanwhile substantially reducing computation and communication costs. The PrivSet mechanism may also have the optimal utility, however it is extremely hard to be proved due to the discreteness of its parameter k. Thus such a continuous analogy of the previous mechanisms with discrete parameters (e.g., in [13, 23, 33, 34, 38]) not only reduces computation/communication costs, but also facilitates theoretical analysis on error bounds.

6. EXPERIMENTS

This section includes an experimental evaluation of both computational efficiency and statistical efficiency. Chosen values of parameters are summarized in Table 4.

6.1 Experimental settings

Datasets. Since existing set-valued mechanisms are data-independent, we generate several datasets by sampling m-sized data from a d-sized domain using reservoir sampling.
Competitors. We compare the wheel mechanism with two state-of-the-art mechanisms: the RAPPOR mechanism [13] with Bloom filter size d and the PrivSet mechanism [33]. PrivSet mechanism is equivalent to k-Subset mechanism [34, 38] for categorical data ($m \equiv 1$), which means it has best-so-far error bounds for both categorical and set-valued data distribution estimation.

It should be noted that the scaled distribution $\frac{\theta}{m}$ lies in the probability simplex, while the distribution estimator $\frac{\hat{\theta}}{m}$ given by all LDP approaches may not. With the prior knowledge that $m = \sum_{i \in [1,d]} \theta_i$, the estimated distributions during each simulation can be optimized by mapping to the probability simplex [37]. All results here are computed based on post-processed distribution estimators.
Evaluation Metrics. The *average running time* is used to measure the computational cost of the data randomization procedure on the user side. The performance metrics regarding data utility (distribution estimation accuracy) include the *total variation error* (TVE, a.k.a. ℓ_1-norm error)

$$|\hat{\theta} - \theta|_1 = \sum_{i \in [1,d]} |\hat{\theta}_i - \theta_i|,$$

and the *maximum absolute error* (MAE, a.k.a. ℓ_∞-norm error)

$$|\hat{\theta} - \theta|_\infty = \max_{i \in [1,d]} |\hat{\theta}_i - \theta_i|.$$

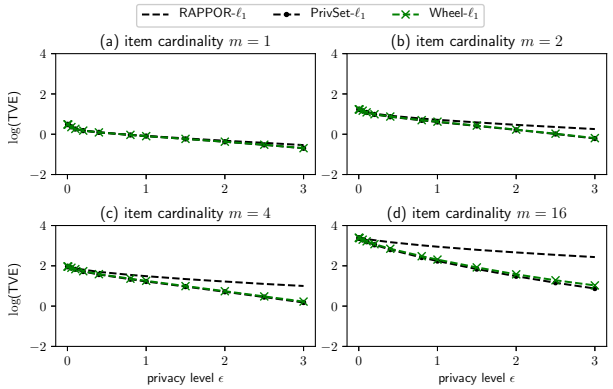
All results are the average value of 200 independent repetitions. Experimental results on extremely low/high privacy regimes (e.g., $\epsilon = 0.0001$ or $\epsilon = 10.0$) are presented in Table 5 as a complement to the following results.

Table 5: Experimental TVE results under extremely low/high privacy budgets.

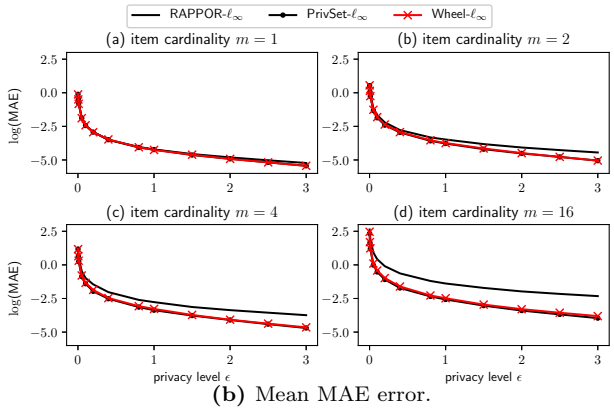
Mechanisms	$\epsilon = 0.0001$	$\epsilon = 1.0$	$\epsilon = 10.0$
RAPPOR	7.21	4.43	0.94
PrivSet	7.20	3.42	0.18
Wheel	7.20	3.73	0.25

We here omitted experiments on communication costs of different approaches, given that their communication complexities have been analyzed in Table 1 and Table 2, and necessary bits of transmitting are theoretically determinable.

6.2 Computational efficiency evaluation

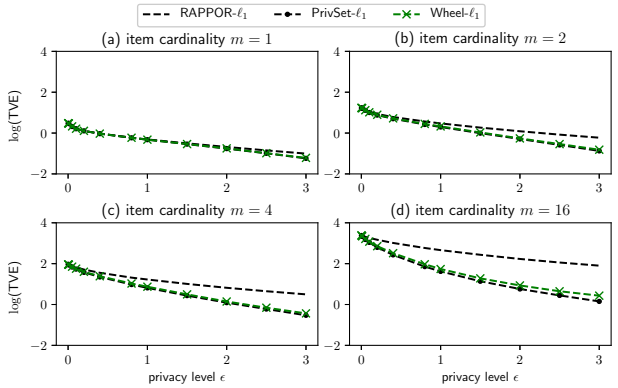


(a) Mean TVE error.

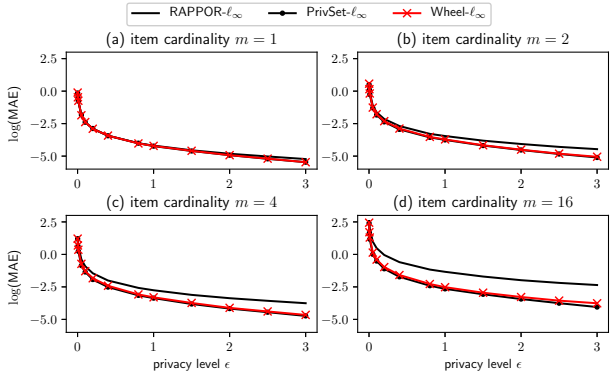


(b) Mean MAE error.

Figure 5: Experimental results on $n = 100\,000$ users, domain size $d = 512$ and the set cardinality m ranges from 1 to 16.



(a) Mean TVE errors.



(b) Mean MAE errors.

Figure 6: Experimental results on $n = 100\,000$ users, domain size $d = 256$ and the set cardinality m ranges from 1 to 16.

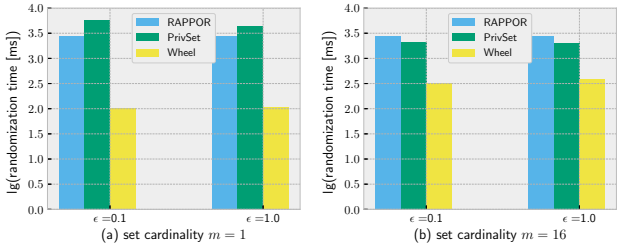


Figure 8: Total randomization time on $n = 1000$ users, domain size $d = 1024$ and the set cardinality $m = 1$ or 16.

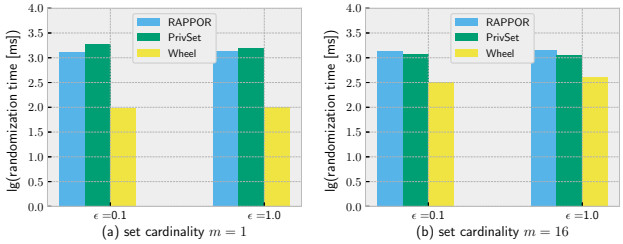


Figure 7: Total randomization time on $n = 1\,000$ users, domain size $d = 512$ and the set cardinality $m = 1$ or 16.

6.2.1 *Vary item cardinality* m

Figure 7 presents the decadic logarithm of user-side running time of three mechanisms varying item cardinality m. We can see that the wheel mechanism is extremely fast; every user finishes data randomization procedure in a few milliseconds, while other mechanisms need tens of milliseconds. The wheel mechanism is 3-10x faster than other mechanisms.

6.2.2 *Vary domain size* d

We here study how the computational costs vary with domain size d. Results of the user-side running time are shown in Figures 8 and 11, when the domain size is 1024 and 2048 respectively. Compared with results in Figure 7, it can be observed that the costs of wheel mechanism are domain-size independent, while the costs of RAPPOR and PrivSet mechanisms grow at least linearly with the domain size. The wheel mechanism is $5 - 100$x faster than other mechanisms.

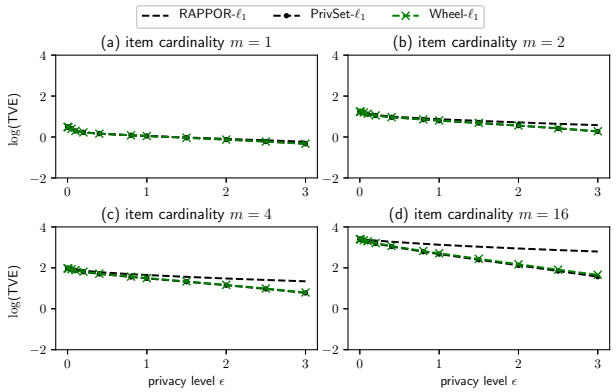
6.3 Statistical efficiency evaluation

6.3.1 *Vary item cardinality* m

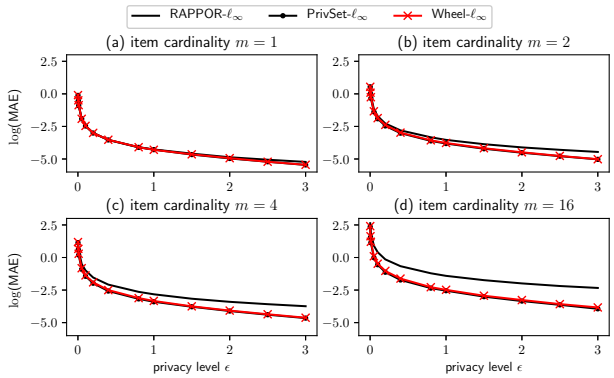
Simulated with 100000 users and the item domain size at 512, Figures 5a and 5b show the natural logarithm of TVE errors and MAE errors for distribution estimation respectively. The wheel mechanism achieves close estimation accuracy to the cost-intensive PrivSet mechanism. Except for cases when the privacy budget is extremely low (and the estimated distribution is almost non-informative), the errors of the wheel mechanism grow with $\sqrt{m}$, as contrast errors of the RAPPOR mechanism grow with m.

6.3.2 *Vary domain size* d

Simulated with 100000 users, Figures 6a and 6b show natural logarithm of TVE errors and MAE errors for distribution estimation respectively when the item domain size is 256, Figure 9a and 9b show results when the item domain

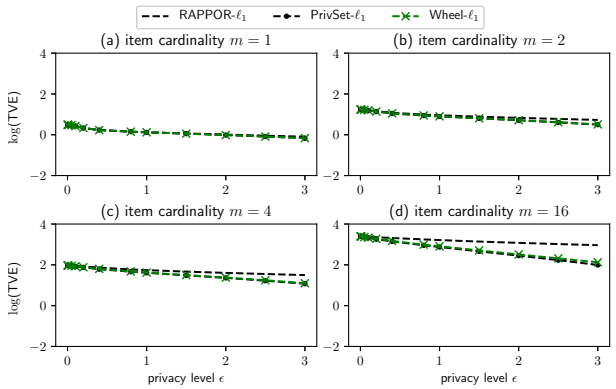


(a) Mean TVE errors.

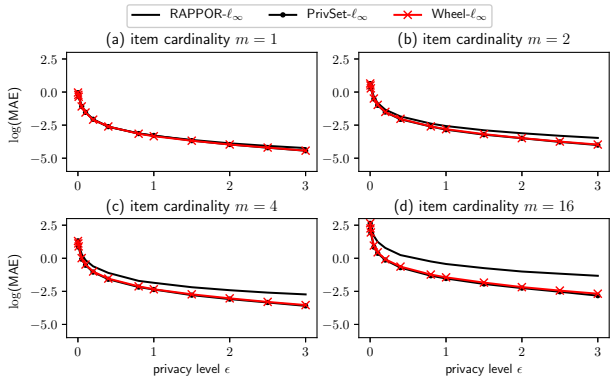


(b) Mean MAE errors.

Figure 9: Experimental results on $n = 100\,000$ users, domain size $d = 1024$ and the set cardinality m ranges from 1 to 16.



(a) Mean TVE errors.



(b) Mean MAE errors.

Figure 10: Experimental results on $n = 10\,000$ users, domain size $d = 1024$ and the set cardinality m ranges from 1 to 16.

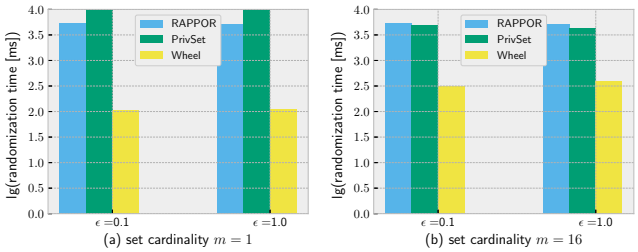


Figure 11: Total randomization time on $n = 1\,000$ users, domain size $d = 2048$ and the set cardinality $m = 1$ or 16.

size is 1024. The TVE and MAE errors of the wheel mechanism grow roughly with $\sqrt{d}$.

6.3.3 *Vary number of users* n

Simulated with 10000 users and the item domain size at 512, Figures 10a and 10b show logarithm of TVE errors and MAE errors for distribution estimation respectively. Compared to the results in Figures 5a and 5b, the estimation errors of all mechanisms increase roughly by $\sqrt{\frac{100000}{10000}}$.

6.4 Summary

Through these experiments, we can conclude that the wheel mechanism only puts domain-independent and minimum computational overheads on the user side, meanwhile achieves theoretic-guaranteed optimal utility for item distribution estimation on the server side. These experimental results confirm our theoretical error bounds $O(\frac{dm}{n\epsilon^2})$ (the TVE/ MAE is usually proportional to the root of mean squared error). It seems that practically the PrivSet mechanism also has excellent statistical efficiency, but its computation / communication overheads on the user side are much higher than the wheel mechanism. Hence for set-valued data distribution estimation under local differential privacy, the Wheel mechanism has great theoretical and practical advantages over existing approaches.

7. CONCLUSION

For set-valued data distribution estimation with ϵ-LDP, this work gave tight minimax error bounds and proposed an efficient & optimal mechanism: the wheel mechanism. The mechanism has the advantage of being domain-independent. Besides, it needs only $O(\min\{m \log m, me^\epsilon\})$ computational costs on the user side and $O(\log(me^\epsilon))$ communication costs between a user and the server. Compared with existing approaches depending on $O(d)$ or $O(\log d)$, the proposed wheel mechanism is practical for large-scale set-valued data aggregation in online services. On the service provider side, the mechanism provides an unbiased distribution estimator with much-improved error bounds from the previous $\Theta(\frac{m^2 d}{n\epsilon^2})$ to the optimal rate of $\Theta(\frac{md}{n\epsilon^2})$. Experimental results validated the computational/statistical efficiency of the mechanism; specifically, it has 3-100x speedup on user-side running time.

8. REFERENCES

[1] J. Acharya, Z. Sun, and H. Zhang. Hadamard response: Estimating distributions privately, efficiently, and with little communication. *arXiv preprint arXiv:1802.04705*, 2018.

[2] R. Bassily and A. Smith. Local, private, efficient protocols for succinct histograms. In *Proceedings of the forty-seventh annual ACM symposium on Theory of computing*, pages 127–135. ACM, 2015.

[3] R. Chen, N. Mohammed, B. C. Fung, B. C. Desai, and L. Xiong. Publishing set-valued data via differential privacy. *PVLDB*, 4(11):1087–1098, 2011.

[4] G. Cormode, T. Kulkarni, and D. Srivastava. Marginal release under local differential privacy. In *Proceedings of the 2018 International Conference on Management of Data*, pages 131–146. ACM, 2018.

[5] G. Cormode, C. Procopiuc, D. Srivastava, and T. T. Tran. Differentially private summaries for sparse data. In *Proceedings of the 15th International Conference on Database Theory*, pages 299–311. ACM, 2012.

[6] G. Cormode, M. Procopiuc, D. Srivastava, and T. T. Tran. Differentially private publication of sparse data. *arXiv preprint arXiv:1103.0825*, 2011.

[7] D. Dachman-Soled, T. Malkin, M. Raykova, and M. Yung. Efficient robust private set intersection. In *International Conference on Applied Cryptography and Network Security*, pages 125–142. Springer, 2009.

[8] E. De Cristofaro and G. Tsudik. Practical private set intersection protocols with linear complexity. In *International Conference on Financial Cryptography and Data Security*, pages 143–159. Springer, 2010.

[9] T. M. T. Do, J. Blom, and D. Gatica-Perez. Smartphone usage in the wild: a large-scale analysis of applications and context. In *Proceedings of the 13th international conference on multimodal interfaces*, pages 353–360. ACM, 2011.

[10] J. C. Duchi, M. I. Jordan, and M. J. Wainwright. Local privacy and statistical minimax rates. In *Foundations of Computer Science (FOCS), 2013 IEEE 54th Annual Symposium on*, pages 429–438. IEEE, 2013.

[11] J. C. Duchi, M. I. Jordan, and M. J. Wainwright. Minimax optimal procedures for locally private estimation. *Journal of the American Statistical Association*, 113(521):182–201, 2018.

[12] C. Dwork. Differential privacy: A survey of results. In *International Conference on Theory and Applications of Models of Computation*, pages 1–19. Springer, 2008.

[13] Ú. Erlingsson, V. Pihur, and A. Korolova. Rappor: Randomized aggregatable privacy-preserving ordinal response. In *Proceedings of the 2014 ACM SIGSAC conference on computer and communications security*, pages 1054–1067. ACM, 2014.

[14] G. Fanti, V. Pihur, and Ú. Erlingsson. Building a rappor with the unknown: Privacy-preserving learning of associations and data dictionaries. *Proceedings on Privacy Enhancing Technologies*, 2016(3):41–61, 2016.

[15] M. L. Fredman and B. Weide. On the complexity of computing the measure of ∪[ai, bi]. *Communications of the ACM*, 21(7):540–544, 1978.

[16] G. Ghinita, Y. Tao, and P. Kalnis. On the anonymization of sparse high-dimensional data. In *2008 IEEE 24th International Conference on Data Engineering*, pages 715–724. Ieee, 2008.

[17] E. Goldman. An introduction to the california consumer privacy act (ccpa). *Santa Clara Univ. Legal Studies Research Paper*, 2019.

[18] A. Greenberg. Apple's 'differential privacy'is about collecting your data–but not your data. *Wired (June 13, 2016)*, 2016.

[19] M. M. Groat, B. Edwards, J. Horey, W. He, and S. Forrest. Enhancing privacy in participatory sensing applications with multidimensional data. In *Pervasive Computing and Communications (PerCom), 2012 IEEE International Conference on*, pages 144–152. IEEE, 2012.

[20] Ş. Gündüz and M. T. Özsu. A web page prediction model based on click-stream tree representation of user behavior. In *Proceedings of the ninth ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 535–540. ACM, 2003.

[21] Y. He and J. F. Naughton. Anonymization of set-valued data via top-down, local generalization. *PVLDB*, 2(1):934–945, 2009.

[22] Y. Huang, D. Evans, and J. Katz. Private set intersection: Are garbled circuits better than custom protocols? In *NDSS*, 2012.

[23] P. Kairouz, K. Bonawitz, and D. Ramage. Discrete distribution estimation under local privacy. In *International Conference on Machine Learning*, pages 2436–2444, 2016.

[24] N. Li, W. Qardaji, D. Su, and J. Cao. Privbasis: Frequent itemset mining with differential privacy. *PVLDB*, 5(11):1340–1351, 2012.

[25] Y. Lindell. Secure multiparty computation for privacy preserving data mining. In *Encyclopedia of Data Warehousing and Mining*, pages 1005–1009. IGI Global, 2005.

[26] A. Machanavajjhala, J. Gehrke, D. Kifer, and M. Venkitasubramaniam. l-diversity: Privacy beyond k-anonymity. In *22nd International Conference on Data Engineering (ICDE'06)*, pages 24–24. IEEE, 2006.

[27] 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 *Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security*, pages 192–203. ACM, 2016.

[28] L. Sweeney. k-anonymity: A model for protecting privacy. *International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems*, 10(05):557–570, 2002.

[29] J. Tang, A. Korolova, X. Bai, X. Wang, and X. Wang. Privacy loss in apple's implementation of differential privacy on macos 10.12. *arXiv preprint arXiv:1709.02753*, 2017.

[30] O. Tene and J. Polonetsky. Privacy in the age of big data: a time for big decisions. *Stan. L. Rev. Online*, 64:63, 2011.

[31] M. Terrovitis, N. Mamoulis, and P. Kalnis. Privacy-preserving anonymization of set-valued data. *PVLDB*, 1(1):115–125, 2008.

[32] P. Voigt and A. Von dem Bussche. *The EU General Data Protection Regulation (GDPR)*, volume 18.

Springer, 2017.

[33] S. Wang, L. Huang, Y. Nie, P. Wang, H. Xu, and W. Yang. Privset: Set-valued data analyses with locale differential privacy. In *IEEE INFOCOM 2018-IEEE Conference on Computer Communications*, pages 1088–1096. IEEE, 2018.

[34] S. Wang, L. Huang, P. Wang, Y. Nie, H. Xu, W. Yang, X.-Y. Li, and C. Qiao. Mutual information optimally local private discrete distribution estimation. *arXiv preprint arXiv:1607.08025*, 2016.

[35] S. Wang, Y. Nie, P. Wang, H. Xu, W. Yang, and L. Huang. Local private ordinal data distribution estimation. In *INFOCOM 2017-IEEE Conference on Computer Communications, IEEE*, pages 1–9. IEEE, 2017.

[36] T. Wang, J. Blocki, N. Li, and S. Jha. Locally differentially private protocols for frequency estimation. In *Proc. of the 26th USENIX Security Symposium*, pages 729–745, 2017.

[37] W. Wang and M. A. Carreira-Perpinán. Projection onto the probability simplex: An efficient algorithm with a simple proof, and an application. *arXiv preprint arXiv:1309.1541*, 2013.

[38] M. Ye and A. Barg. Optimal schemes for discrete distribution estimation under locally differential privacy. *IEEE Transactions on Information Theory*, 2018.

[39] B. Yu. Assouad, fano, and le cam. In *Festschrift for Lucien Le Cam*, pages 423–435. Springer, 1997.

[40] C. Zeng, J. F. Naughton, and J.-Y. Cai. On differentially private frequent itemset mining. *PVLDB*, 6(1):25–36, 2012.