LabelDP Leaks Privacy — A Tightened Correlation-aware Privacy Model for Labeled Training Data

Qiao Xue, Qingqing Ye, *Member, IEEE,* Haibo Hu, *Senior Member, IEEE,* Jian Lou, Jin Li, Chengfang Fang, Jie Shi

Abstract—It is well understood that the accuracy of machine learning models heavily depends on the amount of training data collected from individuals. However, the collection of sensitive information brings privacy risks to users. Recently, differential privacy (DP) has emerged as a rigorous privacy model for sensitive data collection. When applying DP to training data collection, a common practice to improve utility is that labels are sanitized whereas attribute values are not, a.k.a., label differential privacy (LabelDP). In this paper, we point out that LabelDP can hardly guarantee the expected privacy on labels due to the correlation between attributes and labels. To address this privacy leakage, we propose a stronger privacy model, *correlation-aware label local differential privacy* (CLLDP), to protect each individual user with the consideration of correlations between attributes and labels. Under CLLDP, we propose a perturbation protocol k *heads response* (kHR) to estimate the joint probabilistic distribution of attributes and labels. This distribution can be used for a variety of machine learning tasks, such as Naïve Bayes and decision tree, both of which are illustrated in this paper. Through extensive experiments, we show the strong privacy guarantee of CLLDP and its effectiveness in real-life machine learning tasks.

Index Terms—Local differential privacy; Labeled data collection; Joint distribution estimation.

1 Introduction

Machine learning has empowered new business opportunities in all walks of life, especially in financial, social service, and health industry. However, it is often at the cost of collecting sensitive user training data, such as personal income and clinic diagnosis. To prevent the infringement on individual privacy, businesses have been adopting LDP [1]. As a variant of DP model [2] in local setting without a trusted data collector, LDP has been integrated with existing machine learning algorithms [3], [4] to protect both attributes and the label of each training example. However, a critical issue of such approach is dimensionality curse — as the attribute number increases, more noise has to be injected to the original data to retain the same privacy guarantee, resulting in low utility of the training data.

To alleviate the issue, a simple fix is to treat attributes (e.g., a patient's age, gender, blood pressure) and label (e.g., diagnosis of a disease) separately as the latter is usually more sensitive. Known as *LabelDP* [5], [6], [7], each user reports true attributes while perturbing her label by a DP/LDP mechanism. Although LabelDP can enhance utility, it is vulnerable to inference attacks arising from the correlation between attributes and the label [8], which undermines privacy guarantee of DP/LDP on labels. We illustrate this correlation-aware inference attack as below.

TABLE 1
Conditional Distribution of White Blood Cell Level (WBC) given Diagnosis of Fever ($\Pr\{s|l\}$)

	$s = \text{WBC}\downarrow$	$s = \text{WBC}-$	$s = \text{WBC}\uparrow$
$l = \text{Flu}$	0.01	0.22	0.77
$l = \text{HIV}$	0.68	0.16	0.16

Example 1. *Table 1 shows the conditional distribution of white blood cell level (attribute s) given diagnosis of fever (label l) — Flu or HIV, which can be obtained from clinical research literature. As the conditional probability* $\Pr\{\text{WBC}\downarrow|\text{Flu}\}$ *(0.01) is far less than* $\Pr\{\text{WBC}\downarrow|\text{HIV}\}$ *(0.68), an adversary can infer* $l = \text{HIV}$ *with high confidence if* $\text{WBC}\downarrow$ *is collected, no matter how the label is perturbed.*

In essence, the root cause of this vulnerability lies in data correlations between labels and attributes. Even though the labels are perturbed, the above conditional distribution can be estimated from the perturbed labels and original attribute values by noise calibration [9]. Based on these distributions, adversaries can infer the true label that corresponds to the original attribute values, which defeats the privacy guarantee of LabelDP. A feasible solution is to perturb labels and attributes simultaneously with LDP schemes, so that adversaries cannot infer real label value from either noisy label or noisy attributes, even with perfect knowledge about data correlations. But since LDP has to assume the worst-case scenario, i.e., the strongest correlation between label and attributes, this solution suffers from low data utility

- *Qiao Xue, Qingqing Ye and Haibo Hu are with Department of Electrical and Electronic Engineering, The Hongkong Polytechnic University. E-mail: qiaoxue@nuaa.edu.cn; {qqing.ye, haibo.hu}@polyu.edu.hk*
- *Jian Lou is with Sun Yat-sen University. E-mail: louj5@mail.sysu.edu.cn*
- *Jin Li is with Institute of Artificial Intelligence and Blockchain, Guangzhou University. E-mail: lijin@gzhu.edu.cn*
- *Chengfang Fang and Jie Shi are with Huawei International, Singapore. E-mail: {fang.chengfang, shi.jie1}@huawei.com*
- *Corresponding author: Qingqing Ye*

Manuscript received April 19, 2005; revised August 26, 2015.

due to privacy budget split. In this paper, we argue that in practice it is not always necessary to assume the strongest correlations. By **modeling and quantifying the correlation between label and attributes** in the data collection domain, both strong label privacy and high data utility can be achieved at the same time.

In this paper, we propose a novel privacy notion *correlation-aware label local differential privacy* (CLLDP) to bound label privacy leakage in the local setting, by modeling data correlations between the label and attributes with conditional probabilities. These probabilities are common primitives to model data correlations in Markov Chain [10], Bayesian network [11], and Markov Random Field [12], [13]. Under CLLDP model, we propose a perturbation protocol k *heads response* (kHR) to collect user data and estimate their joint distribution without violating user label privacy. The main challenge is that, since there is no trusted collector in CLLDP, proper amount of noise must be injected to local values to satisfy the privacy guarantee while minimizing utility loss. In addition, the parameter k in kHR has a significant impact on the estimation variance, which needs to be optimized with respect to the correlations and privacy budget. We also demonstrate how to apply CLLDP protocol in machine learning. We showcase two classic models, namely Naïve Bayes and decision tree, and adopt kHR as the building block for privacy-preserving machine learning tasks. Experimental results on both synthetic and real datasets confirm the effectiveness of kHR on different computation tasks, as well as its strong label privacy guarantee. Our main contributions are summarized as below:

- We propose a new privacy model CLLDP to tighten the privacy guarantee of label information in LabelDP.
- We design a CLLDP perturbation-calibration protocol to estimate the joint distribution of attributes and labels.
- We present how CLLDP is applied in two classic machine learning models, i.e., Naïve Bayes and decision tree.
- We conduct extensive experiments on both synthetic and real datasets to evaluate the utility of the designed CLLDP protocol on various correlation and dimension settings.

In the rest of the paper, Sec. 2 introduces the preliminaries. Sec. 3 defines the problem and correlation-aware label local differential privacy, followed by the designed CLLDP protocol in Sec. 4. Sec. 5 and Sec. 6 show how the protocol is applied in machine learning. Sec. 7 shows the performance evaluation. Sec. 8 reviews related work and Sec. 9 concludes this paper.

2 PRELIMINARIES

2.1 Differential Privacy

Differential privacy is first proposed in the centralized setting [2], relying on a trusted party to collect users' raw data. Local differential privacy [1] is then proposed in the local setting where users can locally perturb their sensitive data before reporting to an untrusted data collector. DP and LDP are formally defined as follows.

Definition 1. *(Differential Privacy, DP). A randomized mechanism $\mathcal{M}$ that takes as input a dataset D and outputs $o \in O$ is ϵ-DP iff for two neighboring datasets D, D' that only differ in one labeled tuple, and any possible output $o \in O$,*

$$\Pr\{\mathcal{M}(D) = o\} \leq e^{\epsilon} \cdot \Pr\{\mathcal{M}(D') = o\}.$$

Definition 2. *(Local Differential Privacy, LDP). A randomized mechanism $\mathcal{M}$ that takes as input one labeled tuple $d \in D$ and outputs $o \in O$ is ϵ-local differential privacy iff for any two labeled tuples $d, d' \in D$, and any output $o \in O$,*

$$\Pr\{\mathcal{M}(d) = o\} \leq e^{\epsilon} \cdot \Pr\{\mathcal{M}(d') = o\}.$$

Similarly for DP and LDP, by observing output o, an adversary cannot infer with high confidence (controlled by ϵ) whether the input dataset (tuple) is D or D' (d or d'). The privacy budget ϵ indicates the privacy strength. Intuitively, a larger (resp. smaller) ϵ indicates a stronger (resp. weaker) privacy guarantee and less (resp. more) perturbation noise.

2.2 Label Differential Privacy

In the context of machine learning, label differential privacy [5], [6] is proposed to strike a balance between data privacy and model accuracy. Under LabelDP, only labels are considered sensitive and their privacy needs to be protected, while attributes are not sensitive.

Definition 3. *(Label Differential Privacy, LabelDP). A randomized mechanism $\mathcal{M}$ that takes as input a dataset D and outputs $o \in O$ is ϵ-LabelDP iff for two neighboring datasets D, D' that only differ in the label of a single tuple, and any possible output $o \in O$,*

$$\Pr\{\mathcal{M}(D) = o\} \leq e^{\epsilon} \cdot \Pr\{\mathcal{M}(D') = o\}.$$

LabelDP is originally developed under centralized differential privacy, which can be further adapted to the local setting, where the neighboring data are defined as any two tuples d, d' that only differ in the label value. That is, $d = [d^1, \cdots, d^t, l]$ and $d' = [d^1, \cdots, d^t, l']$ where $d^1, \cdots, d^t$ are attribute values and l, l' are different label values. By following the practice in the centralized setting, users can locally sanitize their label values with an ϵ-LDP scheme (e.g., [9]) and release true attributes without perturbation.

However, **for label differential privacy in either centralized or local setting, there remains a critical privacy issue**. They do not consider the correlations between the label and attributes, which can disclose the true label information, and then impair the desired label privacy (as shown in Table 1). To make up this deficiency, we then explore the impact of such correlations on the label privacy, and propose a stronger label privacy definition. To concentrate on the correlation issue, we focus on local setting in the sequel, which is considered to be more challenging than centralized setting [14].

3 LABEL PRIVACY MODEL

In this section, we first describe the problem setting, and then define label correlations with attributes, followed by label privacy leakage. Finally, we introduce Correlation-aware Label Local Differential Privacy.

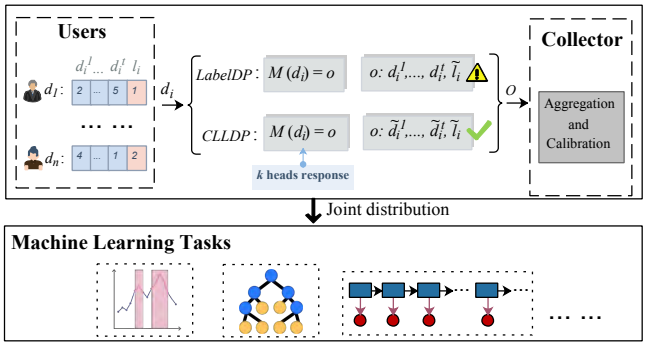


Fig. 1. Overview of label privacy preserving protocols.

3.1 Problem Formulation

As illustrated in Fig. 1, we assume a system of n users and an untrusted collector. Each user has a tuple of t *categorical* attributes $A = \{a^1, a^2, \cdots, a^t\}$ and one label l. The domain size for each attribute a^c is m^c, $c \in [t] = \{1, \cdots, t\}$, and the label set is $L = \{1, 2, \cdots, m^l\}$. Then each user data d_i, $i \in [n]$, is a (t+1)-dimensional vector $d_i = [d_i^1, d_i^2, \cdots, d_i^t, l_i]$, where $d_i^c \in [m^c]$ and $l_i \in L$. The collector gathers user data to estimate the joint distribution of attributes and the label, i.e., $\Pr\{A, L\}$, to build generative or discriminative machine learning models [15]. The former ones, e.g., Naive Bayes and hidden markov models, are essentially statistical models on this joint distribution. The latter ones, e.g, decision tree and SVM, model the conditional distribution of the label given the attribute values. Denoted by $\Pr\{L|A\}$, the conditional distribution can be derived from the joint distribution $\Pr\{A, L\}$. Therefore, the objective of this paper is to estimate the latter while protecting the sensitive *label* information of users. We summarize the main notations in Table 2.

To guarantee the privacy of labels, a naive approach is to employ LDP techniques [9] on the labels, a.k.a., the *LabelDP* privacy model as shown in Fig. 1. However, privacy leakage arises if an adversary has the knowledge of correlations between the attribute and the label, and thus infers the true label with higher confidence than what is guaranteed by LabelDP. In the next subsection, we quantify the label privacy leakage through such correlation, based on which a new privacy model CLLDP is proposed.

As a final note, in this paper we focus on vertical correlation, i.e., correlation between dimensions (i.e., the attribute and label). As is aligned with classic LDP assumption, we do not consider horizontal correlation and assume each user tuple d_i is independent of other tuples.[1]

3.2 Correlations and Label Privacy Leakage

Let $\mathbb{S}$ denote the set of all combinations of the attributes (i.e., $A = \{a^1, a^2, ..., a^t\}$):

$$\mathbb{S} = \{\{a^1\}, \{a^2\}, \cdots, \{a^1, a^2\}, \cdots, \{a^1, a^2, \cdots, a^t\}\} \quad (1)$$

For each combination $s \in \mathbb{S}$, there exists a probabilistic mapping $\mathcal{F}_s : L \to s$ from label value l to attribute values in s, which is denoted by the conditional probabilities given the label value, i.e., $\Pr\{s|l\}$. Table 3 shows two concrete examples. In (a), if label value l is 1, the probability of a^1 being 1 is 0.01, i.e., $\Pr\{\mathcal{F}_{\{a^1\}}(1) = 1\} = 0.01$; in (b), given the label value 2, the probability of a^1a^2 taking the pair of values $(3, 2)$ is 0.12, i.e., $\Pr\{\mathcal{F}_{\{a^1,a^2\}}(2) = (3, 2)\} = 0.12$. When $s = \{a^1, a^2, \cdots, a^t\}$ is the complete set, we simplify $\mathcal{F}_s$ by $\mathcal{F}$, and define $\mathcal{F}$ as the correlations between the label and attributes. Based on it, next we define label privacy leakage of LabelDP where only labels are sanitized.

1. Examples of horizontal correlation include flu or COVID status where the label of one user heavily depends on that of her family or close contacts.

TABLE 2
Notations

Symbols	Description
n	the number of users
L	the set of label values
A	the set of all attributes, $A = \{a^1, \cdots, a^t\}$
$\mathbb{S}$	the set of all combinations of A
s	one attribute combination, $s \in \mathbb{S}$
m^s	the domain size for the attribute(s) s
s_v	one instance of the attribute(s) s, $s_v \in [m^s]$
$\mathcal{F}$, $\mathcal{F}_s$	a probabilistic mapping from L to A/s
ω_s	the adversarial belief to infer a true label by s
k	the maximal number of "1" among noisy vectors
$[n]$	a set of integers, $[n] = \{1, 2, ..., n\}$

TABLE 3
Probabilistic mapping $\mathcal{F}_s$, namely, $\Pr\{s|l\}$. Assume that the domain sizes for a^1, a^2 and l are 3, 2, 2 respectively.

(a) $s = \{a^1\}$

l \ s	1	2	3
1	0.01	0.22	0.77
2	0.68	0.16	0.16

(b) $s = \{a^1, a^2\}$

l \ s	1,1	2,1	$\cdots$	3,2
1	0.01	0.11		0.55
2	0.27	0.15		0.12

Definition 4. *(Label privacy leakage, PL). Given a randomized mechanism $\mathcal{M}$ that takes as input a labeled tuple $d \in D = A \times L$ and outputs $o \in O$, i.e., $\mathcal{M}(d) = o$, for any two label values $l, l' \in L$ and any output $o \in O$, the label privacy leakage is the upper bound of the ratio of two conditional probabilities:*

$$\mathrm{PL} = \sup_{o,l,l'} \frac{\Pr\{o|l\}}{\Pr\{o|l'\}}.$$

The rationale behind this definition is that PL essentially indicates the *effective privacy budget* under the conditional distribution $\mathcal{F}$. We use LabelDP as an example to illustrate it. Recall that in LabelDP, an output o consists of the original attribute values $d^A = [d^1, d^2, ..., d^t]$ (i.e., no perturbation) and a perturbed label by an ϵ-LDP mechanism $\mathcal{M}_{\mathrm{LDP}}$, i.e., $\mathcal{M}_{\mathrm{LDP}}(l) = \tilde{l}$. By Definition 4, the label privacy leakage is:

$$\mathrm{PL} = \sup_{\tilde{l},d^A,l,l'} \frac{\Pr\{\tilde{l}|l\}}{\Pr\{\tilde{l}|l'\}} \cdot \frac{\Pr\{d^A|l\}}{\Pr\{d^A|l'\}} = e^{\epsilon} \cdot \sup_{d^A,l,l'} \frac{\Pr\{d^A|l\}}{\Pr\{d^A|l'\}}.$$

From the above equation, if labels are independent of attributes, the label privacy leakage is e^{ϵ}, i.e., the effective privacy budget equals the input privacy budget as users require. But if $\sup_{d^A,l,l'} \frac{\Pr\{d^A|l\}}{\Pr\{d^A|l'\}}$ is greater than 1, the privacy

leakage exceeds e^ϵ, which weakens the privacy guarantee of LabelDP and breaches the label privacy guarantee users expect. To avoid such extra privacy leakage, one naive solution is to spend a part of privacy budget $\epsilon_c = \log(\frac{\Pr\{d^A|l\}}{\Pr\{d^A|l'\}})$ on the correlation and to use the remaining $\epsilon - \epsilon_c$ on the labels. However, this naive solution does not work in the worst case when $\Pr\{d^A|l'\}$ is 0, i.e., the adversary can deterministically infer or eliminate some label value. Since the ratio of $\frac{\Pr\{d^A|l\}}{\Pr\{d^A|l'\}}$ becomes ∞, we cannot find an effective ϵ_c to rectify extra privacy leakage caused by correlations. To correct this discrepancy, we need a new privacy definition to bound this leakage within e^ϵ.

3.3 Correlation-aware Label Local Differential Privacy

As analyzed in Sec. 3.2, adversaries can utilize their knowledge of data correlations to obtain more label information (i.e., beyond the privacy guarantee of e^ϵ) from attribute values in the output (i.e., o) of a randomized mechanism. To mathematically reflect the confidence of the adversaries inferring a true label from some attribute combination $s \in \mathbb{S}$, we propose the notion of *adversarial belief* ω_s.

Definition 5. *(Adversarial belief ω_s). Given a randomized algorithm $\mathcal{M}(d) = o$ with a labeled tuple $d \in D$ as input and $o \in O$ as output, let $o_s \subset o$ denote the subset output value(s) of attribute combination s. The adversarial belief of s, denoted by ω_s, is the maximum summation of the conditional probabilities of all elements ν in any $o_s \subset o \in O$, given any true label $l \in L$. Formally,*

$$\omega_s = \max_{o \in O, l \in L} \sum\nolimits_{\nu \in o_s} \Pr\{\nu|l\}. \tag{2}$$

In the above, summation over all elements in o_s tightly bounds the conditional probability of o_s on label l. We use an example to explain the above definition. In Table 1, the domain of attribute s is $\{\text{WBC}\downarrow, \text{WBC}-, \text{WBC}\uparrow\}$ and the label set L is {Flu, HIV}. Suppose there is such a mechanism that employs subset selection [16] on attribute s to randomly output two values of s, and employs General Random Response (GRR) [17]) on the label to randomly output a label. For an output $o = \{\text{WBC}\downarrow, \text{WBC}-, \text{HIV}\}$, the corresponding $o_s = \{\text{WBC}\downarrow, \text{WBC}-\}$, and the summation of the conditional probabilities given l=HIV is $\Pr\{\text{WBC}\downarrow|\text{HIV}\}+\Pr\{\text{WBC}-|\text{HIV}\} = 0.68+0.16 = 0.84$, as shown in Table 1. By enumerating all possible $o_s \in O$, i.e., $\{\{\text{WBC}\downarrow, \text{WBC}-\}, \{\text{WBC}-, \text{WBC}\uparrow\}, \{\text{WBC}\downarrow, \text{WBC}\uparrow\}\}$, and enumerating all possible labels {Flu, HIV}, we can obtain that the summations under l = Flu are $\{0.23, 0.99, 0.78\}$ and those under l = HIV are $\{0.84, 0.32, 0.84\}$. As such, $\omega_s = \max\{0.23, 0.99, 0.78, 0.84, 0.32, 0.84\} = 0.99$.

To defend against the adversarial belief, we propose *correlation-aware label local differential privacy* model to guarantee a more rigorous label privacy than LabelDP.

Definition 6. *(Correlation-aware Label Local Differential Privacy, CLLDP). A randomized algorithm $\mathcal{M}$ that takes as input a labeled tuple $d \in D = A \times L$ and outputs $o \in O$ is ϵ-CLLDP with respect to the adversarial belief $\omega = \{\omega_s | s \in \mathbb{S}\}$ for any attribute combination s, iff for any two label values $l, l' \in L$ and any output $o \in O$,*

$$\Pr\{\mathcal{M}(\mathcal{F}(l), l) = o|\omega\} \leq e^\epsilon \cdot \Pr\{\mathcal{M}(\mathcal{F}(l'), l') = o|\omega\},$$

where $\mathcal{F}$ is a probabilistic mapping from the label to attributes A as defined in Sec. 3.2.

By Def. 4 and Def. 6, if a randomized algorithm $\mathcal{M}$ satisfies ϵ-CLLDP, the label privacy leakage of user data (i.e., input) could be bounded within e^ϵ, i.e., $\sup_{o,l,l'} \frac{\Pr\{o|l\}}{\Pr\{o|l'\}} \leq e^\epsilon$, given adversarial belief ω.

In addition, we assume that the adversary's maximum background knowledge can be quantified and upper bounded by ω values. Under the assumption, Def. 6 can always ensure an effective privacy guarantee of e^ϵ. But if an adversary's knowledge exceeds the bound, label privacy leakage beyond e^ϵ would occur. The detailed analysis is deferred to Sec. 4.3 where we will elaborate the setting of ω values and its privacy implications.

CLLDP v.s. LDP. In essence, CLLDP is a general form of LDP, and it degrades to LDP when $\omega_s = 1, \forall s \in \mathbb{S}$. Intuitively, a large (resp. small) ω_s indicates a strong (resp. weak) correlation, i.e., large (resp. small) value of $\Pr\{\nu|l\}$; and $\omega_s = 1$, the maximum value, indicates the strongest correlation between attributes and labels. The strongest correlation can happen in extreme cases where each label $l \in L$ is deterministically mapped to a distinct attribute value ν in the domain, so ω_s is always 1 by Eq. (2). In other words, there is a one-to-one mapping between the label and attributes. In such case, to guarantee the label privacy, CLLDP must provide the same protection on attribute values as labels, which becomes identical to the LDP model.

In the next section, we will study the perturbation protocols that satisfy CLLDP. Once the collector aggregates these perturbed data and estimates the joint distribution of attributes and labels by calibration, many machine learning models can be constructed privately. In Sec. 5 and Sec. 6, we will show two private machine learning algorithms where the CLLDP perturbation protocols serve as a building block.

4 Perturbation Protocol under CLLDP

In this section, we first propose a basic CLLDP solution, based on which, we then present a well-designed CLLDP protocol k heads response, and figure out a strategy to set an optimal k value for utility enhancement.

To simplify the presentation, in what follows, we assume each user data tuple consists of one attribute, i.e., A=a, and one label l, so there is only one attribute combination $s = a$. The domains for A and L are denoted by $[m^s]$ and $[m^l]$ respectively. X is the Cartesian product over A and L, i.e, $X = A \times L$, whose domain size $m = m^s \cdot m^l$. Without loss of generality, the domain for X is denoted by $[m] = \{1, 2, ..., m\}$ where each $x \in [m]$ indicates a pair of values (s_v, l) for the attribute s and the label, $s_v \in [m^s]$, $l \in L$. So the distribution of X is the joint distribution of A and L.

4.1 Basic Solution

The basic solution is inspired by the Unary Encoding [9], [18] which is an LDP scheme for frequency estimation. Firstly, each user i encodes her value $x_i \in [m]$ (indicating her attribute value d_i^s and label l_i) into a binary vector v_i with length m, where only the x_i-th bit $v_i^{x_i}$ is 1 and others are 0. In perturbation, each bit v_i^x ($x \in [m]$) keeps

unchanged (i.e., $\tilde{v}_i^x = v_i^x$) with the probability $p \geq \frac{1}{2}$, or is flipped to $\tilde{v}_i^x = 1 - v_i^x$ with $1-p$, i.e.,

$$\tilde{v}_i^x = \begin{cases} v_i^x, & \text{w.p.} \quad p \geq \frac{1}{2}; \\ 1 - v_i^x, & \text{w.p.} \quad 1-p. \end{cases}$$

At last, the noisy vector $\tilde{v}_i = [\tilde{v}_i^1, \cdots, \tilde{v}_i^m]$ is reported to the collector.

After gathering the perturbed vectors from n users, the collector can derive an empirical frequency of "1" on each bit, i.e., $\tilde{f}_x = \frac{\sum_i \tilde{v}_i^x}{n}$, and the true frequency of each value $x \in [m]$ is estimated by $\hat{f}_x = \frac{\tilde{f}_x - 1 + p}{2p-1}$. The following theorem establishes the CLLDP guarantee of the basic solution.

Theorem 1. *The basic solution is ϵ- correlation-aware label local differential privacy where $\epsilon = \log(\frac{\omega_s \cdot p^2}{(1-p)^2} + 1 - \omega_s)$.*

Proof. Let $\mathcal{M}$ denote the perturbation process in the solution and $\tilde{v}$ be some output from $\mathcal{M}$. The probability of observing $\tilde{v}$ given the true label l is

$$\Pr\{\tilde{v}|l\} = \sum\nolimits_{s_v \in [m^s]} \Pr\{\tilde{v}|x_{s_v}^l\} \Pr\{s_v|l\}$$

where $x_{s_v}^l$ denotes the value specified by a pair of s_v and l with Cartesian product, i.e., $(s_v, l) \Leftrightarrow x_{s_v}^l \in [m]$.

Without loss of generality, we assume that the first k ($\leq m$) bits on $\tilde{v}$ are "1" and the later $m-k$ bits are "0", i.e., $\tilde{v} = [1, \cdots, 1, 0, \cdots, 0]$. For each value $x_{s_v}^l$, the probability of $\Pr\{\tilde{v}|x_{s_v}^l\}$ is

$$\Pr\{\tilde{v}|x_{s_v}^l\} = \begin{cases} \alpha = p^{m-k+1}(1-p)^{k-1}, & \text{if } x_{s_v}^l \in \{1, \cdots, k\}; \\ \beta = p^{m-k-1}(1-p)^{k+1}, & \text{otherwise.} \end{cases}$$

Then the conditional probability is

$$\Pr\{\tilde{v}|l\} = \alpha \cdot \sum\nolimits_{s_v \in S} \Pr\{s_v|l\} + \beta \cdot (1 - \sum\nolimits_{s_v \in S} \Pr\{s_v|l\})$$

where S is a set of attribute values whose corresponding Cartesian products with l are in the set of $\{1, \cdots, k\}$, i.e., $S = \{s_v | x_{s_v}^l \in \{1, \cdots, k\}\} = \{s_v | \tilde{v}^{x_{s_v}^l} = 1\}$.

According to Def. 5, o_s in the basic solution is calculated as $o_s = \{s_v | \tilde{v}^{x_{s_v}^l} = 1, \exists\, l \in L\}$[2]. So we have $S \subseteq o_s$ and $\omega_s \geq \sum_{s_v \in S} \Pr\{s_v|l\}$. Then we get

$$\Pr\{\tilde{v}|l\} \leq \alpha \cdot \omega_s + \beta \cdot (1 - \omega_s) \quad (\text{as } \alpha \geq \beta),$$

and when $S = \emptyset$, the probability reaches the minimum, i.e.,

$$\Pr\{\tilde{v}|l\} \geq \beta.$$

Then the ratio of any two such conditional probabilities with different labels $l, l' \in L$ is bounded by,

$$\frac{\Pr\{\tilde{v}|l\}}{\Pr\{\tilde{v}|l'\}} \leq \frac{\alpha \cdot \omega_s + \beta \cdot (1 - \omega_s)}{\beta} = \frac{\omega_s \cdot p^2}{(1-p)^2} + 1 - \omega_s.$$

By Definition 6,

$$\frac{\Pr\{\mathcal{M}(\mathcal{F}(l), l) = \tilde{v} | \omega = \omega_s\}}{\Pr\{\mathcal{M}(\mathcal{F}(l'), l') = \tilde{v} | \omega = \omega_s\}} = \frac{\Pr\{\tilde{v}|l\}}{\Pr\{\tilde{v}|l'\}} \leq \frac{\omega_s \cdot p^2}{(1-p)^2} + 1 - \omega_s.$$

Thus, $\epsilon = \log(\frac{\omega_s \cdot p^2}{(1-p)^2} + 1 - \omega_s)$. □

2. For example, given $m^s = 3$ and $L = \{1, 2\}$, the domain of (s, L) is $\{(1,1), (2,1), (3,1), (1,2), (2,2), (3,2)\}$, which is indicated by $\{1, 2, 3, 4, 5, 6\}$, i.e, $m = 6$. If $\tilde{v} = 110100$, it means to output $\{1, 2, 4\} \Leftrightarrow \{(\underline{1}, 1), (\underline{2}, 1), (\underline{1}, 2)\}$, so $o_s = \{1, 2\}$.

To satisfy a desired privacy level, i.e., given the privacy budget ϵ, the perturbation probability p is calculated as follows,

$$e^\epsilon = \frac{\omega_s \cdot p^2}{(1-p)^2} + 1 - \omega_s,$$
$$\omega_s \cdot p^2 = (e^\epsilon + \omega_s - 1)(1-p)^2,$$
$$p = \frac{\sqrt{e^\epsilon - 1 + \omega_s}}{\sqrt{e^\epsilon - 1 + \omega_s} + \sqrt{\omega_s}}.$$

When ω_s is large (resp. small), the value of p should be small (resp. large), which corresponds to a heavy (resp. slight) perturbation with the low (resp. high) probability of reporting true values.

In essence, the setting of ω_s depends on the maximal number k of "1" among all possible noisy vectors (i.e., the output set O). To consider an extreme case, a noisy output $\tilde{v}$ (i.e., o) is an all-"1" vector, which indicates that o_s contains all attribute values. Then by Eq. (2), ω_s should be calculated to 1, i.e., the maximum value. Consequently, a large perturbation is introduced to sanitize the privacy information and degrades the data utility. Next we propose an improved CLLDP protocol to address the problem and minimize the utility loss.

4.2 k Heads Response

This subsection presents the designed CLLDP protocol k *heads response* (kHR), whose main idea is to limit the number k of "1" on the noisy vectors during perturbation, so that the collector can set a smaller ω_s value (instead of the maximum 1) with a small-size o_s and enhance the data utility. A method to choose an optimal k is illustrated in Sec. 4.3 for further improving the performance of the designed protocol. In the following, we elaborate kHR in terms of perturbation, calibration and privacy analysis.

In user-side perturbation, each user i encodes her private value $x_i \in [m]$ to a binary vector v_i with length m, where only the x_i-th bit is set to 1 and denoted by $v_i^{x_i}$. Then the bit $v_i^{x_i}$ is perturbed by the following probabilities

$$\tilde{v}_i^{x_i} = \begin{cases} 1, & \text{w.p.} \quad p \geq \frac{k}{m}; \\ 0, & \text{w.p.} \quad 1-p. \end{cases} \tag{3}$$

For other bits v_i^x, $x \in [m] \backslash x_i$, their perturbation is based on the value of $\tilde{v}_i^{x_i}$. If $\tilde{v}_i^{x_i} = 1$, user i will randomly choose $k-1$ ($k \leq \frac{m}{2}$) distinct values from $[m] \backslash x_i$ and set the corresponding bits to 1; otherwise, different k values are uniformly sampled from $[m] \backslash x_i$ and set the bits to 1. For those unsampled bits, they are set to 0. With a subset $K \subset [m] \backslash x_i$ consisting of the sampled values, the perturbation of each "0" bit v_i^x ($x \in [m] \backslash x_i$) can be expressed as

$$\tilde{v}_i^x = \begin{cases} 1, & \text{if } x \in K; \\ v_i^x, & \text{otherwise.} \end{cases} \tag{4}$$

By Eqs. (3)-(4), this perturbation can fix the number of "1" at k on each noisy vector $\tilde{v}_i = [\tilde{v}_i^1, \cdots, \tilde{v}_i^m]$, which is reported to the data collector at last.

Based on the noisy vectors reported by n users, the data collector can derive the empirical frequency $\tilde{f}_x$ of "1" on each bit, i.e., $\tilde{f}_x = \frac{\sum_i \tilde{v}_i^x}{n}$, $x \in [m]$. By the perturbation pro-

Algorithm 1 kHR: user-side perturbation $\mathcal{M}_{k\text{HR}}$

Input: User data (d_i^s, l_i), ω_s, ϵ, k
Output: Noisy vector $\tilde{v}_i$.
1: $x_i = \textit{Cartesian}(d_i^s, l_i)$ // *Cartesian product*
2: Encode x_i to an m-length one-hot vector v_i
3: Calculate the probability p by Eq. (6)
4: Perturb the "1" bit $v_i^{x_i}$ by Eq. (3)
5: **if** $\tilde{v}_i^{x_i} = 1$ **then**
6: Randomly select a set K of $k-1$ values from $[m]\backslash x_i$
7: **else**
8: Randomly select a set K of k values from $[m]\backslash x_i$
9: Perturb each "0" bit v_i^x by Eq. (4), $x \in [m]\backslash x_i$
10: **Report** $\tilde{v}_i = [\tilde{v}_i^1, \cdots, \tilde{v}_i^m]$

tocol, the probability of one false positive report (denoted by q) is calculated by

$$q = \Pr\{\tilde{v}_i^x = 1 | v_i^x = 0\} = \frac{k-p}{m-1}.$$

Then with the empirical frequency $\tilde{f}_x$ and the probability p, the true frequency of each value $x \in [m]$ can be estimated by the following calibration

$$\Phi_{k\text{HR}}: \ \hat{f}_x = \frac{\tilde{f}_x - q}{p - q}. \tag{5}$$

The privacy guarantee of kHR is analyzed by the following theorem.

Theorem 2. *kHR provides ϵ- correlation-aware label local differential privacy where $\epsilon = \log(\omega_s \cdot \frac{(m-k)p}{k(1-p)} + 1 - \omega_s)$.*

Proof. Let $\mathcal{M}_{k\text{HR}}$ denote the perturbation process in k heads response and $\tilde{v}$ be some output from $\mathcal{M}_{k\text{HR}}$. The probability of observing $\tilde{v}$ given the true label l is

$$\Pr\{\tilde{v}|l\} = \sum\nolimits_{s_v \in [m^s]} \Pr\{\tilde{v}|x_{s_v}^l\} \Pr\{s_v|l\}$$

where $x_{s_v}^l$ denotes the value specified by a pair of s_v and l. For each value $x_{s_v}^l$ ($s_v \in [m^s]$), the probability of $\Pr\{\tilde{v}|x_{s_v}^l\}$ is calculated as

$$\Pr\{\tilde{v}|x_{s_v}^l\} = \begin{cases} p \cdot \frac{1}{\binom{m-1}{k-1}} & \text{if } \tilde{v}^{x_{s_v}^l} = 1, \\ (1-p) \cdot \frac{1}{\binom{m-1}{k}} & \text{otherwise.} \end{cases}$$

Then the conditional probability is

$$\Pr\{\tilde{v}|l\} = \frac{p}{\binom{m-1}{k-1}} \cdot \sum_{s_v \in S} \Pr\{s_v|l\} + \frac{1-p}{\binom{m-1}{k}} \cdot (1 - \sum_{s_v \in S} \Pr\{s_v|l\})$$

where S is a set of attribute values whose corresponding bits (Cartesian products) with l are 1 in $\tilde{v}$, i.e., $S = \{s_v | \tilde{v}^{x_{s_v}^l} = 1\}$.

According to Def. 5, o_s in kHR is calculated as $o_s = \{s_v | \tilde{v}^{x_{s_v}^l} = 1, \exists\, l \in L\}$, so $S \subseteq o_s$ and $\omega_s \geq \sum_{s_v \in S} \Pr\{s_v|l\}$. Then we have

$$\Pr\{\tilde{v}|l\} \leq \frac{p \cdot \omega_s}{\binom{m-1}{k-1}} + \frac{(1-p)(1-\omega_s)}{\binom{m-1}{k}},$$

and when $S = \emptyset$, we achieve the minimum conditional probability, i.e.,

$$\Pr\{\tilde{v}|l\} \geq \frac{1-p}{\binom{m-1}{k}}.$$

Algorithm 2 kHR: collector-side calibration $\Phi_{k\text{HR}}$

Input: Noisy vectors $\tilde{v}_1, \cdots, \tilde{v}_n$.
Output: Joint distribution $\hat{f}$.
1: **for** each $x \in [m]$ **do**
2: Calculate the noisy frequency $\tilde{f}_x = \frac{\sum_i \tilde{v}_i^x}{n}$
3: Get the estimated frequency $\hat{f}_x$ by Eq. (5)
4: **Return** $\hat{f} = \{\hat{f}_1, \cdots, \hat{f}_m\}$

Then the ratio of two such conditional probabilities with any two labels $l, l' \in L$ is bounded by

$$\frac{\Pr\{\tilde{v}|l\}}{\Pr\{\tilde{v}|l'\}} \leq \omega_s \cdot \frac{(m-k)p}{k(1-p)} + 1 - \omega_s.$$

By Definition 6,

$$\frac{\Pr\{\mathcal{M}_{k\text{HR}}(\mathcal{F}(l), l) = \tilde{v} | \omega = \omega_s\}}{\Pr\{\mathcal{M}_{k\text{HR}}(\mathcal{F}(l'), l') = \tilde{v} | \omega = \omega_s\}} = \frac{\Pr\{\tilde{v}|l\}}{\Pr\{\tilde{v}|l'\}} \leq \omega_s \cdot \frac{(m-k)p}{k(1-p)} + 1 - \omega_s.$$

Thus, $\epsilon = \log(\omega_s \cdot \frac{(m-k)p}{k(1-p)} + 1 - \omega_s)$. □

By Theorem 2, given the desired label privacy level, i.e., privacy budget ϵ, and the value of ω_s, the probability p should be calculated as

$$p = \frac{ke^\epsilon + k\omega_s - k}{ke^\epsilon + m\omega_s - k} \tag{6}$$

which is used in Eq. (3) for the perturbation.

Algorithm 1 presents the user-side perturbation. Each user i gets her true value $x_i \in [m]$ and encodes it to a one-hot vector (Lines 1-2). Then user i calculates the probability p (Line 3) and perturbs her vector v_i (Lines 4-9). Lastly, $\tilde{v}_i$ is reported to the data collector (Line 10).

Algorithm 2 briefly summarizes the collector-side calibration. For each value $x \in [m]$, the collector first calculates its empirical frequency $\tilde{f}_x$ (Line 2), and calibrates it to gain the estimated result (Line 3).

4.3 The Parameter Setting

In this subsection, we first present a theoretical utility analysis of kHR, and then propose a method to select an appropriate k in terms of decreasing the estimation variance, followed by the discussion on the strategies to achieve a proper value of ω_s.

The setting of k. By the calibration in kHR, i.e., Eq. (5), it is easy to prove that $\Phi_{k\text{HR}}$ is an unbiased estimation [9], i.e.,

$$\mathrm{E}(\hat{f}_x) = \frac{\mathrm{E}(\tilde{f}_x) - q}{p - q} = \frac{f_x \cdot p + (1 - f_x) \cdot q - q}{p - q} = f_x,$$

where f_x denotes the true frequency of $x \in [m]$. Then the variance of this estimation is calculated as [3]

$$\mathrm{Var}[\hat{f}_x] = \frac{q(1-q)}{n(p-q)^2} + \frac{f_x(1-p-q)}{n(p-q)} \approx \frac{q(1-q)}{n(p-q)^2}. \tag{7}$$

3. In practice, the domain is usually large, and the frequencies (i.e., f_x) of most items tend to be small, so the second term in Eq. (7) can be neglected.

By Eq. (7), the estimation variance is mainly related to the probabilities of p and q, both of which vary with the value of k. The following theorem analyzes the variance trend with k under different situations of ω_s.

Theorem 3. *Given privacy budget ϵ, with an increasing k, when $\omega_s < 1$, variance goes up monotonically; while when $\omega_s = 1$, variance firstly goes down until $k = \lceil \frac{m}{e^\epsilon+1} \rceil$ and then increases.*

Proof. We first analyze the variation of the probability of p and q. By Eq. (6), the value of p is related to k and ω_s, so we can write p as the form of a function, i.e., $p = g(k, \omega_s)$. To analyze the impact of k and ω_s on the probability p, we calculate the partial derivative of the function $g(k, \omega_s)$ as follows

$$g'_k = \frac{\omega_s}{(ke^\epsilon + m\omega_s - k)^2}; \quad g'_{\omega_s} = \frac{k-m}{(ke^\epsilon + m\omega_s - k)^2}.$$

By the above equalities, we learn that the value of p increases as k rises (for $\omega_s > 0$) and decreases as ω_s rises (for $k < m$). Also, since $|g'_k| \ll |g'_{\omega_s}|$, we can conclude that the variation of p is mainly dominated by the value of ω_s. According to the calculation of ω_s, i.e., Eq. (2), when $\omega_s < 1$, its value rises with an increasing k (due to the expansion of o_s). Hence, when k increases, ω_s increases and p decreases. About the probability q, it increases when k rises and p decreases by $q = \frac{k-p}{m-1}$. With the variation of p and q under an increasing k, we then analyze the variance trends. By Eq. (7), its numerator only depends on q which rises with an increase of k, so the numerator monotonically increases due to $q \leq \frac{1}{2}$. As for the denominator, it decreases as p drops and q rises under an increasing k. As such, the variance goes up monotonically as k increases before ω_s reaches 1.

Once ω_s is set to 1 (i.e., the maximum), its value no longer increases as k rises, and the probability p is totally controlled by the value of k. Besides, when $\omega_s = 1$, kHR will reduce to the schemes in [16], [19], where they have proved that the variance can be minimized when $k = \lceil \frac{m}{e^\epsilon+1} \rceil$. With that said, at $\omega_s = 1$, with an increasing k, the variance firstly goes down until $k = \lceil \frac{m}{e^\epsilon+1} \rceil$ and then increases.

□

By Theorem 3, the optimal k to minimize the variance must lie in one of the following two points $\{1, \lceil \frac{m}{e^\epsilon+1} \rceil\}$. Based on that, we respectively calculate two variances Var_1 and Var_2 with $k = 1$ and $k = \lceil \frac{m}{e^\epsilon+1} \rceil$, and the optimal k is selected by the following,

$$k = \begin{cases} 1, & \text{if } \mathrm{Var}_1 < \mathrm{Var}_2; \\ \lceil \frac{m}{e^\epsilon+1} \rceil, & \text{otherwise.} \end{cases} \tag{8}$$

The setting of ω_s. In real-world applications, we have two strategies to estimate adversarial belief ω_s for an attribute combination s. First, the collector can exploit historical data or the apriori knowledge to estimate this value. For example, a cancer investigation [20] claims that cigarette smoking is linked to 82% lung cancer cases, by which ω_s between smoking (attribute s) and lung cancer (label) could be 0.82 when the mechanism outputs only one value of s and a label. Second, the collector can alternatively sample some users (e.g., $1/10$) from the population to collect their data with LDP methods [9] and estimate conditional distributions. Then the collector can obtain the value of ω_s with the estimated distributions by Eq. (2).

If the achieved ω_s is larger than or equal to the ground truth, the protocol can always ensure an effective privacy guarantee of e^ϵ. Otherwise, the CLLDP guarantee will be degraded. The label privacy leakage is beyond e^ϵ and increases by $(\frac{\omega^*}{\omega_s} - 1)(e^\epsilon - 1)$, where ω^* is the ground truth[4].

4.4 CLLDP Protocols v.s. LDP Protocols

As discussed in Sec. 3.3, CLLDP is a generalization of LDP. Similarly, the proposed CLLDP mechanisms generalize the corresponding LDP schemes by incorporating the parameter ω_s into the perturbation process. Specifically, when $\omega_s = 1$, i.e., the strongest correlation, the basic solution and kHR are reduced to the corresponding LDP ones, i.e., Unary Encoding (UE) [9] and Subset Selection (SS) [16], which provide the same protection on both labels and attributes. In other cases ($\omega_s < 1$), the perturbation probability varies with the value of ω_s; a smaller ω_s indicates a weaker correlation, leading to lighter perturbation and utility enhancement.

UE and SS are two widely used LDP schemes for data collection, and many LDP protocols adopt them as primitives in perturbation design. Therefore, when converting the existing LDP protocols to their CLLDP version, the primitives can be simply replaced with the corresponding CLLDP mechanisms. For example, Local Hash (LH) [9] is proposed for frequency estimation on large domains. It first maps the original domain to a smaller one using hash functions, and applies Direct Encoding (a special case of SS) on the reduced domain. When analyzing LH under CLLDP, we can follow the same domain-reduction step by hash functions and then apply kHR.

5 Naïve Bayes under CLLDP

In this section, we show how to build a Naive Bayes classifier under CLLDP. We first present implementation details, followed by its privacy analysis and an overall algorithm.

The Naïve Bayes classifier is a popular Bayesian learning model. Given a new instance d with attribute values $[d^1, d^2 \cdots, d^t]$, the classifier predicts the label value by

$$l^* = \arg\max_{l \in L} \Pr\{l\} \cdot \prod_{c=1}^{t} \Pr\{d^c | l\}. \tag{9}$$

So it requires a set of probabilities: (1) the probability of each label, i.e, $\Pr\{l\}$, $l \in L$; and (2) the conditional probability of each attribute value given the label, i.e., $\Pr\{a_v^c | l\}$, $a_v^c \in [m^c]$, $a^c \in A$. To privately construct a Naïve Bayes classifier, we must derive these probabilities from the joint distribution $\Pr\{a^c, L\}$, $a^c \in A$, which can be estimated by kHR.

5.1 Perturbation and Aggregation

To avoid splitting the privacy budget, we divide users into t groups [3], each only reporting an attribute and the label. Specifically, if one group is assigned the attribute a^c, then each user i in this group will perturb her value d_i^c of a^c

4. Given the ground truth ω^*, by Theorem 2, the actual privacy guarantee should be $e^* = \omega^*(\frac{e^\epsilon + \omega_s - 1}{\omega_s}) + 1 - \omega^*$, and the increment is $e^* - e^\epsilon = (\frac{\omega^*}{\omega_s} - 1)(e^\epsilon - 1)$.

Algorithm 3 Naïve Bayes classifier under CLLDP

Input: Dataset D, adversarial belief ω, ϵ, k
Output: Distributions $\Pr\{L\}$ and $\Pr\{a^c|l\}$, $a^c \in A$, $l \in L$
// Users side:
1: **for** each user i **do**
2: Sample $c \leftarrow [t]$
3: $\tilde{v}_i = \mathcal{M}_{k\text{HR}}(d_i^c, l_i, \epsilon, \omega_{a^c}, k)$
4: **Report** $\tilde{v}_i$ and the index c
// Collector side:
5: **for** each attribute a^c **do**
6: $\Pr\{a^c, L\} = \Phi_{k\text{HR}}(\{\tilde{v}_i | i\text{'s index is } c\})$
7: The marginal probability $p_l^c = \sum_{a_v^c} \hat{f}_{a_v^c,l}$
8: Get $\Pr\{a_v^c|l\}$ by Eq. (10)
9: $\Pr\{l\}$ is calculated by $\sum_{c\in[t]} p_l^c/t$, $l \in L$
10: **Return** all distributions $\Pr\{L\}$ and $\Pr\{a^c|l\}$

and her label l_i by kHR with entire ϵ and the adversarial belief ω_{a^c} of a^c. Based on the perturbed values from each group, the data collector can estimate the corresponding joint distribution $\Pr\{a^c, L\}$ with the calibration $\Phi_{k\text{HR}}$. Then the label probability $\Pr\{l\}$ can be calculated by $\sum_{a_v^c} \hat{f}_{a_v^c,l}$ where $\hat{f}_{a_v^c,l}$ is the estimated joint probability of a_v^c and l in $\Pr\{a^c, L\}$. For different groups, the estimated results of $\Pr\{l\}$ are varied, so the collector averages them as the final marginal estimation. The conditional distribution of each attribute given the label can be acquired by

$$\Pr\{a_v^c|l\} = \frac{\hat{f}_{a_v^c,l}}{\sum_{a_{v'}^c \in [m^c]} \hat{f}_{a_{v'}^c,l}}. \tag{10}$$

Therefore, the necessary probabilities are estimated privately and the Naïve Bayes classifier can be trained under the CLLDP privacy guarantee. Since the probability $\Pr\{a_v^c|l\}$ is proportional to $\Pr\{a_v^c, l\}$ (i.e., $\hat{f}_{a_v^c,l}$), the quality of the achieved $\Pr\{a_v^c|l\}$ is upper bounded by the variance of joint estimation, i.e., Eq. (7), with a normalized factor.

5.2 Privacy Analysis

During the construction of a Naïve Bayes classifier, each user just reports one attribute value and her label by sampling. The user true data are perturbed by kHR before reporting, so her label privacy can be guaranteed by CLLDP.

Theorem 4. *The proposed scheme to construct the Naïve Bayes satisfies ϵ-CLLDP.*

Proof. Let $\mathcal{M}_{\text{NB}}$ denote the perturbation procedure. For any two label values $l, l' \in L$, and any output $(\tilde{v}, c)$, $c \in [t]$ from $\mathcal{M}_{\text{NB}}$, since the sampling process of the value c is independent of the perturbation, we have

$$\frac{\Pr\{\mathcal{M}_{\text{NB}}(\mathcal{F}(l), l) = (\tilde{v}, c)|\omega\}}{\Pr\{\mathcal{M}_{\text{NB}}(\mathcal{F}(l'), l') = (\tilde{v}, c)|\omega\}} = \frac{\Pr\{\tilde{v}|l\}}{\Pr\{\tilde{v}|l'\}} \cdot \frac{\Pr\{c\}}{\Pr\{c\}} = \frac{\Pr\{\tilde{v}|l\}}{\Pr\{\tilde{v}|l'\}}$$

Then by Theorem 2,

$$\frac{\Pr\{\mathcal{M}_{\text{NB}}(\mathcal{F}(l), l) = (\tilde{v}, c)|\omega\}}{\Pr\{\mathcal{M}_{\text{NB}}(\mathcal{F}(l'), l') = (\tilde{v}, c)|\omega\}} = \frac{\Pr\{\tilde{v}|l\}}{\Pr\{\tilde{v}|l'\}} \leq e^{\epsilon}.$$

□

5.3 Overall Construction

Algorithm 3 details the construction of private Naïve Bayes classifier under CLLDP. Firstly each user i randomly samples one value c from $[t]$ and invokes the CLLDP protocol to perturb her private values d_i^c and l_i with ϵ and ω_{a^c} (Line 3). The perturbed data $\tilde{v}_i$ with the sampled index c are reported to the collector in Line 4. Based on the perturbed data, the collector estimates the joint distribution by the calibration in Line 6. The marginal probabilities and conditional probabilities are achieved in Lines 7-8. Line 9 takes an average value of marginal probabilities from different groups. Lastly, all achieved probabilities are returned to build a private Naïve Bayes classifier.

6 Decision Tree under CLLDP

This section proposes a strategy to build the decision tree CART (i.e., Classification And Regression Tree) under CLLDP. We first present the implementation details, followed by its privacy analysis and an overall algorithm.

CART is a classification method with a binary tree structure. In CART, each leaf represents a class label; and each branch represents a conjunction of attribute values that lead to a class label (i.e., leaf). A private CART is constructed by a multi-round interaction between users and the collector. Fig. 2 illustrates a whole process with an example of four attributes: u, w, y, z, and two labels. By the example, we elaborate the scheme in the following.

6.1 Perturbation and Aggregation

Round 1: user side. In the first round, each user randomly selects one attribute a^c from A and uses a set SL_1 to store the sampled attribute a^c. By the sampling process, users are divided into several groups, as shown in Fig. 2(a). Each user i perturbs her sampled attribute value d_i^c and label l_i by kHR with privacy budget ϵ_1 and adversarial belief ω_{a^c}. (The division of privacy budget is analyzed in the next subsection, i.e., Eq. (15).) Then the user i reports the perturbed data $\tilde{v}_{i,1}$ and her group index $g_{i,1}$ to the collector.

Round 1: collector side. With the noisy data from users, the collector estimates a joint distribution between each attribute and the label by the calibration $\Phi_{k\text{HR}}$. With the estimated distribution, the collector can gain the root value a_{opt} by Gini index, i.e.,

$$a_{opt} = \underset{a_v^c \in [m^c], c \in [t]}{\arg\min} \text{Gini}(D, a_v^c). \tag{11}$$

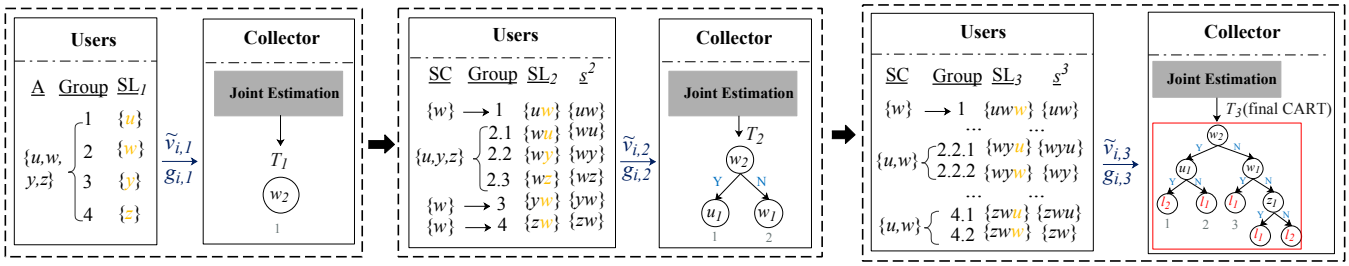
Let $D^{a_v^c}$ (resp. $D^{\neq a_v^c}$) denote a subset of D where $a^c = a_v^c$ (resp. $a^c \neq a_v^c$). The value of $\text{Gini}(D, a_v^c)$ in Eq. (11) is calculated by

$$\text{Gini}(D, a_v^c) = \Pr\{a_v^c\} \cdot \text{Gini}(D^{a_v^c}) + (1 - \Pr\{a_v^c\}) \cdot \text{Gini}(D^{\neq a_v^c})$$

$$\text{Gini}(D^*) = 1 - \sum\nolimits_{l \in L} (\Pr\{l|*\})^2,$$

where the probabilities can be acquired from the estimated joint distributions. At last, the tree T_1 with the root a_{opt} is built in the first round.

Round r ($r > 1$): user side. In round r, each user firstly get an attribute candidate set SC according to her set SL_{r-1} and the tree T_{r-1} in the previous round $r-1$. Specifically, for each unlabeled branch h (i.e., the branch does not end with a label) of tree T_{r-1}, we use a set B_a^h to record each attribute on h, e.g., for the second branch of T_2 in Fig. 2(b), we have $B_a^2 = \{w, w\}$. For each set B_a^h, if SL_{r-1} differs from B_a^h only in one attribute that does not belong to B_a^h, the



(a) Round 1 (b) Round 2 (c) Round 3

Fig. 2. An example of building CART with maximal depth $dep = 3$ under four attributes u, w, y, z. The domains are: $u = \{u_1, u_2, u_3\}, w = \{w_1, w_2, w_3\}, y = \{y_1, y_2\}, z = \{z_1, z_2\}, L = \{l_1, l_2\}$. In the figure, SC is a candidate set; $\mathrm{SL_r}$ ($r \in [dep]$) is a set of sampled attributes; s^r is one attribute combination.

last attribute in B_a^h will be included in the user's candidate set SC. For example, in round 2, each user in group 4 has $\mathrm{SL}_1 = \{z\}$ and $B_a^1 = \{w\}$ for T_1 in previous round 1. $|\mathrm{SL}_1 \backslash B_a^1| = |\{z\}| = 1$ and $z \notin B_a^1$, so the last attribute w in B_a^1 is included in SC for group 4. If $\mathrm{SC} = \emptyset$ after comparison (e.g., group 2 in round 2), the attributes not in SL_{r-1} are contained into SC, i.e., $\mathrm{SC} = A \backslash \mathrm{SL}_{r-1}$.

From the candidate set SC, each user i randomly selects one attribute a and inserts a into the set SL_{r-1} to form SL_r. The set SL_r may include some duplicate attributes, such as SL_3 of group 1 in Fig. 2(c). By removing the duplicates, user i gets an attribute combination s^r, $s^r \in \mathbb{S}$. Let S^r stand for the Cartesian product over the attributes in s^r, denoted by $S^r = \mathit{Cartesian}(s^r)$. With the original values, each user i can find the corresponding value d_i^S for S^r and exploit kHR to sanitize d_i^S and her label l_i with privacy budget ϵ_r and adversarial belief ω_{s^r}. Lastly user i reports the sanitized value $\tilde{v}_{i,r}$ and her group index $g_{i,r}$. Particularly, if $s^r = s^{r-1}$, e.g., group 1 in Fig. 2(b) and (c), the users will not report data and consume privacy budget in this round since the same data have been reported.

Round r ($r > 1$): collector side. With the perturbed data from each user group, the collector estimates a joint distribution of $\Pr\{s^r, L\}$ by the calibration. Sometimes the collector can get several estimated results on the same joint distribution from different user groups. For example, in Fig. 2(b), there are two estimated results of $\Pr\{u, w, L\}$ from group 1 and group 2.1. In the case, the collector takes an average of the different results as the final estimation.

About the tree construction, each unlabeled leaf on T_{r-1} first generates two children to construct the structure of T_r. For each new generated node j, its ancestors' values and their states (Y or N) are summarized in a set PS_j^r. For example, in Fig. 2(c), for the new generated node 2 in round 3, $\mathrm{PS}_2^3 = \{w = w_2, u \neq u_1\}$. Then a subset $D_{r,j}$ of D can be achieved with the tuples that satisfy PS_j^r. Finally, the collector can set the optimal attribute value for j with the estimated distributions by the following equation

$$a_{opt} = \underset{a_v^c \in [m^c], c \in [t]}{\arg\min} \ \mathrm{Gini}(D_{r,j}, a_v^c), \tag{12}$$

where the value of $\mathrm{Gini}(D_{r,j}, a_v^c)$ is calculated by

$$\mathrm{Gini}(D_{r,j}, a_v^c) =$$
$$\Pr\{a_v^c | \mathrm{PS}_j^r\} \cdot \mathrm{Gini}(D_{r,j}^{a_v^c}) + (1 - \Pr\{a_v^c | \mathrm{PS}_j^r\}) \cdot \mathrm{Gini}(D_{r,j}^{\neq a_v^c})$$
$$\mathrm{Gini}(D_{r,j}^*) = 1 - \sum\nolimits_{l \in L} (\Pr\{l | \mathrm{PS}_j^r, *\})^2.$$

Stop. In any round r, if $\mathrm{Gini}(D_{r,j}) \leq \mathrm{Gini}(D_{r,j}, a_v)$ for any attribute value a_v, the new node j is labeled with the most frequent class in $D_{r,j}$, i.e., $l^* = \max_{l \in L} \Pr\{l | \mathrm{PS}_j^r\}$. Also, if the depth of j reaches the given maximum dep and it has not been labeled, e.g., node 4 (z_1) in Fig. 2(c), j will generate two children and label them by the most frequent class in $D_{r,j}^{j_v}$ and $D_{r,j}^{\neq j_v}$ where j_v is the attribute value in node j.

6.2 Privacy Analysis

In this section, we first quantify each user label privacy leakage PL_r by her multi-round reported data (i.e., Theorem 5). To bound PL_r within e^ϵ, we then prove the conditions that must be satisfied (i.e., Theorem 6), which finally leads to an allocation of privacy budget over multiple rounds.

Theorem 5. *During the training phase of CART, given user selected attribute combinations s^1 (i.e., SL_1), $\cdots$, s^r, the total label privacy leakage across r rounds is quantified by*

$$\mathrm{PL}_r \leq 1 - \omega_{s^1} + \omega_{s^r} \prod_{r'=1}^{r} \theta_{r'} + \sum_{r'=1}^{r-1} (\omega_{s^{r'}} - \omega_{s^{r'+1}}) \prod_{r''=1}^{r'} \theta_{r''} \tag{13}$$

where $\theta_\gamma = (e^{\epsilon_\gamma} + \omega_{s^\gamma} - 1)/\omega_{s^\gamma}$, $\gamma \in [r]$.

Proof. Let a denote the sampled attribute in round r and $s^r = s^{r-1} \cup a$. We assume that a is different from the attributes in s^{r-1}. For any two label values, $l, l' \in L$ and any perturbed output $\tilde{v}^1, \cdots, \tilde{v}^r$ across r rounds, by Def. 4, we have,

$$\mathrm{PL}_r = \sup_{\tilde{v}^1, \cdots, \tilde{v}^r, l, l'} \frac{\Pr\{\tilde{v}^1, \cdots, \tilde{v}^r | l\}}{\Pr\{\tilde{v}^1, \cdots, \tilde{v}^r | l'\}}$$
$$= \sup_{\tilde{v}^1, \cdots, \tilde{v}^r, l, l'} \frac{\sum_{s_v^r} \Pr\{\tilde{v}^1, \cdots, \tilde{v}^r | s_v^r, l\} \Pr\{s_v^r | l\}}{\sum_{s_v^r} \Pr\{\tilde{v}^1, \cdots, \tilde{v}^r | s_v^r, l'\} \Pr\{s_v^r | l'\}}.$$

where s_v^r is a value for the attribute combination s^r. Let $\tilde{vs}^{r-1}$ denote the set of $\{\tilde{v}^1, \cdots, \tilde{v}^{r-1}\}$. Since $\tilde{v}^r$ is independent of $\tilde{vs}^{r-1}$, the above equation can be written as,

$$\mathrm{PL}_r$$
$$= \sup_{\tilde{v}^1, \cdots, \tilde{v}^r, l, l'} \frac{\sum_{s_v^r} \Pr\{\tilde{vs}^{r-1} | s_v^{r-1}, l\} \Pr\{\tilde{v}^r | s_v^r, l\} \Pr\{s_v^r | l\}}{\sum_{s_v^r} \Pr\{\tilde{vs}^{r-1} | s_v^{r-1}, l'\} \Pr\{\tilde{v}^r | s_v^r, l'\} \Pr\{s_v^r | l'\}}$$
$$= \sup_{\substack{\tilde{v}^1, \cdots, \tilde{v}^r \\ l, l'}} \frac{\sum_{s_v^{r-1}} \Pr\{\tilde{vs}^{r-1} | s_v^{r-1}, l\} \sum_{a_v} \Pr\{\tilde{v}^r | s_v^r, l\} \Pr\{s_v^r | l\}}{\sum_{s_v^{r-1}} \Pr\{\tilde{vs}^{r-1} | s_v^{r-1}, l'\} \sum_{a_v} \Pr\{\tilde{v}^r | s_v^r, l'\} \Pr\{s_v^r | l'\}}.$$

where s_v^{r-1} (resp. a_v) is the corresponding value for attributes s^{r-1} (resp. a) given s_v^r.

For simplicity, we let $k=1$ in kHR, so $p=\frac{e^{\epsilon_r}+\omega_{s^r}-1}{e^{\epsilon_r}+m^r\omega_{s^r}-1}$, and $\Pr\{\tilde{v}^r|s_v^r,l\}\in\{p,\frac{1-p}{m^r-1}\}$ where m^r is the domain size for s^r. Without loss of generality, we assume that the $x^l_{s^r_{v*}}$-bit (corresponding to (l,s^r_{v*})) is the only "1" in $\tilde{v}^r$. Then let both numerator and denominator of the equation PL_r divides by the value of $\Pr\{\tilde{v}^r|s_v^r,l'\}$ (i.e., $\frac{1-p}{m^r-1}$) , and the following is achieved, i.e.,

$$\mathrm{PL}_r \le \sup_{\tilde{v}^1,\cdot,\tilde{v}^{r-1},l,l'} \frac{\sum_{s_v^{r-1}}\Pr\{\tilde{vs}^{r-1}|s_v^{r-1},l\}\sum_{a_v}\Pr\{s_v^r|l\}}{\sum_{s_v^{r-1}}\Pr\{\tilde{vs}^{r-1}|s_v^{r-1},l'\}\sum_{a_v}\Pr\{s_v^r|l'\}} + \frac{(\frac{p(m^r-1)}{1-p}-1)\cdot\Pr\{\tilde{vs}^{r-1}|s_{v*}^{r-1},l\}\Pr\{s_{v*}^r|l\}}{\sum_{s_v^{r-1}}\Pr\{\tilde{vs}^{r-1}|s_v^{r-1},l'\}\sum_{a_v}\Pr\{s_v^r|l'\}}.$$

With $\sum_{a_v}\Pr\{s_v^r|l\}=\Pr\{s_v^{r-1}|l\}$, we get,

$$\mathrm{PL}_r \le \sup_{\substack{\tilde{v}^1,\cdots,\tilde{v}^{r-1}\\ l,l'}} \frac{\sum_{s_v^{r-1}}\Pr\{\tilde{vs}^{r-1}|s_v^{r-1},l\}\Pr\{s_v^{r-1}|l\}}{\sum_{s_v^{r-1}}\Pr\{\tilde{vs}^{r-1}|s_v^{r-1},l'\}\Pr\{s_v^{r-1}|l'\}} + \frac{(\frac{p(m^r-1)}{1-p}-1)\cdot\Pr\{\tilde{vs}^{r-1}|s_{v*}^{r-1},l\}\Pr\{s_{v*}^r|l\}}{\sum_{s_v^{r-1}}\Pr\{\tilde{vs}^{r-1}|s_v^{r-1},l'\}\Pr\{s_v^{r-1}|l'\}}.$$

For the above inequality, in the right, the first term is PL_{r-1}. Since the noisy vectors in $\tilde{vs}^{r-1}$ are independent given s_v^{r-1},

$$\prod_{r'=1}^{r-1}\frac{1-p}{m^{r'}-1}\le\Pr\{\tilde{vs}^{r-1}|s_v^{r-1},l\}\le\prod_{r'=1}^{r-1}p$$

and

$$\frac{\Pr\{\tilde{vs}^{r-1}|s_{v*}^{r-1},l\}}{\sum_{s_v^{r-1}}\Pr\{\tilde{vs}^{r-1}|s_v^{r-1},l'\}\Pr\{s_v^{r-1}|l'\}}\le\prod_{r'=1}^{r-1}\frac{p(m^{r'}-1)}{1-p}.$$

Let $\theta_r=\frac{p(m^r-1)}{1-p}=\frac{e^{\epsilon_r}+\omega_{s^r}-1}{\omega_{s^r}}$, and due to $\Pr\{s_{v*}^r|l\}\le\omega_{s^r}$, we have,

$$\begin{aligned}\mathrm{PL}_r \le& \mathrm{PL}_{r-1}+(\theta_r-1)\cdot\prod_{r'=1}^{r-1}\theta_{r'}\cdot\omega_{s^r}\\ =&\mathrm{PL}_{r-1}+\omega_{s^r}\cdot\prod_{r'=1}^{r}\theta_{r'}-\omega_{s^r}\cdot\prod_{r'=1}^{r-1}\theta_{r'}.\end{aligned}$$

From $\mathrm{PL}_1=\omega_{s^1}\cdot\theta_1+1-\omega_{s^1}$ (Theorem 2), the following inequality can be achieved recursively, i.e.,

$$\mathrm{PL}_r\le 1-\omega_{s^1}+\omega_{s^r}\cdot\prod_{r'=1}^{r}\theta_{r'}+\sum_{r'=1}^{r-1}(\omega_{s^{r'}}-\omega_{s^{r'+1}})\cdot\prod_{r''=1}^{r'}\theta_{r''}.$$

□

To provide a strong label privacy guarantee, we should assure that the label privacy leakage PL_r is always bounded by e^ϵ, namely,

$$\mathrm{PL}_r\le e^\epsilon,\quad\forall\, r\le dep. \tag{14}$$

When $r=1$, $\mathrm{PL}_1=e^{\epsilon_1}$, which is less than e^ϵ for $\epsilon_1\le\epsilon$. In the case of $r>1$, Theorem 6 demonstrates the conditions that should be satisfied to bound the label privacy leakage within e^ϵ.

Theorem 6. *For any round $r>1$, let $\epsilon^{re}=\epsilon-\log\mathrm{PL}_{r-1}$ denote the residual privacy budget, if $\omega_{s^r}\le\omega_{s^{r-1}}$ and $e^{\epsilon_r}\le\omega_{s^r}\cdot e^{\epsilon^{re}}+1-\omega_{s^r}$, the label privacy leakage PL_r across r rounds is bounded by e^ϵ, i.e., $\mathrm{PL}_r\le e^\epsilon$.*

Proof. Let $\triangle\mathrm{PL}_r$ denote the difference between PL_r and PL_{r-1}, i.e., $\triangle\mathrm{PL}_r=\mathrm{PL}_r-\mathrm{PL}_{r-1}=\omega_{s^r}\cdot(\theta_r-1)\cdot\prod_{r'=1}^{r-1}\theta_{r'}$. If $\omega_{s^r}\le\omega_{s^{r-1}}$, we get,

$$\begin{aligned}\triangle\mathrm{PL}_r &\le \omega_{s^{r-1}}\cdot(\theta_r-1)\cdot\prod_{r'=1}^{r-1}\theta_{r'}\\ &=(\triangle\mathrm{PL}_{r-1}+\omega_{s^{r-1}}\cdot\prod_{r'=1}^{r-2}\theta_{r'})\cdot(\theta_r-1).\end{aligned}$$

If $\omega_{s^{r-1}}\le\omega_{s^{r-2}}\le\cdots\le\omega_{s^1}$, we have,

$$\begin{aligned}\triangle\mathrm{PL}_r &\le (\triangle\mathrm{PL}_{r-1}+\omega_{s^{r-2}}\cdot\prod_{r'=1}^{r-2}\theta_{r'})\cdot(\theta_r-1)\\ &\le(\triangle\mathrm{PL}_{r-1}+\mathrm{PL}_{r-2})\cdot(\theta_r-1)\quad\text{(by Theorem 5)}\\ &=\mathrm{PL}_{r-1}\cdot(\theta_r-1).\end{aligned}$$

That is, $\mathrm{PL}_{r-1}+\triangle\mathrm{PL}_r\le\mathrm{PL}_{r-1}\cdot\theta_r$, i.e., $\mathrm{PL}_r\le\mathrm{PL}_{r-1}\cdot\theta_r$. Then we can achieve $\mathrm{PL}_r\le e^\epsilon$ when $\theta_r\le\frac{e^\epsilon}{\mathrm{PL}_{r-1}}=e^{\epsilon^{re}}$, i.e., $e^{\epsilon_r}\le\omega_{s^r}\cdot e^{\epsilon^{re}}+1-\omega_{s^r}$. □

To satisfy the conditions of $\omega_{s^r}\le\omega_{s^{r-1}}$ in Theorem 6, the k in kHR should be set to the same value over all-round data collection (due to $s^{r-1}\subseteq s^r$). We use a simple example to illustrate it. When k is fixed to 1, if $s^1=\{u\},s^2=\{uv\}$, $\omega_{s^1}=\max\Pr\{u|l\}\le\max\Pr\{uv|l\}=\omega_{s^2}$. On the other hand, to satisfy $e^{\epsilon_r}\le\omega_{s^r}\cdot e^{\epsilon^{re}}+1-\omega_{s^r}$, the privacy budget ϵ_r in each round is set by the following strategy:

$$\epsilon_r=\begin{cases}\frac{\epsilon}{dep}, & r=1\\ \min\{\frac{\epsilon^{re}}{dep-r+1},\ g(\epsilon^{re})\}, & r>1\end{cases} \tag{15}$$

where $\epsilon^{re}=\epsilon-\log\mathrm{PL}_{r-1}$ and $g(\epsilon^{re})=\log(\omega_{s^r}\cdot e^{\epsilon^{re}}+1-\omega_{s^r})$. The following theorem establishes the CLLDP guarantee of the proposed scheme for decision tree.

Theorem 7. *The proposed scheme to construct CART satisfies ϵ-CLLDP.*

Proof. Let $\mathcal{M}_{\mathrm{DT}}$ denote the user-side perturbation. Without loss of generality, we assume that private CART is achieved by r^* rounds, $r^*\le dep$. For any two label values $l,l'\in L$, and any output $\tilde{V}=\{\tilde{v}_1,\cdots,\tilde{v}_{r^*}\}$, $G=\{g_1,\cdots,g_{r^*}\}$ from $\mathcal{M}_{\mathrm{DT}}$ across r^* rounds, since G is sampled independently, we have,

$$\begin{aligned}&\frac{\Pr\{\mathcal{M}_{\mathrm{DT}}(\mathcal{F}(l),l)=(\tilde{V},G)|\omega\}}{\Pr\{\mathcal{M}_{\mathrm{DT}}(\mathcal{F}(l'),l')=(\tilde{V},G)|\omega\}}\\ =&\frac{\Pr\{\tilde{V}|l\}}{\Pr\{\tilde{V}|l'\}}\cdot\frac{\Pr\{G\}}{\Pr\{G\}}=\frac{\Pr\{\tilde{V}|l\}}{\Pr\{\tilde{V}|l'\}}\le\mathrm{PL}_{r^*}.\end{aligned}$$

By Theorem 6, when k is fixed during the data collection and privacy budget is assigned by Eq. (15), we can achieve $\mathrm{PL}_{r^*}\le e^\epsilon$, so the proposed scheme is ϵ-CLLDP. □

6.3 Overall Construction

Algorithm 4 summarizes the overall scheme for decision tree. In initialization, SL_0 and T_0 are set to the empty set, and the label privacy leakage PL_0 for each user is set to 1 (Line 3). In each round r, user i randomly samples an

Algorithm 4 Decision Tree under CLLDP

Input: Dataset D, adversarial belief ω, ϵ, k, depth dep.
Output: The private decision tree T_r

1: **for** $r = 1$ to dep **do**
// Initialization:
2: **if** $r = 1$ **then**
3: $\mathrm{SL}_0 = \emptyset$, $T_0 = \emptyset$, $\mathrm{PL}_0 = 1$
// Users side:
4: **for** each user i **do**
5: Get a candidate set SC with SL_{r-1} and T_{r-1}
6: Sample $a \leftarrow \mathrm{SC}$ *// $\mathrm{SC} = A$ if $r = 1$*
7: $\mathrm{SL}_r = a \cup \mathrm{SL}_{r-1}$ *// multiset operation*
8: Gain s^r by removing duplicates in SL_r
9: $S^r = Cartesian\ (s^r)$ *// Cartesian product*
10: **if** $s^r \neq s^{r-1}$ **then**
11: $\tilde{v}_{i,r} = \mathcal{M}_{k\mathrm{HR}}(d_i^S, l_i, \epsilon_r, \omega_{s^r}, k)$
12: Get PL_r by Eq. (13) and update $\epsilon^{re} = \epsilon - \log \mathrm{PL}_r$
13: **Report** $\tilde{v}_{i,r}$ and group index $g_{i,r}$
// Collector side:
14: **for** each user group g **do**
15: $\Pr\{s^r, L\} = \Phi_{k\mathrm{HR}}(\{\tilde{v}_{i,r} | g_{i,r} = g\})$
16: Construct T_r based on T_{r-1} by Eq. (11) or Eq. (12)
17: **if** all leaves of T_r are labeled **then**
18: **Return** the CART T_r

attribute a from the candidate set SC and insert a into SL_r (Lines 5-7). By removing the duplicates in SL_r, user i gains an attribute combination s^r and the Cartesian product S^r over attributes in s^r (Lines 7-9). If $s^r \neq s^{r-1}$, user i sanitizes her values d_i^S, l_i by kHR, update her remaining privacy budget ϵ^{re} and reports her noisy data $\tilde{v}_{i,r}$ with the group index $g_{i,r}$ (Lines 10-13). At the collector side, based on the perturbed data from each user group, the collector estimate the joint distribution $\Pr\{s^r, L\}$ by the calibration (Line 15). With the estimated distribution, the collector constructs the tree T_r based on T_{r-1} (Line 16). If all leaf nodes are labeled (i.e., stop), the construction is completed and T_r is returned (Lines 17-18).

7 EXPERIMENTS

In this section, we conduct experiments to verify the effectiveness of the proposed CLLDP perturbation protocol kHR, and show its performance in two machine learning models, namely, Naïve Bayes classifier and decision tree.

7.1 Experimental Setup

Datasets. We use four synthetic datasets and three public datasets in the experiments. The synthetic datasets are used to investigate the impact of correlations between attributes and the label. Since multiple attributes can be reduced to one attribute by Cartesian product, we only study one-attribute case to reduce the control parameters. Unless otherwise stated, continuous values are discretized to 10 categorical values.

- *Synthetic datasets*. Each of the four synthetic datasets contains $100,000$ records. The attribute has 50 different values and the label has 5 classes, i.e., $|L| = 5$. In these datasets, label values follow a uniform distribution $\mathrm{unif}(1, 5)$ while the attribute values of the first four classes all follow $\mathrm{unif}(1, 50)$. In other words, the four datasets only differ in the attribute value distribution of the fifth class: $\mathrm{U}(1, 50)$ follows a uniform distribution $\mathrm{unif}(1, 50)$ (i.e., same as the first four), $\mathrm{N}(25, 1)$ follows a normal distribution with mean 25 and variance 1, $\mathrm{N}(25, 0.3)$ follows a normal distribution with mean 25 and variance 0.3, and $\mathrm{D}(25)$ follows a constant value 25.

TABLE 4
Adversarial belief (ω_s) at $k = 1$.

Datasets	$\mathrm{U}(1, 50)$	$\mathrm{N}(25, 1)$	$\mathrm{N}(25, 0.3)$	$\mathrm{D}(25)$
ω_s	0.020	0.383	0.638	1

- *Adult*.[5] This is a real dataset extracted from the 1994 Census bureau database. It contains $32,561$ records with 14 attributes and 2 classes.
- *SmartGrid*.[6] This is a public synthetic dataset to simulate the electrical grid stability. It contains $60,000$ records with 13 attributes and 2 classes.
- *Healthcare*.[7] This is a real dataset containing the basic information of $318,438$ patients with 16 attributes. We regard the last attribute 'Stay' (the length of stay of each patient) as the label and divide it into two classes, namely, ≤ 30 days or > 30 days.

Experiments Design. With the above datasets, we design four sets of experiments. The first one evaluates the joint estimation under CLLDP in synthetic datasets; the second investigates the machine learning tasks under CLLDP in public datasets; the third one compares the privacy loss of CLLDP and LabelDP; and the last one verifies the correctness of variance analysis and the optimal k selection.

All algorithms are implemented in MATLAB, and the experiments are conducted on a desktop computer with Intel Core i7-10700 2.9Ghz CPU and 72GB RAM.

Performance Metrics. (1) To verify the effectiveness of CLLDP-based joint estimation protocols, we use $\mathbf{L_2}$ **norm** (i.e., square error, denoted by err) between the true distribution and the estimated one. Formally,

$$\mathrm{err} = \sqrt{\sum\nolimits_{x=1}^{m} |\hat{f}_x - f_x|^2},$$

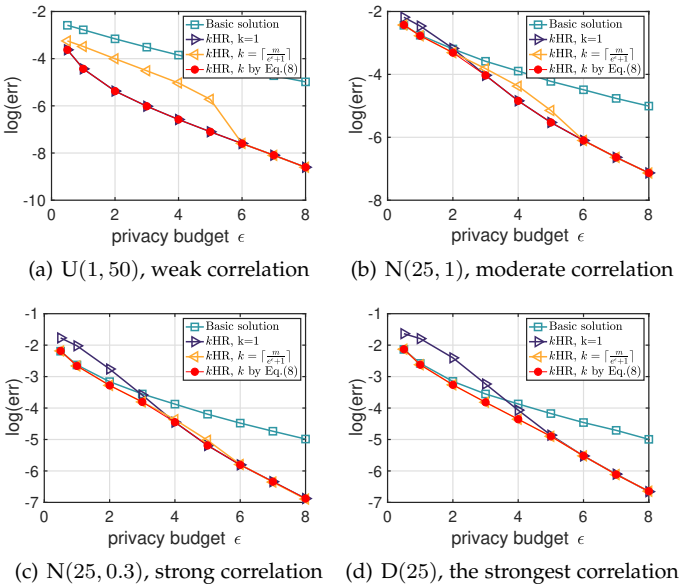
where m is the number of different values in the domain and each $\hat{f}_x$ (resp. f_x) denotes the estimated (resp. true) probability of the x-th value.

(2) We use the **classification accuracy** to evaluate the performance of machine learning models. Given a test dataset, we use the machine learning model trained from CLLDP perturbed data to predict the label of each instance in the test dataset and measure the percentage of correct predictions.

7.2 Joint Distribution Estimation

In this subsection, we evaluate the performance of the proposed CLLDP protocol kHR on joint estimation in four synthetic datasets.

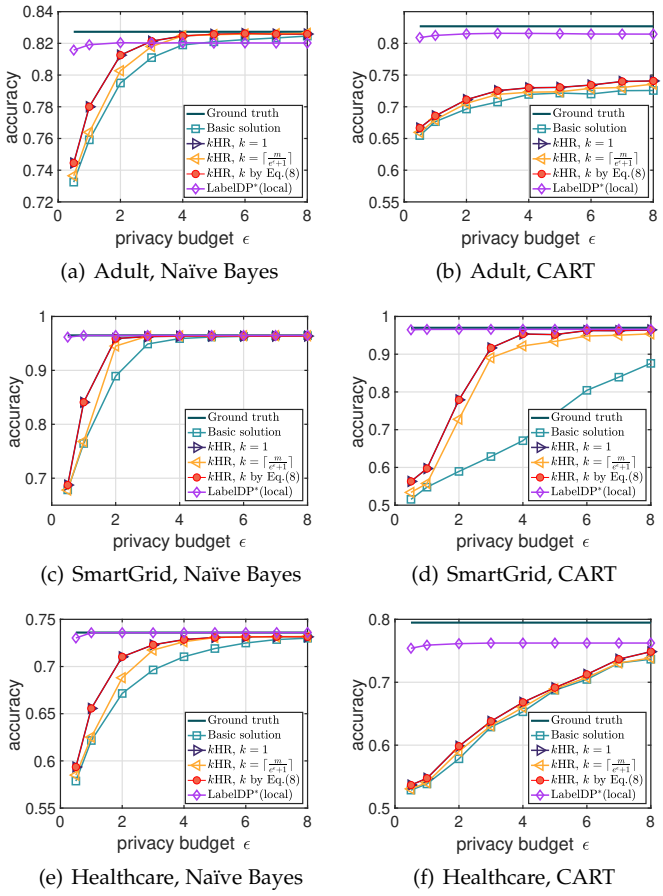
5. https://archive.ics.uci.edu/ml/datasets/Adult
6. https://www.kaggle.com/pcbreviglieri/smart-grid-stability
7. https://www.kaggle.com/nehaprabhavalkar/av-healthcare-analytics-ii



(a) U$(1, 50)$, weak correlation (b) N$(25, 1)$, moderate correlation

(c) N$(25, 0.3)$, strong correlation (d) D(25), the strongest correlation

Fig. 3. CLLDP protocols on joint estimation.



(a) Adult, Naïve Bayes (b) Adult, CART

(c) SmartGrid, Naïve Bayes (d) SmartGrid, CART

(e) Healthcare, Naïve Bayes (f) Healthcare, CART

Fig. 4. CLLDP protocols for Naïve Bayes and decision tree classification.

The setting of ω_s. In synthetic datasets, we have the knowledge of conditional distributions give labels, so the adversarial belief ω_s can be achieved in advance. Table 4 shows the values of ω_s on four datasets at $k = 1$, which indicates that D(25) has the strongest correlation between the attribute and label, while U$(1, 50)$ has the weakest correlation.

Fig. 3 compares the basic solution and the designed kHR ($k = 1, \lceil\frac{m}{e^\epsilon+1}\rceil$ [8] and the k from Eq. (8)) on different datasets by varying privacy budget from 0.5 to 8. We observe that kHR with the k selected by Eq. (8) always achieves the lowest estimation error, which verifies the effectiveness of our k selection strategy. The basic solution must set ω_s to the maximum value 1, which leads to a large perturbation, thus degrading the data utility. In addition, when the correlation is weak (i.e., U$(1, 50)$), the optimal k always lies in the value 1; while when the correlation becomes strong, the optimal k tends to select $\lceil\frac{m}{e^\epsilon+1}\rceil$ under a small ϵ and choose 1 under a large ϵ. In U$(1, 50)$, for any $k \leq \lceil\frac{m}{e^\epsilon+1}\rceil$, ω_s is always less than 1, which makes the variance increase with the growth of k (Theorem 3), so the estimation error is minimized at $k = 1$. Under a strong correlation, ω_s quickly increases to 1 with k. Once $\omega_s = 1$, the variance will decrease as k grows until k reaches to $\lceil\frac{m}{e^\epsilon+1}\rceil$. When ϵ is small, $\lceil\frac{m}{e^\epsilon+1}\rceil$ is large and would bring a big drop in variance. As such, the optimal k tends to choose the large value $\lceil\frac{m}{e^\epsilon+1}\rceil$ under a strong correlation with a small ϵ. Moreover, due to $\lceil\frac{m}{e^\epsilon+1}\rceil = 1$ with a large ϵ, the two curves of $k = 1$ and $k = \lceil\frac{m}{e^\epsilon+1}\rceil$ can coincide in the low privacy region.

7.3 Machine Learning Tasks

We evaluate the performance of the proposed protocols for Naïve Bayes and decision tree classification tasks on three public datasets. For each dataset, we randomly sample 70% of user data for training models and the rest (30%) for testing. For the decision tree, the maximal depth (dep) of CART is set to 5. When applying kHR to decision tree classification, for privacy guarantee (Theorem 6), k is set to 1, $\lceil\frac{m}{e^\epsilon+1}\rceil$ [9] or by Eq. (8) with the selected data in the first round, and do not change in the following rounds.

The setting of ω_s. In machine learning tasks, we sample 10% user data from the training dataset and employ an LDP scheme – subset selection (SS) [10] to estimate the conditional distribution $\mathcal{F}$ of each attribute given the label. With $\mathcal{F}$, we then set a proper ω_{a^c} between each attribute a^c and the label, which is required in the Naïve Bayes and the first round of decision tree. About other rounds in decision tree, the required ω_{s^r} can be approximated by $\omega_{s^{r-1}}$ achieved from the estimated marginal distribution of s^{r-1} in the previous round.

Fig. 4 shows the comparison results for these two tasks varying the privacy budget from 0.5 to 8. kHR with the k selected by Eq. (8) has the same best performance as $k = 1$. This is because the value of ω_s is often less than 1 in the three public datasets and the estimation error can be minimized at $k = 1$. Besides, we observe that the advantage of kHR is not prominent for the decision tree classification on Adult and Healthcare datasets. The main reason is that, on the decision

8. In synthetic datasets, $m = 50 \times 5 = 250$.

9. m is the domain size of Cartesian product of the selected attribute and labels.

10. SS is a highly flexible scheme under LDP. When the domain size is small and ϵ is large, SS simplifies to DE and offers better utility [9]; in the other cases, the performance of SS is similar to OUE, which can achieve the best accuracy in high privacy regimes under a large domain [9].

tree task, kHR cannot make full use of privacy budget. By Eq. (15), we can obtain $\epsilon_r \leq \epsilon^{re}$ where ϵ_r is the consumed privacy budget in round r and ϵ^{re} is the remaining budget. So in the last (dep-th) round, the remaining privacy budget ϵ^{re} cannot be totally consumed unless $\epsilon_{dep} = \epsilon^{re}$ holds when ω_{s^r} is always 1, i.e., the setting of the basic solution. As such, the advantage of kHR is impaired. SmartGrid is a special dataset where the ground-truth tree almost only depends on one attribute. Hence, the accuracy of the joint distribution of this attribute and the label becomes critical to the performance of a trained decision tree. This situation favors kHR, so in the first round it can achieve better accuracy than the baisc solution on this joint estimation with a smaller ω_{s^1}.

The accuracy of LabelDP is generally higher than that of CLLDP, mainly because the former only perturbs on labels. Since attribute values are original, the joint distribution between attributes and labels can be estimated more accurately, leading to high classification accuracy. However, as we have highlighted in the motivation of CLLDP, LabelDP achieves this at the cost of severe privacy leakage on labels, which will be shown in the next subsection.

7.4 Privacy Loss under LabelDP and CLLDP

In the following, we simulate a Bayesian attack to evaluate the privacy loss under both LabelDP and CLLDP models. Specifically, we train two Naïve Bayes models, one using the dataset perturbed by LabelDP where only labels are perturbed by SS and attributes are true, and the other using the dataset perturbed by CLLDP where both labels and attributes are perturbed by kHR. The key information in Naïve Bayes is the probabilities of each attribute conditioning on the label (see Eq. (9)), which can be estimated from the perturbed datasets and used to build classifiers. Once the models are trained, they are then applied back to predict labels from original attribute values (i.e., LabelDP) and perturbed attribute values (i.e., CLLDP). The percentage of correct predictions is used as the metric for privacy loss. The experimental results are plotted in Fig. 5, with the privacy budget ranging from 0.5 to 8. We observe that, even with a small $\epsilon = 0.5$, more than 70% of ground-truth labels are correctly predicted by the trained model with LabelDP (i.e., purple lines). This percentage even reaches 96% in SmartGrid. By contrast, the accuracy of CLLDP predictions (i.e., red lines) from noisy attributes is close to a random guess, especially when ϵ is small, e.g., less than 60% at $\epsilon = 0.5$. Therefore, we can conclude that CLLDP achieves much higher privacy guarantee than LabelDP, as the latter only perturbs labels.

7.5 Verification of the optimal k setting

Finally, we conduct experiments to verify the optimal k setting in Sec. 4.3. Due to space limitation, we mainly present the joint estimation on four synthetic datasets by varying k from 1 to 100 under a fixed privacy budget $\epsilon = 3$. Fig. 6 plots the experimental results, where the k selected by Eq. 8 can enjoy the lowest estimation error, which indicates the correctness of our optimal k setting. Besides, these results also verify the analysis of the variance in Theorem 3. Recall that with the increase of k, the estimation variance rises

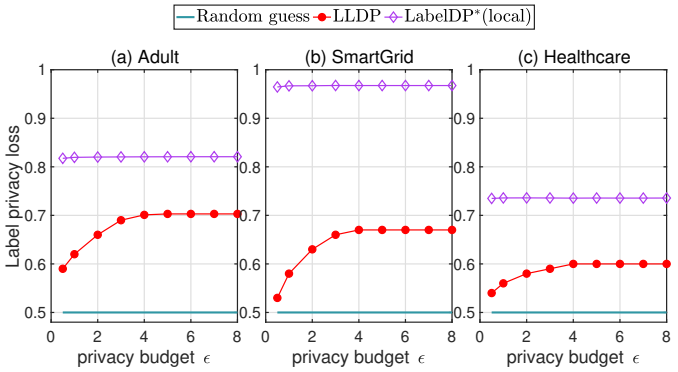
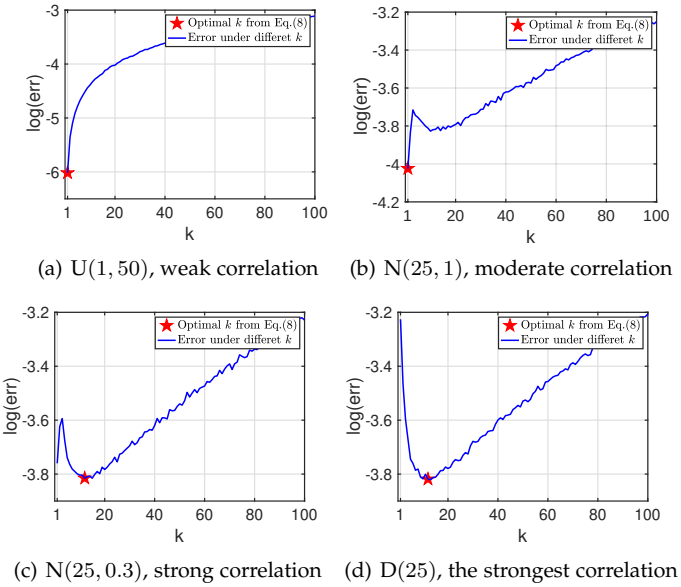


Fig. 5. Label privacy loss under CLLDP and LabelDP



(a) $\mathrm{U}(1, 50)$, weak correlation (b) $\mathrm{N}(25, 1)$, moderate correlation

(c) $\mathrm{N}(25, 0.3)$, strong correlation (d) $\mathrm{D}(25)$, the strongest correlation

Fig. 6. Estimation error with various k values under $\epsilon = 3$.

when $\omega_s < 1$, and when $\omega_s = 1$, the variance decreases until $k = \lceil \frac{m}{e^\epsilon+1} \rceil$ and then increases. In the dataset $\mathrm{U}(1, 50)$, since $\omega_s < 1$ can always hold when $k \leq \lceil \frac{m}{e^\epsilon+1} \rceil = 12$ ($m = 250$, $\epsilon = 3$), the error increases monotonically in Fig. 6(a). In both datasets $\mathrm{N}(25, 1)$ and $\mathrm{N}(25, 0.3)$, $\omega_s < 1$ at $k = 1$ but it quickly increases to 1 as k increases, so their errors first increase, then decrease, and increase again as shown in Figs. 6(b)-6(c). In the dataset $\mathrm{D}(25)$, ω_s is always 1, so the error first decreases and then increases in Fig 6(d). Overall, these experimental results are consistent to our theoretical analysis.

8 Related Work

Under the traditional DP model [2], a trusted data collector can manage personal data of individuals, and generate statistic information with differential privacy guarantee. With the request of untrusted data curators, LDP [1] is proposed, and followed by numerous work on various computation tasks, such as distribution estimation [14], [21], [22], [23], [24], [25], [26], [27], [28], [29], mean value calculation [3], [30], [31], [32], time-series data collection [18], [33], [34], [35] and graph data analysis [36], [37]. Particularly, [26], [27]

propose a RS+FD solution that achieves lower estimation variance. In the training phase of CART, multiple rounds are performed and many marginal distributions are required. If the RS+FD strategy is exploited, all users must report data for every required marginal, which results in an unaffordable communication cost. Hence we choose the simple sampling strategy when training models for efficiency.

To tackle with the problem of correlated data privacy leakage, [38] proposes the *Pufferfish* framework, a generalization of differential privacy. *Pufferfish* consists of three components — $\mathcal{S}$, a set of secrets that need to be protected; $\mathcal{Q}$, a set of secret pairs that need to be indistinguishable; and Θ, a class of distributions that represent the data correlations. The privacy framework requires that the secret pairs in $\mathcal{Q}$ should be indistinguishable when data are generated from any $\theta \in \Theta$. In Pufferfish, by using different models to describe the correlations between data tuples, various privacy definitions and mechanisms are developed, e.g., Bayesian differential privacy [11] with Gaussian Markov random field; dependent differential privacy [39] with a dependence coefficient; and Markov Quilt mechanism [12] with a Bayesian network. These schemes focus on the correlations between different user data, i.e., horizontal correlation, and study the privacy leakage in a centralized setting, which is different from our investigation into the correlations between attributes and the label, i.e., vertical correlation, in a local setting. Hence, the above works are orthogonal to ours.

The methods [10], [40], [41], which explore temporal correlations in the context of time-series data release, are also not comparable to our work. [10], [40] use a transition matrix (i.e., Markov chain) to model the correlation between the values at two consecutive timestamps, where each value is from a discrete space; while [41] considers the time-series from a continuous space and uses a public constant to bound the difference between any two consecutive values over the time period. Our work exploits conditional distributions given labels to describe the correlation of labels with attributes.

[42], [43] utilize the correlations between attributes (i.e., vertical correlation) for data synthesis, where the data collector constructs a Bayesian network (or Junction tree) to explore the correlations, and synthesizes data with the constructed model and a set of low-dimensional noisy marginals. Although we all consider vertical correlation, our work differs from them in the following ways. First, [42], [43] employ the differential privacy model to protect the whole dataset in a centralized setting; while we design a new privacy model to protect the label information of each user in a local setting. Second, their works learn and utilize the vertical correlations to synthesize data; whereas we explore the privacy leakage caused by such correlations to provide a stronger label privacy guarantee. [14], [28] are two other studies on data synthesis under LDP, where joint distributions are directly derived from the (privacy-preserving) synthetic dataset. Our work focuses on developing mechanisms to estimate joint distributions from noisy data under CLLDP.

Our recent work [25] also considers vertical correlations and leverages them to relax LDP under certain assumptions, e.g., some attributes are independent. However, this paper differs from [25] in terms of correlation metrics, privacy models and practical tasks. Notably, we employ conditional probability to capture correlations and derive them from historical or noisy user data, whereas [25] uses correlation coefficients and simply assumes that this information is given; we mainly focus on label privacy whereas [25] wants to protect any one dimension; we aim to achieve multi-dimensional marginals as well as the downstream machine learning models whereas [25] is limited to single-dimensional frequency estimation.

In this work, we focus on correlations between attributes and the label and study the impact of the correlations on label privacy. The comparable work LabelDP [5], a variant of the traditional DP, is proposed to protect label information in the centralized setting. It can be directly extended to the local setting, where only labels need to be sanitized for the label privacy guarantee [7]. Also, [44] considers label-only privacy for machine learning, and propose two approaches PATE (Private Aggregation of Teacher Ensembles) and ALIBI (Additive Laplace with Iterative Bayesian Inference) to strike a good balance between empirical privacy loss and accuracy. [8] first notes the fact that even with the labelDP guarantee, an adversary can still recover the training labels by applying the trained model back to the public training features. [45] provides a different view about privacy guarantees of LabelDP. That is, LabelDP cannot protect against label inference attacks (LIAs); instead, it aims to limit the advantage of an LIA adversary compared to inferring training labels only with a Bayes classifier.

9 Conclusion

In this work, we study the label privacy issues in the local setting, by proposing a new privacy definition named correlation-aware label local differential privacy. CLLDP fully considers the correlation between attributes and labels, thus addressing the vulnerability of existing LabelDP work against correlation-aware adversaries. Based on CLLDP, we design a label privacy-preserving perturbation protocol for joint probabilistic distribution estimation, namely, kHR. To further show the usage of the designed protocol, we then apply it to machine learning models. Finally, through extensive experiments on both synthetic and real datasets, we verify the effectiveness of the proposed CLLDP model and the designed perturbation protocol.

About the future work, we plan to investigate mechanisms for other data types, e.g., continuous data, time-series data, under CLLDP. Intuitively, the proposed privacy notion can be extended to continuous data, where the correlations are characterized by the probability density function (instead of the probability mass function) of attributes conditioning on labels. Alternatively, we can discretize continuous data to categorical values [3], [33], and then apply the proposed protocol to collect them. Besides, we also would like to investigate more complex data analytical tasks (e.g., machine learning models) under CLLDP. Machine learning typically requires many training iterations, which would result in an assignment of the privacy budget, leading to the poor utility. To address this issue, data synthesis could be a promising method to achieve complex models under CLLDP. Specifically, we first acquire some marginals by

CLLDP methods (e.g., kHR), based on which, we synthesize data to replace the original dateset while preserving privacy. As a result, any model can be trained on the synthetic data under CLLDP guarantees.

ACKNOWLEDGMENTS

This work was supported by the National Natural Science Foundation of China (Grant No: 62372122, 92270123 and 62302214), and the Research Grants Council (Grant No: 15208923 and 25207224) and the Innovation and Technology Fund (Grant No: GHP/392/22GD), Hong Kong SAR, China.

REFERENCES

[1] S. P. Kasiviswanathan, H. K. Lee, K. Nissim, S. Raskhodnikova, and A. Smith, "What can we learn privately?" *SIAM Journal on Computing*, vol. 40, no. 3, pp. 793–826, 2011.

[2] C. Dwork, F. McSherry, K. Nissim, and A. Smith, "Calibrating noise to sensitivity in private data analysis," in *Theory of cryptography conference*. Springer, 2006, pp. 265–284.

[3] N. Wang, X. Xiao, Y. Yang, J. Zhao, S. C. Hui, H. Shin, J. Shin, and G. Yu, "Collecting and analyzing multidimensional data with local differential privacy," in *2019 IEEE 35th International Conference on Data Engineering*. IEEE, 2019, pp. 638–649.

[4] Q. Xue, Y. Zhu, and J. Wang, "Joint distribution estimation and naïve bayes classification under local differential privacy," *IEEE Transactions on Emerging Topics in Computing*, vol. 9, no. 4, pp. 2053–2063, 2021.

[5] K. Chaudhuri and D. Hsu, "Sample complexity bounds for differentially private learning," in *Proceedings of the 24th Annual Conference on Learning Theory*. JMLR Workshop and Conference Proceedings, 2011, pp. 155–186.

[6] A. Beimel, K. Nissim, and U. Stemmer, "Private learning and sanitization: Pure vs. approximate differential privacy," in *Approximation, Randomization, and Combinatorial Optimization. Algorithms and Techniques*. Springer, 2013, pp. 363–378.

[7] D. Wang and J. Xu, "On sparse linear regression in the local differential privacy model," in *International Conference on Machine Learning*. PMLR, 2019, pp. 6628–6637.

[8] R. I. Busa-Fekete, U. Syed, S. Vassilvitskii *et al.*, "On the pitfalls of label differential privacy," in *NeurIPS 2021 Workshop LatinX in AI*, 2021.

[9] T. Wang, J. Blocki, N. Li, and S. Jha, "Locally differentially private protocols for frequency estimation," in *26th USENIX Security Symposium*, 2017, pp. 729–745.

[10] Y. Cao, M. Yoshikawa, Y. Xiao, and L. Xiong, "Quantifying differential privacy under temporal correlations," in *2017 IEEE 33rd International Conference on Data Engineering*. IEEE, 2017, pp. 821–832.

[11] B. Yang, I. Sato, and H. Nakagawa, "Bayesian differential privacy on correlated data," in *Proceedings of the 2015 ACM SIGMOD international conference on Management of Data*, 2015, pp. 747–762.

[12] S. Song, Y. Wang, and K. Chaudhuri, "Pufferfish privacy mechanisms for correlated data," in *Proceedings of the 2017 ACM International Conference on Management of Data*, 2017, pp. 1291–1306.

[13] G. Liao, X. Chen, and J. Huang, "Social-aware privacy-preserving mechanism for correlated data," *IEEE/ACM Transactions on Networking*, vol. 28, no. 4, pp. 1671–1683, 2020.

[14] Z. Zhang, T. Wang, N. Li, S. He, and J. Chen, "CALM: Consistent adaptive local marginal for marginal release under local differential privacy," in *CCS*. ACM, 2018, pp. 212–229.

[15] T. Jebara, *Machine learning: discriminative and generative*. Springer Science & Business Media, 2012, vol. 755.

[16] S. Wang, L. Huang, Y. Nie, X. Zhang, P. Wang, H. Xu, and W. Yang, "Local differential private data aggregation for discrete distribution estimation," *IEEE Transactions on Parallel and Distributed Systems*, vol. 30, no. 9, pp. 2046–2059, 2019.

[17] P. Kairouz, S. Oh, and P. Viswanath, "Extremal mechanisms for local differential privacy," in *Advances in neural information processing systems*, 2014, pp. 2879–2887.

[18] 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*. ACM, 2014, pp. 1054–1067.

[19] M. Ye and A. Barg, "Optimal schemes for discrete distribution estimation under locally differential privacy," *IEEE Transactions on Information Theory*, vol. 64, no. 8, pp. 5662–5676, 2018.

[20] F. Islami, A. Goding Sauer, K. D. Miller, R. L. Siegel, S. A. Fedewa *et al.*, "Proportion and number of cancer cases and deaths attributable to potentially modifiable risk factors in the united states," *CA: a cancer journal for clinicians*, vol. 68, no. 1, pp. 31–54, 2018.

[21] 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 *the 2016 ACM SIGSAC Conference on Computer and Communications Security*. ACM, 2016, pp. 192–203.

[22] P. Kairouz, K. Bonawitz, and D. Ramage, "Discrete distribution estimation under local privacy," in *International Conference on Machine Learning*. PMLR, 2016, pp. 2436–2444.

[23] Z. Li, T. Wang, M. Lopuhaä-Zwakenberg, N. Li, and B. Škoric, "Estimating numerical distributions under local differential privacy," in *SIGMOD*, 2020, pp. 621–635.

[24] G. Cormode, T. Kulkarni, and D. Srivastava, "Marginal release under local differential privacy," in *SIGMOD*. ACM, 2018, pp. 131–146.

[25] R. Du, Q. Ye, Y. Fu, and H. Hu, "Collecting high-dimensional and correlation-constrained data with local differential privacy," in *2021 18th Annual IEEE International Conference on Sensing, Communication, and Networking (SECON)*. IEEE, 2021, pp. 1–9.

[26] H. H. Arcolezi, J.-F. Couchot, B. Al Bouna, and X. Xiao, "Random sampling plus fake data: Multidimensional frequency estimates with local differential privacy," in *Proceedings of the 30th ACM International Conference on Information & Knowledge Management*, 2021, pp. 47–57.

[27] G. Varma, R. Chauhan, and D. Singh, "Sarve: synthetic data and local differential privacy for private frequency estimation," *Cybersecurity*, vol. 5, no. 1, p. 26, 2022.

[28] G. Liu, P. Tang, C. Hu, C. Jin, S. Guo, J. Stoyanovich, J. Teubner, N. Mamoulis, E. Pitoura, and J. Mühlig, "Multi-dimensional data publishing with local differential privacy." in *EDBT*, 2023, pp. 183–194.

[29] H. H. Arcolezi, J.-F. Couchot, B. Al Bouna, and X. Xiao, "Improving the utility of locally differentially private protocols for longitudinal and multidimensional frequency estimates," *Digital Communications and Networks*, vol. 10, no. 2, pp. 369–379, 2024.

[30] J. C. Duchi, M. I. Jordan, and M. J. Wainwright, "Minimax optimal procedures for locally private estimation," *Journal of the American Statistical Association*, vol. 113, no. 521, pp. 182–201, 2018.

[31] Q. Ye, H. Hu, X. Meng, H. Zheng, K. Huang, C. Fang, and J. Shi, "PrivKVM*: Revisiting key-value statistics estimation with local differential privacy," *IEEE Transactions on Dependable and Secure Computing*, 2021.

[32] X. Gu, M. Li, Y. Cheng, L. Xiong, and Y. Cao, "PCKV: locally differentially private correlated key-value data collection with optimized utility," in *USENIX Security Symposium*, 2020.

[33] B. Ding, J. Kulkarni, and S. Yekhanin, "Collecting telemetry data privately," in *Proceedings of the 31st International Conference on Neural Information Processing Systems*, 2017, p. 3574–3583.

[34] Q. Ye, H. Hu, N. Li, X. Meng, H. Zheng, and H. Yan, "Beyond value perturbation: Local differential privacy in the temporal setting," in *INFOCOM*. IEEE, 2021.

[35] Q. Xue, Q. Ye, H. Hu, Y. Zhu, and J. Wang, "DDRM: A continual frequency estimation mechanism with local differential privacy," *IEEE Transactions on Knowledge and Data Engineering*, 2022.

[36] H. Sun, X. Xiao, I. Khalil, Y. Yang, Z. Qin, H. W. Wang, and T. Yu, "Analyzing subgraph statistics from extended local views with decentralized differential privacy," in *CCS*. ACM, 2019, pp. 703–717.

[37] Q. Ye, H. Hu, M. H. Au, X. Meng, and X. Xiao, "LF-GDPR:graph metric estimation with local differential privacy," *IEEE Transactions on Knowledge and Data Engineering (TKDE)*, 2020.

[38] D. Kifer and A. Machanavajjhala, "Pufferfish: A framework for mathematical privacy definitions," *ACM Transactions on Database Systems (TODS)*, vol. 39, no. 1, pp. 1–36, 2014.

[39] C. Liu, S. Chakraborty, and P. Mittal, "Dependence makes you vulnberable: Differential privacy under dependent tuples." in *NDSS*, vol. 16, 2016, pp. 21–24.

[40] Y. Xiao and L. Xiong, "Protecting locations with differential privacy under temporal correlations," in *Proceedings of the 22nd ACM SIGSAC Conference on Computer and Communications Security*, 2015, pp. 1298–1309.

[41] E. Bao, Y. Yang, X. Xiao, and B. Ding, "Cgm: an enhanced mechanism for streaming data collection with local differential privacy," *Proceedings of the VLDB Endowment*, vol. 14, no. 11, pp. 2258–2270, 2021.
[42] J. Zhang, G. Cormode, C. M. Procopiuc, D. Srivastava, and X. Xiao, "Privbayes: Private data release via bayesian networks," *ACM Transactions on Database Systems (TODS)*, vol. 42, no. 4, pp. 1–41, 2017.
[43] R. Chen, Q. Xiao, Y. Zhang, and J. Xu, "Differentially private high-dimensional data publication via sampling-based inference," in *Proceedings of the 21th ACM SIGKDD international conference on knowledge discovery and data mining*, 2015, pp. 129–138.
[44] M. Malek Esmaeili, I. Mironov, K. Prasad, I. Shilov, and F. Tramer, "Antipodes of label differential privacy: Pate and alibi," *Advances in Neural Information Processing Systems*, vol. 34, pp. 6934–6945, 2021.
[45] R. Wu, J. P. Zhou, K. Q. Weinberger, and C. Guo, "Does label differential privacy prevent label inference attacks?" in *International Conference on Artificial Intelligence and Statistics*, 2023, pp. 4336–4347.

Qiao Xue is an associate professor in the College of Computer Science and Technology, Nanjing University of Aeronautics and Astronautics, China. Before that, she was a postdoctoral fellow at The Hong Kong Polytechnic University. She received her B.E. degree and Ph.D. degree in Nanjing University of Aeronautics and Astronautics, China in 2015 and 2020, respectively. Her research interests include data privacy and privacy-preserving machine learning.

Qingqing Ye is an assistant professor in the Department of Electrical and Electronic Engineering, The Hong Kong Polytechnic University. She received her PhD degree from Renmin University of China in 2020. She has received several prestigious awards, including Hong Kong RGC Early Career Award, IEEE S&P Travel Award, and National Scholarship. Her research interests include data privacy and security, and adversarial machine learning.

Haibo Hu is a professor in the Department of Electrical and Electronic Engineering, Hong Kong Polytechnic University and the programme leader of BSc (Hons) in Information Security. His research interests include cybersecurity, data privacy, internet of things, and adversarial machine learning. He has published over 180 research papers in refereed journals, international conferences, and book chapters. As principal investigator, he has received over 20 million HK dollars of external research grants from Hong Kong and mainland China. He is the recipient of a number of titles and awards, including IEEE MDM 2019 Best Paper Award, WAIM Distinguished Young Lecturer, ICDE 2020 Outstanding Reviewer, VLDB 2018 Distinguished Reviewer, ACM-HK Best PhD Paper, Microsoft Imagine Cup, and GS1 Internet of Things Award. He is a senior member of ACM, IEEE and CCF, and a certified Cisco CCNA Security Trainer.

Jian Lou is an associate professor at Sun Yat-sen University. Previously, he was a researcher at HIC-ZJU, Zhejiang University, an associate professor at Xidian University, and a postdoc hosted by Prof.Li Xiong in Department of Computer Science, Emory University. He received his PhD degree from Hong Kong Baptist University, and BS from Zhejiang University. His research interests focus on trustworthy machine learning and privacy-preserving machine learning.

Jin Li is currently a professor at Guangzhou University. He got his Ph.D degree in information security from Sun Yat-sen University at 2007. His research interests include design of secure protocols in Artificial Intelligence, Cloud Computing (secure cloud storage and outsourcing computation) and cryptographic protocols. He has published more than 100 papers in international conferences and journals, including IEEE INFOCOM, IEEE TIFS, IEEE TPDS, IEEE TOC and ESORICS etc. His work has been cited more than 18000 times at Google Scholar and the H-Index is 40. He is Editor-in-Chief of International Journal of Intelligent Systems. He also serves as Associate editor for several international journals, including IEEE Transactions on Dependable and Secure Computing, Information Sciences.

Chengfang Fang obtained his Ph.D. degree from National University of Singapore before joining Huawei. He has been working on security and privacy protection in several areas including machine learning, internet of things, mobile device and biometrics for more than 10 years. He has published over 20 research papers and obtained 15 patents in the domain. He is currently a principle researcher in Huawei Singapore Research Center.

Jie Shi is a Principal Researcher in Huawei Singapore Research Center. His research interests include trustworthy AI, machine learning security, data security and privacy, IoT security and applied cryptography. He has over 10 years' research experience and has published over 30 research papers in refereed journals and international conferences. He received his Ph.D degree from Huazhong University of Science and Technology, China.