Does Label Differential Privacy Prevent Label Inference Attacks?

Ruihan Wu[*] Cornell University
Jin Peng Zhou[*] Cornell University
Kilian Q. Weiberger Cornell University
Chuan Guo Meta AI

Abstract

Label differential privacy (label-DP) is a popular framework for training private ML models on datasets with public features and sensitive private labels. Despite its rigorous privacy guarantee, it has been observed that in practice label-DP *does not* preclude label inference attacks (LIAs): Models trained with label-DP can be evaluated on the public training features to recover, with high accuracy, the very private labels that it was designed to protect. In this work, we argue that this phenomenon is not paradoxical and that label-DP is designed to limit the *advantage* of an LIA adversary compared to predicting training labels using the *Bayes classifier*. At label-DP $\epsilon = 0$ this advantage is zero, hence the optimal attack is to predict according to the Bayes classifier and is independent of the training labels. Our bound shows the semantic protection conferred by label-DP and gives guidelines on how to choose ε to limit the threat of LIAs below a certain level. Finally, we empirically demonstrate that our result closely captures the behavior of simulated attacks on both synthetic and real world datasets.

1 INTRODUCTION

Differential privacy (DP) (Dwork et al., 2006, 2014) has become the foundational tool for private learning on sensitive training data. More recently, this framework has been adopted for training *label differentially private* (label-DP) models (Chaudhuri and Hsu, 2011; Ghazi et al., 2021; Esmaeili et al., 2021), where only the label of a training sample is considered sensitive and must be protected. One prominent application for label-DP is online advertisement, where the learning goal is to predict whether a user clicked on an ad or not, which is a private and sensitive label, given the product description for the displayed ad.

Proceedings of the 26[th] International Conference on Artificial Intelligence and Statistics (AISTATS) 2023, Valencia, Spain. PMLR: Volume 206.

Intuitively, label-DP presents an easier task for the learner compared to DP since the training features are assumed to be public. Indeed, prior work showed that label-DP learning algorithms can achieve much higher test accuracy compared to the best DP counterparts on benchmark datasets. However, such models also attain a high accuracy on the training set, which enables an adversary to simply evaluate the model on the public training features to (correctly) predict the private labels (Busa-Fekete et al., 2021)—a method that we refer to as the *simple prediction attack* (SPA). The existence of such a paradoxical adversary raises the question of whether label-DP is truly a meaningful privacy notion to strive for.

In this paper, we take a closer look at the connection between label-DP and label inference attacks (LIAs). We first show that label-DP is unable to upper bound the accuracy of LIAs under arbitrarily small values of the privacy parameter ϵ. This limitation applies not only to label-DP, but *any* model that generalizes will inevitably enable the SPA attack to attain high label inference accuracy. In the extreme case where the learning algorithm perfectly generalizes, the output model becomes the Bayes classifier and the SPA attack's accuracy is determined entirely by the Bayes error rate, which is independent of the training labels.

Our analysis suggests that it is unreasonable to equate label privacy with limiting the accuracy of LIAs in absolute terms. At a high level, such an argument is in-line with the design principle of DP as not to protect against statistical inference (McSherry, 2016; Bun et al., 2021). Instead, we consider the *advantage* of an LIA adversary over predicting training labels according to the Bayes classifier. Such advantage can only have originated from memorizing the training set and therefore leakage of private labels, and vice versa an adversary with zero advantage is no better than the Bayes classifier that is completely independent of the training labels. Under this analytical framework, we show that an ε-label-DP learner can reduce this advantage to $1 - \frac{2}{1+e^{\varepsilon}}$. Importantly, our bound shows that at low ε, even if an label-DP learner achieves high training accuracy, it does not necessarily reveal any sensitive information about the training labels—resolving the aforementioned paradox. Our bound gives semantic meaning to the label-DP ε and can be used as a guideline for calibrating the value of ε for

practical use cases.

We empirically validate the advantage upper bound on both simulated and real world datasets. On the simulated dataset where the Bayes classifier is known, our upper bound dominates advantage of the SPA attack and is fairly tight at both small and large ε values. We also evaluate on the Criteo 1TB Click Logs dataset (Tallis and Yadav, 2018), which closely resembles the learning setting in common applications of label-DP where the ground truth label is very noisy and the marginal label distribution is highly imbalanced. Our result shows that advantage of the SPA attack becomes negative at even moderate values of the privacy parameter ε despite the attack attaining close to 97% label inference accuracy.

2 PRELIMINARIES

Notations. Let $\mathcal{X}, \mathcal{Y}$ denote the feature and label space, respectively, and let $D = (X, \mathbf{y}) \in (\mathcal{X} \times \mathcal{Y})^n$ be a training dataset consisting of n training samples. Let $\mathcal{D}$ be the underlying data distribution. For $i = 1, \ldots, n$, $X_{-i} \in \mathcal{X}^{n-1}$ denotes the training features except for the ith sample.

Differential privacy (DP) (Dwork et al., 2006, 2014) is a standard tool for privacy-preserving data analysis that hides the contribution of any individual training sample to the mechanism's output. In the context of machine learning, this is achieved by randomizing the learner's output and requiring that replacing one data point by another does not lead to a significant change in the output distribution. We restate its formal definition below.

Definition 1 ((ε, δ)-Differential Privacy)**.** *Let $\varepsilon, \delta \in \mathbb{R}^{\geq 0}$. A randomized training algorithm $\mathcal{M} : (\mathcal{X} \times \mathcal{Y})^n \to \mathcal{R}$ with domain $(\mathcal{X} \times \mathcal{Y})^n$ and range $\mathcal{R}$ satisfies (ε, δ)-differential privacy if for any two adjacent datasets $D, D' \in (\mathcal{X} \times \mathcal{Y})^n$, which differ at exactly one data point $(\mathbf{x}, y)$, and for any subset of outputs $S \subseteq \mathcal{R}$, it holds that:*

$$\mathbb{P}[\mathcal{M}(D) \in S] \leq e^{\varepsilon} \cdot \mathbb{P}[\mathcal{M}(D') \in S] + \delta.$$

Label differential privacy (label-DP) (Chaudhuri and Hsu, 2011) is a relaxation of DP where only the privacy of training *labels* must be protected. This setting assumes that the training features are public and/or non-sensitive but the labels are sensitive and are kept secret. Such a scenario arises naturally in several common applications of ML:

1. In online advertising, ads are selected by an ML model for display to maximize click-through rate (CTR)—the percentage of users that will click on the ad (Richardson et al., 2007; McMahan et al., 2013; Chapelle et al., 2014). The model is trained on features such as product and advertiser description, and a binary label of whether the user clicked on a displayed ad or not. In this application, features are publicly accessible and non-sensitive, but the label indicates user interest and is considered sensitive and private.

2. In recommendation systems, the learning goal is to suggest products or webpages to a user based on features such as user profile, search query, and descriptions of products/webpages, which are available to the recommender (Ricci et al., 2011). The training labels are historical data of user rating or click and are considered private.

The existence of a label-only privacy setting motivates the study of label-DP. Different from DP, the notion of adjacency applies only to the label of a single training sample: D and D' are identical except for one data point $(\mathbf{x}, y) \in D$ and $(\mathbf{x}, y') \in D'$; see below for a formal definition.

Definition 2 ((ϵ, δ)-Label Differential Privacy)**.** *Let $\varepsilon, \delta \in \mathbb{R}^{\geq 0}$. A randomized training algorithm $\mathcal{M}$ taking as input a dataset is said to be (ε, δ)-label differentially private ((ε, δ)-label-DP) if for any two training datasets D and D' that differ in the label of a single example, and for any subset S of outputs of $\mathcal{M}$, it holds that*

$$\mathbb{P}(\mathcal{M}(D) \in S) \leq e^{\varepsilon} \cdot \mathbb{P}(\mathcal{M}(D') \in S) + \delta.$$

When $\delta = 0$, we simply refer to $\mathcal{M}$ as ϵ-label-DP.

Label-DP learning algorithms. The first mechanism for achieving label differential privacy is *randomized response* (RR) (Warner, 1965), which (with a certain probability) randomly samples training labels according to a pre-determined distribution before releasing them to the learner. Recent works proposed several label-DP learning algorithms that are inspired by RR:

1. *Label Private Multi-Stage Training* (LP-MST; (Ghazi et al., 2021)) randomly samples training labels $\mathbf{y}_i$ using a learned prior sampling distribution $\mathbb{P}(y|X_i)$ instead of the pre-determined distribution in RR. Such a prior could be learned by observing the top-K predictions using a pre-trained model and limiting RR to this subset of most likely labels. An alternative way is to divide the training process into multiple stages and leverage the model trained in the previous stage as the prior for predicting the top-K labels.

2. *Private Aggregation of Teacher Ensembles with FixMatch* (PATE-FM; (Esmaeili et al., 2021)) uses FixMatch (Sohn et al., 2020)—a semi-supervised learning algorithm—to train several teacher models for private aggregation. Each teacher is trained on all training features together with a subset of revealed labels, with this subset disjoint among different teachers. Finally, a student model is trained using PATE (Papernot et al., 2016) to predict differentially privately aggregated labels from the teachers' predictions given public training features.

3. *Additive Laplace Noise Coupled with Bayesian Inference* (ALIBI; (Esmaeili et al., 2021)) releases differentially private training labels by perturbing one-hot encodings of the labels using the Laplace mechanism (Ghosh et al., 2012). Since post-processing preserves differential privacy (Dwork et al., 2014), the resulting noisy labels can then be denoised

using Bayesian inference to maximize the probability of recovering the clean label.

3 DOES LABEL-DP PREVENT LABEL INFERENCE ATTACKS?

Relaxing DP to label-DP provided the flexibility for designing more specialized private learning algorithms. These methods seem to provide excellent trade-offs between privacy and model utility, as measured by their high test accuracy even at very low ε, δ values. In this section, we take a closer look at the privacy protection offered by label-DP. We argue that not only is label-DP unable to prevent adversaries from inferring the training labels under arbitrarily low values of privacy parameter $\varepsilon > 0$, *any* model that generalizes will inevitably fail to do so as well.

3.1 Label Inference Attack against Label-DP

Label Inference Attack in Vertical Federated Learning. In the setting of *Vertical Federated Learning* (VFL), one party owns the features and the other party owns the labels. The objective is to jointly train a model without leaking private information about the labels. One way to achieve this is using split learning, where the party with features holds the first several layers of the model and the party with labels holds the remaining layers of the model, and the exchange of gradients at the split layer is protected by label-DP. However, Sun et al. (2022) showed that even when the label-DP ε is as small as 0.5, the party with the features can still infer the private labels with a prediction AUC of 0.75.

Label Inference Attack in Label-DP Model Training. We first make the observation that the label-DP guarantee *does not* imply that an adversary cannot leverage the model to infer its training labels. In fact, if the training accuracy is high, the adversary can trivially evaluate the model on the public training set and recover its labels (Busa-Fekete et al., 2021). We refer to this attack as the *simple prediction attack* (SPA) and evaluate it on existing label-DP learning algorithms.

Table 1 shows test accuracy for several label-DP models trained on MNIST (LeCun et al., 1998) and CIFAR10 (Krizhevsky et al., 2009), along with the corresponding SPA attack accuracy, *i.e.*, the model's training accuracy. There is a clear trend that these algorithms can achieve very high test accuracies with strong label-DP guarantee, *i.e.*, low privacy parameter ε. For instance, at $\varepsilon = 0.1$, the test accuracy could reach as high as 97.0 for MNIST and 87.6 for CIFAR10. However, the SPA attack accuracy is almost identical to the model's test accuracy, which (paradoxically) seems to suggest that models leak a tremendous amount of label information even when trained with stringent label-DP guarantees.

3.2 Impossibility of Label Protection under Label-DP

Following the above observation, a natural question to ask is whether the vulnerability of label-DP to the existing label inference attack is due to an insufficiently strong privacy guarantee, *i.e.*, ε, δ being not small enough. We give a definitive negative answer by formalizing *label inference attacks* (LIAs) and showing that label-DP cannot guarantee protection against LIAs even for arbitrarily small values of $\varepsilon, \delta > 0$.

Threat model. The adversary's goal is to design a label inference attack algorithm $\mathcal{A}$ that infers the training label of each sample in the training dataset. The output of $\mathcal{A}$ is a vector of inferred labels $\hat{\mathbf{y}} \in \mathcal{Y}^n$. We assume that the adversary has access to the following information:

1. The adversary has full knowledge of the output $o = \mathcal{M}(X, \mathbf{y})$, where $\mathcal{M}$ is any releasing algorithm. The output o could be the model when we consider the label-DP model training. It could also be a sequential of message passed between parties when label-DP is applied in the federated learning setting.
2. The adversary has full knowledge of the feature matrix $X \in \mathbb{R}^{n \times d}$.
3. (Optional) The adversary has knowledge of the conditional data distribution $\mathbb{P}(y|\mathbf{x})$ of $\mathcal{D}$.

We refer to the threat model with or without the third assumption as the *with-prior* or *priorless* setting. The with-prior setting is not unrealistic: Given access to a separate dataset for the same learning task, the adversary can train a shadow model to estimate the conditional probability $\mathbb{P}(y|\mathbf{x})$. Such an approach is commonly used in membership inference attacks (Shokri et al., 2017).

Expected attack utility. To measure how successful an LIA is at inferring training labels, we define the *attack utility function* $u(\hat{\mathbf{y}}_i, \mathbf{y}_i)$ where $\hat{\mathbf{y}}_i \in \mathcal{Y}$ is the inferred label and $\mathbf{y}_i$ is the ground truth. We assume that $u(\hat{\mathbf{y}}_i, \mathbf{y}_i) \in [0, B]$ for any $\hat{\mathbf{y}}_i, \mathbf{y}_i \in \mathcal{Y}$, *e.g.*, $u(\hat{\mathbf{y}}_i, \mathbf{y}_i) = \mathbb{1}(\hat{\mathbf{y}}_i = \mathbf{y}_i)$ is the zero-one accuracy for classification problems, which is bounded with $B = 1$. For regression problems with bounded label range $\mathcal{Y} \subseteq [-b, b]$, we can define $u(\hat{\mathbf{y}}_i, \mathbf{y}_i) = 4b^2 - (\hat{\mathbf{y}}_i - \mathbf{y}_i)^2$, which is bounded with $B = 4b^2$. We assume $B = 1$ for simplicity; our results can be easily generalized to any $B > 0$.

The *expected attack utility* (EAU) is defined as the expectation of $u(\hat{\mathbf{y}}_i, \mathbf{y}_i)$ over the randomness of the sampling of labels and in the learning algorithm:

$$\mathrm{EAU}(\mathcal{A}, \mathcal{K}) = \mathbb{E}_{\mathbf{y}, \mathcal{M}} \left[\frac{1}{n} \sum_{i=1}^{n} u(\hat{\mathbf{y}}_i, \mathbf{y}_i) \middle| X \right], \quad (1)$$

Table 1: Model test accuracy and attack accuracy of the simple prediction attack (SPA) evaluated on label-DP models trained on MNIST and CIFAR10. SPA attack accuracy is equivalent to training accuracy for classification and is exceptionally high in most cases. Our evaluation shows that a learning algorithm can offer a very stringent label-DP guarantee of $\varepsilon = 0.1$ while failing to prevent label inference attacks.

Algorithm	**MNIST** ($\varepsilon = 1.0$)		**MNIST** ($\varepsilon = 0.1$)		**CIFAR10** ($\varepsilon = 1.0$)		**CIFAR10** ($\varepsilon = 0.1$)	
	Test Acc.	Attack Acc.	Test Acc.	Attack Acc.	Test Acc.	Attack Acc.	Test Acc.	Attack Acc.
LP-1ST	93.3	93.3	20.8	20.9	61.5	61.9	15.5	15.8
LP-1ST (in-domain prior)	97.1	96.5	97.0	96.2	75.4	75.7	66.3	66.3
LP-1ST (out-of-domain pior)	94.6	93.7	86.2	85.2	89.5	89.8	87.6	86.9
PATE-FM	99.3	99.1	23.6	23.0	92.4	92.1	18.6	18.6
ALIBI	96.3	96.3	21.5	20.8	67.5	69.6	13.6	13.9

where $\mathcal{K}$ denotes the adversary's knowledge: $(X, o, \mathbb{P}(y|\mathbf{x}))$ for the with-prior setting and (X, o) for the priorless setting.

Upper bound on expected attack utility. For attack utility functions u that reflect the accuracy of a label inference attack, one may ask whether label-DP can provide a uniform upper bound $U(\varepsilon, \delta)$ such that

$$\mathrm{EAU}\left(\mathcal{A}, \mathcal{K}\right) \leq U(\varepsilon, \delta) \tag{2}$$

holds for any data distribution $\mathcal{D}$, feature matrix X and attack algorithm $\mathcal{A}$. A trivial upper bound $U(\varepsilon, \delta) \leq 1$ follows from the boundedness of u. Unfortunately, we show that this bound is in fact optimal for both the with-prior and the priorless settings.

Theorem 1. *There is no function $U(\varepsilon, \delta)$ that satisfies Equation 2 and is strictly less than 1 at some $\varepsilon, \delta > 0$.*

Proof. We first consider the with-prior setting where the adversary has access to the conditional distribution $\mathbb{P}(y|\mathbf{x})$. For a classification problem with utility function $u(\hat{\mathbf{y}}_i, \mathbf{y}_i) = \mathbb{1}(\hat{\mathbf{y}}_i = \mathbf{y}_i)$, we define the attack $\mathcal{A}_{\mathrm{prior}}$ that predicts training labels according to the Bayes classifier:

$$\mathcal{A}_{\mathrm{prior}}(\mathcal{K}) := \left(\arg\max_{y \in \mathcal{Y}} \mathbb{P}(\mathbf{y}_i = y | X_i) : i = 1, \cdots n \right). \tag{3}$$

The expected attack utility for $\mathcal{A}_{\mathrm{prior}}$ is:

$$\mathrm{EAU}\left(\mathcal{A}_{\mathrm{prior}}, \mathcal{K}\right) = \frac{1}{n} \sum_{i=1}^{n} \max_{y \in \mathcal{Y}} \mathbb{P}(\mathbf{y}_i = y | X_i). \tag{4}$$

In particular, when the label y is deterministic given the feature $\mathbf{x}$, *i.e.*, $\mathbb{P}(y|\mathbf{x}) = 1$ for some $y \in \mathcal{Y}$, this EAU evaluates to 1. Note that $\mathcal{A}_{\mathrm{prior}}$ does not depend on o and is thus valid for any ε, δ, hence $U(\varepsilon, \delta) = 1 \; \forall \varepsilon, \delta > 0$.

For the *priorless* setting, consider again a classification problem with the same utility function u. Denote by $\mathcal{A}_{\mathrm{priorless}}$ the simple prediction attack, which predicts labels using the label-DP trained model f. We will construct a series of settings ($\mathcal{D}^n$, X^n, f^n) such that each f^n is (ε, δ)-label-DP and as $n \to \infty$, $\mathrm{EAU}\left(\mathcal{A}_{\mathrm{priorless}}, \mathcal{K}\right) \to 1$, which shows that $U(\varepsilon, \delta) \geq 1 \; \forall \varepsilon, \delta > 0$.

- *Data construction*: The feature domain $\mathcal{X}$ is $\{-1, 1\}$ and the label space $\mathcal{Y}$ is $\{-1, 1\}$. We construct X^n with $n = 2r$ samples where $X_1^n = \cdots = X_r^n = 1$ and $X_{r+1}^n = \cdots = X_{2r}^n = -1$. The conditional label distribution given the feature is $\mathbb{P}(\mathbf{y}_i^n = X_i^n | X_i^n) = 1$ for all i.
- *Label-DP algorithm for training f^n*: We apply randomized response with $\mathbb{P}(\tilde{\mathbf{y}}_i = \mathbf{y}_i^n) = \frac{e^\varepsilon}{e^\varepsilon + 1}$ to privatize the labels. We then train f^n on the privatized labels to maximize training accuracy, which results in f^n being simply the majority sign function: $f^n(1) = \mathrm{sign}\{\sum_{i=1}^{r} \tilde{\mathbf{y}}_i\}$ and $f^n(-1) = \mathrm{sign}\left\{\sum_{i=r+1}^{2r} \tilde{\mathbf{y}}_i\right\}$.

The fact that f^n is $(\varepsilon, 0)$-label-DP follows from $\tilde{\mathbf{y}}_i$ being the randomized response of $\mathbf{y}_i$ for $i = 1, \ldots, n$ and f^n being a post-processing function. The SPA attack's EAU is equal to the training accuracy of the model f^n, which can be lower bounded using Hoeffding's inequality:

$$\begin{aligned}
&\mathbb{E}\left[\frac{1}{n} \sum_{i=1}^{n} \mathbb{1}\left[\mathcal{A}_{\mathrm{priorless}}(X^n, f^n)_i = \mathbf{y}_i^n\right] \middle| X^n \right] \\
&= \frac{1}{2} \cdot \mathbb{P}\left(\sum_{i=1}^{r} \tilde{\mathbf{y}}_i > 0 \right) + \frac{1}{2} \cdot \mathbb{P}\left(\sum_{i=r+1}^{2r} \tilde{\mathbf{y}}_i < 0 \right) \\
&\geq \mathbb{P}\left(\left| \frac{1}{r} \sum_{i=1}^{r} \frac{1 + \tilde{\mathbf{y}}_i}{2} - \frac{e^\varepsilon}{e^\varepsilon + 1} \right| < \frac{e^\varepsilon}{e^\varepsilon + 1} - \frac{1}{2} \right) \\
&\geq 1 - 2\exp\left(- \left(\frac{e^\varepsilon}{e^\varepsilon + 1} - \frac{1}{2} \right)^2 n \right).
\end{aligned}$$

Taking $n \to \infty$ gives the desired result. □

Theorem 1 shows that for both the with-prior and priorless settings, no non-trivial upper bound for EAU exists for any label-DP privacy parameters $\varepsilon, \delta > 0$. It also validates our rationalization about the experimental result in Table 1 that failure to prevent the SPA attack is to be expected.

4 LABEL-DP PROVABLY BOUNDS ATTACK ADVANTAGE

The impossibility results derived in section 3 suggest that limiting the EAU of label inference attacks may not be a reasonable objective for label privacy. In particular, since the with-prior attack $\mathcal{A}_{\text{prior}}$ in Theorem 1 completely disregards the trained model o, it should be treated as a baseline for measuring the effectiveness of LIAs. Indeed, $\mathcal{A}_{\text{prior}}$ generalizes the Bayes classifier and is optimal among attacks that are independent of the training labels $\mathbf{y}$ by construction. Hence any attack that achieves an EAU equal to or less than that of $\mathcal{A}_{\text{prior}}$ *does not gain any additional information* about the training labels from o. Thus, we refer to

$$\text{EAU}(\mathcal{A}_{\text{prior}}, \mathcal{K}) = \frac{1}{n}\sum_{i=1}^{n} \max_{y\in\mathcal{Y}} \mathbb{E}_{\mathbf{y}_i}\left[u(y, \mathbf{y}_i)|X_i\right]$$

as the *label-independent expected attack utility* (L-EAU), and instead measure the success of a label inference attack $\mathcal{A}$ by defining its *advantage*:

$$\text{Adv}\left(\mathcal{A}, \mathcal{K}\right) := \text{EAU}\left(\mathcal{A}, \mathcal{K}\right) - \text{EAU}(\mathcal{A}_{\text{prior}}, \mathcal{K}). \quad (5)$$

Next, we show that label-DP can effectively reduce the advantage of LIAs to close to 0 when its privacy parameters ε, δ are small. We first prove a distribution-dependent upper bound in Theorem 2 that holds for any label-DP output o and attack algorithm $\mathcal{A}$ but depends on the conditional distribution $\mathbb{P}(y|\mathbf{x})$, and then give a universal upper bound $\text{Adv}\left(\mathcal{A}, \mathcal{K}\right) \leq U(\varepsilon, \delta)$ in Corollary 1 that only depends on the label-DP parameters (ε, δ). Proof is given in Appendix A.

Theorem 2. *Assume each label $\mathbf{y}_i$ is sampled independent of $(\mathbf{y}_{-i}, X_{-i})$. If o satisfies (ε, δ)-label-DP then for any attack algorithm $\mathcal{A}$, we have:*

$$\text{Adv}\left(\mathcal{A}, \mathcal{K}\right) \leq \left(1 - \frac{2}{1+e^{\varepsilon}}(1-\delta)\right) \cdot \left(\frac{1}{n}\sum_{i=1}^{n} \mathbb{E}_{\mathbf{y}_i|X_i}\left[\sup_{y\in\mathcal{Y}} u(y, \mathbf{y}_i)\right]\right).$$

Proof. First note that the adversary's inferred label vector $\hat{\mathbf{y}} = \mathcal{A}(X, \mathcal{M}(X, \mathbf{y}))$ is a random variable that depends on both the sampling of training labels $\mathbf{y}$ and randomness in the learning algorithm $\mathcal{M}$. Then:

$$\mathbb{E}[u(\hat{\mathbf{y}}_i, \mathbf{y}_i)|X, \mathbf{y}_{-i}] = \mathbb{E}_{\mathbf{y}_i|X_i}\left[\mathbb{E}[u(\hat{\mathbf{y}}_i, \mathbf{y}_i)|\mathbf{y}, X]\right], \quad (6)$$

where the equality holds by the assumption that $\mathbf{y}_i$ is independent of X_{-i} and $\mathbf{y}_{-i}$. For each $i = 1, \cdots, n$, let $B(\mathbf{y}_i) = \sup_{y\in\mathcal{Y}} u(y, \mathbf{y}_i)$ be the maximal attack utility attainable when inferring the ground truth label $\mathbf{y}_i$. Consider an alternative label vector $\mathbf{y}'$ that is identical to $\mathbf{y}$ except for $\mathbf{y}'_i$ being replaced with some deterministic label value y^*, and denote by $\hat{\mathbf{y}}' = \mathcal{A}(X, \mathcal{M}(X, \mathbf{y}'))$ the adversary's inferred labels for the model trained on $(X, \mathbf{y}')$. By label-DP, we have:

$$\begin{aligned}
\mathbb{E}[u(\hat{\mathbf{y}}_i, \mathbf{y}_i)|\mathbf{y}, X] &= \int_0^{B(\mathbf{y}_i)} \mathbb{P}(u(\hat{\mathbf{y}}_i, \mathbf{y}_i) > v|\mathbf{y}, X)\,\mathrm{d}v \\
&\leq \int_0^{B(\mathbf{y}_i)} \mathbb{P}(u(\hat{\mathbf{y}}'_i, \mathbf{y}_i) > v|\mathbf{y}, X)\,\mathrm{d}v \\
&\quad + \int_0^{B(\mathbf{y}_i)} \left(1 - \frac{2}{1+e^{\varepsilon}}(1-\delta)\right)\mathrm{d}v \\
&= \mathbb{E}[u\left(\hat{\mathbf{y}}'_i, \mathbf{y}_i\right)|\mathbf{y}, X] \\
&\quad + \left(1 - \frac{2}{1+e^{\varepsilon}}(1-\delta)\right) \cdot B(\mathbf{y}_i),
\end{aligned}$$

where the inequality follows the Remark A.1 in Kairouz et al. (2015) Substituting the above inequality into Equation 6 gives:

$$\begin{aligned}
\mathbb{E}[u(\hat{\mathbf{y}}_i, \mathbf{y}_i)|X, \mathbf{y}_{-i}] &\leq \mathbb{E}\left[u\left(\hat{\mathbf{y}}'_i, \mathbf{y}_i\right)|X, \mathbf{y}_{-i}\right] \\
&\quad + \left(1 - \frac{2}{1+e^{\varepsilon}}(1-\delta)\right) \cdot \mathbb{E}_{\mathbf{y}_i|X_i}\left[B(\mathbf{y}_i)\right] \\
&\leq \max_{y\in\mathcal{Y}} \mathbb{E}_{\mathbf{y}_i|X_i}[u(y, \mathbf{y}_i)] \\
&\quad + \left(1 - \frac{2}{1+e^{\varepsilon}}(1-\delta)\right) \cdot \mathbb{E}_{\mathbf{y}_i|X_i}\left[B(\mathbf{y}_i)\right],
\end{aligned}$$

where the last inequality holds by the fact that $\hat{\mathbf{y}}'_i$ is independent of $\mathbf{y}_i$ conditioned on X and $\mathbf{y}_{-i}$. Finally, we can derive our bound for the advantage $\text{Adv}\left(\mathcal{A}, \mathcal{K}\right)$:

$$\begin{aligned}
&\text{EAU}\left(\mathcal{A}, \mathcal{K}\right) - \frac{1}{n}\sum_{i=1}^{n} \max_{y\in\mathcal{Y}} \mathbb{E}_{\mathbf{y}_i}\left[u(y, \mathbf{y}_i)|X_i\right] \\
&= \frac{1}{n}\sum_{i=1}^{n}\left(\mathbb{E}_{\mathbf{y}_{-i}}\mathbb{E}[u(\hat{\mathbf{y}}_i, \mathbf{y}_i)|X, \mathbf{y}_{-i}] - \max_{y\in\mathcal{Y}} \mathbb{E}_{\mathbf{y}_i}\left[u(y, \mathbf{y}_i)|X_i\right]\right) \\
&\leq \left(1 - \frac{2}{1+e^{\varepsilon}}(1-\delta)\right) \cdot \left(\frac{1}{n}\sum_{i=1}^{n} \mathbb{E}_{\mathbf{y}_i|X_i}\left[\sup_{y\in\mathcal{Y}} u(y, \mathbf{y}_i)\right]\right).
\end{aligned}$$

□

Corollary 1. *Suppose $u(y', y) \in [0, B]$ for any $y', y \in \mathcal{Y}$ and each label $\mathbf{y}_i$ is sampled independent of $(\mathbf{y}_{-i}, X_{-i})$. If o satisfies (ε, δ)-label-DP then for any data distribution $\mathcal{D}$, any feature matrix X and any attack algorithm $\mathcal{A}$, we have:*

$$\text{Adv}\left(\mathcal{A}, \mathcal{K}\right) \leq \left(1 - \frac{2}{1+e^{\varepsilon}}(1-\delta)\right) \cdot B.$$

Interpretation of Theorem 2. We can interpret Theorem 2 by revisiting the example in subsection 3.1. Instead of bounding the EAU of a label inference attack, Theorem 2 shows that label-DP with low (ε, δ) can upper bound the *advantage* to close to 0. This result explains why even with the strong guarantee of ϵ-label-DP at $\varepsilon = 0.1$ in Table 1, the attack utility could still be as high as $80\%+$: Because both MNIST and CIFAR10 admit a high L-EAU (*i.e.*, high accuracy of the Bayes classifier), LIAs that attain $80\%+$ EAU

may not even outperform the label-independent attack $\mathcal{A}_{\text{prior}}$, hence models trained by label-DP algorithms do not leak a significant amount of information about training labels.

Moreover, we observe that the bound for the advantage, or equivalently the EAU, is relative to both the label-DP parameter ε and the underlying distribution. When $\frac{1}{n}\sum_{i=1}^{n}\mathbb{E}_{\mathbf{y}_i|X_i}\left[\sup_{y\in\mathcal{Y}} u(y,\mathbf{y}_i)\right]$ is higher, a higher ε is sufficient to achieve the same level of protection against LIAs. This interpretation is in-line with the design principle of DP, which is meant to limit the difference between the prior and posterior distributions for the underlying data (Dwork et al., 2014; Kasiviswanathan and Smith, 2014). From a practical aspect, one can use Corollary 1 to calibrate the values of ε and δ for the dataset at hand to limit the utility of arbitrary label inference attack.

4.1 Label-DP vs. DP

Remarkably, using DP even when only the label is private can give stronger semantic guarantees against LIAs than the label-DP guarantee in Theorem 2. This is however not true under the threat model defined in subsection 3.2 where the feature matrix X is public, but holds under a weaker threat model where the feature matrix is non-private but *unknown*. This threat model has been considered in Ghazi et al. (2021) and was implicitly used to motivate the randomized response mechanism (Warner, 1965).

In essence, with X unknown, the with-prior attack $\mathcal{A}_{\text{prior}}$ is no longer viable. Instead, the optimal attack without observing the trained model is to guess according to the *marginal distribution* of $\mathbf{y}_i$ for each i, resulting in an EAU of $\max_{y\in\mathcal{Y}}\mathbb{E}_{\mathbf{y}_i}[u(y,\mathbf{y}_i)]$, which is provably smaller than the L-EAU when the feature matrix is known and we denote it as L-EAU^w. For example, when we consider the data from CIFAR10, the L-EAU is able to attain $80\%+$ when feature matrix is known, while L-EAU^w is only 10% in the weaker threat model without the knowledge of the feature matrix. We define the new advantage corresponding to the weaker threat model as

$$\mathrm{Adv}^w\left(\mathcal{A},\mathcal{K}^w\right) := \mathrm{EAU}\left(\mathcal{A},\mathcal{K}^w\right) - \text{L-EAU}^w, \quad (7)$$

where $\mathcal{K}^w = o$ denotes the adversary's knowledge in this weaker setting.

Due to the lack of protection for the feature matrix, label-DP is not capable limit this advantage to 0. This is intuitive: with a successful feature (data) reconstruction attack (Fredrikson et al., 2015; Carlini et al., 2019; Zhu et al., 2019), the adversary will have the knowledge of feature and hence achieve previous higher L-EAU. Instead, DP including the protection of features can successfully limit this advantage of the weaker threat model into 0. Theorem 3 below gives a precise statement; see Appendix A for proof.

Theorem 3. *Assume each data $(X_i,\mathbf{y}_i)$ is sampled independently. If f satisfies (ε,δ)-DP then for any attack algorithm $\mathcal{A}$, we have:*

$$\mathrm{Adv}^w\left(\mathcal{A},\mathcal{K}^w\right) \le \left(1-\frac{2}{1+e^{\varepsilon}}(1-\delta)\right)\cdot\left(\frac{1}{n}\sum_{i=1}^{n}\mathbb{E}_{\mathbf{y}_i}\left[\sup_{y\in\mathcal{Y}} u(y,\mathbf{y}_i)\right]\right).$$

Hence, in the scenario where the adversary does not have access to the feature matrix X, DP gives a stronger guarantee against label inference attacks and can be preferred over label-DP especially when L-EAU is relatively large.

5 EXPERIMENTS

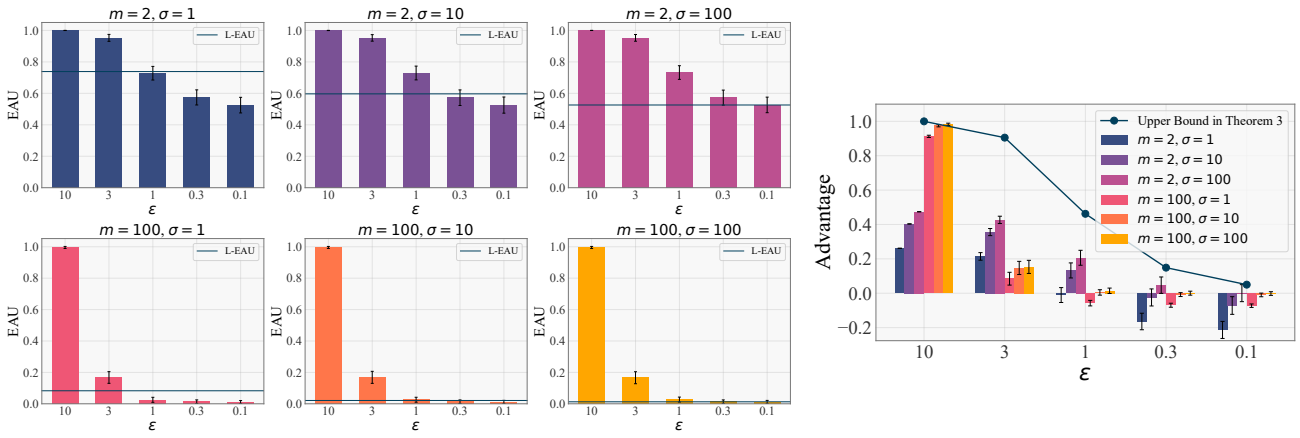
We validate Theorem 2 on both a simulated Gaussian mixture dataset (subsection 5.1) and a real world ads click prediction dataset (subsection 5.2) and show that empirical results obey the theoretically derived upper bound. For full reproducibility, we release our code at `https://github.com/jinpz/label_differential_privacy/`.

5.1 Simulated Dataset with Gaussian Mixture

Data generation. We define a classification setting where the feature space $\mathcal{X}=\mathbb{R}^d$ with $d=100$ and the label space $\mathcal{Y}=\{1,\dots,m\}$ for $m=2$ or $m=100$ classes. For each class i, features are sampled from an isotropic Gaussian $\mathcal{N}(\mathbf{e}_i,\sigma^2 I_d)$ where $\mathbf{e}_i$ is the standard basis vector. We vary $\sigma\in\{1,10,100\}$ and the resulting data distribution $\mathcal{D}$ is the uniform mixture of the m classes' distributions.

Model training. To train a private model that satisfies label-DP, we first draw $n=100$ random samples from the mixture distribution $\mathcal{D}$. Given a target label-DP privacy parameter ϵ, the learning algorithm trains a logistic regressor by generating privatized labels using randomized response (Warner, 1965) to satisfy ϵ-label-DP. This process is repeated multiple times to estimate the EAU of the simple prediction attack; see the following paragraph for details.

Attack evaluation. We evaluate the simple prediction attack (SPA) using the utility function $u(\hat{y},y)=\mathbb{1}\{\hat{y}=y\}$. To estimate the expected attack utility (EAU), we first *fix* a random draw of training features X from the marginal distribution of $\mathcal{D}$. Given X and the Gaussian mixture parameters, we can compute the conditional probability of each $y\in\mathcal{Y}$ using mixture densities and sample the label vector $\mathbf{y}$ accordingly. This process is repeated $T=1000$ times for the same training features to estimate the expectation over $\mathbf{y}$ in Equation 1. We find that the sampling of X does not significantly impact our result. Finally, we can construct the Bayes classifier by choosing the label y for each X_i that maximizes the conditional probability $\mathbb{P}(y|X_i)$ and compute L-EAU directly.



(a) Expected Attack Utility vs. $\mathcal{E}$ (b) Advantage vs. $\mathcal{E}$

Figure 1: **(a)** EAU and L-EAU vs. ε on the simulated dataset with $m = 2, 100$ and $\sigma = 1, 10, 100$. EAU decreases with lower ε (stronger privacy guarantee) and can be lower than L-EAU for moderately low values of ε. **(b)** Attack advantage and theoretical upper bound vs. ε. The upper bound dominates the advantage while being fairly tight for the end values $\varepsilon = 0.1$ and $\varepsilon = 10$ for $m = 100$. The error bar represents standard deviation across $T = 1000$ different draws of label vector $\mathbf{y}$.

Results. In Figure 1(a), we plot EAU and L-EAU against ε for the simulated dataset with $m = 2, 100$ and $\sigma = 1, 10, 100$. For both settings, EAU is close to 100% when $\varepsilon = 10$, and decreasing ε (*i.e.*, stronger privacy guarantee) leads to smaller EAU values. For $m = 2$, although EAU is consistently high even at $\epsilon = 0.1$, L-EAU is also very high, hence the attack does not attain a very large advantage (difference between EAU and L-EAU). For $m = 100$, L-EAU is much lower due to the classification problem being harder, while EAU also drops very quickly as ε decreases.

In Figure 1(b), we plot advantage and the upper bound in Theorem 2 against ε. As expected, the upper bound dominates attack advantage in all settings, and both values decrease monotonically as ε decreases. The upper bound is tight at the two end values $\varepsilon = 0.1$ and $\varepsilon = 10$ for $m = 100$, whereas for $m = 2$ it can be fairly loose. This is because L-EAU for $m = 2$ is at least 0.5 and therefore advantage is always upper bounded by 0.5. Our upper bound could potentially be improved if the minimum L-EAU value can be inferred from the task and/or data distribution.

5.2 Criteo Ads Click-Through Rate (CTR) Prediction

To understand the behavior of LIAs and our advantage bound on real world datasets that closely resemble the usage scenarios of label-DP, we conduct experiments on the Criteo 1TB Click Logs dataset[1] (Tallis and Yadav, 2018) for click-through rate (CTR) prediction.

Dataset description. In CTR prediction, the task is to predict the percentage of users that will click on the ad given ad features. The features consist of 13 integer values and 26 categorical features, while the semantic of these features is undisclosed. The binary label indicates whether a user clicked on the ad or not. The marginal label distribution is heavily skewed with approximately 97% of samples having the label 0, *i.e.*, no click.

The dataset contains more than 4B click log data points spanning across 24 days of data collection. We subsample 1M data points from the first day's entries, and take 80% as the training set, 4% as the validation set and the remaining 16% as the test set. Following the Kaggle competition[2] for this dataset, we evaluate model utility using log loss:

$$\frac{1}{|D_{\text{test}}|} \sum_{(\mathbf{x},y)\in D_{\text{test}}} -y \cdot \log\left(f(\mathbf{x})\right) + (y-1) \cdot \log\left(1 - f(\mathbf{x})\right).$$

Model training. We implemented gradient-boosted decision tree (Friedman, 2001) in CatBoost (Dorogush et al., 2018) as the baseline non-private learning algorithm. We further adapted LP-MST (Ghazi et al., 2021) and PATE (Papernot et al., 2016) to this setting as label-DP learning algorithms; see Appendix B for implementation details. For LP-MST, we considered multiple variants: LP-1ST, LP-1ST (domain prior), LP-1ST (noise correction) and LP-2ST.

Attack evaluation. Since the dataset is heavily skewed towards the label 0, simply predicting the all-zero label vector $\hat{\mathbf{y}}$ can attain an approximately 97% attack accuracy under the zero-one accuracy utility function. To correct for this bias, we consider a *weighted* attack utility function:

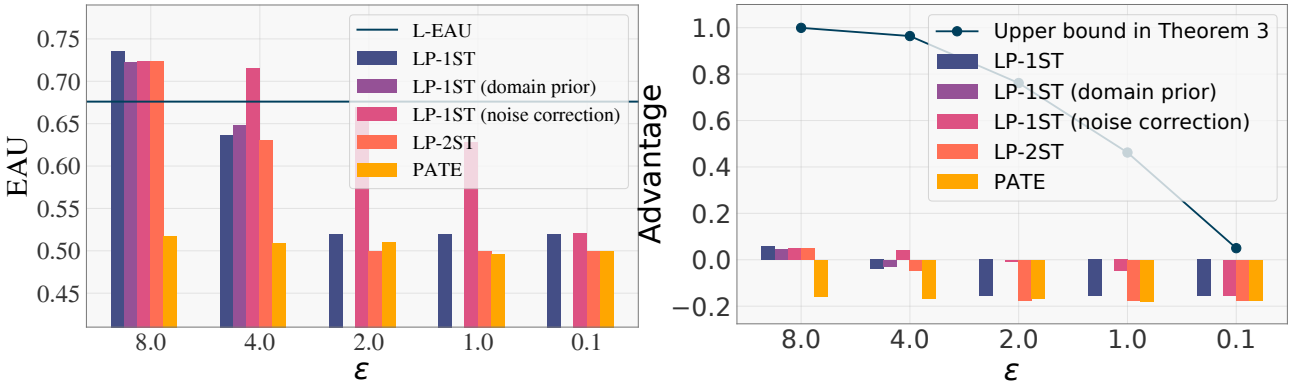
$$u(\hat{y}, y) := \frac{1}{2p_y} \mathbb{1}\{\hat{y} = y\}, \tag{8}$$

[1] https://ailab.criteo.com/download-criteo-1tb-click-logs-dataset

[2] https://www.kaggle.com/c/criteo-display-ad-challenge

Table 2: Log loss of state-of-the-art label-DP algorithms under different ε. When $\varepsilon \leq 2$, none of the label-DP algorithms outperform the constant prediction baseline, which attains a log loss of 0.135. For LP-1ST (domain prior) at $\varepsilon \in \{2, 1, 0.1\}$, the mechanism heavily relied on the prior and returned label 0 with probability 1 for all training samples, so training did not yield any meaningful result.

Label-DP Algorithm	$\varepsilon = \infty$	$\varepsilon = 8$	$\varepsilon = 4$	$\varepsilon = 2$	$\varepsilon = 1$	$\varepsilon = 0.1$
LP-1ST	0.123	0.123	0.129	0.204	0.360	0.651
LP-1ST (domain prior)	0.123	0.123	0.128	-	-	-
LP-1ST (noise correction)	0.123	0.123	0.126	0.151	0.177	0.646
LP-2ST	0.123	0.123	0.129	0.202	0.346	0.530
PATE	0.130	0.151	0.164	0.194	0.248	0.676



(a) Expected attack utility (EAU) vs. ε (b) Advantage vs. ε

Figure 2: Expected attack utility (EAU) and advantage of the simple prediction attack against label-DP models trained on the Criteo dataset. L-EAU of 0.676 is estimated using maximum test accuracy achieved on this dataset among all trained models. Even at $\epsilon = 8$, the EAU for label-DP models is marginally above L-EAU, hence the corresponding advantage is only slightly above 0. In (b), we show that Theorem 2 strictly upper bounds attack advantage but there exists a large gap compared to the computed advantage.

where p_y is the marginal probability of label y. This re-weighting has a meaningful interpretation for the adversary as well: The label 1 represents a user click and is more valuable to infer compared to the label 0 that represents no click, and hence should be given a higher utility when predicted correctly. Under this attack utility, predicting all-0, all-1 or randomly guessing y with probability p_y all attain an EAU of $1/2$. Finally, we adapt the SPA attack accordingly to optimize re-weighted utility:

$$\mathcal{A}_{\text{priorless}}(X, f)_i = \arg\max_{y \in \{0,1\}} \frac{1}{p_y} \cdot (y \cdot f(X_i) + (1-y) \cdot (1 - f(X_i))).$$

Computing EAU. Since the conditional label distribution is unknown, we cannot compute EAU or L-EAU directly as in the simulated dataset experiment in subsection 5.1. Instead, we use a model's training accuracy (weighted according to Equation 8) as an unbiased estimator for the EAU of the SPA attack. For L-EAU, any ML model evaluated on the test set gives a valid lower bound, and we pick the maximum over all models trained on this dataset as an approximation for L-EAU.

Results. Table 2 shows the model utility of label-DP models trained with $\varepsilon \in \{\infty, 8, 4, 2, 1, 0.1\}$. Since the dataset is heavily skewed with $p_0 \approx 0.97$ fraction of samples belonging to class 0, the constant predictor $f(\mathbf{x}) = p_0$ for all $\mathbf{x}$ achieves a log loss of 0.135. In comparison, the label-DP algorithms fail to outperform this naive baseline when $\varepsilon \leq 2$. Our evaluation suggests that there is much room for improvement in existing label-DP learning algorithms for this highly noisy and heavily skewed learning setting.

Next, we plot EAU of the SPA attack along with our estimate of L-EAU in Figure 2(a). The maximum attainable EAU is 1.0, while both EAU of the SPA attack and the estimated L-EAU are not very high, which reflects the noisy nature of this dataset. Moreover, EAU quickly deteriorates to below L-EAU when $\varepsilon = 2$, despite the attack accurately inferring approximately 97% of the training labels by predicting (nearly) all-zero. Finally, we see in Figure 2(b) that advantage of these attacks is very low and can be negative for $\varepsilon \leq 2$. We also evaluate the upper bound in Theorem 2, where the quantity $\mathbb{E}_{\mathbf{y}_i|X_i}\left[\sup_{y \in \mathcal{Y}} u(y, \mathbf{y}_i)\right] = 1$ by construction of u (*cf.* Equation 8). Although this bound again

dominates the computed advantage, there exists a very large gap. We suspect this is due to a number of reasons, including the SPA attack being sub-optimal or due to looseness in label-DP accounting. Future work may be able to design better LIAs that exploit model memorization in other ways or use tighter label-DP accounting to reduce this gap.

6 DISCUSSION AND CONCLUSION

Busa-Fekete et al. (2021) first noted the fact that even with the label-DP guarantee, an adversary can still recover the training labels via the simple prediction attack. Their Bayesian interpretation of this paradox is that the public release of features contributed to this privacy loss and that label-DP cannot mitigate this risk. We offer a different view and advocate that label leakage in absolute terms should not be considered a privacy violation. Instead, with the appropriate metric of success for the adversary, *i.e.*, advantage, we showed that label-DP can indeed prevent label inference attacks. We hope that future work can build upon our analysis to deepen our understanding of the connection between label-DP and LIAs. Lastly, we do not foresee any negative societal impacts for our work.

Limitations. Our paper focuses on deriving LIA advantage bounds for (ϵ, δ)-label-DP. Alternative formulations of DP such as Rényi-DP (Mironov, 2017) and Gaussian DP (Dong et al., 2019) offer much tighter privacy accounting for composition of mechanisms, and hence it may be of interest to derive similar bounds under these accounting frameworks. Moreover, our evaluation on the Criteo dataset is only a preliminary study. Follow-up work on thoroughly analyzing label-DP algorithms and studying their limitations on such challenging datasets is warranted for a better understanding of the privacy-utility trade-offs.

Acknowledgements

RW, JPZ and KQW are supported by grants from the National Science Foundation NSF (IIS-2107161, III-1526012, IIS-1149882, and IIS-1724282), and the Cornell Center for Materials Research with funding from the NSF MRSEC program (DMR-1719875), and SAP America.

References

Bun, M., Desfontaines, D., Dwork, C., Naor, M., Nissim, K., Roth, A., Smith, A., Steinke, T., Ullman, J., and Vadhan, S. (2021). Statistical inference is not a privacy violation. `https://differentialprivacy.org/inference-is-not-a-privacy-violation/`, Last accessed on 2022-05-19.

Busa-Fekete, R. I., Syed, U., Vassilvitskii, S., et al. (2021). On the pitfalls of label differential privacy. In *NeurIPS 2021 Workshop LatinX in AI*.

Carlini, N., Liu, C., Erlingsson, Ú., Kos, J., and Song, D. (2019). The secret sharer: Evaluating and testing unintended memorization in neural networks. In *28th USENIX Security Symposium (USENIX Security 19)*, pages 267–284.

Chapelle, O., Manavoglu, E., and Rosales, R. (2014). Simple and scalable response prediction for display advertising. *ACM Transactions on Intelligent Systems and Technology (TIST)*, 5(4):1–34.

Chaudhuri, K. and Hsu, D. (2011). Sample complexity bounds for differentially private learning. In Kakade, S. M. and von Luxburg, U., editors, *Proceedings of the 24th Annual Conference on Learning Theory*, volume 19 of *Proceedings of Machine Learning Research*, pages 155–186, Budapest, Hungary. PMLR.

Dong, J., Roth, A., and Su, W. J. (2019). Gaussian differential privacy. *arXiv preprint arXiv:1905.02383*.

Dorogush, A. V., Ershov, V., and Gulin, A. (2018). Catboost: gradient boosting with categorical features support. *arXiv preprint arXiv:1810.11363*.

Dwork, C., McSherry, F., Nissim, K., and Smith, A. (2006). Calibrating noise to sensitivity in private data analysis. In *Theory of cryptography conference*, pages 265–284. Springer.

Dwork, C., Roth, A., et al. (2014). The algorithmic foundations of differential privacy. *Found. Trends Theor. Comput. Sci.*, 9(3-4):211–407.

Esmaeili, M. M., Mironov, I., Prasad, K., Shilov, I., and Tramer, F. (2021). Antipodes of label differential privacy: PATE and ALIBI. In *Thirty-Fifth Conference on Neural Information Processing Systems*.

Fredrikson, M., Jha, S., and Ristenpart, T. (2015). Model inversion attacks that exploit confidence information and basic countermeasures. In *Proceedings of the 22nd ACM SIGSAC conference on computer and communications security*, pages 1322–1333.

Friedman, J. H. (2001). Greedy function approximation: a gradient boosting machine. *Annals of statistics*, pages 1189–1232.

Ghazi, B., Golowich, N., Kumar, R., Manurangsi, P., and Zhang, C. (2021). Deep learning with label differential privacy. In *Thirty-Fifth Conference on Neural Information Processing Systems*.

Ghosh, A., Roughgarden, T., and Sundararajan, M. (2012). Universally utility-maximizing privacy mechanisms. *SIAM Journal on Computing*, 41(6):1673–1693.

Kairouz, P., Oh, S., and Viswanath, P. (2015). The composition theorem for differential privacy. In *International conference on machine learning*, pages 1376–1385. PMLR.

Kasiviswanathan, S. P. and Smith, A. (2014). On the'semantics' of differential privacy: A bayesian formulation. *Journal of Privacy and Confidentiality*, 6(1).

Krizhevsky, A., Hinton, G., et al. (2009). Learning multiple layers of features from tiny images.

LeCun, Y., Bottou, L., Bengio, Y., and Haffner, P. (1998). Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2324.

McMahan, H. B., Holt, G., Sculley, D., Young, M., Ebner, D., Grady, J., Nie, L., Phillips, T., Davydov, E., Golovin, D., et al. (2013). Ad click prediction: a view from the trenches. In *Proceedings of the 19th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 1222–1230.

McSherry, F. (2016). Statistical inference considered harmful. `https://github.com/frankmcsherry/blog/blob/master/posts/2016-06-14.md`, Last accessed on 2022-05-19.

Mironov, I. (2017). Rényi differential privacy. In *2017 IEEE 30th Computer Security Foundations Symposium (CSF)*, pages 263–275. IEEE.

Papernot, N., Abadi, M., Erlingsson, U., Goodfellow, I., and Talwar, K. (2016). Semi-supervised knowledge transfer for deep learning from private training data. *arXiv preprint arXiv:1610.05755*.

Ricci, F., Rokach, L., and Shapira, B. (2011). Introduction to recommender systems handbook. In *Recommender systems handbook*, pages 1–35. Springer.

Richardson, M., Dominowska, E., and Ragno, R. (2007). Predicting clicks: estimating the click-through rate for new ads. In *Proceedings of the 16th international conference on World Wide Web*, pages 521–530.

Shokri, R., Stronati, M., Song, C., and Shmatikov, V. (2017). Membership inference attacks against machine learning models. In *2017 IEEE Symposium on Security and Privacy (SP)*, pages 3–18. IEEE.

Sohn, K., Berthelot, D., Carlini, N., Zhang, Z., Zhang, H., Raffel, C. A., Cubuk, E. D., Kurakin, A., and Li, C.-L. (2020). Fixmatch: Simplifying semi-supervised learning with consistency and confidence. *Advances in Neural Information Processing Systems*, 33.

Sun, J., Yang, X., Yao, Y., and Wang, C. (2022). Label leakage and protection from forward embedding in vertical federated learning. *arXiv preprint arXiv:2203.01451*.

Tallis, M. and Yadav, P. (2018). Reacting to variations in product demand: An application for conversion rate (cr) prediction in sponsored search. In *2018 IEEE International Conference on Big Data (Big Data)*, pages 1856–1864. IEEE.

Warner, S. L. (1965). Randomized response: A survey technique for eliminating evasive answer bias. *Journal of the American Statistical Association*, 60(309):63–69.

Zhang, M., Lee, J., and Agarwal, S. (2021). Learning from noisy labels with no change to the training process. In *International Conference on Machine Learning*, pages 12468–12478. PMLR.

Zhu, L., Liu, Z., and Han, S. (2019). Deep leakage from gradients. *Advances in neural information processing systems*, 32.

A Proof of Theorem 3

Theorem 3. *Assume each data $(X_i, \mathbf{y}_i)$ is sampled independently. If f satisfies (ε, δ)-DP then for any attack algorithm $\mathcal{A}$, we have:*

$$\mathrm{Adv}^w\left(\mathcal{A}, \mathcal{K}\right) \leq \left(1 - \frac{2}{1+e^{\varepsilon}}(1-\delta)\right) \cdot \left(\frac{1}{n}\sum_{i=1}^{n} \mathbb{E}_{\mathbf{y}_i}\left[\sup_{y \in \mathcal{Y}} u(y, \mathbf{y}_i)\right]\right).$$

Proof. The proof follows similarly from proof of Theorem 3. First note that the adversary's inferred label vector $\hat{\mathbf{y}} = \mathcal{A}(\mathcal{M}(X, \mathbf{y}))$ is a random variable that depends on both the sampling of training labels $\mathbf{y}$ and randomness in the learning algorithm $\mathcal{M}$. Then:

$$\mathbb{E}[u(\hat{\mathbf{y}}_i, \mathbf{y}_i) | X_{-i}, \mathbf{y}_{-i}] = \mathbb{E}_{\mathbf{y}_i, X_i}\left[\mathbb{E}[u(\hat{\mathbf{y}}_i, \mathbf{y}_i) | \mathbf{y}, X]\right], \tag{9}$$

where the equality holds by the assumption that $\mathbf{y}_i$ is independent of X_{-i} and $\mathbf{y}_{-i}$. By DP, we have:

$$\begin{aligned}
\mathbb{E}[u(\hat{\mathbf{y}}_i, \mathbf{y}_i) | \mathbf{y}, X] &= \int_0^{B(\mathbf{y}_i)} \mathbb{P}(u(\hat{\mathbf{y}}_i, \mathbf{y}_i) > v | \mathbf{y}, X)\, \mathrm{d}v \\
&\leq \int_0^{B(\mathbf{y}_i)} \left(\mathbb{P}(u(\hat{\mathbf{y}}'_i, \mathbf{y}_i) > v | \mathbf{y}, X) + \left(1 - \frac{2}{1+e^{\varepsilon}}(1-\delta)\right)\right) \mathrm{d}v \\
&= \mathbb{E}[u\left(\hat{\mathbf{y}}'_i, \mathbf{y}_i\right) | \mathbf{y}, X] + \left(1 - \frac{2}{1+e^{\varepsilon}}(1-\delta)\right) \cdot B(\mathbf{y}_i),
\end{aligned}$$

where the inequality once again follows the Remark A.1 in Kairouz et al. (2015). Substituting the above inequality into Equation 9 gives:

$$\begin{aligned}
\mathbb{E}[u(\hat{\mathbf{y}}_i, \mathbf{y}_i) | X_{-i}, \mathbf{y}_{-i}] &\leq \mathbb{E}\left[u\left(\hat{\mathbf{y}}'_i, \mathbf{y}_i\right) | X_{-i}, \mathbf{y}_{-i}\right] + \left(1 - \frac{2}{1+e^{\varepsilon}}(1-\delta)\right) \cdot \mathbb{E}_{\mathbf{y}_i, X_i}\left[B(\mathbf{y}_i)\right] \\
&\leq \max_{y \in \mathcal{Y}} \mathbb{E}_{\mathbf{y}_i, X_i}[u(y, \mathbf{y}_i)] + \left(1 - \frac{2}{1+e^{\varepsilon}}(1-\delta)\right) \cdot \mathbb{E}_{\mathbf{y}_i, X_i}\left[B(\mathbf{y}_i)\right],
\end{aligned}$$

where the last inequality holds by the fact that $\hat{\mathbf{y}}'_i$ is independent of $\mathbf{y}_i$ conditioned on X_{-i} and $\mathbf{y}_{-i}$. Finally, we can derive our bound for the advantage $\mathrm{Adv}\left(\mathcal{A}, \mathcal{K}\right)$:

$$\begin{aligned}
\mathrm{EAU}\left(\mathcal{A}, \mathcal{K}\right) - \frac{1}{n}\sum_{i=1}^{n} \max_{y \in \mathcal{Y}} \mathbb{E}_{\mathbf{y}_i}\left[u(y, \mathbf{y}_i)\right] &= \frac{1}{n}\sum_{i=1}^{n} \left(\mathbb{E}_{\mathbf{y}_{-i}} \mathbb{E}[u(\hat{\mathbf{y}}_i, \mathbf{y}_i) | X_{-i}, \mathbf{y}_{-i}] - \max_{y \in \mathcal{Y}} \mathbb{E}_{\mathbf{y}_i}\left[u(y, \mathbf{y}_i)\right]\right) \\
&\leq \left(1 - \frac{2}{1+e^{\varepsilon}}(1-\delta)\right) \cdot \left(\frac{1}{n}\sum_{i=1}^{n} \mathbb{E}_{\mathbf{y}_i}\left[\sup_{y \in \mathcal{Y}} u(y, \mathbf{y}_i)\right]\right).
\end{aligned}$$

□

B Experiment Details

The implementation details of label-DP algorithms for Criteo CTR prediction dataset are listed as below.

1. LP-1ST (RR): LP-1ST is a one-stage version of LP-MST. Notice that it is equivalent to Randomized Response (RR) Warner (1965), which flips each training label to other labels uniformly to satisfy the label-DP.
2. LP-1ST (domain prior): We follow the Algorithm 4 in Ghazi et al. (2021) to compute the prior label distribution for each data and feed these prior distributions to LP-1ST. We set the number of clusters as 100. Before the clustering, we encode all categorical features into one-hot representations and normalize integer features into the range $[0, 1]$.
3. LP-1ST (noise correction): LP-1ST injects uniform noise to the training labels before the training of gradient boost. We can additionally adapt a post-training noise correction method in Zhang et al. (2021) for LP-1ST to achieve better performance.
4. LP-2ST: LP-2ST is the two-stage version of LP-MST.
5. PATE: We make two adaptions for the original PATE to keep the information of those minority labels of 1:
   - When a trained teacher predicts the label for a data point, instead of outputting the label that has maximum probability score, we sample a label from the output probability vector.

- When we aggregate the prediction from each teacher for student's training, instead of outputting the label with maximum count in the noisy histogram of teacher's predictions, we again sample a label from the probability, normalized from the noisy prediction histogram.

Without the above two adaptions, as we empirically verified, all aggregated labels would be 0 and the student model would be meaningless. Moreover, we perform data-independent privacy cost accounting following Papernot et al. (2016) and obtain different ε by varying number of queries with fixed noise level.