Label Differential Privacy and Private Training Data Release

Róbert Busa-Fekete [1] **Andres Muñoz Medina** [1] **Umar Syed** [1] **Sergei Vassilvitskii** [1]

Abstract

We study differentially private mechanisms for sharing training data in machine learning settings. Our goal is to enable learning of an accurate predictive model while protecting the privacy of each user's label. Previous work established privacy guarantees that assumed the features are public and given exogenously, a setting known as *label differential privacy*. In some scenarios, this can be a strong assumption that removes the interplay between features and labels from the privacy analysis. We relax this approach and instead assume the features are drawn from a distribution that depends on the private labels. We first show that simply adding noise to the label, as in previous work, can lead to an arbitrarily weak privacy guarantee, and also present methods for estimating this privacy loss from data. We then present a new mechanism that replaces some training examples with synthetically generated data, and show that our mechanism has a much better privacy-utility tradeoff if the synthetic data is realistic, in a certain quantifiable sense. Finally, we empirically validate our theoretical analysis.

1. Introduction

We consider the problem of privately sharing a dataset with a learner. We are motivated by machine learning competitions such as those hosted by Netflix (Bell and Koren, 2007), Yahoo! (Chapelle and Chang, 2010), Criteo (Diemert et al., 2022) and Kaggle (Kaggle). We consider the problem both from the perspective of individual users who must decide whether to contribute their labeled example to the dataset, as well as a learner whose goal is to train a model that accurately predicts the labels of new examples. We focus on the local model of *differential privacy*, a mathematically rigorous definition of privacy protection (Dwork and Roth, 2014).

[1]Google Research. Correspondence to: All authors <{busarobi,ammedina,usyed,sergeiv}@google.com>.

Proceedings of the 40^{th} International Conference on Machine Learning, Honolulu, Hawaii, USA. PMLR 202, 2023.

Since the dataset will be shared as part of a competition, we also do not want to restrict the learner to any particular training algorithm or model class.

Differentially private mechanisms typically protect all the data belonging to a user. But for many types of real-world training data, only the label contains sensitive information. *Label differential privacy* is a relaxation of differential privacy that protects each training label, but not necessarily the features (Chaudhuri and Hsu, 2011). Early mechanisms that satisfied label differential privacy were bound to a specific training algorithm and were very inefficient (Beimel et al., 2013), while recent work has proposed more practical approaches that are based on applying randomized response to each training label (Esfandiari et al., 2022; Ghazi et al., 2021), which gives the learner the flexibility to use essentially any training algorithm.

A key premise underlying all previous work on label differential privacy is that the features are public knowledge. However, the ramifications of this assumption may not always align with user's privacy expectations. For example, label differential privacy does not protect users from *label inference attacks* (Wu et al., 2022) that use the features to predict the training labels. Also, the assumption that an attacker has *a priori* access to the features is a poor match for many applications, notably when the attacker obtains all training data, including both the features and labels, from the same source.

For example, consider a hospital that wants to launch a machine learning competition to diagnose a sensitive illness from X-ray images. The hospital must ask patients for consent to share their data with external parties. Neither the patients' illness status nor their X-ray images are public knowledge, and while the X-rays are not sensitive *per se*, patients are unlikely to be satisfied with a data sharing mechanism that allows a learner to accurately predict whether they have the illness from their X-rays. At the same time, we want the learner to be able to train a model that can accurately diagnose the illness in new patients from their X-rays. *In other words, we want a model that has low accuracy on the training set, but high accuracy on the test set.*

Our contributions. The traditional formalization of machine learning assumes that in a training set every example

$z = (x, y)$ with features x and label y is drawn from some fixed but unknown joint distribution P on features and labels. A training set $Z = \{z_1, z_2, \ldots, z_t\}$ is generated by drawing each example z_i identically and independently from P.

The standard definition of label differential privacy assumes that the training set Z is given, and the goal is to learn a hypothesis from Z while protecting the label y_i of every user. Note, that by design this eschews worrying about any correlations between features and labels that are present in P. In this work, on the other hand, we assume each x_i is generated from the conditional distribution of P given y_i and ask how to protect the label y_i in this case. To distinguish between the two scenarios, we call the former label differential privacy with *static* features, and the latter as label differential privacy with *conditional* features.

Our specific contributions are as follows:

- We propose a new privacy definition, *label differential privacy with conditional features*. Our definition is an alternative to the existing definition of label differential privacy, which assumes that the features are public knowledge and static. Instead, we assume that the features are drawn from a distribution that depends on the private label. We show that any mechanism that satisfies our new definition also provably prevents label inference attacks on the training set, in contrast to the existing definition.
- We prove that a large class of mechanisms that satisfy label differential privacy with static features do not satisfy label differential privacy with conditional features. All recently proposed mechanisms for label differential privacy are instances of this class. We complement the worst-case analysis with an algorithm for empirically estimating the privacy loss of mechanisms in this class.
- We propose a new mechanism that satisfies label differential privacy with static *or* conditional features. We assume that our mechanism has access to a synthetic feature generator than can produce 'realistic' features.
- We present experiments showing that our mechanism is significantly less vulnerable to label inference attacks on the training set than existing mechanisms that only satisfy label differential privacy with static features, while also enabling learning a model that makes nearly as accurate predictions on the test set.

2. Related Work

Label differential privacy was introduced by Chaudhuri and Hsu (2011), who also proved lower bounds on the sample complexity of any private learner. Beimel et al. (2013) were the first to explicitly describe a learning algorithm that satisfies label differential privacy, but its running time is exponential in the worst-case, since it is based on enumerating a cover of the model class. Tractable mechanisms have been proposed more recently (Bassily et al., 2018; Wang and Xu, 2019; Ghazi et al., 2021; Esfandiari et al., 2022), including mechanisms that generate private training data that can be used as input to any learning algorithm.

While previous work has shown that label-differentially private mechanisms can learn highly accurate predictive models, Wu et al. (2022) observed that these mechanisms are also vulnerable to attacks that use the training features to infer the training labels. Wu et al. (2022) did not propose any methods for preventing label inference attacks, but instead suggested quantifying their severity relative to what can be inferred about the labels by an adversary who knows the Bayes optimal classifier. By contrast, we propose a novel definition of label differential privacy that provably prevents label inference attacks.

Our definition of label differential privacy assumes that a training example's features are drawn from a distribution that depends on its private label. Other privacy frameworks also make distributional assumptions about the data, such as Bayesian differential privacy (Triastcyn and Faltings, 2020). One criticism of these frameworks is that an average-case analysis is inappropriate for establishing privacy guarantees, which should hold in the worst case (Steinke and Ullman, 2020). But we emphasize that in our proposed definition only the *non-private* features are randomly distributed; the private labels can be arbitrary.

3. Preliminaries

Let $\mathcal{X}$ be the feature space and $\mathcal{Y} = \{1, \ldots, k\}$ be the label space, where $k > 1$. A *labeled example* is an element of $\mathcal{X} \times \mathcal{Y}$, typically denoted (x, y), and a *dataset* is an element of $(\mathcal{X} \times \mathcal{Y})^n$, typically denoted $(\mathbf{x}, \mathbf{y})$.

If P is a distribution on $\mathcal{X} \times \mathcal{Y}$ then $P_{\mathcal{Y}}$ is its marginal distribution on labels $\mathcal{Y}$. Also, $P_{\mathcal{X}|y}$ is the conditional distribution of P on features $\mathcal{X}$ given label $y \in \mathcal{Y}$, and $P_{\mathcal{Y}|x}$ is the conditional distribution of P on labels $\mathcal{Y}$ given features $x \in \mathcal{X}$. Let $f_P : \mathcal{Y} \mapsto \mathcal{X}$ be the random function that, given label $y \in \mathcal{Y}$, outputs features $x \in \mathcal{X}$ according to distribution $P_{\mathcal{X}|y}$.

Let z_i denote the ith component of vector $\mathbf{z}$. For any function $f(z)$ let $f^n(\mathbf{z})$ be the function that applies f componentwise to $\mathbf{z}$, such that if $\mathbf{z}' = f^n(\mathbf{z})$ then $z'_i = f(z_i)$. For any distribution P let $\mathbf{z} \sim P^n$ denote that each z_i is independent and distributed according to P.

For any distribution P let $P(z)$ denote the density of P at z. Several of the privacy definitions we study in this paper are based on the following definition of the divergence between

two distributions.

Definition 3.1 (Rényi α-divergence)**.** For any $\alpha > 1$ the *Rényi α-divergence* between distributions P and Q is

$$D_\alpha(P\|Q) = \frac{1}{\alpha-1}\log \mathrm{E}_{z\sim Q}\left[\left(\frac{P(z)}{Q(z)}\right)^\alpha\right].$$

A *mechanism* is a random function. With an abuse of notation, if M is a mechanism then we write $D_\alpha(M(z)\|M(z'))$ to denote the Rényi α-divergence between the distributions of $M(z)$ and $M(z')$.

4. Privacy Definitions

In this section we present and compare several definitions of the privacy of a mechanism. All the mechanisms we study are *local*; given a private labeled example they output a noisy version of the labeled example. Thus a dataset is privately released by having each user independently apply a mechanism to their labeled example.

Definitions 4.1, 4.2 and 4.3 below are based on the framework of Rényi differential privacy (Mironov, 2017), which quantifies how well an attacker can infer the input to a mechanism from its output. Definition 4.4 below quantifies the vulnerability of a mechanism to a label inference attack, and is based on a definition introduced by Wu et al. (2022).

Definition 4.1 (Differential privacy)**.** Let $\alpha > 1$ and $\varepsilon \geq 0$. Mechanism $M : \mathcal{X} \times \mathcal{Y} \mapsto \mathcal{X} \times \mathcal{Y}$ satisfies (α, ε)-*Rényi differential privacy* if for all $x, x' \in \mathcal{X}$ and $y, y' \in \mathcal{Y}$

$$D_\alpha(M(x,y)\|M(x',y')) \leq \varepsilon.$$

Rényi differential privacy is closely related to both ε-differential privacy and (ε, δ)-differential privacy (Dwork and Roth, 2014), with larger values of α indicating stronger privacy. In particular, (∞, ε)-RDP is equivalent to ε-differential privacy, and (α, ε)-RDP implies (ε', δ)-differential privacy for all $\delta \in (0, 1)$ and $\varepsilon' = \varepsilon + \frac{\log(1/\delta)}{\alpha-1}$ (Mironov, 2017).

Definition 4.1 protects the privacy of both the label and the features. But this is overkill for applications where the features are not sensitive, and will therefore introduce much more noise than necessary. An alternative is to protect the privacy of the labels only, by assuming the features are fixed and publicly available.

Definition 4.2 (Label differential privacy with static features)**.** Let $\alpha > 1$ and $\varepsilon \geq 0$. Mechanism $M : \mathcal{X} \times \mathcal{Y} \mapsto \mathcal{X} \times \mathcal{Y}$ satisfies (α, ϵ)-*label Rényi differential privacy* if for all $x \in \mathcal{X}$ and $y, y' \in \mathcal{Y}$

$$D_\alpha(M(x,y)\|M(x,y')) \leq \varepsilon.$$

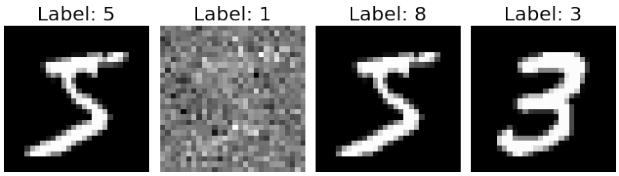

Figure 1. From left to right: The original training example. Noisy label and feature vector (satisfies Definition 4.1). Noisy label only (satisfies Definition 4.2). Noisy label and sample synthetic but realistic examples (satisfies Definition 4.3).

Definition 4.2 is a straightforward generalization of the definition of label differential privacy proposed by Chaudhuri and Hsu (2011), with Rényi differential privacy used in place of ε-differential privacy. Note that the same features x appear on both sides of the inequality in Definition 4.2, which implies that the features are public, and so this definition does not penalize the mechanism for revealing information about the label via the features. But there are many applications where the learner has no *a priori* access to the features.

In this paper we propose a new privacy definition that is suitable for applications where the features are not assumed to be sensitive (as in Definition 4.1), but also not assumed to be public (as in Definition 4.2). Instead, Definition 4.3 assumes the features are randomly drawn from a distribution that depends on the private label.

Definition 4.3 (Label differential privacy with conditional features)**.** Let $\alpha > 1$, $\varepsilon \geq 0$ and P be a distribution on $\mathcal{X} \times \mathcal{Y}$. Mechanism $M : \mathcal{X} \times \mathcal{Y} \mapsto \mathcal{X} \times \mathcal{Y}$ satisfies (α, ε, P)-*label Rényi differential privacy* if for all $y, y' \in \mathcal{Y}$

$$D_\alpha(M(f_P(y), y)\|M(f_P(y'), y')) \leq \varepsilon$$

where $f_P : \mathcal{Y} \mapsto \mathcal{X}$ is the random function that, given input y, outputs x according to $P_{\mathcal{X}|y}$.

Definition 4.3 only penalizes a mechanism for information it reveals about the features if doing so indirectly reveals information about the label via their correlation according to P. Figure 1 shows different noise models and how they fit the above definitions.

Our final privacy definition quantifies the ability of an attacker to recover the labels of the dataset that was input to a mechanism. An *attack algorithm* is a random function that, given a dataset generated by a mechanism, outputs a prediction of each label in the original dataset. Definition 4.4 bounds the average loss of these predictions assuming the dataset is an i.i.d. sample from P.

Definition 4.4 (Expected attack utility)**.** Let $\varepsilon \geq 0$. Let P be a distribution on $\mathcal{X} \times \mathcal{Y}$. Let $\ell : \mathcal{Y} \times \mathcal{Y} \mapsto [0, 1]$ be a loss function. Mechanism $M : \mathcal{X} \times \mathcal{Y} \mapsto \mathcal{X} \times \mathcal{Y}$

has (ε, P)-*expected attack utility* if for any attack algorithm $A : (\mathcal{X} \times \mathcal{Y})^n \mapsto \mathcal{Y}^n$ we have

$$\mathrm{E}_{\substack{(\mathbf{x},\mathbf{y})\sim P^n,\\ (\mathbf{x}',\mathbf{y}')\sim M^n(\mathbf{x},\mathbf{y}),\\ \hat{\mathbf{y}}\sim A(\mathbf{x}',\mathbf{y}')}}\left[\frac{1}{n}\sum_{i=1}^{n}\ell(y_i,\hat{y}_i)\right] \geq 1-\varepsilon.$$

Definition 4.4 is almost identical to the definition of expected attack utility proposed by Wu et al. (2022), except in our definition the features are drawn from a distribution instead of being fixed. As with differential privacy, smaller values of ε in the definition of expected attack utility indicate a stronger privacy guarantee.

Importantly, while a small expected attack utility implies that no attacker can fully recover the *training* labels using the output of the mechanism, it may still be possible to use the output of the mechanism to accurately predict unseen *test* labels. Indeed, in Section 8 we demonstrate empirically that the output of a mechanism with low expected attack utility can be used to estimate a good predictive model.

5. Preventing Label Inference Attacks

Wu et al. (2022) showed that the most widely used definition of label differential privacy (Definition 4.2) does not prevent label inference attacks.

Theorem 5.1 (Wu et al. (2022))**.** *For any $\varepsilon \geq 0$ there exists an (∞, ε)-label Rényi differentially private mechanism M and a distribution P on $\mathcal{X} \times \mathcal{Y}$ such that M does not have $(\tilde{\varepsilon}, P)$-expected attack utility for any $\tilde{\varepsilon} < 1$.*

Note that every mechanism trivially has $(1, P)$-expected attack utility for any distribution P, since the loss function is bounded between 0 and 1. So Theorem 5.1 says that even a mechanism with arbitrarily strong label differential privacy has no non-trivial bound on its expected attack utility when the features are public. This can happen whenever an example's features are a strong predictor of its label, because if the features are public, no amount of noise added to the training labels can prevent a label inference attack.

In contrast to Theorem 5.1, we show in Theorem 5.2 that our definition of label differential privacy (Definition 4.3), which does not assume that the features are public, is strong enough to prevent label inference attacks.

Theorem 5.2. *For any $\varepsilon \geq 0, \delta \in (0,1), \alpha > 1$ and distribution P on $\mathcal{X} \times \mathcal{Y}$, if mechanism M is (α, ε, P)-label Rényi differentially private then M has $(\tilde{\varepsilon}, P)$-expected attack utility, where*

$$\tilde{\varepsilon} = 1 - \left(\exp(-\varepsilon)\delta^{\frac{1}{\alpha-1}}(1-p^*) - (k-1)\delta\right)\ell^*,$$

$p^* = \max_y P_{\mathcal{Y}}(y)$ is the marginal probability of the most common label, and $\ell^* = \min_{y\neq\hat{y}} \ell(y,\hat{y})$ is the minimum loss when the predicted label differs from the true label.

Note that $\tilde{\varepsilon}$ in Theorem 5.2 can be made arbitrarily close to $1-(1-p^*)\ell^*$ from above by choosing ε, δ sufficiently small and α sufficiently large. Also note that if the loss function $\ell(y,\hat{y}) = \mathbf{1}[y \neq \hat{y}]$, then an attacker can trivially achieve expected attack utility of $(\tilde{\varepsilon}, P)$ with $\tilde{\varepsilon} = 1-(1-p^*)\ell^*$ by always predicting the most common label, since that strategy will mispredict $1-p^*$ fraction of the labels on average. Thus Theorem 5.2 is asymptotically tight, and shows that a sufficiently private mechanism (when privacy is defined as in Definition 4.3) can prevent any non-trivial label inference attack.

6. Quantifying Privacy Loss

Many previously proposed mechanisms that satisfy label differential privacy with static features are *model-based*. These mechanisms use a model to replace some of the labels in the original training set with synthetic labels, and then train a new model on the partially-synthetic training set using any non-private learning algorithm. This approach can be iterated in an bootstrap fashion, by first partitioning the original training set, and then successively applying the mechanism and learning algorithm to each partition. It has been shown empirically that this technique can learn high-quality predictive models (Esfandiari et al., 2022; Ghazi et al., 2021).

Mechanism 1 below includes several model-based mechanisms from the literature as special cases (Esfandiari et al., 2022; Ghazi et al., 2021). The mechanism performs randomized response on the label only, choosing a random label from a synthetic distribution that depends on the features. The features themselves are output without any randomization.

Mechanism 1 Model-based randomized response on label

Given: Noise level $\lambda \in [0,1]$; synthetic label distribution $\hat{P}_{\mathcal{Y}|x}$ for each feature $x \in \mathcal{X}$.
Input: Labeled example (x, y).
Output: Noisy labeled example $(x, \tilde{y})$.
with probability $1-\lambda$:
    Let $\tilde{y} = y$.
otherwise:
    Draw $\tilde{y}$ from distribution $\hat{P}_{\mathcal{Y}|x}$.
return $(x, \tilde{y})$.

While Mechanism 1 satisfies label differential privacy with static features, Theorem 6.1 shows that it cannot satisfy any non-trivial guarantee for label differential privacy with conditional features.

Theorem 6.1. *There is a distribution P such that Mechanism 1 does not satisfy (α, ε, P)-label Rényi differential privacy for any $\alpha > 1$ and $\varepsilon < \infty$.*

Theorem 6.1 proves that there *exists* a single distribution P for which Mechanism 1 has arbitrarily bad privacy. In the remainder of this section, we describe methods for empirically estimating the privacy loss of Mechanism 1 with respect to a *given* distribution. For ease of exposition, we focus on the special case of binary labels (*i.e.*, $\mathcal{Y} = \{0, 1\}$) and where each synthetic label distribution $\hat{P}_{\mathcal{Y}|x}$ in Mechanism 1 is the same for all $x \in \mathcal{X}$. Accordingly, the mechanisms we study in this section have the form $M\colon (x, y) \to (x, M'(y))$ where $M'\colon \{0, 1\} \to \{0, 1\}$ is a fixed randomized function.

We begin by providing a semantic interpretation of the concept of label differential privacy with conditional features.

Theorem 6.2. *Let $M\colon (x, y) \to (x, M'(y))$ where $M'(y)$ is a randomized mechanism with output $\widetilde{Y}$. Let P_y denote the distribution over $\mathcal{X} \times \mathcal{Y}$ induced by $(f_P(y), M'(y))$. Then M satisfies*

$$D_\alpha(M(f_P(y), y)||M(f_P(1-y), 1-y)) = \frac{1}{\alpha - 1} \log \mathrm{E}_{P_y}\left[e^{(\alpha-1)\nu(y,\widetilde{Y},X)}\right]$$

where $\nu(y, \widetilde{y}, x)$ is defined by

$$\frac{P(Y = y|X = x, \widetilde{Y} = \widetilde{y})}{P(Y = 1 - y|X = x, \widetilde{Y} = \widetilde{y})} = \frac{e^{\nu(y,\widetilde{y},x)} P(Y = y)}{P(Y = 1 - y)}.$$

That is, label differential privacy with conditional features is proportional to the cumulant generating function at $\alpha - 1$ of the random variable $\nu(y, \widetilde{Y}, X)$.

We call $\nu(y, \widetilde{y}, x)$ from Theorem 6.2 the *instance-based privacy loss function*, since it quantifies the information about the label Y gained by an observer of the output of the mechanism as well as the feature vector x. This interpretation further confirms that our notion of privacy is capturing the information leakage on the label due to the release of features. The following corollary, which is a consequence of Jensen's inequality, makes this connection more explicit:

Corollary 6.3. *The following inequality holds for all $\alpha > 1$*

$$\mathrm{E}_{y\sim P_{\mathcal{Y}}}\left[D_\alpha(M(f_P(y), y)||M(f_P(1\text{–}y), 1\text{–}y))\right] \geq \mathrm{E}[\nu(Y, \widetilde{Y}, X)] \tag{1}$$

That is, if the randomized response mechanism is (α, ϵ, P)-label differentially private, then the expected instance-based privacy loss on the true label must also be less than ϵ. In view of this, the following theorem provides a sufficient condition on P for the mechanism to not be private for any $\epsilon < \infty$.

Theorem 6.4. *The instance-based privacy loss function can be written as*

$$e^{\nu(y,\widetilde{y},x)} = \frac{P(\widetilde{Y} = \widetilde{y}|Y = y)}{P(\widetilde{Y} = y|Y = 1 - y)} \times \frac{P(Y = y|X = x)}{P(Y = 1 - y|X = x)} \frac{P(Y = 1 - y)}{P(Y = y)}. \tag{2}$$

In particular if the set $A = \{x\colon P(Y = 1-y|X = x) = 0\}$ is such that $P_{\mathcal{X}}(A) > 0$ then the expectation in (1) would be infinite and consequently the mechanism wouldn't be label differentially private with conditional features.

A second interpretation of Corollary 6.3 is that any estimate on $\mathrm{E}[\nu(Y, \widetilde{Y}, X)]$ provides a lower bound on the privacy leakage of mechanism M. Instead of directly estimating this quantity however, we opt to estimate the following distribution

$$J(\tau) = P(\nu(X) \geq \tau)$$

where $\nu(x) = \mathrm{E}[\nu(Y, \widetilde{Y}, x)]$. This not only allows us to derive a lower bound on (1) via Markov's inequality (or by integrating $J(\tau)$) but also provides us with a better semantic interpretation of the privacy loss by answering the question: if one were to release data using mechanism M, for what fraction of the users would the information gain on their label be greater than τ? We conclude the section by presenting an estimator for $J(\tau)$. Our estimator requires the feature space $\mathcal{X}$ to have a metric ρ and a standard smoothness assumption on the so called regression function $\eta(x)\colon P(Y = 1|X = x)$.

Assumption 6.5 (Smoothness assumption)**.** Let $\beta \in (0, 1]$, $C_\beta > 0$. We say that P satisfies the measure-smoothness assumption with parameters β, C_β if the following holds for all $x_0, x_1 \in \mathcal{X}$: $|\eta(x_0) - \eta(x_1)| \leq C_\beta \cdot \mu\left(B_{\rho(x_0,x_1)}(x_0)\right)^\beta$, where $B_r(x)$ is the ball of radius r centered at x and $\mu(A) = \Pr[X \in A]$ is the marginal distribution of the features.

Theorem 6.6. *Let $\delta > 0$ and let $\mathcal{S}_n = (x_i, y_i)_{i=1}^n$ denote an i.i.d. sample drawn from P. Suppose that Assumption 6.5 holds for β, C_β. Then there exists a nearest neighbor estimator $\widehat{J}_n$ such that with probability at least $1 - \delta$ over $\mathcal{S}_n$:*

$$J(\tau) \geq \widehat{J}_n(\tau) - 2\sqrt{\frac{\log(2/\delta)}{2n}} - \delta.$$

7. Guaranteeing Label Differential Privacy with Conditional Features

In Section 5 we showed that label differential privacy with conditional features (Definition 4.3), unlike label differential privacy with static features (Definition 4.2), implies protection against label inference attacks. In Section 6 we showed that existing model-based mechanisms do not satisfy label differential privacy with conditional features. Below we describe a new model-based mechanism (Mechanism 2) that achieves *both* privacy guarantees. Unlike previous mechanisms, our mechanism uses a model to generate synthetic *features* instead of synthetic *labels*.

Mechanism 2 performs randomized response on both the label and the features. The mechanism uses a synthetic distribution $\hat{P}_{\mathcal{X}|y}$, close to $P_{\mathcal{X}|y}$ for each possible label y to generate the features in the randomized response.

Mechanism 2 Model-based double randomized response

Given: Noise level $\lambda \in [0, 1]$; synthetic feature distribution $\hat{P}_{\mathcal{X}|y}$ for each label $y \in \mathcal{Y}$.
Input: Labeled example (x, y).
Output: Noisy labeled example $(\tilde{x}, \tilde{y})$.
with probability $1 - \lambda$:
  Let $\tilde{y} = y$.
otherwise:
  Draw $\tilde{y}$ uniformly at random from $\mathcal{Y}$.
with probability $1 - \lambda$:
  Let $\tilde{x} = x$
otherwise
  Draw y' uniformly at random from $\mathcal{Y}$.
  Draw $\tilde{x}$ from distribution $\hat{P}_{\mathcal{X}|y'}$.
return $(\tilde{x}, \tilde{y})$.

Theorem 7.1. *Let $\alpha > 1$. Let P be a distribution on $\mathcal{X} \times \mathcal{Y}$. If $D_\alpha(P_{\mathcal{X}|y} \| \hat{P}_{\mathcal{X}|y}) \leq \Delta$ for each label $y \in \mathcal{Y}$, then Mechanism 2 satisfies (α, ε)-label Rényi differential privacy* and *(α, ε, P)-label Rényi differential privacy for all*

$$\varepsilon \geq 2 \log \left(1 + \frac{(1-\lambda)k}{\lambda}\right) + \Delta.$$

Theorem 7.1 states that the privacy of Mechanism 2 depends on the quality of the synthetic distributions $\hat{P}_{\mathcal{X}|y}$. If each of these synthetic distributions is close to the true distribution $P_{\mathcal{X}|y}$ (as measured by the Rényi divergence between them) then the mechanism satisfies a strong privacy guarantee. Essentially, the synthetic data generated by the mechanism is difficult to distinguish from the original data, which protects its privacy.

To place the privacy guarantee for Mechanism 2 in context, we next consider a simple mechanism that outputs no information about the true features. Instead, Mechanism 3 performs uniform randomized response on the label while always outputting 'dummy' features. Even though the output of this mechanism is obviously not useful for learning, its privacy will serve as a baseline for comparison with Mechanism 2.

Theorem 7.2. *Let $\alpha > 1$. Let P be a distribution on $\mathcal{X} \times \mathcal{Y}$. If Mechanism 3 satisfies (α, ε, P)-label Rényi differential privacy then*

$$\varepsilon \geq \frac{\alpha}{\alpha - 1} \log \left(1 + \frac{(1-\lambda)k}{\lambda}\right) - \frac{1}{\alpha - 1} \log \frac{k}{\lambda}.$$

Comparing Theorems 7.1 and 7.2, we see that in the high-privacy regime (α and λ are large), the bound on Mechanism 2 approaches that of Mechanism 3 (modulo a factor of 2), provided that the synthetic feature distributions are good (in other words, Δ is small). So Mechanism 2 can release information about how features are correlated with labels at almost no additional privacy cost relative to a mechanism that releases no such information.

Mechanism 3 Uniform randomized response on label (and output dummy features)

Given: Noise level $\lambda \in [0, 1]$.
Input: Labeled example (x, y).
Output: Noisy labeled example $(\tilde{x}, \tilde{y})$.
Let $\tilde{x} = \perp$. *// Dummy features*
with probability $1 - \lambda$:
  Let $\tilde{y} = y$.
otherwise:
  Draw $\tilde{y}$ uniformly at random from $\mathcal{Y}$.
return $(\tilde{x}, \tilde{y})$.

In applications, many options exist for realistic synthetic feature generation. For example, deep neural networks for generating extremely realistic language and image data are now widely available (such as GPT (OpenAI, 2023) and Imagen (Saharia et al., 2022), among others). Crucially, these models are trained on publicly available data.

8. Experiments

In this section we validate the theoretical findings in our paper by conducting simulations on real-world datasets. We provide a thorough analysis for the estimation of privacy risk of mechanisms that are only label differential privacy for static features as well as a comparison of the label inference attack for both Mechanism 1 and Mechanism 2.

8.1. Privacy Loss of Randomized Response

We begin by validating the estimator of $J(\tau)$ defined in Theorem 6.6. As we pointed out, $J(\tau)$ can be used to lower bound the privacy budget ϵ with static features (see Corollary 6.3 and Definition 4.2). We present examples that show that the finite-sample estimator can help asses privacy violations and tune the flipping probability λ. We use four large scale binary classification datasets, as described in Table 1. For each dataset we computed an approximate k-nearest neighbor graph under the L_2 distance, using $k = 1000$ for SUSY and $k = 10000$ for the rest. We use the Clopper-Pearson confidence interval (Clopper and Pearson, 1934), which is an exact confidence interval, and thus tighter than the Chernoff bound used in our analysis in the Appendix.

Figure 2 shows the estimates of $J(\tau)$ according to Theorem 6.6. To interpret the figures, consider the `kag14` dataset with label flipping probability $\lambda = 0.01$ (left plot). Note that in

Name	#Train	#Test	#Feat.	$P(Y=1)$
kag14	40M	5.8M	1M	0.256
kdd12	118M	29.9M	54.6M	0.044
kdd10	19.2M	0.7M	29.8M	0.861
SUSY	4.5M	0.5M	18	0.457

Table 1. The main parameters of the benchmark datasets. kag14 dataset used in Kaggle Display Advertising Challenge and it is released by Criteo (Criteo, 2014). kdd12 dataset is the official dataset of KDD Cup 2012 Track 1 (https://www.kaggle.com/c/kddcup2012-track1) and released by Tencent Inc. kdd10 dataset is the official dataset of KDD Cup 2010 (Stamper and Koedinger, 2010). SUSY is taken from UCI repository.

this scenario the mechanism is (α, ϵ)-label differentially for static features for $\epsilon \sim 5.19$. What our plot shows it that for kag14 (at the black cross), $P(\nu(X) > 6) \sim 0.05$. This means that only about 5% of the users have a privacy risk significantly higher than the protection implied by static features.

On the other hand, consider the kdd12 dataset with label flipping probability $\lambda = 0.001$ (right plot). Here, the mechanism is label differentially private for static features for $\epsilon \sim 7.6$. However, for kdd12 we note that there is a non-trivial mass of points for which the privacy leakage, as measured by ν, can be almost 10.

8.2. Evaluation of Mechanism 2

We now empirically verify the theoretical results of our paper. We focus on measuring the label inference attack described in Definition 4.4.

Inspired by Definition 4.4, we measure the label inference accuracy of two different mechanisms on the MNIST (Deng, 2012) dataset: Mechanism 1 (Randomized Response) with $\widehat{P}_{\mathcal{Y}|x}$ constant over x and our proposed Mechanism 2 (Double Randomized Response).

Methodology. Let $\mathcal{X} = [0,1]^{28\times 28}$, $\mathcal{Y} = \{0, 9\}$ and $\mathcal{S} = ((x_i, y_i))_{i=1}^n \subset \mathcal{X} \times \mathcal{Y}$ denote the MNIST dataset with normalized features. For each mechanism M defined above let $(\widetilde{x}_i, \widetilde{y}_i) = M(x_i, y_i)$ and $\widetilde{D} = ((\widetilde{x}_i, \widetilde{y}_i))$ denote the data set released by the mechanism. Let $h\colon \mathcal{X} \to \mathcal{Y}$ denote a model trained on $\widetilde{D}$. We measure the *label inference accuracy* as $\frac{1}{n}\sum_{i=1}^n \mathbf{1}[h(\widetilde{x}_i) = y_i]$.

To understand the utility of the mechanism, we measure the ability of the learned model h to predict on a test sample $\mathcal{S}_T \subset \mathcal{X} \times \mathcal{Y}$. For this we use the tesing data of MNIST and measure the *test accuracy* as $\frac{1}{n}\sum_{(x,y)\in\mathcal{S}_T} \mathbf{1}[h(x) = y]$.

Recall that Mechanism 2 requires a feature generation function. We train a conditional GAN (Mirza and Osindero, 2014) model on the training dataset. We vary $\lambda \in (0, 0.4]$ and generate private datasets using randomized response and double randomized response with parameter λ. For the prediction task, we train a standard convolutional network using an unbiased version of the loss for each dataset (see Appendix G).

The results are present in Figure 3, which shows both the label inference accuracy and test accuracy for both mechanisms—Randomized Response and Double Randomized Response—as a function of λ. We observe that label inference accuracy remains high for the Randomized Rresponse mechanism for all values of λ. On the other hand, as expected, it drops precipitously for the Double Randomized Response mechanism. Thus our proposed mechanism offers much stronger guarantees against label inference attacks.

And yet, as the second plot shows, while there is a degradation in test accuracy as a function of λ, the Double Randomized Response mechanism remains competitive, still achieving accuracy far above 95% even when $\lambda > 1/3$. This validates the ability of the mechanism to produce a training set which can be used to train accurate models.

Finally, we describe a method for estimating the (α, ε, P)-label Rényi differential privacy of Double Randomized Response, which by Theorem 7.1 depends on both the noise level λ and the Rényi divergence $D_\alpha(P_{\mathcal{X}|y} \| \hat{P}_{\mathcal{X}|y})$ between the true and synthetic feature distributions. To estimate the latter quantity, we use a recent method from Birrell et al. (2020), who showed that $D_\alpha(P\|Q)$ can be approximated by plugging the function $\phi^*(x) = \log \frac{P(x)}{Q(x)}$ into a data-dependent variational bound (see their Theorem 3.1 and Corollary 3.2). Consider a random variable X defined by choosing $Z \in \{0, 1\}$ uniformly at random and then drawing X from P if $Z = 1$ and from Q otherwise. It is easy to see by Bayes rule that

$$\phi^*(x) = \log \frac{\Pr[Z = 1 \mid X = x]}{\Pr[Z = 0 \mid X = x]}.$$

Moreover, if $P = P_{\mathcal{X}|y}$ and $Q = \hat{P}_{\mathcal{Y}|y}$ then the conditional distribution $\Pr[Z = 1 \mid X = x]$ is exactly what is learned by the discriminator during GAN training, since the discriminator's goal is to distinguish true features from synthetic features. So we can plug the discriminator's model into the variational approximation from Birrell et al. (2020) to estimate the α-Rényi divergence between the true and synthetic feature distributions, and then apply Theorem 7.1 to estimate the privacy parameter ε of Double Randomized Response as a function of α and the noise level λ. See Figure 4 below for these estimates, which show that even for small values of λ, Double Randomized Response provides a non-trivial privacy guarantee, which improves as λ increases. By contrast, Randomized Response cannot provide a comparable privacy guarantee (see Theorem 6.1).

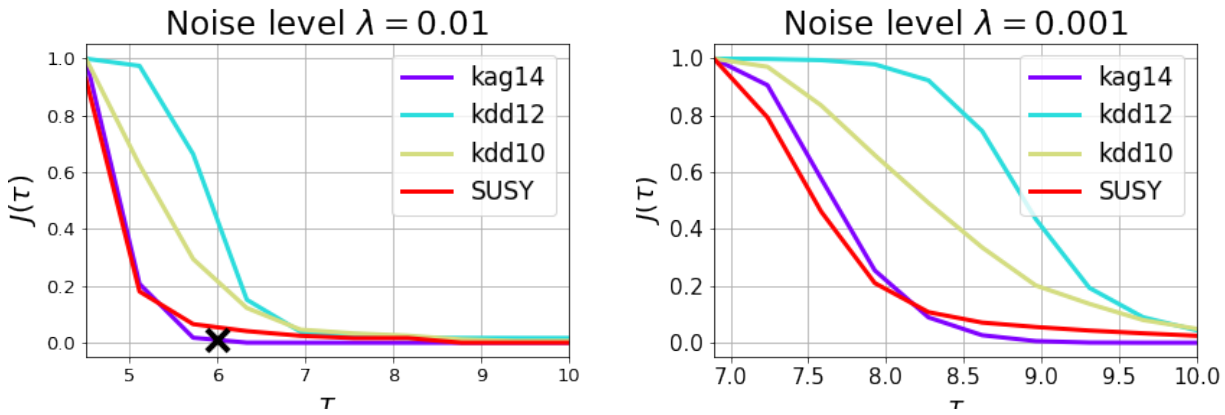

Figure 2. High probability approximation for $J(\tau)$ with $\delta = 0.01$. These graphs show $\hat{J}(\tau)$ defined in Theorem 6.6 computed on the training data.

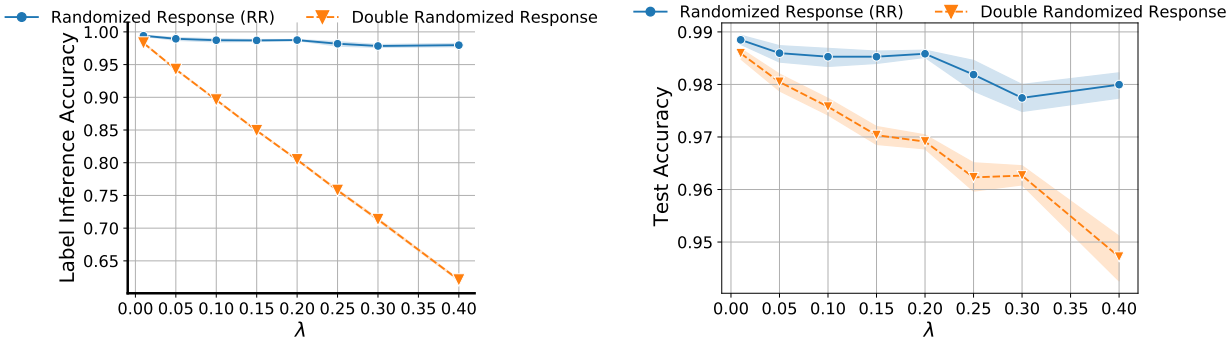

Figure 3. Evaluation of the label inference attack (left) and model accuracy (right). Observe that while Label Inference Attack risk is high for Randomized Response at all levels of λ, for the Double Randomized Response mechanism Label Inference Attack risk drops below 65%) even as the accuracy remains high (above 95%).

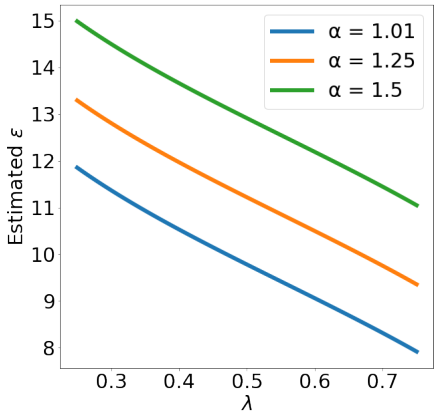

Figure 4. Estimated privacy of Double Randomized Response. The mechanism is estimated to be (α, ε, P)-label Rényi differentially private (Definition 4.3), where ε is determined by λ and α according to the figure. Note that privacy will improve if the synthetic feature distribution becomes more realistic, and that Randomized Response provides no privacy under this definition (Theorem 6.1).

9. Conclusion

We have provided a new framework for understanding the privacy risks releasing a training dataset when only the label is sensitive. By establishing a connection with the label inference attack, we demonstrated that our new privacy definition captures the risks of an attacker learning the sensitive label of a user through correlations in the dataset. We also showed that common mechanisms that protect only the label on a dataset but assume that features are static, do not satisfy our notion of privacy and provided a way for a data curator to evaluate the potential privacy risks of using such mechanisms. Finally, we proposed a new mechanism that not only satisfies the standard notion of label differential privacy for static features but also satisfies our stronger notion of privacy with conditional features. Our experiments demonstrate that when the label inference attack is a concern, our algorithm provides significantly more protection against such attacks at a negligible cost on utility.

References

R. Bassily, O. Thakkar, and A. Guha Thakurta. Model-agnostic private learning. *Advances in Neural Information Processing Systems*, 31, 2018.

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*, pages 363–378. Springer, 2013.

R. M. Bell and Y. Koren. Lessons from the netflix prize challenge. *Acm Sigkdd Explorations Newsletter*, 9(2): 75–79, 2007.

J. Birrell, P. Dupuis, M. A. Katsoulakis, L. Rey-Bellet, and J. Wang. Variational representations and neural network estimation of rnyi divergences, 2020. URL https://arxiv.org/abs/2007.03814.

O. Chapelle and Y. Chang. Yahoo! learning to rank challenge overview. In *Proceedings of the 2010 International Conference on Yahoo! Learning to Rank Challenge - Volume 14*, YLRC'10, page 124. JMLR.org, 2010.

K. Chaudhuri and D. J. Hsu. Sample complexity bounds for differentially private learning. In S. M. Kakade and U. von Luxburg, editors, *COLT 2011 - The 24th Annual Conference on Learning Theory, June 9-11, 2011, Budapest, Hungary*, volume 19 of *JMLR Proceedings*, pages 155–186. JMLR.org, 2011.

C. J. Clopper and E. S. Pearson. The use of confidence or fiducial limits illustrated in the case of the binomial. *Biometrika*, 26(4):404–413, 1934.

Criteo. http://labs.criteo.com/2014/02/kaggle-display-advertising-challenge-dataset/, 2014.

L. Deng. The mnist database of handwritten digit images for machine learning research. *IEEE Signal Processing Magazine*, 29(6):141–142, 2012.

E. Diemert, R. Fabre, A. Gilotte, F. Jia, B. Leparmentier, J. Mary, Z. Qu, U. Tanielian, and H. Yang. Lessons from the adkdd'21 privacy-preserving ml challenge, 2022. URL https://arxiv.org/abs/2201.13123.

C. Dwork and A. Roth. The algorithmic foundations of differential privacy. *Foundations and Trends® in Theoretical Computer Science*, 9(3–4):211–407, 2014.

H. Esfandiari, V. Mirrokni, U. Syed, and S. Vassilvitskii. Label differential privacy via clustering. In *International Conference on Artificial Intelligence and Statistics*, pages 7055–7075. PMLR, 2022.

B. Ghazi, N. Golowich, R. Kumar, P. Manurangsi, and C. Zhang. Deep learning with label differential privacy. *Advances in Neural Information Processing Systems*, 34: 27131–27145, 2021.

Kaggle. Kaggle competitions. https://www.kaggle.com/competitions, 2022.

I. Mironov. Rényi differential privacy. In *2017 IEEE 30th computer security foundations symposium (CSF)*, pages 263–275. IEEE, 2017.

M. Mirza and S. Osindero. Conditional generative adversarial nets. *arXiv preprint arXiv:1411.1784*, 2014.

OpenAI. Gpt-4 technical report, 2023.

H. Reeve and Kabán. Classification with unknown class-conditional label noise on non-compact feature spaces. In A. Beygelzimer and D. Hsu, editors, *Proceedings of the Thirty-Second Conference on Learning Theory*, volume 99 of *Proceedings of Machine Learning Research*, pages 2624–2651, Phoenix, USA, 25–28 Jun 2019. PMLR.

C. Saharia, W. Chan, S. Saxena, L. Li, J. Whang, E. Denton, S. K. S. Ghasemipour, B. K. Ayan, S. S. Mahdavi, R. G. Lopes, T. Salimans, J. Ho, D. J. Fleet, and M. Norouzi. Photorealistic text-to-image diffusion models with deep language understanding, 2022.

N.-M. A. R. S. G. G. Stamper, J. and K. Koedinger. Algebra i 2008-2009. challenge data set from kdd cup 2010 educational data mining challenge. find it at http://pslcdatashop.web.cmu.edu/KDDCup/downloads.jsp, 2010.

T. Steinke and J. Ullman. The pitfalls of average-case differential privacy, 2020. https://differentialprivacy.org/average-case-dp/.

A. Triastcyn and B. Faltings. Bayesian differential privacy for machine learning. In *International Conference on Machine Learning*, pages 9583–9592. PMLR, 2020.

D. Wang and J. Xu. On sparse linear regression in the local differential privacy model. In K. Chaudhuri and R. Salakhutdinov, editors, *Proceedings of the 36th International Conference on Machine Learning*, volume 97 of *Proceedings of Machine Learning Research*, pages 6628–6637. PMLR, 09–15 Jun 2019. URL https://proceedings.mlr.press/v97/wang19m.html.

R. Wu, J. P. Zhou, K. Q. Weinberger, and C. Guo. Does label differential privacy prevent label inference attacks? *arXiv preprint arXiv:2202.12968*, 2022.

A. Proof of Theorem 5.2

We first define some additional notation. Let $s_i : \mathcal{Y}^n \mapsto \mathcal{Y}$ be the function $s_i(\mathbf{y}) = y_i$. Let $\mathbf{y} = (y_i, \mathbf{y}_{-i}) \in \mathcal{Y}^n$ denote that $y_i \in \mathcal{Y}$ is the ith label in $\mathbf{y}$ and $\mathbf{y}_{-i} \in \mathcal{Y}^{n-1}$ are the remaining labels. Recall that $f_P : \mathcal{Y} \mapsto \mathcal{X}$ is the random function that, given input y, outputs x according to conditional distribution $P_{\mathcal{X}|y}$. Also recall that for any function $f(z)$ we let $f^n(\mathbf{z})$ be the function that applies f component-wise to $\mathbf{z}$, such that if $\mathbf{z}' = f^n(\mathbf{z})$ then $z'_i = f(z_i)$.

For any $i \in [n]$ let $\widehat{M}_i : \mathcal{Y}^n \mapsto \mathcal{Y}$ be the mechanism

$$\widehat{M}_i(\mathbf{y}) = s_i(A(M^n(f_P^n(\mathbf{y}), \mathbf{y}))).$$

In other words, given a vector of labels $\mathbf{y}$, mechanism $\widehat{M}_i$ first draws features for each label from the corresponding conditional distribution, applies mechanism M to each labeled example to generate a noisy dataset, and then passes the noisy dataset to the attack algorithm, which outputs a prediction for the ith label. Thus we have

$$\mathrm{E}_{(\mathbf{x},\mathbf{y})\sim P^n,(\mathbf{x}',\mathbf{y}')\sim M^n(\mathbf{x},\mathbf{y}),\hat{\mathbf{y}}\sim A(\mathbf{x}',\mathbf{y}')}\left[\frac{1}{n}\sum_{i=1}^{n}\ell(y_i,\hat{y}_i)\right] = \frac{1}{n}\sum_{i=1}^{n}\mathrm{E}_{\mathbf{y}\sim P_{\mathcal{Y}}^n,\hat{y}_i\sim\widehat{M}_i(\mathbf{y})}[\ell(y_i,\hat{y}_i)]. \tag{3}$$

We will lower bound each term in the sum on the right-hand side of Eq. (3), which will suffice to prove the theorem.

For any $i \in [n]$ and vector of labels $\mathbf{y}_{-i} \in \mathcal{Y}^{n-1}$ let $\widehat{M}_{i,\mathbf{y}_{-i}} : \mathcal{X} \times \mathcal{Y} \mapsto \mathcal{Y}^n$ be the mechanism

$$\widehat{M}_{i,\mathbf{y}_{-i}}(x', y') = s_i(A((x', y'), M^{n-1}(f_P^{n-1}(\mathbf{y}_{-i}), \mathbf{y}_{-i}))).$$

In other words, mechanism $\widehat{M}_{i,\mathbf{y}_{-i}}$ uses the 'hard-wired' labels $\mathbf{y}_{-i}$ to perform the same operations as mechanism $\widehat{M}_i$ to generate $n-1$ noisy labeled examples, but uses its input for the ith noisy labeled example. Both mechanisms then pass the noisy dataset to the attack algorithm, which outputs a prediction for the ith label. Thus for any $i \in [n]$ and vector of labels $\mathbf{y} \in \mathcal{Y}^n$ we have

$$\widehat{M}_i(\mathbf{y}) = \widehat{M}_i(y_i, \mathbf{y}_{-i}) = \widehat{M}_{i,\mathbf{y}_{-i}}(M(f_P(y_i), y_i)).$$

Therefore for all $\mathbf{y}_{-i} \in \mathcal{Y}^{n-1}$ and $y, y' \in \mathcal{Y}$ we have

$$\begin{aligned} D_\alpha(\widehat{M}_i(y, \mathbf{y}_{-i}) \| \widehat{M}_i(y', \mathbf{y}_{-i})) &= D_\alpha(\widehat{M}_{i,\mathbf{y}_{-i}}(M(f_P(y), y)) \| \widehat{M}_{i,\mathbf{y}_{-i}}(M(f_P(y'), y'))) \\ &\le D_\alpha(M(f_P(y), y) \| M(f_P(y'), y')) \\ &\le \varepsilon \end{aligned}$$

where the first inequality follows because Rényi differential privacy is preserved under post-processing (Mironov, 2017) and the second inequality uses Definition 4.3. Thus, by the reduction from Rényi differential privacy to (ε, δ)-differential privacy (Mironov, 2017), for all for all $\mathbf{y}_{-i} \in \mathcal{Y}^{n-1}$ and $y, y', \hat{y} \in \mathcal{Y}$ we have

$$\Pr[\widehat{M}_i(y, \mathbf{y}_{-i}) = \hat{y}] \ge \exp(-\varepsilon') \Pr[\widehat{M}_i(y', \mathbf{y}_{-i}) = \hat{y}] - \delta \tag{4}$$

where $\varepsilon' = \varepsilon + \frac{\log(1/\delta)}{\alpha - 1}$. Let $y^* = \arg\max_y P_{\mathcal{Y}}(y)$ be the most common label. We are now ready to prove a lower bound on each term on the right-hand side of Eq. (3).

$$\begin{aligned} &\mathrm{E}_{\mathbf{y}\sim P_{\mathcal{Y}}^n,\hat{y}_i\sim\widehat{M}_i(\mathbf{y})}[\ell(y_i,\hat{y}_i)] \\ &= \mathrm{E}_{\mathbf{y}_{-i}\sim P_{\mathcal{Y}}^{n-1}}\left[\mathrm{E}_{y_i\sim P_{\mathcal{Y}},\hat{y}_i\sim\widehat{M}_i(y_i,\mathbf{y}_{-i})}[\ell(y_i,\hat{y}_i)]\right] && (5) \\ &= \mathrm{E}_{\mathbf{y}_{-i}\sim P_{\mathcal{Y}}^{n-1}}\left[\sum_{y,\hat{y}} P_{\mathcal{Y}}(y)\Pr\left[\widehat{M}_i(y,\mathbf{y}_{-i}) = \hat{y}\right]\ell(y,\hat{y})\right] \\ &\ge \mathrm{E}_{\mathbf{y}_{-i}\sim P_{\mathcal{Y}}^{n-1}}\left[\sum_{y}\sum_{\hat{y}\neq y} P_{\mathcal{Y}}(y)\Pr\left[\widehat{M}_i(y,\mathbf{y}_{-i}) = \hat{y}\right]\right]\ell^* && (6) \\ &\ge \mathrm{E}_{\mathbf{y}_{-i}\sim P_{\mathcal{Y}}^{n-1}}\left[\sum_{y}\sum_{\hat{y}\neq y} P_{\mathcal{Y}}(y)\left(\exp(-\varepsilon')\Pr\left[\widehat{M}_i(y^*,\mathbf{y}_{-i}) = \hat{y}\right] - \delta\right)\right]\ell^* && (7) \end{aligned}$$

$$= \mathrm{E}_{\mathbf{y}_{-i}\sim P_{\mathcal{Y}}^{n-1}} \left[\exp(-\varepsilon') \sum_{y} \sum_{\hat{y}\neq y} P_{\mathcal{Y}}(y) \Pr\left[\widehat{M}_i(y^*, \mathbf{y}_{-i}) = \hat{y}\right] - (k-1)\delta \right] \ell^* \tag{8}$$

$$= \mathrm{E}_{\mathbf{y}_{-i}\sim P_{\mathcal{Y}}^{n-1}} \left[\exp(-\varepsilon') \sum_{\hat{y}} \Pr\left[\widehat{M}_i(y^*, \mathbf{y}_{-i}) = \hat{y}\right] \sum_{y\neq\hat{y}} P_{\mathcal{Y}}(y) - (k-1)\delta \right] \ell^*$$

$$\geq \mathrm{E}_{\mathbf{y}_{-i}\sim P_{\mathcal{Y}}^{n-1}} \left[\exp(-\varepsilon') \sum_{y\neq y^*} P_{\mathcal{Y}}(y) - (k-1)\delta \right] \ell^* \tag{9}$$

$$= (\exp(-\varepsilon')(1-p^*) - (k-1)\delta)\ell^*$$

$$= \left(\exp(-\varepsilon)\delta^{\frac{1}{\alpha-1}}(1-p^*) - (k-1)\delta\right)\ell^*.$$

Eq. (5) follows because each label y_i is drawn independently from distribution $P_{\mathcal{Y}}$. Eq. (6) follows because the loss function ℓ is non-negative and $\ell^* = \min_{y\neq\hat{y}} \ell(y, \hat{y})$. Eq. (7) follows from Eq. (4). Eq. (8) follows because

$$\sum_{y} \sum_{\hat{y}\neq y} P_{\mathcal{Y}}(y) = \sum_{y} (k-1) P_{\mathcal{Y}}(y) = k-1.$$

Finally, to see why Eq. (9) holds, let $\alpha_{\hat{y}} = \Pr\left[\widehat{M}_i(y^*, \mathbf{y}_{-i}) = \hat{y}\right]$ and $\beta_{\hat{y}} = \sum_{y\neq\hat{y}} P_{\mathcal{Y}}(y)$. We have

$$\sum_{\hat{y}} \Pr\left[\widehat{M}_i(y^*, \mathbf{y}_{-i}) = \hat{y}\right] \sum_{y\neq\hat{y}} P_{\mathcal{Y}}(y) = \sum_{\hat{y}} \alpha_{\hat{y}}\beta_{\hat{y}} \geq \min_{\hat{y}} \beta_{\hat{y}} = \beta_{y^*} = \sum_{y\neq y^*} P_{\mathcal{Y}}(y)$$

where the inequality follows because $\alpha_{\hat{y}} \geq 0$ and $\sum_{\hat{y}} \alpha_{\hat{y}} = 1$.

B. Proof of Theorem 6.1

We assume for simplicity that the feature space $\mathcal{X}$ is finite and contains at least two elements, and also that there are at least two labels in $\mathcal{Y}$. Let $x_1, x_2 \in \mathcal{X}$ be distinct, and let $y_1, y_2 \in \mathcal{Y}$ be distinct. Let P be the distribution on $\mathcal{X} \times \mathcal{Y}$ that assigns probability $\frac{1}{2}$ to (x_1, y_1) and probability $\frac{1}{2}$ to (x_2, y_2). Let M be Mechanism 1. It suffices to show that

$$D_\alpha(M(f_P(y_1), y_1) \| M(f_P(y_2), y_2)) > \varepsilon$$

for any $\varepsilon < \infty$. We have

$$\begin{aligned} D_\alpha(M(f_P(y_1), y_1) \| M(f_P(y_2), y_2)) &= D_\alpha(M(x_1, y_1) \| M(x_2, y_2)) && \because \text{Definition of } P \\ &= \frac{1}{\alpha-1} \log \mathrm{E}_{(\tilde{x},\tilde{y})\sim M(x_2,y_2)} \left[\left(\frac{\Pr[M(x_1, y_1) = (\tilde{x}, \tilde{y})]}{\Pr[M(x_2, y_2) = (\tilde{x}, \tilde{y})]} \right)^\alpha \right] \\ &\geq \frac{1}{\alpha-1} \log \sum_{y} \left(\frac{\Pr[M(x_1, y_1) = (x_1, y)]^\alpha}{\Pr[M(x_2, y_2) = (x_1, y)]^{\alpha-1}} \right), \end{aligned} \tag{10}$$

where the second equality holds by Definition 3.1. Each denominator in Eq. (10) is zero (since M always outputs the same features that it receives as input) and at least one numerator in Eq. (10) must be positive (since M always outputs some label). Since $\alpha > 1$, this proves that Eq. (10) cannot be less than any finite quantity.

C. Proof of Theorem 6.2

Fix the values of $x, y, \widetilde{y}$. And let us consider the distribution of $M(f_P(y), y)$ which by definition is given by P_y.

$$P_y(x, \widetilde{y}) = P(X = x, \tilde{Y} = \tilde{y} | Y = y) = \frac{P(Y = y | X = x, \tilde{Y} = \tilde{y}) P(x = x, \tilde{Y} = \tilde{y})}{P(Y = y)}$$

Similarly we have

$$P_{1-y}(x,\widetilde{y}) = P(X=x,\tilde{Y}=\tilde{y}|Y=1-y) = \frac{P(Y=1-y|X=x,\tilde{Y}=\tilde{y})P(x=x,\tilde{Y}=\tilde{y})}{P(Y=1-y)}$$

Taking the ratio of these expressions, it is easy to see that:

$$\frac{P_y(x,\widetilde{y})}{P_{1-y}(x,\widetilde{y})} = \frac{P(Y=y|X=x,\tilde{Y}=\tilde{y})}{P(Y=1-y|X=x,\tilde{Y}=\tilde{y})}\frac{P(Y=1-y)}{P(Y=y)} =: e^{\nu(y,\widetilde{y},x)}$$

Finally, we have:

$$\begin{aligned}
(1-\alpha)D_\alpha(M(f_P(y),y)||M(f_P(1-y),1-y)) &= \log \mathrm{E}_{(X,\widetilde{Y})\sim P_{1-y}}\left[\left(\frac{P_y(X,\widetilde{Y})}{P_{1-y}(X,\widetilde{Y})}\right)^{\alpha}\right] \\
&= \log \mathrm{E}_{(X,\widetilde{Y})\sim P_{1-y}}\left[\left(\frac{P_y(X,\widetilde{Y})}{P_{1-y}(X,\widetilde{Y})}\right)^{\alpha-1}\frac{P_y(X,\widetilde{Y})}{P_{1-y}(X,\widetilde{Y})}\right] \\
&= \log \mathrm{E}_{(X,\widetilde{Y})\sim P_{y}}\left[\left(\frac{P_y(X,\widetilde{Y})}{P_{1-y}(X,\widetilde{Y})}\right)^{\alpha-1}\right] \\
&= \log \mathrm{E}_{(X,\widetilde{Y})\sim P_{y}}\left[e^{(\alpha-1)\nu(y,\widetilde{Y},X)}\right]
\end{aligned}$$

D. Proof of Theorem 6.6

Lemma D.1. *The average instance based privacy loss of the randomized response mechanism with probability π is given by*

$$\nu(x) = (2\eta(x)-1)\left[\log\tfrac{\eta(x)}{1-\eta(x)} - \log\tfrac{p_+}{1-p_+}\right] + (2\pi-1)\log\tfrac{\pi}{1-\pi}$$

Proof. Straightforward computation yields

$$\begin{aligned}
\mathbb{E}\left[\nu(Y,\widetilde{Y},x)|X=x\right] &= [(1-\eta(x))(1-\pi)]\log\frac{(1-\eta(x))(1-\pi)(1-p+)}{\eta(x)\pi p_+} \\
&+ [(1-\eta(x))\pi]\log\frac{(1-\eta(x))\pi(1-p_+)}{\eta(x)(1-\pi)p_+} \\
&+ \eta(x)\pi\log\frac{\eta(x)\pi p_+}{(1-\eta(x))(1-\pi)(1-p_+)} \\
&+ \eta(x)(1-\pi)\log\frac{\eta(x)(1-\pi))p_+}{(1-\eta(x))\pi(1-p_+)} \\
&= (2\eta(x)-1)\log\frac{\eta(x)}{1-\eta(x)} + (2\pi-1)\log\frac{\pi}{1-\pi} - (2\eta(x)-1)\log\frac{p_+}{1-p_+}
\end{aligned}$$

□

In order to build the estimator for $J(\tau)$ we will require the definition of 2 types of level sets. Let $H\colon [0,1]\to\mathbb{R}$ be given by

$$H(z) = (2z-1)\log\frac{z}{1-z} + (2\pi-1)\log\frac{\pi}{1-\pi} - (2z-1)\log\frac{p_+}{1-p_+}$$

Note that $H(z)$ is symmetric around $1/2$. Therefore its inverse admits two branches. Let $H_0^{-1}:\mathbb{R}\to[0,1/2]$ be its first branch and $H_1^{-1}:\mathbb{R}\to[1/2,1]$ be its second branch. For a fixed τ let $G_\tau = \{x\colon \eta(x)\le H_0^{-1}(\tau)\}$. $G^\tau = \{x\colon \eta(x) > H_1^{-1}(\tau)\}$

Lemma D.2. *The following holds for all τ*

$$J(\tau) = P(\nu(X) > \tau) = P(G_\tau) + P(G^\tau)$$

Proof. The proof is straightforward since $\nu(X) = H(\eta(X))$. Since H is decreasing in $[0, 1/2]$ and increasing in $[1/2, 1]$ it follows that $H(\eta(X)) \geq \tau$ if and only if $\eta(X) < H_0^{-1}(\tau)$ or $\eta(X) > H^{-1}(\tau)$. □

Thus we are left with estimating the probability of these sets. We will show how this can be done using nearest neighbor estimators.

D.1. Nearest neighbor estimators

We will work in the non-parametric regime using k-nearest neighbor estimator which is a *plug-in estimator*, i.e. it estimates $\eta(x)$ by using the conditional empirical distribution using the neighbors of x. We will denote the estimate of η by $\widehat{\eta}(x)$. The motivation for using plug-in estimates is that, under mild assumptions, one can show that the L_1 error of the estimator vanishes under mild smoothness assumption on the conditional distribution of labels, therefore we do not have to deal with approximation error.

Let $\mathcal{S} = \{(x_1, y_1), \ldots, (x_n, y_n)\}$. Given $x \in \mathcal{X}$ we let $\{\tau_{n,q}(x)\}_{q\in[n]}$ be an enumeration of $[n]$ such that for each $q \in [n-1]$, $\rho\left(x, x_{\tau_{n,q}(x)}\right) \leq \rho\left(x, x_{\tau_{n,q+1}(x)}\right)$. In words, given $x \in \mathcal{X}$, $\{\tau_{n,i}(x)\}_{i\in[n]}$ sorts $\{x_1, \ldots, x_n\}$ in increasing order of ρ-distance to x.

The k-nearest neighbor regression estimator $\hat{\eta}_{n,k} : \mathcal{X} \to [0, 1]$ is given by

$$\hat{\eta}_{n,k}(x) := \frac{1}{k} \cdot \sum_{i\in[k]} y_{\tau_{n,i}(x)}. \tag{11}$$

We recall the following lemma which upper bounds the probability measure of the ball around a point $x \in \mathcal{X}$ that contains its k nearest neighbors. The proof immediately follows from the multiplicative Chernoff bound (see, e.g., Lemma 3.2 in (Reeve and Kabán, 2019)).

Lemma D.3. *Given $x \in \mathcal{X}, k \in [n]$ with $k \geq 4\log(1/\delta)$, with probability at least $1-\delta$ over $\mathcal{S} = \{x_i\}_{i\in[n]}$ we have $\mu\left(B_{\rho\left(x, x_{\tau_{n,k}(x)}\right)}(x)\right) \leq \frac{2k}{n}$.*

When we combine Assumption 6.5 with Lemma D.3, we get the following corollary.

Corollary D.4. *Suppose that the measure-smoothness assumption (Assumption 6.5) holds with parameters β, C_β. Then for all $x \in \mathcal{X}, k \in [n]$ with $k \geq 4\log(1/\delta)$, with probability at least $1-\delta$ over $\mathcal{S}$, the following holds for all $j \in [k]$: $\left|\eta(x_{\tau_{n,j}(x)}) - \eta(x)\right| \leq C_\beta \cdot (2k/n)^\beta$.*

Lemma D.5. *Suppose that the measure-smoothness assumption (Assumption 6.5) holds with parameters β, C_β. Let $\mathcal{S} = ((x_i, y_i))_{i=1}^n$ be an i.i.d. sample. Let $\delta > 0$ and let $k > 4\log(2/\delta)$. Then with probability at least $1-\delta$ over the sample $\mathcal{S}$ the following bound holds for all i:*

$$|\widehat{\eta}_{n,k}(x_i) - \eta(x_i)| \leq \sqrt{\frac{\log(8n/\delta)}{2k}} + C_\beta \left(\frac{2k}{n}\right)^\lambda. \tag{12}$$

Proof. Fix a value $x \in \mathcal{X}$. Note that, conditioned on x_i the nearest neighbor rule is a sum of binomial random variables with mean $\sum_{j=1}^k \eta(x_{\tau_{n,i}(x)})$. Therefore by Hoeffding's inequality we have with probability at least $1 - \frac{\delta}{2n}$

$$\left|\widehat{\eta}_{n,k}(x) - \frac{1}{k}\sum \eta(x_{\tau_{n,j}(x)})\right| \leq \sqrt{\frac{\log(8n/\delta)}{2k}}$$

In particular, by the union bound we have that for all i, with probability at least $1 - \delta/2$

$$\left|\widehat{\eta}_{n,k}(x_i) - \frac{1}{k}\sum \eta(x_{\tau_{n,j}(x_i)})\right| \leq \sqrt{\frac{\log(8n/\delta)}{2k}}$$

On the other hand, by Corollary D.4 we know that for our choice of k. With probability at least $1-\delta/2$ over the sample $\mathcal{S}$ the following holds:

$$\left|\frac{1}{k}\sum \eta(x_{\tau_{n,j}(x_i)}) - \eta(x_i)\right| \leq C_\beta \left(\frac{2k}{n}\right)^\lambda$$

Combining these two bounds and using the union bound yields the desired result. □

Lemma D.6. *Let η satisfy the smoothness assumption with parameters β, C_β. Let $\delta > 0$, $k > 4\log(4/\delta)$ and $C := C(n,k,\delta) = \sqrt{\frac{\log(16n/\delta)}{2k}} + C_\beta \left(\frac{2k}{n}\right)^\lambda$. Let $z > 0$ $A = \{x\colon \eta(x) < z\}$. Then with probability at least $1-\delta$ the following bound holds:*

$$P(A) \geq \frac{1}{n}\sum_{i=1}^{n} \mathbb{1}_{\widehat{\eta}(x_i)+C<z} - \sqrt{\frac{\log(2/\delta)}{2n}} - \frac{\delta}{2}$$

Proof. Let U denote the event $|\widehat{\eta}(x_i) - \eta(x_i)| < C(n,k,\delta)$ for all i, by Lemma D.5 we know that $P(U) > 1-\delta/2$. We also know that

$$\begin{aligned}
\frac{1}{n}\sum_{i=1}^{n} \mathbb{1}_{\widehat{\eta}(x_i)+C<z} &= \frac{1}{n}\sum_{i=1}^{n} \mathbb{1}_{\widehat{\eta}(x_i)+C<z}\mathbb{1}_U + \frac{1}{n}\sum_{i=1}^{n} \mathbb{1}_{\widehat{\eta}(x_i)+C<z}\mathbb{1}_{U^c} \\
&\leq \frac{1}{n}\sum_{i=1}^{n} \mathbb{1}_{\eta(x_i)<z}\mathbb{1}_U + \frac{1}{n}\sum_{i=1}^{n} \mathbb{1}_{U^c} \\
&\leq \frac{1}{n}\sum_{i=1}^{n} \mathbb{1}_{\eta(x_i)<z} + \mathbb{1}_{U^c}
\end{aligned}$$

Taking expectations over both sides of the inequality yield:

$$\frac{1}{n}\sum_{i=1}^{n} \mathbb{1}_{\widehat{\eta}(x_i)+C<z} \leq P(A) + P(U^c) \leq P(A) + \delta/2$$

. □

By Hoeffding's inequality we thus have that with probability at least $1-\delta/2$:

$$\frac{1}{n}\sum_{i=1}^{n} \mathbb{1}_{\widehat{\eta}(x_i)+C<z} \leq P(A) + \frac{\delta}{2} + \sqrt{\frac{\log(2/\delta)}{2n}} \tag{13}$$

By the union bound it follows that G and (13) will both happen with probability at least $1-\delta$.

We are now in a position to prove the statement of Proposition 6.6.

Proof. Let $k > 4\log(8/\delta)$ and let $C(n,k,\delta) = \sqrt{\frac{\log(16n/\delta)}{2k}} + C_\beta \left(\frac{2k}{n}\right)^\lambda$. Fix $\tau > 0$ and define

$$\widehat{J}_n(\tau) = \sum_{i=1}^{n} \mathbb{1}_{\widehat{\eta}_{n,k}(x_i)+C\leq H_0^{-1}(\tau)} + \mathbb{1}_{\widehat{\eta}_{n,k}(x_i)-C\geq H_0^{-1}(\tau)} \tag{14}$$

By Lemma D.6 and using the union bound we have that with probability at least $1-\delta$:

$$J(\tau) \geq \widehat{J}_n(\tau) - 2\sqrt{\frac{\log(2/\delta)}{2n}} - \delta.$$

□

E. Proof of Theorem 7.1

Here we prove the privacy properties that Mechanism 2 satisfies. For the rest of the section, we will denote by X a random variable representing the features and Y a random variable representing the label. The measure for this random variables may vary across the proofs. Note also that mechanism M defined by Mechanism 2 can actually be expressed as two independent mechanisms: $M(x, y) = (M_{\mathcal{X}}(x, y), M_{\mathcal{Y}}(x, y))$. We will analyze each mechanism separately.

Lemma E.1. *For all $\alpha > 1$, $M_{\mathcal{Y}}$ satisfies*

$$D_\alpha(M_{\mathcal{Y}}(x, y)\|M_{\mathcal{Y}}(x, y')) \leq \log\left(1 + \frac{k(1-\lambda)}{\lambda}\right)$$

Proof. Fix label y and let P_y denote the distribution over the output of the mechanism when the input label to the mechanism is y.

Let $\tilde{Y}$ denote the random variable representing the output of the mechanism $M_{\mathcal{Y}}$. By definition of the mechanism we have:

$$P_y(\tilde{y}) = P(\tilde{Y} = \tilde{y}|Y = y) = (1-\lambda)\mathbb{1}\tilde{y} = y + \frac{\lambda}{k}.$$

Thus the ratio of densities is bounded as

$$\frac{P_y(\tilde{y})}{P'_y(\tilde{y})} = \frac{(1-\lambda)\mathbb{1}_{\tilde{y}=y} + \frac{\lambda}{k}}{(1-\lambda)\mathbb{1}_{\tilde{y}=y'} + \frac{\lambda}{k}}$$

Note that this ratio is clearly maximizes when $y \neq y'$ and $\tilde{y} = y$.In which case the ratio is given by

$$\frac{P_y(\tilde{y})}{P'_y(\tilde{y})} = \frac{(1-\lambda) + \frac{\lambda}{k}}{\frac{\lambda}{k}} = 1 + k\frac{1-\lambda}{\lambda}.$$

We thus have that

$$D_\alpha(M(x, y)\|M(x, y')) \leq \frac{1}{\alpha - 1}\,\mathrm{E}_{P_y}\log\left[\left(1 + k\frac{1-\lambda}{\lambda}\right)^{\alpha-1}\right].$$

Simplifying this expression shows the result of the lemma. □

Corollary E.2. *For all $\alpha > 1$, Mechanism 2 satisfies (α, ϵ)-label differential privacy if*

$$\epsilon \geq \log\left(1 + \frac{k(1-\lambda)}{\lambda}\right)$$

Proof. The proof follows immediately from the above lemma and the fact that mechanism $M_{\mathcal{X}}$ is applied on the same feature vector. Hence there distribution over both examples is the same. □

We now proceed to show that the mechanism also satisfies (α, ϵ, P)-label differential privacy. We first introduce a very simple lemma.

Lemma E.3. *Let $a, b, k > 0$ and $\lambda \in (0, 1)$. Then*

$$\frac{(1-\lambda) + \lambda/k}{\lambda/k} \leq \frac{(1-\lambda)a + \frac{\lambda}{k}b}{\frac{\lambda}{k}b} \tag{15}$$

if and only if $b \leq a$.

Proof. Simplifying terms and multiplying both sides of (15) we see that the inequality is true if and only if:

$$(1-\lambda)b + \frac{\lambda}{k}b \leq (1-\lambda)a + \frac{\lambda}{k}b.$$

The result follows since all terms in the above inequality are positive. □

Lemma E.4. *Mechanism $M_{\mathcal{X}}$ satisfies the following*

$$D_\alpha(M_{\mathcal{X}}(f_P(y),y)\|M_{\mathcal{X}}(f_P(y),y) \leq \log\left(1+\frac{k(1-\lambda)}{\lambda}\right) + D_\alpha(P_{\mathcal{X}|y}\|\widehat{P}_{\mathcal{X}|y})$$

Proof. For a fixed y let $P_y(\cdot)$ denote the probability distribution associated with the mechanism $M_{\mathcal{X}}(f_P(y),y)$. For a fixed value of x we want to bound the ratio:

$$\frac{P_y(x)}{P'_y(x)}$$

Let $A = \{x\colon \frac{P_y(x)}{P'_y(x)} \leq \frac{\lambda/k+1-\lambda}{\lambda/k}\}$. Then we can decompose the above ratio as

$$\frac{P_y(x)}{P'_y(x)} = \frac{P_y(x)}{P'_y(x)}\mathbb{1}_{x\in A} + \frac{P_y(x)}{P'_y(x)}\mathbb{1}_{x\notin A} \tag{16}$$

By definition of A the first term satisfies:

$$\frac{P_y(x)}{P'_y(x)}\mathbb{1}_{x\in A} \leq \frac{\lambda/k+1-\lambda}{\lambda/k} = \left(1+\frac{k(1-\lambda)}{\lambda}\right)\mathbb{1}_{x\in A}$$

Let us understand the scenario where $x \notin A$. By definition of the mechanism we know that

$$P_y(X=x) = (1-\lambda)P_{\mathcal{X}|y}(x) + \frac{\lambda}{k}\sum_{j=1}^{k}\widehat{P}_{\mathcal{X}|j}(x))$$

Applying the same argument to y' we obtain:

$$\begin{aligned}\frac{P_y(X=x)}{P_{y'}(X=x)} &= \frac{(1-\lambda)P_{\mathcal{X}|y}(x) + \frac{\lambda}{k}\sum_{j=1}^{k}\widehat{P}_{\mathcal{X}|j}(x)}{(1-\lambda)P_{\mathcal{X}|y'}(x) + \frac{\lambda}{k}\sum_{j=1}^{k}\widehat{P}_{\mathcal{X}|j}(x)} \\ &\leq \frac{(1-\lambda)P_{\mathcal{X}|y}(x) + \frac{\lambda}{k}\sum_{j=1}^{k}\widehat{P}_{\mathcal{X}|j}(x)}{\frac{\lambda}{k}\sum_{j=1}^{k}\widehat{P}_{\mathcal{X}|j}(x)}\end{aligned}$$

Moreover for $x \notin A$ we must have, in view of Lemma E.3, that

$$P_{\mathcal{X}|y}(x) \geq \sum_{j=1}^{k}\widehat{P}_{\mathcal{X}|j}(x)$$

Using this inequality and the fact that $\sum_{j=1}^{k}\widehat{P}_{\mathcal{X}|j}(x) \geq \widehat{P}_{\mathcal{X}|y}(x)$ (for the denominator) we obtain

$$\frac{P_y(x)}{P'_y(x)}\mathbb{1}_{x\notin A} \leq \frac{(1-\lambda+\frac{\lambda}{k})P_{\mathcal{X}|y}(x)}{\frac{\lambda}{k}\widehat{P}_{\mathcal{X}|y)(x)}}\mathbb{1}_{x\notin A} = \left(1+\frac{k(1-\lambda)}{\lambda}\right)\frac{P_{\mathcal{X}|y}(x)}{\widehat{P}_{\mathcal{X}|y}(x)}\mathbb{1}_{x\notin A}$$

We can replace this expression in (16) to obtain:

$$\frac{P_y(x)}{P'_y(x)} \leq \left(1+\frac{k(1-\lambda)}{\lambda}\right)\left(\mathbb{1}_{x\in A} + \frac{P_{\mathcal{X}|y}(x)}{\widehat{P}_{\mathcal{X}|y}(x)}\mathbb{1}_{x\notin A}\right)$$

We can now calculate the divergence between the output distributions of the mechanism.

$$(\alpha-1)D_\alpha(M_{\mathcal{X}}(f_P(y),y)\|M_{\mathcal{X}}(f_P(y'),y')) = \log\mathbb{E}_{x\sim P_y}\left[\left(\frac{P_y(x)}{P'_y(x)}\right)^{\alpha-1}\right]$$

$$= \log\mathbb{E}_{x\sim P_y}\left[\left(1+\frac{k(1-\lambda)}{\lambda}\right)^{\alpha-1}\left(\mathbb{1}_{x\in A} + \frac{P_{\mathcal{X}|y}(x)}{\widehat{P}_{\mathcal{X}|y}(x)}\mathbb{1}_{x\notin A}\right)^{\alpha-1}\right]$$

$$= (\alpha-1)\log\left(1+\frac{k(1-\lambda)}{\lambda}\right) + (\alpha-1)\log(P_y(A)) + \log \mathbb{E}_{x\sim P_y}\left[\left(\frac{P_{\mathcal{X}|y}(x)}{\widehat{P}_{\mathcal{X}|y}(x)}\mathbb{1}_{x\notin A}\right)^{\alpha-1}\right]$$

$$\leq (\alpha-1)\log\left(1+\frac{k(1-\lambda)}{\lambda}\right) + \log \mathbb{E}_{x\sim P_y}\left[\left(\frac{P_{\mathcal{X}|y}(x))}{\widehat{P}_{\mathcal{X}|y}(x)}\mathbb{1}_{x\notin A}\right)^{\alpha-1}\right], \tag{17}$$

where we have used the fact that $\log(P_y(A)) \leq 0$. Let us handle the expectation term in the above expression. Note that for $x \notin A$, again by Lemma E.3, we must have that $P_{\mathcal{X}|y}(x) \geq \sum_{j=1}^{k} \widehat{P}_{\mathcal{X}|j}(x)$. Therefore

$$\begin{aligned} P_{\mathcal{X}|y}(x) &\geq (1-\lambda+\frac{\lambda}{k})P_{\mathcal{X}|y}(x) \\ &\geq (1-\lambda)P_{\mathcal{X}|y}(x) + \frac{\lambda}{k}\sum_{j=1}^{k}\widehat{P}_{\mathcal{X}|j}(x) = P_y(x). \end{aligned}$$

In particular for all $x \notin A$ we have that

$$\frac{P_{\mathcal{X}|y}(x)}{P_y(x)} \geq 1$$

Thus we have

$$\begin{aligned} \log \mathbb{E}_{x\sim P_y}\left[\left(\frac{P_{\mathcal{X}|y}(x)}{\widehat{P}_{\mathcal{X}|y}(x)}\mathbb{1}_{x\notin A}\right)^{\alpha-1}\right] &\leq \log \mathbb{E}_{x\sim P_y}\left[\left(\frac{P_{\mathcal{X}|y}(x)}{\widehat{P}_{\mathcal{X}|y}(x)}\mathbb{1}_{x\notin A}\right)^{\alpha-1}\frac{P_{\mathcal{X}|y}(x)}{P_y(x)}\right] \\ &\leq \log E_{x\sim P_{\mathcal{X}|y}}\left[\left(\frac{P_{\mathcal{X}|y}(x)}{\widehat{P}_{\mathcal{X}|y}(x)}\right)^{\alpha-1}\right] \\ &= D_\alpha(P_{\mathcal{X}|y}\|\widehat{P}_{\mathcal{X}|y}) \end{aligned}$$

The proof is finalized by replacing this bound in (17). □

We are now in a position to prove the main theorem for Mechanism 2.

Proof of Theorem 7.1. By Corollary E.2 we know that the mechanism satisifies (α, ϵ)-label differential privacy. To show that it also satisfies (α, ϵ, P)-label differential privacy, notice that the random variable generated by $M_{\mathcal{X}}$ is independent of $M_{\mathcal{Y}}$ conditioned on the true label y. Therefore, by the additive properties of the Rényi divergence for product distributions we have:

$$D_\alpha(M(f_P(y),y)\|M(f_p(y'),y')) = D_\alpha(M_{\mathcal{X}}(f_P(y),y)\|M(f_p(y'),y')) + D_\alpha(M_{\mathcal{Y}}(f_P(y),y)\|M(f_p(y'),y')).$$

The result now follows from Lemmas E.1 and E.4 as well as the assumption that $D_\alpha(P_{\mathcal{X}|y}\|\widehat{P}_{\mathcal{X}|y}) \leq \Delta$. □

F. Proof of Theorem 7.2

Throughout this section, let M be Mechanism 3. Recall that $f_P : \mathcal{Y} \mapsto \mathcal{X}$ is the random function that, given label $y \in \mathcal{Y}$, outputs features $x \in \mathcal{X}$ according to distribution $P_{\mathcal{X}|y}$.

Proof of Theorem 7.2. We shall prove the contrapositive: If

$$\varepsilon < \frac{\alpha}{\alpha-1}\log\left(1+\frac{(1-\lambda)k}{\lambda}\right) - \frac{1}{\alpha-1}\log\frac{k}{\lambda}$$

then M does not satisfy (α, ε, P)-label Rényi differential privacy. By Definition 4.3, it suffices show that there exist labels $y, y' \in \mathcal{Y}$ such that $D_\alpha(M(f_P(y),y)\|M(f_P(y'),y')) > \varepsilon$. Choose any $y, y' \in [k]$ such that $y \neq y'$. We have

$$\mathrm{E}_{(\tilde{x},\tilde{y})\sim M(y')}\left[\left(\frac{\Pr[M(f_P(y),y)=(\tilde{x},\tilde{y})]}{\Pr[M(f_P(y'),y')=(\tilde{x},\tilde{y})]}\right)^{\alpha}\right]$$

$$
\begin{aligned}
&\geq \Pr_{(\tilde{x},\tilde{y})\sim M(f_P(y'),y')}[\tilde{y}=y]\cdot \mathrm{E}_{(\tilde{x},\tilde{y})\sim M(f_P(y'),y')}\left[\left(\frac{\Pr[M(f_P(y),y)=(\tilde{x},\tilde{y})]}{\Pr[M(f_P(y'),y')=(\tilde{x},\tilde{y})]}\right)^{\alpha}\;\middle|\;\tilde{y}=y\right]\\
&=\frac{\lambda}{k}\cdot \mathrm{E}_{(\tilde{x},\tilde{y})\sim M(f_P(y'),y')}\left[\left(\frac{\Pr[M(f_P(y),y)=(\tilde{x},\tilde{y})]}{\Pr[M(f_P(y'),y')=(\tilde{x},\tilde{y})]}\right)^{\alpha}\;\middle|\;\tilde{y}=y\right]\\
&=\frac{\lambda}{k}\cdot\left(\frac{\Pr[M(f_P(y),y)=(\perp,y)]}{\Pr[M(f_P(y'),y')=(\perp,y)]}\right)^{\alpha}\\
&=\frac{\lambda}{k}\left(\frac{(1-\lambda)+\frac{\lambda}{k}}{\frac{\lambda}{k}}\right)^{\alpha}\\
&=\frac{\lambda}{k}\left(1+\frac{(1-\lambda)k}{\lambda}\right)^{\alpha}. \qquad (18)
\end{aligned}
$$

Therefore

$$
\begin{aligned}
D_\alpha(M(f_P(y),y)\|M(f_P(y'),y')) &= \frac{1}{\alpha-1}\log \mathrm{E}_{(\tilde{x},\tilde{y})\sim M(y')}\left[\left(\frac{\Pr[M(f_P(y),y)=(\tilde{x},\tilde{y})]}{\Pr[M(f_P(y'),y')=(\tilde{x},\tilde{y})]}\right)^{\alpha}\right] && \because \text{Definition 3.1}\\
&\geq \frac{1}{\alpha-1}\log\left(\frac{\lambda}{k}\left(1+\frac{(1-\lambda)k}{\lambda}\right)^{\alpha}\right) && \because \text{Eq. (18)}\\
&= \frac{\alpha}{\alpha-1}\log\left(1+\frac{(1-\lambda)k}{\lambda}\right)-\frac{1}{\alpha-1}\log\frac{k}{\lambda}\\
&> \varepsilon && \text{By assumption}
\end{aligned}
$$

□

G. Unbiasing the loss

We discuss how a learner may unbias the loss of a model being trained using the data released by our mechanism. Given a prediction space $\widehat{\mathcal{Y}}$ and loss function $\ell\colon \widehat{\mathcal{Y}}\times\mathcal{Y}\to\mathbb{R}$ we are interested with measuring

$$\mathrm{E}_P[\ell(h(x),y)]$$

However, if $\tilde{X},\tilde{Y}$ are the outputs of Mechanism 2. We may only observe

$$\mathrm{E}[\ell(h(\tilde{X}),\tilde{Y}))]$$

The following lemma provides us with a relation between both terms.

Lemma G.1. *Under the above notation, the following equality holds.*

$$\mathrm{E}_P[\ell(h(x),y)] = \frac{\mathrm{E}[\ell(h(\tilde{X}),\tilde{Y})]-\frac{p}{k}\sum_{j=1}^{k}\mathrm{E}_{\tilde{X}}[\ell(h(\tilde{X},j)]}{(1-p)^2}+\frac{\frac{p}{k}}{1-p}\sum_{j=1}^{k}\mathrm{E}_{x,y\sim\widehat{P}_{\mathcal{X}|i}\times P_{\mathcal{Y}}}[\ell(h(x),y)]$$

Proof. We begin by taking the expectation over the randomness of our mechanism. Fix y and for simplicity let be the distribution of the mechanism for input y.

$$
\begin{aligned}
\mathrm{E}_M[\ell(h(\tilde{X},\tilde{Y})] &= (1-p)E_{M_{\mathcal{X}}}[\ell(h(\tilde{X},y)]+\frac{p}{k}\sum_{j=1}^{k}\mathrm{E}_{M_{\mathcal{X}}}[l(h(\tilde{X},j)]\\
&= (1-p)^2\,\mathrm{E}_{x\sim P_{\mathcal{X}|y}}[\ell(h(x,y)]+\frac{(1-p)p}{k}\sum_{j=1}^{k}\mathrm{E}_{x\sim P_{\mathcal{X}|j}}[\ell(h(x,y)]+\frac{p}{k}\sum_{j=1}^{k}E_{M_{\mathcal{X}}}[l(h(\tilde{X},j)]
\end{aligned}
$$

Taking expectation over the label y we have

$$\mathrm{E}[\ell(h(\tilde{X},\tilde{Y})] = (1-p)^2\,\mathrm{E}_P[\ell(h(x,y)] + \frac{(1-p)p}{k}\sum_{j=1}^{k}\mathrm{E}_{x,y\sim P_{\mathcal{X}|j}\times P_{\mathcal{Y}}}[\ell(h(x,y)] + \frac{p}{k}\sum_{j=1}^{k}\mathrm{E}[l(h(\tilde{X},j)]$$

Solving for $\mathrm{E}_P[\ell(h(x),y)]$ yields the result. □

Note that the above expression for the expected loss on the true distribution can be expressed as two terms. The first one is something we can simulate from the output of Mechanism 2. The second term however, depends on synthetic data distribution $\widehat{P}$ which the learner may not have access to. In our experiments, we approximate the expectation of the loss using only the first term.