BLOCKWISE RANDOMIZED RESPONSE: THEORY AND ALGORITHMS FOR LABEL DIFFERENTIAL PRIVACY

Anonymous authors
Paper under double-blind review

ABSTRACT

In this paper, we introduce a novel block structure and apply it to randomized response-type algorithms, yielding the *BlockRR* framework for label differential privacy. Theoretically, we formulate an optimization problem whose optimal solution corresponds to the BlockRR mechanism. We illustrate that the introduction of $\mathcal{T} \subseteq \tilde{\mathcal{S}}_1$ is designed to enhance the influence of majority labels and mitigate the impact of minority labels and demonstrate the BlockRR mechanism serves as a general framework. In addition, we establish its ϵ-label DP guarantee and conduct a utility analysis by deriving bounds on the excess risk. Numerical results demonstrate that our method gets a better balance between test accuracy and the average of per-class accuracy in the high- and moderate-privacy regimes. In the low-privacy regime, our method reduces to standard RR.

1 INTRODUCTION

In many real-world scenarios, such as ad-click prediction or survey analysis (Dwork et al., 2006a;b), the features (e.g., user interactions) are not sensitive, but the labels (e.g., clicks or private responses) are (Abadi et al., 2016; Chaudhuri et al., 2011). Label Differential Privacy (Label DP) (Chaudhuri & Hsu, 2011) addresses this by applying privacy guarantees solely to the labels, unlike standard Differential Privacy (DP) (Dwork, 2006), which protects both features and labels. This narrower scope allows for better model utility under the same privacy budget (Ghazi et al., 2021; Esfandiari et al., 2022b; Ghazi et al., 2022).

A widely adopted technique in Label DP is the randomized response (RR) mechanism (Warner, 1965), originally developed for social surveys to reduce response bias. Here, RR acts as a label perturbation mechanism: each true label is transformed into a privatized output via a randomized process that satisfies the DP guarantee, and models are then trained on these perturbed labels.

To improve utility, numerous RR variants have been proposed for different learning tasks, including RRWithPrior (Ghazi et al., 2021) for classification, and RRonBins (Ghazi et al., 2022) and RPWithPrior (Liu & Huang, 2026) for regression. Those RR-type algorithms are reviewed in Appendix A.

Class imbalance continues to pose a significant practical obstacle, as classifiers tend to favor majority classes while underperforming on minority ones. This necessitates separate treatment of the two class groups. Furthermore, overall accuracy alone is an insufficient performance metric; the trade-off between overall and per-class accuracy must be explicitly accounted for. Compounding the issue is the absence of comprehensive benchmarking frameworks, which hinders practitioners from selecting the most suitable algorithm for a given application without extensive empirical evaluation.

We address this gap by proposing BlockRR, a general framework for Label DP that generalizes RR-type mechanisms. The core idea is to partition the label spaces (original and privatized) into blocks, applying tailored transformations between specific block pairs. This design subsumes RR and RRWithPrior as special cases defined by specific hyper-parameter settings (i.e., particular block structures). Thus, BlockRR provides a common design space where algorithm comparison reduces to an analysis of hyper-parameter choices, enabling systematic benchmarking and selection. In summary, our contributions can be outlined as follows:

(1) **We introduce a new block structure.** Specifically, we partition the original label space $\mathcal{S}$ into two disjoint subsets: a majority set $\mathcal{S}_1$ and a minority set $\mathcal{S}_2$. Similarly, we then partition $\tilde{\mathcal{S}}$ into

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

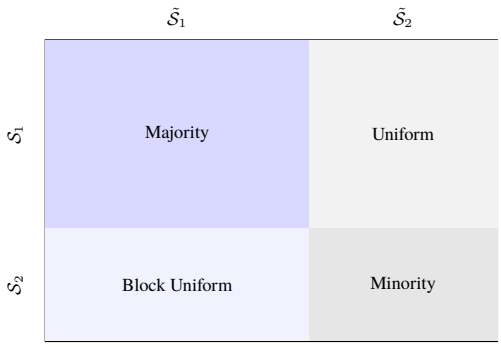

Figure 1: Quadrant view of the blocked label perturbation mechanism.

two privatized candidate sets $\tilde{\mathcal{S}}_1$ and $\tilde{\mathcal{S}}_2$. This creates four mapping regions (Figure 1): the diagonal regions ($\mathcal{S}_1 \to \tilde{\mathcal{S}}_1$ and $\mathcal{S}_2 \to \tilde{\mathcal{S}}_2$) follow standard RR, while the off-diagonal regions use a uniform or block-uniform distribution. The block-uniform design amplifies the influence of majority labels and mitigates that of minority labels.

(2) **Our algorithm is a general mechanism of RR-type algorithms under label DP.** Given the complexity of real-world data, an adaptive algorithm is often required. In this paper, we therefore design a general RR-type algorithm, which allows for easy control to enhancing majority labels and mitigate minority labels, using tunable parameters. Existing RR-type algorithms, including RR, RRWithPrior for classification tasks, correspond to specific configurations of BlockRR, thereby unifying them within our framework and eliminating the need for separate, case-by-case analysis.

(3) **Theoretical Results.** We provide a comprehensive theoretical analysis of the proposed BlockRR framework. Specifically, we formulate an optimization problem whose optimal solution corresponds to the BlockRR algorithm. We further establish the ϵ-label DP guarantee of the mechanism and conduct a utility analysis by deriving bounds on the excess risk.

(4) **We evaluate all three methods on various tasks to guide practitioners to identify an appropriate mechanism for a new application scenario.** we evaluate BlockRR on variants of CIFAR-10, CIFAR-100 and MNIST. The results demonstrate that in the high-privacy and moderate-privacy regimes (i.e., smaller ϵ), our method offers a superior trade-off between overall and per-class accuracy. Furthermore, our proposed method achieves the best performance in the majority of cases. In the low-privacy regime (larger ϵ), all methods converge as $\mathcal{S}_1 = \mathcal{S}$ and $\mathcal{S}_2 = \emptyset$, reducing BlockRR to standard RR.

The remainder of this paper is structured as follows: Section 2 reviews those related works. Sections 3 and 4 provide a general framework of RR-type algorithms under label DP and those theoretical results, respectively. Numerical implementations are discussed in Section 5. Finally, we conclude with our findings and highlight the limitations of this work in Section 6.

2 Related Works

Over the past decade, the field of differential privacy has seen significant theoretical and practical advances. A cornerstone technique is RR (Warner, 1965), introduced by Warner in 1965. Originally designed for anonymous surveys, its core idea is to allow participants to provide untruthful answers with a known probability when reporting sensitive data. This preserves individual privacy while still enabling accurate statistical analysis. With the formalization of the differential privacy framework, RR has become a crucial tool in DP research. Specifically, RR methods can be applied directly during data collection, preventing raw sensitive data from ever being exposed to a central server and thus providing a strong privacy guarantee (Erlingsson et al., 2014).

Additive noise mechanisms form another essential category of differential privacy methods. The Laplace Mechanism (Dwork et al., 2006b) achieves privacy by perturbing query outputs with Laplace-distributed noise. Similarly, the Gaussian Mechanism (Dwork & Roth, 2013) adds noise

from a Gaussian distribution and the Staircase Mechanism (Kairouz et al., 2016) adds noise distributed from a step probability density function.

Within the study of differentially private algorithms, DP-ERM (Chaudhuri et al., 2011) enables private training of regularized empirical risk minimization (ERM) through either output perturbation or objective perturbation. Song et al. (2013) proposed a stochastic gradient descent (SGD) with differentially private updates, which privatizes each parameter update by injecting calibrated noise directly into the computed gradients. For differentially private training of neural networks, Abadi et al. (2016) developed a differentially private SGD algorithm that enforces privacy by clipping each per-sample gradient and adding Gaussian noise to the aggregated batch averages. Beyond these foundational techniques, subsequent work has extended privacy mechanisms into deep learning architectures. For example, randomization layers have been successfully integrated into Convolutional Neural Networks (CNNs) to perturb internal feature representations directly (Mahawaga Arachchige et al., 2020).

In addition, Chaudhuri & Hsu (2011) introduced the concept of Label Differential Privacy (Label DP), which specifically perturbs and protects only the data labels while keeping the features unchanged. Research into deep learning under this framework has advanced significantly. For instance, Malek Esmaeili et al. (2021) proposed both the ALIBI method and the PATE method. The ALIBI method integrates additive Laplace noise with Bayesian denoising, and the PATE method combines the PATE framework with semi-supervised learning (Papernot et al., 2017). Ghazi et al. (2021) propose the RRTop-k method by filtering out minority class samples and upsampling majority classes. They introduced an RRWithPrior mechanism that leverages prior knowledge of the data distribution to mitigate the utility loss. Esfandiari et al. (2022a) introduced novel mechanisms for label differential privacy based on clustering. Their method groups training examples according to their non-private feature vectors, then randomly resamples labels from examples within the same cluster, yielding a noisy-label training set along with a modified loss function.

Label DP has attracted considerable attention in the regression tasks. The RRonBins mechanism (Ghazi et al., 2022) discretizes the output space into a finite set of bins through dynamic programming and applies the randomized response (RR) algorithm to these bins. Another approach is the optimal unbiased randomizers for regression under label differential privacy (Badanidiyuru et al., 2023), which imposes an unbiasedness constraint to enhance utility. More recently, Liu et al. (2026) introduced the RPWithPrior mechanism, a continuous extension of randomized response that models both the true and privatized outputs as continuous random variables, thus eliminating the need for discretization.

3 Blocked Framework of RR-type algorithms under Label DP

3.1 BlockRR

We propose *BlockRR*, a blocked framework for achieving Label DP. Its core mechanism partitions the true label set into disjoint subsets (determined in Subsection 3.2). Formally, let $\mathcal{S}$ be the original label set. We partition it into two disjoint subsets, $\mathcal{S}_1$ and $\mathcal{S}_2$, where $\mathcal{S}_1$ typically contains majority or important labels and $\mathcal{S}_2$ contains minority or less important labels. Let $\tilde{\mathcal{S}} \subseteq \mathcal{S}$ denote the privatized label set, which is correspondingly partitioned into $\tilde{\mathcal{S}}_1 = \tilde{\mathcal{S}} \cap \mathcal{S}_1$ and $\tilde{\mathcal{S}}_2 = \tilde{\mathcal{S}} \setminus \tilde{\mathcal{S}}_1$. Let $B : \mathcal{S} \to \mathcal{X}$ be a mapping such that for every $i \in \{1, 2\}$ and every $y \in \mathcal{S}_i$, we have $y \in B(y) \subseteq \tilde{\mathcal{S}}_i$. Here $\mathcal{X}$ is a set of all subsets of $\mathcal{S}$. As illustrated in Figure 1, the Cartesian product of the true and privatized label sets is thus divided into four regions. We proceed by constructing the overall randomized algorithm in a modular fashion, designing the specific perturbation rules for each region.

In practice, majority labels often dominate the dataset. To improve the performance of Label DP algorithms, we focus on leveraging the information-rich majority labels while appropriately mitigating the influence of minority labels, which may carry limited useful information. To this end, we introduce a subset $\mathcal{T} \subseteq \tilde{\mathcal{S}}_1$, which is a subset of the most dominated privatized labels.

We now define the transition probability for each block. Beginning with the bottom-left block, corresponding to the pair $(\mathcal{S}_2, \tilde{\mathcal{S}}_1)$, we define

$$\Pr(\tilde{Y} = \tilde{y}, \tilde{y} \in \mathcal{T} | Y = y, y \in \mathcal{S}_2) = \frac{1}{|\tilde{\mathcal{S}}|},$$

$$\Pr(\tilde{Y} = \tilde{y}, \tilde{y} \in \tilde{\mathcal{S}}_1 \setminus \mathcal{T} | Y = y, y \in \mathcal{S}_2) = \beta.$$

Similarly, for the bottom-right block, corresponding to the pair $(\mathcal{S}_2, \tilde{\mathcal{S}}_2)$, we define

$$\Pr(\tilde{Y} = \tilde{y}, \tilde{y} \in B(y) | Y = y, y \in \mathcal{S}_2) = e^{\epsilon}\gamma,$$
$$\Pr(\tilde{Y} = \tilde{y}, \tilde{y} \in \tilde{\mathcal{S}}_2 \setminus B(y) | Y = y, y \in \mathcal{S}_2) = \gamma.$$

For the top-left block (corresponding to the pair $\mathcal{S}_1$ and $\tilde{\mathcal{S}}_1$) and the top-right block (corresponding to the pair $\mathcal{S}_1$ and $\tilde{\mathcal{S}}_2$), the function is specified as follows:

$$\Pr(\tilde{Y} = \tilde{y}, \tilde{y} \in B(y) | Y = y, y \in \mathcal{S}_1) = e^{\epsilon}\beta,$$
$$\Pr(\tilde{Y} = \tilde{y}, \tilde{y} \in \tilde{\mathcal{S}}_1 \setminus B(y) | Y = y, y \in \mathcal{S}_1) = \beta,$$
$$\Pr(\tilde{Y} = \tilde{y}, \tilde{y} \in \tilde{\mathcal{S}}_2 | Y = y, y \in \mathcal{S}_1) = \gamma.$$

By the probabilistic property $\mathbb{E}_{\tilde{y}}[\Pr(\tilde{Y} = \tilde{y} | Y = y)] = 1,\ \forall y \in \mathcal{S}$, we have the following system of linear equations

$$\begin{cases} (e^{\epsilon}|B(y)| + |\tilde{\mathcal{S}}_1| - |B(y)|) \cdot \beta + |\tilde{\mathcal{S}}_2| \cdot \gamma = 1,\ \forall y \in \mathcal{S}_1, \\ (|\tilde{\mathcal{S}}_1| - |\mathcal{T}|) \cdot \beta + (e^{\epsilon}|B(y)| + |\tilde{\mathcal{S}}_2| - |B(y)|) \cdot \gamma = 1 - \frac{|\mathcal{T}|}{|\tilde{\mathcal{S}}|},\ \forall y \in \mathcal{S}_2, \end{cases} \tag{3.1}$$

For any $i \in \{1, 2\}$, assume that for every $y \in \mathcal{S}_i$, $y \in B(y) \subseteq \tilde{\mathcal{S}}_i$ with cardinality $|B(y)| = n_i$, where $|\cdot|$ is the cardinality of a set. Additionally, suppose $|\mathcal{T}| = l$. Under these assumptions, solving the linear system in Equation (3.1) yields the following result:

$$\beta = \frac{\beta_1}{\kappa},\ \gamma = \frac{\gamma_1}{\kappa}, \tag{3.2}$$

where

$$\beta_1 = (e^{\epsilon} - 1)n_2 + \frac{l \cdot |\tilde{\mathcal{S}}_2|}{|\tilde{\mathcal{S}}|}, \qquad \gamma_1 = ((e^{\epsilon} - 1)n_2 + l) - \frac{l}{|\tilde{\mathcal{S}}|} \cdot ((e^{\epsilon} - 1)n_1 + |\tilde{\mathcal{S}}_1|),$$
$$\kappa = ((e^{\epsilon} - 1)n_1 + |\tilde{\mathcal{S}}_1|) \cdot ((e^{\epsilon} - 1)n_2 + |\tilde{\mathcal{S}}_2|) - (|\tilde{\mathcal{S}}_1| - l) \cdot |\tilde{\mathcal{S}}_2|.$$

For these four mapping regions (Figure 1), the diagonal regions ($\mathcal{S}_1 \to \tilde{\mathcal{S}}_1$ and $\mathcal{S}_2 \to \tilde{\mathcal{S}}_2$) follow standard RR when $n_1 = n_2 = 1$, while the off-diagonal regions use a uniform or block-uniform distribution. This construction ensures both normalization, $\mathbb{E}_{\tilde{y}}[\Pr(\tilde{Y} = \tilde{y} | Y = y)] = 1$ and ϵ-label DP. For completeness, the whole algorithm is shown in Algorithm 1, denoted as ***BlockRR***.

3.2 Partition by Weight Matrix

This section presents a method for partitioning the true label space, based on a weight matrix derived from the label prior distribution. Given the prior, we define the (i, j)-th entry of the weight matrix as

$$w_{ij} = \Pr(\tilde{Y} = j) \exp\left(-\frac{\mathbb{I}(i \neq j)}{\sigma}\right), \tag{3.3}$$

where $\sigma > 0$ is a hyperparameter and $\mathbb{I}(\cdot)$ is an indicator function. We assume that the distribution of $\tilde{Y}$ aligns with that of Y. Note that $\tilde{\mathcal{S}} \subseteq \mathcal{S}$, we set the marginal distribution of the privatized label $\tilde{Y}$ matches the corresponding values of the true label Y. Formally, this means $\Pr(\tilde{Y} = j) = \Pr(Y = j) / \sum_{j' \in \tilde{\mathcal{S}}} \Pr(Y = j'),\ \forall j \in \tilde{\mathcal{S}}$. In practice, the prior distribution of the true dataset is often unknown. We therefore adopt a two-stage approach: first, a subset of the training data is used to estimate the prior, the remaining data are then used for model training. Specifically, we split the dataset D into two disjoint parts D_1 and D_2 randomly, and obtain the prior estimate $\boldsymbol{p} = [p_1\ \cdots\ p_K]^{\top}$ via Algorithm 3.

Using the weight matrix, we partition the full label set $\mathcal{S}$ into two disjoint subsets: $\mathcal{S}_1 = \{i : w_{ii} \geq w_{ij}, \forall j \in \tilde{\mathcal{S}}\}$ and $\mathcal{S}_2 = \mathcal{S} \setminus \mathcal{S}_1$. Once the partition of $\mathcal{S}$ is obtained, we can apply the BlockRR algorithm to randomize the target dataset. Algorithm 2 details the steps for generating the randomized dataset.

Algorithm 1 BlockRR$_\epsilon(y)$

Input: $\mathcal{S}, \mathcal{S} = \mathcal{S}_1 \cup \mathcal{S}_2, \mathcal{S}_1 \cap \mathcal{S}_2 = \emptyset, \tilde{\mathcal{S}} \subseteq \mathcal{S}, \tilde{\mathcal{S}}_1 = \tilde{\mathcal{S}} \cap \mathcal{S}_1$ and $\tilde{\mathcal{S}}_2 = \tilde{\mathcal{S}} \setminus \tilde{\mathcal{S}}_1, \mathcal{T} \subseteq \tilde{\mathcal{S}}_1$, privacy parameter ϵ, cardinality l and mapping B.
Compute β and γ by Equation (3.2),
if $y \in \mathcal{S}_2$ **then**
    Sample $\tilde{y} \in \tilde{\mathcal{S}}$ according to

$$\Pr(\tilde{Y} = \tilde{y} \mid Y = y) = \begin{cases} \frac{1}{|\tilde{\mathcal{S}}|}, & \text{if } \tilde{y} \in \mathcal{T}, \\ e^{\epsilon} \cdot \gamma, & \text{if } \tilde{y} \in B(y), \\ \gamma, & \text{if } \tilde{y} \notin B(y) \cup \mathcal{T}, \end{cases}$$

else if $y \in \mathcal{S}_1$ **then**
    Sample $\tilde{y}$ with probability:

$$\Pr(\tilde{Y} = \tilde{y} \mid Y = y) = \begin{cases} \gamma & \text{if } \tilde{y} \in \tilde{S}_2, \\ e^{\epsilon} \cdot \beta & \text{if } \tilde{y} \in B(y), \\ \beta & \text{if } \tilde{y} \in \tilde{S}_1 \setminus B(y). \end{cases}$$

end if
Output: Randomized labels $\tilde{y}$.

Algorithm 2 Generation of the randomized dataset.

Input: $\mathcal{S}, \tilde{\mathcal{S}}$, dataset D, privacy parameter ϵ, cardinality l and mapping B.
Split the dataset D into two disjoint parts D_1 and D_2 randomly,
Prior estimation by D_1 with privacy budget ϵ via Algorithm 3,
Compute weight matrix $W = (w_{ij})$ from Equation (3.3),
$\mathcal{S}_1 = \{i : w_{ii} \geq w_{ij}, \forall j \in \tilde{\mathcal{S}}\}$ and $\mathcal{S}_2 = \mathcal{S} \setminus \mathcal{S}_1$,
$\tilde{\mathcal{S}}_1 = \tilde{\mathcal{S}} \cap \mathcal{S}_1$ and $\tilde{\mathcal{S}}_2 = \tilde{\mathcal{S}} \setminus \tilde{\mathcal{S}}_1$,
Compute $\mathcal{T}$ as the set of indices corresponding to the l largest values of prior in $\tilde{\mathcal{S}}_1$,
$\tilde{D}_2 = \emptyset$,
for $(x, y) \in D_2$ **do**
    $\tilde{y} = \text{BlockRR}_\epsilon(y)$,
    $\tilde{D}_2 = \tilde{D}_2 \cup (x, \tilde{y})$.
end for
Output: $\tilde{D}_2$.

4 Theoretical Results

Traditional Randomized Response applies a one-size-fits-all perturbation strategy to all labels, disregarding dataset-specific structures during randomization. To better account for data distribution characteristics, some mechanisms (Ghazi et al., 2021; Liu & Huang, 2026) incorporate prior information to adjust the randomization mechanism. A key limitation of this method, however, is its discarding of minority labels, making it unsuitable for tasks where per-class accuracy matters. To address these shortcomings, we propose the BlockRR algorithm, which introduces configurable, block-wise randomization mechanisms tailored to different regions of the label space.

The design of a suitable randomized mechanism reduces to specifying the conditional probability values $p_{\tilde{y}|y}$. To encode dataset-specific structure, we introduce a weight for each such conditional probability. These weights are organized into a matrix, where the entry $w_{y\tilde{y}}$ corresponding to $(y, \tilde{y})$ is defined as Equation (3.3). When the weight matrix is diagonally dominant, a standard RR mechanism is a suitable choice. However, when the weight matrix is not diagonally dominant, we partition $\mathcal{S}$ into two subsets: $\mathcal{S}_1 = \{i : w_{ii} \geq w_{ij}, \forall j\}$ and $\mathcal{S}_2 = \mathcal{S} \setminus \mathcal{S}_1$. For labels in $\mathcal{S}_2$, the condition $w_{ii} \geq w_{ij}, \forall j$ does not hold. Yet we still aim to keep $p_{\tilde{y}|y}, \forall \tilde{y} \in B(y)$ as large as possible for all $y \in \mathcal{S}_2$ to maintain training performance. To this end, we define a set of dominating labels and

introduce a subset $\mathcal{T} \subseteq \tilde{\mathcal{S}}_1$ as the set of indices corresponding to the l largest values of $\{p_i\}_{i \in \tilde{\mathcal{S}}_1}$ and impose the constraint $p_{\tilde{y}|y} = \frac{1}{|\tilde{\mathcal{S}}|}, \forall \tilde{y} \in \mathcal{T}$ when $y \in \mathcal{S}_2$.

We now formulate the following optimization problem:

$$\begin{aligned} \max \quad & \sum_{y \in \mathcal{S}, \tilde{y} \in \tilde{\mathcal{S}}} w_{y\tilde{y}} p_{\tilde{y}|y} \\ \text{s.t.} \quad & \begin{cases} p_{\tilde{y}|y} \leq e^{\epsilon} \cdot p_{\tilde{y}|y'}, & \forall \tilde{y} \in \tilde{\mathcal{S}}, \forall y, y' \in \mathcal{S}, \\ p_{\tilde{y}|y} = \frac{1}{|\tilde{\mathcal{S}}|}, & \forall y \in \mathcal{S}_2, \forall \tilde{y} \in \mathcal{T}, \\ \sum_{\tilde{y} \in \tilde{\mathcal{S}}} p_{\tilde{y}|y} = 1, & \forall y \in \mathcal{S}, \\ p_{\tilde{y}|y} \geq 0, & \forall \tilde{y} \in \tilde{\mathcal{S}}, \forall y \in \mathcal{S}, \\ p_{\tilde{y}|y} = p_{\tilde{y}|\tilde{y}}, & \forall y \in \mathcal{S}, \forall \tilde{y} \in B(y), \end{cases} \end{aligned} \tag{4.1}$$

where the first constraint enforces the ϵ-label differential privacy requirement, while the third and fourth constraints arise from the probabilistic nature of the conditional distribution (non-negativity and normalization), the final inequality is a direct consequence of our assumption that, for every $i \in \{1, 2\}$ and any $y \in \mathcal{S}_i$, the set $B(y)$ contains y and is contained in $\tilde{\mathcal{S}}_i$.

Theorem 4.1. *For any $i \in \{1, 2\}$, assume that for every $y \in \mathcal{S}_i$, we assume $y \in B(y) \subseteq \tilde{\mathcal{S}}_i$ with cardinality $|B(y)| = n_i$. Suppose $|\mathcal{T}| = l$. Then BlockRR (Algorithm 1) is the optimal solution of the optimization problem in Equation* (4.1).

We defer the proof of Theorem 4.1 to Appendix C. Next, we will illustrate that the introduction of $\mathcal{T} \subseteq \tilde{\mathcal{S}}_1$ is designed to enhance the influence of majority labels and mitigate the impact of minority labels.

Lemma 4.2. *Under the same assumptions as that in Theorem 4.1, β is monotonically increasing and γ is monotonically decreasing for $l \in [0, |\tilde{S}_1|]$. Furthermore,* $\max_l \gamma = \min_l \beta$.

The proof of Lemma 4.2 is in Appendix D. As formalized in Lemma 4.2, the cardinality l governs the strength of minority-label mitigation. The lemma shows that when $l = 0$ (i.e. $\mathcal{T} = \emptyset$), mitigation is entirely absent; when $l = |\tilde{\mathcal{S}}_1|$ (i.e. $\mathcal{T} = \tilde{\mathcal{S}}_1$), mitigation is strongest. Consequently, the parameter γ is maximized and β is minimized at $l = 0$, and under the condition $\beta = \gamma$, BlockRR reduces precisely to the classical RR algorithm.

In addition, BlockRR serves as a general framework. The following theorem demonstrates that the RR and RRWithPrior algorithms correspond to specific configurations of BlockRR, thereby unifying them within our framework and eliminating the need for separate, case-by-case analysis. The RR algorithm admits two valid configuration options (first two lines in Table 1), either of which is acceptable.

Proposition 4.3. *Our proposed algorithm, BlockRR, can recover various existing mechanisms under different parameter configurations:*

Table 1: General schemes under various parameter configurations.

Method	$\mathcal{S}_1$	$\mathcal{S}_2$	$\tilde{\mathcal{S}}_1$	$\tilde{\mathcal{S}}_2$	B	l
RR	$\mathcal{S}$	$\emptyset$	$\mathcal{S}_1$	$\emptyset$	$y \to \begin{cases} \{y\}, & \forall y \in \mathcal{S}_1 \\ \emptyset, & \forall y \in \mathcal{S}_2 \end{cases}$	–
RR	$\mathcal{S}_1$	$\mathcal{S}_2$	$\mathcal{S}_1$	$\mathcal{S}_2$	$y \to \{y\}$	0
RRWithPrior	$\mathcal{S}_1$	$\mathcal{S}_2$	$\mathcal{S}_1$	$\emptyset$	$y \to \begin{cases} \{y\}, & \forall y \in \mathcal{S}_1 \\ \emptyset, & \forall y \in \mathcal{S}_2 \end{cases}$	$\lvert\tilde{\mathcal{S}}_1\rvert$

The difference of the computational complexity is from the separation of the true label set $\mathcal{S}$ among all three mechanisms. There is no separation for the standard RR mechanism, which samples directly from the label set without any preprocessing. In contrast, BlockRR requires an extra step to determine the two disjoint subsets $\mathcal{S}_1$ and $\mathcal{S}_2$ of $\mathcal{S}$, which has a complexity of $O(K^2)$ for the computation of the weight matrix, where K is the number of labels. RRWithPrior further incurs a complexity of $O(K^2)$ to find the optimal k^*.

Next, we will prove the property of label DP, whose proofs are in Appendix E.

Theorem 4.4. *BlockRR$_\epsilon$ (Algorithm 1) is ϵ-Label DP.*

Theorem 4.5. *Algorithm 2 is also ϵ-label DP.*

Finally, we discuss the utility guarantee. we provide a theoretical analysis of the utility gap between the optimal Bayes classifier and the classification models trained with privacy protection. We quantify this gap by deriving formal bounds on the excess risk. Throughout this section, let $\mathbf{x}$ denote the feature random vector, Y the true label random variable, and $\tilde{Y}$ the privatized response obtained via privacy mechanisms.

Definition 4.6 (Excess Risk)**.** For each $y \in \mathcal{S}$, define the conditional probability $\eta_y(\boldsymbol{x}) = \Pr(Y = y \mid \mathbf{x} = \boldsymbol{x})$, $\forall y \in \mathcal{S}$. Let $\{g_{\tilde{Y}}(\boldsymbol{x})\}_{\tilde{Y}\in\tilde{\mathcal{S}}}$ be the classifier trained on randomized data. The Bayes optimal classifier and the optimal label on the classifier trained on the randomized dataset at a point $\boldsymbol{x}$ are given by

$$c^*(\boldsymbol{x}) = \arg\max_{y\in\mathcal{S}} \eta_y(\boldsymbol{x}),\ \tilde{c}(\boldsymbol{x}) = \arg\max_{\tilde{y}\in\tilde{\mathcal{S}}} g_{\tilde{y}}(\boldsymbol{x}).$$

The associated risks under the $0-1$ loss are

$$R^* = \mathbb{E}\left[\mathbb{I}\left(c^*(\mathbf{x}) \neq y\right)\right] = \Pr\left(c^*(\mathbf{x}) \neq y\right),\ \tilde{R} = \mathbb{E}\left[\mathbb{I}\left(\tilde{c}(\mathbf{x}) \neq y\right)\right] = \Pr\left(\tilde{c}(\mathbf{x}) \neq y\right).$$

The *excess risk* is defined as the difference $\tilde{R} - R^*$.

The following theorem formally bounds the excess risk induced by BlockRR label perturbation, whose proof is in Appendix F.

Theorem 4.7. *Use the same notations as Definition 4.6 and let $s_j = \mathbb{I}(j \in \tilde{\mathcal{S}}_1) + \frac{\beta}{\gamma}\mathbb{I}(j \in \tilde{\mathcal{S}}_2)$, $j \in \tilde{\mathcal{S}}$. Define*

$$\Delta(\boldsymbol{x}) = \max_{j\in\tilde{\mathcal{S}}} \left|g_j(\boldsymbol{x}) - s_j\tilde{\eta}_j(\boldsymbol{x})\right|, \qquad \Delta'(\boldsymbol{x}) = \frac{1}{(e^\epsilon - 1)\beta}\Delta(\boldsymbol{x}).$$

When $n_1 = n_2 = 1$, the excess risk between the Bayes optimal classifier and the classifier trained on BlockRR-privatized labels satisfies

$$\tilde{R} - R^* \leq \mathbb{E}_{\boldsymbol{x}\sim\mathbf{x}}[A + 2\Delta'(\boldsymbol{x})], \qquad A = \frac{1}{(e^\epsilon - 1)\beta}\left(\frac{1}{|\tilde{\mathcal{S}}|} - \beta\right)\sum_{\ell\in\mathcal{S}_2}\eta_\ell(\boldsymbol{x}),$$

where $\mathbf{x}$ *is the feature random vector.*

Remark 4.8. In the high-privacy regime (smaller ϵ), $\beta \gg \gamma$ and $\Delta'(\boldsymbol{x})$ grows about ϵ. BlockRR mitigates the utility degradation of vanilla randomized response via the block-wise separation of majority and minority labels ($\mathcal{S}_1/\mathcal{S}_2$). In the low-privacy regime (larger ϵ), we have $\beta \to \gamma$. When l is small, the bound degenerates to the standard excess risk bound for RR. This is consistent with our empirical observation that BlockRR reduces to vanilla RR when $\mathcal{S}_2 = \emptyset$.

5 Numerical Experiments

In this section, we empirically evaluate our proposed label differential privacy mechanism on image classification tasks, comparing it with the baseline RR and RRWithPrior mechanisms under varying privacy budgets ϵ. Subsection 5.1 describes the datasets we use, which is followed by the experimental results on these datasets, respectively. Owing to space constraints, we relegate the detailed experimental configurations, and ablation analyses to Appendix G.

5.1 Dataset

To better reflect real-world class imbalance, we generate some variants from CIFAR-10 (Krizhevsky, 2009), CIFAR-100 (Krizhevsky, 2009) and MNIST (LeCun et al., 2002) via non-uniform sampling (Buda et al., 2018).

The CIFAR-10 dataset contains 60,000 32×32 color images, divided into 10 classes, with 50,000 images used for training (5,000 per class) and 10,000 images used for testing (1,000 per class). To

evaluate those methods, we perform non-uniform sampling on the original CIFAR-10 training set to construct two imbalanced variants. Each derived dataset preserves the original test set but modifies the number of training samples per class as follows:

CIFAR-10$_1$: Training samples per class (0–9):

$$[5000, 4900, 4700, 4600, 4500, 4800, 1000, 1500, 1000, 1500].$$

The test set for each class contains $\frac{1}{10}$ of its corresponding training count.

CIFAR-10$_2$: Training samples per class (0–9):

$$[5000, 4900, 4700, 4600, 4500, 4800, 600, 500, 700, 400].$$

Similarly, the test set for each class is scaled to $\frac{1}{10}$ of its training count.

We also conduct experiments on a variant of MNIST (**MNIST**$_1$). The per-class training sample sizes, selected randomly, are $[5000, 4900, 4700, 4600, 4500, 4800, 600, 500, 700, 400]$, and the test set for each class is scaled to one-tenth of its training count.

For CIFAR-100, we select training samples randomly for 500 samples for the first 90 classes and 400 samples for the subsequent 5 classes and 100 samples for the last 5 classes. the test set for each class is scaled to one-fifth of its training count.

5.2 Experiments on CIFAR-10

Table 2: Test Accuracy (%) on CIFAR-10$_1$ (best in bold, second-best underlined).

Privacy Budget	RR		RRWithPrior		Ours	
	overall	per-class	overall	per-class	overall	per-class
$\epsilon = 0.6$	$\underline{39.75 \pm 1.44}$	**29.14**	37.74 ± 4.06	26.17	$\mathbf{40.31 \pm 2.08}$	$\underline{28.52}$
$\epsilon = 0.8$	46.18 ± 1.51	**37.25**	$\mathbf{47.54 \pm 4.32}$	33.07	$\underline{47.11 \pm 1.43}$	$\underline{35.40}$
$\epsilon = 1.0$	53.66 ± 2.18	**46.01**	$\underline{54.86 \pm 4.19}$	38.42	$\mathbf{55.12 \pm 2.69}$	$\underline{43.30}$
$\epsilon = 2.0$	$\underline{78.74 \pm 0.48}$	**75.96**	73.20 ± 1.53	50.08	$\mathbf{78.86 \pm 0.56}$	$\underline{74.03}$
$\epsilon = 3.0$	$\mathbf{87.67 \pm 0.50}$	**87.21**	83.35 ± 1.93	70.85	$\underline{87.60 \pm 0.51}$	$\underline{86.09}$

For CIFAR-10$_1$, Table 2 presents the quantitative comparisons of RR, RRWithPrior, and our proposed method across privacy budgets, including the overall test accuracy and the average of per-class test accuracy.

Through numerical experiments in the dataset CIFAR-10$_1$, we observe that when $\epsilon \geq 4.0$, the equality $\mathcal{S}_1 = \tilde{\mathcal{S}}_1 = \mathcal{S} = [10]$ holds for all the mechanisms considered above. This result implies that these mechanisms reduce exactly to the standard randomized response (RR) mechanism. Therefore, in such cases, it becomes unnecessary to set aside a separate subset of the dataset for the purpose of prior estimation.

In the high-privacy regime ($\epsilon = 0.6, 0.8, 1.0$), our BlockRR mechanism demonstrates a clear advantage, achieving accuracies of 40.31% ($\epsilon = 0.6$) and 55.12% ($\epsilon = 1.0$), respectively. For the average of per-class accuracy reported in Table 2, our proposed mechanism is superior to RRWithPrior, while maintaining competitive overall accuracy compared to baseline RR. Moreover, our proposed method gets a better balance between test accuracy and the average of per-class accuracy. For moderate budgets ($\epsilon = 2.0, 3.0$), our method attains results comparable to those of the baselines.

Next, we evaluate those mechanisms on CIFAR-10$_2$, reporting the overall accuracy and the average of per-class accuracy in Table 3. In the high-privacy regime ($\epsilon \leq 1.0$), both RRWithPrior and our method achieve higher overall accuracy than RR. Notably, our mechanism demonstrates substantially lower variance than RRWithPrior, indicating superior stability. For $\epsilon \in \{2.0, 3.0, 4.0\}$, both RR and our proposed mechanism show better performance than the RRWithPrior mechanism. Our proposed mechanism ranks first or second among all methods. Consistent with the numerical results on the CIFAR-10$_1$ dataset, all mechanisms degenerate to the standard RR mechanism when $\epsilon \geq 6.0$.

5.3 Experiments on MNIST

We also conduct experiments on a variant of MNIST (**MNIST**$_1$). Table 4 illustrates the quantitative results on the MNIST$_1$ dataset. As shown in the table, RR achieves performance comparable to that of

Table 3: Test Accuracy(%) on CIFAR-10_2 (best in bold, second-best underlined).

Privacy Budget	RR		RRWithPrior		Ours	
	overall	per-class	overall	per-class	overall	per-class
$\epsilon = 0.6$	42.55 ± 1.71	$\underline{27.43}$	$\underline{42.70 \pm 4.62}$	27.24	$\mathbf{43.48 \pm 1.98}$	**27.82**
$\epsilon = 0.8$	48.70 ± 2.59	33.08	$\mathbf{52.12 \pm 3.57}$	**33.48**	$\underline{51.01 \pm 1.29}$	$\underline{33.24}$
$\epsilon = 1.0$	55.82 ± 2.33	**45.95**	$\mathbf{59.48 \pm 5.94}$	38.42	$\underline{58.26 \pm 1.67}$	$\underline{43.30}$
$\epsilon = 2.0$	$\underline{79.09 \pm 0.82}$	**67.96**	78.91 ± 0.76	50.87	$\mathbf{79.14 \pm 0.44}$	$\underline{60.09}$
$\epsilon = 3.0$	$\mathbf{87.52 \pm 0.32}$	**81.67**	84.05 ± 1.93	54.94	$\underline{87.18 \pm 0.30}$	$\underline{80.90}$
$\epsilon = 4.0$	$\underline{90.22 \pm 0.17}$	**89.24**	88.65 ± 1.42	74.01	$\mathbf{90.36 \pm 0.33}$	$\underline{86.01}$

our method. Even under relatively small privacy budgets($\epsilon = 0.6, 0.8$), both methods maintain high accuracy. Moreover, our method demonstrates greater stability than RRWithPrior. When $\epsilon = 4.0$, the accuracy of both methods exceeds 99.5%, which is nearly close to that in the non-private setting. For the MNIST dataset, all mechanisms reduce to the RR mechanism when $\epsilon \geq 6.0$.

Table 4: Test Accuracy (%) on MNIST_1 (best in bold, second-best underlined).

Privacy Budget	RR		RRWithPrior		Ours	
	overall	per-class	overall	per-class	overall	per-class
$\epsilon = 0.6$	$\underline{88.84 \pm 0.57}$	$\underline{57.81}$	61.11 ± 11.87	39.27	$\mathbf{89.02 \pm 0.45}$	**58.06**
$\epsilon = 0.8$	$\mathbf{91.38 \pm 0.68}$	**63.32**	71.77 ± 11.85	46.35	$\underline{91.06 \pm 0.67}$	$\underline{61.87}$
$\epsilon = 1.0$	$\underline{93.73 \pm 0.87}$	$\underline{72.02}$	90.21 ± 4.45	58.27	$\mathbf{93.86 \pm 0.71}$	**72.19**
$\epsilon = 2.0$	$\underline{98.89 \pm 0.13}$	$\underline{96.24}$	92.47 ± 0.05	59.77	$\mathbf{99.02 \pm 0.15}$	**96.79**
$\epsilon = 3.0$	$\underline{99.38 \pm 0.05}$	$\underline{98.23}$	92.78 ± 0.68	60.80	$\mathbf{99.41 \pm 0.08}$	**98.27**
$\epsilon = 4.0$	$\underline{99.52 \pm 0.07}$	$\underline{98.40}$	97.07 ± 1.03	84.10	$\mathbf{99.55 \pm 0.07}$	**98.57**

5.4 Experiments on CIFAR-100

Finally, we evaluate our proposed method on the CIFAR-100 dataset. Numerical comparisons are presented in Table 5. For high privacy budgets ($\epsilon \leq 3.0$), our method performs best; for low privacy budgets ($\epsilon \in \{4.0, 6.0\}$), our method is the second-best only to RR, which achieves the best performance. When $\epsilon \geq 8.0$, all mechanisms reduce to the RR mechanism.

Table 5: Test Accuracy(%) on CIFAR-100 (best in bold, second-best underlined).

Privacy Budget	RR		RRWithPrior		Ours	
	overall	per-class	overall	per-class	overall	per-class
$\epsilon = 1.0$	1.81 ± 0.25	1.77	$\underline{2.02 \pm 0.39}$	$\underline{1.92}$	$\mathbf{2.11 \pm 0.38}$	**2.07**
$\epsilon = 2.0$	7.17 ± 0.39	6.88	$\underline{7.38 \pm 0.95}$	$\underline{7.03}$	$\mathbf{7.69 \pm 0.48}$	**7.39**
$\epsilon = 3.0$	$\underline{21.60 \pm 0.71}$	$\underline{20.79}$	16.69 ± 1.24	15.99	$\mathbf{21.63 \pm 0.89}$	**20.80**
$\epsilon = 4.0$	$\mathbf{43.30 \pm 0.60}$	**41.73**	35.70 ± 1.26	34.17	$\underline{43.15 \pm 0.65}$	$\underline{41.54}$
$\epsilon = 6.0$	$\mathbf{67.08 \pm 0.33}$	**65.32**	65.24 ± 1.17	62.96	$\underline{67.03 \pm 0.23}$	$\underline{65.25}$

6 Conclusion

In this paper, we introduce BlockRR, a novel RR-type mechanism with block structure for label DP. This framework generalizes existed RR and RRWithPrior mechanisms as special cases under specific parameter settings. Theoretically, we formulate an optimization problem whose optimal solution corresponds to the BlockRR mechanism. We establish its ϵ-label DP guarantee and conduct a utility analysis by deriving bounds on the excess risk. Empirically, we evaluate BlockRR on variants of the CIFAR-10, CIFAR-100 and MNIST datasets. Results show that our proposed method shows a better balance between test accuracy and the average of per-class accuracy. We acknowledge the limitations of the current study. Specifically, our theoretical guarantees on the excess risk rely on the assumption $n_1 = n_2 = 1$. The generalization to cases where $B(y)$ has larger cardinality proves to be significantly more challenging, and we defer this extension to future research.

AI USE STATEMENT

In this work, we used generative AI tools for language polishing only. We have not used generative AI tools for other tasks with required disclosure, and the rest of the required disclosure tasks are not applicable to this work. We have reviewed all AI-assisted work. We take responsibility for the final content of this work, including text, claims or artifacts produced with the aid of generative AI.

REFERENCES

Martin Abadi, Andy Chu, Ian Goodfellow, H. Brendan McMahan, Ilya Mironov, Kunal Talwar, and Li Zhang. Deep learning with differential privacy. In *Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security*, CCS '16, pp. 308–318, New York, NY, USA, 2016.

Ashwinkumar Badanidiyuru, Badih Ghazi, Pritish Kamath, Ravi Kumar, Ethan Leeman, Pasin Manurangsi, Avinash V Varadarajan, and Chiyuan Zhang. Optimal unbiased randomizers for regression with label differential privacy. In A. Oh, T. Naumann, A. Globerson, K. Saenko, M. Hardt, and S. Levine (eds.), *Advances in Neural Information Processing Systems*, volume 36, pp. 60226–60246, 2023.

Léon Bottou. Large-scale machine learning with stochastic gradient descent. In *Proceedings of the 19th International Conference on Computational Statistics*, pp. 177–186, 2010.

Maciej Buda, Andrew Maki, and Maciej A Mazurowski. A systematic study of the class imbalance problem in convolutional neural networks. In *2018 IEEE 15th International Symposium on Biomedical Imaging (ISBI 2018)*, pp. 244–248, 2018.

Kamalika Chaudhuri and Daniel Hsu. Sample complexity bounds for differentially private learning. In Sham M. Kakade and Ulrike von Luxburg (eds.), *Proceedings of the 24th Annual Conference on Learning Theory*, volume 19 of *Proceedings of Machine Learning Research*, pp. 155–186, Budapest, Hungary, 09–11 Jun 2011.

Kamalika Chaudhuri, Claire Monteleoni, and Sarwate Anand D. Differentially private empirical risk minimization. *Journal of Machine Learning Research*, 12:1069–1109, March 2011.

Cynthia Dwork. Differential privacy. In *International colloquium on automata, languages, and programming*, pp. 1–12. Springer, 2006.

Cynthia Dwork and Aaron Roth. The algorithmic foundations of differential privacy. *Foundations and Trends in Theoretical Computer Science*, 9(3-4):211 – 487, 2013.

Cynthia Dwork, Krishnaram Kenthapadi, Frank McSherry, Ilya Mironov, and Moni Naor. Our data, ourselves: Privacy via distributed noise generation. In Serge Vaudenay (ed.), *Advances in Cryptology - EUROCRYPT 2006*, pp. 486–503, Berlin, Heidelberg, 2006a.

Cynthia Dwork, Frank McSherry, Kobbi Nissim, and Adam Smith. Calibrating noise to sensitivity in private data analysis. In Shai Halevi and Tal Rabin (eds.), *Theory of Cryptography*, pp. 265–284, Berlin, Heidelberg, 2006b.

Úlfar Erlingsson, Vasyl Pihur, and Aleksandra Korolova. Rappor: Randomized aggregatable privacy-preserving ordinal response. In *Proceedings of the 2014 ACM SIGSAC Conference on Computer and Communications Security*, CCS '14, pp. 1054–1067, New York, NY, USA, 2014.

Hossein Esfandiari, Vahab Mirrokni, Umar Syed, and Sergei Vassilvitskii. Label differential privacy via clustering. In *International Conference on Artificial Intelligence and Statistics*, pp. 7055–7075, 2022a.

Hossein Esfandiari, Vahab Mirrokni, Umar Syed, and Sergei Vassilvitskii. Label differential privacy via clustering. In *International Conference on Artificial Intelligence and Statistics*, pp. 7055–7075. PMLR, 2022b.

Badih Ghazi, Noah Golowich, Ravi Kumar, Pasin Manurangsi, and Chiyuan Zhang. Deep learning with label differential privacy. *Advances in neural information processing systems*, 34:27131–27145, 2021.

Badih Ghazi, Pritish Kamath, Ravi Kumar, Ethan Leeman, Pasin Manurangsi, Avinash V. Varadarajan, and Chiyuan Zhang. Regression with label differential privacy. *ArXiv*, abs/2212.06074, 2022.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 770–778, 2016.

Geoffrey E Hinton, David E Rumelhart, and Ronald J Williams. Learning representations by back-propagating errors. *Nature*, 323(6088):533–536, 1986.

Peter Kairouz, Sewoong Oh, and Pramod Viswanath. Extremal mechanisms for local differential privacy. *Journal of Machine Learning Research*, 17(17):1–51, 2016.

Alex Krizhevsky. Learning multiple layers of features from tiny images. In *Technical Report, University of Toronto*, 2009.

Yann LeCun, Léon Bottou, Yoshua Bengio, and Patrick Haffner. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2324, 2002.

Haixia Liu and Ruifan Huang. RPWithPrior: Label differential privacy in regression. *Transactions on Machine Learning Research*, 2026.

Pathum Chamikara Mahawaga Arachchige, Peter Bertok, Ibrahim Khalil, Dongxi Liu, Seyit Camtepe, and Mohammed Atiquzzaman. Local differential privacy for deep learning. *IEEE Internet of Things Journal*, 7(7):5827–5842, 2020.

Mani Malek Esmaeili, Ilya Mironov, Karthik Prasad, Igor Shilov, and Florian Tramer. Antipodes of label differential privacy: PATE and ALIBI. In M. Ranzato, A. Beygelzimer, Y. Dauphin, P.S. Liang, and J. Wortman Vaughan (eds.), *Advances in Neural Information Processing Systems*, volume 34, pp. 6934–6945, 2021.

Frank D. McSherry. Privacy integrated queries: an extensible platform for privacy-preserving data analysis. In *Proceedings of the 2009 ACM SIGMOD International Conference on Management of Data*, SIGMOD '09, pp. 19–30, New York, NY, USA, 2009.

Nicolas Papernot, Martín Abadi, Úlfar Erlingsson, Ian Goodfellow, and Kunal Talwar. Semi-supervised knowledge transfer for deep learning from private training data. In *International Conference on Learning Representations*, 2017.

Shuang Song, Kamalika Chaudhuri, and Anand D. Sarwate. Stochastic gradient descent with differentially private updates. In *2013 IEEE Global Conference on Signal and Information Processing*, pp. 245–248, 2013.

Stanley L. Warner. Randomized response: a survey technique for eliminating evasive answer bias. *Journal of the American Statistical Association*, 60 309:63–6, 1965.

Hongyi Zhang, Moustapha Cisse, Yann N Dauphin, and David Lopez-Paz. mixup: Beyond empirical risk minimization. *arXiv preprint arXiv:1710.09412*, 2017.

Puning Zhao, Jiafei Wu, Zhe Liu, Li Shen, Zhikun Zhang, Rongfei Fan, Le Sun, and Qingming Li. Enhancing learning with label local differential privacy by vector approximation. In *International Conference on Learning Representations*, 2025.

A Existing RR-type Algorithms

This section reviews RR-type algorithms, which provide privacy guarantees for personal data. We first outline the concepts of differential privacy and label differential privacy before detailing specific algorithms.

Definition A.1 (Differential Privacy, DP (Dwork, 2006))**.** Let $\epsilon, \delta \geq 0$. A randomized mechanism A is said to be (ϵ, δ)-differentially private if, for any two adjacent datasets D, D', and for any subset S of the output space of A, the probability inequality holds, $\Pr[A(D) \in S] \leq e^{\epsilon} \cdot \Pr[A(D') \in S] + \delta$. If $\delta = 0$, then the mechanism A is said to be ϵ-differentially private (ϵ-DP).

Definition A.2 (Label Differential Privacy, Label DP (Chaudhuri & Hsu, 2011))**.** Let $\epsilon, \delta \geq 0$. A randomized mechanism A is said to be (ϵ, δ)-label differentially private if, for any two datasets D, D' that differ by a single label, and for any subset S of the output space of A, the probability inequality holds, $\Pr[A(D) \in S] \leq e^{\epsilon} \cdot \Pr[A(D') \in S] + \delta$. If $\delta = 0$, then the mechanism A is said to be ϵ-label differentially private (ϵ-Label DP).

Next, we introduce the RR mechanism and its key variants.

Definition A.3 (Randomized Response, RR (Dwork & Roth, 2013))**.** Let K be a positive integer, and define $[K] := \{0, 1, ..., K-1\}$. Let ϵ be a parameter, and $y \in [K]$ be the true value, as input to RR_ϵ. When an observer wants to query the value of y, RR_ϵ outputs a random value $\tilde{y} \in [K]$, which satisfies the following probability distribution:

$$\Pr[\tilde{Y} = \tilde{y} \mid Y = \hat{y}] = \begin{cases} \dfrac{e^{\epsilon}}{e^{\epsilon} + K - 1}, & \text{for } \hat{y} = y, \\ \dfrac{1}{e^{\epsilon} + K - 1}, & \text{otherwise.} \end{cases}$$

RRWithPrior is a variant of RR that uses the prior distribution of the label y across the dataset. The algorithm outputs the top k classes with the highest probability, forming the candidate set $\mathcal{Y}_k$ ($\mathcal{Y}_k \subseteq [K]$).

Definition A.4 (Randomized Response with Prior, RRWithPrior (Ghazi et al., 2021))**.** Let K be a positive integer, $[K] = \{0, 1, 2, \ldots, K-1\}$ be the true label set and $\mathcal{Y}_k \subseteq [K]$ be the randomized set consisting of the indices of the top k components. Let ϵ be a parameter, and $y \in [K]$ be the true label. When an observer wants to query the value of $y \in \mathcal{Y}_k$, the randomized output $\tilde{y}$ obeys the following probability distribution:

$$\Pr[\tilde{Y} = \tilde{y} \mid Y = y \in \mathcal{Y}_k] = \begin{cases} \dfrac{e^{\epsilon}}{e^{\epsilon} + k - 1}, & \text{for } \hat{y} = y, \\ \dfrac{1}{e^{\epsilon} + k - 1}, & \text{for } \hat{y} \in \mathcal{Y}_k \setminus \{y\}, \end{cases}$$

If $y \notin \mathcal{Y}_k$, it outputs a value of $\mathcal{Y}_k$ uniformly.

While RR and RRWithPrior are primarily designed for privacy protection in classification tasks, the variants RRonBins and RPwithPrior extend the randomized-response framework to regression tasks.

Definition A.5 (Randomized Response on Bins, RRonBins (Ghazi et al., 2022))**.** Let ϵ be a privacy parameter and $\mathcal{Y}$ the set of true labels. Let $\Phi : \mathcal{Y} \to \tilde{\mathcal{Y}}$ be a deterministic grouping function, where $\tilde{\mathcal{Y}}$ is a finite set of size $|\tilde{\mathcal{Y}}|$. The output distribution of $\tilde{y} = \text{RRonBins}(y)$ is defined as:

$$\Pr[\tilde{Y} = \tilde{y} \mid Y = y] = \begin{cases} \dfrac{e^{\epsilon}}{e^{\epsilon} + |\tilde{\mathcal{Y}}| - 1}, & \text{for } \hat{y} = \Phi(y), \\ \dfrac{1}{e^{\epsilon} + |\tilde{\mathcal{Y}}| - 1}, & \text{otherwise.} \end{cases}$$

Definition A.6 (Regression Privacy with Prior, RPWithPrior (Liu & Huang, 2026))**.** Let $\epsilon > 0$ be a privacy parameter and $\zeta > 0$ a fixed positive constant. Consider a randomized mechanism $\mathcal{A} : \mathcal{Y} \to \tilde{\mathcal{Y}}$, where $y \in \mathcal{Y}$ and $\tilde{y} \in \tilde{\mathcal{Y}}$. For an optimal interval $\mathcal{I} = [A_1, A_2]$, define the neighborhoods $N_y = [y - \zeta, y + \zeta]$ and $N_{\mathcal{I}} = \bigcup_{y \in \mathcal{I}} N_y$. Let $\gamma = 2\zeta + e^{-\epsilon}(A_2 - A_1)$ and $f_{\tilde{Y}|Y}(\tilde{y} \mid y)$ denote the conditional probability density of $\tilde{Y}$ given $Y = y$. Then for any $y \in \mathcal{I}$, $f_{\tilde{Y}|Y}(\tilde{y} \mid y)$ satisfies:

$$f_{\tilde{Y}|Y}(\tilde{y} \mid y) = \begin{cases} 1/\gamma, & \text{for } \tilde{y} \in N_y, \\ e^{-\epsilon}/\gamma, & \text{for } \tilde{y} \in N_{\mathcal{I}} \setminus N_y, \\ 0, & \text{otherwise.} \end{cases}$$

If $y \notin \mathcal{I}$, $f_{\tilde{Y}|Y}(\tilde{y} \mid y) = \dfrac{1}{2\zeta + (A_2 - A_1)}, \forall \tilde{y} \in N_{\mathcal{I}}$.

B Laplace Mechanism for Estimating Probability Distribution

We begin by recalling the Laplace mechanism for probability distribution estimation, as described in (Ghazi et al., 2022).

Algorithm 3 Laplace Mechanism for Estimating Probability Distribution M_ϵ^{Lap}

1: **Input:** Privacy parameter $\epsilon \geq 0$, $D_1 = \{(x_i, y_i)\}_{i=1}^n$,
2: **for** $(x, y) \in D_1$ **do**
3: $\quad h_y \leftarrow$ number of i such that $y_i = y$,
4: $\quad h'_y \leftarrow \max\{h_y + \text{Lap}(2/\epsilon), 0\}$,
5: **end for**
6: **Output:** Distribution $\mathbf{p}$ over $\mathcal{Y}$ such that $p_y = \frac{h'_y}{\sum_{(x,y)\in D_1} h'_y}$.

C Proof of Theorem 4.1

Proof. From the first three constraints in Equation (4.1), we have the following results. For $y \in \mathcal{S}_2$,

$$\begin{aligned} 1 = \sum_{\tilde{y}\in\tilde{\mathcal{S}}} p_{\tilde{y}|y} &= \sum_{\tilde{y}\in B(y)} p_{\tilde{y}|y} + \sum_{\tilde{y}\in\tilde{\mathcal{S}}\setminus(\{B(y)\}\cup\mathcal{T})} p_{\tilde{y}|y} + \sum_{\tilde{y}\in\mathcal{T}} p_{\tilde{y}|y} \\ &\geq \sum_{\tilde{y}\in B(y)} p_{\tilde{y}|y} + e^{-\epsilon} \sum_{\tilde{y}\in\tilde{\mathcal{S}}\setminus(\{B(y)\}\cup\mathcal{T})} p_{\tilde{y}|\tilde{y}} + \sum_{\tilde{y}\in\mathcal{T}} p_{\tilde{y}|y} \end{aligned} \tag{C.1}$$

For $y \in \mathcal{S}_1$,

$$\begin{aligned} 1 = \sum_{\tilde{y}\in\tilde{\mathcal{S}}} p_{\tilde{y}|y} &= \sum_{\tilde{y}\in B(y)} p_{\tilde{y}|y} + \sum_{\tilde{y}\in\tilde{\mathcal{S}}_1\setminus\{B(y)\}} p_{\tilde{y}|y} + \sum_{\tilde{y}\in\tilde{\mathcal{S}}_2} p_{\tilde{y}|y} \\ &\geq \sum_{\tilde{y}\in B(y)} p_{\tilde{y}|y} + e^{-\epsilon} \sum_{\tilde{y}\in\tilde{\mathcal{S}}_1\setminus\{B(y)\}} p_{\tilde{y}|\tilde{y}} + e^{-\epsilon} \sum_{\tilde{y}\in\tilde{\mathcal{S}}_2} p_{\tilde{y}|\tilde{y}}. \end{aligned} \tag{C.2}$$

The equations hold for both Equation (C.1) and Equation (C.2) if and only if $p_{\tilde{y}|y} = e^\epsilon \cdot p_{\tilde{y}|y'}$, for all $y \in \mathcal{S}_1, \tilde{y} \in \tilde{\mathcal{S}} \setminus B(y)$ and for all $y \in \mathcal{S}_2, \tilde{y} \in \tilde{\mathcal{S}} \setminus (B(y) \cup \mathcal{T})$. Equation (C.1) and Equation (C.2) together imply we have a larger feasible region for the following optimization problem compared with Equation (4.1):

$$\begin{aligned} \max \quad & \sum_{y\in\mathcal{S},\tilde{y}\in\tilde{\mathcal{S}}} w_{y\tilde{y}} p_{\tilde{y}|y} \\ \text{s.t.} \quad & \begin{cases} \sum_{\tilde{y}\in B(y)} p_{\tilde{y}|y} + e^{-\epsilon} \sum_{\tilde{y}\in\tilde{\mathcal{S}}\setminus(\{B(y)\}\cup\mathcal{T})} p_{\tilde{y}|\tilde{y}} + \sum_{\tilde{y}\in\mathcal{T}} p_{\tilde{y}|y} \leq 1, \forall y \in \mathcal{S}_2, \\ \sum_{\tilde{y}\in B(y)} p_{\tilde{y}|y} + e^{-\epsilon} \sum_{\tilde{y}\in\tilde{\mathcal{S}}\setminus\{B(y)\}} p_{\tilde{y}|\tilde{y}} + e^{-\epsilon} \sum_{\tilde{y}\in\tilde{\mathcal{S}}_2} p_{\tilde{y}|\tilde{y}} \leq 1, \forall y \in \mathcal{S}_1, \\ p_{\tilde{y}|y} = \frac{1}{|\tilde{\mathcal{S}}|}, \forall y \in \mathcal{S}_2, \forall \tilde{y} \in \mathcal{T}, \\ p_{\tilde{y}|y} \geq 0, \forall \tilde{y} \in \tilde{\mathcal{S}}, \ \forall y \in \mathcal{S}. \end{cases} \end{aligned} \tag{C.3}$$

We know that the optimal solution of linear programming is typically found at the boundary. That is, The equations hold for both Equation (C.1) and Equation (C.2), which implies we have the same optimal solution for both Equation (4.1) and Equation (C.3). To find the optimal solution of Equation (C.3) is to solve the following linear system

$$\begin{cases} \sum_{\tilde{y}\in B(y)} p_{\tilde{y}|y} + e^{-\epsilon} \sum_{\tilde{y}\in\tilde{\mathcal{S}}\setminus(\{B(y)\}\cup\mathcal{T})} p_{\tilde{y}|\tilde{y}} + |\mathcal{T}| \cdot \frac{1}{|\tilde{\mathcal{S}}|} = 1, \ \forall y \in \mathcal{S}_2, \\ \sum_{\tilde{y}\in B(y)} p_{\tilde{y}|y} + e^{-\epsilon} \sum_{\tilde{y}\in\tilde{\mathcal{S}}_1\setminus\{B(y)\}} p_{\tilde{y}|\tilde{y}} + e^{-\epsilon} \sum_{\tilde{y}\in\tilde{\mathcal{S}}_2} p_{\tilde{y}|\tilde{y}} = 1, \ \forall y \in \mathcal{S}_1. \end{cases}$$

For any $y \in \mathcal{S}$, we have $p_{\tilde{y}|y} = p_{\tilde{y}|\tilde{y}}, \forall \tilde{y} \in B(y)$, is a constant and

$$\begin{cases} |B(y)| \cdot p_{\tilde{y}|\tilde{y}} + e^{-\epsilon} \sum_{\hat{y}\in\tilde{\mathcal{S}}\setminus(B(y)\cup\mathcal{T})} p_{\hat{y}|\hat{y}} + |\mathcal{T}| \cdot \frac{1}{|\tilde{\mathcal{S}}|} = 1, \ \forall y \in \mathcal{S}_2, \forall \tilde{y} \in B(y), \\ |B(y)| \cdot p_{\tilde{y}|\tilde{y}} + e^{-\epsilon} \sum_{\hat{y}\in\tilde{\mathcal{S}}_1\setminus B(y)} p_{\hat{y}|\hat{y}} + e^{-\epsilon} \sum_{\hat{y}\in\tilde{\mathcal{S}}_2} p_{\hat{y}|\hat{y}} = 1, \ \forall y \in \mathcal{S}_1, \forall \tilde{y} \in B(y). \end{cases}$$

Note that $p_{\tilde{y}|\tilde{y}}$ is symmetric about $\tilde{y}$ on $\tilde{\mathcal{S}}_1$ and $\tilde{\mathcal{S}}_2$, respectively. Set $p_{\tilde{y}|\tilde{y}} = e^\epsilon \cdot \beta, \forall \tilde{y} \in \tilde{\mathcal{S}}_1$, and $p_{\tilde{y}|\tilde{y}} = e^\epsilon \cdot \gamma, \forall \tilde{y} \in \tilde{\mathcal{S}}_2$, we have

$$\begin{cases} (e^\epsilon |B(y)| + |\tilde{\mathcal{S}}_1| - |B(y)|) \cdot \beta + |\tilde{\mathcal{S}}_2| \cdot \gamma = 1, \ \forall y \in \mathcal{S}_1, \\ (|\tilde{\mathcal{S}}_1| - |\mathcal{T}|) \cdot \beta + (e^\epsilon |B(y)| + |\tilde{\mathcal{S}}_2| - |B(y)|) \cdot \gamma = 1 - \frac{|\mathcal{T}|}{|\tilde{\mathcal{S}}|}, \ \forall y \in \mathcal{S}_2. \end{cases}$$

solving Equation (3.1) yields the BlockRR algorithm. □

D Proof of Lemma 4.2

Proof. Note that

$$\begin{aligned} \kappa =& [(e^\epsilon - 1)n_1 + |\tilde{\mathcal{S}}_1|][(e^\epsilon - 1)n_2 + |\tilde{\mathcal{S}}_2|] - |\tilde{\mathcal{S}}_1||\tilde{\mathcal{S}}_2| + l|\tilde{\mathcal{S}}_2| \\ =& (e^\epsilon - 1)^2 n_1 n_2 + (e^\epsilon - 1)n_1|\tilde{\mathcal{S}}_2| + (e^\epsilon - 1)n_2|\tilde{\mathcal{S}}_1| + l|\tilde{\mathcal{S}}_2| \\ =& (e^\epsilon - 1)^2 n_1 n_2 + (e^\epsilon - 1)(n_1 - n_2)|\tilde{\mathcal{S}}_2| + |\tilde{\mathcal{S}}|\beta_1 := \kappa_1 + |\tilde{\mathcal{S}}|\beta_1, \end{aligned}$$

where $\kappa_1 = (e^\epsilon - 1)^2 n_1 n_2 + (e^\epsilon - 1)(n_1 - n_2)|\tilde{\mathcal{S}}_2|$. Then $\beta_1 = \frac{\kappa - \kappa_1}{|\tilde{\mathcal{S}}|}$ and

$$\beta = \frac{\beta_1}{\kappa} = \frac{\kappa - \kappa_1}{|\tilde{\mathcal{S}}|\kappa} = \frac{1}{|\tilde{\mathcal{S}}|} - \frac{\kappa_1}{|\tilde{\mathcal{S}}|\kappa}.$$

Since κ increases with l, β is monotonically increasing in l. From the expression for γ in the first equation of Equation (3.1),

$$\gamma = \frac{1}{|\tilde{\mathcal{S}}_2|}(1 - (e^\epsilon |B(y)| + |\tilde{\mathcal{S}}_1| - |B(y)|) \cdot \beta),$$

it follows that γ is monotonically decreasing in l. Consequently, β attains its minimum and γ gets its maximum at $l = 0$, and $\max_l \gamma = \min_l \beta$. □

E Proof of Privacy

In the following, we recall the parallel composition theorem in differential privacy.

Let $\{U_i\}_{i=1}^k$ be a partition of a domain U, i.e., the U_i are pairwise disjoint and $\cup_{i=1}^k U_i = U$. For any dataset $D \subseteq U$, the sets $\{D \cap U_i\}_{i=1}^k$ constitute a partition of D. Then, the parallel composition theorem can be stated as follows:

Lemma E.1 (Parallel Composition (McSherry, 2009))**.** *Suppose that $\{M_i : U \to R_i\}_{i=1}^k$ is a family of randomized algorithms, each of which satisfies ϵ-DP, and that $\{U_i\}_{i=1}^k$ is a partition of the domain U. Then the combined mechanism $M(D) := \{M_i(D \cap U_i)\}_{i=1}^k$ still satisfies ϵ-DP.*

Proof. For any adjacent datasets $D, D' \subseteq U$, let $D \cap U_i = D_i$ and $D' \cap U_i = D'_i$. For any $r_i \in R_i$, let $r = [r_1\ r_2\ \dots\ r_k]^\top$. Then

$$\Pr[M(D) = r] = \prod_{i=1}^k \Pr[M_i(D_i) = r_i].$$

In particular, since D and D' are adjacent datasets, there is exactly one pair in $\{(D_i, D'_i)\}_{i=1}^k$ that are adjacent (i.e., differ in exactly one record), while all the other pairs are equal. Hence,

$$\Pr[M(D) = r] = \prod_{i=1}^k \Pr[M_i(D_i) = r_i] \le e^\epsilon \prod_{i=1}^k \Pr[M_i(D'_i) = r_i] = e^\epsilon \cdot \Pr[M(D') = r].$$

Therefore, the parallel composition satisfies ϵ-DP. □

Proof of Theorem 4.4. We only need to prove that for any $y, y' \in \mathcal{S}$ and any $\tilde{y} \in \tilde{\mathcal{S}}$, the following holds:

$$\Pr[\tilde{Y} = \tilde{y} \mid Y = y] \leq e^{\epsilon} \Pr[\tilde{Y} = \tilde{y} \mid Y = y']. \tag{E.1}$$

For any $\tilde{y} \in \tilde{\mathcal{S}}_1$ and any $y \in \mathcal{S}$,

$$\Pr[\tilde{Y} = \tilde{y} \mid Y = y] = \begin{cases} e^{\epsilon} \cdot \beta, & \text{for } y \in \mathcal{S}_1, \tilde{y} \in B(y), \\ \beta, & \text{for } y \in \mathcal{S}_1, \tilde{y} \in \tilde{\mathcal{S}}_1 \setminus B(y) \text{ or } y \in \mathcal{S}_2, \tilde{y} \in \tilde{\mathcal{S}}_1 \setminus \mathcal{T}, \\ \dfrac{1}{|\tilde{\mathcal{S}}|}, & \text{for } y \in \mathcal{S}_2, \tilde{y} \in \mathcal{T}, \end{cases}$$

where $\beta = \dfrac{(e^{\epsilon}-1)|B(y)| + \frac{l|\tilde{\mathcal{S}}_2|}{|\tilde{\mathcal{S}}|}}{(e^{\epsilon}|B(y)| + |\tilde{\mathcal{S}}_1| - |B(y)|)(e^{\epsilon}|B(y)| + |\tilde{\mathcal{S}}_2| - |B(y)|) - (|\tilde{\mathcal{S}}_1| - l)|\tilde{\mathcal{S}}_2|}$.

We only need to verify that $\beta \leq \dfrac{1}{|\tilde{\mathcal{S}}|} \leq e^{\epsilon}\beta$. We have

$$\begin{aligned}
&\frac{1}{|\tilde{\mathcal{S}}|} - \beta \geq 0 \\
\Leftrightarrow\ &\frac{1}{|\tilde{\mathcal{S}}|} - \frac{(e^{\epsilon}-1)|B(y)| + \frac{l|\tilde{\mathcal{S}}_2|}{|\tilde{\mathcal{S}}|}}{(e^{\epsilon}|B(y)| + |\tilde{\mathcal{S}}_1| - |B(y)|)(e^{\epsilon}|B(y)| + |\tilde{\mathcal{S}}_2| - |B(y)|) - (|\tilde{\mathcal{S}}_1| - l)|\tilde{\mathcal{S}}_2|} \geq 0 \\
\Leftrightarrow\ &(e^{\epsilon}|B(y)| + |\tilde{\mathcal{S}}_1| - |B(y)|)(e^{\epsilon}|B(y)| + |\tilde{\mathcal{S}}_2| - |B(y)|) - (|\tilde{\mathcal{S}}_1| - l)|\tilde{\mathcal{S}}_2| \geq \\
&(e^{\epsilon}|B(y)| - |B(y)|)|\tilde{\mathcal{S}}| - l|\tilde{\mathcal{S}}_2| \\
\Leftrightarrow\ &(e^{\epsilon}|B(y)| - |B(y)|)^2 \geq 0.
\end{aligned}$$

Next, we check the other inequality:

$$\begin{aligned}
&e^{\epsilon}\beta - \frac{1}{|\tilde{\mathcal{S}}|} \geq 0 \\
\Leftrightarrow\ &\frac{e^{\epsilon}\left((e^{\epsilon}-1)|B(y)| + \frac{l|\tilde{\mathcal{S}}_2|}{|\tilde{\mathcal{S}}|}\right)}{(e^{\epsilon}|B(y)| + |\tilde{\mathcal{S}}_1| - |B(y)|)(e^{\epsilon}|B(y)| + |\tilde{\mathcal{S}}_2| - |B(y)|) - (|\tilde{\mathcal{S}}_1| - l)|\tilde{\mathcal{S}}_2|} - \frac{1}{|\tilde{\mathcal{S}}|} \geq 0 \\
\Leftrightarrow\ &e^{\epsilon}\left((e^{\epsilon}|B(y)| - |B(y)|) + \frac{l|\tilde{\mathcal{S}}_2|}{|\tilde{\mathcal{S}}|}\right)|\tilde{\mathcal{S}}| \geq \\
&(e^{\epsilon}|B(y)| + |\tilde{\mathcal{S}}_1| - |B(y)|)(e^{\epsilon}|B(y)| + |\tilde{\mathcal{S}}_2| - |B(y)|) - (|\tilde{\mathcal{S}}_1| - l)|\tilde{\mathcal{S}}_2| \\
\Leftrightarrow\ &(e^{\epsilon}-1)^2(|\tilde{\mathcal{S}}| - |B(y)|)|B(y)| + (e^{\epsilon}-1)l|\tilde{\mathcal{S}}_2| \geq 0,
\end{aligned}$$

where the last inequality is from $\epsilon > 0$ and the set $\tilde{\mathcal{S}}$ contains $B(y)$.

For any $\tilde{y} \in \tilde{\mathcal{S}}_2$ (assume $\tilde{\mathcal{S}}_2 \neq \emptyset$) and $y \in \mathcal{S}$,

$$\Pr[\tilde{Y} = \tilde{y} \mid Y = y] = \begin{cases} e^{\epsilon} \cdot \gamma, & \text{for } y \in \mathcal{S}_2, \tilde{y} \in B(y), \\ \gamma, & \text{otherwise.} \end{cases}$$

where

$$\gamma = \frac{(e^{\epsilon}|B(y)| + l - |B(y)|) - \frac{l}{|\tilde{\mathcal{S}}|}(e^{\epsilon}|B(y)| + |\tilde{\mathcal{S}}_1| - |B(y)|)}{(e^{\epsilon}|B(y)| + |\tilde{\mathcal{S}}_1| - |B(y)|) \cdot (e^{\epsilon}|B(y)| + |\tilde{\mathcal{S}}_2| - |B(y)|) - (|\tilde{\mathcal{S}}_1| - l)|\tilde{\mathcal{S}}_2|},$$

which clearly satisfies Equation (E.1). Thus, we conclude that BlockRR$_\epsilon$ satisfies ϵ-Label DP. □

Proof of Theorem 4.5. Both Algorithm 1 and Algorithm 3 satisfy ϵ-label differential privacy. By the parallel composition principle stated in Theorem E.1, Algorithm 2, which composes these two mechanisms, also satisfies ϵ-label differential privacy. □

F Proof of Theorem 4.7

Proposition F.1 (Proposition 1 in (Zhao et al., 2025)). *For any practical classifier $c : \mathcal{X} \to \mathcal{Y}$, the excess risk is*

$$R - R^* = \int \left(\eta^*(\boldsymbol{x}) - \mathbb{E}\left[\eta_{c(\boldsymbol{x})}(\boldsymbol{x})\right]\right) f(\boldsymbol{x}) d\boldsymbol{x},$$

in which R^* is the Bayes risk, $\eta_{c(\boldsymbol{x})}(\boldsymbol{x})$ is just $\eta_j(\boldsymbol{x})$ with $j = c(\boldsymbol{x})$, the expectation is about $c(\cdot)$ due to the randomness of the training dataset, and f is the probability density function (pdf) of random feature vector $\mathbf{x}$*,*

Proof of Theorem 4.7. Let $\eta^*(\boldsymbol{x}) = \max_y \eta_y(\boldsymbol{x}) = \eta_{c^*(\boldsymbol{x})}(\boldsymbol{x})$ and $\tilde{\eta}_{\tilde{c}(\boldsymbol{x})} = \Pr(\tilde{Y} = \tilde{c}(\boldsymbol{x}) \mid \mathbf{x} = \boldsymbol{x})$. Proposition F.1 yields:

$$\tilde{R} - R^* = \int \left(\eta^*(\boldsymbol{x}) - \mathbb{E}\left[\eta_{\tilde{c}(\boldsymbol{x})}(\boldsymbol{x})\right]\right) f(\boldsymbol{x})\, d\boldsymbol{x}.$$

It suffices to derive a pointwise upper bound on $\eta^*(\boldsymbol{x}) - \eta_{\tilde{c}(\boldsymbol{x})}(\boldsymbol{x})$ for any fixed $\boldsymbol{x}$.

By definition, $\tilde{c}(\boldsymbol{x}) = \arg\max_{\tilde{y} \in \tilde{\mathcal{S}}} g_{\tilde{y}}(\boldsymbol{x})$, which implies

$$g_{\tilde{c}(\boldsymbol{x})}(\boldsymbol{x}) \geq g_{c^*(\boldsymbol{x})}(\boldsymbol{x}).$$

Invoking the definition of $\Delta(\boldsymbol{x})$, for all $\tilde{y} \in \tilde{\mathcal{S}}$ we have

$$|g_{\tilde{y}}(\boldsymbol{x}) - s_{\tilde{y}} \tilde{\eta}_{\tilde{y}}(\boldsymbol{x})| \leq \Delta(\boldsymbol{x}).$$

Combining the optimality condition and the approximation bound yields

$$s_{\tilde{c}(\boldsymbol{x})} \tilde{\eta}_{\tilde{c}(\boldsymbol{x})}(\boldsymbol{x}) + \Delta(\boldsymbol{x}) \geq g_{\tilde{c}(\boldsymbol{x})}(\boldsymbol{x}) \geq g_{c^*(\boldsymbol{x})}(\boldsymbol{x}) \geq s_{c^*(\boldsymbol{x})} \tilde{\eta}_{c^*(\boldsymbol{x})}(\boldsymbol{x}) - \Delta(\boldsymbol{x}).$$

Rearranging terms gives the fundamental gap inequality:

$$s_{c^*(\boldsymbol{x})} \tilde{\eta}_{c^*(\boldsymbol{x})}(\boldsymbol{x}) - s_{\tilde{c}(\boldsymbol{x})} \tilde{\eta}_{\tilde{c}(\boldsymbol{x})}(\boldsymbol{x}) \leq 2\Delta(\boldsymbol{x}). \tag{F.1}$$

By the law of total probability, we expand the perturbed posterior as follows:

$$\Pr(\tilde{Y} = j \mid \mathbf{x} = \boldsymbol{x}) = \sum_{\ell} \Pr(\tilde{Y} = j,\, Y = \ell \mid \mathbf{x} = \boldsymbol{x}) = \sum_{\ell} \Pr(\tilde{Y} = j \mid Y = \ell)\, \Pr(Y = \ell \mid \mathbf{x} = \boldsymbol{x}).$$

We analyze three cases separately: for $j \in \mathcal{T}$,

$$\begin{aligned}
\tilde{\eta}_j(\boldsymbol{x}) &= \Pr(Y = j \mid \mathbf{x} = \boldsymbol{x}) e^{\epsilon} \beta + \sum_{\ell \in \mathcal{S}_1 \setminus \{j\}} \Pr(Y = \ell \mid \mathbf{x} = \boldsymbol{x}) \beta + \frac{1}{|\tilde{\mathcal{S}}|} \sum_{\ell \in \mathcal{S}_2} \Pr(Y = \ell \mid \mathbf{x} = \boldsymbol{x}) \\
&= \Pr(Y = j \mid \mathbf{x} = \boldsymbol{x}) e^{\epsilon} \beta + \sum_{\ell \neq j} \Pr(Y = \ell \mid \mathbf{x} = \boldsymbol{x}) \beta + \left(\frac{1}{|\tilde{\mathcal{S}}|} - \beta\right) \sum_{\ell \in \mathcal{S}_2} \Pr(Y = \ell \mid \mathbf{x} = \boldsymbol{x}) \\
&= \Pr(Y = j \mid \mathbf{x} = \boldsymbol{x})(e^{\epsilon} - 1)\beta + \beta + \left(\frac{1}{|\tilde{\mathcal{S}}|} - \beta\right) \sum_{\ell \in \mathcal{S}_2} \Pr(Y = \ell \mid \mathbf{x} = \boldsymbol{x}) \\
&= (e^{\epsilon} - 1)\beta \cdot \eta_j(\boldsymbol{x}) + \beta + \left(\frac{1}{|\tilde{\mathcal{S}}|} - \beta\right) \sum_{\ell \in \mathcal{S}_2} \eta_{\ell}(\boldsymbol{x}),
\end{aligned}$$

for $j \in \tilde{\mathcal{S}}_1 \setminus \mathcal{T}$,

$$\tilde{\eta}_j(\boldsymbol{x}) = \Pr(Y = j \mid \mathbf{x} = \boldsymbol{x}) e^{\epsilon} \beta + \sum_{\ell \in \mathcal{S} \setminus \{j\}} \Pr(Y = \ell \mid \mathbf{x} = \boldsymbol{x}) \beta = (e^{\epsilon} - 1)\beta \cdot \eta_j(\boldsymbol{x}) + \beta.$$

for $j \in \tilde{\mathcal{S}}_2$,

$$\tilde{\eta}_j(\boldsymbol{x}) = \Pr(Y = j \mid \mathbf{x} = \boldsymbol{x}) e^{\epsilon} \gamma + \sum_{\ell \in \mathcal{S} \setminus \{j\}} \Pr(Y = \ell \mid \mathbf{x} = \boldsymbol{x}) \gamma = (e^{\epsilon} - 1)\gamma \cdot \eta_j(\boldsymbol{x}) + \gamma.$$

Based on the closed-form expression of $\tilde{\eta}_j(\boldsymbol{x})$ and Equation (F.1), we split the subsequent analysis into three disjoint cases over the membership of $\tilde{c}(\boldsymbol{x})$ and $c^*(\boldsymbol{x})$.

Case 1: $\tilde{c}(\boldsymbol{x}) \in \mathcal{T},\ c^*(\boldsymbol{x}) \in \mathcal{T}$ **or** $\tilde{c}(\boldsymbol{x}), c^*(\boldsymbol{x}) \in \tilde{\mathcal{S}} \setminus \mathcal{T}$**.** Under this configuration,

$$\eta_{c^*(\boldsymbol{x})}(\boldsymbol{x}) - \eta_{\tilde{c}(\boldsymbol{x})}(\boldsymbol{x}) = \frac{\tilde{\eta}_{c^*(\boldsymbol{x})}(\boldsymbol{x}) - \tilde{\eta}_{\tilde{c}(\boldsymbol{x})}(\boldsymbol{x})}{(e^\epsilon - 1)\beta} \leq \frac{2\Delta(\boldsymbol{x})}{(e^\epsilon - 1)\beta} = 2\Delta'(\boldsymbol{x}).$$

Case 2: $\tilde{c}(\boldsymbol{x}) \in \mathcal{T},\ c^*(\boldsymbol{x}) \in \tilde{\mathcal{S}} \setminus \mathcal{T}$**.**

$$\begin{aligned}
&s_{c^*(\boldsymbol{x})}\tilde{\eta}_{c^*(\boldsymbol{x})}(\boldsymbol{x}) - s_{\tilde{c}(\boldsymbol{x})}\tilde{\eta}_{\tilde{c}(\boldsymbol{x})}(\boldsymbol{x}) \\
&= (e^\epsilon - 1)\beta \cdot \eta_{c^*(\boldsymbol{x})}(\boldsymbol{x}) + \beta + \left(\frac{1}{|\tilde{\mathcal{S}}|} - \beta\right) \sum_{\ell \in \mathcal{S}_2} \eta_\ell(\boldsymbol{x}) - \left((e^\epsilon - 1)\beta \cdot \eta_{\tilde{c}(\boldsymbol{x})}(\boldsymbol{x}) + \beta\right) \\
&= (e^\epsilon - 1)\beta \left(\eta_{c^*(\boldsymbol{x})}(\boldsymbol{x}) - \eta_{\tilde{c}(\boldsymbol{x})}(\boldsymbol{x})\right) + \left(\frac{1}{|\tilde{\mathcal{S}}|} - \beta\right) \sum_{\ell \in \mathcal{S}_2} \eta_\ell(\boldsymbol{x}) \leq 2\Delta(\boldsymbol{x}).
\end{aligned}$$

Rearranging the above inequality immediately yields

$$\eta_{c^*(\boldsymbol{x})}(\boldsymbol{x}) - \eta_{\tilde{c}(\boldsymbol{x})}(\boldsymbol{x}) \leq 2\Delta'(\boldsymbol{x}) - A(\boldsymbol{x}).$$

Case 3: $\tilde{c}(\boldsymbol{x}) \in \tilde{\mathcal{S}} \setminus \mathcal{T},\ c^*(\boldsymbol{x}) \in \mathcal{T}$**.** Similar as Case 2, we have

$$\eta_{c^*(\boldsymbol{x})}(\boldsymbol{x}) - \eta_{\tilde{c}(\boldsymbol{x})}(\boldsymbol{x}) \leq 2\Delta'(\boldsymbol{x}) + A(\boldsymbol{x}).$$

In summary,

$$\eta^*(\boldsymbol{x}) - \eta_{\tilde{c}(\boldsymbol{x})}(\boldsymbol{x}) \leq 2\Delta'(\boldsymbol{x}) + A(\boldsymbol{x}).$$

Misclassification occurs if and only if the Bayes margin $\eta^*(\boldsymbol{x}) - \eta_s(\boldsymbol{x})$ does not exceed the threshold derived above, where

$$\eta_s(\boldsymbol{x}) = \max_{j \neq c^*(\boldsymbol{x})} \eta_j(\boldsymbol{x})$$

is defined as the second-largest posterior probability. Consequently, we obtain the pointwise indicator bound

$$\eta^*(\boldsymbol{x}) - \eta_{\tilde{c}(\boldsymbol{x})}(\boldsymbol{x}) \leq (2\Delta'(\boldsymbol{x}) + A(\boldsymbol{x}))\, \mathbb{I}\left(\eta^*(\boldsymbol{x}) - \eta_s(\boldsymbol{x})\right) \leq 2\Delta'(\boldsymbol{x}) + A(\boldsymbol{x}).$$

Integrating both sides against the feature density $f(\boldsymbol{x})$ recovers the target upper bound for excess risk, which completes the proof. □

G Extra Experiments

This section proceeds as follows: we first outline the experimental details in Subsection G.1, followed by ablation analyses in Subsection G.2.

G.1 Experimental Details

Experiments are conducted over privacy budgets $\epsilon \in \{0.6, 0.8, 1.0, 2.0, 3.0, 4.0, 6.0, 8.0, \infty\}$, where $\epsilon = \infty$ represents the non-private baseline (training on original labels).

For RRWithPrior and BlockRR, the label prior is estimated from the training data via a cross-validation (CV)-like scheme: we reserve 1% of the training set for prior estimation (Algorithm 2) and use the remaining 99% for model training. Since standard RR does not require priors, it is trained on the entire training set, ensuring a fair comparison where each method uses its maximal available data.

All experiments use a ResNet-18 architecture (He et al., 2016), trained from scratch without pre-trained weights. In optimization and training, All models are trained from scratch for 160 epochs with a batch size of 512 for $\epsilon > 0.6$ and for 110 epochs when $\epsilon = 0.6$. Training uses SGD with Nesterov momentum (0.9) (Bottou, 2010) and a cross-entropy loss (Hinton et al., 1986). The learning rate warms up linearly from 0 to 0.4 over the first 15% of epochs, then decays by a factor of 10 at 30%, 60%, and 90% of total training. Weight decay (10^{-4}) is applied for regularization. To improve generalization, we employ Mixup (Zhang et al., 2017) with $\alpha = 8.0$.

Table 6: Ablation results of the parameter l under different global privacy budgets ϵ (test accuracy %).

l	0	1	2	3	4	5	6
$\epsilon = 0.6$	42.52 ± 1.88	42.78 ± 2.47	$\mathbf{43.48 \pm 1.98}$	41.52 ± 2.46	43.22 ± 1.51	42.95 ± 1.48	42.34 ± 1.62
$\epsilon = 0.8$	48.88 ± 1.94	48.79 ± 1.98	49.91 ± 1.90	49.90 ± 2.93	49.44 ± 2.31	48.80 ± 1.80	$\mathbf{51.01 \pm 1.29}$
$\epsilon = 1.0$	55.58 ± 1.91	55.65 ± 1.25	57.10 ± 1.91	56.40 ± 1.59	56.04 ± 1.71	$\mathbf{58.26 \pm 1.67}$	56.92 ± 2.85
$\epsilon = 2.0$	79.03 ± 0.79	78.61 ± 0.67	78.76 ± 0.78	78.77 ± 0.92	78.54 ± 0.74	78.72 ± 1.05	$\mathbf{79.14 \pm 0.44}$
$\epsilon = 3.0$	$\mathbf{87.18 \pm 0.30}$	87.05 ± 0.33	86.99 ± 0.48	86.78 ± 0.42	86.59 ± 0.39	86.09 ± 0.44	85.94 ± 0.35
$\epsilon = 4.0$	$\mathbf{90.36 \pm 0.32}$	90.09 ± 0.42	89.99 ± 0.27	89.85 ± 0.60	89.62 ± 0.38	88.96 ± 0.35	88.78 ± 0.53

G.2 Ablation Study

In the ablation study, we aim to analyze the impact of the hyperparameters l and σ on the performance of our proposed BlockRR mechanism. The following experiments are all conducted on Cifar-10_2.

G.2.1 Effectiveness of l

As demonstrated in Section 5, the performance for CIFAR-10_2 dataset of our proposed mechanism is almost the same as the RR mechanism at $\epsilon > 4.0$, since most of the partitions become $\mathcal{S}_1 = \mathcal{S}$ and $\mathcal{S}_2 = \emptyset$. Consequently, the hyperparameter l only affect elements in $\mathcal{S}_2$, its specific value becomes irrelevant in this regime.

In the following, we conduct experiments across different privacy budgets $\epsilon \in \{0.6, 0.8, 1.0, 2.0, 3.0, 4.0\}$ and vary $l \in \{0, 1, 2, 3, 4, 5, 6\}$. The resulting test accuracies are reported in Table 6.

As shown in Table 6, test accuracy remains stable for high privacy budgets ($\epsilon \leq 1.0$), fluctuating within 2% as l varies from 0 to 6. For instance, at $\epsilon = 0.6$, accuracy ranges from 41.52% ($l = 3$) to 43.44% ($l = 2$). From the viewpoint of the transition probability matrix, according to the definition of l in our proposed method, smaller values of l make the transition probabilities closer to those of RR (and when $l = 0$ the mechanism reduces exactly to RR), while larger values of l move the mechanism towards RRWithPrior. The fact that the optimal l is not at the boundaries (e.g., $l = 2$ for $\epsilon = 0.6$) indicates that the best performance is achieved by an intermediate configuration, blending properties of both RR and RRWithPrior.

When ϵ increases to 3.0, overall accuracy shows a mild negative trend with l, dropping from 87.18% ($l = 0$) to 85.94% ($l = 6$), a decrease of 1.2%. This aligns with the results in Table 3 for $\epsilon = 3.0$: under higher privacy budgets, better performance is achieved when the transition matrix is closer to the standard RR mechanism, which itself is a special case of our proposed framework when $l = 0$. We have a similar result for $\epsilon = 4.0$.

G.2.2 Sensitivity to σ

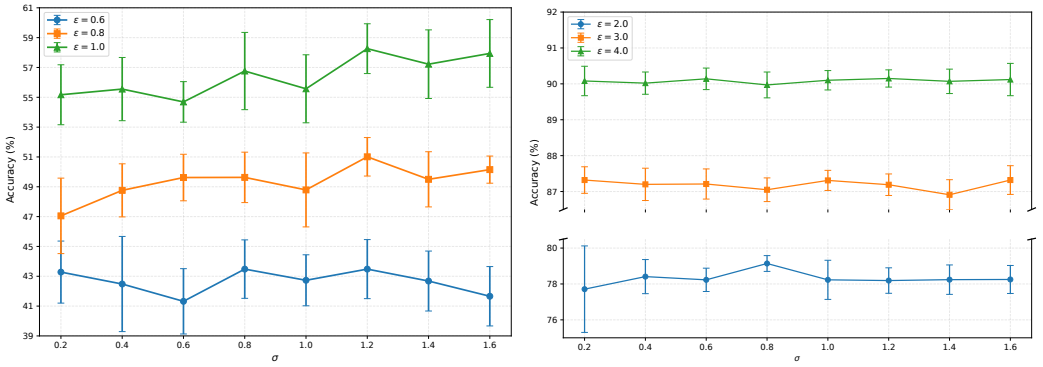

Figure 2: Left: Accuracy (%) versus σ with $\epsilon \in \{0.6, 0.8, 1.0\}$. Right: Accuracy (%) versus σ with $\epsilon \in \{2.0, 3.0, 4.0\}$.

We study the sensitivity of our method to the hyperparameter σ by varying $\sigma \in [0.2, 1.6]$ and report the overall accuracy under different privacy budgets ϵ. Overall, the performance is stable across a wide range of σ, indicating that our method is not overly sensitive to this hyperparameter.

In the strict privacy regime (Figure 2 (Left), $\epsilon \leq 1.0$), accuracy exhibits mild fluctuations as σ changes. For all three budgets ($\epsilon = 0.6, 0.8, 1.0$), very small σ values tend to yield slightly lower accuracy, while increasing σ to a moderate range generally improves performance and the best results are typically achieved around $\sigma = 1.2$.

In the moderate privacy budget regime (Figure 2 (Right), $\epsilon \geq 2.0$), the curves become even flatter. For $\epsilon = 3.0$ and $\epsilon = 4.0$, accuracy remains almost constant across the entire σ range, with variations that are comparable to the error bars. These results confirm that once the privacy budget is less restrictive, the choice of σ has minimal impact on performance.