# Appendix

The Appendix is organized as follows:

- Appendix A provides the instance-level DP definition, corresponding algorithms, the privacy analysis and robustness certification for FL.
- Appendix B provides the DPFL algorithms on both user and instance levels, and the proofs for corresponding privacy guarantees.
- Appendix C specifies our threat models.
- Appendix D provides more details on experimental setups for training and evaluation, certified attack cost of user-level DPFL under different $\epsilon$, robustness evaluation of instance-level DPFL, the addition experimental results on certified accuracy with confidence level, and robustness evaluation on 10-class classification.
- Appendix E provides the proofs for the certified robustness related analysis, including Lemma 1, Theorem 1, 2, 3, 5 and Corollary 1.

## A Instance-level Privacy and Certified Robustness for FL

### A.1 Instance-level Privacy

In this section, we introduce the instance-level DP definition, corresponding algorithm, and the privacy analysis for FL. When DP is used to protect the privacy of individual instance, the trained stochastic FL model should not differ much if one instance is modified. Hence, the adjacent datasets in instance-level DP is defined as those differing by one instance.

**Definition 3** (Instance-level $(\epsilon, \delta)$-DP)**.** *Let $D$ be the dataset that is the union of local training examples from all users. Then, $D$ and $D'$ are adjacent if they differ by one instance. The mechanism $\mathcal{M}$ is instance-level $(\epsilon, \delta)$-DP if it meets Definition 1 with $D$ and $D'$ as adjacent datasets.*

Dopamine [44] provides the first instance-level privacy guarantee under FedSGD [45]. However, it has two limitations. First, its privacy bound is loose. Although FedSGD performs both user and batch sampling during training, Dopamine ignores the privacy gain provided by random user sampling. In this section, we improve the privacy guarantee under FedSGD with privacy amplification via user sampling [8, 2]. This improvement leads to algorithm `InsDP-FedSGD`, to achieve tighter privacy analysis. We defer the algorithm (Algorithm 2) and its privacy guarantee to Appendix B.

Besides the loose privacy bound, Dopamine [44] only allows users to perform *one step of DP-SGD* [2] at each FL round. This restriction limits the efficiency of the algorithm and increases the communication overhead. In practice, users in FL are typically allowed to update their local models for many steps before submitting updates to reduce the communication cost. To solve this problem, we further improve `InsDP-FedSGD` to support multiple local steps at each round. Specifically, we propose a novel instance-level DPFL algorithm `InsDP-FedAvg` (Algorithm 3 in Appendix B) allowing users to train multiple local SGD steps before submitting the updates. In `InsDP-FedAvg`, each user $i$ performs local DP-SGD so that the local training mechanism $\mathcal{M}^i$ satisfies instance-level DP. Then, the server aggregates the updates. We prove that the global mechanism $\mathcal{M}$ preserves instance-level DP using DP parallel composition theorem [21] and moment accountant [2].

Algorithm 3 formally presents the `InsDP-FedAvg` algorithm and the calculation of its privacy budget $\epsilon$. Specifically, at first, local privacy cost $\epsilon_0^i$ is initialized as 0 before FL training. At round $t$, if user $i$ is not selected, its local privacy cost is kept unchanged $\epsilon_t^i \leftarrow \epsilon_{t-1}^i$. Otherwise user $i$ updates local model by running DP-SGD for $V$ local steps with batch sampling probability $p$, noise level $\sigma$ and clipping threshold $S$, and $\epsilon_t^i$ is accumulated upon $\epsilon_{t-1}^i$ via its local moment accountant. Next, the server aggregates the updates from selected users, and leverage $\{\epsilon_t^i\}_{i\in[N]}$ and the parallel composition in Theorem 4 to calculate the global privacy cost $\epsilon_t$. After $T$ rounds, the mechanism $\mathcal{M}$ that outputs the FL global model in Algorithm 3 is instance-level $(\epsilon_T, \delta)$-DP.

**Theorem 4** (`InsDP-FedAvg` Privacy Guarantee)**.** *In Algorithm 3, at round $t$, suppose local mechanism $\mathcal{M}^i$ satisfies $(\epsilon_t^i, \delta)$-DP, then the global mechanism $\mathcal{M}$ satisfies $\left(\max_{i\in[N]} \epsilon_t^i, \delta\right)$-DP.*

The idea behind Theorem 4 is that when $D'$ and $D$ differ one instance, the modified instance only fall into one local dataset, thus parallel composition theorem [21] can be applied. Then the privacy

guarantee considers the worst-case, which is provided by taking the maximum local privacy cost across all the users. The detailed proof is omitted to Appendix B.

### A.2 Certified Robustness of Instance-level DPFL against Poisoning Attacks

**Threat Model.** We consider poisoning attacks where there are $k$ poisoned instances. These instances could be controlled by the same or multiple adversarial users. Our robustness certification is agnostic to the attack methods as long as the number of poisoned instances is constrained.

According to the group DP property (Lemma 1) and the post-processing property for FL model with instance-level $(\epsilon, \delta)$-DP, we prove that our robust certification results proposed for user-level DP are also applicable to instance-level DP. Below is the formal theorem (proof is omitted to Appendix E).

**Theorem 5.** *Suppose a randomized mechanism $\mathcal{M}$ satisfies instance-level $(\epsilon, \delta)$-DP, $D$ and $D'$ differ by $k$ instances. The results in Theorems 1, 2, 3 and Corollary 1 hold for $\mathcal{M}$, $D$ and $D'$.*

**Comparison with existing certified prediction methods in centralized setting.** The form of Theorem 1 is similar with the robustness condition against test-time attack in Proposition 1 of Lecuyer *et al.* [39]. This is because the derived robustness conditions are both rooted in the DP properties, but ours focus on the robustness against training-time attacks in FL, which is more challenging considering the model dynamics. Our Theorem 1 is also different from previous certifiably robust centralized learning against backdoor [56] and label flipping [19]. First, our randomness comes from the inherent training randomness of user/instance-level $(\epsilon, \delta)$-DP, e.g., user subsampling and Gaussian noise; while in [56, 19] the randomness only comes from the *explicitly* added training-time noises. Second, our Theorem 1, 2 hold no matter how $\epsilon$ is achieved, which means that we can add different types of noise, leverage different subsampling strategies or even different FL training protocols to achieve user/instance-level $\epsilon$. However, in [56, 19] different certifications require different types of noise (Laplacian, Gaussian, etc.). Additionally, DP is suitable to characterize the robustness against poisoning since off-the-shelf DP composition theorems track privacy cost $\epsilon$, which naturally captures the training dynamics of ML model parameters and thus provides robustness guarantees in a probabilistic manner without additional assumptions. Otherwise one may need to track the deviations of model parameters by analyzing SGD over training, which is theoretically knotty and often requires strong assumptions on Lipschitz continuity, smoothness or convexity for the trained models.

## B Differentially Private Federated Learning

### B.1 `UserDP-FedAvg`

**Algorithm 1:** `UserDP-FedAvg`.

**Input:** Initial model $w_0$, user sampling probability $q$, privacy parameter $\delta$, clipping threshold $S$, noise level $\sigma$, local datasets $D_1, ..., D_N$, local epochs $E$, learning rate $\eta$.
**Output:** FL model $w_T$ and privacy cost $\epsilon$
**Server executes:**
**for** *each round* $t = 1$ **to** $T$ **do**
  $m \leftarrow \max(q \cdot N, 1)$;
  $U_t \leftarrow$ (random subset of $m$ users);
  **for** *each user* $i \in U_t$ *in parallel* **do**
    $\Delta w_t^i \leftarrow$ `UserUpdate`$(i, w_{t-1})$ ;
  $w_t \leftarrow w_{t-1} + \frac{1}{m}\left(\sum_{i \in U_t}$ `Clip`$(\Delta w_t^i, S) + \mathcal{N}\left(0, \sigma^2 S^2\right)\right)$ ;
  $\mathcal{M}$.*accum_priv_spending*$(\sigma, q, \delta)$ ;
$\epsilon = \mathcal{M}$.*get_privacy_spent*() ;
**return** $w_T, \epsilon$
**Procedure** `UserUpdate`$(i, w_{t-1})$
  $w \leftarrow w_{t-1}$ ;
  **for** *local epoch* $e = 1$ **to** $E$ **do**
    **for** *batch* $b \in$ *local dataset* $D_i$ **do**
      $w \leftarrow w - \eta \nabla l(w; b)$
  $\Delta w_t^i \leftarrow w - w_{t-1}$ ;
  **return** $\Delta w_t^i$
**Procedure** `Clip`$(\Delta, S)$
  **return** $\Delta / \max\left(1, \frac{\|\Delta\|_2}{S}\right)$

In Algorithm 1, $\mathcal{M}$.*accum_priv_spending*() and $\mathcal{M}$.*get_privacy_spent*() are the calls on the moments accountant $\mathcal{M}$ refer to the API of Abadi *et al.* [2].

The privacy guarantee for Algorithm 1 is a generalization of [2]. We recall Proposition 1.

**Proposition 1** (`UserDP-FedAvg` Privacy Guarantee)**.** *There exist constants $c_1$ and $c_2$ so that given user sampling probability $q$, and FL rounds $T$, for any $\varepsilon < c_1 q^2 T$, if $\sigma \geq c_2 \frac{q\sqrt{T \log(1/\delta)}}{\epsilon}$, the randomized mechanism $\mathcal{M}$ in Algorithm 1 is $(\epsilon, \delta)$-DP for any $\delta > 0$.*

*Proof.* The proof follows the proof of Theorem 1 in [2], while the notations have slightly different meanings under FL settings. In Proposition 1, we use $q$ to represent *user-level* sampling probability and $T$ to represent FL training rounds. □

**Discussion** Tian *et al.* [41] divide the user-level privacy into global privacy [29, 46] and local privacy [3]. In both local and global privacy, the norm of each update is clipped. The difference lies in that the noise is added on the aggregated model updates in global privacy because a trusted server is assumed, while the noise is added on each local update in local privacy because it assumes that the central server might be malicious. Algorithm 1 belongs to global privacy.

### B.2 `InsDP-FedSGD`

**Algorithm 2:** `InsDP-FedSGD`.

**Input:** Initial model $w_0$, user sampling probability $q$, privacy parameter $\delta$, local clipping threshold $S$, local noise level $\sigma$, local datasets $D_1, ..., D_N$, learning rate $\eta$, batch sampling probability $p$.
**Output:** FL model $w_T$ and privacy cost $\epsilon$
**Server executes:**
**for** *each round* $t = 1$ **to** $T$ **do**
  $m \leftarrow \max(q \cdot N, 1)$;
  $U_t \leftarrow$ (random subset of $m$ clients);
  **for** *each user* $i \in U_t$ *in parallel* **do**
    $\Delta w_t^i \leftarrow$ `UserUpdate`$(i, w_{t-1})$ ;
  $w_t \leftarrow w_{t-1} + \frac{1}{m} \sum_{i \in U_t} \Delta w_t^i$ ;
  $\mathcal{M}$.*accum_priv_spending*$(\sqrt{m}\sigma, pq, \delta)$
$\epsilon = \mathcal{M}$.*get_privacy_spent*() ;
**return** $w_T, \epsilon$

**Procedure** `UserUpdate`$(i, w_{t-1})$
  $w \leftarrow w_{t-1}$ ;
  $b_t^i \leftarrow$(uniformly sample a batch from $D_i$ with probability $p = L/|D_i|$);
  **for** *each* $x_j \in b_t^i$ **do**
    $g(x_j) \leftarrow \nabla l(w; x_j)$;
    $\bar{g}(x_j) \leftarrow$ `Clip`$(g(x_j), S)$ ;
  $\widetilde{g} \leftarrow \frac{1}{L}\left(\sum_j \bar{g}(x_j) + \mathcal{N}\left(0, \sigma^2 S^2\right)\right)$;
  $w \leftarrow w - \eta \widetilde{g}$ ;
  $\Delta w_t^i \leftarrow w - w_{t-1}$ ;
  **return** $\Delta w_t^i$
**Procedure** `Clip`$(\Delta, S)$
  **return** $\Delta / \max\left(1, \frac{\|\Delta\|_2}{S}\right)$

Under FedSGD, when each local model performs one step of DP-SGD [2], the randomized mechanism $\mathcal{M}$ that outputs the global model preserves the instance-level DP. We can regard the one-step update for the global model in Algorithm 2 as:

$$w_t \leftarrow w_{t-1} - \frac{1}{m} \sum_{i \in U_t} \frac{\eta}{L} \left( \sum_{x_j \in b_t^i} \bar{g}(x_j) + \mathcal{N}\left(0, \sigma^2 S^2\right) \right) \tag{5}$$

**Proposition 2** (`InsDP-FedSGD` Privacy Guarantee)**.** *There exist constants $c_1$ and $c_2$ so that given batch sampling probability $p$, and user sampling probability $q$, the number of selected users each round $m$, and FL rounds $T$, for any $\varepsilon < c_1 (pq)^2 T$, if $\sigma \geq c_2 \frac{pq\sqrt{T \log(1/\delta)}}{\epsilon \sqrt{m}}$, the randomized mechanism $\mathcal{M}$ in Algorithm 2 is $(\epsilon, \delta)$-DP for any $\delta > 0$.*

*Proof.* i) In instance-level DP, we consider the sampling probability of each instance under the combination of user-level sampling and batch-level sampling. Since the user-level sampling probability is $q$ and the batch-level sampling probablity is $p$, each instance is sampled with probability $pq$. ii) Additionally, since the sensitivity of instance-wise gradient w.r.t one instance is $S$, after local gradient descent and server FL aggregation, the equivalent sensitivity of global model w.r.t one instance is $S' = \frac{\eta S}{Lm}$ according to Eq (5). iii) Moreover, since the local noise is $n_i \sim \mathcal{N}(0, \sigma^2 S^2)$ , then the "virtual" global noise is $n = \frac{\eta}{mL} \sum_{i \in U_t} n_i$ according to Eq (5), so $n \sim \mathcal{N}(0, \frac{\eta^2 \sigma^2 S^2}{mL^2})$. Let $\frac{\eta^2 \sigma^2 S^2}{mL^2} = \sigma'^2 S'^2$ such that $n \sim \mathcal{N}(0, \sigma'^2 S'^2)$. Because $S' = \frac{\eta S}{Lm}$, the equivalent global noise level is $\sigma'^2 = \sigma^2 m$, i.e., $\sigma' = \sigma\sqrt{m}$.

In Proposition 2, we use $pq$ to represent *instance-level* sampling probability, $T$ to represent FL training rounds, $\sigma\sqrt{m}$ to represent the equivalent global noise level. The rest of the proof follows the proof of Theorem 1 in [2].

□

### B.3 InsDP-FedAvg

**Algorithm 3:** InsDP-FedAvg.

**Input:** Initial model $w_0$, user sampling probability $q$, privacy parameter $\delta$, local clipping threshold $S$, local noise level $\sigma$, local datasets $D_1, ..., D_N$, local steps $V$, learning rate $\eta$, batch sampling probability $p$.
**Output:** FL model $w_T$ and privacy cost $\epsilon$
**Server executes:**
**for** *each round* $t = 1$ **to** $T$ **do**
  $m \leftarrow \max(q \cdot N, 1)$;
  $U_t \leftarrow$ (random subset of $m$ users);
  **for** *each user* $i \in U_t$ *in parallel* **do**
    $\Delta w_t^i, \epsilon_t^i \leftarrow$ `UserUpdate`$(i, w_{t-1})$ ;
  **for** *each user* $i \notin U_t$ **do**
    $\epsilon_t^i \leftarrow \epsilon_{t-1}^i$ ;
  $w_t \leftarrow w_{t-1} + \frac{1}{m} \sum_{i \in U_t} \Delta w_t^i$ ;
  $\epsilon_t = \mathcal{M}.$*parallel_composition*$(\{\epsilon_t^i\}_{i \in [N]})$
$\epsilon = \epsilon_T$ ;
**return** $w_T, \epsilon$
**Procedure** `UserUpdate`$(i, w_{t-1})$
  $w \leftarrow w_{t-1}$ ;
  **for** *each local step* $v = 1$ **to** $V$ **do**
    $b \leftarrow$(uniformly sample a batch from $D_i$ with probability $p = L/|D_i|$);
    **for** *each* $x_j \in b$ **do**
      $g(x_j) \leftarrow \nabla l(w; x_j)$;
      $\bar{g}(x_j) \leftarrow$ `Clip`$(g(x_j), S)$ ;
    $\widetilde{g} \leftarrow \frac{1}{L}(\sum_j \bar{g}(x_j) + \mathcal{N}\left(0, \sigma^2 S^2\right))$;
    $w \leftarrow w - \eta \widetilde{g}$ ;
    $\mathcal{M}^i.$*accum_priv_spending*$(\sigma, p, \delta)$ ;
  $\epsilon_t^i = \mathcal{M}^i.$*get_privacy_spent*$()$ ;
  $\Delta w_t^i \leftarrow w - w_{t-1}$ ;
  **return** $\Delta w_t^i, \epsilon_t^i$
**Procedure** `Clip`$(\Delta, S)$
  **return** $\Delta / \max\left(1, \frac{\|\Delta\|_2}{S}\right)$

**Lemma 2** (InsDP-FedAvg Privacy Guarantee when $T = 1$)**.** *In Algorithm 3, when $T = 1$, suppose local mechanism $\mathcal{M}^i$ satisfies $(\epsilon^i, \delta)$-DP, then global mechanism $\mathcal{M}$ satisfies $(\max_{i \in [N]} \epsilon^i, \delta)$-DP.*

*Proof.* We can regard federated learning as partitioning a dataset $D$ into $N$ disjoint subsets $\{D_1, D_2, \ldots, D_N\}$. $N$ mechanisms $\{\mathcal{M}^1, \ldots, \mathcal{M}^N\}$ are operated on these $N$ parts separately and each $\mathcal{M}^i$ satisfies its own $\epsilon^i$-DP for $i \in [1, N]$. Note that if $i$-th user is not selected , $\epsilon^i = 0$ because local dataset $D_i$ is not accessed and there is no privacy cost. Without loss of generality, we assume the modified data sample $x'$ ($x \to x'$ causes $D \to D'$) is in the local dataset of $k$-th client $D_k$. Let $D, D'$ be two neighboring datasets ($D_k, D_k'$ are also two neighboring datasets). $\mathcal{M}$ is randomized mechanism that outputs the global model, and $\mathcal{M}^i$ is the randomized mechanism that outputs the local model update $\Delta w^i$. Suppose $w_0$ is the initialized and deterministic global model, and $\{z_1, \ldots, z_N\}$ are randomized local updates. We have a sequence of computations $\{z_1 = \mathcal{M}^1(D_1), z_2 = \mathcal{M}^2(D_2; z_1), z_3 = \mathcal{M}^3(D_3; z_1, z_2) \ldots\}$ and $z = \mathcal{M}(D) = w_0 + \sum_{i=1}^N z_i$. Note that if $i$-th user is not selected , $z_i = 0$. According to the parallel composition [54], we have

$$
\begin{aligned}
&\Pr[\mathcal{M}(D) = z] \\
&= \Pr[\mathcal{M}^1(D_1) = z_1] \Pr[\mathcal{M}^2(D_2; z_1) = z_2] \ldots \Pr[\mathcal{M}^N(D_N; z_1, \ldots, z_{N-1}) = z_N] \\
&\leq \exp(\epsilon^k) \Pr[\mathcal{M}^k(D_k'; z_1, \ldots, z_{k-1}) = z_k] \prod_{i \neq k} \Pr[\mathcal{M}^i(D_i; z_1, \ldots, z_{i-1}) = z_i] \\
&= \exp(\epsilon^k) \Pr[\mathcal{M}(D') = z]
\end{aligned}
$$

So $\mathcal{M}$ satisfies $\epsilon^k$-DP when the modified data sample lies in the subset $D_k$. Consider the worst case of where the modified data sample could fall in, we know that $\mathcal{M}$ satisfies $(\max_{i \in [N]} \epsilon^i)$-DP. □

We recall Theorem 4.

**Theorem 4** (InsDP-FedAvg Privacy Guarantee)**.** *In Algorithm 3, at round $t$, suppose local mechanism $\mathcal{M}^i$ satisfies $(\epsilon_t^i, \delta)$-DP, then the global mechanism $\mathcal{M}$ satisfies $\left(\max_{i \in [N]} \epsilon_t^i, \delta\right)$-DP.*

*Proof.* Again, without loss of generality, we assume the modified data sample $x'$ ($x \to x'$ causes $D \to D'$) is in the local dataset of $k$-th user $D_k$. We first consider the case when all users are selected. At each round $t$, $N$ mechanisms are operated on $N$ disjoint parts and each $\mathcal{M}_t^i$ satisfies own $\epsilon^i$-DP where $\epsilon^i$ is the privacy cost for accessing the local dataset $D_i$ *for one round* (not accumulating over previous rounds). Let $D, D'$ be two neighboring datasets ($D_k, D_k'$ are also two neighboring datasets). Suppose $z_0 = \mathcal{M}_{t-1}(D)$ is the aggregated randomized global model at

round $t-1$, and $\{z_1, \ldots, z_N\}$ are the randomized local updates at round $t$, we have a sequence of computations $\{z_1 = \mathcal{M}_t^1(D_1; z_0), z_2 = \mathcal{M}_t^2(D_2; z_0, z_1), z_3 = \mathcal{M}_t^3(D_3; z_0, z_1, z_2) \ldots\}$ and $z = \mathcal{M}_t(D) = z_0 + \sum_i^N z_i$. We first consider the sequential composition [23] to accumulate the privacy cost over FL rounds. According to parallel composition, we have

$$
\begin{aligned}
&\Pr[\mathcal{M}_t(D) = z] \\
&= \Pr[\mathcal{M}_{t-1}(D) = z_0] \prod_{i=1}^{N} \Pr[\mathcal{M}_t^i(D_i; z_0, z_1, \ldots, z_{i-1}) = z_i] \\
&= \Pr[\mathcal{M}_{t-1}(D) = z_0] \Pr[\mathcal{M}_t^k(D_k; z_0, z_1, \ldots, z_{k-1}) = z_k] \prod_{i \neq k} \Pr[\mathcal{M}_t^i(D_i; z_0, z_1, \ldots, z_{i-1}) = z_i] \\
&\leq \exp(\epsilon_{t-1}) \Pr[\mathcal{M}_{t-1}(D') = z_0] \exp(\epsilon^k) \Pr[\mathcal{M}_t^k(D'_k; z_0, z_1, \ldots, z_{k-1}) = z_k] \prod_{i \neq k} \Pr[\mathcal{M}_t^i(D_i; z_0, z_1, \ldots, z_{i-1}) = z_i] \\
&= \exp(\epsilon_{t-1} + \epsilon^k) \Pr[\mathcal{M}_t(D') = z]
\end{aligned}
$$

Therefore, $\mathcal{M}_t$ satisfies $\epsilon_t$-DP, where $\epsilon_t = \epsilon_{t-1} + \epsilon^k$. Because the modified data sample always lies in $D_k$ over $t$ rounds and $\epsilon_0 = 0$, we can have $\epsilon_t = t\epsilon^k$, which means that the privacy guarantee of global mechanism $\mathcal{M}_t$ is only determined by the local mechanism of $k$-th user over $t$ rounds.

Moreover, moment accountant [2] is known to reduce the privacy cost from $\mathcal{O}(t)$ to $\mathcal{O}(\sqrt{t})$. We can use the more advanced composition, i.e., moment accountant, instead of the sequential composition, to accumulate the privacy cost for local mechanism $\mathcal{M}^k$ over $t$ FL rounds. In addition, we consider user subsampling. As described in Algorithm 3, if the user $i$ is not selected at round $t$, then its local privacy cost is kept unchanged at this round.

Take the worst case of where $x'$ could lie in, at round $t$, $\mathcal{M}$ satisfies $\epsilon_t$-DP, where $\epsilon_t = \max_{i \in [N]} \epsilon_t^i$, local mechanism $M^i$ satisfies $\epsilon_t^i$-DP, and the local privacy cost $\epsilon_t^i$ is accumulated via local moment accountant in $i$-th user over $t$ rounds.

□

## C Threat Models

We consider targeted poisoning attacks of two types. In *backdoor* attacks [31, 17], the goal is to embed a backdoor pattern (i.e., a trigger) during training such that any test input with such pattern will be mis-classified as the target. In *label flipping* attacks [12, 35], the labels of clean training examples from one source class are flipped to the target class while the features of the data are kept unchanged. In FL, the purpose of backdoor attacks is to manipulate local models with backdoored local data, so that the global model would behave normally on untampered data samples while achieving high attack success rate on clean data [6]. Given the same purpose, *distributed backdoor* attack (DBA) [58] decomposes the same backdoor pattern to several smaller ones and embedds them to different local training sets for different adversarial users. The goal of label flipping attack against FL is to manipulate local datasets with flipped labels such that the global model will mis-classify the test data in the source class as the target class. The model replacement [6] is a more powerful approach to perform the above attacks, where the attackers first train the local models using the poisoned datasets and then scale the malicious updates before sending them to the server. This way, the attacker's updates would have a stronger impact on the FL model. We use the model replacement method to perform poisoning attacks and study the effectiveness of DPFL.

For `UserDP-FedAvg`, we consider backdoor, distributed backdoor and label flipping attacks via the model replacement approach. Next, we formalize the attack process and introduce the notations. Suppose the attacker controlling $k$ adversarial users, i.e., there are $k$ attackers out of $N$ users. Let $B$ be the original user set of $N$ benign users, and $B'$ be the user set that contains $k$ attackers. Let $D := \{D_1, D_2, \ldots, D_N\}$ be the union of original benign local datasets in all users. For a data sample $z_j^i := \{x_j^i, y_j^i\}$ in $D_i$, we denote its backdoored version as $z'^i_j := \{x_j^i + \delta_x, y^*\}$, where $\delta_x$ is the backdoor pattern, $y^*$ is the targeted label; the DBA version as $z'^i_j := \{x_j^i + \delta_x^i, y^*\}$, where $\delta_x^i$ is the distributed backdoor pattern for attacker $i$; the label-flipped version as $z'^i_j := \{x_j^i, y^*\}$. Note that the composition of all DBA patterns is equivalent to the backdoor pattern, i.e.,

| Algorithm | Dataset | #training samples | $N$ | $m$ | $E$ | $V$ | batch size | $\eta$ | $S$ | $\delta$ | $\bar{C}$ |
|---|---|---|---|---|---|---|---|---|---|---|---|
| `UserDP-FedAvg` | MNIST | 12665 | 200 | 20 | 10 | / | 60 | 0.02 | 0.7 | 0.0029 | 0.5 |
| `UserDP-FedAvg` | CIFAR-10 | 10000 | 200 | 40 | 5 | / | 50 | 0.05 | 1 | 0.0029 | 0.2 |
| `InsDP-FedAvg` | MNIST | 12665 | 10 | 10 | / | 25 | 60 | 0.02 | 0.7 | 0.00001 | 0.5 |
| `InsDP-FedAvg` | CIFAR-10 | 10000 | 10 | 10 | / | 100 | 50 | 0.05 | 1 | 0.00001 | 2 |

Table 1: Dataset description and parameters

$\sum_{i=1}^{k} \delta_x^i = \delta_x$. We assume attacker $i$ has $\alpha_i$ fraction of poisoned samples in its local dataset $D_i'$. Let $D' := \{D'_1, \ldots, D'_{k-1}, D'_k, D_{k+1}, \ldots, D_N\}$ be the union of local datasets under $k$ attackers. The adversarial user $i$ performs model replacement by scaling the model update with hyperparameter $\gamma$ before submitting it to the server, i.e., $\Delta w_t^i \leftarrow \gamma \Delta w_t^k$.

For `InsDP-FedAvg`, we consider both backdoor and label flipping attacks. Since distributed backdoor and model replacement attack are proposed for adversarial users rather than adversairal instances, we do not consider them for instantce-level DPFL. There are $k$ backdoored or label-flipped instances $\{z'_1, z'_2, \ldots, z'_k\}$, which could be controlled by the same or multiple users.

# D Experimental Details and Additional Results

## D.1 Datasets and Models

We evaluate our robustness certification results with two datasets: MNIST [38] and CIFAR-10 [37]. For each dataset, we use corresponding standard CNN architectures in the differential privacy library [1] of PyTorch [49].

**MNIST**: We study an image classification problem of handwritten digits in MNIST. It is a dataset of 70000 28x28 pixel images of digits in 10 classes, split into a train set of 60000 images and a test set of 10000 images. Except Section D.7, we consider binary classification on classes 0 and 1, making our train set contain 12665 samples, and the test set 2115 samples. The model consists of two Conv-ReLu-MaxPooling layers and two linear layers.

**CIFAR-10**: We study image classification of vehicles and animals in CIFAR-10. This is a harder dataset than MNIST, consisting of 60000 32x32x3 images, split into a train set of 50000 and a test set of 10000. Except Section D.7, we consider binary classification on class airplane and bird, making our train set contain 10000 samples, and the test set 2000 samples. The model consists of four Conv-ReLu-AveragePooling layers and one linear layer. When training on CIFAR10, we follow the standard practice for differential privacy [2, 36] and fine-tune a whole model pre-trained non-privately on the more complex CIFAR100, a similarly sized but more complex benchmark dataset.

## D.2 Training Details

We simulate the federated learning setup by splitting the training datasets for $N$ FL users in an i.i.d manner. FL users run SGD with learning rate $\eta$, momentum $0.9$, weight decay $0.0005$ to update the local models. The training parameter setups are summarized in Table 1. Following McMahan *et al.* [46] that use $\delta \approx \frac{1}{N^{1.1}}$ as privacy parameter, for `UserDP-FedAvg` we set $\delta = 0.0029$ according to the total number of users, and for `InsDP-FedAvg` we set $\delta = 0.00001$ according the total number of training samples. Next we summarize the privacy guarantees and clean accuracy offered when we study the certified prediction and certified attack cost, which are also the training parameters setups when $k = 0$ in Figure 1, 2, 4, 5, 7, 6.

**User-level DPFL** In order to study the user-level certified prediction under different privacy guarantee, for MNIST, we set $\epsilon$ to be $0.2808, 0.4187, 0.6298, 0.8694, 1.8504, 2.8305, 4.8913, 6.9269$, which are obtained by training `UserDP-FedAvg` FL model for 3 rounds with noise level $\sigma = 3.0, 2.3, 1.8, 1.5, 1.0, 0.8, 0.6, 0.5$, respectively (Figure 1(a)). For CIFAR-10, we set $\epsilon$ to be $0.2444, 0.3663, 0.4527, 0.5460, 0.8781, 1.3551, 2.0757, 2.9755, 6.2436$, which are obtained by training `UserDP-FedAvg` FL model for one round with noise level $\sigma = 4.0, 3.0, 2.6, 2.3, 1.7, 1.3, 1.0, 0.8, 0.5$, respectively (Figure 1(b)).

To certify the attack cost under different number of adversarial users $k$ (Figure 2), for MNIST, we set the noise level $\sigma$ to be 2.5. When $k = 0$, after training `UserDP-FedAvg` for $T = 3, 4, 5$ rounds, we obtain FL models with privacy guarantee $\epsilon = 0.3672, 0.4025, 0.4344$ and clean accuracy (average over $M$ runs) $86.69\%, 88.76\%, 88.99\%$. For CIFAR-10, we set the noise level $\sigma$ to be 3.0. After training `UserDP-FedAvg` for $T = 3, 4$ rounds under $k = 0$, we obtain FL models with privacy guarantee $\epsilon = 0.5346, 0.5978$ and clean accuracy $78.63\%, 78.46\%$.

With the interest of certifying attack cost under different user-level DP guarantee (Figure 4, Figure 7), we explore the empirical attack cost and the certified attack cost lower bound given different $\epsilon$. For MNIST, we set the privacy guarantee $\epsilon$ to be $1.2716, 0.8794, 0.6608, 0.5249, 0.4344$, which are obtained by training `UserDP-FedAvg` FL models for 5 rounds under noise level $\sigma = 1.3, 1.6, 1.9, 2.2, 2.5$, respectively, and the clean accuracy for the corresponding models are $99.50\%, 99.06\%, 96.52\%, 93.39\%, 88.99\%$. For CIFAR-10, we set the privacy guarantee $\epsilon$ to be $1.600, 1.2127, 1.0395.0.8530, 0.7616, 0.6543, 0.5978$, which are obtained by training `UserDP-FedAvg` FL models for 4 rounds under noise level $\sigma = 1.5, 1.8, 2.0, 2.3, 2.5, 2.8, 3.0$, respectively, and the clean accuracy for the corresponding models are $85.59\%, 84.52\%, 83.23\%, 81.90\%, 81.27\%, 79.23\%, 78.46\%$.

**Instance-level DPFL** To certify the prediction for instance-level DPFL under different privacy guarantee, for MNIST, we set privacy cost $\epsilon$ to be $0.2029, 0.2251, 0.2484, 0.3593, 0.4589, 0.6373, 1.0587, 3.5691$, which are obtained by training `InsDP-FedAvg` FL models for 3 rounds with noise level $\sigma = 15, 10, 8, 5, 4, 3, 2, 1$, respectively (Figure 1(c)). For CIFAR-10, we set privacy cost $\epsilon$ to be $0.3158, 0.3587, 0.4221, 0.5130, 0.6546, 0.9067, 1.4949, 4.6978$, which are obtained by training `InsDP-FedAvg` FL models for one round with noise level $\sigma = 8, 7, 6, 5, 4, 3, 2, 1$, respectively (Figure 1(d)).

With the aim to study certified attack cost under different number of adversarial instances $k$, for MNIST, we set the noise level $\sigma$ to be 10. When $k = 0$, after training `InsDP-FedAvg` for $T = 4, 9$ rounds, we obtain FL models with privacy guarantee $\epsilon = 0.2383, 0.304$ and clean accuracy (average over $M$ runs) $96.40\%, 96.93\%$ (Figure 6(a)(b)). For CIFAR-10, we set the noise level $\sigma$ to be 8.0. After training `InsDP-FedAvg` for one round under $k = 0$, we obtain FL models with privacy guarantee $\epsilon = 0.3158$ and clean accuracy $61.78\%$ (Figure 5(a)(b)).

In order to study the empirical attack cost and certified attack cost lower bound under different instance-level DP guarantee, we set the privacy guarantee $\epsilon$ to be $0.5016, 0.311, 0.2646, 0.2318, 0.2202, 0.2096, 0.205$ for MNIST, which are obtained by training `InsDP-FedAvg` FL models for 6 rounds under noise level $\sigma = 5, 8, 10, 13, 15, 18, 20$, respectively, and the clean accuracy for the corresponding models are $99.60\%, 98.81\%, 97.34\%, 92.29\%, 88.01\%, 80.94\%, 79.60\%$ (Figure 6 (c)(d)). For CIFAR-10, we set the privacy guarantee $\epsilon$ to be $1.261, 0.9146, 0.7187, 0.5923, 0.5038, 0.4385$, which are obtained by training `InsDP-FedAvg` FL models for 2 rounds under noise level $\sigma = 3, 4, 5, 6, 7, 8$, respectively, and the clean accuracy for the corresponding models are $84.47\%, 80.99\%, 76.01\%, 68.65\%, 63.07\%, 60.65\%$ (Figure 5 (c)(d)).

With the intention of exploring the upper bound for $k$ given $\tau$ under different instance-level DP guarantee, for MNIST, we set noise level $\sigma$ to be $5, 8, 10, 13, 20$, respectively, to obtain instance-DP FL models after 10 rounds with privacy guarantee $\epsilon = 0.6439, 0.3937, 0.3172, 0.2626, 0.2179$ and clean accuracy $99.58\%, 98.83\%, 97.58\%, 95.23\%, 85.72\%$ (Figure 7(c)). For CIFAR-10, we set noise level $\sigma$ to be $3, 4, 5, 6, 7, 8$ and train `InsDP-FedAvg` for $T = 3$ rounds, to obtain FL models with privacy guarantee $\epsilon = 1.5365, 1.1162, 0.8777, 0.7238, 0.6159, 0.5361$ and clean accuracy $84.34\%, 80.27\%, 74.62\%, 66.94\%, 62.14\%, 59.75\%$ (Figure 7(d)).

### D.3 Additional Implementation Details

**(Threat Models)** For the attacks against `UserDP-FedAvg`, by default, the local poison fraction $\alpha = 100\%$, and the scale factor $\gamma = 50$. We use same parameters setups for all $k$ attackers. In terms of label flipping attacks, the attackers swap the label of images in source class (digit 1 for MNIST; bird for CIFAR-10) into the target label

Figure 3: Backdoor pattern (left) and distributed backdoor patterns (right) on CIFAR-10.

(digit 0 for MNIST; airplane for CIFAR-10). In terms of backdoor attacks in MNIST and CIFAR-10, the attackers add a backdoor pattern, as shown in Figure 3 (left), in images and swap the label of any sample with such pattern into the target label (digit 0 for MNIST; airplane for CIFAR-10). In terms of distributed backdoor attacks, Figure 3 (right) shows an example when the triangle pattern is evenly decomposed into $k = 4$ parts, and they are used as the distributed patterns for $k = 4$ attackers respectively. For the cases where there are more or fewer distributed attackers, the similar decomposition strategy is adopted.

For the attacks against `InsDP-FedAvg`, the same target classes and backdoor patterns are used as `UserDP-FedAvg`. The parameters setups are the same for all $k$ poisoned instances.

**(Robustness Certification)** We certified 2115/2000 test samples from the MNIST/CIFAR-10 test sets. In Theorem 3 and Corollary 1 that are related to certified attack cost, $\bar{C}$ specifies the range of $C(\cdot)$. In the implementation, $\bar{C}$ is set to be larger than the maximum empirical attack cost evaluated on the test sets (see Table 1 for details). For each dataset, we use the same $\bar{C}$ for cost function $C$ defined in Example 1 and Example 2. When using Monte-Carlo sampling, we run $M = 1000$ times for certified accuracy, and $M = 100$ times for certified attack cost in all experiments.

**(Machines)** We simulate the federated learning setup (1 server and N users) on a Linux machine with Intel® Xeon® Gold 6132 CPUs and 8 NVidia® 1080Ti GPUs.

**(Libraries)** All code is implemented in Pytorch [49]. Please see the submitted code for full details.

### D.4 Certified Attack Cost of user-level DPFL under Different $\epsilon$

Here we further explore the impacts of different factors on the certified attack cost. Figure 4 presents the empirical attack cost and the certified attack cost lower bound given different $\epsilon$ on user-level DP. It is shown that as the privacy guarantee becomes stronger, i.e. smaller $\epsilon$, the model is more robust achieving higher $J(D')$ and $\overline{J(D')}$. In Figure 7 (a)(b), we train user-level ($\epsilon$, $\delta$) DPFL models, calculate corresponding $J(D)$, and plot the lower bound of $k$ given different attack effectiveness hyperparameter $\tau$ according to Corollary 1. It shows that 1) when the required attack effectiveness is higher, i.e., $\tau$ is larger, more number of attackers is required. 2) To achieve the same effectiveness of attack, fewer number of attackers is needed for larger $\epsilon$, which means that DPFL model with weaker privacy is more vulnerable to poisoning attacks.

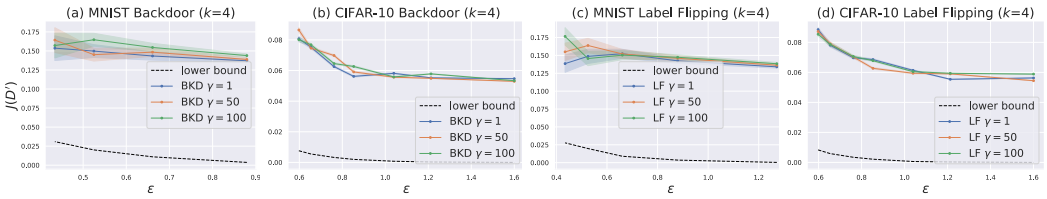


Figure 4: Certified attack cost of user-level DPFL with different $\epsilon$ under different attacks.

### D.5 Robustness Evaluation of Instance-level DPFL

**Certified Prediction.** Figure 1(c)(d) show the instance-level certified accuracy under different $\epsilon$. The optimal $\epsilon$ for K is around $0.3593$ for MNIST and $0.6546$ for CIFAR-10, which is aligned with our observation of the tradeoff between certified accuracy and privacy on user-level DPFL (Section 5.1).

**Certified Attack Cost.** Figure 5 show the certified attack cost on CIFAR-10. From Figure 5 (a)(b), poisoning more instances (i.e., larger $k$) induces lower theoretical and empirical attack cost. From Figure 5 (c)(d), it is clear that instance-level DPFL with stronger privacy guarantee provides higher attack cost both empirically and theoretically, meaning that it is more robust against poisoning attacks. Figure 6 shows the robustness evaluation of instance-level DPFL on MNIST where the results are similar to the results on CIFAR-10 in Figure 5.

Figure 7 (c)(d) show the lower bound of $k$ under different instance-level $\epsilon$ given different $\tau$. Fewer poisoned instances are required to reduce the $J(D')$ to the similar level for a less private DPFL model, indicating that the model is easier to be attacked.

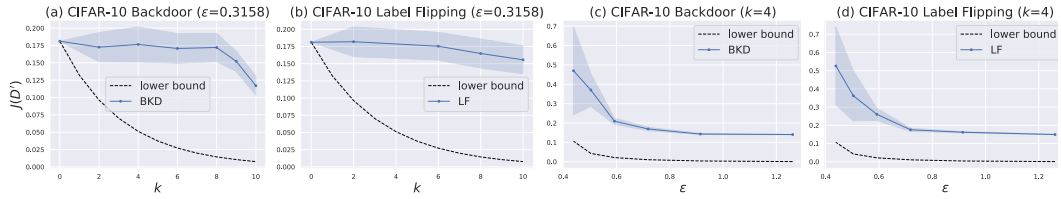


Figure 5: Certified attack cost of instance-level DPFL under different attacks given different number of malicious instances $k$ (a)(b) and different $\epsilon$ (c)(d).

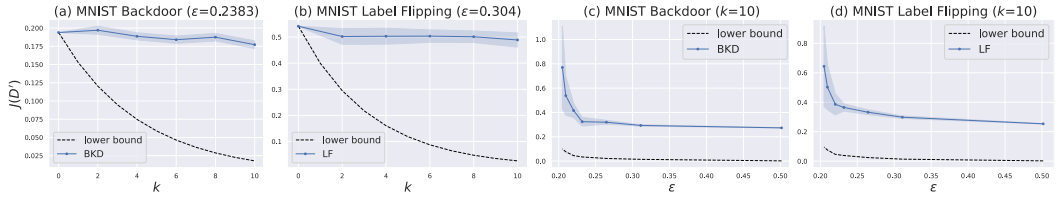


Figure 6: Certified attack cost of instance-level DPFL on MNIST under different attacks given different number of malicious instances $k$ (a)(b) and different $\epsilon$ (c)(d).

### D.6 Certified Accuracy with Confidence Level

Here we present the certified accuracy with confidence level. We use Hoeffding's inequality [33] to calibrates the empirical estimation with one-sided error tolerance $\psi$, i.e., one-sided confidence level $1-\psi$. We first use Monte-Carlo sampling by running the private FL algorithms for $M$ times, with class confidence $f_c^s = f_c(\mathcal{M}(D), x)$ for class $c$ each time. We denote the empirical estimation as $\widetilde{F}_c(\mathcal{M}(D), x) = \frac{1}{M}\sum_{s=1}^{M} f_c^s$. For a test input $x$, suppose $\mathbb{A}, \mathbb{B} \in [C]$ satisfy $\mathbb{A} = \arg\max_{c\in[C]} \widetilde{F}_c(\mathcal{M}(D), x)$ and $\mathbb{B} = \arg\max_{c\in[C]:c\neq\mathbb{A}} \widetilde{F}_c(\mathcal{M}(D), x)$. For a given error tolerance $\psi$, we use Hoeffding's inequality to compute a lower bound $\underline{F_{\mathbb{A}}(\mathcal{M}(D), x)}$ on the class confidence $F_{\mathbb{A}}(\mathcal{M}(D), x)$ and a upper bound $\overline{F_{\mathbb{B}}(\mathcal{M}(D), x)}$ on the class confidence $F_{\mathbb{B}}(\mathcal{M}(D), x)$ according to

$$\underline{F_{\mathbb{A}}(\mathcal{M}(D), x)} = \widetilde{F}_{\mathbb{A}}(\mathcal{M}(D), x) - \sqrt{\frac{\log(1/\psi)}{2M}}, \quad \underline{F_{\mathbb{B}}(\mathcal{M}(D), x)} = \widetilde{F}_{\mathbb{B}}(\mathcal{M}(D), x) + \sqrt{\frac{\log(1/\psi)}{2M}}. \tag{6}$$

$\underline{F_{\mathbb{A}}(\mathcal{M}(D), x)}$ and $\overline{F_{\mathbb{B}}(\mathcal{M}(D), x)}$ are used as the expected class confidences for the evaluation of Theorem 2. We use $\psi = 0.01$ and $M = 1000$ for all experiments.

As shown in Figure 8, we can observe the same tradeoff between $\epsilon$ and certified accuracy as we discussed in Figure 1. In general, the K in Figure 8 is smaller than the K in Figure 1 because we calibrate the empirical estimation according to Eq. (6), and the class confidence gap between top-1 and top-2 class is narrowed.

### D.7 Robustness Evaluation of user-level DPFL on 10-class Classification

Here we report the robustness evaluation of user-level DPFL under backdoor attacks on 10-class classification problem. Figure 10 presents the certified accuracy under different $\epsilon$. We can observe the tradeoff between $\epsilon$ and certified accuracy on MNIST. On CIFAR-10, larger $k$ can be certified with smaller $\epsilon$. The certified K is relatively small because we set large $\epsilon$ to preserve a reasonable accuracy for 10-class classification. Our results can inspire advanced DP mechanisms that provide tighter privacy guarantee (i.e., smaller $\epsilon$) while achieving similar level of accuracy. In terms of certified attack cost, as shown in Figure 9 and 11, the trends are similar to the 2-class results in Figure 2, 4 and 7.

## E Proofs of Certified Robustness Analysis

We restate our Lemma 1 here.

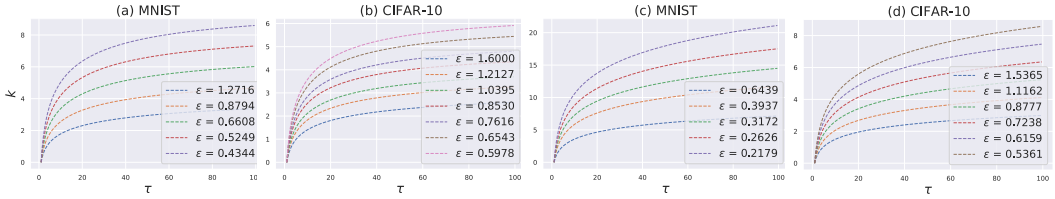


Figure 7: Lower bound of $k$ under user-level $\epsilon$ (a,b) and instance-level $\epsilon$ (c,d) given attack effectiveness $\tau$.

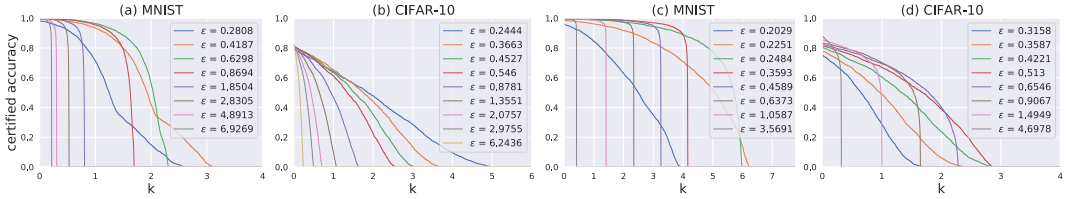


Figure 8: Certified accuracy under 99% confidence of FL satisfying user-level DP (a,b), and instance-level DP (c,d).

**Lemma 1** (Group DP). *For mechanism $\mathcal{M}$ that satisfies $(\epsilon, \delta)$-DP, it satisfies $(k\epsilon, \frac{1-e^{k\epsilon}}{1-e^{\epsilon}}\delta)$-DP for groups of size $k$. That is, for any $d, d' \in \mathcal{D}$ that differ by $k$ individuals, and any $E \subseteq \Theta$ it holds that $\Pr[\mathcal{M}(d) \in E] \leq e^{k\epsilon} \Pr[\mathcal{M}(d') \in E] + \frac{1-e^{k\epsilon}}{1-e^{\epsilon}}\delta$.*

*Proof.* We denote $d$ as $d_0$, $d'$ as $d_k$. $d_i$ differ $i$ individuals with $d_0$. For any $i \in [1, k]$, $d_i$ and $d_{i-1}$ differ by one individual, thus

$$\Pr[M(d_{i-1})] \leq e^{\epsilon} \Pr[M(d_i)] + \delta. \tag{7}$$

By iteratively applying Eq. (7) $k$ times, we have

$$\begin{aligned}\Pr[M(d_0)] &\leq e^{k\epsilon} \Pr[M(d_k)] + (1 + e^{\epsilon} + e^{2\epsilon} + \ldots + e^{(k-1)\epsilon})\delta \\ &= e^{k\epsilon} \Pr[M(d_k)] + \frac{1 - e^{k\epsilon}}{1 - e^{\epsilon}}\delta\end{aligned}$$

□

Before we prove Theorem 1, we introduce the following lemma:

**Lemma 3.** *Suppose a randomized mechanism $\mathcal{M}$ satisfies user-level $(\epsilon, \delta)$-DP. For two user sets $B$ and $B'$ that differ by one user, $D$ and $D'$ are the corresponding training datasets. For a test input $x$, for any $c \in [C]$, $f_c(\mathcal{M}(D), x) \in [0, 1]$ is the class confidence, then the expected class confidence $F_c(\mathcal{M}(D), x) := \mathbb{E}[f_c(\mathcal{M}(D), x)]$ meets the following property:*

$$F_c(\mathcal{M}(D), x) \leq e^{\epsilon} F_c(\mathcal{M}(D'), x) + \delta \tag{8}$$

*Proof.* Define $\Theta(a) := \{\theta : f_c(\theta, x) > a\}$. Then

$$\begin{aligned}F_c(\mathcal{M}(D), x) = \mathbb{E}[f_c(\mathcal{M}(D), x)] &= \int_0^1 \mathbb{P}\left[f_c(\mathcal{M}(D), x) > a\right] da \\ &= \int_0^1 \mathbb{P}\left[\mathcal{M}(D) \in \Theta(a)\right] da \\ &\leq \int_0^1 \left(e^{\epsilon}\mathbb{P}\left[\mathcal{M}(D') \in \Theta(a)\right] + \delta\right) da \\ &= \int_0^1 e^{\epsilon}\mathbb{P}\left[f_c(\mathcal{M}(D'), x) > a\right] da + \int_0^1 \delta da \\ &= e^{\epsilon} F_c(\mathcal{M}(D'), x) + \delta\end{aligned}$$

□

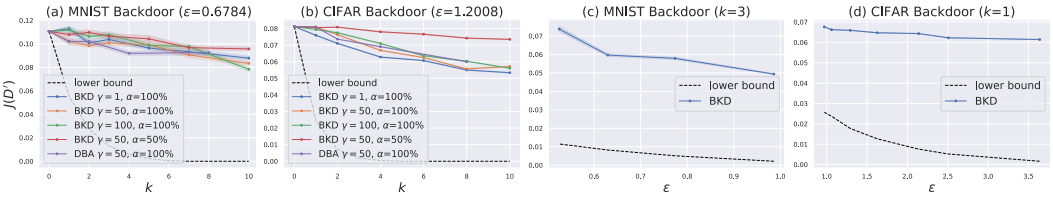


Figure 9: Certified attack cost of user-level DPFL on 10-class classification given different number of malicious instances $k$ (a)(b) and different $\epsilon$ (c)(d).

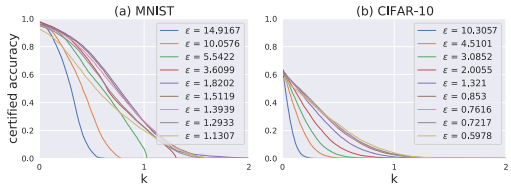


Figure 10: Certified accuracy of FL satisfying user-level DP on 10-class classification.

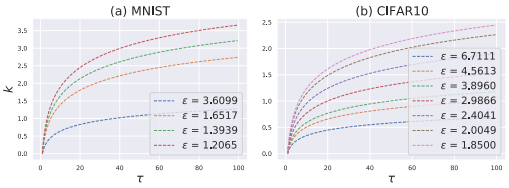


Figure 11: Lower bound of $k$ on 10-class classification under user-level $\epsilon$ given attack effectiveness $\tau$.

We recall Theorem 1.

**Theorem 1** (Condition for Certified Prediction under One Adversarial User)**.** *Suppose a randomized mechanism $\mathcal{M}$ satisfies user-level $(\epsilon, \delta)$-DP. For two user sets $B$ and $B'$ that differ by one user, $D$ and $D'$ are the corresponding training datasets. For a test input $x$, suppose $\mathbb{A}, \mathbb{B} \in [C]$ satisfy $\mathbb{A} = \arg\max_{c\in[C]} F_c(\mathcal{M}(D), x)$ and $\mathbb{B} = \arg\max_{c\in[C]:c\neq\mathbb{A}} F_c(\mathcal{M}(D), x)$, then if*

$$F_{\mathbb{A}}(\mathcal{M}(D), x) > e^{2\epsilon} F_{\mathbb{B}}(\mathcal{M}(D), x) + (1 + e^{\epsilon})\delta, \tag{1}$$

*it is guaranteed that*

$$H(\mathcal{M}(D'), x) = H(\mathcal{M}(D), x) = \mathbb{A}.$$

*Proof.* According to Lemma 3,

$$F_{\mathbb{A}}(\mathcal{M}(D), x) \leq e^{\epsilon} F_{\mathbb{A}}(\mathcal{M}(D'), x) + \delta \tag{9}$$

$$F_{\mathbb{B}}(\mathcal{M}(D'), x) \leq e^{\epsilon} F_{\mathbb{B}}(\mathcal{M}(D), x) + \delta. \tag{10}$$

Then

$$\begin{aligned}
F_{\mathbb{A}}(\mathcal{M}(D'), x) &\geq \frac{F_{\mathbb{A}}(\mathcal{M}(D), x) - \delta}{e^{\epsilon}} && \text{(Because of Eq. 9)} \\
&\geq \frac{e^{2\epsilon} F_{\mathbb{B}}(\mathcal{M}(D), x) + (1 + e^{\epsilon})\delta - \delta}{e^{\epsilon}} && \text{(Because of the given condition Eq. 1)} \\
&= e^{\epsilon} F_{\mathbb{B}}(\mathcal{M}(D), x) + \delta \\
&\geq e^{\epsilon} \left( \frac{F_{\mathbb{B}}(\mathcal{M}(D'), x) - \delta}{e^{\epsilon}} \right) + \delta && \text{(Because of Eq. 10)} \\
&= F_{\mathbb{B}}(\mathcal{M}(D'), x),
\end{aligned}$$

which indicates that the prediction of $\mathcal{M}(D')$ at $x$ is $\mathbb{A}$ by definition. □

Before we prove Theorem 2, we introduce the following lemma:

**Lemma 4.** *Suppose a randomized mechanism $\mathcal{M}$ satisfies user-level $(\epsilon, \delta)$-DP. For two user sets $B$ and $B'$ that differ $k$ users, $D$ and $D'$ are the corresponding training datasets. For a test input $x$, for any $c \in [C]$ , $f_c(\mathcal{M}(D), x) \in [0, 1]$ is the class confidence, then the expected class confidence $F_c(\mathcal{M}(D), x) := \mathbb{E}[f_c(\mathcal{M}(D), x)]$ meets the following property:*

$$F_c(\mathcal{M}(D), x) \leq e^{k\epsilon} F_c(\mathcal{M}(D'), x) + \frac{1 - e^{k\epsilon}}{1 - e^{\epsilon}} \delta \tag{11}$$

*Proof.* Define $\Theta(a) := \{\theta : f_c(\theta, x) > a\}$. Then

$$\begin{aligned}
F_c(\mathcal{M}(D), x) &= \int_0^1 \mathbb{P}\left[f_c(\mathcal{M}(D), x) > a\right] da \\
&= \int_0^1 \mathbb{P}\left[\mathcal{M}(D) \in \Theta(a)\right] da \\
&\leq \int_0^1 \left(e^{k\epsilon}\mathbb{P}\left[\mathcal{M}(D') \in \Theta(a)\right] + \frac{1-e^{k\epsilon}}{1-e^{\epsilon}}\delta\right) da \\
&\qquad \text{(Because of Group DP property in Lemma 1)} \\
&= \int_0^1 e^{k\epsilon}\mathbb{P}\left[f_c(\mathcal{M}(D'), x) > a\right] da + \int_0^1 \frac{1-e^{k\epsilon}}{1-e^{\epsilon}}\delta da \\
&= e^{k\epsilon}F_c(\mathcal{M}(D'), x) + \frac{1-e^{k\epsilon}}{1-e^{\epsilon}}\delta
\end{aligned}$$

□

We recall Theorem 2.[1]

**Theorem 2** (Upper Bound of $k$ for Certified Prediction)**.** *Suppose a randomized mechanism $\mathcal{M}$ satisfies user-level $(\epsilon, \delta)$-DP. For two user sets $B$ and $B'$ that differ $k$ users, $D$ and $D'$ are the corresponding training datasets. For a test input $x$, suppose $\mathbb{A}, \mathbb{B} \in [C]$ satisfy $\mathbb{A} = \arg\max_{c\in[C]} F_c(\mathcal{M}(D), x)$ and $\mathbb{B} = \arg\max_{c\in[C]:c\neq\mathbb{A}} F_c(\mathcal{M}(D), x)$, then $H(\mathcal{M}(D'), x) = H(\mathcal{M}(D), x) = \mathbb{A}$, $\forall k < \mathsf{K}$ where $\mathsf{K}$ is the certified number of adversarial users:*

$$\mathsf{K} = \frac{1}{2\epsilon}\log\frac{F_{\mathbb{A}}(\mathcal{M}(D), x)(e^{\epsilon}-1)+\delta}{F_{\mathbb{B}}(\mathcal{M}(D), x)(e^{\epsilon}-1)+\delta} \tag{2}$$

*Proof.* According to Lemma 4, we have

$$F_{\mathbb{A}}(\mathcal{M}(D), x) \leq e^{k\epsilon}F_{\mathbb{A}}(\mathcal{M}(D'), x) + \frac{1-e^{k\epsilon}}{1-e^{\epsilon}}\delta \tag{12}$$

$$F_{\mathbb{B}}(\mathcal{M}(D'), x) \leq e^{k\epsilon}F_{\mathbb{B}}(\mathcal{M}(D), x) + \frac{1-e^{k\epsilon}}{1-e^{\epsilon}}\delta. \tag{13}$$

We can re-write the given condition $k < \mathsf{K}$ according to Eq. (2) as

$$e^{2k\epsilon}F_{\mathbb{B}}(\mathcal{M}(D), x) + (1+e^{k\epsilon})\frac{1-e^{k\epsilon}}{1-e^{\epsilon}}\delta < F_{\mathbb{A}}(\mathcal{M}(D), x). \tag{14}$$

Then

$$\begin{aligned}
F_{\mathbb{A}}(\mathcal{M}(D'), x) &\geq \frac{F_{\mathbb{A}}(\mathcal{M}(D), x) - \frac{1-e^{k\epsilon}}{1-e^{\epsilon}}\delta}{e^{k\epsilon}} && \text{(Because of Eq. 12)} \\
&> \frac{e^{2k\epsilon}F_{\mathbb{B}}(\mathcal{M}(D), x) + (1+e^{k\epsilon})\frac{1-e^{k\epsilon}}{1-e^{\epsilon}}\delta - \frac{1-e^{k\epsilon}}{1-e^{\epsilon}}\delta}{e^{k\epsilon}} \\
&\qquad \text{(Because of the given condition Eq.14)} \\
&= e^{k\epsilon}F_{\mathbb{B}}(\mathcal{M}(D), x) + \frac{1-e^{k\epsilon}}{1-e^{\epsilon}}\delta \\
&\geq e^{k\epsilon}\left(\frac{F_{\mathbb{B}}(\mathcal{M}(D'), x) - \frac{1-e^{k\epsilon}}{1-e^{\epsilon}}\delta}{e^{k\epsilon}}\right) + \frac{1-e^{k\epsilon}}{1-e^{\epsilon}}\delta && \text{(Because of Eq. 13)} \\
&= F_{\mathbb{B}}(\mathcal{M}(D'), x),
\end{aligned}$$

which indicates that the prediction of $\mathcal{M}(D')$ at $x$ is $\mathbb{A}$ by definition. □

[1] We apologize that there is a typo in Theorem 2 of the submitted main paper, where $\bar{C}$, which equals to 1, should be removed from the Eq. (2). We present the correct Theorem 2 in appendix.

We recall Theorem 3.

**Theorem 3** (Attack Cost with $k$ Attackers)**.** *Suppose a randomized mechanism $\mathcal{M}$ satisfies user-level $(\epsilon, \delta)$-DP. For two user sets $B$ and $B'$ that differ $k$ users, $D$ and $D'$ are the corresponding training datasets. Let $J(D)$ be the expected attack cost where $|C(\cdot)| \leq \bar{C}$. Then,*

$$\begin{aligned}
&\min\{e^{k\epsilon}J(D) + \frac{e^{k\epsilon}-1}{e^{\epsilon}-1}\delta\bar{C}, \bar{C}\} \geq J(D') \geq \max\{e^{-k\epsilon}J(D) - \frac{1-e^{-k\epsilon}}{e^{\epsilon}-1}\delta\bar{C}, 0\}, \quad \textit{if} \quad C(\cdot) \geq 0 \\
&\min\{e^{-k\epsilon}J(D) + \frac{1-e^{-k\epsilon}}{e^{\epsilon}-1}\delta\bar{C}, 0\} \geq J(D') \geq \max\{e^{k\epsilon}J(D) - \frac{e^{k\epsilon}-1}{e^{\epsilon}-1}\delta\bar{C}, -\bar{C}\}, \quad \textit{if} \quad C(\cdot) \leq 0
\end{aligned} \tag{3}$$

*Proof.* We first consider $C(\cdot) \geq 0$. Define $\Theta(a) = \{\theta : C(\theta) > a\}$.

$$\begin{aligned}
J(D) &= \int_0^{\bar{C}} \mathbb{P}\left[C(\mathcal{M}(D)) > a\right] da \\
&= \int_0^{\bar{C}} \mathbb{P}\left[\mathcal{M}(D)) \in \Theta(a)\right] da \\
&\leq \int_0^{\bar{C}} \left(e^{k\epsilon}\mathbb{P}\left[\mathcal{M}(D')) \in \Theta(a)\right] + \frac{1-e^{k\epsilon}}{1-e^{\epsilon}}\delta\right) da \\
&\qquad \text{(Because of Group DP property in Lemma 1)} \\
&= \int_0^{\bar{C}} e^{k\epsilon}\mathbb{P}\left[\mathcal{M}(D')) \in \Theta(a)\right] da + \frac{1-e^{k\epsilon}}{1-e^{\epsilon}}\delta\bar{C} \\
&= \int_0^{\bar{C}} e^{k\epsilon}\mathbb{P}\left[C(\mathcal{M}(D')) > a\right] da + \frac{1-e^{k\epsilon}}{1-e^{\epsilon}}\delta\bar{C} \\
&= e^{k\epsilon}J(D') + \frac{1-e^{k\epsilon}}{1-e^{\epsilon}}\delta\bar{C}
\end{aligned}$$

i.e.,

$$J(D') \geq e^{-k\epsilon}J(D) - \frac{1-e^{-k\epsilon}}{e^{\epsilon}-1}\delta\bar{C}.$$

Switch the role of $D$ and $D'$, we have

$$J(D') \leq e^{k\epsilon}J(D) + \frac{1-e^{k\epsilon}}{1-e^{\epsilon}}\delta\bar{C}.$$

Also note that $0 \leq J(D') \leq \bar{C}$ trivially holds due to $0 \leq C(\cdot) \leq \bar{C}$, thus

$$\min\{e^{k\epsilon}J(D) + \frac{e^{k\epsilon}-1}{e^{\epsilon}-1}\delta\bar{C}, \bar{C}\} \geq J(D') \geq \max\{e^{-k\epsilon}J(D) - \frac{1-e^{-k\epsilon}}{e^{\epsilon}-1}\delta\bar{C}, 0\}.$$

Next we consider $C(\cdot) \leq 0$. Define $\Theta(a) = \{\theta : C(\theta) < a\}$.

$$\begin{aligned}
J(D) &= -\int_{-\bar{C}}^0 \mathbb{P}\left[C(\mathcal{M}(D)) < a\right] da \\
&= -\int_{-\bar{C}}^0 \mathbb{P}\left[\mathcal{M}(D)) \in \Theta(a)\right] da \\
&\geq -\int_{-\bar{C}}^0 \left(e^{k\epsilon}\mathbb{P}\left[\mathcal{M}(D')) \in \Theta(a)\right] + \frac{1-e^{k\epsilon}}{1-e^{\epsilon}}\delta\right) da \\
&\qquad \text{(Because of Group DP property in Lemma 1)} \\
&= -\int_{-\bar{C}}^0 e^{k\epsilon}\mathbb{P}\left[\mathcal{M}(D')) \in \Theta(a)\right] da - \frac{1-e^{k\epsilon}}{1-e^{\epsilon}}\delta\bar{C} \\
&= -\int_{-\bar{C}}^0 e^{k\epsilon}\mathbb{P}\left[C(\mathcal{M}(D')) < a\right] da - \frac{1-e^{k\epsilon}}{1-e^{\epsilon}}\delta\bar{C} \\
&= e^{k\epsilon}J(D') - \frac{1-e^{k\epsilon}}{1-e^{\epsilon}}\delta\bar{C}
\end{aligned}$$

i.e.,

$$J(D') \leq e^{-k\epsilon} J(D) + \frac{1 - e^{-k\epsilon}}{e^{\epsilon} - 1} \delta \bar{C}.$$

Switch the role of $D$ and $D'$, we have

$$J(D') \geq e^{k\epsilon} J(D) - \frac{1 - e^{k\epsilon}}{1 - e^{\epsilon}} \delta \bar{C}.$$

Also note that $-\bar{C} \leq J(D') \leq 0$ trivially holds due to $-\bar{C} \leq C(\cdot) \leq 0$, thus

$$\min\{e^{-k\epsilon} J(D) + \frac{1 - e^{-k\epsilon}}{e^{\epsilon} - 1} \delta \bar{C}, 0\} \geq J(D') \geq \max\{e^{k\epsilon} J(D) - \frac{e^{k\epsilon} - 1}{e^{\epsilon} - 1} \delta \bar{C}, -\bar{C}\}$$

□

We recall Corollary 1.

**Corollary 1** (Lower Bound of $k$ Given $\tau$)**.** *Suppose a randomized mechanism $\mathcal{M}$ satisfies user-level $(\epsilon, \delta)$-DP. Let attack cost function be $C$, the expected attack cost be $J(\cdot)$. In order to achieve $J(D') \leq \frac{1}{\tau} J(D)$ for $\tau \geq 1$ when $0 \leq C(\cdot) \leq \bar{C}$, or achieve $J(D') \leq \tau J(D)$ for $1 \leq \tau \leq -\frac{\bar{C}}{J(D)}$ when $-\bar{C} \leq C(\cdot) \leq 0$, the number of adversarial users should satisfy:*

$$k \geq \frac{1}{\epsilon} \log \frac{(e^{\epsilon} - 1)\, J(D)\tau + \bar{C}\delta\tau}{(e^{\epsilon} - 1)\, J(D) + \bar{C}\delta\tau} \quad \textit{or} \quad k \geq \frac{1}{\epsilon} \log \frac{(e^{\epsilon} - 1)\, J(D)\tau - \bar{C}\delta}{(e^{\epsilon} - 1)\, J(D) - \bar{C}\delta} \quad \textit{respectively.} \tag{4}$$

*Proof.* We first consider $C(\cdot) \geq 0$. According to the lower bound in Theorem 3, when $B'$ and $B$ differ $k$ users, $J(D') \geq e^{-k\epsilon} J(D) - \frac{1 - e^{-k\epsilon}}{e^{\epsilon} - 1} \delta \bar{C}$. Since we require $J(D') \leq \frac{1}{\tau} J(D)$, then $e^{-k\epsilon} J(D) - \frac{1 - e^{-k\epsilon}}{e^{\epsilon} - 1} \delta \bar{C} \leq \frac{1}{\tau} J(D)$. Rearranging gives the result.

Next, we consider $C(\cdot) \leq 0$. According to the lower bound in Theorem 3, when $B'$ and $B$ differ $k$ users, $J(D') \geq e^{k\epsilon} J(D) - \frac{e^{k\epsilon} - 1}{e^{\epsilon} - 1} \delta \bar{C}$. Since we require $J(D') \leq \tau J(D)$, then $e^{k\epsilon} J(D) - \frac{e^{k\epsilon} - 1}{e^{\epsilon} - 1} \delta \bar{C} \leq \tau J(D)$. Rearranging gives the result.

□

We note that all the above robustness certification related proofs are built upon the user-level $(\epsilon, \delta)$-DP property and the Group DP property. According to Definition 2 and Definition 3, the definition of user-level DP and instance-level DP are both induced from DP (Definition 1) despite the different definitions of adjacent datasets. By applying the definition of instance-level $(\epsilon, \delta)$-DP and following the proof steps of Theorem 1, 2, 3 and Corollary 1, we can derive the similar theoretical conclusions for instance-level DP, leading to Theorem 5 to achieve the certifiably robsut FL for free given the DP property.