Federated Model Distillation with Noise-Free Differential Privacy

Lichao Sun[1] , **Lingjuan Lyu**[2*]
[1] Lehigh University, [2] Ant Group
lis221@lehigh.edu, lingjuanlvsmile@gmail.com

Abstract

Conventional federated learning directly averages model weights, which is only possible for collaboration between models with homogeneous architectures. Sharing prediction instead of weight removes this obstacle and eliminates the risk of white-box inference attacks in conventional federated learning. However, the predictions from local models are sensitive and would leak training data privacy to the public. To address this issue, one naive approach is adding the differentially private random noise to the predictions, which however brings a substantial trade-off between privacy budget and model performance. In this paper, we propose a novel framework called FEDMD-NFDP, which applies a Noise-Free Differential Privacy (NFDP) mechanism into a federated model distillation framework. Our extensive experimental results on various datasets validate that FEDMD-NFDP can deliver not only comparable utility and communication efficiency but also provide a noise-free differential privacy guarantee. We also demonstrate the feasibility of our FEDMD-NFDP by considering both IID and non-IID setting, heterogeneous model architectures, and unlabelled public datasets from a different distribution.

1 Introduction

Federated learning (FL) provides a privacy-aware paradigm of model training, which allows a multitude of parties to construct a joint model without directly exposing their private training data [McMahan *et al.*, 2017; Bonawitz *et al.*, 2017; Xu *et al.*, 2021]. Nevertheless, recent works have demonstrated that FL may not always provide sufficient privacy guarantees, as communicating model updates throughout the training process can nonetheless reveal sensitive information [Bhowmick *et al.*, 2018; Melis *et al.*, 2019].

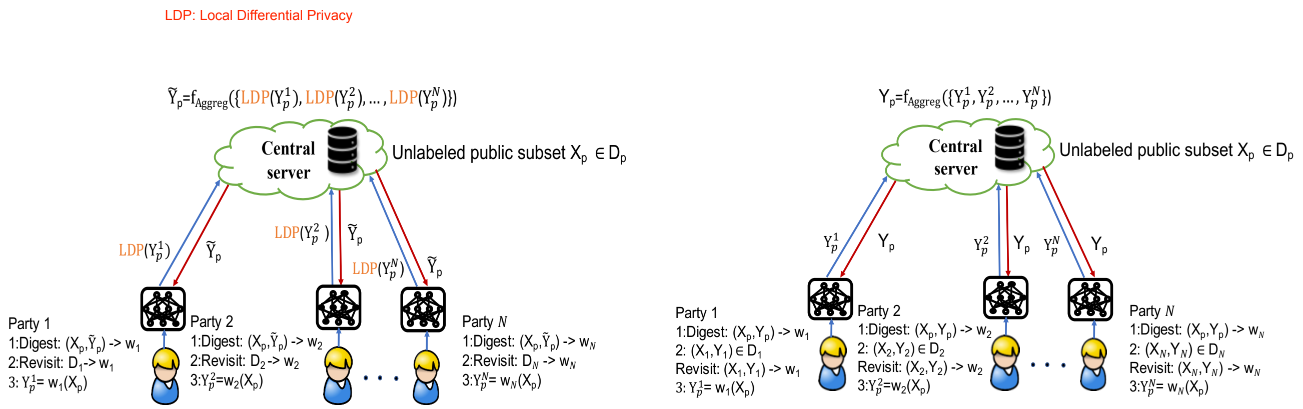
In order to protect training data privacy in FL, various privacy protection techniques have been proposed in the literature [Geyer *et al.*, 2017; McMahan *et al.*, 2018; Bonawitz *et al.*, 2017; Wang *et al.*, 2019b; Zhao *et al.*, 2020; Sun *et al.*, 2020a; Lyu *et al.*, 2020]. From the perspective of differential privacy, most works focus on the centralized differential privacy (CDP) that requires a central trusted party to add noise to the aggregated gradients [Geyer *et al.*, 2017; McMahan *et al.*, 2018]. Moreover, these works are geared to tackle thousands of users for training to converge and achieve an acceptable trade-off between privacy and accuracy [McMahan *et al.*, 2018], resulting in a convergence problem with a small number of parties.

To achieve stronger privacy protection, a few recent works start to integrate local differential privacy (LDP) into federated learning. However, most existing approaches can only support shallow models such as logistic regression and only focus on simple tasks and datasets [Wang *et al.*, 2019b; Zhao *et al.*, 2020]. [Bhowmick *et al.*, 2018] presented a viable approach to large-scale local private model training. Due to the high variance of their mechanism, it requires more than 200 communication rounds and incurs much higher privacy cost, i.e., MNIST ($\epsilon = 500$) and CIFAR-10 ($\epsilon = 5000$). A recent work [Sun *et al.*, 2020a] utilized Local Differential Privacy (LDP) into federated learning. However, in order to achieve a reasonable privacy budget, it uses a splitting and shuffling mechanism that split all parameters of a single model and send them individually to the cloud. This special communication requires tons of communication between clients and clouds.

All the above works considered privacy issues in conventional FL that requires parties to share model weights. Compared with sharing prediction via knowledge transfer, conventional FL system [McMahan *et al.*, 2017; McMahan *et al.*, 2018] suffers from several intrinsic limitations: (1) it requires every party to share their local model weights in each round, thus limiting only to models with homogeneous architectures; (2) sharing model weight incurs a significant privacy issue of local model, as it opens all the internal state of the model to white-box inference attacks; (3) model weight is usually of much higher dimension than model predictions, resulting in huge communication overhead and higher privacy cost.

Inspired by the knowledge transfer algorithms [Buciluă *et al.*, 2006; Hinton *et al.*, 2015], *Federated Model Distillation* (FedMD) shares the knowledge of FL parties' models via their predictions on an unlabeled public set [Li and Wang, 2019]. However, sharing prediction may still leak the privacy of the local data [Papernot *et al.*, 2017]. Currently, there is no reasonable privacy guarantee for sharing model prediction in FL. A naive approach is to add the differentially private

[*] Equal contribution. Order determined by coin toss.


LDP: Local Differential Privacy
$\tilde{Y}_p$=f$_{Aggreg}$({LDP(Y_p^1), LDP(Y_p^2), ..., LDP(Y_p^N)})
Central server
Unlabeled public subset $X_p \in D_p$
LDP(Y_p^1)
$\tilde{Y}_p$
LDP(Y_p^2)
$\tilde{Y}_p$
LDP(Y_p^N)
$\tilde{Y}_p$
Party 1
1:Digest: (X_p,$\tilde{Y}_p$) -> w_1
2:Revisit: D_1-> w_1
3: Y_p^1= w_1(X_p)
Party 2
1:Digest: (X_p,$\tilde{Y}_p$) -> w_2
2:Revisit: D_2-> w_2
3:Y_p^2=w_2(X_p)
Party N
1:Digest: (X_p,$\tilde{Y}_p$) -> w_N
2:Revisit: D_N -> w_N
3:Y_p^N= w_N(X_p)
Y_p=f$_{Aggreg}$({Y_p^1, Y_p^2, ..., Y_p^N})
Central server
Unlabeled public subset $X_p \in D_p$
Y_p^1
Y_p
Y_p^2
Y_p
Y_p^N
Y_p
Party 1
1:Digest: (X_p,Y_p) -> w_1
2: (X_1,Y_1) ∈ D_1
Revisit: (X_1,Y_1) -> w_1
3: Y_p^1= w_1(X_p)
Party 2
1:Digest: (X_p,Y_p) -> w_2
2: (X_2,Y_2) ∈ D_2
Revisit: (X_2,Y_2) -> w_2
3:Y_p^2=w_2(X_p)
Party N
1:Digest: (X_p,Y_p) -> w_N
2: (X_N,Y_N) ∈ D_N
Revisit: (X_N,Y_N) -> w_N
3:Y_p^N= w_N(X_p)

(a) FEDMD-LDP

(b) FEDMD-NFDP

Figure 1: Overview of the naive FEDMD-LDP and our FEDMD-NFDP in each communication round. Each party first updates its model w_i to approach the consensus on the public dataset (Digest); then updates its model w_i on its own sampled subset from private local data (Revisit). Note that the sampled subset $(X_i, Y_i) \in D_i$ is fixed in all rounds.

random noise perturbation to the predictions of local models. However, prior works have shown the significant trade-off between privacy budget and model performance [Bhowmick *et al.*, 2018]. We fill in the above gaps and make the following contributions:

- We propose FEDMD-NFDP, a novel federated model distillation framework with the new proposed noise-free differential privacy (NFDP) mechanism that guarantees each party's privacy without explicitly adding any noise.
- We formally prove that NFDP with both replacement and without replacement sampling strategies can inherently ensure (ϵ, δ)-differential privacy, eliminating noise addition and privacy cost explosion issues explicitly in previous works.
- Extensive experiments on benchmark datasets, various settings (IID and non-IID data distribution), and heterogeneous model architectures, demonstrate that FEDMD-NFDP achieves comparable utility with only a few private samples that are randomly sampled from each party, validating the numerous benefits of our framework.

We remark that sampling can individually serve as a privacy amplification method to tighten the privacy budget of a differentially private algorithm [Balle *et al.*, 2018], which is in sharp contrast with the inherent privacy guarantee of sampling given in this paper.

2 Preliminary

Differential Privacy. DP has become a de facto standard for privacy analysis. DP can either be enforced in a "local" or "global" sense depending on whether the server is trusted. For FL scenarios where data are sourced from multiple parties, while the server is untrusted, DP should be enforced in a "local" manner to enable parties to apply DP mechanisms before data publication, which we term as LDP. Compared with the global model via DP (CDP) [Dwork and Roth, 2014; Abadi *et al.*, 2016], LDP offers a stronger level of protection.

Definition 1. *A randomized mechanism $\mathcal{M}$: $\mathcal{D} \to \mathcal{R}$ with domain $\mathcal{D}$ and range $\mathcal{R}$ satisfies (ϵ, δ)-differential privacy if for all two neighbouring inputs $D, D' \in \mathcal{D}$ and any measurable subset of outputs $S \subseteq \mathcal{R}$ it holds that*

$$\Pr\{\mathcal{M}(D) \in S\} \quad \leq \quad \exp(\epsilon) \cdot \Pr\{\mathcal{M}(D') \in S\} + \delta \ .$$

A formal definition of record-level DP is provided in Def. 1, which bounds the effect of the presence or the absence of a record on the output likelihood within a small factor ϵ. The additive term δ allows that the unlikely responses do not need to satisfy the pure ϵ-DP criterion. In FL, each party can individually apply $\mathcal{M}$ in Definition 1 to ensure record-level LDP.

3 Federated Model Distillation with Noise-Free Differential Privacy

Unlike the existing federated learning algorithms, such as FedAvg [McMahan *et al.*, 2017], FedMD does not force a single global model onto local models. Instead, each local model is updated separately. To support heterogeneous model architectures, we assume that an unlabeled public dataset is available, then parties share the knowledge that they have learned from their training data (their model predictions) in a succinct, black-box and model agnostic manner. To protect local model predictions, each party can explicitly apply LDP mechanisms by adding noise to their local model predictions, as shown in FEDMD-LDP (Figure 1 (a)), or adopt data sampling before training, which inherently ensures LDP of the sampled subset, and the follow-up local model predictions as per the post-processing property of DP [Dwork and Roth, 2014], as demonstrated in FEDMD-NFDP(Figure 1 (b)).

It should be noted that FEDMD-LDP requires each party to explicitly inject noise to ensure DP individually before releasing their local model knowledge to the server. The privacy cost will accumulate as per the dimension of the shared knowledge ($|Y_p| * class$), as well as the communication rounds, resulting in huge privacy costs. Here $|Y_p|$ is the number of the chosen public set and $class$ refers to the class number. In contrast, our FEDMD-NFDP inherently ensures that the released local model knowledge by each party is differentially private via random data sampling process, as indicated in Theorem 1 and Theorem 2.

Algorithm 1 FEDMD-NFDP. Initialization phase does not involve collaboration. D_i and w_i are local dataset and model parameters from i-th party. $Y_p^i[t]$ is the prediction from i-th party on the chosen public subset $X_p \in D_p$ in round t.

```
1:                    **Initialization phase**
2:  Initializes each party $i \in [N]$ with the same pretrained
    model, selects subset $(X_i, Y_i) \in D_i$, and updates their
    weights in parallel:
3:  for $t \in [T_1]$ epochs do                        1，基于本地data训
4:      Update $w_i \leftarrow$ TRAIN $(w_i, X_i, Y_i)$          练local model
5:  end for
6:  Server randomly samples a public subset $X_p[0] \in D_p$
7:  Send $Y_p^i[0] = $ PREDICT$(w_i; X_p[0])$ to the server

8:                    **Collaboration phase**
9:  $Y_p[0] = f_{\text{Aggreg}}(\{Y_p^{i \in [N]}[0]\})$ ▷ Initial aggregation at the
    server
10: for $t \in [R]$ communication rounds do
11:     Server randomly samples a public subset $X_p[t+1] \in$
    $D_p$
12:     for $i \in [N]$ parties do          ▷ Each party updates local
    weight $w_i$ in parallel                2，每个client基于server
13:         for $j \in [T_2]$ epochs do         avg过的data 做KD
14:             Digest: $w_i \leftarrow$ TRAIN $(w_i, X_p[t], Y_p[t])$
15:         end for
16:         for $j \in [T_3]$ epochs do
17:             Revisit: $w_i \leftarrow$ TRAIN $(w_i, X_i, Y_i)$
18:         end for
19:         Send $Y_p^i[t+1] = $ PREDICT$(w_i; X_p[t+1])$ to the
    server
20:     end for
21:     $Y_p[t+1] = f_{\text{Aggreg}}(\{Y_p^{i \in [N]}[t+1]\})$     ▷ Prediction
    aggregation at the server
22: end for
```

每个public data的predict会随着local model的更新而变化

Algorithm 1 describes our FEDMD-NFDP algorithm, which consists of two training phases: (1) during initialization phase, every party i updates its local model weights w_i on a randomly sampled subset $(X_i, Y_i) \in D_i$ from local private training data D_i for T_1 times without any collaboration; (2) during collaboration phase, parties share the knowledge of their local models via their predictions on a subset of public data, X_p. In each round of the collaboration phase, the detailed procedure proceeds as follows:

- Each party uses its local model weights w_i to compute prediction Y_p^i for X_p and shares them with the server.
- The server aggregates the predictions (separately for each public record), i.e., computes $Y_p = f_{\text{Aggreg}}(Y_p^1, \cdots, Y_p^N)$, and sends Y_p to all parties for the next round's local training; f_{Aggreg} is an aggregation algorithm, which is average function throughout this work.
- Each party first updates its local model weights w_i by training on the soft-labeled public data (X_p, Y_p) to approach the consensus on the public dataset (Digest); then training on its previously sampled local subset (Revisit).

Softmax avg

In addition, Algorithm 1 can also support the implementation of FEDMD-LDP. The only difference is the output $Y_p^i[t+1]$ (line 19) should be perturbed by the differentirally private random noise, which can be randomly sampled from either Laplace or Gaussian distribution.

4 Theoretical Analysis

In this work, we consider record-level DP for each party. Below, we formally stated that NFDP with random sampling from each party's training dataset satisfies differential privacy guarantee for each party. In particular, random sampling without replacement and with replacement are two most common sampling strategies, and we prove the (ϵ, δ)-differential privacy for both of them.

Theorem 1. *[NFDP mechanism: (ϵ, δ)-differential privacy of sampling without replacement] Given a training dataset of size n, sampling without replacement achieves $(\ln \frac{n+1}{n+1-k}, \frac{k}{n})$-differential privacy, where k is the subsample size.*

Theorem 2. *[NFDP mechanism: (ϵ, δ)-differential privacy of sampling with replacement] Given a training dataset of size n, sampling with replacement achieves $((k \ln \frac{n+1}{n}, 1 - \left(\frac{n-1}{n}\right)^k)$-differential privacy, where k is the subsample size.*

Lemma 1. *Algorithm 1 using sampling with replacement is consistently more private than using sampling without replacement for any $n > 0$ and $0 < k \leq n$.*

All the related proofs of lemma and theorems can be referred to the Appendix. The nice property of NFDP with random sampling once and the post-processing property of differential privacy [Dwork and Roth, 2014] removes the privacy dependence on the number of queries on the public dataset, allowing a more practical deployment of our NFDP in FEDMD.

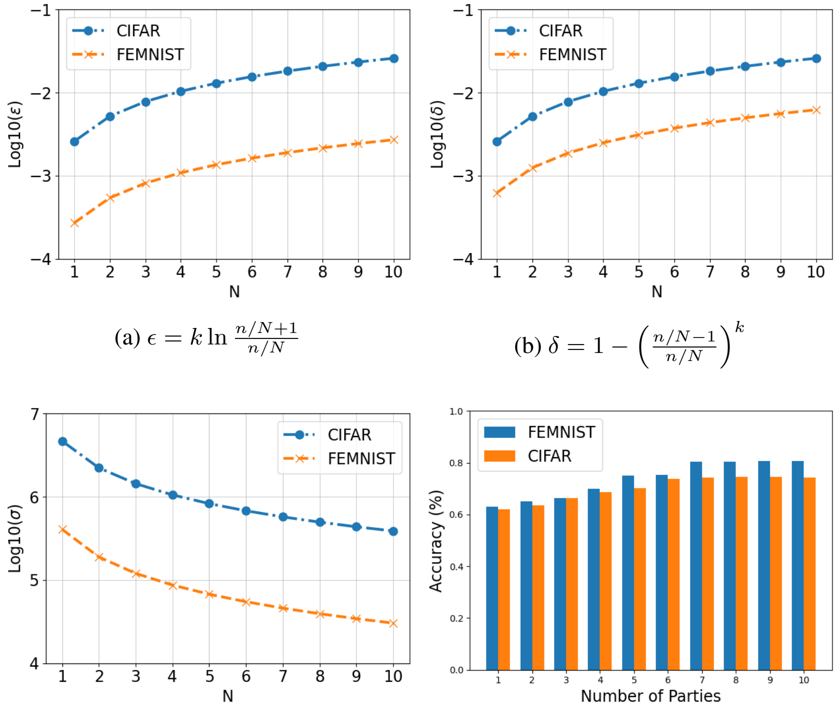
5 Experimental Evaluation

In the experiment, we evaluate on paired datasets, i.e., MNIST/FEDMNIST and CIFAR-10/CIFAR-100. For MNIST/FEMNIST, the public data is the MNIST, and the private data is a subset of the Federated Extended MNIST (FEMNIST) [Caldas *et al.*, 2018], which is built by partitioning the data in Extended MNIST based on the writer of the digit/character. In the IID scenario, the private dataset of each party is drawn randomly from FEMNIST. In the non-IID scenario, each party only has letters written by a single writer, and the task is to classify letters by all writers.

For CIFAR-10/CIFAR-100, the public dataset is the CIFAR-10, and the private dataset is a subset of the CIFAR-100 [Krizhevsky *et al.*, 2009], which has 100 subclasses that fall under 20 superclasses, e.g., bear, leopard, lion, tiger, and wolf belong to large carnivores [Li and Wang, 2019]. In the

Task	Public	Private
IID	MNIST	FEMNIST letters [a-f] classes
IID	CIFAR-10	CIFAR-100 subclasses [0,2,20,63,71,82]
Non-IID	MNIST	FEMNIST letters from one writer
Non-IID	CIFAR-10	CIFAR-100 superclasses [0-5]

Table 1: Summary of datasets



(a) $\epsilon = k \ln \frac{n/N+1}{n/N}$

(b) $\delta = 1 - \left(\frac{n/N-1}{n/N}\right)^k$

(c) $\sigma = c_2 \frac{\sqrt{T \log(1/\delta)}}{\epsilon}$

(d) Accuracy v.s. Number of Parities

Figure 2: [a-c]: Log 10 scale of ϵ, δ for each party in FEDMD-NFDP/FEDMD-LDP v.s. the number of parities N. Note that, we use fixed $k = 3$. Here T is the total number of queries and we have $T = 100000$ in 20 round communications, i.e., 5000 queries on the public dataset in each round. n is the size of private datasets owned by all parties, which is 28800 for FEMNIST and 3000 for CIFAR-10. The log 10 scale of δ of FEDMD-LDP is calculated by ϵ and δ, which are from (a) and (b) by the corresponding N. [d]: accuracy v.s. number of parities.

IID scenario, each party is required to classify test images into correct subclasses. The non-IID scenario is more challenging: each party has data from one subclass per superclass but needs to classify generic test data into the correct superclasses. Therefore, it necessitates knowledge sharing among parties.

Each party's local model is two or three-layer deep neural networks for both MNIST/FEMNIST and CIFAR-10/CIFAR-100. All experiments are implemented by using Pytorch. A single GPU NVIDIA Tesla V100 is used in the experiments. FEMNSIT and CIFAR-10 can be done within an hour at $N = 10$ parties. A summary of the public and private datasets used in this paper is provided in Table 1.

In each communication round, we use a subset of size 5000 that is randomly selected from the entire public dataset. We empirically validate FEDMD-NFDP largely reduces the communication cost without degrading the utility. The number of training epochs in Algorithm 1 and the batch size in the Digest and the Revisit phase may impact the stability of the learning process. We empirically choose $R = 20, T_1 = 20, T_2 = 2, T_3 = 1$ via grid search. We initialize all parties with the same pre-trained model on some labelled data in the same domain. For example, parties training on the private FEMNIST are initialized with the same pre-trained model on some labelled MNIST data.

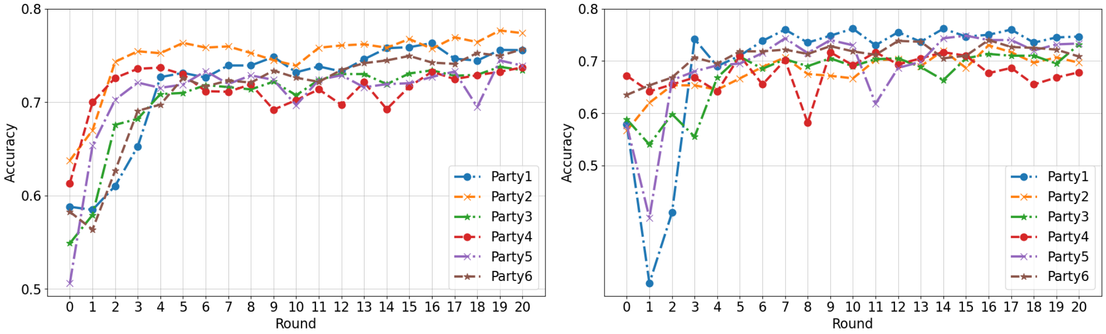
5.1 Baselines

We demonstrate the effectiveness of our proposed FEDMD-NFDP by comparison with the following three frameworks. We omit the comparison with FedAvg as it delivers similar utility as the *Centralized* framework.

1. *Non-private Federated Model Distillation* (FedMD-NP) framework: all parties train on all their local private data, collaborate the public data distillation, and use the aggregation feedbacks to update the local model as same as FedMD. It should be noted that there is no privacy guarantee in this framework.
2. *Centralized* framework: the private data of all parties were pooled into a centralized server to train a global model. We use this as an utility upper bound.
3. FEDMD-LDP framework: FEDMD-LDP requires each party to explicitly add Gaussian noise to locally ensure (ϵ, δ)-DP before releasing their local model knowledge to the server.

5.2 Performance Analysis

Evaluation on privacy budget ϵ. It can be observed from Figure 2(a) that FEDMD-NFDP can ensure strong privacy protection during training and communication. For each party in FEDMD-NFDP we fixed $k = 3$, which means we only



(a) FEMNIST

(b) CIFAR-10

Figure 3: Accuracy v.s. Communication Rounds

FEMNIST	k	ϵ	δ	Accuracy
FEDMD-NP	2880	$+\infty$	1	96.15%
Centralized	2880	$+\infty$	1	98.00%
FedMD-NFDP	16	0.0027	0.0062	80.64%
FedMD-NFDP	60	0.0090	0.0206	88.06%
FedMD-NFDP	300	0.0452	0.0989	93.56%
FedMD-NFDP	2880	0.4342	0.6321	96.63%

CIFAR-10	k	ϵ	δ	Accuracy
FedMD-NP	300	$+\infty$	1	86.88%
Centralized	300	$+\infty$	1	88.83%
FedMD-NFDP	16	0.0260	0.0583	74.40%
FedMD-NFDP	60	0.0867	0.1815	81.58%
FedMD-NFDP	120	0.1734	0.3301	83.57%
FedMD-NFDP	300	0.4336	0.6327	87.38%

Table 2: Comparisons with All Baselines

randomly sample three private data points from each private local dataset. While more parties participate in the training and communication, the number of each private local dataset becomes smaller, since each party's data size equals to the total number of private data n divided by the number of parties N. Due to that, when we increase the number of the parties, the privacy budget ϵ will increase even with a fixed random sample size k. However, the ϵ is still very small, its log 10 scale is close to -2 for CIFAR-10 and -3 for FEMNIST, when the number of parties N is 10.

Evaluation on δ : Similar to ϵ, δ is defined based on Theorem 2. Figure 2(b) shows that the increasing number of parties will increase the δ. The δ is small, its log 10 scale is close to -2 for CIFAR-10 and -3 for FEMNIST, when the number of parties N is 10. Note that, in real life, the private data are collected by each party independently, so more parties would not decrease the local data size in practice.

Evaluation on σ in FedMD-LDP. Besides our proposed approach, the most naive solution is FedMD-LDP. Unlike FedMD-NFDP, Fed-LDP can use all the private dataset for training and only protect the distillation information on the public dataset, as shown in Figure 1(a). However, FedMD is cursed by a massive number of queries and multi-round communications as per the sequential composition in DP [Dwork and Roth, 2014]. Given the same ϵ, δ as in FedMD-NFDP, the σ is a huge number from 4 to 7 in the log 10 scale, compromising the utility of the original information. Due to this reason, we do not report the experimental results of FedMD-LDP in this work, since the prediction results are close to random guess with a huge noise scale of σ for both FEMNIST and CIFAR-10. The only way to maintain utility is to set a very large ϵ and δ for FedMD-LDP, but it will result in meaningless privacy guarantee.

		N	logits	softmax	argmax
FedMD-NFDP	FEMNIST	5	74.99%	75.02%	75.58%
		10	80.74%	80.64%	81.58%
	CIFAR-10	5	69.79%	70.02%	70.01%
		10	74.55%	74.88%	75.12%

Table 3: Different Distillation Approaches

Evaluation on model convergence. Figure 3 presents the accuracy trajectories of each party in our FedMD-NFDP. As shown in Figure 3, all parties can converge to a decent performance within 20 communication rounds, largely reducing communication cost. Due to the complexity of the tasks, FEMNIST shows a slightly better performance than CIFAR-10.

Evaluation on distillation approaches. In the original FedMD [Li and Wang, 2019], they use logits as the distillation approach. However, in our implementation, besides the logits, we also build the distillation with softmax and argmax approaches. Softmax approach returns the soft labels and argmax approach returns the hard label for each query. From Table 3, we can see that the results did not differ too much across different approaches in general. However, we recommend argmax label approach for both FedMD and our system. There are two main reasons: (1) argmax shows slightly better performance than the other two approaches; (2) more importantly, argmax can save much communication cost of each query. Both softmax and logits need to send the float vectors, but argmax only needs to send the integer during communication.

		N	IID	Non-IID
FedMD-NFDP	FEMNIST	10	81.58%	78.36%
	CIFAR-10	10	75.12%	53.18%

Table 4: IID vs Non-IID

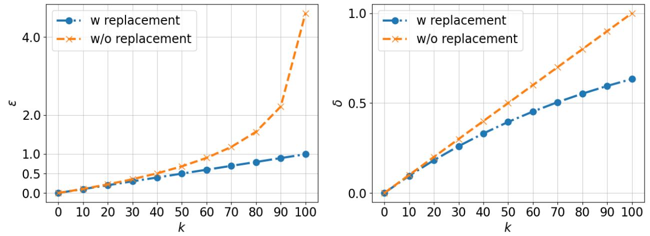
		k	w replacement	w/o replacement
FedMD-NFDP	FEMNIST	300	93.56%	93.63%
	CIFAR-10	60	81.58%	81.13%

Table 5: With replacement vs without replacement sampling

Evaluation on IID and Non-IID distributions. Table 4 shows the evaluation on Non-IID dataset. FedMD-NFDP can achieve a superior performance with a low privacy cost because of the noise-free differential privacy mechanism. Compared with IID, non-IID is definitely more challenging due to the incomplete data information of each class. The detailed settings of our experiments are well introduced in the appendix. From the results, we can see that FEMNIST can do better on Non-IID tasks. The main reason is for CIFAR-10 task, we only use one sub-class during training which hardly train the local model well for other classification. For example, one party has the wolfs dataset during training, but it is hardly to help classify lions correctly as large carnivores.

Evaluation on number of parities. It is not hard to see that more parties can help improve the utility in the federated learning. Figure 2(d) shows that more collaboration can effectively improve the performance of each local model. Although we only have 10 parties in total, but it already can achieve a good performance on complex image dataset, i.e. CIFAR-10. Compared with FedMD-NFDP, previous private collaborate learning framework requires at least hundreds of parties to be robust to the noise perturbation from previous DP and LDP mechanisms [Papernot *et al.*, 2017; Geyer *et al.*, 2017; Bhowmick *et al.*, 2018; Sun *et al.*, 2020a]. In that sense, FedMD-NFDP also is the work that firstly provides a private collaboration system with a small number of parties.

Comparison on with replacement and without replacement samplings. The results in Figure 4 demonstrate the correctness of the lemma 1. It is not hard to see that, while we fix the number of the parties, the increasing number of sampling examples will require a larger privacy budget. Meanwhile, the without replacement sampling spends much more privacy loss than with replacement sampling with the same size of the sampled subset. Furthermore, we evaluate the performance of both sampling strategies, and the results are shown in Table 2. From the results, we find there is no much difference between these two sampling strategies. In summary, we recommend using with replacement sampling for private model training, since it costs less privacy budget than without replacement sampling but achieves the same performance.



(a) n=100

(b) n=100

Figure 4: ϵ and δ comparison between without (i.e., w/o) replacement and with (i.e., w) replacement. k is the size of the sampled subset.

Comparison with baselines. Table 2 shows that FedMD-NFDP can achieve a superior performance with a low privacy cost because of the noise-free differential privacy mechanism. For all methods, we report the average accuracy of 10 parities. When we increase the privacy budget, it can even outperform the FEDMD-NP approach and be comparable to the Centralized approach. Meanwhile, we observe that given a very small (ϵ,δ) with $k = 16$, we can still achieve a competitive performance on CIFAR-10. None of the previous works related to differential privacy in federated learning [Geyer *et al.*, 2017; Bhowmick *et al.*, 2018; Sun *et al.*, 2020a] can achieve comparable performance on CIFAR-10 with such a small ϵ as ours.

Note that, we did not list the utility of the FedMD-LDP, since the utility of each model is close to random guess while we use the same privacy budget as FedMD-NFDP. Put in another way, FedMD-LDP requires an extremely large noise scale to ensure the same level of (ϵ, δ)-DP as FedMD-NFDP, resulting in poor utility. However, it no longer becomes a problem in FedMD-NFDP. After the random sampling with replacement approach in the first step, FedMD-NFDP is already $((k \ln \frac{n+1}{n}, 1 - \left(\frac{n-1}{n}\right)^k)$-differential privacy due to the post-processing property [Dwork and Roth, 2014].

Comparison with previous works. Our results are competitive comparing to the previous works that adopt CDP and LDP. Currently, most of the popular differential privacy approaches, such as DP-SGD [Abadi *et al.*, 2016], PATE [Papernot *et al.*, 2017], are cursed by the number of queries and communication rounds when they are applied to FL. Since each query touches the private information, the large number of queries will cost huge privacy budget in FL with multi-round communications.

6 Discussion

Diversity of public dataset. In federated model distillation, the public dataset requires careful deliberation and even prior knowledge on parties' private datasets. The distribution of the unlabeled public dataset could either match, or differ from the distribution of the private training data available at the parties to some degree. It needs to know how the gap widens when the public dataset becomes more different from the training dataset, the worst case could be from different domains without any overlap. There is also a potential to use the synthetic data from a pre-trained generator (e.g. GAN) as public data to alleviate potential limitations (e.g. acquisition, storage) of real unlabeled datasets. This may open up numerous possibilities for effective and efficient model distillation.

Diversity of local models. FedMD-NFDP allows local models in FL to not only differ in model structure, size, but also numerical precision, offering great benefit for the Internet of Things (IoT) that involves edge devices with diverse hardware configurations and computing resources.

Weighted aggregation. The aggregation step in Algorithm 1 is based on directly averaging of parties' predictions, i.e., f_{Aggreg} corresponds to the average function with equal weight $1/N$, where N is the number of parties. However, parties may contribute to the consensus differently, especially in the extreme cases of model and data heterogeneity. Allocating all the parties with the same weight may negatively impact system utility. We remark that there may exist more advanced weighted average algorithms that can further boost utility. These weights can be used to quantify the contributions from local models, and play important roles in dealing with extremely different models.

Limitations. Based on the privacy analysis of NFDP mechanisms, for both with replacement and without replacement sampling strategies, we require each local party has an adequate size of the dataset. For example, if each local data only has one label, our mechanism can not protect any privacy due to the private training data's size limitation.

In this case, NFDP could be very useful for three scenarios. First, one party is required to provide machine learning as a service (MLaaS) for others, i.e., teacher-student learning framework. While this party contains a large size of private data, NFDP could help it train a privacy guaranteed model. Second, one party has a large dataset, but the data itself is lack of diversity. The model still can not achieve a good performance due to the data diversity limitation. In this case, they need to communicate with others for a better model utility, and we can use NFDP to protect them during their communications. Finally, some learning tasks only require a small fraction of the private training data, such as FedMD [Vinyals *et al.*, 2016; Snell *et al.*, 2017]. NFDP can perform well on these tasks with adequate privacy protection. Besides FedMD, traditional one-shot learning and few-short learning tasks are also suitable to use NFDP for privacy protection for the same reason.

7 Related Work

7.1 Differential Privacy

Differential privacy [Dwork *et al.*, 2006; Dwork and Roth, 2014] provides a mathematically provable framework to design and evaluate a privacy protection scheme. Recently, differential privacy has been applied to FL [Bhowmick *et al.*, 2018; Geyer *et al.*, 2017; McMahan *et al.*, 2018]. Previous works mostly focus on the centralized differential privacy mechanism that requires a trusted party [Geyer *et al.*, 2017; McMahan *et al.*, 2018; Yang *et al.*, 2021], or local differential privacy, in which each user randomizes its gradients locally before sending it to an untrusted aggregator [Sun *et al.*, 2020a], or the hybrid mechanism by combining distributed differential privacy (DDP) with crypto system [Lyu, 2020].

7.2 Knowledge Distillation

Knowledge distillation [Buciluǎ *et al.*, 2006; Hinton *et al.*, 2015] is originally designed to extract class probability produced by a large DNN or an ensemble of DNNs to train a smaller DNN with marginal utility loss. It also offers a powerful tool to share knowledge of a model through its predictions. Knowledge of ensemble of teacher models has been used to train a student model in previous works [Hamm *et al.*, 2016; Papernot *et al.*, 2017; Wang *et al.*, 2019a; Sun *et al.*, 2020b]. For example, Papernot el. al. [Papernot *et al.*, 2017] proposed PATE, a centralized learning approach that uses ensemble of teachers to label a subset of unlabeled public data in a differentially private manner, then trains a student in a semi-supervised fashion [Dwork and Roth, 2014]. We remark that our focus is fundamentally different from the setting of PATE, which requires a trusted aggregator to aggregate the prediction label made by the teacher ensemble and conduct DP mechanisms.

7.3 Federated Learning

Federated learning (FL) has emerged as a promising collaboration paradigm by enabling a multitude of parties to jointly construct a global model without exposing their private training data. In FL, parties do not need to explicitly share their training data, they have full autonomy for their local data. FL generally comes in two forms [McMahan *et al.*, 2017]: FedSGD, in which each client sends every SGD update to the server, and FedAVG, in which clients locally batch multiple iterations of SGD before sending updates to the server, which is more communication efficient.

More recently, FedMD [Li and Wang, 2019] and Cronus [Chang *et al.*, 2019] attempted to apply knowledge distillation to FL by considering knowledge transfer via model distillation, in which, the logits on an unlabeled public dataset from parties' models are averaged. In FedMD, each model is first trained on the public data to align with public logits, then on its own private data. In contrast, Cronus mixes the public dataset (with soft labels) and local private data, then trains local models simultaneously. One obvious benefit of sharing logits is the reduced communication costs, without significantly sacrificing utility. However, both works did not offer any theoretical privacy guarantee for sharing model prediction.

8 Conclusion

In this work, we formulate a new federated model distillation framework with noise-free differential privacy guarantee for each party. We formally prove that NFDP both with replacement and without replacement sampling can inherently ensure (ϵ, δ)-differential privacy, eliminating explicitly noise addition and privacy cost explosion issues in the previous works. Empirical results on various datasets, settings, and heterogeneous model architectures demonstrate that our framework achieves comparable utility by using only a few private samples that are randomly sampled from each party, confirming the effectiveness and superiority of our framework.

In the future, we hope NFDP could support more privacy-preserving machine learning methods, such as semi-supervised learning, pre-training learning, meta-learning, and few-shot learning. Another direction is that we could optimize the random sampling approach with the advanced data analysis for a better promising and practical privacy guarantee mechanism. Last but not least, we could use the NFDP mechanism with advanced machine learning methods to support more applications in real life, such as natural language processing, graph analysis, and medical diagnosis.

References

[Abadi *et al.*, 2016] Martín Abadi, Andy Chu, Ian Goodfellow, H Brendan McMahan, Ilya Mironov, Kunal Talwar, and Li Zhang. Deep learning with differential privacy. In *CCS*, pages 308–318, 2016.

[Balle *et al.*, 2018] Borja Balle, Gilles Barthe, and Marco Gaboardi. Privacy amplification by subsampling: Tight analyses via couplings and divergences. In *NIPS*, 2018.

[Bhowmick *et al.*, 2018] Abhishek Bhowmick, John Duchi, Julien Freudiger, Gaurav Kapoor, and Ryan Rogers. Protection against reconstruction and its applications in private federated learning. *CoRR, arXiv:1812.00984*, 2018.

[Bonawitz *et al.*, 2017] Keith Bonawitz, Vladimir Ivanov, Ben Kreuter, Antonio Marcedone, H Brendan McMahan, Sarvar Patel, Daniel Ramage, Aaron Segal, and Karn Seth. Practical secure aggregation for privacy-preserving machine learning. In *CCS*, pages 1175–1191, 2017.

[Buciluǎ *et al.*, 2006] Cristian Buciluǎ, Rich Caruana, and Alexandru Niculescu-Mizil. Model compression. In *KDD*, 2006.

[Caldas *et al.*, 2018] Sebastian Caldas, Peter Wu, Tian Li, Jakub Konečnỳ, H Brendan McMahan, Virginia Smith, and Ameet Talwalkar. Leaf: A benchmark for federated settings. *arXiv preprint arXiv:1812.01097*, 2018.

[Chang *et al.*, 2019] Hongyan Chang, Virat Shejwalkar, Reza Shokri, and Amir Houmansadr. Cronus: Robust and heterogeneous collaborative learning with black-box knowledge transfer. *CoRR, arXiv:1912.11279*, 2019.

[Dwork and Roth, 2014] Cynthia Dwork and Aaron Roth. The algorithmic foundations of differential privacy. *FOCS*, 2014.

[Dwork *et al.*, 2006] Cynthia Dwork, Krishnaram Kenthapadi, Frank McSherry, Ilya Mironov, and Moni Naor. Our data, ourselves: Privacy via distributed noise generation. In *Eurocrypt*. Springer, 2006.

[Geyer *et al.*, 2017] Robin C Geyer, Tassilo Klein, and Moin Nabi. Differentially private federated learning: A client level perspective. *CoRR, arXiv:1712.07557*, 2017.

[Hamm *et al.*, 2016] Jihun Hamm, Yingjun Cao, and Mikhail Belkin. Learning privately from multiparty data. In *International Conference on Machine Learning*, 2016.

[Hinton *et al.*, 2015] Geoffrey Hinton, Oriol Vinyals, and Jeff Dean. Distilling the knowledge in a neural network. *arXiv preprint arXiv:1503.02531*, 2015.

[Krizhevsky *et al.*, 2009] Alex Krizhevsky, Geoffrey Hinton, et al. Learning multiple layers of features from tiny images. 2009.

[Li and Wang, 2019] Daliang Li and Junpu Wang. Fedmd: Heterogenous federated learning via model distillation. *arXiv preprint arXiv:1910.03581*, 2019.

[Lyu *et al.*, 2020] Lingjuan Lyu, Han Yu, Xingjun Ma, Lichao Sun, Jun Zhao, Qiang Yang, and Philip S Yu. Privacy and robustness in federated learning: Attacks and defenses. *arXiv preprint arXiv:2012.06337*, 2020.

[Lyu, 2020] Lingjuan Lyu. Lightweight crypto-assisted distributed differential privacy for privacy-preserving distributed learning. In *IJCNN*. IEEE, 2020.

[McMahan *et al.*, 2017] Brendan McMahan, Eider Moore, Daniel Ramage, Seth Hampson, and Blaise Aguera y Arcas. Communication-efficient learning of deep networks from decentralized data. In *AIS*, 2017.

[McMahan *et al.*, 2018] H Brendan McMahan, Daniel Ramage, Kunal Talwar, and Li Zhang. Learning differentially private recurrent language models. In *ICLR*, 2018.

[Melis *et al.*, 2019] Luca Melis, Congzheng Song, Emiliano De Cristofaro, and Vitaly Shmatikov. Exploiting unintended feature leakage in collaborative learning. In *SP*, 2019.

[Papernot *et al.*, 2017] Nicolas Papernot, Martín Abadi, Ulfar Erlingsson, Ian Goodfellow, and Kunal Talwar. Semi-supervised knowledge transfer for deep learning from private training data. In *ICLR*, 2017.

[Snell *et al.*, 2017] Jake Snell, Kevin Swersky, and Richard Zemel. Prototypical networks for few-shot learning. In *NIPS*, 2017.

[Sun *et al.*, 2020a] Lichao Sun, Jianwei Qian, Xun Chen, and Philip S Yu. Ldp-fl: Practical private aggregation in federated learning with local differential privacy. *arXiv preprint arXiv:2007.15789*, 2020.

[Sun *et al.*, 2020b] Lichao Sun, Yingbo Zhou, Philip S Yu, and Caiming Xiong. Differentially private deep learning with smooth sensitivity. *arXiv preprint arXiv:2003.00505*, 2020.

[Vinyals *et al.*, 2016] Oriol Vinyals, Charles Blundell, Timothy Lillicrap, Daan Wierstra, et al. Matching networks for one shot learning. In *NIPS*, 2016.

[Wang *et al.*, 2019a] Ji Wang, Weidong Bao, Lichao Sun, Xiaomin Zhu, Bokai Cao, and S Yu Philip. Private model compression via knowledge distillation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pages 1190–1197, 2019.

[Wang *et al.*, 2019b] Ning Wang, Xiaokui Xiao, Yin Yang, Jun Zhao, Siu Cheung Hui, Hyejin Shin, Junbum Shin, and Ge Yu. Collecting and analyzing multidimensional data with local differential privacy. In *ICDE*. IEEE, 2019.

[Xu *et al.*, 2021] Xiaohang Xu, Hao Peng, Lichao Sun, Md Zakirul Alam Bhuiyan, Lianzhong Liu, and Lifang He. Fedmood: Federated learning on mobile health data for mood detection. *arXiv preprint arXiv:2102.09342*, 2021.

[Yang *et al.*, 2021] Carl Yang, Haonan Wang, Ke Zhang, Liang Chen, and Lichao Sun. Secure deep graph generation with link differential privacy, 2021.

[Zhao *et al.*, 2020] Yang Zhao, Jun Zhao, Mengmeng Yang, Teng Wang, Ning Wang, Lingjuan Lyu, Dusit Niyato, and Kwok Yan Lam. Local differential privacy based federated learning for internet of things. *arXiv preprint arXiv:2004.08856*, 2020.

Appendix

In this appendix, we first prove the privacy guarantee of random sampling without the replacement and then with the replacement.

Theorem 1. *[NFDP mechanism: (ϵ, δ)-differential privacy of sampling without replacement] Given a training dataset of size n, sampling without replacement achieves $(\ln \frac{n+1}{n+1-k}, \frac{k}{n})$-differential privacy, where k is the subsample size.*

Proof. D and D' are two neighbouring datasets in the data space $\mathcal{D}$. $|D| = n$ is the size of the dataset. There are two cases including $D = D' \cup \{u\}$ and $D' = D \cup \{u\}$, where u is the the additional sample. Let $\mathcal{M}$ be the random sample mechanism that randomly returns a subset of the data without replacement here. Let $\mathcal{S}$ denotes the all subsets in the joint domain of $\mathcal{M}(D)$ and $\mathcal{M}(D')$. Then, we use $\Gamma(D)$, $\Gamma(D')$ denote all subsets of $\mathcal{M}(D)$ and $\mathcal{M}(D')$ respectively. $S \in \mathcal{S}$ is a subset in the domain, where $|S| = k$ denotes the size of the subset. Then, for a random subset S, we have,

$$\Pr(\mathcal{M}(D) = S) = \begin{cases} \frac{1}{\binom{|D|}{k}}, & \text{if } S \in \Gamma(D), \\ 0, & \text{otherwise.} \end{cases} \tag{1}$$

$$\Pr(\mathcal{M}(D') = S) = \begin{cases} \frac{1}{\binom{|D'|}{k}}, & \text{if } S \in \Gamma(D'), \\ 0, & \text{otherwise.} \end{cases} \tag{2}$$

Case 1 ($D' = D \cup \{u\}$)**:** Due to $D \subseteq D'$, then we have,

$$\Pr(\mathcal{M}(D) \in \Gamma(D)) = 1, \tag{3}$$

$$\Pr(\mathcal{M}(D') \in \Gamma(D)) = \frac{\binom{|D|}{k}}{\binom{|D'|}{k}}. \tag{4}$$

Let R is a random subset of $\mathcal{S}$ and R is composed by two disjoint subsets, i.e., $R = R_D \cup R_{D' \setminus D}$, where $R_D \subseteq \Gamma(D)$ and $R_{D' \setminus D} \in \Gamma(D') \setminus \Gamma(D)$. Then, we have

$$\Pr(\mathcal{M}(D) \in R) \tag{5}$$

$$= \Pr(\mathcal{M}(D) \in R_D) + \Pr(\mathcal{M}(D) \in R_{D' \setminus D}) \tag{6}$$

$$= \Pr(\mathcal{M}(D) \in R_D) + 0 \tag{7}$$

$$= \Pr(\mathcal{M}(D) \in R_D) \tag{8}$$

$$= \Pr(\mathcal{M}(D') \in R_D) \cdot \frac{\binom{|D'|}{k}}{\binom{|D|}{k}} \tag{9}$$

$$= \Pr(\mathcal{M}(D') \in R_D) \cdot \frac{\binom{n+1}{k}}{\binom{n}{k}} \tag{10}$$

$$= \Pr(\mathcal{M}(D') \in R_D) \cdot \frac{n+1}{n+1-k} \tag{11}$$

$$\leq \Pr(\mathcal{M}(D') \in R) \cdot \frac{n+1}{n+1-k} \tag{12}$$

$$\tag{13}$$

Case 2 ($D = D' \cup \{u\}$)**:** Due to $D' \subseteq D$, then we have,

$$\Pr(\mathcal{M}(D) \in \Gamma(D')) = \frac{\binom{|D'|}{k}}{\binom{|D|}{k}}, \tag{14}$$

$$\Pr(\mathcal{M}(D') \in \Gamma(D')) = 1. \tag{15}$$

Let P is a subset of $\Gamma(D) \setminus \Gamma(D')$, then we have

$$\Pr(\mathcal{M}(D) \in P) \leq \Pr(\mathcal{M}(D) \in \Gamma(D) \setminus \Gamma(D')), \tag{16}$$

$$\leq 1 - \frac{\binom{n-1}{k}}{\binom{n}{k}} = \frac{k}{n}. \tag{17}$$

Let R is a random subset of $\mathcal{S}$ and R is composed by two disjoint subsets, i.e., $R = R_{D'} \cup R_{D \setminus D'}$, where $R_{D'} \subseteq \Gamma(D')$ and $R_{D \setminus D'} \subseteq \Gamma(D) \setminus \Gamma(D')$. Then, we have

$$\Pr(\mathcal{M}(D) \in R) \tag{18}$$

$$= \Pr(\mathcal{M}(D) \in R_{D'}) + \Pr(\mathcal{M}(D) \in R_{D \setminus D'}) \tag{19}$$

$$\leq \Pr(\mathcal{M}(D) \in R_{D'}) + \frac{k}{n} \tag{20}$$

$$\leq \Pr(\mathcal{M}(D') \in R_{D'}) \cdot \frac{\binom{|D'|}{k}}{\binom{|D|}{k}} + \frac{k}{n} \tag{21}$$

$$\leq \Pr(\mathcal{M}(D') \in R_{D'}) \cdot \frac{\binom{n-1}{k}}{\binom{n}{k}} + \frac{k}{n} \tag{22}$$

$$\leq \Pr(\mathcal{M}(D') \in R_{D'}) \cdot \frac{n-k}{n} + \frac{k}{n} \tag{23}$$

$$\leq \Pr(\mathcal{M}(D') \in R) \cdot \frac{n-k}{n} + \frac{k}{n} \tag{24}$$

$$\tag{25}$$

Now, we merge the Case 1 and 2 together. Then we have $e^{\epsilon} = \max(\frac{n+1}{n+1-k}, \frac{n-k}{n}) = \frac{n+1}{n+1-k}$ and $\delta = \max(0, \frac{k}{n}) = \frac{k}{n}$. Therefore, NFDP without replacement statisfies $(\ln \frac{n+1}{n+1-k}, \frac{k}{n})$-differential privacy. □

Theorem 2. *[NFDP mechanism: (ϵ, δ)-differential privacy of sampling with replacement] Given a training dataset of size n, sampling with replacement achieves $((k \ln \frac{n+1}{n}, 1 - \left(\frac{n-1}{n}\right)^k)$-differential privacy, where k is the subsample size.*

Proof. Here we use the same notation as the last proof. The proof of replacement is almost similar to the without replacement. First, for a random subset $S \in \mathcal{S}$, we have

$$\Pr(\mathcal{M}(D) = S) = \begin{cases} \frac{1}{|D|^k}, & \text{if } S \in \Gamma(D), \\ 0, & \text{otherwise.} \end{cases} \tag{26}$$

$$\Pr(\mathcal{M}(D') = S) = \begin{cases} \frac{1}{|D'|^k}, & \text{if } S \in \Gamma(D'), \\ 0, & \text{otherwise.} \end{cases} \tag{27}$$

Case 1 ($D' = D \subseteq \{u\}$)**:** Due to $D \in D'$, then we have,

$$\Pr(\mathcal{M}(D) \in \Gamma(D)) = 1, \tag{28}$$

$$\Pr(\mathcal{M}(D') \in \Gamma(D)) = \frac{|D|^k}{|D'|^k}. \tag{29}$$

Let R is a random subset of $\mathcal{S}$ and R is composed by two disjoint subsets, i.e., $R = R_D \cup R_{D' \setminus D}$, where $R_D \subseteq \Gamma(D)$

and $R_{D'\setminus D} \in \Gamma(D') \setminus \Gamma(D)$. Then, we have

$$\Pr(\mathcal{M}(D) \in R) \tag{30}$$

$$= \Pr(\mathcal{M}(D) \in R_D) + \Pr(\mathcal{M}(D) \in R_{D'\setminus D}) \tag{31}$$

$$= \Pr(\mathcal{M}(D) \in R_D) + 0 \tag{32}$$

$$= \Pr(\mathcal{M}(D) \in R_D) \tag{33}$$

$$= \Pr(\mathcal{M}(D') \in R_D) \cdot \frac{|D'|^k}{|D|^k} \tag{34}$$

$$= \Pr(\mathcal{M}(D') \in R_D) \cdot \frac{(n+1)^k}{n^k} \tag{35}$$

$$\leq \Pr(\mathcal{M}(D') \in R) \cdot \left(\frac{n+1}{n}\right)^k \tag{36}$$

$$\tag{37}$$

Case 2 ($D = D' \cup \{u\}$): Due to $D' \subseteq D$, then we have,

$$\Pr(\mathcal{M}(D) \in \Gamma(D')) = \frac{|D'|^k}{|D|^k}, \tag{38}$$

$$\Pr(\mathcal{M}(D') \in \Gamma(D')) = 1. \tag{39}$$

Let P is a subset of $\Gamma(D) \setminus \Gamma(D')$, then we have

$$\Pr(\mathcal{M}(D) \in P) \leq \Pr(\mathcal{M}(D) \in \Gamma(D) \setminus \Gamma(D')), \tag{40}$$

$$\leq 1 - \left(\frac{n-1}{n}\right)^k. \tag{41}$$

Let R is a random subset of $\mathcal{S}$ and R is composed by two disjoint subsets, i.e., $R = R_{D'} \cup R_{D\setminus D'}$, where $R_{D'} \subseteq \Gamma(D')$ and $R_{D\setminus D'} \subseteq \Gamma(D) \setminus \Gamma(D')$. Then, we have

$$\Pr(\mathcal{M}(D) \in R) \tag{42}$$

$$= \Pr(\mathcal{M}(D) \in R_{D'}) + \Pr(\mathcal{M}(D) \in R_{D\setminus D'}) \tag{43}$$

$$\leq \Pr(\mathcal{M}(D) \in R_{D'}) + \left(1 - \left(\frac{n-1}{n}\right)^k\right) \tag{44}$$

$$\leq \Pr(\mathcal{M}(D') \in R_{D'}) \cdot \frac{|D'|^k}{|D|^k} + \left(1 - \left(\frac{n-1}{n}\right)^k\right) \tag{45}$$

$$\leq \Pr(\mathcal{M}(D') \in R_{D'}) \cdot \frac{(n-1)^k}{n^k} + \left(1 - \left(\frac{n-1}{n}\right)^k\right) \tag{46}$$

$$\leq \Pr(\mathcal{M}(D') \in R) \cdot \left(\frac{n-1}{n}\right)^k + \left(1 - \left(\frac{n-1}{n}\right)^k\right) \tag{47}$$

$$\tag{48}$$

Now, we merge the Case 1 and 2 together. Then we have $e^\epsilon = \max(\left(\frac{n+1}{n}\right)^k, \left(\frac{n-1}{n}\right)^k) = \left(\frac{n+1}{n}\right)^k$ and $\delta = \max(0, 1 - \left(\frac{n-1}{n}\right)^k) = 1 - \left(\frac{n-1}{n}\right)^k$. Therefore, NFDP with replacement statisfies $(k \ln \frac{n+1}{n}, 1 - \left(\frac{n-1}{n}\right)^k)$-differential privacy. □

Lemma 1. *Algorithm 1 using sampling with replacement is consistently more private than using sampling without replacement for any $n > 0$ and $0 < k \leq n$.*

Proof. Sampling with replacement is $(k \ln \frac{n+1}{n}, 1 - \left(\frac{n-1}{n}\right)^k)$-differential privacy and sampling without replacement is $(\ln \frac{n+1}{n+1-k}, k/n)$-differential privacy. Let $n \geq 1$, and then if $k = 0$ or $k = 1$,

$$\epsilon : k \ln \frac{n+1}{n} = \ln \frac{n+1}{n+1-k} \tag{49}$$

$$\text{and} \tag{50}$$

$$\delta : 1 - \left(\frac{n-1}{n}\right)^k = k/n \tag{51}$$

If $1 < k \leq n$, we have

$$\epsilon : k \ln \frac{n+1}{n} < \ln \frac{n+1}{n+1-k} \tag{52}$$

$$\text{and} \tag{53}$$

$$\delta : 1 - \left(\frac{n-1}{n}\right)^k < k/n \tag{54}$$

Briefly, we can prove above two inequalities by mathematical induction. First, if $k = 2$, above two inequalities are correct. Then we assume the inequalities are correct while $k = n - 1$. Last, we easily prove if $k = n$, the above two inequalities are still correct. Therefore, for any fixed n, $0 \leq k \leq n$, sampling with replacement is more private than Sampling without replacement. □