

---

# Securing Secure Aggregation: Mitigating Multi-Round Privacy Leakage in Federated Learning

---

**Jinhyun So \***  
ECE Department  
University of Southern California (USC)  
jinhyuns@usc.edu

**Ramy E. Ali \***  
ECE Department  
University of Southern California (USC)  
reali@usc.edu

**Başak Güler**  
ECE Department  
University of California, Riverside  
bguler@ece.ucr.edu

**Jiantao Jiao**  
EECS Department  
University of California, Berkeley  
jiantao@eecs.berkeley.edu

**A. Salman Avestimehr**  
ECE Department  
University of Southern California (USC)  
avestime@usc.edu

## Abstract

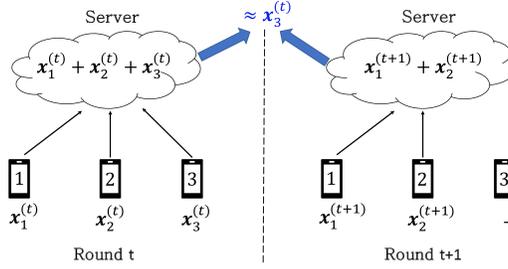
Secure aggregation is a critical component in federated learning, which enables the server to learn the aggregate model of the users without observing their local models. Conventionally, secure aggregation algorithms focus only on ensuring the privacy of individual users in a *single* training round. We contend that such designs can lead to significant privacy leakages over *multiple* training rounds, due to partial user selection/participation at each round of federated learning. In fact, we show that the conventional random user selection strategies for federated learning lead to leaking users' individual models within number of rounds linear in the number of users. To address this challenge, we introduce a secure aggregation framework with multi-round privacy guarantees. In particular, we introduce a new metric to quantify the privacy guarantees of federated learning over multiple training rounds, and develop a structured user selection strategy that guarantees the long-term privacy of each user (over any number of training rounds). Our framework also carefully accounts for the fairness and the average number of participating users at each round. We perform several experiments on MNIST and CIFAR-10 datasets in the IID and the non-IID settings to demonstrate the performance improvement over the baseline algorithms, both in terms of privacy protection and test accuracy.

## 1 Introduction

Federated learning (FL) enables collaborative training of machine learning models over the data collected and stored locally by multiple data-owners (users). Training in FL is coordinated by a central server who maintains a global model that is updated locally by the users. The local updates are then aggregated by the server to update the global model. Throughout the training process, users never share their data with the server, i.e., the data is always kept on device, rather, they only share their local models. However, as has been shown recently, local models may still reveal substantial

---

\*Equal contribution.



**Figure 1:** A multi-round secure aggregation example, where the server can reconstruct the model of user 3. At round  $t$ , the set of participating users is  $\mathcal{S}^{(t)} = \{1, 2, 3\}$ . At round  $t + 1$ , the set of participating users is  $\mathcal{S}^{(t+1)} = \{1, 2\}$ . If the local models do not change significantly over the two rounds (e.g., the models start to converge), the server can reconstruct the model of user 3 from the aggregated models of the two rounds.

information about the local datasets, and the private training data can be reconstructed from the local models through inference or inversion attacks (see e.g., [1, 2, 3, 4]).

To prevent such information leakage, *secure aggregation* protocols are proposed (e.g., [5, 6, 7, 8, 9]) to protect the privacy of individual local models, both from the server and other users, while still allowing the server to learn their aggregate. More specifically, secure aggregation protocols ensure that, at any given round, the server can only learn the aggregate model of the users, and beyond that no further information is revealed about the individual local model of a particular user.

Secure aggregation protocols, however, only ensure the privacy of the individual users in a *single training round*, and do not consider their privacy over multiple training rounds. On the other hand, due to partial user selection [10, 11, 12, 13], the server may be able to reconstruct the individual models of some users using the aggregated models from the previous rounds. As a simple illustrative example, consider the setting in Fig. 1. Although model aggregation in each round is secure (in the sense that the server only learns the aggregate model of the active users), the server can still combine the aggregate models of the two rounds to approximate the individual model update of user 3.

As such motivated, we study long-term user privacy in federated learning. We first introduce a new metric to quantify the privacy guarantees of a secure aggregation protocol over multiple training rounds. Then, we formulate the user selection problem with long-term privacy guarantees, by also taking into account the fairness and the average number of participating users (which then controls the convergence rate) in the selection process. Finally, we develop Multi-RoundSecAgg, a practical user selection strategy that guarantees user privacy across any number of training rounds.

Multi-RoundSecAgg has provable privacy and convergence guarantees, and provides a trade-off between privacy and convergence rate. In particular, as we demonstrate in our theoretical analysis, stronger privacy guarantees require fewer number of participating users per training round, which slows down the convergence of the training. We also show that conventional random user selection strategies for FL, while maximizing the convergence rate, can be vulnerable to attacks that span multiple training rounds. Through extensive numerical experiments on the MNIST and CIFAR-10 datasets on both IID and non-IID settings, we have observed that Multi-RoundSecAgg can achieve comparable accuracy to the random user selection strategy, while ensuring long-term privacy guarantees for the individual users. In summary, our specific contributions are as follows.

1. We introduce a new metric to capture long-term privacy guarantees in FL for the first time. This long-term privacy condition requires that the server cannot reconstruct any individual model using the aggregated models from any number of training rounds. Using this metric, we show that the conventional random user selection schemes can result in leaking the individual user models after a sufficient number of rounds, even if secure aggregation is employed at each round.
2. We propose Multi-RoundSecAgg, a privacy-preserving structured user selection strategy that ensures the long-term privacy of the individual users over any number of training rounds. This strategy also takes into account the fairness of the selection process and the average number of participating users at each round.

3. We demonstrate that Multi-RoundSecAgg creates a trade-off between the long-term privacy guarantee and the average number of participating users. In particular, as the average number of participating users increases, the long-term privacy guarantee becomes weaker.
4. We provide the convergence analysis of Multi-RoundSecAgg, which shows that the long-term privacy guarantee and the average number of participating users control the convergence rate. The convergence rate is maximized when the average number of participating users is maximized (e.g., the random user selection strategy maximizes the average number of participating users at the expense of not providing long-term privacy guarantees). As we require stronger long-term privacy guarantees, the average number of participating users decreases and a larger number of training rounds is required to achieve the same level of accuracy as the random selection strategy.
5. Finally, we conduct extensive experiments in both IID and non-IID settings on MNIST and CIFAR-10 datasets and demonstrate that Multi-RoundSecAgg achieves almost the same test accuracy compared to random selection scheme while providing better long-term privacy guarantees.

## 2 Related Work

The underlying principle of the secure aggregation protocol in [5] is that each pair of users exchange a pairwise secret key which they can use to mask their local models before sharing them with the server. The pairwise masks cancel out when the server aggregates the masked models, allowing the server to learn the aggregate of the local models. These masks also ensure that the local models are kept private, i.e., no further information is revealed beyond the aggregate of the local models. This protocol incurs a significant communication cost due to exchanging and reconstructing the pairwise keys. Recently, several works have developed communication-efficient exchange protocols [6, 7, 9, 14, 15], which are complementary to and can be combined with our work. Another line of work focused on designing partial user selection strategies to overcome the communication bottleneck in FL while speeding up the convergence by selecting the users based on their local loss [10, 11, 12, 13].

Previous works, either on secure aggregation or on partial user selection, however, do not consider the potential privacy leakage as a result of partial user participation and the server observing the aggregated models across multiple training rounds. This is the first study to address this challenge, where we identify a novel metric to quantify the long-term privacy of secure aggregation, and develop a privacy-preserving user selection strategy with provable long-term privacy and fairness guarantees.

## 3 System Model

In this section, we first describe the basic federated learning model in Section 3.1. Next, we introduce the multi-round secure aggregation problem for federated learning and define the key metrics to evaluate the performance of a multi-round secure aggregation protocol in Section 3.2.

### 3.1 Basic Federated Learning Model

We consider a cross-device federated learning setup consisting of a server and  $N$  devices (users). User  $i \in [N]$  has a local dataset  $\mathcal{D}_i$  consisting of  $m_i = |\mathcal{D}_i|$  data samples. The users are connected to each other through the server, i.e., all communication between the users goes through the server (no direct links exist between the users) [16, 5, 17]. The goal is to collaboratively learn a single global model  $\mathbf{x}$  with dimension  $d$ , using the local datasets that are generated, stored, and processed locally by the users. The training task can be represented by minimizing a global loss function,

$$\min_{\mathbf{x}} L(\mathbf{x}) \text{ s.t. } L(\mathbf{x}) = \sum_{i=1}^N w_i L_i(\mathbf{x}), \quad (1)$$

where  $L_i$  is the local loss function of user  $i$  and  $w_i \geq 0$  is a weight parameter assigned to user  $i$  to specify the relative impact of the user, where  $\sum_i w_i = 1$ . A common choice for the weight parameters is  $w_i = \frac{m_i}{m}$ , where  $m = \sum_{i=1}^N m_i$  [17]. We define the optimal model parameters  $\mathbf{x}^*$  and  $\mathbf{x}_i^*$  as  $\mathbf{x}^* = \arg \min_{\mathbf{x} \in \mathbb{R}^d} L(\mathbf{x})$  and  $\mathbf{x}_i^* = \arg \min_{\mathbf{x} \in \mathbb{R}^d} L_i(\mathbf{x})$ .

**Federated Averaging with Partial User Participation.** To solve (1), the most common algorithm is the *FedAvg* (federated averaging) algorithm proposed in [16]. *FedAvg* is an iterative algorithm, where model training is done by repeatedly iterating over individual local updates. At the beginning of each training round, the server sends the current state of the global model, denoted by  $\mathbf{x}^{(t)}$  at round

$t$ , to the users. Each round consists of two phases, local training and aggregation. In the local training phase, user  $i \in [N]$  updates the global model by carrying out  $E (\geq 1)$  local stochastic gradient descent (SGD) steps and sends the updated local model  $\mathbf{x}_i^{(t)}$  to the server. One of key features of cross-device federated learning is partial device participation. Due to various reasons such as being offline, unreliable wireless connectivity, or battery issues, at any given round, only a fraction of the users are available to participate in the protocol. We refer to such users as *available* users throughout the paper. In the aggregation phase, the server selects  $K \leq N$  users among the users that are available in that training round and aggregates the local models of the selected users. After receiving the local updates from the users, the server updates the global model as follows

$$\mathbf{x}^{(t+1)} = \sum_{i \in \mathcal{S}^{(t)}} \mathbf{x}_i^{(t)} = \mathbf{X}^{(t)\top} \mathbf{p}^{(t)}, \quad (2)$$

where  $\mathbf{p}^{(t)} \in \{0, 1\}^N$  denotes a participation vector at round  $t$  whose  $i$ -th entry is 0 when user  $i$  is not selected and 1 when user  $i$  is selected.  $\mathbf{X}^{(t)}$  denotes the concatenation of the weighted local models at round  $t$ , i.e.,  $\mathbf{X}^{(t)} = [w_1 \mathbf{x}_1^{(t)}, \dots, w_N \mathbf{x}_N^{(t)}]^\top \in \mathbb{R}^{N \times d}$ . Finally, the server broadcasts the updated global model  $\mathbf{x}^{(t+1)}$  to the users for the next round.

### 3.2 Multi-round Secure Aggregation

Conventional secure aggregation protocols only consider the privacy guarantees over a single training round. While secure aggregation protocols have provable privacy guarantees at any single round, in the sense that no information is leaked beyond the aggregated models at each round, the privacy guarantees do not extend to attacks *that span multiple training rounds*. Specifically, by using the aggregated models and participation information across multiple rounds, an individual model may be reconstructed from the aggregated models. For instance, consider the following user participation strategy across three training rounds,  $\mathbf{p}^{(1)} = [1, 1, 0]^\top$ ,  $\mathbf{p}^{(2)} = [0, 1, 1]^\top$ , and  $\mathbf{p}^{(3)} = [1, 0, 1]^\top$ . Assume a scenario where the local updates do not change significantly over time (e.g., models start to converge, or the server fixes the global model over consecutive rounds). Then, the server can single out all individual models  $\mathbf{x}_i$  even if a secure aggregation protocol is employed at each round.

In this paper, we study secure aggregation protocols with long-term privacy guarantees (which we term *multi-round secure aggregation*) for the cross-device FL setup which (to the best of our knowledge) has not been studied before. We assume that user  $i \in [N]$  drops from the protocol at each round with probability  $p_i$ .  $\mathcal{U}^{(t)}$  denotes the index set of available users at round  $t$  and  $\mathbf{u}^{(t)} \in \{0, 1\}^N$  is a vector indicating the available users, such that  $\{\mathbf{u}^{(t)}\}_j = \mathbb{1}\{j \in \mathcal{U}^{(t)}\}$  where  $\{\mathbf{u}\}_j$  is  $j$ -th entry of vector  $\mathbf{u}$  and  $\mathbb{1}\{\cdot\}$  is an indicator function. The server selects  $K$  users from  $\mathcal{U}^{(t)}$ , if  $|\mathcal{U}^{(t)}| \geq K$ , based on the history of selected users in previous rounds. The local models of the selected users are then aggregated via a secure aggregation protocol (i.e., by communicating masked models), at the end of which the server learns the aggregate of the local models of the selected users. Our goal is to design a user selection algorithm  $\mathcal{A}^{(t)} : \{0, 1\}^{t \times N} \times \{0, 1\}^N \rightarrow \{0, 1\}^N$ ,

$$\mathcal{A}^{(t)}(\mathbf{P}^{(t)}, \mathbf{u}^{(t)}) = \mathbf{p}^{(t)} \text{ such that } \|\mathbf{p}^{(t)}\|_0 \in \{0, K\}, \quad (3)$$

to prevent the potential information leakage over multiple rounds, where  $\mathbf{p}^{(t)} \in \{0, 1\}^N$  is the participation vector defined in (2),  $\|\mathbf{x}\|_0$  denotes the  $L_0$ -“norm” of a vector  $\mathbf{x}$ ,  $K$  denotes the maximum number of selected users. We note that  $\mathcal{A}^{(t)}$  can be a random function.  $\mathbf{P}^{(t)}$  is a matrix representing the user participation information up to round  $t$ , and is termed the *participation matrix*, given by

$$\mathbf{P}^{(t)} = [\mathbf{p}^{(0)}, \mathbf{p}^{(1)}, \dots, \mathbf{p}^{(t-1)}]^\top \in \{0, 1\}^{t \times N}. \quad (4)$$

If  $|\mathcal{U}^{(t)}| < K$ , the server skips the aggregation phase that round, by setting  $\mathbf{p}^{(t)} = \mathbf{0}$ .

**Key Metrics.** A multi-round secure aggregation protocol can be represented by  $\mathcal{A} = \{\mathcal{A}^{(t)}\}_{t \in [J]}$ , where  $\mathcal{A}^{(t)}$  is the user selection algorithm at round  $t$  defined in (3) and  $J$  is the total number of rounds. The inputs of  $\mathcal{A}^{(t)}$  are a random vector  $\mathbf{u}^{(t)}$ , which indicates the available users at round  $t$ , and the participation matrix  $\mathbf{P}^{(t)}$  defined in (4) which can be a random matrix. Given the participation matrix  $\mathbf{P}^{(J)}$ , where  $J$  is the total number of rounds, we evaluate the performance of the corresponding multi-round secure aggregation protocol through the following three key metrics.

1. **Multi-round Privacy Guarantee.** We consider a security model where the server is honest-but-curious. A multi-round privacy guarantee of  $T$  requires that, any non-zero partial sum of

the local models that the server can construct, through any linear combination  $\mathbf{X}^\top \mathbf{P}^{(J)\top} \mathbf{z}$ , where  $\mathbf{z} \in \mathbb{R}^J \setminus \{\mathbf{0}\}$  and  $J$  is the total number of rounds, must be of the form<sup>2</sup>

$$\mathbf{X}^\top \mathbf{P}^{(J)\top} \mathbf{z} = \sum_{i \in [n]} a_i \sum_{j \in \mathcal{S}_i} \mathbf{x}_j = a_1 \sum_{j \in \mathcal{S}_1} \mathbf{x}_j + a_2 \sum_{j \in \mathcal{S}_2} \mathbf{x}_j + \cdots + a_n \sum_{j \in \mathcal{S}_n} \mathbf{x}_j, \quad (5)$$

where  $|\mathcal{S}_i| \geq T, a_i \neq 0, \forall i \in [n]$  and  $n \in \mathbb{Z}^+$ . Here all the sets  $\mathcal{S}_i$ , the number of sets  $n$ , and each  $a_i$  could all depend on  $\mathbf{z}$ . In equation (5), we consider the worst-case scenario, where the local models do not change over the rounds. That is,  $\mathbf{X}^{(t)} = \mathbf{X} \forall t \in [J]$ . Intuitively, this guarantee ensures that the best that the server can do is to reconstruct a partial sum of  $T$  individual user models which corresponds to the case where  $n = 1$ . When  $T \geq 2$ , this condition implies that the server cannot get any individual user model from the aggregated models of all training rounds (the best it can obtain is the sum of two local models).

**Remark 1.** (Weaker Privacy Definitions). It is worth noting that, one can also define a weaker privacy notion requiring that  $\|\mathbf{P}^{(J)\top} \mathbf{z}\|_0 \geq T$  for any  $\mathbf{z}$  such that  $\mathbf{P}^{(J)\top} \mathbf{z} \neq \mathbf{0}$ . When  $T = 2$ , this definition requires that the server cannot reconstruct any individual model (the best it can do is to obtain a linear combination of two local models). This definition, however, allows constructions in the form of  $a\mathbf{x}_i + b\mathbf{x}_j$  for any  $a, b \in \mathbb{R} \setminus \{0\}$  (this does not violate the definition). When  $a \gg b$ , however, this is almost the same as recovering  $\mathbf{x}_i$  perfectly, hence this privacy criterion is weaker than that of (5).

**Remark 2.** (Multi-round Privacy Guarantee of Random Selection). In Section 6, we empirically show that a random selection strategy in which  $K$  available users are selected at random at each round does not ensure multi-round privacy guarantee even with respect to the weaker definition of Remark 1. Specifically, the individual local models can be reconstructed within a number of rounds that is linear in  $N$ . We also show theoretically in Appendix G that when  $\min(N - K, K) \geq cN$ , where  $c > 0$  is a constant, then the probability that the server can reconstruct all individual models after  $N$  rounds is at least  $1 - 2e^{-c'N}$  for some constant  $c'$  that depends on  $c$ . In addition, we show in Appendix G that a random selection scheme in which the users are selected in an i.i.d fashion according to  $\text{Bern}(\frac{K}{N(1-p)})$  also reveals all individual models to the server after  $N$  rounds with probability that converges to 1 exponentially fast.

**Remark 3.** (Worst-Case Assumption). In Equation (5), we considered the worst-case assumption where the models do not change over time. When the local models change over rounds, the multi-round privacy guarantee becomes even stronger as the number of unknowns increases.

2. **Aggregation Fairness Gap.** The average aggregation fairness gap quantifies the largest gap between any two users in terms of the expected relative number of rounds each user has participated in training. Formally, the average aggregation fairness gap is defined as follows

$$F = \max_{i \in [N]} \limsup_{J \rightarrow \infty} \frac{1}{J} \mathbb{E} \left[ \sum_{t=0}^{J-1} \mathbb{1} \{ \{ \mathbf{p}^{(t)} \}_i = 1 \} \right] - \min_{i \in [N]} \liminf_{J \rightarrow \infty} \frac{1}{J} \mathbb{E} \left[ \sum_{t=0}^{J-1} \mathbb{1} \{ \{ \mathbf{p}^{(t)} \}_i = 1 \} \right], \quad (6)$$

where  $\mathbb{1} \{ \cdot \}$  is an indicator function,  $\{ \mathbf{p}^{(t)} \}_i$  is  $i$ -th entry of the vector  $\mathbf{p}^{(t)}$ , and the expectation is over the randomness of the user selection algorithm  $\mathcal{A}$  and the user availability.

3. **Average Aggregation Cardinality.** The aggregation cardinality quantifies the expected number of models to be aggregated per round. Formally, it is defined as

$$C = \liminf_{J \rightarrow \infty} \frac{\mathbb{E} \left[ \sum_{t=0}^{J-1} \|\mathbf{p}^{(t)}\|_0 \right]}{J}, \quad (7)$$

where the expectation is over the randomness in  $\mathcal{A}$  and the user availability.

### 3.3 Baseline Schemes

In this subsection, we introduce three baseline schemes for multi-round secure aggregation.

**Random Selection.** In this scheme, at each round, the server selects  $K$  users at random from the set of available users. If less than  $K$  users are available, the server skips this round.

**Random Weighted Selection.** This scheme is a modified version of random selection to reduce the average aggregation fairness gap when the dropout probabilities of the users are not equal. In this

<sup>2</sup>We assume that  $w_i = \frac{1}{N}, \forall i \in [N]$  in this paper, but the definition can be modified if this is not the case.

scheme, the server selects  $K$  users at random from the available users with the minimum frequency of participation in previous rounds. If less than  $K$  users are available, the server skips this round.

**User Partitioning (Grouping).** In this scheme, the users are partitioned into  $G = N/K$  equal-sized groups denoted as  $\mathcal{G}_1, \mathcal{G}_2, \dots, \mathcal{G}_G$ . At each round, the server selects one of the groups provided that none of the users in this group has dropped out. If multiple groups are available, in order to reduce the average aggregation fairness gap, the server selects a group including a user with the minimum frequency of participation in previous rounds. If no group is available, the server skips this round.

## 4 Proposed Scheme: Multi-RoundSecAgg

In this section, we present Multi-RoundSecAgg, which has two components as follows.

- The first component designs a family of sets of users that satisfy the multi-round privacy requirement. The inputs of the first component are the number of users  $N$ , the number of users desired to be selected at each round  $K$  and the desired multi-round privacy guarantee  $T$ . The output is a family of sets of  $K$  users that satisfy the multi-round privacy guarantee  $T$ , termed as a *privacy-preserving family*. This family is represented by a matrix  $\mathbf{B}$ .
- The second component selects a set from this designed family to satisfy the fairness guarantee. The inputs to the second component are the privacy-preserving family  $\mathbf{B}$ , the set of available users at round  $t$ ,  $\mathcal{U}^{(t)}$  and the frequency of participation of each user. The output is the set of users that will participate at round  $t$ .

We now describe the two components of Multi-RoundSecAgg in detail.

**Component 1 (Batch Partitioning (BP) of the users to guarantee multi-round privacy).** The first component of Multi-RoundSecAgg designs a family of  $R_{BP}$  sets, where  $R_{BP}$  is given in (8), satisfying the multi-round privacy requirement  $T$ . We denote the  $R_{BP} \times N$  binary matrix corresponding to these sets by  $\mathbf{B} = [\mathbf{b}_1, \dots, \mathbf{b}_{R_{BP}}]^\top$ , where  $\|\mathbf{b}_i\|_0 = K$  for all  $i \in [R_{BP}]$ . That is, the rows of  $\mathbf{B}$  are the characteristic vectors of those sets. The main idea of our proposed scheme is to restrict certain sets of users of size  $T$ , denoted as batches, to either participate together or not participate at all. This guarantees a multi-round privacy  $T$  as we show in Section 5. To construct a family of sets with this property, the users are first partitioned into  $N/T$  batches. At any given round, either all or none of the users of a particular batch participate in training. The server can choose  $K/T$  batches to participate in training, provided that all users in any given selected batch are available. Since there are  $\binom{N/T}{K/T}$  possible sets with this property, then the size of this privacy-preserving family of sets is given by<sup>3</sup>

$$R_{BP} \stackrel{\text{def}}{=} \binom{N/T}{K/T}. \quad (8)$$

In the extreme case when  $T = 1$ , this strategy specializes to the random selection strategy where the server can choose any set from the  $\binom{N}{K}$  possible sets of  $K$  users. In the other extreme case where  $T = K$ , this strategy specializes to the user partitioning strategy where there are  $N/K$  possible sets. We next provide an example to illustrate the construction of  $\mathbf{B}$  further.

**Example 1** ( $N = 8, K = 4, T = 2$ ). In this example, the users are partitioned into 4 batches as  $\mathcal{G}_1 = \{1, 2\}, \mathcal{G}_2 = \{3, 4\}, \mathcal{G}_3 = \{5, 6\}$  and  $\mathcal{G}_4 = \{7, 8\}$ . The server can choose any two batches out of these 4 batches, hence we have  $R_{BP} = \binom{4}{2} = 6$  possible sets. Fig. 2 shows our construction for this example, which ensures  $T = 2$ .

**Component 2 (Available batch selection to guarantee fairness).** At round  $t$ , user  $i \in [N]$  is available to participate in the protocol with a probability  $1 - p_i \in (0, 1]$ . Given the set of available users at round  $t$ ,  $\mathcal{U}^{(t)}$  and the frequencies of participation  $\mathbf{f}^{(t-1)} = (f_1^{(t-1)}, \dots, f_N^{(t-1)})^\top$ , the server selects  $K$  users according to the available rows in  $\mathbf{B}$  and the frequency of participation.  $f_i^{(t)}$  represents the frequency of participation of users  $i$  up to  $t$  rounds, i.e.,  $f_i^{(t)} = \sum_{j=0}^{t-1} \mathbb{1}\{\{\mathbf{p}^{(j)}\}_i = 1\}$ .

To do so, the server first finds the submatrix of  $\mathbf{B}$  denoted by  $\mathbf{B}^{(t)}$  corresponding to the set of available

$$\mathbf{B} = \begin{bmatrix} 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 & 1 & 1 & 0 & 0 \\ 1 & 1 & 0 & 0 & 0 & 0 & 1 & 1 \\ 0 & 0 & 1 & 1 & 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 & 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 \end{bmatrix}$$

**Figure 2:** Example of our construction with  $N = 8, K = 4$  and  $T = 2$ .

<sup>3</sup>We assume for simplicity that  $N/T$  and  $K/T$  are integers.

users at round  $t$ ,  $\mathcal{U}^{(t)}$ . Specifically, the  $i$ -th row of  $\mathbf{B}$  denoted by  $\mathbf{b}_i^\top$  is included in  $\mathbf{B}^{(t)}$  provided that  $\text{supp}(\mathbf{b}_i) \subseteq \mathcal{U}^{(t)}$ . Next, the server selects a row vector from  $\mathbf{B}^{(t)}$ . If  $\mathbf{B}^{(t)}$  is an empty matrix, then the server skips this round. Otherwise, the server selects a row from  $\mathbf{B}^{(t)}$  uniformly at random if  $p_i = p, \forall i \in [N]$ . If the users have different dropout probabilities, the server selects a row from  $\mathbf{B}^{(t)}$  that includes the user with the minimum frequency of participation  $\ell_{\min}^{(t-1)} \stackrel{\text{def}}{=} \arg \min_{i \in \mathcal{U}^{(t)}} f_i^{(t-1)}$ . If there are many such rows, then the server selects one of them at random.

Overall, the algorithm first designs a privacy-preserving family of sets of  $K$  users such that any set can be selected at any given round while ensuring the multi-round privacy guarantee  $T$ . Then a specific set is selected from this family at each round to ensure fairness in the user selection. We describe the two components of Multi-RoundSecAgg in Algorithm 1 and 2, which are provided in Appendix D.

## 5 Theoretical Results

We provide the theoretical and convergence guarantees of Multi-RoundSecAgg in Sec. 5.1 and 5.2.

### 5.1 Theoretical Guarantees of Multi-RoundSecAgg

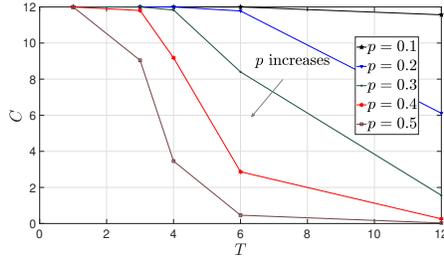
We first establish the theoretical guarantees of Multi-RoundSecAgg in terms of the multi-round privacy guarantee, aggregation fairness gap and average aggregation cardinality.

**Theorem 1.** *Multi-RoundSecAgg with parameters  $N, K, T$  ensures a multi-round privacy guarantee of  $T$ , an aggregation fairness gap  $F = 0$ , and an average aggregation cardinality given by*

$$C = K \left( 1 - \sum_{i=N/T-K/T+1}^{N/T} \binom{N/T}{i} q^i (1-q)^{N/T-i} \right), \quad (9)$$

where  $q = 1 - (1-p)^T$ , when all users have the same dropout probability  $p$ .

We provide the proof of Theorem 1 in Appendix A.



**Figure 3:** An illustration of the trade-off between the multi-round privacy guarantee  $T$  and the average aggregation cardinality  $C$ . In this example,  $N = 120$  and  $K = 12$ .

**Remark 4.** (Trade-off between “Multi-round Privacy Guarantee” and “Average Aggregation Cardinality”). Theorem 1 indicates a trade-off between the multi-round privacy guarantee and the average aggregation cardinality since as  $T$  increases,  $C$  decreases which slows down the convergence rate as we show in Section 5.2. We illustrate this trade-off in Figure 3.

**Remark 5.** (Necessity of Batch Partitioning (BP)). We show that any strategy that satisfies the privacy guarantee in Equation (5) must have a batch partitioning structure, and for given  $N, K, T, K \leq N/2$ , the largest number of distinct user sets in any strategy is at most  $\binom{N/T}{K/T}$ , which is achieved in our design in Section 4. We provide the proof in Appendix C.

**Remark 6.** (Non-linear Reconstructions of Aggregated Models). The privacy criterion in Equation (5) considers linear reconstructions of the aggregated models. One may also consider more general non-linear reconstructions. Long-term privacy guarantees of batch partitioning hold even under such reconstructions, as the users in the same batch always participate together or not participate at all, the server cannot separate individual models within the same batch even through non-linear operations.

## 5.2 Convergence Analysis of Multi-RoundSecAgg

We now provide the convergence guarantees of Multi-RoundSecAgg, by first introducing a few common assumptions [18, 19] needed for our analysis.

**Assumption 1.**  $L_1, \dots, L_N$  in (1) are all  $\rho$ -smooth: for all  $\mathbf{a}, \mathbf{b} \in \mathbb{R}^d$  and  $i \in [N]$ ,  $L_i(\mathbf{a}) \leq L_i(\mathbf{b}) + (\mathbf{a} - \mathbf{b})^\top \nabla L_i(\mathbf{b}) + \frac{\rho}{2} \|\mathbf{a} - \mathbf{b}\|^2$ .

**Assumption 2.**  $L_1, \dots, L_N$  in (1) are all  $\mu$ -strongly convex: for all  $\mathbf{a}, \mathbf{b} \in \mathbb{R}^d$  and  $i \in [N]$ ,  $L_i(\mathbf{a}) \geq L_i(\mathbf{b}) + (\mathbf{a} - \mathbf{b})^\top \nabla L_i(\mathbf{b}) + \frac{\mu}{2} \|\mathbf{a} - \mathbf{b}\|^2$ .

**Assumption 3.** Let  $\xi_i^{(t)}$  be a sample uniformly selected from the local dataset  $\mathcal{D}_k$ . The variance of the stochastic gradients at each user is bounded, i.e.,  $\mathbb{E}\|\nabla L_i(\mathbf{x}_i^{(t)}, \xi_i^{(t)}) - \nabla L_i(\mathbf{x}_i^{(t)})\|^2 \leq \sigma_i^2$  for  $i \in [N]$ .

**Assumption 4.** The expected squared norm of stochastic gradients is uniformly bounded, i.e.,  $\mathbb{E}\|\nabla L_i(\mathbf{x}_i^{(t)}, \xi_i^{(t)})\|^2 \leq G^2$  for all  $i \in [N]$ .

We now state the convergence guarantees of Multi-RoundSecAgg.

**Theorem 2.** Consider a federated learning setup with  $N$  users to train a machine learning model from (1). Assume  $K$  users are selected by Multi-RoundSecAgg with average aggregation cardinality  $C$  defined in (7) to update the global model from (2), and all users have the same dropout rate, hence Multi-RoundSecAgg selects a random set of  $K$  users uniformly from the set of available user sets at each round. Then, the following is satisfied,

$$\mathbb{E}[L(\mathbf{x}^{(J)})] - L^* \leq \frac{\rho}{\gamma + \frac{C}{K}EJ - 1} \left( \frac{2(\alpha + \beta)}{\mu^2} + \frac{\gamma}{2} \mathbb{E}\|\mathbf{x}^{(0)} - \mathbf{x}^*\|^2 \right), \quad (10)$$

where  $\alpha = \frac{1}{N} \sum_{i=1}^N \sigma_i^2 + 6\rho\Gamma + 8(E-1)^2G^2$ ,  $\beta = \frac{4(N-K)E^2G^2}{K(N-1)}$ ,  $\Gamma = L^* - \sum_{i=1}^N L_i^*$ , and  $\gamma = \max\left\{\frac{8\rho}{\mu}, E\right\}$ .

We provide the proof of Theorem 2 in Appendix B.

**Remark 7.** Theorem 2 shows how the average aggregation cardinality affects the convergence. When the average aggregation cardinality is maximized, i.e.,  $C = K$ , the convergence rate in Theorem 2 equals that of random selection algorithm provided in Theorem 3 of [18]. In (10), we have the additional term  $E$  (number of local epochs) in front of  $J$  compared to Theorem 3 of [18] as we use global round index  $t$  instead of using step index of local SGD. As the average aggregation cardinality decreases, a greater number of training rounds is required to achieve the same level of accuracy.

**Remark 8.** (Different Dropout Rates). When the dropout probabilities of the users are not the same, characterizing the theoretical and convergence guarantees of Multi-RoundSecAgg is challenging. This is due to the fact that batch selection based on the frequency of participation breaks the conditional unbiasedness of the user selection, which is required for the convergence guarantee.

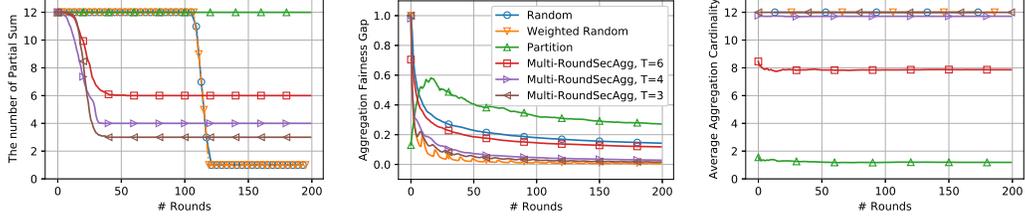
## 6 Experiments

Our experiments consist of two parts. We first numerically demonstrate the performance of Multi-RoundSecAgg compared to the baseline schemes of Section 3.3, in terms of the key metrics defined in Section 3.2. Next, we implement convolutional neural networks (CNNs) for image classification with MNIST [20] and CIFAR-10 [21] datasets to investigate how the key metrics affect the test accuracy.

**Setup.** We consider a FL setting with  $N = 120$  users, where the server aims to choose  $K = 12$  users at every round. We study two settings for partitioning the CIFAR-10 dataset across the users.

- **IID Setting.** In this setting, the 50000 training samples are shuffled and partitioned uniformly across the  $N = 120$  users, where each user receives 417 or 416 samples.
- **Non-IID dataset.** In this setting, we utilize the *data-sharing strategy* proposed in [22], where the 50000 training samples are divided into a globally shared dataset  $\mathcal{G}$  and private dataset  $\mathcal{D}$ . We set  $|\mathcal{G}| = 200$  and  $|\mathcal{D}| = 49800$ . Then, we sort  $\mathcal{D}$  by the labels, partition the sorted dataset into 120 shards of size 415, and assign each of the 120 users one shard. Each user has 200 samples of globally shared data and 415 samples of private dataset with one label.

For both settings, we implement LeNet [23]. While the state-of-the-art models [24, 25] achieve 99% accuracy, LeNet is sufficient for our needs, as our goal is to evaluate various schemes, not achieve the



(a) Multi-round privacy guarantee. (b) Aggregation fairness gap. (c) Average aggregation cardinality.  
**Figure 4:** The key metrics with  $N = 120$  (number of users),  $K = 12$  (number of selected users at each round).

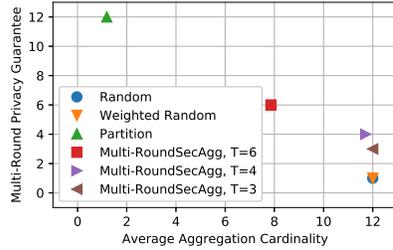
best accuracy. The hyperparameters of our experiment are provided in Appendix F.

**Modeling user dropouts.** At each round, user  $i \in [N]$  drops from the protocol with probability  $p_i$ . In the IID setting,  $p_i$  is selected from  $\{0.1, 0.2, 0.3, 0.4, 0.5\}$  uniformly at random. In the non-IID setting, in order to investigate how the aggregation fairness gap affects the test accuracy, we assume that  $p_i$  depends on the label of private data. Specifically, the dropout probability of the users with the label 0 is 0.1 while each user with label 9 has a dropout probability of 0.5.

**Implemented Schemes.** For the benchmarks, we implement the three baseline schemes introduced in Section 3.3, referred to as *Random*, *Weighted Random*, and *Partition*. For the proposed Multi-RoundSecAgg, we construct three privacy-preserving families of sets with different target multi-round privacy guarantees,  $T = 6$ ,  $T = 4$ , and  $T = 3$  which we refer to as *Multi-RoundSecAgg (T=6)*, *Multi-RoundSecAgg (T=4)*, and *Multi-RoundSecAgg (T=3)*, respectively. One can view Weighted Random and Partition as extreme cases of Multi-RoundSecAgg with  $T = 1$  and  $T = K$ , respectively. Table 1 summarizes the family size  $R$  defined in (8).

**Table 1:** Family size with  $N = 120$ ,  $K = 12$ .

Scheme	Family size (= $R$ )
Random selection	$\sim 10^{16}$
Weighted random selection	$\sim 10^{16}$
User partition	10
Multi-RoundSecAgg, $T=6$	190
Multi-RoundSecAgg, $T=4$	4060
Multi-RoundSecAgg, $T=3$	91389



**Figure 5:** Trade-off between multi-round privacy guarantee versus average aggregation cardinality with  $N = 120$  and  $K = 12$ .

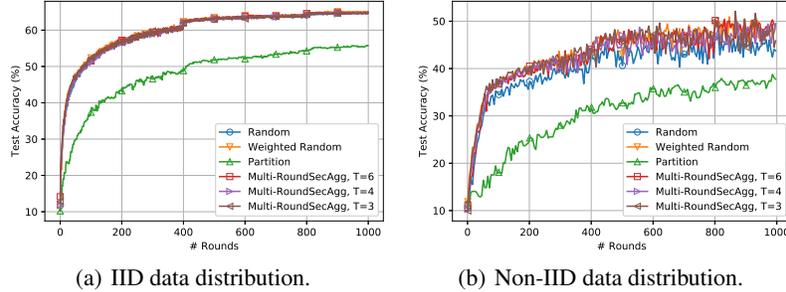
**Key Metrics.** To numerically demonstrate the performance of the six schemes in terms of the key metrics defined in Section 3.2, at each round, we measure the following metrics.

- For the multi-round privacy guarantee, we measure the number of models in the partial sum that the server can reconstruct, which is given by  $T^{(t)} := \min_{z \in \mathbb{R}^J} \|z^\top \mathbf{P}^{(t)}\|_0$ , s.t.  $\mathbf{P}^{(t)\top} z \neq \mathbf{0}$ . This corresponds to the weaker privacy definition of Remark 1. We use this weaker privacy definition as the random selection and random weighted selection strategies provide the worst privacy guarantee even with this weaker definition, as demonstrated later. On the other hand, Multi-RoundSecAgg provides better privacy guarantees with both the strong and the weaker definitions.
- For the aggregation fairness gap, we measure the instantaneous fairness gap,  $F^{(t)} := \max_{i \in [N]} F_i^{(t)} - \min_{i \in [N]} F_i^{(t)}$  where  $F_i^{(t)} = \frac{1}{i+1} \sum_{l=0}^t \mathbb{1}\{\{\mathbf{p}^{(l)}\}_i = 1\}$ .
- We measure the instantaneous aggregation cardinality as  $C^{(t)} := \frac{1}{i+1} \sum_{l=0}^t \|\mathbf{p}^{(l)}\|_0$ .

We demonstrate these key metrics in Figure 4. We make the following key observations.

- Multi-RoundSecAgg achieves better multi-round privacy guarantee than both the random selection and random weighted selection strategies, while user partitioning achieves the best multi-round privacy guarantee,  $T = K = 12$ . However, the partitioning strategy has the lowest aggregation cardinality, which results in the lowest convergence rate as demonstrated later.

- Figure 5 demonstrates the trade-off between the multi-round privacy guarantee  $T$  and the average aggregation cardinality  $C$ . Interestingly, Multi-RoundSecAgg when  $T = 3$  or  $T = 4$  achieves better multi-round privacy guarantee than both the random selection and the weighted random selection strategies while achieving almost the same average aggregation cardinality.



**Figure 6:** Training rounds versus test accuracy of LeNet in [23] on the CIFAR-10 with  $N = 120$  and  $K = 12$ . **Key Metrics versus Test Accuracy.** To investigate how the key metrics affect the test accuracy, we measure the test accuracy of the six schemes in the two settings, the IID and the non-IID settings. Our results are demonstrated in Figure 6. We make the following key observations.

- In the IID setting, the Multi-RoundSecAgg schemes show test accuracies that are comparable to the random selection and random weighted selection schemes while the Multi-RoundSecAgg schemes provide higher levels of privacy. Specifically, the Multi-RoundSecAgg schemes achieve  $T = 3, 4, 6$  based on the privacy-preserving family design while the random selection and random weighted selection schemes have  $T = 1$ , i.e., the server can learn an individual local model.
- In the non-IID setting, Multi-RoundSecAgg not only outperforms the random selection scheme but also achieves a smaller fairness gap as demonstrated in Fig. 4(b).
- In both IID and non-IID settings, the user partitioning scheme has the worst test accuracy as its aggregation cardinality is much smaller than the other schemes as demonstrated in Figure 4(c).

We also implement additional experiments on MNIST and make similar observations in Appendix E.

## 7 Conclusion

Partial user participation may breach user privacy in federated learning, even if secure aggregation is employed at every training round. To address this challenge, we introduced the notion of long-term privacy, which ensures that the privacy of individual models are protected over all training rounds. We developed Multi-RoundSecAgg, a structured user selection strategy that guarantees long-term privacy while taking into account the fairness in user selection and average number of participating users, and showed that Multi-RoundSecAgg provides a trade-off between long-term privacy and average number of participating users (hence the convergence rate). Our experiments on the MNIST and CIFAR-10 datasets on both the IID and non-IID settings show that Multi-RoundSecAgg achieves comparable accuracy to the random selection strategy (which does not ensure long-term privacy), while ensuring long-term privacy guarantees.

## References

- [1] Matt Fredrikson, Somesh Jha, and Thomas Ristenpart. Model inversion attacks that exploit confidence information and basic countermeasures. In *Proceedings of the 22nd ACM SIGSAC Conference on Computer and Communications Security*, pages 1322–1333, 2015.
- [2] Milad Nasr, Reza Shokri, and Amir Houmansadr. Comprehensive privacy analysis of deep learning: Passive and active white-box inference attacks against centralized and federated learning. In *2019 IEEE symposium on security and privacy (SP)*, pages 739–753. IEEE, 2019.
- [3] Ligeng Zhu and Song Han. Deep leakage from gradients. In *Federated Learning*, pages 17–31. Springer, 2020.
- [4] Jonas Geiping, Hartmut Bauermeister, Hannah Dröge, and Michael Moeller. Inverting gradients—how easy is it to break privacy in federated learning? *arXiv preprint arXiv:2003.14053*, 2020.
- [5] 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 *Proceedings of the 2017 ACM SIGSAC Conference on Computer and Communications Security*, pages 1175–1191, 2017.
- [6] Jinhyun So, Başak Güler, and A Salman Avestimehr. Turbo-aggregate: Breaking the quadratic aggregation barrier in secure federated learning. *IEEE Journal on Selected Areas in Information Theory*, 2(1):479–489, 2021.
- [7] Swanand Kadhe, Nived Rajaraman, O Ozan Koyluoglu, and Kannan Ramchandran. Fastsecagg: Scalable secure aggregation for privacy-preserving federated learning. *arXiv preprint arXiv:2009.11248*, 2020.
- [8] Yizhou Zhao and Hua Sun. Information theoretic secure aggregation with user dropouts. *arXiv preprint arXiv:2101.07750*, 2021.
- [9] James Henry Bell, Kallista A Bonawitz, Adrià Gascón, Tancrede Lepoint, and Mariana Raykova. Secure single-server aggregation with (poly) logarithmic overhead. In *Proceedings of the 2020 ACM SIGSAC Conference on Computer and Communications Security*, pages 1253–1269, 2020.
- [10] Yae Jee Cho, Jianyu Wang, and Gauri Joshi. Client selection in federated learning: Convergence analysis and power-of-choice selection strategies. *arXiv preprint arXiv:2010.01243*, 2020.
- [11] Wenlin Chen, Samuel Horvath, and Peter Richtarik. Optimal client sampling for federated learning. *arXiv preprint arXiv:2010.13723*, 2020.
- [12] Yae Jee Cho, Samarth Gupta, Gauri Joshi, and Osman Yağan. Bandit-based communication-efficient client selection strategies for federated learning. *arXiv preprint arXiv:2012.08009*, 2020.
- [13] Monica Ribero and Haris Vikalo. Communication-efficient federated learning via optimal client sampling. *arXiv preprint arXiv:2007.15197*, 2020.
- [14] Minxue Tang, Xuefei Ning, Yitu Wang, Yu Wang, and Yiran Chen. Fedgp: Correlation-based active client selection for heterogeneous federated learning. *arXiv preprint arXiv:2103.13822*, 2021.
- [15] Beongjun Choi, Jy-yong Sohn, Dong-Jun Han, and Jaekyun Moon. Communication-computation efficient secure aggregation for federated learning. *arXiv preprint arXiv:2012.05433*, 2020.
- [16] H Brendan McMahan, Eider Moore, Daniel Ramage, Seth Hampson, and Blaise Aguera y Arcas. Communication-efficient learning of deep networks from decentralized data. In *Int. Conf. on Artificial Int. and Stat. (AISTATS)*, pages 1273–1282, 2017.
- [17] Peter Kairouz, H Brendan McMahan, Brendan Avent, Aurélien Bellet, Mehdi Bennis, Arjun Nitin Bhagoji, Keith Bonawitz, Zachary Charles, Graham Cormode, Rachel Cummings, et al. Advances and open problems in federated learning. *arXiv preprint arXiv:1912.04977*, 2019.
- [18] Xiang Li, Kaixuan Huang, Wenhao Yang, Shusen Wang, and Zhihua Zhang. On the convergence of fedavg on non-iid data. In *International Conference on Learning Representations*, 2019.

- [19] Hao Yu, Sen Yang, and Shenghuo Zhu. Parallel restarted sgd with faster convergence and less communication: Demystifying why model averaging works for deep learning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pages 5693–5700, 2019.
- [20] Yann LeCun, Corinna Cortes, and CJ Burges. MNIST handwritten digit database. <http://yann.lecun.com/exdb/mnist>, 2010.
- [21] Alex Krizhevsky and Geoffrey Hinton. Learning multiple layers of features from tiny images. Technical report, Citeseer, 2009.
- [22] Yue Zhao, Meng Li, Liangzhen Lai, Naveen Suda, Damon Civin, and Vikas Chandra. Federated learning with non-iid data. *arXiv preprint arXiv:1806.00582*, 2018.
- [23] Yann LeCun, Patrick Haffner, Léon Bottou, and Yoshua Bengio. Object recognition with gradient-based learning. In *Shape, contour and grouping in computer vision*, pages 319–345. Springer, 1999.
- [24] Alexander Kolesnikov, Lucas Beyer, Xiaohua Zhai, Joan Puigcerver, Jessica Yung, Sylvain Gelly, and Neil Houlsby. Big transfer (bit): General visual representation learning. *arXiv preprint arXiv:1912.11370*, 6(2):8, 2019.
- [25] Mingxing Tan and Quoc Le. Efficientnet: Rethinking model scaling for convolutional neural networks. In *International Conference on Machine Learning*, pages 6105–6114. PMLR, 2019.
- [26] Sebastian U Stich. Local sgd converges fast and communicates little. *arXiv preprint arXiv:1805.09767*, 2018.
- [27] Tuan Tran. The smallest singular value of random combinatorial matrices. *arXiv preprint arXiv:2007.06318*, 2020.
- [28] Vishesh Jain, Ashwin Sah, and Mehtaab Sawhney. Singularity of discrete random matrices II. *arXiv preprint arXiv:2010.06554*, 2020.

**Organization.** These appendices are organized as follows.

- (A) In Appendix A, we prove Theorem 1.
- (B) In Appendix B, we prove Theorem 2.
- (C) In Appendix C, we show that batch partitioning is necessary to satisfy the multi-round privacy definition given in (5).
- (D) In Appendix D, we provide the two components of Multi-RoundSecAgg which are Algorithm 1 and Algorithm 2.
- (E) Appendix E provides additional experiments on the MNIST dataset.
- (F) Appendix F provides additional details and the hyperparameters of the experiments of Section 6 and Appendix E.
- (G) In Appendix G, we theoretically show that the random selection strategy discussed in Remark 2 that aims to select  $K$  available users at each round and the random selection strategy that selects the users in i.i.d fashion both have a multi-round privacy  $T = 1$  with high probability. We also empirically demonstrate that the local models can be reconstructed accurately when random selection is used.

## A Theoretical Guarantees of Multi-RoundSecAgg: Proof of Theorem 1

In this appendix, we provide the proof of Theorem 1.

*Proof.* 1. First, we prove that Multi-RoundSecAgg ensures a multi-round privacy of  $T$ . We first partition the matrix  $\mathbf{B}$  into  $R \times T$  matrices as  $\mathbf{B} = [\mathbf{B}^{(1)}, \mathbf{B}^{(2)}, \dots, \mathbf{B}^{(N/T)}]$  and the aggregated models as  $\mathbf{X} = [\mathbf{X}^{(1)\top}, \mathbf{X}^{(2)\top}, \dots, \mathbf{X}^{(N/T)\top}]^\top$ . We can then express any linear combination of the aggregated models  $\mathbf{X}^\top \mathbf{B}^\top \mathbf{z}$ , where  $\mathbf{z} \in \mathbb{R}^R \setminus \{\mathbf{0}\}$ , as follows

$$\mathbf{X}^\top \mathbf{B}^\top \mathbf{z} = \sum_{i=1}^{N/T} \mathbf{X}^{(i)\top} \mathbf{B}^{(i)\top} \mathbf{z}. \quad (11)$$

Denote the  $j$ -th column of  $\mathbf{B}^{(i)}$  by  $\mathbf{b}_j^{(i)}$  which is either a zero vector or all ones vector due to the batch partitioning structure. That is,  $\mathbf{b}_j^{(i)} \in \{\mathbf{0}, \mathbf{1}\}$ . Hence,  $\mathbf{B}^{(i)\top} \mathbf{z} \in \{\mathbf{0}, a_i \cdot \mathbf{1}\}$  for some  $a_i \in \mathbb{R} \setminus \{0\}$ . Therefore, we have

$$\mathbf{X}^{(i)\top} \mathbf{B}^{(i)\top} \mathbf{z} = \begin{cases} \mathbf{0} & \mathbf{B}^{(i)\top} \mathbf{z} = 0, \\ a_i \sum_{j=(i-1)T+1}^{iT} x_j & \text{otherwise,} \end{cases} \quad (12)$$

$\forall i \in [N/T]$ , which shows that Multi-RoundSecAgg achieves a multi-round privacy  $T$ .

2. Next, we prove that Multi-RoundSecAgg has an aggregation fairness gap  $F = 0$ .

It is clear that the total number of times user  $i$  is being selected up to time  $J$  is the same as that of user  $j$  who lies in the same batch as user  $i$ . This follows since all users in the same batch either participate together or they do not participate at all.

It suffices to show that the *expected* number of selections of user  $i$  up to time  $J$  is the same as that of user  $j$ , where user  $i$  and user  $j$  are in different batches. The main observation is that our protocol is *symmetric*. Indeed, the only randomness in the system are the user availability randomness and the set selection randomness when there are multiple user sets satisfying the requirements. We note that for any realization of random variables such that the batch of user  $i$  is selected at time  $t$ , there is a corresponding realization of random variables such that the batch of user  $j$  is selected at time  $t$  and all other selections remain exactly the same. Hence,  $F_i = F_j$  for any  $i \neq j$ .

3. Finally, we characterize the average aggregation cardinality of Multi-RoundSecAgg. The average aggregation cardinality can be expressed as follows

$$\begin{aligned}
C &= K (1 - \Pr[\text{No row of } \mathbf{B} \text{ is available}]) \\
&= K \left( 1 - \Pr[\text{At least } \frac{N}{T} - \frac{K}{T} + 1 \text{ batches are not available}] \right) \\
&= K \left( 1 - \sum_{i=N/T-K/T+1}^{N/T} \binom{N/T}{i} q^i (1-q)^{N/T-i} \right), \tag{13}
\end{aligned}$$

where  $q$  is the probability that a certain batch is not available, which is given by  $q = 1 - (1-p)^T$ .  $\square$

## B Convergence Analysis of Multi-RoundSecAgg : Proof of Theorem 2

The proof of Theorem 2 is divided into two parts. In the first part, we introduce a new sequence to represent the local updates in each user with respect to step index while we use the global round index  $t$  for  $\mathbf{x}^{(t)}$  in (2). We carefully define the sequence and the step index, and then provide the convergence analysis of the sequence. In the second part, we bridge the newly defined sequence and  $\mathbf{x}^{(t)}$  in (2), and provide convergence analysis of  $\mathbf{x}^{(t)}$ .

### First Part (Convergence analysis of local model updates).

Let  $w_i^{(j)}$  be the local model updated by user  $i$  at the  $j$ -th step. Note that this step index is different from the global round index  $t$  in (2) as each user updates the local model by carrying out  $E (\geq 1)$  local SGD steps before sending the results to the server. Let  $\mathcal{I}_E$  be the set of global synchronization steps, i.e.,  $\mathcal{I}_E = \{nE | n = 0, 1, 2, \dots\}$ . Importantly, we define the step index  $j$  such it increases from  $nE$  to  $nE + 1$  only when the server does not skip the selection, i.e., there are at least  $K$  available users at step  $nE + 1$  for  $n \in \{0, 1, 2, \dots\}$ . We denote by  $\mathcal{H}_{nE}$  the set selected by Multi-RoundSecAgg at step index  $nE$  and from the definition,  $|\mathcal{H}_{nE}| = K$  for all  $n \in \{0, 1, 2, \dots\}$ . Then, the update equation can be described as

$$v_i^{j+1} = w_i^j - \eta^j \nabla L_i(w_i^j, \xi_i^j), \tag{14}$$

$$w_i^{j+1} = \begin{cases} v_i^{j+1} & \text{if } j+1 \in \mathcal{I}_E \\ \frac{1}{K} \sum_{k \in \mathcal{H}_{j+1}} v_k^{j+1} & \text{if } j+1 \notin \mathcal{I}_E \end{cases}, \tag{15}$$

where we introduce an additional variable  $v_i^{j+1}$  to represent the immediate result of one step SGD from  $w_i^j$ . We can view  $w_i^{j+1}$  as the model obtained after aggregation step (when  $j+1$  is a global synchronization step). Motivated by [26, 18], we define two virtual sequences

$$\bar{v}^j = \frac{1}{N} \sum_{i=1}^N v_i^j, \tag{16}$$

$$\bar{w}^j = \frac{1}{N} \sum_{i=1}^N w_i^j. \tag{17}$$

We can interpret  $\bar{v}^{j+1}$  as the result of single step SGD from  $\bar{w}^j$ . When  $j \notin \mathcal{I}_E$ , both  $\bar{v}^j$  and  $\bar{w}^j$  are not accessible. We also define  $\bar{g}^j = \frac{1}{N} \sum_{i=1}^N \nabla L_i(w_i^j)$  and  $\mathbf{g}^j = \frac{1}{N} \sum_{i=1}^N \nabla L_i(w_i^j, \xi_i^j)$ . Then,  $\bar{v}^{j+1} = \bar{w}^j - \eta^j \mathbf{g}^j$ .

Now, we state our two key lemmas.

**Lemma 1** (Unbiased selection). *When  $j+1 \in \mathcal{I}_E$ , the following is satisfied,*

$$\mathbb{E}_{\mathcal{H}_{j+1}}[\bar{w}^{j+1}] = \bar{v}^{j+1}. \tag{18}$$

*Proof.* Let  $\mathcal{H}_{j+1} = \{i_1, \dots, i_K\}$ . Then, we have

$$\begin{aligned} \mathbb{E}_{\mathcal{H}_{j+1}}[\bar{\mathbf{w}}^{j+1}] &= \frac{1}{K} E_{\mathcal{H}_{j+1}} \left[ \sum_{k \in \mathcal{H}_{j+1}} \mathbf{v}_k^{j+1} \right] = \frac{1}{K} E_{\mathcal{H}_{j+1}} \left[ \sum_{k=1}^K \mathbf{v}_{i_k}^{j+1} \right] = E_{\mathcal{H}_{j+1}}[\mathbf{v}_{i_k}^{j+1}] \\ &= \sum_{k=1}^N \frac{1}{N} \mathbf{v}_k^{j+1} = \bar{\mathbf{v}}^{j+1} \end{aligned} \quad (19)$$

where (19) follows as  $\Pr[i_k = j] = \frac{1}{N}$  for  $i \in [N]$ . This is because the sampling probability of each user is identical due to the symmetry in the construction and the fact that all users have the same dropout probability.  $\square$

Now, we provide the convergence analysis of the sequence  $\bar{\mathbf{w}}^j$  defined in (17). We have,

$$\begin{aligned} \|\bar{\mathbf{w}}^{j+1} - \mathbf{w}^*\|^2 &= \|\bar{\mathbf{w}}^{j+1} - \bar{\mathbf{v}}^{j+1} + \bar{\mathbf{v}}^{j+1} - \mathbf{w}^*\|^2 \\ &= \|\bar{\mathbf{w}}^{j+1} - \bar{\mathbf{v}}^{j+1}\|^2 + \|\bar{\mathbf{v}}^{j+1} - \mathbf{w}^*\|^2 + 2 \left( \bar{\mathbf{w}}^{j+1} - \bar{\mathbf{v}}^{j+1} \right)^\top \left( \bar{\mathbf{v}}^{j+1} - \mathbf{w}^* \right). \end{aligned} \quad (20)$$

When the expectation is taken over  $\mathcal{H}_{j+1}$ , the last term in (20) becomes zero due to Lemma 1. For the second term in (20), we have

$$\|\bar{\mathbf{v}}^{j+1} - \mathbf{w}^*\|^2 \leq (1 - \eta^j \mu) \|\bar{\mathbf{w}}^j - \mathbf{w}^*\|^2 + \alpha (\eta^j)^2, \quad (21)$$

where  $\alpha = \frac{1}{N} \sum_{i=1}^N \sigma_i^2 + 6\rho\Gamma + 8(E-1)^2 G^2$  and (21) directly follows from Lemma 1,2,3 of [18]. The first term in (20) becomes zero if  $j+1 \in \mathcal{I}_E$ , and if  $j+1 \notin \mathcal{I}_E$ , from Lemma 5 of [18], it is bounded by

$$\mathbb{E}_{\mathcal{H}_{j+1}} \|\bar{\mathbf{w}}^{j+1} - \bar{\mathbf{v}}^{j+1}\|^2 \leq \beta (\eta^j)^2, \quad (22)$$

where  $\beta = \frac{4(N-K)E^2 G^2}{K(N-1)}$ . By combining (20) to (22), we have

$$\mathbb{E} \|\bar{\mathbf{w}}^{j+1} - \mathbf{w}^*\|^2 \leq (1 - \eta^j \mu) \|\bar{\mathbf{w}}^j - \mathbf{w}^*\|^2 + (\alpha + \beta) (\eta^j)^2. \quad (23)$$

Then by utilizing the similar induction in [18], we can show that

$$\mathbb{E} \|\bar{\mathbf{w}}^{j+1} - \mathbf{w}^*\|^2 \leq \frac{1}{\gamma + t - 1} \left( \frac{4(\alpha + \beta)}{\mu^2} + \gamma \mathbb{E} \|\bar{\mathbf{w}}^0 - \mathbf{w}^*\|^2 \right), \quad (24)$$

where  $\gamma = \max \left\{ \frac{8\rho}{\mu}, E \right\}$ . By combining (24) with  $\rho$ -smoothness of the global loss function in (1), we have

$$\mathbb{E}[L(\bar{\mathbf{w}}^I)] - L^* \leq \frac{\rho}{\gamma + I - 1} \left( \frac{2(\alpha + \beta)}{\mu^2} + \frac{\gamma}{2} \mathbb{E} \|\bar{\mathbf{w}}^0 - \mathbf{x}^*\|^2 \right). \quad (25)$$

### Second Part (Convergence analysis of global model).

Now, we bridge the sequence  $\bar{\mathbf{w}}^T$  and  $\mathbf{x}^{(t)}$  in (2) to provide the convergence analysis of  $\mathbf{x}^{(t)}$ . Since we define the step index  $j$  such that  $j$  increases from  $nE$  to  $nE + 1$  only when the server does not skip the selection, we have

$$\mathbb{E}[L(\mathbf{x}^{(J)})] = \mathbb{E}[L(\bar{\mathbf{w}}^{(JE\phi)})] \quad (26)$$

where  $\phi$  is the probability that there are at least  $K$  available users at a certain synchronization step, and  $\phi = \frac{C}{K}$  due to the fact that  $C = K \cdot \Pr[\text{at least one row of } \mathbf{B} \text{ is available}] = K\phi$ . By combining (25) and (26), we have that,

$$\mathbb{E}[L(\mathbf{x}^{(J)})] - L^* \leq \frac{\rho}{\gamma + \frac{C}{K} EJ - 1} \left( \frac{2(\alpha + \beta)}{\mu^2} + \frac{\gamma}{2} \mathbb{E} \|\mathbf{x}^{(0)} - \mathbf{x}^*\|^2 \right), \quad (27)$$

which completes the proof.

## C Necessity of Batch Partitioning (BP)

In this appendix, we show that batch partitioning is necessary to satisfy the multi-round privacy guarantee of Equation (5) and our strategy is optimal in the sense that no other strategy can have more distinct user selection sets than our strategy.

*Proof.* Consider any scheme which selects sets from an  $R \times N$  matrix  $\mathbf{V} = [v_1, \dots, v_N]^\top$ . Denote the linear coefficients multiplying them by  $z_i, i \in [R]$ . Then, the  $i$ -th element of  $\mathbf{V}^\top \mathbf{z}$  is given by

$$\{\mathbf{V}^\top \mathbf{z}\}_i = \sum_{j \in \text{supp}(v_i)} z_j. \quad (28)$$

We now claim that we can cluster the entries using equivalence of linear functions to groups, where each group must have a size of at least  $T$  except for the group corresponding to the zero function. To show this, we choose each  $z_i \stackrel{\text{i.i.d.}}{\sim} U[0, 1]$ , and the key observation is that if two entries have different linear functions then their final value after this assignment would be different with probability one. Since the scheme satisfies a multi-round privacy  $T$ , this implies that for each non-zero linear function of the form of Equation (28), there must be at least  $T$  of them. If we group the entries according to the equivalence of linear functions, we get at most  $N/T$  groups (ignoring the group of constant zero).

Then, we show that the total number of possible sets  $R$  is upper-bounded by  $\binom{N/T}{K/T}$ . We observe that the total number of non-zero groups we can choose for each vector is at most  $K/T$  due to the size of each group, so the total number of distinct vectors satisfying the weight requirement is at most

$$R \leq R_{\max} \stackrel{\text{def}}{=} \binom{D}{E}, \quad (29)$$

where  $D \leq N/T$  is the total number of groups corresponding to the non-zero linear functions, and  $E \leq K/T$  is the total number of groups we may select in each round. Next, we have

$$\begin{aligned} R_{\max} &= \binom{D}{E} \\ &\stackrel{(i)}{\leq} \binom{N/T}{E} \\ &\stackrel{(ii)}{\leq} \binom{N/T}{K/T} = R_{\text{BP}}, \end{aligned} \quad (30)$$

where (i) follows since  $\binom{D}{E}$  is monotonically increasing w.r.t  $D$ , and (ii) follows as  $\binom{D}{E}$  is monotonically increasing w.r.t  $E$  if  $E \leq D/2$ .  $\square$

## D The Two Components of Multi-RoundSecAgg : Algorithms 1 and 2

In Algorithm 1 and 2, we describe the two components of Multi-RoundSecAgg, which is proposed in Section 4.

---

### Algorithm 1 Batch Partitioning Privacy-preserving Family Generation

---

**Input:** Number of users  $N$ , row weight  $K$  and the desired multi-round privacy guarantee  $T$ .

**Output:** Privacy-preserving Family  $\mathbf{B} \in \{0, 1\}^{R_{\text{BP}} \times N}$ , where  $R_{\text{BP}} = \binom{N/T}{K/T}$

**Initialization:**  $\mathbf{B} = \mathbf{0}_{R_{\text{BP}} \times N}$ .

- 1: Partition index sets  $\{1, 2, \dots, N\}$  into  $\frac{N}{T}$  sets,  $\mathcal{G}_1, \dots, \mathcal{G}_{\frac{N}{T}}$ , where  $|\mathcal{G}_i| = T$  for all  $i \in [\frac{N}{T}]$ .
  - 2: Generate all possible sets each of which is union of  $\frac{K}{T}$  sets out of  $\frac{N}{T}$  sets  $(\mathcal{G}_1, \dots, \mathcal{G}_{\frac{N}{T}})$  without replacement. Denote the generated sets by  $\mathcal{L}_1, \dots, \mathcal{L}_{R_{\text{BP}}}$ .
  - 3: **for**  $i = 1, 2, \dots, R_{\text{BP}}$  **do**
  - 4:     **for**  $j = 1, 2, \dots, N$  **do**
  - 5:         **if**  $j \in \mathcal{L}_i$  **then**  $\{b_i\}_j = 1$
-

---

**Algorithm 2** Available Batch Selection
 

---

**Input:** A family of sets  $\mathbf{B}$ , set of available users  $\mathcal{U}^{(t)}$ , the frequency of participation vector  $\mathbf{f}^{(t-1)}$ , and the selection mode  $\lambda$ .  
 $\triangleright \lambda = 0$  when  $p_i = p, \forall i \in [N]$  and 1 otherwise

**Output:** A participation vector  $\mathbf{p}^{(t)}$ .

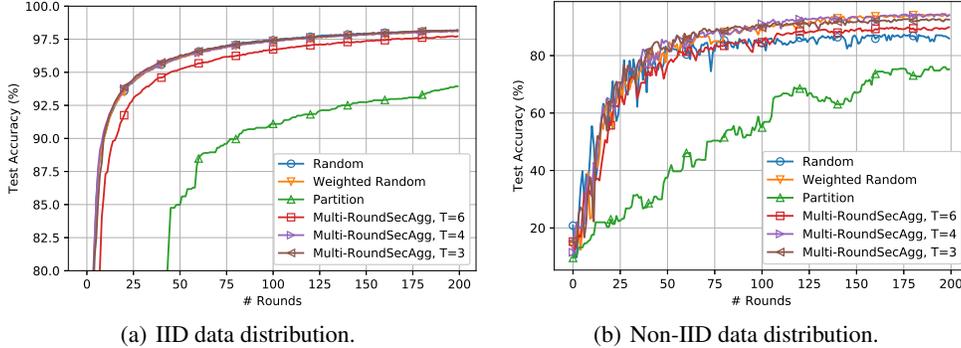
**Initialization:**  $\mathbf{B}^{(t)} = [\ ]$ ,  $\ell_{\min}^{(t-1)} := \arg \min_{i \in \mathcal{U}^{(t)}} f_i^{(t-1)}$ .

- 1: **for**  $i = 1, 2, \dots, R_{\text{BP}}$  **do**
  - 2:   **if**  $\text{supp}(\mathbf{b}_i) \subseteq \mathcal{U}^{(t)}$  **then**  $\mathbf{B}^{(t)} = [\mathbf{B}^{(t)\top}, \mathbf{b}_i]^\top$ .
  - 3: **if**  $\mathbf{B}^{(t)} = [\ ]$  **then**
  - 4:    $\mathbf{b}_{r^{(t)}}^{(t)} = \mathbf{0}$ .
  - 5: **else if**  $\lambda = 0$  **then**  $\triangleright$  Uniform selection
  - 6:   Select a row from  $\mathbf{B}^{(t)}$ ,  $\mathbf{b}_{r^{(t)}}^{(t)}$ , uniformly at random.
  - 7: **else**  $\triangleright$  Fairness-aware selection
  - 8:   Select a row from  $\mathbf{B}^{(t)}$ ,  $\mathbf{b}_{r^{(t)}}^{(t)}$ , uniformly at random from the rows that include  $\ell_{\min}^{(t-1)}$ .
  - 9:  $\mathbf{p}^{(t)} = \mathbf{b}_{r^{(t)}}^{(t)}$ .
  - 10: Update  $\mathbf{f}^{(t)} = \mathbf{f}^{(t-1)} + \mathbf{p}^{(t)}$
- 

## E Additional Experiments

To further investigate the performance of Multi-RoundSecAgg, we implement a simple CNN [16] with two  $5 \times 5$  convolution layers, a fully connected layer with ReLU activation, and a final Softmax output layer. This standard model has 1,663,370 parameters and is sufficient for our needs, as our goal is to evaluate various schemes, not to achieve the best accuracy. We study the two settings for partitioning the MNIST dataset across the users.

- **IID Setting.** In this setting, the 60000 training samples are shuffled and partitioned uniformly across the  $N = 120$  users, where each user receives 500 samples.
- **Non-IID dataset.** In this setting, we first sort the dataset by the digit labels, partition the sorted dataset into 120 shards of size 500, and assign each of the 120 users one shard. This is similar to the pathological non-IID partitioning setup proposed in [16], where our partition is an extreme case as each user has only one digit label while each user in [16] has two.



**Figure 7:** Training rounds versus test accuracy of CNN in [16] on the MNIST with  $N = 120$  and  $K = 12$ .

We measure the test accuracy of the six schemes in the two settings, the IID and the Non-IID settings, for both IID and non-IID settings. Our results are demonstrated in Figure 7. We make the following key observations, which are similar to the observations on the CIFAR-10 dataset.

- In the IID setting, the Multi-RoundSecAgg schemes show comparable test accuracy to the random selection and random weighted selection schemes while the Multi-RoundSecAgg schemes provide better multi-round privacy guarantee  $T$ .
- In the non-IID setting, the Multi-RoundSecAgg schemes outperform the random selection scheme while showing comparable test accuracy to the weighted random selection scheme. This

is because Multi-RoundSecAgg schemes have better aggregation fairness gaps as demonstrated in Figure 4(b), which results in better test accuracy in the non-IID setting.

- In both IID and non-IID settings, the user partitioning scheme has the worst test accuracy as its average aggregation cardinality is much smaller than the other schemes.

## F Experiment Details

In this section, we provide more details about the experiments of Section 6 and Appendix E.

We summarize the test accuracy of CIFAR-10 and MNIST dataset in Table 2 and Table 3, respectively. For both dataset, we run experiments five times with different random seeds and present the average value of the test accuracy in Table 2 and Table 3.

**Table 2:** Test accuracy of LeNet in [23] on the CIFAR-10 dataset with  $N = 120$  and  $K = 12$ .

Scheme	IID Setting	Non-IID Setting
Random selection	64.64%	45.20%
Weighted random selection	65.06%	47.89%
User partition	55.70%	37.74%
Multi-RoundSecAgg, T=6	65.01%	46.35%
Multi-RoundSecAgg, T=4	64.95%	47.00%
Multi-RoundSecAgg, T=3	64.80%	47.21%

**Table 3:** Test accuracy of the CNN in [16] on the MNIST dataset with  $N = 120$  and  $K = 12$ .

Scheme	IID Setting	Non-IID Setting
Random selection	98.21%	85.79%
Weighted random selection	98.10%	94.04%
User partition	93.94%	75.26%
Multi-RoundSecAgg, T=6	97.72%	89.88%
Multi-RoundSecAgg, T=4	98.11%	92.51%
Multi-RoundSecAgg, T=3	98.15%	94.16%

**Hyperparameters and computing resources.** For a fair comparison between 6 schemes, we find the best learning rate from  $\{0.1, 0.03, 0.01, 0.003, 0.001, 0.0003, 0.0001\}$ . Given the choice of the best learning rate  $\eta$ ,  $\eta$  is decayed to  $0.4\eta$  every 400 and 800 rounds to train the LeNet on the CIFAR-10 dataset while  $\eta$  is not decayed in the CNN on the MNIST dataset. To train the LeNet on the CIFAR-10 dataset, we use the mini-batch size of 50 and  $E = 1$  local epoch for both IID and Non-IID settings. To train the CNN on the MNIST dataset, we use the mini-batch size of 100 and  $E = 1$  local epoch for both IID and Non-IID settings. All experiments are conducted with users equipped with 3.4 GHz 4 cores i-7 Intel CPU and NVIDIA Geforce 1080, and the users communicate amongst each other through Ethernet to transfer the model parameters.

## G Multi-round Privacy Analysis of the Conventional Random User Selection Strategies

In this appendix, we first theoretically study the multi-round privacy of two random user selection strategies, and show that they have a very weak multi-round privacy of  $T = 1$  with high probability (for the case where  $p_i = p, \forall i \in [N]$ ). Furthermore, we also provide additional experiments showing that the server can reconstruct the local models of all users with high accuracy when a random selection strategy is used. In the theoretical analysis, to simplify the problem, we assume that the model of the users have converged and don't change from one round to the next. However, in the experiments, we empirically evaluate the error in approximating the individual models of the users (via least-squares error estimation), and show that the server can approximate individual models with very small error.

## G.1 Theoretical Analysis of the Random Selection Strategies

We start by our theoretical results, where we consider the following two random selection schemes.

1.  **$K$ -uniform Random Selection.** In this scheme, at round  $t$ ,  $K$  users are selected uniformly at random from the set of available users  $\mathcal{U}^{(t)}$  if  $|\mathcal{U}^{(t)}| \geq K$ . Otherwise, the server skips this round.
2. **I.I.D Random Selection.** In this scheme, at round  $t$ , each user is selected with probability  $\frac{K}{N(1-p)}$  independently from the other available users, where  $K < N(1-p)$ . Hence, the expected number of selected users at each round is  $K$  user.

For both schemes, we show that the server can reconstruct all individual models after  $N$  rounds in the worst-case scenario (assuming that the models do not change over  $N$  rounds). Specifically, we show that the participation matrices in both schemes have full rank with high probability after  $N$  rounds. This, in turn, implies that the server can reconstruct all local models after  $N$  rounds with high probability in both schemes. We provide our results formally next in Theorem 3.

**Theorem 3.** (*Random Selection Schemes have multi-round privacy guarantee of  $T = 1$ ).*)

1. Consider the  $K$ -uniform random selection scheme, where  $\min(K, N - K) \geq cN$ . In this scheme, the server can reconstruct all individual models of the  $N$  users after  $N$  rounds with probability at least

$$1 - 2e^{-c'N}, \quad (31)$$

for some constant  $c' > 0$  that depends on  $c$ .

2. Consider the i.i.d random selection scheme, where the users are selected according to  $\text{Bern}(\frac{K}{N(1-p)})$  distribution and let  $t = K/N$ . In this scheme, the server can reconstruct the individual models of the  $N$  users after  $N$  rounds with probability at least

$$1 - 2N(1-t)^N - (1 + o_N(1))N(N-1)(t^2 + (1-t)^2)^N, \quad (32)$$

which converges to 1 exponentially fast if  $t \in (0, 1/2)$  is a fixed constant.

*Proof.* We first note that if the participation matrix has full rank after  $N$  rounds, then the server can reconstruct the model of each individual user. Hence, we analyze the probability of the  $N \times N$  participation matrix being full rank. We now consider each scheme separately.

1. In the  $K$ -uniform random selection scheme, the probability that the participation matrix after  $N$  rounds  $\mathbf{P}^{(N)}$  has full rank is lower-bounded as follows [27], when  $\min(K, N - K) \geq cN$ ,

$$\Pr[\mathbf{P}^{(N)} \text{ has full rank}] \geq 1 - 2e^{-c'N},$$

for some constant  $c' > 0$  that depends on  $c$ . Hence, it follows that the server can reconstruct all individual models with probability at least  $1 - 2e^{-c'N}$ .

2. In the i.i.d random selection scheme, the probability that the participation matrix after  $N$  rounds  $\mathbf{P}^{(N)}$  has full rank is lower-bounded as follows [28]

$$\Pr[\mathbf{P}^{(N)} \text{ has full rank}] \geq 1 - 2N(1-t)^N - (1 + o_N(1))N(N-1)(t^2 + (1-t)^2)^N,$$

which converges to 1 exponentially fast if  $t = K/N \in (0, 1/2)$  is a fixed constant. Hence, it follows that the probability the server can reconstruct all individual models is lower-bounded by the same probability.

□

**Remark 9.** Our experimental results in Section 6 also show that the multi-round privacy guarantee of the  $K$ -uniform random selection scheme goes to 1 after almost  $N$  rounds as shown in Fig. 4(a).

## G.2 Experimental Results

We now empirically evaluate the error in approximating the individual models of the users (via least-squares error estimation), and show that the server can approximate individual models of all users with very small error. To do so, we implement a reconstruction algorithm utilizing the least-squares method, and measure the  $L_2$  distance between the original model and reconstructed model.

We consider a FL setting with  $N = 40$  users, where the server aims to choose  $K = 8$  users at every round, to train the CNN in [16] on the MNIST dataset with IID setting. After a sufficiently large number of rounds  $t_0$ , the global model at the server converges and does not change much across the rounds, which results in that local models also do not change much across the rounds. Then, we have

$$\mathbf{X}_{\text{global}}^{(t_0:t_1)} = \mathbf{P}^{(t_0:t_1)} \mathbf{X}_{\text{individual}}^{(t_0)} + \mathbf{Z}, \quad (33)$$

where  $\mathbf{X}_{\text{global}}^{(t_0:t_1)}$  denotes the concatenate of the global models from round  $t_0$  to round  $t_1 - 1$ , i.e.,  $\mathbf{X}_{\text{global}}^{(t_0:t_1)} = [\mathbf{x}^{(t_0)}, \dots, \mathbf{x}^{(t_1-1)}]^\top \in \mathbb{R}^{(t_1-t_0) \times d}$  for  $t_1 > t_0$ ,  $\mathbf{P}^{(t_0:t_1)} \in \{0, 1\}^{(t_1-t_0) \times N}$  is the participation matrix from round  $t_0$  to round  $t_1 - 1$ ,  $\mathbf{X}_{\text{individual}}^{(t_0)}$  denotes the concatenate of the individual models at round  $t_0$ , i.e.,  $\mathbf{X}_{\text{individual}}^{(t_0)} = [\mathbf{x}_1^{(t_0)}, \dots, \mathbf{x}_N^{(t_0)}]^\top \in \mathbb{R}^{N \times d}$ , and  $\mathbf{Z}$  denotes the perturbation (or noise) incurred by the local updates across the rounds.

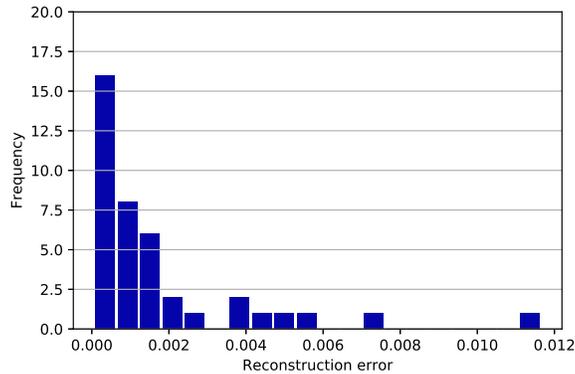
The server can then estimate  $\mathbf{X}_{\text{individual}}^{(t_0)}$  by utilizing the least-squares method as follows

$$\hat{\mathbf{X}}_{\text{individual}}^{(t_0)} = \left( \mathbf{P}^{(t_0:t_1)\top} \mathbf{P}^{(t_0:t_1)} \right)^{-1} \mathbf{P}^{(t_0:t_1)\top} \mathbf{X}_{\text{global}}^{(t_0:t_1)}, \quad (34)$$

and we measure the reconstruction error as follows

$$e_i^{(t_0)} = \frac{\|\mathbf{x}_i^{(t_0)} - \hat{\mathbf{x}}_i^{(t_0)}\|^2}{\|\mathbf{x}_i^{(t_0)}\|^2}, \quad (35)$$

where  $\hat{\mathbf{x}}_i^{(t_0)}$  denotes the reconstructed model of user  $i$ , which corresponds to  $i$ -th row of  $\hat{\mathbf{X}}_{\text{individual}}^{(t_0)}$  in (34). Figure 8 shows the histogram of the reconstruction error of the individual models of  $N = 40$  users with  $t_0 = 260$  and  $t_1 = 300$ . We observe that the  $K$ -uniform random selection scheme has a very small average reconstruction error  $\frac{1}{N} \sum_{i=1}^N e_i^{(t_0)} = 1.72 \times 10^{-3}$ , which implies that the server can reconstruct all local models as the  $K$ -uniform random selection scheme has the multi-round privacy guarantee  $T = 1$ .



**Figure 8:** Histogram of the reconstruction error defined in (35) of  $N = 40$  users when the  $K (= 8)$ -uniform random selection scheme is used to train the CNN [16] on the MNIST dataset.