

# One-Shot Federated Learning for Model Clustering and Learning in Heterogeneous Environments

Aleksandar Armacki  
Carnegie Mellon University, Pittsburgh, PA  
aarmacki@andrew.cmu.edu

Dragana Bajovic  
Faculty of Technical Sciences, University of Novi Sad, Novi Sad  
dbajovic@uns.ac.rs

Dusan Jakovetic  
Faculty of Sciences, University of Novi Sad, Novi Sad  
dusan.jakovetic@dmi.uns.ac.rs

Soumya Kar  
Carnegie Mellon University, Pittsburgh, PA  
soumyak@andrew.cmu.edu \*

## Abstract

We propose a communication efficient approach for federated learning in heterogeneous environments. The system heterogeneity is reflected in the presence of  $K$  different data distributions, with each user sampling data from only one of  $K$  distributions. The proposed approach requires only one communication round between the users and server, thus significantly reducing the communication cost. Moreover, the proposed method provides strong learning guarantees in heterogeneous environments, by achieving the optimal mean-squared error (MSE) rates in terms of the sample size, i.e., matching the MSE guarantees achieved by learning on all data points belonging to users with the same data distribution, provided that the number of data points per user is above a threshold that we explicitly characterize in terms

---

\*This work is supported by the European Union's Horizon 2020 Research and Innovation program under grant agreement No 957337. The paper reflects only the view of the authors and the Commission is not responsible for any use that may be made of the information it contains.

of system parameters. Remarkably, this is achieved without requiring any knowledge of the underlying distributions, or even the true number of distributions  $K$ . Numerical experiments illustrate our findings and underline the performance of the proposed method.

## 1 Introduction

Federated learning (FL) is a paradigm where many users collaborate, with the aim of jointly training a model [1]. Formally, the goal is to solve the problem

$$\arg \min_{\theta \in \Theta} F(\theta) = \frac{1}{m} \sum_{i=1}^m F_i(\theta), \quad (1)$$

where  $\Theta \subset \mathbb{R}^d$  is the parameter space,  $m \in \mathbb{N}$  represents the number of users, while  $F_i : \mathbb{R}^d \mapsto \mathbb{R}$ ,  $i = 1, \dots, m$  is the loss of user  $i$ .

Unlike in centralized learning, where a single user has access to all the data, in FL each user stores its data locally. This is an important feature, as typically huge amounts of users participate in the process and generate enormous amounts of data, therefore imposing significant storage cost for a single user. Additionally, the nature of the data is often sensitive, which incentivizes the users to keep their data private. The training process is coordinated by a central server, which typically includes updating and sharing the global model with the users.

While such an approach helps alleviate the storage and computation burden for any single user, it imposes significant communication costs on the system as a whole [2]. To tackle this issue, different approaches have been proposed, such as quantization [3], [4], [5], gradient sparsification [6], [7], specialized user sampling [8], [9], [10], local methods [11], [12], [13] and one-shot methods [14], [15], [16], [17], [18], to name a few. Another issue associated with training a global model comes from system heterogeneity. Users that participate in FL often contain datasets generated by different distributions, making the system as a whole highly heterogeneous. A global model can therefore be very bad for an individual user [19], [20].

One way to alleviate the issues stemming from training a global model is for each user to train their own model. Formally, the goal of such an approach is to solve the problem

$$\arg \min_{\theta_1, \dots, \theta_m \in \Theta} F_L(\theta_1, \dots, \theta_m) = \frac{1}{m} \sum_{i=1}^m F_i(\theta_i).$$

Such an approach leads to models that can be trained locally and eliminates the need for any communication. However, it is completely oblivious to any underlying similarities that might exist between users. Moreover, a strictly local approach often suffers from data imbalance - while some users generate abundance of data, the majority of users generate only a few data samples. This results in the majority of users being unable to learn useful models on their own [20].

Many different approaches that address the shortcomings of the global and purely local models have been proposed. One such approach is personalized federated learning (PFL). The goal of PFL is to learn models that fit each individual user, while utilizing the federation to produce models that generalize better. Many approaches to personalization have been proposed, such as multi-task learning [21], [22], [23], meta-learning [24], [25], fine-tuning [26], [27].

Another closely related approach is clustered federated learning (CFL). The underlying assumption in clustered federated learning is the presence of  $K$  different data distributions  $\mathcal{D}_k$ ,  $k \in [K]$ , with each user sampling their data from only one of  $K$  distributions. This leads to a natural clustering of users, i.e., we can define clusters  $\{C_k\}_{k \in [K]}$ , given by

$$C_k = \{i \in [m] : \text{user } i\text{'s data follows distribution } \mathcal{D}_k\}.$$

Since each user contains data from exactly one of  $K$  distributions, the clusters form a disjoint partition of the set of all users, i.e., we have

$$\cup_{k \in [K]} C_k = [m] \text{ and } C_k \cap C_j = \emptyset, \forall k \neq j.$$

In such a scenario the goal is to learn  $K$  models associated with the underlying clusters, so that users belonging to the same clusters have the same models. This is somewhat different from the classical PFL approaches, where the goal is to produce  $m$  models, one for each user. Allowing for  $1 \leq K \leq m$ , clustered federated learning can again be seen as an intermediary between the global and local learning, with  $K = 1$  resulting in a global model, while  $K = m$  resulting in purely local models. There are many works assuming a clustered structure among users, such as [28], [29], [30], [31], [32].

While existing works in CFL focus on dealing with heterogeneity and personalization aspects, none of them focus on communication efficiency. The aim of this paper is to provide a method for CFL that maintains the benefits of standard clustering-based approaches, while simultaneously achieving communication efficiency. This is achieved by developing a one-shot federated learning method, that requires only one round of communication.

**Literature review.** We next review the related literature, in particular, one-shot methods in FL and methods for CFL.

*One-shot methods* in the context of FL have been studied in [14], [15], [16], [17], [18]. [14], study one-shot averaging methods. Each user trains a model on the local data and the server produces the final model by averaging all the users’ models. The authors show that, for strongly convex loss  $f_i$ , the methods can achieve the same MSE guarantees as centralized learning, i.e., order-optimal rates in terms of sample size, provided that the number of samples available to each user is higher than a threshold. [15] propose to train  $K \leq m$  ensemble based methods for supervised and semi-supervised problems. [16] propose a one-shot distillation method, wherein the users send a distilled version of their local dataset to the server, which then performs the global model training. [17] study one-shot methods in federated settings under constraints on the communication budget. The proposed method is, under certain regimes, order-optimal up to logarithmic factors, while simultaneously relaxing the higher-order smoothness assumptions made in [14]. [18] introduce a one-shot FL method in heterogeneous settings, designed for data clustering. The methods [14] and [17] provide strong theoretical guarantees<sup>1</sup>, however, they assume that the data across all users follows a single distribution  $\mathcal{D}$ . Therefore, they focus on training a single global model to be used for all users. In modern FL systems the data across different users typically comes from different distributions, hence violating the IID assumptions made in prior works. Moreover, the user heterogeneity stemming from this phenomena is known to hamper the global model [19]. The methods [15] and [16] consider heterogeneous settings, but provide no theoretical analysis of their methods. To the best of our knowledge, no theoretical results for one-shot methods are established under the presence of heterogeneity, i.e., multiple data distributions in the system.

*CFL* has been studied in [28], [29], [30], [31], [32]. [28] and [29] propose similar methods, that iteratively estimate cluster membership and perform model updates. [29] show an exponential convergence rate up to an error floor that is order-optimal up to a logarithmic factor, in the number of samples and users. [30] propose a robust algorithm for CFL, under the presence of adversarial users. If there are no adversarial users (the setting that we consider in this paper), the method is order-optimal up to a logarithmic factor. [31] propose a method for CFL that can be applied to any standard FL method, as a fine-tuning step. The method is based on successive bi-

---

<sup>1</sup>The method [18] provides a theoretical analysis under heterogeneous settings. However, the method is not a general learning method, but a method designed for clustering.

partitioning of the current set of users, based on cosine similarity and does not require prior knowledge of the number of clusters  $K$ . However, when a bi-partitioning is performed, each partition needs to do a full FL training on the newly formed partition/federation, potentially requiring multiple rounds of model re-training and communication. [32] propose a method that aims to simultaneously infer the clustering of users and train models. The proposed method does not require knowledge of  $K$  and establishes explicit conditions under which the true clustering can be recovered. All of the methods require potentially many rounds of communication and model training. The methods [28], [29] and [30] require multiple communication rounds and prior knowledge of the number of clusters  $K$ . While the methods in [31] and [32] do not require knowledge of  $K$ , they require many rounds of communication.

**Contributions.** In this paper we propose a one-shot method for CFL, that is able to deal with system heterogeneity. We study the convergence guarantees of the method, in terms of the MSE with respect to the number of samples, as in [14], [29] and [30]. Our contributions can be summarized as follows:

1. We propose a one-shot method for CFL under the presence of multiple data distributions (i.e., system heterogeneity). The method is communication efficient, by only requiring one round of communication. Moreover, the proposed method does not require knowledge of the true number of clusters  $K$ .
2. We show that, for strongly convex costs, the method achieves the order-optimal MSE guarantees in terms of sample size, i.e., it matches the order-optimal MSE guarantees of centralized learning, provided that users have sufficient number of samples. This establishes regimes in which communication beyond the first round is not necessary for achieving order-optimality.
3. We show that, compared to existing methods, our algorithm reduces communication cost by a factor of  $\mathcal{O}\left(\frac{\kappa}{p} \log\left(\frac{2D}{\varepsilon}\right)\right)$ , while improving the rates by a factor logarithmic in the number of samples and users.
4. We explicitly derive the expression for the requirements on the number of data points for the users to achieve the order-optimal MSE rate and show how it depends on various system parameters, like the size of clusters, difficulty of the clustering problem, as well as problem related parameters (e.g., strong convexity constant).

5. We propose a method for discovering the underlying clustering structure of the users and establish conditions under which the method recovers the true cluster membership. The proposed method does not require prior knowledge of the true number of underlying distributions  $K$ .
6. We verify our theoretical findings via numerical experiments on linear regression problems, showing the proposed method achieves the order-optimal MSE rate and matches the performance of oracle methods that know the true cluster membership beforehand.

**Paper organization.** The rest of the paper is organized as follows. Section 2 introduces the relevant background and formally states the problem. Section 3 describes the proposed method. Section 4 presents the main results of the paper. Section 5 presents numerical results. Finally, Section 6 concludes the paper. The remainder of the section introduces the notation used throughout the paper.

**Notation.** The set of real numbers is denoted by  $\mathbb{R}$ , while  $\mathbb{R}^d$  denotes the corresponding  $d$ -dimensional vector space.  $\mathbb{N}$  denotes the set of positive integers.  $\langle \cdot, \cdot \rangle$  denotes the Euclidean inner product and  $\| \cdot \|$  denotes the induced norm. In a slight abuse of notation, we will also use  $\| \cdot \|$  to denote the corresponding matrix norm.  $[m]$  denotes the set of positive integers up to and including  $m \in \mathbb{N}$ , i.e.,  $[m] = \{1, 2, \dots, m\}$ . For a collection of sets  $\{S_k\}_{k \in [K]}$ , we use  $S_{(k)}$  to denote the  $k$ -th largest set. The notation  $\mathcal{O}(\cdot)$ ,  $\Omega(\cdot)$  refers to the standard "big O" and "big Omega" notation, respectively, i.e., for two non-negative sequences  $\{a_n\}_{n \in \mathbb{N}}$  and  $\{b_n\}_{n \in \mathbb{N}}$ , the relation  $a_n = \mathcal{O}(b_n)$  implies the existence of a global constant  $C_1 > 0$  and  $n_1 \in \mathbb{N}$ , such that  $a_n \leq C_1 b_n$ , for all  $n \geq n_1$ , while  $a_n = \Omega(b_n)$  implies the existence of a global constant  $C_2 > 0$  and  $n_2 \in \mathbb{N}$ , such that  $a_n \geq C_2 b_n$ , for all  $n \geq n_2$ .

## 2 Problem formulation and preliminaries

In this section, we begin by formally stating the problem of interest. We introduce some assumptions and discuss their implications. In Subsections 2.1 and 2.2 we introduce the method from [14] and convex clustering, respectively. We begin by introducing the notions of population and empirical loss.

Consider  $m$  users,  $i = 1, \dots, m$ , that participate in a federated learning system. The goal of standard FL approach is to train a shared model, by

solving (1), where  $F_i : \Theta \mapsto \mathbb{R}$  is the *population loss* of user  $i$ , given by

$$F_i(\theta) = \mathbb{E}_{X_i \sim \mathcal{D}_i}[\ell(\theta; X_i)]. \quad (2)$$

Here,  $\Theta \subset \mathbb{R}^d$  is the parameter space,  $\mathcal{D}_i$  is the data distribution of user  $i$ ,  $X_i \in \mathcal{X}$  is the data generating random variable distributed according to  $\mathcal{D}_i$ ,  $\mathcal{X} \subset \mathbb{R}^{d'}$  is the data space, while  $\ell : \Theta \times \mathcal{X} \mapsto \mathbb{R}$  is a loss function.

In practice, users only have access to a finite data sample, hence the aim of federated learning systems is to solve

$$\arg \min_{\theta \in \Theta} f(\theta) = \frac{1}{m} \sum_{i=1}^m f_i(\theta), \quad (3)$$

where  $f_i : \Theta \mapsto \mathbb{R}$  is the *empirical loss* of user  $i$ , given by

$$f_i(\theta) = \frac{1}{n_i} \sum_{j=1}^{n_i} \ell(\theta; x_{ij}). \quad (4)$$

Here,  $n_i \in \mathbb{N}$  represents the number of data samples available to user  $i$ , while  $x_{ij} \in \mathcal{X}$ ,  $j = 1, \dots, n_i$ , represents independent, identically distributed (IID) samples available to user  $i$ , sampled from the population  $X_i$ . However, as we proceed to argue in the reminder of the section, solving (1)-(3) is not an optimal approach under the presence of strong user heterogeneity. To that end, we formally state the main assumption used throughout the paper.

**Assumption 1.** *There exist  $K$  different data distributions in the system, with  $1 < K < m$ , such that each user samples their data from only one of the distributions, i.e., for each  $i \in [m]$ , we have  $\mathcal{D}_i = \mathcal{D}_k$ , for some  $k \in [K]$ . Moreover, the population optimal models of each cluster  $\theta_k^* := \arg \min_{\theta_k \in \Theta} F_k(\theta_k)$ ,  $k \in [K]$ , satisfy*

$$\min_{k \neq l} \|\theta_k^* - \theta_l^*\| > 0.$$

Denote by  $D$  the minimal distance between population optima of different distributions, i.e.,

$$D = \min_{k, l \in [K], k \neq l} \|\theta_k^* - \theta_l^*\| > 0.$$

Assumption 1 provides a natural partitioning of the set of all users  $[m]$ , given by

$$C_k = \{i \in [m] : \text{user } i\text{'s data follows distribution } \mathcal{D}_k\}.$$

We then have  $\cup_{k \in [K]} C_k = [m]$  and  $C_k \cap C_l = \emptyset$ , for all  $k \neq l$ . We will denote the resulting partition of  $[m]$  by  $\mathcal{C}$ , i.e.,  $\mathcal{C} = \{C_k\}_{k \in [K]}$ .

**Remark 1.** *Assumption 1 can be interpreted as a measure of distance between different distributions. Intuitively, it states that the optimal model corresponding to one of the  $K$  different populations will not be a good model for any other population. In general, Assumption 1 can be relaxed, to allow for existence of  $m$  different distributions, one corresponding to each user, while requiring that some of them are sufficiently close. We refer the reader to Lemma 7 in the Appendix, for a formal result of this argument.*

**Remark 2.** *Assumption 1 also gives a lower bound on the error caused by using models coming from different clusters. To see this, let  $i, j \in [m]$  be two users such that  $i \in C_k$ ,  $j \in C_l$ ,  $k \neq l$  and  $n_i \gg n_j$ . If user  $j$ , due to a lack of available samples, decides to use the model trained by user  $i$ , given by  $\hat{\theta}_i = \arg \min_{\theta_i \in \Theta} f_i(\theta_i)$ , then the error stemming from such an approximation is*

$$\|\hat{\theta}_i - \theta_l^*\| \geq \|\theta_k^* - \theta_l^*\| - \|\theta_k^* - \hat{\theta}_i\|. \quad (5)$$

*By Assumption 1, the first term on the right hand side of (5) is lower bounded by  $D$ . For the second term, under certain regularity conditions (to be formalized below), we can apply results from learning theory, e.g., [33], to get*

$$\|\hat{\theta}_i - \theta_l^*\| \geq D - \Omega\left(\frac{1}{\sqrt{n_i}}\right).$$

*The above equation tells us that the error floor of using a model from a different cluster grows with both  $D$  and the number of samples available to user  $i$ . For example, as we will show in Corollary 1 ahead, for  $D > 2\sqrt{\frac{|C_{(1)}|}{|C_{(K)}|}} \geq 2$  that allows our method to achieve order-optimal MSE rates, the error floor in the ideal case of balanced cluster sizes becomes*

$$\|\hat{\theta}_i - \theta_l^*\| > 2 - \Omega\left(\frac{1}{\sqrt{n_i}}\right) = \Omega(1),$$

*thus showing that the error of using a model coming from a different cluster has a constant error floor.*

Under Assumption 1, the population loss in (1) can be rewritten as

$$F(\theta) = \sum_{k=1}^K \frac{|C_k|}{m} F_k(\theta). \quad (6)$$

Similarly, the empirical loss in (3) can be cast as

$$f(\theta) = \sum_{k=1}^K \frac{|C_k|}{m} g_k(\theta), \quad (7)$$



where  $g_k : \Theta \mapsto \mathbb{R}$  is the cluster-wise loss

$$g_k(\theta) = \frac{1}{|C_k|} \sum_{i \in C_k} f_i(\theta).$$

Note that both (6) and (7) have a clear clustering structure. These formulations suggest a natural approach model training, where the goal is to find  $K$  different models, one corresponding to each cluster. Thereafter, each user from the cluster is assigned the cluster-wide optimal model. Formally, the goal is to find  $K$  models, by solving

$$\arg \min_{\theta_1, \dots, \theta_K \in \Theta} \sum_{k=1}^K \frac{|C_k|}{m} g_k(\theta_k). \quad (8)$$

To see why such an approach is optimal recall that, for a user  $i \in C_k$ , the optimal population model is given by  $\theta_k^* = \arg \min_{\theta_k \in \Theta} F_k(\theta_k)$ . If we denote the empirical risk minimizers (ERMs) as  $\hat{\theta}_i = \arg \min_{\theta_i \in \Theta} f_i(\theta_i)$ , under some regularity conditions (to be formalized below), we know from, e.g., [33], that the following MSE guarantee holds

$$\mathbb{E} \|\theta_k^* - \hat{\theta}_i\|^2 = \mathcal{O} \left( \frac{1}{n_i} \right),$$

where  $n_i$  is the number of samples available to user  $i$ . Let  $\hat{\theta}_k$  be minimizer of the empirical cluster-wise cost, i.e.,  $\hat{\theta}_k = \arg \min_{\theta_k \in \Theta} g_k(\theta_k)$ . We then have the following MSE guarantee

$$\mathbb{E} \|\theta_k^* - \hat{\theta}_k\|^2 = \mathcal{O} \left( \frac{1}{\sum_{i \in C_k} n_i} \right),$$

which shows the clear benefits of clustered learning, as  $\frac{1}{\sum_{i \in C_k} n_i} < \frac{1}{n_i}$ , for all  $i \in C_k$ . On the other hand, by Assumption 1 and cluster design, the benefits of further merging (and/or modifying) the clusters, in terms of sample size, can potentially be significantly outweighed by the distribution skew between two different clusters (recall Remark 2).

We now state the rest of the assumptions used throughout the paper.

**Assumption 2.** *The parameter space  $\Theta \subset \mathbb{R}^d$  is a compact, convex set, with  $\theta_k^* \in \text{int } \Theta$ , for all  $k \in [K]$ .*

**Remark 3.** *Assumption 2 is a standard assumption on the parameter space in statistical learning literature, e.g., [14], [33], [34].*

**Remark 4.** Assumption 2 implies the existence of a global constant  $R > 0$  such that, for all  $\theta \in \Theta$

$$\|\theta\| \leq R.$$

**Assumption 3.** For any fixed  $x \in \mathcal{X}$ , the loss function  $\ell(\cdot; x) : \Theta \mapsto \mathbb{R}$  is:

1. Nonnegative, i.e., for any  $\theta \in \Theta$ ,  $\ell(\theta; x) \geq 0$ .
2. Convex, i.e., for any  $\theta, \theta' \in \Theta$ , we have

$$\ell(\theta'; x) \geq \ell(\theta; x) + \langle \nabla \ell(\theta; x), \theta' - \theta \rangle.$$

3. Smooth, i.e., there exists a constant  $L > 0$  such that, for any  $\theta, \theta' \in \Theta$ , we have

$$\ell(\theta'; x) \leq \ell(\theta; x) + \langle \nabla \ell(\theta; x), \theta' - \theta \rangle + \frac{L}{2} \|\theta' - \theta\|^2.$$

**Remark 5.** Assumption 3 requires  $\ell$  to be non-negative, convex and smooth. It is a straightforward exercise to show that this in turns implies non-negativity, convexity and smoothness of all of  $F$ ,  $f$ ,  $F_k$  and  $g_k$ ,  $k \in [K]$  and  $f_i$ ,  $i \in [m]$ .

**Remark 6.** Note that the constant  $L$  for the smoothness condition is independent of the choice of  $x$ , i.e.,  $L$  is a global constant that holds for any choice of  $x \in \mathcal{X}$ .

**Remark 7.** Recall that, under convexity of  $\ell$ , the smoothness condition is equivalent to Lipschitz continuous gradient of  $\ell$ , i.e., for any fixed  $x \in \mathcal{X}$  and any  $\theta, \theta' \in \Theta$ , we have

$$\|\nabla \ell(\theta; x) - \nabla \ell(\theta'; x)\| \leq L \|\theta - \theta'\|.$$

From Remark 7, we can see that, for each fixed  $x \in \mathcal{X}$ , the gradients of  $\ell$  are continuous. Using the compactness of  $\Theta$ , we can conclude that  $\ell$  has bounded gradients over  $\Theta$ , for any fixed  $x \in \mathcal{X}$ . Denote by  $S$  the global gradient bound, i.e.,  $S = \sup_{x \in \mathcal{X}, \theta \in \Theta} \|\nabla \ell(\theta; x)\|$ .

Next, from Remark 5 and compactness of  $\Theta$ , we can conclude that each  $F_k$ ,  $k \in [K]$  and each  $f_i$ ,  $i \in [m]$ , have bounded gradients on  $\Theta$ . Denote the corresponding gradient bounds by  $G_{F_k}$  and  $G_{f_i}$ , respectively, i.e.,  $G_{F_k} := \max_{\theta \in \Theta} \|\nabla F_k(\theta)\|$ ,  $k \in [K]$  and  $G_{f_i} := \max_{\theta \in \Theta} \|\nabla f_i(\theta)\|$ ,  $i \in [m]$ . Appealing to the mean value theorem, we can conclude that  $F_k$  is Lipschitz continuous, with constant  $G_{F_k}$ ,  $k \in [K]$ .

The next assumption considers the behaviour of the cluster population losses  $F_k$ ,  $k \in [K]$ .

**Assumption 4.** For each  $k \in [K]$ , the population loss  $F_k(\theta) = \mathbb{E}_{X_k \sim \mathcal{D}_k}[\ell(\theta; X_k)]$  is strongly convex, i.e., there exists a constant  $\mu_{F_k} > 0$ , such that, for all  $\theta, \theta' \in \Theta$ , we have

$$F_k(\theta') \geq F_k(\theta) + \langle \nabla F_k(\theta), \theta' - \theta \rangle + \frac{\mu_{F_k}}{2} \|\theta - \theta'\|^2.$$

**Remark 8.** In addition to requirements of Assumption 3, that implies convexity of each  $F_k$ , we require them to be even "better" behaved, i.e., we require them to be strongly convex. This will facilitate the rest of the analysis and allow for stronger bounds to be obtained.

**Assumption 5.** For each  $k \in [K]$ , there exists a neighborhood  $U_k = \{\theta \in \Theta : \|\theta - \theta_k^*\| \leq \rho_k\}$ , where  $\theta_k^* = \arg \min_{\theta \in \Theta} F_k(\theta)$ ,  $\rho_k > 0$ , such that, for any fixed  $x \in \mathcal{X}$ , the loss  $\ell$  has Lipschitz continuous Hessian, i.e., there exists a constant  $P_k > 0$ , such that, for any  $\theta, \theta' \in U_k$ , we have

$$\|\nabla^2 \ell(\theta; x) - \nabla^2 \ell(\theta'; x)\| \leq P_k \|\theta - \theta'\|.$$

**Remark 9.** Assumption 5 requires each population loss to be well-behaved in a neighborhood of the optimal model. Akin to Assumption C in [14], this assumption is required for averaging methods to work. We refer the reader to [14] and references therein, for an elaborate discussion on this requirement.

Note that, for each  $k \in [K]$ , the set  $U_k$  is compact. Using the continuity of the Hessian on  $U_k$ , via Assumption 5, we can conclude that the Hessian of  $\ell$  is bounded on all  $U_k, k \in [K]$ . Denote the bounding constants as  $H_k, k \in [K]$ , i.e.,

$$H_k = \sup_{x \in \mathcal{X}, \theta \in \Theta} \|\nabla^2 \ell(\theta; x)\|, k \in [K].$$

Assumptions 2-5 are standard in the literature, e.g., the reader is referred to [14], [34] and the references therein. The authors in [14] require that the population loss be strongly convex only in a neighborhood of the optimal model, which is more relaxed than the requirement of Assumption 4. However, this condition is required in [34], whose results we apply to construct a high probability bound in the following sections.

Note that the formulation (8) implicitly assumes the knowledge of the true clustering structure. In reality, the distributions and their associated clustering structures are not known. Moreover, even the exact number of different distributions,  $K$ , is typically not available. Therefore, the formulation (8) is impossible to obtain and solve in practice. In what follows, we propose a method that is able to deal with these issues, by correctly identifying the true clusters and producing models that offer the same order-optimal MSE guarantees, as the models with knowledge of the true clustering structure, obtained by (8).

## 2.1 The method from [14]

The authors in [14] study the problem of finding the optimal model for (1), under the assumption that all the distributions are the same, i.e.,  $\mathcal{D}_k = \mathcal{D}$ , for all  $k \in [K]$ . They propose the following two-step method, that requires only one round of communication:

1. Each user  $i$  obtains a local model  $\hat{\theta}_i$ , by solving  $\hat{\theta}_i = \arg \min_{\theta \in \Theta} f_i(\theta)$  and sends it to the server.
2. The sever receives the local models and produces the final model by averaging, i.e.,  $\bar{\theta} = \frac{1}{m} \sum_{i=1}^m \hat{\theta}_i$ .

Assuming  $n_i = n$ ,  $i \in [m]$ , for some  $n \in \mathbb{N}$ , the authors show that, when  $n \geq m$ , the method results in the order-optimal MSE, i.e., we have

$$\mathbb{E} \|\bar{\theta} - \theta^*\|^2 = \mathcal{O} \left( \frac{1}{mn} \right).$$

Here,  $\theta^* = \arg \min_{\theta \in \Theta} F(\theta)$  is the optimal model for the entire population.

## 2.2 Convex clustering

Convex clustering is a well-studied approach to clustering, e.g.. [35], [36] [37], wherein the clustering problem is formulated as a strongly convex optimization problem with group lasso regularization. As such, the method is guaranteed to have a unique solution and, moreover, does not require knowledge of the true number of clusters  $K$ . Formally, for a given dataset  $A = \{a_1, \dots, a_n\} \subset \mathbb{R}^d$ , the problem of convex clustering is formulated as

$$\arg \min_{u_1, \dots, u_n \in \mathbb{R}^d} \frac{1}{2} \sum_{i=1}^n \|a_i - u_i\|^2 + \lambda \sum_{i < j} \|u_i - u_j\|, \quad (9)$$

where  $\lambda > 0$  is a tunable parameter. Let  $\mathcal{V} = \{V_k\}_{k \in [K]}$  be a partition of  $A$ , such that  $\cup_{k \in [K]} V_k = A$  and  $V_k \cap V_l = \emptyset$ ,  $k \neq l$ . The authors in [37, Corollary 7] show that, if  $\lambda$  satisfies

$$\max_{k \in [K]} \frac{\text{diam}(V_k)}{|V_k|} \leq \lambda < \min_{\substack{k \neq l \\ k, l \in [K]}} \frac{\|c(V_k) - c(V_l)\|}{2n - |V_k| - |V_l|}, \quad (10)$$

the partition, i.e., the clustering, is recovered, in the sense that, for a mapping  $\psi(x_i) = u_i^*$ , we have  $u_i^* = u_j^*$ , for all  $i, j \in V_k$  and  $u_i^* \neq u_j^*$ , for all  $i \in V_k$ ,  $j \in V_l$ ,  $k \neq l$ . Here,  $\{u_i^*\}_{i=1}^n = \{u_i^*(\lambda)\}_{i=1}^n$  is the (unique) optimal solution produced by (9),  $\text{diam}(S) = \max\{\|x - y\| : x, y \in S\}$ , is the diameter of a set  $S \subset \mathbb{R}^d$ , while  $c(S) = \frac{1}{|S|} \sum_{x \in S} x$ , is the centroid of  $S$ .

### 3 Algorithm design

In this section, we outline our one-shot algorithm for FL in heterogeneous environments. Subsection 3.1 describes the proposed one-shot method. Subsection 3.2 outlines some considerations when applying the method in practice.

#### 3.1 The proposed method

In order to deal with the presence of multiple data distributions, we propose a method that works as follows:

1. Each user  $i$  obtains a local model  $\hat{\theta}_i$ , by solving  $\hat{\theta}_i = \arg \min_{\theta_i \in \Theta} f_i(\theta_i)$  and sends it to the server.
2. The server receives the local models  $\{\hat{\theta}_i\}_{i=1}^m$ , chooses a value  $\lambda > 0$  (check Subsection 3.2 ahead) and runs the convex clustering algorithm (9), with the local models as inputs, resulting in  $K'$  clusters  $C' = \{C'_{k'}\}_{k' \in [K']}$ .
3. The server then averages the local models according to the resulting clusters, i.e., for each obtained cluster  $C'_{k'}$ ,  $k' \in [K']$ , the server performs  $\bar{\theta}_{k'} = \frac{1}{|C'_{k'}|} \sum_{i \in C'_{k'}} \hat{\theta}_i$ .
4. The server then sends the models to each user, corresponding to their cluster assignment, i.e., each user  $i \in C'_{k'}$  receives the model  $\bar{\theta}_{k'}$ .

Note that the main difference between the method in [14] and the proposed method is in step 2, where the server performs clustering of the models. This step is necessary, as we aim to identify the true clustering structure, and produce a model that maintains the guarantees of the clustered approach (8). We chose the convex clustering method, e.g., [37], [35], as it does not require knowledge of the exact number of clusters  $K$ . Note that, if knowledge of the number of clusters was available, a simpler algorithm, like K-means, e.g., [38], [39], or gradient clustering, e.g., [40], can be applied.

#### 3.2 Practical considerations

The lower and upper bounds in the recovery condition (10) both depend on the recovered clustering, which in turns depends on the value of  $\lambda$ , via (9). This shows that (10) (and (12) ahead) can only be verified in "a posteriori" manner, after (9) is solved. Therefore, choosing an appropriate value of  $\lambda$

can be difficult in practice. In this subsection we provide an algorithm that includes practical guidelines on choosing an appropriate value of the parameter  $\lambda$ , elaborating on step 2) from the previous subsection. The algorithm works as follows:

1. Each user  $i$  obtains a local model  $\hat{\theta}_i$ , by solving  $\hat{\theta}_i = \arg \min_{\theta_i \in \Theta} f_i(\theta_i)$  and sends it to the server.
2. The server receives the local models  $\{\hat{\theta}_i\}_{i=1}^m$  and chooses a range of strictly increasing values of  $\lambda$ ,  $\{\lambda_1, \lambda_2, \dots, \lambda_N\}$ , such that solving the convex clustering problem (9) results in the number of clusters  $K_{\lambda_i}$  satisfying  $K_{\lambda_1} = m$  and  $K_{\lambda_N} = 1$ <sup>2</sup>. The server runs the convex clustering algorithm for each value of  $\lambda_i$  and verifies the condition (10).
  - (a) If the condition (10) is verified for some values of  $\lambda_i$ , the server takes a value  $\lambda_i$  (and the associated clustering) such that the same number of clusters  $K_{\lambda_i}$  is recovered for the largest number of  $\lambda_i$ 's verifying (10).
  - (b) If the condition (10) is not verified for any value of  $\lambda_i$ , the server takes a value  $\lambda_i$  (and the associated clustering) such that the violation of the condition (10) is minimal, i.e., take a  $\lambda_i$  such that the difference between the associated lower and upper bounds is the smallest.
3. The server then averages the local models according to the resulting clusters  $\mathcal{C}' = \{C'_{k'}\}_{k' \in [K']}$ , i.e., for each obtained cluster  $C'_{k'}$ ,  $k' \in [K]$ , the server performs  $\bar{\theta}_{k'} = \frac{1}{|C'_{k'}|} \sum_{i \in C'_{k'}} \hat{\theta}_i$ .
4. The server then sends the models to each user, corresponding to their cluster assignment, i.e., each user  $i \in C'_{k'}$  receives the model  $\bar{\theta}_{k'}$ .

The procedure in step 2 is known as "clusterpath", e.g., [36]. The intuition behind it is to either take a value of  $\lambda$  that results in a clustering that is the likeliest to be "true", or to take a value of  $\lambda$  for which the resulting clustering is the likeliest to be "close" to a true clustering. Note that in general, the recovery guarantees of convex clustering hold only when  $\lambda$  satisfies (10). However, in practice, convex clustering is known to perform well even when

---

<sup>2</sup>From the formulation of convex clustering (9), it is obvious that, for  $\lambda$  sufficiently small, the optimal solution is going to be  $u_i^* = a_i$ ,  $i \in [m]$ , i.e.,  $K_\lambda = m$ . On the other hand, the authors in [41] show that, for  $\lambda$  sufficiently large, we have  $K_\lambda = 1$ . Hence, the choices of  $\lambda$  guaranteeing  $K_\lambda = m$  and  $K_\lambda = 1$  always exist.

the condition (10) is not met, e.g., [37] show that exact clustering can be recovered even for values of  $\lambda$  not in (10), with, e.g., [36], [42], validating the performance on real data, without the knowledge of (10).

## 4 Theoretical guarantees

In this section, we present the theoretical guarantees of the proposed method. In this section, for the sake of simplicity, we assume  $n_i = n$ , for all  $i \in [m]$ . Subsection 4.1 introduces some technical details and lemmas used in our work and presents the main result of the paper. Subsection 4.2 offers a detailed comparison of our method with the method from [29]. Subsection 4.3 presents the MSE guarantees if the exact solutions of the local empirical risks are replaced by approximate ones.

### 4.1 Main result

Specializing (10) to our method, we can see that, for the clustering in step 3 of our approach to be correct, i.e., to have  $K' = K$  and for all  $k \in [K]$ , a unique  $k' \in [K']$  to satisfy  $C'_{k'} = C_k$ , we need the following condition satisfied

$$\max_{k \in [K]} \frac{\text{diam}(W_k)}{|W_k|} \leq \lambda < \min_{\substack{k \neq l \\ k, l \in [K]}} \frac{\|c(W_k) - c(W_l)\|}{2m - |W_k| - |W_l|}, \quad (11)$$

where  $W_k = \{\hat{\theta}_i : i \in C_k\}$ ,  $k \in [K]$  is the cluster containing the ERM's of all users belonging to cluster  $C_k$ . Using the definitions of  $\text{diam}(\cdot)$ ,  $c(\cdot)$  and  $W_k$ ,  $k \in [K]$ , we get that (11) is equivalent to

$$\max_{k \in [K], i, j \in C_k} \frac{\|\hat{\theta}_i - \hat{\theta}_j\|}{|C_k|} \leq \lambda < \min_{\substack{k \neq l \\ k, l \in [K]}} \frac{\|\bar{\theta}_k - \bar{\theta}_l\|}{2m - |C_k| - |C_l|}. \quad (12)$$

**Remark 10.** *Note that in general, condition (11) (and equivalently (12)) might not hold. However, in what follows, we consider all the possible outcomes and quantify the probability of (12) being satisfied.*

Next, we state some important results used in the rest of the section, from [14] and [34].

**Lemma 1** (Theorem 3 in [34]). *Under Assumptions 1-4, for any  $k \in [K]$ , any  $i \in C_k$  and any  $0 < \delta < \frac{1}{2}$ ,  $\epsilon > 0$ , with probability at least  $1 - 2\delta$ , we*

have

$$F_k(\hat{\theta}_i) - F_k(\theta_k^*) \leq \frac{16R^2LC(\epsilon, \delta)}{n} + \frac{8R\|\nabla f_i(\theta_k^*)\| \log \frac{2}{\delta}}{n} \\ + \frac{8LF_k(\theta_k^*) \log \frac{2}{\delta}}{\mu_{F_k} n} + \left( 8RL + G_{F_k} + \frac{4RLC(\epsilon, \delta)}{n} \right) \epsilon,$$

where  $C(\epsilon, \delta) := 2 \left( \log \frac{2}{\delta} + d \log \frac{6R}{\epsilon} \right)$ .

**Lemma 2** (Theorem 1 in [14]). *Under Assumptions 1-5, for each  $k \in [K]$  and  $\bar{\theta}_k = \frac{1}{|C_k|} \sum_{i \in C_k} \hat{\theta}_i$ , we have*

$$\mathbb{E} \|\bar{\theta}_k - \theta_k^*\|^2 \leq \frac{2E_k}{n|C_k|} + \frac{5}{\mu_{F_k}^2 n^2} (H_k^2 \log d + E_k) E_k \\ + \mathcal{O}(|C_k|^{-1} n^{-2}) + \mathcal{O}(n^{-3}),$$

where  $E_k := \mathbb{E} \|\nabla^2 F_k(\theta_k^*)^{-1} \nabla \ell(\theta_k^*; X)\|^2$ .

Note that the original results in [14] and [34] concern the global population loss (1) and the corresponding empirical loss (3). These directly translate to each individual cluster in our framework, i.e., to each component in (8). Additionally, note that Lemma 2 assumes knowledge of the true clusters  $C_k$ , as the averaging is performed across the true clusters.

We are now ready to state the main result of the paper.

**Theorem 1.** *Let Assumptions 1-5 hold. If the number of samples per user satisfies  $n \geq 3$  and moreover*

$$\frac{n}{\log n} > \frac{2M(2m - |C_{(K-1)}| - |C_{(K)}|)^2}{|C_{(K)}|^2(D - 2\gamma)^2},$$

where  $\beta \geq 1$  and  $0 < \gamma < \frac{D}{2}$  are tunable parameters, while  $M = M(\beta) = \max_{i,j \in C_k, k \in [K]} M_{ik} + M_{jk}$ , and for all  $i \in C_k$ ,  $k \in [K]$

$$M_{ik} = \frac{64R^2L(\log 2 + d \log 6R + (d+1)\beta)}{\mu_{F_k}} \\ + \frac{16LF_k(\theta_k^*)(\log 2 + \beta)}{\mu_{F_k}^2} + \frac{16R\|\nabla f_i(\theta_k^*)\|(\log 2 + \beta)}{\mu_{F_k}} \\ + \frac{2G_{F_k} + 16RL(1 + \log 2 + d \log 6R + (d+1)\beta)}{\mu_{F_k}},$$



then, for any choice of  $\lambda \in \left[ \sqrt{\frac{2M \log n}{n}}, \frac{|C_{(K)}|(D-2\gamma)}{2m-|C_{(K-1)}|-|C_{(K)}|} \right)$ , we have that, for all  $k \in [K]$ , the models produced by the proposed method achieve the MSE

$$\begin{aligned} \mathbb{E} \|\bar{\theta}_k - \theta_k^*\|^2 &\leq \frac{2E_k}{n|C_k|} + \frac{4K\tilde{E}R^2}{n|C_{(K)}|\gamma^2} + \frac{4KR^2|\tilde{C}|^2}{n^\beta} \\ &+ \mathcal{O}\left(\frac{\log d}{n^2}\right) + \mathcal{O}\left(\frac{K \log d}{n^2\gamma^2}\right) + \mathcal{O}\left(\frac{1}{n^2|C_k|}\right) \\ &+ \mathcal{O}\left(\frac{K}{n^2|C_{(K)}|\gamma^2}\right) + \mathcal{O}\left(\frac{1}{n^3}\right) + \mathcal{O}\left(\frac{K}{n^3\gamma^2}\right), \end{aligned}$$

where  $E_k = \mathbb{E} \|\nabla^2 F_k(\theta_k^*)^{-1} \nabla \ell(\theta_k^*; X)\|^2$ ,  $\tilde{E} = \frac{1}{K} \sum_{k \in [K]} E_k$  and  $|\tilde{C}|^2 = \frac{1}{K} \sum_{k \in [K]} |C_k|^2$ .

Theorem 1 provides the MSE rate of the proposed method. If, in addition to the conditions of Theorem 1, we have  $n \geq |C_{(1)}|$ , then, for the choice of  $\beta \geq 2$ , we have that the MSE rate is dominated by the first two terms, i.e.,

$$\frac{2E_k}{n|C_k|} + \frac{4K\tilde{E}R^2}{n|C_{(K)}|\gamma^2}. \quad (13)$$

Since  $0 < \gamma < \frac{D}{2}$  is a tunable parameter, if  $D > 2\sqrt{\frac{|C_{(1)}|}{|C_{(K)}|}}$ , we can choose

$\gamma = \sqrt{\frac{|C_{(1)}|}{|C_{(K)}|}}$ , so that (13) becomes

$$\frac{2E_k}{n|C_k|} + \frac{4K\tilde{E}R^2}{n|C_{(1)}|}.$$

This observation directly leads to the following corollary.

**Corollary 1.** *Let conditions of Theorem 1 hold. If additionally  $D > 2\sqrt{\frac{|C_{(1)}|}{|C_{(K)}|}}$*

*and  $n \geq |C_{(1)}|$ , then for the choices of  $\beta \geq 2$  and  $\gamma = \sqrt{\frac{|C_{(1)}|}{|C_{(K)}|}}$ , we have the following MSE, for all  $k \in [K]$*

$$\mathbb{E} \|\bar{\theta}_k - \theta_k^*\|^2 \leq \mathcal{O}\left(\frac{1}{n|C_k|}\right).$$

Corollary 1 shows that, if the populations are sufficiently separated, our method can achieve the order-optimal MSE rate for each cluster, provided

that users have sufficient number of samples available. This rate is equivalent to the rate achieved by training a centralized learner on each cluster and as we discuss in Subsection 4.2 ahead, it is a stronger result compared to the current literature, where the convergence rate depends on the size of the smallest cluster,  $|C_{(K)}|$ . Remarkably, this is achieved with significant communication savings, requiring only a single communication round. Some remarks are now in order.

**Remark 11.** *The MSE guarantees in Lemma 2 are established without any requirements on the sample size  $n$ . This stems from the fact that the method from Lemma 2 can be seen as an oracle method that knows the true clustering structure. On the other hand, the sample size requirement in Theorem 1 stems from the fact that our method does not know the true clustering, hence a sufficiently large sample size that guarantees the true clustering can be recovered, is required.*

**Remark 12.** *Recall that the parameters  $\beta \geq 1$  and  $0 < \gamma < \frac{D}{2}$  are tunable. From Theorem 1, we can see that both parameters offer a trade-off between convergence speed and sample requirements. In particular, larger values of  $\beta$  and  $\gamma$  result in faster convergence, at the expense of higher sample requirements.*

**Remark 13.** *Recall the condition on the number of samples, given by*

$$\frac{n}{\log n} > \frac{2M(2m - |C_{(K-1)}| - |C_{(K)}|)^2}{|C_{(K)}|^2(D - 2\gamma)^2},$$

where  $\beta \geq 1$  and  $0 < \gamma < \frac{D}{2}$  are tunable parameters, while  $M = \max_{i,j \in C_k, k \in [K]} M_{ik} + M_{jk}$ , and for all  $i \in C_k, k \in [K]$

$$\begin{aligned} M_{ik} = & \frac{64R^2L(\log 2 + d \log 6R + (d+1)\beta)}{\mu_{F_k}} \\ & + \frac{16LF_k(\theta_k^*)(\log 2 + \beta)}{\mu_{F_k}^2} + \frac{16R\|\nabla f_i(\theta_k^*)\|(\log 2 + \beta)}{\mu_{F_k}} \\ & + \frac{2G_{F_k} + 16RL(1 + \log 2 + d \log 6R + (d+1)\beta)}{\mu_{F_k}}. \end{aligned}$$

We can identify three components of the condition that quantify the complexity of different aspects of the system:

- $M$  - quantifies the difficulty of the learning problems, as it depends on problem parameters, such as the dimension of the parameter space  $d$ ,

the smoothness and strong convexity parameters  $L$ ,  $G_{F_k}$ ,  $\mu_{F_k}$ , etc. It also depends on the population minimal value  $F_k(\theta_k^*)$  and the proximity of the population and local empirical risks in the form of  $\|\nabla f_i(\theta_k^*)\|$ . Hence, an easier learning problem implies smaller  $M$ .

- $\frac{(2m-|C_{(K)}|-|C_{(K-1)}|)^2}{|C_{(K)}|^2}$  - quantifies how well balanced the clusters are. For example, when the clusters are well balanced, so that  $|C_k| = \frac{m}{K}$ , for all  $k \in [K]$ , we have  $\frac{(2m-|C_{(K)}|-|C_{(K-1)}|)^2}{|C_{(K)}|^2} = 4(K-1)^2$ , while in the extreme case of  $|C_{(K)}| = |C_{(K-1)}| = 1$ , we have  $\frac{(2m-|C_{(K)}|-|C_{(K-1)}|)^2}{|C_{(K)}|^2} = 4(m-1)^2$ . As  $K \leq m$ , this again shows that balanced clusters are favored over unbalanced ones.
- $(D-2\gamma)^{-2}$  - quantifies the difficulty of the clustering problem. If  $D$  is smaller, population optima corresponding to different populations are closer to one another and it is more difficult to cluster the local ERM's correctly, hence requiring more samples. On the other hand, for larger  $D$ , the clustering problem becomes easier and we require fewer samples per user for correct clustering.

**Remark 14.** Recall the condition on  $D$  in Corollary 1,  $D > 2\sqrt{\frac{|C_{(1)}|}{|C_{(K)}|}}$ . When the clusters are well balanced, so that  $|C_k| = \frac{m}{K}$ ,  $k \in [K]$ , our method can achieve order-optimal rates if the minimal separation between population optima of different clusters is  $D > 2$ , i.e., independent of any problem parameters. On the other hand, in the worst case, we can have  $D > 2\sqrt{m-1}$ , if there are only two clusters  $C_1, C_2$ , such that  $|C_1| = 1$ ,  $|C_2| = m-1$ .

*Proof of Theorem 1.* We start by noting that, for any event  $\Psi$ , we have

$$\mathbb{E}\|\hat{\theta}_k - \theta_k^*\|^2 = \mathbb{E}\|\hat{\theta}_k - \theta_k^*\|^2 \mathbb{I}_\Psi + \mathbb{E}\|\hat{\theta}_k - \theta_k^*\|^2 \mathbb{I}_{\Psi^c}, \quad (14)$$

where  $\mathbb{I}_\Psi$  is the indicator random variable. We now proceed to define a specific event  $\Psi$  and establish the resulting bounds.

Applying Lemma 1 for the choice of  $\delta = \epsilon = \frac{1}{n^\beta}$ , for some  $\beta > 0$ , we get

that, for all  $k \in [K]$  and all  $i \in C_k$ , we have

$$\begin{aligned} \|\hat{\theta}_i - \theta_k^*\|^2 &\leq \frac{32R^2LC(\epsilon, \delta)}{n\mu_{F_k}} + \frac{16LF_k(\theta_k^*)(\log 2 + \beta \log n)}{n\mu_{F_k}^2} \\ &\quad + \frac{16R\|\nabla f_i(\theta_k^*)\|(\log 2 + \beta \log n)}{n\mu_{F_k}} \\ &\quad + \frac{\left(16RL + 2G_{F_k} + \frac{8RLC(\epsilon, \delta)}{n}\right)}{n^\beta \mu_{F_k}}, \end{aligned} \quad (15)$$

with probability at least  $1 - \frac{2}{n^\beta}$ , where  $C(\epsilon, \delta) = 2(\log 2 + d \log 6R + (d+1)\beta \log n)$ . Here, we used strong convexity of  $F_k$ , which implies

$$\|\hat{\theta}_i - \theta_k^*\|^2 \leq \frac{2}{\mu_{F_k}} \left( F_k(\hat{\theta}_i) - F_k(\theta_k^*) \right).$$

Note that, for  $\beta \geq 1$  and  $n \geq 3$ , the dominating term in (15), in terms of the number of samples  $n$ , is of the order  $\mathcal{O}\left(\frac{\log n}{n}\right)$ . We can therefore upper-bound the right-hand side of (15) by  $\frac{M_{ik} \log n}{n}$ , where  $M_{ik}$ ,  $i \in C_k$ ,  $k \in [K]$  is defined as follows

$$\begin{aligned} M_{ik} &= \frac{64R^2L(\log 2 + d \log 6R + (d+1)\beta)}{\mu_{F_k}} \\ &\quad + \frac{16LF_k(\theta_k^*)(\log 2 + \beta)}{\mu_{F_k}^2} + \frac{16R\|\nabla f_i(\theta_k^*)\|(\log 2 + \beta)}{\mu_{F_k}} \\ &\quad + \frac{2G_{F_k} + 16RL(1 + \log 2 + d \log 6R + (d+1)\beta)}{\mu_{F_k}}. \end{aligned}$$

As  $\frac{M_{ik} \log n}{n}$  is an upper bound on the right-hand side of (15), we can therefore conclude that, for any  $i \in C_k$ ,  $k \in [K]$

$$\mathbb{P}\left(\|\hat{\theta}_i - \theta_k^*\|^2 \leq \frac{M_{ik} \log n}{n}\right) \geq 1 - \frac{2}{n^\beta}. \quad (16)$$

Next, define the events

$$\begin{aligned} \Sigma_{ij} &= \left\{ \omega : \|\hat{\theta}_i - \hat{\theta}_j\|^2 \leq \frac{2(M_{ik} + M_{jk}) \log n}{n} \right\}, \\ \Upsilon_i &= \left\{ \omega : \|\hat{\theta}_i - \theta_k^*\|^2 \leq \frac{M_{ik} \log n}{n} \right\}. \end{aligned}$$

for all  $i, j \in C_k$ ,  $i \neq j$ ,  $k \in [K]$ . Noting that

$$\|\widehat{\theta}_i - \widehat{\theta}_j\|^2 \leq 2\|\widehat{\theta}_i - \theta_k^*\|^2 + 2\|\widehat{\theta}_j - \theta_k^*\|^2,$$

we can conclude that, for all  $i, j \in C_k$ ,  $k \in [K]$

$$\mathbb{P}(\Upsilon_i \cap \Upsilon_j) \leq \mathbb{P}(\Sigma_{ij}). \quad (17)$$

For  $\Sigma = \cap_{i,j \in C_k, i \neq j, k \in [K]} \Sigma_{ij}$ , we then get the following bound

$$\begin{aligned} \mathbb{P}(\Sigma) &\geq 1 - \sum_{\substack{i \neq j \\ i, j \in C_k \\ k \in [K]}} \mathbb{P}(\Sigma_{ij}^c) \geq 1 - \sum_{\substack{i \neq j \\ i, j \in C_k \\ k \in [K]}} \mathbb{P}((\Upsilon_i \cap \Upsilon_j)^c) \\ &\geq 1 - \sum_{\substack{i \neq j \\ i, j \in C_k \\ k \in [K]}} \frac{4}{n^\beta} \geq 1 - \frac{2}{n^\beta} \sum_{k \in [K]} |C_k| (|C_k| - 1) \\ &\geq 1 - \frac{2K|\widetilde{C}|^2}{n^\beta}, \end{aligned}$$

where  $|\widetilde{C}|^2 = \frac{1}{K} \sum_{k \in [K]} |C_k|^2$ , the first inequality follows from the union bound, the second inequality follows from (17), while the third inequality follows from the union bound and (16). Next, for any  $k, l \in [K]$ , we have that

$$\|\bar{\theta}_k - \bar{\theta}_l\| \geq \|\theta_k^* - \theta_l^*\| - \|\bar{\theta}_k - \theta_k^*\| - \|\bar{\theta}_l - \theta_l^*\|. \quad (18)$$

For any  $\gamma > 0$  and any  $k \in [K]$ , applying Chebyshev's inequality and Lemma 2, we get the following bound

$$\begin{aligned} \mathbb{P}(\|\bar{\theta}_k - \theta_k^*\| > \gamma) &\leq \frac{\mathbb{E}\|\bar{\theta}_k - \theta_k^*\|^2}{\gamma^2} \leq \frac{2E_k}{n|C_k|\gamma^2} \\ &+ \frac{5(H_k^2 \log d + E_k)E_k}{\mu_{F_k}^2 n^2 \gamma^2} + \mathcal{O}\left(\frac{1}{|C_k|n^2 \gamma^2}\right) + \mathcal{O}\left(\frac{1}{n^3 \gamma^2}\right). \end{aligned} \quad (19)$$

Define the event  $\Lambda = \cap_{k \in [K]} \{\omega : \|\bar{\theta}_k - \theta_k^*\| \leq \gamma\}$ . We then have

$$\begin{aligned}
\mathbb{P}(\Lambda) &\geq 1 - \sum_{k \in [K]} \mathbb{P}(\|\bar{\theta}_k - \theta_k^*\| > \gamma) \\
&\geq 1 - \sum_{k \in [K]} \left( \frac{2E_k}{n|C_k|\gamma^2} + \frac{5(H_k^2 \log d + E_k)E_k}{\mu_{F_k}^2 n^2 \gamma^2} \right. \\
&\quad \left. + \mathcal{O}\left(\frac{1}{|C_k|n^2 \gamma^2}\right) + \mathcal{O}\left(\frac{1}{n^3 \gamma^2}\right) \right) \\
&\geq 1 - \frac{2K\tilde{E}}{n|C_{(K)}|\gamma^2} - \frac{5K(\tilde{H}^2 \log d + \tilde{E})E_{\max}}{\mu_{F_{\min}}^2 n^2 \gamma^2} \\
&\quad - \Omega\left(\frac{K}{|C_{(K)}|n^2 \gamma^2}\right) - \Omega\left(\frac{K}{n^3 \gamma^2}\right),
\end{aligned}$$

where  $\tilde{E} = \frac{1}{K} \sum_{k \in [K]} E_k$ ,  $\tilde{H}^2 = \frac{1}{K} \sum_{k \in [K]} H_k^2$ ,  $E_{\max} = \max_{k \in [K]} E_k$  and  $\mu_{F_{\min}} = \min_{k \in [K]} \mu_{F_k}$ . Recall that  $D = \min_{k \neq l} \|\theta_k^* - \theta_l^*\|$ . Applying (18), we then have that on  $\Lambda$ , for any  $k, l \in [K]$

$$\|\bar{\theta}_k - \bar{\theta}_l\| \geq D - 2\gamma, \quad (20)$$

which is valid for any  $\gamma < \frac{D}{2}$ . Next, notice that on  $\Sigma$ , for all  $i, j \in C_k$ ,  $k \in [K]$ , we have

$$\|\hat{\theta}_i - \hat{\theta}_j\| \leq \sqrt{\frac{2(M_{ik} + M_{jk}) \log n}{n}}. \quad (21)$$

Plugging (20) and (21) in (12), we get that the true clustering can be recovered if

$$\sqrt{\frac{2M \log n}{n}} < \frac{|C_{(K)}|(D - 2\gamma)}{2m - |C_{(K-1)}| - |C_{(K)}|}, \quad (22)$$

where  $M = \max_{i,j \in C_k, k \in [K]} (M_{ik} + M_{jk})$ . For (22) to hold we need the number of samples per user to be such that

$$\frac{n}{\log n} > \frac{2M(2m - |C_{(K-1)}| - |C_{(K)}|)^2}{|C_{(K)}|^2(D - 2\gamma)^2}. \quad (23)$$

We then have that on  $\Psi = \Sigma \cap \Lambda$ , if the number of samples per user satisfies (23), the true clustering can be recovered and Lemma 2 applies to our

method. On the other hand, we have

$$\begin{aligned}
\mathbb{P}(\Psi) &\geq \mathbb{P}(\Sigma) + \mathbb{P}(\Lambda) - 1 \\
&\geq 1 - \frac{2K\tilde{E}}{n|C_{(K)}|\gamma^2} - \frac{5K\left(\tilde{H}^2 \log d + \tilde{E}\right) E_{\max}}{\mu_{F_{\min}}^2 n^2 \gamma^2} \\
&\quad - \Omega\left(\frac{K}{|C_{(K)}|n^2 \gamma^2}\right) - \Omega\left(\frac{K}{n^3 \gamma^2}\right) - \frac{2K|\tilde{C}|^2}{n^\beta},
\end{aligned}$$

which implies

$$\begin{aligned}
\mathbb{P}(\Psi^c) &\leq \frac{2K\tilde{E}}{n|C_{(K)}|\gamma^2} + \frac{2K|\tilde{C}|^2}{n^\beta} + \mathcal{O}\left(\frac{K}{|C_{(K)}|n^2 \gamma^2}\right) \\
&\quad + \frac{5K\left(\tilde{H}^2 \log d + \tilde{E}\right) E_{\max}}{\mu_{F_{\min}}^2 n^2 \gamma^2} + \mathcal{O}\left(\frac{K}{n^3 \gamma^2}\right)
\end{aligned}$$

Combining everything in (14), we finally get

$$\begin{aligned}
\mathbb{E}\|\bar{\theta}_k - \theta_k^*\|^2 &\leq \mathbb{E}\|\hat{\theta}_k - \theta_k^*\|^2 \mathbb{I}_\Psi + R^2 \mathbb{P}(\Psi^c) \\
&\leq \frac{2E_k}{n|C_k|} + \frac{2K\tilde{E}R^2}{n|C_{(K)}|\gamma^2} + \frac{2KR^2|\tilde{C}|^2}{n^\beta} + \mathcal{O}\left(\frac{1}{|C_k|n^2}\right) \\
&\quad + \frac{5E_k}{\mu_{F_k}^2 n^2} (H_k^2 \log d + E_k) + \frac{5R^2 K E_{\max}}{\mu_{F_{\min}}^2 n^2 \gamma^2} (\tilde{H}^2 \log d + \tilde{E}) \\
&\quad + \mathcal{O}\left(\frac{K}{|C_{(K)}|n^2 \gamma^2}\right) + \mathcal{O}\left(\frac{1}{n^3}\right) + \mathcal{O}\left(\frac{K}{n^3 \gamma^2}\right),
\end{aligned}$$

for  $n$  satisfying (23).  $\square$

## 4.2 Comparison with order-optimal CFL methods

In this section we compare the results from Theorem 1 with the guarantees of other CFL methods. As discussed in the Introduction, many methods for CFL have been proposed, with various requirements and guarantees. For example, we can split the methods into the ones requiring knowledge of the number of clusters  $K$ , e.g., [28], [29], [30] and the ones not requiring it, e.g., [31], [32]. On the other hand, we can split them into the ones that estimate the true clustering iteratively, e.g., [28], [29], [31], [32] and the ones that perform clustering only once during training, [30]. While all the proposed methods offer certain advantages (as illustrated by the previous classification), we specifically compare our method with two methods, namely

*Iterative Federated Clustering Algorithm* (IFCA), from [29], and the method from [30]. The main reasons for comparing with these specific methods are: 1) both methods analyze their performance in terms of statistical guarantees; 2) both methods are order-optimal, up to logarithmic factors and 3) both methods derive explicit requirements for the number of communication rounds. In what follows, due to some similarities of the methods and their performances, we will provide a detailed outline of IFCA, while highlighting where [30] differs significantly.

IFCA is an iterative algorithm for CFL that alternates between the following two steps: inferring cluster membership and updating the models. To that end, IFCA is initialized by first producing  $K$  different models  $\{\theta_k^0\}_{k \in [K]}$ , where the superscript denotes the iteration counter. The method then proceeds as follows, for  $t = 0, \dots, T - 1$ :

1. The server broadcasts  $\{\theta_k^t\}_{k \in [K]}$  to each user.
2. Each user evaluates the models on their local data and chooses the model  $\theta_{(i)}^t$ , where  $(i) = \arg \min_{k \in [K]} f_i(\theta_k^t)$ .
3. Each user computes the local stochastic gradient  $g_i^t = \tilde{\nabla} f_i(\theta_{(i)}^t)$ , evaluated at  $\theta_{(i)}^t$ . Users send the gradients back to the server, along with a one-hot encoding vector  $s_i \in \mathbb{R}^K$ , such that  $s_{ij} = \begin{cases} 1, & (i) = j \\ 0, & (i) \neq j \end{cases}$ , notifying the server which model was updated by user  $i$ .
4. The server forms clusters of users that updated specific models, based on the received tokens  $\{s_i\}_{i \in [m]}$  and performs the model update, i.e.,  $\theta_k^{t+1} = \theta_k^t - \alpha \frac{1}{|C_k^t|} \sum_{i \in C_k^t} g_i^t$ , where  $\alpha > 0$  is the step-size, while  $C_k^t = \{i \in [m] : s_{ik} = 1\}$  is the cluster of users that updated model  $k$  at iteration  $t$ .

On the other hand, the method in [30] can be seen as a modular method, as it depends on the following three steps:

1. Each user trains the local ERMs and sends them to the server.
2. The server performs a clustering procedure.
3. A FL algorithm is run on the resulting clusters for  $T$  iterations to produce the final models.



From the algorithm above, we can see that both IFCA and the method [30] require knowledge of the true number of distributions  $K$  (or at least a good estimate), which is typically unavailable or would require running a separate learning algorithm in practice (e.g., community detection). Secondly, both require  $T$  rounds of communication, whereas our method requires a single round of server-user communication. Comparing to our algorithm, IFCA alleviates the computational requirements on the server side, by only requiring the server to average the received models. On the other hand, our algorithm assumes that the server has enough computation resources to run the convex clustering algorithm and perform inference on the underlying clustering structure, significantly decreasing the communication cost.

**Assumptions.** Similarly to our algorithm, IFCA assumes that the population risks  $F_k$ ,  $k \in [K]$  are  $L$ -smooth and  $\mu_{F_k}$  strongly convex. The method [30] requires a stronger assumption - namely, that the loss function  $\ell$  is strongly convex. Additionally, IFCA assumes bounded variance of  $\ell$  with respect to all  $\mathcal{D}_k$ ,  $k \in [K]$ , i.e.,

$$\mathbb{E}_{X \sim \mathcal{D}_k} [(\ell(\theta; X) - F_k(\theta))^2] \leq \eta^2,$$

for some  $\eta > 0$ . Intuitively, this assumption is made to ensure that the empirical loss  $f_i$  of user  $i \in C_k$  stays close to the true population loss  $F_k$ , enabling clustering inference via the local loss. An additional assumption made by IFCA, that is required for the convergence of the algorithm is *sufficiently close initialization*, i.e., for all  $k \in [K]$ , the authors require

$$\|\theta_k^0 - \theta_k^*\| \leq \left(\frac{1}{2} - \alpha_0\right) D \sqrt{\frac{\mu_{F_{\min}}}{L}},$$

where  $\alpha_0 \in (0, \frac{1}{2})$  is a tunable parameter that determines the proximity of the initialization to the true population optima. Note that such an assumption is quite strong, as it requires  $\|\theta_k^0 - \theta_k^*\| < \frac{1}{2}D$ , for all  $k \in [K]$ . In order to find such an initialization, the knowledge of underlying clusters, as well as  $D$ , would have to be available. The method [30], like our method, does not require such an assumption. IFCA requires three further assumptions:

1.  $|C_{(K)}| \gtrsim \log(mn)^3$ , i.e., the size of the smallest cluster has to grow at least logarithmically in the number of total samples available in the system;

---

<sup>3</sup>Note that the authors in [29] use  $n' = \frac{n}{2T}$  in their theoretical analysis, i.e., they require that each user contains  $n = 2Tn'$  samples and all conditions in the original paper are expressed in terms of  $n'$ . However, for the sake of simplicity, we will represent the conditions in terms of  $n$ , effectively reducing the original sample size requirement by a factor of  $2T$ .

2.  $n \gtrsim \frac{K\eta^2}{\alpha_0^2 \mu_{F_{\min}}^2 D^4}$ , i.e., each user contains a sufficient number of samples;
3.  $D \geq \tilde{\mathcal{O}} \left( \max \left\{ \alpha_0^{-2/5} n^{-1/5}, \alpha_0^{-1/3} m^{-1/6} n^{-1/3} \right\} \right)$ , i.e., the population optimal models across different populations are sufficiently well separated.

Here, the operator  $x \gtrsim y$  indicates the existence of global constant  $C$  that does not depend on the problem parameters, such that  $x \geq Cy$  (the operator  $x \lesssim y$  is defined similarly), while  $\tilde{\mathcal{O}}(\cdot)$  hides logarithmic factors that do not depend on  $m$  and  $n$ . On the other hand, both our method and the method [30] do not require any assumptions on the separation parameter  $D$  or on the size of the smallest cluster  $|C_{(K)}|$ , effectively covering the cases for which IFCA might fails, i.e., highly unbalanced clusters, e.g.,  $|C_{(K)}| = \mathcal{O}(1)$  and small separation between optimal models of different populations. Comparing the requirements on the number of samples of our method, IFCA and [30], we have, respectively,

$$\begin{aligned} \frac{n}{\log n} &> \frac{2M(2m - |C_{(K-1)}| - |C_{(K)}|)^2}{|C_{(K)}|^2(D - 2\gamma)^2}, \\ n &\gtrsim \frac{K\eta^2}{\alpha_0^2 \mu_{F_{\min}}^2 D^4}, \\ n &\geq \frac{G_{F_{\max}}^2 L \log m}{\mu_{F_{\min}}^3}, \end{aligned}$$

where  $G_{F_{\max}} = \max_{k \in [K]} G_{F_k}$ . We can first see that that for our method, the requirement is expressed in terms of  $n/\log n$ , which, for  $n \geq 3$  is always more relaxed than placing requirements directly on  $n$ . Comparing the right-hand sides of the inequalities, we can see that the dependence of IFCA on  $K$  is much better, as (recall Remark 13) the term  $\frac{(2m - |C_{(K-1)}| - |C_{(K)}|)^2}{|C_{(K)}|^2}$  evaluates to  $4(K - 1)^2$  in the best case, while being  $4(m - 1)^2$  in the worst case. The method [30] depends logarithmically on  $m$ . For  $D > 1$ , the dependence of IFCA is again better, while, for  $D < 1$ , our method has a much better dependence. Finally, the dependence on the problem parameters, encapsulated in  $M$ , are again better for IFCA, as typically one would expect  $M > \frac{\eta^2}{\mu_{F_{\min}}^2}$ .

However, we stress that IFCA and [30] are iterative algorithms, allowing for multiple rounds of communication, whereas the method we propose is a one-shot method. Therefore, the higher requirements on the number of samples are to be expected, but uncover regimes in which communicating beyond one round to achieve order-optimality is redundant. Additionally,

our method does not require knowledge of  $K$ , while both IFCA and [30] assume the knowledge of the true value of  $K$ . All of these facts lead to less strict requirements of IFCA on the number of samples per user, with respect to different problem parameters (in some cases).

**Guarantees.** The guarantees of IFCA are given in terms of high probability bounds, while our guarantees, expressed in terms of the MSE, are sharper. IFCA provides the following guarantee (Corollary 2 in [29]): after  $T = \frac{8mL}{|C_{(K)}|\mu_{F_{\min}}} \log\left(\frac{2D}{\varepsilon}\right)$  communication rounds, with probability at least  $1 - \delta$

$$\|\theta_k^T - \theta_k^*\| \leq \varepsilon,$$

where

$$\begin{aligned} \varepsilon \lesssim & \frac{\sigma_{\max}KL \log(mn) (m/|C_{(K)}|)^2}{\mu_{F_{\min}} \delta \sqrt{n|C_{(K)}|}} + \tilde{\mathcal{O}}\left(\frac{1}{n\sqrt{m}}\right) \\ & + \frac{\eta^2 L^2 (m/|C_{(K)}|)^2 K \log(mn)}{\mu_{F_{\min}}^4 \delta D^4 n}. \end{aligned} \quad (24)$$

We can see that, assuming  $n \geq |C_{(1)}|$ , the dominating term in (24) becomes

$$\|\theta_k^T - \theta_k^*\| = \mathcal{O}\left(\frac{\log(mn)}{\sqrt{n|C_{(K)}|}}\right),$$

for all  $k \in [K]$ , which is almost order-optimal, up to a logarithmic factor and dependence on the smallest cluster size. The guarantees of [30] are similar, i.e., via Theorem 1 in [30], we have that: after  $T = \mathcal{O}\left(\frac{L+\mu_{F_{\max}}}{\mu_{F_{\min}}} \log\left(\frac{\mu_{F_{\max}}}{2\varepsilon}\right)\right)$  communication rounds, with high probability, for all  $k \in [K]$

$$\|\theta_k^T - \theta_k^*\| \leq \mathcal{O}\left(\frac{\log mn}{\sqrt{n|C_k|}}\right).$$

On the other hand, from Theorem 1, for  $n \geq |C_{(1)}|$ , we have

$$\mathbb{E}\|\bar{\theta}_k - \theta_k^*\| = \mathcal{O}\left(\frac{1}{\sqrt{n|C_{(K)}|}}\right),$$

for all  $k \in [K]$ , which is almost order-optimal, with the dependence on the smallest cluster size. Therefore, we can see that our method removes

the logarithmic dependence on the total number of samples, of both IFCA and [30], while simultaneously reducing the communication cost by a factor of  $\mathcal{O}\left(\frac{\kappa}{p} \log\left(\frac{2D}{\varepsilon}\right)\right)$  with respect to IFCA (and similar with respect to [30]), where  $\kappa = \frac{L}{\mu_{F_{\min}}} \geq 1$  is the condition number, while  $p = \frac{|C_{(K)}|}{m} < 1$  is the fraction of users belonging to the smallest cluster, reflecting the difficulty of the clustering problem.

However, we can see that Theorem 1 provides guarantees in terms of the size of the smallest cluster,  $|C_{(K)}|$ , while Theorem 1 in [30] provides the guarantees in terms of the true cluster size  $|C_k|$ , for each  $k \in [K]$ . Applying Corollary 1, for  $D > 2\sqrt{\frac{|C_{(1)}|}{|C_{(K)}|}}$ , our method matches the dependence on individual cluster sizes of [30], while removing the logarithmic dependence on the total number of samples, thus achieving the order-optimal rate

$$\mathbb{E}\|\bar{\theta}_k - \theta_k^*\| = \mathcal{O}\left(\frac{1}{\sqrt{n|C_{(k)}|}}\right),$$

for all  $k \in [K]$ , while still providing a reduction in communication cost by a factor of  $\mathcal{O}\left(\frac{\kappa}{p} \log\left(\frac{2D}{\varepsilon}\right)\right)$ . Hence, we can see that our method provides order-optimal convergence guarantees, improving on the guarantees of both IFCA and [30] by a factor logarithmic in the total number of samples in the system. Remarkably, this is achieved while simultaneously reducing the communication cost by a factor of  $\mathcal{O}\left(\frac{\kappa}{p} \log\left(\frac{2D}{\varepsilon}\right)\right)$  and without requiring any knowledge of the underlying structure, while both IFCA and [30] assume knowledge of  $K$ .

### 4.3 Inexact ERMs

In this section we consider replacing the ERM model  $\hat{\theta}_i = \arg \min_{\theta_i \in \Theta} f_i(\theta_i)$ ,  $i \in [m]$ , by an *inexact estimate*, i.e., an estimate  $\tilde{\theta}_i \in \Theta$ , such that

$$\|\tilde{\theta}_i - \hat{\theta}_i\| \leq \varepsilon, \tag{25}$$

for some  $\varepsilon > 0$ . To that end, we need an additional assumption on the strong convexity of the empirical losses  $f_i$ ,  $i \in [m]$ .

**Assumption 6.** *For all  $i \in [m]$  the empirical loss  $f_i$  is strongly convex, i.e., there exists a constant  $\mu_{f_i} > 0$ , such that, for all  $\theta, \theta' \in \Theta$ , we have*

$$f_i(\theta') \geq f_i(\theta) + \langle \nabla f_i(\theta), \theta' - \theta \rangle + \frac{\mu_{f_i}}{2} \|\theta - \theta'\|^2.$$

Denote by  $\mu_f = \min_{i \in [m]} \mu_{f_i}$ .

**Remark 15.** Note that in general, Assumption 6 allows for the loss function  $\ell$  to be convex, as long as the average across local samples,  $f_i(\theta) = \frac{1}{n} \sum_{j=1}^n \ell(\theta; x_{ij})$  is strongly convex.

Assumption 6 allows for each user to apply iterative solvers, to obtain parameters  $\hat{\theta}_i$  that satisfy (25). A standard choice is the stochastic gradient descent (SGD) algorithm [43]. SGD follows a simple update rule, given by

$$\theta^{t+1} = \theta^t - \eta^t g^t,$$

where  $\theta^t$  is the estimate of the parameter of interest at iteration  $t$ ,  $\eta^t$  is the step-size and  $g^t$  is a stochastic gradient, evaluated at  $\theta^t$ .

SGD can be implemented in both the online setting, where users only have access to a single stochastic gradient at a time and in the batch setting, where users have access to the entire local dataset. Additionally, SGD offers the most general guarantees with respect to the mini-batch size and can be implemented even with a mini-batch size of 1. We discuss at the end of the section how different assumptions can allow for the implementation of more efficient algorithms, in terms of the local iteration complexity per user. Next, we state an additional assumption on the stochastic gradients of  $f_i$ .

**Assumption 7.** For each  $i \in [m]$  and all  $\theta \in \Theta$ , stochastics gradient  $g_i$  of  $f_i$ , evaluated at  $\theta$ , are unbiased, i.e.,  $\mathbb{E}[g_i] = \nabla f_i(\theta)$ . Additionally, the stochastic gradients have bounded variance, i.e., there exists a  $\sigma_i > 0$ , such that for all  $\theta \in \Theta$ , we have

$$\mathbb{E}\|g_i - \nabla f_i(\theta)\|^2 \leq \sigma_i^2.$$

**Remark 16.** Assumption 7 is standard in the analysis of stochastic algorithms, e.g., [44], [45], [46].

**Remark 17.** Recall the discussion in Section 2 and Remarks 5-7, that imply bounded gradients of  $f_i$ , with constant  $G_{f_i}$ . Combining with Assumption 7, it then follows that, for all  $\theta \in \Theta$

$$\mathbb{E}\|g_i\|^2 \leq 2\mathbb{E}\|g_i - \nabla f_i(\theta)\|^2 + 2\|\nabla f_i(\theta)\|^2 \leq 2\sigma_i^2 + 2G_{f_i}^2.$$

Define  $\Gamma_i^2 := 2\sigma_i^2 + 2G_{f_i}^2$ ,  $i \in [m]$  and denote by  $\Gamma^2 = \max_{i \in [m]} \Gamma_i^2$ . We now state two well-known result on the convergence of SGD from [44], used in the rest of the section.

**Lemma 3** (Lemma 1 in [44]). *Under Assumptions 2, 3, 6 and 7, for all  $i \in [m]$ , if we set the step-size rule of SGD as  $\eta^t = \frac{1}{\mu_{f_i} t}$ , it holds for any  $T \geq 1$  and any  $i \in [m]$  that*

$$\mathbb{E}\|\theta_i^T - \hat{\theta}_i\|^2 \leq \frac{4\Gamma_i^2}{\mu_{f_i}^2 T}.$$

**Lemma 4** (Lemma 2 in [44]). *Let Assumptions 2, 3, 6 and 7 hold and let  $\|g^t\|^2 \leq \Gamma^2$ , with probability 1. Then, for all  $i \in [m]$  and any  $\delta \in (0, 1/e)$ ,  $T \geq 4$ , if we set the step-size rule of SGD as  $\eta^t = \frac{1}{\mu_{f_i} t}$ , it holds with probability  $1 - \delta$ , for any  $t \in \{8, \dots, T-1, T\}$  and any  $i \in [m]$ , that*

$$\|\theta_i^t - \hat{\theta}_i\|^2 \leq \frac{12\Gamma^2}{\mu_{f_i}^2 t} + 8G(121G + 1) \frac{\log(\log(t)/\delta)}{t}.$$

We are now ready to state and prove counterparts of Lemmas 1 and 2, when an inexact ERM estimator is used.

**Lemma 5.** *Let Assumptions 1-4, 6 and 7 hold and  $\|g^t\|^2 \leq \Gamma^2$  with probability 1. If each user runs SGD locally for  $T$  iterations, with the step-size rule  $\eta^t = \frac{1}{\mu_f t}$ , to produce  $\theta_i^T$ ,  $i \in [m]$  and  $T$  is chosen such that  $T \geq 15$  and  $\frac{T}{\log \log(T)} \geq \left( \frac{12\Gamma^2}{\mu_f^2} + 8G(121G + 1)(1 + \log \frac{1}{\delta}) \right) \frac{1}{\epsilon^2}$ , then for any  $k \in [K]$ , any  $i \in C_k$  and any  $\epsilon > 0$ ,  $0 < \delta < \frac{1}{3}$ , with probability at least  $1 - 3\delta$ , we have, for any  $i \in C_k$ ,  $k \in [K]$*

$$\begin{aligned} F_k(\theta_i^T) - F_k(\theta_k^*) &\leq \frac{16R^2 LC(\epsilon, \delta)}{n} + \frac{8R\|\nabla f_i(\theta_k^*)\| \log \frac{2}{\delta}}{n} \\ &\quad + \frac{8LF_k(\theta_k^*) \log \frac{2}{\delta}}{\mu_{F_k} n} + \left( 8RL + G_{F_k} + \frac{4RLC(\epsilon, \delta)}{n} \right) \epsilon \\ &\quad + \epsilon G_{F_k}, \end{aligned}$$

where  $C(\epsilon, \delta) := 2 \left( \log \frac{2}{\delta} + d \log \frac{6R}{\epsilon} \right)$ .

*Proof.* For any  $\theta \in \Theta$ , any  $k \in [K]$  and any  $i \in C_k$ , we have

$$F_k(\theta) - F_k(\theta_k^*) \leq \left| F_k(\theta) - F_k(\hat{\theta}_i) \right| + F_k(\hat{\theta}_i) - F_k(\theta_k^*). \quad (26)$$

We can bound the second term on the right hand side of (26) using Lemma 1. To bound the first term, we use Lipschitz continuity of  $F_k$  (recall the discussion in Section 2), to get

$$\left| F_k(\theta) - F_k(\hat{\theta}_i) \right| \leq G_{F_k} \|\theta - \hat{\theta}_i\|. \quad (27)$$

Next, applying Lemma 4, we have that, with probability at least  $1 - \delta$

$$\|\theta_i^T - \hat{\theta}_i\|^2 \leq \frac{12\Gamma^2}{\mu_f^2 T} + 8G(121G + 1) \frac{\log(\log(T)/\delta)}{T}.$$

As  $T \geq 15$ , we can use the following upper-bound, with probability at least  $1 - \delta$

$$\begin{aligned} \|\theta_i^T - \hat{\theta}_i\|^2 &\leq \frac{12\Gamma^2}{\mu_f^2} \frac{\log \log(T)}{T} \\ &\quad + 8G(121G + 1) \left(1 + \log \frac{1}{\delta}\right) \frac{\log \log(T)}{T}. \end{aligned}$$

From the conditions of the Lemma, we can then conclude that

$$\|\theta_i^T - \hat{\theta}_i\| \leq \varepsilon. \quad (28)$$

Plugging (28) into (27) and combining in (26), we finally get that, with probability at least  $1 - \delta$

$$F_k(\theta_i^T) - F_k(\theta_k^*) \leq \varepsilon G_{F_k} + F_k(\hat{\theta}_i) - F_k(\theta_k^*).$$

The result is completed by applying Lemma 1 to the second term on the right hand side of the final inequality.  $\square$

**Lemma 6.** *Let Assumptions 1-7 hold and each user runs SGD locally for  $T$  iterations, to produce  $\theta_i^T$ . If  $T \geq \frac{4\Gamma^2}{\mu_f^2 \varepsilon}$ , then for  $\tilde{\theta}_k = \frac{1}{|C_k|} \sum_{i \in C_k} \theta_i^T$ ,  $k \in [K]$ , we have*

$$\begin{aligned} \mathbb{E}\|\tilde{\theta}_k - \theta_k^*\|^2 &\leq \frac{4E_k}{n|C_k|} + \frac{10}{\mu_{F_k}^2 n^2} (H_k^2 \log d + E_k) E_k \\ &\quad + \mathcal{O}(|C_k|^{-1} n^{-2}) + \mathcal{O}(n^{-3}) + \varepsilon, \end{aligned}$$

where  $E_k := \mathbb{E}\|\nabla^2 F_k(\theta_k^*)^{-1} \nabla \ell(\theta_k^*; X)\|^2$ .

*Proof.* From Lemma 3, we know that, for each  $i \in [m]$ , running SGD locally for  $T \geq \frac{4\Gamma^2}{\mu_f^2 \varepsilon}$  iterations results in

$$\mathbb{E}\|\theta_i^T - \hat{\theta}_i\|^2 \leq \varepsilon. \quad (29)$$

Define the across-cluster average of  $\varepsilon$ -inexact approximations as  $\tilde{\theta}_k = \frac{1}{|C_k|} \sum_{i \in C_k} \theta_i^T$ . We then have

$$\mathbb{E}\|\tilde{\theta}_k - \theta_k^*\|^2 \leq 2\mathbb{E}\|\bar{\theta}_k - \theta_k^*\|^2 + 2\mathbb{E}\|\tilde{\theta}_k - \bar{\theta}_k\|^2, \quad (30)$$

where  $\bar{\theta}_k = \frac{1}{|C_k|} \sum_{i \in C_k} \hat{\theta}_i$ . We can bound the first term on the right-hand side of (30) using Lemma 2. For the second term, we use (29), to obtain

$$\mathbb{E} \|\tilde{\theta}_k - \bar{\theta}_k\|^2 \leq \frac{1}{|C_k|} \sum_{i \in C_k} \mathbb{E} \|\theta_i^T - \hat{\theta}_i\|^2 = \varepsilon.$$

Combining the results and plugging in (30), we get

$$\begin{aligned} \mathbb{E} \|\tilde{\theta}_k - \theta_k^*\|^2 &\leq \frac{4E_k}{n|C_k|} + \frac{10}{\mu_{F_k}^2 n^2} (H_k^2 \log d + E_k) E_k \\ &\quad + \mathcal{O}(|C_k|^{-1} n^{-2}) + \mathcal{O}(n^{-3}) + \varepsilon, \end{aligned}$$

which completes the proof.  $\square$

Lemmas 5 and 6 give us the counterparts of Lemmas 1 and 2 in the case where an approximate solution to the ERM is used instead of the exact one. We can apply them to prove the following.

**Theorem 2.** *Let Assumptions 1-7 hold and  $\|g^t\|^2 \leq \Gamma^2$  with probability 1. If each user runs SGD locally for  $T$  iterations to produce  $\theta_i^T$ ,  $i \in [m]$  and the number of samples per user  $n$  and the number of local iterations  $T$  are such that  $n > 3$ ,  $T \geq \max \left\{ 15, \frac{4\Gamma^2}{\mu_f^2 \varepsilon} \right\}$  and moreover*

$$\begin{aligned} \frac{n}{\log n} &> 2M \left( \frac{(D - 2\gamma)^2 |C_{(K)}|^2}{(2m - |C_{(K)}| - |C_{(K-1)}|)^2} - 4\varepsilon S_F \right)^{-1}, \\ \frac{T}{\log \log(T)} &\geq \left( \frac{12\Gamma^2}{\mu_f^2} + 8G(121G + 1)(1 + \beta \log n) \right) \frac{1}{\varepsilon^2}, \end{aligned}$$

where  $\beta \geq 1$  and  $0 < \gamma < \frac{D}{2}$  are tunable parameters,  $S_F = \max_{k \in [K]} \frac{G_{F_k}}{\mu_{F_k}}$ , while  $M = M(\beta) = \max_{i,j \in C_k, k \in [K]} M_{ik} + M_{jk}$ , and for all  $i \in C_k$ ,  $k \in [K]$

$$\begin{aligned} M_{ik} &= \frac{64R^2 L (\log 2 + d \log 6R + (d+1)\beta)}{\mu_{F_k}} \\ &\quad + \frac{16LF_k(\theta_k^*)(\log 2 + \beta)}{\mu_{F_k}^2} + \frac{16R \|\nabla f_i(\theta_k^*)\|(\log 2 + \beta)}{\mu_{F_k}} \\ &\quad + \frac{2G_{F_k} + 16RL (1 + \log 2 + d \log 6R + (d+1)\beta)}{\mu_{F_k}}, \end{aligned}$$



then, for any choice of  $\lambda \in \left[ \sqrt{\frac{2M \log n}{n}} + 4\varepsilon S_F, \frac{|C_{(K)}|(D-2\gamma)}{2m-|C_{(K-1)}|-|C_{(K)}|} \right)$ , we have that, for all  $k \in [K]$ , the models produced by the inexact method achieve the MSE

$$\begin{aligned} \mathbb{E}\|\tilde{\theta}_k - \theta_k^*\|^2 &\leq \frac{4E_k}{n|C_k|} + \frac{4K\tilde{E}R^2}{n|C_{(K)}|\gamma^2} + \frac{3KR^2|\tilde{C}|^2}{n^\beta} \\ &+ \frac{10E_k}{\mu_{F_k}^2 n^2} (H_k^2 \log d + E_k) + \frac{10R^2 K E_{\max}}{\mu_{F_{\min}}^2 n^2 \gamma^2} (\tilde{H}^2 \log d + \tilde{E}) \\ &+ \mathcal{O}\left(\frac{1}{|C_k|n^2}\right) + \mathcal{O}\left(\frac{K}{|C_{(K)}|n^2\gamma^2}\right) + \mathcal{O}\left(\frac{1}{n^3}\right) \\ &+ \mathcal{O}\left(\frac{K}{n^3\gamma^2}\right) + \varepsilon \left(1 + \frac{2R^2 K}{\gamma^2}\right), \end{aligned}$$

where  $E_k = \mathbb{E}\|\nabla^2 F_k(\theta_k^*)^{-1} \nabla \ell(\theta_k^*, X)\|^2$ ,  $E_{\max} = \max_{k \in [K]} E_k$ ,  $\tilde{E} = \frac{1}{K} \sum_{k \in [K]} E_k$ ,  $\tilde{H} = \frac{1}{K} \sum_{k \in [K]} H_k$  and  $|\tilde{C}|^2 = \frac{1}{K} \sum_{k \in [K]} |C_k|^2$ .

We can provide an analogue to Corollary 1 in the inexact ERM scenario.

**Corollary 2.** *Let conditions of Theorem 2 hold. If additionally  $D > 2\sqrt{\frac{|C_{(1)}|}{|C_{(K)}|}}$*

*and  $n \geq |C_{(1)}|$ , then for the choices of  $\beta \geq 2$  and  $\gamma = \sqrt{\frac{|C_{(1)}|}{|C_{(K)}|}}$ , we have the following MSE, for all  $k \in [K]$*

$$\mathbb{E}\|\tilde{\theta}_k - \theta_k^*\|^2 \leq \mathcal{O}\left(\frac{1}{n|C_k|} + \varepsilon\right).$$

The proof of Theorem 2 follows the same idea as the proof of Theorem 1, replacing the results of Lemmas 1 and 2 with results from Lemmas 5 and 6. For the sake of brevity, we omit the proof. Some comments are now in order.

**Remark 18.** *Comparing the MSE rates of Theorem 1 and Theorem 2, we can see that the main difference is the presence of an additional term in Theorem 2, that being*

$$\varepsilon \left(1 + \frac{2R^2 K}{\gamma^2}\right),$$

*with  $\varepsilon > 0$  representing the accuracy up to which we solve the local ERM. We can therefore see that, as long as the local ERMs are solved up to precision  $\varepsilon = \mathcal{O}\left(\frac{1}{n|C_{(1)}|}\right)$ , the rates of Theorem 1 are recovered, i.e., the final MSE*

is of the order  $\mathcal{O}\left(\frac{1}{n|C_{(k)}|}\right)$ , for all  $k \in [K]$ . This in turns leads to a local iteration requirement of  $T \geq \max\left\{15, \frac{4n|C_{(1)}|\Gamma^2}{\mu_\ell^2}\right\}$  and

$$\frac{T}{\log \log(T)} \geq \left(\frac{6L\Gamma^2}{\mu_\ell^2} + 4LG(121G + 1)(1 + \beta \log n)\right)n^2|C_{(1)}|^2.$$

**Remark 19.** We can see from Corollary 2 that, if we solve the local problems up to precision  $\varepsilon = \frac{1}{n|C_{(1)}|}$ , we again obtain the order-optimal MSE rates

$$\mathbb{E}\|\tilde{\theta}_k - \theta_k^*\|^2 = \mathcal{O}\left(\frac{1}{n|C_k|}\right),$$

for all  $k \in [K]$ .

**Remark 20.** Note that the sample size requirement implicitly places a requirement on the precision up to which we solve the local ERMs, i.e., we have

$$\varepsilon < \frac{(D - 2\gamma)^2|C_{(K)}|^2}{4S_F(2m - |C_{(K)}| - |C_{(K-1)}|)^2}.$$

This requirement can again be seen in terms of the "problem difficulty", with respect to different system aspects. For example, if the clusters are well separated, so that  $D - 2\gamma$  is large, we can solve the local ERMs up to moderate, or even low precision, while for clusters that are not well separated, we need to solve the local ERMs to high precision in order to achieve the optimal rates. Similarly, if the clusters are well balanced, i.e.,  $|C_k| = \frac{m}{K}$ , for all  $k \in [K]$ , the term  $\frac{|C_{(K)}|^2}{(2m - |C_{(K)}| - |C_{(K-1)}|)^2}$  evaluates to  $\frac{1}{4(K-1)^2}$ , while in the extreme case of  $|C_{(K)}| = |C_{(K-1)}| = 1$ , the term evaluates to  $\frac{1}{4(m-1)^2}$ . For  $K \ll m$ , we see that balanced clusters (easier clustering problem) again lead to a lower precision requirement than the imbalanced clusters case. Finally, recall that  $S_F = \max_{k \in [K]} \frac{G_{F_k}}{\mu_{F_k}}$ , where  $G_{F_k}$  is the Lipschitz constant of  $F_k$  (not the gradient!), while  $\mu_{F_k}$  is the strong convexity constant of  $F_k$ , hence showing that, if  $F_k$ 's are strongly convex (high  $\mu_{F_k}$ ) and don't have big jumps (low Lipschitz constant  $G_{F_k}$ ), the overall precision to which we have to solve the local ERMs is relaxed.

**Remark 21.** The choice of SGD as the local solver is based on the flexibility offered by the algorithm. The results from Lemmas 3 and 4 do not depend on either the setting being online or locally stored data, nor do they place any requirement on the mini-batch size used. This however leads to sub-optimal

dependence on  $\varepsilon$  in the requirements on the number of local iterations each user has to run.

**Remark 22.** *If all the  $n$  local data samples were available to each user, variance reduction methods such as SAGA [45] and SVRG [46] could be applied, making the number of iterations  $T$  dependence on  $\varepsilon$  only logarithmical, i.e.,  $T = \mathcal{O}(\log \frac{1}{\varepsilon})$ .*

**Remark 23.** *Finally, we remark that Assumption 6 is the most general form assumption on the loss function and as such, leads to the requirement of solving the ERM to precision  $\varepsilon^2$ . As shown in [33], Theorem 2, if the loss is a generalized linear loss, then it suffices to solve the ERM up to precision  $\varepsilon$ . While such an assumption is satisfied by a certain class of strongly convex loss functions, such as support vector machines, linear and logistic regression, it is less general than Assumption 6.*

## 5 Numerical experiments

In this section we present numerical experiments on linear regression problem. All of the experiments are implemented in python. To solve the local empirical risk problems, we use CVXPY [47]. The results presented in subsections below are averaged across 20 runs.

We consider a linear regression problem, where the data generating process for each cluster is given by

$$y = \langle x, u_k^* \rangle + \epsilon,$$

where  $\epsilon \sim \mathcal{N}(0, 1)$ , i.e.,  $\epsilon$  follows a standard Gaussian distribution. The number of clusters is set to  $K = 10$ . The vectors  $u_k^*$  are  $d$ -dimensional, with  $d = 20$ , and each component is drawn from a uniform distribution, independent of one another. Specifically, we drew  $u_k^*$ 's as:  $u_{1i}^* \sim \mathcal{U}([1, 2])$ ,  $u_{2i}^* \sim \mathcal{U}([4, 5])$ ,  $u_{3i}^* \sim \mathcal{U}([7, 8])$ ,  $u_{4i}^* \sim \mathcal{U}([10, 11])$  and  $u_{5i}^* \sim \mathcal{U}([13, 14])$ , with  $u_6^*$  through  $u_{10}^*$  begin generated from the corresponding negative intervals, i.e.,  $u_6^* \sim \mathcal{U}([-2, -1])$ , through to  $u_{10i}^* \sim \mathcal{U}([-14, -13])$ , respectively, for all  $i \in [d]$ . Such a choice of  $u_k^*$ 's ensures that  $D > 0$ . Each cluster is assigned a total of  $N_k = 100000$  points, where the datapoints  $x$  are generated as follows: for each  $x \in \mathbb{R}^d$ , we choose 5 random components in  $[d]$  that are distributed according to  $\mathcal{N}(0, 1)$ , while the other components are set to zero. A similar setup was considered in [14], with  $K = 1$ .

To measure the error, we use the quadratic loss, i.e.,

$$\ell((x, y); u) = (y - \langle x, u \rangle)^2.$$

Under the proposed loss, we have that  $u_k^*$ 's are the population optimal models, i.e.,  $u_k^* = \arg \min_u F_k(u)$ ,  $k \in [K]$ .

We consider a FL system with  $m = 100$  users and a balanced clustering, i.e.,  $|C_k| = \frac{m}{K} = 10$ , for all  $k \in [K]$ . Each user  $i \in C_k$  is assigned  $n$  points uniformly at random, from the corresponding sample  $N_k$ , such that no data point is assigned to two different users, effectively simulating an IID distribution of data within clusters. We benchmark the proposed method with the following methods:

- *Oracle Averaging* - an oracle method that knows the true clusters beforehand and applies the averaging method from [14] on each individual cluster, i.e.,

$$\bar{u}_k = \frac{1}{|C_k|} \sum_{i \in C_k} \hat{u}_i, \quad (31)$$

with  $\hat{u}_i$  the local ERM of user  $i$  and  $C_k$ ,  $k \in [K]$  being the true underlying clustering;

- *Cluster Oracle* - an oracle method that contains all of the data points assigned to the users from the same clusters, i.e., a total of  $\frac{mn}{K}$  data points per cluster and trains the models on all of the data, i.e.,

$$u_k = \arg \min_u \frac{K}{m} \sum_{i \in C_k} f_i(u),$$

with  $f_i$ 's given by (4);

- *Local ERMs* - ERMs trained on each user's local data;
- *Naive averaging* - the method from [14], that averages the local ERMs across all users, oblivious to system heterogeneity.

Cluster Oracle is the equivalent of centralized learning, i.e., is the method that trains on all the data available in the cluster, achieving the best order-optimal MSE rate  $\mathcal{O}\left(\frac{1}{n|C_k|}\right)$  (e.g., [33]). On the other hand, [14] show that Oracle Averaging matches the performance of Cluster Oracle if the sample size is above a threshold. Therefore, using Cluster Oracle and Oracle Averaging as benchmarks illustrates: 1) how fast our method attains the order-optimal MSE rate and 2) the additional requirements on the sample size to reach the order-optimal rate, compared to Oracle Averaging, that stem from not knowing the true clustering.

To measure the quality of performance, we present the average normalized MSE, i.e., for each of the above estimators, we compute

$$\frac{1}{m} \sum_{i=1}^m \frac{\|\tilde{u}_i - u_{(i)}^*\|^2}{\|u_{(i)}^*\|^2}, \quad (32)$$

where  $u_{(i)}^*$  denotes the population optima associated with user  $i$ , while  $\tilde{u}_i$  is the estimator associated with user  $i$ . For example, if we measure the performance of Oracle Averaging estimator from (31), (32) evaluates to

$$\frac{1}{K} \sum_{k \in [K]} \frac{\|\bar{u}_k - u_k^*\|^2}{\|u_k^*\|^2}.$$

To select the parameter  $\lambda$ , we first compute the lower and upper bounds in (12). If the condition is satisfied, so that the lower bound is strictly smaller than the upper bound, we choose  $\lambda$  uniformly at random from the interval defined by the lower and upper bounds in (12). Otherwise, for simplicity, we take  $\lambda$  to be equal to the upper bound.

Figure 1 presents the performance using the linear regression models. On  $y$ -axis we plot the averaged normalized MSE (32), while on the  $x$ -axis, we present the number of samples  $n$  available to each user. We can see that, for a small number of samples (less than 300), our method clusters each user to an individual cluster, effectively performing like the local ERMs. This can be explained by the fact that in the small sample size regime, the condition (10) is not satisfied (with high probability) and typically the upper bound will be small, hence resulting in a large number of clusters. The results can potentially be improved by running clusterpath, but for illustrative purposes, we went with the simple choice of setting  $\lambda$  to be equal to the upper bound. On the other hand, as  $n$  grows, we see a sharp phase transition in the quality of our estimator, in the interval between 300 and 400 samples, after which the performance of our method matches the order-optimal performance of both the oracle methods, as predicted by the theory. Oracle Averaging performs slightly worse than Cluster Oracle in the small sample regime, but quickly matches the performance of Cluster Oracle, as expected. The difference in the number of samples required for reaching order-optimal rates of our proposed method and the Oracle Averaging (450 and 350 samples required, respectively), as outlined above, stems from the additional requirements of our method to produce an exact clustering. Finally, we see that the naive averaging method consistently performs badly, as it is completely oblivious to the clustering structure, hence illustrating that a global model approach can be bad in the presence of system heterogeneity.

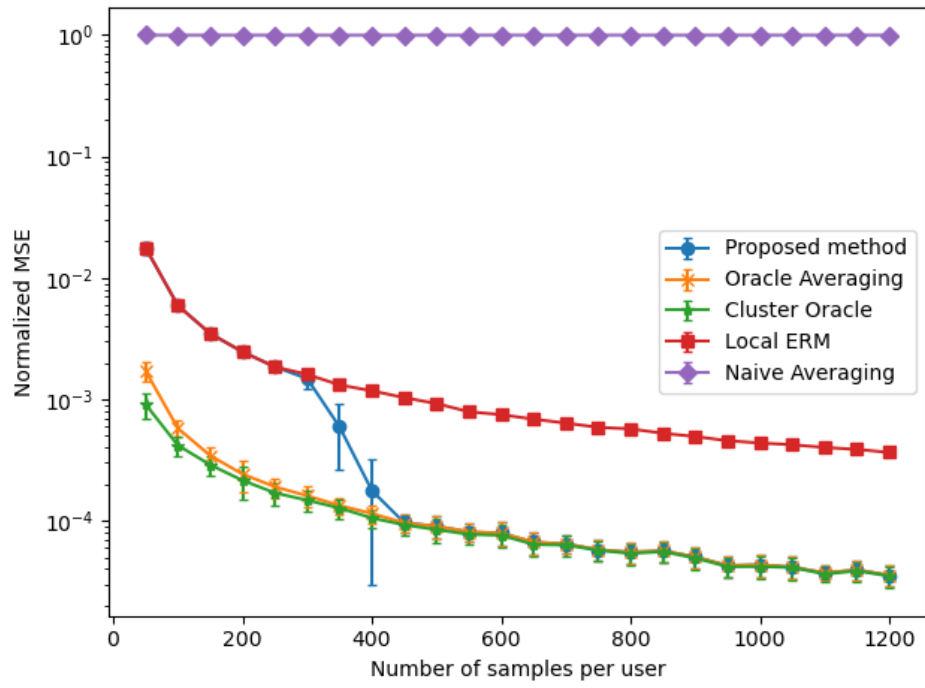


Figure 1: Performance of different methods for linear regression, versus the number of samples available per user. We can see that our method matches the order-optimal MSE rates for a sufficiently large sample size.

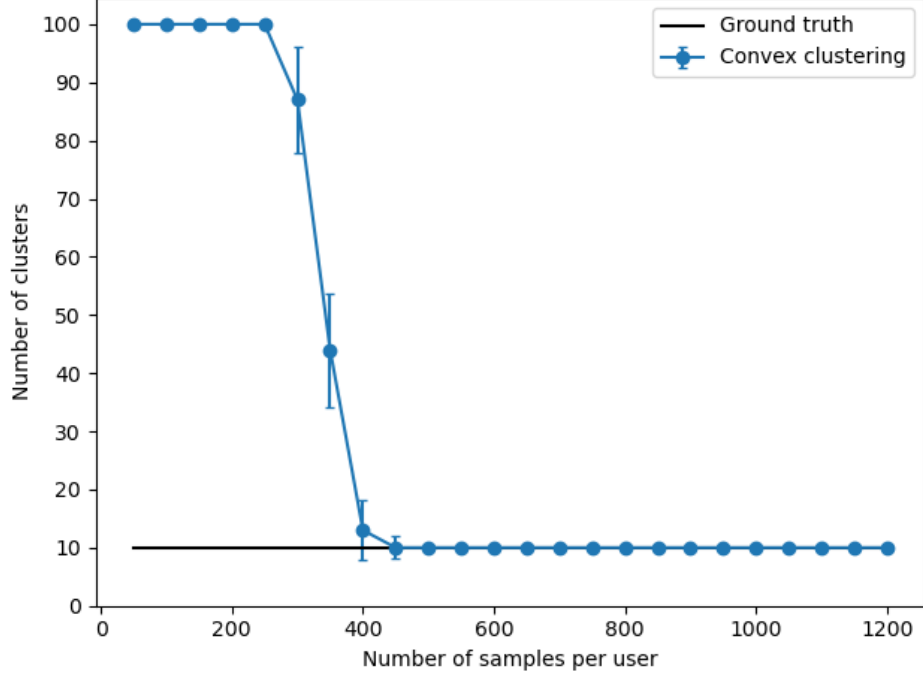


Figure 2: Number of clusters produced by convex clustering for linear regression, versus the number of samples available per user. We can see that convex clustering is able to recover the exact clustering for a sufficiently large sample size available to each user.

Figure 2 presents the performance of convex clustering. On  $y$ -axis, we plot the number of clusters produced by the convex clustering algorithm. On  $x$ -axis, we again plot the number of samples  $n$ . Figure 2 is consistent with the results from Figure 1, as it shows that, for small  $n$  (less than 300), convex clustering clusters each user separately, which, due to the low sample regime and our sub-optimal choice of  $\lambda$ , is to be expected. On the other hand, there is a sharp phase transition in the number of clusters for  $n$  between 300 and 400, after which convex clustering consistently produces  $K' = 10$  clusters. Moreover, we can see that the clustering produced by the convex clustering method is correct, as our method matches the performance of both oracle methods that know the true clustering.

## 6 Conclusion

We proposed a one-shot approach for CFL, based on a simple inference and averaging scheme. The proposed approach is communication efficient, as it requires a single round of communication. Moreover, our theoretical analysis showed that the method provides order-optimal MSE rates, in terms of the sample size. Compared to the state-of-the-art algorithms that require multiple rounds of communication, our method improve the existing results by a factor that is logarithmic in the total number of samples in the system, our metod provides significant communication reduction. Remarkably, unlike other methods that require knowledge of  $K$ , e.g., [29], [28],[30], our method does not require any knowledge of the underlying number of clusters  $K$ . Numerical experiments corroborate our findings.

## References

- [1] B. McMahan, E. Moore, D. Ramage, S. Hampson, and B. A. y. Arcas, “Communication-Efficient Learning of Deep Networks from Decentralized Data,” in *Proceedings of the 20th International Conference on Artificial Intelligence and Statistics*, ser. Proceedings of Machine Learning Research, A. Singh and J. Zhu, Eds., vol. 54. PMLR, 20–22 Apr 2017, pp. 1273–1282. [Online]. Available: <https://proceedings.mlr.press/v54/mcmahan17a.html>
- [2] J. Konečný, H. B. McMahan, F. X. Yu, P. Richtárik, A. T. Suresh, and D. Bacon, “Federated learning: Strategies for improving communication efficiency,” *arXiv preprint arXiv:1610.05492*, 2016.
- [3] D. Alistarh, D. Grubic, J. Li, R. Tomioka, and M. Vojnovic, “Qsgd: Communication-efficient sgd via gradient quantization and encoding,” in *Advances in Neural Information Processing Systems*, I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, Eds., vol. 30. Curran Associates, Inc., 2017. [Online]. Available: <https://proceedings.neurips.cc/paper/2017/file/6c340f25839e6acdc73414517203f5f0-Paper.pdf>
- [4] A. Reisizadeh, A. Mokhtari, H. Hassani, A. Jadbabaie, and R. Pedarsani, “Fedpaq: A communication-efficient federated learning method with periodic averaging and quantization,” in *Proceedings of the Twenty Third International Conference on Artificial Intelligence and Statistics*, ser. Proceedings of Machine Learning Research, S. Chiappa



- and R. Calandra, Eds., vol. 108. PMLR, 26–28 Aug 2020, pp. 2021–2031. [Online]. Available: <https://proceedings.mlr.press/v108/reisizadeh20a.html>
- [5] A. Koloskova, S. Stich, and M. Jaggi, “Decentralized stochastic optimization and gossip algorithms with compressed communication,” in *Proceedings of the 36th International Conference on Machine Learning*, ser. Proceedings of Machine Learning Research, K. Chaudhuri and R. Salakhutdinov, Eds., vol. 97. PMLR, 09–15 Jun 2019, pp. 3478–3487. [Online]. Available: <https://proceedings.mlr.press/v97/koloskova19a.html>
  - [6] S. U. Stich, J.-B. Cordonnier, and M. Jaggi, “Sparsified sgd with memory,” in *Advances in Neural Information Processing Systems*, S. Bengio, H. Wallach, H. Larochelle, K. Grauman, N. Cesa-Bianchi, and R. Garnett, Eds., vol. 31. Curran Associates, Inc., 2018. [Online]. Available: <https://proceedings.neurips.cc/paper/2018/file/b440509a0106086a67bc2ea9df0a1dab-Paper.pdf>
  - [7] J. Wangni, J. Wang, J. Liu, and T. Zhang, “Gradient sparsification for communication-efficient distributed optimization,” in *Advances in Neural Information Processing Systems*, S. Bengio, H. Wallach, H. Larochelle, K. Grauman, N. Cesa-Bianchi, and R. Garnett, Eds., vol. 31. Curran Associates, Inc., 2018. [Online]. Available: <https://proceedings.neurips.cc/paper/2018/file/3328bdf9a4b9504b9398284244fe97c2-Paper.pdf>
  - [8] Y. J. Cho, J. Wang, and G. Joshi, “Client selection in federated learning: Convergence analysis and power-of-choice selection strategies,” *arXiv preprint arXiv:2010.01243*, 2020.
  - [9] W. Chen, S. Horvath, and P. Richtarik, “Optimal client sampling for federated learning,” *arXiv preprint arXiv:2010.13723*, 2020.
  - [10] M. Ribero and H. Vikalo, “Communication-efficient federated learning via optimal client sampling,” *arXiv preprint arXiv:2007.15197*, 2020.
  - [11] S. U. Stich, “Local sgd converges fast and communicates little,” in *ICLR 2019-International Conference on Learning Representations*, no. CONF, 2019.
  - [12] A. Khaled, K. Mishchenko, and P. Richtarik, “Tighter theory for local sgd on identical and heterogeneous data,” in *Proceedings of the*

- Twenty Third International Conference on Artificial Intelligence and Statistics*, ser. Proceedings of Machine Learning Research, S. Chiappa and R. Calandra, Eds., vol. 108. PMLR, 26–28 Aug 2020, pp. 4519–4529. [Online]. Available: <https://proceedings.mlr.press/v108/bayoumi20a.html>
- [13] K. Mishchenko, G. Malinovsky, S. Stich, and P. Richtarik, “ProxSkip: Yes! Local gradient steps provably lead to communication acceleration! Finally!” in *Proceedings of the 39th International Conference on Machine Learning*, ser. Proceedings of Machine Learning Research, K. Chaudhuri, S. Jegelka, L. Song, C. Szepesvari, G. Niu, and S. Sabato, Eds., vol. 162. PMLR, 17–23 Jul 2022, pp. 15 750–15 769. [Online]. Available: <https://proceedings.mlr.press/v162/mishchenko22b.html>
  - [14] Y. Zhang, M. J. Wainwright, and J. C. Duchi, “Communication-efficient algorithms for statistical optimization,” in *Advances in Neural Information Processing Systems*, F. Pereira, C. J. C. Burges, L. Bottou, and K. Q. Weinberger, Eds., vol. 25. Curran Associates, Inc., 2012. [Online]. Available: <https://proceedings.neurips.cc/paper/2012/file/e7f8a7fb0b77bcb3b283af5be021448f-Paper.pdf>
  - [15] N. Guha, A. Talwalkar, and V. Smith, “One-shot federated learning,” *arXiv preprint arXiv:1902.11175*, 2019.
  - [16] Y. Zhou, G. Pu, X. Ma, X. Li, and D. Wu, “Distilled one-shot federated learning,” *arXiv preprint arXiv:2009.07999*, 2020.
  - [17] S. Salehkaleybar, A. Sharifnassab, and S. J. Golestani, “One-shot federated learning: Theoretical limits and algorithms to achieve them,” *Journal of Machine Learning Research*, vol. 22, no. 189, pp. 1–47, 2021. [Online]. Available: <http://jmlr.org/papers/v22/19-1048.html>
  - [18] D. K. Dennis, T. Li, and V. Smith, “Heterogeneity for the win: One-shot federated clustering,” in *Proceedings of the 38th International Conference on Machine Learning*, ser. Proceedings of Machine Learning Research, M. Meila and T. Zhang, Eds., vol. 139. PMLR, 18–24 Jul 2021, pp. 2611–2620. [Online]. Available: <https://proceedings.mlr.press/v139/dennis21a.html>
  - [19] J. Wang, Q. Liu, H. Liang, G. Joshi, and H. V. Poor, “Tackling the objective inconsistency problem in heterogeneous federated optimization,” *Advances in neural information processing systems*, vol. 33, pp. 7611–7623, 2020.

- [20] T. Yu, E. Bagdasaryan, and V. Shmatikov, “Salvaging federated learning by local adaptation,” *arXiv preprint arXiv:2002.04758*, 2020.
- [21] V. Smith, C.-K. Chiang, M. Sanjabi, and A. S. Talwalkar, “Federated multi-task learning,” in *Advances in Neural Information Processing Systems*, I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, Eds., vol. 30. Curran Associates, Inc., 2017. [Online]. Available: <https://proceedings.neurips.cc/paper/2017/file/6211080fa89981f66b1a0c9d55c61d0f-Paper.pdf>
- [22] F. Hanzely and P. Richtárik, “Federated learning of a mixture of global and local models,” *arXiv preprint arXiv:2002.05516*, 2020.
- [23] F. Hanzely, S. Hanzely, S. Horváth, and P. Richtárik, “Lower bounds and optimal algorithms for personalized federated learning,” *Advances in Neural Information Processing Systems*, vol. 33, pp. 2304–2315, 2020.
- [24] A. Fallah, A. Mokhtari, and A. Ozdaglar, “Personalized federated learning with theoretical guarantees: A model-agnostic meta-learning approach,” in *Advances in Neural Information Processing Systems*, H. Larochelle, M. Ranzato, R. Hadsell, M. F. Balcan, and H. Lin, Eds., vol. 33. Curran Associates, Inc., 2020, pp. 3557–3568. [Online]. Available: <https://proceedings.neurips.cc/paper/2020/file/24389bfe4fe2eba8bf9aa9203a44cdad-Paper.pdf>
- [25] M. Khodak, M.-F. F. Balcan, and A. S. Talwalkar, “Adaptive gradient-based meta-learning methods,” in *Advances in Neural Information Processing Systems*, H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, and R. Garnett, Eds., vol. 32. Curran Associates, Inc., 2019. [Online]. Available: <https://proceedings.neurips.cc/paper/2019/file/f4aa0dd960521e045ae2f20621fb4ee9-Paper.pdf>
- [26] C. T. Dinh, N. Tran, and J. Nguyen, “Personalized federated learning with moreau envelopes,” *Advances in Neural Information Processing Systems*, vol. 33, pp. 21 394–21 405, 2020.
- [27] T. Li, S. Hu, A. Beirami, and V. Smith, “Ditto: Fair and robust federated learning through personalization,” in *International Conference on Machine Learning*. PMLR, 2021, pp. 6357–6368.
- [28] Y. Mansour, M. Mohri, J. Ro, and A. T. Suresh, “Three approaches for personalization with applications to federated learning,” *arXiv preprint arXiv:2002.10619*, 2020.

- [29] A. Ghosh, J. Chung, D. Yin, and K. Ramchandran, “An efficient framework for clustered federated learning,” *IEEE Transactions on Information Theory*, pp. 1–1, 2022.
- [30] A. Ghosh, J. Hong, D. Yin, and K. Ramchandran, “Robust federated learning in a heterogeneous environment,” *arXiv preprint arXiv:1906.06629*, 2019.
- [31] F. Sattler, K.-R. Müller, and W. Samek, “Clustered federated learning: Model-agnostic distributed multitask optimization under privacy constraints,” *IEEE Transactions on Neural Networks and Learning Systems*, vol. 32, no. 8, pp. 3710–3722, 2021.
- [32] A. Armacki, D. Bajovic, D. Jakovetic, and S. Kar, “Personalized federated learning via convex clustering,” *arXiv preprint arXiv:2202.00718*, 2022.
- [33] K. Sridharan, S. Shalev-Shwartz, and N. Srebro, “Fast rates for regularized objectives,” in *Advances in Neural Information Processing Systems*, D. Koller, D. Schuurmans, Y. Bengio, and L. Bottou, Eds., vol. 21. Curran Associates, Inc., 2008. [Online]. Available: <https://proceedings.neurips.cc/paper/2008/file/abd815286ba1007abfbb8415b83ae2cf-Paper.pdf>
- [34] L. Zhang, T. Yang, and R. Jin, “Empirical risk minimization for stochastic convex optimization:  $O(1/n)$ - and  $O(1/n^2)$ -type of risk bounds,” in *Proceedings of the 2017 Conference on Learning Theory*, ser. Proceedings of Machine Learning Research, S. Kale and O. Shamir, Eds., vol. 65. PMLR, 07–10 Jul 2017, pp. 1954–1979. [Online]. Available: <https://proceedings.mlr.press/v65/zhang17a.html>
- [35] A. Panahi, D. Dubhashi, F. D. Johansson, and C. Bhattacharyya, “Clustering by sum of norms: Stochastic incremental algorithm, convergence and cluster recovery,” in *Proceedings of the 34th International Conference on Machine Learning*, ser. Proceedings of Machine Learning Research, D. Precup and Y. W. Teh, Eds., vol. 70. PMLR, 06–11 Aug 2017, pp. 2769–2777. [Online]. Available: <https://proceedings.mlr.press/v70/panahi17a.html>
- [36] T. D. Hocking, A. Joulin, F. Bach, and J.-P. Vert, “Clusterpath: An algorithm for clustering using convex fusion penalties,” in *Proceedings of the 28th International Conference on International Conference on*

- Machine Learning*, ser. ICML'11. Madison, WI, USA: Omnipress, 2011, p. 745–752.
- [37] D. Sun, K.-C. Toh, and Y. Yuan, “Convex clustering: Model, theoretical guarantee and efficient algorithm.” *J. Mach. Learn. Res.*, vol. 22, pp. 9–1, 2021.
  - [38] S. Lloyd, “Least squares quantization in pcm,” *IEEE Transactions on Information Theory*, vol. 28, no. 2, pp. 129–137, 1982.
  - [39] A. Banerjee, S. Merugu, I. S. Dhillon, J. Ghosh, and J. Lafferty, “Clustering with bregman divergences.” *Journal of machine learning research*, vol. 6, no. 10, 2005.
  - [40] A. Armacki, D. Bajovic, D. Jakovetic, and S. Kar, “Gradient based clustering,” in *Proceedings of the 39th International Conference on Machine Learning*, ser. Proceedings of Machine Learning Research, K. Chaudhuri, S. Jegelka, L. Song, C. Szepesvari, G. Niu, and S. Sabato, Eds., vol. 162. PMLR, 17–23 Jul 2022, pp. 929–947. [Online]. Available: <https://proceedings.mlr.press/v162/armacki22a.html>
  - [41] W. Ben-Ameur, P. Bianchi, and J. Jakubowicz, “Robust distributed consensus using total variation,” *IEEE Transactions on Automatic Control*, vol. 61, no. 6, pp. 1550–1564, 2016.
  - [42] E. C. Chi and K. Lange, “Splitting methods for convex clustering,” *Journal of Computational and Graphical Statistics*, vol. 24, no. 4, pp. 994–1013, 2015, pMID: 27087770. [Online]. Available: <https://doi.org/10.1080/10618600.2014.948181>
  - [43] H. Robbins and S. Monro, “A Stochastic Approximation Method,” *The Annals of Mathematical Statistics*, vol. 22, no. 3, pp. 400 – 407, 1951. [Online]. Available: <https://doi.org/10.1214/aoms/1177729586>
  - [44] A. Rakhlin, O. Shamir, and K. Sridharan, “Making gradient descent optimal for strongly convex stochastic optimization,” in *Proceedings of the 29th International Conference on Machine Learning*, 2012, pp. 1571–1578.
  - [45] A. Defazio, F. Bach, and S. Lacoste-Julien, “Saga: A fast incremental gradient method with support for non-strongly convex composite objectives,” in *Advances in Neural Information Processing Systems*, Z. Ghahramani, M. Welling, C. Cortes, N. Lawrence,

and K. Weinberger, Eds., vol. 27. Curran Associates, Inc., 2014. [Online]. Available: <https://proceedings.neurips.cc/paper/2014/file/ede7e2b6d13a41ddf9f4bdef84fdc737-Paper.pdf>

- [46] R. Johnson and T. Zhang, “Accelerating stochastic gradient descent using predictive variance reduction,” in *Advances in Neural Information Processing Systems*, C. Burges, L. Bottou, M. Welling, Z. Ghahramani, and K. Weinberger, Eds., vol. 26. Curran Associates, Inc., 2013. [Online]. Available: <https://proceedings.neurips.cc/paper/2013/file/ac1dd209cbcc5e5d1c6e28598e8cbbe8-Paper.pdf>
- [47] S. Diamond and S. Boyd, “CVXPY: A Python-embedded modeling language for convex optimization,” *Journal of Machine Learning Research*, vol. 17, no. 83, pp. 1–5, 2016.

## A Appendix

In this Appendix we show conditions under which it is suitable for two users with different distributions to form a cluster by merging their respective data. As in the main body, we use  $\theta_k^*$  and  $\hat{\theta}_i$  to denote the population optimal of cluster  $k \in [K]$  and local ERM of user  $i \in [m]$ , respectively, i.e.,  $\theta_k^* = \arg \min_{\theta \in \Theta} F_k(\theta)$  and  $\hat{\theta}_i = \arg \min_{\theta \in \Theta} f_i(\theta)$ . We then have the following result.

**Lemma 7.** *Let Assumption 3 hold and assume  $\ell$  is strongly convex. Let all users sample data from a unique distribution, i.e., each user  $i$  samples data following  $\mathcal{D}_i$ ,  $i \in [m]$ . Denote by  $\mathcal{D}_k$  the mixture of distributions  $\mathcal{D}_i$  and  $\mathcal{D}_j$ , i.e., the distribution such that  $F_k(\theta) = p_i F_i(\theta) + p_j F_j(\theta)$ , where  $0 < p_i, p_j < 1$ , such that  $p_i + p_j = 1$ . If the distributions  $\mathcal{D}_i$  and  $\mathcal{D}_j$  are such that*

$$\|\theta_i^* - \theta_j^*\|^2 < \epsilon,$$

*then, with high probability*

$$\|\hat{\theta}_k - \theta_m^*\|^2 = \mathcal{O}\left(\frac{1}{n_i + n_j} + \epsilon\right),$$

*with  $m = i, j$ , where  $\hat{\theta}_k = \arg \min_{\theta \in \Theta} p_i f_i(\theta) + p_j f_j(\theta)$ .*

Some remarks are now in order.

**Remark 24.** Lemma 7 tells us that, as long as  $\frac{1}{n_i+n_j} + \epsilon < \min \left\{ \frac{1}{n_i}, \frac{1}{n_j} \right\}$ , i.e.,  $\epsilon < \frac{\min\{n_i, n_j\}}{\max\{n_i, n_j\}(n_i+n_j)}$ , we have that the model trained on the joint datasets of users  $i$  and  $j$  is beneficial to both users. For example, when  $n_i = n, \forall i \in [m]$ , the condition on  $\epsilon$  evaluates to  $\epsilon < \frac{1}{2n}$ .

**Remark 25.** If  $\frac{1}{n_i+n_j} + \epsilon < \min \left\{ \frac{1}{n_i}, \frac{1}{n_j} \right\}$ , Lemma 7 tells us that it is beneficial to treat the users  $i$  and  $j$  as belonging to the same cluster. Therefore, averaging the local ERM's trained by users  $i$  and  $j$  leads to mutual benefits, even though the two users come from different, but mutually close distributions (as measured by the distance of the population optima). Therefore, it is beneficial to treat users  $i, j$  as belonging to the same cluster, justifying the assumption that  $1 < K < m$ .

*Proof of Lemma 7.* Applying the results of [33], we have that, with high probability

$$\|\hat{\theta}_i - \theta_i^*\|^2 = \mathcal{O} \left( \frac{1}{n_i} \right).$$

Denote by  $\theta_k^*$  the population optima of the mixture distribution  $\mathcal{D}_k$ . We then have

$$\begin{aligned} \|\hat{\theta}_k - \theta_i^*\|^2 &\leq 2\|\hat{\theta}_k - \theta_k^*\|^2 + 2\|\theta_k^* - \theta_i^*\|^2 \\ &\leq \mathcal{O} \left( \frac{1}{n_i + n_j} \right) + 2\|\theta_k^* - \theta_i^*\|^2, \end{aligned} \quad (33)$$

where the second inequality again follows from [33]. Using strong convexity of  $F$ 's, we have that

$$\begin{aligned} \frac{\mu}{2} \|\theta_k^* - \theta_i^*\|^2 &\leq F_k(\theta_i^*) - F_k(\theta_k^*) \\ &= p_i F_i(\theta_i^*) + p_j F_j(\theta_i^*) - p_i F_i(\theta_k^*) - p_j F_j(\theta_k^*) \\ &\leq p_j (F_j(\theta_i^*) - F_j(\theta_k^*)) \\ &= p_j (F_j(\theta_i^*) - F_j(\theta_j^*) + F_j(\theta_j^*) - F_j(\theta_k^*)) \\ &\leq p_j (F_j(\theta_i^*) - F_j(\theta_j^*)), \end{aligned}$$

where we used the fact that  $\theta_m^* = \arg \min_{\theta \in \Theta} F_m(\theta)$ ,  $m \in \{i, j\}$  in the second and third inequalities, respectively, with  $\mu$  the strong convexity parameter of  $\ell$ . Finally, using  $L$ -Lipschitz continuous gradients of  $F$ 's, we get that

$$F_j(\theta_i^*) - F_j(\theta_j^*) \leq \frac{L}{2} \|\theta_i^* - \theta_j^*\|^2 = \mathcal{O}(\epsilon). \quad (34)$$

Combining (33) and (34), we get, with high probability

$$\|\widehat{\theta}_k - \theta_i^\star\|^2 = \mathcal{O}\left(\frac{1}{n_i + n_j} + \epsilon\right).$$

Analogous results can be obtained for user  $j$ , hence the claim follows.  $\square$