Personalized Federated Learning with Contextualized Generalization

Xueyang Tang\textsuperscript{1}, Song Guo\textsuperscript{1,2,*} and Jingcai Guo\textsuperscript{1,2,*}

\textsuperscript{1}Department of Computing, The Hong Kong Polytechnic University, Hong Kong, China
\textsuperscript{2}The Hong Kong Polytechnic University Shenzhen Research Institute, Shenzhen, China
csxxtang@comp.polyu.edu.hk, song.guo@polyu.edu.hk, jingcai.guo@gmail.com

Abstract
The prevalent personalized federated learning (PFL) usually pursues a trade-off between personalization and generalization by maintaining a shared global model to guide the training process of local models. However, the sole global model may easily transfer deviated context knowledge to some local models when multiple latent contexts exist across the local datasets. In this paper, we propose a novel concept called contextualized generalization (CG) to provide each client with fine-grained context knowledge that can better fit the local data distributions and facilitate faster model convergence, based on which we properly design a framework of PFL, dubbed CGPFL. We conduct detailed theoretical analysis, in which the convergence guarantee is presented and \(\mathcal{O}(\sqrt{K})\) speedup over most existing methods is granted. To quantitatively study the generalization-personalization trade-off, we introduce the ‘generalization error’ measure and prove that the proposed CGPFL can achieve a better trade-off than existing solutions. Moreover, our theoretical analysis further inspires a heuristic algorithm to find a near-optimal trade-off in CGPFL. Experimental results on multiple real-world datasets show that our approach surpasses the state-of-the-art methods on test accuracy by a significant margin.

1 Introduction
Recently, personalized federated learning (PFL) has emerged as an alternative to conventional federated learning (FL) to cope with the statistical heterogeneity of local datasets (a.k.a., Non-I.I.D. data). Different from conventional FL that focuses on training a shared global model to explore the global optimality of the whole system, i.e., minimizing the averaged loss of clients, the PFL aims at developing a personalized model (distinct from the individually trained local model which usually fail to work due to the insufficient local data and the limited diversity of local dataset) for each client to properly cover diverse data distributions. To develop the personalized model, each user needs to incorporate some context information into the local data, since the insufficient local data cannot present the complete context which the personalized model will be applied to [Kairouz \textit{et al.}, 2019]. However, the context is generally latent and can be hardly featurized in practice, especially when the exchange of raw data is forbidden. In the existing PFLs, the latent context knowledge can be considered to be transferred to the local users via the global model update. During the PFL training, the personalization usually requires personalized models to fit local data distributions as well as possible, while the generalization needs to exploit the common context knowledge among clients by collaborative training. Thus, the PFL is indeed pursuing a trade-off between them to achieve better model accuracy than the traditional FL. More specifically, the server-side model is trained by aggregating local model updates from each client and hence can obtain the common context knowledge covering diverse data distributions. Such knowledge can then be offloaded to each client and contributes to the generalization of personalized models.

Despite the recent PFL approaches have reported better performance against conventional FL methods, they may still be constrained in personalization by using sole global model as the guidance during the training process. Concretely, our intuition is that: If there exists multiple latent contexts across local data distributions, then contextualized generalization can provide fine-grained context knowledge and further facilitate the personalized models toward better recognition accuracy and faster model convergence. We thus argue one potential bottleneck of current PFL methods is the loss of generalization diversity with only one global model. Worse still, the global model may also easily degrade the overall performance of PFL models due to negative knowledge transfers between the disjoint contexts.

In this paper, we design a novel PFL training framework, dubbed CGPFL, by involving the proposed concept, i.e., contextualized generalization (CG), to handle the challenge of the context-level heterogeneity. More specifically, we suppose the participating clients can be covered by several latent contexts based on their statistical characteristics and each latent context can be corresponded to a generalized model maintained in the server. The personalized models are dynamically associated with the most pertinent generalized model and guided by it with fine-grained contextualized gen-
eralization in an iterative manner. We formulate the process as a bi-level optimization problem considering both the global models with contextualized generalization maintained in the server and the personalized models trained locally in clients.

The main contributions of this work are summarized as follows:

- To the best of our knowledge, we are the first to propose the concept of contextualized generalization (CG) to provide fine-grained generalization and seek a better trade-off between personalization and generalization in PFL, and further formulate the training as a bi-level optimization problem that can be solved effectively by our designed CGPFL algorithm.

- We conduct detailed theoretical analysis to provide the convergence guarantee and prove that CGPFL can obtain a $O(\sqrt{K})$ times acceleration over the convergence rate of most existing algorithms for non-convex and smooth case. We further derive the generalization error bound of CGPFL and demonstrate that the proposed contextualized generalization can constantly help reach a better trade-off between personalization and generalization in terms of generalization error against the state-of-the-arts works.

- We provide a heuristic improvement of CGPFL, dubbed CGPFL-Heur, by minimizing the generalization bound in the theoretical analysis, to find a near-optimal trade-off between personalization and generalization. CGPFL-Heur can achieve a near-optimal accuracy with negligible additional computation in the server, while retaining the same convergence rate as that of CGPFL.

- Experimental results on multiple real-world datasets demonstrate that our proposed methods, i.e., CGPFL and CGPFL-Heur, can achieve higher model accuracy than the state-of-the-art PFL methods in both convex and non-convex cases.

### 2 Related Work

Considering that one shared global model can hardly fit the heterogeneous data distributions, some recent FL works [Ghosh et al., 2020; Sattler et al., 2020; Briggs et al., 2020; Mansour et al., 2020] try to cluster the participating clients into multiple groups and develop corresponding number of shared global models by aggregating the local updates. After the training process, the obtained global models are offloaded to the corresponding clients for inference. Since these methods only reduce the FL training into several sub-groups, of which each global model is still shared by their in-group clients, the personalization is scarce and the offloaded models can still hardly cover the heterogeneous data distributions across the in-group clients. Specifically, *IFCA* [Ghosh et al., 2020] requires each client to calculate the losses on all global models to estimate its cluster identity during each iteration, and result in significantly higher computation cost. *CFL* [Sattler et al., 2020] demonstrates that the conventional FL even cannot converge in some Non-I.I.D. settings and provides intriguing perspective for clustered FL with bi-partitioning clustering. However, it can only work for some special Non-I.I.D. case described as ‘same feature & different labels’ [Hsieh et al., 2020]. *FL+HC* [Briggs et al., 2020] divides the clients clustering and the model training processes separately, and only conducts the clustering once at a manually defined step, while the training remains the same as conventional FL. Last, three effective PFL approaches are proposed in [Mansour et al., 2020], of which the user clustering method is very similar to *IFCA* [Ghosh et al., 2020].

Most recently, the PFL approaches have attracted increasing attention [Kairouz et al., 2019]. Among them, a branch of works [Hanzely and Richtárik, 2020; Hanzely et al., 2020; Deng et al., 2020] propose to mix the global model on the server with local models to acquire the personalized models. More concretely, Hanzely et al. [Hanzely et al., 2020; Hanzely and Richtárik, 2020] formulate the mixture problem as a combined optimization of the local and global models, while APFL [Deng et al., 2020] straightforwardly mixes them with an adaptive weight. *KT-pFL* [Zhang et al., 2021] exploits the knowledge distillation (KD) to transfer the generalization information to local models and allows the training of heterogeneous models in FL setting. Differently, *FedPer* [Ariavazhagan et al., 2019] splits the personalized models into two separate parts, of which the base layers are shared by all the clients and trained on the server, and the personalized layers are trained to adapt to individual data and maintain the privacy properties on local devices. *MOCHA* [Smith et al., 2017] considers the model training on the clients as relevant tasks and formulate this problem as a distributed multi-task learning objective. Fallah et al. [Fallah et al., 2020] make use of the model agnostic meta learning (*MAML*) to implement the PFL, of which the obtained meta-model contains the generalization information and can be utilized as a good initialization point of training.

### 3 Problem Formulation

We start by formalizing the FL task and then introduce our proposed method. Given $N$ clients and the their Non-I.I.D. datasets $D_1, \ldots, D_i, \ldots, D_N$ that subject to the underlying distributions as $D_1, \ldots, D_i, \ldots, D_N (D_i \in \mathbb{R}^{dxn_i} \text{ and } i \in [N])$. Every client $i$ has $m_i$ instances $z_{i,j} = (x_{i,j}, y_{i,j}), j \in [m_i]$, where $x$ is the data features and $y$ denotes the label. Hence, the objective function of the conventional FL can be described as [Li et al., 2021]:

$$
\min_{\omega \in \mathbb{R}^d} \{G(\omega) := G(f_1(\omega; \bar{D}_1), ..., f_N(\omega; \bar{D}_N))\},
$$

where $\omega$ is the global model and $f_i : \mathbb{R}^d \rightarrow \mathbb{R}, i \in [N]$ denotes the expected loss function over the data distribution of client $i$: $f_i(\omega; \bar{D}_i) = \mathbb{E}_{z \sim \bar{D}_i} [f_i(\omega; z_{i,j})]$. The function $G(\cdot)$ denotes the aggregation method to obtain the global model $\omega$. For example, *FedAvg* [McMahan et al., 2017] applies $G(\omega) = \sum_{i=1}^{N} \frac{m_i}{m} f_i(\omega)$ to do the aggregation, where $m$ is the total number of instances on local devices.

To handle the challenge of rich statistical diversities in PFL, especially in the cases where the local datasets belong to several latent contexts, our CGPFL propose to maintain $K$ context-level generalized models in the server to guide the training of personalized models on the clients. During training, the local training process based on its local dataset can...
i.e., the average estimation error. In this paper, we use $L$ to derive the context-level generalized models by minimizing $\Omega^*$, which can be used to evaluate the estimation error, and we can further investigate the locally obtained $\omega^*_i$, $\lambda$ is a hyper-parameter and $C_k$ denotes the corresponding context that client $i$ belongs to. Considering the latent contexts are represented in disjoint subspaces respectively, the function $G(\cdot)$ can be decomposed as $G(\omega_1, ..., \omega_K; C_K) = \frac{1}{K} \sum_{k=1}^{K} G_k(\omega_k; C_k)$.

In general, there exists two alternative strategies to generate the context-level generalized models. The intuitive one is to solve the inner-level objective $\min_{\Omega \in \mathbb{R}^{d \times K}} G(\omega_1, ..., \omega_K)$ based on local datasets, which is similar to IFCA [Ghosh et al., 2020]. However, the computation overhead is high in the local devices while their available computation resources are usually limited. Comparing the local objective that trains a generalized model $\omega^*_i$ based on local dataset, i.e., $\omega^*_i = \arg \min_{\omega} f_i(\omega; \hat{D}_i)$, with that of the personalized model, i.e., $\theta^*_i = \arg \min_{\theta} f_i(\theta; \hat{D}_i) + \lambda \rho(\theta, \omega^*_i)$, we notice that the locally obtained $\theta^*_i$ can be regarded as the distributed estimation of $\omega^*_i$. In this way, the regularizer $\rho(\theta^*_i, \omega^*_i)$ can be used to evaluate the estimation error, and we can further derive the context-level generalized models by minimizing the average estimation error. In this paper, we use $L2$-norm, i.e., $\rho(\theta^*_i, \omega^*_i) = \frac{1}{2} \|\theta^*_i - \omega^*_i\|^2$ as the regularizer, which is also adopted in various prevalent PFL methods [Hanzely and Richtárik, 2020; Hanzely et al., 2020; T Dinh et al., 2020; Li et al., 2021], and has been empirically demonstrated to be superior over other regularizers, e.g., the symmetrized KL divergence [Li et al., 2021]. Hence, we formulate our overall objective as:

$$\min_{\Omega \in \mathbb{R}^{d \times N}} \frac{1}{N} \sum_{i=1}^{N} \left\{ F_i(\theta_{i}) := f_i(\theta_{i}) + \lambda \frac{1}{2} \|\theta_{i} - \omega^*_i\|^2 \right\}, i \in C^*_K,$$

s.t. $\Omega^*, C^*_K = \arg \min_{\Omega \in \mathbb{R}^{d \times K}, C_K \in \mathbf{K}} \sum_{k=1}^{K} q_k \sum_{j \in C_k} p_{k,j} \|\theta_j - \omega^*_j\|^2$, (2)

We adopt $p_{k,j} = \frac{1}{|C_k|}$ and $q_k = \frac{|C_k|}{N}$ in this paper, where $C_k \in \mathbf{K}$ denotes the latent context $k$, and $|C_k|$ is the number of clients that belong to the context $k$. Intriguingly, the inner-level objective is exactly the classic objective of $k$-means clustering [Lloyd, 1982]. We notice that when $K = 1$, the above objective is equivalent to the overall objective in [T Dinh et al., 2020], which means that the objective in [T Dinh et al., 2020] can be regarded as a special case ($K = 1$) of ours.

4 Design of CGPFL

In this section, we introduce our proposed CGPFL in detail. The key idea is to dynamically relate the clients to $K$ latent contexts based on their uploaded local model updates, and then develop a generalized model for each context by aggregating the updates from each user group. These generalized models are utilized to guide the training directions of personalized models and transfer contextualized generalization to them. Both the personalized models and the generalized models are trained in parallel, so we can denote the model parameters in matrix form. The generalized models can be written as $\Omega_K := [\omega_1, ..., \omega_K] \in \mathbb{R}^{d \times K}$, and the corresponding local approximations are $\Omega_{i,R} := [\hat{\omega}_{1,R}, ..., \hat{\omega}_{i,R}, ..., \hat{\omega}_{N,R}]$, where $R$ is the number of local iterations and $\hat{\omega}_{i,R}, \omega_k \in \mathbb{R}^d, \forall i \in [N], k \in [K]$. In this paper, we use capital characters to represent matrices unless stated otherwise.

4.1 CGPFL: Algorithm

We design an effective alternating optimization framework to minimize the overall objective in (2). Specifically, the upper-level problem can be decomposed into $N$ separate subproblems with fixed generalized models and to be solved on local devices in parallel. Next, we can further settle the inner-level problem to derive the generalized models with fixed personalized models. Since the solution to the subproblems of the upper-level objective has been well-explored in recent PFL methods [T Dinh et al., 2020; Li et al., 2021; Hanzely et al., 2020], we hereby mainly focus on the inner-level problem. We alternately update the context-level generalized models $\Omega_K$ and the context indicator $C_K$ to obtain the optimal generalized models. We view the personalized models, i.e., $\Theta_i = [\theta_1, ..., \theta_N]$, as private data, and distributionally update the context-level generalized models $\Omega_K$ on clients with fixed context indicator $C_K$. During each server round, the server conducts $k$-means clustering on uploaded local parameters $\Omega_{i,R}$ to cluster the clients into $K$ latent contexts, and the clustering results $C_K$ are re-arranged to the matrix form as $P^t \in \mathbb{R}^{N \times K}$. For example, if client $i$, $i \in [N]$ is clustered into the context $C_j$, $j \in [K]$ (where $C_j, j \in [K]$ are sets, the union $\bigcup_{j \in [K]} C_j$ and intersection $\bigcap_{j \in [K]} C_j$ are the set $[N]$ and empty set, respectively), the element $(P^t)_{i,j}$ is defined as $\begin{cases} 1, & i,j, \text{ otherwise}. \end{cases}$ In this way, the elements of every column in $P^t$ amount to 1, i.e., $\sum_{i=1}^{N} (P^t)_{i,j} = 1, \forall i, j$.

When considering the relationship between the consecutive $P^t$, we can formulate the iterate as $P^{t+1} = P^t Q^t$, where $Q^t \in \mathbb{R}^{K \times K}$ is a square matrix. We can find that to maintain the above property of $P^t$ (s.t.), the matrix $Q^t$ must satisfies that:

$$\sum_{j=1}^{K} (Q^t)_{j,k} = 1, \forall k, t \quad \text{and} \quad \sum_{k=1}^{K} (Q^t)_{j,k} = 1, \forall j, t. \quad (3)$$

It is noticed that the clustering is based on the latent model parameters $\Omega(t+1)$ that depends on $\Omega(t)$, and the latest gradient updates given by clients. Hence, $P^{t+1}$ is determined by and only by $P^t$ and $Q^t$. Then we can consider this global iteration as a discrete-time Markov chain and $Q^t$ corresponds the transition probability matrix.
Algorithm 1 CGPFL: Personalized Federated Learning with Contextualized Generalization

Input: \( \Theta_0^i, \Omega_k^i, P^0, T, R, S, K, \lambda, \eta, \alpha, \beta, \delta \)

Output: \( \Theta_T^i \)

1: for \( t = 0 \) to \( T - 1 \) do
2: Server sends \( \Omega_k^i \) to clients according to \( P^t \).
3: for local device \( i = 1 \) to \( N \) in parallel do
4: Initialization: \( \Omega_k^i = \Omega_k^0 \).
5: Local update for the subproblem of \( G(\Theta_i, \Omega_k^i) \):
6: for \( r = 0 \) to \( R - 1 \) do
7: for \( s = 0 \) to \( S - 1 \) do
8: Update personalized model: \( \theta_i^{s+1} = \theta_i^s - \eta \nabla F_i(\theta_i^s) \).
9: end for
10: Local update: \( \omega_i^{t+1} = \omega_i^t - \beta \nabla \omega_i G_i(\theta_i^{t+1}), \omega_i^{t+1} \).
11: end for
12: end for
13: Clients send back \( \omega_i^{t+1} \) and server conducts clustering (e.g., \( k \)-means++) on models \( \omega_i^{t+1} \) to obtain \( P^{t+1} \).
14: Global aggregation: \( \Omega_k^{t+1} = \Omega_k^t - \alpha (\Omega_k^t - \Omega_k^{t,R}) P^{t+1} \).
15: end for
16: return The personalized models \( \Theta_T^i \).

During each local round, the clients need to first utilize local datasets to solve the regularized optimization problem, i.e., the upper-level objective in (2) with fixed \( \omega_i^{t+1} \) to obtain a \( \delta \)-approximate solution \( \tilde{\theta}_i(\omega_i^{t+1}) \). Then, each client is required to calculate the gradients \( \nabla_g_i G_i(\tilde{\theta}_i(\omega_i^{t+1})), \omega_i^{t+1} \) with fixed \( \tilde{\theta}_i(\omega_i^{t+1}) \) and update the model using \( \omega_i^{t+1} = \omega_i^t - \beta \nabla \omega_i G_i(\theta_i^{t+1}), \omega_i^{t+1} \), where \( \beta \) is the learning rate and \( \nabla \omega_i G_i(\theta_i^{t+1}), \omega_i^{t+1} = \frac{\partial}{\partial \theta} \nabla r(\theta_i(\omega_i^{t+1})), \omega_i^{t+1} \). To reduce the communication overhead, our CGPFL allows the clients to process several local iterations before uploading the latest model parameters to the server. The details of CGPFL is given in algorithm 1, from which we can summarize the parameters update process as:

\[
\Omega_k^{t+1} = \frac{\partial}{\partial \theta} \nabla \omega_i G_i(\theta_i^{t+1}), \omega_i^{t+1}, \omega_i^{t+1} \rightarrow \Omega_k^{t+1} = \lambda \Omega_k^t - \alpha (\Omega_k^t - \Omega_k^{t,R} P^{t+1}) \rightarrow \Omega_k^{t+1},
\]

where \( P^{t+1} = P^t Q^t \) and \( J^t P^t = I_K \) (\( J^t \in \mathbb{R}^{K \times N} \) and \( I_K \) is an identity matrix), \( \forall t \).

4.2 Convergence Analysis

Since the inner-level objective in (2) is non-convex, we focus on analyzing the convergence rate under the smooth case. Firstly, we can write the local updates as:

\[
\Omega_k^{t+1} = \Omega_k^t - \beta R H_1^t,
\]

where \( H_1^t = \frac{1}{R} \sum_{r=0}^{R-1} H_1^{t,r} \) and \( H_1^{t,r} = \frac{1}{R} (\Omega_k^r - \Theta_i(\Omega_k^{t,K})) \). Based on (5) and the update process in (4), we can obtain the global updates as:

\[
\omega_i^{t+1} = (1 - \alpha) \omega_i^t + \alpha \Omega_k^{t+1} P^{t+1}
\]

Definition 1 (L-smooth) (i.e., L-Lipschitz gradient) If a function \( f \) satisfies \( \| \nabla f(\omega) - \nabla f(\omega') \| \leq L \| \omega - \omega' \| \), \( \forall \omega, \omega' \), we say \( f \) is L-smooth.

Assumption 1 (Smoothness) The loss functions \( f_i \) is L-smooth and \( G(\omega_k) \) is \( L_G \)-smooth, \( \forall i, k \).

Assumption 2 (Bounded intra-context diversity) The variance of local gradients to the corresponding context-level generalized models is upper bounded by:

\[
\frac{1}{| G_k |} \sum_{i \in G_k} \| \nabla G_k(i, \omega_k) - \nabla G_k(\omega_k) \|^2 \leq \delta_i^2, \forall k \in [K],
\]

where \( G_k(i, \omega_k) := r(\theta_i, \omega_k) \).

Assumption 3 (Bounded parameters and gradients) The generalized model parameters \( \Omega_k^i \) and the gradients \( \nabla G_k(\Omega_k^i) \) are upper bounded by \( \rho_0 \) and \( \rho_g \), respectively.

\[
\| \nabla \omega_i G_i(\theta_i^{t+1}), \omega_i^{t+1} \| \leq \rho_0 \quad \text{and} \quad \| \nabla G_k(\Omega_k^i) \|^2 \leq \rho_g^2, \quad \forall t
\]

where \( \rho_0 \) and \( \rho_g \) are finite non-negative constants, and \( \nabla G_k(\Omega_k^i) := [\nabla G_1(\omega_1^i), \ldots, \nabla G_k(\omega_k^i), \ldots, \nabla G_K(\omega_K^i)] \).

Proposition 1 [T Dinh et al., 2020] The deviation between the \( \delta \)-approximate and the optimal solution is upper bounded by \( \delta \). That is:

\[
\mathbb{E} \left[ \left\| \tilde{\Theta}_i(\omega_i^{t+1}) - \Theta_i(\omega_i^{t+1}) \right\|^2 \right] \leq N \delta^2, \forall r, t.
\]

Remark 4.1 Theorem 4.1 shows that the proposed CGPFL can achieve a convergence rate of \( O(1/\sqrt{KNRT}) \), which is \( O(\sqrt{K}) \) times faster than what most of the state-of-the-art works ([Karimireddy et al., 2020; Deng et al., 2020; Reddi et al., 2020] achieved (i.e., \( O(1/\sqrt{NR}) \)) in non-convex FL setting. The detailed proof of convergence can be found in the full version of this paper [Tang et al., 2021].
4.3 Generalization Error

We analyze the generalization error of CGPFL in this section. Before starting the analysis, we first introduce two important definitions as follows.

Definition 2 (Complexity) Let \( \mathcal{H} \) be a hypothesis class (corresponding to \( \omega \in \mathbb{R}^d \) in neural network), and \( |D| \) be the size of dataset \( D \), the complexity of \( \mathcal{H} \) can be expressed by the maximum disagreement between two hypotheses on the dataset \( D \):

\[
\lambda_{\mathcal{H}}(D) = \sup_{h_1, h_2 \in \mathcal{H}} \frac{1}{|D|} \sum_{(x,y) \in D} |h_1(x) - h_2(x)|. \tag{9}
\]

Definition 3 (Label-discrepancy) Consider a hypothesis class \( \mathcal{H} \), the label-discrepancy between two data distributions \( D_1 \) and \( D_2 \) is given by:

\[
disc_{\mathcal{H}}(D_1, D_2) = \sup_{h \in \mathcal{H}} |\mathcal{L}_{D_1}(h) - \mathcal{L}_{D_2}(h)|, \tag{10}
\]

where \( \mathcal{L}_D(h) = \mathbb{E}_{(x,y) \in D} [(h(x), y)] \).

Theorem 4.2 (Generalization error bound of CGPFL) When Assumption 1 is satisfied, with probability at least \( 1 - \delta \), the following holds:

\[
\sum_{i=1}^N m_i \left\{ \mathcal{L}_{D_i}(\hat{h}_i) - \min_{h \in \mathcal{H}} \mathcal{L}_{D_i}(h) \right\} \\
\leq 2 \sqrt{\frac{\log \frac{N}{\delta}}{m}} + \sqrt{\frac{dK}{m} \log \frac{em}{\delta}} + (\lambda + \frac{L}{2}) \text{cost}(\Theta^*, \Omega^*; K) \\
+ \sum_{i=1}^N m_i \left\{ 2B \lambda_{\mathcal{H}}(D_i) + \text{disc}(D_i, \bar{D}_i) \right\},
\]

where \( B \) is a positive constant with \( |\mathcal{L}_D(h_1) - \mathcal{L}_D(h_2)| \leq B \lambda_{\mathcal{H}}(D), \forall h_1, h_2 \in \mathcal{H} \). Besides, \( \hat{h}_i \) is given by \( \hat{h}_i = \arg\min_\theta \left\{ \mathcal{L}_{D_i}(h(\theta_i)) + \|\theta_i - \omega^*_i\|^2 \right\} \) and \( \text{cost}(\Theta^*, \Omega^*; K) = \sum_{i=1}^N m_i \min_{k \in [K]} \|\theta^*_i - \omega^*_k\|^2 \).

Remark 4.2 Theorem 4.2 gives the generalization error bound of CGPFL. When \( K = 1 \), it yields the error bound of PFL with single global model [Li et al., 2021; T Dinh et al., 2020; Hanzely and Richtárik, 2020; Hanzely et al., 2020]. As the number of contexts increases, the second terms become larger, while the last term get smaller. Hence, our CGPFL can always reach better personalization-generalization trade-off by adjusting the number of contexts \( K \), and further achieve higher accuracy than the existing PFL methods. The detailed proof of generalization error can be found in the full version of this paper [Tang et al., 2021].

4.4 CGPFL-Heur: The Heuristic Improvement

As discussed, Theorem 4.2 indicates that there exists an optimal \( K^\ast\) (\( K^\ast \in [K] \)) to achieve the minimal generalization error bound that corresponds to the highest model accuracy. Theoretically, the optimal \( K^\ast \) can be obtained by minimizing the generalization bound in Theorem 4.2. We can find that the first and the third term have no relationship with the number of latent contexts, that is, they are irrelevant to \( K \). Therefore, we can obtain an optimal \( K^\ast \) by minimizing the following expression:

\[
e(K) := \sqrt{\frac{dK}{m} \log \frac{em}{\delta}} + \mu \cdot \text{cost}(\Theta^*, \Omega^*; K), \tag{11}
\]

where \( \mu \) is a hyper-parameter which is induced by the unknown constant \( L \). The above objective can be solved in the server along with the clustering. In the down-to-earth experiments, we notice that the latent context structure can be learned efficiently in the first few rounds. Based on this observation, we believe that CGPFL-Heur can efficiently figure out a near-optimal solution \( K \) by operating the solver of (11) only in the first few rounds (in the experimental part, we only operate the solver at the first global round), and after that, the obtained \( K \) will no longer be updated. In this way, CGPFL-Heur can reach a near-optimal trade-off between generalization and personalization with negligible additional computation in the server. Moreover, in view of the fact that we only need to operate the solver in the first few rounds, CGPFL-Heur can retain the same convergence rate as CGPFL.

5 Experiments

5.1 Experimental Setup

Dataset Setup: Three datasets including MNIST [LeCun et al., 1998], CIFAR10 [Krizhevsky, 2009], and Fashion-MNIST (FMNIST) [Xiao et al., 2017] are used in our experiments. To generate Non-I.I.D. datasets for the clients, we split the whole dataset as follows. 1) MNIST: we distribute the train-set containing 60,000 digital instances into 40 clients, and each of them is only provided with 3 classes out of total 10. The number of instances obtained by each client is randomly chosen from the range of \([400, 5000]\), of which 75% are used for training and the remaining 25% for testing. 2) CIFAR10: We distribute the whole dataset containing 60,000 instances into 40 clients, and each of them is also provided with 3 classes out of total 10. The number of instances obtained by each client is randomly chosen from the range of \([400, 5000]\). The train/test split remains 75%/25%.

3) Fashion-MNIST: It’s a more challenging replacement of MNIST, and the Non-I.I.D. splitting is the same as MNIST.

Competitors: We compare our CGPFL and CGPFL-Heur with seven state-of-the-art works: one traditional FL method, FedAvg [McMahan et al., 2017]; one typical cluster-based FL method, IFCA [Ghosh et al., 2020]; and five most recent PFL models, APFL [Deng et al., 2020], Per-FedAvg [Fallah et al., 2020], L2SGD [Hanzely and Richtárik, 2020], pFedMe [T Dinh et al., 2020], and Ditto [Li et al., 2021].

Model Architectures: 1) For strongly convex case, we use a \( l_2 \)-regularized multinomial logistic regression model (MLR) with the softmax and cross-entropy loss, in line with [T Dinh et al., 2020]; 2) For the non-convex case, we apply a neural network (DNN) with one hidden layer of size 128 and a softmax layer at the end for evaluation. In addition, we apply a CNN that has two convolutional layers and two fully connected layers for the CIFAR10. All competitors and our algorithms are based on the same configurations and fine-tuned to their best performances.

5.2 Overall Performance

The comprehensive comparison results of our CGPFL and CGPFL-Heur are shown in Table 1. It can be observed that our methods outperform the competitors with large margins.
for both non-convex and convex cases on all datasets, even if IFCA works with a good initialization. Besides, although we only provide the proof of convergence rate under non-convex case, as shown in Figure 1 and Figure 2, the extensive experiments further demonstrate that our methods constantly obtain better performance against multiple state-of-the-art PFL methods,\footnote{\cite{T-Dinh2020}} significantly obtain better performance against multiple state-of-the-art methods.\footnote{\cite{our-work}} The results are demonstrated in Figure 3(a). On the other hand, we conduct the CGPFL-Heur training with an appropriate $\mu$ and keep other parameters same as that of the above evaluation. As shown in Figure 3(a), we distinguish the results of CGPFL-Heur using red-star points. Besides, we make comparisons between the performance of a state-of-the-art PFL algorithm, pFedMe \cite{T-Dinh2020} with our proposed CGPFL with optimal $K$ and CGPFL-Heur in Figure 3(b). The results in Figure 3(a) and Figure 3(b) demonstrate that our designed heuristic algorithm CGPFL-Heur can effectively reach a near-optimal trade-off and consequently achieve the near-optimal model accuracy.

### 6 Conclusion

In this paper, we propose a novel personalized federated learning framework to handle the challenge of statistical heterogeneity (Non-I.I.D), especially contextual heterogeneity in the federated setting. To the best of our knowledge, we are the first to propose the concept of contextualized generalization (CG) for personalized federated learning and further formulate it to a bi-level optimization problem that is solved effectively. Our method provides fine-grained generalization knowledge for personalized models which can prompt higher test accuracy and facilitate faster model convergence. Experimental results on real-world datasets demonstrate the effectiveness of our method over the state-of-the-art works.
Acknowledgments

This research was supported by fundings from the Key-Area Research and Development Program of Guangdong Province (No. 2021B0101400003), Hong Kong RGC Research Impact Fund (No. R5060-19), General Research Fund (No. 152221/19E, 152203/20E, and 152244/21E), the National Natural Science Foundation of China (61872310 and 62102327), and Shenzhen Science and Technology Innovation Commission (JCYJ202001091942008673).

References

[Arivazhagan et al., 2019] Manoj Ghuhan Arivazhagan, Vinay Aggarwal, Aaditya Kumar Singh, and Sunav Choudhary. Federated learning with personalization layers. arXiv preprint arXiv:1912.00818, 2019.

[Briggs et al., 2020] Christopher Briggs, Zhong Fan, and Peter Andras. Federated learning with hierarchical clustering of local updates to improve training on non-iid data. In 2020 International Joint Conference on Neural Networks (IJCNN), pages 1–9. IEEE, 2020.

[Deng et al., 2020] Yuyang Deng, Mohammad Mahdi Kamani, and Mehrdad Mahdavi. Adaptive personalized federated learning. arXiv preprint arXiv:2003.13461, 2020.

[Fallah et al., 2020] Alireza Fallah, Aryan Mokhtari, and Asuman Ozdaglar. Personalized federated learning with theoretical guarantees: A model-agnostic meta-learning approach. Advances in Neural Information Processing Systems, 33, 2020.

[Ghosh et al., 2020] Avishek Ghosh, Jichan Chung, Dong Yin, and Kannan Ramchandran. An efficient framework for clustered federated learning. Advances in Neural Information Processing Systems, 33, 2020.

[Hanzely and Richtárik, 2020] Filip Hanzely and Peter Richtárik. Federated learning of a mixture of global and local models. arXiv preprint arXiv:2002.05516, 2020.

[Hanzely et al., 2020] Filip Hanzely, Slavomír Hanzely, Samuel Horváth, and Peter Richtárik. Lower bounds and optimal algorithms for personalized federated learning. Advances in Neural Information Processing Systems, 33, 2020.

[Hsieh et al., 2020] Kevin Hsieh, Amar Panishayee, Onur Mutlu, and Phillip Gibbons. The non-iid data quagmire of decentralized machine learning. In International Conference on Machine Learning, pages 4387–4398. PMLR, 2020.

[Kairouz et al., 2019] Peter Kairouz, H Brendan McMahan, Brendan Avent, Aurélien Bellet, Mehdi Bennis, Arjun Nitin Bhagoji, Keith Bonawitz, Zachary Charles, Graham Cormode, Rachel Cummings, et al. Advances and open problems in federated learning. arXiv preprint arXiv:1912.04977, 2019.

[Karimireddy et al., 2020] Sai Praneeth Karimireddy, Satyen Kale, Mehryar Mohri, Sashank Reddi, Sebastian Stich, and Ananda Theertha Suresh. Scaffold: Stochastic controlled averaging for federated learning. In International Conference on Machine Learning, pages 5132–5143. PMLR, 2020.

[Krizhevsky, 2009] Alex Krizhevsky. Learning multiple layers of features from tiny images. Technical report, University of Toronto, 2009.

[LeCun et al., 1998] Yann LeCun, Léon Bottou, Yoshua Bengio, and Patrick Haffner. Gradient-based learning applied to document recognition. Proceedings of the IEEE, 86(11):2278–2324, 1998.

[Li et al., 2021] Tian Li, Shengyuan Hu, Ahmad Beirami, and Virginia Smith. Ditto: Fair and robust federated learning through personalization. In International Conference on Machine Learning, pages 6357–6368. PMLR, 2021.

[Lloyd, 1982] Stuart Lloyd. Least squares quantization in pcm. IEEE transactions on information theory, 28(2):129–137, 1982.

[Mansour et al., 2020] Yishay Mansour, Mehryar Mohri, Jae Ro, and Ananda Theertha Suresh. Three approaches for personalization with applications to federated learning. arXiv preprint arXiv:2002.10619, 2020.

[McMahan et al., 2017] Brendan McMahan, Eider Moore, Daniel Ramage, Seth Hampson, and Blaise Agua y Arcas. Communication-efficient learning of deep networks from decentralized data. In Artificial intelligence and statistics, pages 1273–1282. PMLR, 2017.

[Reddi et al., 2020] Sashank J Reddi, Zachary Charles, Manzil Zaheer, Zachary Garrett, Keith Rush, Jakub Konečný, Sanjiv Kumar, and Hugh Brendan McMahan. Adaptive federated optimization. In International Conference on Learning Representations, 2020.

[Sattler et al., 2020] Felix Sattler, Klaus-Robert Müller, and Wojciech Samek. Clustered federated learning: Model-agnostic distributed multitask optimization under privacy constraints. IEEE Transactions on Neural Networks and Learning Systems, 2020.

[Smith et al., 2017] Virginia Smith, Chao-Kai Chiang, Maziar Sanjabi, and Ameet S Talwalkar. Federated multi-task learning. Advances in neural information processing systems, 30, 2017.

[T Dinh et al., 2020] Canh T Dinh, Nguyen Tran, and Tuan Dung Nguyen. Personalized federated learning with moreau envelopes. Advances in Neural Information Processing Systems, 33, 2020.

[Tang et al., 2021] Xueyang Tang, Song Guo, and Jingcai Guo. Personalized federated learning with clustered generalization. arXiv preprint arXiv:2106.13044, 2021.

[Xiao et al., 2017] Han Xiao, Kashif Rasul, and Roland Vollgraf. Fashion-mnist: a novel image dataset for benchmarking machine learning algorithms. arXiv preprint arXiv:1708.07747, 2017.

[Zhang et al., 2021] Jie Zhang, Song Guo, Xiaosong Ma, Haozhao Wang, Wenchao Xu, and Feijie Wu. Parameterized knowledge transfer for personalized federated learning. Advances in Neural Information Processing Systems, 34, 2021.