Fast Heterogeneous Federated Learning with Hybrid Client Selection

Duanxiao Song\textsuperscript{1}  Guangyuan Shen\textsuperscript{1,2}  Dehong Gao\textsuperscript{1,2}  Libin Yang\textsuperscript{1}  Xukai Zhou\textsuperscript{1}  Shirui Pan\textsuperscript{3}  Wei Lou\textsuperscript{4}  Fang Zhou\textsuperscript{1}

\textsuperscript{1}Department of Cybersecurity, Northwestern Polytechnical University, China  
\textsuperscript{2}Alibaba Group, China  
\textsuperscript{3}School of Information and Communication Technology, Griffith University, Australia  
\textsuperscript{4}Department of Computing, The Hong Kong Polytechnic University, Hong Kong, China

Abstract

Client selection schemes are widely adopted to handle communication-efficient problems in recent studies of Federated Learning (FL). However, the large variance of the model updates aggregated from the randomly-selected unrepresentative subsets directly slows the FL convergence. We present a novel clustering-based client selection scheme to accelerate the FL convergence by variance reduction. Simple yet effective schemes are designed to improve the clustering effect and control the effect fluctuation, therefore, generating the client subset with certain representativeness of sampling. Theoretically, we demonstrate the improvement of the proposed scheme in variance reduction. We also present the tighter convergence guarantee of the proposed method thanks to the variance reduction. Experimental results confirm the exceed efficiency of our scheme compared to alternatives.

1 INTRODUCTION

Federated Learning (FL) is a distributed learning paradigm for training a global model from data scattered across different clients\cite{Konecny2015}. During the training process, all the clients need to operate data locally and transfer the model updates between servers and themselves back and forth. Such a training process may raise many challenges, with communication cost often being the critical bottleneck\cite{Kairouz2021a}.

Many studies have found that different clients might transfer similar (or redundant) model updates to the server, which is a waste of communication costs\cite{Balakrishnan2022, Fraboni2021, Karimireddy2020b}. Client selection schemes are widely adopted to reduce the waste of communication costs, e.g., Federated Averaging (FedAvg) where only the randomly-selected subset of clients transfer their model updates to the server instead of yielding all clients involved\cite{McMahan2017}. FedAvg can maintain the learning efficiency with reduced communication costs, since the random selection schemes can reduce the redundant update transmission. When FedAvg meets client sets with low similarity, the server can not accurately pick out the representative clients. The large variance of the update gradient aggregated from the unrepresentative subset directly slows the training convergence\cite{Fraboni2021}.

To accelerate the training convergence by variance reduction, many client selection criteria have been proposed in recent literature, e.g., importance sampling, where the probabilities for clients to be selected is proportional to their importance measured by the norm of updates\cite{Chen2022}, data variability\cite{Rizk2021}, and test accuracy\cite{Mohammed2020}. However, importance sampling could not effectively capture the similarities among the clients. As shown in Figure\textsuperscript{1}(a), applying importance sampling could cause learning inefficiency as the clients transfer excessive important yet similar updates to the server. To make better use of the similarities among clients, some researchers propose raw gradient-based cluster sampling schemes. As shown in Figure\textsuperscript{1}(b), they first group the clients with similar model updates together\cite{Fraboni2021, Muhammad2020a}, and apply random selection within each cluster. However, cluster sampling schemes only work under the strong assumption that we can have a decent clustering effect all the time, i.e., almost all clusters have small intra-cluster distances, with little variation from cluster to cluster. Unfortunately, the effect of raw gradient-based clustering methods faces the following problems: (i) Poor Effect, the high dimension gradient of a client is too complicated to be an appropriate cluster feature, which can not bring small intra-cluster distances. (ii) Effect Fluctuation, due to the limitations of the clustering algorithm, the clustering effect
tends to fluctuate greatly, i.e., clusters tend to have diverse heterogeneity. In practice, such diverse heterogeneity always leads to the intra-cluster distances of different clusters varying greatly. After clustering with poor and fluctuant effects, there always exists some clusters with low client similarity (please refer to Figure 2). Applying client selection in the clusters with low client similarity may slow the overall training convergence. Besides, cluster sampling methods require all the clients to return the raw gradient for better selection, which runs against the communication reduction objective.

We propose HCSFed, a novel Hybrid Clustering-based client Selection scheme for heterogeneous Federated learning that further accelerates the FL convergence. To improve the poor clustering effect, different from using the high dimension gradient, we adopt the compressed gradient as the cluster feature. Adopting the compressed gradient as a cluster feature not only allows a better clustering effect but also is communication-efficient since the compression operation filters out the redundant information within each gradient. Besides, to control the effect fluctuation, we reset the number of clients to be selected in each cluster based on the client similarity and the size of the cluster, which we term as sample size re-allocation. As shown in Figure 1(c), the sample size re-allocation scheme makes the cluster with low similarity have more clients to be selected. The sampling effect becomes stable because the poor effect of random sampling in the clusters with low similarity is greatly mitigated by the sample size re-allocation scheme. Theoretically, we demonstrate the improvement HCSFed in variance reduction. Extensive experiments show the exceed efficiency of the proposed scheme compared to the cutting-edge FL client selection criteria in both convex and non-convex settings. Our main contributions are summarized below:

- To the best of our knowledge, this work provides the first insight into modeling the diverse heterogeneity of clusters which leads to the client sampling effect fluctuation. (please refer to our sample size re-allocation part section 3)
- We present the explicit cross-client variance comparison among different client selection methods which is a new perspective to watch the convergence improvement of the advanced selection method. We also provide detailed discussion and proof of Theorem 2 proving that HCSFed has a convergence guarantee. (please refer to our theoretical analysis part section 4)
- A series of experiments in different settings verify the effectiveness of our proposed method and the correctness of our theoretical results.

2 SYSTEM SETUP AND PRIOR WORK

The Federated Learning Optimization Settings. In federated learning, a total number of $N$ clients aim to jointly solve the following distributed optimization model:

$$\min_{w \in \mathbb{R}^d} F(w) := \sum_{k=1}^{N} \omega_k F_k(w) := \frac{1}{n_k} \sum_{j=1}^{n_k} f(w; x_{k,j}),$$

where $\omega_k$ is the weight of the $k$th client, s.t. $\omega_k \geq 0$ and $\sum_{k=1}^{N} \omega_k = 1$. Suppose the $k$th client possesses $n_k$ training data: $\{x_{k,1}, x_{k,2}, \ldots , x_{k,n_k}\}$. The local objective function $F_k(w)$ is defined as the average of $f(w; x_{k,j})$, which is a user-specified loss function (possibly non-convex) made with model parameters $w \in \mathbb{R}^d$ and training data $x_{k,j}$.

FedAvg Description. In the $t$th train round of the FedAvg, the server broadcasts the latest global model $w_t$ to all the clients. The $k$th client sets $w_t^k \leftarrow w_t$ and performs local training for $E$ epochs:

$$w_{t+1}^k \leftarrow w_{t}^k - \eta_t \nabla F_k \left( w_{t}^k, \xi_{t}^k \right),$$

where $\eta_t$ is the learning rate and $\xi_{t}^k$ is a data sample uniformly selected from the local data. In the communication round, the server aggregates the local model updates to produce the new global model $w_{t+E}$. In fact, to reduce the communication cost, FedAvg randomly generates a subset $S_t$ consisting of $m$ clients from the entire client set and aggregates the model updates, $\{w_{t+1}^1, w_{t+1}^2, \ldots , w_{t+1}^m\}$,

$$w_{t+E} \leftarrow \sum_{k=1}^{N} \omega_k w_{t+E}^k \leftarrow \sum_{k \in S_t} \frac{N}{m} \omega_k w_{t+E}^k.$$
is still large, which directly slows the FL convergence. To accelerate the convergence by variance reduction, many client selection criteria have been proposed in recent literature, including importance-based and clustering-based methods.

**Importance Sampling.** Importance sampling is a non-uniform client selection method, where the probabilities for clients to be chosen are proportional to their importance Katharopoulos and Fleuret [2018], Zhao and Zhang [2015]. Different methods to measure the importance are proposed, including data variability Rizk et al. [2021], the norm of updates Chen et al. [2022], test accuracy Mohammed et al. [2020], local rounds Singh et al. [2019], and local loss Cho et al. [2020]. Such methods can yield better selection by activating the clients with more importance. However, they could not capture the similarities among the updates of clients, therefore, triggering the communication redundancy.

**Cluster Sampling.** To make better use of similarities among clients, some researchers propose raw gradient-based cluster sampling schemes that group the clients with similar gradients together Fraboni et al. [2021], Mohammed et al. [2020]. However, the effect of cluster sampling strongly relies on the performance of the clustering process. Unfortunately, the high dimension gradient of a client is too complicated to be an appropriate cluster feature, which can not bring a good clustering effect. Besides, meticulous one-by-one client selection Balakrishnan et al. [2022], Dieuleveut et al. [2021] is introduced to FL. It could achieve better selection outcomes but with extra expensive computation for solving each noncommittal submodular maximization. We want to claim that cluster sampling needs to transmit the raw gradient information of all the clients to the server, which would bring unacceptable communication costs. And if the server has received all the updates, why don’t we aggregate them immediately? When all the clients participate in the training, there is no cross-client anymore.

**Difference from SCAFFOLD.** While SCAFFOLD Karimireddy et al. [2020], Gurbunov et al. [2021], Karimireddy et al. [2020a] appears to be similar to our method, there are fundamental differences. SCAFFOLD augments the updates with extra “control-variate” item that is also transmitted along with the updates to reduce the variance, while we reduce the variance by selecting a more representative subset. The former focuses on update control, while the latter focuses on better update selection, indicating that they are orthogonal and compatible with each other.

### 3 METHOD

How to explore and make use of the similarities among clients with acceptable resource costs is the key issue in constructing client selection criteria. We propose HCSFed, a selection scheme that exploits three components (cluster sampling, sample size re-allocation, importance selection) guaranteeing convergence speedup by variance reduction.

**The Compressed Gradient-Based Cluster Selection.** To make use of the similarity among clients, we develop cluster selection that groups the similar clients together based on the client characteristic. The following is a mathematical description of cluster selection. Generating a subset $S_h$ consisting of $m$ clients by clustering selection can be regarded as selecting $m$ clients using $H$ different probability distributions $\{ P_h \}_{h=1}^H$ where $P_h = \{ p_k^h \}_{k=1}^N$. For example, if we select a client from the $h$th cluster, the probability distribution $P_h$ will be used, and the $p_k^h$ comes as follows:

$$p_k^h = \begin{cases} 0, & \text{if } k \text{th client } \notin h \text{th cluster} \\ \frac{1}{N_h}, & \text{if } k \text{th client } \in h \text{th cluster} \end{cases}$$

where $N_h$ denotes the number of clients in the $h$th cluster.

**Algorithm 1: ClientClustering**

**Input:** compressed gradient of all the clients in the $t$th round $\{ X^k_t \}_{k=1}^N$.

**Output:** The number of the clusters $H$.

1. Initialize Randomly select $H$ clients as cluster centers $\{ \mu_1, \mu_2, \ldots, \mu_H \}$.
2. Initialize $C_i = \emptyset$ $(1 \leq i \leq H)$.
3. **repeat**
   4. **for each client** $k = 1, 2, \ldots, N$ **do**
      5. $\lambda_k = \arg \min_{i \in \{1, 2, \ldots, H\}} \|X^k_t - \mu_i\|_2$;
      6. $C_{\lambda_k} = C_{\lambda_k} \cup \{ k \text{th client with } X^k_t \}$;
   **end**
8. **for each cluster** $i = 1, 2, \ldots, H$ **do**
   9. New center $\mu_i = \frac{1}{|C_i|} \sum_{X^k_t \in C_i} X^k_t$;
10. $\mu_i \leftarrow \mu_i$;
11. **end**
12. **until** $\forall i = \{1, 2, \ldots, H\}, \mu_i = \mu_i$ **;**

**Output:** $G = \{ C_1, C_2, \ldots, C_H \}$;

**Difference from SCAFFOLD.** While SCAFFOLD Karimireddy et al. [2020], Gurbunov et al. [2021], Karimireddy et al. [2020a] appears to be similar to our method, there are fundamental differences. SCAFFOLD augments the updates with extra “control-variate” item that is also transmitted along with the updates to reduce the variance, while we reduce the variance by selecting a more representative subset. The former focuses on update control, while the latter focuses on better update selection, indicating that they are orthogonal and compatible with each other.

---

1. Typical approaches like PCA cannot be adopted in our work: each client operates data locally, which means reduced dimension $d'$ is strictly lower than the min value of the number of samples $m$ and the number of raw dimension $d$. 

2008
Sample Size Re-allocation Scheme. Concentrating on the diverse heterogeneity of clusters, we make the first attempt to develop a sample size re-allocation scheme that pays more attention to the cluster with great heterogeneity. Here, it redetermines the sample size $m_h$, namely, the number of clients sampled from the $h^{th}$ cluster, by considering both the cluster size $N_h$ and variability.

$$m_h = \frac{N_h S_h}{\sum_{h=1}^{H} N_h S_h} \cdot m,$$  \hspace{1cm} (5)

where $S_h = \frac{1}{N_h-1} \sum_{j=1, j \neq i}^{N_h} \|X_i - X_j\|_2^2$, $m_h$ denotes the number of clients in the subset $S_i$ from the $h^{th}$ cluster s.t. $\sum_{h=1}^{H} m_h = m$. $S_h$ denotes the variability of the $h^{th}$ cluster where we use Cluster Cohesion based on the compressed model updates to approximate. The introduction of sample size re-allocation can further reduce the variance by assigning more sample chance to those clusters with greater heterogeneity. Due to the limited total number of clients to be selected, sometimes some clusters with great heterogeneity still cannot have an adequate sample chance, which indeed need extra care.

Importance Selection. We introduce importance selection to optimize the probabilities for clients to be selected in one cluster, which is based on the norm of the compressed gradient, i.e., $GC(G_h^t) = X_h^t$. We re-determine the probability for each client to be sampled in $i^{th}$ round as follows,

$$p_i^k = \frac{\|GC(G_i^t)\|}{\sum_{k=1}^{N_h} \|GC(G_k^t)\|} = \frac{\|X_i^k\|}{\sum_{k=1}^{N_h} \|X_k^k\|},$$  \hspace{1cm} (6)

if $k^{th}$ client $\in h^{th}$ cluster in $i^{th}$ round.

After completing the importance selection, the client with a higher norm of gradient (more importance) will have a higher probability to be sampled, which brings representative outcomes under an inadequate sampling ratio. Importance selection provides a fine-grained optimization over cluster selection with sample size re-allocation, i.e., assigning more attention to the more representative clients.

These ideas of selection discussed in this subsection together constitute the HCSFed to optimize the client selection in FL. HCSFed selects representative clients via allocating appropriate probability for each client to be selected, with the promise of reducing the variance between the model.
**Algorithm 2: HCSFed**

**Input:** updates in the $t^{th}$ round of all clients $\{G_t^k\}_{k=1}^N$

1. Initialize $w_0$;
2. Initialize $S_t = \emptyset$;
3. Server executes:
   4. for each round $t = 1, 2, \cdots$ do
      5. $m \leftarrow \max(qN, 1)$;
      6. $G \leftarrow \text{ClientClustering}(X_t, H)$;
      7. for each cluster $h = 1, 2, \cdots, H$ do
         8. $m_h \leftarrow \frac{\sum_{k \in S_h} N_k}{N_h}$;
         9. compute $\{v_h^k\}_{k \in N_h}$ using Equation [6];
      10. $S_t = S_t \cup m_h$ clients selected with $P_h$;
   11. for client $i \in S_t$ in parallel do
      12. $w_h^{k+1} \leftarrow \text{ClientUpdate}(k, w_h^k)$;
   13. end
   14. $w_{t+1} \leftarrow \sum_{k \in S_t} \frac{N}{m} w_k^{t+1}$;
   15. end
   16. ClientUpdate$(k, w_k^t)$:
   17. for each local epoch $i$ from 1 to $E$ do
      18. $w_h^{k+1} \leftarrow w_h^{k+1} - \eta \nabla F_k(w_h^{k+1}, c_h^{k+1})$;
   19. end
   20. if $k \in S_t$ then
      21. return $w_h^{k+1} \leftarrow w_h^{k+1}, X_h^{k+1} \leftarrow \text{GC}(G_t^k)$;
   22. else
      23. return $X_h^{k+1} \leftarrow \text{GC}(G_t^k)$;
   24. end

Update aggregated from the sampled subset $S_t$ and the entire set $K$. Algorithm 2 summarizes the main steps of HCSFed.

4. **THEORETICAL ANALYSIS**

4.1 **VARIANCE RELATIONSHIP AMONG SELECTION SCHEMES**

**Theorem 1** (Variance Reduction). If the population is large compared to the subset, $\frac{N}{N_k}$, $\frac{1}{m_h}$, and $\frac{1}{N_h}$ are negligible, then the cross-client variance of different selection schemes satisfy

$$\mathbb{V}(w_{\text{hybrid}}) \leq \mathbb{V}(w_{\text{cludiv}}) \leq \mathbb{V}(w_{\text{cluster}}) \leq \mathbb{V}(w_{\text{rand}}),$$

where $w_{\text{hybrid}}, w_{\text{cludiv}}, w_{\text{cluster}}, w_{\text{rand}}$ denote the model update aggregated from the subset $S_t$ that generated by HCSFed, clustering selection scheme under re-allocation, clustering selection scheme under plain allocation and simple random selection scheme, respectively.

**Proof Sketch.** Below we provide a proof sketch to reveal the relationship among the variance of different client selection schemes, and then we naturally state when our HCSFed scheme could achieve variance reduction. We defer the details of proof to Appendix B.

**Comparison of Random and Clustering Selection.** We derive Equation [7] to show the relationship between $\mathbb{V}(w_{\text{cludiv}})$ and $\mathbb{V}(w_{\text{rand}})$:

$$\mathbb{V}(w_{\text{rand}}) = \mathbb{V}(w_{\text{cludiv}}) + \frac{\sum_{h=1}^H {N_h} \| W_h - W(K) \|^2}{mN},$$

where $K$ denotes the set of all the clients. We have $\mathbb{V}(w_{\text{cludiv}}) < \mathbb{V}(w_{\text{rand}})$, unless $\forall h \in \{1, \cdots, H\}, W_h = W(K)$, i.e., each cluster has the same averaged model update with the entire population, which indicates that all the clusters are homogeneous in terms of the mean model update.

**Comparison of Plain Clustering and Clustering with Re-allocation.** We derive Equation [8] to show the relationship between $\mathbb{V}(w_{\text{cludiv}})$ and $\mathbb{V}(w_{\text{cluster}})$:

$$\mathbb{V}(w_{\text{cluster}}) = \mathbb{V}(w_{\text{cludiv}}) + \frac{N_h (S_h - S)^2}{mN},$$

We have $\mathbb{V}(w_{\text{cludiv}}) < \mathbb{V}(w_{\text{cluster}})$, unless $\forall h \in \{1, \cdots, H\}, S_h = S$, i.e., each cluster has equal variability, which indicates that all the clusters are homogeneous in terms of the variability.

**Comparison of Clustering with Re-allocation and HCSFed.** We derive Equation [9] to show the relationship between $\mathbb{V}(w_{\text{cludiv}})$ and $\mathbb{V}(w_{\text{hybrid}})$:

$$\mathbb{V}(w_{\text{hybrid}}) = \mathbb{V}(w_{\text{cludiv}}) + \left( \frac{\sum_{i=1}^{N_h} \| G_i \|^2}{N_h} \right) \frac{\sum_{i=1}^{N_h} \left( I_i - \frac{1}{N_h} \right)^2}{N_h},$$

We have $\mathbb{V}(w_{\text{hybrid}}) < \mathbb{V}(w_{\text{cludiv}}), \forall i \in \{1, \cdots, N_h\}, I_i = \frac{\sum_{i=1}^{N_h} \| G_i \|^2}{N_h} = \frac{1}{N_h}, $ i.e., each client in $i^{th}$ cluster has equal norm of model update.

**Remark 1.** Theorem 1 verifies the capability of HCSFed in variance reduction. The proof sketch indicates that the variance difference of different selections will vanish only if all the clusters and the norm of the gradients are identical. The proposed approach can achieve variance reduction, since each cluster never has identical update mean, variability, and norm in heterogeneous FL.

4.2 **CONVERGENCE BEHAVIOR OF HCSFED**

We emphasize that the convergence analysis of FedAvg with random selection is NOT our contribution, we just follow the previous analysis work Li et al. [2019] to verify that HCSFed could maintain the same convergence guarantees.
Table 1: Required rounds for different methods with convex model (logistic regression) to achieve 80\% accuracy on non-IID MNIST and FMNIST. 200+ indicates 80\% accuracy was not reached after 200 rounds.

| Methods   | Sampling Ratio 10\% | Sampling Ratio 30\% | Sampling Ratio 50\% |
|-----------|---------------------|---------------------|---------------------|
|           | Num. of Rounds | Speedup | Num. of Rounds | Speedup | Num. of Rounds | Speedup |
| Random    | 96 | (1.0×) | 42 | (1.0×) | 28 | (1.0×) |
| SCAFFOLD | 74 | (1.3×) | 36 | (1.2×) | 21 | (1.3×) |
| Importance Cluster | 35 | (2.7×) | 24 | (1.8×) | 22 | (1.3×) |
| HCSFed    | 31 | (3.1×) | 16 | (2.6×) | 8 | (3.5×) |
|           | 9 | (10.7×) | 8 | (5.2×) | 8 | (3.5×) |
| Random    | 200+ | (1.0×) | 200+ | (1.0×) | 158 | (1.0×) |
| SCAFFOLD | 180 | (>1.1×) | 144 | (>1.4×) | 115 | (>1.4×) |
| Importance Cluster | 180 | (>1.1×) | 116 | (>1.7×) | 100 | (>1.6×) |
| HCSFed    | 145 | (>1.4×) | 104 | (>1.9×) | 75 | (2.1×) |
|           | 81 | (>2.5×) | 80 | (>2.5×) | 75 | (2.1×) |

as a random selection in the worst case and in most cases, HCSFed will enjoy tighter convergence.

We next state the assumptions used in our theorem and proof, which are common in FL optimization literature, e.g., Zhou and Cong [2017], Li et al. [2019], Wang and Joshi [2021], Stich [2018], Haddadpour and Mahdavi [2019], Haddadpour et al. [2019]. Assume each function $F_k (k \in [N])$ is $\mu$-strongly convex and $L$-smooth. Suppose that for all $k \in [N]$ and all $t$, the variance and expectation of stochastic gradients in each client on random samples $\xi$ are bounded by $\sigma_k^2$ and $G^2$, i.e., $E \left[ \left\| \nabla F_k (w^k_t, \xi) - \nabla F_k (w^k_t) \right\|^2 \right] \leq \sigma_k^2$ and $E \left[ \left\| \nabla F_k^2 (w^k_t, \xi) \right\|^2 \right] \leq G^2$, respectively. And $\Gamma$ is used to quantify the heterogeneity, where $\Gamma = F^* - \sum_{k=1}^{N} p_k F_k^*$.

**Theorem 2 (Convergence Bound).** Let Assumptions above hold and $L$, $\mu$, $\sigma_k$, $G$ be defined therein. Consider FedAvg when sampling $m$ clients, then HCSFed satisfies:

$$
E [F(w_T)] - F^* \leq O \left( \frac{\sum_{k=1}^{N} p_k^2 \sigma_k^2 + E^2 G^2 + \gamma G^2}{\mu T} \right) + O \left( \frac{L \Gamma}{\mu T} \right) + O \left( \frac{E^2 G^2}{m \mu T} \right) - O \left( \frac{\rho_0^2}{m} \right) - O \left( \frac{\lambda^2}{m} \right)
$$

(10)

**Remark 3.** Theorem [2] Li et al. [2019] uses Lemma 1-3 to bound the error of FedAvg with full participation corresponding to the first term, while using Lemma 4 to verify the unbiased property and Lemma 5 to bind the variance resulting from the client selection. Naturally, HCSFed satisfies the Lemma 5, since the Theorem 7 shows that HCSFed could achieve variance reduction. The dependency between the Lemmas and Theorem 2 indicates that the proof the Lemma 4 is enough to maintain the convergence bound. We defer the proof to Appendix A.3.

We demonstrate the convergence guarantee of HCSFed under $\mu$-strongly convex assumption. As for the non-convex loss function, we refer to the previous proved proposition that FedAvg with any unbiased selections maintains the same FL convergence bound of random selection (Theorem 2 in Fraboni et al. [2021]).

**5 EXPERIMENT**

**5.1 EXPERIMENTAL SETUP**

We run logistic regression (convex) and a fully connected network with one hidden layer of 50 nodes (non-convex) on MNIST [LeCun 1998]. As for CIFAR-10 [Krizhevsky et al. 2009] and FMNIST [Xiao et al. 2017], we use the same classifier of FedAvg [McMahan et al. 2017] composed of 3 convolutional and 2 fully connected layers. We partition the dataset into 100 clients under both IID and non-IID data distribution. IID data partition strategy employs the idea of random split to create a uniform federated dataset. As for non-IID data partition, we use Dirichlet distribution to partition the entire client set. Dirichlet distribution, i.e., $\text{Dir}(\alpha)$, gives to each client the respective partitioning across labels by changing the value of $\alpha$. Specifically, the lower the value of $\alpha$, the more heterogeneous the dataset is. The schematic table of the Dirichlet distribution is presented in Appendix.
A.1. As for the hyperparameters, $N$ is the number of all clients, $q$ is the sampling ratio, $nSGD$ is the times of SGD running locally, $\eta$ is the learning rate, $B$ is the batch size, $T$ is the terminate round. For better comparison, we set all the hyperparameters following the FL optimization work FedProx [Li et al., 2020].

5.2 MAIN RESULTS

Our evaluation target is to compare the convergence speed of different methods in convex and non-convex FL context. We choose image classification as our downstream task and compare HCSFed with simple random sampling [McMahan et al., 2017], norm-based importance sampling [Chen et al., 2022], cluster sampling [Fraboni et al., 2021], IFCA [Ghosh et al., 2020], and SCAFFOLD [Karimireddy et al., 2020b]. For better representation of the convergence speed, we follow the previous FL acceleration work setting [Karimireddy et al., 2020b] and report the required rounds for different methods with convex model to reach the target test accuracy in Table 1. We can observe that HCSFed consistently achieves the fastest convergence (has the fewest required rounds to reach the target accuracy) across all settings, surpassing not only the client selection schemes but also the SOTA variance-reduced method SCAFFOLD [Karimireddy et al., 2020b].

For better representation of the convergence speed, we follow the previous FL acceleration work setting [Karimireddy et al., 2020b] and report the required rounds for different methods with convex model to reach the target test accuracy in Table 1. We can observe that HCSFed consistently achieves the fastest convergence (has the fewest required rounds to reach the target accuracy) across all settings, surpassing not only the client selection schemes but also the SOTA variance-reduced method SCAFFOLD [Karimireddy et al., 2020b]. We also run HCSFed with non-convex model on different datasets. The result of non-convex experiment is presented in Figure 3. HCSFed consistently requires the fewest rounds to reach the target accuracy. All the results reveal that HCSFed can greatly accelerate the convergence, especially in the low sampling ratio and data heterogeneity case. As we emphasized in the theoretical analysis, the improvement of HCSFed depends on the sampling ratio and the client heterogeneity.

Figure 4: Visualization of Sensitivity Study Results on MNIST (non-IID).

The lower the sampling ratio and the more heterogeneous the dataset, the more improvement HCSFed can enjoy. Naturally, such great improvement in convergence speed can be attributed to the two key components in HCSFed: (i) Different from the previous raw gradient-based methods, we introduce gradient compression to improve the clustering effect by using the more expressive cluster feature representation.

Figure 5: Required rounds for random, cluster, importance sampling, SCAFFOLD, IFCA and HCSFed to achieve 60% accuracy with $q = 0.1$, $N = 100$, $nSGD = 50$, $\eta = 0.01$, $B = 50$ on MNIST and FMNIST, with $\alpha \in \{0.01, 0.001\}$, $q = 0.1$, $N = 100$, $nSGD = 80$, $\eta = 0.05$, $B = 50$ on CIFAR-10.
(ii) To further guarantee the robustness of the sampling effect, we introduce a sample size re-allocation scheme and extra importance sampling to control the sampling effect fluctuation in the cluster with low client similarity. For more details, please refer to Appendix C.

### 5.3 PARAMETER SENSITIVITY STUDY

**The Number of Clusters (H).** This experiment demonstrates the sensitivity of HCSFed to the different numbers of clusters. We run convex model-based HCSFed on non-IID MNIST with different numbers of clusters, i.e. with \( H \in \{5, 6, \ldots, 10\} \). We report the best classification accuracy and the required rounds for HCSFed with different numbers of clusters to reach the best test accuracy in Figure 4(a). HCSFed achieves stable classification accuracy and similar convergence speeds (measured by the required rounds to achieve the best accuracy), which highlights the robust efficiency of HCSFed. The results also reveal that the clustering effect of HCSFed is robust regardless of the number of clusters. Such merit might attribute to the sample size re-allocation scheme or the importance sampling scheme that we use to control the clustering effect fluctuation.

![Figure 4(a)](image)

*Figure 4(a): Visualization of Sensitivity Study Results on CIFAR-100.*

| Compression Rate | Rounds | Best Accuracy |
|------------------|--------|---------------|
| 300              | 30     |               |
| 350              | 32     |               |
| 400              | 34     |               |
| 450              | 36     |               |
| 500              | 38     |               |
| 550              | 40     |               |
| 600              | 42     |               |
| 650              | 44     |               |
| 700              | 46     |               |

![Figure 5](image)

*Figure 5: Visualization of Sensitivity Study Results on CIFAR-10 (\( \alpha = 0.01 \)).

**The Compression Rate (R).** We also evaluate how different compression rates affect the global model performance. We run convex model-based HCSFed on non-IID MNIST with different compression rates (R), i.e. with \( R \in \{5\%, 10\%, 20\%, 40\%, 100\%\} \). We report the best accuracy and the required rounds of HCSFed with different compression rates to reach the best test accuracy in Figure 4(b). The results show that HCSFed achieves different convergence speed as the compression rate changes. The noticeable information is that both too high and too low compression rates do not achieve good learning efficiency. Intuitively, the low compression rate cannot ensure the integrity of gradient information while the high compression rate is too complicated to learn efficiently. When we set the compressed rate to 100\%, i.e., without gradient compression, the results show that both the convergence speed and final accuracy of the model are not satisfactory, which confirms the effectiveness of gradient compression. In practice, following the previous compression work [Han et al., 2015], we set the compression rate with the highest within-group sum of squares to get a good sampling effect.

To further verify the influence of parameters, we provide the sensitivity study results on CIFAR-10 in Figure 5 and CIFAR-100 in Figure 6.

![Figure 6](image)

**5.4 ABSTRACTION STUDY**

We run logistic regression (convex) on non-IID MNIST with \( q = 0.1, nSGD = 50 \) to analyze the individual contributions of different components of HCSFed in both final accuracy and convergence speed. In Figure 7(a), HCSFed is represented as the green line starting from the accuracy of 10\% and ending at 91\%, whereas FedAvg is the yellow line starting also from the same position as HCSFed but ending at 81\% after 200 rounds, i.e., 10\% less. We report the required rounds for different methods to achieve 80\% accuracy in Figure 7(b). Compared with the plain FedAvg, FedAvg combined with cluster (blue) or importance sampling (purple) can accelerate the convergence speed (mea-
sured by the required rounds to achieve the target accuracy) by 2.6× and 2.1× times. The experimental results confirm the important role of the two in improving learning efficiency. Impressively, compared with importance sampling, the improvement of the clustering process is more obvious. This is because clustering can make better use of the correlation among clients, thereby selecting more representative clients to participate in model training. We also compare FedAvg combined with cluster, FedAvg combined with cluster and re-allocation, FedAvg combined with cluster and importance sampling, and HCSFed. All the latter methods can achieve a faster convergence in model training than the former, which validates the effectiveness of re-allocation and extra importance sampling. We also make a comparison with the result of SCAFFOLD after convergence, which further demonstrates that the effect of HCSFed is convincing. These results indicate that each component of HCSFed is complementary to each other, and also show the correctness of the variance reduction theory.

![Graph](image)

(a) The test accuracy of individual components

![Graph](image)

(b) The impact of individual components

Figure 7: Visualization of Ablation Study Results.

6 CONCLUDING REMARKS

In this paper, we propose HCSFed, a novel clustering-based selection scheme to accelerate the training convergence by variance reduction. Theoretically, we demonstrate the improvement of the proposed scheme in variance reduction. We present the convergence guarantee of HCSFed under the convex assumption. Experimental results demonstrate the superiority and the effectiveness of our method with both the convex and non-convex models. In the future, the co-variance among different clients and unavailable client settings will be considered, since it may also lead to performance degradation in FL. In the theoretical aspect, following the latest convergence analysis [Khaled et al., 2020], we plan to loose the widely used but strict assumption “bounded gradient” and convexity for more general analysis of the faster convergence achieved by HCSFed.

References

Ravikumar Balakrishnan, Tian Li, Tianyi Zhou, Nageen Himayat, Virginia Smith, and Jeff Bilmes. Diverse client selection for federated learning via submodular maximization. In International Conference on Learning Representations, 2022.

Tianyi Chen, Georgios Giannakis, Tao Sun, and Wotao Yin. Lag: Lazily aggregated gradient for communication-efficient distributed learning. Advances in Neural Information Processing Systems, 2018.

Wenlin Chen, Samuel Horváth, and Peter Richtárik. Optimal client sampling for federated learning. Transactions on Machine Learning Research, 2022. ISSN 2835-8856. URL https://openreview.net/forum?id=8GvRCWKHI4

Yae Jee Cho, Jianyu Wang, and Gauri Joshi. Client selection in federated learning: Convergence analysis and power-of-choice selection strategies. arXiv preprint arXiv:2010.02143, 2020.

Aymeric Dieuleveut, Gersende Fort, Eric Moulines, and Geneviève Robin. Federated expectation maximization with heterogeneity mitigation and variance reduction. arXiv preprint arXiv:2111.02083, 2021.

Yann Fraboni, Richard Vidal, Laetitia Kameni, and Marco Lorenzi. Clustered sampling: Low-variance and improved representativity for clients selection in federated learning. In International Conference on Machine Learning, 2021.

Avishek Ghosh, Jichan Chung, Dong Yin, and Kannan Ramchandran. An efficient framework for clustered federated learning. Advances in Neural Information Processing Systems, 33:19586–19597, 2020.

Eduard Gorbunov, Konstantin P Burlachenko, Zhize Li, and Peter Richtárik. Marina: Faster non-convex distributed learning with compression. In International Conference on Machine Learning, pages 3788–3798. PMLR, 2021.

Farzin Haddadpour and Mehrdad Mahdavi. On the convergence of local descent methods in federated learning. arXiv preprint arXiv:1910.14425, 2019.

Farzin Haddadpour, Mohammad Mahdi Kamani, Mehrdad Mahdavi, and Viveck Cadambe. Local sgd with periodic averaging: Tighter analysis and adaptive synchronization. Advances in Neural Information Processing Systems, 2019.
Song Han, Huizi Mao, and William J Dally. Deep compression: Compressing deep neural networks with pruning, trained quantization and huffman coding. *arXiv preprint arXiv:1510.00149*, 2015.

Peter Kairouz, H Brendan McMahan, Brendan Avent, Aurélien Bellet, Mehdi Bennis, Arjun Nitin Bhagoji, Kallista Bonawitz, Zachary Charles, Graham Cormode, Rachel Cummings, et al. Advances and open problems in federated learning. *Foundations and Trends in Machine Learning*, 14(1–2):1–210, 2021.

Sai Praneeth Karimireddy, Martin Jaggi, Satyen Kale, Mehryar Mohri, Sashank J Reddi, Sebastian U Stich, and Ananda Theertha Suresh. Mime: Mimicking centralized stochastic algorithms in federated learning. *arXiv preprint arXiv:2008.03606*, 2020a.

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*, 2020b.

Angelos Katharopoulos and François Fleuret. Not all samples are created equal: Deep learning with importance sampling. In *International Conference on Machine Learning*, 2018.

Ahmed Khaled, Konstantin Mishchenko, and Peter Richtárik. Tighter theory for local sgd on identical and heterogeneous data. In *International Conference on Artificial Intelligence and Statistics*, 2020.

Jakub Konečný, Brendan McMahan, and Daniel Ramage. Federated optimization: Distributed optimization beyond the datacenter. *arXiv preprint arXiv:1511.03575*, 2015.

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

Yann LeCun. The mnist database of handwritten digits. *http://yann.lecun.com/exdb/mnist/*, 1998.

Tian Li, Anit Kumar Sahu, Manzil Zaheer, Maziar Sanjabi, Ameet Talwalkar, and Virginia Smith. Federated optimization in heterogeneous networks. *Proceedings of Machine Learning and Systems*, 2020.

Xiang Li, Kaixuan Huang, Wenhao Yang, Shusen Wang, and Zhihua Zhang. On the convergence of fedavg on non-iid data. In *International Conference on Learning Representations*, 2019.

Brendan McMahan, Eider Moore, Daniel Ramage, Seth Hampson, and Blaise Agura y Arcas. Communication-efficient learning of deep networks from decentralized data. In *International Conference on Artificial Intelligence and Statistics*, 2017.

Ihab Mohammed, Shadha Tabatabai, Ala Al-Fuqaha, Faissal El Bouanani, Junaid Qadir, Basheer Qolomany, and Mohsen Guizani. Budgeted online selection of candidate iot clients to participate in federated learning. *IEEE Internet of Things Journal*, 2020.

Khalil Muhammad, Qinlin Wang, Diarmuid O’Reilly-Morgan, Elias Tragos, Barry Smyth, Neil Hurley, James Geraci, and Aonghus Lawlor. Fedfast: Going beyond average for faster training of federated recommender systems. In *International Conference on Knowledge Discovery & Data Mining*, 2020.

Elsa Rizk, Stefan Vlaski, and Ali H Sayed. Optimal importance sampling for federated learning. In *IEEE International Conference on Acoustics, Speech and Signal Processing*, 2021.

Navjot Singh, Deepesh Data, Jemin George, and Suhas Diggavi. Sparq-sgd: Event-triggered and compressed communication in decentralized stochastic optimization. *arXiv preprint arXiv:1910.14280*, 2019.

Sebastian U Stich. Local sgd converges fast and communicates little. In *International Conference on Learning Representations*, 2018.

Jianyu Wang and Gauri Joshi. Cooperative sgd: A unified framework for the design and analysis of local-update sgd algorithms. *Journal of Machine Learning Research*, 22:1–50, 2021.

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

Peilin Zhao and Tong Zhang. Stochastic optimization with importance sampling for regularized loss minimization. In *International Conference on Machine Learning*, 2015.

Fan Zhou and Guojing Cong. On the convergence properties of a k-step averaging stochastic gradient descent algorithm for nonconvex optimization. *arXiv preprint arXiv:1708.01012*, 2017.