A. Proof of Theorem 2

Theoretical guarantees regarding the convergence of FedAvg were given in (Wang et al., 2020). The proof relies on Assumptions 1 to 3. The full proof is provided in (Wang et al., 2020) for MD sampling where the MD sampling is shown to satisfy Lemma 5. In Section A.1, we reproduce the proof provided in (Wang et al., 2020) for Lemma 5 and, in Section A.2, we show that clustered sampling satisfying Proposition 1 also satisfies Lemma 5. As a result, FedAvg when sampling clients with MD or clustered sampling has identical asymptotic behavior.

Lemma 5. *Suppose we are given $z_1, z_2, ..., z_n, x \in \mathbb{R}^d$. Let $l_1, l_2, ..., l_m$ be the index of the sampled clients and S be the set of sampled clients. We have*

$$\mathbb{E}_S \left[\frac{1}{m} \sum_{j=1}^{m} z_{l_j} \right] = \sum_{i=1}^{n} p_i z_i, \tag{24}$$

and

$$\mathbb{E}_S \left[\left\| \frac{1}{m} \sum_{j=1}^{m} z_{l_j} \right\|^2 \right] \leq 3 \sum_{i=1}^{n} p_i \left\| z_i - \nabla \mathcal{L}_i(x) \right\|^2 + 3 \left\| \nabla \mathcal{L}(x) \right\|^2 + \frac{3}{m} (\beta^2 \left\| \nabla \mathcal{L}(x) \right\|^2 + \kappa^2). \tag{25}$$

A.1. Proof of Lemma 5 for Theorem 1 adapted from (Wang et al., 2020)

Proof. Clients are selected with MD sampling. We denote by $l_1, l_2, ..., l_m$ the m indices of the sampled clients which are iid sampled from a multinomial distribution supported on $\{1, ..., n\}$ satisfying $\mathbb{P}(l_x = i) = p_i$ and $\sum_{i=1}^{n} p_i = 1$.

By definition, MD sampling satisfies equation (24).

Regarding equation (25), we have:

$$\frac{1}{m} \sum_{j=1}^{m} z_{l_j} = \left(\frac{1}{m} \sum_{j=1}^{m} z_{l_j} - \frac{1}{m} \sum_{j=1}^{m} \nabla \mathcal{L}_{l_j}(x) \right) + \left(\frac{1}{m} \sum_{j=1}^{m} \mathcal{L}_{l_j}(x) - \nabla \mathcal{L}(x) \right) + \nabla \mathcal{L}(x). \tag{26}$$

Using the Jensen inequality on the $\left\| \cdot \right\|^2$ operator, we get:

$$\mathbb{E} \left[\left\| \frac{1}{m} \sum_{j=1}^{m} z_{l_j} \right\|^2 \right] \leq 3 \, \mathbb{E} \left[\left\| \frac{1}{m} \sum_{j=1}^{m} (z_{l_j} - \nabla \mathcal{L}_{l_j}(x)) \right\|^2 \right] + 3 \, \mathbb{E} \left[\left\| \frac{1}{m} \sum_{j=1}^{m} \nabla \mathcal{L}_{l_j}(x) - \nabla \mathcal{L}(x) \right\|^2 \right] + 3 \left\| \nabla \mathcal{L}(x) \right\|^2 \tag{27}$$

Using the Jensen inequality, we get the following upper bound for the first term:

$$\mathbb{E} \left[\left\| \frac{1}{m} \sum_{j=1}^{m} (z_{l_j} - \nabla \mathcal{L}_{l_j}(x)) \right\|^2 \right] \leq \mathbb{E} \left[\frac{1}{m} \sum_{j=1}^{m} \left\| z_{l_j} - \nabla \mathcal{L}_{l_j}(x) \right\|^2 \right] \tag{28}$$

$$= \sum_{i=1}^{n} p_i \left\| z_i - \nabla \mathcal{L}_i(x) \right\|^2, \tag{29}$$

where the equality follows from equation (24).

By definition, MD sampling is unbiased, i.e. $\mathbb{E}\left[\nabla \mathcal{L}_{l_j}(x) \right] = \nabla \mathcal{L}(x)$. Therefore, we get the following upper bound for the second term:

$$\mathbb{E} \left[\left\| \frac{1}{m} \sum_{j=1}^{m} \nabla \mathcal{L}_{l_j}(x) - \nabla \mathcal{L}(x) \right\|^2 \right] = \mathbb{E} \left[\frac{1}{m^2} \sum_{j=1}^{m} \left\| \nabla \mathcal{L}_{l_j}(x) - \nabla \mathcal{L}(x) \right\|^2 \right] \tag{30}$$

$$= \frac{1}{m} \sum_{i=1}^{n} p_i \| \nabla \mathcal{L}_i(x) - \nabla \mathcal{L}(x) \|^2 \tag{31}$$

$$= \frac{1}{m} \sum_{i=1}^{n} p_i \left\| \nabla \mathcal{L}_i(x) \right\|^2 - \frac{1}{m} \left\| \nabla \mathcal{L}(x) \right\|^2 \tag{32}$$

$$\leq \frac{1}{m} [(\beta^2 - 1) \left\| \nabla \mathcal{L}(x) \right\|^2 + \kappa^2] \tag{33}$$

$$\leq \frac{1}{m} [\beta^2 \left\| \nabla \mathcal{L}(x) \right\|^2 + \kappa^2], \tag{34}$$

where the first inequality comes from using Assumption 3.

Finally, substituting equation (29) and (34) in equation (27) completes the proof.

□

A.2. Proof of Lemma 5 for Theorem 2

Proof. Clients are selected with clustered sampling. The m clients indices $l_1, l_2, ..., l_m$ are still independently sampled but no longer identically. Each index l_k is sampled from a distribution W_k. Each client can be sampled with probability $\mathbb{P}(l_k = i) = r_{k,i}$.

Clustered sampling follows Proposition 1 and therefore satisfies equation (24).

Equation (27) holds for any sampling schemes. Therefore, we also use it to prove equation (25) for clustered sampling. Using the same steps as for the proof of Lemma 5 for MD

sampling, we bound the first term of equation (27) as:

$$\mathbb{E}\left[\left\|\frac{1}{m}\sum_{j=1}^{m}(z_{l_j}-\nabla\mathcal{L}_{l_j}(x))\right\|^2\right] \leq \sum_{i=1}^{n} p_i \left\|z_i-\nabla\mathcal{L}_i(x)\right\|^2. \tag{35}$$

Before bounding the second term, we define $\nabla\mathcal{L}_{W_k}(x)$ as the expected gradient of the distribution W_k with respects to the parameters x, i.e.

$$\nabla\mathcal{L}_{W_k}(x) \coloneqq \mathbb{E}_{l_k\sim W_k}\left[\nabla\mathcal{L}_{l_k}(x)\right] = \sum_{i=1}^{n} r_{k,i}\nabla\mathcal{L}_i(x) \tag{36}$$

Using this definition, we bound the second term as

$$\begin{aligned}
&\mathbb{E}\left[\left\|\frac{1}{m}\sum_{k=1}^{m}\nabla\mathcal{L}_{l_k}(x)-\nabla\mathcal{L}(x)\right\|^2\right] \\
&= \mathbb{E}\left[\left\|\frac{1}{m}\sum_{k=1}^{m}(\nabla\mathcal{L}_{l_k}(x)-\nabla\mathcal{L}_{W_k}(x))\right\|^2\right] && (37)\\
&= \frac{1}{m^2}\sum_{k=1}^{m}\mathbb{E}\left[\left\|\nabla\mathcal{L}_{l_k}(x)-\nabla\mathcal{L}_{W_k}(x)\right\|^2\right] && (38)\\
&= \frac{1}{m^2}\sum_{k=1}^{m}\sum_{i=1}^{n} r_{k,i}\left\|\nabla\mathcal{L}_i(x)-\nabla\mathcal{L}_{W_k}(x)\right\|^2 && (39)\\
&= \frac{1}{m^2}[\sum_{i=1}^{n} mp_i\left\|\nabla\mathcal{L}_i(x)\right\|^2-\sum_{k=1}^{m}\left\|\nabla\mathcal{L}_{W_k}(x)\right\|^2] && (40)\\
&\leq \frac{1}{m}[\beta^2\left\|\nabla\mathcal{L}(x)\right\|^2+\kappa^2], && (41)
\end{aligned}$$

where the last inequality comes from using Assumption 3 and equation (38) and (40) are obtained with equation (36).

Finally, substituting equation (35) and (41) in equation (27) completes the proof.

□

Equation (32) and (40) allow us to theoretically identify the convergence improvement of clustered sampling over MD sampling.

We define by $B_{MD} = \frac{1}{m}\sum_{i=1}^{n} p_i\left\|\nabla\mathcal{L}_i(x)\right\|^2 - \frac{1}{m}\left\|\nabla\mathcal{L}(x)\right\|^2$, equation (32), and $B_{Cl} = \frac{1}{m}\sum_{i=1}^{n} p_i\left\|\nabla\mathcal{L}_i(x)\right\|^2 - \frac{1}{m^2}\sum_{k=1}^{m}\left\|\nabla\mathcal{L}_{W_k}(x)\right\|^2$, equation (40). Using the Jensen inequality, we get

$$-\sum_{k=1}^{m}\frac{1}{m^2}\left\|\nabla\mathcal{L}_{W_k}(x)\right\|^2 \leq -\frac{1}{m}\left\|\sum_{k=1}^{m}\frac{1}{m}\nabla\mathcal{L}_{W_k}(x)\right\|^2 \tag{42}$$

$$= -\frac{1}{m}\left\|\nabla\mathcal{L}(x)\right\|^2 \tag{43}$$

with equality if and only if $\forall k,l,\ \nabla\mathcal{L}_{W_k}(x) = \nabla\mathcal{L}_{W_l}(x)$. Thus, $B_{Cl} \leq B_{MD}$ with equality if and only if all the clients have the same data distribution or the considered clustered sampling is MD sampling.

B. MD and Clustered Sampling Comparison

B.1. Client aggregation weight variance

As in Section 3, we denote by S_{MD} and $S_C(t)$ the random variables associated respectively to MD and clustered sampling. Also in Section 3, we have shown that

$$\mathrm{Var}_{S_{MD}}\left[\omega_i(S_{MD})\right] = \frac{1}{m^2}mp_i(1-p_i), \tag{44}$$

and

$$\mathrm{Var}_{S_C(t)}\left[\omega_i(S_C(t)\right] = \frac{1}{m^2}\sum_{k=1}^{m} r_{k,i}^t(1-r_{k,i}^t). \tag{45}$$

Hence, we get:

$$\begin{aligned}
&\mathrm{Var}_{S_{MD}}\left[\omega_i(S_{MD})\right] - \mathrm{Var}_{S_C(t)}\left[\omega_i(S_C(t)\right] && (46)\\
&= \frac{1}{m^2}[mp_i(1-p_i) - \sum_{k=1}^{m} r_{k,i}^t(1-r_{k,i}^t)] && (47)
\end{aligned}$$

We consider an unbiased clustered sampling. Therefore, the sum of probability for client i over the m clusters satisfies $\sum_{k=1}^{m} r_{k,i}^t = mp_i$ giving:

$$\begin{aligned}
&\mathrm{Var}_{S_{MD}}\left[\omega_i(S_{MD})\right] - \mathrm{Var}_{S_C(t)}\left[\omega_i(S_C(t)\right] && (48)\\
&= \frac{1}{m^2}[\sum_{k=1}^{m} {r_{k,i}^t}^2 - mp_i^2] && (49)
\end{aligned}$$

Using the Cauchy-Schwartz inequality, we get: $\sum_{k=1}^{m} {r_{k,i}^t}^2 \times \sum_{k=1}^{m} 1^2 \geq \left(\sum_{k=1}^{m} r_{k,i}^t \times 1\right)^2 = (mp_i)^2$ due to the unbiased aspect of the considered clustered sampling. As such, we get:

$$\mathrm{Var}_{S_{MD}}\left[\omega_i(S_{MD})\right] - \mathrm{Var}_{S_C(t)}\left[\omega_i(S_C(t)\right] \geq 0, \tag{50}$$

with equality if and only if $r_{k,i}^t = p_i$.

B.2. Probability for a client to be sampled at least once

In Section 3, we have shown that

$$p(\{i \in S_{MD}\}) = 1-(1-p_i)^m \tag{51}$$

and

$$p(\{i \in S_C(t)\}) = 1 - \prod_{k=1}^{m}(1 - r_{k,i}^t). \quad (52)$$

Hence, we get:

$$p(\{i \in S_{MD}\}) - p(\{i \in S_C(t)\}) \quad (53)$$

$$= \prod_{k=1}^{m}(1 - r_{k,i}^t) - (1 - p_i)^m \quad (54)$$

We consider an unbiased clustered sampling. Therefore, when using the inequality of arithmetic and geometric means, we get:

$$\prod_{k=1}^{m}(1-r_{k,i}^t) \leq \left(\frac{\sum_{k=1}^{m}(1 - r_{k,i}^t)}{m}\right)^m = (1-p_i)^m, \quad (55)$$

with equality if and only if $r_{k,i}^t = p_i$. Finally, we get:

$$p(\{i \in S_{MD}\}) \geq p(\{i \in S_C(t)\}) \quad (56)$$

C. Explaining Algorithm 1 and 2

Algorithms 1 and 2 can be written in term of data ratio p_i instead of samples number n_i. While in both cases the algorithms would be correct, it turns out to be simpler to work with quantities of samples $n_i = p_i M$ instead which are integers. Therefore, without loss of generality, we denote by $r'_{k,i}$ the number of samples allocated by client i to distribution k. We retrieve the sampling probability of client i in distribution W_k with $r_{k,i} = \frac{r'_{k,i}}{M}$.

Also, without loss of generality, we prove Algorithms 1 and 2 at iteration t and therefore we use in the proofs $r_{k,i}$ and W_k instead of $r_{k,i}^t$ and W_k^t.

C.1. Algorithm 1

We illustrate in Figure 3 the clients allocation scheme of Algorithm 1 introduced in Section 4, by considering how a client i is associated to the m distributions. Theorem 3 states that Algorithm 1 provides a sampling scheme satisfying Proposition 1 with complexity $\mathcal{O}(n\log(n))$ which we prove in Section 4 and in the following proof.

Proof. In term of complexity, the while loop for the client allocation, as illustrated in Figure 3, either change client or distribution at every step and is thus done in complexity $\mathcal{O}(n+m)$. Sampling client is relevant if $m < n$. Therefore the allocation complexity is equivalent to $\mathcal{O}(n+m) = \mathcal{O}(n)$. Also, sorting n elements is done in complexity $\mathcal{O}(n\log(n))$. Therefore, Algorithm 1 overall complexity is $\mathcal{O}(n\log(n)$. □

C.2. Algorithm 2

We illustrate in Figure 4, the clients allocation scheme of Algorithm 2 introduced in Section 5 by considering how a client i is associated to the m distributions. Theorem 4 states that Algorithm 2 provides a sampling scheme satisfying Proposition 1 and takes time complexity $\mathcal{O}(n^2d + X)$. We prove these statements in Section 5 and the following proof.

Proof. With identical reasoning as for Algorithm 1, clients are allocated in complexity $\mathcal{O}(n)$. Computing the similarity between two clients requires d elementary operations, where d is the number of parameters in the model, and has thus complexity $\mathcal{O}(d)$. Computing the similarity matrix requires computing $\frac{n(n-1)}{2}$ client similarities and thus has total complexity $\mathcal{O}(n^2d)$. Computing the similarity tree depends on the *clustering method* which we consider has complexity $\mathcal{O}(X)$. Transforming the tree as discussed in Section 5 requires going through its $n-1$ nodes and thus has time complexity $\mathcal{O}(n)$. Cutting the tree requires considering at most every nodes and has thus complexity $\mathcal{O}(n)$. Lastly, the tree is cut in at most n branches and sorting them takes therefore complexity $\mathcal{O}(n\log(n)$. Finally, combining all these time complexities gives for Algorithm 2 a time complexity of $\mathcal{O}(n^2d + X)$.

□

In practice, the m distributions are computed at every iteration, while the server is required to compute the similarity between sampled clients and all the other clients. Therefore the similarity matrix can be estimated in complexity $\mathcal{O}(nmd)$, and Algorithm 2 has complexity $\mathcal{O}(nmd + X)$.

D. Additional Experiments

We describe in Section 6 the different datasets used for the experiments and how we use the Dirichlet distribution to partition CIFAR10 in realistic heterogeneous federated datasets. In all the experiments, we consider a batch size of 50. For every CIFAR10 dataset partition, the learning rate is selected in $\{0.001, 0.005, 0.01, 0.05, 0.1\}$ to minimize FedAvg with MD sampling training loss.

D.1. CIFAR10 partitioning illustration

In Figure 5, we show the influence of α on the resulting federated dataset heterogeneity. $\alpha = 10$ provides almost an iid dataset and identical class percentages, column (a) , and same number of samples per class, column (b). With $\alpha = 0.001$, we get a very heterogeneous dataset with almost only one class per client translating into some classes much more represented than others due to the unbalanced nature aspect of the created federated dataset, cf Section 6.

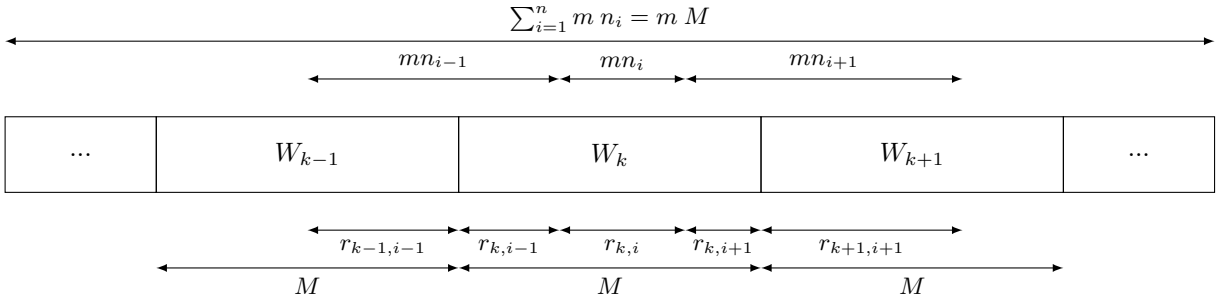

Figure 3. Illustration of the clients allocation scheme of Algorithm 1. Clients are considered in decreasing importance of their number of samples and always allocate client samples to distributions that already received samples but do not yet have M of them. As a result, after allocating a client, all distributions except at most one have 0 or M samples. Client i is only sampled in W_k because every distribution with index inferior to k are filled with clients of index inferior to i, and because there is enough room in W_k to receive all the samples that need to be allocated for client i.

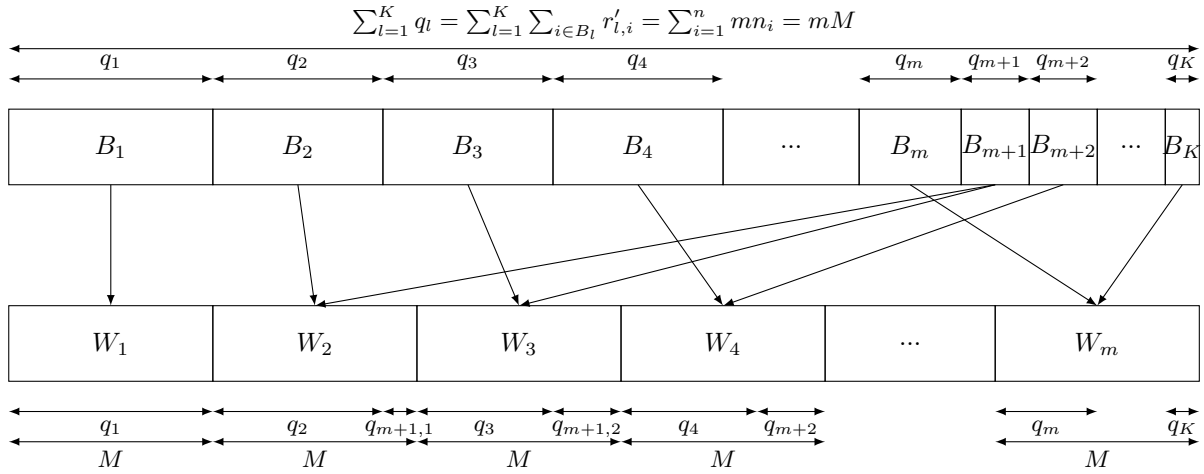

Figure 4. Illustration of the clients allocation scheme of Algorithm 2. After the tree is split in K groups of clients, the groups are ordered and we consider without loss of generality that their number of samples are inversely proportional to their index. With Algorithm 2, the first m groups, i.e. B_1 to B_m, are each associated to one distribution, i.e. W_1 to W_m. The remaining groups are considered one after the other and split among the remaining slots in the groups. Each distribution has M samples from clients participating to the FL process.

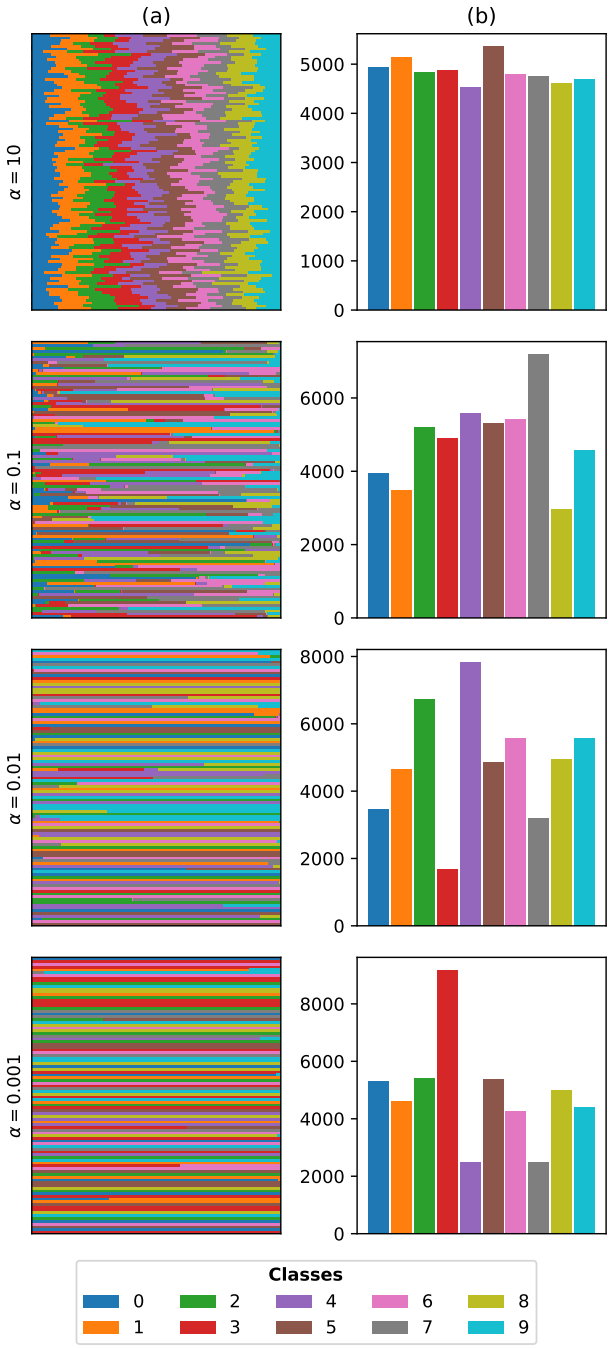

Figure 5. Effect of α on the resulting clients partitioning when using a Dirichlet distribution. Plots in column (a) represent the percentage of each class owned by the clients. Plots in column (b) give for every class its total number of samples across clients. We consider in this work $\alpha \in \{0.001, 0.01, 0.1, 10\}$.

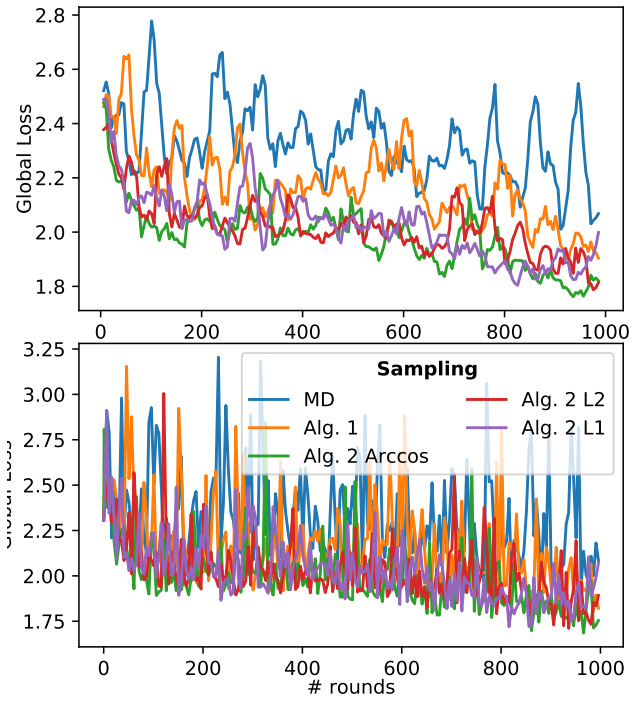

Figure 6. Effect of the similarity measure chosen for Algorithm 2 on the training loss convergence. We consider the evolution of the global loss, equation (1), in function of the server iteration t. For clarity concerns, we plot the global loss obtained with rolling mean over 50 server iterations (top) and the raw global loss (bottom). We consider CIFAR partitioned with Dir($\alpha = 0.01$), learning rate $lr = 0.05$, $N = 100$ SGD, and $m = 10$ sampled clients.

D.2. Influence of the similarity measure

Figure 6 shows the effect similarity measures (Arccos, L2, and L1) have on training global loss convergence. We retrieve that Algorithm 1 outperforms MD sampling by reducing clients aggregation weight variance. We remind that the hierarchical tree is obtained using Ward's method in this work. We notice that the tree similarity measures gives similar performances when using Algorithm 2 with Ward hierarchical clustering method.. This justifies the use of Arccos similarity for the other experiments.

D.3. More details on Figure 2

For sake of clarity, we note that the training loss displayed in Figures 2 is computed as the rolling mean over 50 iterations. In Figure 7, we provide the raw training global loss with the testing accuracy at every server iteration.

D.4. Influence of m the number of sampled clients, and N the number of SGD run

We also investigates the influence the number of sampled clients m and the number of SGD run N have on the FL

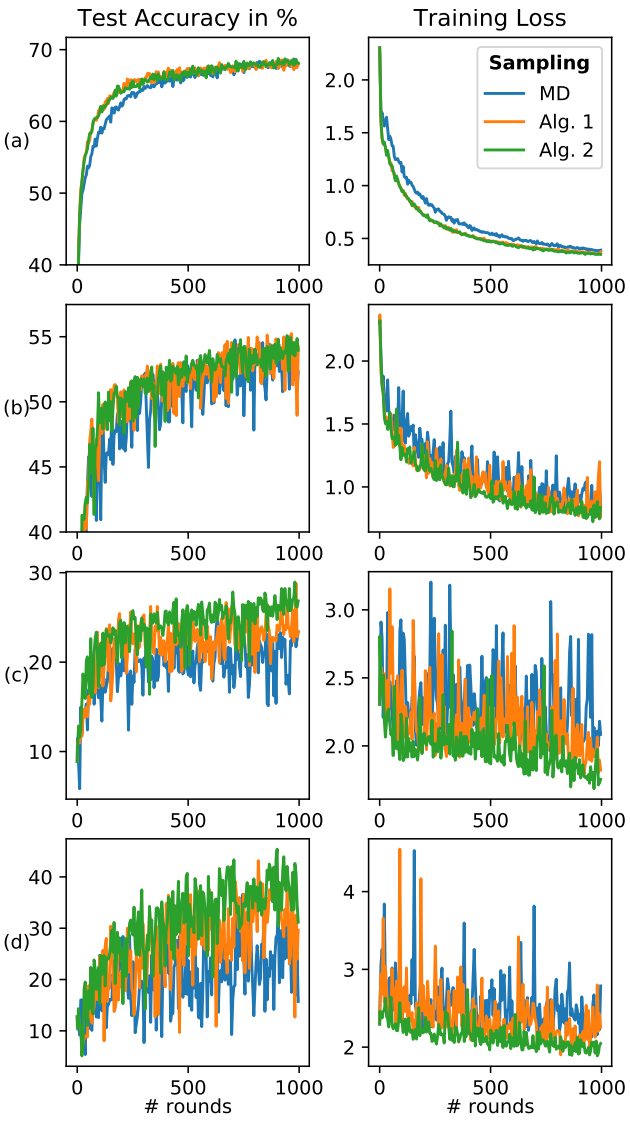

Figure 7. We investigate the improvement provided by clustered sampling on federated unbalanced datasets partitioned from CIFAR10 using a Dirichlet distribution with parameter $\alpha \in \{0.001, 0.01, 0.1, 10\}$ for respective row (a), (b), (c), (d). We use $N = 100$, $m = 10$, and respective learning rate for each dataset $lr = \{0.05, 0.05, 0.05, 0.1\}$.

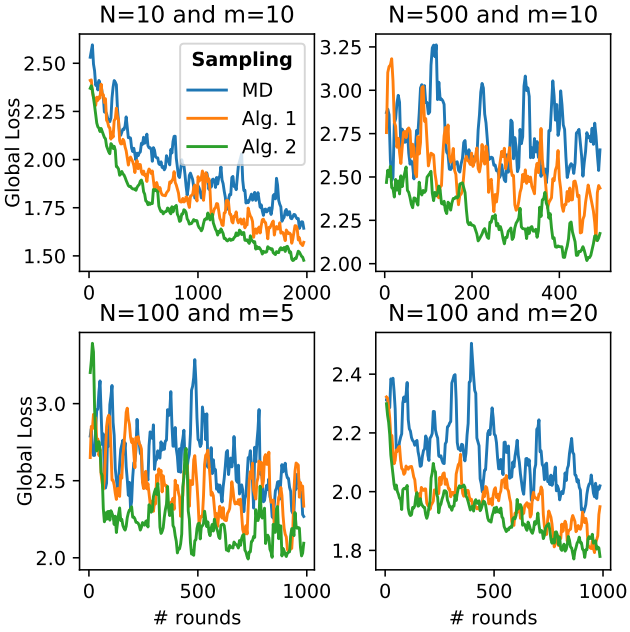

Figure 8. We consider the federated dataset partitioned from CIFAR10 using a Dirichlet distribution with parameter $\alpha = 0.01$. We investigate the influence of N, the number of SGD run by each client, and m, the number of sampled clients, on the training loss convergence. For each plot, experiments in first row use respectively $lr = \{0.1, 0.05\}$ and for second row $lr = \{0.05, 0.05\}$.

convergence speed and smoothness in Figure 8. We notice that the more important the amount of local work N is, and the faster clustered sampling convergence speed is. With more local work, clients better fit their data. In non-iid dataset this translates in more forgetting on the classes and samples which are not part of the sampled clients. Regarding the amount of sampled clients m, we notice that with a smaller amount of sampled clients the improvement of clustered sampling over MD sampling is more important. We associate this result to the better data representativity of clustered sampling. For the same reason, when we increase the number of sampled clients, we see faster convergence for both MD and clustered sampling. The performance of clustered sampling is closer but still better than the one of MD sampling.

For sake of clarity, we note that the training loss displayed in Figures 8 is computed as the rolling mean over 50 iterations. In Figure 7, we provide the raw training global loss with the testing accuracy at every server iteration.

D.5. Local regularization

With FedProx (Li et al., 2018), every client's local loss function is equipped with a regularization term forcing the updated model to stay close to the current global model, i.e.

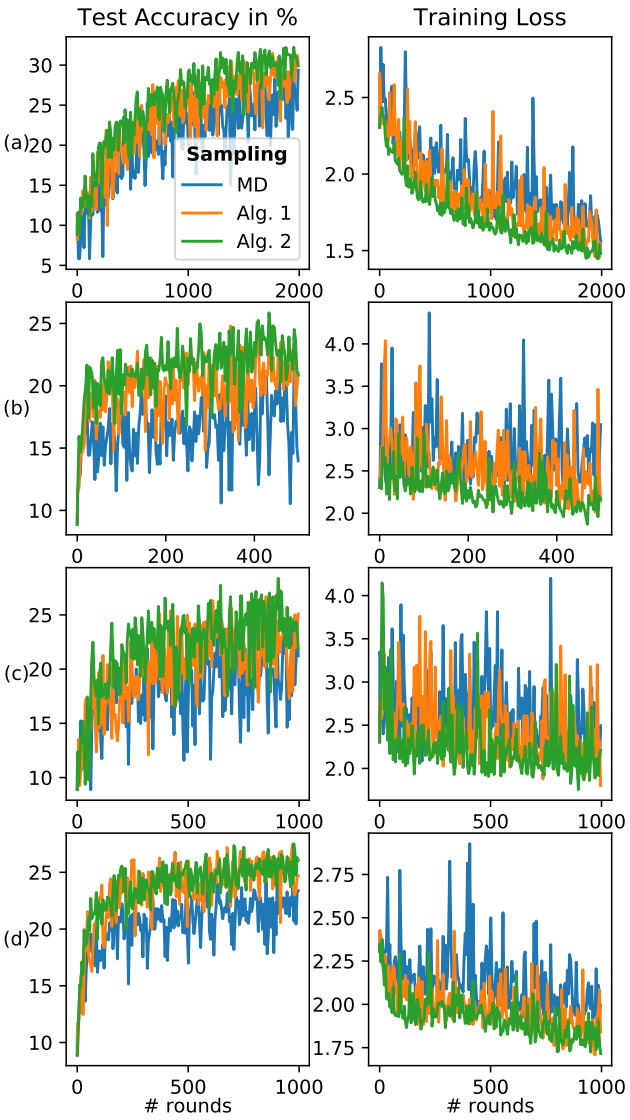

Figure 9. We consider the federated dataset partitioned from CIFAR10 using a Dirichlet distribution with parameter $\alpha = 0.01$. We investigate the influence of N the number of SGD run by each client in the first two rows with $N = 10$ and $N = 500$ for $m = 10$ and the influence of sampled clients with $m = 5$ and $m = 20$ for $N = 100$ in the last two rows. For each dataset, we use respective learning rate $lr = \{0.1, 0.05, 0.05, 0.05\}$.

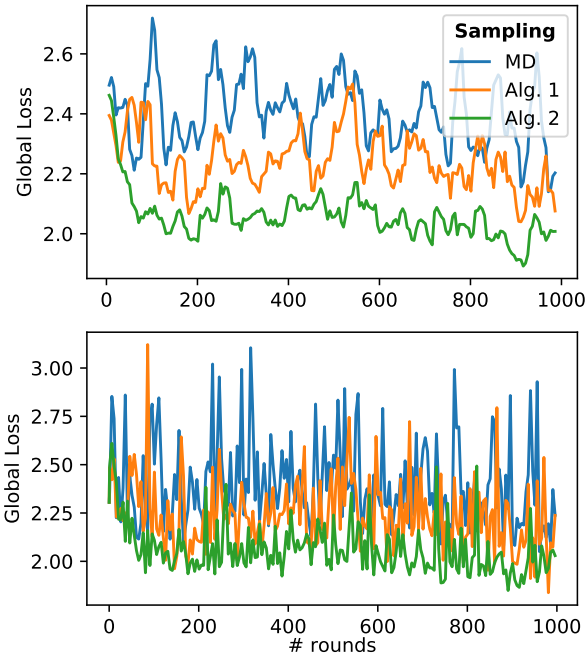

Figure 10. Training loss convergence for FL with FedProx local loss function regularization ($\mu = 0.1$). We consider CIFAR10 partitioned with Dir($\alpha = 0.01$), learning rate $lr = 0.05$, $m = 10$ sampled clients, and $N = 100$ SGD.

$$\mathcal{L}_i'(\theta_i^{t+1}) = \mathcal{L}_i(\theta_i^{t+1}) + \frac{\mu}{2}\left\|\theta_i^{t+1} - \theta^t\right\|^2 \qquad (57)$$

where θ_{t+1} is the updated local model of client i and θ^t is the current global model. μ is the hyperparameter monitoring the regularization and is common for all the clients. This framework enables smoother federated learning processes.

We try a range of regularization term $\mu \in \{0.001, 0.01, 0.1, 1.\}$ and keep $\mu = 0.1$ maximizing the performances of FedAvg with regularization and MD sampling. We notice in Figure 10 that Algorithm 1 and 2 still outperform MD sampling.