FedDiv

...

***Abstract*—Federated learning is a promising bridge which connect machine learning and medicine. There are many aggregation strategies for federated learning now especially personalized methods which show relatively excellent performance. But most of them excessive pay attention to accuracy while ignore specificity and sensitiveness which are used in medicine more generally. We propose a method to strike a balance between these metric mentioned before. It is basing on divergence among federated clients' models and we name it as FedDiv. In total, this strategy aligns encoders or extractors and makes personalized decoder or classifier by divergence. In addition, we design a new structure which assembles several sub-encoder as one encoder and each sub-encoder focuses on one category data to weaken the influence to minority caused by majority. In this paper, we validate this strategy in three medical datasets from public and handle these data as different datasets with different proportion of labels and modals. The experiment shows that our method perform excellent over others in different distribute data.**

***Index Terms*—federated learning, personalized, features align**

I. Introduction

With the development of artificial intelligence, deep learning algorithms are receiving increasing attention in healthcare. Meanwhile, deep learning is suitable for efficiently handling massive high-quality medical data accumulated by hospitals [1]. One of the bridge of hospitals' data and deep learning is federated learning which enable machines to learn generalize or personalized features in data from different hospitals confidentially [2]. Although people believe that characteristics obtained from different hospitals should be more general than from single one, non-IID data limit the occurrence of this assumption [3]. The demography characteristics in data vary according to the geographical location of the hospital. The quality and quantity of cases are also different when we compare the data between comprehensive hospital and specialized hospital. If we ignore these differences during federated learning, the outcome maybe disappointing.

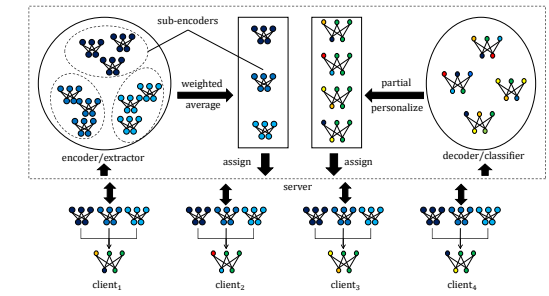
Many people in different filed have provided solutions to these problems and they achieved relatively ideal results in terms of accuracy [4]. The most popular strategy to overcome the problem that decreases aggregated models' accuracy caused by non-IID data is personalizing [5]. What hospitals taking part in federated learning do is like that way. Hospitals must adhere to principles, but they do not completely obey it during treating and preserve their own subtle differences. This is understandable because many of them are regional and keep their habits. Thus, all federated clients get an identical model maybe not an excellent strategy in medicine [6]. However, it is not enough just taking personalizing aggregating strategy in medical field where people attache importance to sensitiveness and specificity too. There are several reason that personalizing aggregating strategy can not strike balanced between accuracy and sensitiveness or specificity. Firstly, some methods align features just by averaging which is not best way validated by our experiment such as FedAMP [7] and L2GD [8]. Secondly, strategies like FedRod [9] mainly consider about majorities that way causes aggregated models possess higher accuracy meanwhile its' specificity is relatively low. Finally, those take account models calibration doing not discriminate encoder and decoder similar to FedABC [10] which makes aggregated models across a dilemma between higher accuracy or specificity.

To strike a balance among metrics such as accuracy, sensitiveness and specificity, we proposes a universal method for improving model structure in federated learning and a new personalizing aggregation algorithm basing on **federated divergence** (FedDiv) applies to this structure. We first divide model as encoder/extractor and decoder/classifier. And then, model's encoder in each client is further divided into several sub-encoder performing single task. Each of them catches different feature with same input and has maximum influence in its' field being aggregated by sever. That is like an expert who have maximum influence in their major field. And we think that input in identical category data share same feature. Different encoder pays more attention to different category data favoring to extract different category feature and increasing specificity. The structure is displayed by sub-figure (b) in Figure 1.

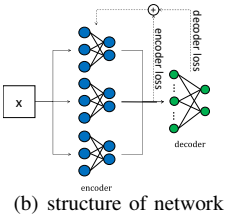
When comes to aggregating classifier's or decoder's parameters, we refer to making guidelines. When experts have inconsistent views during making guidelines, they would had reservation and did further research until they reach an agreement. Inspired by this, we argue that discrepancy among models decides the extent of aggregation in each communication round. While our aggregation strategy's main idea is generating the personalized classifier or decoder according to it's divergence between clients. We differentiate the classifiers' parameters as personalized and generic components. When some component of local classifiers' parameters are great different we consider it as personalized one which doesn't take part in aggregating. And we aggregate it in average if the other part of classifiers' parameters have small divergence. Sub-figure (c) in Figure 1 shows the aggregation region.

When identical category data shares shame features, the proportion of these features in whole data is related with the ratio of identical category data account on total. Thus, personalizing decoder or classifier will get higher accuracy or sensitiveness in non-IID data. But excessive personalizing decoder hinders the aggregated models learning global knowledge and features aligning. So, we combine encoder calibration and decoder personalizing to strike a balance between specificity and accuracy. In addition wo control the extent of personalizing decoders

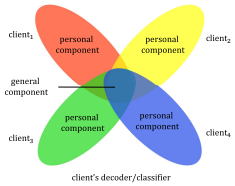
* ...



(a) aggregation of encoders and decoders



(b) structure of network



(c) decoders' components

Fig. 1. Each client shares one network structure which is consist of several sub-encoders and one decoder. The server aggregates sub-encoders' and partial decoders' parameters which are considered as general component from clients through weighted averaging. Personal component which are great different among client in decoders are maintained its' parameters doing not take part in aggregation. Sub-encoders align features through encoder loss prompting models' specificity. Decoder keep its' personal parameters to ensure models' accuracy in different distribution data.

through their parameters' divergence to ensure aggregating global knowledge. Sub-figure (a) in Figure 1 shows whole aggregation process.

Our contributions are as follow:

- We propose sub-encoder which major in extracting one category data feature. This structure aligns identical category features in different clients.
- We design an aggregating strategy suiting for sub-encoders' parameters. Each sub-encoder is assigned maximum weight in its' major field.
- Different from other personalized strategy which applied regularization to aggregate, FedDiv reserves part raw weight through divergence among clients. This way is efficiently and effectively that suits large-scale federated learning.
- We improve models' performance in sensitiveness and specificity by aligning sub-encoders and prompt their accuracy by personalizing decoder.

II. Related work

A. personalized federated learning

Traditionally, the global model is updated by using distributed private data in federated learning, and lots of optimization methods have been proposed. FedAvg [11] and FedProx [12] are representative strategies in generic federated learning. With the advanced study on federated learning, researchers encounter severe problem of non-IID data [**?**]. Nowadays, personalized federated learning has drawn much attention due to its' higher accuracy [4]. There are many personalized methods to aggregate models including regularization [**?**], clustering [**?**], feature aligning [**?**] and so on. Specifically, the regularization strategy's basic idea is to guide the local model aggregating direction by global information. Among them, FedAMP [7] and L2GD [8] learn personalized models **being lead by each other local models**. Similar to FedAMP, FPFC [13] combines clustering and regularization. In many strategies, the cost of computational resources will increase high with the participants mounting up, when introducing regularization to federated learning. That may limit these methods applied into federated learning. However, strategies based on clustering methods like IFCA [14] and FedBayes [15] would raise the communication cost. We think that the process of federated learning is dynamic and the extent of personalizing is decided by the guiding ability of global information. Thus, we assume that adjusting the models' extent of personalizing through divergence would reach similar effect at a low cost way.

Comparing with those existing methods, the proposed method gains following advantages. First, xxx. Second, xxxx.

B. personalized federated learning based on feature aligning

It is not enough that purely personalized aggregate models and most people merge model calibration into model personalization. Someone had chosen realistic or virtually balanced data to align models [16]–[18]. However, this data may not obey actual distribute or even violate the principle of federated learning. Others had used divergence calibrate the models but they didn't consider whole category consistency [19], [20]. Normalization in data is one way to align features being extracted by encoders but it still could not avoid distribution of data in various clients affect to model training [21]. FedRoD [9] suggested that the features can be aligned by balanced softmax loss and this strategy had performed excellently in metric of accuracy. But it ignores the minority while pay attention to majority which would make it in disadvantage when we compare sensitiveness and specificity. To ease this problem meanwhile calibrate models, we design sub-encoders which pay attention to one class data. The superiority includes xxxx.

Contrastive learning is another way to calibrate models. Among unsupervised or semi-supervised learning, it plays an important role. Weiming Zhuang et al. choose siamese network to execute that task [22]. And this network aims to find out the difference between different category meanwhile search for identical features among same group. However, without the guidance of label, the learning efficiency will be low and the model will perform below expectations. Felix X. Yu et al [23] propose similar ideas to train networks with only positive labels though they used distance instead of divergence to align features. FedIIC [24] considered the difference and similarities among classes, but it took so much hyper-parameters that the complexity was increased. Although FedABC [10] transformed multi-classify task to double-classify task, disproportion between positive and negative samples would hinder classifiers discriminate them. Different from these strategies, we calibrate extractors or encodes by limiting divergence for each category features to extract more generic features.

III. Preliminaries

In this section, we briefly introduce the generalizing or personalizing methods of federated learning. Generally, generic models are easily be transplanted to new clients but it's accuracy are normally lower than personal models. Personalizing federated learning usually get higher quality models but there are some problems like sensitiveness and specificity maintained to be solved [5].

A. Federated learning

The global loss function learned from several clients is defined as

$$\min_{W} L(W) := \sum_{k=1}^{K} \frac{L_k(W)}{K}.$$

Here, W denotes model's parameters; K represents the total number of clients. As the representative aggregation strategy, FedAvg iteratively update model between server and clients [11]. At server, it performs

$$W = \sum_{k=1}^{K} \frac{n_k}{n} W^k,$$

where W^k is obtained by performing

$$W^k = \underset{U}{\operatorname{argmin}}\, L^k(U).$$

Intuitively, federated learning aggregates every clients' knowledge by using distributed data. But it will be mislead by unbalance of data. Especially when some clients' data is non-IID, the generic federated learning strategy will be impeded [3].

B. Personalized federated learning

C. Tradeoff between personalized and generic federate learning

Someone argues that the data distribution varies in clients for their different attribute. Learning a global model may ignore clients' unique characteristic. Personalized federated learning suggests that we should train local model for each client from a global model. There are many methods to deal with it such as regularization, fine tuning and so on. These methods produce many excellent results. There are some shortages waiting for being overcomed in applications such as multi-center medical analysis.

- Some methods are trying to trade off non-IID data against IID data. When it executes well in non-IID data, it may perform poorly during dealing with IID data. While the medical industry cases includes both non-IID data and IID data.
- Many strategies pay attention to accuracy while overlook others metrics such as sensitiveness and specificity. Hospitals need different indexes in different scenes for example it requires high sensitiveness for screening experiment while making a definite diagnosis must be high specificity.
- Lots of algorithms could not solve the problem that disadvantage cases are covered by advantage cases. If the number of one category instances far more than another, the advantage category data are possibly covering whole networks causing the disadvantage category data learning insufficiently.

To overcome the problems mentioned above, we propose FedDiv. The strategy aggregates encoder and decoder in different way. We divide encoder into many small parts and aggregate decoder according to the data divergence.

IV. FedDiv

In deep learning, the network usually is consist of encoder/extractor and decoder/classifier. And in this section we call it encoder and decoder. While, single encoder may be mislead by unbalanced data easier than multi-encoder. We can train personal encoder for each local data but it is unnecessary. Because each category data should has identical feature. And it should not be noised by distribution of data when the encoder extracts features. Because of depending on the possibility output by encoder, the decoder should be changed to adapt various data with different distribute.

A. Multiple Encoder

We hold the idea that encoder should abstract same features from identical category data, but single encoder will be mislead by non-IID data. We designed a loss function for encoders each of that focus on one category data. We assumed that if there are category-special feature, the more convergent the feature is, the more general it will be. And the loss function can be formula as follow:

$$L_e(\boldsymbol{\theta}_1, ..., \boldsymbol{\theta}_C; \boldsymbol{x}) = \sum_{i=1}^{C} D(\beta_i(\boldsymbol{l})\phi_i(\boldsymbol{\theta}_i; \boldsymbol{x})). \tag{1}$$

Here, $D(\cdot)$ represents divergence and we we choose **standard deviation** (STD) to measure the features' divergence output by encoder in this article. $\phi_i(\cdot)$ is ith encoder or extractor in the net. It outputs features with the form as:

$$\phi_i(\boldsymbol{\theta}_i; \boldsymbol{x}) = F.$$

$\boldsymbol{\theta}_i$ is the parameter of ith encoder ϕ_i. C is the number of category. And $F \in R^{B\times N}$ represent features which are extracted by ϕ_i from inputs $\boldsymbol{x} \in R^{B\times 1}$. $\beta_i(\boldsymbol{l})$ is focus coefficient lead by label $\boldsymbol{l} \in R^{B\times 1}$. B is the number of encoders. N is the number of features output by one encoder from ith input $\boldsymbol{x_i}$. Finally, the $D(\cdot)$ is formulated as:

$$D(\beta_i(\boldsymbol{l})\phi_i(\boldsymbol{\theta}_i; \boldsymbol{x})) = \sum_{j=1}^{N} \frac{std(\beta_i(\boldsymbol{l}) \odot F_{:,j})}{N}. \tag{2}$$

Here, $F_{:,j}$ represents jth feature which is extracted by encoders. N represents that the input can be extracted to N features. We called Equation 1 as divergence loss.

There are some different between classification task and segmentation task. Because in classification task the encoder extracts features belong to one category data only. While all category features exist in identical outputs when encoders take segmentation task.

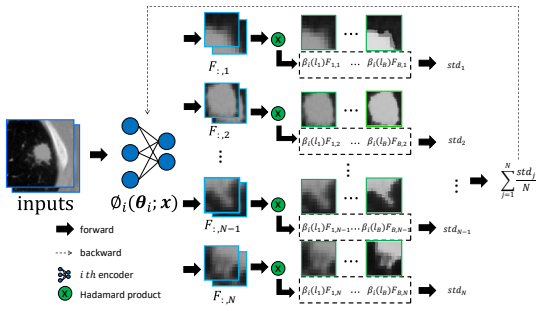


Fig. 2. **Divergence loss in classification task**. Each category data can be extracted as n unique features. Focus coefficient $\beta_i(\boldsymbol{l})$ in Equation 3 varies to different category data.

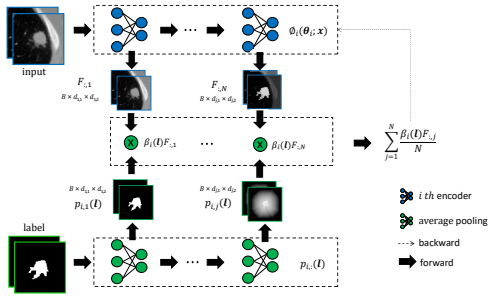


Fig. 3. **Divergence loss in segmentation task**. The target's relative position is unchanging in encoder's output though size of output is unequal to that of input. The extent of divergence should be negative correlation with the distance of the target's center.

1) Classification task: In classification task, each category input can be extracted as N unique features. If their divergences are small, we consider that the feature's variation should be small in new data. So, we hold some opinion:

- Identical category data share same features.
- If features extracted by encoder from identical category data in training datasets are highly concentrating, it is quite possible that these features also exist in test training datasets and their variation is small.

According to above mentioned, the focus coefficient $\beta_i(\boldsymbol{l})$ is formulated as:

$$\beta_i(\boldsymbol{l}) = e^{\lambda(2\delta(\boldsymbol{l}-i)-1)}. \tag{3}$$

Here, $\delta(\cdot)$ is unit pulse sequence. $\beta_i \in R^{B\times 1}$ is focus coefficient. Figure 2 shows this procession.

2) Segmentation task: Segmentation task is different from classification task, because all the category features exist in identical outputs. There are several assumption in the segmentation task:

- Most time, targets' edge are blurry and dimmed. We need enhance the edge's features.
- The features from background are small than from target in medical data. To balance this difference, features from background should have larger coefficient.
- The target's relative position is unchanging in encoder's output with encoder becoming deeper. And the encoders' outputs are tensor with different scale.

From above assumption, we think that the extent of divergence should be negative correlation with the distance of the target's center and we use average pooling to replace it. So we design $\beta_i(\boldsymbol{l})$ as:

$$\beta_i(\boldsymbol{l}) = \lambda e^{-p_{i,:}(\boldsymbol{l}^\gamma)}.$$

Here, $p_{i,j}(\cdot)$ represents average pooling which output consequence with same size as $F_{i,j}$. Labels $\boldsymbol{l}_i$ are tensors and feature $F_{i,j} \in R^{d_{j,1}\times d_{j,2}}$ means j th feature from ith input. The feature's size is related with the depth of encoder. $p_{i,j}(\boldsymbol{l}^\gamma) \in R^{d_{j,1}\times d_{j,2}}$ means ith input label that is averaged pooling matching with identical size of $F_{i,j}$. γ is hyper-parameters which determines the average pooling edge sharpness of ith input. λ controls the average pooling strength of input. We can refer the Figure 3.

We connect the encoders with a decoder and make a decision. On the whole, this network is composed of several encoders and one decoder. Its loss function is formula as two parts:

$$\min_{\Theta} L(\Theta) = L^\star(\Theta; X) + \alpha L_e(\Theta; X). \tag{4}$$

Here, K is the number of client. $L^\star$ is risk loss function. L_e is the divergence loss. Normally, we choose entropy-cross loss function as $L^\star$ that is mainly extracting the dividable features. But the features caught by entropy-cross loss function may not be general because it focuses on features' divisibility in training data. So, we add divergence loss to re-catch the encoder's attention on the generalization of features.

B. Personal Decoder

Thinking about the scene that the divergences of some encoders' output are tiny in same class data while decoders' parameters which correspond with the outputs vary widely in different client, we guess that these features encoded by encoders is client-only. Generally speaking, if the symptom in chinese pneumonia patients is cough while that in America more likely is dyspnea, we should aware that this symptom maybe regional or ethnically diverse. The symptom represents the output of encoders and different symptom make different decision meaning that decoders' parameters vary among countries. In discussion, they will have reservation if they disagree others' opinions. Thus, referring to similar scene, we personalized decoder according to its divergence. This procession can be indicated by Algorithm 1. And we can transform the mentions above to equation:

$$\varphi_t^k = \varphi_t^{k,*} - (\varphi_t^{k,*} - \bar{\varphi}_t^k) g(std(\Phi_t^k)) \tag{5}$$

Here, φ_t^k is the parameters which should be sent to kth client after aggregating in tth iterations. $\varphi_t^{k,*}$ is the decoder's parameters returned by kth client in tth iterations. $\bar{\varphi}_t^k$ is mean value of n clients nearing kth client. Φ_t^k is a matrix being

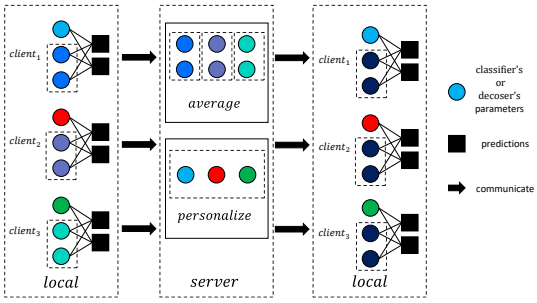


Fig. 4. **Aggregation of classifier's or decoder's parameters**. Parameters will be aggregated in average when the local parameters' divergence is small. If their divergence is large, the local parameters will be reserved and dose not take part in aggregating. What the parameter should be, is decided by Equation 6.

consisted of parameters from n clients nearing kth client. $g(\cdot)$ is the selection function according to the divergence of clients' parameters. In this paper, we formulate the selection function as:

$$g(x) = e^{-(\lambda x)^{\gamma}} \tag{6}$$

If $\gamma > 1$, the function includes three part:

- $g = 1$, means that local parameters are mean value when the divergence is small.
- $g \approx kx + b$, means that local parameters are linearly leaded by mean value.
- $g = 0$, means that local parameters are personalized value when the divergence is big.

We set $\gamma = 10$ that will transform it to be hard selection function. When $\lambda x > 1$, we think x is personal parameter while x is global parameter if $\lambda x < 1$. And we should notice that parameters' divergence will be larger with aggregation turns increasing. It can be regard as there are more local parameters in the end of aggregation. We consider that this phenomenon is benefit to aggregation because in the beginning of aggregation the models are insufficient training and they need more global parameters to guide the local learning. Figure 4 shows the procession of aggregation.

Algorithm 1 personalized aggregation

Server : $(t = 1, 2, ..., T)$

while $t < T + 1$ **do**
  get decoders' parameters $\varphi_t^{k,*}$ from clients
  aggregate $\varphi_t^{k,*}$ by eq 5 and outcome φ_t^k
  Send φ_t^k to client.

Client k **:** $(t = 1, 2, ..., T)$

while $t < T + 1$ **do**
  get decoder's parameters φ_t^k from server
  $\theta_t^{i,k,*} \leftarrow \min_{\theta}[f(\bar{\theta}_{t-1}^i, \varphi_t^k; x) + \alpha L_e(\bar{\theta}_{t-1}^i, \varphi_t^k; x)]$
  Send encoder's parameters $\theta_t^{i,k,*}$ to server.

C. Algorithm

We argue that identical data shares same feature in different clients. Thus, the encoders' parameters in each client should be same. We aggregate each class-special encoder through class weighted average in Algorithm 2. And we formulate the aggregation function:

$$\bar{\theta}_t^i = \sum_{k=1}^{K} \frac{N_k^i \theta_t^{i,k,*}}{N^i} \tag{7}$$

The Equation 7 outcomes same parameters in identical category-favor encoders which handle certain kind of category data in preference to others. Where, $\bar{\theta}_t^i$ is aggregation weight in class-favor encoder. N_k^i is the number of ith class in kth client while $N^i = \sum_{k=1}^{K} N_k^i$ is total number of ith class data. $\theta_t^{i,k,*}$ is kth client returned ith class-favor encoder's parameters.

Algorithm 2 Class weighted aggregation

Server : $(t = 1, 2, ..., T)$

while $t < T + 1$ **do**
  get encoders' parameters $\theta_t^{i,k,*}$ from clients
  aggregate $\theta_t^{i,k,*}$ by eq 7 and outcome $\bar{\theta}_t^i$
  Send $\bar{\theta}_t^i$ to client.

Client k **:** $(t = 1, 2, ..., T)$

while $t < T + 1$ **do**
  get encoder's parameters $\bar{\theta}_t^i$ from server
  $\varphi_t^{k,*} \leftarrow \min_{\varphi} f(\bar{\theta}_t^i, \varphi_{t-1}^k; x)$
  Send decoder's parameters $\varphi_t^{k,*}$ to server.

Algorithm 3 FedDiv

Server :
  Initialize: send $\bar{\theta}_0^i, \varphi_0^k$ to clients
  Aggregate $\boldsymbol{\theta}$ according Algorithm 2
  Aggregate $\boldsymbol{\varphi}$ according Algorithm 1
Client :
  Update $\boldsymbol{\theta}$ and send $\boldsymbol{\varphi}$ according Algorithm 2
  Update $\boldsymbol{\varphi}$ and send $\boldsymbol{\theta}$ according Algorithm 1

Here, $\boldsymbol{\theta}$ and $\boldsymbol{\varphi}$ represents sub-encoders' parameters and decoder's parameters respectively. It describes the whole process of FedDiv. We use the method of alternatively optimizing between encoder and decoder. It transmits whole parameters in each two communications, therefor the strategy doesn't increase the cost of communication. In a word, we firstly find general feature through class-favor encoders and get personal model through personalized decoder.

V. Experiments

In this section, we will display the performance of FedDiv and compare it with others strategies including two generic federated leaning algorithms and twelve personalized federated learning algorithms. We chose FedAvg [11] and FedProx [12] as comparable algorithm. While the personalized algorithms include FedAMP [7], FedRoD [9], L2GD [8] and so on.

A. Setting and material

*1) **Metric**:* In classification task, we used **best mean accuracy** (BMACC) [7] to evaluate total performance of strategy and **best mean balance accuracy** (BMBACC), **best mean class-average F1 score** (BMCAF1) and **best mean specificity** (BMSpec) to estimate their performance in each class. BMACC is the highest averaging accuracy of all clients in whole communicating period. BMBACC is the official metric of the ISIC 2019 Challenge [25]. They can be formula as:

$$BMBACC = \frac{1}{N_{client}N_{class}} \sum_{k=1}^{N_{client}} \sum_{i=1}^{N_{class}} \frac{TP_{i,k}}{TP_{i,k} + FN_{i,k}}.$$

$$BMACC = \frac{1}{N_{client}} \sum_{k=1}^{N_{client}} \frac{\sum_{i=1}^{N_{class}} TP_{i,k}}{N_k}. \quad (8)$$

In fact, BMBACC is mean sensitive in each category data and we adhere the reference [25] calling it BMBACC. Corresponding to it, **best mean specificity** (BMSpec) can be formulated as:

$$BMSpec = \frac{1}{N_{client}N_{class}} \sum_{k=1}^{N_{client}} \sum_{i=1}^{N_{class}} \frac{TN_{i,k}}{FP_{i,k} + TN_{i,k}}.$$

To more effective estimate the strategy's ability of distinguish different category data, we refer to BMBACC and defined F1 score in different category as BMCAF1 which can be formula as follow:

$$BMCAF1 = \frac{1}{N_{client}N_{class}} \sum_{k=1}^{N_{client}} \sum_{i=1}^{N_{class}} \frac{2TP_{i,k}}{FP_{i,k} + FN_{i,k} + 2TP_{i,k}}.$$

Here, N_{client} is the number of clients. N_{class} is the number of class. $TP_{i,k}, TN_{i,k}, FP_{i,k}$ and $FN_{i,k}$ represent true positive, true negative, false positive and false negative of class i in kth client respectively. And N_k is number of total sample in k th client.

In segmentation task, we choose **best mean intersection over union** (BMIoU) and **beast mean dice** (BMDice) as metric which are general used in segmentation task to measure strategies' performance [26]. They can be formula as follow respectively:

$$BMIoU = \frac{1}{N_{client}N_{class}} \sum_{k=1}^{N_{client}} \sum_{i=1}^{N_{class}} \frac{TP_{i,k}}{FP_{i,k} + FN_{i,k} + TP_{i,k}}.$$

$$BMDice = \frac{1}{N_{client}N_{class}} \sum_{k=1}^{N_{client}} \sum_{i=1}^{N_{class}} \frac{2TP_{i,k}}{FP_{i,k} + FN_{i,k} + 2TP_{i,k}}.$$

The meaning of above two equations is same as Equation 8. In addition, we add **best mean sensitivity** (BMSens) as metric in segmentation task:

$$BMSens = \frac{1}{N_{client}N_{class}} \sum_{k=1}^{N_{client}} \sum_{i=1}^{N_{class}} \frac{TP_{i,k}}{TP_{i,k} + FN_{i,k}}.$$

For the reason of sensitivity and specificity possessing identical value in two target segmentation task or two category classification task, we display **best mean specificity** (BMSpec) in four target segmentation task and three category classification task only.

*2) **Datasets and model**:* We used **lung nodule analysis 2016** (LUNA16) datasets [27] in binary label classification task and DFUC2021 [28] datasets in three label classification task. We divided LUNA16 datasets into four proportion including 1:1, 1:2, 1:4 and 1:7 which are ratio about the number of two class samples. These ratios are denoted by $\delta = \mathbf{1}, \delta = \mathbf{2}, \delta = \mathbf{4}$ and $\delta = \mathbf{7}$ respectively. And we set the **diabetic foot ulcers datasets 2021** (DFUC2021) datasets as 1:1:1, 1:1:2, 1:1:3 and 1:1:4 in same way. Similar to LUNA16 datasets, they are denoted by $\delta = \mathbf{1}, \delta = \mathbf{2}, \delta = \mathbf{3}$ and $\delta = \mathbf{4}$ respectively. In this way, we built label unbalanced datasets in different extent.

In segmentation task, we handled **multimodal brain tumor segmentation challenge 2017** (BraTS 2017) datasets [29] as single target and three target data that was rebuilt as four model. To build non-IID data, we hid one, two, three modal data to clients respectively. For comparison, we added balance data which includes all modal data. Finally, there are four different distribute data include no-lack and lack one, two, and three modal data and we used $l = 0$, $l = 1$, $l = 2$ and $l = 3$ to denote them. Finally, we got modal unbalanced datasets in different extent.

There are five clients taking part in federated learning. And all of the clients' network base on densenet in classification task and unet in segmentation task. Each client own itself advantage class, for example client 1's number of positive samples is two time than negative samples while client 2 is contrary to client 1. We choose FedAvg [11], FedProx [12], FedAMP [7], FedRoD [9], L2GD [8], FPFC [13], IFCA [14], SuPerFed [30], Ditto [31], pFedME [32], FedPer [33], FedRep [34], pFedNet [35] and FedABC [10] as comparable algorithm. But because of different network being adopted by different strategy, we control the number of network's parameters in same. All the clients and strategies are completed in PyTorch 2.0 running on ASUS with Intel(R) Core(TM) i7-12700k CPU, NVIDIA GeForce RTX 3090 and Ubuntu 20.04.

B. Result on features calibrating

We designed loss function which was formula as Equation 4. And different α selected by us would made different consequence. We use LUNA16 in classification task and BraTS 2017 in segmentation task showing this results with $\lambda = 0$ in Equation 5. The results of classification task and segmentation task were displayed in Figure 5.

These results suggest that calibrating the multi-encoder is benefit to improve the model's accuracy after aggregating especially in balance data. With the data distribution's extent of unbalance increasing, the effect of classification task will be weakened especially when the data is extremely unbalanced such as $\delta = 7$ in LUNA16. It is because outputs of sub-encoder are not match with decoder. The effect of calibration is covered

by averaged classifier or decoder when the data distribution is unbalanced. So, we add another experiment shown in Figure 6. In this experiment, we set $\lambda = 100$ which weaken the effect of decoder on encoders. And it indicates that the optimal value nears 0.2 that is similar to others experiment's result.

We can find that despite the different distribute of data, optimal value is close in themselves' datasets either in LUNA16 which represents label unbalanced or BraTS 2017 which represents modal unbalance. This phenomenon further shows that we extract dividable features which are more general before decoder or classifier by divergence loss function in Equation 4. Many people focus on features' dividable but they ignore its' generalization. In this paper, we use entropy-cross loss function to get dividable features and use divergence loss function to lead it being more general.

C. Result on personalized aggregating

From the Equation 5, we can analyze that if $\lambda x > 1$, x should be considered as personalizing parameter when the select function $g = e^{-(\lambda x)^{\gamma}}$ and $\gamma = +\infty$. Thus, the number of personalizing parameters and λ should be in direct ratio. In this paper we fixed γ as 10. And we defined that personalizing ratio is the number of parameters' STD which multiplied by λ is larger than 1 being divided by the number of total parameters. We used LUNA16 datasets with $\delta = 7$ and BraTS 2017 datasets with $l = 3$ to validate this assumption.

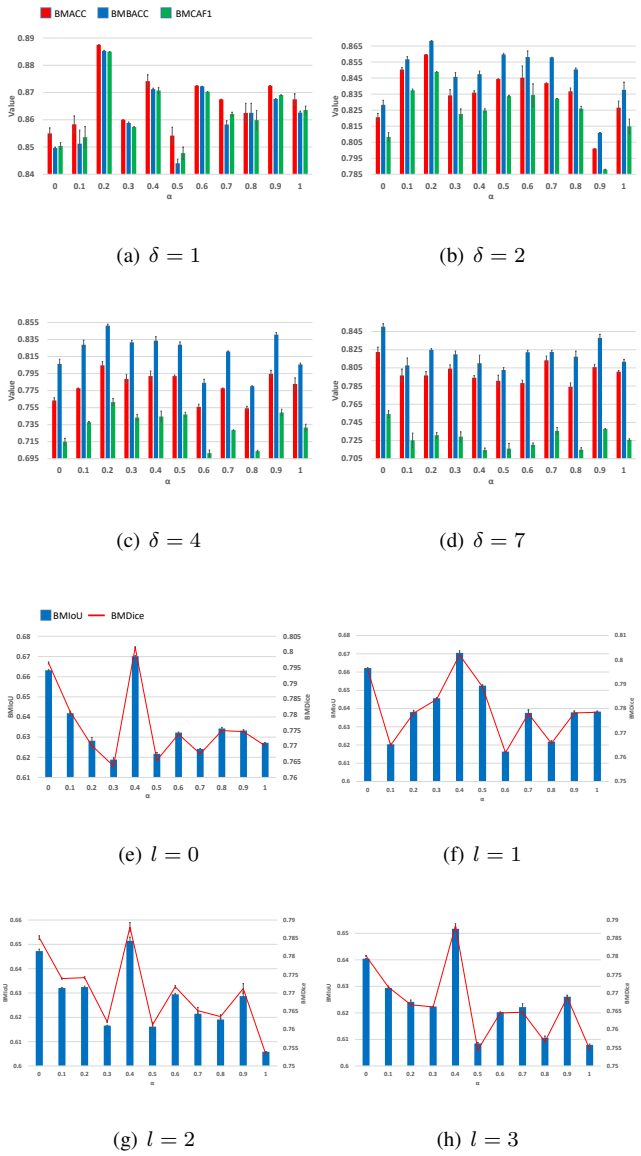
We got results on Figure 7 which are divided into four sub-figures. Sub-figure (a) and (b) shown personalizing ratio and classifier's parameters' mean STD between different clients in classification task and sub-figure (c) and (d) shown identical metric in segmentation task. The personalizing ratio in classification task presents stepped increase while it increase at first and then decrease in segmentation task. The reason we guessing is that the number of classifier's parameters in classification task is far lower than decoder's in segmentation task. But in total, personalizing ratio increases with λ becoming bigger. And we find that there are threshold in λ and personalizing ratio is not zero if λ is bigger than it.

To study personalizing of classifier or decoder, we select λ varying from 0 to 100 in Equation 5 with $\alpha = 0$ in Equation 4. The results of classification task which are shown from sub-figure (a) to (d) and segmentation task which are shown from sub-figure (e) to (h) in Figure 8.

In classification task, as label becoming unbalanced, higher BMACC requires larger λ while BMBACC and BMCAF1 require relatively smaller λ. However in segmentation task, BMIoU and BMDice are almost no change when modal becomes unbalanced. From the result of these task in different data distribution, we concluded that personalizing should be related with proportion of category instead of proportion of modals' composition. And slightly personalizing will get high BMACC in label unbalanced data.

D. Compare with other strategies

We compared 14 strategies include FedAvg, FedProx, FedAMP, FedRoD, L2GD, FPFC, IFCA, SuPerFed, Ditto, pFedME, FedPer, FedRep, pFedNet and FedABC to validate



(a) $\delta = 1$ (b) $\delta = 2$

(c) $\delta = 4$ (d) $\delta = 7$

(e) $l = 0$ (f) $l = 1$

(g) $l = 2$ (h) $l = 3$

Fig. 5. **Result on features calibrating**. Calibrating the multi-encoder's outputs is benefit to improve the model's accuracy after aggregating. Sub-fig (a) to (d) are classification task training by LUNA16 and (e) to (h) are segmentation task training by BraTS 2017. Despite the different distribute of data, optimal value is near to $\alpha = 0.2$ in LUNA16 and $\alpha = 0.4$ in BraTS 2017.

our strategy is superior performance in both classification task and segmentation task. We use BMACC, BMBACC and BMCAF1 in classification task and BMIoU, BMDice and BMSpec in segmentation task as the metric to compare strategies' performance.

1) Classification task: We conducted classification task by LUNA16 with two label and DFUC 2021 with three label. These results are displayed in Table I and Table II. These tables shown our strategy's advantage over other strategies to some extent in both binary classification and multi classification task.

In our compared strategies, FedRoD shows relative excellent performance for BMACC especially in the unbalanced data.

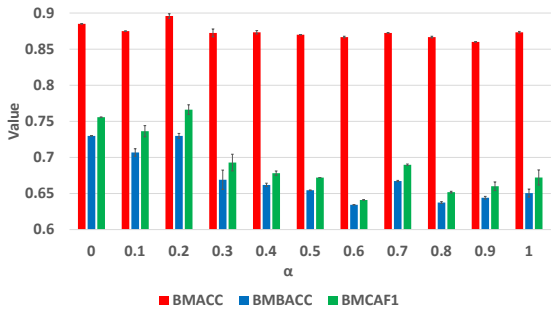
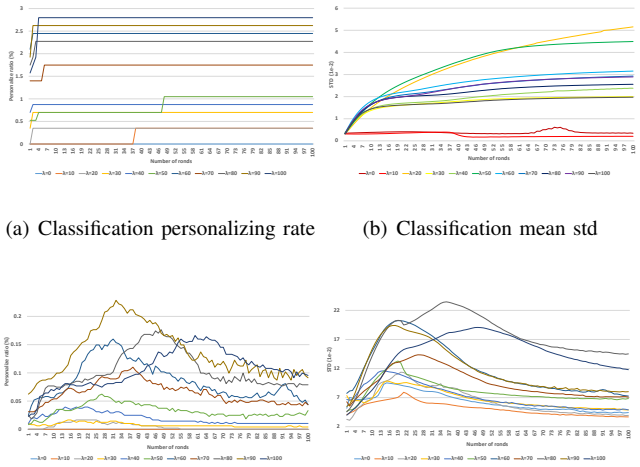


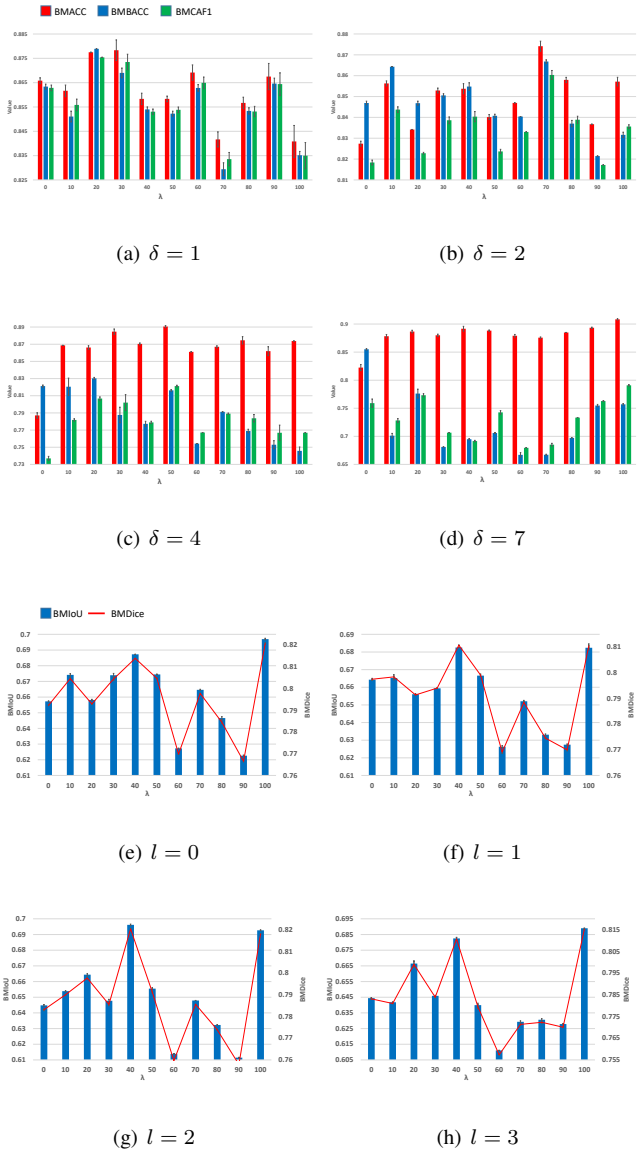
Fig. 6. **Complement experiment when** $\delta = 7$. We add another experiment on LUNA16 when $\delta = 7$ and $\lambda = 100$. The result shows that the optimal coefficient value is $\aleph = 0.2$.



(a) Classification personalizing rate (b) Classification mean std

(c) Segmentation personalizing rate (d) Segmentation mean std

Fig. 7. **Personalizing ratio and STD**. Sub-figure (a) and (b) shown personalizing ratio and classifier's parameters' mean STD between different clients in classification task and sub-figure (c) and (d) shown identical metric in segmentation task. In total, personalizing ratio increases with λ becoming larger.



(a) $\delta = 1$ (b) $\delta = 2$

(c) $\delta = 4$ (d) $\delta = 7$

(e) $l = 0$ (f) $l = 1$

(g) $l = 2$ (h) $l = 3$

Fig. 8. **Result on personalizing classifier or decoder**. The results of classification task which are shown from sub-figure (a) to (d) and segmentation task which are shown from sub-figure (e) to (h). Personalizing should be related with proportion of category instead of modals' composition

The reason we guess is that FedRoD focus on the advantage data. Interms of BMACC, when the data distribution is extremely unbalanced the profit of correctly classifying advantaged category surpass the loss of mistakenly classifying disadvantaged category. Thus, the shortcoming of FedRod is that it's performance for BMBACC and BMCAF1 which pay more attention to disadvantaged category relative poor. While in medical industry minor group do important to diagnosis such as rare disease.

Meanwhile, we couldn't ignore major popular otherwise high misdiagnose rate will impact hospitals' quality of health care. Thus, FedAvg, pFedNet and IFCA may not suitable strategies to screen in medicine though they have relative high BMBACC and BMCAF1. In other hand, IFCA needs repeatedly communicating between clients and server that will consume higher costs.

Our aggregation algorithm not only improves models performance in BMACC which majorly used in daily screening but also in BMBACC and BMCAF1 which majorly used in making a definite diagnosis. And it working well in different data distribute of binary or multi classification in terms of medical data.

2) Segmentation task: Some algorithms are not considering segmentation task. But in theory they can be used to that through deforming to some extent. We handled BraTS 2017 datasets as two targets and four targets in segmentation task.

To build non-IID data, we treat the datasets' four channels as four views. It is hided to clients in different extent and we use l denoting the this extent. If $l = n$, there are n kind of view being hided to clients that views vary from clients. We called that as modal unbalanced. And these results are shown by Table III and Table IV.

In two targets segmentation task, FedAvg, FedRoD and IFCA get relative considerable BMIoU and BMDice regardless of modal unbalanced. We guess that the volume in space of two targets is equal in expectation. When it comes to four targets segmentation, their BMIoU and BMDice are relative low. We assume that it was caused by their overlook about minorities.

FedAMP and L2GD achieved relatively high BMIoU and BMDice in extremely modal unbalanced data in four targets segmentation task. But their methods to aggregate models' weight would cost higher calculate resource especially when the quantity of clients is large. And their performance in other distribution are not well. We think that their gradient directions are lead by all clients and it is easily mislead by some outliers.

Segmentation task in medical diagnosis is relative unimportant, but they absolutely play significant role in surgery. If the machine's sensitive is not enough, the health organize may be resected wrongly which is intolerable in brain surgery. While cancer will be remained in body which maybe fatal when machine's specificity is low. And our strategy not only considers sensitive but also specificity which has higher accuracy comparing to other strategies.

VI. Conclusion

In this paper, we consider the medical actual scenery and compare different strategies in special metrics which are more accepted by medicine. Thinking about that there different kind of data distribution in real works, we built different types of distribution data including label unbalanced and modal unbalance. And we validated that our strategy is relative advance over others we compared in different scene. But supervised learning may not suite large scale federated learning, we will pay more attention to semi-supervised or unsupervised learning base on this work in the future.

References

[1] G. J. S. Litjens, T. Kooi, B. E. Bejnordi, A. A. A. Setio, F. Ciompi, M. Ghafoorian, J. van der Laak, B. van Ginneken, and C. I. Sánchez, "A survey on deep learning in medical image analysis," *Medical image analysis*, vol. 42, pp. 60–88, 2017.

[2] S. Saha and T. Ahmad, "Federated transfer learning: concept and applications," *Intelligenza Artificiale*, vol. 15, pp. 35–44, 2020.

[3] Q. Li, Y. Diao, Q. Chen, and B. He, "Federated learning on non-iid data silos: An experimental study," *2022 IEEE 38th International Conference on Data Engineering (ICDE)*, pp. 965–978, 2021.

[4] H. Zhu, J. Xu, S. Liu, and Y. Jin, "Federated learning on non-iid data: A survey," *ArXiv*, vol. abs/2106.06843, 2021.

[5] V. Kulkarni, M. Kulkarni, and A. Pant, "Survey of personalization techniques for federated learning," *2020 Fourth World Conference on Smart Trends in Systems, Security and Sustainability (WorldS4)*, pp. 794–797, 2020.

[6] A. Z. Tan, H. Yu, L. zhen Cui, and Q. Yang, "Towards personalized federated learning," *IEEE transactions on neural networks and learning systems*, vol. PP, 2021.

[7] Y. Huang, L. Chu, Z. Zhou, L. Wang, J. Liu, J. Pei, and Y. Zhang, "Personalized cross-silo federated learning on non-iid data," in *AAAI Conference on Artificial Intelligence*, 2020.

[8] F. Hanzely and P. Richtárik, "Federated learning of a mixture of global and local models," *ArXiv*, vol. abs/2002.05516, 2020.

[9] H.-Y. Chen and W.-L. Chao, "On bridging generic and personalized federated learning for image classification," in *International Conference on Learning Representations*, 2022.

[10] D. Wang, L. Shen, Y. Luo, H. Hu, K. Su, Y. Wen, and D. Tao, "Fedabc: Targeting fair competition in personalized federated learning," *ArXiv*, vol. abs/2302.07450, 2023.

[11] H. B. McMahan, E. Moore, D. Ramage, S. Hampson, and B. A. y Arcas, "Communication-efficient learning of deep networks from decentralized data," in *International Conference on Artificial Intelligence and Statistics*, 2016.

[12] A. K. Sahu, T. Li, M. Sanjabi, M. Zaheer, A. Talwalkar, and V. Smith, "Federated optimization in heterogeneous networks," *arXiv: Learning*, 2018.

[13] X. Yu, Z. Liu, Y. Sun, and W. Wang, "Clustered federated learning based on nonconvex pairwise fusion," *ArXiv*, vol. abs/2211.04218, 2022.

[14] A. Ghosh, J. Chung, D. Yin, and K. Ramchandran, "An efficient framework for clustered federated learning," *IEEE Transactions on Information Theory*, vol. 68, pp. 8076–8091, 2020.

[15] X. Zhang, Y. Li, W. Li, K. Guo, and Y. Shao, "Personalized federated learning via variational bayesian inference," in *International Conference on Machine Learning*, 2022.

[16] M. Luo, F. Chen, D. Hu, Y. Zhang, J. Liang, and J. Feng, "No fear of heterogeneity: Classifier calibration for federated learning with non-iid data," *ArXiv*, vol. abs/2106.05001, 2021.

[17] X. Shang, Y. Lu, Y. ming Cheung, and H. Wang, "Fedic: Federated learning on non-iid and long-tailed data via calibrated distillation," *2022 IEEE International Conference on Multimedia and Expo (ICME)*, pp. 1–6, 2022.

[18] Y. Zhao, M. Li, L. Lai, N. Suda, D. Civin, and V. Chandra, "Federated learning with non-iid data," *ArXiv*, vol. abs/1806.00582, 2018.

[19] C. Li, P.-H. Huang, Y.-T. Ma, H. Hung, and S. Huang, "Robust aggregation for federated learning by minimum γ-divergence estimation," *Entropy*, vol. 24, 2022.

[20] L. Zhang, Y. Luo, Y. Bai, B. Du, and L. yu Duan, "Federated learning for non-iid data via unified feature learning and optimization objective alignment," *2021 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 4400–4408, 2021.

[21] X. Li, M. Jiang, X. Zhang, M. Kamp, and Q. Dou, "Fedbn: Federated learning on non-iid features via local batch normalization," *ArXiv*, vol. abs/2102.07623, 2021.

[22] W. Zhuang, Y. Wen, and S. Zhang, "Divergence-aware federated self-supervised learning," *ArXiv*, vol. abs/2204.04385, 2022.

[23] F. X. Yu, A. S. Rawat, A. K. Menon, and S. Kumar, "Federated learning with only positive labels," in *International Conference on Machine Learning*, 2020.

[24] N. Wu, L. Yu, X. Yang, K.-T. Cheng, and Z. Yan, "Fediic: Towards robust federated learning for class-imbalanced medical image classification," 2022.

[25] ——, "Federated learning with imbalanced and agglomerated data distribution for medical image classication," 2022.

[26] Z. Cai and N. Vasconcelos, "Cascade r-cnn: High quality object detection and instance segmentation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 43, pp. 1483–1498, 2019.

[27] K. Yan, J. Cai, A. P. Harrison, D. Jin, J. Xiao, and L. Lu, "Universal lesion detection by learning from multiple heterogeneously labeled datasets," *ArXiv*, vol. abs/2005.13753, 2020.

[28] M. H. Yap, B. Cassidy, J. M. Pappachan, C. O'Shea, D. Gillespie, and N. D. Reeves, "Analysis towards classification of infection and ischaemia of diabetic foot ulcers," *2021 IEEE EMBS International Conference on Biomedical and Health Informatics (BHI)*, pp. 1–4, 2021.

[29] B. H. Menze, A. Jakab, S. Bauer, J. Kalpathy-Cramer, K. Farahani, J. Kirby, Y. Burren, N. Porz, J. Slotboom, R. Wiest, L. Lanczi, E. Gerstner, M.-A. Weber, T. Arbel, B. B. Avants, N. Ayache, P. Buendia, D. L. Collins, N. Cordier, J. J. Corso, A. Criminisi, T. Das, H. Delingette, . Demiralp, C. R. Durst, M. Dojat, S. Doyle, J. Festa, F. Forbes, E. Geremia, B. Glocker, P. Golland, X. Guo, A. Hamamci, K. M. Iftekharuddin, R. Jena, N. M. John, E. Konukoglu, D. Lashkari, J. A. Mariz, R. Meier, S. Pereira, D. Precup, S. J. Price, T. R. Raviv, S. M. S. Reza, M. Ryan, D. Sarikaya, L. Schwartz, H.-C. Shin, J. Shotton, C. A. Silva, N. Sousa, N. K. Subbanna, G. Szekely, T. J. Taylor, O. M. Thomas, N. J. Tustison, G. Unal, F. Vasseur, M. Wintermark, D. H. Ye, L. Zhao, B. Zhao, D. Zikic, M. Prastawa, M. Reyes, and K. Van Leemput, "The multimodal brain tumor image segmentation benchmark (brats)," *IEEE Transactions on Medical Imaging*, vol. 34, no. 10, pp. 1993–2024, 2015.

[30] M. Bi, Z. Zhao, J. Yang, and Y. Wang, "Comparison of case-based learning and traditional method in teaching postgraduate students of medical oncology," *Medical Teacher*, vol. 41, pp. 1124 – 1128, 2019.

[31] T. Li, S. Hu, A. Beirami, and V. Smith, "Ditto: Fair and robust federated learning through personalization," in *International Conference on Machine Learning*, 2020.

[32] C. T. Dinh, N. H. Tran, and T. D. Nguyen, "Personalized federated learning with moreau envelopes," *ArXiv*, vol. abs/2006.08848, 2020.

[33] M. G. Arivazhagan, V. Aggarwal, A. K. Singh, and S. Choudhary, "Federated learning with personalization layers," *ArXiv*, vol. abs/1912.00818, 2019.
[34] L. Collins, H. Hassani, A. Mokhtari, and S. Shakkottai, "Exploiting shared representations for personalized federated learning," in *International Conference on Machine Learning*, 2021.
[35] Y. Zhao, Q. Liu, X. Liu, and K. He, "Medical federated model with mixture of personalized and sharing components," *ArXiv*, vol. abs/2306.14483, 2023.

TABLE I
RESULTS OF BINARY LABEL CLASSIFICATION TASK ON LUNA16

(a) $\delta = 1$

	BMACC	BMBACC	BMCAF1
FedAvg	87.00±0.00	85.78±0.00	86.39±0.00
FedProx	82.92±0.24	81.11±0.04	81.68±0.06
Ditto	84.50±0.00	83.56±0.09	83.78±0.07
FedPer	77.08±0.62	77.05±0.21	76.68±0.40
FedRep	82.67±0.72	82.34±0.57	82.25±0.69
FedRoD	87.00±0.00	86.40±0.06	86.56±0.04
FPFC	76.25±0.00	73.96±0.13	74.41±0.11
IFCA	85.08±0.12	83.93±0.12	84.44±0.12
pFedME	79.58±0.31	77.90±0.20	78.35±0.24
SuperFed	69.92±1.30	70.25±0.72	68.61±1.11
FedAMP	79.33±0.31	78.17±0.27	78.44±0.30
L2GD	77.00±0.20	75.65±0.07	75.94±0.09
pFedNet	85.50±0.00	85.40±0.00	85.25±0.00
FedABC	71.00±0.00	71.61±0.10	70.83±0.02
FedDiv	**88.5±0.20**	**88.18±0.14**	**88.19±0.19**

(b) $\delta = 2$

	BMACC	BMBACC	BMCAF1
FedAvg	83.93±0.21	84.49±0.22	82.66±0.20
FedProx	79.76±0.12	80.49±0.33	78.31±0.12
Ditto	80.87±0.42	81.45±0.49	79.16±0.33
FedPer	79.51±0.12	73.19±0.83	74.53±0.81
FedRep	83.50±0.12	78.97±0.42	80.21±0.27
FedRoD	86.39±0.12	84.69±0.04	84.49±0.11
FPFC	76.45±0.43	73.79±0.26	73.23±0.32
IFCA	84.95±0.21	85.25±0.22	83.65±0.22
pFedME	78.15±0.12	77.51±0.45	75.89±0.43
SuperFed	75.94±0.64	71.22±0.73	70.95±1.13
FedAMP	72.28±0.12	74.29±0.67	73.20±0.21
L2GD	76.70±0.60	73.84±0.20	73.19±0.65
pFedNet	82.40±0.00	84.55±0.05	81.48±0.02
FedABC	76.79±0.21	72.31±0.22	70.22±0.21
FedDiv	**86.99±0.00**	**85.40±0.00**	**85.30±0.00**

(c) $\delta = 4$

	BMACC	BMBACC	BMCAF1
FedAvg	84.76±0.31	84.99±0.30	79.80±0.57
FedProx	82.66±0.66	84.12±0.65	77.77±0.57
Ditto	86.20±0.83	81.27±1.13	77.18±0.48
FedPer	83.84±0.41	62.11±1.02	62.96±1.22
FedRep	86.78±0.43	69.92±1.04	73.24±1.43
FedRoD	**91.75±0.31**	84.28±0.81	86.26±0.28
FPFC	82.66±0.12	71.65±0.17	72.19±0.17
IFCA	88.72±0.12	86.93±0.07	83.59±0.18
pFedME	84.76±0.12	81.48±0.52	76.89±0.23
SuperFed	81.23±0.60	64.91±0.88	63.05±1.11
FedAMP	84.26±0.60	76.47±0.25	75.25±1.23
L2GD	85.44±0.31	76.53±0.49	76.22±0.99
pFedNet	78.79±0.21	85.43±0.40	74.78±0.27
FedABC	73.23±0.36	73.26±0.92	66.82±0.61
FedDiv	91.16±0.21	**87.13±0.83**	**86.35±0.29**

(d) $\delta = 7$

	BMACC	BMBACC	BMCAF1
FedAvg	81.25±0.00	81.41±0.00	75.65±0.00
FedProx	79.92±0.12	80.73±0.35	72.50±0.21
Ditto	86.83±0.72	82.19±1.43	78.06±1.23
FedPer	83.58±0.12	55.06±0.14	51.91±0.13
FedRep	86.08±0.24	66.79±1.41	66.85±0.71
FedRoD	89.83±0.12	78.93±0.48	80.63±0.43
FPFC	86.33±0.12	67.93±0.00	68.86±0.00
IFCA	88.08±0.62	85.19±0.15	80.55±0.17
pFedME	86.67±0.42	77.01±0.24	75.67±0.44
SuperFed	83.42±0.12	54.77±0.14	51.67±0.11
FedAMP	86.58±0.12	69.48±0.19	70.77±0.17
L2GD	86.08±0.12	72.42±0.25	73.16±0.54
pFedNet	87.33±0.31	**86.41±0.70**	80.47±0.66
FedABC	74.83±0.12	76.88±0.48	67.34±0.20
FedDiv	**91.83±0.12**	85.18±0.26	**85.34±0.45**

TABLE II
RESULTS OF THREE LABEL CLASSIFICATION TASK ON DFUC 2021

(a) $\delta = 1$

	BMACC	BMBACC	BMCAF1	BMSpec
FedAvg	84.71±0.00	75.89±0.00	75.31±0.00	89.02±0.00
FedProx	82.93±0.52	73.10±1.02	72.50±1.13	87.62±0.43
Ditto	82.87±0.17	73.04±0.10	72.32±0.09	87.58±0.06
FedPer	82.52±0.08	71.59±0.15	70.89±0.23	86.95±0.04
FedRep	80.74±0.30	69.32±0.15	68.69±0.07	85.77±0.02
FedRoD	86.43±0.22	78.71±0.32	77.95±0.52	90.23±0.16
FPFC	78.43±0.22	65.47±0.35	64.89±0.25	84.26±0.11
IFCA	85.42±0.00	77.16±0.00	76.33±0.00	89.46±0.00
pFedME	80.44±0.00	69.18±0.09	68.60±0.17	85.89±0.04
SuperFed	80.03±0.47	67.82±0.36	67.07±0.31	85.25±0.22
FedAMP	80.27±0.00	69.38±0.13	68.45±0.14	85.85±0.04
L2GD	79.44±0.08	67.72±0.27	67.07±0.17	85.05±0.09
pFedNet	84.18±0.15	75.41±0.30	74.76±0.26	88.56±0.09
FedABC	77.66±0.17	65.44±0.53	64.69±0.37	83.60±0.16
FedDiv	**86.84±0.15**	**78.88±0.21**	**78.41±0.22**	**90.48±0.06**

(b) $\delta = 2$

	BMACC	BMBACC	BMCAF1	BMSpec
FedAvg	86.29±0.24	79.59±0.28	79.10±0.29	89.91±0.18
FedProx	81.85±0.07	72.73±0.07	72.16±0.03	86.71±0.04
Ditto	82.88±0.00	73.91±0.31	72.93±0.13	87.35±0.09
FedPer	81.33±0.30	67.59±0.63	67.43±0.76	85.20±0.27
FedRep	81.05±0.20	68.04±0.33	68.21±0.08	85.17±0.22
FedRoD	86.53±0.11	77.88±0.19	78.14±0.25	89.63±0.03
FPFC	79.51±0.11	67.74±0.06	68.01±0.09	84.51±0.06
IFCA	85.22±0.07	78.16±0.00	77.42±0.00	89.08±0.00
pFedME	83.25±0.18	74.57±0.15	74.03±0.20	87.70±0.06
SuperFed	79.74±0.24	67.10±0.25	67.18±0.36	84.45±0.08
FedAMP	80.30±0.35	69.04±0.53	69.00±0.55	85.23±0.27
L2GD	80.21±0.34	68.87±0.39	68.92±0.38	85.05±0.20
pFedNet	85.36±0.13	78.59±0.12	77.47±0.09	89.41±0.07
FedABC	77.59±0.26	65.82±0.27	65.05±0.27	83.31±0.22
FedDiv	**87.32±0.07**	**79.62±0.21**	**79.74±0.21**	**90.34±0.03**

(c) $\delta = 3$

	BMACC	BMBACC	BMCAF1	BMSpec
FedAvg	81.40±0.11	74.60±0.27	71.25±0.17	85.94±0.11
FedProx	79.41±0.06	70.49±0.45	67.97±0.30	84.30±0.25
Ditto	81.98±0.36	69.84±0.54	68.22±0.33	84.38±0.08
FedPer	81.94±0.06	64.50±0.65	64.71±0.61	82.48±0.19
FedRep	81.90±0.25	65.83±0.63	66.69±0.47	83.26±0.12
FedRoD	**84.16±0.15**	72.37±0.09	72.10±0.15	86.22±0.10
FPFC	80.04±0.25	64.20±0.03	63.66±0.21	82.33±0.05
IFCA	84.04±0.06	**78.85±0.05**	75.39±0.07	88.26±0.03
pFedME	79.68±0.19	66.61±0.32	65.75±0.23	82.91±0.10
SuperFed	79.76±0.15	61.17±0.34	60.43±0.43	81.10±0.11
FedAMP	81.40±0.28	67.93±0.05	67.27±0.18	84.07±0.05
L2GD	80.50±0.10	66.99±0.43	66.01±0.38	83.23±0.22
pFedNet	80.31±0.39	74.03±0.39	70.28±0.48	85.49±0.27
FedABC	73.92±0.28	61.79±0.20	59.46±0.39	80.06±0.19
FedDiv	84.08±0.06	78.84±0.23	**75.54±0.16**	**88.38±0.09**

(d) $\delta = 4$

	BMACC	BMBACC	BMCAF1	BMSpec
FedAvg	83.81±0.00	78.17±0.03	73.14±0.05	87.67±0.03
FedProx	77.91±0.31	70.72±0.41	65.20±0.54	83.37±0.17
Ditto	83.24±0.63	69.89±0.15	67.90±0.55	84.38±0.09
FedPer	85.56±0.05	62.25±0.26	63.44±0.19	81.55±0.06
FedRep	84.17±0.09	64.05±0.15	64.61±0.26	82.09±0.06
FedRoD	86.22±0.09	72.17±0.44	71.84±0.23	85.46±0.31
FPFC	84.30±0.24	69.43±0.17	68.62±0.28	84.68±0.14
IFCA	85.00±0.08	76.86±0.16	71.60±0.24	86.89±0.08
pFedME	83.84±0.48	72.61±0.23	69.65±0.25	85.97±0.04
SuperFed	83.84±0.37	62.64±0.10	63.01±0.61	81.08±0.09
FedAMP	84.50±0.00	69.88±0.11	68.47±0.33	84.85±0.01
L2GD	84.47±0.45	70.85±0.35	69.19±0.10	85.31±0.06
pFedNet	79.07±0.05	76.69±0.09	68.69±0.08	85.70±0.08
FedABC	72.51±0.12	62.21±0.14	56.86±0.18	79.17±0.19
FedDiv	**87.81±0.09**	**78.93±0.18**	**74.44±0.11**	**88.08±0.07**

TABLE III
RESULTS OF TWO TARGET SEGMENTATION TASK
ON BRATS 2017

(a) $l = 0$

	BMIoU	BMDice	BMSens
FedAvg	69.78 ±0.06	82.11 ±0.05	82.22 ±0.02
FedProx	56.70 ±0.28	72.10 ±0.24	72.00 ±0.26
Ditto	67.66 ±0.07	80.62 ±0.06	80.75 ±0.02
FedPer	43.82 ±0.10	58.77 ±0.12	61.59 ±0.08
FedRep	60.80 ±0.04	75.53 ±0.03	75.84 ±0.02
FedRoD	68.77 ±0.03	81.42 ±0.02	81.39 ±0.02
FPFC	61.01 ±0.09	75.69 ±0.07	75.90 ±0.01
IFCA	69.76 ±0.02	82.11 ±0.02	82.15 ±0.01
pFedME	65.02 ±0.12	78.70 ±0.07	78.82 ±0.03
SuperFed	43.82 ±0.11	58.76 ±0.13	61.59 ±0.08
FedAMP	63.25 ±0.08	77.38 ±0.07	77.46 ±0.10
L2GD	62.94 ±0.10	77.15 ±0.06	77.26 ±0.01
pFedNet	65.14 ±0.01	78.85 ±0.00	79.44 ±0.01
FedABC	51.33 ±0.02	67.73 ±0.01	68.07 ±0.01
FedDiv	**71.61 ±0.05**	**83.37 ±0.03**	**83.31 ±0.05**

(b) $l = 1$

	BMIoU	BMDice	BMSens
FedAvg	69.16 ±0.03	81.71 ±0.02	81.74 ±0.01
FedProx	59.47 ±0.18	74.50 ±0.15	74.57 ±0.15
Ditto	65.99 ±0.22	79.44 ±0.17	79.51 ±0.13
FedPer	38.06 ±0.25	51.62 ±0.33	57.46 ±0.19
FedRep	60.51 ±0.03	75.33 ±0.02	75.49 ±0.02
FedRoD	68.63 ±0.12	81.33 ±0.08	81.32 ±0.05
FPFC	61.08 ±0.01	75.75 ±0.00	75.80 ±0.02
IFCA	69.27 ±0.03	81.78 ±0.02	81.71 ±0.01
pFedME	62.68 ±0.09	76.99 ±0.06	77.04 ±0.10
SuperFed	41.64 ±0.21	56.03 ±0.25	60.13 ±0.16
FedAMP	61.45 ±0.02	76.00 ±0.01	75.93 ±0.02
L2GD	61.38 ±0.03	75.96 ±0.02	75.97 ±0.01
pFedNet	66.11 ±0.10	79.53 ±0.07	79.50 ±0.03
FedABC	50.90 ±0.01	67.35 ±0.01	67.62 ±0.00
FedDiv	**70.04 ±0.08**	**82.33 ±0.06**	**82.34 ±0.06**

(c) $l = 2$

	BMIoU	BMDice	BMSens
FedAvg	68.48 ±0.01	81.25 ±0.01	81.24 ±0.01
FedProx	60.04 ±0.10	74.90 ±0.09	74.84 ±0.10
Ditto	67.36 ±0.13	80.40 ±0.08	80.34 ±0.03
FedPer	42.40 ±0.14	55.96 ±0.16	61.29 ±0.11
FedRep	62.69 ±0.04	76.96 ±0.03	76.95 ±0.03
FedRoD	69.56 ±0.09	82.00 ±0.06	82.04 ±0.05
FPFC	62.96 ±0.01	77.14 ±0.01	77.09 ±0.02
IFCA	69.78 ±0.02	82.16 ±0.01	82.19 ±0.01
pFedME	63.25 ±0.12	77.41 ±0.09	77.41 ±0.05
SuperFed	42.73 ±0.09	56.11 ±0.11	61.55 ±0.07
FedAMP	62.62 ±0.08	76.87 ±0.08	76.81 ±0.07
L2GD	62.72 ±0.11	76.96 ±0.09	76.89 ±0.06
pFedNet	66.13 ±0.03	79.55 ±0.02	79.64 ±0.02
FedABC	52.40 ±0.03	68.32 ±0.02	68.29 ±0.02
FedDiv	**70.56 ±0.01**	**82.71 ±0.01**	**82.68 ±0.04**

(d) $l = 3$

	BMIoU	BMDice	BMSens
FedAvg	69.39 ±0.05	81.88 ±0.03	81.85 ±0.01
FedProx	59.34 ±0.24	74.35 ±0.17	74.32 ±0.18
Ditto	66.71 ±0.22	79.89 ±0.16	80.00 ±0.15
FedPer	46.52 ±0.15	60.10 ±0.17	64.62 ±0.12
FedRep	63.27 ±0.03	77.32 ±0.03	77.40 ±0.02
FedRoD	70.08 ±0.09	82.37 ±0.06	**82.49 ±0.06**
FPFC	64.64 ±0.02	78.37 ±0.02	78.40 ±0.02
IFCA	69.52 ±0.02	81.98 ±0.02	82.02 ±0.06
pFedME	64.80 ±0.07	78.48 ±0.06	78.46 ±0.05
SuperFed	45.73 ±0.10	58.41 ±0.11	64.02 ±0.08
FedAMP	64.48 ±0.14	78.28 ±0.10	78.39 ±0.09
L2GD	64.47 ±0.00	78.28 ±0.00	78.40 ±0.02
pFedNet	65.81 ±0.01	79.31 ±0.01	79.38 ±0.03
FedABC	52.31 ±0.01	68.05 ±0.01	68.07 ±0.01
FedDiv	**70.23 ±0.06**	**82.46 ±0.04**	82.44 ±0.06

TABLE IV
RESULTS OF FOUR TARGET SEGMENTATION TASK
ON BRATS 2017

(a) $l = 0$

	BMIoU	BMDice	BMSpec	BMSens
FedAvg	32.69 ±0.04	44.34 ±0.03	85.82 ±0.02	49.48 ±0.03
FedProx	12.49 ±0.05	17.74 ±0.08	75.40 ±0.02	26.09 ±0.04
Ditto	31.82 ±0.07	43.44 ±0.10	85.47 ±0.07	48.49 ±0.17
FedPer	11.60 ±0.00	16.22 ±0.01	75.09 ±0.00	25.13 ±0.00
FedRep	28.64 ±0.01	40.34 ±0.02	83.92 ±0.01	44.70 ±0.03
FedRoD	33.54 ±0.04	45.56 ±0.04	**86.09 ±0.01**	**50.26 ±0.04**
FPFC	28.43 ±0.06	40.00 ±0.08	83.81 ±0.03	44.27 ±0.09
IFCA	32.80 ±0.07	44.46 ±0.08	85.88 ±0.03	49.65 ±0.09
pFedME	30.26 ±0.04	41.96 ±0.03	84.65 ±0.04	46.52 ±0.08
SuperFed	11.60 ±0.00	16.21 ±0.01	75.08 ±0.00	25.13 ±0.00
FedAMP	29.51 ±0.08	41.15 ±0.10	84.31 ±0.04	45.55 ±0.17
L2GD	29.28 ±0.06	40.96 ±0.05	84.26 ±0.02	45.63 ±0.04
pFedNet	32.34 ±0.08	43.82 ±0.10	85.64 ±0.05	48.58 ±0.19
FedABC	20.00 ±0.00	32.52 ±0.01	79.50 ±0.00	33.94 ±0.01
FedDiv	**34.06 ±0.12**	**47.66 ±0.09**	85.66 ±0.07	50.19 ±0.18

(b) $l = 1$

	BMIoU	BMDice	BMSpec	BMSens
FedAvg	32.89 ±0.02	44.58 ±0.04	85.82 ±0.02	49.06 ±0.10
FedProx	12.24 ±0.01	17.27 ±0.01	75.32 ±0.00	25.82 ±0.01
Ditto	31.57 ±0.05	43.18 ±0.01	85.23 ±0.04	47.52 ±0.22
FedPer	11.68 ±0.01	16.32 ±0.01	75.10 ±0.00	25.16 ±0.00
FedRep	28.82 ±0.01	40.56 ±0.01	83.89 ±0.00	44.60 ±0.01
FedRoD	33.74 ±0.01	45.94 ±0.03	**86.07 ±0.01**	50.20 ±0.04
FPFC	29.29 ±0.11	41.29 ±0.13	84.06 ±0.03	44.91 ±0.15
IFCA	32.94 ±0.05	44.64 ±0.06	85.88 ±0.03	49.29 ±0.09
pFedME	30.70 ±0.02	42.45 ±0.02	84.83 ±0.02	46.69 ±0.10
SuperFed	11.69 ±0.00	16.33 ±0.01	75.11 ±0.00	25.16 ±0.00
FedAMP	29.91 ±0.07	41.58 ±0.06	84.45 ±0.03	45.66 ±0.10
L2GD	29.85 ±0.07	41.54 ±0.09	84.43 ±0.05	45.67 ±0.19
pFedNet	32.37 ±0.08	43.91 ±0.08	85.66 ±0.04	48.56 ±0.08
FedABC	19.92 ±0.00	32.30 ±0.00	79.48 ±0.00	33.59 ±0.00
FedDiv	**35.91 ±0.12**	**50.46 ±0.12**	85.86 ±0.04	**51.14 ±0.07**

(c) $l = 2$

	BMIoU	BMDice	BMSpec	BMSens
FedAvg	32.45 ±0.08	44.10 ±0.10	85.67 ±0.05	48.89 ±0.07
FedProx	11.92 ±0.01	16.73 ±0.02	75.20 ±0.00	25.57 ±0.01
Ditto	31.77 ±0.02	43.48 ±0.03	85.40 ±0.06	48.40 ±0.23
FedPer	32.49 ±0.06	45.04 ±0.07	85.09 ±0.01	48.35 ±0.05
FedRep	29.82 ±0.04	41.52 ±0.04	84.47 ±0.01	45.96 ±0.06
FedRoD	32.95 ±0.06	44.81 ±0.05	85.94 ±0.03	49.66 ±0.10
FPFC	31.86 ±0.09	44.43 ±0.09	84.80 ±0.04	47.46 ±0.11
IFCA	32.67 ±0.02	44.37 ±0.02	85.79 ±0.02	49.25 ±0.13
pFedME	30.95 ±0.08	42.65 ±0.06	84.99 ±0.03	47.21 ±0.11
SuperFed	30.83 ±0.06	43.19 ±0.07	84.39 ±0.03	45.93 ±0.03
FedAMP	30.26 ±0.02	42.00 ±0.02	84.68 ±0.03	46.51 ±0.13
L2GD	30.38 ±0.08	42.11 ±0.05	84.74 ±0.04	46.72 ±0.10
pFedNet	31.39 ±0.01	42.87 ±0.01	85.20 ±0.02	47.40 ±0.06
FedABC	20.67 ±0.00	33.31 ±0.00	79.82 ±0.00	34.56 ±0.01
FedDiv	**35.62 ±0.12**	**49.90 ±0.06**	**86.01 ±0.04**	**51.82 ±0.09**

(d) $l = 3$

	BMIoU	BMDice	BMSpec	BMSens
FedAvg	32.54 ±0.05	44.21 ±0.04	85.73 ±0.01	48.87 ±0.08
FedProx	11.96 ±0.04	16.79 ±0.09	75.16 ±0.01	25.25 ±0.02
Ditto	33.51 ±0.06	45.06 ±0.06	86.26 ±0.04	50.06 ±0.04
FedPer	15.39 ±0.03	21.10 ±0.03	77.10 ±0.01	29.06 ±0.03
FedRep	33.29 ±0.07	45.69 ±0.06	85.42 ±0.03	48.99 ±0.04
FedRoD	35.03 ±0.01	46.77 ±0.02	86.79 ±0.01	51.45 ±0.04
FPFC	35.15 ±0.04	47.34 ±0.05	86.19 ±0.02	50.79 ±0.04
IFCA	32.39 ±0.01	44.12 ±0.03	85.66 ±0.03	48.90 ±0.15
pFedME	32.66 ±0.05	44.20 ±0.10	85.77 ±0.01	48.92 ±0.09
SuperFed	15.67 ±0.03	21.43 ±0.03	77.16 ±0.01	29.32 ±0.03
FedAMP	37.47 ±0.03	51.76 ±0.03	86.26 ±0.02	52.33 ±0.07
L2GD	37.25 ±0.13	51.60 ±0.20	86.14 ±0.03	52.03 ±0.20
pFedNet	31.49 ±0.06	42.96 ±0.06	85.35 ±0.03	47.65 ±0.06
FedABC	20.92 ±0.00	33.31 ±0.00	79.95 ±0.00	34.97 ±0.00
FedDiv	**38.71 ±0.13**	**52.93 ±0.12**	**86.94 ±0.05**	**54.22 ±0.14**