Evaluation and Analysis of Different Aggregation and Hyperparameter Selection Methods for Federated Brain Tumor Segmentation

Ece Isik-Polat[1](✉), Gorkem Polat[1], Altan Kocyigit[1],
and Alptekin Temizel[1,2]

[1] Graduate School of Informatics, Middle East Technical University, Ankara, Turkey
{eceisik,gorkem.polat,kocyigit,atemizel}@metu.edu.tr
[2] Neuroscience and Neurotechnology Center of Excellence, Ankara, Turkey

Abstract. Availability of large, diverse, and multi-national datasets is crucial for the development of effective and clinically applicable AI systems in the medical imaging domain. However, forming a global model by bringing these datasets together at a central location, comes along with various data privacy and ownership problems. To alleviate these problems, several recent studies focus on the federated learning paradigm, a distributed learning approach for decentralized data. Federated learning leverages all the available data without any need for sharing collaborators' data with each other or collecting them on a central server. Studies show that federated learning can provide competitive performance with conventional central training, while having a good generalization capability. In this work, we have investigated several federated learning approaches on the brain tumor segmentation problem. We explore different strategies for faster convergence and better performance which can also work on strong Non-IID cases.

Keywords: Federated learning · Collaborative learning · Brain tumor segmentation · Medical imaging

1 Introduction

Computer-aided approaches utilizing deep learning models have become prominent in the domain of medical image processing [18]. The amount and diversity of training data used to develop these models are important for model success and generalizability [25–27]. Currently, the inadequacy of medical data sources and labeled data have become a bottleneck and led to poor performance of the deep learning based solutions [30]. In order to overcome these issues, there are several initiatives to form diverse datasets to train reliable and robust models that have good generalization ability and clinical usability. EndoCV Challenges incorporates diverse endoscopy video frames from several institutions worldwide, including different modalities and organs to utilize deep learning methods to detect

A. Crimi and S. Bakas (Eds.): BrainLes 2021, LNCS 12963, pp. 405–419, 2022.
https://doi.org/10.1007/978-3-031-09002-8_36

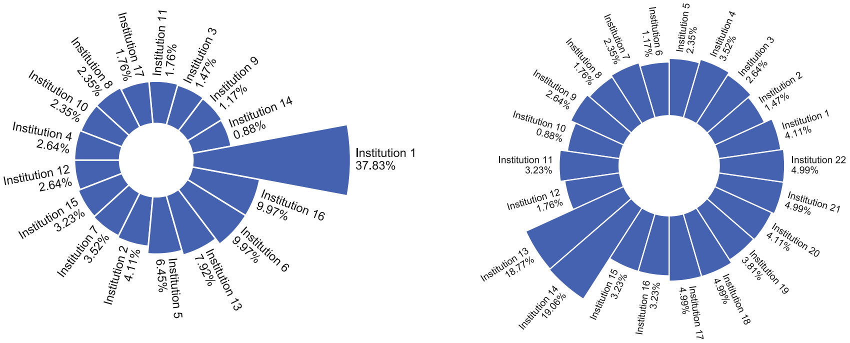
artifacts and diseases [1,2,21,22]. BraTS Challenges brings multi-institutional multi-parametric magnetic resonance imaging (mpMRI) scans for the analysis of brain tumors and the dataset has been continuously growing [6]. Although these initiatives are very important for reliable and clinical-ready models, they are not feasible to scale because it requires a tremendous work. First of all, it is difficult to represent whole distribution (e.g., minority and under-represented groups) as it requires healthy collaborations with many institutions and immense annotations. Secondly, data properties such as image modalities and resolutions are in a constant change that leads to distribution shift over time; therefore, collecting and processing all the data for once does not work either. Moreover, due to the data privacy regulations, collecting sensitive patient data from different institutions and hospitals is not always applicable. The federated learning (FL) concept offers a solution in such situations where data privacy and ownership are a problem by enabling collaborators train a common global model without disclosing their local data [15,19]. Several studies employing FL approaches in the medical domain have reported successful results [8,25,26]. These studies have drawn the attention of researchers into FL for medical imaging and made it a popular research field recently.

In this study, we propose various FL approaches for the Federated Tumor Segmentation (FeTS) Challenge [20]. For the Task-1 of the challenge, the participants are provided with an FL environment setup that is based on the OpenFL [24] framework and they are requested to develop strategies for the development of the methods in order to extract much of the knowledge from the collaborators. In this task, the participants are allowed to modify four functions: 1) custom performance metrics, 2) collaborator selection, 3) hyperparameter selection, and 4) custom aggregator. Our proposed methods took the 3rd place in the competition.

2 Related Work

Recently, the use of FL has been increasing in the medical field. In [11], Huang et al. proposed Loss-based Adaptive Boosting Federated Averaging (LoAdaBoost FedAvg) on critical care database data called as MIMIC-III [13]. In this method, the collaborators with higher losses than the previous round median loss are retrained before sending to the server for model aggregation. In [16], Li et al. have proposed a federated learning system for brain tumor segmentation on BraTS 2018 dataset [6] and have shown the trade-off between privacy protection costs and model performance. Similarly, in [26], Sheller et al. have compared federated learning and other data private collaborative learning approaches such as institutional incremental learning and cyclic institutional incremental learning on brain tumor segmentation task. This study has shown that FL can overcome institutional biases and form a global model that has better generalization where data amount and data diversity are inadequate. In [8], Dou et al. have used FL architecture to detect chest CT abnormalities in COVID-19 patients and showed that federated global model outperforms in terms of generalizability on external datasets better than individual models and their ensemble.

3 Data



(a) Natural split, *partitioning_1*.

(b) Artificial split, *partitioning_2*: 5 largest institutions are splitted further according to the tumor sizes.

Fig. 1. The data distribution in the training dataset splits.

The Federated Tumor Segmentation (FeTS) challenge 2021 is the first challenge in the federated medical imaging area. The challenge data set is composed of multi-institutional magnetic resonance images from the International Brain Tumor Segmentation (BraTS) challenge and other independent institutions in the FeTS initiative [3–5,20,24]. The training set contains 341 images, institution-based split of which is given in Fig. 1. The validation and the tests sets contain 111 and 166 images, respectively. The segmentation annotations of the challenge dataset were performed by annotators whose experience levels vary with respect to their clinical and academic backgrounds. Then, these annotations were approved by two experienced board-certified neuroradiologists with more than 12 years of experience [20].

4 Methods

4.1 Aggregator

In a real-life FL setting, the data distribution of collaborators is non independent identically distributed (non-IID) because collaborators may have different data distribution and the number of observations. The difference in device capabilities, user demographic information, or geographic location can be major reasons for the non-IIDness [14,19].

When collaborators have access to differing amounts of data and when they use the same number of epochs E in their local training, they would perform different numbers of local updates τ. If a collaborator has n_i samples, number of local gradient descent (GD) iterations is $\tau_i = En_i/B$, where B is the mini-batch size. In [29], Wang et al. have shown that the heterogeneity in collaborators' local progresses causes convergence to a stationary point of mismatched objective function, which is different from the true objective, when vanilla weighted averaging is used. Instead, they propose FedNova, a normalized averaging method that prevents bias toward clients performing more local updates. The shared global model is updated as in Eq. 1.

$$\boldsymbol{x}^{t+1} = \boldsymbol{x}^t - \tau_{\text{eff}} \sum_{i=1}^{m} p_i \frac{\varDelta_i^t}{\tau_i^t} \tag{1}$$

where p_i denotes the relative sample size of the collaborator i (i.e., $p_i = n_i/n$ where n is the total number of samples), $\tau_{\text{eff}} = \sum_{i=1}^{m} p_i \tau_i$, $\varDelta_i^t = \boldsymbol{x}^t - \boldsymbol{x}_i^{t+1}$, and m is the total number of collaborators. Since the number of samples n_i for collaborator i is directly proportional to the number of local iteration τ_i and the relative sample size p_i, this formula can be rewritten as in Eq. 2.

$$\tau_i = \frac{p_i n E}{B}$$

$$\tau_{\text{eff}} = \sum_{i=1}^{m} \frac{p_i^2 n E}{B}$$

$$\boldsymbol{x}^{t+1} = \boldsymbol{x}^t - \frac{nE}{B}(p_1^2 + p_2^2 + \dots + p_m^2) \sum_{i=1}^{m} \frac{B}{nE} \varDelta_i^t$$

$$\boldsymbol{x}^{t+1} = \boldsymbol{x}^t - \underbrace{(p_1^2 + p_2^2 + \dots + p_m^2)}_{constant} \sum_{i=1}^{m} \varDelta_i^t$$

$$\boldsymbol{x}^{t+1} = \boldsymbol{x}^t - \gamma \sum_{i=1}^{m} \varDelta_i^t \tag{2}$$

where γ refers to the aggregator learning rate, which can be increased or decreased according to FL training needs. As given in Eq. 2, FedNova corresponds to a uniform averaging with adjustable step size (or learning rate) on the aggregator. FedNova aims to prevent exacerbation of client drifts caused by relative sample sizes p_i. When there is a significant difference between the number of samples in the collaborators, as in the FeTS Challenge dataset, FedAvg creates a bias toward the collaborators having more samples (Fig. 2). Although validation set (named Val-1 in Sect. 5) results reported during the training may seem good as its data distribution directly comes from the training set, out-of-distribution performance results may not be satisfactory. Wang et al. [29] have shown that FedNova generally achieves 6–9% higher accuracy than FedAvg on a non-IID version of CIFAR-10 dataset.

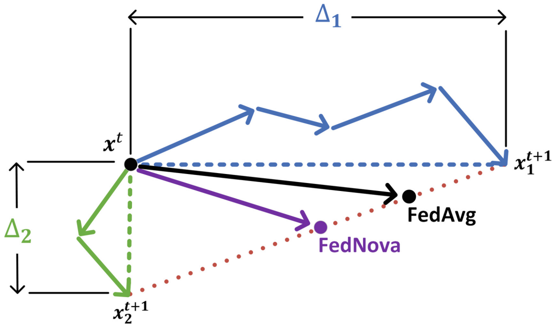


Fig. 2. Naive weighted averaging (FedAvg) creates bias toward collaborators having higher number of samples, which may adversely affect the out-of-distribution performance. On the other hand, FedNova gives equal weights to all collaborators acting as a regularizer.

Another approach to deal with convergence issues when collaborators' data distribution is non-IID is Federated Averaging with server momentum (FedAvgM). The momentum on top of the Stochastic Gradient Descent (SGD) has proven to provide a significant success in accelerating the training and dampening the oscillations [9]. In [10], Hsu et al. have shown that as the level of non-IIDness increases, the performance of the FedAvgM stays relatively constant while federated averaging falls rapidly. Moreover, [23] has shown the improved effect of adaptive optimizers such as Adam and RMSProp, which are based on momentum, on top of the federated averaging.

In FedAvgM, the average of the gradients are added to the accumulated gradient which is multiplied by a β parameter to adjust effect of the momentum as shown in Eq. 3. Then this weighted accumulated gradient is used to update weights of the current communication round as in Eq. 4. Here, an aggregator learning rate γ can be used to adjust the step size on the server (in our experiments β is chosen as 0.9 and γ is chosen as 1).

$$\boldsymbol{\Delta w}^{t+1} = \sum_{i=1}^{m} p_i \boldsymbol{\Delta w_i}^{t+1}$$

$$\boldsymbol{v}^{t+1} = \beta \boldsymbol{v}^t + \boldsymbol{\Delta w}^{t+1} \tag{3}$$

$$\boldsymbol{w}^{t+1} = \boldsymbol{w}^t - \gamma \boldsymbol{v}^{t+1} \tag{4}$$

where p_i denotes the relative sample size of the collaborator i (i.e., $p_i = n_i/n$ where n is the total number of samples), and m is the total number of collaborators.

Along with FedNova and FedAvgM, other aggregator functions (Table 1) have been implemented and experimented in the FeTS Challenge. However, in this article, only the results for FedNova and FedAvgM are presented. Please visit https://github.com/eceisik/FeTS_Challenge_METU_FL_Team to see all implemented methods by the METU FL Team.

Table 1. The list of other aggregator methods implemented.

Function name	Explanation
Make aggregation with improved nodes	All collaborators participate in each round of FL, but only those that have improved validation scores participate in the main model aggregation
Coordinate-wise median aggregation	Main model weights are determined by taking the median of the collaborators weights. Median is more robust to the outliers and extreme values than mean

4.2 Collaborator Selection

How to choose collaborators that will take part in each round is another important dimension of the FeTS Challenge. We used "all_collaborators_train" as a collaborator choice function and all collaborators participated in each FL round.

We implemented two alternative collaborator choice functions given in Table 2. If the focus is on the convergence time metric, the method called as "choose random nodes with faster ones" could be more preferable. This method does not introduce any extra communication delays, because once a random collaborator is selected, only those that are faster than the selected one participate in the training for the FL round (i.e., selected collaborator creates an upper bound for the other selected collaborators in terms of time). Although the number of collaborators participate in each round varies, the working mechanism tends to favor the fastest collaborators. Being fast, in this case, depends on two factors namely the amount of available computation/communication resources and the number of samples in a collaborator. On the other hand, the institutions having fewer patient images may be over represented, which is a disadvantage of this method.

4.3 Hyperparameter Selection

For the hyperparameter selection, an adaption of AdaComm [28] with a learning rate scheduling scheme is used. AdaComm [28] is an adaptive communication strategy that saves communication delay and enables fast convergence by federated averaging less frequently in early training rounds and later increasing communication frequency. In [28], experimental converge analysis was examined on wall-clock time instead of communication round. It is shown that using more local updates in the early rounds of training resulted in a faster decrease in loss but also a higher error. For this reason, it starts with a large number of updates per round and gradually decreases as the model starts to converge.

In the original version of AdaComm, the method is based on the number of local updates in an IID setting. However, in the challenge, the data distribution is extremely uneven. While Institute-1 has 37.83% of the data, Institute-14 has 0.88% of the whole training data (Fig. 1). Using the same number of local updates for each collaborator could potentially cause over-representation of some

Table 2. The list of other collaborator choice methods implemented.

Function name	Explanation
Choose random nodes with faster ones	For the first round, all collaborators participate in the training and round time statistics are recorded. After the first round, a random collaborator is selected and participates in the current FL round with other collaborators that are faster than itself
Random collaborators train	A random subset of collaborators are selected for each FL round

small data provider institutions. By considering the non-IID nature of the data distribution, our aggregation method mechanism, and the fact that the number of local updates is directly proportional to the number of epochs, we adapted this method based on the decaying number of epoch (AdaptiveEpoch). Basically, the number of epochs per round at each FL round decays according to the relative difference between the initial loss and current round loss as stated in Eq. 5.

$$E_t = \left\lceil \sqrt{\frac{F(x_{T=t})}{F(x_{T=0})}} E_0 \right\rceil \tag{5}$$

where T denotes the number of FL rounds, t denotes the round number, E_t denotes the number of epochs at a given round t, and $F(x)$ is the objective function with respect to model parameters denoted by x.

Learning rate scheduling is a commonly used technique to train deep neural networks in a centralized manner [9]. Studies show that learning rate scheduling is also necessary for FedAvg to converge to an optimum point of loss function [17]. However, there are many strategies for scheduling and there is no benchmark for their performances. In this study, we have adopted *decay learning rate on plateau* approach. This strategy brings two new parameters namely *patience* and *decay factor*. In our implementation, learning rate scheduling tracks the target performance metric, which is the mean Dice score for ET, TC and WT labels, and if there is no improvement on the target performance metric for a *patience* number of round, the learning rate is updated by scaling with the *decay factor*. Experiments show that learning rate scheduling provides faster convergence, more relaxed learning rate selection, higher convergence score, and reduced oscillations when training converges [9]. The list of hyperparameter selection methods are given in Table 3. For AdaptiveEpoch initial epoch E_0 is set to 8; for the LR scheduling, the initial LR is set to 0.0002 and patience is set as 15. For the constant hyperparameters, default values were used (LR = 0.00005, epoch per round = 1).

Table 3. The list of other hyperparameter selection methods.

Function name	Explanation
Constant hyperparameters	Use fixed hyperparameters for each FL round
LRScheduling hyperparameters	Learning rate decays according to the value of average Dice score with given patience scheme
AdaptiveEpoch	The number of epochs per round decays according to the decrease in the initial loss

5 Experimental Results and Discussion

Before the FL training, the training dataset is split into train and validation sets as 80% and 20%, respectively. The performance results of the aggregated and individual models on validation sets are logged at each FL round (it is integrated with the FeTS Challenge source code). Unless otherwise stated, all reported performance metrics and loss graphs belong to this validation set of *partitioning_2.csv*. The mean Dice score refers to the average of Dice scores of ET, TC, and WT labels.

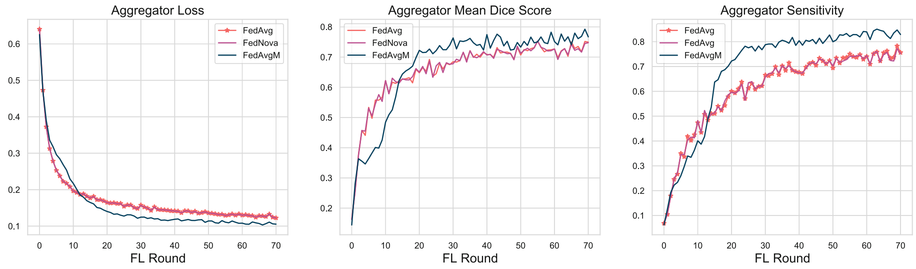


Fig. 3. The performance comparison of FedAvg, FedNova, and FedAvgM.

Figure 3 shows the performances of FedAvg, FedNova, and FedAvgM on aggregator mean dice score, aggregator loss, and aggregator sensitivity metrics. Since medical datasets may contain institutional biases [26] and FedAvg have an undesirable effect of favoring these biases, FedNova is expected to have better performance on the non-training sets. However, since samples of institutions' distributions of training and validation sets are similar to each other, we observe nearly identical performance for both FedAvg and FedNova. Yet, models built with FedNova are expected to have better inferences on the out-of-distribution dataset [29], and as such, they are expected to be more suitable for real-life use-case scenarios. On the other hand, FedAvgM outperforms both FedAvg and FedNova on all metrics. Therefore, we have preferred FedAvgM as the aggregator method in the FeTS challenge.

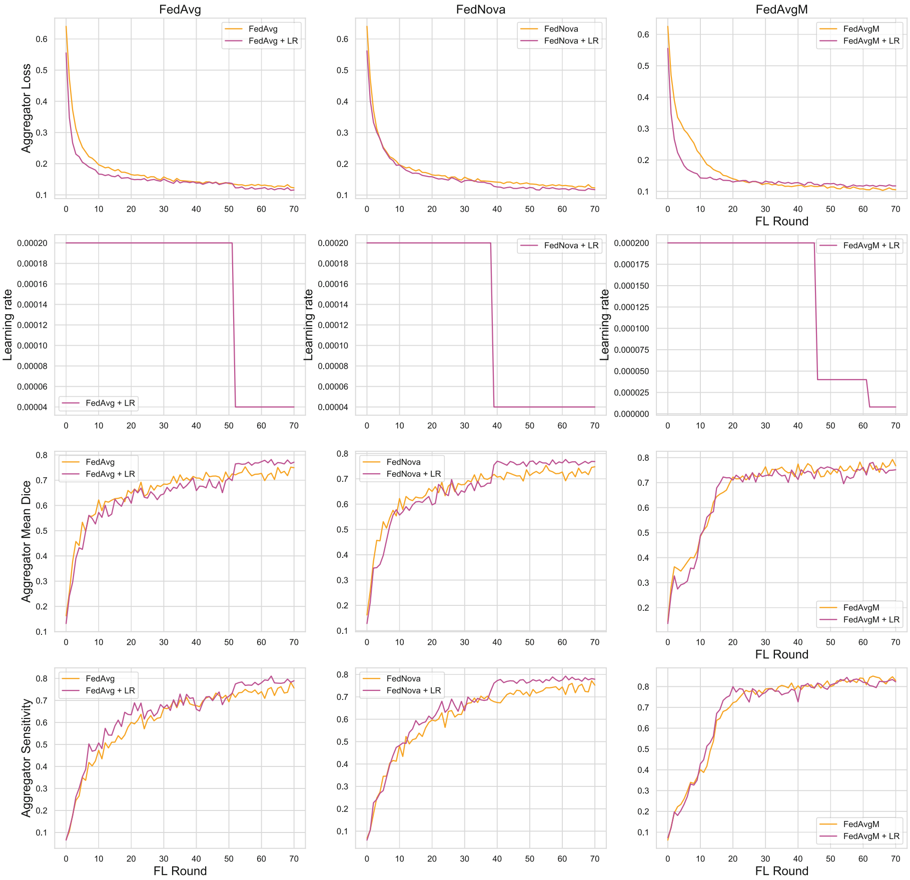


Fig. 4. The impacts of LR scheduling on FedAvg, FedNova, and FedAvgM.

Figure 4 shows the effect of the LR scheduling approach on each aggregation method. For both FedAvg and FedNova, it can be observed on both loss function and performance metrics that LR scheduling has an evident effect on their performances. In particular, a sharp increase on performance metrics occurs when the LR is decayed. On the other hand, LR scheduling has no improvement on FedAvgM. One possible reason might be since FedAvgM converges much faster than the FedAvg and FedNova, it may have directly reached the optimum region where it does not need any scheduling. However, it should be noted that we have used fixed values for starting learning rate, decay rate, and patience parameters; therefore, more experiments with different set of values should be performed to make a comment on effect of LR scheduling on FedAvgM.

AdaptiveEpoch helps training converge in fewer rounds with higher performance due to having more local epochs than using the constant hyperparameters as seen in Fig. 5. The AdaptiveEpoch method improves the performance of both FedAvg, FedNova, and FedAvgM on aggregator loss, aggregator mean dice score, and aggregator sensitivity metrics. The improvement achieved on aggregator methods by AdaptiveEpoch is much more significant than the LR scheduling. The performance increase can be observed both on loss and performance metrics.

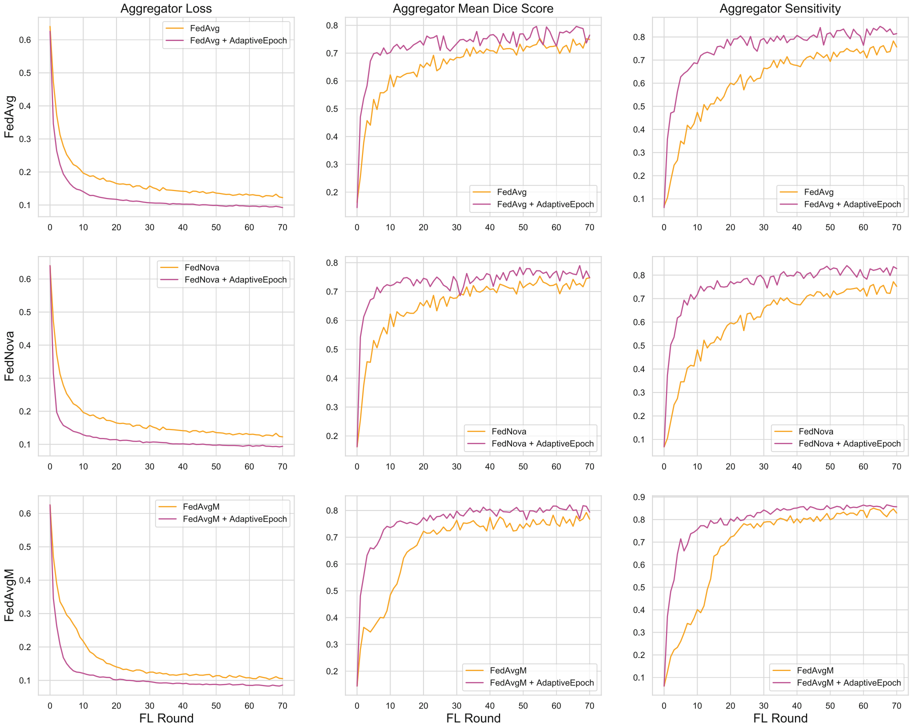


Fig. 5. The impacts of adaptive epoch on FedAvg, FedNova, and FedAvgM.

Figure 6 shows the performance comparison of different hyperparameter strategies on FedAvgM. Accordingly, LR scheduling, AdaptiveEpoch, and AdaptiveEpoch+LR scheduling improves the baseline model performance. AdaptiveEpoch and AdaptiveEpoch+LR scheduling provides faster convergence than LR scheduling. However, there is no significant difference between AdaptiveEpoch and AdaptiveEpoch+LR scheduling. Due to the time and resource constraints, the number of FL round was set to 70 for all experiments, which in turn limited the effect of LR scheduling and AdaptiveEpoch+LR scheduling due to incomplete decaying of LR.

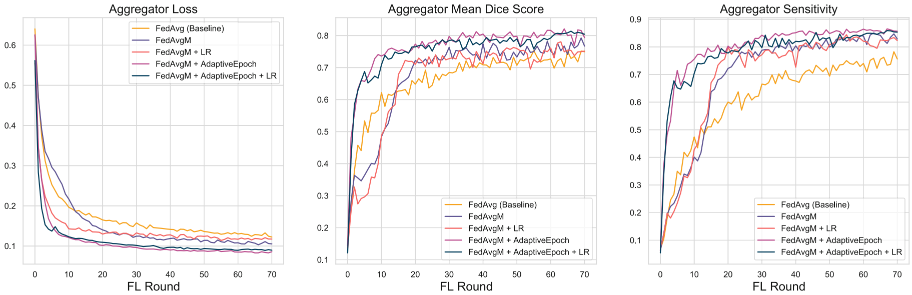


Fig. 6. The impacts of hyperparameter setting strategies on FedAvgM

Table 4 shows the mean dice score and convergence score obtained on the validation set. These experiments are performed by using *partitioning-2* as data split. The convergence score is computed as the area under the validation learning curve where the horizontal axis is the runtime, and the vertical axis is the performance. Most of the time, FedAvgM outperforms others and achieves the best mean Dice score and the convergence score for all hyperparameter choice strategies except for LR scheduling. It is expected and in line with the results that are presented in Fig. 4. Nevertheless, the convergence score is based on the validation set reported during the FL training; therefore, the comparison of convergence scores on an out-of-distribution set is still an open question.

Table 4. The mean Dice score and convergence scores on the validation set.

	Validation mean dice			Convergence score		
	FedAvg	FedNova	FedAvgM	FedAvg	FedNova	FedAvgM
Constant Hyperparameters	0.753	0.753	**0.793**	0.770	0.770	**0.788**
LR Scheduling	**0.782**	0.776	0.781	**0.761**	0.759	0.744
AdaptiveEpoch	0.796	0.790	**0.821**	0.797	0.797	**0.804**
AdaptiveEpoch+LR Scheduling	0.783	0.787	**0.814**	0.781	0.785	**0.795**

Table 5 presents the results of the our challenge submission on the challenge test set with convergence score of 0.770. The results are provided by the FeTS initiative.

Table 5. The scores were obtained on Leader Board 2 of Task 1 that our team (METU FL) won the 3^{rd} rank in the FeTS Challenge.

	μ	σ	Q_1	Q_2	Q_3
Dice_ET	0.719	0.268	0.676	0.811	0.887
Dice_WT	0.794	0.228	0.807	0.882	0.917
Dice_TC	0.741	0.286	0.674	0.870	0.931
Sensitivity_ET	0.817	0.277	0.812	0.930	0.974
Sensitivity_WT	0.860	0.236	0.869	0.942	0.977
Sensitivity_TC	0.832	0.278	0.861	0.945	0.981
Specificity_ET	0.999	0.002	0.999	0.999	0.999
Specificity_WT	0.998	0.004	0.998	0.998	0.999
Specificity_TC	0.999	0.003	0.999	0.999	0.999
Hausdorff95_ET	34.430	96.022	1.414	2.449	9.027
Hausdorff95_WT	19.199	57.799	3.162	5.431	10.355
Hausdorff95_TC	35.288	91.651	2.236	6.240	17.051

6 Conclusion

In this study, we perform comprehensive experiments to compare different hyperparameter selection strategies and aggregation methods. The experiments reveal that FedAvgM has better performance than FedAvg and FedNova. Moreover, it is shown that the AdaptiveEpoch approach provides performance increase and faster convergence. However, LR scheduling is not effective with FedAvgM or AdaptiveEpoch. Therefore, it can be said that methods that work well individually may not work well together when combined, or one can reduce the effectiveness of the other. For instance, while AdaptiveEpoch results in better validation mean dice scores and convergence scores than using constant hyperparameter strategy, when it is combined with LR scheduling, all mean dice and convergence scores get worse for all aggregation methods (see Table 4).

During the experiments, all collaborators have participated in the local training process for all rounds. Instead, collaborator choosing methods such as clustering collaborators based on the update similarity or increasing the likelihood of being chosen collaborators that improved the performance for the random collaborator choice can be utilized to improve performance.

Moreover, in the medical image domain, there is generally high interobserver variability in annotations, which can be considered label noise. For example, if an institution's label quality is low, the model coming from that institution will adversely affect the global model; therefore, weights coming from that institution should be handled carefully. There are defense mechanisms such as KRUM [7], BARFED [12], or trimmed mean [31] that can overcome the attacks in federated learning to some extent. These defense strategies may be used to overcome the label noise.

7 GPU Training Times

Computation time and cost, as well as energy consumption, are important factors determining the direction of future research and adoption of the technology in real life. Table 6 shows the detailed GPU training times of the experiments that are run on single NVIDIA A100-80GB GPU. LR scheduling has no significant effect on the training times. On the other hand, although AdaptiveEpoch strategy brings an increase in performance metrics, its usage nearly doubles the total training time due to longer round times.

Table 6. The detailed GPU training times (hour).

	FedAvg	FedNova	FedAvgM	Total
Constant Hyperparameters	36.7	36.4	34.8	**107.9**
LR Scheduling	35.1	36.7	36.8	**108.6**
AdaptiveEpoch	67.2	69.0	68.9	**205.1**
AdaptiveEpoch+LR Scheduling	65.5	65.8	66.4	**197.7**
Total	**204.5**	**207.9**	**206.9**	**619.3**

Acknowledgment. This work has been supported by Middle East Technical University Scientific Research Projects Coordination Unit under grant number GAP-704-2020-10071. The numerical calculations reported in this paper were performed using TUBITAK ULAKBIM, High Performance and Grid Computing Center (TRUBA resources).

References

1. Ali, S., et al.: Deep learning for detection and segmentation of artefact and disease instances in gastrointestinal endoscopy. Med. Image Anal. **70**, 102002 (2021)
2. Ali, S., et al.: An objective comparison of detection and segmentation algorithms for artefacts in clinical endoscopy. Sci. Rep. **10**(1), 1–15 (2020)
3. Bakas, S., et al.: Segmentation labels and radiomic features for the pre-operative scans of the TCGA-GBM collection. The cancer imaging archive. Nat. Sci. Data **4**, 170117 (2017)
4. Bakas, S., et al.: Segmentation labels and radiomic features for the pre-operative scans of the TCGA-LGG collection. Cancer Imaging Archive **286** (2017)
5. Bakas, S., et al.: Advancing the cancer genome atlas glioma MRI collections with expert segmentation labels and radiomic features. Sci. Data **4**(1), 1–13 (2017)
6. Bakas, S., et al.: Identifying the best machine learning algorithms for brain tumor segmentation, progression assessment, and overall survival prediction in the brats challenge. arXiv preprint arXiv:1811.02629 (2018)
7. Blanchard, P., El Mhamdi, E.M., Guerraoui, R., Stainer, J.: Machine learning with adversaries: byzantine tolerant gradient descent. In: Proceedings of the 31st International Conference on Neural Information Processing Systems, pp. 118–128 (2017)

8. Dou, Q., et al.: Federated deep learning for detecting COVID-19 lung abnormalities in CT: a privacy-preserving multinational validation study. NPJ Digital Med. **4**(1), 1–11 (2021)
9. Goodfellow, I., Bengio, Y., Courville, A.: Deep Learning. MIT Press, Cambridge (2016). http://www.deeplearningbook.org
10. Hsu, T.M.H., Qi, H., Brown, M.: Measuring the effects of non-identical data distribution for federated visual classification. arXiv preprint arXiv:1909.06335 (2019)
11. Huang, L., Yin, Y., Fu, Z., Zhang, S., Deng, H., Liu, D.: LoAdaBoost: loss-based AdaBoost federated machine learning with reduced computational complexity on IID and non-IID intensive care data. PLoS ONE **15**(4), e0230706 (2020)
12. Isik-Polat, E., Polat, G., Kocyigit, A.: BARFED: byzantine attack-resistant federated averaging based on outlier elimination. arXiv preprint arXiv:2111.04550 (2021)
13. Johnson, A.E., et al.: MIMIC-III, a freely accessible critical care database. Sci. Data **3**(1), 1–9 (2016)
14. Kairouz, P., et al.: Advances and open problems in federated learning. arXiv preprint arXiv:1912.04977 (2019)
15. Li, T., Sahu, A.K., Talwalkar, A., Smith, V.: Federated learning: challenges, methods, and future directions. IEEE Signal Process. Mag. **37**(3), 50–60 (2020)
16. Li, W., et al.: Privacy-preserving federated brain tumour segmentation. In: Suk, H.-I., Liu, M., Yan, P., Lian, C. (eds.) MLMI 2019. LNCS, vol. 11861, pp. 133–141. Springer, Cham (2019). https://doi.org/10.1007/978-3-030-32692-0_16
17. Li, X., Huang, K., Yang, W., Wang, S., Zhang, Z.: On the convergence of FedAvg on non-IID data. arXiv preprint arXiv:1907.02189 (2019)
18. Litjens, G., et al.: A survey on deep learning in medical image analysis. Med. Image Anal. **42**, 60–88 (2017)
19. McMahan, B., Moore, E., Ramage, D., Hampson, S., Arcas, B.A.: Communication-efficient learning of deep networks from decentralized data. In: Artificial Intelligence and Statistics, pp. 1273–1282. PMLR (2017)
20. Pati, S., et al.: The federated tumor segmentation (FeTS) challenge (2021)
21. Polat, G., Isik Polat, E., Kayabay, K., Temizel, A.: Polyp detection in colonoscopy images using deep learning and bootstrap aggregation. In: Proceedings of the 3rd International Workshop and Challenge on Computer Vision in Endoscopy (EndoCV 2021) @ ISBI, vol. 2886, pp. 90–100 (2021)
22. Polat, G., Sen, D., Inci, A., Temizel, A.: Endoscopic artefact detection with ensemble of deep neural networks and false positive elimination. In: EndoCV@ ISBI, pp. 8–12 (2020)
23. Reddi, S.J., et al.: Adaptive federated optimization. In: International Conference on Learning Representations (2021). https://openreview.net/forum?id=LkFG3lB13U5
24. Reina, G.A., et al.: OpenFL: an open-source framework for federated learning. arXiv preprint arXiv:2105.06413 (2021)
25. Rieke, N., et al.: The future of digital health with federated learning. NPJ Digital Med. **3**(1), 1–7 (2020)
26. Sheller, M.J., et al.: Federated learning in medicine: facilitating multi-institutional collaborations without sharing patient data. Sci. Rep. **10**(1), 1–12 (2020)
27. Sun, C., Shrivastava, A., Singh, S., Gupta, A.: Revisiting unreasonable effectiveness of data in deep learning era. In: Proceedings of the IEEE International Conference on Computer Vision (ICCV), October 2017

28. Wang, J., Joshi, G.: Adaptive communication strategies to achieve the best error-runtime trade-off in local-update sgd. In: Talwalkar, A., Smith, V., Zaharia, M. (eds.) Proceedings of Machine Learning and Systems, vol. 1, pp. 212–229 (2019). https://proceedings.mlsys.org/paper/2019/file/c8ffe9a587b126f152ed3d89a146b445-Paper.pdf
29. Wang, J., Liu, Q., Liang, H., Joshi, G., Poor, H.V.: Tackling the objective inconsistency problem in heterogeneous federated optimization. In: Advances in Neural Information Processing Systems 33 (2020)
30. Yang, Q., Liu, Y., Chen, T., Tong, Y.: Federated machine learning: concept and applications. ACM Trans. Intell. Syst. Technol. (TIST) **10**(2), 1–19 (2019)
31. Yin, D., Chen, Y., Kannan, R., Bartlett, P.: Byzantine-robust distributed learning: towards optimal statistical rates. In: International Conference on Machine Learning, pp. 5650–5659. PMLR (2018)