Towards the Practical Utility of Federated Learning in the Medical Domain

Hyeonji Hwang* LOCALH@KAIST.AC.KR
KAIST, Republic of Korea

Seongjun Yang*† SEONGJUNYANG@KRAFTON.COM
KRAFTON, Republic of Korea

Daeyoung Kim DAEYOUNG.K@KAIST.AC.KR
KAIST, Republic of Korea

Radhika Dua† RADHIKADUA@GOOGLE.COM
Google Research, India

Jong-Yeup Kim JYKIM@KYUH.AC.KR
College of Medicine, Konyang University, Republic of Korea

Eunho Yang EUNHOY@KAIST.AC.KR
Edward Choi EDWARDCHOI@KAIST.AC.KR
KAIST, Republic of Korea

Abstract

Federated learning (FL) is an active area of research. One of the most suitable areas for adopting FL is the medical domain, where patient privacy must be respected. Previous research, however, does not provide a practical guide to applying FL in the medical domain. We propose empirical benchmarks and experimental settings for three representative medical datasets with different modalities: longitudinal electronic health records, skin cancer images, and electrocardiogram signals. The likely users of FL such as medical institutions and IT companies can take these benchmarks as guides for adopting FL and minimize their trial and error. For each dataset, each client data is from a different source to preserve real-world heterogeneity. We evaluate six FL algorithms designed for addressing data heterogeneity among clients, and a hybrid algorithm combining the strengths of two representative FL algorithms. Based on experiment results from three modalities, we discover that simple FL algorithms tend to outperform more sophisticated ones, while the hybrid algorithm consistently shows good, if not the best performance. We also find that a frequent global model update leads to better performance under a fixed training iteration budget. As the number of participating clients increases, higher cost is incurred due to increased IT administrators and GPUs, but the performance consistently increases. We expect future users will refer to these empirical benchmarks to design the FL experiments in the medical domain considering their clinical tasks and obtain stronger performance with lower costs.

Data and Code Availability Every detail of the data availability and code for every experiment in this study is stated on the official repository. [1] For eICU dataset, completing the CITI "Data or Specimens Only Research" course and a formal request are necessary.

1. Introduction

Federated learning (FL) is a distributed machine learning framework in which each client does not share its data but instead shares model parameters, thus preserving data privacy. FL is divided into cross-device and cross-silo FL (Kairouz et al., 2021; Wang et al., 2021). Typically, mobile devices are the clients in a cross-device setting, and hence the Internet connectivity and the efficiency of training in each device are the critical factors. On the other hand, the well-known example of the cross-silo FL is in the medical

* These authors contributed equally
† Work done at KAIST

1. https://github.com/wns823/medical_federated

domain where medical institutions are participating. There are relatively fewer clients and the Internet connectivity is not critical due to LAN connections (Kairouz et al., 2021). Recently, researchers have begun to test FL on medical datasets with varying success (Pfohl et al., 2019; Brisimi et al., 2018; Dou et al., 2021; Boughorbel et al., 2019; Liu et al., 2018; Sheller et al., 2020). However, existing studies do not fully consider the real-world situation of adopting FL in the medical domain.

The main stakeholders will be hospitals and IT service providers. Hospitals want to take advantage of machine learning models using their large data for various reasons; to use the models for clinical decision-making or to increase efficiency. FL enables multiple distributed data holders to collaboratively train a shared model and get generalizable AI models (Peng et al., 2022). The hospital can take advantage of the knowledge from other hospitals, especially for unseen data. Large IT companies or startups are not able to reach the medical dataset due to privacy issues. They can develop practical machine learning models with real patient records via FL. Achieving high performance at a low monetary cost is an important concern of these likely users.

However, applying FL involves a lot of considerations other than a model architecture or data construction due to its complicated process (i.e. local training and model aggregation). Moreover, it is more expensive than single-site learning to change the settings as there are many clients related and communication rounds are needed to get a satisfactory result. The first consideration is that they have to choose an appropriate FL algorithm. There are various modalities of data from the medical institutions: structured, image, or signal data. A suitable algorithm for each modality may vary and this area has not been fully explored. In addition to choosing which algorithm to use, there are still many other decisions waiting for them to make; the number of training epochs in each client server, the type of normalization technique to use, and the number of participating clients. They should also consider the approximate power consumption of the FL framework because it is directly related to the monetary cost. The guidance for these FL settings will decrease the expense of trial and error and help the users to concentrate on better model architecture or data strategy.

In this work, for the first time, we test well-known FL algorithms on three representative real-world medical datasets with different modalities involving structured (i.e., tabular), visual, and signal data. To be more realistic, each dataset is from a different source respectively so the data distributions preserve real-world heterogeneity. We select FL algorithms designed for heterogeneous data distributions among clients to observe whether they solve data heterogeneity generated by real-world healthcare applications. We provide practical benchmarks including the normalization techniques and the number of local training epochs. We also evaluate the performance of each FL algorithm in terms of monetary costs, such as power consumption and the number of participating clients. We combine two FL algorithms and test the hybrid method (`FedPxN`) on all three settings. `FedPxN` constantly shows comparable or better performance compared to other methods. We recommend using this version to minimize the expense of trial and error in choosing an FL algorithm.

2. Related Works

FL algorithms on data that are not independent and identically distributed (i.i.d.) The main challenge of FL in the medical domain is a non-i.i.d. problem (Rieke et al., 2020; Li et al., 2022) because of factors such as different specific protocols, medical devices, and local demographics. FedAvg (McMahan et al., 2017) is a widely known framework in FL but does not ensure training convergence when data are heterogeneous over local clients (Li et al., 2019; Hsu et al., 2019). Therefore, a large number of methods focus on addressing the non-i.i.d. problem. FedProx (Li et al., 2020) adds a proximal term in the local objective of the FedAvg framework to solve the heterogeneity problem, and FedOpt (Reddi et al., 2020) applies adaptive optimization in global aggregation to stabilize convergence for heterogeneous data. The inconsistency of local and global objectives due to the data heterogeneity of each local client is handled in Scaffold (Karimireddy et al., 2020) and FedDyn (Acar et al., 2021). Scaffold computes and aggregates control variates, whereas FedDyn uses a dynamic regularizer to solve the problem. Most FL methods (McMahan et al., 2017; Li et al., 2020; Reddi et al., 2020; Karimireddy et al., 2020; Acar et al., 2021) are validated in a label-heterogeneous experimental setting by partitioning the same sourced dataset into multiple clients. In contrast, FedBN (Li et al., 2021) considered the non-i.i.d problem that can occur due to feature shifts in different data

sources, and proposed to aggregate local models without batch normalization layers to handle this problem. Similarly, SiloBN (Andreux et al., 2020) aggregates local models without local batch normalization statistics. FedDAR (Zhong et al., 2023) decouples domain-specific prediction heads and a shared encoder in order to tackle a non-i.i.d setting where there is a similarity between each domain.

FL in Healthcare The medical domain is an active area of FL nowadays because of the importance of patient privacy. Prior study on FL in the medical domain (Lee and Shin, 2020) has achieved performance comparable to that of centralized learning. However, these studies focused only on electronic health records (EHR), and electrocardiogram (ECG) signals. Moreover, they randomly sample the data to create heterogeneity. However, in reality, heterogeneity also exists in the feature space. In our work, we focus on diverse and important modalities in healthcare (images, EHR, and ECG) and also consider a realistic scenario in which the clients are from different hospitals. Effective FL frameworks for specific tasks have been empirically demonstrated for each specific modality such as EHR and medical images in (Huang et al., 2019; Kim et al., 2017; Liu et al., 2019; Park et al., 2021; Xu et al., 2020). However, these results depended on architecture or were validated only on specific modalities.

Normalization layer for FL In (Diao et al., 2020; Hsieh et al., 2020), the authors compared performance when using various normalization layers in the local model for image classification tasks. Our study observes which normalization is effective in eight clinical tasks.

3. Methods

FL methods are often designed to minimize the weighted average of local objective function of clients as follows:

$$\min_{w} F(w) := \sum_{k=1}^{K} \frac{n_k}{n} F_k(w), \tag{1}$$

where K is the number of clients, n_k the number of examples in each client k, n the total data size of all clients, and F_k the local objective function of each client k. Data heterogeneity is a key and common challenge in solving Eq. 1. In this section, we propose a hybrid FL method (`FedPxN`) based on the analysis of the different training patterns of the parameters of each client in FedBN (Li et al., 2021) and deployment of the proximal term (Li et al., 2020).

3.1. FedAvg

FedAvg (McMahan et al., 2017) is the de facto standard algorithm in FL (Appendix Algorithm 1). In this framework, first, the central server sends the global model w_t to the clients in each communication round t. Then, each client k sends the updated model $w_{t,k}$ back to the server after local training. Next, the central server averages all client models considering the data size ratio n_k/n. Also, FedAvg reduces the number of communication rounds required for model convergence by updating the model after multiple epochs E of local training. Every baseline of our work is based on FedAvg, so we use the notation used in FedAvg for describing other methods.

3.2. FedProx

To solve the non-i.i.d problem, FedProx (Li et al., 2020) introduces an L_2 regularization term $\|w_{t,k} - w_t\|_2^2$ to the local objective function F_k of the FedAvg framework as follows:

$$\tilde{F}_k(w_{t,k}; b) = F_k(w_{t,k}; b) + \frac{\mu}{2}\|w_{t,k} - w_t\|_2^2, \tag{2}$$

where F is the objective function and μ is a hyperparameter that controls the degree of regularization. The local model updates are restricted by the regularization term so that they are closer to the global model.

3.3. FedBN

In FedBN (Li et al., 2021), the local models are aggregated without batch normalization layers in order to handle feature shifts among clients. The study evaluated FedBN's effectiveness through experiments using datasets from different sources. In our study, we use different types of normalization (i.e., batch normalization (BN) (Ioffe and Szegedy, 2015), group normalization (GN) (Wu and He, 2018), and layer normalization (LN) (Ba et al., 2016) for each task to maximize the performance of the models. Hence, we extend FedBN in such a way that all layers of the local models except the normalization layers are aggregated.

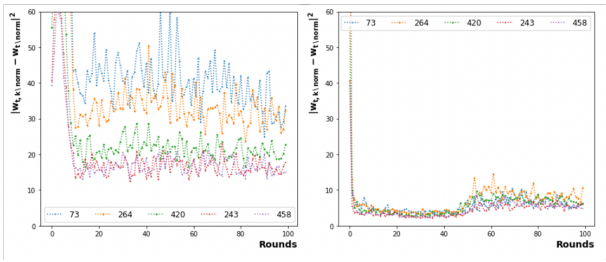
3.4. Hybrid Algorithm: FedPxN

Because the statistics used in the normalization layers are different in each local client $w_{t,k}$, it could be challenging for the aggregated model w_t to capture the distribution of data collected from different sources. A simple solution is to aggregate the local client models without the normalization layers, as in FedBN.

The normalization layers in FedBN naturally move towards their own local optimum depending on the data distribution of each client. However, we suspect that the other non-normalization layers might also move towards the local optimum inconsistent with the global optimum because of the effect of the normalization layers in the model of each client. Therefore, we measure the L_2 distance between all $w_{t,k\backslash norm}$, and $w_{t\backslash norm}$, where the former are the local model parameters except for the normalization layers in client k, and the latter are the global model parameters except the normalization layers in each round t, as follows:

$$\|w_{t,k\backslash norm} - w_{t\backslash norm}\|_2^2, \tag{3}$$

where t indicates each communication round, k denotes each local client. As shown on Figure 1a, each client's model $w_{t,k\backslash norm}$ has evolved in different directions from the global model $w_{t\backslash norm}$. We hypothesize that a proximal term (Equation 3) inspired by FedProx will help the local models in FedBN to progress towards the global optimum. As Figure 1b shows, the added proximal term prevents $w_{t,k\backslash norm}$ from deviating too much from $w_{t\backslash norm}$.



(a) FedBN (b) FedPxN

Figure 1: L_2 distance between each client's model parameters and the global parameters, excluding the normalization layers, when using (a) FedBN and (b) FedPxN. Both show results of the mortality prediction task, in which all models were equipped with LN, and trained using the five clients with the largest samples.

Based on this observation, we propose FedPxN, an algorithm encouraging the normalization layers to adapt to each client's unique feature distribution and the other layers to follow the global optimum in order to improve the overall performance. In FedPxN, local models are updated by the local objective with the proximal term of the other layers of the global model during local training and then aggregated without the normalization layers. The detailed algorithm is presented in the Appendix Algorithm 2.

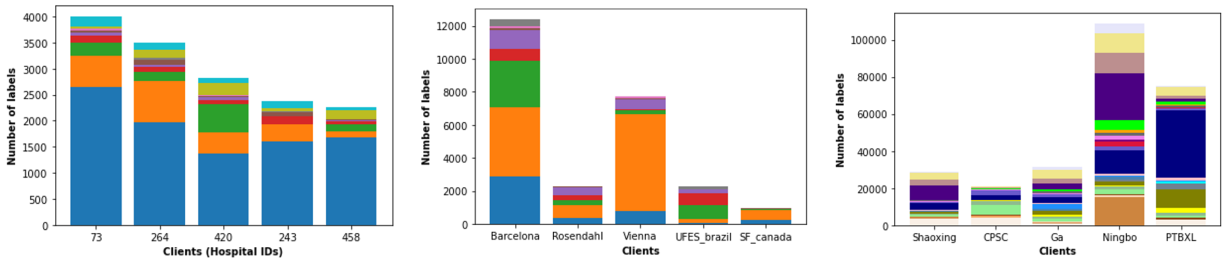
4. Experimental Setup

In this section, we introduce the experimental settings and different medical datasets used to evaluate all FL frameworks for the medical domain in terms of their practical utility. First of all, we compare our approach with recent well-known FL methods for solving the non-i.i.d problem including FedAvg (McMahan et al., 2017), FedProx (Li et al., 2020), FedBN (Li et al., 2021), FedOpt (FedAdam, FedAdagrad, FedYoGi) (Reddi et al., 2020), and FedDyn (Acar et al., 2021). More information about FL methods is described in Appendix A.

For all our experiments, we assumed the full participation of clients in all communication rounds by LAN connection. For all FL methods in each of the tasks in our study, we used the same deep learning model, data size, input size, and GPU in order to conduct a fair comparison. The CPU spec for our experimental setting is AMD EPYC 7502 32-Core Processor(2.5GHz). We used NVIDIA GeForce RTX 3090 with a RAM size of 24G and the corresponding CUDA version was 11.4. Further, we fixed the total number of training epochs for each local client in all FL methods, where this number is defined as the number of local epochs times the number of communication rounds. If not specified, our default setting for local epochs is a single epoch. We also conducted the experiments by setting various communication rounds and local epochs. More details are described in Appendix B. Also, Experiment results about various combinations of communication rounds and local epochs are discussed in Section 5.2.

4.1. Electronic Health Records

We first evaluated FL methods on the eICU dataset (Pollard et al., 2018) by taking advantage of an available benchmark dataset (McDermott et al., 2021), which consists of intensive care unit (ICU) records of patients aged 15 years or over. The benchmark dataset contains 71477 ICU stays across 59 hospitals, where ICU stays range between 540 and 4008 for each hospital. To conduct experiments in the FL



(*a*) eICU Acuity (*b*) skin cancer datasets (*c*) ECG datasets

Figure 2: Label distributions of eICU, skin cancer, and ECG datasets. Colors indicate the labels for each client. The label distribution shown for eICU is for the final acuity prediction task for the five clients with the largest datasets.

setting, we used the 5, 10, 20, and 30 hospitals with the most ICU stays from the dataset and performed the six clinical prediction tasks; two mortality prediction tasks (mort_24h, mort_48h), two discharge perdiction tasks (disch_24h, disch_48h), length of stay prediction (LOS), and final acuity prediction (Acuity). Mort_24h, mort_48h, and LOS are binary classification tasks. Disch_24h, disch_48h, and Acuity are 10-way classification tasks. More details about the clinical tasks are in Appendix B.

Among all tasks that have heterogeneous label distribution, as an example, we show the label distribution of the Final acuity prediction task in Figure 2a. While the most dominant label is shared by all clients, the label distributions and data volumes vary across the clients. We used a Transformer classifier (Vaswani et al., 2017) with LN or GN in all tasks. Full details are given in Appendix B. We utilized the AUROC and AUPRC scores to evaluate the performance of the trained model.

4.2. Skin cancer image dataset

We also evaluated all FL methods on skin cancer datasets. To be more specific, we constructed five clients which are from different sources (Barcelona, Vienna, Rosendahl, UFES_brazil, SF_canada). The details of data construction are illustrated in Appendix B. We observe that the dominant label is almost the same for all clients and all clients have an imbalanced dataset. Further, we observe that the second dominant label is more different for each client, and the total number of samples significantly varies from one client to another. The details of the label distribution and size of the data for each client are shown in Figure 2b.

Then, we formulated an 8-way classification task that uses an image as input and predicts the type of skin cancer. Also, we used an EfficientNet-B0 (Tan and Le, 2019) with BN or GN. To evaluate the performance of the model, we use two metrics, namely AUROC and AUPRC. More details are also listed in Appendix B.

4.3. Electrocardiogram (ECG) dataset

We also evaluated the FL methods on ECG signals. We used the PhysioNet 2021 (Reyna et al., 2021) dataset and split it into five clients based on the different hospitals (Shaoxing, CPSC, Ga, Ningbo, PTBXL). There are 26 labels related to cardiac diseases. In Figure 2c, the label distribution of different clients for these data reveals that the total number of samples varies substantially from one client to another, and the data are imbalanced. Moreover, the dominant label in the data from each client varies.

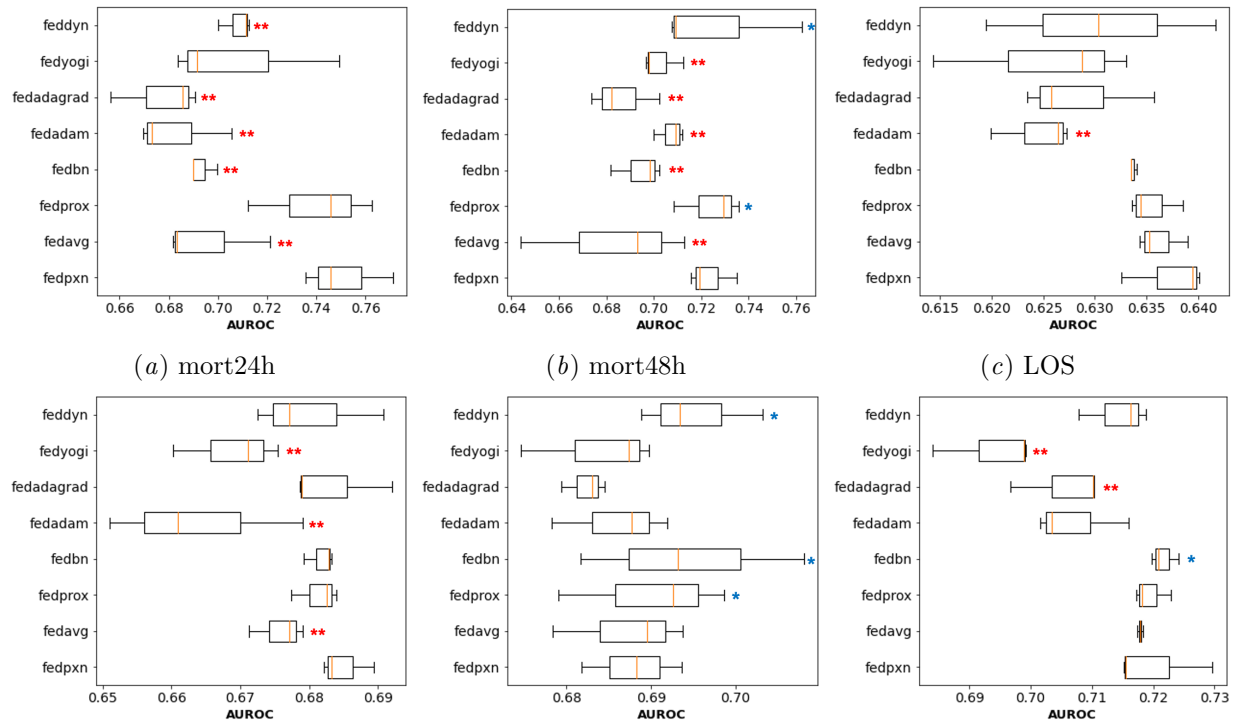
Our objective is to solve a multi-label prediction task to diagnose 26 types of cardiac diseases in which a 12-lead ECG sample is given as input. We trained a ResNet-NC-SE (Kang and Wen, 2022) with BN or GN because it showed the best performance on 12-lead ECG readings in PhysioNet 2021 to the best of our knowledge (Hao and Jingsu, 2022; WEN and KANG, 2022). Following Oh et al. (2022), we evaluated the performance of our approach by measuring the CinC score, the official evaluation metric used by the Physionet 2021 challenge that takes into account domain knowledge on cardiovascular diseases. For more details, please refer to Appendix B.

4.4. Statistical analysis

We conducted extensive experiments to provide comprehensive benchmarks. For example, for eICU dataset, there are 6 tasks, 8 algorithms, and 2 nor-

Table 1: AUROC results for the eICU dataset using the data of the five largest clients. For each FL method, bold indicates the better normalization technique (LN or GN). We indicate the highest average AUROC results for all six tasks in blue.

		FedAvg	FedProx	FedBN	FedAdam	FedAdagrad	FedYoGi	FedDyn	FedPxN
mort_24h	LN	67.85±2.52	73.29±2.13	**69.31±0.56**	**68.27±1.98**	**67.76±1.85**	60.75±3.78	**70.80±0.68**	74.05±1.61
	GN	**69.54±2.23**	**74.02±2.57**	69.05±1.64	67.60±2.39	67.21±0.77	**70.82±3.57**	69.79±2.09	**75.07±1.83**
mort_48h	LN	**68.33±3.55**	72.37±1.42	68.38±0.81	69.61±1.31	**68.62±1.47**	68.69±0.56	67.99±1.23	72.22±1.87
	GN	67.81±0.58	**72.46±1.42**	**69.42±1.08**	**70.71±0.64**	68.32±0.51	**70.23±0.86**	**72.64±3.11**	**72.33±1.02**
LOS	LN	63.23±0.31	62.97±0.34	63.04±0.13	**62.46±0.40**	62.61±0.57	62.31±0.71	61.97±0.53	**63.73±0.42**
	GN	**63.61±0.24**	**63.55±0.27**	**63.36±0.03**	62.36±0.85	**62.83±0.65**	**62.54±0.98**	**63.05±1.11**	63.68±0.18
disch_24h	LN	**67.58±0.40**	67.95±0.56	**68.18±0.22**	**66.37±1.42**	66.50±0.32	66.26±1.02	66.99±0.87	67.98±0.57
	GN	67.38±0.61	**68.13±0.34**	67.63±0.56	66.34±1.39	**68.31±0.76**	**66.89±0.78**	**68.01±0.95**	**68.49±0.39**
disch_48h	LN	**68.72±0.80**	68.49±0.97	69.37±0.58	**68.59±0.70**	68.20±0.59	**68.39±0.81**	**69.51±0.73**	68.51±0.22
	GN	67.90±1.19	**69.01±1.00**	**69.43±1.32**	68.23±0.55	**68.23±0.27**	68.09±0.54	69.31±1.52	**68.79±0.59**
Acuity	LN	71.60±0.20	71.56±0.09	71.99±0.34	**70.70±0.78**	**70.58±0.79**	**69.40±0.87**	70.74±0.48	71.40±0.35
	GN	**71.79±0.04**	**71.94±0.30**	**72.16±0.23**	69.24±0.69	68.97±0.37	69.15±0.44	**71.43±0.58**	**72.01±0.83**
Average	LN	67.88	69.44	68.38	**67.67**	**67.38**	65.97	68	69.65
	GN	**68.01**	**69.85**	**68.51**	67.42	67.31	**67.96**	**69.04**	**70.06**



(a) mort24h (b) mort48h (c) LOS

(d) disch24h (e) disch48h (f) Acuity

Figure 3: Results of the statistical test with eICU datsets in the 5 clients setting. Two red astericks(**) show that FedPxN wins significantly(i.e. p-value is less than 0.05). One blue asterick(*) shows that FedPxN loses with average score, but insignificant(i.e. p-value is over 0.05). `FedPxN` never loses significantly.

malization techniques for each method. Also, we trained each model in the setting of 5, 10, 20, and 30 clients. For each experiment, each client should train the model respectively and the model should be aggregated. We had to go through this communication round 100, 20, or 10 times depending on the number of local training epochs. The same thing happened for the skin cancer dataset and ECG dataset. Due to this extensive process, we took 3 random seeds for each experiment. Each experiment is indepen-

Table 2: AUROC and AUPRC results for the skin cancer dataset. Bold indicates the best normalization strategy for each method. We indicate the best performance in blue.

		AUROC						AUPRC					
		Barcelona	Rosendahl	Vienna	UFES_brazil	SF_canada	Avg	Barcelona	Rosendahl	Vienna	UFES_brazil	SF_canada	Avg
FedAvg	BN	95.19±0.75	88.44±0.50	96.84±0.36	76.40±1.79	84.20±1.00	**88.19±0.03**	53.40±5.33	36.25±2.07	56.51±5.15	25.81±0.95	22.93±2.51	**39.12±1.24**
	GN	90.78±0.51	84.50±1.37	94.63±0.74	72.72±1.33	76.24±1.46	83.78±0.29	35.81±3.92	29.91±1.60	39.59±2.50	23.36±0.97	20.06±1.07	29.75±1.66
FedProx	BN	95.82±0.43	87.56±1.61	96.40±0.93	76.79±0.06	85.26±3.19	**88.37±0.68**	58.63±2.64	36.47±3.38	61.80±4.76	25.92±0.55	23.36±1.28	**41.24±1.33**
	GN	90.28±0.24	82.70±1.26	94.51±0.45	71.88±1.75	73.58±2.71	82.59±0.25	32.54±2.56	30.01±2.07	40.11±1.52	23.86±0.44	19.79±1.09	29.26±1.19
FedBN	BN	95.74±0.40	86.92±3.48	97.99±0.57	85.58±1.65	79.19±2.80	**89.08±1.10**	59.39±0.47	40.73±4.78	64.86±2.52	35.82±1.69	17.94±1.09	**43.75±1.08**
	GN	91.59±0.19	87.02±0.26	95.33±0.36	85.14±0.76	80.25±2.79	87.86±0.54	39.35±1.22	36.64±2.07	41.06±3.96	31.99±1.30	22.80±1.15	34.37±1.49
FedAdam	BN	64.50±3.55	65.53±2.89	68.50±2.40	64.64±2.24	71.46±9.06	66.93±2.80	13.70±0.39	17.88±1.26	14.74±0.16	18.17±0.84	17.55±1.03	16.41±0.66
	GN	91.97±0.26	80.63±1.83	95.22±0.55	70.65±2.14	75.77±1.10	**82.85±0.32**	40.34±6.00	29.47±1.30	42.68±2.69	24.39±1.13	20.32±1.16	**31.44±2.29**
FedAdagrad	BN	62.42±3.43	62.89±3.33	66.54±1.42	63.94±2.29	70.84±8.58	65.33±1.80	13.78±0.61	18.34±2.35	14.54±0.14	18.45±1.19	17.75±0.48	16.57±0.86
	GN	92.26±0.89	84.28±0.52	95.77±0.13	70.99±0.69	76.22±1.20	**83.90±0.11**	42.38±6.66	32.73±6.66	45.59±3.03	23.02±0.69	20.84±2.24	**32.91±1.72**
FedYoGi	BN	62.29±5.18	65.61±2.92	65.83±3.78	64.26±2.16	72.32±7.92	66.06±2.47	13.87±0.81	19.07±0.95	14.89±0.53	18.63±0.64	18.22±0.26	16.94±0.45
	GN	91.10±1.49	79.96±0.95	95.19±0.69	68.20±1.90	75.17±1.42	**81.92±0.76**	38.65±4.30	29.51±1.99	40.54±8.61	22.56±0.71	21.14±2.13	**30.48±2.99**
FedDyn	BN	80.75±0.85	83.45±0.62	91.98±0.42	72.62±1.20	85.61±0.92	82.88±0.06	20.73±0.53	24.14±0.27	21.20±0.29	23.55±0.46	21.27±0.22	22.18±0.09
	GN	88.42±0.14	86.35±1.12	95.47±0.18	82.73±0.98	82.19±1.07	**87.03±0.21**	28.61±1.11	31.05±1.18	41.38±3.58	31.93±0.71	20.60±1.06	**30.71±1.04**
`FedPxN`	BN	95.93±0.30	88.81±1.86	97.64±0.92	84.36±1.70	84.51±3.62	**90.25±1.27**	59.42±0.53	38.77±2.75	69.17±4.15	34.81±1.42	17.85±1.11	**44.00±1.30**
	GN	91.60±0.15	87.61±0.87	94.49±0.93	84.39±0.30	78.35±3.03	87.29±0.88	36.71±0.78	35.39±3.36	41.17±0.03	31.82±1.18	19.80±0.61	32.98±1.03

dent, but as there are three samples for each, we used the Wilcoxon rank-sum test (i.e. Mann–Whitney U test)(Mann and Whitney, 1947) to evaluate the statistical significance of the difference between each algorithm. The Wilcoxon rank-sum test is the non-parametric version of the two-sample t-test. We considered the result is statistically significant if a p-value is less than 0.05.

5. Results

5.1. Experimental results for each dataset

5.1.1. eICU dataset

We present the results of the FL methods for six clinical prediction tasks on the eICU dataset described in Section 4.1. We present the experiment results using the data from the five largest clients with hospital IDs 73, 264, 420, 243, and 458 in Table 1.

As shown in Table 1 and Figure 3, there is no clear winner in the eICU dataset that outperforms all other algorithms across all tasks. The tendency varies even within the same discharge prediction task according to the time window (disch-24, disch-48). Therefore, we averaged AUROC of all six tasks to give better suggestions to the community.

We observe that most FL methods perform better when trained with GN instead of LN. We also observe that the average performance of our approach `FedPxN` trained using GN across all tasks is better than that of existing FL methods and our approach trained using LN. We report the result of the Wilcoxon rank-sum test that shows the hybrid method never loses significantly on Figure 3. `FedPxN` also outperforms all FL methods in terms of AUPRC, which can be viewed on Appendix Table 6.

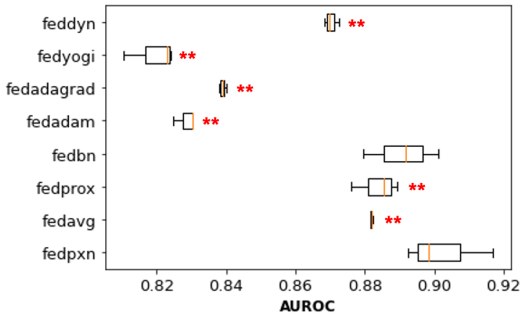
5.1.2. Skin cancer image dataset

We present the results of FL methods on the skin cancer image dataset for the 8-way classification task with five clients for both BN and GN. Table 2 reveals that training with BN is better than GN in the FedAvg, FedProx, FedBN, and `FedPxN` when comparing clients' average AUROC and AUPRC results. In contrast, GN is better than BN in FedAdam, FedAdagrad, FedYoGi and FedDyn. `FedPxN` yields the overall best performance when trained with BN and GN. When using GN, FedBN also shows good performance when compared to other methods. According to Figure 4a, FedPxN never loses compared to each method with its better normalization technique and always wins significantly except FedBN.

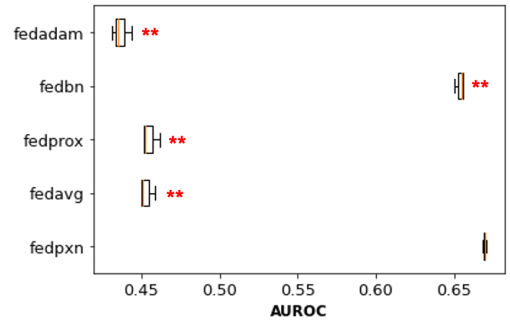
Surprisingly, training was extremely unstable when FedDyn was trained using BN. Specifically, a NaN value was found in the BN parameter during training. To address the problem, we tried 1) extensive hyper-parameter tuning, 2) allowing BN's rescaling parameters to be aggregated (Andreux et al., 2020) 3) allowing none of BN parameters to be aggregated (Li et al., 2021), but none of the approaches helped improve the performance of FedDyn.

5.1.3. ECG dataset

We present the results of the FL methods for the 26 multi-label prediction task on ECG signals in Table 3. All the results indicate that training with BN is more effective in FedAvg, FedProx, FedBN and `FedPxN` whereas training with GN is better for FedAdam. Our proposed approach, `FedPxN`, shows overall the best performance based on the average CinC score when trained with BN or GN. `FedPxN` always wins significantly as shown in Figure 4b.



(a) skin cancer datasets



(b) ECG datasets

Figure 4: Results of the statistical test with the skin cancer and ECG datasets. Two red astericks(**) show that FedPxn wins significantly(i.e. p-value is less than 0.05). For the skin cancer datasets, FedPxN always wins significantly except one case. For the ECG datasets, FedPxN always wins significantly.

FedDyn failed to train with either BN or GN (similar to Section 5.1.2). To address this failure in training, we tried various modifications similar to those stated in Section 5.1.2. However, we still encountered failure in training. Therefore, we do not report the results of FedDyn.

Table 3: CinC scores for the ECG dataset. We indicate the best normalization strategy for each method in bold. The highest average CinC is indicated in blue.

		Shaoxing	CPSC	Ga	Ningbo	PTBXL	**Average**
FedAvg	BN	65.8±1.3	45.1±1.0	39.0±0.3	63.5±1.0	13.1±1.4	**45.3±0.4**
	GN	65.4±1.0	46.1±0.4	37.6±0.8	60.7±0.2	11.1±1.1	44.17±0.6
FedProx	BN	66.9±1.4	47.3±0.8	40.0±0.1	58.7±0.9	14.7±0.4	**45.53±0.5**
	GN	65.3±0.6	46.2±0.1	37.1±0.1	59.4±1.0	10.4±0.8	43.68±0.1
FedBN	BN	78.9±0.7	72.5±0.8	46.5±0.5	76.7±0.3	52.2±1.9	**65.37±0.3**
	GN	77.6±0.3	68.5±1.5	49.2±0.4	75.9±0.5	49.7±0.3	64.19±0.3
FedAdam	BN	61.6±1.3	43.9±3.0	33.1±1.6	56.9±5.1	4.3±1.3	39.95±1.2
	GN	64.3±1.1	44.7±0.5	37.4±1.0	60.5±1.2	11.5±0.9	**43.67±0.6**
FedPxN	BN	78.0±1.0	72.7±0.3	51.7±1.5	76.4±0.5	55.8±1.1	**66.93±0.1**
	GN	78.4±0.6	70.0±1.1	48.7±0.7	75.9±0.5	51.8±0.8	64.95±0.4

5.2. The number of local training epochs

We investigated the effect of the number of local epochs on the performance of the FL algorithms. We conducted experiments on the three datasets (eICU, skin cancer, and ECG) using 1, 5, and 10 local epochs, while fixing the number of total training epochs. The performance was averaged across three seeds. Figure 5 presents the results on the eICU dataset, where the images in the top and bottom rows show the plots of models trained with GN and LN, respectively. Each row contains four plots corresponding to models trained and tested on $5, 10, 20$, and 30 clients by AUROC score averaged across all six tasks. Figure Figure 6 presents the results of models trained with BN or GN and varying local epochs on the skin cancer and ECG dataset.

Figures 5 and 6 reveal that the performance of most FL methods including `FedPxN` consistently decreases as the number of local epochs increases. This demonstrates that FL methods perform well with fewer local epochs and more global updates. Because the number of total training epochs is fixed, fewer local epochs indicate more communication rounds, which would require good internet connectivity and might result in increased bandwidth costs. However, in medicine, a model's performance is the utmost priority, as it has a high impact on a patient's life and a wrong prediction could lead to fatal errors. Therefore, although more communication rounds increase communication costs, a better performing FL model is preferable in medicine.

5.3. Overall guidance from the results

The first step to apply FL is to decide the FL algorithm. `FedPxN` shows favorable performances for every dataset. For the structured data, there was no clear winner, but `FedPxN` is always included in the top 3 powerful methods, often with FedProx and FedDyn. For the image or signal data, `FedPxN` is a clear winner overall and FedBN follows. In addition, `FedPxN` has relatively less fluctuation with its performance, so it can be the first option to consider. FedAvg is the most basic and simple algorithm but never wins in our experiments. Unfortunately, FedOpt variants were poor with the image or signal dataset.

The next step is to design a communication process. According to our results above, it is better to train local clients' models with a single local epoch and increase the number of communication rounds

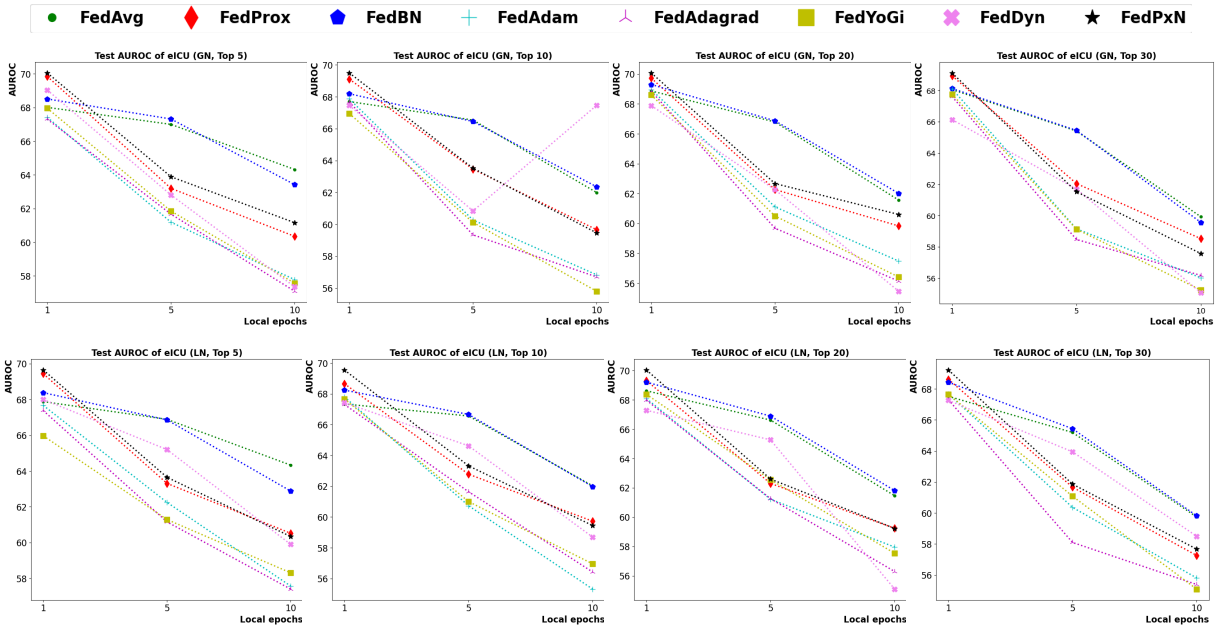


Figure 5: Test AUROC results for the eICU dataset as the number of local training epochs increases.

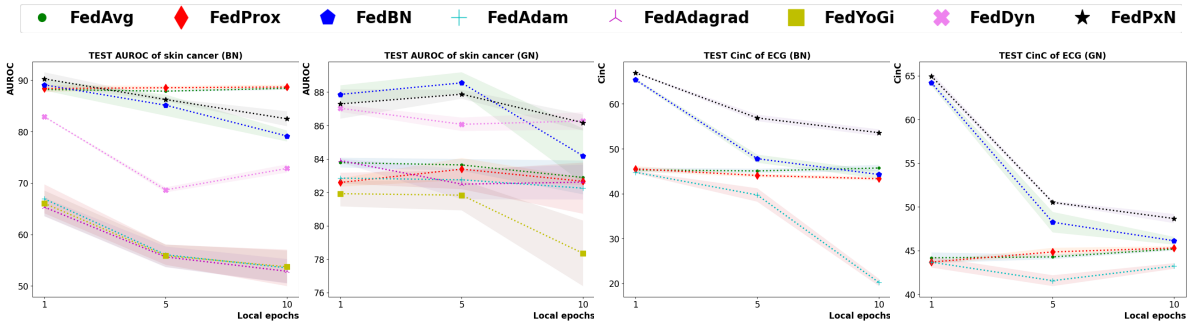


Figure 6: Test AUROC and CinC results for the skin cancer image and ECG datasets measured as the number of local training epochs increases.

instead. This guidance will be a good starting point for the hospitals or the IT companies to adopt FL.

6. Discussion

We observe which method is cost-effective for optimal performance by measuring power consumption each FL method requires to achieve the best performance. As the number of clients increases in FL, more IT administrators and GPUs will be needed, and the related monetary costs will increase accordingly. So we note which methods are effective as the number of clients increases using the eICU database. Furthermore, we explain our limitations in this section.

6.1. Power consumption

We measured the power consumed while conducting the experiments using a single local epoch. We report the results for one seed value only for this experiment in Table 4. To understand the monetary costs of each method to achieve optimal performance, we conducted experiments using the normalization technique that showed better performance in Section 5.1. During training, we queried the NVIDIA System Management Interface [2] at regular intervals to measure the power consumption of GPU and averaged the measured values (Strubell et al., 2019).

2. https://developer.nvidia.com/nvidia-system-management-interface

Then, we multiplied the obtained average values by elapsed time. The power consumption of all FL methods is comparable. The difference between max and min power consumption among all methods in each dataset is between 0.103-0.596 kWh. The number of hyperparameters varies depending on the FL method (see Appendix C). So, the hyperparameter-tuning cost of FedAvg and FedBN is the cheapest considering the number of hyperparameters. Practically, FedBN is the cheapest technique for all tasks on the three datasets. The power consumption of `FedPxN` is 1.03-1.17 times more than that of FedBN in Table 4.

Table 4: Power consumption of each method for the tasks in the eICU, skin cancer images, and ECG datasets. In eICU, we measured total power consumption of six tasks with 30 clients.

	FedAvg	FedProx	FedBN	FedPxN	FedAdam	FedAdagrad	FedYoGi	FedDyn
eICU								
	GN	GN	LN	LN	GN	GN	GN	LN
elapsed time (h)	5.66	5.82	4.51	4.51	4.81	4.84	5.14	5.13
Power consumption (kWh)	0.542	0.560	0.510	0.526	**0.479**	0.484	0.518	0.582
Skin cancer images								
	BN	BN	BN	BN	GN	GN	GN	GN
elapsed time (h)	10.53	10.62	10.29	10.27	10.56	10.67	10.61	9.49
Power consumption (kWh)	3.034	3.280	**2.989**	3.266	3.046	3.097	3.100	3.072
ECG								
	BN	BN	BN	BN	GN			
elapsed time (h)	9.14	11.49	9.15	10.71	9.58			
Power consumption (kWh)	2.955	3.484	**2.888**	3.401	3.016			

6.2. Administrators and GPUs

6.2.1. Training with more clients

We performed experiments in which we trained the FL methods with 10, 20, 30 largest clients and reported the average AUROC of all six tasks using the eICU in Appendix Table 7. GN outperforms LN in most FL methods trained with $5 - 30$ clients. In addition, we observe that `FedPxN` yields the best AUROC in each setting (5, 10, 20, 30 clients) despite the similar monetary costs requirement of different FL methods.

6.2.2. Testing on the top-30 client dataset with varying training clients

We evaluated all FL methods on the test set from the top 30 clients, while varying the number of clients the models were trained on. The performance is measured by the average AUROC of all six tasks. As depicted in Figure 7, the AUROC results of all methods except FedDyn consistently increase when both LN and GN are used.

This demonstrates that training the model on more clients generalizes the model due to the availability of heterogeneous data and improves model performance. FedProx obtains the best performance with both GN and LN when the FL methods are trained on 20 or fewer clients. This implies that training with FedProx using fewer hospitals can provide better performance for hospitals not involved in training. However, there is only a marginal difference between the AUROC of FedProx and `FedPxN`. For example, the AUROC of FedProx and `FedPxN` are 67.15 and 66.79 when training with 20 clients' data using GN. Also, `FedPxN` yields the optimal performance when trained on the data of all 30 clients.

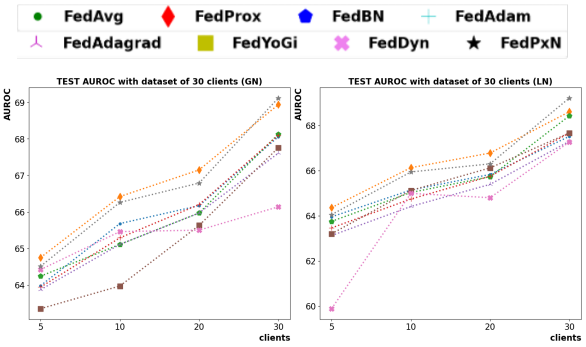


Figure 7: Test AUROC results for the five largest clients and all 30 clients measured using the models trained on varying numbers of clients.

6.3. Transmitting cost

For a fair comparison, we fixed the number of total communication rounds for each task, and therefore the transmitting fee was not within our scope. Yet, we assume that the transmitting fee is insignificant. Institutions such as hospitals usually pay the fee for the internet connection on a monthly or yearly basis. In our experimental settings, the model parameters were approximately 4.4MB, 16MB, and 32MB and the cost of exchanging model parameters 100-300 times is sufficiently covered by the amount paid. Therefore, there is no practical need to calculate the transmitting fee separately.
However, if the hospital uses the cloud service, it would be better to consider it carefully because the transmission cost may vary depending on the bandwidth, and increase proportionally with the model size. In that case, it would be an interesting future work to analyze the FL framework that guarantees generally good performance while minimizing the communication rounds by fast convergence.

6.4. Limitations

Although we provide comprehensive benchmarks on three medical modalities for the first time, there are some limitations. We conducted experiments across multiple GPU's but on the same machine for a fair experimental setting. It would have been even more realistic if the experiments were conducted across independent test sites. Plus, eICU database is across independent hospitals, but they are still from the United States. Therefore, it does not fully reflect real-world multi-national heterogeneity. For medical images in our experimental setting, we tested FL methods for a specific cancer and task. The next step is to validate FL methods across different clinical tasks or cancer types (e.g., medical image segmentation, breast cancer prediction).

7. Conclusion

In this work, we provide practical benchmarks of various FL algorithms on the real-world heterogeneous datasets in the clinical domain with three modalities: structured, image, and signal data. We find out that a hybrid algorithm, `FedPxN`, that introduces the regularization term from FedProx to the FedBN framework is simple but effective. It has a slightly higher power consumption than the most economical method (FedBN), but mostly outperforms the other methods and never loses significantly. We expect users to refer to these experimental results to design their FL settings and to select a suitable FL algorithm considering their clinical tasks and obtain stronger performance with lower costs.

Institutional Review Board (IRB) This research does not require IRB approval.

Acknowledgments

This work was supported by the KAIST Key Research Institute (Interdisciplinary Research Group) Project, Institute of Information & Communications Technology Planning & Evaluation (IITP) grant (No.2019-0-00075), and the Korea Health Industry Development Institute (KHIDI) grant (No.HR21C0198, No.HI22C1518), funded by the Korea government (MSIT, MOHW).

References

Durmus Alp Emre Acar, Yue Zhao, Ramon Matas, Matthew Mattina, Paul Whatmough, and Venkatesh Saligrama. Federated learning based on dynamic regularization. In *International Conference on Learning Representations*, 2021.

Mathieu Andreux, Jean Ogier du Terrail, Constance Beguier, and Eric W Tramel. Siloed federated learning for multi-centric histopathology datasets. In *Domain Adaptation and Representation Transfer, and Distributed and Collaborative Learning*, pages 129–139. Springer, 2020.

Jimmy Lei Ba, Jamie Ryan Kiros, and Geoffrey E Hinton. Layer normalization. *arXiv preprint arXiv:1607.06450*, 2016.

Tariq Bdair, Nassir Navab, Shadi Albarqouni, et al. Semi-supervised federated peer learning for skin lesion classification. *Machine Learning for Biomedical Imaging*, 1(April 2022 issue):1–10, 2022.

Sabri Boughorbel, Fethi Jarray, Neethu Venugopal, Shabir Moosa, Haithum Elhadi, and Michel Makhlouf. Federated uncertainty-aware learning for distributed hospital ehr data. *arXiv preprint arXiv:1910.12191*, 2019.

Theodora S Brisimi, Ruidi Chen, Theofanie Mela, Alex Olshevsky, Ioannis Ch Paschalidis, and Wei Shi. Federated learning of predictive models from federated electronic health records. *International journal of medical informatics*, 112:59–67, 2018.

Bill Cassidy, Connah Kendrick, Andrzej Brodzicki, Joanna Jaworek-Korjakowska, and Moi Hoon Yap. Analysis of the isic image datasets: usage, benchmarks and recommendations. *Medical image analysis*, 75:102305, 2022.

Noel Codella, Veronica Rotemberg, Philipp Tschandl, M Emre Celebi, Stephen Dusza, David Gutman, Brian Helba, Aadi Kalloo, Konstantinos Liopyris, Michael Marchetti, et al. Skin lesion analysis toward melanoma detection 2018: A challenge hosted by the international skin imaging collaboration (isic). *arXiv preprint arXiv:1902.03368*, 2019.

Enmao Diao, Jie Ding, and Vahid Tarokh. Heterofl: Computation and communication efficient federated learning for heterogeneous clients. In *International Conference on Learning Representations*, 2020.

Qi Dou, Tiffany Y So, Meirui Jiang, Quande Liu, Varut Vardhanabhuti, Georgios Kaissis, Zeju Li, Weixin Si, Heather HC Lee, Kevin Yu, et al. Federated deep learning for detecting covid-19 lung abnormalities in ct: a privacy-preserving multinational validation study. *NPJ digital medicine*, 4(1): 1–11, 2021.

Wen Hao and Kang Jingsu. Investigating deep learning benchmarks for electrocardiography signal processing. *arXiv preprint arXiv:2204.04420*, 2022. doi: 10.48550/ARXIV.2204.04420.

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

Tzu-Ming Harry Hsu, Hang Qi, and Matthew Brown. Measuring the effects of non-identical data distribution for federated visual classification. *arXiv preprint arXiv:1909.06335*, 2019.

Li Huang, Andrew L Shea, Huining Qian, Aditya Masurkar, Hao Deng, and Dianbo Liu. Patient clustering improves efficiency of federated machine learning to predict mortality and hospital stay time using distributed electronic medical records. *Journal of biomedical informatics*, 99:103291, 2019.

Sergey Ioffe and Christian Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. In *International conference on machine learning*, pages 448–456. PMLR, 2015.

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

Jingsu Kang and Hao Wen. A study on several critical problems on arrhythmia detection using varying-dimensional electrocardiography. *Physiological Measurement*, 2022.

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

Jeremy Kawahara, Sara Daneshvar, Giuseppe Argenziano, and Ghassan Hamarneh. Seven-point checklist and skin lesion classification using multitask multimodal neural nets. *IEEE journal of biomedical and health informatics*, 23(2):538–546, 2018.

Yejin Kim, Jimeng Sun, Hwanjo Yu, and Xiaoqian Jiang. Federated tensor factorization for computational phenotyping. In *Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 887–895, 2017.

Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

Geun Hyeong Lee and Soo-Yong Shin. Federated learning on clinical benchmark data: performance assessment. *Journal of medical Internet research*, 22(10):e20891, 2020.

Qinbin Li, Yiqun Diao, Quan Chen, and Bingsheng He. Federated learning on non-iid data silos: An experimental study. In *IEEE International Conference on Data Engineering*, 2022.

Tian Li, Anit Kumar Sahu, Manzil Zaheer, Maziar Sanjabi, Ameet Talwalkar, and Virginia Smith. Federated optimization in heterogeneous networks. *Proceedings of Machine Learning and Systems*, 2: 429–450, 2020.

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

Xiaoxiao Li, Meirui Jiang, Xiaofei Zhang, Michael Kamp, and Qi Dou. Fed{bn}: Federated learning on non-{iid} features via local batch normalization. In *International Conference on Learning Representations*, 2021.

Dianbo Liu, Timothy Miller, Raheel Sayeed, and Kenneth D Mandl. Fadl: Federated-autonomous deep learning for distributed electronic health record. *arXiv preprint arXiv:1811.11400*, 2018.

Dianbo Liu, Dmitriy Dligach, and Timothy Miller. Two-stage federated phenotyping and patient representation learning. In *Proceedings of the conference. Association for Computational Linguistics. Meeting*, volume 2019, page 283. NIH Public Access, 2019.

Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. In *International Conference on Learning Representations*, 2018.

Henry B. Mann and Douglas R. Whitney. On a test of whether one of two random variables is stochastically larger than the other. *Annals of Mathematical Statistics*, 18:50–60, 1947.

Matthew McDermott, Bret Nestor, Evan Kim, Wancong Zhang, Anna Goldenberg, Peter Szolovits, and Marzyeh Ghassemi. A comprehensive ehr timeseries pre-training benchmark. In *Proceedings of the Conference on Health, Inference, and Learning*, pages 257–278, 2021.

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

Jungwoo Oh, Hyunseung Chung, Joon-myoung Kwon, Dong-gyun Hong, and Edward Choi. Lead-agnostic self-supervised learning for local and global representations of electrocardiogram. In *Conference on Health, Inference, and Learning*, pages 338–353. PMLR, 2022.

Andre GC Pacheco, Gustavo R Lima, Amanda S Salomão, Breno Krohling, Igor P Biral, Gabriel G de Angelo, Fábio CR Alves Jr, José GM Esgario, Alana C Simora, Pedro BC Castro, et al. Pad-ufes-20: a skin lesion dataset composed of patient data and clinical images collected from smartphones. *Data in brief*, 32:106221, 2020.

Sangjoon Park, Gwanghyun Kim, Jeongsol Kim, Boah Kim, and Jong Chul Ye. Federated split task-agnostic vision transformer for COVID-19 CXR diagnosis. In A. Beygelzimer, Y. Dauphin, P. Liang, and J. Wortman Vaughan, editors, *Advances in Neural Information Processing Systems*, 2021.

Le Peng, Gaoxiang Luo, Andrew Walker, Zachary Zaiman, Emma K Jones, Hemant Gupta, Kristopher Kersten, John L Burns, Christopher A

Harle, Tanja Magoc, Benjamin Shickel, Scott D Steenburg, Tyler Loftus, Genevieve B Melton, Judy Wawira Gichoya, Ju Sun, and Christopher J Tignanelli. Evaluation of federated learning variations for COVID-19 diagnosis using chest radiographs from 42 US and European hospitals. *Journal of the American Medical Informatics Association*, 30(1):54–63, 10 2022. ISSN 1527-974X. doi: 10.1093/jamia/ocac188. URL https://doi.org/10.1093/jamia/ocac188.

Stephen R Pfohl, Andrew M Dai, and Katherine Heller. Federated and differentially private learning for electronic health records. *arXiv preprint arXiv:1911.05861*, 2019.

Tom J Pollard, Alistair EW Johnson, Jesse D Raffa, Leo A Celi, Roger G Mark, and Omar Badawi. The eicu collaborative research database, a freely available multi-center database for critical care research. *Scientific data*, 5(1):1–13, 2018.

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

Matthew A Reyna, Nadi Sadr, Erick A Perez Alday, Annie Gu, Amit J Shah, Chad Robichaux, Ali Bahrami Rad, Andoni Elola, Salman Seyedi, Sardar Ansari, et al. Will two do? varying dimensions in electrocardiography: the physionet/computing in cardiology challenge 2021. In *2021 Computing in Cardiology (CinC)*, volume 48, pages 1–4. IEEE, 2021.

Tal Ridnik, Emanuel Ben-Baruch, Nadav Zamir, Asaf Noy, Itamar Friedman, Matan Protter, and Lihi Zelnik-Manor. Asymmetric loss for multi-label classification. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 82–91, October 2021.

Nicola Rieke, Jonny Hancox, Wenqi Li, Fausto Milletari, Holger R Roth, Shadi Albarqouni, Spyridon Bakas, Mathieu N Galtier, Bennett A Landman, Klaus Maier-Hein, et al. The future of digital health with federated learning. *NPJ digital medicine*, 3(1):1–7, 2020.

Micah J Sheller, Brandon Edwards, G Anthony Reina, Jason Martin, Sarthak Pati, Aikaterini Kotrotsou, Mikhail Milchenko, Weilin Xu, Daniel Marcus, Rivka R Colen, et al. Federated learning in medicine: facilitating multi-institutional collaborations without sharing patient data. *Scientific reports*, 10(1):1–12, 2020.

Huan Song, Deepta Rajan, Jayaraman J Thiagarajan, and Andreas Spanias. Attend and diagnose: Clinical time series analysis using attention models. In *Thirty-second AAAI conference on artificial intelligence*, 2018.

Emma Strubell, Ananya Ganesh, and Andrew McCallum. Energy and policy considerations for deep learning in nlp. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 3645–3650, 2019.

Mingxing Tan and Quoc Le. Efficientnet: Rethinking model scaling for convolutional neural networks. In *International conference on machine learning*, pages 6105–6114. PMLR, 2019.

Philipp Tschandl, Cliff Rosendahl, and Harald Kittler. The ham10000 dataset, a large collection of multi-source dermatoscopic images of common pigmented skin lesions. *Scientific data*, 5(1):1–9, 2018.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017.

Jianyu Wang, Zachary Charles, Zheng Xu, Gauri Joshi, H Brendan McMahan, Maruan Al-Shedivat, Galen Andrew, Salman Avestimehr, Katharine Daly, Deepesh Data, et al. A field guide to federated optimization. *arXiv preprint arXiv:2107.06917*, 2021.

Hao WEN and Jingsu KANG. torch_ecg: An ECG Deep Learning Framework Implemented using PyTorch, 2022. URL https://zenodo.org/record/6435048.

Yuxin Wu and Kaiming He. Group normalization. In *Proceedings of the European conference on computer vision (ECCV)*, pages 3–19, 2018.

Jie Xu, Zhenxing Xu, Peter Walker, and Fei Wang. Federated patient hashing. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 6486–6493, 2020.

Aoxiao Zhong, Hao He, Zhaolin Ren, Na Li, and Quanzheng Li. FedDAR: Federated domain-aware representation learning. In *The Eleventh International Conference on Learning Representations*, 2023. URL https://openreview.net/forum?id=6P9Y25Plj16.

Appendix A. FL methods

In this section, we describe the FL methods that are evaluated in the experiments.

A.1. FedAvg

Algorithm 1: Federated Averaging

Input: number of clients K, number of communication rounds T, number of local epochs E, Data $D := (D_1, D_2, ..., D_K)$, learning rate η
Output: model parameter w_T

Server executes:
initialize model parameters w_0
for $t = 0, ..., T-1$ **do**
  for *each client* $k \in K$ **do**
    $w_{t,k} \leftarrow w_t$
    $w_{t,k} \leftarrow \text{LocalTraining}(k,\ w_{t,k},\ D_k)$
  end
  $w_{t+1} \leftarrow \sum_{k=1}^{K} \frac{n_k}{n}\ w_{t,k}$
end

LocalTraining(k, $w_{t,k}$, D_k)**:**
for $e = 0, ..., E-1$ **do**
  for *batch* $b \leftarrow$ *(x, y) of* D_k **do**
    $w_{t,k} \leftarrow w_{t,k}$ - $\eta \nabla F_k(w_{t,k}; b)$
  end
end
return $w_{t,k}$

A.2. FedOpt

FedOpt (Reddi et al., 2020) applies adaptive optimization to model the aggregation stage in FedAvg. First, the pseudo gradient $\Delta_{t,k} := w_{t,k} - w_t$ of each client k is calculated after local training at each round t. Second, it calculates Δ_t by averaging each pseudo gradient $\Delta_{t,k}$. Third, the momentum $m_t \leftarrow \beta_1 m_{t-1} + (1-\beta_1)\Delta_t$ is calculated. Then, v_t is calculated using different adaptation techniques (FedAdam, FedAdagrad, FedYoGi) as follows:

$$\begin{aligned} &\text{FedAdam} \\ &v_t \leftarrow \beta_2 v_{t-1} + (1-\beta_2)\Delta_t^2, \\ &\text{FedAdagrad} \\ &v_t \leftarrow v_{t-1} + \Delta_t^2, \\ &\text{FedYoGi} \\ &v_t \leftarrow v_{t-1} - (1-\beta_2)\Delta_t^2 sign(v_{t-1} - \Delta_t^2), \end{aligned} \tag{4}$$

where β_1, β_2 are the hyperparameters for adaptation. Finally, a global model is updated using m_t and v_t as follows:

$$w_{t+1} \leftarrow w_t + \eta_g \frac{m_t}{\sqrt{v_t} + \gamma} \quad , \tag{5}$$

where η_g is the server learning rate and γ represents the degree of adaptivity for each algorithm.

A.3. FedPxN

Algorithm 2: Federated learning with Proximal regularization eXcept local Normalization (`FedPxN`)

Notation: number of clients K, number of communication rounds T, number of local epochs E, Data $D := (D_1, D_2, ..., D_K)$, learning rate η, normalization layers *norm*

Server executes:
initialize model parameters w_0
for $t = 0, ..., T-1$ **do**
  for *each client* $k \in K$ **do**
    $w_{t,k\backslash norm} \leftarrow w_{t\backslash norm}$
    $w_{t,k} \leftarrow \text{LocalTraining}(k,\ w_{t,k},\ D_k)$
  end
  $w_{t+1\backslash norm} \leftarrow \sum_{k=1}^{K} \frac{n_k}{n}\ w_{t,k\backslash norm}$
end

LocalTraining(k, $w_{t,k}$, D_k)**:**
for $e = 0, ..., E-1$ **do**
  for *batch* $b \leftarrow$ *(x, y) of* D_k **do**
    $R = \|w_{t,k\backslash norm} - w_{t\backslash norm}\|^2$
    $w_{t,k} \approx \arg\min_w F_k(w; b) + \frac{\mu}{2} R$
  end
end
return $w_{t,k}$

A.4. FedDyn

In FedDyn (Acar et al., 2021), the local model $w_{t,k}$ is updated by adding a penalized risk function to the objective function F_k of FedAvg during local training. The risk objective of each client is dynamically updated using both local and global model as:

$$\begin{aligned} \hat{F}_k(w_{t,k}; b) = F_k(w_{t,k}; b) - \Delta F_k(w_{t-1,k}; b) w_{t,k} \\ + \frac{\alpha}{2} \|w_{t,k} - w_{t-1}\|^2, \end{aligned} \tag{6}$$

where α is the hyperparameter controlling the degree of regularization. Theoretically, local models converge to the global model if they converge in local training of FedDyn.

Appendix B. Details of Experimental Setup

For each dataset, the total number of training epochs was set to 100, 300 and 200 respectively, which was

sufficient for all models using all FL methods to converge. After training, among model weights from all communication rounds $w_1, \ldots w_T$, the w_t that showed the best average validation performance across all clients is chosen as the final model weight. Then we use the final model weight to calculate the average test performance across all clients. We repeated all experiments three times and report the mean test performance.

eICU database The benchmark dataset in McDermott et al. (2021) contains labs, vitals, and demographic information. The labs and vitals were measured for at least 5% of all observed time-points. The benchmark dataset contains 71477 ICU stays across 59 hospitals, where ICU stays range between 540 and 4008 for each hospital. To conduct experiments in the FL setting, we used the 5, 10, 20, and 30 hospitals with the most ICU stays (with total of 14962, 25198, 39501, and 50434 ICU stays) and performed the six clinical prediction tasks :

- **Mortality prediction** (mort_24h, mort_48h)
  This task aims to predict whether the recorded time of death is within 24/48 hours.
  Input: first 24 hours of data;
  Type: binary classification.
- **Length-of-stay prediction** (LOS)
  The LOS task aims to predict whether the patient's total stay is more than three days.
  Input: first 24 hours of data;
  Type: binary classification.
- **Discharge prediction** (disch_24h, disch_48h)
  This task predicts whether the patient is discharged within the next 24/48 hours. If the patient is discharged, this task further aims to predict the next place of the patient (e.g., a skilled nursing facility or home).
  Input: first 24 hours of data;
  Type: 10-way classification.
- **Final acuity prediction** (Acuity)
  This task aims to predict whether a patient dies or is discharged. If the patient dies, it also predicts when the patient dies (e.g., in ICU or in hospital). If the patient is discharged, this task also aims to predict the next place of the patient.
  Input: first 24 hours of data;
  Type: 10-way classification.

When using the five clients with the largest samples, the label distribution of six prediction tasks is described in Appendix table 5. We used a 2-layer Transformer encoder model (Vaswani et al., 2017) followed by two fully connected (FC) layers, similar to (Song et al., 2018). We also applied Layer Normalization (LN) between the FC layers. We conducted additional experiments in which we replaced the LN used in the Transformer encoder and FC layers with Group Normalization (GN). We randomly split the ICU stays of each client using the ratio of 7:1.5:1.5 to form training, validation, and test sets. To train the models for binary classification, and 10-way classification tasks, we used the binary cross-entropy loss and cross-entropy loss, respectively. We used the Adam optimizer (Kingma and Ba, 2014), a batch size of 256, a single local epoch, 100 communication rounds, and 100 total training epochs in all of the tasks. We also tested varying combinations of the number of local epochs and communication rounds, such as 5 & 20 and 10 & 10, but 1 & 100 generally gave the best performance for all FL methods in all tasks.

Skin cancer dataset Following Cassidy et al. (2022), we used the ISIC19 (Codella et al., 2019) and HAM10000 (Tschandl et al., 2018) datasets, removing image samples that appear in both datasets from the ISIC19 dataset. Then, we split the HAM10000 (Tschandl et al., 2018) dataset into two datasets based on the source of the data sample. Hence, we formed three clients from the ISIC19 and HAM10000 datasets. The clients are from the Hospital Clinic de Barcelona (Barcelona), Medical University of Vienna, Austria (Vienna), and Queensland University, Australia (Rosendahl). In addition to these three clients, we formed two more clients, PAD-UFES (Pacheco et al., 2020) and Derm7pt (Kawahara et al., 2018), following Bdair et al. (2022). These two clients are from the Federal University of Espirito Santo, Brazil (UFES_brazil), and Simon Frazer University, Canada (SF_canada). So, our setting contains a dataset of three countries in the northern hemisphere and two countries in the southern hemisphere. Due to the ozone hole, countries in the southern hemisphere may have more skin cancer cases than in the northern hemisphere (Li et al., 2022). Then, the label distributions vary between clients. Therefore, our setting has a non-i.i.d problem that is likely in the real world. We split the data of each client into training, validation, and test sets in the ratio of 7 : 1.5 : 1.5. We used cross-entropy loss for the skin cancer classification task. Then, We also tested LN, but it showed consistently worse performance than BN or GN. We

Table 5: Label information of six tasks in eICU when using five clients with the largest samples (hospital id : 73, 264, 420, 243, 458). NaN is unlabled data. We used only labeled data in all experiments.

Task	Hospital ID	NaN	0	1	2	3	4	5	6	7	8	9	Total
mort_24h	73	6	3978	24	0	0	0	0	0	0	0	0	4008
	264	6	3438	51	0	0	0	0	0	0	0	0	3495
	420	9	2754	55	0	0	0	0	0	0	0	0	2818
	243	3	2343	35	0	0	0	0	0	0	0	0	2381
	458	3	2218	39	0	0	0	0	0	0	0	0	2260
mort_48h	73	11	3951	46	0	0	0	0	0	0	0	0	4008
	264	20	3384	91	0	0	0	0	0	0	0	0	3495
	420	21	2705	92	0	0	0	0	0	0	0	0	2818
	243	10	2322	49	0	0	0	0	0	0	0	0	2381
	458	8	2177	75	0	0	0	0	0	0	0	0	2260
LOS	73	0	2669	1339	0	0	0	0	0	0	0	0	4008
	264	0	2152	1343	0	0	0	0	0	0	0	0	3495
	420	0	1622	1196	0	0	0	0	0	0	0	0	2818
	243	0	1537	844	0	0	0	0	0	0	0	0	2381
	458	0	1512	748	0	0	0	0	0	0	0	0	2260
disch_24h	73	459	2132	1101	175	0	68	30	12	21	8	2	4008
	264	386	1979	794	215	0	44	26	10	26	0	15	3495
	420	404	1623	499	105	0	124	28	22	8	3	2	2818
	243	296	1259	667	89	0	0	37	5	25	0	3	2381
	458	345	1109	705	39	0	32	16	5	7	0	2	2260
disch_48h	73	792	1266	1456	263	0	103	59	23	29	15	2	4008
	264	555	1202	1163	373	0	76	44	15	45	0	22	3495
	420	540	1050	725	175	0	228	49	25	14	5	7	2818
	243	413	764	946	143	0	0	62	6	42	0	5	2381
	458	511	641	947	63	0	54	25	7	10	0	2	2260
Acuity	73	0	2642	609	255	138	47	46	28	2	45	196	4008
	264	0	1977	790	180	91	27	114	0	37	144	135	3495
	420	0	1379	392	555	79	42	30	11	12	220	98	2818
	243	0	1599	326	1	162	8	77	0	14	50	144	2381
	458	0	1674	128	138	54	11	18	0	6	176	55	2260

used the Adam optimizer, a batch size of 128, a single local epoch, 300 communication rounds, and 300 total training epochs. In our experiments, we resized the input images to $256 \times 256 \times 3$. In the dataset, We tested the combinations of the number of local epochs and communication rounds, such as 5 & 150 and 10 & 30, but 1 & 300 generally gave the best performance for all FL methods in all tasks.

ECG dataset As mentioned earlier, We used the PhysioNet 2021 (Reyna et al., 2021) dataset and split it into five clients. The clients are Shaoxing People's Hospital, China (Shaoxing), CPSC 2018, China (CPSC), Georgia 12-lead Challenge Database, USA (Ga), Ningbo First Hospital, China (Ningbo), and Physikalisch-Technische Bundesanstalt, Germany (PTBXL). Following Oh et al. (2022), we extracted samples with a sampling frequency 500Hz and divided the data into 5-second segments. Our aim is to solve a multi-label prediction task in which a 12-lead ECG sample is given as input, and the objective is to diagnose 26 types of cardiac diseases. We randomly split the data in the ratio of 8:1:1 to form training, validation and test sets. Then, We used ResNet-NC-SE (Kang and Wen, 2022) with asymmetric loss (Ridnik et al., 2021) because it showed the best performance on 12-lead ECG readings in PhysioNet 2021 to the best of our knowledge (Hao and Jingsu, 2022; WEN and KANG, 2022). We used the AdamW optimizer (Loshchilov and Hutter, 2018), a batch size of 64, a single local epoch, 200 communication rounds, and 200 total training epochs. For the FedOpt-based methods, we only report the results of FedAdam for the ECG experiments because the other FedOpt-based methods, FedAdagrad and FedYoGi, showed similar performance, as in the eICU and skin cancer experiments.

Appendix C. Hyperparameters of FL methods

In the section, We show the hyperparameters of FL methods. First of all, We use the same batch size for all FL methods in each dataset (256 in eICU, 128 in Skin cancer images, 64 in ECG). Then, the search space for other parameters is as follows :

- Learning rate(η) := [0.1, 0.03, 0.01, 0.003, 0.001, 0.0001]
- Mu(μ) := [1.0, 0.1, 0.01, 0.001, 0.0001]
- Feddyn alpha(α) := [0.0001, 0.001, 0.01, 0.1]
- Server learning rate(η_g) := [0.1, 0.03, 0.01, 0.003, 0.001, 0.0001]
- Tau(γ) := [0.0001, 0.001, 0.01, 0.1]

Also, the parameters to be tuned for each FL method are as follows :

- FedAvg, FedBN – η
- FedProx, FedPxN – η, μ
- FedOpt – η, η_g, γ
- FedDyn – η, α

Appendix D. More experimental results

In the section, we include more information and experimental results. (Table 6 - 8)

Table 6: AUPRC results for the eICU dataset using the data of the five largest clients. For each FL method, bold indicates the better normalization technique (LN or GN). We indicate the highest average AUPRC results for all six tasks in blue.

		FedAvg	FedProx	FedBN	FedAdam	FedAdagrad	FedYoGi	FedDyn	FedPxN
mort_24h	LN	**10.75±4.79**	**12.47±2.79**	10.72±1.77	**10.16±2.39**	**13.30±3.63**	8.94±3.91	**9.81±1.07**	13.07±2.77
	GN	10.58±1.03	12.29±2.15	**11.42±3.23**	10.06±1.75	12.46±1.32	**10.39±2.70**	8.27±2.39	**15.64±0.19**
mort_48h	LN	12.92±0.49	13.14±1.77	12.66±1.18	**14.50±0.95**	12.78±0.80	**13.35±0.55**	10.13±2.75	13.13±1.29
	GN	**13.84±0.46**	**13.20±2.09**	**14.07±0.44**	14.15±0.94	**15.77±1.38**	12.90±2.54	**14.81±2.84**	**13.90±1.41**
LOS	LN	47.69±0.39	47.49±0.04	46.87±0.09	46.61±0.39	47.03±0.14	46.85±0.62	46.36±0.29	**48.04±0.65**
	GN	**47.73±0.30**	**47.96±0.31**	**47.68±0.55**	**46.93±0.69**	**47.20±0.39**	**46.98±0.91**	**47.70±1.25**	48.00±0.39
disch_24h	LN	**21.49±0.48**	**21.38±0.51**	**22.15±0.39**	20.89±0.41	21.01±0.13	20.93±0.38	20.78±0.68	21.88±0.51
	GN	21.39±0.56	21.19±0.20	21.59±0.77	**21.00±0.32**	**21.12±0.53**	**21.09±0.53**	**22.06±0.19**	**22.21±0.22**
disch_48h	LN	22.03±1.04	**21.35±0.13**	**22.56±1.18**	**22.99±1.52**	**23.02±2.21**	22.58±1.48	21.13±0.44	**22.44±2.06**
	GN	**22.17±1.01**	21.17±0.02	21.76±1.21	22.92±1.59	22.85±1.12	**23.10±1.61**	**21.65±0.85**	22.40±1.57
Acuity	LN	**21.11±0.30**	21.28±0.06	21.22±0.20	**20.87±0.79**	**20.70±0.79**	**20.85±1.35**	21.03±0.21	**21.46±0.45**
	GN	20.97±0.13	**21.54±0.44**	**21.31±0.16**	20.59±1.20	19.46±0.30	20.57±1.32	**21.94±0.48**	21.34±0.40
Average	LN	22.66	22.85	22.70	**22.67**	22.97	22.25	21.54	23.33
	GN	**22.78**	**22.89**	**22.97**	22.61	**23.14**	**22.51**	**22.74**	**23.91**

Table 7: Average AUROC results for all six tasks for the eICU dataset using the data of the $5, 10, 20, 30$ largest clients. For each FL method and each client setting, bold indicates the better normalization (LN or GN). We indicate the highest AUROC for each client setting in blue.

	5 clients		10 clients		20 clients		30 clients	
	LN	GN	LN	GN	LN	GN	LN	GN
FedAvg	67.88	**68.01**	67.34	**67.7**	68.62	**68.87**	67.52	**68.06**
FedProx	69.44	**69.85**	68.66	**69.11**	69.31	**69.73**	68.62	**68.94**
FedBN	68.38	**68.51**	**68.24**	68.19	69.21	**69.3**	**68.43**	68.13
FedAdam	**67.67**	67.42	67.8	**67.87**	68.09	**68.96**	67.67	**68.1**
FedAdagrad	**67.38**	67.31	67.31	**67.68**	67.98	**68.85**	67.3	**67.61**
FedYoGi	65.97	**67.96**	**67.67**	66.95	68.38	**68.6**	67.66	**67.76**
FedDyn	68	**69.04**	67.43	**67.46**	67.29	**67.89**	**67.26**	66.14
`FedPxN`	69.65	**70.06**	**69.55**	69.5	70.04	**70.08**	**69.22**	69.12

Table 8: Average AUPRC results for all six tasks for the eICU dataset using the data of the $5, 10, 20, 30$ largest clients. For each FL method and each client setting, bold indicates the better normalization (LN or GN). We indicate the highest AUPRC for each client setting in blue.

	5 clients		10 clients		20 clients		30 clients	
	LN	GN	LN	GN	LN	GN	LN	GN
FedAvg	22.66	**22.78**	22.24	**22.40**	24.15	**24.41**	25.02	**25.14**
FedProx	22.85	**22.89**	22.69	**23.00**	25.10	**25.36**	25.88	**26.33**
FedBN	22.70	**22.97**	**22.53**	22.52	24.16	**24.87**	**25.26**	25.06
FedAdam	**22.67**	22.61	22.88	**22.99**	24.55	**25.28**	24.75	**25.72**
FedAdagrad	22.97	**23.14**	22.42	**22.91**	23.89	**25.22**	25.03	**25.69**
FedYoGi	22.25	**22.51**	**22.80**	22.01	24.31	**24.48**	**25.26**	25.05
FedDyn	21.54	**22.74**	**22.47**	21.73	23.81	**23.93**	**24.73**	24.48
FedPxN	23.33	**23.91**	23.28	**23.39**	25.30	**25.72**	26.06	**26.29**