FLamby: Datasets and Benchmarks for Cross-Silo Federated Learning in Realistic Healthcare Settings

Jean Ogier du Terrail[1] **Samy-Safwan Ayed**[2] **Edwige Cyffers**[3] **Felix Grimberg**[4]
Chaoyang He[5] **Regis Loeb**[1] **Paul Mangold**[3] **Tanguy Marchand**[1]
Othmane Marfoq[2] **Erum Mushtaq**[6] **Boris Muzellec**[1] **Constantin Philippenko**[7]
Santiago Silva[2] **Maria Teleńczuk**[1] **Shadi Albarqouni**[8,9] **Salman Avestimehr**[5,6]
Aurélien Bellet[3] **Aymeric Dieuleveut**[7] **Martin Jaggi**[4]
Sai Praneeth Karimireddy[10] **Marco Lorenzi**[2] **Giovanni Neglia**[2] **Marc Tommasi**[3]
Mathieu Andreux[1]
[1]Owkin, Inc, [2] Inria, Université Côte d'Azur, Sophia Antipolis, France
[3]Univ. Lille, Inria, CNRS, Centrale Lille, UMR 9189 - CRIStAL, F-59000 Lille, France
[4]EPFL [5]FedML, Inc. [6]University of Southern California
[7]CMAP, UMR 7641, École Polytechnique, Institut Polytechnique de Paris
[8]University Hospital Bonn [9]Helmholtz Munich [10]University of California, Berkeley
{jean.du-terrail, regis.loeb, tanguy.marchand, boris.muzellec,
maria.telenczuk, mathieu.andreux}@owkin.com, {samy-safwan.ayed,
edwige.cyffers, paul.mangold, othamne.marfoq,
santiago-smith.silva-rincon, aurelien.bellet,
marco.lorenzi, giovanni.neglia, marc.tommasi}@inria.fr
{felix.grimberg, martin.jaggi}@epfl.ch,
ch@fedml.ai, {emushtaq, avestime}@usc.edu,
{constantin.philippenko, aymeric.dieuleveut}@polytechnique.edu,
shadi.albarqouni@ukbonn.de, sp.karimireddy@berkeley.edu

Abstract

Federated Learning (FL) is a novel approach enabling several clients holding sensitive data to collaboratively train machine learning models, without centralizing data. The cross-silo FL setting corresponds to the case of few (2–50) reliable clients, each holding medium to large datasets, and is typically found in applications such as healthcare, finance, or industry. While previous works have proposed representative datasets for cross-device FL, few realistic healthcare cross-silo FL datasets exist, thereby slowing algorithmic research in this critical application. In this work, we propose a novel cross-silo dataset suite focused on healthcare, FLamby (Federated Learning AMple Benchmark of Your cross-silo strategies), to bridge the gap between theory and practice of cross-silo FL. FLamby encompasses 7 healthcare datasets with natural splits, covering multiple tasks, modalities, and data volumes, each accompanied with baseline training code. As an illustration, we additionally benchmark standard FL algorithms on all datasets. Our flexible and modular suite allows researchers to easily download datasets, reproduce results and re-use the different components for their research. FLamby is available at www.github.com/owkin/flamby.

1 Introduction

Recently it has become clear that, in many application fields, impressive machine learning (ML) task performance can be reached by scaling the size of both ML models and their training data while

36th Conference on Neural Information Processing Systems (NeurIPS 2022) Track on Datasets and Benchmarks.

keeping existing well-performing architectures mostly unaltered [118, 76, 24, 109]. In this context, it is often assumed that massive training datasets can be collected and centralized in a single client in order to maximize performance. However, in many application domains, data collection occurs in distinct sites (further referred to as clients, e.g., mobile devices or hospitals), and the resulting local datasets cannot be shared with a central repository or data center due to privacy or strategic concerns [42, 18].

To enable cooperation among clients given such constraints, Federated Learning (FL) [99, 73] has emerged as a viable alternative to train models across data providers without sharing sensitive data. While initially developed to enable training across a large number of small clients, such as smartphones or Internet of Things (IoT) devices, it has been then extended to the collaboration of fewer and larger clients, such as banks or hospitals. The two settings are now respectively referred to as *cross-device* FL and *cross-silo* FL, each associated with specific use cases and challenges [73].

On the one hand, cross-device FL leverages edge devices such as mobile phones and wearable technologies to exploit data distributed over billions of data sources [99, 16, 14, 103]. Therefore, it often requires solving problems related to edge computing [53, 87, 129], participant selection [73, 131, 23, 44], system heterogeneity [73], and communication constraints such as low network bandwidth and high latency [113, 93, 51]. On the other hand, cross-silo initiatives enable to untap the potential of large datasets previously out of reach. This is especially true in healthcare, where the emergence of federated networks of private and public actors [112, 115, 105], for the first time, allows scientists to gather enough data to tackle open questions on poorly understood diseases such as triple negative breast cancer [40] or COVID-19 [34]. In cross-silo applications, each silo has large computational power, a relatively high bandwidth, and a stable network connection, allowing it to participate to the whole training phase. However, cross-silo FL is typically characterized by high inter-client dataset heterogeneity and biases of various types across the clients [105, 40].

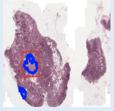
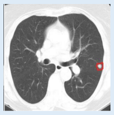
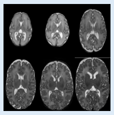
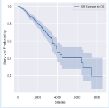
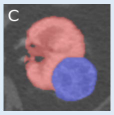
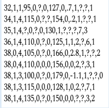
As we show in Section 2, publicly available datasets for the cross-silo FL setting are scarce. As a consequence, researchers usually rely on heuristics to artificially generate heterogeneous data partitions from a single dataset and assign them to hypothetical clients. Such heuristics might fall short of replicating the complexity of natural heterogeneity found in real-world datasets. The example of digital histopathology [126], a crucial data type in cancer research, illustrates the potential limitations of such synthetic partition methods. In digital histopathology, tissue samples are extracted from patients, stained, and finally digitized. In this process, known factors of data heterogeneity across hospitals include patient demographics, staining techniques, storage methodologies of the physical slides, and digitization processes [71, 45, 59]. Although staining normalization [81, 35] has seen recent progress, mitigating this source of heterogeneity, the other highlighted sources of heterogeneity are difficult to replicate with synthetic partitioning [59] and some may be unknown, which calls for actual cross-silo cohort experiments. This observation is also valid for many other application domains, e.g. radiology [52], dermatology [10], retinal images [10] and more generally computer vision [122].

In order to address the lack of realistic cross-silo datasets, we propose FLamby, an open source cross-silo federated dataset suite with natural partitions focused on healthcare, accompanied by code examples, and benchmarking guidelines. Our ambition is that FLamby becomes the reference benchmark for cross-silo FL, as LEAF [19] is for cross-device FL. To the best of our knowledge, apart from some promising isolated works to build realistic cross-silo FL datasets (see Section 2), our work is the first standard benchmark allowing to systematically study healthcare cross-silo FL on different data modalities and tasks.

To summarize, our contributions are threefold:

1. We build an open-source federated cross-silo healthcare dataset suite including 7 datasets. These datasets cover different tasks (classification / segmentation / survival) in multiple application domains and with different data modalities and scale. Crucially, all datasets are partitioned using natural splits.
2. We provide guidelines to help compare FL strategies in a fair and reproducible manner, and provide illustrative results for this benchmark.
3. We make open-source code accessible for benchmark reproducibility and easy integration in different FL frameworks, but also to allow the research community to contribute to FLamby development, by adding more datasets, benchmarking types and FL strategies.

Table 1: Overview of the datasets, tasks, metrics and baseline models in FLamby. For Fed-Camelyon16 the two different sizes refer to the size of the dataset before and after tiling.

Dataset	Fed-Camelyon16	Fed-LIDC-IDRI	Fed-IXI	Fed-TCGA-BRCA	Fed-KITS2019	Fed-ISIC2019	Fed-Heart-Disease
Input (x)	Slides	CT-scans	T1WI	Patient info.	CT-scans	Dermoscopy	Patient info.
Preprocessing	Matter extraction + tiling	Patch Sampling	Registration	None	Patch Sampling	Various image transforms	Removing missing data
Task type	binary classification	3D segmentation	3D segmentation	survival	3D segmentation	multi-class classification	binary classification
Prediction (y)	Tumor on slide	Lung Nodule Mask	Brain mask	Risk of death	Kidney and tumor masks	Melanoma class	Heart disease
Center extraction	Hospital	Scanner Manufacturer	Hospital	Group of Hospitals	Group of Hospitals	Hospital	Hospital
Thumbnails							32,1,1,95,0,?,0,127,0,.7,1,?,?,1 34,1,4,115,0,?,?,154,0,.2,1,?,?,1 35,1,4,?,0,?,0,130,1,?,?,?,7,3 36,1,4,110,0,?,0,125,1,1,2,?,6,1 38,0,4,105,0,?,0,166,0,2.8,1,?,?,2 38,0,4,110,0,0,0,156,0,0,2,?,3,1 38,1,3,100,0,?,0,179,0,-1.1,1,?,?,0 38,1,3,115,0,0,0,128,1,0,2,?,7,1 38,1,4,135,0,?,0,150,0,0,?,?,3,2
Original paper	Litjens *et al.* 2018	Armato *et al.* 2011	Perez *et al.* 2021	Liu *et al.* 2018	Heller *et al.* 2019	Tschandl *et al.* 2018 / Codella *et al.* 2017 / Combalia *et al.* 2019	Janosi *et al.* 1988
# clients	2	5	3	5	6	5	4
# examples	399	1,018	566	1, 088	96	23, 247	740
# examples per center	239, 150	670, 205, 69, 74	311, 181, 74	311, 196, 206, 162 51	12, 14, 12, 12, 16, 30	12413, 3954, 3363, 225 819, 439	303, 261, 46, 130
Model	DeepMIL [66]	Vnet [100, 102]	3D U-net [25]	Cox Model [33]	nnU-Net [69]	efficientnet [119] + linear layer	Logistic Regression
Metric	AUC	DICE	DICE	C-index	DICE	Balanced Accuracy	Accuracy
Size	50G (850G total)	115G	444M	115K	54G	9G	40K
Image resolution	0.5 µm / pixel	$\sim$1.0 × 1.0 × 1.0 mm / voxel	$\sim$ 1.0 × 1.0 × 1.0 mm / voxel	NA	$\sim$1.0 × 1.0 × 1.0 mm / voxel	$\sim$0.02 mm / pixel	NA
Input dimension	10, 000 x 2048	128 x 128 x 128	48 x 60 x 48	39	64 x 192 x 192	200 x 200 x 3	13

This paper is organized as follows. Section 2 reviews existing FL datasets and benchmarks, as well as client partition methods used to artificially introduce data heterogeneity. In Section 3, we describe our dataset suite in detail, notably its structure and the intrinsic heterogeneity of each federated dataset. Finally, we define a benchmark of several FL strategies on all datasets and provide results thereof in Section 4.

2 Related Work

In FL, data is collected locally in clients in different conditions and without coordination. As a consequence, clients' datasets differ both in size (unbalanced) and in distribution (non-IID) [99]. The resulting *statistical heterogeneity* is a fundamental challenge in FL [84, 73], and it is necessary to take it into consideration when evaluating FL algorithms. Most FL papers simulate statistical heterogeneity by artificially partitioning classic datasets, e.g., CIFAR-10/100 [80], MNIST [83] or ImageNet [37], on a given number of clients. Common approaches to produce synthetic partitions of classification datasets include associating samples from a limited number of classes to each client [99], Dirichlet sampling on the class labels [61, 133], and using Pachinko Allocation Method (PAM) [86, 110] (which is only possible when the labels have a hierarchical structure). In the case of regression tasks, [107] partitions the *superconduct* dataset [20] across 20 clients using Gaussian Mixture clustering based on T-SNE representations [124] of the features. Such synthetic partition approaches may fall short of modelling the complex statistical heterogeneity of real federated datasets. Evaluating FL strategies on datasets with natural client splits is a safer approach to ensuring that new strategies address real-world issues.

For *cross-device* FL, the LEAF dataset suite [19] includes five datasets with natural partition, spanning a wide range of machine learning tasks: natural language modeling (Reddit [127]), next character prediction (Shakespeare [99]), sentiment analysis (Sent140 [47]), image classification (CelebA [90]) and handwritten-character recognition (FEMNIST [28]). TensorFlow Federated [15] complements LEAF and provides three additional naturally split federated benchmarks, i.e., StackOverflow [120], Google Landmark v2 [62] and iNaturalist [125]. Further, FLSim [111] provides cross-device examples based on LEAF and CIFAR10 [80] with a synthetic split, and FedScale [82] introduces a large FL benchmark focused on mobile applications. Apart from iNaturalist, the aforementioned datasets target the cross-device setting.

To the best of our knowledge, no extensive benchmark with natural splits is available for *cross-silo* FL. However, some standalone works built cross-silo datasets with real partitions. [48] and [97] partition Cityscapes [30] and iNaturalist [125], respectively, exploiting the geolocation of the picture acquisition site. [60] releases a real-world, geo-tagged dataset of common mammals on Flickr. [94] gathers a federated cross-silo benchmark for object detection created using street cameras. [31] partitions Vehicle Sensor Dataset [41] and Human Activity Recognition dataset [7] by sensor and by individuals, respectively. [95] builds an iris recognition federated dataset across five clients using multiple iris datasets [128, 135, 136, 108]. While FedML [55] introduces several cross-silo benchmarks [56, 132, 54], the related client splits are synthetically obtained with Dirichlet sampling and not based on a natural split. Similarly, FATE [1] provides several cross-silo examples but, to the best of our knowledge, none of them stems from a natural split.

In the medical domain, several works use natural splits replicating the data collection process in different hospitals: the works [5, 21, 11, 74, 130, 22] respectively use the Camelyon datasets [89, 13, 12], the CheXpert dataset [67], LIDC dataset [8], the chest X-ray dataset [78], the IXI dataset [130], the Kaggle diabetic retinopathy detection dataset [49]. Finally, the works [6, 50, 91] use the TCGA dataset [121] by extracting the Tissue Source site metadata.

Our work aims to give more visibility to such isolated cross-silo initiatives by regrouping seven medical datasets, some of which listed above, in a single benchmark suite. We also provide reproducible code alongside precise benchmarking guidelines in order to connect past and subsequent works for a better monitoring of the progress in cross-silo FL.

3 The FLamby Dataset Suite

3.1 Structure Overview

The FLamby datasets suite is a Python library organized in two main parts: datasets with corresponding baseline models, and FL strategies with associated benchmarking code. The suite is modular, with a standardized simple application programming interface (API) for each component, enabling easy re-use and extensions of different components. Further, the suite is compatible with existing FL software libraries, such as FedML [55], Fed-BioMed [117], or Substra [46]. Listing 1 provides a code example of how the structure of FLamby allows to test new datasets and strategies in a few lines of code, and Table 1 provides an overview of the FLamby datasets.

Dataset and baseline model. The FLamby suite contains datasets with a natural notion of client split, as well as a predefined task and associated metric. A train/test set is predefined for each client to enable reproducible comparisons. We further provide a baseline model for each task, with a reference implementation for training on pooled data. For each dataset, the suite provides documentation, metadata and helper functions to: 1. download the original pooled dataset; 2. apply preprocessing if required, making it suitable for ML training; 3. split each original pooled dataset between its natural clients; and 4. easily iterate over the preprocessed dataset. The dataset API relies on PyTorch [104], which makes it easy to iterate over the dataset with natural splits as well as to modify these splits if needed.

FL strategies and benchmark. FL training algorithms, called *strategies* in the FLamby suite, are provided for simulation purposes. In order to be agnostic to existing FL libraries, these strategies are provided in plain Python code. The API of these strategies is standardized and compatible with the dataset API, making it easy to benchmark each strategy on each dataset. We further provide a script performing such a benchmark for illustration purposes. We stress the fact that it is easy to alternatively use implementations from existing FL libraries.

3.2 Datasets, Metrics and Baseline Models

We provide a brief description of each dataset in the FLamby dataset suite, which is summarized in Table 1. In Section 3.4, we further explore the heterogeneity of each dataset, as displayed in Figure 1.

Fed-Camelyon16. Camelyon16 [89] is a histopathology dataset of 399 digitized breast biopsies' slides with or without tumor collected from two hospitals: Radboud University Medical Center (RUMC) and University Medical Center Utrecht (UMCU). By recovering the original split information we build a federated version of Camelyon16 with **2** clients. The task consists in binary classification of each slide, which is challenging due to the large size of each image ($10^5 \times 10^5$ pixels at 20X magnification), and measured by the Area Under the ROC curve (AUC).

As a baseline, we follow a weakly-supervised learning approach. Slides are first converted to bags of local features, which are one order of magnitude smaller in terms of memory requirements, and a model is then trained on top of this representation. For each slide, we detect regions with a matter-detection network and then extract features from each tile with an ImageNet-pretrained Resnet50, following state-of-the-art practice [32, 92]. Note that due to the imbalanced distribution of tissue in the different slides, a different number of features is produced for each slide: we cap the total number of tiles to 10^5 and use zero-padding for consistency. We then train a DeepMIL architecture [65], using its reference implementation [66] and hyperparameters from [36]. We refer to Appendix C for more details.

Fed-LIDC-IDRI. LIDC-IDRI [8, 64, 26] is an image database [26] study with 1018 CT-scans (3D images) from The Cancer Imaging Archive (TCIA), proposed in the LUNA16 competition [114]. The task consists in automatically segmenting lung nodules in CT-scans, as measured by the DICE score [39]. It is challenging because lung nodules are small, blurry, and hard to detect. By parsing the metadata of the CT-scans from the provided annotations, we recover the manufacturer of each scanning machine used, which we use as a proxy for a client. We therefore build a **4**-client federated version of this dataset, split by manufacturer. Figure 1b displays the distribution of voxel intensities in each client.

As a baseline model, we use a VNet [100] following the implementation from [102]. This model is trained by sampling 3D-volumes into 3D patches fitting in GPU memory. Details of the sampling procedure are available in Appendix D.

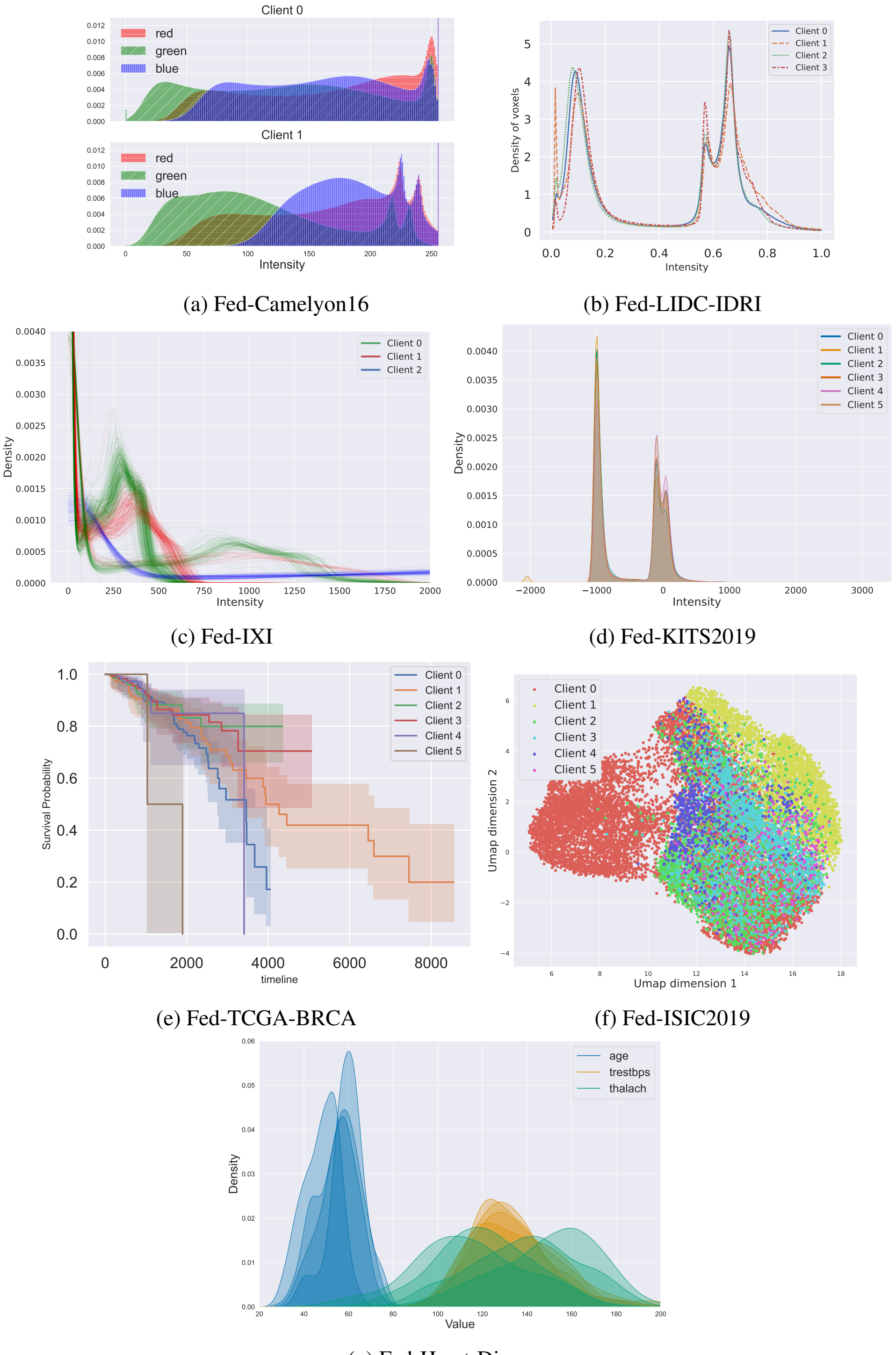
Fed-IXI. This dataset is extracted from the Information eXtraction from Images - IXI database [38], and has been previously released by Perez *et* al. [2, 106] under the name of *IXITiny*. IXITiny provides a database of brain T1 magnetic resonance images (MRIs) from **3** hospitals (Guys, HH, and IOP). This dataset has been adapted to a brain segmentation task by obtaining spatial brain masks using a state-of-the-art unsupervised brain segmentation tool [63]. The quality of the resulting supervised segmentation task is measured by the DICE score [39].

The image pre-processing pipeline includes volume resizing to $48 \times 60 \times 48$ voxels, and sample-wise intensity normalization. Figure 1c highlights the heterogeneity of the raw MRI intensity distributions between clients. As a baseline, we use a 3D U-net [25] following the implementation of [3]. Appendix E provides more detailed information about this dataset, including demographic information, and about the baseline.

Fed-TCGA-BRCA. The Cancer Genome Atlas (TCGA)'s Genomics Data Commons (GDC) portal [101] contains multi-modal data (tabular, 2D and 3D images) on a variety of cancers collected in many different hospitals. Here, we focus on clinical data from the BReast CAncer study (BRCA), which includes features gathered from 1066 patients. We use the Tissue Source Site metadata to split data based on extraction site, grouped into geographic regions to obtain large enough clients. We end up with **6** clients: USA (Northeast, South, Middlewest, West), Canada and Europe, with patient counts varying from 51 to 311. The task consists in predicting survival outcomes [72] based on the patients' tabular data (39 features overall), with the event to predict being death. This survival task is akin to a ranking problem with the score of each sample being known either directly or only by lower bound (right censorship). The ranking is evaluated by using the concordance index (C-index) that measures the percentage of correctly ranked pairs while taking censorship into account.

As a baseline, we use a linear Cox proportional hazard model [33] to predict time-to-death for patients. Figure 1e highlights the survival distribution heterogeneity between the different clients. Appendix F provides more details on this dataset.

Fed-KITS2019. The KiTS19 dataset [57, 58] stems from the Kidney Tumor Segmentation Challenge 2019 and contains CT scans of 210 patients along with the segmentation masks from 79 hospitals. We recover the hospital metadata and extract a **6**-client federated version of this dataset by removing hospitals with less than 10 training samples. The task consists of both kidney and tumor segmentation, labeled 1 and 2, respectively, and we measure the average of Kidney and Tumor DICE scores [39] as our evaluation metric.



(a) Fed-Camelyon16

(b) Fed-LIDC-IDRI

(c) Fed-IXI

(d) Fed-KITS2019

(e) Fed-TCGA-BRCA

(f) Fed-ISIC2019

(g) Fed-Heart-Disease

Figure 1: Heterogeneity of FLamby datasets. Best seen in color. 1a: Color histograms per client. 1b, 1c and 1d: Voxel intensity distribution per client. 1e: Kaplan-Meier survival curves per client. 1f: UMAP of deep network features of the raw images, colored by client. 1g: Per-client histograms of several features. Differences between client distributions are sometimes obvious and sometimes subtle. Some clients are close in the feature space, some are not and different types of heterogeneity are observed with different data modalities.

The preprocessing pipeline comprises intensity clipping followed by intensity normalization, and resampling of all the cases to a common voxel spacing of 2.90x1.45x1.45 mm. As a baseline, we use the nn-Unet library [69] to train a 3D nnU-Net, combined with multiple data augmentations including scaling, rotations, brightness, contrast, gamma and Gaussian noise with the batch generators framework [68]. Appendix G provides more details on this dataset.

Fed-ISIC2019. The ISIC2019 dataset [123, 27, 29] contains dermoscopy images collected in 4 hospitals. We restrict ourselves to 23,247 images from the public train set due to metadata availability reasons, which we re-split into train and test sets. The task consists in image classification among 8 different melanoma classes, with high label imbalance (prevalence ranging 49% to less than 1% depending on the class). We split this dataset based on the imaging acquisition system used: as one hospital used 3 different imaging technologies throughout time, we end up with a **6**-client federated version of ISIC2019. We measure classification performance through balanced accuracy, defined as the average recall on each class.

As an offline preprocessing step, we follow recommendations and code from [9] by resizing images to the same shorter side while maintaining their aspect ratio, and by normalizing images' brightness and contrast through a color consistency algorithm. As a baseline classification model, we fine-tune an EfficientNet [119] pretrained on ImageNet, with a weighted focal loss [88] and with multiple data augmentations. Figure 1f highlights the heterogeneity between the different clients prior to preprocessing. Appendix H provides more details on this dataset.

Fed-Heart-Disease. The Heart-Disease dataset [70] was collected in **4** hospitals in the USA, Switzerland and Hungary. This dataset contains tabular information about 740 patients distributed among these four clients. The task consists in binary classification to assess the presence or absence of heart disease. We preprocess the dataset by removing missing values and encoding non-binary categorical variables as dummy variables, which gives 13 relevant attributes. As a baseline model, we use logistic regression. Appendix I provides more details on this dataset.

3.3 Federated Learning Strategies in FLamby

The following standard FL algorithms, called *strategies*, are implemented in FLamby. We rely on a common API for all strategies, which allows for efficient benchmarking of both datasets and strategies, as shown in Listing 1. As we focus on the cross-silo setting, we restrict ourselves to strategies with full client participation.

FedAvg [99]. FedAvg is the simplest FL strategy. It performs iterative round-based training, each round consisting in local mini-batch updates on each client followed by parameter averaging on a central server. As a convention, we choose to count the number of local updates in batches and not in local epochs in order to match theoretical formulations of this algorithm; this choice also applies to strategies derived from FedAvg. This strategy is known to be sensitive to heterogeneity when the number of local updates grows [85, 77].

FedProx [85]. In order to mitigate statistical heterogeneity, FedProx builds on FedAvg by introducing a regularization term to each local training loss, thereby controlling the deviation of the local models from the last global model.

Scaffold [77]. Scaffold mitigates client drifts using control-variates and by adding a server-side learning rate. We implement a full-participation version of Scaffold that is optimized to reduce the number of bits communicated between the clients and the server.

Cyclic Learning [22, 116]. Cyclic Learning performs local optimizations on each client in a sequential fashion, transferring the trained model to the next client when training finishes. Cyclic is a simple sequential baseline to other federated strategies. For Cyclic, we define a round as a full cycle throughout all clients. We implement both such cycles in a fixed order or in a shuffled order at each round.

FedAdam [110], **FedYogi [110]**, **FedAdagrad [110].** FedAdam, FedYogi and FedAdagrad are generalizations of their respective single-centric optimizers (Adam [79], Yogi [134] and Adagrad [96]) to the FL setting. In all cases, the running means and variances of the updates are tracked at the server level.

```python
# Import relevant dataset, strategy, and utilities
from flamby.datasets.fed_camelyon16 import FedCamelyon16, Baseline, BaselineLoss, NUM_CLIENTS, metric
from flamby.strategies import FedProx
from flamby.utils import evaluate_model_on_tests, get_nb_max_rounds

# Define number of local updates and number of rounds
num_updates = 100
num_rounds = get_nb_max_rounds(num_updates)
# Dataloaders for train and test
training_dataloaderss = [
    DataLoader(FedCamelyon16(center=i, train=True, pooled=False), batch_size=BATCH_SIZE, shuffle=True)
    for i in range(NUM_CLIENTS)
]
test_dataloaders = [
    DataLoader(FedCamelyon16(center=i, train=False, pooled=False), batch_size=BATCH_SIZE, shuffle=False)
    for i in range(NUM_CLIENTS)
]
# Define local model and loss
model_baseline = Baseline()
loss_baseline = BaselineLoss()
# Define and train strategy
strategy = FedProx(training_dataloaders, model_baseline, loss_baseline, torch.optim.SGD, LR, num_updates, num_rounds)
model_final = strategy.run()[0]
# Evaluate final FL model on test sets
results_per_client = evaluate_model_on_tests(model_final, test_dataloaders, metric)
```

Listing 1: Code example from the FLamby dataset suite: on the Fed-Camelyon16 dataset, we use the FedProx Federated Learning strategy to train the pre-implemented baseline model.

3.4 Dataset Heterogeneity

We qualitatively illustrate the heterogeneity of the datasets of FLamby. For each dataset, we compute a relevant statistical distribution for each client, which differs due to the differences in tasks and modalities of the datasets. We comment the results displayed in Figure 1 in the following. Appendix M provides a more quantitative exploration of this heterogeneity.

For the **Fed-Camelyon16** dataset, we display the color histograms (RGB values) of the raw tissue patches in each client. We see that the RGB distributions of both clients strongly differ. For both **Fed-LIDC-IDRI** and **Fed-KITS2019** datasets, we display histograms of voxel intensities. In both cases, we do not note significant differences between clients. For the **Fed-IXI** dataset, we display the histograms of raw T1-MRI images, showing visible differences between clients. For **Fed-TCGA-BRCA**, we display Kaplan-Meier estimations of the survival curves [75] in each client. As detailed in Appendix F, pairwise log-rank tests demonstrate significant differences between some clients, but not all. For the **Fed-ISIC2019**, we use a 2-dimensional UMAP [98] plot of the features extracted from an Imagenet-pretrained Efficientnetv1 on the raw images. We see that some clients are isolated in distinct clusters, while others overlap, highlighting the heterogeneity of this dataset. Last, for the **Fed-Heart-Disease** dataset, we display histograms for a subset of features (age, resting blood pressure and maximum heart rate), showing that feature distributions vary between clients.

4 FL Benchmark Example with FLamby

In this section, we detail the guidelines we follow to perform a benchmark and provide results thereof. These guidelines might be used in the future to facilitate fair comparisons between potentially novel FL strategies and existing ones. However, we stress that FLamby also allows for any other experimental setup thanks to its modular structure, as we showcase in Appendices L.1 and L.2. The FLamby suite further provides a script to automatically reproduce this benchmark based on configuration files.

Train/test split. We use the per-client train/test splits, including all clients for training. Performance is evaluated on each local test dataset, and then averaged across the clients. We exclude model personalization from this benchmark: therefore, a single model is evaluated at the end of training. We refer to Appendix L.2 for more results with model personalization.

Hyperparameter tuning and Baselines. We distinguish two kinds of hyperparameters: those related to the machine learning (ML) part itself, and those related to the FL strategy. We tune these parameters separately, starting with the machine learning part. All experiments are repeated with 5 independent runs, except for FED-LIDC-IDRI where only 1 training is performed due to a long training time.

For each dataset, the ML hyperparameters include the model architecture, the loss and related hyperparameters, including local batch size. These ML hyperparameters are carefully tuned with cross-validation on the pooled training data. The resulting ML model gives rise to the **pooled baseline**. We use the same ML hyperparameters for training on each client individually, leading to **local baselines**.

For the FL strategies, hyperparameters include e.g. local learning rate, server learning rate, and other relevant quantities depending on the strategies. For each dataset and each FL strategy, we use the same model as in the pooled and local baselines, with fixed hyperparameters. We then only optimize FL strategies-related hyperparameters.

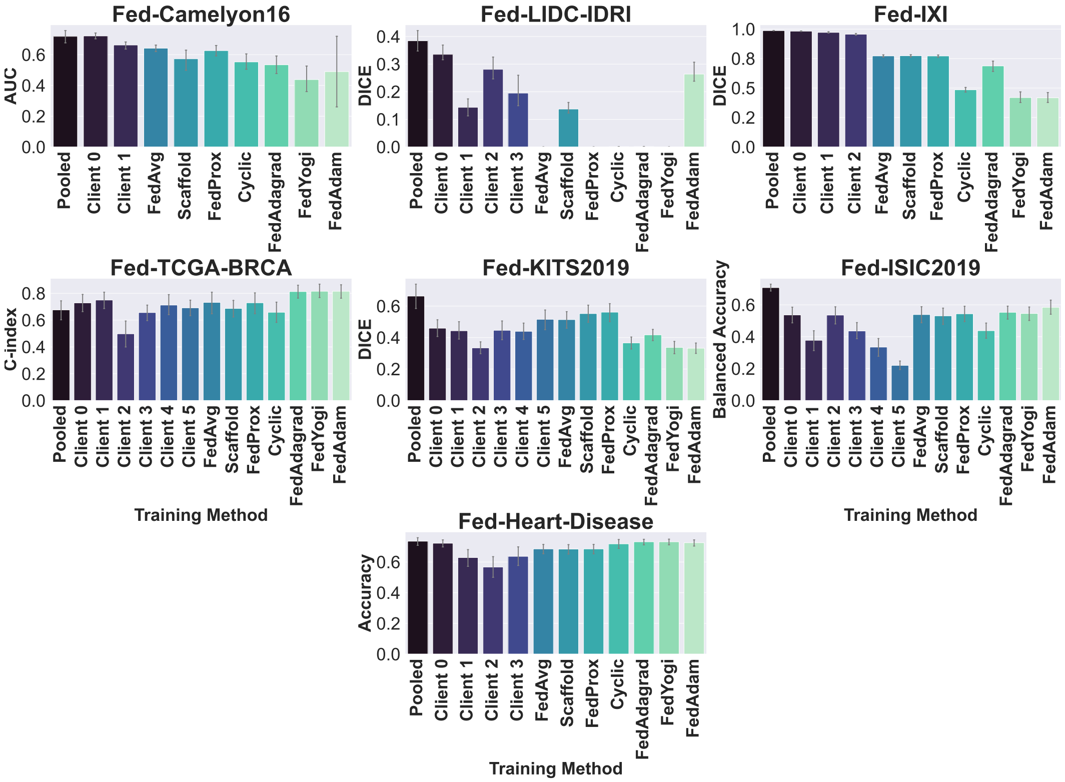


Figure 2: Benchmark results on FLamby for each dataset. For all metrics, higher is better, see Section 3.2 for metric details and Section 4 for experimental details. For Fed-LIDC-IDRI, multiple strategies fail converging, leading to zero DICE. Except for Fed-TCGA-BRCA and Fed-Heart-Disease, federated strategies fall short of reaching the pooled performance, but improve over the local ones.

Federated setup. For all strategies and datasets, the number of rounds $T_{\max}$ is fixed to perform approximately as many epochs on each client as is necessary to get good performance when data is pooled. Note that, as we use a single batch size B and a fixed number E of local steps, the notion of epoch is ill-defined; we approximate it as follows. Given n^P_{epochs}, the number of epochs required to train the baseline model for the pooled dataset, n_T the total number of samples in the distributed training set, K the number of clients, we define

$$T_{\max} = n^P_{epochs} \cdot \lfloor n_T/K/B/E \rfloor \tag{1}$$

where $\lfloor \cdot \rfloor$ denotes the floor operation. In our benchmark, we use $E = 100$ local updates for all datasets. Note that this restriction in the total number of rounds may have an impact on the convergence of federated strategies. We refer to Appendix J for more details on this benchmark.

Benchmark results. The test results of the benchmark are displayed in Figure 2. Note that test results are uniformly averaged over the different local clients. We observe strikingly different behaviours across datasets.

No local training or FL strategy is able to reach a performance on par with the pooled training, except for Fed-TCGA-BRCA and Fed-Heart-Disease. It is remarkable that both of them are tabular, low-dimensional datasets, with only linear models. Still, for Fed-KITS2019 and Fed-ISIC2019, some FL strategies outperform local training, showing the benefit of collaboration, but falling short of reaching pooled performance. For Fed-Camelyon16, Fed-LIDC-IDRI and Fed-IXI, the current results do not indicate any benefit in collaborative training.

Among FL strategies, we note that for the datasets where an FL strategy outperforms the pooled baselines, FedOpt variants (FedAdagrad, FedYogi and FedAdam) reach the best performance. Further, the Cyclic baseline systematically underperforms other strategies. Last, but not least, FedAvg does not reach top performance among FL strategies, except for Fed-Camelyon16 and Fed-IXI, it remains a competitive baseline strategy.

These results show the difficulty of tuning properly FL strategies, especially in the case of heterogeneous cross-silo datasets. This calls for the development of more robust FL strategies in this setting.

5 Conclusion

In this article we introduce FLamby, a modular dataset suite and benchmark, comprising multiple tasks and data modalities, and reflecting the heterogeneity of real-world healthcare cross-silo FL use cases. This comprehensive benchmark is needed to advance the understanding of cross-silo healthcare data collection on FL performance.

Currently, FLamby is limited to healthcare datasets. In the longer run and with the help of the FL community, it could be enriched with datasets from other application domains to better reflect the diversity of cross-silo FL applications, which is possible thanks to its modular design. Regarding machine learning backends, FLamby only provides PyTorch [104] code: supporting other backends, such as TensorFlow [4] or JAX [17], is a relevant future direction if there is such demand from the community. Further, our benchmark currently does not integrate all constraints of cross-silo FL, especially privacy aspects, which are important in this setting.

In terms of FL setting, the benchmark mainly focuses on the heterogeneity induced by natural splits. In order to make it more realistic, future developments might include in depth study of Differential Privacy (DP) training [42], cryptographic protocols such as Secure Aggregation [16], Personalized FL [43], or communication constraints [113] when applicable. As we showcase in Appendices L.1 for DP and L.2 for personalization, the structure of FLamby makes it possible to quickly tackle such questions. We hope that the scientific community will use FLamby for cross-silo research purposes on real data, and contribute to further develop it, making it a reference for this research topic.

Acknowledgments and Disclosure of Funding

The authors thank the anonymous reviewers, ethics reviewer, and meta-reviewer for their feedback and ideas, which significantly improved the paper and the repository. The authors listed as Owkin, Inc. employees are supported by Owkin, Inc. The works of E.M. and S.A. is supported, in part, by gifts from Intel and Konica Minolta. This work was supported by the Swiss State Secretariat for Education, Research and Innovation (SERI) under contract number 22.00133, by the Inria Explorator Action FLAMED and by the French National Research Agency (grant ANR-20-CE23-0015, project PRIDE and ANR-20-THIA-0014 program `AI_PhD@Lille`). This project has also received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 847581 and is co-funded by the Region Provence-Alpes-Côte d'Azur and IDEX UCAJEDI. A.D.'s research was supported by the *Statistics and Computation for AI* ANR Chair and by *Hi!Paris*. C.P. received support from *Accenture Labs* (Sophia Antipolis, France).

References

[1] Fate (federated ai technology enabler). `https://github.com/FederatedAI/FATE`. Accessed: 2022-10-12.

[2] Ixitiny dataset. `https://torchio.readthedocs.io/datasets.html#torchio.datasets.ixi.IXITiny`. Accessed: 2022-05-18.

[3] unet 0.7.7. `https://pypi.org/project/unet/0.7.7/`. Accessed: 2022-02-02.

[4] Martín Abadi, Ashish Agarwal, Paul Barham, Eugene Brevdo, Zhifeng Chen, Craig Citro, Greg S. Corrado, Andy Davis, Jeffrey Dean, Matthieu Devin, Sanjay Ghemawat, Ian Goodfellow, Andrew Harp, Geoffrey Irving, Michael Isard, Yangqing Jia, Rafal Jozefowicz, Lukasz Kaiser, Manjunath Kudlur, Josh Levenberg, Dandelion Mané, Rajat Monga, Sherry Moore, Derek Murray, Chris Olah, Mike Schuster, Jonathon Shlens, Benoit Steiner, Ilya Sutskever, Kunal Talwar, Paul Tucker, Vincent Vanhoucke, Vijay Vasudevan, Fernanda Viégas, Oriol Vinyals, Pete Warden, Martin Wattenberg, Martin Wicke, Yuan Yu, and Xiaoqiang Zheng. TensorFlow: Large-scale machine learning on heterogeneous systems, 2015. Software available from tensorflow.org.

[5] 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.

[6] Mathieu Andreux, Andre Manoel, Romuald Menuet, Charlie Saillard, and Chloé Simpson. Federated survival analysis with discrete-time Cox models. *arXiv preprint arXiv:2006.08997*, 2020.

[7] Davide Anguita, Alessandro Ghio, Luca Oneto, Xavier Parra Perez, and Jorge Luis Reyes Ortiz. A public domain dataset for human activity recognition using smartphones. In *Proceedings of the 21th international European symposium on artificial neural networks, computational intelligence and machine learning*, pages 437–442, 2013.

[8] Samuel G Armato III, Geoffrey McLennan, Luc Bidaut, Michael F McNitt-Gray, Charles R Meyer, Anthony P Reeves, Binsheng Zhao, Denise R Aberle, Claudia I Henschke, Eric A Hoffman, et al. The lung image database consortium (LIDC) and image database resource initiative (IDRI): a completed reference database of lung nodules on ct scans. *Medical physics*, 38(2):915–931, 2011.

[9] Aman Arora. Siim-isic melanoma classification - my journey to a top 5% solution and first silver medal on kaggle. `https://amaarora.github.io/2020/08/23/siimisic.html`. Accessed: 2022-02-02.

[10] Aldo Badano, Craig Revie, Andrew Casertano, Wei-Chung Cheng, Phil Green, Tom Kimpe, Elizabeth Krupinski, Christye Sisson, Stein Skrøvseth, Darren Treanor, et al. Consistency and standardization of color in medical imaging: a consensus report. *Journal of digital imaging*, 28(1):41–52, 2015.

[11] Pragati Baheti, Mukul Sikka, KV Arya, and R Rajesh. Federated learning on distributed medical records for detection of lung nodules. In *VISIGRAPP (4: VISAPP)*, pages 445–451, 2020.

[12] Peter Bandi, Oscar Geessink, Quirine Manson, Marcory Van Dijk, Maschenka Balkenhol, Meyke Hermsen, Babak Ehteshami Bejnordi, Byungjae Lee, Kyunghyun Paeng, Aoxiao Zhong, et al. From detection of individual metastases to classification of lymph node status at the patient level: the camelyon17 challenge. *IEEE transactions on medical imaging*, 38(2):550–560, 2018.

[13] Babak Ehteshami Bejnordi, Mitko Veta, Paul Johannes Van Diest, Bram Van Ginneken, Nico Karssemeijer, Geert Litjens, Jeroen AWM Van Der Laak, Meyke Hermsen, Quirine F Manson, Maschenka Balkenhol, et al. Diagnostic assessment of deep learning algorithms for detection of lymph node metastases in women with breast cancer. *Jama*, 318(22):2199–2210, 2017.

[14] Abhishek Bhowmick, John Duchi, Julien Freudiger, Gaurav Kapoor, and Ryan Rogers. Protection against reconstruction and its applications in private federated learning. *arXiv preprint arXiv:1812.00984*, 2018.

[15] Keith Bonawitz, Hubert Eichner, Wolfgang Grieskamp, Dzmitry Huba, Alex Ingerman, Vladimir Ivanov, Chloe Kiddon, Jakub Konečnỳ, Stefano Mazzocchi, Brendan McMahan, et al. Towards federated learning at scale: System design. *Proceedings of Machine Learning and Systems*, 1:374–388, 2019.

[16] Keith Bonawitz, Vladimir Ivanov, Ben Kreuter, Antonio Marcedone, H Brendan McMahan, Sarvar Patel, Daniel Ramage, Aaron Segal, and Karn Seth. Practical secure aggregation for privacy-preserving machine learning. In *proceedings of the 2017 ACM SIGSAC Conference on Computer and Communications Security*, pages 1175–1191, 2017.

[17] James Bradbury, Roy Frostig, Peter Hawkins, Matthew James Johnson, Chris Leary, Dougal Maclaurin, George Necula, Adam Paszke, Jake VanderPlas, Skye Wanderman-Milne, and Qiao Zhang. JAX: composable transformations of Python+NumPy programs, 2018.

[18] Talha Burki. Pharma blockchains AI for drug development. *The Lancet*, 393(10189):2382, 2019.

[19] Sebastian Caldas, Sai Meher Karthik Duddu, Peter Wu, Tian Li, Jakub Konečnỳ, H Brendan McMahan, Virginia Smith, and Ameet Talwalkar. Leaf: A benchmark for federated settings. *arXiv preprint arXiv:1812.01097*, 2018.

[20] Rich Caruana, Thorsten Joachims, and Lars Backstrom. KDD-Cup 2004: results and analysis. *ACM SIGKDD Explorations Newsletter*, 6(2):95–108, 2004.

[21] Arunava Chakravarty, Avik Kar, Ramanathan Sethuraman, and Debdoot Sheet. Federated learning for site aware chest radiograph screening. In *2021 IEEE 18th International Symposium on Biomedical Imaging (ISBI)*, pages 1077–1081. IEEE, 2021.

[22] Ken Chang, Niranjan Balachandar, Carson Lam, Darvin Yi, James Brown, Andrew Beers, Bruce Rosen, Daniel L Rubin, and Jayashree Kalpathy-Cramer. Distributed deep learning networks among institutions for medical imaging. *Journal of the American Medical Informatics Association*, 25(8):945–954, 2018.

[23] Zachary Charles, Zachary Garrett, Zhouyuan Huo, Sergei Shmulyian, and Virginia Smith. On large-cohort training for federated learning. *arXiv preprint arXiv:2106.07820*, 2021.

[24] Aakanksha Chowdhery, Sharan Narang, Jacob Devlin, Maarten Bosma, Gaurav Mishra, Adam Roberts, Paul Barham, Hyung Won Chung, Charles Sutton, Sebastian Gehrmann, et al. Palm: Scaling language modeling with pathways. *arXiv preprint arXiv:2204.02311*, 2022.

[25] Özgün Çiçek, Ahmed Abdulkadir, Soeren S Lienkamp, Thomas Brox, and Olaf Ronneberger. 3d u-net: learning dense volumetric segmentation from sparse annotation. In *International conference on medical image computing and computer-assisted intervention*, pages 424–432. Springer, 2016.

[26] Kenneth Clark, Bruce Vendt, Kirk Smith, John Freymann, Justin Kirby, Paul Koppel, Stephen Moore, Stanley Phillips, David Maffitt, Michael Pringle, et al. The cancer imaging archive (TCIA): maintaining and operating a public information repository. *Journal of digital imaging*, 26(6):1045–1057, 2013.

[27] Noel CF Codella, David Gutman, M Emre Celebi, Brian Helba, Michael A Marchetti, Stephen W Dusza, Aadi Kalloo, Konstantinos Liopyris, Nabin Mishra, Harald Kittler, et al. Skin lesion analysis toward melanoma detection: A challenge at the 2017 international symposium on biomedical imaging (ISBI), hosted by the international skin imaging collaboration (ISIC). In *2018 IEEE 15th international symposium on biomedical imaging (ISBI 2018)*, pages 168–172. IEEE, 2018.

[28] Gregory Cohen, Saeed Afshar, Jonathan Tapson, and Andre Van Schaik. EMNIST: Extending MNIST to handwritten letters. In *2017 International Joint Conference on Neural Networks (IJCNN)*, pages 2921–2926. IEEE, 2017.

[29] Marc Combalia, Noel CF Codella, Veronica Rotemberg, Brian Helba, Veronica Vilaplana, Ofer Reiter, Cristina Carrera, Alicia Barreiro, Allan C Halpern, Susana Puig, et al. Bcn20000: Dermoscopic lesions in the wild. *arXiv preprint arXiv:1908.02288*, 2019.

[30] Marius Cordts, Mohamed Omran, Sebastian Ramos, Timo Rehfeld, Markus Enzweiler, Rodrigo Benenson, Uwe Franke, Stefan Roth, and Bernt Schiele. The cityscapes dataset for semantic urban scene understanding. In *Proc. of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016.

[31] Luca Corinzia, Ami Beuret, and Joachim M Buhmann. Variational federated multi-task learning. *arXiv preprint arXiv:1906.06268*, 2019.

[32] Pierre Courtiol, Eric W Tramel, Marc Sanselme, and Gilles Wainrib. Classification and disease localization in histopathology using only global labels: A weakly-supervised approach. *arXiv preprint arXiv:1802.02212*, 2018.

[33] David R Cox. Regression models and life-tables. *Journal of the Royal Statistical Society: Series B (Methodological)*, 34(2):187–202, 1972.

[34] Ittai Dayan, Holger R Roth, Aoxiao Zhong, Ahmed Harouni, Amilcare Gentili, Anas Z Abidin, Andrew Liu, Anthony Beardsworth Costa, Bradford J Wood, Chien-Sung Tsai, et al. Federated learning for predicting clinical outcomes in patients with covid-19. *Nature medicine*, 27(10):1735–1743, 2021.

[35] Kevin de Haan, Yijie Zhang, Jonathan E Zuckerman, Tairan Liu, Anthony E Sisk, Miguel FP Diaz, Kuang-Yu Jen, Alexander Nobori, Sofia Liou, Sarah Zhang, et al. Deep learning-based transformation of H&E stained tissues into special stains. *Nature communications*, 12(1):1–13, 2021.

[36] Olivier Dehaene, Axel Camara, Olivier Moindrot, Axel de Lavergne, and Pierre Courtiol. Self-supervision closes the gap between weak and strong supervision in histology. *arXiv preprint arXiv:2012.03583*, 2020.

[37] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, pages 248–255. Ieee, 2009.

[38] Brain development team. Ixi dataset. `https://brain-development.org/ixi-dataset/`. Accessed: 2022-02-02.

[39] Lee R Dice. Measures of the amount of ecologic association between species. *Ecology*, 26(3):297–302, 1945.

[40] Jean Ogier du Terrail, Armand Leopold, Clément Joly, Constance Beguier, Mathieu Andreux, Charles Maussion, Benoit Schmauch, Eric W Tramel, Etienne Bendjebbar, Mikhail Zaslavskiy, et al. Collaborative federated learning behind hospitals' firewalls for predicting histological response to neoadjuvant chemotherapy in triple-negative breast cancer. *medRxiv*, 2021.

[41] Marco F Duarte and Yu Hen Hu. Vehicle classification in distributed sensor networks. *Journal of Parallel and Distributed Computing*, 64(7):826–838, 2004.

[42] Cynthia Dwork, Aaron Roth, et al. The algorithmic foundations of differential privacy. *Found. Trends Theor. Comput. Sci.*, 9(3-4):211–407, 2014.

[43] Alireza Fallah, Aryan Mokhtari, and Asuman Ozdaglar. Personalized federated learning: A meta-learning approach. *arXiv preprint arXiv:2002.07948*, 2020.

[44] Yann Fraboni, Richard Vidal, Laetitia Kameni, and Marco Lorenzi. On the impact of client sampling on federated learning convergence. *arXiv preprint arXiv:2107.12211*, 2021.

[45] Yu Fu, Alexander W Jung, Ramon Viñas Torne, Santiago Gonzalez, Harald Vöhringer, Artem Shmatko, Lucy R Yates, Mercedes Jimenez-Linan, Luiza Moore, and Moritz Gerstung. Pan-cancer computational histopathology reveals mutations, tumor composition and prognosis. *Nature Cancer*, 1(8):800–810, 2020.

[46] Mathieu N Galtier and Camille Marini. Substra: a framework for privacy-preserving, traceable and collaborative machine learning. *arXiv preprint arXiv:1910.11567*, 2019.

[47] Alec Go, Richa Bhayani, and Lei Huang. Twitter sentiment classification using distant supervision. *CS224N project report, Stanford*, 1(12):2009, 2009.

[48] Xuan Gong, Abhishek Sharma, Srikrishna Karanam, Ziyan Wu, Terrence Chen, David Doermann, and Arun Innanje. Ensemble attention distillation for privacy-preserving federated learning. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 15076–15086, 2021.

[49] Ben Graham. Kaggle diabetic retinopathy detection competition report. *University of Warwick*, pages 24–26, 2015.

[50] Gozde N Gunesli, Mohsin Bilal, Shan E Ahmed Raza, and Nasir M Rajpoot. Feddropoutavg: Generalizable federated learning for histopathology image classification. *arXiv preprint arXiv:2111.13230*, 2021.

[51] Farzin Haddadpour, Mohammad Mahdi Kamani, Aryan Mokhtari, and Mehrdad Mahdavi. Federated learning with compression: Unified analysis and sharp guarantees. In *International Conference on Artificial Intelligence and Statistics*, pages 2350–2358. PMLR, 2021.

[52] Peter F Hahn, Michael A Blake, and Giles WL Boland. Adrenal lesions: attenuation measurement differences between ct scanners. *Radiology*, 240(2):458–463, 2006.

[53] Chaoyang He, Murali Annavaram, and Salman Avestimehr. Group knowledge transfer: Federated learning of large CNNs at the edge. *arXiv preprint arXiv:2007.14513*, 2020.

[54] Chaoyang He, Keshav Balasubramanian, Emir Ceyani, Carl Yang, Han Xie, Lichao Sun, Lifang He, Liangwei Yang, Philip S Yu, Yu Rong, et al. Fedgraphnn: A federated learning system and benchmark for graph neural networks. *arXiv preprint arXiv:2104.07145*, 2021.

[55] Chaoyang He, Songze Li, Jinhyun So, Xiao Zeng, Mi Zhang, Hongyi Wang, Xiaoyang Wang, Praneeth Vepakomma, Abhishek Singh, Hang Qiu, et al. Fedml: A research library and benchmark for federated machine learning. *arXiv preprint arXiv:2007.13518*, 2020.

[56] Chaoyang He, Alay Dilipbhai Shah, Zhenheng Tang, Di Fan1Adarshan Naiynar Sivashunmugam, Keerti Bhogaraju, Mita Shimpi, Li Shen, Xiaowen Chu, Mahdi Soltanolkotabi, and Salman Avestimehr. Fedcv: A federated learning framework for diverse computer vision tasks. *arXiv preprint arXiv:2111.11066*, 2021.

[57] Nicholas Heller, Fabian Isensee, Klaus H Maier-Hein, Xiaoshuai Hou, Chunmei Xie, Fengyi Li, Yang Nan, Guangrui Mu, Zhiyong Lin, Miofei Han, et al. The state of the art in kidney and kidney tumor segmentation in contrast-enhanced ct imaging: Results of the kits19 challenge. *Medical Image Analysis*, page 101821, 2020.

[58] Nicholas Heller, Niranjan Sathianathen, Arveen Kalapara, Edward Walczak, Keenan Moore, Heather Kaluzniak, Joel Rosenberg, Paul Blake, Zachary Rengel, Makinna Oestreich, et al. The kits19 challenge data: 300 kidney tumor cases with clinical context, ct semantic segmentations, and surgical outcomes. *arXiv preprint arXiv:1904.00445*, 2019.

[59] Frederick M Howard, James Dolezal, Sara Kochanny, Jefree Schulte, Heather Chen, Lara Heij, Dezheng Huo, Rita Nanda, Olufunmilayo I Olopade, Jakob N Kather, et al. The impact of site-specific digital histology signatures on deep learning model accuracy and bias. *Nature communications*, 12(1):1–13, 2021.

[60] Kevin Hsieh, Amar Phanishayee, Onur Mutlu, and Phillip Gibbons. The non-IID data quagmire of decentralized machine learning. In *International Conference on Machine Learning (ICML)*, pages 5819–5830. PMLR, 2020.

[61] 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.

[62] Tzu-Ming Harry Hsu, Hang Qi, and Matthew Brown. Federated visual classification with real-world data distribution. In *European Conference on Computer Vision*, pages 76–92. Springer, 2020.

[63] Juan Eugenio Iglesias, Cheng-Yi Liu, Paul M Thompson, and Zhuowen Tu. Robust brain extraction across datasets and comparison with publicly available methods. *IEEE transactions on medical imaging*, 30(9):1617–1634, 2011.

[64] S. G. Armato III, G. McLennan, L. Bidaut, M. F. McNitt-Gray, C. R. Meyer, A. P. Reeves, B. Zhao, D. R. Aberle, C. I. Henschke, E. A. Hoffman, E. A. Kazerooni, H. MacMahon, E. J. R. Van Beek, D. Yankelevitz, A. M. Biancardi, P. H. Bland, M. S. Brown, R. M. Engelmann, G. E. Laderach, D. Max, R. C. Pais, D. P. Y. Qing, R. Y. Roberts, A. R. Smith, A. Starkey, P. Batra, P. Caligiuri, A. Farooqi, G. W. Gladish, C. M. Jude, R. F. Munden, I. Petkovska, L. E. Quint, L. H. Schwartz, B. Sundaram, L. E. Dodd, C. Fenimore, D. Gur, N. Petrick, J. Freymann, J. Kirby, B. Hughes, A. V. Casteele, S. Gupte ans M. Sallam, M. D. Heath, M. H. Kuhn, E. Dharaiya, R. Burns, D. S. Fryd, M. Salganicoff, V. Anand, U. Shreter, S. Vastagh, B. Y. Croft, and L. P. Clarke. Data from lidc-idri [data set]. the cancer imaging archive., 2015.

[65] Maximilian Ilse, Jakub Tomczak, and Max Welling. Attention-based deep multiple instance learning. In *International conference on machine learning*, pages 2127–2136. PMLR, 2018.

[66] Maximilian Ilse, Jakub M. Tomczak, and Max Welling. Attention-based deep multiple instance learning. `https://github.com/AMLab-Amsterdam/AttentionDeepMIL`. Accessed: 2022-02-02.

[67] Jeremy Irvin, Pranav Rajpurkar, Michael Ko, Yifan Yu, Silviana Ciurea-Ilcus, Chris Chute, Henrik Marklund, Behzad Haghgoo, Robyn Ball, Katie Shpanskaya, et al. Chexpert: A large chest radiograph dataset with uncertainty labels and expert comparison. In *Proceedings of the AAAI conference on artificial intelligence*, volume 33, pages 590–597, 2019.

[68] F Isensee, P Jäger, J Wasserthal, D Zimmerer, J Petersen, S Kohl, J Schock, A Klein, T RoSS, S Wirkert, et al. batchgenerators—a python framework for data augmentation. *Zenodo https://doi. org/10.5281/zenodo*, 3632567, 2020.

[69] Fabian Isensee, Paul F Jaeger, Simon AA Kohl, Jens Petersen, and Klaus H Maier-Hein. nnu-net: a self-configuring method for deep learning-based biomedical image segmentation. *Nature methods*, 18(2):203–211, 2021.

[70] Andras Janosi, William Steinbrunn, Matthias Pfisterer, and Robert Detrano. Heart disease data set, 1988.

[71] Andrew Janowczyk, Ren Zuo, Hannah Gilmore, Michael Feldman, and Anant Madabhushi. Histoqc: an open-source quality control tool for digital pathology slides. *JCO clinical cancer informatics*, 3:1–7, 2019.

[72] Stephen P Jenkins. Survival analysis. *Unpublished manuscript, Institute for Social and Economic Research, University of Essex, Colchester, UK*, 42:54–56, 2005.

[73] Peter Kairouz, H. Brendan McMahan, Brendan Avent, Aurélien Bellet, Mehdi Bennis, Arjun Nitin Bhagoji, Kallista Bonawitz, Zachary Charles, Graham Cormode, Rachel Cummings, Rafael G. L. D'Oliveira, Hubert Eichner, Salim El Rouayheb, David Evans, Josh Gardner, Zachary Garrett, Adrià Gascón, Badih Ghazi, Phillip B. Gibbons, Marco Gruteser, Zaid Harchaoui, Chaoyang He, Lie He, Zhouyuan Huo, Ben Hutchinson, Justin Hsu, Martin Jaggi, Tara Javidi, Gauri Joshi, Mikhail Khodak, Jakub Konecný, Aleksandra Korolova, Farinaz Koushanfar, Sanmi Koyejo, Tancrède Lepoint, Yang Liu, Prateek Mittal, Mehryar Mohri, Richard Nock, Ayfer Özgür, Rasmus Pagh, Hang Qi, Daniel Ramage, Ramesh Raskar, Mariana Raykova, Dawn Song, Weikang Song, Sebastian U. Stich, Ziteng Sun, Ananda Theertha Suresh, Florian Tramèr, Praneeth Vepakomma, Jianyu Wang, Li Xiong, Zheng Xu, Qiang Yang, Felix X. Yu, Han Yu, and Sen Zhao. Advances and open problems in federated learning. *Foundations and Trends® in Machine Learning*, 14(1–2):1–210, 2021.

[74] Georgios Kaissis, Alexander Ziller, Jonathan Passerat-Palmbach, Théo Ryffel, Dmitrii Usynin, Andrew Trask, Ionésio Lima, Jason Mancuso, Friederike Jungmann, Marc-Matthias Steinborn, et al. End-to-end privacy preserving deep learning on multi-institutional medical imaging. *Nature Machine Intelligence*, 3(6):473–484, 2021.

[75] Edward L Kaplan and Paul Meier. Nonparametric estimation from incomplete observations. *Journal of the American statistical association*, 53(282):457–481, 1958.

[76] Jared Kaplan, Sam McCandlish, Tom Henighan, Tom B Brown, Benjamin Chess, Rewon Child, Scott Gray, Alec Radford, Jeffrey Wu, and Dario Amodei. Scaling laws for neural language models. *arXiv preprint arXiv:2001.08361*, 2020.

[77] 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.

[78] Daniel S Kermany, Michael Goldbaum, Wenjia Cai, Carolina CS Valentim, Huiying Liang, Sally L Baxter, Alex McKeown, Ge Yang, Xiaokang Wu, Fangbing Yan, et al. Identifying medical diagnoses and treatable diseases by image-based deep learning. *Cell*, 172(5):1122–1131, 2018.

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

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

[81] Amal Lahiani, Irina Klaman, Nassir Navab, Shadi Albarqouni, and Eldad Klaiman. Seamless virtual whole slide image synthesis and validation using perceptual embedding consistency. *IEEE Journal of Biomedical and Health Informatics*, 25(2):403–411, 2020.

[82] Fan Lai, Yinwei Dai, Sanjay Singapuram, Jiachen Liu, Xiangfeng Zhu, Harsha Madhyastha, and Mosharaf Chowdhury. Fedscale: Benchmarking model and system performance of federated learning at scale. In *International Conference on Machine Learning*, pages 11814–11827. PMLR, 2022.

[83] Yann LeCun and Corinna Cortes. MNIST handwritten digit database. 2010.

[84] Tian Li, Anit Kumar Sahu, Ameet Talwalkar, and Virginia Smith. Federated learning: Challenges, methods, and future directions. *IEEE Signal Processing Magazine*, 37(3):50–60, 2020.

[85] 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.

[86] Wei Li and Andrew McCallum. Pachinko allocation: DAG-structured mixture models of topic correlations. In *Proceedings of the 23rd international conference on Machine learning*, pages 577–584, 2006.

[87] Wei Yang Bryan Lim, Nguyen Cong Luong, Dinh Thai Hoang, Yutao Jiao, Ying-Chang Liang, Qiang Yang, Dusit Niyato, and Chunyan Miao. Federated learning in mobile edge networks: A comprehensive survey. *IEEE Communications Surveys & Tutorials*, 22(3):2031–2063, 2020.

[88] Tsung-Yi Lin, Priya Goyal, Ross B. Girshick, Kaiming He, and Piotr Dollár. Focal loss for dense object detection. *CoRR*, abs/1708.02002, 2017.

[89] Geert Litjens, Peter Bandi, Babak Ehteshami Bejnordi, Oscar Geessink, Maschenka Balkenhol, Peter Bult, Altuna Halilovic, Meyke Hermsen, Rob van de Loo, Rob Vogels, et al. 1399 H&E-stained sentinel lymph node sections of breast cancer patients: the CAMELYON dataset. *GigaScience*, 7(6):giy065, 2018.

[90] Ziwei Liu, Ping Luo, Xiaogang Wang, and Xiaoou Tang. Deep learning face attributes in the wild. In *Proceedings of the IEEE international conference on computer vision*, pages 3730–3738, 2015.

[91] Ming Y Lu, Richard J Chen, Dehan Kong, Jana Lipkova, Rajendra Singh, Drew FK Williamson, Tiffany Y Chen, and Faisal Mahmood. Federated learning for computational pathology on gigapixel whole slide images. *Medical image analysis*, 76:102298, 2022.

[92] Ming Y Lu, Drew FK Williamson, Tiffany Y Chen, Richard J Chen, Matteo Barbieri, and Faisal Mahmood. Data-efficient and weakly supervised computational pathology on whole-slide images. *Nature biomedical engineering*, 5(6):555–570, 2021.

[93] Yunlong Lu, Xiaohong Huang, Ke Zhang, Sabita Maharjan, and Yan Zhang. Low-latency federated learning and blockchain for edge association in digital twin empowered 6g networks. *IEEE Transactions on Industrial Informatics*, 17(7):5098–5107, 2020.

[94] Jiahuan Luo, Xueyang Wu, Yun Luo, Anbu Huang, Yunfeng Huang, Yang Liu, and Qiang Yang. Real-world image datasets for federated learning. *arXiv preprint arXiv:1910.11089*, 2019.

[95] Zhengquan Luo, Yunlong Wang, Zilei Wang, Zhenan Sun, and Tieniu Tan. Fediris: Towards more accurate and privacy-preserving iris recognition via federated template communication. *CVPRW*, 2022.

[96] Agnes Lydia and Sagayaraj Francis. Adagrad—an optimizer for stochastic gradient descent. *Int. J. Inf. Comput. Sci*, 6(5):566–568, 2019.

[97] Othmane Marfoq, Chuan Xu, Giovanni Neglia, and Richard Vidal. Throughput-Optimal Topology Design for Cross-Silo Federated Learning. In *34th Conference on Neural Information Processing Systems (NeurIPS 2020)*, Vancouver, Canada, December 2020. NeurIPS 2020.

[98] Leland McInnes, John Healy, and James Melville. Umap: Uniform manifold approximation and projection for dimension reduction. *arXiv preprint arXiv:1802.03426*, 2018.

[99] 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.

[100] Fausto Milletari, Nassir Navab, and Seyed-Ahmad Ahmadi. V-net: Fully convolutional neural networks for volumetric medical image segmentation. In *2016 fourth international conference on 3D vision (3DV)*, pages 565–571. IEEE, 2016.

[101] TCGA Research Network. Tensorflow federated stack overflow dataset. `https://www.cancer.gov/tcga`. Accessed: 2022-05-18.

[102] Adaloglou Nikolaos. Deep learning in medical image analysis: a comparative analysis of multi-modal brain-mri segmentation with 3d deep neural networks. Master's thesis, University of Patras, 2019. `https://github.com/black0017/MedicalZooPytorch`.

[103] Chaoyue Niu, Fan Wu, Shaojie Tang, Lifeng Hua, Rongfei Jia, Chengfei Lv, Zhihua Wu, and Guihai Chen. Billion-scale federated learning on mobile clients: a submodel design with tunable privacy. In *Proceedings of the 26th Annual International Conference on Mobile Computing and Networking*, pages 1–14, 2020.

[104] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, et al. Pytorch: An imperative style, high-performance deep learning library. *Advances in neural information processing systems*, 32, 2019.

[105] Sarthak Pati, Ujjwal Baid, Maximilian Zenk, Brandon Edwards, Micah Sheller, G Anthony Reina, Patrick Foley, Alexey Gruzdev, Jason Martin, Shadi Albarqouni, et al. The federated tumor segmentation (FeTS) challenge. *arXiv preprint arXiv:2105.05874*, 2021.

[106] Fernando Pérez-García, Rachel Sparks, and Sebastien Ourselin. Torchio: a python library for efficient loading, preprocessing, augmentation and patch-based sampling of medical images in deep learning. *Computer Methods and Programs in Biomedicine*, 208:106236, 2021.

[107] Constantin Philippenko and Aymeric Dieuleveut. Bidirectional compression in heterogeneous settings for distributed or federated learning with partial participation: tight convergence guarantees. *arXiv preprint arXiv:2006.14591*, 2020.

[108] P Jonathon Phillips, Kevin W Bowyer, Patrick J Flynn, Xiaomei Liu, and W Todd Scruggs. The iris challenge evaluation 2005. In *2008 IEEE Second International Conference on Biometrics: Theory, Applications and Systems*, pages 1–8. IEEE, 2008.

[109] Aditya Ramesh, Prafulla Dhariwal, Alex Nichol, Casey Chu, and Mark Chen. Hierarchical text-conditional image generation with clip latents, 2022.

[110] Sashank Reddi, Zachary Charles, Manzil Zaheer, Zachary Garrett, Keith Rush, Jakub Konečnỳ, Sanjiv Kumar, and H Brendan McMahan. Adaptive federated optimization. *arXiv preprint arXiv:2003.00295*, 2020.

[111] Meta AI Research. Federated learning simulator (flsim). `https://github.com/facebookresearch/FLSim/tree/main/examples`, 2012.

[112] 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.

[113] Felix Sattler, Simon Wiedemann, Klaus-Robert Müller, and Wojciech Samek. Sparse binary compression: Towards distributed deep learning with minimal communication. In *2019 International Joint Conference on Neural Networks (IJCNN)*, pages 1–8. IEEE, 2019.

[114] Arnaud Arindra Adiyoso Setio, Alberto Traverso, Thomas De Bel, Moira SN Berens, Cas Van Den Bogaard, Piergiorgio Cerello, Hao Chen, Qi Dou, Maria Evelina Fantacci, Bram Geurts, et al. Validation, comparison, and combination of algorithms for automatic detection of pulmonary nodules in computed tomography images: the LUNA16 challenge. *Medical image analysis*, 42:1–13, 2017.

[115] 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.

[116] Micah J Sheller, G Anthony Reina, Brandon Edwards, Jason Martin, and Spyridon Bakas. Multi-institutional deep learning modeling without sharing patient data: A feasibility study on brain tumor segmentation. In *International MICCAI Brainlesion Workshop*, pages 92–104. Springer, 2018.

[117] Santiago Silva, Andre Altmann, Boris Gutman, and Marco Lorenzi. Fed-BioMed: A general open-source frontend framework for federated learning in healthcare. In *Domain Adaptation and Representation Transfer, and Distributed and Collaborative Learning*, pages 201–210. Springer, 2020.

[118] Chen Sun, Abhinav Shrivastava, Saurabh Singh, and Abhinav Gupta. Revisiting unreasonable effectiveness of data in deep learning era. In *Proceedings of the IEEE international conference on computer vision*, pages 843–852, 2017.

[119] Mingxing Tan and Quoc V. Le. Efficientnet: Rethinking model scaling for convolutional neural networks. *CoRR*, abs/1905.11946, 2019.

[120] Tensorflow. Tensorflow federated stack overflow dataset. `https://www.tensorflow.org/federated/api_docs/python/tff/simulation/datasets/stackoverflow/load_data`, 2019.

[121] Katarzyna Tomczak, Patrycja Czerwińska, and Maciej Wiznerowicz. The cancer genome atlas (TCGA): an immeasurable source of knowledge. *Contemporary oncology*, 19(1A):A68, 2015.

[122] Antonio Torralba and Alexei A Efros. Unbiased look at dataset bias. In *CVPR 2011*, pages 1521–1528. IEEE, 2011.

[123] 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.

[124] Laurens Van der Maaten and Geoffrey Hinton. Visualizing data using t-SNE. *Journal of machine learning research*, 9(11), 2008.

[125] G. Van Horn, O. Mac Aodha, Y. Song, Y. Cui, C. Sun, A. Shepard, H. Adam, P. Perona, and S. Belongie. The inaturalist species classification and detection dataset. In *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8769–8778, 2018.

[126] Mitko Veta, Josien PW Pluim, Paul J Van Diest, and Max A Viergever. Breast cancer histopathology image analysis: A review. *IEEE transactions on biomedical engineering*, 61(5):1400–1411, 2014.

[127] Michael Völske, Martin Potthast, Shahbaz Syed, and Benno Stein. TL;DR: Mining Reddit to learn automatic summarization. In *Proceedings of the Workshop on New Frontiers in Summarization*, pages 59–63, Copenhagen, Denmark, September 2017. Association for Computational Linguistics.

[128] Zhuoshi Wei, Tieniu Tan, and Zhenan Sun. Nonlinear iris deformation correction based on gaussian model. In *International Conference on Biometrics*, pages 780–789. Springer, 2007.

[129] Qi Xia, Winson Ye, Zeyi Tao, Jindi Wu, and Qun Li. A survey of federated learning for edge computing: Research problems and solutions. *High-Confidence Computing*, page 100008, 2021.

[130] Guoyang Xie, Jinbao Wang, Yawen Huang, Yefeng Zheng, Feng Zheng, Jingkuang Song, and Yaochu Jin. FedMed-GAN: Federated multi-modal unsupervised brain image synthesis. *arXiv preprint arXiv:2201.08953*, 2022.

[131] Haibo Yang, Minghong Fang, and Jia Liu. Achieving linear speedup with partial worker participation in non-iid federated learning. *arXiv preprint arXiv:2101.11203*, 2021.

[132] Bill Yuchen Lin, Chaoyang He, Zihang Zeng, Hulin Wang, Yufen Huang, Christophe Dupuy, Rahul Gupta, Mahdi Soltanolkotabi, Xiang Ren, and Salman Avestimehr. Fednlp: Benchmarking federated learning methods for natural language processing tasks. *Findings of NAACL*, 2022.

[133] Mikhail Yurochkin, Mayank Agarwal, Soumya Ghosh, Kristjan Greenewald, Nghia Hoang, and Yasaman Khazaeni. Bayesian nonparametric federated learning of neural networks. In *International Conference on Machine Learning*, pages 7252–7261. PMLR, 2019.

[134] Manzil Zaheer, Sashank Reddi, Devendra Sachan, Satyen Kale, and Sanjiv Kumar. Adaptive methods for nonconvex optimization. *Advances in neural information processing systems*, 31, 2018.

[135] Hui Zhang, Zhenan Sun, and Tieniu Tan. Contact lens detection based on weighted lbp. In *2010 20th International Conference on Pattern Recognition*, pages 4279–4282. IEEE, 2010.

[136] Qi Zhang, Haiqing Li, Zhenan Sun, and Tieniu Tan. Deep feature fusion for iris and periocular biometrics on mobile devices. *IEEE Transactions on Information Forensics and Security*, 13(11):2897–2912, 2018.

Checklist

1. For all authors...
   (a) Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope? [Yes]
   (b) Did you describe the limitations of your work? [Yes] See Section 5.
   (c) Did you discuss any potential negative societal impacts of your work? [Yes] See Section A.
   (d) Have you read the ethics review guidelines and ensured that your paper conforms to them? [Yes] We point out that we not collect data ourselves. We did a thorough background check on each dataset regarding compliance with these guidelines. We refer to the detailed appendix of each dataset for specific details.
2. If you are including theoretical results...
   (a) Did you state the full set of assumptions of all theoretical results? [N/A] Our work does not contain theoretical results.
   (b) Did you include complete proofs of all theoretical results? [N/A] Our work does not contain theoretical results.
3. If you ran experiments (e.g. for benchmarks)...
   (a) Did you include the code, data, and instructions needed to reproduce the main experimental results (either in the supplemental material or as a URL)? [Yes] See Abstract.
   (b) Did you specify all the training details (e.g., data splits, hyperparameters, how they were chosen)? [Yes] See supplementary and code provided.
   (c) Did you report error bars (e.g., with respect to the random seed after running experiments multiple times)? [Yes] We reported error bars as we report the average error on the local test sets across multiple seeds. For the largest one, we did not use multiple seeds, but observed empirically a smaller variance in the results due to larger local test set sizes.
   (d) Did you include the total amount of compute and the type of resources used (e.g., type of GPUs, internal cluster, or cloud provider)? [Yes] See Appendix J.1
4. If you are using existing assets (e.g., code, data, models) or curating/releasing new assets...
   (a) If your work uses existing assets, did you cite the creators? [Yes] See Section 3.
   (b) Did you mention the license of the assets? [Yes] See code and supplementary.
   (c) Did you include any new assets either in the supplemental material or as a URL? [Yes] See link in abstract.
   (d) Did you discuss whether and how consent was obtained from people whose data you're using/curating? [Yes] We are only repurposing existing assets. We did a thorough background check on each dataset on this issue. We refer to the detailed appendix of each dataset for specific details.
   (e) Did you discuss whether the data you are using/curating contains personally identifiable information or offensive content? [Yes] We are only repurposing existing assets. We did a thorough background check on each dataset on this issue. We refer to the detailed appendix of each dataset for specific details.
5. If you used crowdsourcing or conducted research with human subjects...
   (a) Did you include the full text of instructions given to participants and screenshots, if applicable? [N/A] We are only repurposing existing assets.
   (b) Did you describe any potential participant risks, with links to Institutional Review Board (IRB) approvals, if applicable? [N/A] We are only repurposing existing assets.
   (c) Did you include the estimated hourly wage paid to participants and the total amount spent on participant compensation? [N/A] We are only repurposing existing assets.