Federated Learning Test for Collaborative Machine Learning

Bernard Chang[1,*], *Çinar* Turhan[1], *Maria* Esteva[2], *Maša* Prodanović[1], *Jeremy T.* First[3], *Yang* Ning[3], *Yuliana* Zapata[3], *Glen* Gettemy[3], *Caio* Graco Pereira Santos[4] and *Jonatas* Castro Einsielder[4]
[1]Hildebrand Department of Petroleum and Geosystems Engineering, The University of Texas at Austin, Austin, TX, USA
[2]Texas Advanced Computing Center, The University of Texas at Austin, Austin, TX, USA
[3]bp, Houston, TX
[4]Petrobras, Rio de Janeiro, Brazil

Abstract. Machine and deep learning (ML/DL) have emerged as powerful tools for driving innovation in industry and sciences. Typically, a data-driven model trains on a large, centralized dataset to achieve reasonable prediction accuracy while still generalizing well to unseen data. However, data sharing can be challenging when collaborating across multiple groups because of privacy concerns or sheer data size. Here, we test a distributed machine learning framework using MS-Net, a deep learning model to predict the velocity field and permeability given a pore scale image of a porous medium - a fundamental task in digital rock physics. In the framework, known as federated learning (FL), a central server distributes copies of the central model to several clients, each client trains a model on its own set of training data, and the central server subsequently aggregates the client-side model parameter updates. We propose a set of geometric characterizations to test the quality of the training data quality without requiring the sharing of sensitive data. We successfully obtained the approval from participating companies (bp and Petrobras), conducted tutorials on the computational tools in the workflow, and trained the model. We report on the training performance of this framework using an established network design to predict velocity fields of imaged rock samples.

1 Introduction

Machine and deep learning (ML/DL) are now omni-present tools for data-driven innovation in industry and sciences. Specifically in the field of digital rock physics, ML/DL models can accurately predict temporal and spatial phenomena in a variety of complex geophysical systems based on an image of the porous media (see [1, 2] for recent reviews). For example, the Prodanović research group successfully applied DL to predict 3D velocity fields in image data stored in the DRP using a convolutional neural network (CNN) approach named MS-Net [3, 4]. This work has been combined with diffusion for the prediction of concentration fields [5] and extended to electric current transport [6]. These approaches use segmented images as a starting point, and most notably, they constitute the only working DL prediction for fractured or otherwise heterogeneous media.

One common hurdle to realizing the potential of ML/DL in digital rocks physics is that data-driven models train on large, centralized datasets and the datasets should be reasonably diverse. Data in digital rock physics are typically large in scale and at high resolution; due to innovations in imaging techniques, datasets are often GB to TB in size. Generally, there is no shortage of datasets, and we curate many open datasets in the Digital Rocks Portal (DRP) [7]. However, when collaborating across multiple groups, data sharing can be challenging because of export control regulations, business sensitivities, privacy concerns, large data volumes, formatting standards, and hardware differences. Most data remain behind firewalls. Additionally, in the case of traditional curve fitting, a learned model can still be effectively communicated and used by others via the functional relationship and its parameters; however, the same is not necessarily true for ML models or neural network parameters. In such cases, models often need to be retrained with new and different datasets, which is time-consuming and some of the learning is ultimately lost.

In recent years, decentralized training of ML models has emerged as a resource-efficient and privacy-promoting alternative to centralized solutions. In particular, federated learning (FL) [8, 9] allows collaborative training of ML models without collecting massive amounts of potentially sensitive private data from the participating agents and, instead, exchanges the learned neural network parameters. Intellectual property concerns and export regulations may still hinder the free exchange of these network parameters; however, the extent to which these networks are subject to such policies ultimately reduce to contractual agreements between participating parties. While we later describe our encounters with some of these issues, the primary

* Corresponding author: bcchang@utexas.edu

motivation for this work focuses on the technical capabilities of FL. Those agents (*i.e.*, clients) may be devices (known as a cross-device scenario) or institutions (known as a cross-silo scenario).

An illustration of a typical FL system is shown in Figure 1. Training in FL systems is organized into rounds. In each round, the coordinator, which is a server, shares a copy of an ML model with a set of clients; the clients update the model on their local, potentially private and large datasets, and then report the updates to the server. The server then combines the individual model weights using a predetermined *aggregation strategy*, updates the weights of the central model, and redistributes copies to the clients for continued training. The end product after several rounds of training is typically the aggregated central model that has learned from the clients' local datasets without the explicit transfer of the training data. In certain FL topologies, such as personalized FL, the product could be the local models that are tuned to the clients' individual data distributions, but still contain general base knowledge from a global model.

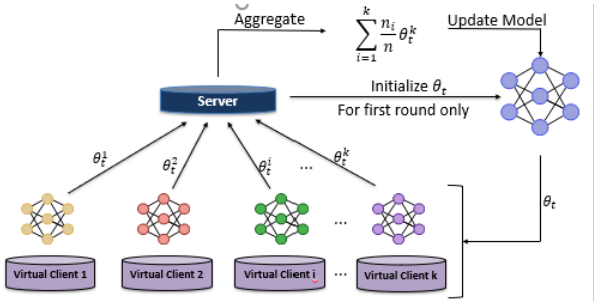

Fig. 1. A schematic of federated learning collaborative training, where data remains with clients while training the model, but the error is aggregated for simultaneous learning.

Table 1. The number of mentions of AI topics by search terms in major journals as of March 1, 2024. DL refers to deep learning, ML to machine learning, NN to neural network, and FL to federated learning. For FL, the number in parentheses indicates the number of actual implementations.

Journal collection	DL	ML	NN	FL
AGU Journals	10985	4056	4057	2 (0)
Geoscience World Journals	5902	2973	3241	2 (1)
Physical Review Journals	4171	7423	9537	6 (2)
OnePetro Journals	2426	3393	1172	0

FL remains largely unknown in geoscience applications. While many AI topics are mentioned in the literature, only three FL implementations (for rainfall prediction or seismic networks) currently appear in the major journals (Table 1). None are applied to digital rocks physics. Here, we revisit the MS-Net framework [3] for predicting the permeabilities of imaged porous medium samples under the context of FL with the motivation to generalize the model beyond the open data in the DRP.

Establishing that datasets are AI ready is fundamental for any ML application [10], [11], and this is especially important when the data itself cannot be disclosed due to proprietary or access restrictions. AI ready data is complete, properly documented, unbiased, and has been assessed on features relevant to the scientific domain. We propose a set of benchmark measures based on Minkowski functionals [12], [13], a scale-independent heterogeneity classifier [14], and morphology drainage, as suggested by [15]. This basic information about the data was shared with the server (as well as presented in this paper) and provided an understanding of the quality of data used for training without sharing the datasets themselves. All clients used the same Python code for the benchmark measures and the same, open source lattice-Boltzmann method solver, LBPM [15], to produce training velocity datasets with mutually agreed-upon input settings and boundary conditions.

The proposed work accomplishes two goals. First, it shows how to foster collaboration among separate companies, without sharing sensitive data, which helps to accelerate technical advancements beyond this particular application in digital rock physics. The framework can be adopted for training on any spatial data from pore scale (this work) to field scale. Second, in the context of digital rocks physics, this work allows for improved real-time assessment of flow and transport properties. When applied to large libraries of imaged rock samples, this work can accelerate uncertainty analysis workflows in an otherwise computationally expensive domain or allow the closer coupling of dynamic processes. Beyond bridging data privacy issues, the implications of this workflow ameliorate the difficulties of handling large data locality issues for any ML/DL training application.

2 Methods

2.1 Input data preparation

Each client started from a set of 3D segmented images of porous media with two numerical labels identifying the pore-space and solid-phase. Two example cross-sections are shown in Figure 2. While segmentation itself is an important topic (for review, see [16]), the segmentation workflows are beyond the scope of this work, and each client was free to implement the segmentation independently. Each phase contains many separate objects that can be measured or characterized. Due to the complexity of porous media, there is little agreement on standard measures to characterize the porous media geometry and topology. Here, we chose Minkowski functionals as emergent key descriptors supported by theory [13]. We further use a heterogeneity classification curve recently introduced by our group and evaluated on DRP data [14].

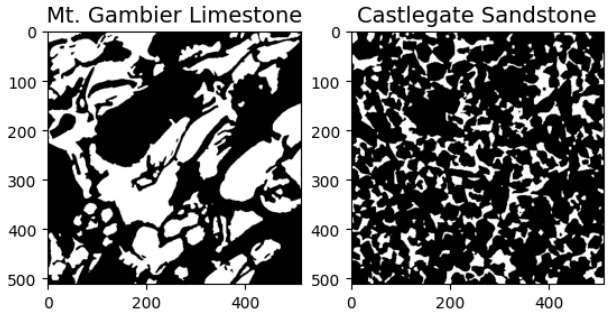

Fig. 2. Cross-sections of two example training images, Gambier limestone and Castlegate sandstone from the Digital Rocks Portal [17]. The solid-phase is shown in black, and the pore-space in white.

Each client prepared at least 100 subdirectories, each storing an imaged dataset of the same nominal size (256^3 spatial grid of points), following an agreed upon nomenclature: <1-digit client #>_<3-digit dataset #>_<3-digit side length> (e.g., 0_003_256). Each dataset was stored using the same file format (volumetric tiff file), with a one-byte numerical value per spatial point, where 0 denotes solid phase and 1 denotes pore space. To eliminate images with few meaningful training points, images were required to contain a minimum of 10% pore space values across the volume. In the same folder, metadata containing the voxel length (in microns) of the image were saved as a text file.

Note that the data size (256^3 numerical cells) in this exercise is not meant to be a representative elementary volume. Rather, we chose a standard smaller image size in order to focus on the performance of the FL framework itself. We also note that, in MS-Net, we trained to predict the velocity field of an image before assessing the effective permeability. Various studies show that computed velocity fields, porosities, and permeability values can change when a digital rock sample has limited field of view or when the segmentation method does not fully capture the under-resolved regions of the image [21, 22]. For example, a conservative segmentation threshold for a coarse resolution image could lead to an artificial increase in pore throat size, resulting in an overestimation of permeability. Additionally, under-resolved images may not represent details of the pore structure well, causing underrepresentation of available flow channels. The distributed nature of data collection in this experiment means that resolution, segmentation procedure, and field of view are not uniform, though this challenge is not unique to FL. In principle, any well-resolved image can participate. We later describe an assessment criterion for image resolution and segmentation using morphological drainage simulations. There were no assumptions made on the representativeness beyond a comparison with Darcy's law to calculate effective permeability.

Clients agreed to use LBPM [15] to compute the velocity fields using exactly the same input specifications and boundary conditions (see Appendix). We chose LBPM as it is open-source, scalable, and has both CPU and GPU implementations. Nevertheless, any simulation package for computing velocity fields could be used in principle. The extent to which the dataset collection should be standardized is somewhat task specific. For example, the use of, for instance, periodic boundary conditions (versus pressure boundary conditions) can alter the velocity field near the boundary and these images are relatively small datasets, we did not want those differences to play a role in the convergence of the model, though it is possible the boundary conditions would have no effect on permeability calculation.

In theory, the need for data collection standardization should not vastly differ between FL and conventional ML settings. Issues tend to arise when the data between clients is not identically and independently distributed (non-IID) In essence, uniformly applied preprocessing techniques allow centralized ML models to handle non-IID training data more easily than FL models. FL mainly relies on local preprocessing, aggregation strategies, and model topology (*e.g.* personalized models, clustered FL, hierarchical FL, etc.) to handle non-IID training data. However, consistent and quality training data always helps ensure the reliability of the trained model.

A universally used data format in digital rock physics does not exist. We agreed to use tiff as an input format as opposed to raw binary arrays (still common in applications, including DRP) as the format resolves issues such as machine endianness and data size across different hardware.

2.2 Input data assessment

For each dataset, an image characterization was performed via a common Python script on the client side, computing geometric characterization metrics as a proxy of data quality.

First, the Minkowski functionals – pore volume, surface area, integral mean curvature, and Euler characteristic were computed using the open-source Quantimpy package [18]. For a body, Y, with a sufficiently smooth surface, δY, in 3D space, the Minkowski functionals are formulated as

$$V = \int_Y dv\,, \tag{1}$$

$$S = \int_{\delta Y} ds\,, \tag{2}$$

$$H = \int_{\delta Y} \frac{1}{2}\left[\frac{1}{R_1} + \frac{1}{R_2}\right] ds\,, \text{and} \tag{3}$$

$$X = \int_{\delta Y} \frac{1}{R_1 R_2} ds \tag{4}$$

where V, S, H, and X are the volume, surface area, integral mean curvature, and integral Gaussian curvature, respectively. dv is a volume element, ds is a surface element, and R_1 and R_2 are the principal radii of curvature of ds. The Gauss-Bonnet theorem relates X to the Euler characteristic, χ, by

$$X = 2\pi\chi. \quad (5)$$

The Euler characteristic is a topological invariant that describes the connectivity of an object. For a 3D object, χ can be expressed using the Betti numbers,

$$\chi = \beta_0 - \beta_1 + \beta_2, \quad (6)$$

where β_0 is the number of connected components, β_1 is the number of loops, and β_2 is the number of cavities.

The functionals were computed for the pore-space in the image. By way of example, the computed Minkowski functionals for the two images in Figure 2 are presented in Table 2. All images were padded with a 0 to isolate connected pore-spaces, disconnected from the volume boundaries. While LBPM contains similar characterizations, we chose QuantimPy for its Python implementation and so that the geometric characterizations need not be performed on high performance computing resources. All codes for characterizing the datasets are available on GitHub in the DPM Tools package [19].

Table 2. Minkowski functionals computed for the images in Fig. 2. The length, L, is assumed to be 1 voxel in the numbers reported below.

	Mt. Gambier Limestone	**Castlegate Sandstone**
Porosity	0.436	0.206
Surface area [L^2]	8.32e6	1.20e7
Integral mean curvature [L]	1.92e5	8.00e5
Euler characteristic (no dimension)	-2.85e3	-1.34e4

Next, we classified the heterogeneity/homogeneity of each image, following recent work in the Prodanović group [14]. The Python implementation of the classifier is also available on GitHub [19] and analyzes the variance in porosity in a moving window of increasing radius. As discussed in [14], this classifier was previously used to classify segmented images from the DRP as homogeneous or heterogeneous with high accuracy, provided no fractures were present. An example of the classification is shown in Figure 3. The starting radius for the moving window for every image is based on the maximum inscribed sphere radius, and the reported radius is relative to the starting radius.

Finally, the morphological drainage was computed using the LBPM software, following the algorithm proposed by [15, 20]. This is a fast proxy for a two-phase flow simulation of drainage [15, 20] and, in our application, gauges the inscribed radius of a representative pore-throat for the image. A discussion of the technique is available in [15], where the authors argue that for the two-phase flow simulation, the computed pore-throat radius must to be at least 5 voxels to resolve thin films. This discretization constraint is largely due to the numerics of the lattice Boltzmann solver, rather than that of the aforementioned finite resolution limitation of imaging techniques. Here, we define a *crossover saturation* as the point where the morphological drainage curve intersects a predetermined critical pore-throat radius (see Figure 4). Because of the relatively simpler finite differencing scheme for calculating single phase flow fields, we set the critical radius to 3 voxels and required that all training images have crossover saturations ≤ 0.40. The critical radius falls slightly below the lower threshold of 5 voxels where finite resolution begins to introduce bias to the computed flow measures as proposed by Saxena et al. [21, 22]. However, we found that too few samples were able to pass the crossover saturation at this critical radius to obtain a meaningful predictive model. This criterion assured adequate resolution in each image used in the FL training dataset.

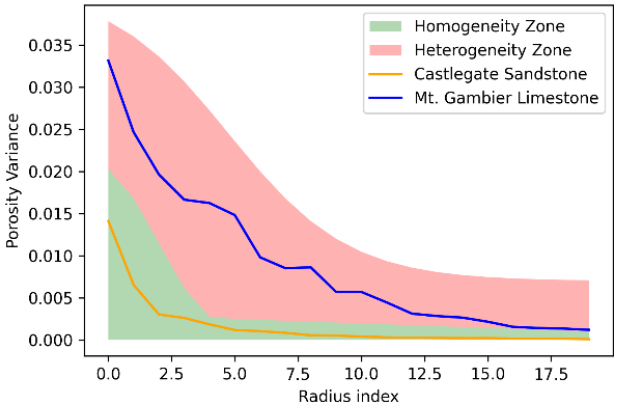

Fig. 3. The heterogeneity assessment curve provides a scale independent measure that quantifies the sample heterogeneity [14]. Here, the Mt. Gambier limestone would be classified as heterogeneous and the Castlegate sandstone as homogeneous. The zones have been predetermined based on data in the DRP.

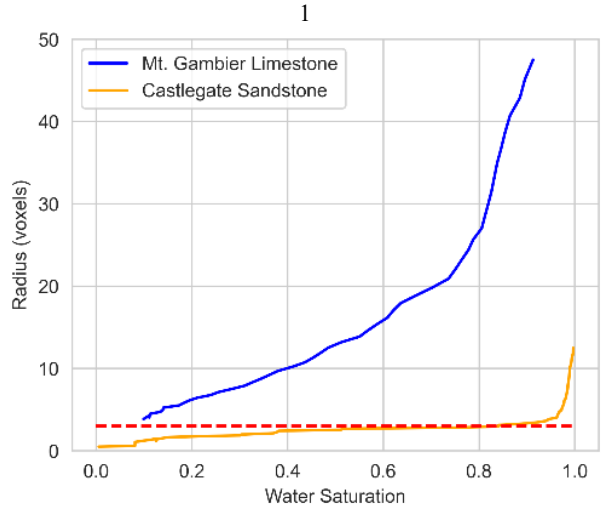

Fig. 4. Morphological drainage is a fast simulation available in both PoresPy and LBPM that provides information on whether the majority of the pore-throats, which control drainage, are resolved in the image. The red line indicates the radius of the throat deemed critical in our work.

Each client assembled an aggregate text file to return to the server with the following scalars for each image: (1) the voxel length, (2) four Minkowski functionals (*e.g.*, see Table 2), (3) the maximum inscribed radius in voxels (in order to normalize the heterogeneity curve in Figure 3), and (4) the x- and y-axis values of the

heterogeneity classifier curve. During aggregation, the server masked the names of the dataset using the following nomenclature: <client #>_<dataset #>_<image side length>. In the following results, the characterizations are shown in an aggregate, statistical form and never individually.

The methods described in this section are based on the state-of-the-art in digital rock physics and are integral in nature. They result in fewer than 100 values that are used to uniquely describe each imaged sample, and these values are the only values shared with the server, apart from the MS-Net model weights. It is impossible to recover the original sample from these few metrics, ensuring data privacy and sensitives were preserved among the clients.

2.3. Central Model for FL

We employed a system of fully connected CNNs called MS-Net, illustrated in Figure 5, as our central model. Previously, this model was shown to provide accurate predictions of single-phase velocity fields and permeabilities in a variety of artificial and geologic porous and fractured media sampled from the DRP, as shown in Figure 6 [3].

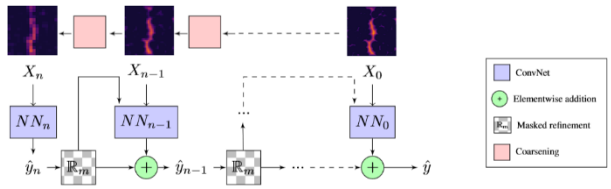

Fig. 5. The hierarchical MS-Net model architecture employed in this work for predicting permeabilities from segmented digital rocks images. Reprinted from [3].

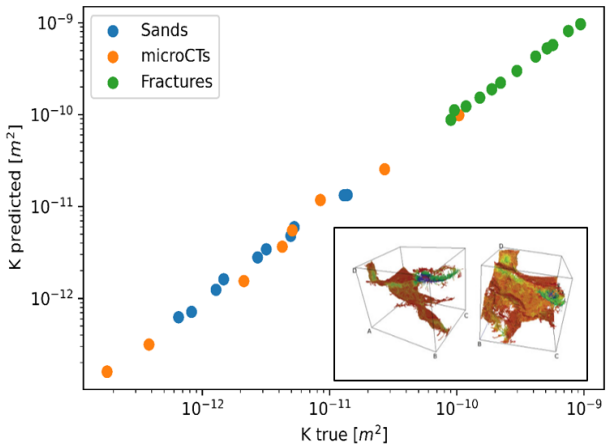

Fig. 6. The permeability predictions by MS-Net on a wide variety of imaged porous and fractured media from DRP against their true values. To our knowledge this is the only method that correctly predicts fractured rocks (such as the examples shown in the inserts). Data reproduced from [3].

2.4. FL framework

We implemented the FL framework using Flower [23]. A central server, located at Texas Advanced Computing Center, The University of Texas at Austin (TACC), distributed the central model to the clients. Each client then trained a model on its own set of training data. The central server subsequently aggregated the client-side model parameters (Figure 1). Clients included in this test are bp (Houston, TX), Petrobras (Rio de Janeiro, Brazil), and the Prodanović group at The University of Texas at Austin (Austin, TX).

Prior to coordinating the FL training, we organized an internal workshop on geometric characterization methods, LBPM, and MS-Net to align on data formats, image assessments, and software packages to be used in the training.

2.4.1 Aggregation strategy

In the traditional FL workflow, the server plays the crucial role of maintaining the central model and coordinating model updates among the clients. The process by which the central server combines the clients' models is referred to as aggregation. In parameter-based aggregation, the server combines the trainable parameters (*e.g.* weights, gradients, etc.) of the client model.

Several aggregation strategies have been proposed in the literature [24]. Here, we chose Federated Averaging (FedAvg), one of the earliest and most commonly used strategies [8]. During aggregation, clients' parameters are weighted and averaged to update the global model by,

$$w_s^{t+1} \leftarrow \sum_{k \in S_t} \frac{n_k}{n} w_k^{t+1}$$

where S_t is the set of clients selected in a particular round, t, n_k/n is the weighting factor, which accounts for the number of a client's data examples, n_k, and the total number of data examples, n, w_k^{t+1} are the updated weights of a client, k, after a local training round, and w_s^{t+1} is the aggregated global model. Because of the limited number of participants in this study, we aggregated all three clients after each communication round.

While FedAvg is widely used and straightforward to implement, it has known convergence issues [25, 26] and several other aggregation strategies have been proposed to improve the convergence of FL. Some notable strategies include: Federated Proximal (FedProx) [27], which addresses prioritization of minimizing the global objective function over minimization of clients' local objective functions; Federated Stochastic Gradient Descent (FedSGD) [8], where clients perform local stochastic gradient descent and the gradients are sent to the central model; SCAFFOLD [28], which uses variance reduction to addresses client-drift resulting from data heterogeneity; and MOON [29], which reduces discrepancies between local clients and the central server by introducing a contrastive loss as a regularization term. Nevertheless, we chose FedAvg for this experiment for simplicity and ease of implementation.

The optimizer states play an important role in ensuring the effectiveness of clients' local training, but exist on a level below the communication and aggregation processes. In a federated setting, the optimizers are not

typically synchronized across clients because they are specific to the local data distributions. It is important to note that proper model aggregation operates under the assumption that client networks encode similar feature representations at the same respective weight locations. Redistributing an aggregated model that contains feature encodings at weight locations overly different from the local model can confuse the local optimizer and severely degrade the training performance. Further discussion on prevention of variable feature encodings can be found in section 2.4.2. Assuming scenarios where there are few clients with unchanging local data distributions, such as the case here, retaining optimizer states between training rounds can lead to faster model convergence.

Finally, because of different security standards in each company, it was not possible to open a live SSL connection between the clients and the server during the training. We mitigated this issue by delivering client model weights to the server at TACC *via* SCP or FTPS, manually aggregating, then pushing the aggregated models back to the clients *via* SCP/FTPS. This practical limitation constrained our aggregation to 10 rounds with 10 local epochs per round, for a total of 100 epochs.

2.4.2 Mitigation of issues related to data heterogeneity

Like traditional machine learning, the canonical goal of the introduced FL topology is to minimize the objective function of a single central model. Because the central model is trained with data located across several devices, one must consider the possibility of varying data distributions when designing an FL workflow.

In FL settings, heterogeneity issues can manifest in numerous ways. Hardware heterogeneity is a common problem in practical applications; however, we do not address this here because the participating clients each have access to similar high-performance computing resources and communication is handled manually. Data space heterogeneity has shown to hinder local model performances and the global model convergence [25, 30, 31, 32]. Mitigating the deterioration of training performance is an open area of research that goes beyond the scope of this study. However, we describe and employ some strategies that are fundamental to making FL possible.

Successful training of a global model hinges on the ability to merge the client models' weights or gradients. One must consider that different model instances encode different information in the same weight location. Although these differences in information encoding cannot be entirely avoided, its effects can be limited by appropriately designed workflows.

We synchronize the model architecture and initializations. Though it is possible to federally train client models that differ from the central model (as is typically the case when training with small edge devices), using identical architectures between the server and clients ensures that the structure of layers, neurons, and connections remain the same across all participating clients. We also initialize the weights of the server and client models using the same random seed so that there is a stronger possibility that the models encode the data representations in a similar way.

The choice of aggregation strategy also plays an important role. The aggregation strategy used here, FedAvg, implicitly balances the differences in information encoding by giving more influence on clients that supply more training data in the central model's parameter updates. More robust aggregation strategies and regularization techniques can be used to prevent the client models from drifting too far from the global model.

Discrepancies in the sizes and distributions of client training data manifest as non-IID. The degree to which non-IID datasets affect model training depends largely on the aggregation strategy. Models using FedAvg have known convergence difficulties when training on non-IID datasets. We focus our effort on preventing clients from supplying non-IID datasets by standardizing the training data collection process by using the same LBM simulation software with identical boundary conditions and post-processing. We enforce minimum resolution and pore volume requirements to ensure the quality of simulations. Target velocity fields are normalized to enhance training performance.

Additional measures may also be deployed to mitigate the effects of non-IID datasets. Synchronizing normalization statistics (e.g., in batch normalization), domain adaptation techniques, matching features statistics, and other alignment techniques aim to standardize feature representations, making parameter aggregation more effective. Some other aggregation strategies (e.g. SCAFFOLD) have been designed to relax the constraints on non-IID data and are readily available to be implemented in future work.

As an interesting aside, personalized FL is an approach that allows each client to maintain a model instance that locally fine-tunes the global model with its own data distribution. This may provide a better solution to data heterogeneity as the global model can learn general features, while personalization addresses local specifics.

3 Results

3.1. Client data characterization

The client data characterizations (see section 2.2) were aggregated to represent each client's proposed datasets. In the spirit of privacy-preserving FL, we do not show the actual training datasets. In total, Petrobras contributed 40% of the training samples, bp contributed 36%, and UT contributed 24% after filtering by the above-mentioned criteria.

The cumulative distributions of the four Minkowski functionals for each of the three participating clients are shown in Figures 7-10.

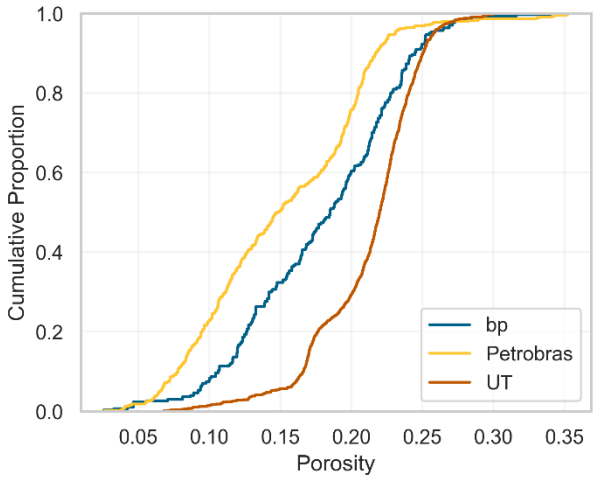

Fig. 7. Porosity data CDF of participating clients.

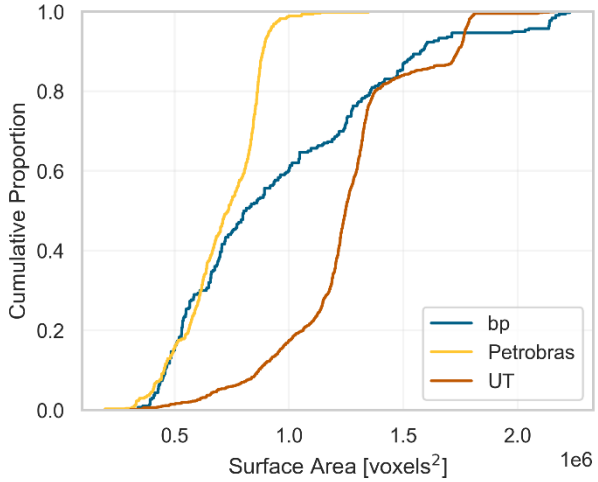

Fig. 8. Surface area data CDFs for participating clients.

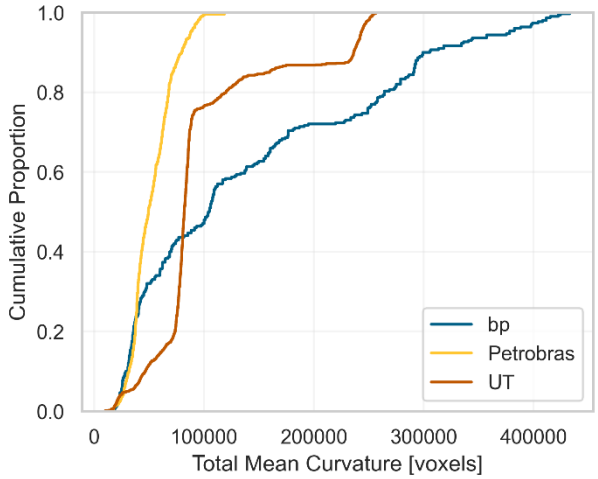

Fig. 9. Total mean curvature data CDFs for participating clients.

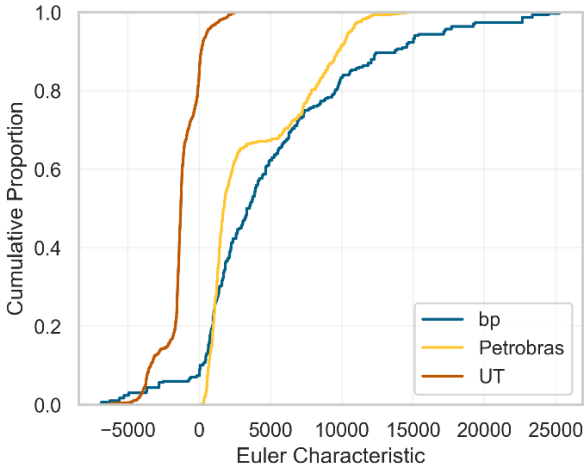

Fig. 10. Euler characteristic data CDFs for participating clients.

The porosity distribution (Figure 7) indicated that the Petrobras dataset (yellow) contained the largest proportion of tight samples, and the UT dataset (orange) had more samples with larger porosity values.

Considering the porosity distributions, the surface area (Figure 8) and integral mean curvature distributions (Figure 9) suggested that the samples in the Petrobras dataset (yellow) contained the fewest pores, with little variation in pore shapes. In contrast, the bp dataset (blue) contained the largest variation in pore shapes.

We observed a large difference in the distributions of Euler characteristic (Figure 10) between the UT dataset (orange) and the company partners' datasets (blue and yellow). The larger Euler characteristic suggested that the samples contained many connected components. The difference in distributions likely can be attributed to the removal of disconnected pores during preprocessing for the UT dataset, which was not performed on the bp and Petrobras datasets.

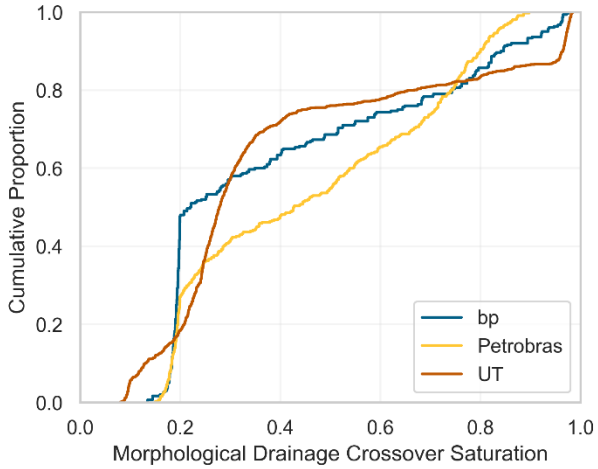

Fig. 11. Cumulative distribution of the saturation at which morphological drainage curve crosses the inscribed radius of three for all clients.

The crossover saturation provides a proxy metric for image resolution. Considering a critical crossover saturation of 0.4, the cumulative distributions (Figure 11) showed that the UT dataset (orange) had the largest proportion of well-resolved samples. The Petrobras dataset (yellow) contained the smallest proportion of samples with throat radii of at least 3 voxels. Note that this is not indicative of image quality but may be due to the nature of the rocks themselves in the datasets.

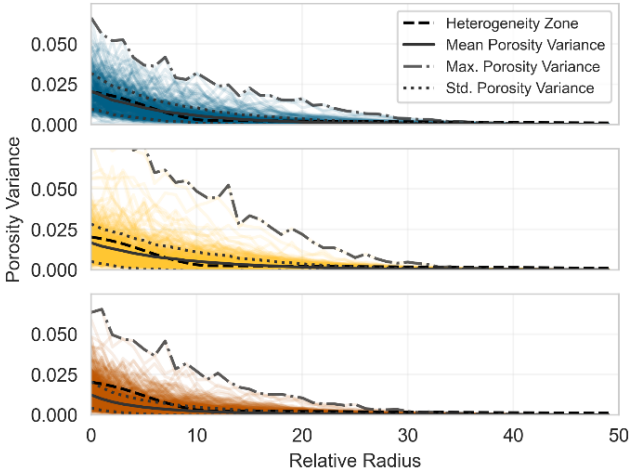

Fig. 12. Homogeneity/heterogeneity classification curves for all client data (bp top, Petrobras middle and UT bottom). The black dashed line represents the boundary of the homogeneous/heterogeneous zones. Curves above this line are considered heterogeneous. The mean, ± 1 standard deviation, and upper bound of the sample porosity variances are shown to highlight differences in dataset heterogeneity.

In Figure 12, we show the statistical ranges of the homogeneity/heterogeneity classification curves for each client. The mean and ranges of the curves indicate that the bp dataset had the highest average degree of heterogeneity. However, the upper bound of the curves suggested that the Petrobras dataset contained several highly heterogeneous samples.

3.2. Training without FL aggregation

In order to establish a baseline, each participant trained MS-Net locally for 100 epochs. The same hyperparameters, initialization seed, and respective local datasets were used as the federated training to provide a more direct comparison.

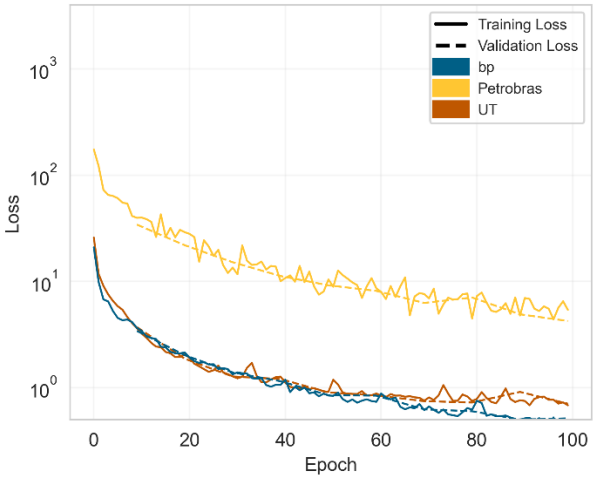

Fig. 13. Local model training and validation losses for each client. The same local training set was used in both local and federated training settings.

The loss curves for each client's local training are shown in Figure 13. All models showed continuously decreasing losses and relatively smooth convergence behavior. UT and bp training performed similarly, achieving loss values of less than 10^0. The Petrobras training was more unstable with loss values approximately one order of magnitude larger than those of UT and bp. This could be attributed to the presence of noisy or inconsistent data, which is apparent from the characterization assessment.

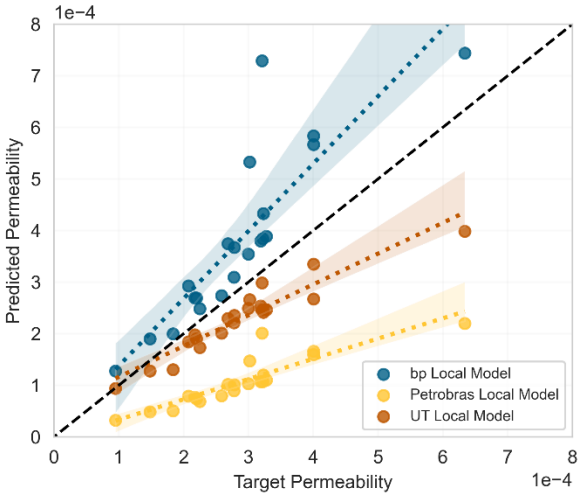

Fig. 14. Locally-trained models' permeability in lattice units^2 (LU^2) predictions on an unseen test set from DRP for model assessment. The black dashed line has a slope of one and represents perfect predictions.

Figure 14 presents the permeability predictions of each model on 25 blind test samples after 100 epochs of local training. The test set was not involved in training but

was comprised of sandstone samples from DRP. Because the UT dataset was similarly sampled from DRP, the data distributions were similar, which accounts for the closer predictions. Because of the federated setting, we did not access bp and Petrobras data for similar testing. The bp model tended to overpredict permeability values whereas the UT and Petrobras models both under-predicted the permeability.

3.3 FL training results

In this subsection, we present the results of the federated training. Note that the client and central models were each initialized with the same set of random parameters to mitigate differences in the model encodings of clients' training data distributions. A transfer learning approach is another valid approach to addressing this problem where federated training begins from a previous, centrally trained model, such as the model in [3].

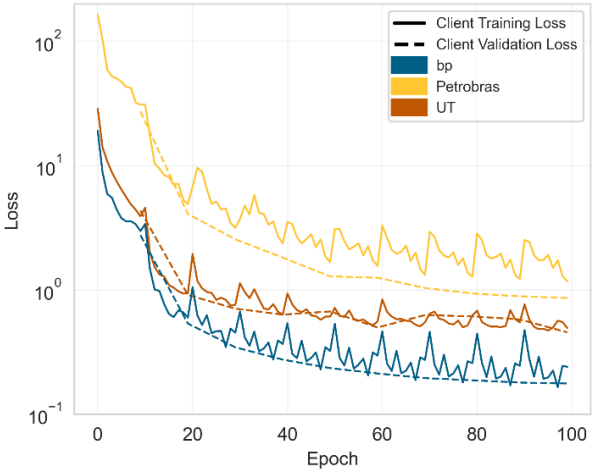

Fig. 15. Training and validation loss curves for each client. Local loss peaks appear after each communication round, when the aggregated model is copied back to the clients.

Figure 15 shows the loss curves for each client through the federated training process. The redistribution of the combined global model naturally induces local peaks in the client loss curve because of the interplay between minimizing the local cost function for the clients' particular data representations and integrating feature encodings from other client models during model aggregation. When applied at scale, the FL loss curves have been shown to be almost as smooth as centralized training. At small scales, such as those envisioned for digital rock applications, heterogeneities in client datasets have larger impacts. Further steps need to be taken to lessen the impacts of non-IID training sets. Advanced aggregation techniques and data augmentation and regularization can help the client models become more robust to the diversity in data distributions and reduce the occurrences of these peaks.

All federated models saw improved loss curves when compared to their respective locally trained models. The federated UT client model saw the smallest improvement whereas the federated Petrobras client model saw the most significant. The bp client model achieved the smallest loss value.

The magnitudes of the federated loss curves indicate further improvement is needed in model training before using them in a predictive capacity. The same approaches as traditional ML can be applied, including changes in the network hyperparameters and regularization. It is also important to note that we did not maintain the clients' optimizer states between communication rounds. Though the local peaks in the loss function would likely still be present, the clients may see better convergence in local training epochs if the optimizer states were maintained.

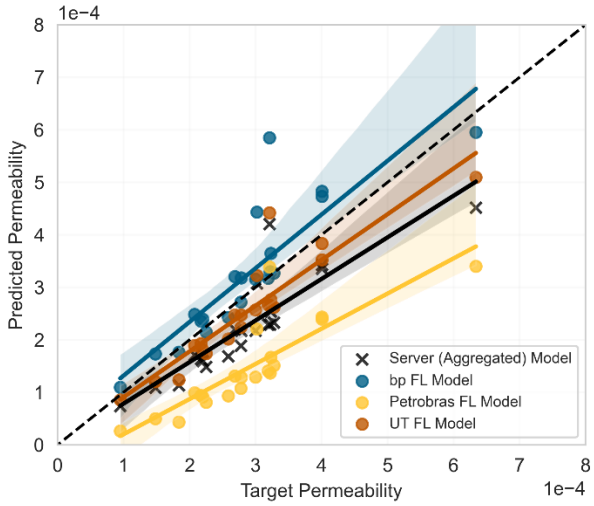

Fig. 16. Federated learning server and client models' permeability predictions (in LU^2) on the same unseen test set as used in the local model assessment. Shaded regions represent the 95% confidence interval of the permeability prediction. The black dashed line has a slope of one and represents perfect model predictions.

Next, we performed a blind permeability test (Figure 16) on the same 25 sandstone samples as the local training test (*c.f.*, Figure 14). The permeability predictions for all three federated models saw substantial improvement over their locally trained counterparts. Again, the UT client model (orange) predicted closest to expectation, likely because the training data distribution was most like the test set. The bp client model (blue) still over-predicted the absolute permeability values, whereas the Petrobras (yellow) and aggregated server models consistently under-predicted them.

The aggregated server model predictions were closest to, though consistently worse than, the UT model. This behavior is expected as it exemplifies the functionality of the FedAvg aggregation strategy. The Petrobras dataset contained the largest number of training samples, resulting in the largest weighting factor in the FedAvg aggregation strategy. On the other hand, the learned weights of the bp and UT models offset the server model's tendency toward the largest contributing client. This results in a server model that falls within an intermediate range and highlights the need to better handle non-IID datasets. Further standardization of the input features and targets could help, though it may be overly restrictive. An aggregation strategy that can better handle non-IID datasets may be a more appropriate strategy for digital rock applications. Communicating the normalization parameters and retaining optimizer states

between training rounds could also help with model convergence.

Finally, we see similar behavior across all models when comparing a particular sample prediction to their respective regression lines, suggesting that the models are encoding feature representations analogously. This further validates the efficacy of the aggregation step where parameters are simply averaged together.

3.4 Model comparison

Here, we demonstrated that FL can improve the generalizability of local models.

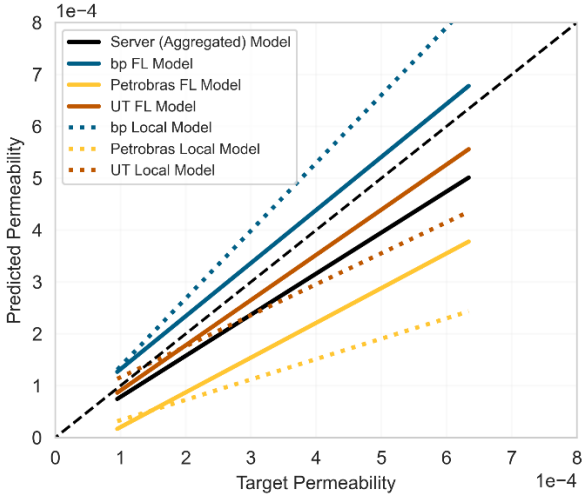

Fig. 17. Comparison between the locally trained and federated client permeability prediction (in LU^2) regression models.

Figure 17 compares the regression models for the client permeability predictions. The regression analysis shows that the bp client model predicted closest expectation. All federated models showed improvements in generalization over their respective locally trained models, with the most substantial improvements at larger permeability values. Assessment of the target velocity fields and permeabilities would help explain this behavior, but these were not evaluated in this work.

Table 3. Mean squared error between locally trained and federated client permeability predictions.

Client	Local Model	Client Model
bp	1.70e-08	5.39e-09
Petrobras	3.71e-08	2.33e-08
UT	5.82e-09	3.05e-09

Table 3 lists the mean squared error of the permeability predictions for the 25 test samples between the locally trained and federated client models after 10 training rounds. These confirm the findings shown in Figure 17, that all federated models achieved higher accuracies in their permeability predictions.

There are a variety of likely reasons for the improvements in predictive performance. For example, the removal of disconnected pores in the UT dataset reduced the effect of negative velocities due to phenomena such as recirculation. During the model aggregation process, the recirculation behavior in disconnected pores could have been "unlearned" by the Petrobras dataset. In any case, the presented experiment confirms that exposing the clients to data that they would otherwise be unable to see can help improve the generalizability of their models.

4 Discussion and Summary

We have designed data assessment protocol and performed the first FL test in the specific application of predicting velocity fields and ultimately permeability based on digital rock images. We trust that the data assessment measures proposed are adequate for concise integral data description without sharing the data itself. We have not performed any assessment of velocity fields used in training beyond agreeing to the same algorithm and boundary conditions; this is reserved for future work.

The experimental process presented some unique challenges that need to be addressed before performing larger scale experiments. Corporate and national export policies prevented the use of an open SSL channel for clients and server to communicate during the training (see Section 2.4), forcing aggregation to be done manually every 10 epochs for the total of 100 effective training epochs in this test. An open protocol could automate the process and allow for more frequent communication between the aggregation server and client training (*i.e.*, less local epochs per round), improving global minimization of the loss function. Further, it handles the maintenance of the local optimizer states between communication rounds and allow for the training to be extended to more epochs. Nevertheless, this proof-of-concept test demonstrated the promise of FL in training a model synchronously without the explicit exchange of training data.

In the blind test, the results showed that the FL client models had greater capacity for generalization than their counterparts that were trained solely on local data. It is important to note that the blind test set was prepared by the UT client and consisted of data from the same sampling pool as their training and validation data - though there was no overlap between them. The test set consisted entirely of sandstone samples. At the time of this writing, bp has prepared its own test set and reported similar improvement in generalization for their own models. Petrobras is actively preparing their own test set. All models will be evaluated on these additional test sets as the method continues to be evaluated. These results will be presented in a future version of this paper. Finally, an evaluation of the model's predictions on different lithologies would a more useful assessment of an FL model's generalizability and is left for future work.

We also plan to conduct several sensitivity analyses as we overcome communication challenges. These include varying aggregation strategy or the number of local training epochs before the aggregation is done, improving the central network architecture, and even altering the aggregation taxonomy. Future work will focus on implementing aggregation strategies that are more resistant to data heterogeneity and exploring applications of federated learning to different training tasks.

References

1. A. Rabbani, A.M. Fernando, R. Shams, A. Singh, P. Mostaghimi, and M. Babaei, Water Resour. Res. **57**, e2020WR029472, (2021)
2. Y.D. Wang, M.J. Blunt, R.T. Armstrong, and P. Mostaghimi, Earth Sci. Rev. **215**, 103555, (2021)
3. J.E. Santos, Y. Yin, H. Jo, et al., Transp. Porous Med. **140**, 241–272, (2021)
4. J.E. Santos, D. Xu, H. Jo, C.J. Landry, M. Prodanović, and M.J. Pyrcz, Adv. Water Resour. **138**, 103539, (2020)
5. A. Marcato, J.E. Santos, G. Boccardo, H. Viswanathan, D. Marchisio, and M. Prodanović, Chem. Eng. J. **455**, 140367, (2023)
6. B. Chang, J. Santos, R. Victor, H. Viswanathan, and M. Prodanovic, SPE Annu. Tech. Conf. Exhib. (2022)
7. M. Prodanović, M. Esteva, J. Mcclure, et al., E3S Web of Conferences. **367**, (2023)
8. H.B. McMahan, E. Moore, D. Ramage, S. Hampson, and B.A.Y. Arcas, Intl. Conf. Artif. Intell. and Stat. (2016)
9. P. Kairouz, H.B. McMahan, B. Avent, et al., Found. Trends Mach. Learn. **14**, 1–210, (2021)
10. T. Christensen, C. Ladino, and D. Clarkin, NOAA Artif. Intell. Workshop. (2020)
11. U.S. Department of Commerce, Online. https://www.commerce.gov/news/blog/2024/01/preparing-open-data-age-ai, (2024)
12. C. H. Arns, M. A. Knackstedt, and K. Mecke, J. Microsc. **240**, 181–196, (2010)
13. R.T. Armstrong, J.E. McClure, V. Robins, et al., Transp. Porous Med. **130**, 305–335, (2019)
14. A. Mohamed and M. Prodanović, Transp. Porous Med. **150**, 257–284, (2023)
15. J.E. McClure, Z. Li, M. Berrill, and T. Ramstad, Comput. Geosci. **25**, 871–895, (2021)
16. P. Iassonov, T. Gebrenegus, and M. Tuller, Water Resour. Res. **45**, (2009)
17. A. Sheppard and G. Schroeder-Turk, Digit. Rocks Portal. (2015)
18. A.M.P. Boelens and H.A. Tchelepi, SoftwareX. **16**, 100823, (2021)
19. Digital Porous Media, DPM Tools. [Source code] https://github.com/digital-porous-media/dpm_tools, (2024)
20. M. Hilpert and C.T. Miller, Adv. Water Resour. **24**, 243–255, (2001)
21. N. Saxena, A. Hows, R. Hofmann, et al., Adv. Water Resour. 134, 103419, (2019)
22. L. Leu, S. Berg, F. Enzmann, R.T. Armstrong, and M. Kersten, Transp. Porous Med. 105, 451–469, (2014)
23. D.J. Beutel, T. Topal, A. Mathur, X. Qiu, T. Parcollet, and N. Lane, ArXiv. (2020)
24. P. Qi, D. Chiaro, A. Guzzo, M. Ianni, G. Fortino, and F. Piccialli, Future Gener. Comput. Syst. **150**, 272–293, (2024)
25. B. Chang, R. Victor, M. Esteva, M. Prodanović, Am. Geophys. Union. (2023)
26. X. Li, K. Huang, W. Yang, S. Wang, and Z. Zhang, ArXiv. (2019)
27. A.K. Sahu, T. Li, M. Sanjabi, M. Zaheer, A. Talwalkar, and V. Smith, ArXiv: Learning. (2018)
28. S.J. Reddi, Z.B. Charles, M. Zaheer, et al., ArXiv. (2020)
29. Q. Li, B. He, and D. Song, IEEE/CVF Conf. Comput. Vis. Pattern Recognit. 10708–10717, (2021)
30. D. Gao, X. Yao, and Q. Yang, ArXiv. (2022)
31. K. Stacke, G. Eilertsen, J. Unger, and C. Lundström, IEEE J. Biomed. Health Inform. **25**, 325–336, (2021)
32. A.M. Abdelmoniem, C.-Y. Ho, P. Papageorgiou, and M. Canini, IEEE Internet Things J. **10**, 14071–14083, (2023)

Appendix

Boundary conditions for the LBPM simulation

We impose consistent boundary conditions across different clients. We use constant pressure BCs to avoid boundary layer effects, that said the choice of BC is the users' prerogative and is not expected to significantly affect the results of this study. The following provides the relevant input configurations for LBPM that were used for all training images:

```
Domain {
  Filename="segmented.raw"
  voxel_length=1.0
  N = 256, 256, 256
  n = 256, 256, 256
  nproc = 1, 1, 1
  ReadType ="8bit"
  // key values set by image labeling
  ReadValues = 0, 1
  WriteValues = 0, 1
  // boundary conditions
  BC = 3
}
MRT {
  timestepMax = 100000
  tau = 1.0
  F = 0.0, 0.0, 0.0
  din = 1.001
  dout = 0.999
  tolerance = 0.00001
}
```

FL comparison with a local model trained with all datasets

We are unable to perform a direct comparison between the FL model and a local model trained with all datasets supplied in this study as we agreed not to transfer any of the training samples provided by bp and Petrobras. Instead, we show results of some preliminary work conducted prior to this experiment using only open-access data from DRP.

In this study, we compare the training performances of four FL models containing two participating clients each with that of a centralized model that sees the exact aggregate training set (Figure 18). The model used in the preliminary study is identical to the one used in this experiment.

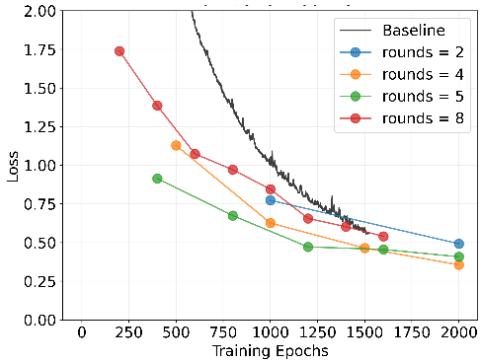

Fig. 18. Comparison between the locally trained and federated model training on identical aggregate datasets.

We observe that the overall training performance of the baseline model is maintained when applied in a federated setting.

Timing profile and comments on scalability

As part of the preliminary study, we simulated an FL workflow with three clients and computed the average run time over 10 runs on a desktop computer for each major step in the FL algorithm and normalized by the total time to perform equivalent centralized training (Figure 19).

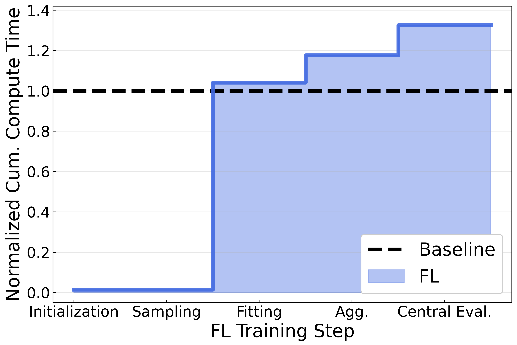

Fig. 19. Average timing profile of simulated FL workflow with three clients over 10 runs on a desktop computer for each major step in the FL algorithm. Averaged times are normalized by the total time to perform equivalent centralized training

FL requires some additional communication overhead over centralized training. The impact of this overhead heavily depends on the size of the model and FL hyperparameters.

We have not directly tested the scalability of FL for digital rocks applications; however, the workflow has already been widely applied across the technology and medical industries, often with thousands of clients. Some issues have already been identified when applying this framework on a large scale. These include diverse hardware capabilities leading to straggler clients, bandwidth limitations and network latency due to increased communication overhead, maintaining training data representativity, and increased likelihood of adversarial attacks.