

FedXDS: Leveraging Model Attribution Methods to counteract Data Heterogeneity in Federated Learning

Maximilian Andreas Hoefler¹ * Karsten Mueller¹ Wojciech Samek^{1,2,3}

¹Fraunhofer Heinrich Hertz Institute, Berlin, Germany

²Technical University of Berlin, Berlin, Germany

³Berlin Institute for the Foundations of Learning and Data (BIFOLD)

Abstract

Explainable AI (XAI) methods have demonstrated significant success in recent years at identifying relevant features in input data that drive deep learning model decisions, enhancing interpretability for users. However, the potential of XAI beyond providing model transparency has remained largely unexplored in adjacent machine learning domains. In this paper, we show for the first time how XAI can be utilized in the context of federated learning. Specifically, while federated learning enables collaborative model training without raw data sharing, it suffers from performance degradation when client data distributions exhibit statistical heterogeneity. We introduce FedXDS (Federated Learning via XAI-guided Data Sharing), the first approach to utilize feature attribution techniques to identify precisely which data elements should be selectively shared between clients to mitigate heterogeneity. By employing propagation-based attribution, our method identifies task-relevant features through a single backward pass, enabling selective data sharing that aligns client contributions. To protect sensitive information, we incorporate metric privacy techniques that provide formal privacy guarantees while preserving utility. Experimental results demonstrate that our approach consistently achieves higher accuracy and faster convergence compared to existing methods across varying client numbers and heterogeneity settings. We provide theoretical privacy guarantees and empirically demonstrate robustness against both membership inference and feature inversion attacks. Code is available at <https://github.com/MaxH1996/FedXDS>.

1. Introduction

Federated learning (FL) [19] has received significant attention in recent years, proving to be an effective method for collaborative distributed machine learning while keeping

local datasets private. However, the practical deployment of federated learning systems faces fundamental challenges regarding data heterogeneity and privacy [11, 35].

Data heterogeneity, characterized by non-IID data across clients, leads to contradicting model updates and significantly degraded performance [12, 31]. Current approaches address this challenge by aligning client distributions using proximal optimization terms [16] and mitigating client drift [12, 14], while others focus on flattening the loss landscape to facilitate better model aggregation [20, 21]. In addition, recent works have shown that sharing even a small portion of client data globally can significantly improve local model generalization [34]. Such shared data introduces centralization, giving clients access to a more homogeneous distribution, which reduces update divergence and mitigates heterogeneity [34].

However, sharing raw data is not feasible due to privacy constraints. Introducing differential privacy [9] by adding noise to the shared data offers a potential solution, however this can cause considerable performance loss. Alternatively, instead of sharing raw data, abstract feature representations or partial data could be released using generator-type approaches [30, 33, 36]. However, the obtained features are not as potent as raw data in mitigating statistical heterogeneity, and can also be prone to privacy leaks and additionally introduce significant computational overhead due to generators. Hence, the challenge arises of *how to retain the performance improvement from data sharing while simultaneously guaranteeing privacy*.

To address this challenge we propose FedXDS (Federated Learning via XAI-guided Data Sharing) which entails a novel approach wherein we leverage methods from the field of eXplainable AI (XAI) [8] enabling performance enhancing data sharing while preserving privacy.

Specifically, we leverage propagation-based attribution methods, which have been shown to reliably identify input features that consistently contribute to model predictions [2, 8]. Moreover, attribution maps yield a pixel-wise relevance score which highlights regions in the input space

*Correspondence to maximilian.andreas.hoefler@hhi.fraunhofer.de

which focus on semantically meaningful features, while suppressing spurious correlations and background noise. In our setup local client data is filtered through the attribution mechanism, i.e., we only share the input features which the attribution map deems most relevant for model predictions. This retains the information needed for model generalization while discarding irrelevant information.

We show that our method, when combined with differential-metric privacy techniques [6, 9], offers stronger privacy guarantees compared to applying similar privacy mechanisms directly to raw input features. Specifically, by discarding task-irrelevant information we effectively reduce the dimensionality of the raw features. This allows us to selectively protect only the task-relevant regions rather than uniformly protecting all input pixels, and thus obtain better utility.

In addition, propagation-based attributions can be obtained via a single backward pass over the local dataset, which only needs to be performed once in the entire FL process.

Our main contributions can be summarized as:

- A novel federated learning algorithm leveraging propagation-based attributions to address the challenge of retaining performance from data sharing with privacy guarantees in heterogeneous environments.
- A privacy-preserving mechanism that utilizes attribution-guided dimensionality reduction to achieve metric differential privacy. We perform theoretical and empirical privacy evaluations on our feature sharing approach through membership inference attacks and feature inversion, demonstrating that we can attain strong privacy guarantees.
- Extensive experiments on image datasets benchmarking our method against current state of the art, demonstrating superior performance in terms of accuracy and convergence.

2. Preliminaries

2.1. Federated Learning

Federated Learning [19] is a distributed machine learning paradigm where multiple clients collaborate to train a model under the orchestration of a central server, without sharing their raw data. Let K be the number of clients, each with a local dataset $\mathcal{D}_k = \{(x_i^k, y_i^k)\}_{i=1}^{n_k}$, where $x_i^k \in \mathcal{X}$ is an input sample and $y_i^k \in \mathcal{Y}$ is its corresponding label. The goal is to learn a global model $f_\theta : \mathcal{X} \rightarrow \mathcal{Y}$ parameterized by θ , which minimizes the empirical risk:

$$\min_{\theta} \frac{1}{N} \sum_{k=1}^K \sum_{i=1}^{n_k} \mathcal{L}(f_\theta(x_i^k), y_i^k) \quad (1)$$

where $N = \sum_{k=1}^K n_k$ is the total number of samples

across all clients, and \mathcal{L} is a suitable loss function.

The FedAvg algorithm [19] iteratively aggregates model updates from selected clients S_t , where each client k performs local updates $\theta_k^t = \theta^{t-1} - \eta \nabla \mathcal{L}_k(\theta^{t-1})$ before the server averages these updates as $\theta^t = \frac{1}{|S_t|} \sum_{k \in S_t} \theta_k^t$.

2.2. Differential Privacy in Metric Spaces

Standard differential privacy (DP) [9] provides guarantees against membership inference by bounding the output distribution change when a single dataset element is modified. For continuous data such as image embeddings, classical adjacency notions are restrictive, making *metric differential privacy* [6] more appropriate:

Definition 1 (Metric Privacy). [6] *Let (X, d_X) be a metric space and Z a set of possible outputs. A randomized mechanism $A : X \rightarrow \Delta(Z)$ is (ϵ, δ) -metric private if for all $x, x' \in X$ and every measurable set $U \subseteq Z$:*

$$\Pr[A(x) \in U] \leq \exp(\epsilon d_X(x, x')) \Pr[A(x') \in U] + \delta.$$

This definition reduces to standard DP when $d_X(x, x') = 1$ for all adjacent datasets, but naturally captures similarity in continuous spaces. In our work, we define (X, d_X) where $X = \mathbb{R}^d$ represents image data and $d_X(x, x') = \|x - x'\|_2$ is the ℓ_2 distance between inputs.

To guarantee privacy, we calibrate noise based on the sensitivity of a query function $f : X \rightarrow \mathbb{R}^m$, which measures output changes relative to input changes. In our work we choose the following sensitivity measure as used in [4, 10, 26]:

Definition 2 (Sensitivity). *For a function $f : X \rightarrow \mathbb{R}^m$, the sensitivity is:*

$$\Delta_f = \max_{x, x' \in X} \frac{\|f(x) - f(x')\|}{\|x - x'\|}.$$

This definition captures the largest possible output difference relative to input difference, directly controlling how much noise is needed for privacy. This notion of sensitivity offers several advantages, which we discuss in the supplementary material. Importantly, when Δ_f is small, less noise is required to achieve the same privacy guarantee, preserving more signal and improving utility—particularly valuable for high-dimensional data like images.

For privacy preservation, we employ the Gaussian mechanism, which adds noise to the output of our query function proportional to the sensitivity Δ_f :

Theorem 1 (Gaussian Mechanism). [6, 9] *For a function $f : \mathcal{D} \rightarrow \mathbb{R}^d$ with sensitivity Δ_f , adding Gaussian noise with scale σ satisfying:*

$$\sigma \geq \frac{\Delta_f \sqrt{2 \log(1.25/\delta)}}{\epsilon} \quad (2)$$

ensures $f(x) + \mathcal{N}(0, \sigma^2 I)$ is (ϵ, δ) -metric private.

This mechanism forms the foundation of our privacy-preserving feature sharing approach, allowing us to bound and quantify the privacy guarantees of our federated learning framework.

2.3. Neural Network Attribution Methods

We consider several gradient-based attribution techniques in our work. The simplest being Gradient \times Input [25], which computes attributions through element-wise multiplication of input and gradient:

$$\mathcal{A}_{\text{grad}}(f_{\theta}, \mathbf{x}) = \mathbf{x} \odot \frac{\partial f_{\theta}}{\partial \mathbf{x}} \quad (3)$$

Integrated Gradients (IG) [28] addresses gradient saturation by accumulating gradients along a path from baseline \mathbf{x}' to input:

$$\mathcal{A}_{\text{IG}}(f_{\theta}, \mathbf{x}) = (\mathbf{x} - \mathbf{x}') \odot \int_0^1 \frac{\partial f_{\theta}(\mathbf{x}' + \alpha(\mathbf{x} - \mathbf{x}'))}{\partial \mathbf{x}} d\alpha \quad (4)$$

SmoothGrad [27] reduces attribution noise by averaging gradients over perturbed inputs:

$$\mathcal{A}_{\text{smooth}}(f_{\theta}, \mathbf{x}) = \frac{1}{N} \sum_{i=1}^N \frac{\partial f_{\theta}}{\partial (\mathbf{x} + \epsilon)} \quad (5)$$

Lastly we use Layer-wise Relevance Propagation (LRP) with the ϵ -rule according to [3]. This recursively propagates relevance scores $R^{(l)}$ from layer l to $l - 1$ using:

$$R_j = \sum \frac{a_j w_{jk}}{\epsilon + \sum_{0,j} a_j w_{jk}} R_k \quad (6)$$

where w_{jk} are layer weights, a_j are layer activations, and ϵ is a small stabilizing term.

3. Related Work

Federated Learning was introduced with FedAvg [19], enabling collaborative model training without sharing raw data, but suffers under statistical heterogeneity. Regularization-based methods mitigate client drift by constraining local updates, such as FedProx [16], SCAF-FOLD [12], and FedDyn [1], with extensions like FedBN [17], MOON [15], FedNova [29], FedSAM [21], and FedDISCO [32] targeting specific aspects like feature shifts or loss geometry. A more direct solution is data or knowledge sharing: FedDF and FedAux [18, 24] distill client knowledge using a public dataset; FedGen [36] synthesizes class-conditional features for distribution alignment; FedFTG [33] generates pseudo-data for feature-level transfer; and FedFed [30] applies VAE-based feature distillation. While effective, these methods often introduce significant computational and privacy costs due to data synthesis and sharing. In contrast, FedXDS shares compact

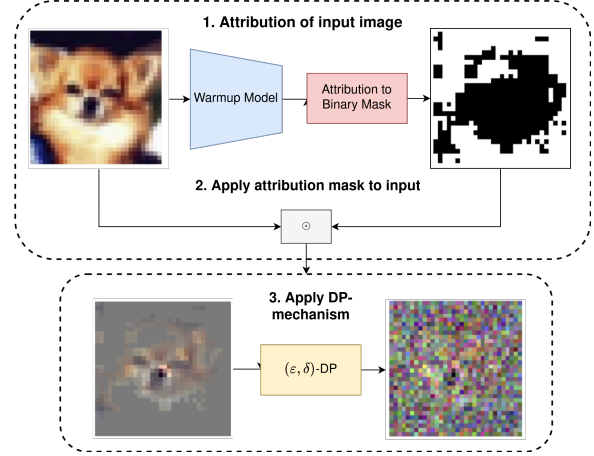


Figure 1. Illustration of the DP attribution-guided feature extraction. The process consists of four main steps: (1) Computing attribution scores using a warmup model to identify important features, (2) Creating and applying a binary mask based on the attributions, (3) Adding Gaussian noise for metric privacy.

Algorithm 1 Attribution-Guided Private Representation Extraction.

Require: D_k : client dataset, f_{θ} : model, s : sparsity level, (ϵ, δ) : privacy params, S : sensitivity, θ_{warmup}
Ensure: Private dataset D_k^p

- 1: **for** $\mathbf{x}_i^k \in D_k$ **do**
- 2: $\mathbf{h} \leftarrow \mathcal{A}(f_{\theta_{\text{warmup}}}, \mathbf{x}_i^k)$; $h^{(s)} \leftarrow s$ -th largest in \mathbf{h}
- 3: $\mathbf{m} \leftarrow [\mathbf{h}[i] \geq h^{(s)}]$; $\mathbf{x}_m \leftarrow \mathbf{x}_i^k \odot \mathbf{m}$
- 4: $\tilde{\mathbf{x}}_i^k \leftarrow \mathbf{x}_m + \mathcal{N}(0, \sigma^2)$ $\triangleright \sigma$ from (ϵ, δ)
- 5: Add $(\tilde{\mathbf{x}}_i^k, y_i^k)$ to D_k^p
- 6: **end for**
- 7: **return** D_k^p

attribution-based feature subsets derived via XAI, avoiding generators while enhancing privacy, efficiency, and performance in heterogeneous settings. We provide an extense related works in the supplementary material.

4. FedXDS Approach

Consider a federated learning system with N clients, where each client k holds local data \mathcal{D}_k . In our setting all clients share the same neural network architecture f_{θ} . Our goal is to learn a global model Θ_G which generalizes well on all client datasets.

4.1. Attribution-Guided Feature Selection

The first stage of our method identifies discriminative features using a pre-trained warmup model, with parameters θ_{warmup} . This is achieved by training with FedAvg for R_{warmup} rounds, before sharing representations. Then, for each input sample $\mathbf{x} \in \mathbb{R}^{H \times W \times 3}$, where H and W are the

Algorithm 2 Privacy-Preserving Federated Training

Require: $\{D_k\}_{k=1}^K$: client datasets, λ : knowledge weight,
 T : rounds, E : local epochs, η : learning rate, R_{warmup}

- 1: Initialize θ_{warmup} using R_{warmup} rounds of FedAvg
- 2: Obtain D_k^p using Algorithm 1 for all $k \in [K]$ in parallel
- 3: Server: $D_g \leftarrow \bigcup_{k=1}^K D_k^p$
- 4: **for** round $t = 1$ to T **do**
- 5: Select client subset $S_t \subseteq [K]$
- 6: **for** $k \in S_t$ **in parallel do**
- 7: $\theta_k^t \leftarrow \theta^{t-1}$ \triangleright Download global model
- 8: **for** epoch $e = 1$ to E **do**
- 9: **for** batch B from D_k, B_g from D_g **do**
- 10: $\mathcal{L}_k^t \leftarrow \frac{1}{|B_k|} \sum_{(\mathbf{x}, y) \in B_k} \ell(f_{\theta_k^t}(\mathbf{x}), y) +$
 $\lambda \frac{1}{|B_g|} \sum_{(\mathbf{x}_g, y_g) \in B_g} \ell(f_{\theta_k^t}(\mathbf{x}_g), y_g)$
- 11: $\theta_k^t \leftarrow \theta_k^t - \eta \nabla \mathcal{L}$
- 12: **end for**
- 13: **end for**
- 14: **end for**
- 15: Server: $\theta^t \leftarrow \frac{1}{|S_t|} \sum_{k \in S_t} \theta_k^t$ \triangleright FedAvg
- 16: **end for**

dimensions of the image with three color channels, in the client dataset \mathcal{D}_k , we compute relevances using an attribution method \mathcal{A} from Section 2.3 :

$$\mathbf{h} = \mathcal{A}(f_\theta, \mathbf{x}) \quad (7)$$

where $\mathbf{h} \in \mathbb{R}^{H \times W}$ is the pixel-wise importance score of an input image \mathbf{x} . These attribution scores quantify the importance of each input feature to the model’s prediction. To focus on the most relevant features, we create a binary mask \mathbf{m} , as shown in Equation 8, by retaining features whose attribution scores exceed a threshold determined by the desired sparsity level s :

$$\mathbf{m}[i] = \begin{cases} 1 & \text{if } \mathbf{h}[i] \geq h^{(s)} \\ 0 & \text{otherwise} \end{cases} \quad (8)$$

where $h^{(s)}$ represents the s -th largest value in the attribution scores. This mask identifies the subset of features that are most crucial for the model’s decision-making process.

We define our feature selection function $f_A : \mathbb{R}^{H \times W \times 3} \rightarrow \mathbb{R}^{H \times W \times 3}$ as the application of the attribution mask \mathbf{m} to each image $\mathbf{x} \in \mathcal{D}_k$ through:

$$f_A(\mathbf{x}) = \mathbf{x} \odot \mathbf{m} \quad (9)$$

This operation emphasizes the most informative parts of the input while suppressing less relevant details. However, directly sharing $f_A(\mathbf{x})$ would still pose privacy risks. Therefore, we design a privacy mechanism \mathcal{M} that adds calibrated noise to these masked features (Section re/subsec:privacy):

$$\mathcal{M}(\mathbf{x}) = f_A(\mathbf{x}) + \mathcal{N}(0, \sigma^2 \mathbf{I}) \quad (10)$$

where σ is determined based on privacy requirements. The resulting privacy-protected features, along with their corresponding labels, are then aggregated into a new dataset $\mathcal{D}_k^p = \{(\mathcal{M}(\mathbf{x}), y) | (\mathbf{x}, y) \in \mathcal{D}_k\}$ and shared with the server. The full data generated process is outlined in Algorithm 1 and visualized in Figure 1.

4.2. Local Training using Shared Data

After the privacy-protected features have been obtained, the server aggregates private representations from all clients into a global dataset $\mathcal{D}_g = \bigcup_{k=1}^N \mathcal{D}_k^p$, and sends this dataset to each client. The clients then optimize a composite objective that balances local task performance with knowledge from the global private dataset:

$$\begin{aligned} \min_{\theta} [& \mathbb{E}_{(\mathbf{x}, y) \sim \mathcal{D}_k} [\ell(f_\theta(\mathbf{x}), y)] \\ & + \lambda \mathbb{E}_{(\mathbf{x}, y) \sim \mathcal{D}_g} [\ell(f_\theta(\mathbf{x}), y)] \end{aligned} \quad (11)$$

The hyperparameter λ controls the trade-off between local specialization and global knowledge integration. This formulation allows clients to benefit from collective learning while maintaining their own task-specific performance and ensuring privacy. The procedure is outlined in Algorithm 2.

4.3. Privacy Discussion

We build on the insight that sharing features derived from raw client data can mitigate data heterogeneity in federated learning [34]. Nevertheless, directly sharing these features poses significant privacy risks. A straightforward approach would be to apply differential privacy (DP) to the raw data itself; however, the high dimensionality of image data forces the addition of very large noise levels, thereby significantly degrading utility.

To address this challenge, we propose a two-step approach. First, we use an attribution-guided masking strategy (Section 4.1) to identify and retain only task-relevant pixels. Second, we add Gaussian noise (Theorem 1) to the resulting masked features to achieve the desired ϵ -metric privacy (Definition 1, Definition 2).

This design offers two main advantages. First, by sparsifying input features that do not contribute to classification predictions, we reduce the overall sensitivity of the data. According to Theorem 1, lower sensitivity requires less noise to meet the same privacy budget ϵ , thus improving utility. Second, in contrast to random or magnitude-based sparsification that may unintentionally discard valuable information, our attribution-guided approach specifically preserves the pixels most critical for the learning task while eliminating irrelevant ones.

The primary goal of our privacy mechanism is to protect against membership inference attacks (MIA) and feature inversion, which we empirically validate in section 6.

In the following section, we formally demonstrate how our strategy guarantees metric privacy, while a comprehensive discussion of privacy assumptions appears in the appendix.

4.3.1. Sensitivity Analysis of Attribution-Based Masking

A critical component of ensuring privacy is understanding the sensitivity of our feature selection function $f(\mathbf{x})$ defined in Equation 9. From Theorem 1, the noise required to attain a certain privacy level ε depends on the sensitivity Δ_f . Our goal is to minimize this sensitivity before applying the privacy mechanism \mathcal{M} .

For our feature selection function $f_A(\mathbf{x}) = \mathbf{x} \odot \mathbf{m}$, we can derive a strict bound on sensitivity. Since each coordinate of the mask satisfies $\mathbf{m}[i] \in \{0, 1\}$, for any two inputs $\mathbf{x}, \mathbf{x}' \in \mathbb{R}^{H \times W \times 3}$, we have, in the worst case:

$$|\mathbf{m}[i]\mathbf{x}[i] - \mathbf{m}[i]\mathbf{x}'[i]| \leq |\mathbf{x}[i] - \mathbf{x}'[i]|$$

Thus, in aggregate:

$$\|f_A(\mathbf{x}) - f_A(\mathbf{x}')\|_2 \leq \|\mathbf{x} - \mathbf{x}'\|_2$$

This shows that our masking operation is non-expansive with respect to the ℓ_2 norm. Rearranging the equation:

$$\Delta_f = \frac{\|f_A(\mathbf{x}) - f_A(\mathbf{x}')\|_2}{\|\mathbf{x} - \mathbf{x}'\|_2} \leq 1 \quad (12)$$

where Δ_f is the sensitivity we defined in the preliminaries in Definition 2. Without our attribution-based masking, the sensitivity would be unbounded for arbitrary transformations. Moreover, in the worst case f_A is the identity, i.e. $f_A(x) = x$, meaning that the attribution method deems all pixels important. This implies that our method would reduce to the naive approach of adding noise to the raw features. In practice, this rarely occurs, since this would already require a highly sparsified image. Nevertheless, our approach safeguards against this worst-case scenario. More discussion can be found in the supplementary material.

4.3.2. Privacy Mechanism and Guarantees

Based on the sensitivity bound established above, we design our privacy mechanism \mathcal{M} (Equation 10) by adding Gaussian noise calibrated to this reduced sensitivity:

$$\mathcal{M}(\mathbf{x}) = f_A(\mathbf{x}) + \mathcal{N}(0, \sigma^2 \mathbf{I})$$

where σ is chosen according to Theorem 1 for an (ε, δ) -level of privacy given sensitivity $\Delta_f \leq 1$. This ensures that for all input images in the client dataset, $\mathbf{x}, \mathbf{x}' \in D_k$, our method satisfies (ε, δ) -metric privacy.

Our theoretical privacy guarantees are empirically validated through membership inference and feature inversion attacks Section 6, where attribution-masked features consistently show stronger protection than unmasked features under equivalent privacy budgets.

5. Experimental

5.1. Experimental Setup

We evaluate FedXDS on standard image classification benchmarks (CIFAR-10, CIFAR-100, Tiny-ImageNet) and real-world federated datasets from the LEAF framework [5] (CelebA, FEMNIST) that naturally exhibit heterogeneous distributions. For the standard benchmarks, we simulate statistical heterogeneity using Dirichlet-based partitioning with concentration parameters $\alpha \in \{0.05, 0.1\}$, where smaller values indicate greater heterogeneity. We conduct experiments with both 10 and 100 clients, setting participation rates to 0.5 and 0.1, respectively, and train for $R = 200$ communication rounds. To compute relevance-based attributions, we evaluate four attribution methods in our FL framework: Layerwise Relevance Propagation (FedXLRP) [3], Integrated Gradients (FedXIG) [28], SmoothGrad (FedXSG) [27], and Input \times Gradient (FedX-Grad). Unless otherwise stated we use $\varepsilon = 20$ sparsity level $s = 70\%$, and $\lambda = 0.5$.

5.2. Baselines

The experimental results in Table 1 shows that FedXLRP consistently outperforms all baselines across varying client numbers and data distributions. This advantage amplifies in more challenging scenarios ($K=100, \alpha=0.05$). FedFed ranks second, while FedAvg and FedAux degrade significantly with increased client numbers. On CIFAR-100, performance gaps narrow, though FedXLRP maintains its edge, particularly with higher client counts and heterogeneity. Other attribution methods (FedXIG, FedXGrad, FedXSG) perform well but remain below FedXLRP, highlighting attribution method choice importance. We discuss this point in Subsection 7.1. In addition, communication efficiency results Table 2 reinforce FedXLRP's advantages. To reach 70% accuracy on CIFAR-10 with 10 clients, FedXLRP requires only 14 rounds versus 49 for FedAvg and 28 for FedFTG. With 100 clients, FedXLRP needs 23 rounds to reach 60% accuracy compared to FedAvg's 79. CIFAR-100 shows similar patterns, with FedXLRP achieving target accuracies in substantially fewer rounds across client configurations. Among baselines, only FedFed shows comparable efficiency.

5.3. Experiments on Real-World Datasets

We also evaluate FedXDS on real-world datasets using implementations from the LEAF [5] and [7] frameworks. Specifically, we consider CelebA and FEMNIST, which exhibit a distinct form of non-IIDness known as distribution shift, introducing an additional challenge for federated learning. For CelebA, we follow the implementation of [36], while for FEMNIST, we use the setup from [7], conducting experiments with 10 and 100 clients, respectively.

Table 1. Performance comparison across different federated learning methods, including standard deviations from 5 runs. Results show the top-1 test accuracy (%) on CIFAR-10, CIFAR-100, and Tiny-ImageNet under varying numbers of clients (K) and Dirichlet concentration parameters (α). Bold values indicate the best performance in each column.

Dataset	CIFAR-10				CIFAR-100				Tiny-ImageNet			
	K=10		K=100		K=10		K=100		K=10		K=100	
	$\alpha = 0.05$	$\alpha = 0.1$	$\alpha = 0.05$	$\alpha = 0.1$	$\alpha = 0.05$	$\alpha = 0.1$	$\alpha = 0.05$	$\alpha = 0.1$	$\alpha = 0.05$	$\alpha = 0.1$	$\alpha = 0.05$	$\alpha = 0.1$
FedAvg	64.16 ± 0.71	72.90 ± 0.55	55.53 ± 0.82	60.94 ± 0.68	43.82 ± 0.45	44.57 ± 0.51	32.99 ± 0.73	35.59 ± 0.62	30.30 ± 0.59	34.41 ± 0.48	26.20 ± 0.81	28.79 ± 0.75
FedProx	71.56 ± 0.65	74.31 ± 0.49	51.89 ± 0.91	65.01 ± 0.53	44.33 ± 0.38	48.67 ± 0.47	37.79 ± 0.55	38.43 ± 0.49	33.27 ± 0.42	34.54 ± 0.51	24.15 ± 0.88	30.23 ± 0.67
FedDyn	70.62 ± 0.58	78.25 ± 0.34	63.09 ± 0.61	69.41 ± 0.45	48.53 ± 0.41	49.42 ± 0.39	37.14 ± 0.68	41.60 ± 0.54	33.03 ± 0.48	35.69 ± 0.41	29.52 ± 0.63	32.37 ± 0.59
SCAFFOLD	71.35 ± 0.61	75.62 ± 0.42	58.51 ± 0.79	63.82 ± 0.62	46.61 ± 0.33	47.27 ± 0.44	35.28 ± 0.24	38.87 ± 0.58	31.53 ± 0.53	35.76 ± 0.39	28.35 ± 0.59	30.49 ± 0.61
FedSAM	70.15 ± 0.88	77.53 ± 0.51	62.45 ± 0.72	68.78 ± 0.59	47.95 ± 0.35	48.81 ± 0.41	36.88 ± 0.51	40.92 ± 0.47	32.21 ± 0.55	35.03 ± 0.44	28.98 ± 0.66	31.84 ± 0.52
FedDISCO	69.73 ± 0.44	76.98 ± 0.63	61.82 ± 0.68	68.11 ± 0.65	47.31 ± 0.52	48.15 ± 0.58	36.17 ± 0.67	40.25 ± 0.61	31.78 ± 0.61	34.68 ± 0.52	28.51 ± 0.71	31.33 ± 0.64
FedFed	79.12 ± 0.48	82.58 ± 0.31	75.35 ± 0.57	78.87 ± 0.43	52.27 ± 0.49	56.45 ± 0.38	45.41 ± 0.76	49.28 ± 0.52	34.99 ± 0.45	36.49 ± 0.37	33.28 ± 0.58	34.89 ± 0.47
FedFTG	75.21 ± 0.53	78.44 ± 0.39	70.84 ± 0.31	74.85 ± 0.41	44.91 ± 0.62	54.15 ± 0.45	40.43 ± 0.39	46.47 ± 0.48	32.40 ± 0.57	33.79 ± 0.49	30.53 ± 0.51	32.26 ± 0.55
FedGen	68.34 ± 0.68	74.65 ± 0.47	58.90 ± 0.75	63.20 ± 0.66	44.29 ± 0.55	50.12 ± 0.51	36.70 ± 0.62	39.82 ± 0.57	34.17 ± 0.44	36.33 ± 0.42	29.45 ± 0.68	31.60 ± 0.61
FedAux	59.72 ± 0.85	73.70 ± 0.51	53.13 ± 0.88	65.11 ± 0.59	44.05 ± 0.48	44.85 ± 0.53	36.54 ± 0.65	38.39 ± 0.61	27.83 ± 0.72	34.31 ± 0.49	24.75 ± 0.82	30.38 ± 0.69
FedDF	60.56 ± 0.77	72.40 ± 0.58	59.20 ± 0.72	63.47 ± 0.64	30.13 ± 0.81	45.47 ± 0.50	34.46 ± 0.69	36.77 ± 0.64	29.07 ± 0.65	34.75 ± 0.46	28.42 ± 0.62	30.51 ± 0.63
FedXLRP	81.72 ± 0.35	83.46 ± 0.28	77.02 ± 0.63	80.27 ± 0.39	52.07 ± 0.45	58.09 ± 0.33	46.25 ± 0.82	52.63 ± 0.41	36.85 ± 0.39	38.64 ± 0.32	34.64 ± 0.52	33.18 ± 0.51
FedXIG	72.22 ± 0.59	75.89 ± 0.44	64.02 ± 0.69	68.61 ± 0.55	49.21 ± 0.48	55.46 ± 0.41	42.34 ± 0.58	47.89 ± 0.50	33.59 ± 0.51	35.30 ± 0.45	29.77 ± 0.64	31.93 ± 0.58
FedXGrad	71.38 ± 0.62	73.99 ± 0.52	64.52 ± 0.65	66.49 ± 0.61	50.13 ± 0.43	54.21 ± 0.46	42.11 ± 0.60	46.98 ± 0.53	33.18 ± 0.54	34.40 ± 0.48	30.00 ± 0.61	30.93 ± 0.60
FedXSG	71.89 ± 0.60	74.63 ± 0.49	63.75 ± 0.71	67.80 ± 0.58	51.57 ± 0.39	55.22 ± 0.42	40.91 ± 0.63	47.61 ± 0.51	33.43 ± 0.52	34.74 ± 0.47	29.64 ± 0.65	31.53 ± 0.59

Table 2. Communication efficiency comparison across different federated learning methods. Results show the number of communication rounds (mean ± std over 5 runs) needed to achieve target accuracy thresholds (70% and 60% for CIFAR-10; 40% and 30% for CIFAR-100; 35% and 30% for Tiny-ImageNet) with different numbers of clients (K=10 and K=100) and heterogeneity parameter $\alpha = 0.1$. **Lower** values indicate better communication efficiency. Best results are in bold.

Method	CIFAR-10		CIFAR-100		Tiny-ImageNet	
	<i>acc=70%</i>	<i>acc=60%</i>	<i>acc=40%</i>	<i>acc=30%</i>	<i>acc=35%</i>	<i>acc=30%</i>
	K=10	K=100	K=10	K=100	K=10	K=100
FedAvg	49 ± 5	79 ± 7	24 ± 4	34 ± 5	60 ± 6	90 ± 8
FedProx	32 ± 3	55 ± 4	28 ± 4	32 ± 4	45 ± 4	65 ± 5
FedDyn	29 ± 3	51 ± 4	24 ± 3	32 ± 3	48 ± 5	60 ± 4
SCAFFOLD	33 ± 4	52 ± 5	25 ± 3	38 ± 4	46 ± 4	69 ± 6
FedSAM	34 ± 4	58 ± 5	27 ± 4	35 ± 4	50 ± 5	64 ± 5
FedDISCO	36 ± 3	61 ± 6	28 ± 4	37 ± 5	52 ± 5	67 ± 6
FedFed	15 ± 2	22 ± 4	13 ± 4	12 ± 3	12 ± 2	15 ± 2
FedFTG	28 ± 3	57 ± 4	22 ± 3	28 ± 3	30 ± 3	40 ± 4
FedGen	51 ± 5	66 ± 6	21 ± 4	39 ± 5	55 ± 5	70 ± 6
FedAux	38 ± 4	45 ± 4	19 ± 3	36 ± 4	42 ± 4	50 ± 5
FedDF	84 ± 7	90 ± 8	28 ± 4	45 ± 5	100 ± 8	110 ± 9
FedXLRP	14 ± 2	23 ± 3	11 ± 3	13 ± 2	10 ± 4	12 ± 2
FedXIG	35 ± 4	39 ± 4	16 ± 2	21 ± 3	38 ± 4	45 ± 4
FedXGrad	41 ± 4	46 ± 5	13 ± 2	26 ± 3	44 ± 4	50 ± 5
FedXSG	41 ± 5	51 ± 4	21 ± 3	33 ± 4	42 ± 4	48 ± 5

Additional implementation details can be found in the supplementary material. The results in Table 3 show a clear trend where FedXDS consistently achieves the highest accuracy across both datasets, demonstrating its effectiveness in handling distribution shift. Notably, methods incorporating additional data generation or augmentation, such as FedFTG and FedGen, also perform well, but FedXDS surpasses them, suggesting that relevance-guided data sharing provides a stronger mechanism for improving generaliza-

tion in heterogeneous federated settings.

The results in Table 4 demonstrate that integrating FedXDS with FL methods consistently improves the performance. In all cases, the addition of FedXDS yields substantial accuracy gains, particularly in the highly heterogeneous setting with K=100 clients. For instance, FedAvg, which traditionally suffers from performance degradation in non-IID scenarios, improves from 60.94% to 80.29% when combined with FedXDS. Similar improvements are observed for

Table 3. Accuracy comparison (mean \pm std % over 5 runs) on CelebA and FEMNIST. Best results are in bold. We use the LRP variant of FedXDS.

Method	CelebA	FEMNIST
FedAvg	87.23 \pm 0.71	84.71 \pm 0.68
FedProx	87.79 \pm 0.65	85.66 \pm 0.61
SCAFFOLD	86.36 \pm 0.56	84.24 \pm 0.59
FedDyn	88.13 \pm 0.52	86.45 \pm 0.34
FedSAM	90.03 \pm 0.15	87.56 \pm 0.44
FedDISCO	89.78 \pm 0.61	87.29 \pm 0.62
FedFTG	90.31 \pm 0.43	87.88 \pm 0.67
FedGen	89.65 \pm 0.59	86.37 \pm 0.55
FedDF	88.56 \pm 0.63	85.28 \pm 0.66
FedAux	89.27 \pm 0.60	85.94 \pm 0.62
FedFed	90.76 \pm 0.83	88.28 \pm 0.71
FedXDS	91.55 \pm 0.48	89.03 \pm 0.35

Table 4. Performance comparison (mean \pm std % over 5 runs) on CIFAR-10 for different base FL methods and their FedXDS (using LRP) variants under two heterogeneity settings ($K = 10$ and $K = 100$). The FedXDS variants consistently improve performance.

Method	$K = 10$	$K = 100$
FedAvg	72.90 \pm 0.55	60.94 \pm 0.68
FedAvg + FedXDS	83.46 \pm 0.31	80.29 \pm 0.42
FedProx	74.31 \pm 0.49	65.01 \pm 0.53
FedProx + FedXDS	84.05 \pm 0.35	80.73 \pm 0.45
FedDyn	70.62 \pm 0.58	69.41 \pm 0.45
FedDyn + FedXDS	84.20 \pm 0.33	79.56 \pm 0.48
SCAFFOLD	71.35 \pm 0.61	63.82 \pm 0.62
SCAFFOLD + FedXDS	83.89 \pm 0.38	79.72 \pm 0.44

FedProx, FedDyn, and SCAFFOLD, with FedXDS enhancing their robustness to data heterogeneity while maintaining their inherent advantages.

The magnitude of improvement suggests that FedXDS effectively mitigates the adverse effects of statistical heterogeneity by facilitating better knowledge transfer across clients. Notably, FedDyn, which incorporates regularization-based local adaptation, exhibits the smallest absolute improvement with FedXDS, indicating that its existing adaptation mechanisms already partially address heterogeneity. In contrast, methods like FedAvg, without correction for client drift, benefit more from FedXDS.

6. Empirical Privacy Analysis

6.1. Membership Inference Attack

We conduct membership inference attacks following [30] to evaluate our methodology’s privacy guarantees. We investigate whether training on masked and noised features reveals

Table 5. Test accuracy (mean \pm std % over 5 runs) for different sparsification levels s across various attribution methods. Best results are in bold. Results for CIFAR10 at $K = 10$, $\alpha = 0.1$.

s	Gradient	SmoothGrad	Int. Grad.	LRP
60	75.99 \pm 0.52	75.63 \pm 0.55	77.10 \pm 0.48	84.74 \pm 0.31
65	74.19 \pm 0.56	73.15 \pm 0.61	76.31 \pm 0.51	84.01 \pm 0.35
70	73.99 \pm 0.59	74.63 \pm 0.54	75.89 \pm 0.49	83.46 \pm 0.28
75	70.54 \pm 0.65	70.67 \pm 0.63	71.68 \pm 0.58	83.01 \pm 0.33
80	68.31 \pm 0.71	70.45 \pm 0.66	70.98 \pm 0.62	82.67 \pm 0.36
85	68.80 \pm 0.68	68.13 \pm 0.73	69.73 \pm 0.67	81.88 \pm 0.41

membership in the underlying dataset by using the globally DP data to train a shadow model, then testing if clients can infer membership by querying this model and training a random forest classifier. Figure 2a shows recall values for different privacy parameters ϵ . Results demonstrate that non-masked features, despite DP protection, offer inferior privacy protection compared to masked data. This can be explained by the fact that sparsity significantly mitigates attacks through norm reduction from attribution masks (Equation 8).

6.2. Feature Inversion Attack

While metric private features prevent direct inference of individual data features, a sophisticated adversary might train a denoising autoencoder to learn mappings between protected representations and original images. We conduct feature inversion attacks and report the SSIM score between original and reconstructed images. Figure 2b reveals inversion attack success increases with lower sparsity and higher privacy budgets. Unmasked features consistently exhibit the highest SSIM, confirming that raw features under identical DP constraints provide inferior protection compared to our approach. Additional validations in supplementary material, including *visualization* of inversion results for non-masked features and FedFed [30] features, confirming both cases’ vulnerability to reconstruction, whereas our method provides superior protection.

7. Ablation Studies and Discussion

7.1. XAI Method Comparison Across Sparsification

Table 5 demonstrates that FedXDS maintains stable performance across different sparsification levels, with all attribution methods achieving above 68% accuracy, even at high sparsity thresholds. Notably, LRP consistently outperforms other methods, preserving accuracies between 81.88% and 84.74%, while others degrade more sharply under extreme sparsification. LRP’s robustness stems from its layer-wise relevance propagation mechanism, which is fundamentally different from gradient-based approaches. Instead of measuring local sensitivity, LRP redistributes the

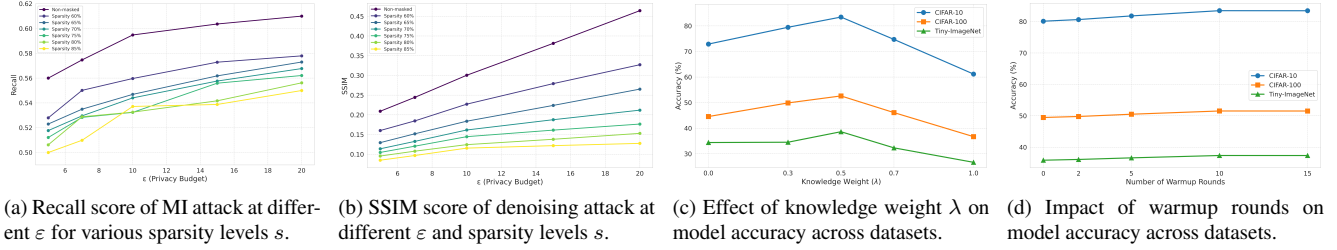


Figure 2. Evaluation of privacy-utility trade-offs and hyperparameter effects in our framework.

model’s output prediction backward through the network, layer by layer, conserving relevance at each step. This process aggregates relevance into contiguous, semantically coherent regions that represent structured object parts. In stark contrast, methods like SmoothGrad or Integrated Gradients identify ‘hotspot’ pixels that are locally influential but often disconnected from each other. Consequently, under high sparsification, LRP retains meaningful visual patterns, whereas other methods are left with a sparse collection of isolated points that lack sufficient context for accurate classification. We conclude that *structural coherence* in saliency maps is an essential criterion when selecting the attribution method. This finding is corroborated by current literature [2, 13, 23], which confirms LRP’s superior explanation quality and reinforces that our work is consistent with established findings on the importance of explanation quality for downstream tasks. More discussion can be found in the supplementary material.

7.2. Privacy vs. Utility

We investigate how the privacy budget ϵ impacts model performance across attribution methods. Table 6 shows that increasing ϵ improves accuracy for all methods, demonstrating the expected privacy-utility trade-off: stronger privacy (lower ϵ) leads to reduced performance due to greater noise. Among all methods, LRP consistently achieves the highest accuracy across privacy levels outperforming the next best (Integrated Gradients) by 7-8 % and maintaining robustness even under strict privacy constraints. SmoothGrad shows modest improvements over basic gradients, though less pronounced than Integrated Gradients. These results underscore LRP’s ability to extract stable, task-relevant features leading to better generalization with less privacy leakage, aligning with prior work on explanation fidelity [3, 22].

7.3. Ablation Study on the Knowledge Weight

We evaluate how the knowledge weight λ , as defined in Equation 11, affects performance across CIFAR-10, CIFAR-100, and Tiny-ImageNet. Figure 2c shows that intermediate values ($\lambda = 0.5$) consistently outperform both extremes, with CIFAR-10 accuracy peaking at 83.46% compared to 72.90% ($\lambda = 0.0$, equivalent to FedAvg) and

Table 6. Ablation study showing test accuracy () for different privacy budgets (ϵ) across XAI-based federated learning methods. Results for CIFAR10 at $K = 10$, $\alpha = 0.1$.

ϵ	Gradient	SmoothGrad	Int. Grad.	LRP
5	70.55 \pm 0.62	71.23 \pm 0.59	72.45 \pm 0.52	80.06 \pm 0.33
7	71.45 \pm 0.58	72.15 \pm 0.55	73.35 \pm 0.47	80.89 \pm 0.31
10	72.25 \pm 0.51	72.95 \pm 0.48	74.15 \pm 0.41	81.65 \pm 0.29
15	73.12 \pm 0.45	73.85 \pm 0.43	75.02 \pm 0.38	82.11 \pm 0.25
20	73.99 \pm 0.42	74.63 \pm 0.39	75.89 \pm 0.35	83.46 \pm 0.28

61.18% ($\lambda = 1.0$, synthetic data equal weight). Our ablation thus shows that balancing the contribution of synthetic data is critical for performance.

7.4. Warmup and Efficiency

Figure 2d shows accuracy slightly improves with more warmup rounds, plateauing at 10, suggesting this is the optimal trade-off between convergence and communication cost. Our approach is also computationally efficient. Attribution masks are computed in a single backward pass during warmup, incurring minimal overhead. In contrast, methods like FedGen and FedFTG retrain generators every round, and FedFed requires costly VAE training—making FedXDS more suitable for resource-constrained settings. We provide more discussion in the supplementary material.

8. Conclusion

In this work, we introduced FedXDS, leveraging attribution methods to address statistical heterogeneity, privacy, and computational efficiency in federated learning. By using attribution maps to extract and share privacy-preserved task-relevant features, our approach enables effective knowledge transfer while ensuring strong privacy guarantees. Experiments show that FedXDS consistently outperforms existing methods in accuracy and communication efficiency. Moreover, the sparsity induced by relevance attribution enhances privacy, mitigating feature inversion and membership inference risks. These findings highlight attribution methods as a promising tool for improving federated learning while preserving privacy and efficiency.

9. Acknowledgements

This work was supported by the European Union’s Horizon Europe research and innovation programme (EU Horizon Europe) as grant ACHILLES (101189689).

References

- [1] Durmus Alp Emre Acar, Yue Zhao, Ramon Matas, Matthew Mattina, Paul Whatmough, and Venkatesh Saligrama. Federated learning based on dynamic regularization. In *International Conference on Learning Representations*, 2021. 3
- [2] Leila Arras, Ahmed Osman, and Wojciech Samek. Clevrxai: A benchmark dataset for the ground truth evaluation of neural network explanations. *Information Fusion*, 81:14–40, 2022. 1, 8
- [3] Sebastian Bach, Alexander Binder, Grégoire Montavon, Frederick Klauschen, Klaus-Robert Müller, and Wojciech Samek. On pixel-wise explanations for non-linear classifier decisions by layer-wise relevance propagation. *PLOS ONE*, 10(7):1–46, 2015. 3, 5, 8
- [4] Jeremiah Blocki, Avrim Blum, Anupam Datta, and Or Sheffet. Differentially private data analysis of social networks via restricted sensitivity. In *Proceedings of the 4th conference on Innovations in Theoretical Computer Science*, pages 87–96, 2013. 2
- [5] Sebastian Caldas, Sai Meher Karthik Duddu, Peter Wu, Tian Li, Jakub Konečný, H Brendan McMahan, Virginia Smith, and Ameet Talwalkar. Leaf: A benchmark for federated settings. *arXiv preprint arXiv:1812.01097*, 2018. 5
- [6] Konstantinos Chatzikokolakis, Miguel E Andrés, Nicolás Emilio Bordenabe, and Catuscia Palamidessi. Broadening the scope of differential privacy using metrics. In *Privacy Enhancing Technologies: 13th International Symposium, PETS 2013, Bloomington, IN, USA, July 10-12, 2013. Proceedings 13*, pages 82–102. Springer, 2013. 2
- [7] Daoyuan Chen, Dawei Gao, Weirui Kuang, Yaliang Li, and Bolin Ding. pfl-bench: A comprehensive benchmark for personalized federated learning. *Advances in Neural Information Processing Systems*, 35:9344–9360, 2022. 5
- [8] Rudresh Dwivedi, Devam Dave, Het Naik, Smiti Singhal, Rana Omer, Pankesh Patel, Bin Qian, Zhenyu Wen, Tejal Shah, Graham Morgan, et al. Explainable ai (xai): Core ideas, techniques, and solutions. *ACM Computing Surveys*, 55(9):1–33, 2023. 1
- [9] Cynthia Dwork. Differential privacy. In *International colloquium on automata, languages, and programming*, pages 1–12. Springer, 2006. 1, 2
- [10] Oluwaseyi Feyisetan and Shiva Kasiviswanathan. Private release of text embedding vectors. In *Proceedings of the First Workshop on Trustworthy Natural Language Processing*, pages 15–27, 2021. 2
- [11] Peter Kairouz, H Brendan McMahan, Brendan Avent, Aurélien Bellet, Mehdi Bennis, Arjun Nitin Bhagoji, Kallista Bonawitz, Zachary Charles, Graham Cormode, Rachel Cummings, et al. Advances and open problems in federated learning. *Foundations and trends® in machine learning*, 14(1–2): 1–210, 2021. 1
- [12] Sai Praneeth Karimireddy, Satyen Kale, Mehryar Mohri, Sashank Reddi, Sebastian Stich, and Ananda Theertha Suresh. SCAFFOLD: Stochastic controlled averaging for federated learning. In *Proceedings of the 37th International Conference on Machine Learning*, pages 5132–5143. PMLR, 2020. 1, 3
- [13] Sebastian Lapuschkin, Stephan Wäldchen, Alexander Binder, Grégoire Montavon, Wojciech Samek, and Klaus-Robert Müller. Unmasking clever hans predictors and assessing what machines really learn. *Nature communications*, 10(1):1096, 2019. 8
- [14] Bo Li, Mikkel N Schmidt, Tommy S Alstrøm, and Sebastian U Stich. On the effectiveness of partial variance reduction in federated learning with heterogeneous data. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3964–3973, 2023. 1
- [15] Qinbin Li, Bingsheng He, and Dawn Song. Model-contrastive federated learning. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 10713–10722, 2021. 3
- [16] 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. 1, 3
- [17] Xiaoxiao Li, Meirui Jiang, Xiaofei Zhang, Michael Kamp, and Qi Dou. Fed{bn}: Federated learning on non-{iid} features via local batch normalization. In *International Conference on Learning Representations*, 2021. 3
- [18] Tao Lin, Lingjing Kong, Sebastian U Stich, and Martin Jaggi. Ensemble distillation for robust model fusion in federated learning. *Advances in neural information processing systems*, 33:2351–2363, 2020. 3
- [19] H. Brendan McMahan, Eider Moore, Daniel Ramage, Seth Hampson, and Blaise Agüera y Arcas. Communication-efficient learning of deep networks from decentralized data. In *Proceedings of the 20th International Conference on Artificial Intelligence and Statistics (AISTATS)*. PMLR, 2017. 1, 2, 3
- [20] Matias Mendieta, Taojiannan Yang, Pu Wang, Minwoo Lee, Zhengming Ding, and Chen Chen. Local learning matters: Rethinking data heterogeneity in federated learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 8397–8406, 2022. 1
- [21] Zhe Qu, Xingyu Li, Rui Duan, Yao Liu, Bo Tang, and Zhuo Lu. Generalized federated learning via sharpness aware minimization. In *International conference on machine learning*, pages 18250–18280. PMLR, 2022. 1, 3
- [22] Wojciech Samek, Alexander Binder, Grégoire Montavon, Sebastian Lapuschkin, and Klaus-Robert Müller. Evaluating the visualization of what a deep neural network has learned. *IEEE Transactions on Neural Networks and Learning Systems*, 28(11):2660–2673, 2017. 8
- [23] Wojciech Samek, Grégoire Montavon, Sebastian Lapuschkin, Christopher J. Anders, and Klaus-Robert Müller. Explaining deep neural networks and beyond: A review of methods and applications. *Proceedings of the IEEE*, 109(3): 247–278, 2021. 8

- [24] Felix Sattler, Tim Korjakow, Roman Rischke, and Wojciech Samek. Fedaux: Leveraging unlabeled auxiliary data in federated learning. *IEEE Transactions on Neural Networks and Learning Systems*, 34(9):5531–5543, 2021. 3
- [25] Karen Simonyan, Andrea Vedaldi, and Andrew Zisserman. Deep inside convolutional networks: Visualising image classification models and saliency maps. In *International Conference on Learning Representations (ICLR)*, 2014. 3
- [26] Abhishek Singh, Praneeth Vepakomma, Vivek Sharma, and Ramesh Raskar. Posthoc privacy guarantees for collaborative inference with modified propose-test-release. *Advances in Neural Information Processing Systems*, 36:26438–26451, 2023. 2
- [27] Daniel Smilkov, Nikhil Thorat, Been Kim, Fernanda Viégas, and Martin Wattenberg. Smoothgrad: removing noise by adding noise. *arXiv preprint arXiv:1706.03825*, 2017. 3, 5
- [28] Mukund Sundararajan, Ankur Taly, and Qiqi Yan. Axiomatic attribution for deep networks. In *International Conference on Machine Learning*, pages 3319–3328. PMLR, 2017. 3, 5
- [29] Jianyu Wang, Qinghua Liu, Hao Liang, Gauri Joshi, and H. Vincent Poor. Tackling the objective inconsistency problem in heterogeneous federated optimization. In *Proceedings of the 34th International Conference on Neural Information Processing Systems*, Red Hook, NY, USA, 2020. Curran Associates Inc. 3
- [30] Zhiqin Yang, Yonggang Zhang, Yu Zheng, Xinmei Tian, Hao Peng, Tongliang Liu, and Bo Han. Fedfed: Feature distillation against data heterogeneity in federated learning. *Advances in Neural Information Processing Systems*, 36, 2024. 1, 3, 7
- [31] Mang Ye, Xiuwen Fang, Bo Du, Pong C Yuen, and Dacheng Tao. Heterogeneous federated learning: State-of-the-art and research challenges. *ACM Computing Surveys*, 56(3):1–44, 2023. 1
- [32] Rui Ye, Mingkai Xu, Jianyu Wang, Chenxin Xu, Siheng Chen, and Yanfeng Wang. Feddisco: Federated learning with discrepancy-aware collaboration. In *International Conference on Machine Learning*, pages 39879–39902. PMLR, 2023. 3
- [33] Lin Zhang, Li Shen, Liang Ding, Dacheng Tao, and Ling-Yu Duan. Fine-tuning global model via data-free knowledge distillation for non-iid federated learning. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 10174–10183, 2022. 1, 3
- [34] Yue Zhao, Meng Li, Liangzhen Lai, Naveen Suda, Damon Civin, and Vikas Chandra. Federated learning with non-iid data. *arXiv preprint arXiv:1806.00582*, 2018. 1, 4
- [35] Yue Zhao, Meng Li, Liangzhen Lai, Naveen Suda, Damon Civin, and Vikas Chandra. Federated learning with non-iid data. *arXiv preprint arXiv:1806.00582*, 2018. 1
- [36] Zhuangdi Zhu, Junyuan Hong, and Jiayu Zhou. Data-free knowledge distillation for heterogeneous federated learning. In *Proceedings of the 38th International Conference on Machine Learning*, pages 12878–12889. PMLR, 2021. 1, 3, 5