Journal of open research software

ProFed: A Benchmark for Proximity-Based Non-IID Federated Learning

SOFTWARE METAPAPER

DAVIDE DOMINI
CHRISTIAN OTTE INGEMANN
GIANLUCA AGUZZI
LUKAS ESTERLE
MIRKO VIROLI

*Author affiliations can be found in the back matter of this article

ubiquity press

ABSTRACT

Federated Learning (FL) has emerged as a key paradigm in machine learning but its performance often deteriorates under non-independent and identically distributed (non-IID) client data. Such heterogeneity frequently reflects geographic factors—for example, regional linguistic variations or localized traffic patterns—leading to IID data within regions but with non-IID distributions across them. However, existing FL algorithms are typically evaluated by randomly splitting non-IID data across devices, disregarding their spatial distribution.

To address this gap, we introduce ProFed, a benchmark that simulates data splits with varying degrees of skewness across different regions. We incorporate several skewness methods from the literature and apply them to well-known datasets, including MNIST, FashionMNIST, Extended MNIST, CIFAR-10, CIFAR-100, and UTKFace. Our goal is to provide researchers with a standardized framework to evaluate FL algorithms more effectively and consistently against established baselines.

CORRESPONDING AUTHOR:
Davide Domini
Department of Computer Science and Engineering, University of Bologna, Cesena, Italy
davide.domini@unibo.it

KEYWORDS:
Federated Learning; benchmarks; performance evaluation; non-IID data

TO CITE THIS ARTICLE:
Domini D, Otte Ingemann C, Aguzzi G, Esterle L, Viroli M. 2026 ProFed: A Benchmark for Proximity-Based Non-IID Federated Learning. *Journal of Open Research Software*, 14: 13. DOI: https://doi.org/10.5334/jors.624

(1) OVERVIEW

INTRODUCTION

Federated Learning (FL) [1] has gained significant interest in the last years. It has been introduced to address privacy problems during learning from users' data. In fact, this framework allows the training of a shared global model without the need of collecting the data on a central server.

Different studies [2, 3] have shown that while FL achieves good learning performance compared to classical learning approaches on homogeneously distributed data, it drops performance when data are *non-independently and identically distributed* (non-IID). For instance, in urban traffic prediction scenarios, data patterns exhibit strong spatial correlations: traffic flows within specific city districts often share common characteristics while differing significantly from patterns observed in other areas. This geographical dependency implies that models trained on data from one district typically achieve higher accuracy when predicting traffic patterns within the same area compared to predictions in different districts.

In the literature, various algorithms—like Scaffold [4], FedProx [5], and many others [6–8]—have been proposed to tackle data heterogeneity. These approaches typically assume that client data are distributed without considering specific patterns or structures. However, in real world scenarios, particularly highly distributed systems [9] (e.g., in edge computing or spatial-aware scenarios), it is common for data from geographically close devices to be more similar to each other compared to data from devices farther apart (Figure 1). This phenomenon is driven by the fact that devices in the same region often experience similar environments and make comparable observations [10, 11]. Several studies (e.g., [12–14, 34]) have attempted to tackle this scenario by proposing algorithms that cluster clients based on similarity metrics, under the assumption that clients within the same cluster have IID data while clusters themselves exhibit non-IID properties. Nevertheless, the lack of standardized benchmarks for evaluating such approaches remains limited. Existing benchmarks [15, 16] often rely on synthetic data splits or arbitrary partitioning schemes that fail to capture the realistic geographic clustering observed in practice.

To bridge this gap, we introduce ProFed,[1] a novel benchmark designed specifically for proximity-based non-IID FL providing a more realistic and complete evaluation setting. ProFed leverages well-known computer vision datasets from PyTorch [17] and TorchVision [18]—like MNIST [19], CIFAR10 [20], CIFAR100 [21], and UTKFace [22]—and incorporates established data partitioning methods from the literature, such as Dirichlet distribution-based splits [23, 24]. Moreover, by enabling researchers to control the degree of data skewness, this approach allows for fine-grained experimentation and analysis. Its effectiveness is further demonstrated by its adoption in several scientific contributions (e.g., [25, 26]).

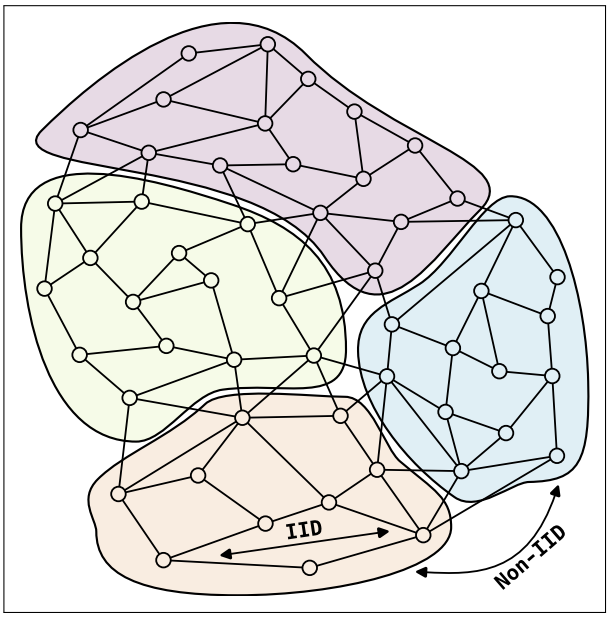

Figure 1 Spatial data distribution: homogeneous within subregions, non-IID across subregions.

RELATED SOFTWARE AND MOTIVATION

Over the years, several benchmarks have been proposed for FL, typically focusing on standard datasets split homogeneously across multiple clients, such as [27, 28]. However, recent work has shifted its focus toward addressing various data shifts. For instance, FedScale [29] offers a comprehensive platform for evaluating multiple aspects of FL at scale, including system efficiency, statistical efficiency, privacy, and security. FedScale incorporates a diverse set of realistic datasets and takes into account client resource constraints.

Similarly, LEAF [30], another relevant framework, emphasizes reproducibility through its open-source datasets, metrics, and reference implementations. LEAF provides granular metrics that assess not only model performance but also the computational and communication costs associated with training in federated settings. Additionally, LEAF supports multiple configurations, enabling users to explore different facets of FL.

Motivation

Despite proposed benchmarks already being valuable resources for the research community, they do not consider one aspect that is crucial in real-world scenarios: *the spatial distribution of devices*. In fact, in many applications, devices are geographically distributed, and the collected data is often correlated with their location. This is where ProFed comes into play, providing a benchmark that simulates data splits with varying

degrees of skewness across different regions, enabling researchers to evaluate FL algorithms in a more realistic and complete setting.

IMPLEMENTATION AND ARCHITECTURE

Before delving into the implementation details of ProFed, we first formalize the considered scenario—this will help the reader to better understand some design choices. As depicted in Figure 1, we consider a spatial area $A = \{a_1, \dots, a_k\}$ divided into k distinct contiguous subregions. Each subregion a_j has a unique data distribution Θ_j and provides specific localized information. This means that, given two regions i, j and the respective data distributions Θ_i and Θ_j, a sample d' from Θ_i is distinctively dissimilar from a sample d'' from Θ_j (namely, the data is non-IID). Whereas, giving two data distributions Θ_i and Θ_j from the same region i, d' and d'' sample from same Θ_i, their difference $m(d', d'')$ is negligible (namely, the data is homogeneous).

This dissimilarity can be quantified using a specific distance metric $m(d', d'')$, which determines the disparity between two distributions. Formally, given an error bound δ, the dissimilarity intra-region and inter-region can be quantified as follows:

$$\forall i \neq j, \forall d, d' \in \Theta_i, \forall d'' \in \Theta_j : m(d, d') \leq \delta < m(d, d'') \quad (1)$$

In A, a set of *sensor nodes* $S = \{s_1, \dots, s_n\}$ ($n \gg |A|$) are deployed—for instance, these sensor nodes may be smartphones or cameras in cars. Each sensor node is assumed to be capable of processing data and to have enough computational power to be able to participate in the FL process. Locally, each node i creates a dataset D_i of samples perceived from the data distribution Θ_j of its respective region j. In this work, we consider a general classification task where each sample d in the data distribution Θ_j consists of a feature vector x and a label y. Therefore, the complete local dataset D_i is represented as $D_i = \{(x_1, y_1), \dots, (x_m, y_m)\}$.

Implementation Details

ProFed implementation is based on PyTorch [17] and TorchVision [18], as it has been specifically designed to facilitate and standardize research experiments within the scenario described above. ProFed provides an API to partition the supported datasets and to generate experimental scenarios that follow the proposed system model. In particular, given the number of regions and the number of devices per region, it enables the creation of region-aware data partitions such that devices belonging to the same region receive datasets sampled from the same underlying data distribution. This design allows the benchmark to reproduce realistic proximity-based non-IID scenarios, where data are homogeneous within regions and heterogeneous across regions, as illustrated in Figure 1. In the following, we detail the implemented methods to synthesize skewed datasets, the supported datasets and the API of the benchmark.

Data Distribution

As part of our analysis, we reviewed several studies in the literature on non-IID FL to identify and select the most commonly used partitioning methods. We observed that several works employed *Dirichlet* distribution for data partitioning. This approach results in each party having instances of most labels, although the distribution is highly imbalanced, with some labels being underrepresented and others heavily overrepresented. The degree of skewness can be adjusted using the concentration parameter α, where lower values yield more skewed distributions. In the literature, α values typically range from 0.1 to 1.0, with $\alpha = 0.5$ commonly used for moderate heterogeneity. An example of this distribution, for five subregions and the MNIST dataset (with 10 classes), is represented in Figure 2b. The second data distribution considered is *hard* partitioning, where each party has access to only a subset of labels. This creates a significantly more skewed distribution, making it considerably more challenging for learning algorithm stability. An example of this distribution is represented in Figure 2c. In ProFed, we enable fine-grained control over the data distribution, allowing either balanced label subsets across regions or customizable cardinality per subregion. For comparative analysis, we also implement an IID split as a baseline distribution (Figure 2a). While existing approaches typically apply these partitioning methods directly at the device level (e.g., [31, 23]), our framework introduces an intermediate layer of regional clustering. ProFed first distributes data heterogeneously among subregions and then splits them homogeneously among devices within the same subregion, thereby creating clustered heterogeneity—see Figure 1. To support extensibility, ProFed enables custom partitioning based on a user-specified distribution matrix. This distribution is represented as an $N \times M$ matrix where N is the number of labels and M is the number of subregions. Each cell (i, j) indicates the proportion of instances with label i assigned to subregion j.

Supported datasets

ProFed supports various TorchVision datasets widely used in computer vision research (details in Table 1). We include MNIST [19] as a baseline: grayscale 28×28 pixel images of handwritten digits across 10 classes. ProFed extends this with Fashion MNIST [32] (clothing items, 10 classes) and Extended MNIST [33] (EMNIST, Latin alphabet letters, 27 classes). For color images, ProFed includes CIFAR10 and CIFAR100 with 32×32 RGB images. CIFAR10 has 10 classes while CIFAR100 has 100, offering greater complexity. Beyond classification tasks, ProFed incorporates the UTKFace dataset [22], containing over 23,000 face images based on 200×200 pixel

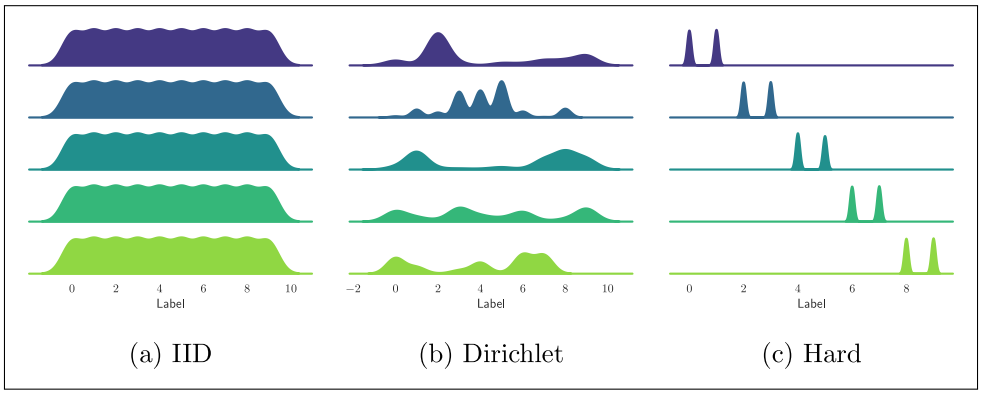

Figure 2 Data distribution patterns across five subregions: **(a)** IID data, **(b)** Dirichlet (non-IID), and **(c)** Hard (Highly non-IID). Each color represents a different subregion.

DATASET	TRAINING SIZE	TEST SIZE	FEATURES	TARGETS
MNIST	60,000	10,000	784	10
Fashion MNIST	60,000	10,000	784	10
EMNIST	124,800	20,800	784	27
CIFAR-10	50,000	10,000	3,072	10
CIFAR-100	50,000	10,000	3,072	100
UTKFace	20,150	3,557	120,000	[1; 116]

Table 1 Summary of the characteristics of the datasets included in the benchmark. The first five datasets are designed for classification tasks, with target values corresponding to discrete classes. In contrast, the last dataset is used for a regression task, where the target values span a continuous range.

RGB images, for age regression tasks. All classification datasets maintain balanced class distributions (e.g., CIFAR10 provides 6,000 training instances per class).

Benchmark API

ProFed has been designed with usability and ergonomics in mind. To achieve this, its API provides all the necessary methods to manage the referenced use case seamlessly—an example is provided in Listing 1. First, ProFed allows users to download the selected dataset directly and automatically generate training and validation subsets. Second, and most important, given a dataset and a predefined number of subregions, it enables users to distribute data among subregions following the specified distribution strategy. Finally, once the data distribution among subregions is established, ProFed facilitates the creation of datasets for individual devices. Each device-specific dataset is represented as an instance of the `Subset` class from PyTorch, ensuring full compatibility with existing learning algorithms.

QUALITY CONTROL

To evaluate ProFed's effectiveness and usability, we conducted experiments using supported datasets with three state-of-the-art algorithms: FedAvg [1], FedProx [5], and Scaffold [4]. Data were synthetically partitioned using supported methods: IID, Dirichlet ($\alpha = 0.5$), and hard partitioning. Experiments varied the number of subregions $A \in \{3, 6, 9\}$. All implementations utilized PyTorch with consistent hyperparameters across approaches. A multi-layer perceptron with 128 hidden neurons was trained for 30 global rounds. Each global round comprised two local epochs per device with

```
from ProFed.partitioner import Environment, Region, download_dataset,
    split_train_validation, partition_to_subregions

mapping_devices_area = distribute_nodes_spatially(devices, number_subregions)

train_data, test_data = download_dataset('EMNIST')
train_data, validation_data = split_train_validation(train_data, 0.8)

environment = partition_to_subregions(train_data, validation_data, 'Hard',
    number_subregions, seed)

mapping = {}

for region_id, devices in mapping_devices_area.items():
    mapping_devices_data = environment.from_subregion_to_devices(
        region_id,
        len(devices))
    for device_index, data in mapping_devices_data.items():
        device_id = devices[device_index]
        mapping[device_id] = data
```

Listing 1 An example of how ProFed is used to partition the EMNIST dataset among devices.

batch size 32, ADAM optimizer (learning rate 10^{-3}, weight decay 10^{-4}). Experiments were repeated with five random seeds for statistical robustness, totaling 120 experimental configurations.

All code is publicly available under a permissive license for reproducibility purposes.[2]

Results were systematically collected during training, validation, and testing phases. We first established a baseline using homogeneous data distribution (IID) to assess model stability and accuracy. FedAvg was the only algorithm evaluated under IID conditions, whereas FedProx and Scaffold were not considered in this setting, as they are variants of FedAvg whose distinguishing mechanisms are specifically designed to address non-IID data distributions, and therefore provide meaningful differences primarily under data heterogeneity. Under IID conditions, FedAvg demonstrates stable convergence and achieves high accuracy. This stability is evident in both validation and testing phases. Validation convergence is shown in the first column of Figure 3, where accuracy increases monotonically toward the optimum. Test stability is demonstrated in Table 2, where the model maintains high accuracy with minimal variance.

The critical impact of data distribution becomes apparent when transitioning from IID to non-IID conditions. Under heterogeneous data distribution, significant performance degradation occurs. All evaluated algorithms fail to handle extreme data skewness effectively, resulting in reduced accuracy and convergence instability. This degradation is particularly pronounced under hard partitioning. The bottom row of Figure 3 illustrates hard partitioning across nine regions, where validation accuracy drops from 80% (IID) to 50%. Testing results exhibit similar trends. While FedAvg achieves stable performance above 95% under IID conditions, Dirichlet partitioning introduces substantial instability, evidenced by increased accuracy variance. Performance decline intensifies under hard partitioning, highlighting fundamental limitations of current FL approaches in handling extreme data heterogeneity. These findings indicate that state-of-the-art algorithms inadequately address spatially distributed data

ALGORITHM	IID	DIRICHLET	HARD
FedAvg	0.95 ± 0.001	0.9 ± 0.04	0.81 ± 0.01
FedProx	✗	0.886 ± 0.04	0.86 ± 0.01
Scaffold	✗	0.889 ± 0.06	0.81 ± 0.01

Table 2 Results on the test set for different algorithms with different partitioning methods.

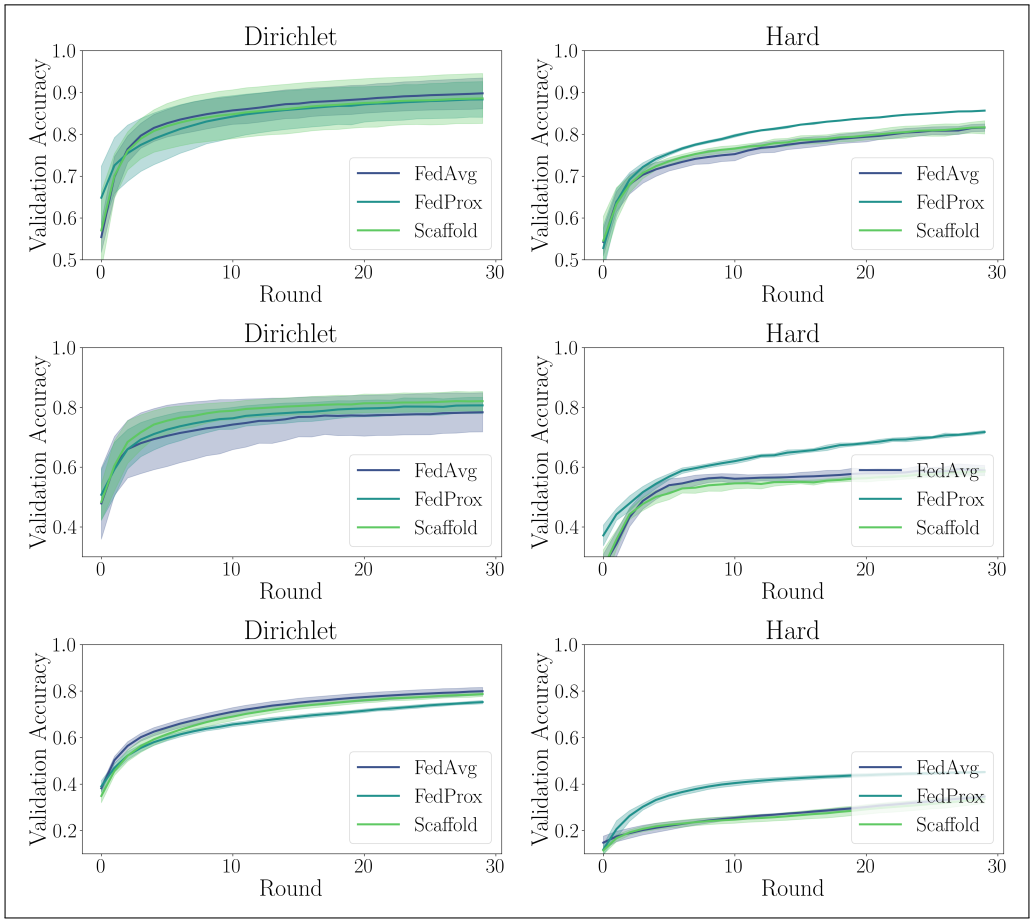

Figure 3 Validation accuracy results across MNIST, FashionMNIST, and EMNIST datasets using Dirichlet and hard partitioning methods.

challenges, necessitating further research toward more robust solutions.

CONCLUSIONS AND FUTURE WORK

In this paper, we presented ProFed, a benchmark designed to support reproducible and realistic evaluation of FL algorithms under proximity-based non-IID data distributions.

The experimental results presented in this paper are meant to demonstrate the practical usability of ProFed and illustrate the types of analyses the benchmark enables. An important direction for future work is a deeper empirical investigation of when and how region-level partitioning leads to qualitatively different learning dynamics compared to standard non-IID strategies, such as client-level Dirichlet splits. This includes studying convergence behavior, robustness to heterogeneity, and potential changes in algorithm rankings when considering a broader set of FL methods and more fine-grained diagnostic metrics. Finally, future developments of ProFed will focus on extending the benchmark with additional datasets and incorporating a broader set of baseline algorithms, including recent approaches in clustered and personalized FL, to further enhance its generality and applicability.

(2) AVAILABILITY

OPERATING SYSTEM

ProFed is platform-independent and runs on any operating system supporting Python 3.12 and above, including all recent versions of Windows, Linux, and macOS.

PROGRAMMING LANGUAGE

Python (v3.12+).

ADDITIONAL SYSTEM REQUIREMENTS

None.

DEPENDENCIES

ProFed requires torch (v2.7.0+), numpy (v2.2.2+), torchvision (v0.22.0), datasets (v3.6.0+), fsspec (v2025.3.0+), tensorflow-datasets (v4.9.9+). These dependencies are automatically installed when installing ProFed from PyPI using `pip install ProFed`.

LIST OF CONTRIBUTORS

1. Davide Domini; University of Bologna, Cesena, Italy.
2. Christian Ingemann Otte; University of Aarhus, Aarhus, Denmark.
3. Gianluca Aguzzi; University of Bologna, Cesena, Italy.
4. Lukas Esterle; University of Aarhus, Aarhus, Denmark.
5. Mirko Viroli; University of Bologna, Cesena, Italy.

SOFTWARE LOCATION

Archive

Name: Zenodo
Persistent identifier: doi: 10.5281/zenodo.16367696
Licence: MIT License
Publisher: Davide Domini
Version published: 0.7.3
Date published: 23/07/25

Code repository

Name: GitHub
Identifier: https://github.com/davidedomini/ProFed
Licence: MIT License
Date published: 23/07/25

LANGUAGE

English.

(3) REUSE POTENTIAL

Our benchmark is designed to be easily reused by other researchers to generate client datasets for FL experiments, following the various distribution strategies we provide. It can serve as a standardized tool for creating reproducible experimental setups, allowing comparisons across different studies. Furthermore, the benchmark is highly extensible: researchers can contribute by adding new datasets or implementing additional data partitioning strategies to create diverse non-IID scenarios. Contributions are welcome via pull requests on the project's public repository, and issues or questions can be raised through the repository's issue tracker. At present, support is provided on a best-effort basis through community discussion and maintainer responses, and researchers requiring further assistance are encouraged to contact the corresponding author via email.

NOTES

1 https://github.com/davidedomini/ProFed.
2 https://github.com/davidedomini/experiments-2025-jors.

FUNDING STATEMENT

This work was supported by the Italian PRIN project 'CommonWears' (2020 HCWWLP) and by the FAIR foundation, funded by the European Commission under the NextGenerationEU programme (PNRR, M4C2, Investimento 1.3, Partenariato Esteso PE00000013, Spoke 8 'Pervasive AI'). Lukas Esterle was supported by the Independent Research Fund Denmark through the FLOCKD project under the grant number 1032-00179B.

COMPETING INTERESTS

The authors have no competing interests to declare.

AUTHOR AFFILIATIONS

Davide Domini orcid.org/0009-0006-8337-8990
Department of Computer Science and Engineering, University of Bologna, Cesena, Italy

Christian Otte Ingemann
Department of Electrical and Computer Engineering, University of Aarhus, Aarhus, Denmark

Gianluca Aguzzi orcid.org/0000-0002-1553-4561
Department of Computer Science and Engineering, University of Bologna, Cesena, Italy

Lukas Esterle orcid.org/0000-0002-0248-1552
Department of Electrical and Computer Engineering, University of Aarhus, Aarhus, Denmark

Mirko Viroli orcid.org/0000-0003-2702-5702
Department of Computer Science and Engineering, University of Bologna, Cesena, Italy

REFERENCES

1. **McMahan B, Moore E, Ramage D, Hampson S, Agüera y Arcas B.** Communication-efficient learning of deep networks from decentralized data. In *Proceedings of the International Conference on Artificial Intelligence and Statistics, volume 54 of Proceedings of Machine Learning Research*. PMLR; 2017. pp. 1273–1282.
2. **Ma X, Zhu J, Lin Z, Chen S, Qin Y.** A state-of-the-art survey on solving non-iid data in federated learning. *Future Gener. Comput. Syst.* 2022;135:244–258. DOI: https://doi.org/10.1016/j.future.2022.05.003
3. **Huang Y, Chu L, Zhou Z, Wang L, Liu J, Pei J, Zhang Y.** Personalized cross-silo federated learning on non-iid data. In: *EAAI 2021*. AAAI Press; 2021. pp. 7865–7873. DOI: https://doi.org/10.1609/aaai.v35i9.16960
4. **Karimireddy SP, Kale S, Mohri M, Reddi SJ, Stich SU, Suresh AT.** SCAFFOLD: stochastic controlled averaging for federated learning. In: *ICML 2020, volume 119 of Proceedings of Machine Learning Research*. PMLR; 2020. pp. 5132–5143.
5. **Li T, Sahu AK, Zaheer M, Sanjabi M, Talwalkar A, Smith V.** Federated optimization in heterogeneous networks. In: *Proceedings of the Third Conference on Machine Learning and Systems*. Austin, TX, USA: MLSys, 2020. mlsys.org
6. **Chen X, Xiao C, Liu Y.** Confusion-resistant federated learning via diffusion-based data harmonization on non-iid data. In: Globerson A, Mackey L, Belgrave D, Fan A, Paquet U, Tomczak JM, Zhang C, editors. *Advances in Neural Information Processing Systems 38: Annual Conference on Neural Information Processing Systems 2024*. Vancouver, BC, Canada: NeurIPS; 2024. DOI: https://doi.org/10.52202/079017-4368
7. **Sattler F, Wiedemann S, Müller KR, Samek W.** Robust and communication-efficient federated learning from non-i.i.d. data. *IEEE Trans. Neural Networks Learn. Syst.* 2020;31(9):3400–3413. DOI: https://doi.org/10.1109/TNNLS.2019.2944481
8. **Domini D, Farabegoli N, Aguzzi G, Viroli M, Esterke L.** Decentralized proximity-aware clustering for collective self-federated learning. *Internet of Things* 2026;35:101841. DOI: https://doi.org/10.1016/j.iot.2025.101841
9. **Domini D.** Towards self-adaptive cooperative learning in collective systems. In: *ACSOS 2024 – Companion*. Aarhus, Denmark, September 16–20, 2024. IEEE; 2024. pp. 158–160. DOI: https://doi.org/10.1109/ACSOS-C63493.2024.00049
10. **Esterle L.** Deep learning in multiagent systems. In: *Deep Learning for Robot Perception and Cognition*. Elsevier; 2022. pp. 435–460. DOI: https://doi.org/10.1016/B978-0-32-385787-1.00022-1
11. **Malucelli N, Domini D, Aguzzi G, Viroli M.** Neighbor-based decentralized training strategies for multi-agent reinforcement learning. In: Hong J, Battiato S, Esposito C, Park JW, Przybylek A, editors. *Proceedings of the 40th ACM/SIGAPP Symposium on Applied Computing, SAC 2025, Catania International Airport*. Catania, Italy, 31 March 2025 – 4 April 2025. ACM; 2025. pp. 1250–1257. DOI: https://doi.org/10.1145/3672608.3707923
12. **Ghosh A, Chung J, Yin D, Ramchandran K.** An efficient framework for clustered federated learning. *IEEE Trans. Inf. Theory* 2022;68(12):8076–8091. DOI: https://doi.org/10.1109/TIT.2022.3192506
13. **Domini D, Aguzzi G, Farabegoli N, Viroli M, Esterle L.** Proximity-based self-federated learning. In: *ACSOS 2024*. Aarhus, Denmark, September 16–20, 2024. IEEE; 2024. pp. 139–144. DOI: https://doi.org/10.1109/ACSOS61780.2024.00033
14. **Li X, Chen X, Tang B, Wang S, Xuan Y, Zhao Z.** Unsupervised graph structure-assisted personalized federated learning. In: *Proceedings of the European Conference on Artificial Intelligence, volume 372 of Frontiers in Artificial Intelligence and Applications*. IOS Press; 2023. pp. 1430–1438. DOI: https://doi.org/10.3233/FAIA230421
15. **Li Q, Diao Y, Chen Q, He B.** Federated learning on non-iid data silos: An experimental study. In: *Proceedings of the International Conference on Data Engineering*. IEEE; 2022. pp. 965–978. DOI: https://doi.org/10.1109/ICDE53745.2022.00077
16. **Huang W, Ye M, Shi Z, Wan G, Li H, Du B, Yang Q.** Federated learning for generalization, robustness, fairness: A survey and benchmark. *IEEE Trans. Pattern Anal. Mach. Intell.* 2024;46(12):9387–9406. DOI: https://doi.org/10.1109/TPAMI.2024.3418862
17. **Ansel J,** et al. Pytorch 2: Faster machine learning through dynamic python bytecode transformation and graph compilation. In: *ASPLOS*. ACM; 2024. pp. 929–947. DOI: https://doi.org/10.1145/3620665.3640366
18. **TorchVision maintainers and contributors.** TorchVision: PyTorch's Computer Vision library, November 2016.

19. **LeCun Y, Cortes C, Burges C,** et al. Mnist handwritten digit database, 2010.
20. **Krizhevsky A, Nair V, Hinton G.** Cifar-10 (Canadian institute for advanced research).
21. **Krizhevsky A, Nair V, Hinton G.** Cifar-100 (Canadian institute for advanced research).
22. **Zhang Z, Song Y, Qi H.** Age progression/regression by conditional adversarial autoencoder. In: *2017 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017*. Honolulu, HI, USA, July 21–26, 2017. IEEE Computer Society; 2017. pp. 4352–4360. DOI: https://doi.org/10.1109/CVPR.2017.463
23. **Lin T, Kong L, Stich SU, Jaggi M.** Ensemble distillation for robust model fusion in federated learning. In: *NeurIPS 2020*; 2020.
24. **Wang J, Liu Q, Liang H, Joshi G, Poor HV.** Tackling the objective inconsistency problem in heterogeneous federated optimization. In: *NeurIPS 2020*; 2020.
25. **Domini D, Erhan L, Aguzzi G, Cavallaro L, Zenoozi AD, Liotta A, Viroli M.** Sparse self-federated learning for energy efficient cooperative intelligence in society 5.0. *CoRR*, abs/2507.07613; 2025. DOI: https://doi.org/10.1109/IJCNN64981.2025.11228400
26. **Domini D, Aguzzi G, Esterle L, Viroli M.** FBFL: A field-based coordination approach for data heterogeneity in federated learning. *CoRR*, abs/2502.08577; 2025.
27. **He C**, et al. Fedml: A research library and benchmark for federated machine learning. *CoRR*, abs/2007.13518; 2020.
28. **Beutel DJ, Topal T, Mathur A, Qiu X, Parcollet T, Lane ND.** Flower: A friendly federated learning research framework. *CoRR*, abs/2007.14390; 2020.
29. **Lai F, Dai Y, Singapuram SSV, Liu J, Zhu X, Madhyastha HV, Chowdhury M.** Fedscale: Benchmarking model and system performance of federated learning at scale. In: *ICML 2022, volume 162 of Proceedings of Machine Learning Research*. PMLR; 2022. pp. 11814–11827.
30. **Caldas S, Wu P, Li T, Konečn'y J, McMahan HB, Smith V, Talwalkar A.** LEAF: A benchmark for federated settings. *CoRR*, abs/1812.01097; 2018.
31. **Elvebakken MF, Iosifidis A, Esterle L.** Adaptive parameterization of deep learning models for federated learning. *CoRR*, abs/2302.02949; 2023.
32. **Kingma DP, Ba J.** Adam: A method for stochastic optimization. *CoRR*, abs/1412.6980; 2014.
33. **Cohen G, Afshar S, Tapson J, van Schaik A.** EMNIST: an extension of MNIST to handwritten letters. *CoRR*, abs/1702.05373; 2017. DOI: https://doi.org/10.1109/IJCNN.2017.7966217
34. **Domini D, Aguzzi G, Esterle L, Viroli M.** Field-based coordination for federated learning. In: *COORDINATION 2024, volume 14676 of Lecture Notes in Computer Science*. Springer; 2024. pp. 56–74. DOI: https://doi.org/10.1007/978-3-031-62697-5_4

TO CITE THIS ARTICLE:
Domini D, Otte Ingemann C, Aguzzi G, Esterle L, Viroli M. 2026 ProFed: A Benchmark for Proximity-Based Non-IID Federated Learning. *Journal of Open Research Software*, 14: 13. DOI: https://doi.org/10.5334/jors.624

Submitted: 08 September 2025 **Accepted:** 11 February 2026 **Published:** 02 March 2026