An Experimental Framework for Studying Non-IID Data in Federated Learning for Network Telemetry

Johny M. B. Ribeiro[1], Konstantinos Vandikas[2], Maria Valéria Marquezini[3], Christian Esteve Rothenberg[4], Rafael Pasquini[1]

[1]Faculdade de Computação (FACOM)
Universidade Federal de Uberlândia (UFU) – Uberlândia – MG – Brazil

[2]Ericsson Research – Kista – Sweden

[3]Ericsson Research Brazil – Indaiatuba – SP – Brazil

[4]Faculdade de Engenharia Elétrica e Computação (FEEC)
Universidade Estadual de Campinas (UNICAMP) – Campinas – SP – Brazil

{johny.ribeiro,rafael.pasquini}@ufu.br

{konstantinos.vandikas,maria.marquezini}@ericsson.com

chesteve@unicamp.br

Abstract. *The increasing complexity of emerging 5G and 6G network environments has intensified the need for data-driven automation under heterogeneous and dynamic conditions. Federated Learning (FL) is a promising paradigm in this context. This paper presents an experimental framework to generate realistic Non-Independent and Identically Distributed (Non-IID) datasets through controlled execution of a distributed service and telemetry collection, aiming to improve the applicability of FL in network automation. Using Apache Cassandra as a representative cloud-native application, we construct datasets exhibiting temporal and structural heterogeneity. We statistically characterize these datasets and evaluate their impact on regression models and horizontal federated learning using a Wide & Deep architecture. Results show that while horizontal federation improves generalization compared to direct cross-dataset transfer, its performance degrades under pronounced structural Non-IID conditions, highlighting both its potential and limitations.*

1. Introduction

The growing demand for real-time and data-intensive applications across domains such as finance, healthcare, autonomous driving, and the Internet of Things (IoT) has accelerated the deployment of large-scale 5G infrastructures and intensified research toward 6G networks. In this context, artificial intelligence (AI)-driven automation has become essential for managing the increasing complexity of modern network environments [Maduranga et al. 2024, Jawad et al. 2023].

Contemporary 5G and emerging 6G networks integrate computing, storage, and networking resources under highly dynamic conditions driven by traffic variability, user mobility, and service diversity. To address these challenges, AI-based solutions and network management paradigms such as Zero-Touch Network Management (ZTM) and Au-

tonomic Network Management (ANM) have been proposed as key enablers of network automation [Maduranga et al. 2024, Coronado et al. 2022].

A major obstacle to the effective application of machine learning in network automation lies in the heterogeneity of the data collected across network domains. Telemetry is gathered from disparate sources, including user devices, radio access networks (RANs), and cloud-based services, each exhibiting distinct behavioral and operational characteristics that vary across geographical regions and deployment contexts. As a result, datasets are often highly heterogeneous, leading to Non-Independent and Identically Distributed (Non-IID) data, which poses significant challenges for model training, generalization, and reuse across network locations [Lu et al. 2024, Criado et al. 2022].

Federated Learning (FL) has emerged as a promising paradigm to address data privacy, security, and heterogeneity concerns in distributed environments. Under the FL paradigm, participants collaboratively train a shared model by performing local updates on private data and exchanging only model parameters or gradients with a central aggregation server. This approach enables cooperation across organizational or geographical boundaries while preserving data locality. However, the effectiveness of FL is strongly affected by the presence of Non-IID data, which can severely impair model convergence and global performance.

In this paper, we argue that advancing FL for network automation requires experimental frameworks capable of generating realistic, controllable, and reproducible Non-IID datasets that reflect the operational behavior of modern networked services. Rather than proposing a new FL algorithm, this work focuses on the design and validation of an experimental framework that systematically induces distributional and structural heterogeneity through controlled service execution. By leveraging Apache Cassandra as a representative distributed service and combining workload-driven infrastructure dynamics with fine-grained telemetry collection, we construct datasets that capture diverse operational regimes under varying degrees of heterogeneity.

The main contributions of this paper are as follows: (i) the design of an experimental framework that integrates service execution, infrastructure monitoring, and controlled workload generation to produce realistic Non-IID datasets; (ii) a comprehensive statistical characterization of these datasets, highlighting temporal dependencies, distributional shifts, and structural heterogeneity; and (iii) an experimental evaluation that quantifies the impact of Non-IID conditions on regression models and Horizontal Federated Learning (HFL), exposing both benefits and limitations as heterogeneity increases.

The remainder of this paper is organized as follows. Section 2 reviews related work on FL, Non-IID data, and time-series modeling in distributed systems. Section 3 presents the proposed experimental framework and details the dataset generation process. Section 4 reports and discusses the experimental results, including standalone learning, cross-dataset evaluation, and HFL scenarios. Finally, Section 5 concludes the paper and outlines directions for future work.

2. Related Work

FL was introduced by [McMahan et al. 2017] as a decentralized learning paradigm in which model training is performed locally on distributed devices, while only model

updates are exchanged with a central aggregator. This approach improves data privacy and reduces communication overhead. According to the taxonomy proposed by [Yang et al. 2019], FL can be broadly classified into horizontal (HFL), vertical (VFL), and transfer-based settings (FTL). This work focuses on HFL, where participants share a common feature space but observe different data samples, enabling a controlled analysis of Non-IID effects under a uniform input representation.

A major challenge in FL arises when local datasets are Non-IID, a condition that naturally characterizes 5G and 6G network environments. Network deployments span geographically and functionally diverse contexts, leading to variations in traffic patterns, mobility, workload intensity, and environmental conditions. As a result, data distributions observed across different network locations and time periods may exhibit temporal dependence, distributional shifts, and structural heterogeneity.

To characterize these properties, several complementary statistical analyses are commonly employed in the literature. Temporal dependence is typically assessed using the Autocorrelation Function (ACF) and the Durbin–Watson test [Bruce et al. 2020], which quantify serial correlation and violations of the independence assumption in time-series data. Distributional heterogeneity over time can be evaluated through comparisons across fixed temporal windows and the Kolmogorov–Smirnov (K–S) test [Dodge 2008], which detect statistically significant shifts in empirical distributions. Non-stationary behavior is further identified through rolling statistics, where changes in first- and second-order moments indicate evolving data-generating processes and regime transitions. From a FL perspective, these effects imply that local empirical risk minimization is performed over dynamically evolving objectives, increasing client drift and degrading global model convergence during aggregation. Together, these methods provide a formal and complementary characterization of temporal and distributional Non-IID behavior in distributed and networked environments.

Given the temporal nature of network telemetry data, forecasting models often rely on lagged observations to capture time-based dependencies. Prior work has investigated different lag-selection strategies, including ACF-based heuristics, adaptive LSTM-based approaches, and representation-driven methods for identifying optimal look-back horizons [Surakhi et al. 2021, Leites et al. 2024, Tang et al. 2025]. These works show the trade-off between capturing relevant temporal patterns and controlling model complexity.

Several mitigation strategies have been proposed to address the impact of Non-IID data in FL. Data-sharing approaches distribute small representative subsets among clients to reduce model divergence [Zhao et al. 2018, Zhao et al. 2022], while clustering-based methods, such as Ensemble Federated Learning (EFL), group clients according to data similarity before aggregation [Zhao et al. 2024]. FedProx introduces a proximal regularization term in the local objective to constrain client updates and improve convergence under heterogeneous data distributions [Li et al. 2018]. A comprehensive overview of these approaches is provided by the survey in [Lu et al. 2024].

In this work, we adopt a Wide & Deep learning architecture, which combines linear memorization with non-linear generalization and has been successfully applied to regression tasks with heterogeneous features [Cheng et al. 2016, Kim et al. 2020, Pölsterl et al. 2020]. While most prior FL studies evaluate Non-IID effects using syn-

thetic datasets, this paper introduces a realistic experimental framework based on network telemetry collected from a distributed Apache Cassandra cluster. The resulting datasets capture naturally emerging temporal and structural Non-IID characteristics, enabling a systematic evaluation of HFL under realistic network conditions.

3. Dataset Generation

Apache Cassandra was selected as the target application for dataset generation because it is widely adopted in modern distributed systems, including large-scale cloud services, social networks, and data-intensive backend platforms. Beyond its popularity, Cassandra provides a service-level abstraction that inherently couples network, storage, and computational resources, allowing the joint observation of infrastructure and application behavior. By operating the service under controlled and varying load conditions, it becomes possible not only to monitor service-level KPIs, but also to deliberately drive the underlying infrastructure through different operational states. This controlled interaction between service demand and infrastructure response enables the generation of realistic datasets that capture diverse performance regimes representative of contemporary and emerging 5G and 6G environments.

The experimental infrastructure, illustrated in Figure 1, consists of an Apache Cassandra cluster deployed over a Docker Swarm environment. Five Cassandra nodes are interconnected in a ring topology, providing all database services required for data storage and retrieval. Each node is hosted on a dedicated physical or virtual machine and is co-located with monitoring containers responsible for collecting node-level telemetry and container-level telemetry, composing our infrastructure features – X.

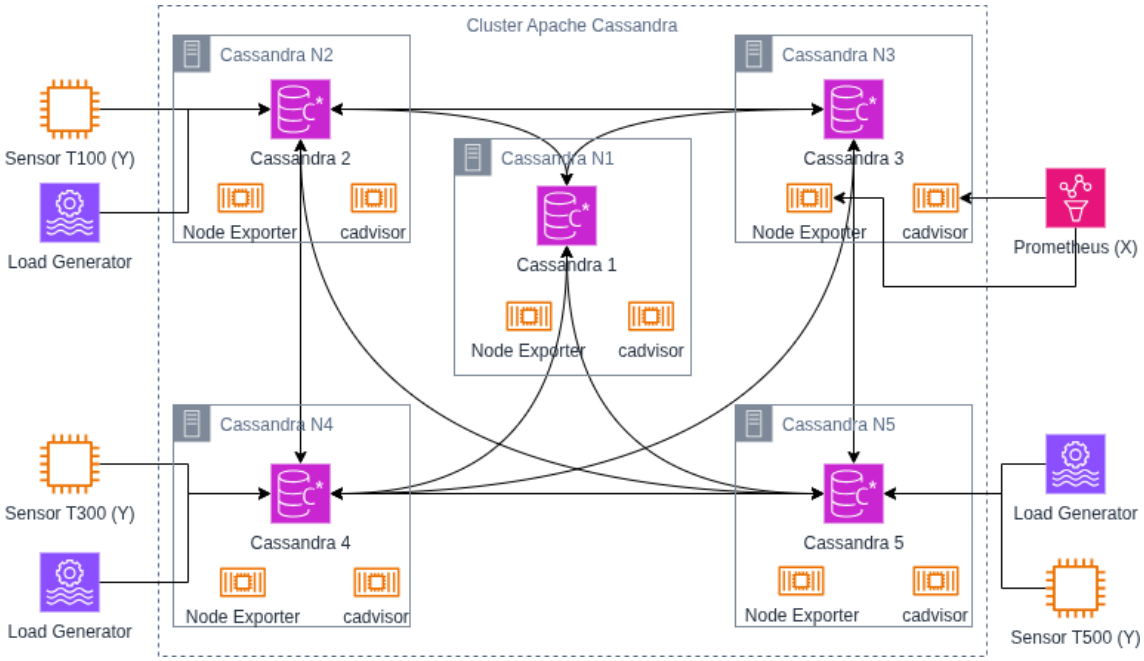

Figure 1. Apache Cassandra cluster deployed in a ring topology, monitored by Prometheus (to collect infrastructure features – X), by sensor clients (to collect service-level KPIs – Y), and stressed by a load generator that conducts the service through different states in a controlled manner.

To explicitly induce structural heterogeneity, three database schemas were defined containing 100, 300, and 500 columns, denoted as T100, T300, and T500, respectively. All columns correspond to attributes extracted from a real-world dataset originally collected for network monitoring and analysis [Stadler et al. 2017][1]. Rather than generating synthetic values, this dataset is replayed during the experiments to populate write operations and to serve as the source for read operations, preserving realistic data semantics and access patterns.

For each experiment, a subset of attributes defines the active schema, which is used consistently for both write and read operations. Increasing schema width directly affects data volume, memory footprint, serialization cost, and access latency, thereby inducing systematic changes in service behavior. This design enables a controlled investigation of how structural differences at the application layer translate into measurable distributional and temporal effects in the collected telemetry from the infrastructure.

Metrics collection is performed using Prometheus[2], configured with a one-second sampling interval for node-level and container-level telemetry, both composing the infrastructure feature set (X). Grafana[3] is integrated for real-time visualization and monitoring. For visual clarity, Prometheus is depicted in Figure 1 as being associated with Cassandra N3; however, it collects telemetry from all nodes and components of the experimental infrastructure shown in the figure. After each experimental run, a custom extraction script retrieves the collected metrics from Prometheus and stores them in CSV format, forming the feature datasets used for machine learning.

In total, Prometheus collects 2,155 telemetry features from the experimental environment, which consists of five Apache Cassandra nodes. Each node contributes 431 features, corresponding to an identical set of infrastructure and container-level metrics. Although the data stored in Cassandra are replicated across all nodes, the underlying infrastructure resources and service configurations were deliberately kept homogeneous. This design choice ensures that each node represents a standardized operational unit, enabling horizontal federation under a shared feature space. In this sense, each node can be interpreted as a replicated site within a larger networked system, such as standardized telecom sites deployed across different locations within a metropolitan area, each exposed to distinct workload dynamics.

While the proposed framework naturally allows federating all five nodes, this study focuses on three representative entry points—Cassandra N2, Cassandra N4 and Cassandra N5—corresponding, respectively, to the T100, T300, and T500 schema configurations. These nodes constitute the effective participants in the federated learning experiments presented in this paper, allowing us to isolate and analyze the impact of increasing structural heterogeneity while preserving a strictly HFL setting.

Two distinct client components interact with the Cassandra cluster during the experiments: a *load generator* and a *sensor client*. The load generator is responsible for stressing the environment. It follows a sinusoidal traffic pattern that injects varying levels of load into the Cassandra cluster, ranging from light to heavy utilization. By modulating

[1]https://github.com/rafaelpasquini/traces-jnsm-2017
[2]https://prometheus.io/
[3]https://grafana.com/

the number of concurrent clients over time, the load generator induces controlled oscillations in response times, enabling the dataset to capture diverse operational regimes that naturally emerge in real-world deployments.

The sensor client executes a constant workload of 100 Transactions Per Second (TPS), composed of 70% write and 30% read operations. In addition to issuing requests, it records service-level latency metrics and computes the statistical structures used to derive the target variable (Y). Specifically, the sensor client collects key performance indicators separately for read and write operations, reflecting their distinct execution paths and complexity within the database system. The collected metrics include minimum, maximum, mean, and standard deviation of latency, as well as the 75th, 95th, 98th, 99th, and 99.9th percentiles. These statistics are computed using three labeling strategies:

- **Cumulative**: a monotonically growing structure that stores all observed latency samples since the beginning of the experiment;
- **Drained**: a bounded structure that accumulates samples over fixed 15-minute intervals and is periodically flushed; and
- **Windowed**: a fixed-size circular buffer that retains only the most recent five minutes of latency samples.

Among the set of latency-related metrics collected during the experiments, this study focuses on the 95th percentile of the latency distribution as the target variable. Throughout the remainder of this paper, this metric is referred to as *P95*. The choice of P95 is motivated by its widespread adoption in network and service performance analysis, where upper-tail latency provides a more informative characterization of user-perceived quality than central tendency measures.

Latency labels are derived by computing P95 over the samples stored in each statistical structure. Under the *cumulative* strategy, all observed latency samples are aggregated since the beginning of the experiment, resulting in strong temporal smoothing but progressively incorporating historical outliers into the upper tail. The *drained* strategy limits this accumulation by periodically resetting the sample set at fixed intervals, thereby reducing long-term bias while still combining past and recent observations within each window. In contrast, the *windowed* strategy computes P95 over a fixed-size sliding window, producing near real-time labels that preserve short-term variability and more accurately reflect instantaneous system conditions in non-stationary environments.

Datasets were generated through three independent 12-hour experimental runs, one for each schema configuration (T100, T300, and T500) [4]. The sensor client maintained a constant throughput of 100 TPS throughout all experiments, while the load generator followed a sinusoidal pattern with a baseline of 46 concurrent clients, an amplitude of 44, a peak of 90 clients, and a period of 30 minutes.

From a FL perspective, the Non-IID characteristics induced by the proposed framework manifest along three complementary dimensions: temporal, distributional, and structural Non-IID. Temporal Non-IID arises from workload-driven dependencies over time, distributional Non-IID emerges from shifts across operational regimes, and structural Non-IID is induced by differences in schema width and service complexity.

[4]https://github.com/EricssonResearch/telemetry_sbrc_2026

A statistical characterization was performed to validate these properties. Figures 2 and 3 reveal strong temporal dependencies and significant distributional shifts across time windows. These effects are consistently observed across all configurations and become more pronounced as schema width increases.

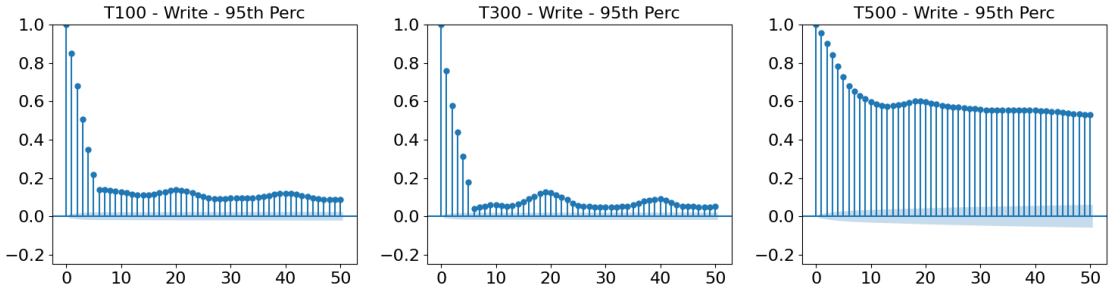

Figure 2. ACF for write operations across schema configurations (T100, T300, and T500). Persistently high ACF indicates strong temporal dependence.

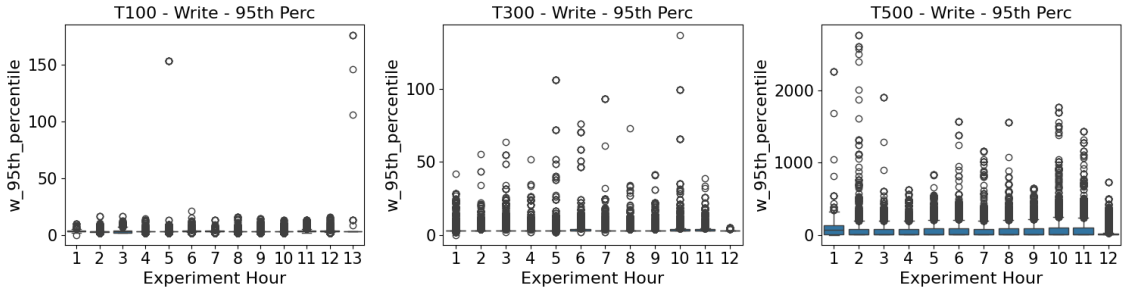

Figure 3. Distribution comparison across hourly time windows for write operations under different schema configurations (T100, T300, and T500). Shifts in distributional shape over time indicate Non-IID behavior. The y-axis label *w_95th_percentile* denotes the windowed 95th latency percentile (P95).

Table 1 summarizes the results of the Durbin–Watson and Kolmogorov–Smirnov tests. Durbin–Watson statistics well below 1.5 indicate strong positive autocorrelation, while Kolmogorov–Smirnov test p-values of zero confirm statistically significant distributional differences across time windows. Notably, the T500 configuration exhibits the strongest deviations, highlighting how increased schema complexity amplifies temporal and distributional heterogeneity. From a FL perspective, this pronounced heterogeneity leads to higher client drift and increased aggregation difficulty, posing additional challenges for model convergence and stability under realistic Non-IID conditions.

Table Size	Durbin–Watson	Kolmogorov–Smirnov
T100	0.36	0.0
T300	0.57	0.0
T500	0.45	0.0

Table 1. Statistical validation of Non-IID behavior across schema configurations.

4. Experiments and Results

Building upon the dataset generation process described in Section 3, we now evaluate the behavior of learning models under increasing structural and distributional heterogeneity. This section describes the experimental environment, the machine learning models

adopted, and the sequence of analyzes conducted to evaluate the impact of Non-IID data on regression performance, model generalization, and HFL.

The experimental environment is hosted on a physical server cluster located at FACOM/UFU, managed via Proxmox VE[5]. Eight virtual machines were provisioned to segment the workload and services. Six virtual machines compose a Docker Swarm cluster, one virtual machine provides gateway services, and one virtual machine is dedicated to load generation and sensor client execution. Each virtual machine is provisioned with 16 GB of RAM and 4 vCPUs. Storage is allocated such that seven virtual machines utilize an 80 GB SSD, while the Prometheus instance is assigned a 300 GB SSD to support extended telemetry retention. All virtual machines run Ubuntu 24.04 LTS.

The Docker Swarm cluster consists of one manager node responsible for orchestration and service routing, including Prometheus, Grafana, and Traefik, and five worker nodes hosting the Apache Cassandra database instances. Each Cassandra instance is pinned to a specific worker node, using local persistent storage and a dedicated network port. Data availability is ensured through a keyspace configured with a replication factor of three, using the `SimpleStrategy` class, which is suitable for the single-datacenter, single-rack deployment used in this study. This controlled setup allows isolating workload- and schema-induced effects from hardware-induced variability.

As previously described, three independent experimental runs were conducted, each lasting 12 hours and corresponding to a single schema configuration (T100, T300, and T500). During each run, only one schema was active, ensuring that the collected datasets reflect isolated operational conditions associated with a specific schema width. Across all experiments, a constant request rate of 100 TPS (70% write and 30% read) was maintained for target labeling, while a time-varying workload was applied by the load genarator to induce fluctuations in system. After dataset collection, all analyses were performed offline using the recorded telemetry.

All experiments employ a neural network based on the Wide & Deep learning paradigm defined in Section 2. The wide component is a linear regressor that maps the full input feature vector X directly to the target output Y. Each input vector X is composed of 435 features, including 431 telemetry features collected from Prometheus at the selected entry point node for a given schema, and four engineered temporal features. These temporal features are derived using a lag strategy consisting of $y_{t-1}, y_{t-2}, y_{t-3}$, and a rolling mean of the last five observations, following the methodology described by [Leites et al. 2024]. Although the Cassandra cluster consists of five nodes, only three of them act as entry points in this study—corresponding to the T100, T300, and T500 configurations—and constitute the effective participants in the HFL experiments. The target variable Y corresponds to the P95 latency.

In parallel, the deep component consists of an initial fully connected layer with 512 units, followed by two residual-style blocks structured as Linear $\rightarrow$ Normalization $\rightarrow$ GELU $\rightarrow$ Dropout. A final linear layer produces a scalar output. The predictions from the wide and deep components are summed to generate the final estimate. The network architecture is illustrated in Figure 4. The same architecture and hyperparameters were used across all experiments to ensure comparability of results.

[5]https://www.proxmox.com/en/products/proxmox-virtual-environment/overview

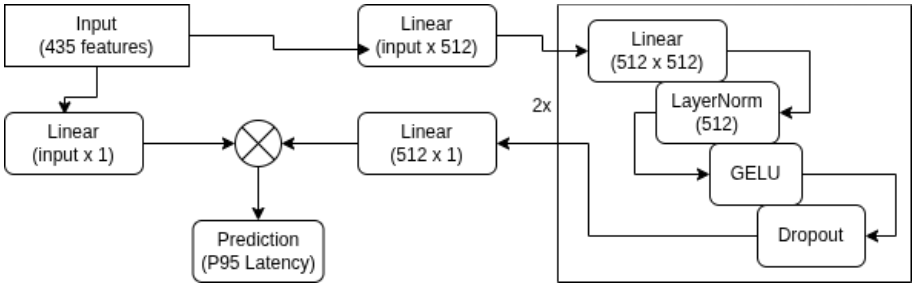

Figure 4. Wide & Deep neural network architecture used in all experiments.

Prior to feature engineering and model training, an outlier detection and imputation step was applied to the datasets. Observations with Z-score values exceeding ± 1.96 were identified as outliers. This threshold corresponds to a 95% confidence interval under an approximate normality assumption. These samples were treated as missing values and imputed using interpolation to preserve temporal continuity. Special care was taken to handle boundary conditions at the beginning and end of each time series. This preprocessing step reduces the impact of extreme values while maintaining the temporal structure.

The evaluation follows a three-stage methodology designed to progressively analyze the effects of Non-IID data, model transferability, and HFL. The first two stages rely exclusively on standalone learning, in which independent regression models are trained and evaluated without any form of federation. HFL is employed only in the third stage, where multiple participants collaboratively train a shared model.

To mitigate the challenges posed by data heterogeneity in the federated setting, the FedProx framework is adopted for local client-side optimization in Stage 3, in conjunction with the FedAvg algorithm, which serves as the primary mechanism for parameter aggregation at the server. All reported NMAE values correspond to the mean and standard deviation computed over ten independent training rounds using different random seeds.

Stage 1 – Individual Dataset Analysis: Each dataset (T100, T300, and T500) is analyzed independently in a standalone setting. For each schema, separate regression models are trained for read and write operations using the P95 latency as the target variable. Three labeling strategies are evaluated: cumulative, drained, and windowed. Model performance is assessed using the Normalized Mean Absolute Error (NMAE).

Stage 2 – Cross-Dataset Evaluation: To assess model transferability under Non-IID conditions, models trained on one dataset in Stage 1 are evaluated on the remaining datasets. For example, a model trained on T100 is evaluated on T300 and T500. This stage quantifies the degradation in predictive performance when models are applied to environments with different structural and distributional characteristics, without retraining, fine-tuning, or any form of federation.

Stage 3 – FL Evaluation: HFL experiments are conducted using different combinations of participants. A global federation involving all three datasets (T100, T300, and T500) is evaluated, as well as pairwise federations (T100+T300, T100+T500, and T300+T500). This stage investigates the impact of distributional and structural heterogeneity on model aggregation and convergence, with particular emphasis on the influence of T500.

Table 2 summarizes the results of the individual dataset analysis conducted in

Stage 1. Models trained using cumulative and drained labeling strategies achieved low NMAE values, reflecting the smoothing effect introduced by long-term aggregation. The windowed labeling strategy resulted in significantly higher NMAE, particularly for read operations, indicating increased stochasticity and stronger temporal variability.

		T100	T300	T500
		NMAE ± Std. Dev	**NMAE ± Std. Dev**	**NMAE ± Std. Dev**
Cumulative	**Read**	0.03% ± 0.0001	0.05% ± 0.0000	0.07% ± 0.0000
	Write	0.02% ± 0.0001	0.03% ± 0.0001	0.08% ± 0.0000
Drained	**Read**	0.42% ± 0.0001	0.85% ± 0.0001	6.67% ± 0.0438
	Write	0.21% ± 0.0001	0.10% ± 0.0001	9.85% ± 0.1040
Windowed	**Read**	12.27% ± 0.0002	26.67% ± 0.0006	19.15% ± 0.0005
	Write	5.11% ± 0.0002	5.24% ± 0.0002	26.73% ± 0.0006

Table 2. NMAE with standard deviation for individual dataset analysis of Stage 1.

To better understand the behavior of the windowed target, Figure 5 shows the empirical CDFs of the P95 latency for read and write operations. For read operations, T100 concentrates at lower latency values, while T300 and especially T500 exhibit increased dispersion, with distribution shifts beginning at earlier percentiles. For write operations, T100 and T300 remain similar until the upper tail, whereas T500 deviates substantially earlier, indicating a pronounced impact of schema width on tail latency. Although T100 and T300 each exhibit Non-IID behavior individually, the distributional differences between them remain relatively subtle, whereas T500 presents a markedly distinct and more severe Non-IID profile. This behavior is expected, as windowed labeling intentionally preserves short-term variability rather than smoothing it out, and reflects scenarios in networked systems where environments with comparable infrastructure and operational characteristics coexist alongside substantially different deployments.

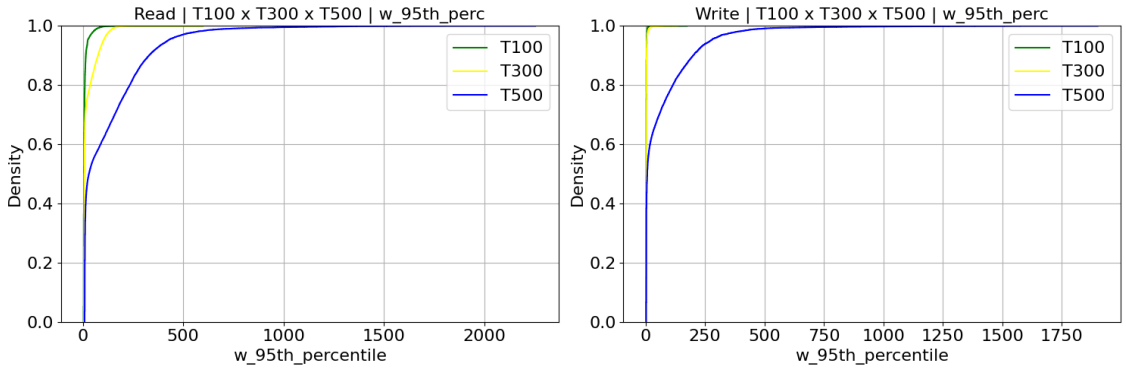

Figure 5. CDF of windowed P95 latency for read and write across table schemas.

Cross-dataset evaluation results for Stage 2 are presented in Table 3. While models trained on T100 and T300 exhibit moderate degradation when applied to each other, both fail to generalize to the T500 environment. Conversely, models trained on T500 perform poorly when evaluated on smaller schemas, suggesting that the learned representations are highly specialized. These results indicate not only distributional, but also structural Non-IID effects induced by increasing schema width. From a networking perspective, this behavior reflects scenarios in which models trained in one locality or operational context cannot be directly transferred to others due to fundamental differences in infrastructure, workload characteristics, and service complexity. Such structural heterogeneity

represents a major challenge for the adoption of machine learning in the orchestration and automation of large-scale 5G and 6G network environments.

	Model T100	**Model T300**	**Model T500**
T100	5.11% ± 0.0002	9.79% ± 0.0009	2862.71% ± 2.3615
T300	10.84% ± 0.0015	5.27% ± 0.0002	4585.93% ± 4.1012
T500	94.10% ± 0.0003	93.73% ± 0.0003	26.73% ± 0.0006

Table 3. Cross-dataset NMAE and standard deviation (write operations).

The cross-dataset degradation observed in Stage 2 is consistent with the distributional differences of the target variable previously highlighted through CDF analysis. Figure 6 provides a complementary and more granular view by presenting the *Kernel Density Estimation (KDE)* of the windowed P95 latency for write operations across schema configurations. While the CDFs emphasize divergence in the upper tail, the KDE reveals differences in the overall shape and mass concentration of the distributions. In particular, the T500 distribution exhibits a broader spread and a heavier tail, indicating a higher density of extreme latency values, whereas T100 and T300 present narrower modes and more concentrated density regions. This visual contrast helps explain why models trained on smaller schemas fail to generalize to T500 and why models trained on T500 transfer poorly to T100 and T300.

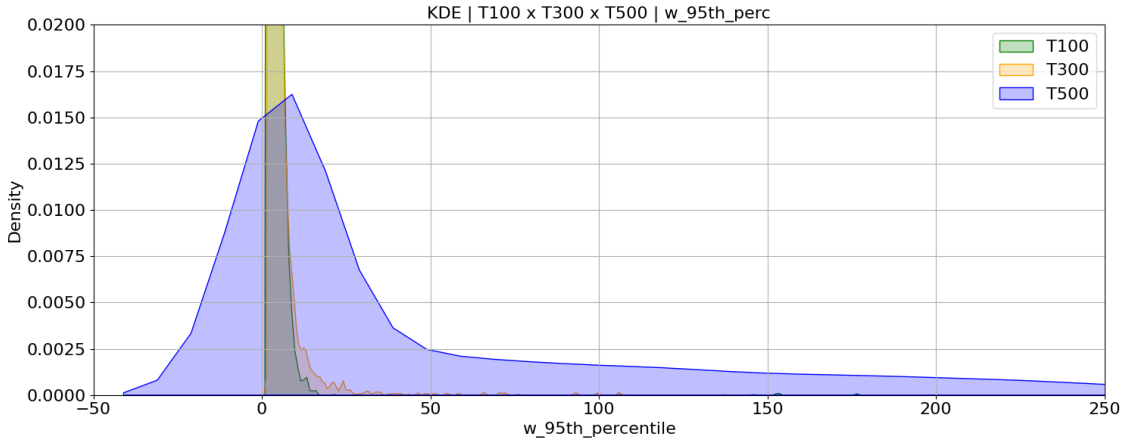

Figure 6. KDE of windowed P95 latency for write operations across table sizes.

The HFL results for Stage 3 are summarized in Table 4. A global federation involving all three datasets fails to converge across local distributions, reflecting the severe heterogeneity introduced by the T500 configuration. In contrast, the federation of T100 and T300 achieves low NMAE values, indicating successful aggregation under moderate heterogeneity. However, when compared to the corresponding individual models from Stage 1 (Table 2), the federated models for T100 and T300 exhibit a slight degradation in accuracy. This behavior highlights a well-known trade-off in HFL, where aggregation may reduce local specialization in exchange for improved generalization.

Notably, when compared to the cross-dataset evaluation results (Table 3), the federated models for T100 and T300 significantly improve predictive performance. This indicates that, even under distributional differences, horizontal federation provides tangible benefits over direct model transfer across heterogeneous environments. For the T500

	Federated Models			
	T100 & T300 & T500	**T100 & T300**	**T100 & T500**	**T300 & T500**
T100	132.12% ± 0.0225	7.46% ± 0.0004	282.05% ± 0.0791	-
T300	123.08% ± 0.0309	8.52% ± 0.0006	-	284.17% ± 0.0601
T500	86.22% ± 0.0005	-	78.77% ± 0.0014	78.64% ± 0.0013

Table 4. NMAE and standard deviation for HFL experiments.

configuration, although the federated results remain suboptimal in absolute terms, federation yields substantial improvements when contrasted with cross-dataset evaluation. This suggests that, despite its pronounced structural and distributional Non-IID characteristics, partial knowledge sharing through federation still mitigates extreme performance degradation. Nevertheless, T500 acts as a dominant outlier that compromises global aggregation, particularly when combined with less complex schemas.

Our framework adopts a HFL paradigm under the assumption of a homogeneous feature space across participants. Despite the application of outlier filtering, the structural heterogeneity induced by increasing schema width limits the effectiveness of global aggregation, particularly in the presence of highly divergent participants such as T500. Feature selection was intentionally avoided throughout the main experimental analysis in order to preserve a common input space and maintain a strictly horizontal federation, as independent feature selection would naturally shift the problem toward a vertical federated learning (VFL) setting.

Nevertheless, preliminary experiments conducted outside the main evaluation pipeline indicate that feature selection can substantially mitigate the impact of structural Non-IID effects. Using collinearity-based filtering, the dimensionality of the input space was reduced from 435 features to 334 for T100, 328 for T300, and 200 for T500. Under this reduced representation, the predictive performance of T500 improved significantly, with NMAE values dropping to approximately 11%, compared to the 19–26% range observed in the standalone analysis of Stage 1. While these results are exploratory and not included as a separate evaluation stage, they provide evidence that alternative learning paradigms, including VFL and more advanced feature-aware or representation-based approaches, are promising directions for future work. These findings reinforce that the proposed framework is not only capable of exposing the limitations of HFL under realistic Non-IID conditions, but also provides the necessary experimental control and observability to support the systematic investigation of more advanced FL strategies.

5. Conclusions and Future Work

This paper presented an experimental framework for investigating FL under realistic Non-IID conditions derived from networked service execution. By combining controlled workload generation, fine-grained telemetry collection, and statistical characterization, we demonstrated how increasing structural and distributional heterogeneity impacts regression performance and HFL. The experimental results show that while horizontal federation can improve generalization compared to direct model transfer, its effectiveness degrades under pronounced structural Non-IID conditions, highlighting fundamental limitations in highly heterogeneous environments.

As future work, several extensions of the proposed framework can be explored. Outlier handling strategies may be further refined, and feature selection mechanisms can

be incorporated to reduce redundancy and isolate the most informative variables. Candidate approaches include collinearity-aware selection, statistical methods such as SelectKBest, and latent representations derived from Autoencoder networks.

Feature selection in federated settings shifts the paradigm from HFL to VFL due to emerging structural heterogeneity. This transition is supported by studies on feature optimization: [Ickin et al. 2021] used VFL for efficient QoE prediction, while [Arachchige et al. 2024] employed eccentricity-based analysis to quantify metric contributions. Furthermore, [Ickin 2023] introduced automated pruning to mitigate communication overhead and training time.

Acknowledgements

This work was supported by Ericsson Telecomunicações Ltda., and by the Sao Paulo Research Foundation (FAPESP), grant 2021/00199-8, CPE SMARTNESS. This work made use of artificial intelligence tools to assist in the organization, revision, and linguistic refinement of the manuscript. All technical content, experimental design, analysis, and interpretations are the responsibility of the authors.

References

Arachchige, T. K., Ickin, S., Abghari, S., and Boeva, V. (2024). Clients Behavior Monitoring in Federated Learning via Eccentricity Analysis. In *2024 IEEE International Conference on Evolving and Adaptive Intelligent Systems (EAIS)*, pages 1–8, Madrid, Spain. IEEE.

Bruce, P., Bruce, A., and Gedeck, P. (2020). *Practical Statistics for Data Scientists: 50+ Essential Concepts Using R and Python*. O'Reilly Media, Inc., Sebastopol, CA, 2 edition.

Cheng, H.-T., Koc, L., Harmsen, J., Shaked, T., Chandra, T., Aradhye, H., Anderson, G., Corrado, G., Chai, W., Ispir, M., Anil, R., Haque, Z., Hong, L., Jain, V., Liu, X., and Shah, H. (2016). Wide & Deep Learning for Recommender Systems. arXiv:1606.07792 [cs].

Coronado, E., Behravesh, R., Subramanya, T., Fernandez-Fernandez, A., Siddiqui, M. S., Costa-Perez, X., and Riggio, R. (2022). Zero Touch Management: A Survey of Network Automation Solutions for 5G and 6G Networks. *IEEE Communications Surveys & Tutorials*, 24(4):2535–2578.

Criado, M. F., Casado, F. E., Iglesias, R., Regueiro, C. V., and Barro, S. (2022). Non-IID data and Continual Learning processes in Federated Learning: A long road ahead. *Information Fusion*, 88:263–280.

Dodge, Y. (2008). *Kolmogorov–Smirnov Test*, pages 283–287. Springer New York, New York, NY.

Ickin, S. (2023). Automated Feature Selection with Local Gradient Trajectory in Split Learning. In *NOMS 2023-2023 IEEE/IFIP Network Operations and Management Symposium*, pages 1–7, Miami, FL, USA. IEEE.

Ickin, S., Fiedler, M., and Vandikas, K. (2021). QoE Modeling on Split Features with Distributed Deep Learning. *Network*, 1(2):165–190.

Jawad, A. T., Maaloul, R., and Chaari, L. (2023). A comprehensive survey on 6G and beyond: Enabling technologies, opportunities of machine learning and challenges. *Computer Networks*, 237:110085.

Kim, M., Lee, S., and Kim, J. (2020). A Wide & Deep Learning Sharing Input Data for Regression Analysis. In *2020 IEEE International Conference on Big Data and Smart Computing (BigComp)*, pages 8–12, Busan, Korea (South). IEEE.

Leites, J., Cerqueira, V., and Soares, C. (2024). Lag Selection for Univariate Time Series Forecasting using Deep Learning: An Empirical Study. arXiv:2405.11237 [stat].

Li, T., Sahu, A. K., Zaheer, M., Sanjabi, M., Talwalkar, A., and Smith, V. (2018). Federated Optimization in Heterogeneous Networks. arXiv:1812.06127 [cs].

Lu, Z., Pan, H., Dai, Y., Si, X., and Zhang, Y. (2024). Federated Learning With Non-IID Data: A Survey. *IEEE Internet of Things Journal*, 11(11):19188–19209.

Maduranga, M. W. P., Tilwari, V., Rathnayake, R. M. M. R., and Sandamini, C. (2024). AI-Enabled 6G Internet of Things: Opportunities, Key Technologies, Challenges, and Future Directions. *Telecom*, 5(3):804–822.

McMahan, B., Moore, E., Ramage, D., Hampson, S., and Arcas, B. A. y. (2017). Communication-Efficient Learning of Deep Networks from Decentralized Data. In Singh, A. and Zhu, J., editors, *Proceedings of the 20th International Conference on Artificial Intelligence and Statistics*, volume 54 of *Proceedings of Machine Learning Research*, pages 1273–1282. PMLR.

Pölsterl, S., Sarasua, I., Gutiérrez-Becker, B., and Wachinger, C. (2020). A Wide and Deep Neural Network for Survival Analysis from Anatomical Shape and Tabular Clinical Data. In Cellier, P. and Driessens, K., editors, *Machine Learning and Knowledge Discovery in Databases*, volume 1167, pages 453–464. Springer International Publishing, Cham. Series Title: Communications in Computer and Information Science.

Stadler, R., Pasquini, R., and Fodor, V. (2017). Learning from Network Device Statistics. *Journal of Network and Systems Management*, 25(4):672–698.

Surakhi, O., Zaidan, M. A., Fung, P. L., Hossein Motlagh, N., Serhan, S., AlKhanafseh, M., Ghoniem, R. M., and Hussein, T. (2021). Time-Lag Selection for Time-Series Forecasting Using Neural Network and Heuristic Algorithm. *Electronics*, 10(20):2518.

Tang, D., Yang, N., Li, Y., Zhu, Z., Jin, Z., and Yuan, D. (2025). Optimal Look-back Horizon for Time Series Forecasting in Federated Learning. arXiv:2511.12791 [cs].

Yang, Q., Liu, Y., Chen, T., and Tong, Y. (2019). Federated Machine Learning: Concept and Applications. arXiv:1902.04885 [cs].

Zhao, Y., Li, M., Lai, L., Suda, N., Civin, D., and Chandra, V. (2018). Federated learning with non-iid data.

Zhao, Z., Feng, C., Hong, W., Jiang, J., Jia, C., Quek, T. Q. S., and Peng, M. (2022). Federated Learning With Non-IID Data in Wireless Networks. *IEEE Transactions on Wireless Communications*, 21(3):1927–1942.

Zhao, Z., Wang, J., Hong, W., Quek, T. Q. S., Ding, Z., and Peng, M. (2024). Ensemble Federated Learning With Non-IID Data in Wireless Networks. *IEEE Transactions on Wireless Communications*, 23(4):3557–3571.