A Practical Framework for Federated Fuzzy Clusterwise Regression, Preprocessing and Hyperparameter Tuning

Morris Stallmann, Gerhard Weiss, Anna Wilbik
Department of Advanced Computing Sciences (DACS)
Maastricht University
Maastricht, The Netherlands
{m.stallmann, gerhard.weiss, a.wilbik}@maastrichtuniversity.nl

***Abstract*—Federated learning has the potential to enable novel use cases in environments with data sharing constraints. While practitioners acknowledge the potential and researchers show applicability to many different use case domains, this new paradigm has not been widely adopted in practice yet. Various studies identify impact factors on adoption, including the communication overhead of the FL approach, non-identically or independently distributed (non-IID) training data, data quality problems, complexity of implementing best practices from (centralized) machine learning operations (MLOps), just to name a few. Moreover, federated regression problems remain an underexplored area as recent research activity has shown. To address the barriers to federated learning adoption, we introduce an end-to-end federated regression framework. Its core component is a novel federated fuzzy clusterwise regression method (FedFCR), complemented by federated protocols for preprocessing (to ensure data quality) and hyperparameter optimization. Through extensive numerical experiments, we find that FedFCR achieves competitive predictive performance, comparable to other recently introduced federated fuzzy regression methods, but with lower communication overhead. Moreover, our experiments show that FedFCR is robust against non-IIDness in training data. In summary, we hope that our practical end-to-end regression framework with federated protocols for preprocessing and hyperparameter optimization, low communication overhead, and robustness against non-IID data helps magnifying the adoption of federated learning in practice.**

***Index Terms*—federated learning, fuzzy clustering, fuzzy inference system, federated regression, clusterwise regression, adoption, federated hyperparaemeter optimization, non-IID data**

I. Introduction

Federated learning (FL) is machine learning on distributed data without transferring it into a central location. It promises to enable the use of data locked in silos due to privacy regulations or other data sharing constraints. In the FL approach of this work, a *central server* coordinates the learning process. It shares a global model that *clients* update using their private data. Instead of sharing the data, the clients share model updates that the central server aggregates to a global model, often by applying an averaging algorithm named $FedAvg$ or variants thereof [1].

The research community has demonstrated the applicability of FL to a wide range of use case domains such as Internet of Things [2], healthcare [3], autonomous driving and vehicular networks [4], predictive maintenance [5]. Recently, federated regression models have been identified as an underexplored area, and some works explore using federated fuzzy methods to achieve good predictive performance paired with explainability, privacy preservation, and adaptivity [6]–[8].

However, there are still relatively few reports about real-world applications of FL, with [9] and [10] being notable exceptions. The authors of [9] explicitly mention robustness and hyperparameter optimization as practical challenges, and in [10] an FL approach that is robust against non-independent or identically distributed (IID) training data is chosen.

Several recent studies investigate the state of federated learning adoption in practice and why it is still lagging behind despite many research breakthroughs. On the one hand, studies find that practitioners recognize the huge potential of FL [11] in enabling novel use cases, but also identify multiple reasons for not adopting the technology in practice. In the same study, multiple impact factors are identified: Data quality and interoperability concerns, management support, legal clarity and collaboration management, and complex technical implementation are among the most influential impact factors. Moreover, the authors of [12] have identified data heterogeneity, system heterogeneity, and communication overhead as key challenges in federated learning adoption. In [13], a seven-stage model for the adoption of federated learning in practice is proposed and validated. Besides stages like project initialization, business case validation, or collaboration management, their model also includes more technical steps like federated data preprocessing to ensure data quality and alignment with (centralized) MLOps best practices such as experimentation, code, data, and model versioning, monitoring, and automation [14].

These concerns motivate us to address key concerns in federated learning adoption by designing a federated regression framework that shows good performance on IID and non-IID data, has low communication overhead to enable quick experimentation and application in communication-constrained environments, uses a federated protocol for preprocessing to ensure consistent data quality, and is embedded into a federated hyperparameter optimization framework.

We choose a fuzzy regression method or fuzzy inference system as the main building block of such a framework for

multiple reasons: Fuzzy regression methods have already been demonstrated to be effective in the federated setting (see Section II), fuzzy methods handle uncertain and imprecise data well (see e.g. [15]), and they are generally considered explainable. Furthermore, (fuzzy) clusterwise regression is known to handle non-IID data well in the centralized case (see Section II), motivating its application to a similar problem in the federated setting.

Our main contribution is a novel federated fuzzy clusterwise regression method that requires only two rounds of communication between the central server and clients. It is composed of two stages: Robust federated fuzzy c-means clustering and federated fuzzy clusterwise ridge regression with a fuzzy federated averaging mechanism. The method shows good performance on various regression datasets under IID and non-IID assumptions, and results in an interpretable model. However, we acknowledge that less communication-efficient frameworks achieve higher prediction accuracy on a number of benchmark datasets.

Our secondary contributions are federated protocols for preprocessing and hyperparameter optimization so that the federated fuzzy clusterwise regression method is embedded into an end-to-end learning framework. To the best of our knowledge, this is one of the first practical end-to-end frameworks for learning federated regression models.

In the remainder of this paper, we will first introduce related work and the foundations of our framework in Section II before describing the end-to-end framework in Section III. Section IV is about the experimental evaluation, while Section V concludes this work with final remarks.

II. Background and Foundations

A. Fuzzy Inference Systems

Fuzzy inference systems (FIS) map inputs (features) to outputs (target values) by applying fuzzy logic. The process of learning such a system from data typically involves two main phases: Rule induction or structure identification and rule base optimization or parameter identification [16], [17]. In the first phase, a suitable number of rules is derived, e.g., by partitioning the feature space into fuzzy sets by applying fuzzy clustering techniques or utilizing expert knowledge [16]. In the second phase, the rules' parameters are optimized to achieve an objective, e.g., minimizing the prediction error.

Recent works have demonstrated how (centralized) FIS can be translated into the federated setting [17], how fuzzy computation aids the federated averaging process [18], and how federated FIS can be tailored to use cases such as fall detection [19] or trust evaluation in cloud computing [20]. All of these approaches assume scenarios where multiple rounds of communication between the central server and the clients are feasible. However, that assumption may not hold true in all practical use cases (see e.g. [12]).

The authors of [6] introduced a communication-efficient federated FIS with the primary concern of explainability. In their approach, clients first identify structure locally, optimize the rules' parameters locally, and share those with the central server. The central server generates global rules while maintaining explainability. While the approach indeed results in an interpretable model with reasonable performance, it remains unclear whether the approach is prone to performance degradation under the non-IID assumption. Moreover, the approach could be improved by recognizing global structure (see Section III-B1) rather than local structure.

1) Fuzzy Clusterwise Regression: Fuzzy clusterwise regression models are fuzzy inference systems, where the structure identification is accomplished through fuzzy clustering and the parameter optimization is accomplished through fitting a (often linear) regression model. These approaches have been found to handle data heterogeneity well in the centralized case while still being interpretable by humans [21], [22]. The clustering model identifies K regions where the relationship between features X_k and targets y_k can be approximated by linear functions for $1, \ldots, K$. Fuzzy clusters have an advantage over crisp clusters as they are more robust against noise in the features [15]. For example, a small perturbation in one feature may change a crisp assignment while only having minimal impact on membership values. This robustness against data heterogeneity and noise motivates the application of fuzzy clusterwise regression in the federated learning setting.

A recent article introduced a related approach in the FL setting, where Gaussian clustering and fuzzy regression are combined [8] to achieve good performance in various benchmark datasets, named eGauss+$_{FR}$. This work focuses on evolving clusters on datastreams and, as such, the approach is designed for multiple rounds of communication. Moreover, one of the authors' main assumptions is that the relationship between X and y is the same at all clients, which may make it unsuitable in the case of non-IID data.

2) Federated Fuzzy Regression Trees: Lastly, the authors of [7] recently introduced a federated version of fuzzy regression trees (FRT), dubbed FL-FRT. In their approach, clients submit statistics of their data that the central server aggregates to grow FRTs in multiple rounds of communication. A key concern in their approach is privacy preservation and model explainability. In their experimental evaluation, the authors show that both can be accomplished while also demonstrating good predictive performance. However, it remains open whether the method is able to deal with non-IID data since it was not the focus the study and was not investigated.

III. Proposed Framework

Our framework consists of three components: the preprocessing component, the learning component, and the hyperparameter optimization component. We will introduce each of them in subsequent sections in detail. For a high-level overview of how the components interplay, see Figure 1.

A. Federated Preprocessing

The preprocessing in our framework consists of two steps: First, clipping extreme values to quantiles q_{lower} and q_{upper}. Second, each feature x is scaled into the range of $[0, 1]$:

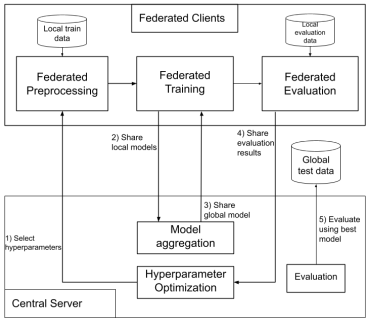

Fig. 1. Schematic Overview of the Federated Learning and Hyperparameter Optimization Process.

$x_{scaled} = \frac{x - min(x)}{max(x) - min(x)}$. Note that both preprocessing steps require the calculation of k-th ranked elements.

1) Federated Computation of k-th Ranked Element: Methods for calculating k-th ranked elements in privacy-preserving settings were introduced in [23]–[25]. In particular, the method introduced in [25] has the lowest communication complexity, as the authors note. Their method requires the total number of data points $|X_i|$ as well as $a := \min X_i$ and $b := \max X_i$ of a feature vector X_i to be known to the central server. Each client c shares their local $|X_i^c|$, $\min X_i^c$, $\max X_i^c$ such that the central server can derive the global numbers. Then, the following is repeated until the k-th ranked object is identified:

1) The server shares a number $\mu = \lfloor \frac{a+b}{2} \rfloor$ with the clients.
2) The clients report back how many elements in their local datasets are smaller and greater than μ.
3) Based on that information, the central server updates μ by adding or subtracting a constant s. The clients repeat the local calculations and share the results.

B. Federated Fuzzy Clusterwise Ridge Regression

Clusterwise regression is known to handle data heterogeneity well in the centralized case. In clusterwise regression problems, data heterogeneity is interpreted as different clusters of observations, where each cluster k has a different joint distribution $P(X_k, Y_k)$. Therefore, clusterwise regression aims to identify those clusters of observations and then learn a separate prediction model for each cluster k. In a federated setting, first, global clusters X_k must be derived from local data, even in the non-IID case (Section III-B1). Second, the K models must be learned according to a federated protocol (Section III-B2).

1) Training Stage 1: Robust Federated Fuzzy c-Means Clustering: The first phase of model training aims to identify structure for use in the second phase. Our framework applies a federated fuzzy c-means formulation that was introduced in [26], because it was specifically designed for and evaluated on non-IID data with good robustness properties and it can be restricted to just one round of communication.

The objective in (federated) fuzzy c-means clustering is to choose cluster centers $c_1, c_2, \ldots, c_K$ that minimize $J_m(U, c) = \sum_{i=1}^{n} \sum_{j=1}^{K} (u_{ij})^m ||x_i - c_j||^2$ on some distributed dataset $X = \bigcup_{j=1}^{C} X_j$, where

$$u_{ij}([c]_{k=1}^{K}) := \frac{1}{\sum_{k=1}^{K} \left(\frac{||x_i - c_j||^2}{||x_i - c_k||^2} \right)^{\frac{2}{m-1}}} \quad (1)$$

is the membership degree of data point x_i to cluster j. Further: $\sum_{j}^{K} u_{ij} = 1$.

The federated solution algorithm in this framework first applies local optimization and then k-means averaging [26]:

1) The central server randomly selects a client that calculates the initial global centers $c_1, \ldots, c_K$ and shares the initial centers with the clients.
2) Locally, the minimum of $J_m(U^j, c^j)$ with X_j is found.
3) Each client shares local centers $c_1^j, \ldots, c_K^j =: c^j$.
4) The central server computes the global centers by applying the k-means clustering algorithm with local centers $[c]^1, \ldots, [c]^C$ and shares the resulting global centers.

2) Training Stage 2: Fuzzy Clusterwise Ridge Regression: By sharing the global cluster centers, the clients are informed about the global structure to utilize in the regression model learning phase.

In this phase, client c solves one optimization problem for each fuzzy cluster $k = 1, \ldots, K$ exactly:

$$\min_{\omega_c^k, \beta_c^k} \left(\sum_{i=1}^{N_c} u_{ik} (x_i \omega_c^k + \beta_c^k - y_i)^2 + \alpha ||\omega_c||_2^2 \right), x_i \in X^c \quad (2)$$

where u_{ik} is the membership degree, calculated according to Equation (1). After solving the problem, each client shares the optimal solutions $\xi_c^k = (\hat{\omega_c}^k, \hat{\beta_c}^k)$ with the central server.

Moreover, each client shares the fuzzy cardinality of all K cluster sets $A_c^k := \sum_{i=1}^{N_c} u_{ik}$ with the central server. Using that information, the central server computes the final ensemble of K models by applying a fuzzy federated averaging mechanism: $\hat{\omega_k} = \sum_{c=1}^{C} \frac{A_c^k}{\Delta_k} \hat{\omega_c}^k$, and $\hat{\beta}_k = \sum_{c=1}^{C} \frac{A_c^k}{\Delta_k} \hat{\beta}_c^k$, where $\Delta_k = \sum_{c=1}^{C} A_c^k$. The result is shared with the clients for inference on local data. Note that, since all clients solve the problem exactly, there is only one round of communication between the central server and each client required.

To calculate the prediction for a new data point x, first the membership vector u (Equation (1)) is computed. Second, the final prediction is calculated as the weighted sum of all K regression models: $M_{pred}(x) = \sum_{k=1}^{K} u_k (\hat{\omega}_k x + \hat{\beta}_k)$.

Utilizing the global structure from stage 1 promises robustness against non-IIDness, provided the global structure identification is robust against non-IID data. This is a slightly different take on addressing non-IIDness compared to more established methods. Popular approaches are either regularized versions of $FedAvg$, cluster clients with compatible data, or augment local training data to achieve robustness [27]. In contrast, our approach aims to identify compatible subregions in the global data space.

C. Hyperparameter Tuning

The third component of the end-to-end framework is a federated protocol for hyperparameter optimization. Let w be a vector of all hyperparameters in the preprocessing, clustering,

and clusterwise regression stages (see Table I for an overview). Then, we define the optimization problem as

$$\min_w \sqrt{\frac{1}{N}\sum_{i=1}^{N}(M_w(x_i)-y_i)^2} =: \min_w f(w), \qquad (3)$$

where M_w is a composite function of the preprocessing, cluster, and clusterwise regression models: $M = M_{pre} \circ M_{cluster} \circ M_{pred}$. M's analytical form is opaque such that we consider the objective a black-box function for the purpose of hyperparameter optimization.

Black-box optimization or derivative-free optimization is a common problem in ML or neural network hyperparameter optimization and is often approached by Bayesian Optimization (BO) methods [28]–[30], which optimize a surrogate $\hat{f}(w) \approx f(w)$. Our framework applies the hyperparameter optimization framework introduced in [29] with Tree-Structured Parzen Estimators (TPE) informing the search [31][1].

The federated protocol of the optimization is as follows:

1) The server chooses a set of parameters w_t in iteration t.
2) Model M_{w_t} is trained as described above.
3) Clients $c = 1, \ldots, C$ evaluate objective (3) on local validation sets X_c^{val} and report back the results ρ_t^c as well as $N_c := |X_c^{val}|$.
4) The server calculates the global objective value as the average of the local values: $\rho^t = \frac{1}{N}\sum_{c=1}^{C} N_c \rho_t^c$.
5) Based on (w_t, ρ_t), the server updates its prior of the surrogate $\hat{f}(w)$ and chooses w_{t+1}.
6) Repeat steps 1) - 5) for a predefined number of iterations.

For a detailed description of the TPE, see [31] and [32].

IV. Experimental Evaluation and Discussion

The main goal of the experiments is to study how our two-shot regression framework performs under IID and non-IID assumptions, where performance is measured by the root mean squared error (RMSE), R^2 score, and mean absolute percentage error (MAPE).

In the IID case, we compare the performance with other recently introduced federated learning fuzzy regression frameworks, namely the explainable regression trees introduced in [6], FL-FRT [7], and the eGauss+ regression method [8].

To study the impact of non-IID data, we compare the performance on the same datasets using the same hyperparameters under the IID and non-IID assumptions. Note that [6]–[8] do not report results on non-IID data.

A. Datasets

We follow the experimental evaluation of related works and evaluate our method on a subset of regression datasets in the KEEL dataset repository [33]. Our framework is applied to all datasets in the regression category[2] under the assumption of IID and non-IID scenarios and we report the results on the datasets that were used in the evaluation in [6]–[8] for comparison.

[1]Implementation we use: https://optuna.readthedocs.io/en/stable/index.html

[2]https://sci2s.ugr.es/keel/category.php?cat=reg, accessed 13.01.2026

Dataset	q_{lower}	q_{upper}	m	K	α
Mortgage	-	-	3.4885	6	0.0019
Treasury	-	-	2.442	14	0.2527
Weather Iz.	-	-	7.8992	14	2.0308
Delta Elev.	0.02	1.0	3.7293	17	0.1576
Elevators	0.02	0.92	-	1	0.3299
House	0.08	0.9	1.1634	26	0.9016
California	0.03	0.96	1.897	19	2.771

TABLE I
Optimal Hyperparameters per dataset.

To create a federated scenario from the KEEL datasets, data $(x, y) \in (X, Y)$ are randomly distributed among $C = 10$ clients according to a distribution function $Dir(\alpha)$, where α determines the "non-IIDness" of the federated scenario (see next Section). To create the IID scenario, we choose $\alpha = 100$, and to create the non-IID scenario, we choose $\alpha = 0.5$.

B. Non-IID Scenario Creation

To simulate federated non-IID scenarios, we sample from a Dirichlet distribution, similar to related works (see, e.g., [27], [34]). The approach requires data to be labeled. Therefore, we create artificial labels based on percentiles: If $perc_{j*10} \leq y_i < perc_{j*(10+1)}$, then we assign label j to (x_i, y_i), where $perc_k$ is the k-th percentile and $j = 0, 1 \ldots, 9$. For each artificial label l and client c, a label proportion vector $p_c = (p_{c,0}, p_{c,1}, \ldots, p_{c,9})$ is sampled according to a Dirichlet distribution with concentration parameter α: $p_c \sim Dir(\alpha)$. That is, the lower α, the higher the non-IIDness.

C. Experimental Protocol

The experimental protocol consists of two main stages. While in the first stage, the optimal hyperparameters are determined by applying our optimization framework, the second stage focuses on calculating the final evaluation metrics.

Moreover, the experimental stages are executed once with preprocessing and once without. Code to reproduce the experiments is available[3].

1) Hyperparameter Search: In the first phase, the goal is to determine the optimal set of hyperparameters $\hat{\xi}$ and fix them for the evaluation phase (see Table I for an overview of the hyperparameters). The following steps are executed:

1) Data is split into train (80%) and test (20%) sets.
2) The data is distributed among $C = 10$ clients according to the distribution function $Dir(\alpha)$ (see Section IV-B).
3) The clients define their local validation datasets (10%).
4) The central server starts and controls the hyperparameter search as described in Section III-C for a total of $r \in \{50, 100\}$ rounds, where $r = 50$ when preprocessing is not applied as the search space is smaller in this case.

All optimal parameters can be found in Table I.

2) Calculation of Evaluation Metrics: With the fixed set of hyperparameters $\hat{\xi}$, the federated training and evaluation are repeated for ten evaluation rounds with different train-test splits. Specifically, in one evaluation round the following steps are executed subsequently:

1) Execute steps 1)-3) from the previous Section IV-C1.
2) Train the federated fuzzy clusterwise regression model as described in Sections III-A and III-B.
3) Calculate MAPE, R^2, and RMSE on the global test set.

[3]https://github.com/stallmo/fed-fuzzy-clusterwise-regression.

Dataset	FedFCR			FL-FRT [7]		eGauss+ Regr. [8]		Expl. Fuzzy Regr. [6]	
	MAPE	R^2	RMSE	R^2	RMSE	R^2	RMSE	R^2	RMSE
Mortgage	0.001	0.999	0.112	(0.979)	0.121 (0.430)	1.000	0.12	-	0.12
Treasury	0.018	0.996	0.223	(0.958)	0.206 (0.634)	0.995	0.24	-	0.40
Weather Iz.	0.016	0.992	1.26	(0.990)	1.27 (1.394)	0.993	1.22	-	1.19
Delta Elev. RMSE $\times 10^{-3}$	0.650	0.613	1.48	-	1.43	-	-	-	-
Elevators RMSE $\times 10^{-3}$	0.100	0.770	3.21	-	2.65	-	-	-	-
House RMSE $\times 10^{4}$	0.463	0.442	3.92	-	3.77	-	-	-	-
California RMSE $\times 10^{4}$	0.291	0.663	6.71	-	5.87	-	-	-	6.95

TABLE II

COMPARISON IN THE IID SCENARIO WITH OTHER FEDERATED REGRESSION METHODS' PREDICTION ACCURACY AS REPORTED BY THE AUTHORS, WHERE VALUES IN PARENTHESES WERE REPORTED BY [8].

Each evaluation round's test results are recorded, and we report the average over all rounds (see Tables II). Depending on whether the performance was better with or without preprocessing, the better results are reported in Table II.

D. Results

1) Impact of Preprocessing: In four out of seven datasets, applying preprocessing leads to better performance, where the performance gain is significant in two cases (Elevators and House datasets). Surprisingly, we observe a slight performance drop in the remaining three datasets.

2) IID Scenario: In the IID scenario, we compare our method's performance with recent fuzzy regression methods [6]–[8] on seven different datasets from the KEEL repository.

All related methods are evaluated on three datasets (Mortgage, Treasury, Weather Izmir). The differences in performance is almost negligible. Note that in all three datasets the performance is extraordinarily high (R^2> 0.99) so that these problems can be considered easier than the remaining ones

For the more challenging problems (Delta Elevators, Elevators, House), only [7] (FL-FRT) reports results. This method requires multiple rounds of communication, but also achieves a better prediction accuracy on all three datasets. The differences range from around 3%-4% lower RMSE on the Delta Elevator and House datasets to around 21% better RMSE on the Elevators dataset. That is the largest performance difference we observe in our experiments. On the California dataset, FL-FRT performs best as well (by around 14%). Our method FedFCR achieves a better performance than [6] and is second.

Overall, we observe that our FedFCR method performs similarly to other recent methods in the IID scenario, but does not always achieve the best performance. For an overview of all results, see Table II.

In addition to the predictive performance, we also note that all optimal hyperparameters vary across the datasets, showing the need for federated hyperparameter optimization (Table I).

3) Non-IID Scenario: Another goal of the FedFCR method is robustness against non-IID data. To study the impact of non-IID data, we repeat the evaluation phase described in Section IV-C2 with the same optimal hyperparameters as in the IID case (Table I), i.e., without rerunning the hyperparameter optimization. All results on the same seven benchmark datasets can be found in Table III.

Generally, the impact of non-IID data is small in our experiments. In the "easy" Mortgage, Treasury, and Weather Izmir datasets, the impact in absolute terms is almost negligible even though it may seem large in relative terms.

Dataset	IID			Non-IID			Non-IID impact in %		
	MAPE	R^2	RMSE	MAPE	R^2	RMSE	MAPE	R^2	RMSE
Mortgage	0.010	0.999	0.113	0.011	0.998	0.126	-9.93%	-0.03%	-12.09%
Treasury	0.018	0.996	0.223	0.021	0.994	0.255	-19.00%	-0.19%	-14.38%
Weather Iz.	0.016	0.992	1.270	0.016	0.993	1.240	1.33%	0.04%	2.37%
Delta Elev. RMSE $\times 10^{-3}$	0.651	0.613	1.482	0.629	0.602	1.502	3.32%	-1.82%	-1.34%
Elevators RMSE $\times 10^{-3}$	0.100	0.770	3.217	0.100	0.751	3.339	0.05%	-2.52%	-3.78%
House RMSE $\times 10^{4}$	0.463	0.442	3.924	0.478	0.413	4.025	-3.19%	-6.67%	-2.56%
California RMSE $\times 10^{4}$	0.291	0.663	6.711	0.296	0.651	6.816	-1.57%	-1.73%	-1.56%

TABLE III

COMPRISON OF PERFORMANCE IN IID AND NON-IID SCENARIOS.

In the remaining datasets, the impact is also limited and remains in the range of $\pm 4\%$ relative decrease, with one exception (R^2 score on the House dataset). While we do observe better performance in some instances (i.e., in the Izmir, Delta Elevator, and Elevator datasets), we note that in the majority of test cases the predictive performance decreases. The highest relative performance decrease can be observed in the House dataset (ignoring the easy datasets). Here, the MAPE increased from 0.463 to 0.478, the R^2 score decreased from 0.442 to 0.413 and the RMSE increased from 3.924 to 4.025. All in all, we observe fairly stable performance under the non-IID assumption, but notice that our method is not completely immune to heterogeneity in the distributed data.

E. Discussion

1) Preprocessing: The impact of preprocessing was significant in two datasets: Elevators and House. In both datasets the scale of features is widely different, such that data normalization is required to meaningfully cluster the data. Recall that in the clustering phase, we optimize J_m (Section III-B1) that uses the Euclidean distance and, hence, is strongly impacted by features of different scale.

Interestingly, we observe a slight performance drop after applying preprocessing in the "easy" datasets Mortgage, Treasury, and Weather Izmir. There are two reasons: First, the feature scales are not as different as in the House and Elevators datasets, such that the clustering does not benefit from normalization as much. Second, since we enforce the application of clipping, we suspect that this may lead to loss of information that the learning algorithm needs to improve its performance.

2) IID Scenarios: We note that the communication-efficient FedFCR can achieve the same performance as other recent federated fuzzy regression methods with fewer rounds of communication on easy datasets, and performs comparably but less accurately on other datasets.

These findings lead us to the following suggestions: If a use case's top priority is predictive performance without any constraints on the number of communication rounds, we suggest using the FL-FRT method. However, if communication is costly or constrained, or the datasets are known to be "easy" to solve, we believe that FedFCR is a particularly effective option, especially if the data are also expected to be non-IID. Lastly, if the use case requires continuously updating clusters with streaming data, [8] may be a good choice.

3) Non-IID Scenarios: FedFCR is the only method that was explicitly evaluated in non-IID scenarios. Overall, we observe good (but not perfect) robustness against non-IIDness.

We offer an explanation for the small but noticeable impact that can guide further research. Since the optimization problems (2) are solved exactly, we assume the source of the difference is the clustering, i.e., a difference in the membership matrix U. In [26], [35], it was already observed that non-IIDness can lead to a better separation in federated clustering. That is, cluster centers can be farther apart in the federated non-IID case than in the IID case. As a consequence, the same data points have higher membership to their closest cluster in the non-IID case, the same effect as decreasing fuzziness m. That creates a new situation where the optimal IID parameters may no longer be optimal in the non-IID case.

V. Concluding Remarks

To address key concerns in federated learning adoption (communication overhead, robustness against non-IIDness, data quality, hyperparameter optimization), this work introduces a novel federated fuzzy clusterwise regression (FedFCR) method, as well as federated protocols for preprocessing and hyperparameter optimization. FedFCR only requires two rounds of communication, and its predictive performance is comparable to other recently introduced federated fuzzy regression methods. However, we acknowledge that other (less communication-efficient) methods achieve better predictive performance on some benchmark datasets. Moreover, we observe good robustness against non-IID data, but not immunity.

Future research directions include the improvement of predictive performance in IID and non-IID scenarios, e.g., by merging cluster and regression model learning into a single stage with the same objective, or by choosing more appropriate hyperparameters in the non-IID case.

References

[1] P. Kairouz *et al.*, "Advances and open problems in federated learning," *Found. Trends Mach. Learn.*, vol. 14, no. 1–2, p. 1–210, Jun. 2021.

[2] A. Imteaj *et al.*, "A Survey on Federated Learning for Resource-Constrained IoT Devices," *IEEE Internet of Things Journal*, vol. 9, no. 1, pp. 1–24, Jan. 2022.

[3] D. C. Nguyen *et al.*, "Federated Learning for Smart Healthcare: A Survey," *ACM Computing Surveys*, vol. 55, no. 3, pp. 60:1–60:37, Feb. 2022.

[4] Y. Fu *et al.*, "A Secure Personalized Federated Learning Algorithm for Autonomous Driving," *IEEE Transactions on Intelligent Transportation Systems*, vol. 25, no. 12, pp. 20 378–20 389, Dec. 2024.

[5] A. A. Purkayastha *et al.*, "Federated Learning for Predictive Maintenance: A Survey of Methods, Applications, and Challenges," in *2024 MWSCAS*. IEEE, Aug. 2024, pp. 238–242.

[6] J. L. Corcuera Barcena *et al.*, "An Approach to Federated Learning of Explainable Fuzzy Regression Models," in *2022 FUZZ-IEEE*. IEEE, Jul. 2022, pp. 1–8.

[7] J. L. Corcuera Bárcena *et al.*, "Increasing trust in AI through privacy preservation and model explainability: Federated Learning of Fuzzy Regression Trees," *Information Fusion*, vol. 113, p. 102598, Jan. 2025.

[8] M. Ožbot *et al.*, "Evolving Gaussian Systems as a Framework for Federated Regression Problems," *IEEE Transactions on Fuzzy Systems*, vol. 33, no. 10, pp. 3736–3746, Oct. 2025.

[9] H. R. Roth *et al.*, "Federated Learning for Breast Density Classification: A Real-World Implementation," in *Domain Adaptation and Representation Transfer, and Distributed and Collaborative Learning*, S. Albarqouni *et al.*, Eds. Springer, 2020, vol. 12444, pp. 181–191.

[10] A. Hard *et al.*, "Federated Learning for Mobile Keyboard Prediction," 2018.

[11] T. Müller *et al.*, "Revealing the Impacting Factors for the Adoption of Federated Machine Learning in Organizations," in *Hawaii International Conference on System Sciences 2024*, 2024.

[12] T. Z. Sana *et al.*, "Advancing Federated Learning: A Systematic Literature Review of Methods, Challenges, and Applications," *IEEE Access*, vol. 13, pp. 153 817–153 844, 2025.

[13] T. Müller *et al.*, "A Pathway for the Practical Adoption of Federated Machine Learning Projects," in *2023 PACIS*, 2023.

[14] D. Kreuzberger *et al.*, "Machine Learning Operations (MLOps): Overview, Definition, and Architecture," May 2022.

[15] A. Gonzalez *et al.*, "Dealing with uncertainty and imprecision by means of fuzzy numbers," *International Journal of Approximate Reasoning*, vol. 21, no. 3, pp. 233–256, Aug. 1999.

[16] S. Guillaume, "Designing fuzzy inference systems from data: An interpretability-oriented review," *IEEE Transactions on Fuzzy Systems*, vol. 9, no. 3, pp. 426–443, Jun. 2001.

[17] A. Wilbik *et al.*, "Towards a Federated Fuzzy Learning System," in *2021 FUZZ-IEEE*. IEEE, Jul. 2021, pp. 1–6.

[18] K. Erenturk, "A Novel Fuzzy Inference System-Based Federated Learning Approach," *UNEC Journal of Computer Science and Digital Technologies*, 2025.

[19] B. Pękala *et al.*, "A Novel Method for Human Fall Detection Using Federated Learning and Interval-Valued Fuzzy Inference Systems," *Journal of Artificial Intelligence and Soft Computing Research*, vol. 15, no. 1, pp. 77–90, Jan. 2025.

[20] T. Le Vinh *et al.*, "Federated Learning-Based Trust Evaluation With Fuzzy Logic for Privacy and Robustness in Fog Computing," *IEEE Access*, vol. 13, pp. 137 952–137 972, 2025.

[21] Miin-Shen Yang *et al.*, "On cluster-wise fuzzy regression analysis," *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, vol. 27, no. 1, pp. 1–13, Feb. 1997.

[22] J. Viaña *et al.*, "Explainable fuzzy cluster-based regression algorithm with gradient descent learning," *Complex Engineering Systems*, vol. 2, no. 2, p. 8, 2022.

[23] G. Aggarwal *et al.*, "Secure Computation of the Median (and Other Elements of Specified Ranks)," *Journal of Cryptology*, vol. 23, no. 3, pp. 373–401, Jul. 2010.

[24] A. Tueno *et al.*, "Secure Computation of the k-th-Ranked Element in a Star Network," in *Financial Cryptography and Data Security*, J. Bonneau *et al.*, Eds. Springer, 2020, vol. 12059, pp. 386–403.

[25] G. Chandran *et al.*, "Comparison-based MPC in Star Topology," in *19th International Conference on Security and Cryptography*. SCITEPRESS - Science and Technology Publications, 2022, pp. 69–82.

[26] M. Stallmann *et al.*, "On a Framework for Federated Cluster Analysis," *Applied Sciences*, vol. 12, no. 20, p. 10455, Jan. 2022.

[27] H. Zhu *et al.*, "Federated learning on non-IID data: A survey," *Neurocomputing*, vol. 465, pp. 371–390, Nov. 2021.

[28] J. Snoek *et al.*, "Practical Bayesian Optimization of Machine Learning Algorithms," in *Advances in Neural Information Processing Systems*, F. Pereira *et al.*, Eds., vol. 25. Curran Associates, Inc., 2012.

[29] T. Akiba *et al.*, "Optuna: A Next-generation Hyperparameter Optimization Framework," in *25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*. ACM, Jul. 2019, pp. 2623–2631.

[30] R. Turner *et al.*, "Bayesian Optimization is Superior to Random Search for Machine Learning Hyperparameter Tuning: Analysis of the Black-Box Optimization Challenge 2020," 2021.

[31] J. Bergstra *et al.*, "Algorithms for Hyper-Parameter Optimization," in *Advances in Neural Information Processing Systems*, J. Shawe-Taylor *et al.*, Eds., vol. 24. Curran Associates, Inc., 2011.

[32] S. Watanabe, "Tree-structured Parzen estimator: Understanding its algorithm components and their roles for better empirical performance," *ArXiv*, vol. abs/2304.11127, 2023.

[33] J. Alcala-Fdez *et al.*, "KEEL Data-Mining Software Tool: Data Set Repository, Integration of Algorithms and Experimental Analysis Framework," *Journal of Multiple-Valued Logic and Soft Computing*, vol. 17, pp. 255–287, 2010.

[34] Q. Li *et al.*, "Federated Learning on Non-IID Data Silos: An Experimental Study," in *2022 IEEE ICDE*. IEEE, May 2022, pp. 965–978.

[35] D. K. Dennis *et al.*, "Heterogeneity for the win: One-shot federated clustering," in *38th International Conference on Machine Learning*, M. Meila *et al.*, Eds., vol. 139. PMLR, 18–24 Jul 2021, pp. 2611–2620.