An Approach to Federated Learning of Explainable Fuzzy Regression Models

José Luis Corcuera Bárcena, Pietro Ducange, Alessio Ercolani, Francesco Marcelloni, Alessandro Renda,
Department of Information Engineering, University of Pisa, Largo Lucio Lazzarino 1, 56122 Pisa, Italy
Email: {pietro.ducange, francesco.marcelloni, alessandro.renda}@unipi.it, joseluis.corcuera@phd.unipi.it

***Abstract*—Federated Learning (FL) has been proposed as a privacy preserving paradigm for collaboratively training AI models: in an FL scenario data owners learn a shared model by aggregating locally-computed partial models, with no need to share their raw data with other parties. Although FL is today extensively studied, a few works have discussed federated approaches to generate explainable AI (XAI) models. In this context, we propose an FL approach to learn Takagi-Sugeno-Kang Fuzzy Rule-based Systems (TSK-FRBSs), which can be considered as XAI models in regression problems. In particular, a number of independent data owner nodes participate in the learning process, where each of them generates its own local TSK-FRBS by exploiting an ad-hoc defined procedure. Then, these models are forwarded to a server that is responsible for aggregating them and generating a global TSK-FRBS, which is sent back to the nodes. An appropriate aggregation strategy is proposed to preserve the explainability of the global TSK-FRBS.**

A thorough experimental analysis highlights that the proposed approach brings benefits, in terms of accuracy, to data owners participating in the federation preserving the privacy of the data. Indeed, the accuracy achieved by the global TSK-FRBS is higher than the ones of the TSK-FRBSs learned by exploiting only local training data.

***Index Terms*—TSK fuzzy system, federated learning, explainability, regression**

I. Introduction

The ever increasing pervasiveness of Artificial Intelligence in the daily process of individuals, companies and institutions stimulated awareness, among users and regulatory bodies, of the importance of adherence to certain widely acknowledged ethical principles. The European Commission, for example, has promoted the creation of the "Ethic Guidelines for Trustworthy AI" [1], which describes the requirements an AI system must meet to achieve trustworthiness. Specifically, the privacy requirement is considered paramount for the data owners, who are often reluctant to share their data to other parties. Evidently, however, this can be an impediment to create accurate and reliable AI models, as they are typically data-hungry in their learning stage. Since data collection for centralized training is therefore not viable as it violates the privacy requirement, alternative paradigms for decentralized learning have been recently proposed. In particular, Federated Learning (FL) [2] has gained interest as it enables collaborative training of an AI model through the aggregation of locally-computed update, without disclosure of private data from the involved participants. Thus, remodeling, adapting, and analyzing traditional AI algorithms in a federated fashion represents one of the most compelling challenges in the current AI research landscape.

An equally important key aspect for users' trust in AI systems is the ability to understand how the model works, and to know the reasons that led to a certain output: the branch of Explainable AI (XAI) is concerned with these aspects and is deemed crucial for the practical deployment of AI systems on a large scale. In this context, the adoption of inherently interpretable models can play a key role: rule-based systems, for instance, are generally considered more inherently interpretable than other commonly employed models, such as random forests and deep neural networks, since the inference process is very much akin to the one used in human reasoning. Fuzzy Rule-Based Systems (FRBSs) feature even higher interpretability thanks to the linguistic representation of numerical variables and have proven to achieve competitive levels of performance for classification and regression tasks [3].

In this paper, we propose a federated approach to learn XAI models (Fed-XAI): different users (also referred to, in this work, as clients or data owners) participate in collaborative learning of an XAI model thus benefiting from the knowledge coming from the other participants without, however, exposing their own raw data. Specifically, we consider the federated learning of an adapted version of the first-order Takagi-Sugeno-Kang FRBS (TSK-FRBS) [4], which has proved to be effective for modelling complex systems in regression and control tasks and consists of a set of *if-then* rules in which the consequent part is a linear combination of the input variables.

In a nutshell, our proposal can be summarized as follows: each data owner learns a modified TSK-FRBS from local data: we revisit the traditional approach for building TSK models to pursue high level of interpretability. Then, data owners share the model with one central server, which merges the received models to produce a global TSK-FRBS. The TSK-FRBS is finally sent back to the data owners that can use it for local inference. Our work entails the following contributions:

- we propose a novel approach for building highly interpretable TSK-FRBS;
- we define a novel aggregation strategy for federated learning of TSK-FRBS, so that different participants can collaborate to learn a global model without sharing private data;
- we demonstrate the effectiveness of the proposed strategy for Fed-XAI with a thorough experimental analysis.

The rest of the paper is organized as follows: Section II describes some related works. Section III provides some preliminaries on the problem statement for the federated setting and on TSK-FRBSs. In Section IV we describe our modified interpretable TSK-FRBS, whereas Section V describes the proposed approach for federated learning of such a model. Section VI describes the experimental setup and results. In Section VII, we draw some conclusions.

II. Related Works

The literature related to TSK-FRBSs is extremely extensive, starting from their introduction in the 1980s. Since we cannot exhaustively cover all significant contributions, we just review the works that have recently discussed the interpretability of TSK-FRBSs and their adaptation to the FL setting.

A remarkable contribution for the design of FRBSs has been recently presented in [5]: specifically, authors introduced PyFUME, a Python library for the estimation of antecedent and consequent parameters of an FRBS. Concerning their transparency, these models are labeled by the authors as "(light) grey box" models: on one side, in fact, linguistic fuzzy rules are easily comprehensible to human beings; on the other side, however, the procedure adopted for the estimation of the antecedent parameters substantially undermines the interpretability of the whole system. Such a procedure, described in [6], exploits clustering for partitioning data in the input-output product space and estimates antecedent parameters by fitting the convex envelop of the projected membership values for each discovered cluster. Compared to the traditional clustering-based approach [7], the procedure implemented in PyFUME pursues more specific membership functions (through the removal of outlying cluster membership values), but it still exhibits the following problem: inevitably, the estimated membership functions will not meet the criteria, generally deemed crucial for interpretability of FRBSs [8], of coverage, completeness, distinguishability and complementarity, as they are automatically derived from data.

Since early works in FL literature [9], [10], most solutions revolve around the original proposal of Federated Averaging (FedAvg), as a protocol for executing Stochastic Gradient Descent (SGD) in a federated manner. Specifically, in [9] the authors showed that deep neural network models can be collaboratively trained for tackling image classification and language modeling tasks. However, the adoption of FL for training inherently interpretable models has not yet been adequately studied. To the best of our knowledge the only work in this direction has been recently published by Zhu et al. [11], who proposed an approach for federated learning of TSK-FRBSs for horizontally partitioned data. The approach involves two steps: the first step consists in collaborative structure identification based on federated FCM clustering [12], [13]; once the fuzzy sets are determined, the second step consists in collaborative estimating the local consequents of each rule, each associated with a cluster. Both steps are inspired by the FedAvg approach: for both FCM centroids and rule consequents parameters, the gradients of related and appropriate cost functions are evaluated by each client based on its local data; then, gradients are transmitted to the central server, which is in charge of aggregating the gradients to update the model parameters accordingly and of sharing the results with the clients. The approach has proved to be effective in modelling some real-world datasets, but the resulting model cannot be considered highly interpretable since the estimation of antecedents parameters is data driven and the inference process combines the implications of all the activated rules, as in the classical TSK-FRBS. Furthermore, it requires careful setting of several hyperparameters, including the learning rate and the maximum number of iterations, which can have a strong impact on model convergence.

Notably, our proposed approach differs from [11] and from the classical FL paradigm in two aspects: first, it entails a *one-shot* communication scheme and not an iterative, gradient-based, algorithm. Second, merging rule-based models requires defining appropriate procedures, necessarily different from the weighted average of models or gradients tensors carried out within FedAvg and its variants.

III. Background

In this section, we first describe the problem setting and introduce the notation for the FL scenario. Then, we provide some preliminaries about TSK-FRBSs.

A. Federated Learning problem statement

Let $\{C_m\}_{m=1}^{M}$ be M parties, i.e. data owners, who wish to train an AI model by consolidating their respective data $\{(\mathbf{x}_i^1, y_i^1)\}_{i=1}^{N_1}, \{(\mathbf{x}_i^2, y_i^2)\}_{i=1}^{N_2}, \ldots, \{(\mathbf{x}_i^M, y_i^M)\}_{i=1}^{N_M}$. In an FL process the parties collaboratively learn a model under the orchestration of a central server and without exposing their private data to others. In this work, we focus on the scenario of horizontally partitioned data: the M private datasets may have different sizes but their instances are represented in the same F-dimensional attribute space. We assume that the domain of definition of the attributes are known a-priori, and they are also known to the server.

B. The TSK-FRBS models

This section describes the basics of TSK-FRBSs: for the sake of clarity and to avoid burdening the notation we omit the index $m = \{1 \ldots M\}$, although in this work the models are built based on *local* datasets $\{C_m\}_{m=1}^{M}$.

Let $X = \{X_1, X_2, \ldots, X_F\}$ be a set of input variable and Y the output variable. A generic record of the dataset is in the form $\mathbf{x} = [x_1, x_2, \ldots, x_F]^T$ and has an associated target value y. Let U_f, $f = 1, 2, \ldots, F$, be the universe of discourse of variable X_f and $P_f = \left\{A_{f,1}, A_{f,2}, \ldots, A_{f,T_f}\right\}$ be a fuzzy partition over U_f with T_f fuzzy sets, each labeled with a linguistic term. Finally, let K be the number of rules in the rule base. The generic k^{th} rule is in the form:

$$R_k : \textbf{IF } X_1 \textit{ is } A_{1,j_{k,1}} \textbf{ AND } \ldots \textbf{ AND } X_F \textit{ is } A_{F,j_{k,F}} \quad \textbf{THEN } y_k = \gamma_{k,0} + \sum_{i=1}^{F} \gamma_{k,i} \cdot x_i \tag{1}$$

where $j_{k,n} \in [1, T_f]$ identifies the index of the fuzzy set of partition P_f. In first order TSK-FRBS the consequent function of the generic rule R_k is a linear combination of the elements of $\mathbf{x}$, parameterized by the vector of coefficients $\boldsymbol{\gamma_k} = \{\gamma_{k,0}, \gamma_{k,1}, \ldots \gamma_{k,F}\}$.

The most popular approach to learn TSK-FRBS consists of two stages: structure identification and model parameter identification. In the former stage, the number of rules and the conditional part of the rules are determined; this is typically done either with grid-partitioning of the input space or exploiting fuzzy clustering methods [14]. In the latter stage, with fixed antecedents, parameters of the local linear models are learned by pseudo-inversion or by applying the recursive least square method. Alternative approaches have been proposed for optimizing TSK-FRBSs, including genetic algorithms [15] or mini batch gradient descent [16], [17].

When an input pattern $\mathbf{x}_i$ is fed to a TSK-FRBS, the strength of activation of each rule is computed as follows:

$$w_k(\mathbf{x}) = \prod_{f=1}^{F} \mu_{f,j_{k,f}}(x_f) \quad \text{for} \quad k = 1, \ldots, K \tag{2}$$

where $\mu_{f,j_{k,f}}(x_f)$ is the membership degree of x_f to the fuzzy set $A_{f,j_{k,f}}$. In the traditional TSK-FRBSs, the inference process generates an output as the weighted average of the K outputs obtained from as many rules. Formally:

$$\hat{y}(\mathbf{x}) = \sum_{k=1}^{K} \left(\frac{w_k(\mathbf{x})}{\sum_{h=1}^{K} w_h(\mathbf{x})} \right) \cdot y_k(\mathbf{x}) \tag{3}$$

IV. Enforcing Interpretability in TSK-FRBSs

The proposed approach for generating a first-order TSK-FRBS consists of rules antecedent generation and consequent parameters estimation. As detailed in Section V, the procedure is executed locally by all clients participating in the FL process. Although in this work we refer to the federated scenario, the procedure described in this section is general.

A. Rule antecedent generation

The rule antecedent generation step is inspired by the Wang & Mendel approach [18] and encompasses three steps: (i) fuzzy partitioning of the input space, (ii) numerosity reduction through fuzzy clustering of training data, and (iii) generation of antecedents based on centroids.

First, a strong triangular uniform fuzzy partitioning is defined on each normalized input attribute f using T_f fuzzy sets. In the rest of the paper we will consider the same granularity for all inputs, and specifically $T_f = T = 3$: this firstly induces less complex models and moreover guarantees a high level of semantic interpretability thanks to the adoption of just three linguistic terms for labelling the fuzzy sets: *Low*, *Medium* and *High*. From the point of view of inherent interpretability, there is another fundamental difference with the standard approaches adopted for structure identification: the use of strong uniform fuzzy partitions ensures high interpretability since they satisfy the properties of coverage, completeness, distinguishability and complementarity; on the other hand, when fuzzy sets are generated through clustering, the modeling of the data distribution can be more accurate, but the above properties are not necessarily satisfied.

The antecedents are generated similarly to the Wang & Mendel's approach: however, instead of determining the membership degree of all training samples, we first execute the Fuzzy *C*-means clustering algorithm [19] on the training samples and then we just consider the cluster centers for assessing their membership degree to the fuzzy sets defined over the input attributes. A toy example, featuring a unique input attribute Z and an output variable Y, is reported in Fig. 1: the FCM algorithm is executed on the $Z \times Y$ input-output product space; cluster centers, represented as red dots, are projected onto the input space and their membership degree to the predetermined fuzzy sets is assessed.

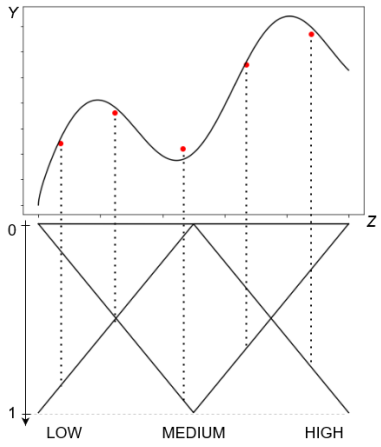

Fig. 1. Rules antecedent generation on an 1-dimensional toy dataset. (top) Training set summarization obtained by the execution of the FCM algorithm - red dots indicate cluster centers. (bottom) Cluster centers projection onto the input attribute partitioned in three fuzzy sets.

For a given input component, a *condition* is added by considering the fuzzy set with the maximum membership degree; for multi-dimensional input, this procedure is repeated for each input component and the conditions are joined with the AND operator to generate the antecedent part of a rule.

In the example of Fig. 1, three antecedents (i.e., three rules) are obtained:

$$\begin{aligned} &Antecedent1 : \textbf{IF } x \; is \; Low \\ &Antecedent2 : \textbf{IF } x \; is \; Medium \\ &Antecedent3 : \textbf{IF } x \; is \; High \end{aligned} \tag{4}$$

Let C_{fcm} be the number of clusters used in the FCM procedure, F the number of input attributes and T the number of fuzzy sets per attribute. The upper bound for the number of rules K generated by the above procedure is:

$$K \leq min(T^F, C_{fcm}) = K_{max} \tag{5}$$

The clustering therefore helps to summarize the training set and to limit the overall number of rules, especially when the dimensionality of the dataset is high. The choice of parameter

C_{fcm} entails a trade-off between complexity and modelling power. Indeed, low values of C_{fcm} reduce the complexity, resulting in a higher global interpretability and lower communication and computational costs, but may increase the error.

B. Consequent parameter estimation

The first-order consequents corresponding to the K antecedents are determined in the consequent parameter estimation step. We adopt the local least-square approach as in [5]: for each k^{th} antecedent, consequent parameters $\boldsymbol{\gamma_k} = \{\gamma_{k,0}, \gamma_{k,1}, \dots \gamma_{k,F}\}$ are estimated with a weighted least-squared method. Specifically, each training sample is weighted by its strength of activation of the rule (Eq. 2).

C. Inference process and rule weight

Given an input pattern, the inference process of the *traditional* TSK-FRBS consists in the evaluation of the weighted average of the outputs inferred from the rules, as presented in Eq. 3. This undermines the interpretability of the model: in fact, the coefficients used in the linear model for calculating the output will be different for any different input pattern. With the aim of enhancing interpretability, we adopt the *maximum matching* inference rule strategy: the output of the system is determined by using the rule with the highest strength of activation. Since the antecedent of this rule is expressed linguistically, a user can easily understand which combination of input values has determined the output.

Furthermore, we associate a weight with each rule. When more than one rule is activated with the same largest strength, we select the rule with the highest weight. Further, in case no rule is activated by the input pattern, the rule with the highest weight is used to produce an output (default strategy). The *rule weight* RW_k associated with each rule R_k is determined by the following procedure.

Let $AE_k(\mathbf{x}, y) = |y - y_k(\mathbf{x})|$ be the absolute error evaluated based on the value $y_k(\mathbf{x})$ predicted for the generic input training sample $(\mathbf{x}, y)$ considering rule R_k. Let AE_{min} and AE_{max} be the minimum and the maximum values over the errors obtained considering all the activated rules for all the samples in the training set. For each sample and for each rule, we evaluate the *quality of prediction* $\mu(AE_k(\mathbf{x}, y))$ according to the following function:

$$\mu(AE_k(\mathbf{x}, y)) = 1 - \frac{AE_k(\mathbf{x}, y) - AE_{min}}{AE_{max} - AE_{min}} \tag{6}$$

. Intuitively, the lower the error, the higher the quality of prediction.

Let TS be the training set and N its cardinality. We compute the *fuzzy confidence* and the *fuzzy support* of a rule R_k as follows:

$$Conf_k = \frac{\sum_{(\mathbf{x},y)\in TS} w_k(\mathbf{x}) \cdot \mu(AE_k(\mathbf{x}, y))}{\sum_{(\mathbf{x},y)\in TS} w_k(\mathbf{x})} \tag{7}$$

$$Supp_k = \frac{\sum_{(\mathbf{x},y)\in TS} w_k(\mathbf{x}) \cdot \mu(AE_k(\mathbf{x}, y))}{N} \tag{8}$$

Fuzzy confidence can be regarded as the average quality of prediction associated with the training samples, each weighted by the strength of activation of the rule. Fuzzy support differs from fuzzy confidence only in the denominator and it gets higher values when a larger number of instances activate the rule and result in high prediction quality.

Finally, we define the *rule weight* RW_k as the harmonic mean of fuzzy confidence and support. Formally,

$$RW_k = 2 \times \frac{Supp_k \times Conf_k}{Supp_k + Conf_k} \tag{9}$$

Notably, the rule weight will not only be used for driving the default strategy, but also for the rule aggregation step in the proposed approach for federated learning of TSK-FRBS. A thorough description of such an approach is provided in the following section.

V. Proposed approach for Federated Learning of a global TSK-FRBS

A schematic view of the proposed approach is shown in Fig. 2. The overall approach encompasses the following steps.

- ***Communication step A***: configuration of the learning process;
- ***Step 1***: local learning of TSK-FRBSs;
- ***Communication step B***: transmission of local models to the central server;
- ***Step 2***: federated learning of the global TSK-FRBS: aggregation of the models;
- ***Communication step C***: transmission of the aggregated model to the clients;

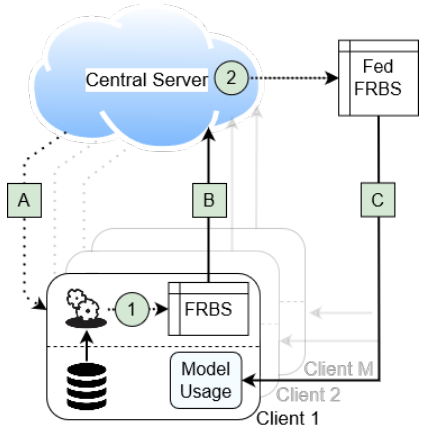

Fig. 2. Overview of the proposed approach. Squared markers (A, B, C) denote *communication* steps. Circle markers denote *local learning* (1) and *model aggregation* (2) steps.

At the beginning (***A***) the central server configures the learning process, by sending a set of hyperparameters to each data owner. Such set includes (i) the domain of definition of the attributes for data normalization, (ii) the number of fuzzy sets T_f $(f = 1, \dots, F)$ for fuzzy partitioning of input attributes, and (iii) the number of clusters C_{fcm} for the FCM algorithm..

Once received the hyperparameters, each client can train a model based on its local data (***Step 1***): local learning of the

first-order TSK-FRBSs is performed as described in Section IV. Notably, in order to make the downstream procedure of rules aggregation workable and effective, it is essential that all clients rely on the same partitioning of the input space. If this were not the case, a rule expressed in the form of Eq. 1 generated on a given client cannot be safely used on another client since the fuzzy sets might not coincide. Therefore, leveraging the information received from the server, every client normalizes input and output attributes in the range $[0, 1]$ and builds the local model starting from the same fuzzy partitioning of the input space. Then, each client transmits the local model to the server (***B***), which is in charge of generating the global model by aggregating the received local models. The aggregation step (***Step 2***) is described in detail in Section V-A. Finally the aggregated model, named as Fed-FRBS in Fig. 2, is transmitted to each client (***C***), where it is employed in place of the local model for the regression task.

A. Federated TSK-FRBS: aggregation of the local models

Let M be the number of clients that participate in the FL process. When the clients have sent the local models (communication step ***B***), the central server will have the information schematized in Table I, i.e. the juxtaposition of the rules collected from the M clients.

TABLE I
RULES COLLECTED BY THE CENTRAL SERVER FROM M PARTICIPANTS IN THE FL PROCESS. THE HORIZONTAL ARROW ($\Leftarrow$) IS USED TO INDICATE CONFLICTING RULES.

	Antecedent	***Consequent***	***Rule Weight***	
Client 1	$ant_{1,1}$	$cons_{1,1}$	$rw_{1,1}$	
	...	...	...	
	$ant_{1,i}$	$cons_{1,i}$	$rw_{1,i}$	$\Leftarrow$
	...	...	...	
	ant_{1,K_1}	$cons_{1,K_1}$	rw_{1,K_1}	
...	...			
Client m	$ant_{m,1}$	$cons_{m,1}$	$rw_{m,1}$	
	...	...	...	
	$ant_{m,j}$	$cons_{m,j}$	$rw_{m,j}$	$\Leftarrow$
	...	...	...	
	ant_{m,K_m}	$cons_{m,K_m}$	rw_{m,K_m}	
...	...			
Client M	$ant_{M,1}$	$cons_{M,1}$	$rw_{M,1}$	
	...	...	...	
	$ant_{M,k}$	$cons_{M,k}$	$rw_{M,k}$	
	...	...	...	
	ant_{M,K_M}	$cons_{M,K_M}$	rw_{M,K_M}	

Since this *global* rule base aggregates knowledge from different sources, it is likely that some *conflicting rules*, i.e. rules, originated from different clients, with the same antecedent but different consequents, occur. These conflicts have to be solved. As an example, suppose that the i-th rule of Client 1 and the j-th rule of Client m are conflicting rules, that is, $ant_{1,i} \equiv ant_{m,j}$ but the consequents are different; the two rules are marked with the symbol '$\Leftarrow$' in Table I.

We propose the following strategy for handling conflicts among rules. Let CR be the set of conflicting rules for a specific antecedent. Let $\vec{\Gamma}_l$ and rw_l be the consequent vector of coefficients and the rule weight of the l^{th} rule in CR, respectively. A single rule is obtained from CR as follows:

- the new *antecedent* is the same of the rules in CR;
- the coefficients of the new *consequent* ($\vec{\Gamma}$) are estimated as the weighted average of the coefficients of the consequents in CR, each weighted by the respective *rule weight*;
- the *rule weight* ($\widehat{rw}$) associated with the rule is computed as the average of the rule weights in CR.

Formally:

$$\vec{\Gamma} = \frac{\sum_{l=1}^{|CR|} \vec{\Gamma}_l \cdot rw_l}{\sum_{l=1}^{|CR|} rw_l} \tag{10}$$

$$\widehat{rw} = \frac{1}{|CR|} \cdot \sum_{l=1}^{|CR|} rw_l \tag{11}$$

Once all conflicts have been handled, the resulting rule base represents the *federated* model, that can be sent back to the clients for local inference (communication step ***C***).

VI. EXPERIMENTAL ANALYSIS

In this section, we first describe our experimental setup, including details regarding the datasets exploited, the simulation of the federated setting and the configuration parameters. Then, we report the results of the FL experiments. Finally, we also compare our proposed approach with the state-of-the-art proposal for building TSK FRBSs [5] discussed in Section II and implemented in PyFUME.

A. Experimental Setup

We employ four well-known regression datasets available within the KEEL-dataset repository [20], namely Weather Izmir, Treasury, Mortgage and California. A summary of the datasets is reported in Table II.

TABLE II
DATASETS DESCRIPTION

Dataset	**Abbreviation**	**Dimensionality (F)**	**Samples (N)**
Weather Izmir	WI	9	1461
Treasury	TR	15	1049
Mortgage	MO	15	1049
California	CA	8	20460

To simulate the distributed scenario, we randomly split each dataset in five parts (each with the same amount of instances), assuming the involvement of as many participants. Our main objective is to assess the performance of our proposed approach for federated learning of TSK-FRBSs; to this aim, we consider three scenarios:

- **Federated** model: we adopt the approach described in Section V: at the end of the aggregation process, each client tests the final global model locally.
- **Local** model: each client locally learns and tests its TSK-FRBS. This scenario does not entail any form of collaborative learning.

- **Centralized** model: each participant shares its training data with the central server which can build a TSK-FRBS using the *overall* training set, obtained by the union of the *local* training sets. The model is shared with the participants that test it locally. This scenario represents the ideal case where all data can be used for training, but of course violates the privacy requirement.

The quality of prediction of the FRBSs is evaluated through the *Mean Squared Error* (MSE):

$$MSE = \frac{1}{N_{test}} \sum_{i=1}^{N_{test}} (y_i - \hat{y}_i)^2 \tag{12}$$

where N_{test} is the number of samples considered for the evaluation, y_i and $\hat{y}_i$ are the ground truth value and the predicted value associated with the i-th instance of the test set, respectively. Results are evaluated in terms of average values over five-fold cross-validation: for a fair comparison, at each iteration of the cross-validation, the same *local* split is used for the three scenarios.

As per the hyperparameter configuration, we set the values consistently across the three scenarios as follows:

- $T_f = 3, \quad \forall f \in \{1, \ldots, F\}$, to ensure high semantic interpretability, as detailed in Section IV-A;
- $C_{fcm} = 30$, as the number of clusters set for the FCM algorithm, in order to summarize the training set and to limit the overall number of rules.

We have verified that the resulting FRBSs are not particularly sensitive to the choice of C_{fcm}. However, the automatic tuning of such a parameter represents an interesting future development of this work.

B. Federated Learning Experiments

Table III reports the results obtained with the proposed *federated TSK-FRBSs*. Results are compared with those obtained in the *local* and *centralized* scenarios. As 5-fold cross-validation is adopted, for each client and dataset, we report the average MSE on the test and training sets. The dataset partitioning over the clients is maintained also in the *centralized setting*, just for the purpose of performance assessment.

Table III suggests that federated scenario always outperforms, on average, the *local* one. This outcome is relevant since it demonstrates how, in the considered setting, every client can benefit from joining the FL process: the model built in a collaborative manner without sharing private raw data features a higher generalization capability than the locally built ones. On the other hand, intuitively, the *centralized* scenario, which employs the union of the five local training sets, achieves comparable or better performance than the *federated* one. While it is clear that the increased availability of training data is crucial for building more accurate models, it should be considered that gathering scattered data into a single server for centralized processing is not always feasible due to privacy concern or communication constraints. It is worth underlining that the gaps between the *centralized* and the *federated* scenarios are more evident for Treasury and Mortgage datasets:

TABLE III
EXPERIMENTAL RESULTS: FOR EACH DATASET AND SCENARIO (LOCAL, FEDERATED, CENTRALIZED), THE AVERAGE MSE OVER CROSS-VALIDATION IS REPORTED FOR EACH CLIENT, ALONG WITH THE OVERALL AVERAGE VALUES.

	Local		**Federated**		**Centralized**	
Client ID	*Train*	*Test*	*Train*	*Test*	*Train*	*Test*
Weather Izmir						
1	1.33	*2.02*	1.44	*1.57*	1.40	*1.54*
2	1.09	*1.62*	1.25	*1.41*	1.22	*1.34*
3	0.96	*1.40*	1.25	*1.32*	1.22	*1.29*
4	1.07	*7.10*	1.23	*1.30*	1.20	*1.28*
5	1.19	*1.64*	1.41	*1.51*	1.38	*1.46*
Avg.	1.13	*2.76*	1.32	*1.42*	1.28	*1.38*
Treasury ($\times 10^{-3}$)						
1	7.11	*377.40*	82.20	*112.72*	21.97	*46.13*
2	19.28	*192.70*	53.64	*79.41*	37.69	*51.35*
3	7.72	*337.25*	429.38	*174.18*	26.86	*41.97*
4	9.31	*110.47*	72.86	*378.61*	20.51	*41.69*
5	10.37	*133.83*	57.04	*40.85*	13.24	*20.37*
Avg.	10.76	*230.33*	139.02	*157.15*	24.06	*40.30*
Mortgage ($\times 10^{-3}$)						
1	2.29	*78.08*	9.70	*15.96*	5.20	*7.55*
2	1.44	*15.08*	9.14	*7.35*	3.47	*5.22*
3	1.22	*38.18*	14.61	*9.52*	3.31	*5.22*
4	1.54	*53.84*	9.38	*35.90*	4.24	*8.83*
5	1.09	*43.36*	14.78	*5.14*	3.74	*4.98*
Avg.	1.52	*45.71*	11.52	*14.77*	3.99	*6.36*
California ($\times 10^{9}$)						
1	4.73	*4.87*	4.75	*4.86*	4.77	*4.78*
2	4.62	*4.73*	4.57	*4.58*	4.60	*4.62*
3	4.71	*4.89*	4.71	*4.74*	4.72	*4.75*
4	4.77	*5.10*	5.23	*5.34*	5.18	*5.24*
5	4.70	*4.82*	4.63	*4.64*	4.65	*4.68*
Avg.	4.71	*4.88*	4.78	*4.83*	4.78	*4.81*

such datasets are characterized by a relatively high number (15) of features and a low number (1049, reduced to around 200 for each client) of instances. Indeed, we can argue that such a small number of instances used to estimate TSK parameters does not allow for accurate modeling of these datasets; although model aggregation improves performance compared to the local setting, it still delivers a worse generalization capability compared to the centralized case. The low data regime of the two datasets is also the underlying reason for the severe overtraining that affects the *local* approach: the average MSE measured on the training set is low (also compared to the *federated* and *centralized* scenarios) but the local models lack of generalization capability on the test sets. Conversely, with a larger dataset (i.e., California, 20460 instances) the relative differences of performance between the three approaches become smaller and also the overtraining phenomenon is less prominent. Notably the performance obtained on the four datasets are comparable to those reported in the literature [21].

We performed the pairwise Wilcoxon signed-rank test [22] to assess possible statistical differences in performances

among the three scenarios: for each dataset, the *federated* approach is selected as the control one and is compared with the *local* and the *centralized* ones. Each distribution consists of 25 values of MSE measured on the test sets, derived from the iterations of the cross-validation over the involved clients. Table IV reports the results of the test: $\mathbf{R}^+$ and $\mathbf{R}^-$ denote, respectively, the sum of ranks for the evaluations in which the federated model outperformed the other one, and the sum of ranks for the opposite outcome.

TABLE IV
RESULTS OF THE WILCOXON SIGNED-RANK TEST ON THE MSE VALUES OBTAINED ON THE TEST SETS.

DS	$\mathbf{R}^+$	$\mathbf{R}^-$	p-value	Hypothesis ($\alpha = 0.05$)
Federated vs Local				
WI	314	11	0.0000	Rejected (>)
TR	230	95	0.0710	Not Rejected (=)
MO	309	16	0.0000	Rejected (>)
CA	237	88	0.0451	Rejected (>)
Federated vs Centralized				
WI	91	234	0.0551	Not Rejected (=)
TR	5	320	0.0000	Rejected (<)
MO	24	301	0.0000	Rejected (<)
CA	231	94	0.0667	Not Rejected (=)

The statistical hypothesis of equivalence can be rejected whenever the p-value is lower than the level of significance α. Results suggest that, with $\alpha = 0.05$, the *federated* approach statistically outperforms the *local* model in three out of the four datasets. On the other hand it is outperformed by the *centralized* approach in the *small* datasets (TR and MO), but it achieves competitive performance on WI and CA. Notably, the equivalence hypothesis would always be rejected by relaxing the level of significance, e.g., with $\alpha = 0.1$.

Finally, it is worth underlining that a necessary but not sufficient condition for the *global* interpretability of an FRBS is the limited size of its rule base, i.e., a reduced complexity [23]. Thus, we measured the complexity of the three approaches as the average number of rules of the FRBSs. Table 5 reports the results. We can observe that, in the *federated* and *centralized* scenarios, the average number of rules is computed over five values (obtained by using the cross-validation procedure), whereas for the *local* scenario we averaged over the 25 values corresponding to as many local models.

TABLE V
MODEL COMPLEXITY: AVERAGE NUMBER OF RULES OF THE TSK-FRBSS.

Dataset	Local	Centralized	Federated
Weather Izmir (WI)	13.96	13.40	27.80
Treasury (TR)	21.36	21.20	42.40
Mortgage (MO)	21.60	21.00	46.00
California (Ca)	8.80	8.60	10.20

First of all, results confirm that the number of rules generated for a TSK-FRBSs, either learned locally or centrally, is always lower than the upper bound K_{max} (see Eq. 5) and also lower than C_{fcm}, i.e. the number of clusters used in the FCM procedure, arbitrarily set to 30. Furthermore, the average number of rules of the *local* and the *centralized* scenarios are very similar. The *federated* model is generally more complex than the local and centralized counterparts, due to the rules aggregation procedure, but its complexity is still limited and in the same order of magnitude.

The interpretability of our TSK-FRBSs can be explained as follows: the antecedent of a generic rule R_k identifies a specific region of the attribute space; within this region, the predicted output is evaluated as a linear combination of the input variables, which is expressed in the consequent part of the rule. The coefficient vector $\boldsymbol{\gamma_k} = \{\gamma_{k,0}, \gamma_{k,1}, \dots \gamma_{k,F}\}$ describes the effect of each attribute on the output value. In the following, a rule generated on the *california* dataset is shown as an illustrative example:

R_k: *IF longitude (x_1) is Low AND latitude (x_2) is Medium and housingMedianAge (x_3) is Medium AND totalRooms (x_4) is Low AND totalBedrooms (x_5) Low AND population (x_6) is Low AND households (x_7) is Low AND medianIncome (x_8) is Medium THEN* $medianHouseValue = 0.83 - 1.08 \cdot x_1 - 0.95 \cdot x_2 + 0.08 \cdot x_3 + 0.41 \cdot x_4 + 2.18 \cdot x_5 - 5.29 \cdot x_6 + 0.27 \cdot x_7 + 1.28 \cdot x_8$

Indeed, we can get that the house value is strongly influenced by number of bedrooms (increases with $x5$) and by the population (decreases with $x6$).

C. Comparison with a state-of-the-art approach

With the objective to enforce interpretability, we proposed a purposely designed approach to learn TSK-FRBSs. In this section, we show that this approach allows achieving performance similar to state-of-the-art approaches, regardless of the federated setting. We adopt the recently delivered PyFUME implementation of TSK-FRBS [5], [6] as comparison. We recall the two main differences among our approach and the PyFUME version:

- our approach considers a predefined strong uniform fuzzy partition over the input attributes, whereas PyFUME estimates the parameters of the fuzzy sets by executing a clustering algorithm and then fitting the convex envelop of the projected membership values for each cluster;
- our approach adopts an inference strategy based on the maximum matching, whereas PyFUME implements the classical averaging of all the activated rules (Eq. 3).

We compared our approach (TSK-SC, i.e. single consequent since we adopt a maximum matching inference rule) and PyFUME under the *centralized* setting. For PyFUME, we set the number of clusters so as to achieve a complexity comparable to the one of our system (number of rules of the centralized approach in Table V). We also evaluate the impact of the inference strategy adopted in our approach, by replacing the single consequent policy with the classical averaging (TSK-AC, i.e. averaging consequents). Table VI summarizes the results.

Results suggest that, at equal system complexity, the performance of our approach is comparable to, or even better than, the one obtained by PyFUME. Furthermore, the inference process based on averaging only slightly outperforms the

TABLE VI
COMPARISON OF OUR APPROACH (SC = SINGLE CONSEQUENTS, AC = AVERAGING CONSEQUENTS) AND STATE OF THE ART APPROACH (PYFUME) FOR BUILDING TSK FRBSS.

	TSK-SC		**TSK-AC**		**PyFUME [5], [6]**	
Dataset	Train	Test	Train	Test	Train	Test
WI	1.28	1.38	1.28	1.37	1.48	1.52
TR	24.06	40.30	24.42	39.18	32.07	62.93
MO	3.99	6.36	4.29	6.14	4.49	8.22
CA	4.78	4.81	4.82	4.85	4.62	4.64

one based on single consequent, although it entails a lower semantic interpretability.

VII. CONCLUSION

In this paper, a solution for Federated Learning (FL) of XAI models has been proposed. Specifically, we introduced a novel approach for aggregating first-order TSK-FRBSs learned locally in clients participating in the federation. The local models are sent to a central server which is in charge of aggregating them by resolving possible conflicts between rules. The TSK-FRBS we adopt is a variant purposely modified in order to achieve high level of interpretability: unlike the classical data driven approaches, our proposal relies on a uniform fuzzy partitioning of the input space; furthermore, a maximum matching inference rule is used for improving the interpretability. In our experimental analysis, we compared the proposed federated approach with the *local* learning (which entails no collaboration) and *centralized* learning (which entails centralization of raw data). Results, evaluated in terms of MSE on four regression datasets, show that the federated approach generally outperforms the local one. On the other hand, it is generally outperformed by the centralized approach, which, however, is unfeasible in privacy-sensitive applications. We also validated our variant of TSK-FRBS by comparing it with a recently proposed state-of-the-art classical TSK-FRBS learning. We observed that the interpretability gain ensured by our implementation does not decrease the accuracy.

Finally, as future work, we aim to address the main challenge of our approach, that is, devising an automated procedure for tuning the hyperparameters of our system, namely, the number C_{fcm} of clusters and the granularity T_f of the fuzzy partitions. We argue that this can be addressed by introducing a preliminary communication round in the FL process.

REFERENCES

[1] E. Commission, C. Directorate-General for Communications Networks, and Technology, *Ethics guidelines for trustworthy AI*. Publications Office, 2019.

[2] Q. Yang, Y. Liu, T. Chen, and Y. Tong, "Federated machine learning: Concept and applications," *ACM T. Intel. Sysy. Tec.*, vol. 10, no. 2, pp. 1–19, 2019.

[3] A. Fernandez, F. Herrera, O. Cordon, M. J. del Jesus, and F. Marcelloni, "Evolutionary fuzzy systems for explainable artificial intelligence: Why, when, what for, and where to?" *IEEE Comput. Intell. Mag.*, vol. 14, no. 1, pp. 69–81, 2019.

[4] T. Takagi and M. Sugeno, "Fuzzy identification of systems and its applications to modeling and control," *IEEE T SYST MAN CYB*, no. 1, pp. 116–132, 1985.

[5] C. Fuchs, S. Spolaor, M. S. Nobile, and U. Kaymak, "pyfume: a python package for fuzzy model estimation," in *2020 IEEE Int'l Conf. on fuzzy systems (FUZZ-IEEE)*. IEEE, 2020, pp. 1–8.

[6] C. Fuchs, A. Wilbik, and U. Kaymak, "Towards more specific estimation of membership functions for data-driven fuzzy inference systems," in *2018 IEEE Int'l Conf. on Fuzzy Systems (FUZZ-IEEE)*. IEEE, 2018, pp. 1–8.

[7] J. M. C. Sousa and U. Kaymak, *Fuzzy decision making in modeling and control*. World Scientific, 2002, vol. 27.

[8] M. J. Gacto, R. Alcalá, and F. Herrera, "Interpretability of linguistic fuzzy rule-based systems: An overview of interpretability measures," *Inf. Sci.*, vol. 181, no. 20, pp. 4340–4360, 2011.

[9] B. McMahan, E. Moore, D. Ramage, S. Hampson, and B. A. y Arcas, "Communication-efficient learning of deep networks from decentralized data," in *Artificial Intelligence and Statistics*. PMLR, 2017, pp. 1273–1282.

[10] J. Konečnỳ, H. B. McMahan, D. Ramage, and P. Richtárik, "Federated optimization: Distributed machine learning for on-device intelligence," *arXiv preprint arXiv:1610.02527*, 2016.

[11] X. Zhu, D. Wang, W. Pedrycz, and Z. Li, "Horizontal federated learning of takagi-sugeno fuzzy rule-based models," *IEEE T FUZZY SYST*, 2021.

[12] W. Pedrycz, "Federated fcm: Clustering under privacy requirements," *IEEE T FUZZY SYST*, 2021.

[13] J. L. Corcuera Bárcena, F. Marcelloni, A. Renda, A. Bechini, and P. Ducange, "A federated fuzzy c-means clustering algorithm," in *Int'l Workshop on Fuzzy Logic and Applications 2021*, vol. 3074, 2021, pp. 1–9.

[14] D. Kukolj, "Design of adaptive takagi–sugeno–kang fuzzy models," *Applied Soft Computing*, vol. 2, no. 2, pp. 89–103, 2002.

[15] O. Cord *et al.*, *Genetic fuzzy systems: evolutionary tuning and learning of fuzzy knowledge bases*. World Scientific, 2001, vol. 19.

[16] Y. Cui, D. Wu, and J. Huang, "Optimize tsk fuzzy systems for classification problems: Minibatch gradient descent with uniform regularization and batch normalization," *IEEE T FUZZY SYST*, vol. 28, no. 12, pp. 3065–3075, 2020.

[17] D. Wu, Y. Yuan, J. Huang, and Y. Tan, "Optimize tsk fuzzy systems for regression problems: Minibatch gradient descent with regularization, droprule, and adabound (mbgd-rda)," *IEEE T FUZZY SYST*, vol. 28, no. 5, pp. 1003–1015, 2020.

[18] L.-X. Wang and J. M. Mendel, "Generating fuzzy rules by learning from examples," *IEEE T SYST MAN CYB*, vol. 22, no. 6, pp. 1414–1427, 1992.

[19] J. C. Bezdek, *Fuzzy-Mathematics in Pattern Classification.* Cornell University, 1973.

[20] J. Alcalá-Fdez, A. Fernández, J. Luengo, J. Derrac, S. García, L. Sánchez, and F. Herrera, "Keel data-mining software tool: data set repository, integration of algorithms and experimental analysis framework." *J MULT-VALUED LOG S*, vol. 17, 2011.

[21] R. Alcalá, P. Ducange, F. Herrera, B. Lazzerini, and F. Marcelloni, "A multiobjective evolutionary approach to concurrently learn rule and data bases of linguistic fuzzy-rule-based systems," *IEEE T FUZZY SYST*, vol. 17, no. 5, pp. 1106–1122, 2009.

[22] F. Wilcoxon, "Individual comparisons by ranking methods," in *Breakthroughs in statistics*. Springer, 1992, pp. 196–202.

[23] J. M. Alonso, P. Ducange, R. Pecori, and R. Vilas, "Building explanations for fuzzy decision trees with the expliclas software," in *2020 IEEE Int'l Conf. on Fuzzy Systems (FUZZ-IEEE)*. IEEE, 2020, pp. 1–8.