Information Fusion 113 (2025) 102598

Contents lists available at ScienceDirect

Information Fusion

journal homepage: www.elsevier.com/locate/inffus

Full length article

# Increasing trust in AI through privacy preservation and model explainability: Federated Learning of Fuzzy Regression Trees

José Luis Corcuera Bárcena, Pietro Ducange, Francesco Marcelloni, Alessandro Renda *

*University of Pisa, Department of Information Engineering, Largo L. Lazzarino, Pisa, 56122, Italy*

ARTICLE INFO

Dataset link: https://sci2s.ugr.es/keel/datasets.php, https://www.dcc.fc.up.pt/~ltorgo/Regression/DataSets.html, http://docenti.ing.unipi.it/g.nardini/ai6g_qoe_dataset.html

*Keywords:*
Federated Learning
Fuzzy Regression Trees
Regression models
Explainable Artificial Intelligence

ABSTRACT

Federated Learning (FL) lets multiple data owners collaborate in training a global model without any violation of data privacy, which is a crucial requirement for enhancing users' trust in Artificial Intelligence (AI) systems. Despite the significant momentum recently gained by the FL paradigm, most of the existing approaches in the field neglect another key pillar for the trustworthiness of AI systems, namely explainability. In this paper, we propose a novel approach for FL of fuzzy regression trees (FRTs), which are generally acknowledged as highly interpretable by-design models. The proposed FL procedure is designed for the scenario of horizontally partitioned data and is based on the transmission of aggregated statistics from the clients to a central server for the tree induction procedure. It is shown that the proposed approach faithfully approximates the ideal case in which the tree induction algorithm is applied on the union of all local datasets, while still ensuring privacy preservation. Furthermore, the FL approach brings benefits, in terms of generalization capability, compared to the local learning setting in which each participant learns its own FRT based only on the private, local, dataset. The adoption of linear models in the leaf nodes ensures a competitive level of performance, as assessed by an extensive experimental campaign on benchmark datasets. The analysis of the results covers both the aspects of accuracy and interpretability of FRT. Finally, we discuss the application of the proposed federated FRT to the task of Quality of Experience forecasting in an automotive case-study.

## 1. Introduction

Data access limitations in decentralized settings, mainly imposed by privacy requirements, hamper the application of traditional machine learning (ML) approaches for knowledge extraction from data. Federated learning (FL) [1,2] emerged to fill this need, enabling multiple parties to collaboratively train an ML model without any disclosure of private raw data. In fact, data owners are typically reluctant to share their data with other parties, also within the perimeter of the legitimate interests of service providers. In a nutshell, during an FL process a shared global model is learned through proper aggregation of locally-computed updates from remote data owners. Essentially, FL removes the need to collect data at a central node for training an ML model. The increasing attention recently received by FL is also driven by regulatory and legislative efforts put in place worldwide to increase the trustworthiness of AI systems. For instance, the "AI ACT"[1] (2021), approved by the European Parliament on March 13th, 2024, introduces a common regulatory and legal framework for AI. Such legislation aims to ensure that AI systems developed and used in Europe abide by the EU rights and values, including human oversight, safety, privacy, transparency, non-discrimination, and social and environmental well-being.

Ultimately, data privacy emerges as a key enabler for trustworthiness, but it is not the only technical requirement to consider in the design of an AI system. Another fundamental requirement is *transparency*, with reference to both the traceability of the learning process (starting from the data gathering phase) and the capability of understanding the structure and the functioning of the ML model itself. This latter challenge is the focus of an entire branch of AI which is widely known as Explainable AI (XAI) [3–6]. According to the "Ethics guidelines for trustworthy AI" [7], in fact, "*[...] AI systems and their decisions should be explained in a manner adapted to the stakeholder concerned*". Indeed, the capability to understand how an AI system works is a building block of trust, and is paramount in a number of high stakes applications in the health, finance, and law domains.

It is widely agreed that two major approaches can be employed to achieve explainability in AI systems [3,4]: the adoption of ML models that are interpretable by-design (i.e., transparent models), and the

* Corresponding author.
*E-mail addresses:* jose.corcuera@ing.unipi.it (J.L. Corcuera Bárcena), pietro.ducange@unipi.it (P. Ducange), francesco.marcelloni@unipi.it (F. Marcelloni), alessandro.renda@unipi.it (A. Renda).

[1] https://artificialintelligenceact.eu/the-act/, visited July 2024.

https://doi.org/10.1016/j.inffus.2024.102598
Received 23 May 2024; Received in revised form 20 July 2024; Accepted 22 July 2024
Available online 25 July 2024

so-called post-hoc explainability techniques (e.g., local explanations, explanations by example, explanations by simplification and feature relevance). The property of transparency that characterizes the former approach is generally accorded to models like decision trees (DTs), regression trees (RTs) and rule-based systems (RBSs): in fact, such models can be represented as collection of "IF *antecedent* THEN *consequent*" rules, also resulting in an inference process highly understandable to a human observer. The latter category of approaches (post-hoc techniques) is applied, instead, on models which are not interpretable by-design and therefore referred to as opaque or "black boxes". Neural networks (NNs) and ensemble models are glaring examples of such kind of models.

While research on FL and XAI topics have both reached considerable development and maturity, the combination of the two is just in its infancy, with only few works attempting to meet simultaneously the requirements of privacy and explainability. Most of the algorithmic approaches in the field of FL, in fact, stems from the original proposal of Federated Averaging (FedAvg) [1,8] as a protocol for executing stochastic gradient descent (SGD) in a decentralized setting: as a consequence, FL is mainly adopted with Deep Learning (DL) and NN models, whose training method is based on parameters optimization through SGD. Recently, with the aim of filling this gap and providing a further leap forward towards trustworthy AI, the Fed-XAI concept has been introduced [9,10]. The acronym stands for *Fed*erated Learning of *XAI* models and indicates methodological and technological solutions that leverage the FL approach for privacy preservation while ensuring the explainability of the AI system itself.

Achieving the Fed-XAI goal is not straightforward, either using post-hoc or ex-ante explainability techniques. As for the former approach, some works have advocated the use of post-hoc explainability techniques for horizontal [11], vertical [12,13] or both [14] FL settings. We recall that in the vertical FL setting data is partitioned over the feature space rather than the instance space. However, ensuring consistency of local explanations while preserving privacy of raw data remains an open challenge. Furthermore, post-hoc methods imply higher computational cost and complexity of the whole AI pipeline, so it is generally considered appropriate to investigate first interpretable by-design approaches [15].

Regarding the FL of inherently interpretable models, which do not require post-hoc techniques, one of the major challenges is represented by the need to revisit their learning algorithm to comply with the federated setting. In fact the training stage of such models does not rely on the optimization of a differentiable cost function (e.g., the cross-entropy, in the case of an NN-based classifier). Thus, the standard approach based on FedAvg cannot be directly used. Some work has recently focused on FL of fuzzy RBSs [16,17] and FL of DTs [18,19], which are however only suitable for addressing classification tasks. Few works have addressed FL of tree-based ensemble models [20,21], whose interpretability is however severely compromised compared to that of individual trees.

The problem of learning interpretable regression trees in the federated setting has not been adequately investigated so far, although such solutions would be valuable in high-stake applications and scenarios, such as healthcare or finance, where privacy and transparency represent imperative needs [22,23]. The approach proposed in this paper addresses these needs by enabling the FL of Fuzzy Regression Trees (FRTs) and thus it pertains to the field of Fed-XAI. An FRT [24–26] enhances the traditional RT with concepts from fuzzy set theory and has proven to achieve competitive performance in regression tasks [27]. The adoption of fuzziness is typically useful in scenarios characterized by vagueness and/or noise. Furthermore, the linguistic representation of numerical variables endows the model with explicit semantic interpretability [28].

Our approach is designed for the horizontal FL setting, in which training instances of multiple data owners are described by the same set of features. The main challenge, indeed, consists in the adaptation for the federated setting of a traditional FRT induction algorithm, e.g., the one described in [26]: the hierarchical partition of the feature space and the estimation of the regression models in the leaves must take place on the basis of a decentralized dataset, where each chunk is private to its owner. In our proposal the data owners collaborate in learning an FRT model by sharing locally computed statistics with a central server, which is in charge of orchestrating the global tree induction procedure. The iterative process is organized in rounds, as it is customary in the FL paradigm, and results in a global federated FRT model which features a high generalization capability because it exploits knowledge extracted from decentralized data (yet without requiring raw data sharing). Experimental results indicate that a *federated* FRT outperforms in general a *local* FRT, namely a model learned by using solely data stored in the client, thus implying an incentive for data owners to participate in the federation. Furthermore, we formally prove and empirically verify that the proposed approach for FL of FRT is a faithful approximation of the application of the traditional FRT learning algorithm on the dataset consisting of the union of all local datasets. In other words, the federated FRT is substantially equivalent to the FRT obtained by transferring all the raw data to a central server and applying traditional FRT learning (we denote this approach as *centralized* FRT). We note that in most real-world cases the centralized FRT cannot be applied due to privacy limitations.

A summary of the main contributions of the paper is provided in the following:

- A novel approach, based on the exchange of summary statistics computed by the participants, for FL of FRTs over horizontally partitioned data is introduced.[2] To the best of our knowledge, this is the first proposal of an FRT learning scheme that is compliant with the federated setting, thus ensuring the privacy of data owners.
- The interpretability of the Federated FRT is analyzed and discussed from both the global perspective (i.e., related to the model structure) and the local perspective (i.e., related to the inference process).
- A comprehensive experimental analysis is carried out to validate the proposed approach: the FRT learned in a federated fashion is shown to outperform FRTs learned locally on each participant and to achieve substantially equivalent results compared to the centralized setting. A statistical test is also employed to compare regression metrics across the different learning settings.
- The proposed approach for Federated FRT is adopted in the context of an innovative AI-empowered service in an automotive case-study for the task of Quality of Experience (QoE) forecasting [29]. The proposed approach is shown to achieve competitive performance with respect to alternative state-of-the-art approaches for QoE forecasting and to be robust with respect to the fraction of participants involved in the FL process.

The rest of the paper is organized as follows: Section 2 discusses related works. Section 3 describes the background related to FRT, detailing about the tree construction procedure and the inference process. Section 4 introduces our approach for FL of FRT. In Section 5 we describe the experimental setup and discuss results and findings. Finally, Section 6 reports some concluding remarks.

## 2. Related works

In this section, we first provide some preliminaries regarding FL and then discuss the most relevant related works concerning FL of XAI models.

[2] Source code available: https://github.com/Unipisa/FederatedFuzzyRegressionTree

### 2.1. Federated learning preliminaries

FL has been introduced by Google Research group in 2017 [1,8]. Since then, most approaches to FL have focused on the horizontal setting and the star topology (i.e., relying on the orchestration of a central server). Furthermore, algorithmic solutions typically stem from the established FedAvg protocol for model aggregation, which enables collaborative SGD optimization with an iterative, round-based procedure. At each round the following steps are performed: (i) the server sends out the current global model to the data owners (or a subset thereof); (ii) each selected data owner updates the model by performing some epochs of SGD on its local data; (iii) each selected data owner sends back the updated model to the server; (iv) the server takes the average of the locally updated models, weighted according to the number of instances, to obtain a new global model. Recently, most extensions of FedAvg were introduced with the aim of addressing FL in heterogeneous settings [30,31] or to enhance privacy and security of FL approaches [32]. Much less attention has been devoted so far to the aspect of explainability of models learned in a federated fashion, which can be of the utmost importance in areas such as healthcare [23,33,34] and networking [35].

### 2.2. Federated learning of explainable AI models

The activity indicated with the term Fed-XAI aims to combine the FL paradigm with the benefits brought by XAI approaches. An overview of the most relevant efforts in this field has been provided in two recent review papers [9,10]. Several recent works [14,36] have focused on SHAP [37], as one of the most popular post-hoc techniques to explain the predictions of an opaque ML model in terms of feature importance through the so-called Shapley values. The primary challenge is that SHAP relies, at explanation time, on a reference dataset which is typically composed of training instances. This poses a significant obstacle in the FL setup, since each participant has access only to its local training data. Consequently, different participants may yield different explanations for an output value obtained from identical input instances and using the same FL model. In other words, the adoption of FL to ensure privacy undermines consistency of explanations among participants. Furthermore, the adoption of a post-hoc technique typically entails additional overhead in terms of computation compared to inherently interpretable models.

The FL of interpretable by-design models has recently gained significant momentum. As observed in [9], Takagi–Sugeno-Kang Fuzzy Rule-Based Systems (TSK-FRBS) [38] have been mostly considered as XAI models to be learnt in a federated fashion [16,17,39]. A TSK-FRBS consists in a collection of linguistic rules, in which the consequent part is typically a linear function of the input variables. An example of the generic $r$th rule is as follows:

$$\begin{aligned} R_r : \textbf{IF} \quad & X_1 \ is \ B_{1,j_{r,1}} \quad \textbf{AND} \quad \dots \quad \textbf{AND} \quad X_F \ is \ B_{F,j_{r,F}} \\ & \textbf{THEN} \quad y_r = \gamma_{r,0} + \sum_{i=1}^{F} \gamma_{r,i} \cdot x_i \end{aligned} \tag{1}$$

where $F$ is the total number of variables, $B_{i,j_{r,i}}$ identifies the $j$th fuzzy set of the fuzzy partition over the $i$th variable considered in the $r$th rule, and $\gamma_{r,i}$ are the coefficient of the linear model, with $i = 0, \dots, F$.

The rule base is typically determined using a data-driven approach that involves two stages: first, the fuzzy partition of each input attribute is derived, thus resulting in the antecedent part of the rules; then, the consequent part of each rule (i.e., the coefficients of the linear model) is estimated. Two recent proposals for federated TSK-FRBS [16,17] exploit clustering for the antecedent generation stage. Authors of [17] consider a local clustering procedure followed by an aggregation stage carried out on the server aimed at merging similar clusters. Authors of [16], instead, adopt a federated version of the Fuzzy C-Means algorithms for the identification of the global clusters. Based on the discovered clusters, the antecedent parameters (i.e., membership function for each fuzzy set) are typically determined through Gaussian fitting of the convex envelop of the project membership degrees of training instances to each cluster. As per the second stage (consequent part estimation), both works consider the application of a federated gradient-based learning schemes.

In one of our recent works [39] we have proposed an alternative method for federated TSK-FRBS. With the goal of enhancing model interpretability, we adopt fuzzy uniform partitions with a limited number of fuzzy sets (3 to 5, typically). Thus, the fuzzy sets are distinguishable and offer high semantic interpretability, which is not guaranteed with the data-driven clustering procedure. Furthermore, the FL process yields the global model in an one-shot procedure instead of requiring multiple rounds: first, each data owner learns a TSK-FRBS based on its local data and sends it to the server. Then, the server aggregates the received rules by juxtaposing the rule bases collected from clients and by resolving possible conflicts (i.e., distinct rules having antecedents referring to identical or overlapping regions of the attribute space but different consequents). We have shown that the resulting federated TSK-FRBS outperforms models generated locally, but it is generally outperformed by the centralized TSK-FRBS. This approach was conceived and employed in the context of the Hexa-X Flagship EU project on 6G.[3] Furthermore, under the framework of Hexa-X, Fed-XAI has been awarded as key innovation[4] by the EU Innovation Radar. We have discussed the potential application of the Fed-XAI paradigm in advanced 5G towards 6G systems for an automated vehicle networking case study [29]; the design of an application enabling Fed-XAI in edge computing environments has been described in [40], whereas the actual deployment and experimentation of the federated TSK-FRBS has been presented in [35] for a task of Quality of Experience (QoE) in a vehicular networking use case. In [41] an extension of the OpenFL library [42] has been introduced to facilitate the FL of inherently interpretable AI models.

TSK-FRBSs however struggle to handle high dimensional datasets [43]: the set of candidate rules grows exponentially with the number of features, thus jeopardizing the accuracy and the interpretability of the system. RTs can be considered functionally equivalent to TSK-FRBSs, as they serve for regression tasks and the rules can be extracted by following the branches from the root to the leaf nodes. Differently from TSK-FRBSs, however, the antecedent part of the rules derived from an RT originates from a recursive, tree-growing, procedure, which therefore provides the model with an inherent feature selection capability. Some recent works [24,26] have demonstrated the effectiveness of FRTs for modeling complex systems in regression tasks. However, to the best of our knowledge, no paper investigated so far FL of an FRT. Indeed, few works in this direction focus on DTs (suitable for classification task). The IBM Federated Learning framework [18] supports FL of DT based on the ID3 algorithm. In a nutshell, a single DT is generated at the central server leveraging some aggregated statistics being sent by the clients based on their local data at each round. In this way, the server can compute the information gain and perform the split accordingly, thus recursively growing the DT. Usual stopping conditions (e.g., maximum depth of the tree or minimum information gain) are adopted. The approach introduced in this manuscript has points in common with such a proposal, as we also rely on the transmission of aggregated information for the generation of a single federated tree on the server side. However, we focus on (fuzzy) RT rather than DT: we need to adapt to the federated setting both the criterion for splitting nodes and the estimation of the model parameters in the leaves, so that they are functional for regression tasks rather than classification ones. Furthermore, unfortunately, the IBM framework is not open-source. In [19] authors discuss FL approach for tree-based models, where a

[3] https://hexa-x.eu, visited July 2024

[4] https://www.innoradar.eu/innovation/45988, visited July 2024.

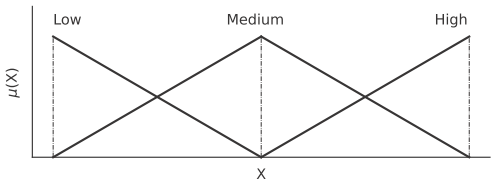


**Fig. 1.** An example of uniform fuzzy partition with three triangular fuzzy sets.

privacy preserving technique (Partially Homomorphic Encryption) is adopted. All clients contribute to build the structure of the DT by providing iterative encrypted statistics to a super client which is in charge of selecting the most relevant attribute and the corresponding splitting point. In the experimental analysis, authors show that the inclusion of the privacy preserving technique entails a slight loss in accuracy. Unlike our approach, conceived for horizontally partitioned data, their proposal is tailored on vertically partitioned data.

Finally, it is worth mentioning few recent works that address federated tree-based ensemble learning. Authors in [21] propose the application of a federated version of the AdaBoost algorithm posing minimal constraints on the learning settings of the clients (so that DTs and support vector machines can be used) and without relying on gradient-based methods. A federated Random Survival Forest has been proposed in [20]. However, the interpretability of ensemble models is significantly reduced compared with that of single trees, failing one of the basic requirements for trustworthiness considered in this paper.

To summarize, a major shortcoming of the extensive FL literature consists in disregarding the goal of explainability and focusing on privacy preserving ML for opaque models. The area of Fed-XAI aims to fill this gap and has recently gained increasing attention. In this context, the problem of learning interpretable RTs in the federated setting has not been investigated so far and is the main focus of our work.

## 3. Background

In this section, we introduce the basic concepts of FRT and the FRT induction algorithm for the *centralized* setting, as proposed in [26]. Furthermore, we introduce the necessary notation that will be exploited for the novel federated approach.

### 3.1. Fuzzy regression trees

An RT is a directed acyclic graph, where each non-leaf node represents a test on an input variable. Each path from the root to one leaf corresponds to a sequence of tests, which aims to isolate a subspace of the input space.

Let $TR = \{(\mathbf{x}_1, y_1), \ldots, (\mathbf{x}_Z, y_Z)\}$ be the training set of $Z$ instances, $\mathbf{X} = \{X_1, \ldots, X_F\}$ be the set of input variables, and $(\mathbf{x}_i, y_i)$ the generic $i$th training instance. We assume that each real input variable $X_f$ is partitioned by using $T_f$ fuzzy sets. Let $\mathbf{P}_f = \{B_{f,1}, \ldots, B_{f,T_f}\}$ be the partition of input variable $X_f$. An example of uniform fuzzy partition with three triangular fuzzy sets, respectively labeled as *low*, *medium* and *high*, is shown in Fig. 1.

The tests in the internal nodes use these fuzzy sets in the form of "$X_f$ is $B_{f,j}$". The membership degree of an instance $\mathbf{x}_i$ with respect to the fuzzy set $B_{f,j}$ is represented as $\mu_{B_{f,j}}(x_{i,f})$. Since fuzzy sets generally overlap, an input instance may activate more than one leaf node.

Leaf nodes are characterized by a regression model defined over the input variables. Different regression models have been used in the literature, although all referred to the traditional centralized learning setting. For instance M5 [44] generates first-order polynomials using the overall set of input variables. CART [45], which is one of the most known algorithms for generating RTs, uses zero-order polynomials, leading to simpler and usually more robust models.

In the case of a zero-order polynomial regression model, the scalar value assigned to each leaf node $LN^{(K)}$ is $\phi^{LN^{(K)}}(\mathbf{X}) = c^{LN^{(K)}}$, which is computed as a weighted average of the output values $y_i$ of all the training set instances that activate such leaf node (Eq. (2)). In the following, we indicate with $r$ the generic leaf $LN^{(K)}$ to avoid overburden the notation.

$$\phi^{(r)}(\mathbf{X}) = c^{(r)} = \frac{\sum_{(\mathbf{x}_i, y_i)|w^{(r)}(\mathbf{x}_i)>0} \left(y_i \cdot w^{(r)}\left(\mathbf{x}_i\right)\right)}{\sum_{(\mathbf{x}_i, y_i)|w^{(r)}(\mathbf{x}_i)>0} \left(w^{(r)}\left(\mathbf{x}_i\right)\right)} \tag{2}$$

where

$$w^{(r)}\left(\mathbf{x}_i\right) = \prod_{k=1}^{K} \mu_{f^{(k)}}(x_{i,f^{(k)}}) \tag{3}$$

The term $\mu_{f^{(k)}}(x_{i,f^{(k)}})$ is the membership degree of $x_{i,f^{(k)}}$ to the fuzzy set $B_{f^{(k)},j}$ of the partition of each input variable $X_{f^{(k)}}$ chosen in each node $N^{(k)}$ in the path from the root ($k = 1$) to the leaf node $r = LN^{(K)}$ (with $k = K$). We observe that only the instances $\mathbf{x}_i$ with $w^{(r)}(\mathbf{x}_i) > 0$ are considered in the computation of $c^{(r)}$.

A first-order polynomial regression model employs in any leaf a linear model defined as in Eq. (4).

$$\phi^{(r)}(\mathbf{X}) = \gamma_0^{(r)} + \sum_{f=1}^{F} \gamma_f^{(r)} \cdot X_f \tag{4}$$

The coefficients $\boldsymbol{\gamma}^{(r)} = \left\{\gamma_0^{(r)}, \gamma_1^{(r)}, \ldots, \gamma_F^{(r)}\right\}$ can be estimated by applying locally a weighted least-squared method.

Let $\mathbf{S}^{(r)} = [\mathbf{x}_i \in TR \mid w^{(r)}(\mathbf{x}_i) > 0]$ be the vector of instances with non-null strength of activation to the leaf node $r$, $\mathbf{y}^{(r)}$ be the vector of associated target values, and $\mathbf{w}^{(r)}$ be the vector of associated values of strength of activation. Finally, let $\mathbf{A}^{(r)} = \mathbf{S}^{(r)} \odot \mathbf{w}^{(r)}$ be the set of instances in $\mathbf{S}^{(r)}$, each weighted by its strength of activation to the leaf node $r$.[5] According to the weighted least-squared method, the coefficients $\boldsymbol{\gamma}^{(r)}$ are estimated as follows:

$$\boldsymbol{\gamma}^{(r)} = \left(\mathbf{A}^{(r)^T}\mathbf{A}^{(r)}\right)^{-1}\mathbf{A}^{(r)^T}\mathbf{y}^{(r)} \tag{5}$$

where T indicates the transpose, and a unit column vector is appended to the data matrix $\mathbf{A}^{(r)}$ for the estimate of the coefficient $\gamma_0^{(r)}$.

Notably, for any given rule, the linear regression model considers the whole set of $F$ input variables, even if typically only a subset of them appears in the antecedent part.

Fig. 2 shows an example of multi-way FRT for regression tasks. The represented tree exploits only two features, namely $X_1$ at the root and $X_2$ at the second level. Both features are partitioned into three levels (as in the example of Fig. 1), thus generating exactly three output branches. Finally, each leaf node is associated with a first-order regression model.

### 3.2. Partition Fuzzy gain, Fuzzy variance and Fuzzy mean

The FRT induction procedure consists in a hierarchical partition of the feature space based on the training instances [26]. Such a partition is generated recursively and requires the definition of a criterion for the choice of the most suitable input variable to be used in the node under consideration. The first selected attribute is used in the root of the tree and originates in a multi-way FRT as many branches and corresponding child nodes as the number of fuzzy sets that constitutes its partition. For each child node, a new attribute is selected, considering only the instances of the training set that have a membership value higher than 0 to the fuzzy set associated with the respective branch. The procedure terminates when specific termination conditions are met.

The choice of the input variable to be used in each decision node is made by using the Partition Fuzzy Gain ($PFGain$) adopted in [24,26]. $PFGain$ is based on the concept of Fuzzy Variance.

[5] The symbol $\odot$ is used to indicate the Hadamard (element-wise) product.

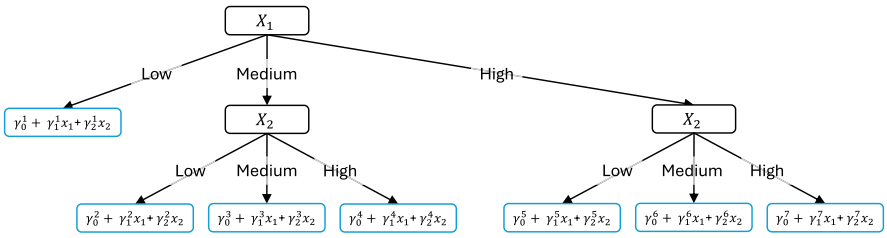


**Fig. 2.** Toy example of a multi-way FRT.

Formally, let $N^{(k)}$ be a generic node in FRT and let $w^{N^{(k)}}(\mathbf{x}_i) = \prod_{t=1}^{k} \mu_{f^{(t)}}(x_{i,f^{(t)}})$ be the strength of activation of instance $(\mathbf{x}_i, y_i) \in TR$ to node $N^{(k)}$ computed along the path $R^{(k)}$ from the root $N^{(1)}$ to $N^{(k)}$. In the root, $w^{N^1}(\mathbf{x}_i) = 1$ for all the instances in $TR$. Let $\mathbf{S}^{N^{(k)}} = [(\mathbf{x}_i, y_i) \in TR \mid w^{N^{(k)}}(\mathbf{x}_i) > 0]$ be the vector of instances with non-null strength of activation to node $N^{(k)}$.

First of all, we introduce the fuzzy mean and the fuzzy variance for the instances in $\mathbf{S}^{N^{(k)}}$. Then, we define the fuzzy mean and the fuzzy variance of the instances in the support of a generic fuzzy set $B_{f^{(k)},j}$ when the instances in $\mathbf{S}^{N^{(k)}}$ are partitioned by using $\mathbf{P}_{f^{(k)}}$. Finally, we introduce the definition of $PFGain$.

The *Fuzzy Mean* $FM^{N^{(k)}}$ of node $N^{(k)}$ is defined as the mean of the output values $y_i$ of the instances $(\mathbf{x}_i, y_i)$ in $\mathbf{S}^{N^{(k)}}$, weighted by the strength of activation $w^{N^{(k)}}(\mathbf{x}_i)$:

$$FM^{N^{(k)}} = \frac{\sum_{(\mathbf{x}_i,y_i)\in\mathbf{S}^{N^{(k)}}} \left(y_i \cdot w^{N^{(k)}}(\mathbf{x}_i)\right)}{\sum_{(\mathbf{x}_i,y_i)\in\mathbf{S}^{N^{(k)}}} \left(w^{N^{(k)}}(\mathbf{x}_i)\right)} \tag{6}$$

The *Fuzzy Variance* $FVar^{N^{(k)}}$ of node $N^{(k)}$ is defined as follows:

$$FVar^{N^{(k)}} = \frac{\sum_{(\mathbf{x}_i,y_i)\in\mathbf{S}^{N^{(k)}}} \left(y_i - FM^{N^{(k)}}\right)^2 \cdot \left(w^{N^{(k)}}(\mathbf{x}_i)\right)}{\sum_{(\mathbf{x}_i,y_i)\in\mathbf{S}^{N^{(k)}}} \left(w^{N^{(k)}}(\mathbf{x}_i)\right)} \tag{7}$$

By simple algebraic transformations, the fuzzy variance can also be expressed in an alternative form, which will be useful in the formulation of the federated approach:

$$FVar^{N^{(k)}} = \frac{WSS^{N^{(k)}}}{WS^{N^{(k)}}} - \left[\frac{WLS^{N^{(k)}}}{WS^{N^{(k)}}}\right]^2 \tag{8}$$

where the weighted linear sum (WLS), weighted squared sum (WSS) and sum of the weights (WS) are defined as follows:

$$WSS^{N^{(k)}} = \sum_{(\mathbf{x}_i,y_i)\in\mathbf{S}^{N^{(k)}}} y_i^2 \cdot w^{N^{(k)}}(\mathbf{x}_i) \tag{9}$$

$$WLS^{N^{(k)}} = \sum_{(\mathbf{x}_i,y_i)\in\mathbf{S}^{N^{(k)}}} y_i \cdot w^{N^{(k)}}(\mathbf{x}_i) \tag{10}$$

$$WS^{N^{(k)}} = \sum_{(\mathbf{x}_i,y_i)\in\mathbf{S}^{N^{(k)}}} w^{N^{(k)}}(\mathbf{x}_i) \tag{11}$$

Let $\mathbf{S}_{f,1}^{N^{(k)}}, \ldots, \mathbf{S}_{f,T_F}^{N^{(k)}}$ be the vectors of instances in $\mathbf{S}^{N^{(k)}}$, contained in the supports of the fuzzy sets $B_{f^{(k)},1}, \ldots, B_{f^{(k)},T_f}$ of the partition $P_f$ tested for splitting node $N^{(k)}$. The fuzzy mean $FM^{N^{(k)}}(B_{f^{(k)},j})$ of the output values computed for the instances in $\mathbf{S}_{f,j}^{N^{(k)}}$ is defined as the mean of the $y_i$ weighted by the product between the strength of activation of $x_{i,f^{(k)}}$ to the node $N^{(k)}$ and the membership degree of $x_{i,f^{(k)}}$ to $B_{f^{(k)},j}$:

$$FM^{N^{(k)}}(B_{f^{(k)},j}) = \frac{\sum_{(\mathbf{x}_i,y_i)\in\mathbf{S}_{f^{(k)},j}^{N^{(k)}}} \left(y_i \cdot w^{N^{(k)}}(\mathbf{x}_i) \cdot \mu_{B_{f^{(k)},j}}(x_{i,f^{(k)}})\right)}{\sum_{(\mathbf{x}_i,y_i)\in\mathbf{S}_{f^{(k)},j}^{N^{(k)}}} \left(w^{N^{(k)}}(\mathbf{x}_i) \cdot \mu_{B_{f^{(k)},j}}(x_{i,f^{(k)}})\right)} \tag{12}$$

The fuzzy variance $FVar^{N^{(k)}}(B_{f^{(k)},j})$ of the output values computed for the instances of the support of fuzzy set $B_{f^{(k)},j}$ in the node $N^{(k)}$ is defined as follows:

$$FVar^{N^{(k)}}(B_{f^{(k)},j}) = \frac{\sum_{(\mathbf{x}_i,y_i)\in\mathbf{S}_{f,j}^{N^{(k)}}} \left[y_i - FM^{N^{(k)}}(B_{f^{(k)},j})\right]^2 \cdot w^{N^{(k)}}(\mathbf{x}_i) \cdot \mu_{B_{f^{(k)},j}}(x_{i,f^{(k)}})}{\sum_{(\mathbf{x}_i,y_i)\in\mathbf{S}_{f,j}^{N^{(k)}}} w^{N^{(k)}}(\mathbf{x}_i) \cdot \mu_{B_{f^{(k)},j}}(x_{i,f^{(k)}})} \tag{13}$$

Like $FVar^{N^{(k)}}$, also $FVar^{N^{(k)}}(B_{f^{(k)},j})$ can be expressed in the alternative form:

$$FVar^{N^{(k)}}(B_{f^{(k)},j}) = \frac{WSS^{N^{(k)}}(B_{f^{(k)},j})}{WS^{N^{(k)}}(B_{f^{(k)},j})} - \left[\frac{WLS^{N^{(k)}}(B_{f^{(k)},j})}{WS^{N^{(k)}}(B_{f^{(k)},j})}\right]^2 \tag{14}$$

where

$$WSS^{N^{(k)}}(B_{f^{(k)},j}) = \sum_{(\mathbf{x}_i,y_i)\in\mathbf{S}_{f^{(k)},j}^{N^{(k)}}} y_i^2 \cdot w^{N^{(k)}}(\mathbf{x}_i) \cdot \mu_{B_{f^{(k)},j}}(x_{i,f^{(k)}}) \tag{15}$$

$$WLS^{N^{(k)}}(B_{f^{(k)},j}) = \sum_{(\mathbf{x}_i,y_i)\in\mathbf{S}_{f^{(k)},j}^{N^{(k)}}} y_i \cdot w^{N^{(k)}}(\mathbf{x}_i) \cdot \mu_{B_{f^{(k)},j}}(x_{i,f^{(k)}}) \tag{16}$$

$$WS^{N^{(k)}}(B_{f^{(k)},j}) = \sum_{(\mathbf{x}_i,y_i)\in\mathbf{S}_{f^{(k)},j}^{N^{(k)}}} w^{N^{(k)}}(\mathbf{x}_i) \cdot \mu_{B_{f^{(k)},j}}(x_{i,f^{(k)}}) \tag{17}$$

The *Partition Fuzzy Gain* $PFGain^{N^{(k)}}(P_{f^{(k)}})$ obtained by adopting the fuzzy partition $P_{f^{(k)}}$ over the input variable $X_{f^{(k)}}$ is defined as follows:

$$PFGain^{N^{(k)}}(P_{f^{(k)}}) = FVar^{N^{(k)}} - \sum_{j=1}^{T_f} FVar^{N^{(k)}}(B_{f^{(k)},j}) \cdot W^{N^{(k)}}(B_{f^{(k)},j}) \tag{18}$$

where:

$$W^{N^{(k)}}(B_{f^{(k)},j}) = \frac{WS^{N^{(k)}}(B_{f^{(k)},j})}{\sum_{j=1}^{T_f} WS^{N^{(k)}}(B_{f^{(k)},j})} \tag{19}$$

Let $X_{\hat{f}}$ be the input variable with the highest $PFGain^{N^{(k)}}$. Fuzzy sets $B_{\hat{f},j}$ are used to split the instances $S^{N^{(k)}}$ in node $N^{(k)}$ into $T_{\hat{f}}$ child nodes $N_j^{(k+1)}$, $j = 1, \ldots, T_{\hat{f}}$. The strength of activation $w^{N_j^{(k+1)}}(\mathbf{x}_i)$ of a generic instance $\mathbf{x}_i$ to child node $N_j^{(k+1)}$ is computed as $w^{N_j^{(k+1)}}(\mathbf{x}_i) = w^{N^{(k)}}(\mathbf{x}_i) \cdot \mu_{B_{\hat{f},j}}(x_{i,\hat{f}})$.

### 3.3. Tree construction and inference process

In the proposed algorithm the following criteria are employed to stop the tree growth:

- when the cardinality of the set $\mathbf{Z}^{N^{(k)}}$ of instances that strongly activate the node $N^{(k)}$ ($\mathbf{Z}^{N^{(k)}} = \{(\mathbf{x}_i, y_i) \in TR \mid w^{N^{(k)}}(\mathbf{x}_i) \geq 0.5^k\}$) is lower than a fraction $th_{min\_instances\_split}$ of the instances in the training set $TR$. The threshold for the *strong* activation is set to 0.5 raised to the tree level at which $N^{(k)}$ is located: in other words, we are assuming that at each level of the path the instances belong to the corresponding fuzzy set with a membership degree higher than 0.5;
- when the highest $PFGain$ computed for a node is lower than a fixed threshold $th_{PFGain}$;

Once the tree has been generated, each leaf node is associated with a regression model. In our proposal, we employ first-order polynomial regression models (see Eq. (4)).

The path $p_r$ from the root to a generic leaf node $r = LN^{(K)}$ at the $K$th level can be described by the following rule $R_r$:

$$R_r : \textbf{IF } X_{r(2)} \textit{ is } B_{r(2),j_{r(2)}} \textbf{ AND } \ldots \textbf{ AND } X_{r(K)} \textit{ is } B_{r(K),j_{r(K)}} \quad \textbf{THEN} \quad Y = \phi^{(r)}(\mathbf{X}) = \gamma_0^{(r)} + \sum_{f=1}^{F} \gamma_f^{(r)} \cdot X_f \tag{20}$$

where $X_{r(k)}$ and $B_{r(k),j_{r(k)}}$ are, respectively, the input variable and the fuzzy set of the corresponding partition which allow reaching the node at the $k$th level of path $p_r$ and contribute to the strength of activation for this node (we recall that $k = 1$ identifies the root node).

Given an input pattern $\hat{\mathbf{x}}$, the inference process generates an output based on the maximum matching strategy: only the rule with the highest strength of activation is used for estimating the output value. We adopted this inference strategy because it guarantees a higher interpretability than other possible inference strategies. Compared to aggregating the output values obtained by all the activated rules, in fact, using a single rule may reduce the modeling capability of an FRT. However, it has been shown that the adoption of a maximum matching approach does not particularly degrade the modeling power compared to the weighted average strategy, yet ensuring a higher level of interpretability [26].

The strength of activation of the rule $R_r$ is computed using the product T-norm (it coincides with the strength of activation of the corresponding leaf node, formulated in Eq. (3)):

$$w_r(\hat{\mathbf{x}}) = \prod_{k=1}^{K} \mu_{B_{r(k),j_{r(k)}}}(\hat{x}_{r(k)}) \tag{21}$$

The use of the product as a T-norm operator (see Eq. (21)) has an obvious implication on the inference process: since the terms $\mu_{B_{r(k),j_{r(k)}}}(\hat{x}_{r(k)})$ are in the range $[0, 1]$, the maximum matching approach will in general prioritize short rules, i.e., those associated with leaf nodes *closer* to the root of the FRT. To compensate for this phenomenon, we consider the normalized strength of activation $\tilde{w}_r(\hat{\mathbf{x}})$, which is defined as follows:

$$\tilde{w}_r(\hat{\mathbf{x}}) = \frac{w_r(\hat{\mathbf{x}})}{\bar{w}_r(TR)} \tag{22}$$

where $\bar{w}_r(TR)$ is the average strength of activation for all instances $\mathbf{x}_i$ in the training set with $w_r(\mathbf{x}_i) > 0$.

## 4. Federated learning of FRT

In our proposal, we focus on a horizontal FL scenario with a centralized communication topology, which involves a central server orchestrating the learning process and aggregating the statistics received from $M$ different participants that wish to collaboratively build an FRT. Let $TR_1 = \left\{\mathbf{x}_1^1, \mathbf{x}_2^1, ..., \mathbf{x}_{N_1}^1\right\}$, $TR_2 = \left\{\mathbf{x}_1^2, \mathbf{x}_2^2, ..., \mathbf{x}_{N_2}^2\right\}$, $\ldots$, $TR_M = \left\{\mathbf{x}_1^M, \mathbf{x}_2^M, ..., \mathbf{x}_{N_M}^M\right\}$ be the $M$ private training set of every client, each of them having a variable cardinality and the same $F$-dimensional feature space: $x_i^m = \left\{x_{i,1}^m, x_{i,2}^m, ..., x_{i,F}^m\right\}$. We assume that each feature has already been partitioned by applying a uniform triangular fuzzy partition (see Fig. 1), and both the server and the data owners know how each feature has been partitioned. Concerning the privacy model, we assume honest participants and a *semi-honest*, or *honest-but-curious*, central server. Notably, this is considered as a realistic scenario for the horizontal partitioning case [2]. The server can try to retrieve private raw data based on the updates from the data owners, but always adhering to the protocol defined for the execution of the ML algorithm. Our approach, however, incorporates a mechanism to avoid the disclosure of raw information from data owners, thus ensuring privacy preservation.

The federated version of the FRT learning algorithm is derived from the centralized version described in Section 3.1, and is shown to provide a faithful approximation of it. The complete pseudocodes for the federated versions of the FRT learning, consisting of the tree growing procedure and the estimation of the coefficients of the linear regression models in the leaves, are presented in Algorithm 1 and Algorithm 2, respectively.

**Table 1**
Statistics for each node under consideration for possible splitting transmitted by each data owner $m$ to the server.

| Statistics of the node $N_q^{(t)}$ | Statistics of the node $N_q^{(t)}$ for each feature $f_q^{(t)}$ available for splitting and for each fuzzy set $B_{f_q^{(t)},j}$ in its partition $\mathbf{P}_{f_q^{(t)}} = \left\{B_{f_q^{(t)},j}\right\}_{j=1}^{T_f}$ |
|---|---|
| $WSS_m^{N_q^{(t)}}$ | $WSS_m^{N_q^{(t)}}\left(B_{f_q^{(t)},j}\right)$ |
| $WLS_m^{N_q^{(t)}}$ | $WLS_m^{N_q^{(t)}}\left(B_{f_q^{(t)},j}\right)$ |
| $WS_m^{N_q^{(t)}}$ | $WS_m^{N_q^{(t)}}\left(B_{f_q^{(t)},j}\right)$ |
| $\left\|\mathbf{S}_m^{N_q^{(t)}}\right\|$ | $\left\|\mathbf{S}_m^{N_q^{(t)}}\left(B_{f_q^{(t)},j}\right)\right\|$ |

### 4.1. Federated FRT growing procedure

At each round $t$, each participant computes the statistics for each node under consideration for possible splitting, considering only the set of features which have not been used along the path from the root to the node. We recall that we adopt a multi-way FRT and therefore along a path from the root to a node a feature can be used only once. Obviously, at $t = 1$, the set of nodes under consideration for possible splitting and the corresponding set of features for each node include, respectively, only the root and all the features. More specifically, let $\mathbf{V}^{(t)}$ be the set of pairs $\left(N_q^{(t)}, \mathbf{H}_q^{(t)}\right)$, where $\mathbf{H}_q^{(t)}$ is the set of features available for splitting node $N_q^{(t)}$. At the beginning of the federated procedure (Algorithm 1, *line 1*), the server initializes the $FRT^{(1)}$ with an empty (null) node representing the root, and set $\mathbf{V}^{(1)} = \left\{\left(N_1^{(1)}, \mathbf{H}_1^{(1)}\right)\right\}$ and $stop\_flag = FALSE$ (*line 2*). Then, at each round the server transmits the $FRT^{(t)}$ and the set $\mathbf{V}^{(t)}$ to the data owners (*line 4*). For each node $N_q^{(t)}$ in the set $\mathbf{V}^{(t)}$, each data owner computes the statistics reported in Table 1 and defined in Section 3.2 (*line 5*).

The privacy protection constraint requires that such statistics are nullified to prevent disclosure of raw data (*line 6*) in some specific cases, which are reported in the following: (i) when the number of instances with non-null strength of activation to a child node is less than or equal to 2, by solving a simple system of equations; (ii) when the sum of the weights of the instances reaching a child node is greater than zero and the sum of the weights of the instances reaching the adjacent child nodes is null; in the case of strong triangular fuzzy partition, this means that all the instances falling in the child node have the same value for the attribute (which yields membership degree equal to 1 for the fuzzy set characterizing the node); (iii) finally, the same disclosure pertains the first split of the root node, when the sum of the weights of the instances reaching a child node is equal to the number of instances with non-null strength of activation to the child node itself. Such cases are captured in the function *Check&Nullify()*, reported at the end of the pseudocode in Algorithm 1.

The, potentially nullified, statistics are transmitted to the server (*line 6*), which use them first to check stopping condition and then to possibly continue growing the tree. For each node $N_q^{(t)}$ under consideration for possible splitting, and for each feature $f_q^{(t)}$ available at the node, the server computes the value of $PFGain^{N_q^{(t)}}\left(P_{f_q^{(t)}}\right)$ (as in Eq. (18)). The computation of $PFGain$ in the federated setting is equivalent to the centralized one. Indeed, the following terms can be computed by the server based on the statistics received from the clients:

$$FVar^{N_q^{(t)}} = \frac{\sum_{m=1}^{M} WSS_m^{N_q^{(t)}}}{\sum_{m=1}^{M} WS_m^{N_q^{(t)}}} - \left[\frac{\sum_{m=1}^{M} WLS_m^{N_q^{(t)}}}{\sum_{m=1}^{M} WS_m^{N_q^{(t)}}}\right]^2 \tag{23}$$

**Algorithm 1** *FRT_Growing*: **Horizontal Federated FRT growing**

**Input:** Horizontally partitioned dataset, scattered over $M$ data owners.
**Output:** Fully grown Federated FRT.
**Initialization stage**
*Server:*
1: Initialize an FRT with an empty (null) node $N_1^{(1)}$ and set $\mathbf{H}_1^{(1)} = \{1, \dots, F\}$ and $\mathbf{V}^{(1)} = \left\{ \left( N_1^{(1)}, \mathbf{H}_1^{(1)} \right) \right\}$
2: *stop_flag* = $FALSE$
**Tree growing stage**
*Server:*
3: At each round $t$, with $t$ starting from 1:
4: Send to each data owner the $FRT^{(t)}$ and the set $\mathbf{V}^{(t)} = \left\{ \left( N_1^{(t)}, \mathbf{H}_1^{(t)} \right), ..., \left( N_Q^{(t)}, \mathbf{H}_Q^{(t)} \right) \right\}$ of pairs consisting of the node to evaluate for possible splitting, and the corresponding set of features available for the splitting
***Computation of the statistics***
*Each data owner m:*
5: Compute the statistics as reported in Table 1 for each node $N_q^{(t)} \in \mathbf{V}^{(t)}$ and for each feature $f \in \mathbf{H}_q^{(t)}$
6: Execute the *Check&Nullify()* function and transmit statistics to the server
*Server:*
7: $\mathbf{V}^{(t+1)} \leftarrow \emptyset$
8: **for** each pair $\left( N_q^{(t)}, \mathbf{H}_q^{(t)} \right) \in \mathbf{V}^{(t)}$ **do**
9: evaluate $\hat{f} = \text{argmax}_{f \in \mathbf{H}_q^{(t)}} PFGain^{N_q^{(t)}}(P_f)$, $PFGain$ evaluated as in Eq. (18)
10: **if** $StopCondition(N_q^{(t)})$ **then**
11: mark node $N_q^{(t)}$ as leaf node
12: **else**
13: split $N_q^{(t)}$ based on $\hat{f}$ into $T_{\hat{f}}$ child nodes: $\{N_q^{(t)}(B_{\hat{f},1}), \dots, N_q^{(t)}(B_{\hat{f},T_{\hat{f}}})\}$
14: $\mathbf{H}_q^{(t+1)} \leftarrow \mathbf{H}_q^{(t)} \setminus \{\hat{f}\}$
15: **if** $\mathbf{H}_q^{(t+1)} \neq \emptyset$ **then**
16: $\mathbf{V}^{(t+1)} \leftarrow \mathbf{V}^{(t+1)} \cup \left\{ \left( N_q^{(t)}(B_{\hat{f},1}), \mathbf{H}_q^{(t+1)} \right), \dots, \left( N_q^{(t)}(B_{\hat{f},T_{\hat{f}}}), \mathbf{H}_q^{(t+1)} \right) \right\}$
17: **else**
18: mark nodes $\{N_q^{(t)}(B_{\hat{f},1}), \dots, N_q^{(t)}(B_{\hat{f},T_{\hat{f}}})\}$ as leaf nodes
19: **if** $\mathbf{V}^{(t+1)} == \emptyset$ **then**
20: *stop_flag* = $TRUE$
21: Send *stop_flag* to each data owner
***Termination***
*Each data owner m & Server:*
22: **if** NOT *stop_flag* **then**
23: Proceed with the next round (line 4)
*Function*: *Check&Nullify()*
**Input:** Values of statistics computed by each data owner as summarized in Table 1.
**Output:** Values of statistics in Table 1, possibly nullified to enforce privacy.
24: **for** each pair $\left( N_q^{(t)}, \mathbf{H}_q^{(t)} \right) \in \mathbf{V}^{(t)}$ **do**
25: **for** each feature $f$ available for the splitting of $N_q^{(t)}$ **do**
26: **for** each $B_{f_q^{(t)},j} \in P_f$ **do**
27: **if** $|\mathbf{S}_m^{N_q^{(t)}}(B_{f_q^{(t)},j})| \leq 2$ or $\left( WS_m^{N_q^{(t)}}(B_{f_q^{(t)},j}) > 0 \text{ and } WS_m^{N_q^{(t)}}(B_{f_q^{(t)},l}) = 0 \quad \forall l \in \{j-1, j+1\} \right)$ **then**
28: $[WSS_m^{N_q^{(t)}}(B_{f_q^{(t)},j}),\ WLS_m^{N_q^{(t)}}(B_{f_q^{(t)},j}),\ WS_m^{N_q^{(t)}}(B_{f_q^{(t)},j}),\ |\mathbf{S}_m^{N_q^{(t)}}(B_{f_q^{(t)},j})|] = [0,0,0,0]$
29: **if** $WS_m^{N_1^{(1)}}(B_{f_1^{(1)},j}) = |\mathbf{S}_m^{N_1^{(1)}}(B_{f_1^{(1)},j})|$ **then**
30: $[WSS_m^{N_1^{(1)}}(B_{f_1^{(1)},j}),\ WLS_m^{N_1^{(1)}}(B_{f_1^{(1)},j}),\ WS_m^{N_1^{(1)}}(B_{f_1^{(1)},j}),\ |\mathbf{S}_m^{N_1^{(1)}}(B_{f_1^{(1)},j})|] = [0,0,0,0]$

$$FVar^{N_q^{(t)}}(B_{f_q^{(t)},j}) = \frac{\sum_{m=1}^{M} WSS_m^{N_q^{(t)}}(B_{f_q^{(t)},j})}{\sum_{m=1}^{M} WS_m^{N_q^{(t)}}(B_{f_q^{(t)},j})} - \left[ \frac{\sum_{m=1}^{M} WLS_m^{N_q^{(t)}}(B_{f_q^{(t)},j})}{\sum_{m=1}^{M} WS_m^{N_q^{(t)}}(B_{f_q^{(t)},j})} \right]^2 \tag{24}$$

$$W^{N_q^{(t)}}(B_{f_q^{(t)},j}) = \frac{\sum_{m=1}^{M} WS_m^{N_q^{(t)}}(B_{f_q^{(t)},j})}{\sum_{j=1}^{T_f} \sum_{m=1}^{M} WS_m^{N_q^{(t)}}(B_{f_q^{(t)},j})}. \tag{25}$$

In other words, in the computation of the $PFGain$, we exploit the alternative forms for the calculation of the fuzzy variance reported in Eqs. (8) and (14).

As in the centralized version, the server does not perform the splitting of the node $N_q^{(t)}$ and marks it as a leaf node if any of the following conditions occurs:

- The overall cardinality of the set of instances that strongly activate $N_q^{(t)}$ is lower than a fraction $th_{min\_instances\_split}$ of the instances in the training set $TR$. Formally:

$$\frac{\sum_{m=1}^{M} |\mathbf{Z}_m^{N_q^{(t)}}|}{\sum_{m=1}^{M} |TR_m|} \leq th_{min\_instances\_split} \tag{26}$$

- The highest $PFGain$ computed for a node is lower than a fixed threshold $th_{PFGain}$;

If none of the stopping criteria is met, the feature with the largest $PFGain$ is selected and is used to split the node into $T_f$ child nodes (*line 13*). The selected feature is removed from the list of features to be considered for splitting newly generated nodes (*line 14*). If such list is not empty, the child nodes are added to the list of nodes to be split in the next round (*line 16*). Otherwise, the child nodes are marked as leaf nodes (*line 18*).

Once all the nodes under consideration for possible splitting in the current round $t$ have been analyzed, the server evaluates whether the tree construction procedure should continue: if all the nodes are marked as leaf nodes (that is, the set $V^{(t+1)}$ is empty) the tree growing phase is completed; otherwise, the server transmits the updated global tree structure and the updated set of pairs to each data owner and the recursive tree growing procedure continues with a new communication round (*line 4*) .

It is worth emphasizing that, net of the Check&Nullify() procedure, the federated tree growing procedure is equivalent to the centralized one. As empirically shown in Section 5.2, the impact on the final outcome of this privacy protection measure is rather limited, indicating that the cases which trigger the nullification are fairly uncommon.

The FRT growing procedure described in Algorithm 1 results in a fully-grown FRT, which is provided in input to the federated linear regression model estimation described in the next subsection.

### 4.2. Federated linear regression model estimation at the FRT leaves

Algorithm 2 defines in detail the procedure adopted to compute the regression model in a federated way to be used in each leaf node of the global FRT.

**Algorithm 2** *LeavesRegressionEstimate*: **Federated linear regression model estimation at the leaves**

**Input:** Horizontally partitioned dataset, scattered over $M$ data owners; fully grown Federated FRT
**Output:** Federated FRT with first-order regression models at leaf nodes.
*Server:*
1: Send the fully grown Federated FRT to each data owner
*Each data owner m:*
2: Evaluate $\mathbf{A}_m^{(r)T}\mathbf{A}_m^{(r)}$ and $\mathbf{A}_m^{(r)T}\mathbf{y}_m^{(r)}$ for each leaf node $r$ and send the results to the server
*Server:*
3: Compute the coefficients $\boldsymbol{\gamma}^{(r)}$ for each leaf node by using Eq. (27)
4: Send the Federated FRT with first-order regression models at leaf nodes to each data owner

**Table 2**
Datasets description.

| Dataset | Abbreviation | Source | Dimensionality (F) | Instances (N) |
|---|---|---|---|---|
| Mortgage | MO | Keel | 15 | 1049 |
| Treasury | TR | Keel | 15 | 1049 |
| Weather Izmir | WI | Keel | 9 | 1461 |
| Kinematics | KI | Torgo | 8 | 8192 |
| Puma8NH | PU | Torgo | 8 | 8192 |
| Computer Activity | CO | Keel | 21 | 8192 |
| Delta Elevators | DE | Torgo | 6 | 9517 |
| House | HO | Keel | 16 | 22784 |
| Elevators | EL | Keel | 18 | 16599 |
| California | CA | Keel | 8 | 20460 |

First, the server transmits the global tree structure to each data owner (Algorithm 2, *line 1*). For each leaf node $r = LN^{(K)}$ each data owner $m$ computes the products $\mathbf{A}_m^{(r)^T}\mathbf{A}_m^{(r)}$ and $\mathbf{A}_m^{(r)^T}\mathbf{y}_m^{(r)}$ and transmits the results to the server (*line 2*). Then, the server can estimate the coefficients $\boldsymbol{\gamma}^{(r)}$ as follows:

$$\boldsymbol{\gamma}^{(r)} = \left(\sum_{m=1}^{M}\left(\mathbf{A}_m^{(r)^T}\mathbf{A}_m^{(r)}\right)\right)^{-1}\sum_{m=1}^{M}\left(\mathbf{A}_m^{(r)^T}\mathbf{y}_m^{(r)}\right) \tag{27}$$

Notably, Eq. (27) leads to the same results of the traditional weighted least-squared method (Eq. (5)) since it holds that

$$\mathbf{A}^{(r)^T}\mathbf{A}^{(r)} = \sum_{m=1}^{M}\left(\mathbf{A}_m^{(r)^T}\mathbf{A}_m^{(r)}\right) \tag{28}$$

and

$$\mathbf{A}^{(r)^T}\mathbf{y}^{(r)} = \sum_{m=1}^{M}\left(\mathbf{A}_m^{(r)^T}\mathbf{y}_m^{(r)}\right) \tag{29}$$

It is worth underlining that the estimation of the coefficient in the federated setting is equivalent to the centralized one. Moreover, since a client only transmits matrices $\mathbf{A}_m^{(r)^T}\mathbf{A}_m^{(r)}$ and $\mathbf{A}_m^{(r)^T}\mathbf{y}_m^{(r)}$, no disclosure of raw data takes place.

## 5. Experimental analysis

In this section, we first describe our experimental setup, in terms of learning settings, configuration parameters, exploited datasets and evaluation metrics. Then, we report and discuss the results of our experiments considering both model performances and explainability aspects.

### 5.1. Experimental setup

We employ ten well-known regression datasets from the Keel [46] and Torgo's[6] repositories, namely Mortgage, Treasury, Weather Izmir, Kinematics, Puma8NH, Computer Activity, Delta Elevators, House, Elevators and California. A summary of the datasets is reported in Table 2.

To simulate the distributed scenario, we randomly split each dataset into five parts, each with approximately the same number of instances, assuming the involvement of as many participants. Our main objective is to assess the performance of our proposed approach for FL of FRT; to this aim, consistently with the relevant literature [23,35,39,47], we consider the following three learning settings:

- **Local Learning (LL)**: each client locally learns an FRT with the traditional approach. In this setting the involved parties do not leverage any form of collaborative learning;
- **Federated Learning (FL)**: the approach described in Section 4 is adopted;
- **Centralized Learning (CL)**: the training sets of the participants are merged into a unique training set, which is used to train a global FRT. This setting represents the ideal case where scattered data can be used for training, but it evidently violates the privacy requirement.

Fig. 3 reports a schematic representation of the three learning settings.

To fairly evaluate the generalization capability of an FRT, we adopt a 5-fold cross-validation (CV): at each iteration of the CV, each client tests the FRT generated by LL, FL or CL on 1/5 of its data. Notably, the same data split is used for a fair comparison of the three learning settings.

The quality of the prediction is evaluated in terms of *Root Mean Squared Error* (RMSE) (see Eq. (30)):

$$RMSE = \sqrt{\frac{1}{N_{\text{test}}}\sum_{i=1}^{N_{\text{test}}}(y_i - \hat{y}_i)^2} \tag{30}$$

where $N_{\text{test}}$ is the number of instances 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.

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

- $T_f = 5, \forall f \in \{1, \ldots, F\}$, to ensure high semantic interpretability, as described in Section 3;
- $th_{PFGain} = 0.0001$ for all datasets to limit the complexity of the tree itself;
- $th_{min\_instances\_split} = 0.1$ for all the datasets to prevent overfitting.

Similarly to relevant recent works [23,48], a robust scaling (using 2.5 and 97.5 percentiles) is applied to the input variables to remove outliers and clip the distribution in the range $[0, 1]$.

### 5.2. Experimental results

Table 3 reports the results obtained through our experimental campaign. We show the average RMSE values on the training and test sets over 5-fold CV for each client. Furthermore we report the average results over the five clients. Results obtained in the FL setting are compared with those obtained in the LL and CL settings (the lowest average RMSE values for each dataset are presented in bold).

On the one hand, we observe that the FL approach always outperforms the LL approach in terms of RMSE on the test set. This confirms that participants benefit from collaborating in the FL process. Indeed, FL allows generating a model with a higher generalization capability than LL, preserving data privacy.

On the other hand, as expected, results obtained through FL are very similar to those obtained through CL. Indeed, we have shown and discussed in Section 4 how the federated approach be equivalent to the centralized one in both the construction of the FRT and the estimation of the linear models implemented in the leaves. The small differences found in some datasets are due to the activation of the nullification strategy to avoid the disclosure of information concerning raw data. We recall that the nullification prevents clients from sending statistics to the server for some training instances. This explains why the results obtained by the FL model are slightly different from the ones achieved by the CL model (not necessarily worse, but different). It is worth recalling, however, that CL violates the privacy requirements. Further, it introduces a high communication overhead when the dataset is very large.

The validity of the interpretable-by-design FRT model is also attested by the fact that the accuracy of the prediction is comparable to that reported in the specialized literature on the same datasets [26,49,50].

[6] https://www.dcc.fc.up.pt/~ltorgo/Regression/DataSets.html, visited July 2024.

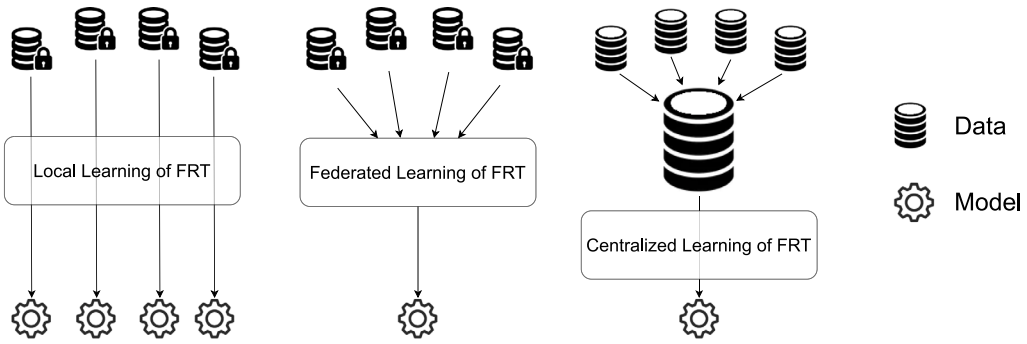


**Fig. 3.** Schematic representation of the three learning settings: (left) local learning; (center) federated learning, (right) centralized learning.

**Table 3**
Experimental results: average RMSE results.

| Client | Local | | Federated | | Centralized | | Local | | Federated | | Centralized | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | Train | *Test* | Train | *Test* | Train | *Test* | Train | *Test* | Train | *Test* | Train | *Test* |
| | Mortgage ($\times 10^{-1}$) | | | | | | Computer Activity | | | | | |
| 1 | 0.30 | *11.45* | 1.01 | *0.94* | 1.01 | *0.97* | 1.92 | *3.09* | 2.24 | *2.46* | 2.24 | *2.46* |
| 2 | 0.52 | *3.08* | 1.00 | *1.31* | 1.00 | *1.31* | 1.92 | *8.09* | 2.30 | *2.46* | 2.30 | *2.46* |
| 3 | 0.44 | *3.45* | 0.95 | *1.46* | 0.95 | *1.46* | 1.83 | *3.22* | 2.21 | *2.49* | 2.21 | *2.49* |
| 4 | 0.27 | *14.28* | 0.85 | *0.96* | 0.85 | *0.96* | 1.91 | *3.18* | 2.28 | *2.49* | 2.28 | *2.49* |
| 5 | 0.71 | *2.26* | 0.89 | *1.38* | 0.88 | *1.38* | 1.88 | *4.17* | 2.26 | *2.67* | 2.26 | *2.67* |
| Avg | 0.45 | *6.91* | 0.94 | ***1.21*** | 0.94 | ***1.22*** | 1.89 | *4.35* | 2.26 | ***2.51*** | 2.26 | ***2.51*** |
| | Treasury ($\times 10^{-1}$) | | | | | | Delta Elevators ($\times 10^{-3}$) | | | | | |
| 1 | 4.21 | *155.41* | 1.50 | *2.02* | 1.50 | *2.01* | 1.36 | *1.51* | 1.40 | *1.49* | 1.40 | *1.49* |
| 2 | 4.32 | *22.87* | 1.60 | *2.15* | 1.60 | *2.16* | 1.35 | *1.48* | 1.41 | *1.44* | 1.41 | *1.44* |
| 3 | 6.26 | *11.22* | 1.40 | *2.71* | 1.40 | *2.73* | 1.35 | *1.38* | 1.40 | *1.36* | 1.40 | *1.36* |
| 4 | 1.67 | *13.64* | 1.52 | *1.69* | 1.52 | *1.70* | 1.38 | *1.48* | 1.43 | *1.44* | 1.43 | *1.44* |
| 5 | 0.82 | *14.46* | 1.57 | *1.74* | 1.56 | *1.74* | 1.36 | *1.47* | 1.41 | *1.42* | 1.41 | *1.42* |
| Avg | 3.46 | *43.52* | 1.52 | ***2.06*** | 1.51 | *2.07* | 1.36 | *1.46* | 1.41 | ***1.43*** | 1.41 | ***1.43*** |
| | Weather Izmir | | | | | | Elevators ($\times 10^{-3}$) | | | | | |
| 1 | 0.61 | *2.04* | 0.96 | *1.13* | 0.95 | *1.13* | 2.44 | *2.71* | 2.60 | *2.52* | 2.60 | *2.52* |
| 2 | 0.64 | *1.89* | 0.98 | *1.28* | 0.98 | *1.27* | 2.36 | *2.76* | 2.52 | *2.65* | 2.52 | *2.65* |
| 3 | 0.58 | *2.02* | 0.93 | *1.34* | 0.93 | *1.38* | 2.41 | *2.92* | 2.60 | *2.72* | 2.60 | *2.72* |
| 4 | 0.63 | *1.87* | 1.04 | *1.36* | 1.04 | *1.39* | 2.41 | *2.72* | 2.58 | *2.62* | 2.58 | *2.62* |
| 5 | 0.63 | *1.63* | 1.02 | *1.24* | 1.03 | *1.23* | 2.40 | *2.89* | 2.57 | *2.74* | 2.57 | *2.74* |
| Avg | 0.62 | *1.89* | 0.99 | ***1.27*** | 0.99 | *1.28* | 2.40 | *2.80* | 2.57 | ***2.65*** | 2.57 | ***2.65*** |
| | Kinematics ($\times 10^{-1}$) | | | | | | House ($\times 10^{4}$) | | | | | |
| 1 | 1.79 | *2.06* | 1.94 | *2.04* | 1.94 | *2.04* | 3.41 | *3.99* | 3.66 | *3.64* | 3.66 | *3.64* |
| 2 | 1.80 | *2.02* | 1.92 | *1.96* | 1.92 | *1.96* | 3.50 | *4.33* | 3.73 | *3.82* | 3.73 | *3.82* |
| 3 | 1.78 | *1.95* | 1.93 | *1.92* | 1.93 | *1.92* | 3.43 | *4.20* | 3.67 | *3.89* | 3.67 | *3.89* |
| 4 | 1.78 | *1.93* | 1.94 | *1.93* | 1.94 | *1.93* | 3.35 | *4.05* | 3.58 | *3.74* | 3.58 | *3.74* |
| 5 | 1.81 | *2.04* | 1.91 | *1.97* | 1.91 | *1.97* | 3.40 | *4.10* | 3.63 | *3.76* | 3.63 | *3.76* |
| Avg | 1.79 | *2.00* | 1.93 | ***1.96*** | 1.93 | ***1.96*** | 3.42 | *4.14* | 3.65 | ***3.77*** | 3.65 | ***3.77*** |
| | Puma8NH | | | | | | California ($\times 10^{4}$) | | | | | |
| 1 | 2.93 | *3.32* | 3.12 | *3.18* | 3.12 | *3.18* | 5.44 | *6.27* | 5.71 | *6.11* | 5.71 | *6.11* |
| 2 | 2.91 | *3.26* | 3.07 | *3.18* | 3.07 | *3.18* | 5.45 | *6.03* | 5.72 | *5.85* | 5.72 | *5.85* |
| 3 | 3.00 | *3.27* | 3.16 | *3.15* | 3.16 | *3.15* | 5.56 | *5.99* | 5.82 | *5.82* | 5.82 | *5.82* |
| 4 | 2.98 | *3.40* | 3.15 | *3.29* | 3.15 | *3.29* | 5.53 | *5.88* | 5.75 | *5.71* | 5.75 | *5.71* |
| 5 | 2.98 | *3.28* | 3.15 | *3.18* | 3.15 | *3.18* | 5.52 | *6.18* | 5.72 | *5.86* | 5.72 | *6.06* |
| Avg | 2.96 | *3.31* | 3.13 | ***3.20*** | 3.13 | ***3.20*** | 5.50 | *6.07* | 5.74 | ***5.87*** | 5.74 | *5.91* |

Interestingly, FL and CL approaches are less prone to overtraining compared to the LL one. This is particularly noticeable on smaller datasets (i.e., those having fewer instances) such as MO, TR, WI, but it is in general observable for all the datasets. Indeed, the limited amount of data available for local training does not allow the creation of a model that generalizes well on the test set. On the other hand, FL and CL approaches leverage (although in different ways) all the training instances for the induction of the FRT, thus increasing the generalization capability of the generated FRT.

Table 3 gives an indication of the relative performance between LL, FL and CL by showing the average RMSE values over cross-validation for each client. A more detailed explanation can be derived from Fig. 4, which plots, for each dataset and for each iteration of the CV, the empirical cumulative distribution functions (ECDFs) of: (i) the difference between the RMSE values of the FL setting and the RMSE values of the LL setting (indicated as $\Delta_{FL-LL}$, dark blue), and (ii) the difference between the RMSE values of the FL setting and the RMSE values of the CL setting (indicated as $\Delta_{FL-CL}$, light blue).

Each plot is made up of 25 points (5 clients $\times$ 5 iterations of the CV) sorted for increasing values. Evidently, if a point lies in the negative half-plane it means that the first term (FL-RMSE) is lower (and thus better) than the second term (either LL- or CL-RMSE). Finally, it can be observed that the dark blue curve lies almost always in the negative half-plane, regardless of the dataset considered, confirming that FL generally outperforms LL. It is equally evident that the light blue curve tends to trail the y-axis, meaning that the difference between FL and CL is limited or negligible.

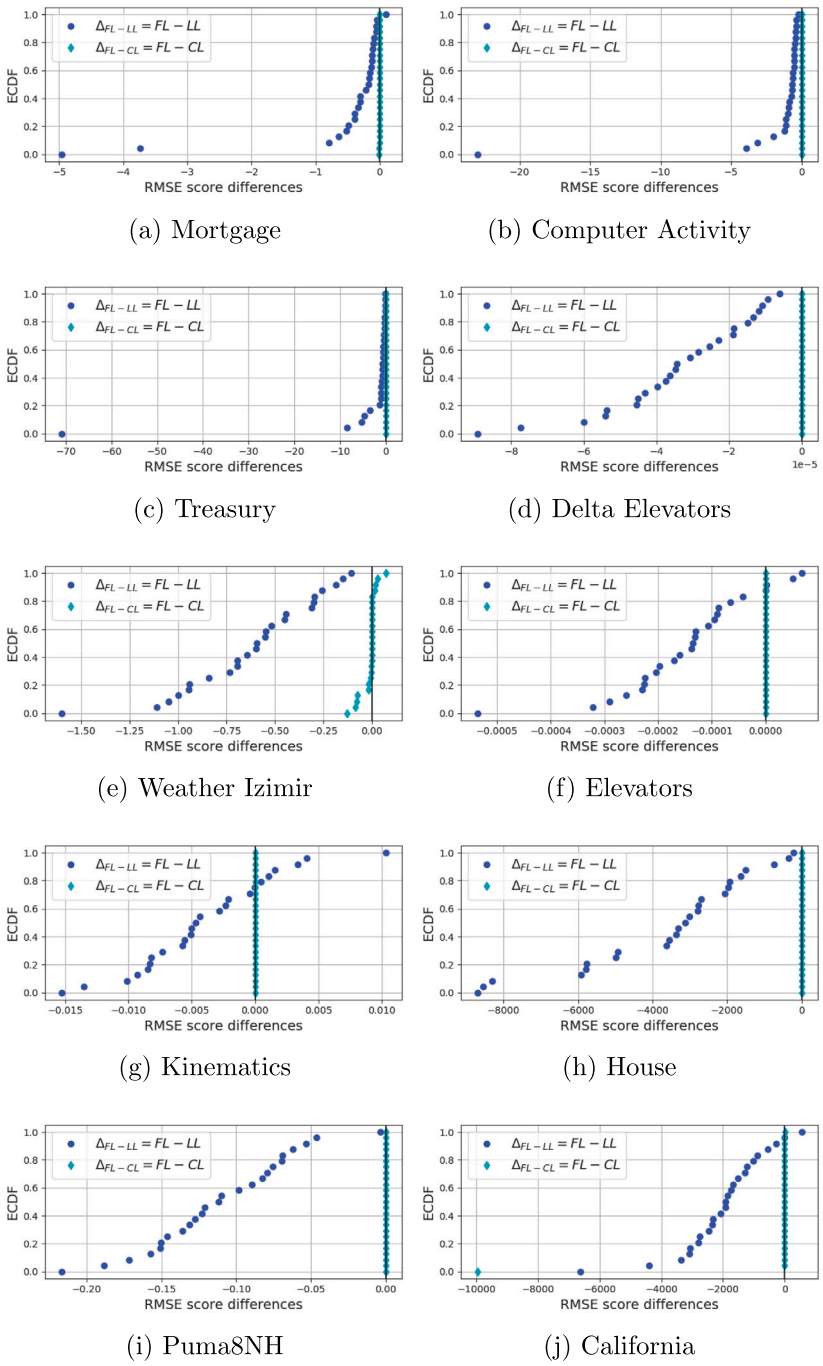


**Fig. 4.** Empirical cumulative distribution function (ECDF) of the differences of RMSE scores between FL and LL ($\Delta_{FL-LL}$, dark blue circles) and between FL and CL ($\Delta_{FL-LL}$, light blue diamonds).

A statistical test, namely the pairwise Wilcoxon signed-rank test [51], is adopted to compare the RMSE values across the different learning settings for each dataset: we evaluate whether there exist statistical differences in the RMSE values between FL and LL, and between FL and CL. For each learning setting and for each dataset the distribution consists of 25 RMSE values. Note that for 6 out of 10 datasets the adoption of the Wilcoxon test between FL and CL is pointless since the results obtained with the two approaches are exactly equivalent. Results are reported in Table 4.

$R^{\pm}$ denotes the sum of the ranks of the differences above or below zero, whichever is smaller. The statistical hypothesis of equivalence can be rejected whenever the $p$-value is lower than the level of significance $\alpha$. The typical choice of $\alpha = 0.05$, combined with the evidence from Table 3 and Fig. 4, leads to the expected outcomes that (i) FL statistically outperforms LL on all datasets, and (ii) the hypothesis of equivalence between FL and CL cannot be rejected on any dataset. In other words, there is no significant difference between the generalization capability of the FRTs learned according to the FL and CL paradigms. The latter however requires data centralization and is simply unfeasible in decentralized scenarios where privacy preservation is a mandatory constraint. Healthcare and finance are glaring examples of domains in which privacy and transparency constitute imperative needs and represent real-world scenarios in which our approach can be applied. As a regression model, the proposed FRT can also be applied for timeseries forecasting tasks in industrial applications: an example

**Table 4**
Results of the Wilcoxon Signed-Rank test. A $p$-value lower than the significance level ($\alpha = 0.05$) is highlighted in bold.

| Dataset | FL vs. LL | | FL vs. CL | |
|---|---|---|---|---|
| | $R^{\pm}$ | $p$-value | $R^{\pm}$ | $p$-value |
| Mortgage | 6.0 | **0.0000** | 100.0 | 0.0957 |
| Treasury | 0.0 | **0.0000** | 126.0 | 0.3388 |
| Weather Izmir | 0.0 | **0.0000** | 133.0 | 0.4418 |
| Kinematics | 54.0 | **0.0025** | *Exactly equivalent* | |
| Puma8NH | 0.0 | **0.0000** | *Exactly equivalent* | |
| Computer Activity | 0.0 | **0.0000** | *Exactly equivalent* | |
| Delta Elevators | 0.0 | **0.0000** | *Exactly equivalent* | |
| Elevators | 13.0 | **0.0000** | *Exactly equivalent* | |
| House | 0.0 | **0.0000** | *Exactly equivalent* | |
| California | 4.0 | **0.0000** | 126.0 | 0.3388 |

within the context of next generation wireless networks is discussed in Section 5.4

The experimental outcomes also provide a cue for possible future developments of this work aimed at improving performance metrics of our Federated FRT: the federation strategy does not allow much room for improvement since the FRT built with the federated approach is substantially equivalent to the centralized one. A further optimization of the model therefore involves the tuning of the model configuration parameters, which characterize its operation. As an example, a federated data-driven approach for fuzzy partitioning of input variables could be investigated. In this work we consider a uniform triangular fuzzy partition: on the one hand, this allows for an identical partitioning for all participants and ensures high interpretability, as discussed in Section 1; on the other hand, however, the current approach is not data-driven, i.e., it does not exploit the actual distribution of data for partitioning variables.

### 5.3. Interpretability of FRT

Besides privacy preservation, also model interpretability is a key pillar for the trustworthiness of an AI system. It is widely acknowledged that the general concept of interpretability can be dissected into two aspects: global and local interpretability. As discussed in [26] *global* refers to the structural properties of the model, whereas *local* refers to how the inference process is carried out for any single instance.

In tree-based models, the global interpretability can be quantified exploiting some measure of model complexity [4]. The higher the complexity, the lower the interpretability: in fact, the less complex and more compact the RT, the easier to understand its internal behavior for humans. A first proxy to model complexity can be given by the total number of nodes in the tree (#Nodes). Since each path from the root node to a leaf node is a sequence of tests on input variables, an induced tree can be transformed into a rule-base consisting of a number of rules equal to the number of leaves (#Leaves). The maximum length of the antecedent part of a rule in the rule-base is given by the maximum depth of the tree (Depth). Although the number of nodes and leaves, and the tree depth can be used to assess model complexity, they do not take into account the inherent complexity of the regression model associated with each leaf node. Identical trees, with same depth, number of nodes and number of leaves, would actually have different complexity if they differ in the order of the model implemented in the leaves: interpretability is higher when the leaves are associated with constant values, whereas it decreases as the order of the regression models increases. To capture this aspect, we also consider the total number of parameters $C_{FRT}$ of the model, which can be computed as in [26]:

$$C_{FRT} = IN_{FRT} + \sum_{LN \in FRT} N_{Coeff}(LN) \tag{31}$$

where $IN$ is the total number of internal nodes, $LN$ is a generic leaf node in the FRT, and $N_{Coeff}(LN)$ is the number of the coefficients of the linear model used in $LN$. As an example, consider the toy FRT reported in Fig. 2: the total number of nodes is #Nodes = 10, the number of leaves is #Leaves = 7, and Depth = 2. Given the number of internal nodes equal to 3 and that each first-order regression model has 3 parameters, the total number of parameters is $C_{FRT} = 3 + 7 \times 3 = 24$. The quantities defined above (#Nodes, #Leaves, Depth, $C_{FRT}$) are evaluated for the FRTs generated by using LL, FL and CL, and are reported in Table 5 as average values over cross-validation.

Obviously the values of the metrics vary from one dataset to another, as they underlie different tasks, and have different numbers of features and instances. Table 5 suggests, however, that induced trees entails in general a limited complexity, with a number of rules in the range of a hundred or less (#Leaves). With the exception of the California dataset, the maximum depth (Depth) is always lower than the dimensionality of the dataset (number of features $F$). This is relevant for interpretability for two reasons. First, shorter antecedents result in simpler and therefore more interpretable rules. Second, this suggests that the FRT induction algorithm has inherently performed a feature selection process, getting rid of those that are unimportant for the antecedent part of the rules. We recall that each leaf node implements a first-order regression model considering the whole set of $F$ input variables, regardless of which ones are tested in the path from the root to the leaf: this explains the high number of parameters ($C_{FRT}$), generally an order of magnitude higher than the number of nodes and leaves.

The comparative analysis of the three learning settings leads to the following observations. First, we have further evidence that the FL approach is essentially equivalent to the CL one, as the average values of the metrics are often equal or similar. Second, it is interesting to observe that the FRTs obtained by FL and CL have a complexity similar to the ones generated by LL. The maximum discrepancy holds for smaller datasets (e.g., in the case of Mortgage the complexity metrics are 20/25% higher in FL/CL compared to LL), but the differences are negligible in most cases and sometimes the local model is even more complex than the federated or centralized. Thus, we can argue that the gain in accuracy obtained by FL compared to LL occurs without jeopardizing the global interpretability of the models.

Local interpretability of the proposed FRT is highly enhanced through the adoption of the maximum matching policy: in fact, it ensures that only one rule is considered in the inference process, thus providing an easy and intuitive explanation of how an output is generated. The antecedent part of the rule consists in a sequence of tests which isolates a specific region of the attribute space. The coefficient of the linear model in the consequent part reports the effect of each input variable on the output value. In the following we report a rule extracted from an FRT generated by FL on the California dataset, as an example:

$$\begin{aligned} R_k : \textbf{IF } & MedianIncome \ is \ Medium \ \textbf{AND} \ HousingMedianAge \ is \\ & VeryHigh \\ \textbf{THEN} : \ & MedianHouseValue = 0.84 - 1.16 \cdot Longitude \\ & - 1.51 \cdot Latitude \\ & + 0.55 \cdot HousingMedianAge - 0.10 \cdot TotalRooms \\ & + 1.02 \cdot TotalBedrooms \\ & - 0.78 \cdot Population + 0.13 \cdot Households + 0.78 \cdot MedianIncome \end{aligned} \tag{32}$$

One can notice that, in case of medium median income and very high housing median age, the target variable is negatively affected mostly by the coordinates, latitude (coefficient −1.16) and longitude (coefficient −1.51), and by the population (coefficient −0.78), whereas it is positively affected mostly by the total number of bedrooms (coefficient +1.02) and by the median income (coefficient +0.78).

**Table 5**
Model complexity measures: average values over cross-validation.

| Dataset | Local learning | | | | Federated learning | | | | Centralized learning | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | #Nodes | #Leaves | Depth | $C_{FRT}$ | #Nodes | #Leaves | Depth | $C_{FRT}$ | #Nodes | #Leaves | Depth | $C_{FRT}$ |
| MO | 156.28 | 93.68 | 11.52 | 1561.48 | 184.40 | 128.40 | 12.00 | 2110.40 | 185.00 | 129.20 | 12.20 | 2123.00 |
| TR | 89.04 | 56.96 | 8.96 | 943.44 | 98.20 | 70.20 | 8.00 | 1151.20 | 98.80 | 70.60 | 8.00 | 1157.80 |
| WI | 109.16 | 77.48 | 7.36 | 806.48 | 112.80 | 88.00 | 6.80 | 904.80 | 119.60 | 93.60 | 7.40 | 962.00 |
| KI | 36.60 | 30.28 | 2.76 | 278.84 | 30.00 | 25.00 | 2.00 | 230.00 | 30.00 | 25.00 | 2.00 | 230.00 |
| PU | 39.40 | 32.52 | 2.88 | 299.56 | 31.00 | 25.80 | 2.20 | 237.40 | 31.00 | 25.80 | 2.20 | 237.40 |
| CO | 83.64 | 63.44 | 6.04 | 1415.88 | 86.20 | 69.20 | 5.80 | 1539.40 | 86.20 | 69.20 | 5.80 | 1539.40 |
| DE | 40.12 | 33.08 | 3.12 | 238.60 | 35.00 | 29.00 | 3.00 | 209.00 | 35.00 | 29.00 | 3.00 | 209.00 |
| EL | 77.36 | 61.12 | 7.08 | 1177.52 | 81.60 | 64.40 | 9.60 | 1240.80 | 81.60 | 64.40 | 9.60 | 1240.80 |
| HO | 95.00 | 76.60 | 6.28 | 1320.60 | 97.00 | 78.60 | 6.20 | 1354.60 | 97.00 | 78.60 | 6.20 | 1354.60 |
| CA | 118.08 | 91.76 | 8.00 | 852.16 | 121.60 | 95.20 | 8.00 | 883.20 | 123.60 | 97.20 | 8.00 | 901.20 |

**Table 6**
RMSE scores for the QoE case study: fine-grained results on the test set for the three learning settings. Best values highlighted in bold.

| Client | Run 1 | | | Run 2 | | | Run 3 | | | Run 4 | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | LL | FL | CL | LL | FL | CL | LL | FL | CL | LL | FL | CL |
| 1 | 0.230 | **0.220** | **0.220** | 0.300 | **0.286** | **0.286** | 0.262 | **0.242** | 0.243 | 0.279 | 0.269 | **0.259** |
| 2 | 0.221 | 0.191 | **0.189** | 0.265 | 0.267 | **0.264** | 0.306 | 0.308 | **0.296** | 0.287 | **0.259** | 0.260 |
| 3 | 0.237 | **0.222** | 0.223 | 0.225 | 0.223 | **0.222** | 0.182 | **0.166** | 0.167 | 0.290 | 0.290 | **0.282** |
| 4 | 0.285 | 0.263 | **0.260** | 0.317 | **0.301** | 0.302 | **0.284** | 0.300 | 0.294 | **0.246** | 0.253 | 0.249 |
| 5 | **0.209** | 0.221 | 0.220 | 0.155 | **0.142** | 0.144 | 0.248 | **0.218** | **0.218** | 0.248 | **0.244** | **0.244** |
| 6 | 0.279 | **0.246** | 0.248 | 0.254 | 0.246 | **0.243** | 0.256 | 0.250 | **0.235** | 0.320 | **0.302** | **0.302** |
| 7 | 0.196 | **0.191** | **0.191** | 0.416 | 0.188 | **0.176** | 0.260 | 0.254 | **0.253** | 0.217 | **0.198** | **0.198** |
| 8 | 0.276 | 0.270 | **0.266** | 0.244 | **0.241** | 0.242 | 0.322 | **0.290** | 0.292 | **0.244** | 0.248 | 0.248 |
| 9 | 0.225 | **0.151** | 0.152 | 0.221 | 0.223 | **0.216** | **0.273** | 0.286 | 0.289 | 0.240 | 0.224 | **0.221** |
| 10 | 0.333 | **0.302** | 0.305 | **0.263** | 0.265 | 0.268 | 0.229 | 0.221 | **0.212** | 0.308 | **0.258** | **0.258** |
| 11 | **0.239** | 0.248 | 0.249 | 0.240 | 0.229 | **0.223** | 0.282 | **0.267** | 0.270 | 0.324 | **0.263** | 0.265 |
| 12 | 0.246 | 0.238 | **0.234** | 0.200 | **0.181** | 0.188 | **0.190** | 0.198 | 0.192 | 0.164 | 0.158 | **0.156** |
| 13 | 0.232 | **0.182** | 0.184 | 0.424 | **0.268** | 0.277 | 0.193 | 0.186 | **0.183** | 0.264 | **0.256** | 0.259 |
| 14 | 0.295 | 0.202 | **0.198** | 0.226 | 0.186 | **0.184** | 0.339 | **0.332** | 0.339 | 0.269 | **0.262** | 0.266 |
| 15 | 0.290 | 0.280 | **0.271** | 0.260 | 0.212 | **0.209** | 0.259 | 0.240 | **0.238** | 0.255 | **0.243** | 0.249 |

### 5.4. A next-generation wireless network case study: quality of experience forecasting

In this section we report on the application of the proposed Federated FRT for a regression task in an automative case study within the context of next generation wireless networks. The objective is to forecast the Quality of Experience (QoE) perceived on vehicular User Equipment (UE) while experiencing a video stream. Being able to correctly predict the perceived quality of the video is crucial for enabling innovative services such as *see-through* or *tele-operated driving* [29].

The dataset[7] used for this case study has been presented in [52]. Some preliminary results have been reported in some of our recent works, with specific focus on the adoption of a federated approach for learning TSK-FRBS [35,48] or on the centralized learning setting, both in *static* [52] and *streaming* [53] scenarios.

In the following we summarize the key aspects of the experimental setup: the dataset results from a simulation campaign and consists of a set of Quality of Service (QoS) and QoE metrics collected for 15 vehicles (clients, or participants) while acquiring a video stream in motion. In each of the 24 independent replicas (or runs) of the simulation, video-streams flow from a video server towards the vehicles and each vehicle collects metrics for 120 s. As in previous works [35,48], we consider, for each client, 20 runs as the training set and 4 runs as the test set. For a fair comparison of the results, we adopt the same preprocessing steps carried out in the related works. In summary, the dataset consists of 15 features and 42 700 instances (35 487 for training and 7213 for test), horizontally partitioned across 15 clients.

Experiments with the proposed FRT were carried out considering the same learning settings and the same configuration parameters discussed in Section 5.1. Table 6 reports the results in terms of RMSE for the three learning settings for each client and for each test run.

[7] Dataset publicly available: http://docenti.ing.unipi.it/g.nardini/ai6g_qoe_dataset.html, visited July 2024.

**Table 7**
RMSE scores for the QoE case study: average test results and comparison with state-of-art approaches.

| | LL | FL | CL |
|---|---|---|---|
| FRT | 0.260 | 0.239 | 0.238 |
| TSK-FRBS [48] | 0.287 | 0.251 | 0.235 |
| MLP-NN [48] | 0.245 | 0.242 | 0.230 |

A glance at the table confirms what has been extensively discussed in the previous section: RMSE values are generally lower for the FL setting compared to the LL one; furthermore, FL and CL obtain very similar results.

In the following we report the average results obtained by the proposed FRT and two alternative state-of-art approaches (discussed in [48]), namely TSK-FRBS and Multi Layer Perceptron Neural Network (MLP-NN). The former represents one of the first approaches investigated in the field of Fed-XAI, and consists of a highly interpretable by-design model; results are evaluated in terms of test RMSE scores (see Table 7).

It can be observed that the proposed FRT achieves competitive performance. Differently from the TSK-FRBS, our proposed procedure for Federated FRT proves to be lossless compared to the centralized approach, which is simply not viable in most real-world case studies. Although in the LL setting the FRT exhibits a higher generalization capability compared to the TSK-FRBS, the discrepancy from the FL setting is still significant.

Results of the Federated FRT are better than those of Federated TSK-FRBS and comparable to the ones of Federated MLP-NN. The approach based on neural network, however, does not guarantee the same level of interpretability of FRT and TSK-FRBS, neither from a local nor a global perspective. The comparison with the black-box model, resulting in similar performance metrics, suggests that our FRT represents a viable

**Table 8**
Complexity measures for the interpretable by-design models, namely FRT and TSK-FBRS, on the QoE case study.

| | Number of rules | | | Avg. rule length | | | Max. rule length | | | Total number of parameters | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | LL | FL | CL | LL | FL | CL | LL | FL | CL | LL | FL | CL |
| FRT | 178.1 | 181.0 | 220.0 | 6.7 | 7.2 | 7.2 | 11.5 | 10.0 | 11.0 | 4086.0 | 4 193.0 | 5 107.0 |
| TSK-FRBS [48] | 289.1 | 997.0 | 997.0 | 15.0 | 15.0 | 15.0 | 15.0 | 15.0 | 15.0 | 8962.1 | 30 907.0 | 30 907.0 |

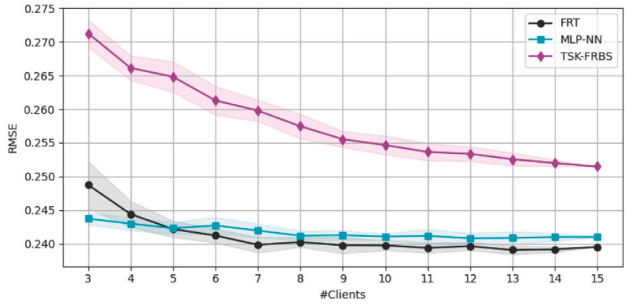


**Fig. 5.** Average RMSE values obtained by the federated FRT, MLP-NN and TSK-FRBS approaches w.r.t. the number of clients randomly selected at each FL round. Shaded region indicates the standard deviation.

approach in real-world scenarios where privacy and explainability are pursued together.

The interpretability of FRT and TSK-FRBS approaches is assessed through several complexity measures. The comparison is enabled since an induced FRT can be represented as a collection of rules. Specifically, we evaluate the total number of rules (which corresponds to #Leaves in an FRT), the average and maximum *rule length*, intended as the number of conditions in the antecedent part of each rule (which correspond to average and maximum Depth of an FRT), and the total number of parameters (which is computed as in Eq. (31) for an FRT).

The complexity of TSK-FRBSs and FRTs provides further insights on their interpretability: Table 8 highlights that FRTs are not only more accurate but also less complex than TSK-FRBSs, featuring a lower number of rules regardless of the learning setting. Also, as highlighted in Section 2, FRTs have an inherent feature selection capability, whereas TSK-FRBSs always consider all the features in the antecedent part of the rules. This motivates the difference between the approaches in terms of rule length and total number of parameters.

Finally, we assess the sensitivity of the proposed Federated FRT and the two state of art approaches, MLP-NN and TSK-FRBS, with respect to the number of clients participating at each round. We simulate a scenario in which, at each FL round, the central server does not receive updates from a subset of the clients, because they become unreachable or because they are deliberately excluded from the FL process to cut the communication costs. In our experiments, at the beginning of each round, we randomly select a fixed number of clients that participate in the learning stage (varying the number in the range from 3 to 15). Notably, the final federated model is shared with all the clients for testing.

We performed 10 repetitions varying the seed used for randomly sampling the clients. Fig. 5 reports the results in terms of average test RMSE scores.

As expected, the performance of the Federated FRT tends to increase with the increase in the number of selected clients, indicating that in general a high number of participants is beneficial. Slightly lower performance occurs when the number of clients is significantly low (3, 4) but it stabilizes around the average value of 0.240 already when less than half of the clients participate in the FL process. This outcome suggests that, in this experimental case study, our federated learning of FRT is reasonably robust with respect to the fraction of clients participating in the process. The federated learning of MLP-NN is also robust to the number of participants: the quite simple model presented in [48] achieves performance comparable to that of the FRT on the QoE dataset even when only a few clients participate in the FL process. Federated TSK-FRBS, instead, is quite sensitive to the number of participants and its performance metrics are consistently worse than the other models.

## 6. Conclusion

In this paper, we have proposed a novel approach for Federated Learning (FL) of Fuzzy Regression Trees (FRTs) over horizontally partitioned datasets. Revisiting the FRT induction algorithm for the federated setting makes it possible to build an accurate model from decentralized data without violating data owners' privacy. At the same time, the federated FRT retains the property of transparency which is typical of tree-based models. The proposed approach represents indeed a step forward towards trustworthy AI, by fulfilling the two key technical requirements of privacy and explainability of AI systems. The federated FRT induction procedure matches the traditional one, applied on a single dataset as a whole, and is shown to provide a faithful approximation of it: during the iterative tree-growing stage, the server updates the structure of the global federated FRT based on aggregated statistics received at each round from the data owners. Furthermore, a strategy of statistics nullification is implemented on the clients' side to avoid any disclosure of raw data in some specific cases. Once the FRT structure has been determined, the server collects aggregated statistics from the data owner to estimate the parameters of the linear regression model to be implemented in each leaf node. A thorough experimental analysis on several benchmark datasets confirms that the proposed FL procedure is substantially equivalent to the traditional centralized one, which, however, poses the (often unfeasible) requirement of gathering scattered data on a single node. In addition, the federated FRT statistically outperforms models learned by using solely data stored on each client, attesting the benefit of collaborative learning. The resulting federated FRTs are also highly interpretable both from a global and a local perspective. As for the former property, the model can be represented as a set of rules, in which the antecedent part is expressed in linguistic terms thanks to the adoption of concepts from fuzzy set theory. As for the latter property, the maximum matching policy ensures that the output generated for a given input instance is determined by just a single rule, i.e. the one most strongly activated. The experimental analysis is extended with the application of the proposed federated FRT to the task of Quality of Experience forecasting in a realistic automotive case-study: the effectiveness of the proposed method is demonstrated also in relation to other state-of-the-art approaches including neural networks.

As future work, we intend to investigate approaches for tuning FRT configuration parameters, e.g., those determining the fuzzy partitioning of each input variable, in the federated setting. Another interesting future development of this work concerns the integration of post-hoc explainability techniques with black-box models generated by FL. In particular, we aim to compare our federated FRT in terms of accuracy and explainability with these integrated systems.

## Funding

This work has been partly funded by the PON 2014–2021 "Research and Innovation", DM MUR 1062/2021, Project title: "Progettazione e

sperimentazione di algoritmi di federated learning per data stream mining", PNRR - M4C2 - Investimento 1.3, Partenariato Esteso PE00000013 - "FAIR - Future Artificial Intelligence Research" - Spoke 1 "Human-centered AI" and the PNRR "Tuscany Health Ecosystem" (THE) (Ecosistemi dell'Innovazione) - Spoke 6 - Precision Medicine & Personalized Healthcare (CUP I53C22000780001) under the NextGeneration EU programme, and by the Italian Ministry of University and Research (MUR) in the framework of the FoReLab and CrossLab projects (Departments of Excellence).

## CRediT authorship contribution statement

**José Luis Corcuera Bárcena:** Software, Methodology, Investigation, Data curation, Conceptualization. **Pietro Ducange:** Writing – review & editing, Supervision, Methodology, Investigation, Conceptualization. **Francesco Marcelloni:** Writing – review & editing, Writing – original draft, Supervision, Methodology, Investigation, Conceptualization. **Alessandro Renda:** Writing – review & editing, Writing – original draft, Methodology, Investigation, Conceptualization.

## Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

## Data availability

The datasets Mortgage, Treasury, Weather Izmir, Computer Activity, House, Elevators, California are publicly available at https://sci2s.ugr.es/keel/datasets.php. The datasets Kinematics, Puma8NH, Delta Elevators, are publicly available at https://www.dcc.fc.up.pt/~ltorgo/Regression/DataSets.html. The Quality of Experience forecasting dataset is publicly available at http://docenti.ing.unipi.it/g.nardini/ai6g_qoe_dataset.html.

## References

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

[2] Q. Yang, Y. Liu, T. Chen, Y. Tong, Federated machine learning: Concept and applications, ACM Trans. Intell. Syst. Technol. 10 (2) (2019) 1–19.

[3] A.B. Arrieta, N. Diaz-Rodriguez, J. Del Ser, A. Bennetot, S. Tabik, A. Barbado, S. Garcia, S. Gil-Lopez, D. Molina, R. Benjamins, et al., Explainable Artificial Intelligence (XAI): Concepts, taxonomies, opportunities and challenges toward responsible AI, Inf. Fusion 58 (2020) 82–115.

[4] R. Guidotti, A. Monreale, S. Ruggieri, F. Turini, F. Giannotti, D. Pedreschi, A survey of methods for explaining black box models, ACM Comput. Surv. 51 (5) (2018) http://dx.doi.org/10.1145/3236009.

[5] R. Guidotti, A. Monreale, D. Pedreschi, F. Giannotti, Principles of explainable artificial intelligence, in: Explainable AI Within the Digital Transformation and Cyber Physical Systems, Springer, 2021, pp. 9–31.

[6] S. Ali, T. Abuhmed, S. El-Sappagh, K. Muhammad, J.M. Alonso-Moral, R. Confalonieri, R. Guidotti, J. Del Ser, N. Díaz-Rodríguez, F. Herrera, Explainable Artificial Intelligence (XAI): What we know and what is left to attain Trustworthy Artificial Intelligence, Inf. Fusion 99 (2023) 101805, http://dx.doi.org/10.1016/j.inffus.2023.101805, URL https://www.sciencedirect.com/science/article/pii/S1566253523001148.

[7] E. Commission, C. Directorate-General for Communications Networks, Technology, Ethics Guidelines for Trustworthy AI, Publications Office, 2019, http://dx.doi.org/10.2759/177365.

[8] J. Konečný, H.B. McMahan, D. Ramage, P. Richtárik, Federated optimization: Distributed machine learning for on-device intelligence, 2016, http://dx.doi.org/10.48550/ARXIV.1610.02527.

[9] J.L.C. Bárcena, M. Daole, P. Ducange, F. Marcelloni, A. Renda, F. Ruffini, A. Schiavo, Fed-XAI: Federated learning of explainable artificial intelligence models, in: XAI.It 2022: 3rd Italian Workshop on Explainable Artificial Intelligence, Co-Located with AI*IA 2022, 2022, URL https://ceur-ws.org/Vol-3277/paper8.pdf.

[10] R. López-Blanco, R.S. Alonso, A. González-Arrieta, P. Chamoso, J. Prieto, Federated learning of explainable artificial intelligence (FED-XAI): A review, in: S. Ossowski, P. Sitek, C. Analide, G. Marreiros, P. Chamoso, S. Rodríguez (Eds.), Distributed Computing and Artificial Intelligence, 20th International Conference, Springer Nature Switzerland, Cham, 2023, pp. 318–326.

[11] J. Fiosina, Interpretable privacy-preserving collaborative deep learning for taxi trip duration forecasting, in: International Conference on Vehicle Technology and Intelligent Transport Systems, International Conference on Smart Cities and Green ICT Systems, Springer, 2022, pp. 392–411.

[12] P. Chen, X. Du, Z. Lu, J. Wu, P.C. Hung, EVFL: An explainable vertical federated learning for data-oriented Artificial Intelligence systems, J. Syst. Archit. 126 (2022) 102474, http://dx.doi.org/10.1016/j.sysarc.2022.102474, URL https://www.sciencedirect.com/science/article/pii/S1383762122000583.

[13] G. Wang, Interpret federated learning with shapley values, 2019, arXiv preprint arXiv:1905.04519.

[14] A. Bogdanova, A. Imakura, T. Sakurai, DC-SHAP method for consistent explainability in privacy-preserving distributed machine learning, Hum.-Centric Intell. Syst. 3 (3) (2023) 197–210, http://dx.doi.org/10.1007/s44230-023-00032-4.

[15] C. Rudin, Stop explaining black box machine learning models for high stakes decisions and use interpretable models instead, Nat. Mach. Intell. 1 (5) (2019) 206–215.

[16] X. Zhu, D. Wang, W. Pedrycz, Z. Li, Horizontal federated learning of Takagi–Sugeno fuzzy rule-based models, IEEE Trans. Fuzzy Syst. 30 (9) (2022) 3537–3547, http://dx.doi.org/10.1109/TFUZZ.2021.3118733.

[17] A. Wilbik, P. Grefen, Towards a federated fuzzy learning system, in: 2021 IEEE International Conference on Fuzzy Systems, FUZZ-IEEE, IEEE, 2021, pp. 1–6.

[18] H. Ludwig, N. Baracaldo, G. Thomas, Y. Zhou, A. Anwar, S. Rajamoni, Y. Ong, J. Radhakrishnan, A. Verma, M. Sinn, M. Purcell, A. Rawat, T. Minh, N. Holohan, S. Chakraborty, S. Whitherspoon, D. Steuer, L. Wynter, H. Hassan, S. Laguna, M. Yurochkin, M. Agarwal, E. Chuba, A. Abay, IBM federated learning: an enterprise framework white paper V0.1, 2020, http://dx.doi.org/10.48550/ARXIV.2007.10987, URL https://arxiv.org/abs/2007.10987.

[19] Y. Wu, S. Cai, X. Xiao, G. Chen, B.C. Ooi, Privacy preserving vertical federated learning for tree-based models, 13 (12) (2020) 2090–2103.

[20] A. Archetti, M. Matteucci, Federated survival forests, in: 2023 International Joint Conference on Neural Networks (IJCNN), IEEE, 2023, http://dx.doi.org/10.1109/ijcnn54540.2023.10190999.

[21] M. Polato, R. Esposito, M. Aldinucci, Boosting the federation: Cross-silo federated learning without gradient descent, in: 2022 International Joint Conference on Neural Networks, IJCNN, IEEE, 2022, http://dx.doi.org/10.1109/IJCNN55064.2022.9892284.

[22] N. Bussmann, P. Giudici, D. Marinelli, J. Papenbrock, Explainable machine learning in credit risk management, Comput. Econ. 57 (1) (2021) 203–216, http://dx.doi.org/10.1007/s10614-020-10042-0.

[23] J.L. Corcuera Bárcena, P. Ducange, F. Marcelloni, A. Renda, F. Ruffini, Federated learning of explainable artificial intelligence models for predicting Parkinson's disease progression, in: L. Longo (Ed.), Explainable Artificial Intelligence, Springer Nature Switzerland, Cham, 2023, pp. 630–648.

[24] J. Cózar, F. Marcelloni, J.A. Gámez, L. de la Ossa, Building efficient fuzzy regression trees for large scale and high dimensional problems, J. Big Data 5 (1) (2018) 1–25.

[25] A. Segatori, F. Marcelloni, W. Pedrycz, On distributed fuzzy decision trees for big data, IEEE Trans. Fuzzy Syst. 26 (1) (2017) 174–192.

[26] A. Bechini, J.L.C. Bárcena, P. Ducange, F. Marcelloni, A. Renda, Increasing accuracy and explainability in fuzzy regression trees: An experimental analysis, in: 2022 IEEE International Conference on Fuzzy Systems, FUZZ-IEEE, IEEE, 2022, pp. 1–8.

[27] A. Fernandez, F. Herrera, O. Cordon, M.J. del Jesus, F. Marcelloni, Evolutionary fuzzy systems for explainable artificial intelligence: Why, when, what for, and where to? IEEE Comput. Intell. Mag. 14 (1) (2019) 69–81.

[28] M.J. Gacto, R. Alcalá, F. Herrera, Interpretability of linguistic fuzzy rule-based systems: An overview of interpretability measures, Inform. Sci. 181 (20) (2011) 4340–4360.

[29] A. Renda, P. Ducange, F. Marcelloni, D. Sabella, M.C. Filippou, G. Nardini, G. Stea, A. Virdis, D. Micheli, D. Rapone, et al., Federated learning of explainable AI models in 6G systems: Towards secure and automated vehicle networking, Information 13 (8) (2022) 395.

[30] H. Zhu, J. Xu, S. Liu, Y. Jin, Federated learning on non-IID data: A survey, Neurocomputing 465 (2021) 371–390, http://dx.doi.org/10.1016/j.neucom.2021.07.098, URL https://www.sciencedirect.com/science/article/pii/S0925231221013254.

[31] L. Yang, J. Huang, W. Lin, J. Cao, Personalized federated learning on non-IID data via group-based meta-learning, ACM Trans. Knowl. Discov. Data 17 (4) (2023) http://dx.doi.org/10.1145/3558005.

[32] R. Gosselin, L. Vieu, F. Loukil, A. Benoit, Privacy and security in federated learning: A survey, Appl. Sci. 12 (19) (2022) http://dx.doi.org/10.3390/app12199901, URL https://www.mdpi.com/2076-3417/12/19/9901.

[33] L. Li, Y. Fan, M. Tse, K.-Y. Lin, A review of applications in federated learning, Comput. Ind. Eng. 149 (2020) 106854.

[34] S. Aich, N.K. Sinai, S. Kumar, M. Ali, Y.R. Choi, M.-I. Joo, H.-C. Kim, Protecting personal healthcare record using blockchain & federated learning technologies, in: 2022 24th International Conference on Advanced Communication Technology, ICACT, IEEE, 2022, pp. 109–112.

[35] J.L.C. Bárcena, P. Ducange, F. Marcelloni, G. Nardini, A. Noferi, A. Renda, F. Ruffini, A. Schiavo, G. Stea, A. Virdis, Enabling federated learning of explainable AI models within beyond-5G/6G networks, Comput. Commun. (2023) http://dx.doi.org/10.1016/j.comcom.2023.07.039, URL https://www.sciencedirect.com/science/article/pii/S0140366423002724.

[36] L. Corbucci, R. Guidotti, A. Monreale, Explaining black-boxes in federated learning, in: L. Longo (Ed.), Explainable Artificial Intelligence, Springer Nature Switzerland, Cham, 2023, pp. 151–163.

[37] S.M. Lundberg, S.-I. Lee, A unified approach to interpreting model predictions, in: I. Guyon, U.V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, R. Garnett (Eds.), Advances in Neural Information Processing Systems, Vol. 30, Curran Associates, Inc., 2017.

[38] T. Takagi, M. Sugeno, Fuzzy identification of systems and its applications to modeling and control, IEEE Trans. Syst. Man Cybern. (1) (1985) 116–132.

[39] J.L.C. Bárcena, P. Ducange, A. Ercolani, F. Marcelloni, A. Renda, An approach to federated learning of explainable fuzzy regression models, in: 2022 IEEE International Conference on Fuzzy Systems, FUZZ-IEEE, IEEE, 2022, pp. 1–8.

[40] A. Bechini, M. Daole, P. Ducange, F. Marcelloni, A. Renda, An application for federated learning of XAI models in edge computing environments, in: 2023 IEEE International Conference on Fuzzy Systems, FUZZ-IEEE, IEEE, 2023, pp. 1–8.

[41] M. Daole, A. Schiavo, J.L. Corcuera Bárcena, P. Ducange, F. Marcelloni, A. Renda, OpenFL-XAI: Federated learning of explainable artificial intelligence models in Python, SoftwareX 23 (2023) 101505, http://dx.doi.org/10.1016/j.softx.2023.101505.

[42] G.A. Reina, A. Gruzdev, P. Foley, O. Perepelkina, M. Sharma, I. Davidyuk, I. Trushkin, M. Radionov, A. Mokrov, D. Agapov, et al., OpenFL: An open-source framework for Federated Learning, 2021, arXiv preprint arXiv:2105.06413.

[43] J. Cózar, L. de la Ossa, J.A. Gámez, TSK-0 fuzzy rule-based systems for high-dimensional problems using the apriori principle for rule generation, in: C. Cornelis, M. Kryszkiewicz, D. Ślezak, E.M. Ruiz, R. Bello, L. Shang (Eds.), Rough Sets and Current Trends in Computing, Springer International Publishing, Cham, 2014, pp. 270–279.

[44] J.R. Quinlan, Learning with continuous classes, in: 5th Australian Joint Conference on Artificial Intelligence, Vol. 92, World Scientific, 1992, pp. 343–348.

[45] L. Breiman, J. Friedman, R. Olshen, C. Stone, Cart, in: Classification and Regression Trees, Wadsworth and Brooks/Cole, Monterey, CA, USA, 1984.

[46] J. Alcalá-Fdez, A. Fernández, J. Luengo, J. Derrac, S. García, L. Sánchez, F. Herrera, Keel data-mining software tool: data set repository, integration of algorithms and experimental analysis framework, J. Mult.-Valued Log. Soft 17 (2011).

[47] E. Bakopoulou, B. Tillman, A. Markopoulou, FedPacket: A federated learning approach to mobile packet classification, IEEE Trans. Mob. Comput. 21 (10) (2022) 3609–3628, http://dx.doi.org/10.1109/TMC.2021.3058627.

[48] J.C. Bárcena, P. Ducange, F. Marcelloni, A. Renda, F. Ruffini, A. Schiavo, Federated TSK models for predicting quality of experience in B5G/6G networks, in: 2023 IEEE International Conference on Fuzzy Systems, FUZZ-IEEE, IEEE, 2023, pp. 1–8.

[49] R. Alcalá, P. Ducange, F. Herrera, B. Lazzerini, F. Marcelloni, A multiobjective evolutionary approach to concurrently learn rule and data bases of linguistic fuzzy-rule-based systems, IEEE Trans. Fuzzy Syst. 17 (5) (2009) 1106–1122.

[50] M. Antonelli, P. Ducange, F. Marcelloni, Genetic training instance selection in multiobjective evolutionary fuzzy systems: A coevolutionary approach, IEEE Trans. Fuzzy Syst. 20 (2) (2011) 276–290.

[51] F. Wilcoxon, Individual comparisons by ranking methods, in: Breakthroughs in Statistics, Springer, 1992, pp. 196–202.

[52] J.L. Corcuera Bárcena, P. Ducange, F. Marcelloni, G. Nardini, A. Noferi, A. Renda, G. Stea, A. Virdis, Towards trustworthy AI for QoE prediction in B5G/6G networks, in: First International Workshop on Artificial Intelligence in beyond 5G and 6G Wireless Networks, AI6G 2022, 2022.

[53] J.C. Bárcena, P. Ducange, F. Marcelloni, A. Renda, F. Ruffini, Hoeffding regression trees for forecasting quality of experience in B5G/6G networks, in: First Workshop on Online Learning from Uncertain Data Streams, OLUD 2022, 2022, http://dx.doi.org/10.5281/zenodo.7024541, URL https://ceur-ws.org/Vol-3380/paper3.pdf.