Mining Trustworthy Symbolic Regression Models in Federated Settings

Mattia Billa*, Veronica Guidetti*, Luca La Rocca†, Federica Mandreoli*
*Department of Physics, Informatics and Mathematics, University of Modena and Reggio Emilia, Italy
†Department of Education and Humanities, University of Modena and Reggio Emilia, Italy
Email: {mattia.billa, veronica.guidetti, luca.larocca, federica.mandreoli}@unimore.it

***Abstract*—Symbolic Regression (SR) is an interpretable machine learning technique that discovers closed-form expressions from data, increasingly adopted in scientific and industrial applications. However, its use in privacy-sensitive domains has remained limited, as conventional SR methods require centralized data access for parameter estimation and model selection.**

We propose Bayesian Federated Symbolic Regression (BFSR), the first framework enabling SR in horizontal federated learning scenarios with formal Bayesian grounding. BFSR formulates federated SR as a sequential process that jointly performs marginal likelihood-based model selection and distributed Bayesian parameter inference, allowing clients to refine local posteriors without sharing data. We instantiate this framework via a two-stage strategy built upon genetic programming: during the evolutionary algorithm, we estimate model quality via the global Bayesian information criterion computed through Gaussian posterior fusion under a large-sample approximation; in the second stage, we apply full sequential Bayesian inference to a subset of candidate models for principled uncertainty quantification and refined marginal likelihood estimation.

Empirical evaluations on six datasets under varying levels of data heterogeneity and different numbers of clients show that BFSR consistently outperforms existing federated baselines in predictive accuracy and model interpretability, also enabling uncertainty quantification. These results establish BFSR as a scalable and trustworthy solution for federated modeling, well-suited to high-stakes domains requiring transparency and reliable epistemic uncertainty estimates.

***Index Terms*—Explainable AI, Federated learning, Bayes methods, Genetic algorithms**

I. Introduction

Symbolic Regression (SR) is a machine learning technique aiming to find the closed-form expression that best fits a given dataset [1]–[5]. During the last years, SR showed remarkable results in non-linear data-driven modeling, even on small training datasets [6], and was shown to generalize better than more powerful but black-box artificial intelligence methods in hundreds of tasks, providing interpretable results in most cases. For this reason, it was recently applied to high-stakes domains such as healthcare, to mine data-driven models for predicting binary, real-valued, and time-dependent events (see, e.g., [5], [7]–[11]), underlying its capacity to enhance different classes of statistical models.

Despite the growing interest in SR, most applications were explored in centralized data setups. In contrast, high-stakes domains often involve decentralized, privacy-sensitive data, where local models need to be validated on external datasets without compromising data confidentiality. This challenge can be overcome by Federated Learning (FL) [12] which enables the collaborative training of a global model without sharing local data. FL techniques, typically developed for and applied to deep learning models, can be classified as horizontal or vertical depending on whether the clients share the same features or samples, respectively [13].

The application of FL techniques to SR received some attention only during the last years [14], [15]. The first and, to date, only horizontal FL approach [14] uses a non-parametric clustering method (Mean Shift) to aggregate model fitness across clients. While innovative, this strategy precludes global parameter estimation and thus hinders parameter calibration within symbolic models. Consequently, model selection is restricted to randomly sampled formulas with fixed coefficients, leading to inefficient exploration of the functional space. Given that parameter tuning significantly enhances convergence and predictive performance in SR [16], and that distributed parameter inference is central to FL, this omission represents a fundamental limitation.

Beyond the need for decentralized parameter estimation in symbolic models to enhance accuracy, we argue that achieving generalization and trustworthiness hinges on robust uncertainty quantification techniques. Symbolic methods are often applied to tabular datasets, such as electronic health records, which are typically small to medium in size and exhibit high heterogeneity in both dimensionality and value distributions across sources. Uncertainty quantification is thus crucial to evaluate how effectively the global model adapts to local data distributions and to determine whether the aggregated data can be meaningfully captured by a single, unified model, i.e., whether it plausibly originates from a common underlying data-generating process.

In deep learning, addressing distributional heterogeneity in federated settings has received considerable attention in recent years. Since the introduction of FedAvg [17], the most widely adopted FL algorithm for neural networks, numerous approaches have been proposed to improve convergence and model robustness under non-IID conditions [18]. These include regularization-based strategies [19] as well as Bayesian FL (BFL) frameworks, which, in addition to improving performance in heterogeneous and data-scarce settings, provide a principled means of quantifying model uncertainty [20]–[22].

However, applying off-the-shelf FL solutions to SR does not suffice for obtaining accurate decentralized point or

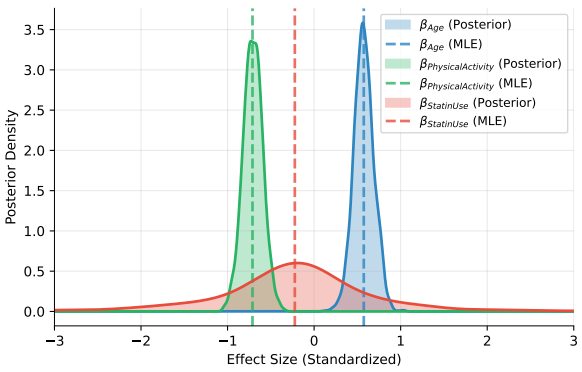


Fig. 1. Illustrative example: posterior parameter distribution (shaded areas) and maximum likelihood estimates (dashed lines).

distributional parameter estimates, due to several fundamental limitations. First, decentralized point parameter estimation techniques developed for deep learning models are not naturally suited to SR due to structural differences. While the architecture and number of trainable parameters in neural networks are fixed a priori, the most widely used method for exploring the model space in SR leverages genetic programming, involving the continuous generation and evolution of model structures. The dynamic nature of SR significantly alters the number and identity of parameters over time, and may also lead to uncontrolled formula bloat [23]. To address this, conventional FL methods must be augmented with statistical model selection criteria that can guide the search toward parsimonious and interpretable expressions. Finally, while standard BFL techniques may, in principle, provide uncertainty estimates for individual distributed SR models, their application to genetic programming remains computationally prohibitive. BFL approaches are typically tailored for static NN architectures and often rely on techniques such as knowledge distillation [21] or iterative estimation of local posterior distributions [22]. Such methods are computationally intensive and thus impractical in evolutionary computation scenarios where model structures and parameters evolve continuously.

Motivating Example. To illustrate the importance of Bayesian methods for uncertainty quantification in federated symbolic machine learning, consider a clinical example involving the prediction of visceral adipose tissue (VAT) percentage across heterogeneous patient populations, for instance, urban tertiary hospitals versus rural clinics, using a linear model of the form: $VAT(\%) \sim \beta_{\text{Age}} \cdot \text{Age} + \beta_{\text{PhysicalActivity}} \cdot \text{PhysicalActivity} + \beta_{\text{Statin}} \cdot \text{Statin} + \ldots$. While covariates such as age and physical activity may exhibit stable effects across sites, others, like medication use, can be context-dependent. For example, statins may correlate with lower VAT in urban settings, where they often reflect proactive cardiovascular care, but with higher VAT in rural clinics, where they may be prescribed reactively in patients with existing metabolic dysfunction.

Standard FL approaches based on point estimates, such as maximum likelihood estimation (MLE), would average these divergent effects, potentially producing a misleading modest protective association biased toward the dominant data source. In contrast, Bayesian methods yield full posterior distributions over model parameters, capturing the elevated uncertainty around the statin coefficient (Figure 1) compared to the other parameters. This posterior uncertainty reflects underlying population heterogeneity and highlights the unreliability of generalizing this effect across contexts, enabling more cautious and context-aware inference.

Our contribution. To address the need for well-calibrated decentralized symbolic models with principled uncertainty quantification, we propose Bayesian Federated Symbolic Regression (BFSR), a novel framework for horizontal federated SR with formal Bayesian foundations. Specifically, we:

- Formulate BFSR as a sequential Bayesian process combining marginal likelihood-based model selection with iterative refinement of parameter posteriors across decentralized clients.
- Introduce a lightweight implementation integrating BFSR into genetic programming-based SR via a two-stage inference: (1) model selection during evolution using a Gaussian posterior fusion approximation of the Bayesian Information Criterion (BIC), and (2) full Bayesian inference on a selected subset of models to quantify uncertainty and finalize model choice based on marginal likelihood.
- Demonstrate BFSR's superior performance, calibration, and interpretability compared to the state-of-the-art on four SRBench datasets, a synthetic benchmark, and a real-world clinical dataset with varying heterogeneity and client counts.

The paper is structured as follows: Section II overviews GPSR and Bayesian Inference. Section III details the proposed approach. Section IV describes the GPSR configuration, the datasets used in the experiments, and the methods for controlling data distribution and heterogeneity across clients. The experimental results are presented and discussed in Section V. Finally, Section VI discusses the related work and summarizes the relevant differences and relation with our approach, with conclusions drawn in Section VII.

II. Preliminaries

To motivate our approach to embedding Bayesian learning in federated SR, we begin by briefly summarizing the foundations of evolutionary SR and Bayesian inference.

A. Genetic Programming-based SR (GPSR)

The original and most widely used method for SR, namely GPSR, leverages a genetic algorithm to perform statistical model selection and optimization.

Like any genetic algorithm, GPSR relies on three core components: (i) an encoding and sampling scheme to represent and generate candidate solutions, (ii) an exploration strategy to define and traverse solution neighborhoods, and (iii) a ranking

Algorithm 1 GPSR pseudo-code

1: **Input:** P, G, T, $\mathcal{D}$
2: $\mathcal{P} \leftarrow GenerateIndividuals(P)$
3: $EvaluatePopulation(\mathcal{P}, \mathcal{D})$
4: **for** generation $g = 1 \dots G$ **do**
5: $\quad \mathcal{P} \leftarrow_{+} GenOffspring(\mathcal{P}, T)$
6: $\quad \mathcal{P} \leftarrow Simplify(\mathcal{P})$
7: $\quad EvaluatePopulation(\mathcal{P}, \mathcal{D})$
8: $\quad$ Drop and replace invalids/duplicates
9: $\quad \mathcal{P} \leftarrow Select(\mathcal{P}, P)$
10: **Output:** $\mathcal{P}$

mechanism to select individuals for reproduction and survival. In its basic form, GPSR encodes closed-form expressions as binary trees, with internal nodes as operators and leaves as constants or features, selected recursively. Tree depth is regulated by two hyperparameters: *parsimony* (initial operator probability at the root) and *parsimony decay* (reducing this probability with depth). Neighborhood exploration uses genetic operations like point mutation (replacing a subtree with a random one) and crossover (swapping subtrees between two formulas). Solutions are ranked via a scalar fitness function, typically a regression metric such as the mean squared error, evaluated on the dataset of interest $\mathcal{D}$. Reproduction eligibility is determined through tournament selection: T candidates are randomly sampled and evaluated, and the best is selected to reproduce. Larger T values make selection more deterministic.

The GPSR algorithm we implemented in this work is outlined in Algorithm 1. It begins by initializing a population $\mathcal{P}$ of P randomly generated formulas (step 2), followed by the evaluation of their individual fitness values (step 3). The population then evolves over G generations: offspring are generated via genetic operations (step 5) and added to the existing population. Each formula in $\mathcal{P}$ is then simplified using symbolic calculus (step 6) and re-evaluated for fitness (step 7). To avoid redundancy, symbolic equivalence and identical fitness scores are used to identify and remove duplicate formulas (step 8), which are replaced with new random individuals. Symbolic calculus also plays a key role in mitigating convergence issues arising from functional redundancies [24]–[26]. Finally, selection is performed based on fitness values (step 9). The algorithm outputs a ranked list of formulas along with their corresponding fitness values (step 10).

Originally devised for centralized data, this formulation can be systematically extended to federated environments through appropriate modifications to the performance evaluation procedure. Specifically, the *EvaluatePopulation* steps must be redefined to incorporate a federated evaluation protocol for each symbolic model in $\mathcal{P}$.

B. Bayesian Inference (BI)

Statistical model selection aims to identify models that best explain observed data by balancing between predictive accuracy and model complexity. In the Bayesian framework, this is achieved by assessing the probability that a given model generated the data [27].

Consider a regression problem in which the goal is to predict the values of an unknown function $f : \mathbb{R}^d \to \mathbb{R}$, parameterized by a vector $\theta \in \mathbb{R}^{p_f}$, given a dataset $\mathcal{D} = \{(x_j, y_j)\}_{j=1}^n$, where $x_j \in \mathbb{R}^d$ denote the input features and $y_j \in \mathbb{R}$ the corresponding target values. Assuming additive Gaussian noise with fixed variance σ^2, we model each y_j as being drawn from a normal distribution centered at the model prediction: $y_j \sim \mathcal{N}(f(x_j, \theta), \sigma^2)$. Assuming that the y_j are conditionally independent given θ, the likelihood under model f factorizes and is given by:

$$p(\mathcal{D}|\theta, f) = \prod_{j=1}^{n} \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left\{-\frac{(y_j - f(x_j, \theta))^2}{2\sigma^2}\right\}.$$

To simplify notation, we omit σ from the likelihood term $p(\mathcal{D}|\theta, f)$, though it remains an explicit parameter of the model and is inferred alongside θ.

Then, the *marginal likelihood* or model evidence, which quantifies how well model f explains the data, is computed via Bayes' theorem as:

$$p(\mathcal{D}|f) = \int p(\theta|f)\, p(\mathcal{D}|\theta, f)\, d\theta,$$

where $p(\theta|f)$ is the parameter prior under model f. Finally, if $p(f)$ is the prior probability of model f, its posterior probability will be given by $p(f|\mathcal{D}) \propto p(\mathcal{D}|f)p(f)$.

Bayesian inference offers two principal advantages: (i) a principled framework for uncertainty quantification via the posterior distribution over parameters,

$$p(\theta|\mathcal{D}, f) \propto p(\theta|f)\, p(\mathcal{D}|\theta, f),$$

and (ii) the capacity to incrementally incorporate new information. Given an additional independent dataset $\mathcal{D}'$, the posterior distribution and marginal likelihood can be updated recursively as:

$$p(\theta|\mathcal{D} \cup \mathcal{D}', f) \propto p(\theta|\mathcal{D}, f)\, p(\mathcal{D}'|\theta, f),$$

$$p(\mathcal{D} \cup \mathcal{D}'|f) = p(\mathcal{D}'|\mathcal{D}, f)\, p(\mathcal{D}|f),$$

where $p(\mathcal{D}'|\mathcal{D}, f) = \int p(\theta|\mathcal{D}, f)\, p(\mathcal{D}'|\theta, f)\, d\theta$ is the predictive density.

This sequential structure aligns naturally with the goals of FL, which seeks to match centralized performance while operating over decentralized data with privacy constraints. Notably, the result does not depend on the update order.

III. Bayesian Federated Symbolic Regression

Let consider a federated regression problem involving K clients, indexed by $\mathcal{K} = \{1, 2, \dots, K\}$ where each client $i \in \mathcal{K}$ holds a local dataset $\mathcal{D}_i = \{(x_j^{(i)}, y_j^{(i)})\}_{j=1}^{n_i}$, with $x_j^{(i)} \in \mathbb{R}^d$ and $y_j^{(i)} \in \mathbb{R}$.

To address this problem, we propose BFSR, a novel regression analysis that searches the space $\mathcal{F}$ of real-valued, closed-form expressions $f : \mathbb{R}^d \to \mathbb{R}$, each parameterized by $\theta \in \mathbb{R}^{p_f}$, to find the symbolic expression $f^* \in \mathcal{F}$

that best explains the collective data while quantifying model uncertainty. Assuming for simplicity equal prior probabilities for all models, this objective is formalized as:

$$f^* = \arg\max_{f \in \mathcal{F}} p(\mathcal{D}_{1:K}|f),$$

along with estimation of the parameter posterior $p(\theta|\mathcal{D}_{1:K}, f^*)$.

Given the aforementioned properties of GPSR and BI, f^* can be selected by an enhanced version of GPSR that treats model parameters θ as random variables and estimates marginal likelihoods accordingly. Specifically, each candidate model f in the evolving population must be evaluated as follows (step 7, Algorithm 1): the server first initializes a prior distribution $p(\theta|f)$. Then, marginal likelihoods $p(\mathcal{D}_{1:i}|f)$ and posteriors $p(\theta|\mathcal{D}_{1:i}, f)$ are computed sequentially across clients $i = 1, \ldots, K$, using only their local data $\mathcal{D}_i$. Once all clients have contributed, each model f is associated with a global posterior and marginal likelihood, facilitating principled model selection.

Nonetheless, performing full Bayesian inference for every model across all GPSR iterations remains computationally prohibitive [28]. To mitigate this, we propose adopting a two-step strategy depicted in Figure 2: during GPSR's evolutionary phase, we employ the BIC, a crude approximation of the marginal likelihood, to guide model selection; then, at the end of GPSR's evolution, we identify a set of models with sufficient theoretical foundation and perform full Bayesian inference using Sequential Monte Carlo (SMC) sampling [29]. This final step allows us to compute accurate marginal likelihoods and quantify parameter uncertainty.

A. Step 1: Streamlined model selection during GPSR

To streamline model space exploration during GPSR's evolutionary phase, we approximate model quality using the BIC. Originally proposed by Schwarz [30], the BIC provides a large-sample approximation to the model marginal likelihood [31]. While its computation is straightforward in centralized regression problems, extending BIC estimation to federated settings requires a principled approach to recover the global maximum a posteriori estimate, which coincides with the global MLE under a flat prior.

In federated scenarios where data are partitioned across disjoint subsets and the parameter prior is uniform, the global posterior distribution can be approximated by aggregating local Gaussian posteriors. For each subset $\mathcal{D}_i$, a Laplace approximation yields a local posterior $p(\theta|\mathcal{D}_i, f) \approx \mathcal{N}(\mu_i, \Sigma_i)$ where μ_i and Σ_i are the estimated local mean vector and covariance matrix respectively. Assuming conditional independence between subsets, the global posterior $p(\theta|\mathcal{D}_{1:K}, f)$ is proportional to the product of these local approximations. This fusion results in a Gaussian posterior with precision matrix $\Lambda = \sum_i \Sigma_i^{-1}$ and mean vector $\mu = \Lambda^{-1} \sum_i \Sigma_i^{-1} \mu_i$.

In the large-sample limit, each local posterior concentrates around its corresponding MLE, $\hat{\theta}_f^{(i)}$, and the covariance scales as $\Sigma_i \approx \frac{1}{n_i} \mathcal{I}_i^{-1}$, where n_i denotes the number of observations in $\mathcal{D}_i$, and $\mathcal{I}_i$ is the observed Fisher information. If the Fisher information is approximately constant across subsets, either due to homogeneous input distributions or because the data are drawn from a common true model with sufficiently informative covariates, the global mean reduces to a weighted average of local estimates:

$$\mu \approx \frac{1}{\sum_i n_i} \sum_i n_i \hat{\theta}_f^{(i)} \equiv \hat{\theta}_f.$$

This coincides with the well-known FedAvg aggregation rule [17], which receives a theoretical foundation under Bayesian learning in the large-sample regime [22]. However, unlike deep learning models where FedAvg typically requires multiple rounds of client-server interaction for convergence, SR models are interpretable and have few parameters. As a result, a single round of communication suffices for parameter aggregation and model evaluation.

Algorithm 2 EVALUATEPOPULATIONFEDERATED

1: **Input:** Population $\mathcal{P}$, local datasets $\{\mathcal{D}_i\}_{i \in \mathcal{K}}$
2: **for** each model $f \in \mathcal{P}$ **do**
3: **for** each client $i \in \mathcal{K}$ **in parallel do**
4: $\hat{\theta}_f^{(i)} \leftarrow \arg\max_\theta p(\mathcal{D}_i|\theta, f)$ *local MLE*
5: Send $\hat{\theta}_f^{(i)}$ and n_i to server
6: $\hat{\theta}_f \leftarrow \frac{1}{\sum_i n_i} \sum_i n_i \hat{\theta}_f^{(i)}$ *server global estimate*
7: Server sends $\hat{\theta}_f$ to all clients
8: **for** each client $i \in \mathcal{K}$ **in parallel do**
9: $\mathcal{L}_f^{(i)} \leftarrow p(\mathcal{D}_i|\hat{\theta}_f, f)$ *local likelihood*
10: Send $\log \mathcal{L}_f^{(i)}$ to server
11: Server computes global BIC(f)
12: **Output:** BIC(f) for all $f \in \mathcal{P}$

To enable Bayesian federated model evaluation within GPSR using BIC, we replace the centralized evaluation step in Algorithm 1, *EvaluatePopulation*, with a federated procedure described in Algorithm 2, and illustrated on the left-hand side of Figure 2.

In this procedure, clients receive the candidate population $\mathcal{P}$ from the server, compute their local MLEs, $\hat{\theta}_f^{(i)}$, via numerical optimization and return them to the server (steps 4 and 5). The server aggregates these to obtain a global parameter estimate $\hat{\theta}$, which is then broadcast back to the clients (steps 6 and 7). Each client evaluates its local likelihood $p(\mathcal{D}_i|\hat{\theta}_f, f)$ under $\hat{\theta}$ and sends the result back to the server (steps 9 and 10).

This enables the server to estimate a global likelihood and compute BIC score (step 11) as:

$$\text{BIC}(f) = p_f \log(n_0) - 2 \log \left[\prod_{i \in \mathcal{K}} p(\mathcal{D}_i|\hat{\theta}_f, f) \right],$$

where p_f is the number of free parameters in model f, and $n_0 = \sum_{i \in \mathcal{K}} n_i$ is the total sample size. After training, the algorithm returns a population of symbolic expressions, each associated with a BIC score.

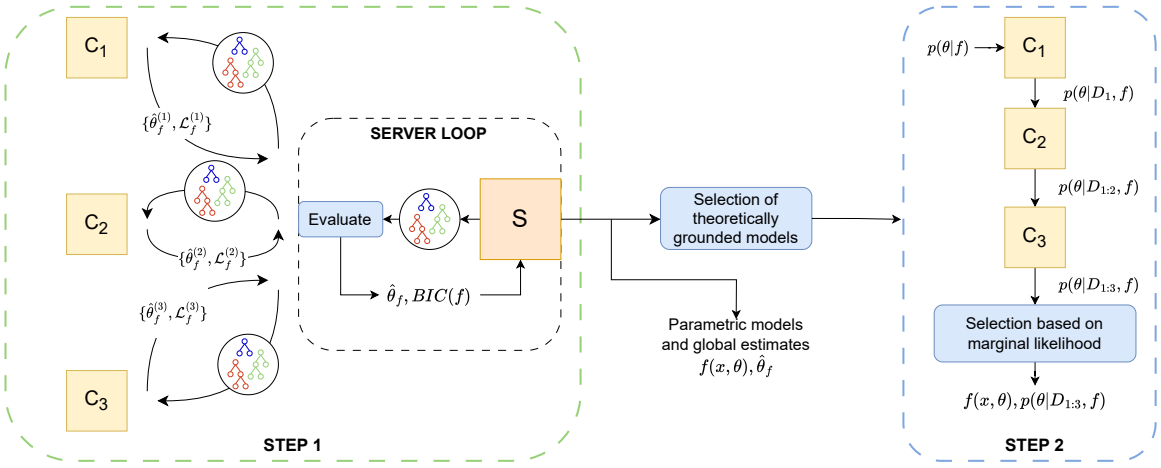


Fig. 2. Visualization of BFSR with a three-client setup. C_1, C_2, and C_3 indicate the clients, and S indicates the server.

B. Step 2: Refined model selection via full Bayesian inference

While ranking models by their BIC values allows for the identification of a single best-scoring model, this approach has important limitations. First, the BIC is an asymptotic approximation to the marginal likelihood and may be insufficiently accurate to correctly discriminate among models, particularly in finite-sample regimes. Second, selecting only the model $f_0 \in \mathcal{P}$ with the minimum BIC does not offer strong theoretical guarantees of its superiority over other candidates.

To address these limitations, we adopt a principled Bayesian model comparison framework to identify the subset of models in $\mathcal{P}$ that are plausibly supported by the observed data. Specifically, we leverage the asymptotic relationship between BIC differences and the Bayes factor:

$$\frac{p(\mathcal{D}_{1:K}|f)}{p(\mathcal{D}_{1:K}|f_0)} \approx \exp\left(-\frac{1}{2}\Delta\text{BIC}\right),$$

where $\Delta\text{BIC} = \text{BIC}(f) - \text{BIC}(f_0)$. According to the conventional interpretation of Bayes factors [32], values of $\Delta\text{BIC} > 6$ constitute strong evidence against model f in favor of f_0. Based on this threshold, we define the set of supported models $\mathcal{S} \subset \mathcal{P}$ as those satisfying $\Delta\text{BIC} \leq 6$, i.e., models that are not strongly disfavored relative to the best candidate f_0.

Furthermore, to mitigate the impact of approximating the marginal likelihood via BIC and assess the uncertainty of the models in $\mathcal{S}$ on the distributed datasets, we perform full sequential Bayesian inference across all clients. The overall procedure is illustrated in the right-hand panel of Figure 2.

For each model $f \in \mathcal{S}$, we initialize the prior over the parameters $p(\theta|f)$ as a multivariate normal distribution $\mathcal{N}(\hat{\theta}_f, \Sigma)$ centered at the point estimates $\hat{\theta}_f$, with diagonal covariance given by $\Sigma_{jj} = \max\{\text{range}_{i\in\mathcal{K}}(\hat{\theta}_j^{(i)}), 1\}$. The prior over the standard deviation of the likelihood, σ, is specified as an inverse gamma distribution with shape parameter $\alpha = 2$ and scale parameter $\beta = 1$.

We perform sequential inference using the SMC sampler provided in the PyMC library [33]. At each step, SMC estimates the incremental marginal likelihood $\log(p(\mathcal{D}_i|\mathcal{D}_{1:i-1}, f))$ by accumulating the normalized particle weights [34]. These values are then summed across all steps to approximate the overall marginal log-likelihood $\log(p(\mathcal{D}_{1:K}|f))$, which then serves as the criterion for model selection.

To address the computational challenges of high-dimensional parameter distributions in sequential inference, we approximate intermediate posteriors $p(\theta|\mathcal{D}_{1:i}, f)$ as multivariate normals. Using PyMC's *prior_from_idata*, we estimate the posterior mean μ and covariance matrix Σ from MCMC samples. The covariance is then factorized via Cholesky decomposition $\Sigma = LL^\top$, enabling efficient sampling through the affine transformation $\theta \sim \mu + Lz$, where $z \sim \mathcal{N}(0, I)$. This approach preserves the first two moments of the posterior and facilitates downstream inference steps.

To ease the comparison with the state-of-the-art and maintain interpretability, we avoid creating a marginal weighted ensemble. Instead, we simply select the model with the highest marginal likelihood across clients in $\mathcal{S}$.

IV. Experimental Setup

A. GPSR algorithm setup

Our implementation of GPSR follows the steps described in Section II-A. In particular, we set parsimony to 0.9 and parsimony decay to 0.95. These values are standard in the literature and were applied uniformly across methods, ensuring a fair comparison. Operators and features are selected with a uniform probability distribution. Constants are treated as an additional feature class and are initialized from a uniform distribution over $[-1, 1]$. The default set of operators used is $\mathcal{O} = \{+, -, *, /, \ln(.), e^{(.)}, **, \sin(.)\}$, while genetic operators encompass mutation, crossover, operator insertion, deletion, and replacement, and feature replacement. The ranking induced by the fitness function (the global BIC estimate) is used in both population selection and offspring creation through tournament selection (tournament size T is fixed to 3 to limit selection pressure). In every experiment, we evolve a

TABLE I
DATASET CHARACTERISTICS.

Dataset	d	B	$\mathbb{R}$	total	train	test
F5-noise	2	0	2	5000	600	1250
Pollen	4	0	4	3790	600	948
Houses	8	0	8	20330	1200	5083
Echo M.	9	3	6	17233	1800	4309
DEXA	9	4	5	3570	2400	893
Breast T.	13	9	4	114890	3000	28723

population of $P = 100$ individuals for $G = 100$ generations without early stopping. All experiments are repeated 10 times[1].

B. Simulating realistic federated settings

To simulate realistic federated settings, we incorporate both sample size and statistical heterogeneity in data distribution across clients. Sample size heterogeneity is simulated by either distributing an equal number of samples per client or following a power-law distribution, where the client with the most data has ten times more samples than the one with the least data. This can be achieved by ensuring $n\sum_{i=1}^{K}\gamma^{i-1} = N$, where N is the number of samples used for training, $\gamma^{K-1} = 0.1$ and each client $i = 1, \ldots, K$ holds $n_i = \gamma^{i-1}n$ data.

To generate non-IID client training data, we adapt the Dirichlet-based mechanism in [35] for regression. In a classification setting, for each class j, one would sample $\vec{q}_j \sim \text{Dir}_K(\alpha\vec{p})$, where p represents the prior class distribution, and allocate a proportion $q_{j,i}$ of instances from class j to client i. The parameter α controls statistical heterogeneity: $\alpha = 100$ mimics identical distributions across clients, while lower values increase heterogeneity by concentrating class examples within specific clients. For regression, this approach is adapted by sampling based on Q quantile classes of the target variable y. Specifically, each client draws a distribution $\vec{p}_i \sim \text{Dir}_Q(\alpha\vec{u})$, where $u_j \equiv 1$, and receives a proportion $p_{i,j}$ of instances from quantile class j. As α decreases, the client distributions deviate more from the IID case, amplifying heterogeneity.

C. Datasets

We evaluate regression methods using four datasets from SRBench [3], a synthetic dataset (f5-noise) from [14], and a private real-world dataset with Electronic Health Records (DEXA). Table I summarizes the key characteristics of these datasets: the number d and types (B for binary, $\mathbb{R}$ for real) of features, total records, and total number of records sampled for training (N in Section IV-B) and testing.

The test set comprises 25% of the total records, while the training set is sampled across the clients according to statistical heterogeneity and distributed based on sample heterogeneity. We consider two levels of sample heterogeneity, $\gamma^{K-1} \in \{1, 0.1\}$, and three levels of statistical heterogeneity, $\alpha \in \{100, 10, 1\}$, using deciles as quantiles ($Q = 10$). So, for each dataset, we simulate 6 different settings, for a total of 36 configurations. All methods are tested simulating the presence of $K = 3$, as in [14], and $K = 10$ clients[2].

V. RESULTS

The experiments are structured around three primary objectives. First, we conduct a comparative analysis of BFSR and the state-of-the-art method for horizontal FL in SR, Mean Shift [14], focusing on both accuracy and interpretability. Second, we investigate the uncertainty quantification capabilities of BFSR, demonstrating that it generates well-calibrated uncertainty estimates. Finally, we explore the scalability of BFSR by evaluating its performance in a scenario with an increased number of clients. All tests were run on a computing cluster with nodes featuring 2×24-core Intel Cascade Lake 8260 CPUs and 384 GB RAM. Mean Shift used 4 cores and 32 GB RAM; BFSR used 8 cores and 64 GB RAM. On average, Mean Shift completed in approximately 40 minutes per run, whereas BFSR required about 60 minutes (50 minutes for step 1 and 10 minutes for step 2).

A. Comparison with the State-of-the-Art

1) Predictive Accuracy: Our first aim is to compare the state-of-the-art method for horizontal FL in SR, i.e., Mean Shift by [14], with BFSR. We do this by computing the *coefficient of determination* (R^2) on the test set, a widely used metric in the SR community [3]. To obtain an R^2 estimate from the BFSR approach, we aggregate samples from the posterior predictive distribution to produce a point estimate. This requires, for each new data point j with feature vector x_j, a predicted value $\hat{y}_j$. We use the posterior predictive mean and approximate it via Monte Carlo averaging:

$$\hat{y}_j = \int f(x_j, \theta)p(\theta|\mathcal{D}_{1:K}, f)d\theta \approx \frac{1}{S}\sum_{s=1}^{S} f(x_j, \theta^{(s)}),$$

where $\theta^{(s)}$, $s = 1, \ldots, S$, is a posterior sample.

Table II presents the predictive performance (R^2, mean $\pm$SD) across all experimental settings in the three-client setup. A paired, two-tailed t-test with significance level $\alpha = 0.05$ is used to assess the statistical significance of differences between BFSR and Mean Shift. Bold values indicate the best performance, while underlined values denote statistically significant improvements.

The results show that BFSR outperforms Mean Shift in 31 out of 36 scenarios overall, with statistically significant improvements in 22 cases. Exceptions arise in scenarios with low R^2 values, such as the Breast T. dataset, where limited data informativeness leads to comparable performance between the methods. Another exception is the Echo M. dataset, where performance saturates, and both methods achieve results close to those of a centralized setup.

For reference, Table III reports the performance of BFSR in a centralized setting, where the data, sampled as described

[1]The source code is available at: https://github.com/mattiabilla/BFSR

[2]Due to computational constraints, when using ten clients, we evaluate only the most heterogeneous configuration.

TABLE II
PREDICTIVE PERFORMANCE (R^2, MEAN $\pm$ SD) AND MODEL COMPLEXITY (C, MEAN $\pm$ SD) ACROSS METHODS IN THE 3-CLIENT SETUP.

Configuration			Accuracy (R^2)		Complexity (C)	
γ^2	α	Dataset	Mean Shift	BFSR	Mean Shift	BFSR
1	100	F5-noise	0.31 ± 0.14	**0.77 ± 0.13**	**17.7 ± 9.2**	30.4 ± 25.0
		Pollen	0.59 ± 0.09	**0.78 ± 0.09**	23.6 ± 6.7	**16.3 ± 7.5**
		Houses	0.51 ± 0.04	**0.55 ± 0.04**	23.9 ± 8.8	**19.3 ± 7.0**
		Echo M.	**0.42 ± 0.01**	0.42 ± 0.02	23.0 ± 11.1	**12.8 ± 3.4**
		DEXA	0.49 ± 0.03	**0.57 ± 0.03**	**22.9 ± 6.7**	26.6 ± 5.9
		Breast T.	**0.03 ± 0.01**	0.02 ± 0.03	**34.3 ± 22.5**	39.5 ± 28.7
	10	F5-noise	0.27 ± 0.15	**0.82 ± 0.08**	**10.8 ± 3.6**	25.5 ± 14.9
		Pollen	0.58 ± 0.12	**0.78 ± 0.12**	27.0 ± 12.4	**14.4 ± 4.5**
		Houses	0.50 ± 0.03	**0.53 ± 0.06**	25.5 ± 6.2	**18.9 ± 5.7**
		Echo M.	0.41 ± 0.02	**0.42 ± 0.02**	22.8 ± 14.9	**10.4 ± 4.6**
		DEXA	0.47 ± 0.04	**0.59 ± 0.02**	**19.0 ± 8.0**	25.3 ± 4.0
		Breast T.	0.02 ± 0.01	**0.04 ± 0.02**	56.2 ± 53.0	**25.2 ± 15.2**
	1	F5-noise	0.18 ± 0.18	**0.76 ± 0.14**	**11.3 ± 5.0**	45.4 ± 23.1
		Pollen	0.65 ± 0.10	**0.78± 0.10**	24.6 ± 8.6	**13.3 ± 3.2**
		Houses	0.47 ± 0.10	**0.53 ± 0.04**	19.5 ± 6.7	**17.9 ± 8.4**
		Echo M.	0.38 ± 0.05	**0.40 ± 0.03**	16.9 ± 7.3	**11.1 ± 6.3**
		DEXA	0.37 ± 0.15	**0.55 ± 0.04**	28.3 ± 14.0	**22.8 ± 5.2**
		Breast T.	**0.00 ± 0.04**	-0.02 ± 0.04	59.2 ± 52.3	**29.1 ± 16.4**
0.1	100	F5-noise	0.24 ± 0.14	**0.77 ± 0.11**	**11.6 ± 4.8**	34.0± 21.2
		Pollen	0.60 ± 0.12	**0.78 ± 0.01**	23.7 ± 8.4	**17.9± 4.4**
		Houses	0.52 ± 0.04	**0.55 ± 0.04**	28.0 ± 13.5	**20.2± 4.9**
		Echo M.	0.42 ± 0.02	**0.43 ± 0.02**	21.7 ± 7.9	**17.2± 5.5**
		DEXA	0.45 ± 0.06	**0.59 ± 0.03**	**21.2 ± 5.6**	25.5± 3.5
		Breast T.	0.01 ± 0.02	**0.03 ± 0.02**	43.7 ± 26.9	**41.4± 23.0**
	10	F5-noise	0.22 ± 0.17	**0.78± 0.09**	**13.1 ± 5.7**	44.2 ± 15.6
		Pollen	0.54 ± 0.10	**0.78 ± 0.01**	24.4 ± 11.1	**16.6 ± 3.1**
		Houses	0.48 ± 0.03	**0.56 ± 0.03**	23.7 ± 11.1	**22.9± 3.4**
		Echo M.	0.41 ± 0.02	**0.43 ± 0.01**	26.2 ± 12.7	**13.0 ± 2.7**
		DEXA	0.45 ± 0.05	**0.58 ± 0.03**	**22.5 ± 6.1**	26.0 ± 5.3
		Breast T.	0.02 ± 0.01	**0.02 ± 0.02**	42.3 ± 15.1	**36.2 ± 15.5**
	1	F5-noise	0.00 ± 0.31	**0.79 ± 0.09**	**6.6 ± 5.6**	27.0 ± 20.1
		Pollen	0.55 ± 0.13	**0.78 ± 0.01**	28.7 ± 6.8	**15.8 ± 3.0**
		Houses	0.47 ± 0.04	**0.54 ± 0.03**	24.5 ± 6.8	**17.7 ± 5.1**
		Echo M.	**0.39 ± 0.03**	0.39 ± 0.05	20.6 ± 10.1	**14.0 ± 5.6**
		DEXA	0.44 ± 0.05	**0.55 ± 0.04**	**17.0 ± 5.9**	26.6 ± 5.4
		Breast T.	**-0.03 ± 0.03**	-0.05 ± 0.08	47.1 ± 34.2	**38.9 ± 19.4**

in Section IV-B, is aggregated before training. Interestingly, the accuracy obtained in this configuration is comparable to that achieved under distributed scenarios. However, it is worth noting that the Breast T. dataset remains difficult to fit, even when trained centrally.

2) Model Interpretability: In Symbolic Regression, the interpretability of the resulting formulas is a key concern. To quantify this aspect, we use the concept of *complexity*, defined as the total number of mathematical operators, features, and free parameters within a model. This metric provides a practical proxy for interpretability, as more complex expressions tend to be harder to understand and analyze. For further details on this measure, we refer the reader to [3].

Table II summarizes the model complexity (C, mean $\pm$ SD) of the tested methods in the three-client setup. A paired, two-tailed t-test with significance level $\alpha = 0.05$ is used to assess differences between BFSR and Mean Shift. As before, bold values indicate the best performance, and underlined values denote statistically significant differences. The results show that BFSR produces less verbose models for most of the datasets, even though this difference is not statistically significant. Conversely, Mean Shift generates significantly shorter formulas for the synthetic dataset (F5-noise); however, these shorter formulas come at the cost of very low accuracy.

Since BFSR achieves higher predictive accuracy without compromising interpretability, it represents a more effective alternative to Mean Shift.

B. Evaluating Uncertainty Quantification

We assess BFSR uncertainty quantification and calibration using the Within 50 (W_{50}) and Within 95 (W_{95}) scores, also referred to as Prediction Interval Coverage Probability

TABLE III
PREDICTIVE PERFORMANCE (R^2, MEAN $\pm$ SD) AND MODEL COMPLEXITY (C, MEAN $\pm$ SD) FOR THE CENTRALIZED SETUP.

Dataset	**Accuracy** (R^2)	**Complexity** (C)
F5-noise	0.88 $\pm$ 0.04	57.5 $\pm$ 49.8
Pollen	0.78 $\pm$ 0.01	13.4 $\pm$ 2.3
Houses	0.63 $\pm$ 0.02	31.8 $\pm$ 8.0
Echo M.	0.41 $\pm$ 0.02	13.9 $\pm$ 6.6
DEXA	0.58 $\pm$ 0.04	25.6 $\pm$ 2.7
Breast T.	0.01 $\pm$ 0.03	39.4 $\pm$ 33.4

(PICP) [36], which measure the proportion of test values that fall within the 50% and 95% posterior predictive intervals, respectively. Let $\text{PI}_\alpha(x_j)$ denote the $(1-\alpha)\%$ central posterior predictive interval for input x_j, estimated from the empirical quantiles of the predictive samples:

$$\text{PI}_\alpha(x_j) = \left[q_{\alpha/2}(x_j),\ q_{1-\alpha/2}(x_j)\right],$$

where $q_p(x_j)$ is the p-th quantile of $\{f(x_j, \theta^{(s)})\}_{s=1}^{S}$. Then, the Within α score is given by:

$$\text{W}_\alpha := \frac{1}{N_{\text{test}}} \sum_{j=1}^{N_{\text{test}}} \mathbb{1}_{\{y_j \in \text{PI}_\alpha(x_j)\}}.$$

Values of W_{50} and W_{95} closer to 0.50 and 0.95, respectively, indicate better calibration performance. Specifically, values below these thresholds suggest that the model is overconfident, while values above suggest underconfidence.

In Figure 3, we represent these metrics for each setting. BFSR exhibits good calibration for almost all scenarios and datasets, except for F5-noise where the high variance of W_{50} is caused by the synthetic nature of this dataset and its relatively low noise level, which can lead to significant miscalibration when the true model is not found. This also justifies the very low level of confidence shown for W_{95}. However, even in this case, there is no clear pattern of miscalibration as heterogeneity increases.

These results suggest that BFSR is capable of identifying well-defined models and that the approximation of the posterior distributions using multivariate normal distributions does not impose a significant limitation on the model's accuracy or expressiveness.

C. Scalability to More Clients

So far, our evaluation focuses on a three-client federated setup, consistent with prior work [14]. To examine whether the observed performance holds with a larger number of clients, we extend our experiments to a more challenging ten-client federated setting. Due to computational and space constraints, here we consider only the most heterogeneous configuration.

Results (Table IV) show that BFSR maintains performance similar to that observed in the three-client setup. In some cases, we even see slight improvements, though not statistically significant, which may be attributed to implicit regularization effects and enhanced ensemble behavior when scaling to more clients. This hypothesis is also supported by the reduced complexity of the selected model compared to the three-client setup. Importantly, BFSR continues to deliver well-calibrated uncertainty estimates, underscoring its robustness and scalability in challenging federated environments.

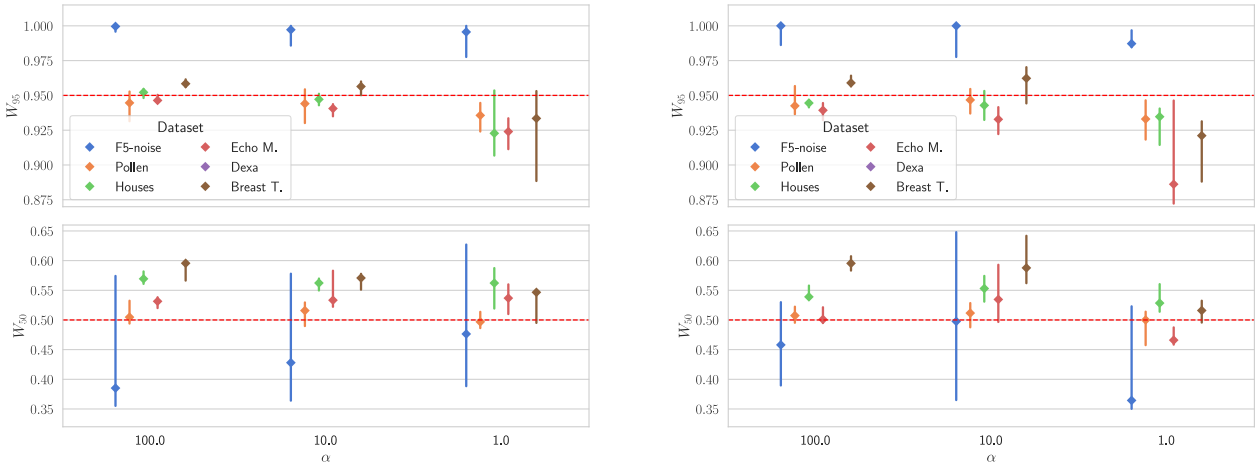
VI. RELATED WORKS AND DISCUSSION

FL has seen rapid advances in handling non-IID data, with methods like FedAvg [17] often struggling to converge to optimal global models [37], [38]. BFL has emerged as a promising direction for incorporating uncertainty and improving robustness also in data limited settings [20]–[22]. However, BFL methods are typically designed for fixed neural network architectures and rely on assumptions or techniques (e.g., posterior factorization, knowledge distillation) that are ill-suited for SR, where model structures evolve over time.

In contrast, the application of FL to SR remains largely unexplored. To date, only one method has addressed horizontal FL in this context [14], relying on Mean Shift to aggregate model fitness across clients. This heuristic mechanism for model selection precludes parameter estimation and limits the search to symbolic expressions with fixed coefficients. This definitely compromises model search and convergence, as also reflected in our empirical comparison (Tables II, IV). By contrast, our BFSR framework introduces a principled approach to model selection via marginal likelihood estimation (or its approximation). Furthermore, it enables the aggregation of local client estimates to obtain either point-wise parameter estimates (after step 1) or distribution-wise estimates (after step 2). This not only improves predictive performance but also allows for uncertainty quantification, a critical aspect in many scientific and data-scarce applications.

Bayesian approaches to SR have recently gained traction, but have only been applied to centralized settings. Prior work has focused on model selection and the construction of model priors based on scientific corpora [28], [39] or domain-specific knowledge [40]. Marginal likelihood is often approximated using BIC or Laplace methods to ensure tractability [6], [28], [39], sometimes supplemented by priors that penalize operator complexity. Full Bayesian inference has also been explored, most notably by [29], who employed SMC and fractional Bayes factors for robust marginal likelihood estimation.

As our method should operate in federated settings where data sources are private and mostly heterogeneous, using informed priors is often impractical and risks introducing bias. For this reason, we adopt a flat prior over model structures and rely on BIC, combined with symbolic simplification via symbolic calculus, to constrain formula complexity, avoiding the need for prior-based bloat control. This operation proves effective, as shown by the modest model complexities in Table II. Moreover, instead of choosing between crude approximation or rigorous estimate of the marginal likelihood, we propose a two-stage approach: BIC is used to rapidly filter candidate models, followed by SMC to refine comparisons among theoretically grounded contenders. This hybrid pipeline



(a) $\gamma^2 = 1$ (b) $\gamma^2 = 0.1$

Fig. 3. Uncertainty Quantification of BFSR (W_{95} and W_{50}, Median $\pm$ IQR) in the 3-client setup.

TABLE IV
PREDICTIVE PERFORMANCE (R^2, MEAN $\pm$ SD) AND MODEL COMPLEXITY (C, MEAN $\pm$ SD) ACROSS METHODS AND UNCERTAINTY QUANTIFICATION (W_α, MEAN $\pm$ SD) FOR BFSR IN THE 10-CLIENT AND MOST HETEROGENEOUS SETUP ($\gamma^9 = 0.1$, $\alpha = 1$).

	Accuracy (R^2)		Complexity C		UQ (BFSR only)	
Dataset	**Mean Shift**	**BFSR**	**Mean Shift**	**BFSR**	$\mathbf{W_{95}}$	$\mathbf{W_{50}}$
F5-noise	-0.03 $\pm$ 0.23	**0.85 $\pm$ 0.05**	**9.4 $\pm$ 6.5**	21.3 $\pm$ 14.5	0.98 $\pm$ 0.03	0.46 $\pm$ 0.15
Pollen	0.57 $\pm$ 0.13	**0.78 $\pm$ 0.01**	26.6 $\pm$ 19.1	**11.7 $\pm$ 1.6**	0.94 $\pm$ 0.02	0.50 $\pm$ 0.03
Houses	0.47 $\pm$ 0.04	**0.53 $\pm$ 0.04**	17.2 $\pm$ 7.0	**15.0 $\pm$ 4.5**	0.94 $\pm$ 0.02	0.54 $\pm$ 0.06
Echo M.	0.36 $\pm$ 0.05	**0.41 $\pm$ 0.02**	18.4 $\pm$ 7.7	**13.3 $\pm$ 4.4**	0.93 $\pm$ 0.03	0.52 $\pm$ 0.05
DEXA	0.49 $\pm$ 0.03	**0.59 $\pm$ 0.03**	**15.7$\pm$ 5.0**	26.0 $\pm$ 3.1	0.95 $\pm$ 0.02	0.48 $\pm$ 0.03
Breast T.	-0.06 $\pm$ 0.21	**0.01 $\pm$ 0.01**	19.6 $\pm$ 17.2	**12.8 $\pm$ 5.5**	0.94 $\pm$ 0.04	0.55 $\pm$ 0.07

is well-suited to FL in SR, where full posterior inference at each generation step would be computationally prohibitive.

Finally, we emphasize the assumptions underlying the global BIC approximation derived via Gaussian posterior fusion in the large-sample regime, a formulation that mirrors, and is formally equivalent to, the FedAvg aggregation method. This approach typically presumes data homogeneity to ensure that the Fisher information matrices are approximately constant across clients, thereby justifying the weighted averaging of local estimators. This same assumption underlies the known limitations of FedAvg in heterogeneous settings. However, the requirement of distributional homogeneity can be substantially relaxed when clients share a common true regression model that is smooth in its parameters and locally identifiable. Under this condition, even in the presence of differing covariate distributions, local Fisher information matrices may remain sufficiently similar to enable effective aggregation. While such assumptions are often violated in overparameterized neural networks, symbolic regression models are typically low-dimensional, smooth, and interpretable, making local parameter estimation more stable and consistent. This expands the practical applicability of Gaussian posterior fusion, even under substantial data heterogeneity. The increased viability of this approach is empirically supported by the results in Section V, where model performance remains stable across both homogeneous and highly heterogeneous data configurations.

VII. CONCLUSIONS

This paper presents BFSR, a novel symbolic regression framework in horizontal federated learning. Designed for privacy-sensitive, decentralized environments, BFSR addresses key challenges in federated modeling, including statistical heterogeneity, limited data per client, and the need for interpretable models with quantified uncertainty.

BFSR is implemented in a two-stage Bayesian inference pipeline. The first stage performs efficient model selection during the evolutionary search via a global BIC, where global parameter estimates are computed through Gaussian posterior fusion in the large-sample limit. In the second stage, BFSR applies full Bayesian inference over a filtered subset of candidate models, enabling refined model selection, parameter estimation, and uncertainty quantification.

Extensive experiments across diverse data heterogeneity regimes demonstrate that BFSR consistently outperforms existing approaches in accuracy and scalability, also quantifying epistemic uncertainty. Its ability to produce interpretable and

robust symbolic models makes it especially suited for federated data mining tasks in real-world settings, such as healthcare and engineering, where privacy, model transparency, and trustworthiness are critical.

ACKNOWLEDGMENT

We acknowledge the CINECA award under the ISCRA initiative, for the availability of high performance computing resources and support.

REFERENCES

[1] N. Makke and S. Chawla, "Interpretable scientific discovery with symbolic regression: a review," *Artificial Intelligence Review*, vol. 57, no. 1, p. 2, 2024.

[2] Y. Mei, Q. Chen, A. Lensen, B. Xue, and M. Zhang, "Explainable artificial intelligence by genetic programming: A survey," *IEEE Transactions on Evolutionary Computation*, vol. 27, no. 3, pp. 621–641, 2023.

[3] W. La Cava, B. Burlacu, M. Virgolin, M. Kommenda, P. Orzechowski, F. O. de França, Y. Jin, and J. H. Moore, "Contemporary symbolic regression methods and their relative performance," *Advances in Neural Information Processing Systems*, vol. 2021, no. DB1, p. 1, 2021.

[4] J. R. Koza, "Genetic programming as a means for programming computers by natural selection," *Statistics and Computing*, vol. 4, no. 2, pp. 87–112, 1994.

[5] D. Angelis, F. Sofos, and T. E. Karakasidis, "Artificial intelligence in physical sciences: Symbolic regression trends and perspectives," *Archives of Computational Methods in Engineering*, vol. 30, no. 6, pp. 3845–3865, 2023.

[6] C. Wilstrup and J. Kasak, "Symbolic regression outperforms other models for small data sets," *arXiv preprint arXiv:2103.15147*, 2021.

[7] W. G. La Cava, P. C. Lee, I. Ajmal, X. Ding, P. Solanki, J. B. Cohen, J. H. Moore, and D. S. Herman, "A flexible symbolic regression method for constructing interpretable clinical prediction models," *npj Digital Medicine*, vol. 6, no. 1, p. 107, 2023.

[8] J. A. Hughes, S. Houghten, and J. A. Brown, "Gait model analysis of Parkinson's disease patients under cognitive load," in *2020 IEEE Congress on Evolutionary Computation (CEC)*, 2020, pp. 1–8.

[9] M. Virgolin, T. Alderliesten, A. Bel, C. Witteveen, and P. A. N. Bosman, "Symbolic regression and feature construction with GP-GOMEA applied to radiotherapy dose reconstruction of childhood cancer survivors," in *Proceedings of the Genetic and Evolutionary Computation Conference*. ACM, 2018, pp. 1395–1402.

[10] V. Guidetti, G. Dolci, E. Franceschini, E. Bacca, G. J. Burastero, D. Ferrari, V. Serra, F. Di Benedetto, C. Mussini, and F. Mandreoli, "Death after liver transplantation: Mining interpretable risk factors for survival prediction," in *2023 IEEE 10th International Conference on Data Science and Advanced Analytics (DSAA)*, 2023, pp. 1–10.

[11] D. Ferrari, V. Guidetti, Y. Wang, and V. Curcin, "Multi-objective symbolic regression to generate data-driven, non-fixed structure and intelligible mortality predictors using EHR: Binary classification methodology and comparison with state-of-the-art," in *AMIA Annual Symposium Proceedings*, vol. 2022, 2023, pp. 442–451.

[12] J. Wen, Z. Zhang, Y. Lan, Z. Cui, J. Cai, and W. Zhang, "A survey on federated learning: Challenges and applications," *International Journal of Machine Learning and Cybernetics*, vol. 14, no. 2, pp. 513–535, 2023.

[13] Q. Yang, Y. Liu, T. Chen, and Y. Tong, "Federated machine learning: Concept and applications," *ACM Transactions on Intelligent Systems and Technology (TIST)*, vol. 10, no. 2, pp. 1–19, 2019.

[14] J. Dong, J. Zhong, W.-N. Chen, and J. Zhang, "An efficient federated genetic programming framework for symbolic regression," *IEEE Transactions on Emerging Topics in Computational Intelligence*, vol. 7, no. 3, pp. 858–871, 2022.

[15] D. Nguyen Duy, M. Affenzeller, and R. Nikzad-Langerodi, "Towards vertical privacy-preserving symbolic regression via secure multiparty computation," in *Proceedings of the Companion Conference on Genetic and Evolutionary Computation*, 2023, pp. 2420–2428.

[16] M. Kommenda, B. Burlacu, G. Kronberger, and M. Affenzeller, "Parameter identification for symbolic regression using nonlinear least squares," *Genetic Programming and Evolvable Machines*, vol. 21, no. 3, pp. 471–501, 2020.

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

[18] X. Li, K. Huang, W. Yang, S. Wang, and Z. Zhang, "On the convergence of FedAvg on non-IID data," *arXiv preprint arXiv:1907.02189*, 2019.

[19] T. Li, A. K. Sahu, M. Zaheer, M. Sanjabi, A. Talwalkar, and V. Smith, "Federated optimization in heterogeneous networks," *Proceedings of Machine Learning and Systems*, vol. 2, pp. 429–450, 2020.

[20] X. Zhang, Y. Li, W. Li, K. Guo, and Y. Shao, "Personalized federated learning via variational Bayesian inference," in *International Conference on Machine Learning*. PMLR, 2022, pp. 26 293–26 310.

[21] H.-Y. Chen and W.-L. Chao, "FedBE: Making Bayesian model ensemble applicable to federated learning," *arXiv preprint arXiv:2009.01974*, 2020.

[22] L. Liu, X. Jiang, F. Zheng, H. Chen, G.-J. Qi, H. Huang, and L. Shao, "A Bayesian federated learning framework with online Laplace approximation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 46, no. 1, pp. 1–16, 2024.

[23] M. Virgolin, E. Medvet, T. Alderliesten, and P. A. Bosman, "Less is more: A call to focus on simpler models in genetic programming for interpretable machine learning," *arXiv preprint arXiv:2204.02046*, 2022.

[24] M. Virgolin and S. P. Pissis, "Symbolic regression is NP-hard," *Transactions on Machine Learning Research*, 2022.

[25] G. Kronberger, "Local optimization often is ill-conditioned in genetic programming for symbolic regression," in *2022 24th International Symposium on Symbolic and Numeric Algorithms for Scientific Computing (SYNASC)*. IEEE, 2022, pp. 304–310.

[26] G. Kronberger, F. Olivetti de Franca, H. Desmond, D. J. Bartlett, and L. Kammerer, "The inefficiency of genetic programming for symbolic regression," in *International Conference on Parallel Problem Solving from Nature*. Springer, 2024, pp. 273–289.

[27] J. Zhang, Y. Yang, and J. Ding, "Information criteria for model selection," *Wiley Interdisciplinary Reviews: Computational Statistics*, vol. 15, no. 5, p. e1607, 2023.

[28] R. Guimerà, I. Reichardt, A. Aguilar-Mogas, F. A. Massucci, M. Miranda, J. Pallarès, and M. Sales-Pardo, "A Bayesian machine scientist to aid in the solution of challenging scientific problems," *Science Advances*, vol. 6, no. 5, p. eaav6971, 2020.

[29] G. F. Bomarito, P. E. Leser, N. Strauss, K. Garbrecht, and J. D. Hochhalter, "Automated learning of interpretable models with quantified uncertainty," *Computer Methods in Applied Mechanics and Engineering*, vol. 403, p. 115732, 2023.

[30] G. Schwarz, "Estimating the dimension of a model," *The Annals of Statistics*, vol. 6, no. 2, pp. 461–464, 1978.

[31] H. Jeffreys, *The Theory of Probability*. OUP, 1998.

[32] R. E. Kass and A. E. Raftery, "Bayes factors," *Journal of the American Statistical Association*, vol. 90, no. 430, pp. 773–795, 1995.

[33] O. Abril-Pla, V. Andreani, C. Carroll, L. Dong, C. J. Fonnesbeck, M. Kochurov, R. Kumar, J. Lao, C. C. Luhmann, O. A. Martin *et al.*, "PyMC: a modern, and comprehensive probabilistic programming framework in Python," *PeerJ Computer Science*, vol. 9, p. e1516, 2023.

[34] P. Del Moral, A. Doucet, and A. Jasra, "Sequential Monte Carlo samplers," *Journal of the Royal Statistical Society Series B: Statistical Methodology*, vol. 68, no. 3, pp. 411–436, 2006.

[35] M. Yurochkin, M. Agarwal, S. Ghosh, K. Greenewald, N. Hoang, and Y. Khazaeni, "Bayesian nonparametric federated learning of neural networks," in *International Conference on Machine Learning*. PMLR, 2019, pp. 7252–7261.

[36] L. Sluijterman, E. Cator, and T. Heskes, "How to evaluate uncertainty estimates in machine learning for regression?" *Neural Networks*, vol. 173, p. 106203, 2024.

[37] Y. Zhao, M. Li, L. Lai, N. Suda, D. Civin, and V. Chandra, "Federated learning with non-iid data," *arXiv preprint arXiv:1806.00582*, 2018.

[38] H. Zhu, J. Xu, S. Liu, and Y. Jin, "Federated learning on non-iid data: A survey," *Neurocomputing*, vol. 465, pp. 371–390, 2021.

[39] D. Bartlett, H. Desmond, and P. Ferreira, "Priors for symbolic regression," in *Proceedings of the Companion Conference on Genetic and Evolutionary Computation*, 2023, pp. 2402–2411.

[40] T. Schneider, A. Totounferoush, W. Nowak, and S. Staab, "Probabilistic regular tree priors for scientific symbolic reasoning," *arXiv preprint arXiv:2306.08506*, 2023.