


## Article

# Stochastic Configuration Based Fuzzy Inference System with Interpretable Fuzzy Rules and Intelligence Search Process

Wei Zhou <sup>1,2,\*</sup> , Hongxing Li <sup>1</sup> and Menghong Bao <sup>2</sup><sup>1</sup> School of Control Science and Engineering, Dalian University of Technology, Dalian 116024, China<sup>2</sup> School of Applied Mathematics, Beijing Normal University, Zhuhai 519087, China

\* Correspondence: zhouwei81@bnu.edu.cn

**Abstract:** In this paper, a stochastic configuration based fuzzy inference system with interpretable fuzzy rules (SCFS-IFRs) is proposed to improve the interpretability and performance of the fuzzy inference system and determine autonomously an appropriate model structure. The proposed SCFS-IFR first accomplishes a fuzzy system through interpretable linguistic fuzzy rules (ILFRs), which endows the system with clear semantic interpretability. Meanwhile, using an incremental learning method based on stochastic configuration, the appropriate architecture of the system is determined by incremental generation of ILFRs under a supervision mechanism. In addition, the particle swarm optimization (PSO) algorithm, an intelligence search technique, is used in the incremental learning process of ILFRs to obtain better random parameters and improve approximation accuracy. The performance of SCFS-IFRs is verified by regression and classification benchmark datasets. Regression experiments show that the proposed SCFS-IFRs perform best on 10 of the 20 data sets, statistically significantly outperforming the other eight state-of-the-art algorithms. Classification experiments show that, compared with the other six fuzzy classifiers, SCFS-IFRs achieve higher classification accuracy and better interpretation with fewer rules.

**Keywords:** interpretable linguistic fuzzy rules; fuzzy system; incremental learning; stochastic configuration; particle swarm optimization

**MSC:** 68T05

**Citation:** Zhou, W.; Li, H.; Bao, M. Stochastic Configuration Based Fuzzy Inference System with Interpretable Fuzzy Rules and Intelligence Search Process. *Mathematics* **2023**, *11*, 614. <https://doi.org/10.3390/math11030614>

Academic Editor: Junzo Watada

Received: 30 November 2022

Revised: 20 January 2023

Accepted: 21 January 2023

Published: 26 January 2023



**Copyright:** © 2023 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

## 1. Introduction

Neuro-fuzzy systems [1,2], which inherit the advantages of both fuzzy inference systems and neural networks, have attracted sustained attention. The fuzzy rules in neuro-fuzzy systems make the system interpretable and the trainable parameters make the precision of the system improvable via learning.

Interpretability is a very important research direction for neuro-fuzzy systems. Since interpretability is considered as the ability to express the meaning of a model in an understandable way, it is of great help to understand the function of the model and what is learned from it [3]. Although we have not yet found a way to measure interpretability mathematically, a fuzzy inference system with high interpretability should have comprehensible fuzzy partition and simple fuzzy rules. We should use fuzzy sets that are distinguishable and semantically clear and use as few fuzzy rules as possible to build an interpretable rule base. In [4], a fuzzy extreme learning machine (FELM) uses ILFRs to improve the interpretability of an extreme learning machine (ELM). ILFRs use several fuzzy sets to equally divide the input universe of discourse in order to better endow the fuzzy sets with semantic labels and generate random matrices to select features and to combine rules. ILFRs have been concerned in neuro-fuzzy modeling because of their advantages in interpretability. In [5,6], two deep fuzzy classifiers which adopt ILFRs are proposed. In order to improve the model transparency of the fuzzy broad learning system (FBLs),

a Compact FBLS (CFBLS) using ILFRs has been improved in [7]. Although there are a large number of neuro-fuzzy systems which have been developed to improve interpretability and approximation accuracy, there is still room for further improvement. It is still a significant research problem to achieve good approximation accuracy with an appropriate number of fuzzy rules and determine the number of these fuzzy rules autonomously rather than through trial and error.

The most commonly used fuzzy system in the neural fuzzy model is the Takagi-Sugeno-Kang (TSK), which has been widely applied in many fields, such as regression [8], classification [9], model approximation [10], adaptive control [11], business [12], etc. Neural-fuzzy systems adopt the form of a neural network to represent the fuzzy system, and the training algorithm in the neural network is used to realize the extraction of fuzzy rules and the establishment of the fuzzy system without tedious manual intervention. The back propagation (BP) algorithm, a commonly used training algorithm in neural network, has been used for training neuro-fuzzy systems [13]. However, as datasets grow in size and dimension, the training algorithm of the neuro-fuzzy system has encountered many difficulties, such as training rate, overfitting, low interpretability, etc.

Because of the effectiveness and efficiency of randomized learning techniques, they have great potential in developing fast learning algorithms for neuro-fuzzy systems. In [14], the proposed Bernstein fuzzy system uses the ELM training system to simultaneously approximate smooth functions and their derivatives. In the generalized Bernstein fuzzy system, the premise parameters are randomly generated and the consequent parameters are determined by solving a linear optimization problem. In [15], a regularized extreme learning adaptive neuro-fuzzy inference system (R-ELANFIS) is designed to improve the generalization performance of a neuro-fuzzy system. In [16], a broad learning system (BLS) is proposed for fast learning of a large volume of data, which randomly assign the values of input parameters and expand the neurons in a broad manner without a retraining process. In [17], a TSK fuzzy system is combined with a broad learning system to establish a fuzzy broad learning system.

The scope of random parameters plays an important role in the training process of randomized learning approaches and has been proved to affect approximation capability [18]. A stochastic configuration network (SCN) is proposed in [19] to improve randomized learning techniques. SCN is built incrementally by assigning the random parameters under a supervisory mechanism and adaptively setting their range. The supervisory mechanism of SCN is helpful for ensuring universal approximation and good performance. Various SCN models have been developed over the past decade. In [20], a deep SCN is developed to randomly assign the weights of each layer under inequality constraints. In [21], a deep stacked SCN is established to automatically extract hidden units from data streams. In [22], the driving amount is introduced into the structure of SCN to improve the generalization approximation ability.

In this paper, a novel stochastic configuration based fuzzy inference system called SCFS-IFRs is proposed to improve the interpretability and performance of the fuzzy system, which uses an adaptive incremental learning approach based on the stochastic configuration algorithm to determine an appropriate architecture autonomously and ensure universal approximation and continuous performance improvement of SCFS-IFRs. The contribution of SCFS-IFRs can be summarized as follows:

- (1) For high interpretability, SCFS-IFRs implement a fuzzy system through ILFRs and determine the appropriate number of fuzzy rules by an adaptive incremental learning algorithm. ILFRs better endow fuzzy sets with semantic labels by equally dividing the input universe of discourse and the rule combination matrix is used to select activated fuzzy sets. They help to improve the semantic expression of fuzzy rules. Moreover, the incremental learning method of SCFS-IFRs which determines a suitable number of fuzzy rules, is also beneficial for improving interpretability and avoiding the problem of “rule explosion”.

- (2) For good performance, a kind of supervisory mechanism and an intelligence search process are to be considered in the adaptive incremental learning algorithm of SCFS-IFRs. The inequality constraints that the premise parameters of fuzzy rules need to satisfy are given, which can ensure that the performance of SCFS-IFRs will be improved along with the increase in rules. The PSO algorithm, an intelligence search technique, is used in the process of stochastic configuration to obtain better random parameters.

The rest of this article is organized as follows. In Section 2, we review the TSK fuzzy system and SCN. In Section 3, we elaborate the proposed SCFS-IFRs. In Section 4, numerical experiments verify the performance of the proposed SCFS-IFRs. Some concluding remarks are given in Section 5.

The abbreviations and variables used in this paper are listed in Tables 1 and 2.

**Table 1.** Table of abbreviations.

Abbreviations	Full Name
ILFRs	interpretable linguistic fuzzy rules
PSO	particle swarm optimization
ELM	extreme learning machine
FELM	fuzzy extreme learning machine
BLS	broad learning system
FBLS	fuzzy broad learning system
CFBLS	compact fuzzy broad learning system
TSK	Takagi-Sugeno-Kang
TSK-FC	TSK fuzzy classifier
BP	back propagation
R-ELANFIS	regularized extreme learning adaptive neuro-fuzzy inference system
SCN	stochastic configuration network
RVFLN	random vector functional link neural network
HPFNN	hierarchical polynomial-based fuzzy neural networks
BL-DFIS	broad learning based dynamic fuzzy inference system
SCFS-IFRs	stochastic configuration based fuzzy inference system with interpretable fuzzy rules

**Table 2.** Table of variables.

Variables	Interpretation
$u_k$	center of the $k$ th fuzzy set, predefined constant
$\sigma_L$	width of fuzzy set in the $L$ th fuzzy rule, probabilistic variables
$C_L$	rule combination matrix of the $L$ th rule, probabilistic variables
$a_L$	weight parameters of the $L$ th rule
$e_L$	residual error of system with $L$ rule
$\varepsilon$	predefined tolerance level
$r$	real variable with adaptively values between 0 and 1
$\{\mu_L\}$	real sequence with a limit of zero and adaptively values
$\lambda$	predefined regular coefficient
$I$	identity matrix
$L_{max}$	predefined maximum number of rules
$K$	predefined number of fuzzy sets for each feature
$b_\sigma$	predefined upper bound of the width parameter $\sigma_L$
$(V_C^t, V_\sigma^t)$	velocity vectors of population of PSO
$w$	predefined inertia weight factor of PSO
$c_1, c_2$	predefined positive acceleration constants of PSO
$r_1, r_2$	random factors of PSO, probabilistic variables
$n_{max}$	predefined the maximum number of iterations of PSO
$T$	predefined the population size of PSO

## 2. Preliminaries

### 2.1. Takagi-Sugeno-Kang Fuzzy Inference System

The Takagi-Sugeno-Kang (TSK) fuzzy system proposed in [23], as one of the most important fuzzy systems, has been applied extensively in many different fields [24]. The consequent part of the fuzzy rule is the function of the input variables rather than the fuzzy set. Specifically, given an input vector  $\mathbf{x} = (x_1, x_2, \dots, x_m)^T$ , the typical fuzzy rules are of the form:

$$R_l : \text{If } x_1 \text{ is } A_{l1} \text{ and } \dots \text{ and } x_m \text{ is } A_{lm}, \text{ then } y = f_l(\mathbf{x}), l = 1, 2, \dots, L \quad (1)$$

where  $A_{lj}$  is an antecedent fuzzy set of variable  $x_j$  for the rule  $R_l$ ,  $f_l(\mathbf{x})$  is the consequent function of the rule  $R_l$  and  $L$  is the number of fuzzy rules.

Although the popular first-order TSK fuzzy system sets the consequent function  $f_l(\mathbf{x})$  as linear function of the input variables,  $f_l(\mathbf{x})$  can be any form function. The linear combination of a subset of the Chebyshev polynomial has been adopted by the rule consequent for improving the non-linear expression ability of TSK fuzzy system in [25], which is of the form.

$$f_l(\mathbf{x}) = \phi(\mathbf{x})^T \mathbf{a}_l \quad (2)$$

where  $\phi(\mathbf{x}) = (1, T_1(x_1), \dots, T_1(x_m), T_2(x_1), \dots, T_n(x_m))^T$  is the expanded input vector of form,  $\mathbf{a}_l \in \mathbb{R}^{nm+1}$  is the weight of the linear combination,  $T_{k+1}(x_j)$  is the Chebyshev polynomial of the form  $T_{k+1}(x_j) = 2x_j T_k(x_j) - T_{k-1}(x_j)$  with  $T_0(x_j) = 1, T_1(x_j) = x_j$  and  $n$  is the highest degree of the used Chebyshev polynomial.

For the  $l$ th fuzzy rule  $R_l$ , the activation level is:

$$\tau_l(\mathbf{x}) = \prod_{j=1}^m \mu_{A_{lj}}(x_j) \quad (3)$$

where  $\mu_{A_{lj}}(x_j)$  is the membership function of  $A_{lj}$ .

As pointed out in [26–28], the output system (1) can be calculated as

$$y = \sum_{l=1}^L \tau_l(\mathbf{x}) f_l(\mathbf{x}) = \sum_{l=1}^L \tau_l(\mathbf{x}) \phi(\mathbf{x})^T \mathbf{a}_l \quad (4)$$

In the above TSK fuzzy system, the parameters of membership function  $\mu_{A_{lj}}$  and the consequent weight  $\mathbf{a}_l$  need to be determined. Traditional gradient descent algorithms can be used to identify these parameters, but is usually time-consuming with a great deal of input data. Many learning algorithms [29–31] have been proposed for accelerating the parameter identification procedure and developing the approximation performance. For example, the premise parameters can be generated randomly and the consequent parameters can be quickly calculated using a pseudoinverse.

### 2.2. Stochastic Configuration Network

The stochastic configuration network (SCN) [19], which develops a randomized method, generates random parameters under a supervisory mechanism. It can guarantee the approximation capability and determine an appropriate architecture.

For a giving target function  $f : \mathbb{R}^m \rightarrow \mathbb{R}$ , an SCN with  $L - 1$  hidden nodes is denoted as

$$y = \sum_{j=1}^{L-1} \beta_j g(\mathbf{w}_j^T \mathbf{x} + b_j) \quad (5)$$

where  $g(\mathbf{w}_j^T \mathbf{x} + b_j)$  is the activation function with random parameters  $\mathbf{w}_j \in \mathbb{R}^m$  and  $b_j \in \mathbb{R}$ ,  $\beta_j \in \mathbb{R}$  is the output weight of the hidden node. The residual error is denoted as

$$e_{L-1} = f - \sum_{j=1}^{L-1} \beta_j g(\mathbf{w}_j^T \mathbf{x} + b_j) \quad (6)$$

If  $e_{L-1}$  is bigger than a predefined error tolerance  $\varepsilon$ , the SCN framework incrementally adds hidden nodes until the residual error reaches  $\varepsilon$ . The supervisory mechanism of an inequality constraint is of the form

$$\langle e_{L-1}, g_L \rangle^2 \geq (1 - r - \mu_L) \|e_{L-1}\|^2 \quad (7)$$

where  $r \in (0, 1)$  and  $\{\mu_L\}$  is a sequence with  $\lim_{L \rightarrow +\infty} \mu_L = 0$  and  $0 < \mu_L \leq (1 - r)$  and  $g_L(\mathbf{x}) = g(\mathbf{w}_L^T \mathbf{x} + b_L)$  is the random basis function of the new hidden node.

The output parameters  $\beta_j$ ,  $j = 1, 2, \dots, L$  are calculated by

$$[\beta_1^*, \beta_2^*, \dots, \beta_L^*] = \underset{\beta}{\operatorname{argmin}} \|f - \sum_{j=1}^L \beta_j g_j\|^2 \quad (8)$$

The leading model  $y = \sum_{j=1}^L \beta_j g_j$  shows continuous performance improvement and the universal approximation of SCN have been proved in [19].

Although the existence of random parameters satisfying the constraint conditions (7) has been proved theoretically, how to find better parameters among these satisfying constraint conditions is still a meaningful problem. In this paper, the PSO algorithm is applied to find better parameters to further improve the performance of the system.

### 3. Stochastic Configuration Based Fuzzy Inference System with Interpretable Fuzzy Rules and Intelligence Search Process

The SCFS-IFRs model establishes a TSK fuzzy system of form (1) by adopting the following new method.

- (1) The ILFRs with the rule combination matrix are used to construct the TSK fuzzy system to guarantee model interpretability and avoid the “rule explosion” problem.
- (2) An adaptive incremental learning algorithm based on stochastic configuration is proposed to train the SCFS-IFR model and fix the suitable number of fuzzy rules.
- (3) The PSO algorithm is adopted to search for better parameters to improve the performance of the system in the process of stochastic configuration.

This section elaborates the proposed SCFS-IFRs. Firstly, the general structure of the SCFS-IFRs is introduced and then their adaptive learning algorithm is presented.

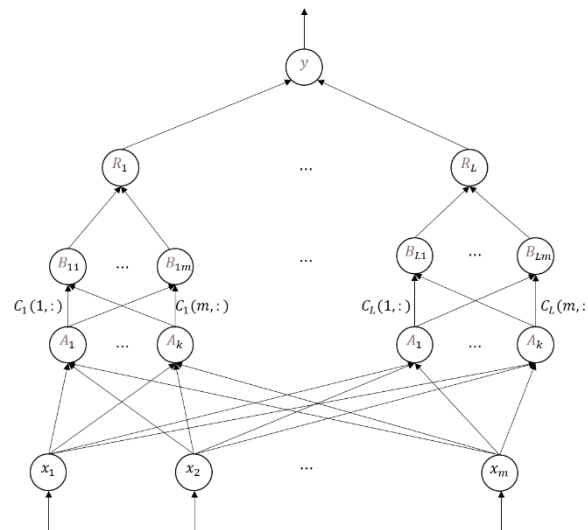
#### 3.1. General Structure of SCFS-IFRs

The details of SCFS-IFRs shown in Figure 1 are given as follows. Given a dataset with inputs  $\mathbf{X} = (\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_N)^T$  and outputs  $\mathbf{Y} = (\mathbf{y}_1, \mathbf{y}_2, \dots, \mathbf{y}_N)^T$ , where  $\mathbf{x}_i = (x_{i1}, x_{i2}, \dots, x_{im})$  and  $\mathbf{y}_i = (y_{i1}, y_{i2}, \dots, y_{it})$ ,  $j = 1, 2, \dots, N$  and suppose all input features are normalized to  $[0, 1]$ . We equally divide  $(0, 1)$  into  $K - 1$  parts, and correspondingly, each feature is transformed into  $K$  fuzzy sets  $\{A_k\}_{k=1}^K$  which are defined as follows

$$\mu_{A_k}(x_{ij}) = \exp\left(-\frac{(x_{ij} - u_k)^2}{2\sigma^2}\right), \quad k = 1, 2, \dots, K, \quad j = 1, 2, \dots, m \quad (9)$$

where  $\mu_{A_k}(\cdot)$  is the Gaussian membership function of the  $k$ th fuzzy set,  $x_{ij}$  is the  $j$ th dimension of input feature  $\mathbf{x}_i$ ,  $u_k = \frac{k-1}{K}$  is the center of fuzzy set  $A_k$  and  $\sigma$  is the randomly generated width. Here, the  $K$  fuzzy sets can be assigned  $K$  explicit linguistic labels. For instance, when  $K = 5$ , the five fuzzy sets can be assigned 5 linguistic labels: very low, low,

medium, high, and very high. Compared with the random allocation of center parameters, the traditional equidistant partition method makes the fuzzy set more interpretable but may lead to “rule explosion”.



**Figure 1.** Illustration of the SCFS-IFRs.

Motivated by [6], the rule combination matrix  $C \in \{0, 1\}^{m \times K \times L}$  is used to determine which fuzzy sets of the input features are used to generate the  $L$  fuzzy rules, where  $C(j, k, l) = 1$  represents the  $k$ th fuzzy set of the  $j$ th feature in which the  $l$ th fuzzy rule will be activated; conversely, it will not be activated. To ensure that at least one fuzzy set is activated for each feature in each rule, we require that  $\sum_{k=1}^K C(j, k, l) \geq 1, j = 1, 2, \dots, m, l = 1, 2, \dots, L$ .  $B_{lj}$  denotes the union of all activated fuzzy sets of the  $j$ th feature in the  $l$ th fuzzy rule, i.e.,

$$B_{lj} = \bigcup_{C(j,k,l)=1} A_k, j = 1, 2, \dots, m \quad (10)$$

The SCFS-IFRs with  $L$  ILFRs take the following form:

$$R_l : \text{if } x_{i1} \text{ is } B_{l1} \text{ and } x_{i2} \text{ is } B_{l2} \text{ and } \dots \text{ and } x_{im} \text{ is } B_{lm}, \text{ then } y = \phi(\mathbf{x}_i)^T \mathbf{a}_l, \\ l = 1, 2, \dots, L \quad (11)$$

We can adopt the operator of algebraic sum  $s(a, b) = a + b - ab$  to calculate the membership function of  $B_{lj}$  as

$$\mu_{B_{lj}}(x_{ij}) = 1 - \prod_{k=1}^K (1 - C(j, k, l) \mu_{A_k}(x_{ij})), j = 1, 2, \dots, m, l = 1, 2, \dots, L \quad (12)$$

Therefore, the output and residual error of the SCFS-IFRs (11) on dataset  $\{\mathbf{X}, \mathbf{Y}\}$  can be calculated as

$$\mathbf{S}_L = \sum_{l=1}^L \begin{pmatrix} \tau_l(\mathbf{x}_1) \phi(\mathbf{x}_1)^T \\ \vdots \\ \tau_l(\mathbf{x}_N) \phi(\mathbf{x}_N)^T \end{pmatrix} \mathbf{a}_l = \sum_{l=1}^L \eta_l \mathbf{a}_l \quad (13)$$

$$\mathbf{e}_L = \mathbf{Y} - \mathbf{S}_L \quad (14)$$

where  $\tau_l(\mathbf{x}_i) = \prod_{j=1}^m B_{lj}(x_{ij}) = \prod_{j=1}^m \left(1 - \prod_{k=1}^K (1 - C(j, k, l) \mu_{A_k}(x_{ij}))\right)$  is the activation level of the  $l$ th fuzzy rule,  $a_l$  is the weight of the linear combination of the expanded input vector  $\phi(\mathbf{x}_i)$  and denotes

$$\eta_l = \begin{pmatrix} \tau_l(\mathbf{x}_1) \phi(\mathbf{x}_1)^T \\ \vdots \\ \tau_l(\mathbf{x}_N) \phi(\mathbf{x}_N)^T \end{pmatrix} \quad (15)$$

**Remark 1.** A SCFS-IFRs model is implemented in the framework of the TSK fuzzy system. The semantically clear fuzzy sets generated by equidistant partition and the rule combination matrix used to generate fuzzy rules contribute to the high interpretability of SCFS-IFRs.

For instance, let us have  $L = 2$  fuzzy rules,  $m = 2$  features for each input sample  $\mathbf{x}_i$ ,  $K = 5$  fuzzy sets with linguistic labels: very low, low, medium, high, and very high. Given

$$C(:, :, 1) = \begin{pmatrix} 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 \end{pmatrix}, \quad C(:, :, 2) = \begin{pmatrix} 0 & 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 0 \end{pmatrix}$$

The two ILFRs can be expressed as

- $R_1$  : if  $x_{i1}$  is very low or low and  $x_{i2}$  is medium and  $x_{i3}$  is high or very high, then  $y = \phi(\mathbf{x}_i)^T \mathbf{a}_1$   
 $R_2$  : if  $x_{i1}$  is very high and  $x_{i2}$  is very low and  $x_{i3}$  is medium or high, then  $y = \phi(\mathbf{x}_i)^T \mathbf{a}_2$

In addition, the rule combination matrix defined in this paper allows multiple fuzzy sets corresponding to each feature in a fuzzy rule to be activated, which improves the semantic expression of fuzzy rules and makes it possible to express more complex relations with fewer rules.

### 3.2. Adaptive Incremental Learning for SCFS-IFRs

We propose an adaptive incremental learning algorithm to train the proposed SCFS-IFRs, which can be helpful in determining an appropriate model structure autonomously and ensure the approximation capability of SCFS-IFRs. In order to further improve the approximation accuracy, the PSO algorithm is adopted to search for better random parameters in the process of stochastic configuration.

The adaptive incremental learning algorithm starts with an initial system, which can be either an empty system or a system with  $L_0$  rules generated by other traditional methods. Given training dataset  $\{X, Y\}$ , new fuzzy rules are increased incrementally in the light of a supervisory mechanism of stochastic configuration for improving the approximation performance until a predefined tolerance level  $\varepsilon$  is met.

Suppose that there is SCFS-IFRs  $S_{L-1}$  with  $L - 1$  fuzzy rules of the form (11). The output and residual error of  $S_{L-1}$  on dataset  $\{X, Y\}$  are expressed in the form

$$S_{L-1} = \sum_{l=1}^{L-1} \eta_l a_l, \quad e_{L-1} = Y - S_{L-1}$$

If  $\|e_{L-1}\|$  does not reach the predefined tolerance level  $\varepsilon$ , i.e.,  $\|e_{L-1}\| \geq \varepsilon$ , a new fuzzy rule  $R_L$  should be generated to construct the new SCFS-IFRs  $S_L$  with  $L$  fuzzy rules. There are three parameters that should be determined for generating rule  $R_L$ : the rule combination matrix (denoted by  $C_L = C(:, :, L)$ ), the width of gaussian membership function (denoted by  $\sigma_L$ ) and the weight of the linear combination in the consequent part (denoted by  $a_L$ ).



The premise parameters  $C_L$  and  $\sigma_L$  are randomly produced to satisfy the following inequalities

$$\|\eta_L(\eta_L^T \eta_L + \lambda I_M)^{-1}(\eta_L^T e_{L-1})\|^2 \geq (1 - r - \mu_L)\|e_{L-1}\|^2 \quad (16)$$

and the consequent parameter  $a_L$  can be evaluated by

$$a_L = (\eta_L^T \eta_L + \lambda I)^{-1}(\eta_L^T e_{L-1}) \quad (17)$$

where  $\lambda$  is the regular coefficient,  $I$  is the identity matrix,  $\eta_L$  is of form (15) and  $r \in (0, 1)$  and  $\{\mu_L\}$  is a sequence with  $\lim_{L \rightarrow \infty} \mu_L = 0$  and  $0 < \mu_L \leq (1 - r)$ .

The output and the residual error of the  $S_L$  can be represented as

$$S_L = \sum_{l=1}^L \eta_l a_l, \quad e_L = Y - S_L = e_{L-1} - \eta_L a_L$$

and the improvement brought about from the new rule  $R_L$  can be calculated as

$$P_L = \|e_L\|^2 - \|e_{L-1}\|^2 = \|e_{L-1} - \eta_L a_L\|^2 - \|e_{L-1}\|^2 = -\|\eta_L(\eta_L^T \eta_L + \lambda I_M)^{-1}(\eta_L^T e_{L-1})\|^2. \quad (18)$$

The improvement  $P_L$  provides the target function for the choice of parameters  $C_L$  and  $\sigma_L$ . There may be many values of parameters  $C_L$  and  $\sigma_L$  that satisfy the inequality constraint (16). We adopt the PSO algorithm to search for better random parameters  $C_L$  and  $\sigma_L$  which satisfy the inequality constraint (16) and have smaller values of improvement  $P_L$ , in order to improve the system performance.

In the process of stochastic configuration using the PSO algorithm, firstly  $T$  pairs of parameters  $\{(C_L^t, \sigma_L^t)\}_{t=1}^T$  within  $\{0, 1\}^{m \times K} \times (0, b_\sigma)$  are randomly initialized as the position of initial population; correspondingly,  $T$  pairs of parameters  $\{(V_C^t, V_\sigma^t)\}_{t=1}^T$  within  $(-1, 1)^{m \times K} \times (-b_\sigma, b_\sigma)$  are randomly initialized as the velocity vectors of the initial population. The fitness value  $P_L^t$  of particle  $t$  can be calculated by Equation (18).

At each iteration, the positions and velocity vectors can be updated as

$$(V_C^t, V_\sigma^t)_{new} = w(V_C^t, V_\sigma^t)_{old} + c_1 r_1 ((C_L^t, \sigma_L^t)_p - (C_L^t, \sigma_L^t)_{old}) + c_2 r_2 ((C_L^g, \sigma_L^g) - (C_L^t, \sigma_L^t)_{old}), \quad (19)$$

$$(C_L^t, \sigma_L^t)_{temp} = (C_L^t, \sigma_L^t)_{old} + (V_C^t, V_\sigma^t)_{new} \quad (20)$$

where  $w$  is the inertia weight factor,  $c_1$  and  $c_2$  are positive acceleration constants,  $r_1$  and  $r_2$  are random factors sampled from a uniform distribution  $U(0, 1)$ ,  $(C_L^t, \sigma_L^t)_p$  are  $(C_L^g, \sigma_L^g)$  represent the best position in the history of particle  $t$  and the whole population, respectively. In this paper, we set the parameters of PSO as  $w = 0.9$ ,  $c_1 = c_2 = 2$ . Through  $n_{max}$  iterations of PSO, we can obtain better parameters to generate new fuzzy rules to improve the performance of the system.

In this way, we gradually generate new fuzzy rules one by one and add them to the system until the residual error reaches the predefined error tolerance  $\varepsilon$  or the number of rules exceeds the given maximum rules number  $L_{max}$ . The adaptive incremental learning algorithm can be summarized as follows.

Adaptive incremental learning algorithm of SCFS-IFRs:

Given dataset with inputs  $X = (x_1, x_2, \dots, x_N)^T$  and outputs  $Y = (y_1, y_2, \dots, y_N)^T$ , the predefined maximum number of rules  $L_{max}$ , the error tolerance  $\varepsilon$ , the population size  $T$ , the maximum number of iterations of the PSO  $n_{max}$ , the number of fuzzy sets  $K$  for each feature, the upper bound  $b_\sigma$  of the width parameter and the regularization parameter  $\lambda$ .

1. Initialize  $e = Y$ ,  $L = 0$ ,  $0 < r < 1$ ;
2. Set center of Gaussian membership function  $u_k = \frac{k-1}{K}$ ,  $k = 1, 2, \dots, K$ ;



3. While  $L < L_{max}$  and  $\|e\| > \varepsilon$ , Do
4. Set  $\mu_L = (1 - r)/(L + 1)$ ;
5. Randomly initialize  $T$  particles with position and velocity vectors as  $\{(C_L^t, \sigma_L^t)\}_{t=1}^T$  and  $\{(V_C^t, V_\sigma^t)\}_{t=1}^T$
6. For  $i = 1$  to  $n_{max}$  do
7. Calculate the fitness value  $\{P_L^t\}_{t=1}^T$  according to Equation (18);
8. Update the positions  $\{(C_L^t, \sigma_L^t)\}_{t=1}^T$  and velocity vectors  $\{(V_C^t, V_\sigma^t)\}_{t=1}^T$  using Equations (19) and (20);
9. End for
10. Set the best position as  $(C_L^g, \sigma_L^g)$  as  $(C_L, \sigma_L)$ ;
11. Calculate  $P_L$  according to Equation (18);
12. if  $P_L \leq -r\|e_{L-1}\|^2$  Calculate  $\eta_L$  and  $a_L$  according to Equations (15) and (17), respectively;
13. Else randomly take  $\tau \in (0, r)$ , renew  $r := r + \tau$ , return to Step 4;
14. End if
15. Update  $e := e - \eta_L a_L$ ;
16. End while
17. Return  $\{(C_l, \sigma_l), a_l\}_{l=1}^L$ .

In this way, the adaptive incremental learning algorithm determines the structure and parameters of SCFS-IFRs. The random parameters  $(C_L, \sigma_L)$  are adaptively determined according to the inequality constraint (16) and the appropriate architecture is determined. In the adaptive incremental learning algorithm, while the random parameters  $(C_L, \sigma_L)$  are optimized, the number of fuzzy sets  $K$  for each feature, the regularization parameter  $\lambda$ , and the degree of Chebyshev polynomial  $n$  are predefined and do not need to be optimized.

**Remark 2.** The approximation capability of SCFS-IFRs can be analyzed as follows. If the random parameters  $(C_L, \sigma_L)$  of the new fuzzy rule satisfies the inequality constraint (16) and the consequent parameter  $a_L$  is evaluated by Equation (17), we can find that  $\|e_L\|^2 - \|e_{L-1}\|^2 \leq -(1 - r - \mu_L)\|e_{L-1}\|^2$  according to Equation (18). This means that  $\|e_L\|^2 \leq (r + \mu_L)\|e_{L-1}\|^2$ . Since  $\lim_{L \rightarrow \infty} \mu_L = 0$ , it is obvious that  $\lim_{L \rightarrow \infty} \|e_L\|^2 = 0$ , which proves the approximation capability of SCFS-IFRs.

#### 4. Performance Evaluation

In this section, the performance of the proposed SCFS-IFRs is investigated. The proposed SCFS-IFRs is compared with some representative models such as RVFLN [32], ELM [33], RVFL-RS [34], HPFNN [35], SCN [19], SCBLS [36], BLS [16], FBLS [7], TSK-FC [37], L2TSK-FC [37], HID-TSK [6], CFBLs [7], FELM [4] and BL-DFIS [38] through numerical experiments on 20 regression datasets and 10 classification datasets.

For a fair comparison, we use the same training strategies for SCFS-IFRs as [6,7], i.e., a ten-fold cross validation. We repeat every experiment 30 times and report the average result. The optimal settings of parameters are obtained by grid search method. The number of fuzzy sets  $K$  for each feature is chosen from  $\{3, 5, 7\}$ , while the regularization parameter  $\lambda$  is chosen from  $\{10^{-3}, 10^{-2}, 10^{-1}\}$ , and the degree of Chebyshev polynomial  $n$  is chosen from  $\{1, 2, 3, 4\}$ . The predefined maximum number of rules  $L_{max}$  is set as 25, the population size  $T$  is set as 50 and the maximum number of iterations of the PSO  $n_{max}$  is set as 10.

##### 4.1. Regression

20 regression datasets which are selected from the UCI [39] and KEEL[40] repositories are employed to evaluate the performance of the SCFS-IFRs in regression tasks. The relevant details of the 20 datasets are summarized in Table 3.

**Table 3.** Details of 20 regression datasets from UCI and KEEL.

Dataset	No. of Features	No. of Data	Dataset	No. of Features	No. of Data
Airfoil	5	1503	Baseball	16	337
Concrete	8	1030	Izmir	9	1461
Auto MPG	7	392	Ankara	9	1609
Yacht	6	308	Electrical	4	1056
QSAR	6	908	Computer	21	8192
Treasury	15	1049	Laser	4	993
Abalone	8	4177	Prices	13	506
Categorical	7	4052	Cool	8	768
Auto MPG6	5	392	Wine	11	1599
Energy	6	365	Heat	8	786

Table 4 compares the testing RMSE of the proposed models with another eight algorithms: RVFLN [32], ELM [33], RVFL-RS [34], HPFNN [35], SCN [19], SCBLS [36], BLS [16] and FBLS [7]. The results of RVFLN, ELM, RVFL-RS and HPFNN are borrowed from [35], and the results of SCN, SCBLS, BLS, FBLS are obtained from the source code provided by the cited author. It should be noted that, in order to be consistent with the results in [35], only the input features are normalized in this experiment. In Table 4, the bold items represent the smallest testing RMSE. Our SCFS-IFRs shows the smallest testing RMSE in 10 of 20 datasets while HPFNN shows the smallest testing RMSE in five datasets. It can be seen that SCFS-IFRs have better generalization performance than the other eight algorithms.

**Table 4.** Comparisons of testing RMSE on 20 datasets.

Dataset	RVFLN	ELM	RVFL-RS	HPFNN	SCN	SCBLS	BLS	FBLS	SCFS-IFRs
Airfoil	3.0120	2.9575	3.0120	3.6479	5.2468	4.9016	5.1255	5.2006	<b>2.7497</b>
Concrete	6.5454	6.8064	6.7564	6.7854	7.2041	6.7368	7.1660	7.0735	<b>5.5644</b>
Auto MPG	3.0211	2.8833	2.9047	<b>2.5723</b>	3.1761	3.1633	3.1780	3.1271	2.6370
Yacht	3.4028	3.3144	4.7810	0.8783	9.3073	4.3551	6.9055	5.2876	<b>0.7443</b>
QSAR	0.9410	0.9469	0.9274	0.9166	0.8986	<b>0.8763</b>	0.8697	0.8798	0.9160
Treasury	0.2086	0.2020	0.1968	0.2305	0.2250	0.2149	0.2328	0.2272	0.1815
Abalone	2.1506	2.1263	2.1260	<b>2.0966</b>	2.2502	2.2228	2.1830	2.1872	2.0994
Categorical	0.3260	0.3865	0.2012	<b>0.0780</b>	0.1130	0.7924	1.0201	0.9321	0.0804
Auto MPG6	3.0642	2.9866	3.9217	2.6323	2.0850	2.1359	2.1748	<b>2.0788</b>	2.6878
Energy	0.3898	0.3980	0.3905	0.3883	0.3881	0.3791	0.3811	<b>0.3714</b>	0.3930
Baseball	723.70	833.72	717.55	<b>615.14</b>	711.36	678.40	666.68	642.84	693.55
Izmir	1.2032	1.3781	1.2186	1.1028	1.1402	1.1389	1.1426	1.1511	<b>1.0810</b>
Ankara	1.4364	1.7054	1.4122	1.2754	1.2453	1.2825	1.3104	1.2811	<b>1.1982</b>
Electrical	102.40	107.16	83.215	115.26	110.75	99.984	113.691	111.974	<b>80.311</b>
Computer	4.9992	5.5109	4.9039	2.9284	4.2178	3.4578	2.9228	4.6363	<b>2.4164</b>
Laser	6.5630	6.2108	<b>4.9542</b>	7.0169	6.3307	6.3058	6.2107	6.1545	5.8172
Prices	4.7295	4.1906	4.0259	3.6176	6.1395	4.7395	5.7109	5.7305	<b>3.0717</b>
Cool	2.0263	2.1077	1.6723	1.7730	3.0224	2.9430	2.8301	2.0390	<b>1.3136</b>
Wine	0.6440	0.6483	0.6448	<b>0.6290</b>	0.6600	0.6552	0.6571	0.6616	0.6294
Heat	1.1406	1.3795	<b>0.5085</b>	0.7662	2.3875	2.3985	2.4324	1.2139	0.5349

In order to observe the statistical differences between SCFS-IFRs and the other eight algorithms in the 20 datasets, the Friedman test and the Holm post-hoc test are applied. The result of the Friedman test is displayed in Table 5. We can see that the SCFS-IFRs has the top rank and the  $p$ -value is less than the significance level 0.1 which implies that there are significant differences in the ranking of generalization performance of these algorithms.

**Table 5.** Result of Friedman tests (significance level of 0.10).

Algorithm	SCFS-IFRs	HPFNN	RVFL-RS	SCBLS	RVFLN	FBLs	ELM	BLS	SCN
Rank	2.300	3.750	4.575	5.050	5.475	5.500	5.950	6.100	6.300
<i>p</i> -value					0.00001				

Then we compare SCFS-IFRs with other models using the Holm post-hoc test and the result is shown in Table 6. The null hypothesis  $H_0$  in the Holm test is that the ranking of the two comparative methods has no significant difference. The significance level  $\alpha$  is set as 0.1. If the adjusted *p*-value of the Holm test is less than the significance level  $\alpha$ , this means we should reject the null hypothesis  $H_0$  and there are remarkable differences between the two comparative methods. From the Holm test results in Table 6, we can see that the proposed SCFS-IFRs are significantly better than RVFLN, ELM, RVFL-RS, HPFNN, SCN, BLS, FBLs and SCBLS.

**Table 6.** Result of Holm post-hoc tests (significance level of 0.10).

Comparison	Statistic	Adjusted <i>p</i> -Value	Result
SCFS-IFRs vs SCN	4.6188	0.00003	$H_0$ is rejected
SCFS-IFRs vs BLS	4.3878	0.00008	$H_0$ is rejected
SCFS-IFRs vs ELM	4.2146	0.00015	$H_0$ is rejected
SCFS-IFRs vs FBLs	3.6950	0.0011	$H_0$ is rejected
SCFS-IFRs vs RVFLN	3.6661	0.0011	$H_0$ is rejected
SCFS-IFRs vs SCBLS	3.1754	0.00449	$H_0$ is rejected
SCFS-IFRs vs RVFL-RS	2.6269	0.01723	$H_0$ is rejected
SCFS-IFRs vs HPFNN	1.6743	0.09407	$H_0$ is rejected

#### 4.2. Classification

We select the following 10 datasets to evaluate the performance of the proposed SCFS-IFRs in classification tasks. The relevant details of the 10 classification datasets are summarized in Table 7.

**Table 7.** Details of 10 classification datasets from UCI and KEEL.

Dataset	No. of Data	No. of Features	No. of Classes
Diabetes	768	8	2
Australia	690	14	2
Balance	625	4	3
Magic04	19,020	10	2
Monk2	432	6	2
Pageblocks	5472	10	5
Sonar	208	60	2
Specfheart	267	44	2
Breast	699	9	2
Wine	178	13	3

The average classification accuracies for the proposed SCFS-IFRs and the other six fuzzy models, TSK-FC [37], L2TSK-FC [37], HID-TSK [6], CFBLs [7], FELM [4] and BL-DFIS [38], for the 10 classification datasets are reported in Table 8. The results of TSK-FC [37], L2TSK-FC, HID-TSK and CFBLs are borrowed from [7] and the results of FELM and BL-DFIS are borrowed from [38]. The bold items represent the highest testing classification accuracy. The comparisons of classification accuracy in Table 8 show that, SCFS-IFRs have the highest average classification accuracy. In terms of the classification accuracy of each data set, SCFS-IFRs have the highest classification accuracy in six of 10 data sets, while the accuracy of classification of SCFS-IFRs in the other four data sets is still comparable.

**Table 8.** Comparisons of classification accuracies on 10 datasets.

Dataset	TSK-FC	L2TSK-FC	HID-TSK	CFBLS	FELM	BL-DFIS	SCFS-IFRs
Diabetes	0.6687	0.6432	0.7182	<b>0.7764</b>	0.7512	0.7724	0.7738
Australia	0.7298	0.7768	0.8337	<b>0.8750</b>	0.8477	0.8657	0.8594
Balance	0.7079	0.7114	0.8416	0.9066	0.8720	0.9141	<b>0.9152</b>
Magic04	0.6487	0.7424	0.8160	0.8533	0.8260	0.8629	<b>0.8631</b>
Monk2	0.6332	0.6062	0.6599	0.9353	0.6598	0.7482	<b>0.9720</b>
Pageblocks	0.9025	0.8938	0.9067	0.9550	0.9355	0.9592	<b>0.9603</b>
Sonar	0.4930	0.6620	0.6585	0.7591	0.6895	0.8223	<b>0.8243</b>
Specfheart	0.7516	0.7450	0.7520	<b>0.8335</b>	0.7921	0.8130	0.7622
Breast	0.8846	0.9469	0.9512	<b>0.9750</b>	0.9669	0.9674	0.9691
Wine	0.9037	0.9183	0.9076	0.9740	0.9676	0.9878	0.9885
Average	0.7324	0.7646	0.8045	0.8843	0.8308	0.8713	<b>0.8888</b>

The incremental learning algorithm is another advantage of SCFS-IFRs. This helps to achieve a compact architecture while maintaining good performance. Table 9 displays the comparisons of the average number of fuzzy rules that all seven models need to use in order to achieve the accuracy shown in Table 8. The bold items represent the minimum number of rules. From Table 9, It can be observed that the proposed SCFS-IFRs can achieve good classification accuracy with few rules.

**Table 9.** Comparisons of average number of fuzzy rules on 10 datasets.

Dataset	TSK-FC	L2TSK-FC	HID-TSK	CFBLS	FELM	BL-DFIS	SCFS-IFRs
Diabetes	3.1	2.0	9.0	8.4	18	<b>1.1</b>	3.5
Australia	7.0	10.0	9.0	15.7	19	2.3	<b>1.7</b>
Balance	4.3	3.9	9.3	9.2	20	2.2	<b>1.2</b>
Magic04	13.4	10.8	36.3	18.9	20	14.3	<b>10.4</b>
Monk2	8.5	7.6	7.0	5.9	18	8.9	<b>3.8</b>
Pageblocks	7.2	<b>7.0</b>	16.2	18.9	17	12.6	12.0
Sonar	8.5	9.3	24.0	8.1	20	<b>2.2</b>	4.6
Specfheart	7.1	7.5	9.3	9.2	20	<b>1.0</b>	1.3
Breast	14.6	13.9	12.0	7.6	20	<b>3.3</b>	5.75
Wine	10.0	5.7	42.0	13.7	20	<b>1.2</b>	2.4
Average	8.37	7.77	17.41	11.56	19.2	4.91	4.67

Finally, similar to Section 4.2, we use the Friedman test and the Holm post-hoc test to observe the statistical differences between SCFS-IFRs and the other six algorithms on the 10 classification datasets. The result of the Friedman test is displayed in Table 10. We can see that the SCFS-IFR has the highest ranking and the  $p$ -value is  $1.339 \times 10^{-9} \ll 0.10$ . This implies that there are significant differences in the ranking of these algorithms.

**Table 10.** Result of Friedman tests (significance level of 0.10).

Algorithm	SCFS-IFRs	CFBLS	BL-DFIS	FELM	HID-TSK	L2TSK-FC	TSK-FC
Rank	2.300	3.750	4.575	5.050	5.475	5.500	5.950
$p$ -value	$1.339 \times 10^{-9}$						

The result of the Holm post-hoc test is listed in Table 11. The null hypothesis  $H_0$  in the Holm test is that the rankings of the two comparative methods have no significant difference. From Table 11, we can see that the proposed SCFS-IFRs is significantly better than TSK-FC, L2TSK-FC, HID-TSK and FELM, but there is no significant difference in the comparison results between SCFS-IFRs, BL-DFIS and CFBLS.

**Table 11.** Result of Holm post-hoc tests (significance level of 0.10).

Comparison	Statistic	Adjusted <i>p</i> -Value	Result
SCFS-IFRs vs TSK-FC	5.07198	0	H0 is rejected
SCFS-IFRs vs L2TSK-FC	4.65794	0.00002	H0 is rejected
SCFS-IFRs vs HID-TSK	3.51933	0.00173	H0 is rejected
SCFS-IFRs vs FELM	2.38073	0.05184	H0 is rejected
SCFS-IFRs vs BL-DFIS	0.62106	1	H0 is accepted
SCFS-IFRs vs CFBLs	0.41404	1	H0 is accepted

However, SCFS-IFRs has an advantage over BL-DFIS and CFBLs in terms of interpretability. First, it can be seen from Table 9 that the average number of rules of SCFS-IFRs is 4.67, far less than the average number of rules for CFBLs, 11.56. The fewer the number of rules, the better the interpretability. Then, BL-DFIS use mapped features by ELM-AE as the premise variable of fuzzy rules, which is not conducive to the interpretation of the rule.

#### 4.3. The Interpretability of SCFS-IFRs

Although a mathematical measure of interpretability has not been found, a fuzzy inference system with high interpretability should have clear semantic expression and simple fuzzy rules. We discuss the interpretability of SCFS-IFRs through one experimental result on the Monk2 task.

In this experiment, the input universe is divided into four equal parts, which implies five fuzzy subsets with five linguistic labels as shown in Section 3.1. The five fuzzy subsets have five fixed centers which are  $\{0, 0.25, 0.5, 0.75, 1\}$ . In this experiment, two fuzzy rules generated by the proposed SCFS-IFRs are presented in Table 12. The widths of fuzzy subsets and the rule combination matrix are randomly assigned under the supervisory mechanism (16) and optimized by the proposed adaptive incremental learning algorithm of SCFS-IFRs. The widths of the two fuzzy rules are 0.8139 and 0.4826. The rule combination matrixes of two fuzzy rules are:

$$C_1 = \begin{pmatrix} 0 & 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 1 & 1 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 1 \end{pmatrix}$$

and

$$C_2 = \begin{pmatrix} 1 & 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 \\ 1 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 \\ 1 & 1 & 0 & 0 & 0 \end{pmatrix}$$

where  $C_l(j, k) = 1$  represents the  $k$ th fuzzy set of the  $j$ th input variable in the  $l$ th fuzzy rule will be activated; conversely, it will not be activated.

The activated fuzzy sets and the polynomial consequent functions are listed in the two parts of Table 12, respectively. Since the highest degree of the used Chebyshev polynomial is set as 1, the polynomial consequent functions are linear functions.

In SCFS-IFRs, the fixed language labels of the antecedents make the intuitive interpretation of the antecedents possible, and the rule combination matrixes allow activation of more than one fuzzy set, corresponding to each feature in a fuzzy rule, which improves the semantic expression of fuzzy rules. This shortens the length of fuzzy rules while maintaining the performance of the model and avoids the “rule explosion” problem.

**Table 12.** Rule description for the Monk2 dataset.

	IF	THEN
$R_1$	$x_1$ is medium or high and $x_2$ is high and $x_3$ is very low and $x_4$ is low or medium or high and $x_5$ is low and $x_5$ is medium or very high	$y_1 = 0.761 + 0.011x_1 - 0.608x_2 - 0.038x_3 + 0.051x_4 - 0.026x_5 + 0.016x_6$ $y_2 = 0.219 + 0.020x_1 - 0.892x_2 - 0.102x_3 + 0.125x_4 - 0.015x_5 - 0.043x_6$
$R_2$	$x_1$ is very low or low and $x_2$ is very low and $x_3$ is high or very high and $x_4$ is very low or medium and $x_5$ is very high and $x_5$ is very low and low	$y_1 = 0.279 + 0.060x_1 + 0.082x_2 + 0.187x_3 + 0.174x_4 - 0.547x_5 - 0.134x_6$ $y_2 = -0.024 - 0.067x_1 - 0.342x_2 - 0.164x_3 - 0.079x_4 + 0.774x_5 + 0.260x_6$

## 5. Conclusions

By incorporating stochastic configuration and ILFRs into a TSK fuzzy system, a novel neuro-fuzzy system named SCFS-IFRs is established for classification and regression data modeling. The SCFS-IFRs uses a stochastic configuration based incremental learning approach to determine the system architecture and the random parameters. Interpretable fuzzy rules are incrementally generated and the premise parameters are randomly selected based on stochastic configuration strategy. In addition, the PSO algorithm, an intelligence search technique, is used in the process of stochastic configuration to obtain better random parameters and further improve the approximation accuracy.

Hence, the proposed SCFS-IFRs improves the interpretability and approximation accuracy of the neuro-fuzzy system at the same time. We tested the model's performance in both the regression task and the classification task. In the regression experiments, the results of 20 regression datasets which are selected from UCI [39] and KEEL [40] repositories show that the proposed SCFS-IFRs performs best on 10 of the 20 data sets while statistically significantly outperforming the other eight state-of-the-art algorithms. In the classification experiments, SCFS-IFRs achieves higher classification accuracy and better interpretation with fewer rules, compared with the other six fuzzy classifiers. In the future, a broad learning system can be considered to combine with SCFS-IFRs in order to improve approximation performance and decrease training time. In addition, it would be an interesting problem to analyze the approximation bound of SCFS-IFRs theoretically.

**Author Contributions:** Conceptualization, W.Z. and H.L.; methodology, W.Z.; software, W.Z.; validation, W.Z. and H.L. and M.B.; formal analysis, W.Z.; investigation, W.Z.; resources, M.B.; data curation, W.Z.; writing—original draft preparation, W.Z.; writing—review and editing, H.L.; visualization, W.Z.; supervision, W.Z.; project administration, W.Z.; funding acquisition, H.L. All authors have read and agreed to the published version of the manuscript.

**Funding:** This research was funded by the National Natural Science Foundation (NNSF) of China under Grant (61773088, 12071056).

**Institutional Review Board Statement:** Not applicable.

**Informed Consent Statement:** Not applicable.

**Data Availability Statement:** Not applicable.

**Conflicts of Interest:** The authors declare no conflict of interest.

## References

1. Jang, J.-S.R. ANFIS: Adaptive-Network-Based Fuzzy Inference System. *IEEE Trans. Syst. Man Cybern.* **1993**, *23*, 665–685. [CrossRef]
2. Kar, S.; Das, S.; Ghosh, P.K. Applications of neuro fuzzy systems: A brief review and future outline. *Appl. Soft Comput.* **2015**, *15*, 243–259. [CrossRef]
3. Pedrycz, W.; Chen, S.-M. (Eds.) *Interpretable Artificial Intelligence: A Perspective of Granular Computing*; Springer: Basingstoke, UK, 2021.
4. Wong, S.Y.; Yap, K.S.; Yap, H.J.; Tan, S.C.; Chang, S.W. On Equivalence of FIS and ELM for Interpretable Rule-Based Knowledge Representation. *IEEE Trans. Neural Netw. Learn. Syst.* **2015**, *26*, 1417–1430. [CrossRef] [PubMed]



5. Zhou, T.; Ishibuchi, H.; Wang, S. Stacked Blockwise Combination of Interpretable TSK Fuzzy Classifiers by Negative Correlation Learning. *IEEE Trans. Fuzzy Syst.* **2018**, *26*, 3327–3341. [\[CrossRef\]](#)
6. Zhang, Y.; Ishibuchi, H.; Wang, S. Deep Takagi–Sugeno–Kang Fuzzy Classifier with Shared Linguistic Fuzzy Rules. *IEEE Trans. Fuzzy Syst.* **2018**, *26*, 1535–1549. [\[CrossRef\]](#)
7. Feng, S.; Chen, C.L.P.; Xu, L.; Liu, Z. On the Accuracy–Complexity Tradeoff of Fuzzy Broad Learning System. *IEEE Trans. Fuzzy Syst.* **2021**, *29*, 2963–2974. [\[CrossRef\]](#)
8. Siminski, K. Prototype based granular neuro-fuzzy system for regression task. *Fuzzy Sets Syst.* **2022**, *449*, 5. [\[CrossRef\]](#)
9. Feng, S.; Chen, C.L.P.; Zhang, C.-Y. A Fuzzy Deep Model Based on Fuzzy Restricted Boltzmann Machines for High-dimensional Data Classification. *IEEE Trans. Fuzzy Syst.* **2020**, *28*, 1344–1355. [\[CrossRef\]](#)
10. Sadjadi, E.N.; Herrero, J.G.; Molina, J.M.; Moghaddam, Z.H. On Approximation Properties of Smooth Fuzzy Models. *Int. J. Fuzzy Syst.* **2018**, *20*, 2657–2667. [\[CrossRef\]](#)
11. Mahmoud, T.A.; Abdo, M.I.; Elsheikh, E.A.; Elshenawy, L.M. Direct adaptive control for nonlinear systems using a TSK fuzzy echo state network based on fractional-order learning algorithm. *J. Frankl. Inst.* **2021**, *358*, 9034–9060. [\[CrossRef\]](#)
12. Rajab, S.; Sharma, V. A review on the applications of neuro-fuzzy systems in business. *Artif. Intell. Rev.* **2018**, *49*, 481–510. [\[CrossRef\]](#)
13. Gaxiola, F.; Melin, P.; Valdez, F.; Castillo, O. Interval type-2 fuzzy weight adjustment for backpropagation neural networks with application in time series prediction. *Inf. Sci.* **2014**, *260*, 1–14. [\[CrossRef\]](#)
14. Wang, D.-G.; Song, W.-Y.; Li, H.-X. Approximation properties of ELM-fuzzy systems for smooth functions and their derivatives. *Neurocomputing* **2015**, *149*, 265–274. [\[CrossRef\]](#)
15. Kv, S.; Pillai, G. Regularized extreme learning adaptive neuro-fuzzy algorithm for regression and classification. *Knowl.-Based Syst.* **2017**, *127*, 100–113. [\[CrossRef\]](#)
16. Chen, C.L.P.; Liu, Z. Broad Learning System: An Effective and Efficient Incremental Learning System Without the Need for Deep Architecture. *IEEE Trans. Neural Netw. Learn. Syst.* **2018**, *29*, 10–24. [\[CrossRef\]](#)
17. Feng, S.; Chen, C.L.P. Fuzzy Broad Learning System: A Novel Neuro-Fuzzy Model for Regression and Classification. *IEEE Trans. Cybern.* **2020**, *50*, 414–424. [\[CrossRef\]](#)
18. Li, M.; Wang, D. Insights into randomized algorithms for neural networks: Practical issues and common pitfalls. *Inf. Sci.* **2017**, *382–383*, 170–178. [\[CrossRef\]](#)
19. Wang, D.; Li, M. Stochastic Configuration Networks: Fundamentals and Algorithms. *IEEE Trans. Cybern.* **2017**, *47*, 3466–3479. [\[CrossRef\]](#)
20. Wang, D.; Li, M. Deep Stochastic Configuration Networks with Universal Approximation Property. In Proceedings of the 2018 International Joint Conference on Neural Networks (IJCNN), Rio de Janeiro, Brazil, 8–13 July 2018; pp. 1–8. [\[CrossRef\]](#)
21. Pratama, M.; Wang, D. Deep stacked stochastic configuration networks for lifelong learning of non-stationary data streams. *Inf. Sci.* **2018**, *495*, 150–174. [\[CrossRef\]](#)
22. Wang, Q.; Dai, W.; Ma, X.; Shang, Z. Driving amount based stochastic configuration network for industrial process modeling. *Neurocomputing* **2020**, *394*, 61–69. [\[CrossRef\]](#)
23. Sugeno, M.; Kang, G. Structure identification of fuzzy model. *Fuzzy Sets Syst.* **1988**, *28*, 15–33. [\[CrossRef\]](#)
24. Ojha, V.; Abraham, A.; Snášel, V. Heuristic design of fuzzy inference systems: A review of three decades of research. *Eng. Appl. Artif. Intell.* **2019**, *85*, 845–864. [\[CrossRef\]](#)
25. Pratama, M.; Lu, J.; Anavatti, S.; Lughofer, E.; Lim, C.-P. An incremental meta-cognitive-based scaffolding fuzzy neural network. *Neurocomputing* **2016**, *171*, 89–105. [\[CrossRef\]](#)
26. Wang, S.; Chung, F.; HongBin, S.; Dewen, H. Cascaded centralized TSK fuzzy system: Universal approximator and high interpretation. *Appl. Soft Comput.* **2005**, *5*, 131–145. [\[CrossRef\]](#)
27. Zhou, T.; Chung, F.-L.; Wang, S. Deep TSK Fuzzy Classifier with Stacked Generalization and Triplely Concise Interpretability Guarantee for Large Data. *IEEE Trans. Fuzzy Syst.* **2017**, *25*, 1207–1221. [\[CrossRef\]](#)
28. Buckley, J. Sugeno type controllers are universal controllers. *Fuzzy Sets Syst.* **1993**, *53*, 299–303. [\[CrossRef\]](#)
29. Zou, W.; Xia, Y.; Dai, L. Fuzzy Broad Learning System Based on Accelerating Amount. *IEEE Trans. Fuzzy Syst.* **2022**, *30*, 4017–4024. [\[CrossRef\]](#)
30. Yousefi, J. A modified NEFCLASS classifier with enhanced accuracy-interpretability trade-off for datasets with skewed feature values. *Fuzzy Sets Syst.* **2021**, *413*, 99–113. [\[CrossRef\]](#)
31. Han, H.-G.; Sun, C.; Wu, X.; Yang, H.; Qiao, J. Training Fuzzy Neural Network via Multi-Objective Optimization for Non-linear Systems Identification. *IEEE Trans. Fuzzy Syst.* **2022**, *30*, 3574–3588. [\[CrossRef\]](#)
32. Pao, Y.-H.; Takefuji, Y. Functional-link net computing: Theory, system architecture, and functionalities. *Computer* **1992**, *25*, 76–79. [\[CrossRef\]](#)
33. Huang, G.-B.; Zhou, H.; Ding, X.; Zhang, R. Extreme Learning Machine for Regression and Multiclass Classification. *IEEE Trans. Syst. Man Cybern. Part B Cybern.* **2012**, *42*, 513–529. [\[CrossRef\]](#) [\[PubMed\]](#)
34. Vuković, N.; Petrović, M.; Miljković, Z. A comprehensive experimental evaluation of orthogonal polynomial expanded random vector functional link neural networks for regression. *Appl. Soft Comput.* **2018**, *70*, 1083–1096. [\[CrossRef\]](#)
35. Zhang, C.; Oh, S.-K.; Fu, Z. Hierarchical polynomial-based fuzzy neural networks driven with the aid of hybrid network architecture and ranking-based neuron selection strategies. *Appl. Soft Comput.* **2021**, *113*, 107865. [\[CrossRef\]](#)



36. Zhou, W.; Wang, D.; Li, H.; Bao, M. Stochastic configuration broad learning system and its approximation capability analysis. *Int. J. Mach. Learn. Cybern.* **2022**, *13*, 797–810. [[CrossRef](#)]
37. Deng, Z.; Choi, K.-S.; Chung, F.-L.; Wang, S. Scalable TSK Fuzzy Modeling for Very Large Datasets Using Mini-mal-Enclosing-Ball Approximation. *IEEE Trans. Fuzzy Syst.* **2011**, *19*, 210–226. [[CrossRef](#)]
38. Bai, K.; Zhu, X.; Wen, S.; Zhang, R.; Zhang, W. Broad Learning Based Dynamic Fuzzy Inference System with Adaptive Structure and Interpretable Fuzzy Rules. *IEEE Trans. Fuzzy Syst.* **2022**, *30*, 3270–3283. [[CrossRef](#)]
39. Dua, D.; Graff, C. UCI Machine Learning Repository. University of California, Irvine, School of Information and Computer Sciences. 2017. Available online: <http://archive.ics.uci.edu/ml> (accessed on 26 February 2022).
40. Triguero, I.; Gonzalez, S.; Moyano, J.M.; Garcia, S.; Fernandez, A. KEEL 3.0: An Open Source Software for Multi-Stage Analysis in Data Mining. *Int. J. Comput. Intell. Syst.* **2017**, *10*, 1238–1249. [[CrossRef](#)]

**Disclaimer/Publisher’s Note:** The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.