applied sciences

MDPI

Article

Adaptive Neuro-Fuzzy Inference System Predictor with an Incremental Tree Structure Based on a Context-Based Fuzzy Clustering Approach

Chan-Uk Yeom and Keun-Chang Kwak *

Department of Control and Instrumentation Engineering, Chosun University, Gwangju 61452, Korea; walt18@naver.com

* Correspondence: kwak@chosun.ac.kr; Tel.: +82-62-230-6086

Received: 08 November 2020; Accepted: 26 November 2020; Published: 27 November 2020

Abstract: We propose an adaptive neuro-fuzzy inference system (ANFIS) with an incremental tree structure based on a context-based fuzzy C-means (CFCM) clustering process. ANFIS is a combination of a neural network with the ability to learn, adapt and compute, and a fuzzy machine with the ability to think and to reason. It has the advantages of both models. General ANFIS rule generation methods include a method employing a grid division using a membership function and a clustering method. In this study, a rule is created using CFCM clustering that considers the pattern of the output space. In addition, multiple ANFISs were designed in an incremental tree structure without using a single ANFIS. To evaluate the performance of ANFIS in an incremental tree structure based on the CFCM clustering method, a computer performance prediction experiment was conducted using a building heating-and-cooling dataset. The prediction experiment verified that the proposed CFCM-clustering-based ANFIS shows better prediction efficiency than the current grid-based and clustering-based ANFISs in the form of an incremental tree.

Keywords: adaptive neuro-fuzzy inference system (ANFIS); context-based fuzzy C-means (CFCM) clustering; fuzzy trees; incremental fuzzy inference system

1. Introduction

In optimizing the nonlinear system model, the neuro-fuzzy system has exhibited better performance than the model based on the existing linear system [1–11]. In the case of a neuro-fuzzy system that simulates human learning ability, decision judgment, etc., rather than a mathematical calculation technique, the performance of the model may vary depending on the type of learning model or learning method.

A grid-based rule generation approach and a clustering-based rule generation method can be separated into the adaptive neuro-fuzzy inference system (AFNIS) model rule generation technique. Studies on the grid-based rule generation method include the following: Dovzan [12] proposed a hyperplane-based fuzzy space partitioning method by defining the superplane dividing the problem space and introducing principal component analysis, in which the distance to the superplane is used as a metric instead of the center-oriented cluster. In order to automatically design interpretable fuzzy partitions with maximal granularity, Castiello [13] suggested a dual clustering (DC) method. DC is advanced and works in a two-step phase for classification problems. The first step identifies a cluster of multidimensional samples to derive a prototype with class labels. In the second step, these prototype one-dimensional projections are further clustered at the same time along each dimension, minimizing the number of clusters for each function. Alexandridis [14] proposed a new algorithm to train radial base function (RBF) networks to produce models with increased accuracy and brevity.

The proposed approach is based on the asymmetric deformation of the algorithm of the fuzzy means (FM) with the potential to calculate the number and position of the centers of the silver-winged node RBF, while linear regression is used for the synaptic weights. Verstraete [15] proposed a new approach to remap grid data using the additional data provided so that the system can automatically assume the underlying distribution. The proposed method uses correlation data to imitate intelligent reasoning to provide insight into the distribution of the original data. In the grid-based rule generation method, when the dimension of the input increases or the number of membership functions (MFs) increases, the rule of the neuro-fuzzy system model increases exponentially. Various studies have been undertaken to solve these problems. A typical example is a clustering method in which a given input space and an output space are divided into subspaces, each having a meaning to give a preamble MF.

Studies on creating a rule using a clustering method include the following: Lee [16] introduced an enhanced Mobile Sensor Network (MSN) Low-Energy Adaptive Clustering Layer Protocol to not only prolong the life of the network but also reduce the failure of the package using the fuzzy inference method. Su [17] proposed a belief-peak-based clustering method as an idea, with evidence that all data objects in each sample subsection led to a belief in the possibility that the sample would become a cluster center. Xu [18] proposed a concise zero-order Sugeno–Takagi (TSK) inference system based on enhanced soft subspace clustering (ESSC) and sparse leading (SL) to improve the clarity and interpretability of fuzzy reasoning systems. Sujil [19] proposed wind power generation prediction agents for multiple-agent-based energy management systems in smart microgrids using subtraction clustering and fuzzy clustering methods. A fuzzy-based hyper-round strategy (FHRP) was introduced by Neamatollani [20] to plan clustering operations easily and flexibly. The FHRP performs clustering at the start of all hyper-rounds (HRs) consisting of several rounds other than each round, and the length of the HR is not fixed during the network life and is calculated using the fuzzy reasoning system. To improve the classification and rule-based analytical performance for unbalanced datasets, Gu [21] proposed an imbalanced TSK purge classifier (IB-TSK-FC) for TSK fuzzy classifiers. A hierarchical fuzzy inference tree (HFIT) was constructed by Ojha [22]. In order to construct a natural hierarchy that supports simplicity, HFIT incorporates many low-dimensional fuzzy logic structures with a structure close to the ideal tree. This natural hierarchy provides a high level of approximation accuracy. The clustering-based rule generation method belongs to a cluster that satisfies a given condition by measuring the degree of similarity with each pattern, under the assumption that there are multiple patterns in one nonlinear data space.

Because the information used to create rules has uncertainty, the MFs of the conditional and conclusion parts of the corresponding rules have uncertainty. Studies have been performed to adjust the form of the MF to minimize this instability. Shi [23] proposed the fountain differential proportional–integral–derivative (PID) and fountain differential type 1 purge PID controller to solve this problem because the fountain differential gap type 2 purge PID controller cannot handle the uncertainty of the system. In describing the system's instability dependent on the general type 2 fuzzy logic system, the proposed controller will thoroughly exploit the benefits of the general type 2 fuzzy logic system. The definition of conditional fuzzy sets was suggested by Wang [24] and proved that type 2 fuzzy sets are united with conditional fuzzy sets. Both the conditional fuzzy set and the fuzzy form 2 set are fuzzy relationships for the primary and secondary variables' product space. The distinction is that the primary and secondary variables are usually independent of each one in the conditional fuzzy set system. To resolve the effects of human-made artifacts and a self-regulating interval type 2 neural purge inference system (SRIT2NFIS) to deal with these intrinsic anomalies, Das [25] proposed a powerful general spatial pattern characteristic pursuit algorithm (RoCSP). Das [26] indicated an emerging neural purge inference method (IT2FIS) gap type 2 and its total sequential learning algorithm. Meta-aware learning manages the learning process by choosing the best learning strategy and lets the recommended IT2FIS efficiently estimate the relationship between input and output. The evolving IT2FIS using meta-cognitive learning algorithms is called McTI2FIS. Zhou [27] conducted a study on how footprint of uncertainty (FOU) affects the analysis structure of a wide range of IT2 Mamdani and TSK controllers (i.e., input–output mathematical relationships). A recent application of the hybrid learning approach to the optimization of membership and non-membership functions of the newly developed Type 2 Interval Intuitive Fuzzy Logic Method (IT2IFLS) of the TSK Fuzzy Rationing System using neural networks was introduced by Eyoh [28]. Sumati [29] proposed the gap type 2 mutual subset purge neural inference system (IT2MSFuNIS). A reciprocal subset measurement between the two gap type 2 fuzzy sets is derived and used to determine the similarity between IT2FS inputs and sex items. Biglarbegian [30] proposed a new reasoning mechanism for the interval type 2 TSK fuzzy logic control system (IT2 TSK FLCS) when the condition is a type 2 fuzzy set, and the conclusion is a constant. Gracia [31] proposed a complete framework for type 2 FLS that uses up-to-date perceptions of IT2 FS (a set of gap type 2 fuzzy sets in a typical subsubsidiary form) in which secondary ratings could be nonconvex T1 FS.

As a result of confirming the studies of the ANFIS models summarized above, the existing study focuses on the model rule generation method. In this research, we propose a context-based fuzzy C-means (CFCM) clustering-based rule generation approach instead of a general clustering-based rule generation methodology that takes into account the patterns of the input space as well as the output space and ANFIS in the form of an incremental tree structure rather than a single structure. Whereas general clustering methods only take the input space into account, the CFCM clustering approach often takes the output space pattern into account, so that the cluster can be generated more accurately. There are many inputs when using big data in numerous application fields. In the neuro-fuzzy system, as the number of inputs increases, the number of rules increases exponentially. Therefore, it creates meaningful rules by designing a point-of-point tree structure using multiple ANFISs rather than a single ANFIS structure. To evaluate the performance of ANFIS in an incremental tree structure based on the CFCM clustering method, a computer performance prediction experiment was conducted using a building heating-and-cooling dataset [32]. The building heating-and-cooling dataset is a dataset used for energy efficiency forecasting created by Xifara. It consists of eight input variables and two output variables and has a data size of 768 × 10.

The remainder of this paper is structured as follows. Section 1 explains the background of the study. Section 2 describes the method and structure of ANFIS rule creation. In Section 3, the proposed method, ANFIS, with an incremental tree structure based on the CFCM clustering method, is described. Section 4 analyzes the predictive performance of the proposed method, and Section 5 addresses conclusions and future research plans.

2. ANFIS

Fuzzy inference has the characteristic of effectively explaining the system by organizing professional empirical knowledge that is difficult to quantitatively express in the form of MFs and fuzzy rule bases. [33]. In addition, because neural networks [34] have learning ability, they are highly flexible in the configuration of the system, and they have excellent parallel processing and fault tolerance capabilities. Neuro-fuzzy system neural network theories are actively studied in various fields.

A typical example of this neuro-fuzzy system is ANFIS. The premise of ANFIS depends on how the rule is created. The structure of the conclusion section consists of the form of the first equation stone, the TSK [35] model. Section 2.1 describes how to create rules to determine the premise.

2.1. Rule Creation Method

You divide all dimensions of the input space consisting of input variables into separate areas when you deduce a fuzzy law and organize them into segmentation and conquest methods that allow the resulting values of the inference in those areas to be determined. In other words, the premise of a fuzzy rule splits the input space into several regions, and the product of inference from each of those areas is the conclusion of the fuzzy rule. The creation of these fuzzy rules is closely connected to how the input space is separated. ANFIS has a method of grid-based rule development and a method of clustering-based rule creation, mainly based on generating rules consisting of input variables at all stages. Figure 1 shows how to create grid-based rules and how to create clustering-based rules based on ANFIS.

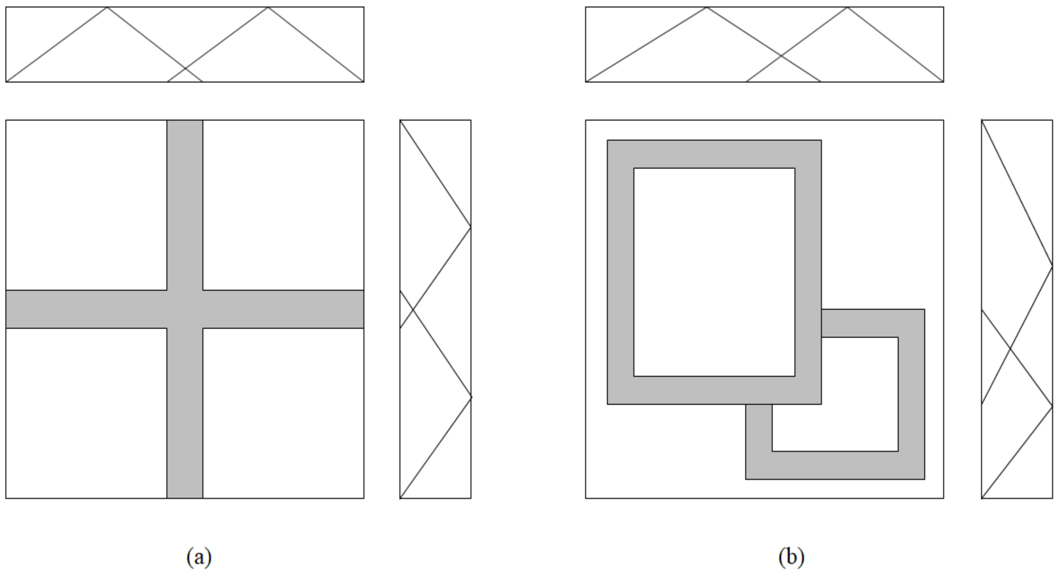


Figure 1. Adaptive neuro-fuzzy inference system (ANFIS) rule creation methods: (**a**) grid-based rule creation method and (**b**) clustering-based rule creation method.

Grid partitioning [36] is a method of dividing space into the same structure as the grid so that there is no overlap in the input space. Generally, the application of the grid partitioning method produces uniformly specific partitioning areas, that is, areas with fuzzy rules, which facilitate the analysis of fuzzy rules. When the number of input variables is minimal, that is, when the input space dimension is low, grid partitioning is used. If there are 10 input variables, for example, each input variable is split into two member functions, or $2^7 = 128$ specific areas. In other terms, for each particular region, one rule is made, and the total number of rules is 128, which is a very complex structure. Therefore, where the number of input variables is limited, the grid partitioning approach is mainly used.

By improving the C-means clustering approach suggested by Bezdek [37,38], the fuzzy C-means (FCM) clustering method is based on a fuzzy set and the least square method. By listing the values belonging to the data in a cluster according to the degree of belonging of each data object in a cluster, the FCM clustering system distinguishes particular subdivided regions. The methods of FCM clustering include the m vector $x_i, i = 1, 2, \ldots, m,$ set in c fuzzy clusters and locate the center in each cluster as it minimizes the objective function of the non-similar calculation. In standard methods of clustering, every data point belongs to a cluster with a membership of 0 or 1. However, there is a gap in the degree of membership of the arbitrary date between 0 and 1 in the FCM clustering process, and it belongs to n clusters. The number of clusters is fixed by the user. The number of clusters here is the number of laws that are fuzzy. Next, we explain the procedure for FCM clustering methods.

Step 1: To have some value between 0 and 1 that satisfies the parameter and membership matrix, initialize

$$u_{ij} = \left[\sum_{k=1}^{cx} \frac{\left\|x_j - v_i\right\|^{\frac{2}{m-1}}}{\left\|x_j - v_i\right\|}\right]^{-1} \quad (1)$$

here, the Euclidean standard is used to measure the distance between the input data and the middle of the cluster:

$$d_{ik} = d(x_k - v_i) = \left[\sum_{j=1}^{n} \left(x_{ki} - v_{ij}\right)^2\right]^{\frac{1}{2}} \quad (2)$$

Step 2: The current cluster's center value is determined by the input data value $E = \{e_1, e_2, \ldots, e_k\}$ and the previously acquired MF u_{ik}:

$$v_{ik} = \frac{\sum_{k=1}^{n}(u_{ik})^m x_{kj}}{\sum_{k=1}^{n}(u_{ik})^m} \tag{3}$$

Step 3: The membership matrix u_{ik} is continuously modified with increasing numbers of repetitions using the center value v_{ij} and input data E obtained in step 2, r:

$$u_{ik}^{(r+1)} = \frac{1}{\sum_{j=1}^{c}\left[\frac{d_{ik}^{r}}{d_{jk}^{r}}\right]^{2/m-1}} \tag{4}$$

Step 4: The above procedure is repeated until the repeated membership matrix U^r and U^{r+1} error is less than any threshold value given by the membership matrix U^r or U^{r+1}

$$\Delta = \|U^{r+1} - U^r\| = \max_{ik}|u_{ik}^{r+1} - u_{ik}^{r}| \tag{5}$$

2.2. Structure

A type of neuro-fuzzy inference method proposed by Jang [35] is the ANFIS model. For given input and output data, the ANFIS model utilizes the least square method and back propagation algorithms to optimally approximate the parameters used in the MF and output. A model consisting of two inputs and n TSK rules with one output and an output from the first linear equation defines the fuzzy inference mechanism briefly:

$$\begin{gathered}\text{Rule}^1: \text{If } \mathrm{X}_1 \text{ is } A_1 \text{ and } X_2 \text{ is } B_1, \text{then } y = k_{10} + k_{11}X_1 + k_{12}X_2 \\ \ldots \\ \text{Rule}^n: \text{If } \mathrm{X}_1 \text{ is } A_1 \text{ and } X_2 \text{ is } B_1, \text{then } y = k_{n0} + k_{n1}X_1 + k_{n2}X_2\end{gathered} \tag{6}$$

here, X_1 and X_2 represent input variables, and A_1 and A_2 are fuzzy sets of X_1. Similarly, B_1 and B_2 represent fuzzy sets of X_2, and k_{i0}, k_{i1} and k_{i2} represent sets of arguments set in rule i. The ANFIS model, a forward network structure which consists of two input variables and five levels with four fuzzy rules, is shown in Figure 2. Nodes have multiple functions on each layer of the ANFIS model that are refined through the learning process. The line of relation between two nodes indicates only the flow path between the nodes and has no weight. Next, for the ANFIS model, we define any layer structure and process.

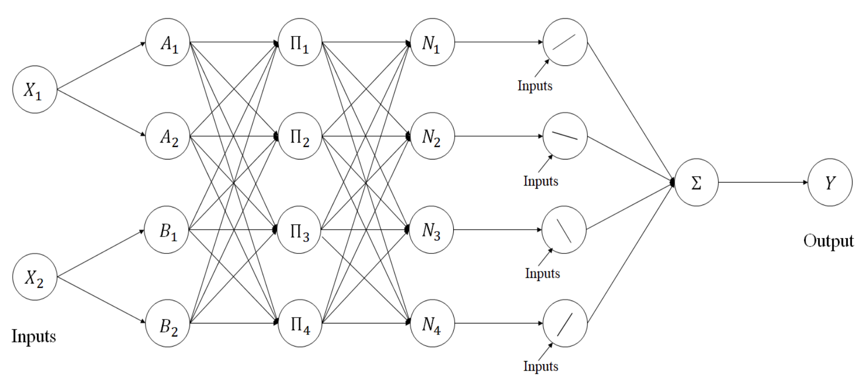


Figure 2. ANFIS structure.

Layer 1: Every node in the first layer is able to output values belonging to the language level in the first layer:

$$O_i^1 = u_{A_i}(x), O_{i+1}^1 = u_{b_i}(y), i = 1, 2 \tag{7}$$

The MF selects and uses the following Gaussian MF:

$$u_{A_i}(x) = \exp\left\{-\left(\frac{x-c_i}{a_i}\right)^2\right\} \tag{8}$$

In addition to the Gaussian MF, a variety of MFs are available, and the learning process selects parameter values that minimize errors.

Layer 2: Every node in the second layer receives a membership value seen in the conditional part of the fuzzy rule in the second layer and outputs it as a weight multiplied by the rule:

$$O_i^2 = w_i = u_{A_i}(x) \times u_{B_i}(y), i = 1, 2 \tag{9}$$

The output on each node shows that the fuzzy rule is sufficient.

Layer 3: In the third layer, each node calculates the ratio of point firepower in rule i to the sum of all point fire forces using:

$$O_i^3 = \overline{w_i} = \frac{w_i}{w_1 + w_2}, i = 1, 2 \tag{10}$$

The values obtained are displayed as normalized values.

Layer 4: Every node in the fourth layer conducts an operation in the fourth layer that multiplies the output function of the conclusion component of each law by the uniform fit:

$$O_i^4 = \overline{w_i} f_i = \overline{w_i}(p_i x + q_i y + r_i), i = 1, 2 \tag{11}$$

where w_i is the Layer 3 output and the p_i, q_i and r_i output function parameters denote the parameters of the conclusion.

Layer 5: Each node consists of one single node in the fifth and last layer. The output value is determined on the basis of all input values in the lower layer by using:

$$O_i^5 = y_i^* = \sum_{i=1}^{2} \overline{w_i} f_i = \frac{\sum w_i f_i}{\sum w_i} \tag{12}$$

The output value has a continuous type value, not a fuzzy set type.

3. ANFIS with an Incremental Tree Structure Based on the CFCM Clustering Method

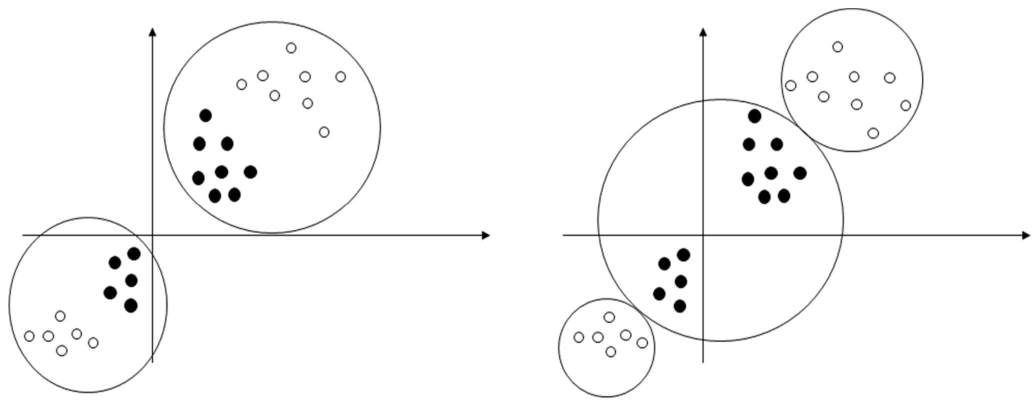
The number of laws increases exponentially as the number of inputs to the fuzzy system increases. The computational utility of the fuzzy system is decreased by this large rule base. It also makes the function of the fuzzy system difficult to understand and complicates the modification of rules and MF parameters. Since many implementations have a small supply of training data, the possibility of generalization of tuned fuzzy structures is diminished by a broad rule base.

The fuzzy inference system (FIS) can be implemented as a tree with smaller interconnected FIS objects to solve this challenge, not as a single monolithic FIS entity. This fuzzy trees are also called hierarchical enemy fuzzy structures [39] since fuzzy systems are organized in a hierarchical tree structure. The output of a low-level fuzzy system is used as an input to a high-level fuzzy system in the tree structure. The fuzzy tree is more effective and easier to grasp in terms of computing than a single FIS with the same number of entries.

3.1. CFCM-Clustering-Based Rule Creation Method

CFCM clustering is a tool proposed by Pedrycz [40] to construct clusters and partition clusters in order to maintain pattern characteristics related to output variable similarities, as well as input space data. In the output variables, a traditional clustering approach does not take patterns into account but generates only clusters using the Euclidean distance between the centroid cluster and the input data. In comparison, by taking into account not just the pattern of input data but also the pattern of output variables, the CFCM clustering approach generates a cluster, facilitating more detailed space segmentation than the conventional clustering method.

The variations between FCM clustering and the strategies of CFCM clustering are seen in Figure 3. The FCM clustering approach produces two clusters if there is data in the input space, so it gives an initial centroid value and then uses the Euclidean distance between the middle and the data. The CFCM clustering process, by comparison, takes into account the output variable patterns and generates three clusters by taking into account the black and white characteristics of the data in the input space. Next, the CFCM clustering process procedures are defined.



(**a**) (**b**)

Figure 3. Comparison of clusters between fuzzy C-means (FCM) and context-based FCM clustering methods: (**a**) FCM clustering method and (**b**) Context-based fuzzy C-means (CFCM) clustering method.

Step 1: Let m $(1 < m < \infty)$ and set the number of clusters, c $(2 \le c \le n)$.

Step 2: Set the initial partition matrix U and the threshold value ε, and select the number of repetitions:

$$U\left(\left[u_{ij}\right] i = 1, \ldots, c, j = 1, \ldots, n\right) \tag{13}$$

Step 3: Compute the center of each cluster, c_i $(i = 1, 2, \ldots, c)$, using the membership matrix U:

$$c_i = \frac{\sum_{j=1}^{n} u_{ij}^{m} x_j}{\sum_{j=1}^{n} u_{ij}^{m}} \tag{14}$$

Step 4: The partitioning matrix U is modified with the center value of cluster c:

$$u_{ij} = \frac{f_j}{\sum_{k=1}^{c} \left(\frac{d_{ij}}{d_{kj}}\right)^{2/(m-1)}} \tag{15}$$

here, f_j represents x_j's degree of inclusion in the created cluster. The linguistic type specified in the output variable is, in other words, represented as a fuzzy set $A, \{A: B \to [0, 1]\}$ and computed by an algorithm of fuzzy equalization. Then, the membership value of y_j in A can be expressed by $f_j = A\left(y_j\right), i = 1, 2, \ldots, n$.

Step 5: If $\|J^r - J^{r+1}\| \le$ ε is met, where

$$J = \sum_{j=1}^{n} \sum_{i=1}^{c} u_{ij}^{m} \left\|x_j - c_i\right\|^2, \tag{16}$$

the procedure above will be stopped. Otherwise, proceed again from Step 3.

For ANFIS models, the methods of CFCM clustering mentioned above apply as follows: Input space data in Layer 1 is broken into input space by CFCM clustering, which outputs the value by considering the output variable pattern. In Layer 2, the values belonging to the previous layer are taken, the weights multiplied by the rules are given, and the impulse force proportion in Layer 3 is expressed as a normalized value. The normalized Layer 4 values are multiplied by the final output function and the final output is determined using Layer 5's weighted average.

3.2. ANFIS with an Incremental Tree Structure

For applications, several fuzzy tree structures are available. The input values are integrated into multiple stages in the incremental tree system used in this analysis to optimize the output values at various steps. The previous diagram, for example shows a three-level incremental fuzzy tree with a FIS_i^n fuzzy inference method, where i represents the FIS index of the nth level. At each step, there is only one fuzzy inference method in an incremental fuzzy tree; $i = 1$, that is. The jth input of the ith FIS at level n is indicated in the previous figure by input x_{ij}^n, where the kth output of the ith FIS at level n is indicated by the x_{ik}^n input. $n = 3$, $j = 1$ or 2 and $k = 1$ in the figure. Each FIS has a complete m^2 rule set if each input has m MFs. The total number of laws, then, is $nm^2 = 3 \times 3^2 = 27$. The monolithic ($n = 1$) FIS is seen in Figure 4 with four inputs ($j = 1, 2, 3, 4$) and three MFs ($m = 3$).

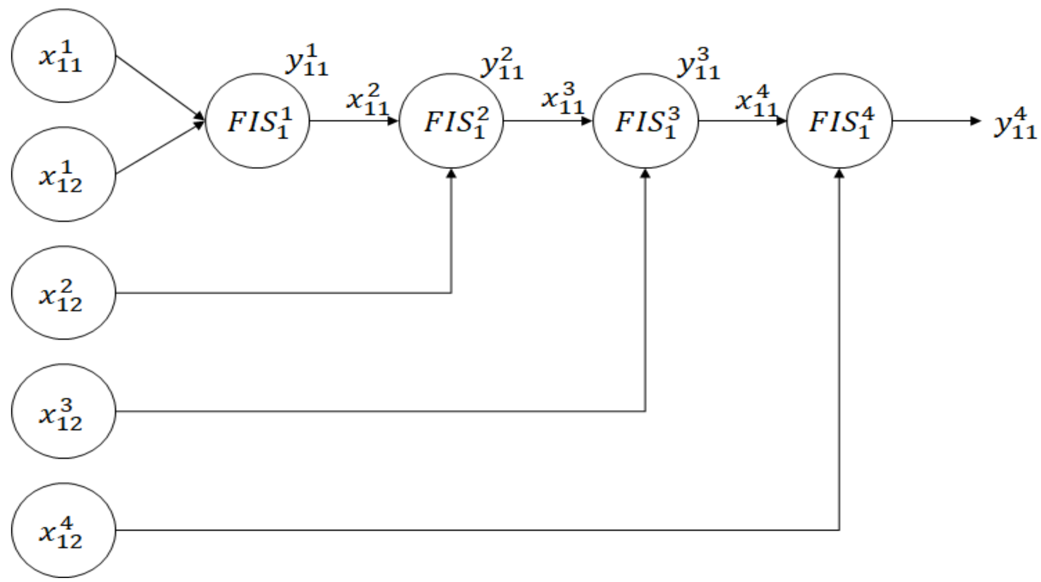
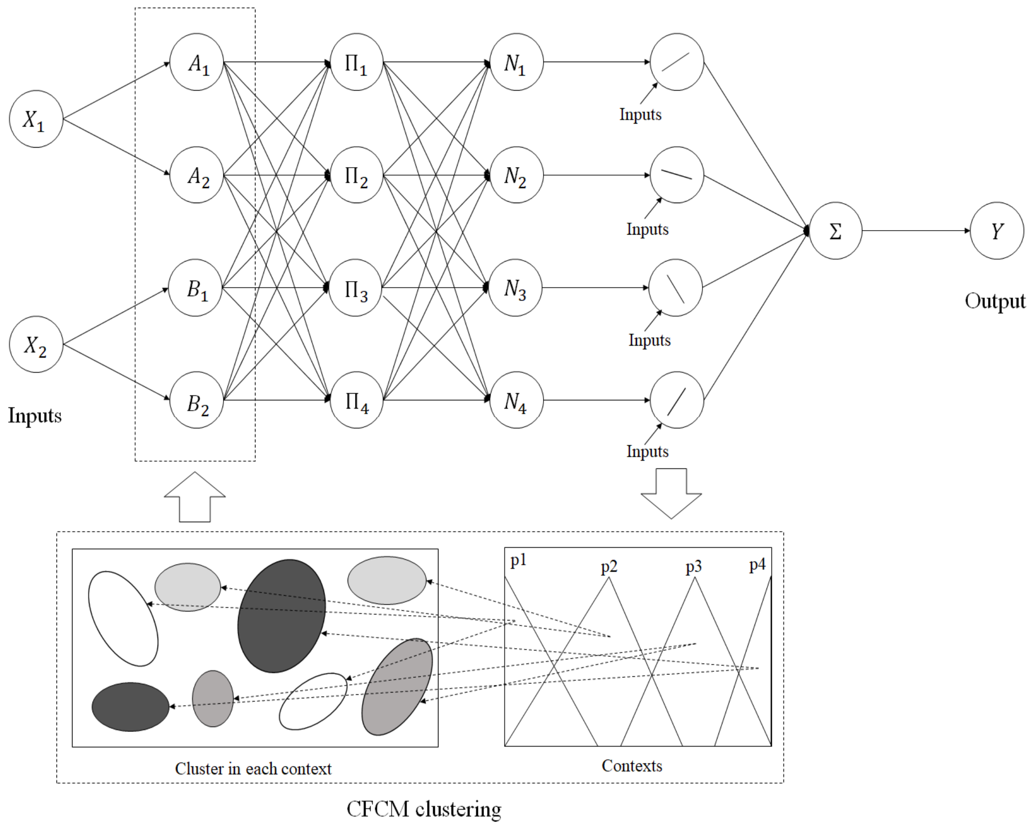
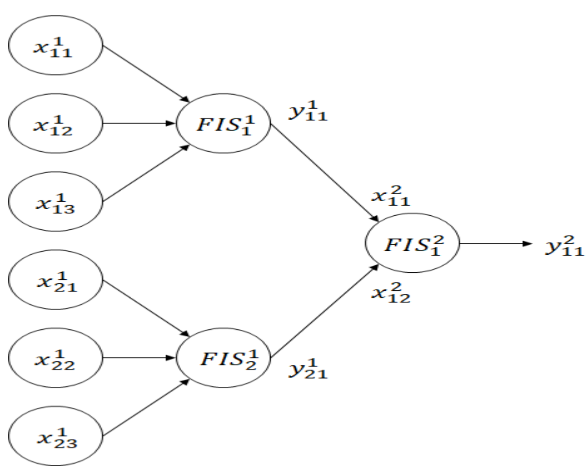


Figure 4. Tree structure in incremental form.

Therefore, with the number of input sets, the cumulative number of rules in the incremental fuzzy tree is linear. Based on the contribution to the final output value, input selection at various levels of the incremental fuzzy tree uses input rank. Generally, the input value that contributes the most is used at the lowest level, and at the highest level, the input value that contributes the least is used. This implies that the input value of the low-rank depends upon the input value of the high-rank. In the incremental fuzzy tree, irrespective of other important inputs, each input value usually contributes to some degree to the inference method. In this paper, to prevent over-generation of fuzzy rules due to large-scale databases and to generate meaningful rules, we propose a CFCM-ANFIS with an incremental tree structure rather than a single type of CFCM-ANFIS. As seen in Figure 5, this allows one to rank inputs using existing data to create the fuzzy tree.



(**a**)



(**b**)

Figure 5. Design of three ANFISs with an incremental tree structure: (**a**) ANFIS structure based on CFCM clustering and (**b**) ANFIS structure based on incremental-tree-structured CFCM clustering.

4. Experiment and Analysis

In this section, to evaluate the predicted performance of ANFIS with an incremental tree structure based on CFCM clustering methods described in Section 3, experiments were conducted to predict computer performance using the computer hardware dataset. In this experiment, for the predictive performance of ANFIS using the grid-based rule generation method, which is a representative ANFIS, and ANFIS using the FCM clustering-based rule generation method, as well as ANFIS using the proposed method, the incremental tree structure-based CFCM clustering-based rule generation method is compared and analyzed.

4.1. Building Heating-and-Cooling Dataset

The building heating-and-cooling dataset is a dataset [41,42] used for energy efficiency forecasting created by Xifara. It consists of eight input variables and two output variables and has a data size of 768 × 10. Relative compaction, surface area, wall area, roof area, total height, direction, glazing area and distribution of the glazing area are input variables. The heating and cooling loads are the output factors, but this analysis uses only the heating load. To conduct the experiment, the building heating dataset was equally divided into learning and verification sets and data values were normalized to between 0 and 1.

4.2. Experimental Method and Analysis of Results

The predicted performance of grid-based AFNIS and FCM clustering-based ANFIS, which are general rule generation methods, and of the increasing tree structure based on the CFCM clustering method proposed in this study were compared and analyzed. As described above, a grid-based ANFIS creates rules by dividing the input space into lattices, and FCM clustering-based ANFIS clusters the input space using FCM clustering to create rules. The proposed method uses CFCM clustering in input and output spaces to create contexts and clusters to create rules.

First, the grid-based ANFIS experiment confirmed the predicted performance by increasing the MF by 1 from 2 to 5. By adjusting the number of clusters and the fuzzification coefficient, the FCM-clustering-based ANFIS experiment was carried out. The number of clusters increased by 2 from 2 to 20 and the coefficient of fuzzification was set at 2 to confirm the performance anticipated. Finally, for the experiment on AFNIS using the incremental tree structure based on the CFCM clustering method, which is the method proposed in this study, we designed three CFCM-clustering-based ANFISs with two inputs and one output as an incremental tree structure. The entire input variable was then ranked according to the correlation coefficient and used as input to each ANFIS. In the CFCM clustering method, the number of contexts (p) increased by 2 from 2 to 6, and the number of clusters (c) increased by 2 from 2 to 20, confirming the predicted performance. Each ANFIS was performed 10 iterations, and the value with the minimum verification root mean square error (RMSE) was used as the result value. All experiments for Grid-ANFIS, FCM-ANFIS, Incremental-CFCM-ANFIS, LR and RBFN were conducted using Matlab, and the OS is a window10 environment. Table 1 summarizes the prediction performance of the grid-based ANFIS, and Figure 6 shows the prediction results of the grid-based ANFIS. As can be seen in Table 1, when there are two MFs, 256 rules are created, and the verification root mean square error (RMSE) is ~2.2471. When there are three, four, or five MFs, the number of rules increases exponentially, indicating a calculation error.

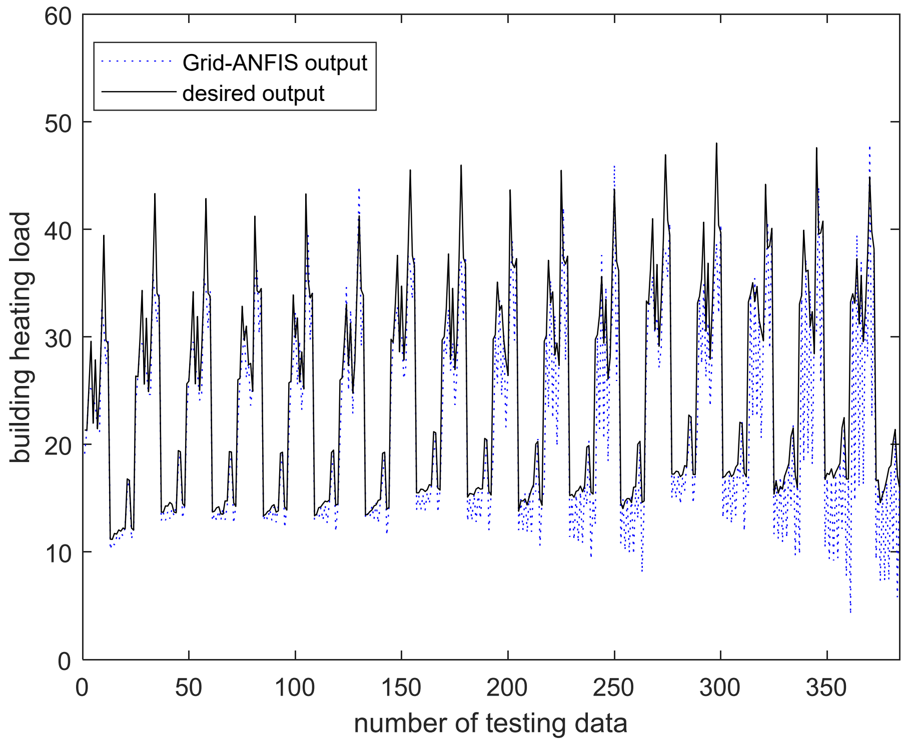


Figure 6. Comparison of the predicted and actual output values of grid-based ANFIS.

Table 1. Prediction experiment results from grid-based ANFIS.

Algorithm	Number of MFs*	Number of Rules	Training RMSE*	Testing RMSE
Grid-ANFIS	2	256	0.6728	2.2471
	3	-	-	-
	4	-	-	-
	5	-	-	-

MFs (Membership Functions)

RMSE (Root Mean Square Error)

Table 2 summarizes the prediction performance of ANFIS based on FCM clustering, and Figure 7 shows the prediction results. As can be seen in Table 2, when there are 10 clusters, 10 rules are created and the verification RMSE is ~2.0671. The prediction efficiency of the CFCM-based ANFIS in the form of an incremental tree is summarized in Table 3, which is the approach suggested in this report. Figure 8 shows the prediction results. As can be seen in Table 3, when there are 6 contexts and 20 clusters, 120 rules are created and the verification RMSE is ~1.8705, yielding the best prediction performance. Table 4 compares and analyzes the prediction performance of grid-ANFIS, FCM-ANFIS and Incremental-CFCM-ANFIS, and the linear regression (LR) model and the radial basis function network (RBFN) model used for prediction problems. For LR and RBFN, the verification RMSE values were approximately 3.04 and 25.45 respectively, and a fuzzy rule was not generated. In Grid-ANFIS, if there are 2 membership functions, 256 rules are created and the verification RMSE value is about 2.25. In FCM-ANFIS, when the number of clusters is 10, 10 rules are created and the verification RMSE value is about 2.07. Finally, the proposed method shows that when the number of contexts is 6 and the number of clusters is 20, 120 rules are generated and the verification RMSE value is about 1.87, showing the best prediction performance.

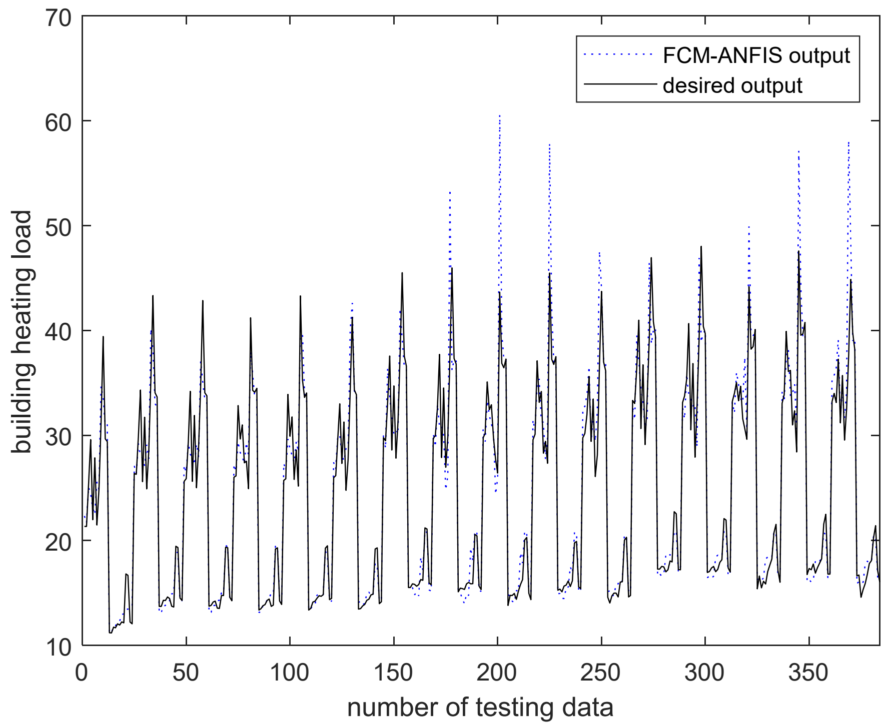


Figure 7. Comparison of the predicted and actual output values of FCM clustering-based ANFIS.

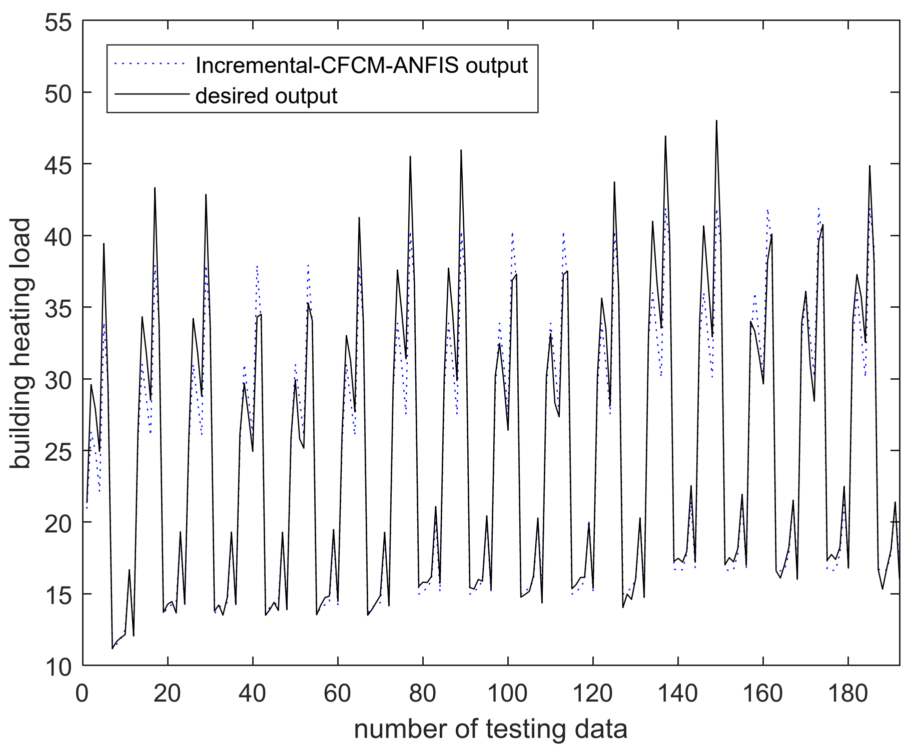


Figure 8. Comparison of ANFIS predicted values with actual output values in an incremental tree structure based on CFCM clustering.

Table 2. Prediction experiment results from FCM-clustering-based ANFIS.

Algorithm	Number of Clusters	Number of Rules	Training RMSE	Testing RMSE
FCM-ANFIS	2	2	2.5042	2.6548
	4	4	1.7882	2.3150
	6	6	1.7814	2.2088
	8	8	1.6466	2.1173
	10	10	1.6286	2.0671
	12	12	1.6702	2.1142
	14	14	1.0611	2.2425
	16	16	1.3782	3.2958
	18	18	1.4028	3.5073
	20	20	1.1791	7.1332

Table 3. Results from ANFIS prediction experiment on the incremental tree structure based on CFCM clustering.

Algorithm	Number of Contexts	Number of Clusters	Number of Rules	Training RMSE	Testing RMSE
Incremental-CFCM-ANFIS	2	2	4	2.6915	3.1126
		4	8	2.3334	2.7517
		6	12	1.7173	2.0464
		8	16	1.5364	1.8881
		10	20	1.5260	1.8742
		12	24	1.5238	1.8738
		14	28	1.5241	1.8725
		16	32	1.5343	1.9118
		18	36	1.5240	1.8725
		20	40	1.5240	1.8724
	4	2	8	2.5414	2.9155
		4	16	1.5981	1.9542
		6	24	1.5240	1.8730
		8	32	1.5241	1.8729
		10	40	1.5240	1.8726
		12	48	1.5296	1.8764
		14	56	1.5248	1.8724
		16	64	1.5241	1.8719
		18	72	1.5241	1.8717
		20	80	1.5241	1.8716
	6	2	12	1.8964	2.2700
		4	24	1.5186	1.8796
		6	36	1.5232	1.8732
		8	48	1.5253	1.8729
		10	60	1.5241	1.8719
		12	72	1.5241	1.8716
		14	84	1.5242	1.8713
		16	96	1.5242	1.8711
		18	108	1.5242	1.8710
		20	120	1.5242	1.8705

Table 4. Analysis of experimental results from ANFIS based on grid-based ANFIS, FCM-clustering-based ANFIS and the CFCM-clustering-based incremental tree structure.

Algorithm	Hyperparameters	Number of Rules	Training RMSE	Testing RMSE
Linear regression (LR)	-	-	3.3453	3.0352
Radial basis function network (RBFN)	Learning rate (0.0001)	-	26.9523	25.4493
Grid-ANFIS	2 MFs	256	0.6728	2.2471
FCM-ANFIS	10 clusters	10	1.6286	2.0671
Incremental-CFCM-ANFIS	6 contexts, 20 clusters	120	1.5242	1.8705

5. Conclusions

ACFCM-based incremental tree-structured ANFIS was proposed. To confirm the validity of the proposed method, the prediction performance was compared with the commonly used grid-based ANFIS and clustering-based ANFIS. As a result of the experiment, the CFCM-based incremental tree-structured ANFIS proposed in this study was confirmed to be superior to the existing ANIFS model in terms of performance. In addition, it was confirmed that generating meaningful rules rather than multiple rules can improve prediction performance. As a future research plan, we plan to design a multi-ANFIS in various forms, rather than an incremental tree structure, and apply an optimization algorithm to generate meaningful rules.

Author Contributions: C.-U.Y. suggested the idea for this study and performed the experiments. K.-C.K. designed the experimental method. Both authors wrote and critically revised the paper. All authors have read and agreed to the published version of the manuscript.

Funding: This work was supported by the "Human Resources Program in Energy Technology" of the Korea Institute of Energy Technology Evaluation and Planning (KETEP) and was granted financial resources from the Ministry of Trade, Industry & Energy, Korea. (No. 20194030202410). This research was supported by the Basic Science Research Program through the National Research Foundation of Korea (NRF) and funded by the Ministry of Education (No. 2017R1A6A1A03015496).

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

References

1. Khan, M.A.; Algarni, F. A healthcare monitoring system for the diagnosis of heart disease in the IoMT cloud environment using MSSO-ANFIS. *IEEE Access* **2020**, *8*, 122259–122269.
2. Liu, M.; Dong, M.; Wu, C. A new ANFIS for parameter prediction with numeric and categorical inputs. *IEEE Trans. Autom. Sci. Eng.* **2010**, *7*, 645–653.
3. Son, Y.S.; Kim, H.J.; Kim, J.T. A video-quality control scheme using ANFIS architecture in a DASH environment. *Korean Soc. Broad Eng.* **2018**, *23*, 104–114.
4. Kannadasan, K.; Edla, D.R.; Yadav, M.H.; Bablani, A. Intelligent-ANFIS model for predicting measurement of surface roughness and geometric tolerances in three-Axis CNC milling. *IEEE Trans. Instrum. Meas.* **2020**, *69*, 7683–7694.
5. Penghui, L.; Ewees, A.A.; Beyaztas, B.H.; Qi, C.; Salih, S.Q.; Ansari, N.A.; Bhagat, S.K.; Yaseen, Z.M.; Singh, V.P. Metaheuristic optimization algorithms hybridized with artificial intelligence model for soil temperature prediction: Novel model. *IEEE Access* **2020**, *8*, 51884–51904.
6. Hwang, D.H.; Bae, Y.C. A prediction of bid price using MLP and ANFIS. *J. Korean Inst. Intell. Syst.* **2020**, *30*, 309–314.
7. Krasopoulos, C.T.; Beniakar, M.E.; Kladas, A.G. Multicriteria PM motor design based on ANFIS evaluation of EV driving cycle efficiency. *IEEE Trans. Transp. Electrif.* **2018**, *4*, 525–535.
8. Hasnony, I.M.E.; Barakat, S.I.; Mostafa, R.R. Optimized ANFIS model using hybrid metaheuristic algorithms for Parkinson's disease prediction in IoT environment. *IEEE Access* **2020**, *8*, 119252–119270.

9. Morshedizadeh, M.; Kordestani, M.; Carriveau, R.; Ting, D.S.K.; Saif, M. Power production prediction of wind turbines using a fusion of MLP and ANFIS networks. *IET Renew. Power Gener.* **2018**, *12*, 1025–1033.
10. Khosravi, A.; Nahavandi, S.; Creighon, D. Prediction interval construction and optimization for adaptive neurofuzzy inference systems. *IEEE Trans. Fuzzy Syst.* **2011**, *19*, 983–988.
11. Elbaz, K.; Shen, S.L.; Sun, W.J.; Yin, Z.Y.; Zhou, A. Prediction model of shield performance during tunneling via incorporating improved particle swarm optimization into ANFIS. *IEEE Access* **2020**, *8*, 39659–39671.
12. Dovzan, D.; Skrjanc, I. Fuzzy space partitioning based on hyperplanes defined by eigenvectors for takagi-sugeno fuzzy model identification. *IEEE Trans. Ind. Electron.* **2019**, *67*, 5144–5153.
13. Castiello, C.; Fanelli, A.M.; Lucarelli, M.; Mencar, C. Interpretable fuzzy partitioning of classified data with variable granularity. *Appl. Soft Comput.* **2019**, *74*, 567–582.
14. Alexandrisdis, A.; Chondrodima, E.; Sarimveis, H. Radial basis function network training using a nonsymmetric partition of the input space and particle swarm optimization. *IEEE Trans. Neural Netw. Learn. Syst.* **2013**, *24*, 219–230.
15. Verstraete, J. The spatial disaggregation problems: Simulating reasoning using a fuzzy inference system. *IEEE Trans. Fuzzy Syst.* **2016**, *25*, 627–641.
16. Lee, J.S.; Teng, C.L. An enhanced hierarchical clustering approach for mobile sensor networks using fuzzy inference systems. *IEEE Internet Things J.* **2017**, *4*, 1095–1103.
17. Su, Z.G.; Denoeux, T. BPEC: Belief-peaks evidential clustering. *IEEE Trans. Fuzzy Syst.* **2019**, *27*, 111–123.
18. Xu, P.; Deng, Z.; Cui, C.; Zhang, T.; Choi, K.S.; Gu, S.; Wang, J. Concise fuzzy system modeling integrating soft subspace clustering and sparse learning. *IEEE Trans. Fuzzy Syst.* **2019**, *27*, 2176–2189.
19. Sujil, A.; Kumar, R.; Bansal, R.C. FCM clustering-ANFIS-based PV and wind generation forecasting agent for energy management in a smart microgrid. *J. Eng.* **2019**, *2019*, 4852–4857.
20. Neamatollani, P.; Naghibzadeh, M.; Abrishammi, S. Fuzzy-based clustering-task scheduling for lifetime enhancement in wireless sensor networks. *IEEE Sens. J.* **2017**, *17*, 6831–6844.
21. Gu, X.; Chung, F.L.; Ishibuchi, H.; Wang, S. Imbalanced TSK fuzzy classifier by cross-class bayesian fuzzy clustering and imbalance learning. *IEEE Trans. Syst. ManCybern. Syst.* **2017**, *47*, 2005–2020.
22. Ojha, V.K.; Snasel, V.; Abraham, A. Multiobjective programming for type-2 hierarchical fuzzy inference trees. *IEEE Trans. Fuzzy Syst.* **2018**, *26*, 915–936.
23. Shi, J.Z. A fractional order general type-2 fuzzy PID controller design algorithm. *IEEE Access* **2020**, *8*, 52151–52172.
24. Wang, L.X. A new look at type-2 fuzzy sets and type-2 fuzzy logic systems. *IEEE Trans. Fuzzy Syst.* **2016**, *25*, 693–706.
25. Das, A.K.; Sundaram, S.; Sundararajan, N. A self-regulated interval type-2 neuro-fuzzy inference system for handling nonstationarities in EEG signals for BCI. *IEEE Trans. Fuzzy Syst.* **2016**, *24*, 1565–1577.
26. Das, A.K.; Subramanian, K.; Sundaram, S. An evolving interval type-2 neurofuzzy inference system and its metacognitive sequential learning algorithm. *IEEE Trans. Fuzzy Syst.* **2015**, *23*, 2080–2093.
27. Zhou, H.; Ying, H.; Zhang, C. Effects of increasing the footprints of uncertainty on analytical structure of the classes of interval type-2 mamdani and TS fuzzy controllers. *IEEE Trans. Fuzzy Syst.* **2019**, *27*, 1881–1890.
28. Eyoh, I.; John, R.; Maere, G.; Kayacan, E. Hybrid learning for interval type-2 intuitionistic fuzzy logic systems as applied to identification and prediction problems. *IEEE Trans. Fuzzy Syst.* **2018**, *26*, 2672–2685.
29. Sumati, V.; Patvardhan, S. Interval type-2 mutual subsethod fuzzy neural inference system (IT2MSFuNIS). *IEEE Trans. Fuzzy Syst.* **2018**, *26*, 203–215.
30. Biglarbegian, M.; Melek, W.W.; Mendel, J.M. On the stability of interval type-2 TSK fuzzy logic control systems. *IEEE Trans. Syst. ManCybern. Part. B* **2010**, *40*, 798–818.
31. Gracia, G.R.; Hagras, H.; Pomares, H.; Ruiz, I.R. Toward a fuzzy logic system based on general forms of interval type-2 fuzzy sets. *IEEE Trans. Fuzzy Syst.* **2019**, *27*, 2381–2395.
32. UUCI Machine Learning Repository. Available online: https://archive.ics.uci.edu/ml/datasets (accessed on 27 November 2020).
33. Sugeno, M.; Yasukawa, T. A fuzzy-logic based approach to qualitative modeling. *IEEE Trans. Fuzzy Syst.* **1993**, *1*, 7–31.
34. Haykin, S. *Neural Networks*. Macmillan Inc.: New York, NY, USA, 1994.
35. Jang, J.S.R. ANFIS: Adaptive-network based fuzzy inference system. *IEEE Trans. Syst. ManCybern.* **1993**, *23*, 665–685.

36. Jang, J.S.R.; Sun, C.T.; Mizutani, E. *Neuro-Fuzzy and Soft Computing: A Computational Approach to Learning and Machine Intelligence*; Prentice Hall: Upper Saddle River, NJ, USA, 1997.
37. Bezdek, J.C. *Pattern Recognition with Fuzzy Objective Function Algorithms*; Plenum, springer.com/gp/book/9781475704525, 1981.
38. Bezdek, J.C. Fuzzy Mathematics in Pattern Classification. PhD Thesis, Applied Math Center, link.springer.com/chapter/10.1007/3-540-27335-2_5, Cornell University, Ithaca, NY, USA, 1973.
39. Siddique, N.; Adeli, H. *Computational Intelligence: Synergies of Fuzzy Logic. Neural Networks and Evolutionary Computing*; Wiley: Hoboken, NJ, USA, 2013.
40. Pedrycz, W. Conditional fuzzy C-means. *Pattern Recognit. Lett.* **1996**, *17*, 625–632.
41. Archive.ics.uci.edi/ml/datasets/Energy+efficiency.
42. Tsanas, A.; Xifara, A. Accurate quantitative estimation of energy performance of residential buildings using statistical machine learning tools. *Energy Build.* **2012**, *49*, 560–567.