Global Analysis and Discrete Mathematics
Volume 6, Issue 1, pp. 143–178
ISSN: 2476-5341 *Research Article*

Prediction of Fuzzy Nonparametric Regression Function: A Comparative Study of a New Hybrid Method and Smoothing Methods

Mahdi Danesh · Sedigheh Danesh · Tahereh Razzaghnia · Ali Maleki

Received: 7 January 2021 / Accepted: 22 August 2021

Abstract In this paper, the fuzzy regression model is considered with crisp inputs and symmetric triangular fuzzy output. This study aims to formulate the fuzzy inference system based on the Sugeno inference model for the fuzzy regression function prediction by the fuzzy least-squares problem-based on Diamond's distance. In this study, the fuzzy least-squares problem is used to optimize consequent parameters, and the results are derived based on the V-fold crossvalidation, so that the validity and quality of the proposed method can be guaranteed. The proposed method is used to reduce the bias and the boundary effect of the estimated underlying regression function. Also, a comparative study of the fuzzy nonparametric regression function prediction is carried out between the proposed model and smoothing methods, such as k-nearest neighbor (k-NN), kernel smoothing (KS), and local linear smoothing (LLS). Different approaches are illustrated by some examples and the results are compared. Comparing the results indicates that, among the various prediction models, the proposed model is the best, decreasing the boundary effect significantly. Also, in comparison with different methods, in both one-dimensional and two-dimensional inputs, it may be considered the best candidate for the prediction.

M. Danesh
Imam Khomeini International University - Buin Zahra Higher Education Center of Engineering and Technology, Buein Zahra, Ghazvin, Iran

S. Danesh
Young Researchers and Elite Club, East Tehran Branch, Islamic Azad university, Tehran, Iran.

T. Razzaghnia (Corresponding Author)
Department of Statistics, North Tehran Branch, Islamic Azad University, Tehran, Iran.
E-mail: razaghnia@riau.ac.ir

A. Maleki
Department of Statistics, West Tehran Branch, Islamic Azad University, Tehran, Iran.

Keywords fuzzy nonparametric regression · k-nearest neighbor smoothing · kernel smoothing · local linear smoothing · adaptive neuro-fuzzy inference system · V-fold cross-validation

Mathematics Subject Classification (2010) 62G08 · 68M10

1 Introduction

The fuzzy logic and fuzzy systems are one of the ways available to address uncertainty in engineering and manufacturing applications [7]. Tanaka, Uejima ,and Asia [48] suggested fuzzy regression analysis. Many different fuzzy regression approaches have been proposed by different researchers. In general, there are two approaches in fuzzy regression analysis: linear programming-based method (see, e.g. [26,29,33,35]) and fuzzy least squares method (see, e.g. [1,27,36–38,47]). The first method is based on minimizing fuzziness as an optimization criterion. The second method uses least-squares of errors as a fitting criterion. The advantages of the first approach are its simplicity in programming and computations, while that the degree of fuzziness between the observed values and the predicted values is minimized by using a fuzzy least-squares method. Tanaka et al [26,49,50] regarded fuzzy data as a possibility distribution, and the deviations between the observed values and the estimated values are supposed to be due to the fuzziness of the system structure. Nadaraya [31] claimed that there exist wellknown methods for regression parameters estimation from empirical data. It is an approximation to the regression line on the basis of sampling when the sample size increase unboundedly. Nadaraya [32] examined some properties of the multivariate empirical probability density with kernels of arbitrary form in the case where the true probability density has a Taylor expansion in arguments about each point of independent variables. Epanechnikov [10] claimed that some non-parametric estimations of a true multivariate probability density can be applied to solve various problems involving the statistical tests of hypotheses. Kim and Chen [25] have addressed conceptual, mythological characteristics of fuzzy and non-parametric regression methods and assessed their relative performances under various conditions. They have also proposed a guideline for selecting the appropriate prediction methods among fuzzy regression, nonparametric regression, and least square regression with compression of these methods for descriptive and predictive purposes. Also, statistical nonparametric smoothing techniques have achieved significant development in recent years. For instance, Loftsgaarden, and Quesesberry [27] introduced the k-NN weight sequence. Cheng et al [2] and Wang et al [51] fuzzified the k-nearest neighbor and the kernel smoothing methods, and the local linear smoothing technique respectively to predict the fuzzy nonparametric regression function. Also, Razaghnia, and Danesh [39] predicted the fuzzy nonparametric regression function for trapezoidal data, and Farnoosh et al [12] proposed fuzzy ridge regression method for the fuzzy nonparametric regression function prediction. In recent years, the prediction of the regression parameters has gained great attention among the researchers of neural networks. Ishibushi, and Tanaka [18, 19] have suggested several fuzzy nonparametric regression methods by using traditional backpropagation networks. Fausett [13] has proposed the fundamental concepts of neural networks such as architectures, algorithms, and applications. Also, Ishibuchi, Kwon, and Tanaka [17] have introduced a learning algorithm of fuzzy neural networks with triangular fuzzy weights. Fuzzy neural networks have been applied for the fuzzy regression (see, e.g. [3, 4, 30, 34]). Fuzzy reasoning system was proposed by Takagi and Sugeno [44]. Jang [20] proposed an adaptive network-based on fuzzy inference system (ANFIS). Some new soft computing techniques, such as artificial neural networks, fuzzy inference systems, evolutionary computation, and their hybrids have been successfully employed for developing predictive models to estimate the needed parameters. These techniques have more attraction in many research fields. For example, Tamer, Kamel, and Hassan [46] have introduced the application of the ANFIS approach for fault classification in transmission lines. In medical field, Kenar Koohi, Soleimanjahi, and Falahi [23] have applied the ANFIS method to predict human Papillomavirus and in the geography field, Talei, Chua, and Wong [45] have used the ANFIS method in evaluating rainfall and discharge inputs. In industry, the ANFIS method has been applied to predict by Zarandi et al. [14] and Rizal et al [40]. So, Cheng, and Lee [3] formulated the ANFIS model for fuzzy regression analysis using linear programming. Dalkilic & Apaydin [5, 6] used the ANFIS model to analyze the switching regression, and estimate the fuzzy regression parameters Danesh and Khalili developed an optimized adaptive neuro-fuzzy inference system model to fuse the surface image, motor current, strain and vibration signal features for tool wear monitoring [8]. In this paper, the ANFIS model is formulated for analyzing fuzzy regression using the fuzzy least squares based on Diamond's distance, also, stability of the proposed method is shown by V-fold cross-validation technique. This paper aims to obtain the consequence parameters using the fuzzy least-squares Problembased on Diamond's distance in the training algorithm of the ANFIS method and, then, applying the proposed method for the fuzzy regression function prediction with crisp inputs and fuzzy output to increase efficiency and decrease boundary effect of the estimated underlying regression function, and finally, comparing the obtained results with different smoothing methods, such as k-NN, KS and LLS methods. In this work, it is presented that the boundary effect of the proposed method is significantly less than smoothing methods. Also, the proposed method has more efficiency compared with smoothing methods and decreases error values of GOF and BIAS simultaneously. In the proposed approach, to obtain the premise parameters, the center of the observed fuzzy outputs are trained using the gradient descent method. The obtained premise parameters are applied in the consequence section and the consequence parameters are obtained using the fuzzy least squares method. Hence, the ANFIS method is fuzzified using this method. Also, V-fold cross-validation technique is used as a validation method.

The advantage of the paper is used V-fold Cross Validation technique for training ANFIS network. The proposed method and the three smoothing methods are compared with the CV scale. The conducted simulation experiments are shown that the performance of the proposed method is better than the smoothing methods, which reduces the CV. in the proposed approach when the observation numbers are increased, the accuracy is increased, in comparison with the existing smoothing methods. Generally, the proposed method (V-fold cross-validation technique with triangular data) is reduced the fuzziness of the system and has faster adaptation. Also, the proposed method has reduced the bias and the boundary effect of the predicted underlying regression function.

The organization of this paper is as follows:

In the next section, the concepts and formulations of the k-nearest neighbor smoothing, the kernel smoothing, the local linear smoothing, and the ANFIS methods are explained. In the following, the ANFIS method is expanded for the fuzzy nonparametric regression function prediction and the consequence parameters are obtained by using the fuzzy least-squares based on Diamond's distance. Then three different examples are described for fitting fuzzy nonparametric regression functions by using different approaches. Error-values are used to numerically evaluate the performance of each method. Finally, these models are compared together to discover the most suitable one and the results are annualized.

2 Preliminaries

In this section, the concepts and formulations of the various models will be explained.

Definition 1 Suppose that $X = (l_X, a_X, r_X)$ is a $L-R$ fuzzy number so that a_X, l_X and r_X are the center, the lower and the upper limits being this fuzzy number, respectively. The membership function of $X = (l_X, a_X, r_X)$ is

$$\mu_X(z) = \begin{cases} L\left(\dfrac{a_X - z}{a_x - l_X}\right) , & l_X < z < a_x \\ R\left(\dfrac{z - a_X}{r_x - a_X}\right) , & a_X < z < r_x \\ 0 & \text{otherwise.} \end{cases} \tag{1}$$

Let $A = (l_A, a_A, r_A)$ and $B = (l_B, a_B, r_B)$ be any two $L - R$ fuzzy numbers. If $L(X) = R(X)1 - |X|$, we have triangular fuzzy number.

So, the distance between A and B can be expressed as [9]:

$$d^2(A, B) = (l_B - l_A)^2 + (a_B - a_A)^2 + (r_B - r_A)^2 \tag{2}$$

This distance measures the closeness between the membership functions of two triangular fuzzy numbers. When $d^2(A, B) = 0$, it means that the membership functions A and B are equal. Also, the result of a fuzzy addition of triangular fuzzy numbers is a triangular fuzzy number again.

Definition 2 The function $f(x)$ is a mapping from x to Y where $x = (x_0, x_1, \ldots, x_p)$ is a p-dimentional vector crisp independent variable. Domain is assumed to be $D \subseteq R^p$ Consider the following the fuzzy regression model:

$$Y = f(x)\{+\}\epsilon \tag{3}$$

where Y has the fuzzy structure and ϵ is an observed error with conditional mean zero and variance $\sigma^2(x)$ given x. Y_j is j^{th} the response variable. The response variable Y_j can also be represented as a triangular fuzzy number. The space of all triangular fuzzy numbers are denoted by $TN(R)$, i.e, $TN(R) = \{Y_j j : Y_j = (a_j, \alpha_j, \beta_j)\}$ with α_j as the center, $\alpha_j = a_j - l_j$ and $\beta_j = \alpha_j - r_j$ as the left and the right spread of triangular fuzzy number Y_j where l_j and r_j are lower and upper limits being this fuzzy number. A symmetric triangular fuzzy number Y_j can be written as $TN(R) = \{Y_j : Y_j = (\alpha_j, \beta_j)\}$ where α_j and β_j are the center, and the spread of a symmetric triangular fuzzy number respectively and $\beta_j = r_j - a_j = a_j - l_j$. However, in many practical situations, the relationship between the input and output variable is frequently unknown. Thus, nonparametric regression is needed where the functional form is unknown. In this study, a fuzzy regression model is considered with crisp inputs and triangular fuzzy output that the functional form is unknown.

2.1 Forecasting methods

In the kinds of literature and practical applications, smoothing methods are widely used as the forecasting methods for nonparametric regression. In this section, for facilitating our comparison, the fuzzy k-nearest neighbor smoothing (k-NN), the fuzzy kernel smoothing (KS) and the local linear smoothing (LLS) procedures are briefly described for fitting the fuzzy nonparametric regression function respectively.

2.2 k-Nearest neighbor smoothing method (k-NN)

Consider the fuzzy nonparametric regression model as, in Eq. (3). The k-nearest neighbor smoothing method is described for the regression function estimation $f(x)$ at $x \in D$ The estimation of a regression function $f(x)$ is defined as [27]:

$$\hat{f}(x) = \left(\hat{l}(x), \hat{a}(x), \hat{r}(x)\right) = \left(\sum_{j=1}^{n} \omega_j(x) l_{y_j}, \sum_{j=1}^{n} \omega_j(x) a_{y_j}, \sum_{j=1}^{n} \omega_j(x) r_{y_j}\right), \tag{4}$$

where a_{y_j}, l_{y_j} and r_{y_j} are the lower, the center and the upper limits of the observed $NT(R)$ fuzzy output Y_j and $\omega_j(x)$ $(j = 1, \ldots, n)$ is a weight sequence at x. Also, $\omega_j(x)$ is expressed as:

$$\omega_j(x) = \begin{cases} \dfrac{1}{k} \ , j \in J(x) \ , \quad j = 1, \ldots \\ 0 \quad \text{otherwise}, \end{cases} \tag{5}$$

where $J(x)$ is a set of j and x_j $(j = 1, \dots, n)$ is one of the k-nearest observations to x So, by substituting $\omega_j(x)$ in Eq. (4), it can be written as follows:

$$\hat{f}(x) = \left(\hat{l}(x), \hat{a}(x), \hat{r}(x)\right) = \left(\frac{1}{k}\sum_{j=1}^{n} l_{y_j}, \frac{1}{k}\sum_{j=1}^{n}(x)a_{y_j}, \frac{1}{k}\sum_{j=1}^{n}(x)r_{y_j}\right). \tag{6}$$

Also, the inclusion condition, which was proposed by Tanaka et al [47] for the fuzzy regression, can be approximated in selecting the appropriate k value.

2.3 Kernel smoothing method (KS)

In this method, the kernel estimation is still presented by Eq. (4), but the weight sequence $\omega_j(x)$ is generated by [27]:

$$\omega_j(x) = \frac{K_h(x - x_j)}{\sum_{h=1}^{n} K_h(x - x_j)} = \frac{K\left(\frac{x - x_j}{h}\right)}{\sum_{h=1}^{n} K\left(\frac{x - x_j}{h}\right)}, \qquad j = 1, \dots, n \tag{7}$$

where $k_h(\cdot)$ is a kernel function. By substituting Eq. (7) in Eq. (4), $\hat{f}(x)$ for $x \in D$ can be expressed as:

$$\hat{f}(x) = \left(\hat{l}(x), \hat{a}(x), \hat{r}(x)\right) = \left(\frac{\sum_{h=1}^{n} K_h((x - x_j)l_{y_j})}{\sum_{h=1}^{n} K_h(x - x_j)}, \frac{\sum_{h=1}^{n} K_h((x - x_j)a_{y_j})}{\sum_{h=1}^{n} K_h(x - x_j)}, \frac{\sum_{h=1}^{n} K_h((x - x_j)r_{y_j})}{\sum_{h=1}^{n} K_h(x - x_j)}\right). \tag{8}$$

Also, for multidimensional case with variables $x_j = (x_{j1}, \dots, x_{jp})$ $(j = 1, \dots, n)$, the following the kernel weights can be used:

$$\omega_j(x) = \frac{\prod_{i=1}^{p} K_h(x_i - x_{ji})}{\sum_{h=1}^{n} K_h(x - x_j)} \tag{9}$$

2.4 Local linear smoothing method (LLS)

The same above discussions, consider the fuzzy nonparametric regression model in Eq. (3). We shall estimate $l(x)$, $a(x)$ and $r(x)$ in $f(x) = (l(x), a(x), r(x))$. So, the estimation $f(x)$ in x_0 can be expressed as:

$$\hat{f}(x_0) = \left(\hat{l}(x_0), \hat{a}(x_0), \hat{r}(x_0)\right) \tag{10}$$

Moreover, suppose $l(x)$, $a(x)$ and $r(x)$ have continuous partial derivatives to each element x_i in domain D. Then for a given $x_0 = (x_{01}, x_{02}, \ldots, x_{0p}) \in D$ and with Taylor's expansion, $l(x)$, $a(x)$ and $r(x)$ can be locally estimated in a neighborhood of x_0, respectively, by the following linear functions:

$$l(x) \cong \tilde{l}(x) = l(x_0) + l^{(x_1)}(x_0)(x_1 - x_{01}) + \cdots + l^{(x_p)}(x_0)(x_p - x_{0p}) \tag{11}$$

$$a(x) \cong \tilde{a}(x) = a(x_0) + a^{(x_1)}(x_0)(x_1 - x_{01}) + \cdots + a^{(x_p)}(x_0)(x_p - x_{0p}) \tag{12}$$

$$r(x) \cong \tilde{r}(x) = r(x_0) + r^{(x_1)}(x_0)(x_1 - x_{01}) + \cdots + r^{(x_p)}(x_0)(x_p - x_{0p}) \tag{13}$$

where $l^{(x_i)}(x_0)$, $a^{(x_i)}(x_0)$ and $r^{(x_i)}(x_0)$ are partial derivatives of $l(x)$, $a(x)$ and $r(x)$ with respect to each element x_i at x_0 $(i = 1, \ldots, p)$. Based on Diamond's distance Eq. (2) and with the observed data (x_j, Y_j) $(j = 1, \ldots, n)$, the following locally weighted least-squares problem is obtained for estimating $f(x_0) = \left(l(x_0), a(x_0), r(x_0)\right)$ in the local linear smoothing technique. That is, minimize

$$\begin{aligned}
\sum_{j=1}^{n} d^2(f_j, \hat{f}_j) &= \sum_{j=1}^{n} d^2\Big((l_{y_j}, a_{y_j}, r_{y_j}), (\hat{l}(x_j), \hat{a}(x_j), \hat{r}(x_j))\Big) K_h(\|x_j - x_0\|) \\
&= \sum_{j=1}^{n} \left(l_{y_j} - l(x_0) \sum_{i=1}^{p} l^{(x_i)}(x_0)(x_{ji} - x_{0i})\right)^2 K_h(\|x_j - x_0\|) \\
&+ \sum_{j=1}^{n} \left(a_{y_j} - a(x_0) \sum_{i=1}^{p} a^{(x_i)}(x_0)(x_{ji} - x_{0i})\right)^2 K_h(\|x_j - x_0\|) \\
&+ \sum_{j=1}^{n} \left(r_{y_j} - r(x_0) \sum_{i=1}^{p} r^{(x_i)}(x_0)(x_{ji} - x_{0i})\right)^2 K_h(\|x_j - x_0\|)
\end{aligned} \tag{14}$$

with respect to $l(x_0)$, $a(x_0)$ and $r(x_0)$ and $l^{(x_i)}(x_0)$, $a^{(x_i)}(x_0)$ and $r^{(x_i)}(x_0)$ $(i = 1, \ldots, p)$. $k_h(\cdot)$ is kernel function with the smoothing parameter h, where

$$K_h(\|x_j - x_0\|) = \frac{K\left(\dfrac{\|x_j - x_0\|}{h}\right)}{h}, \quad j = 1, \ldots, n \tag{15}$$

is a sequence of weights at x_0 and $\|x_j - x_0\|$ is Euclidean distance between x_j and x_0. Between many types of kernel functions, Epanechnikov's kernel (k_1)

and Gaussion kernel (k_2) have been used more. These kernels can be defined, respectively, by the following:

$$k_1(x) = \begin{cases} 0.75(1-x^2) & \text{, if } |x| \leq 1, \\ 0 & \text{otherwise,} \end{cases} \tag{16}$$

and

$$k_2(x) = (2\pi)^{-\frac{1}{2}} \exp\left(-\frac{x^2}{2}\right). \tag{17}$$

The degree of smoothness can be improved to estimate $\hat{a}(x)$, $\hat{l}(x)$ and $\hat{r}(x)$ by using the role of smoothing parameter h in function $k_h(\cdot)$. So, $\hat{f}(x_0) = \left(\hat{l}(x_0), \hat{a}(x_0), \hat{r}(x_0)\right)$ can be obtained by solving weighted least-squares problem (14). Therefore, taking the partial derivatives of the objective function (14) with respect to these unknown parameters to be zero, these linear equations can be separately solved to obtain the estimates of these parameters. According to the principle of the weighted leastsquares and using matrix notations, can be obtained:

$$\begin{aligned} &\left(\hat{l}(x_0), l^{(x_1)}(x_0), \ldots, l^{(x_p)}(x_0)\right)^T \\ &= \left(X^T(x_0)W(x_0;h)X(x_0)\right)^{-1} X^T(x_0)W(x_0;h)L_Y, \end{aligned} \tag{18}$$

$$\begin{aligned} &\left(\hat{a}(x_0), a^{(x_1)}(x_0), \ldots, a^{(x_p)}(x_0)\right)^T \\ &= \left(X^T(x_0)W(x_0;h)X(x_0)\right)^{-1} X^T(x_0)W(x_0;h)A_Y, \end{aligned} \tag{19}$$

$$\begin{aligned} &\left(\hat{r}(x_0), r^{(x_1)}(x_0), \ldots, r^{(x_p)}(x_0)\right)^T \\ &= \left(X^T(x_0)W(x_0;h)X(x_0)\right)^{-1} X^T(x_0)W(x_0;h)R_Y, \end{aligned} \tag{20}$$

$$X(x_0) = \begin{cases} 1 & x_{11} - x_{01} - \cdots - x_{1p} - x_{0p} \\ 1 & x_{21} - x_{01} - \cdots - x_{2p} - x_{0p} \\ \vdots & \\ 1 & x_{n1} - x_{01} - \cdots - x_{np} - x_{0p} \end{cases} \tag{21}$$

$$L_Y = \begin{pmatrix} l_{y_1} \\ l_{y_2} \\ \vdots \\ l_{y_n} \end{pmatrix}, \quad A_Y = \begin{pmatrix} a_{y_1} \\ a_{y_2} \\ \vdots \\ a_{y_n} \end{pmatrix}, \quad R_Y = \begin{pmatrix} r_{y_1} \\ r_{y_2} \\ \vdots \\ r_{y_n} \end{pmatrix} \tag{22}$$

and

$$W(x_0;h) = \mathrm{Diag}\Big(K_h(\|x_1 - x_0\|), K_h(\|x_2 - x_0\|), \ldots, K_h(\|x_n - x_0\|)\Big) \tag{23}$$

Therefore, $W(x_0;h)$ is a diagonal matrix with its diagonal elements being $K_h(\|x_j - x_0\|)$ $(j = 1, \ldots, n)$. So, the estimated regression function $\hat{f}(x)$ in x_0

can be expressed by:

$$\hat{f}(x_0) = \Big(\hat{l}(x_0), \hat{a}(x_0), \hat{r}(x_0)\Big) = \Big(e_1^T H(x_0;h)L_Y, e_1^T H(x_0;h)A_Y, e_1^T H(x_0;h)R_Y\Big) \tag{24}$$

where

$$e_1^T = (1, 0, \ldots, 0)^T,$$

and

$$H(x_0;h) = \Big(X^T(x_0)W(x_0;h)X(x_0)\Big)^{-1} X^T(x_0)W(x_0;h). \tag{25}$$

In addition, the symbol "T" indicates the transpose of a matrix. More details are provided in [2].

2.5 Smoothing parameter selection

There are a few approaches for selecting the optimal value of smoothing parameter (see, e.g. [11,15,16]). For selecting the optimal value of smoothing parameter, fuzzified cross-validation method is used [43]. The fuzzified cross-validation method is expressed as:

$$\begin{aligned} CV(h) &= \frac{1}{n}\sum_{j=1}^{n} d^2\Big(Y_j, \hat{f}_{(j)}(x_j;h)\Big) \\ &= \frac{1}{n}\sum_{j=1}^{n} \Big(l_{y_j}, \hat{l}_{(j)}(x_j;h)\Big)^2 + \Big(a_{y_j}, \hat{a}_{(j)}(x_j;h)\Big)^2 + \Big(r_{y_j}, \hat{r}_{(j)}(x_j;h)\Big)^2 \end{aligned} \tag{26}$$

so that

$$\hat{f}_{(j)}(x_j;h) = \Big(\hat{l}_{(j)}(x_j;h), \hat{a}_{(j)}(x_j;h), \hat{r}_{(j)}(x_j;h)\Big) \tag{27}$$

be the predicted fuzzy regression function at input x_j with the smoothing parameter h. In which, the j^{th} observation (x_j, Y_j) is omitted in the process of performing the fitting procedure and is computed $\hat{f}_{(j)}(x_j;h)$ for each element x_j $(j = 1, \ldots, n)$. Choose as the optimal value such that

$$CV(h) = \min_{h>0} CV(h). \tag{28}$$

In practice, we may compute $CV(h)$ for a series of values of h to search for. For more details, see [12].

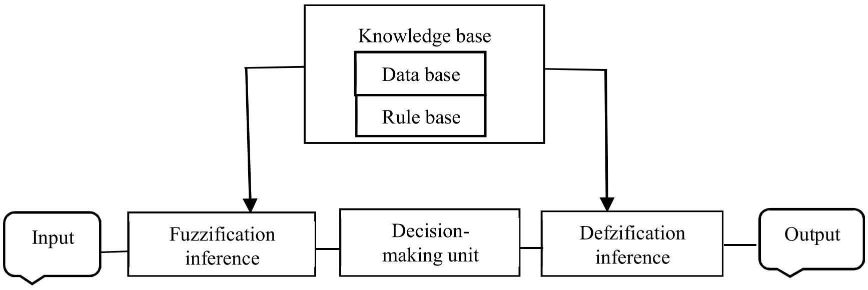

Fig. 1: Fuzzy inference system.

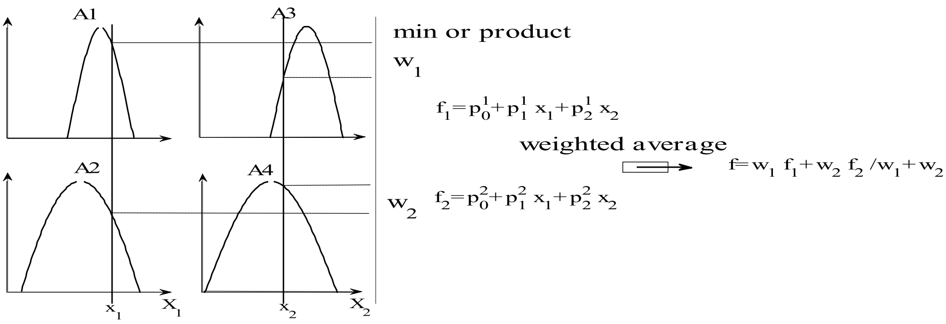

Fig. 2: The reasoning scheme.

2.6 Adaptive neuro-fuzzy inference system (ANFIS)

The ANFIS method is a famous hybrid neuro-fuzzy network for modeling complex systems [20, 21]. It is one of the best tradeoffs between neural and fuzzy systems that provides smoothness due to the fuzzy logic (FL) interpolation and adaptability due to the NN backpropagation. Hence, the advantages of a fuzzy system can be combined with a learning algorithm. The fuzzy inference system forms useful computing based on concepts of fuzzy if-then rules. A fuzzy inference system consists of three components that are shown in Fig. 1 [42]. Firstly, a rule base contains a selection of fuzzy rules. Secondly, a database defines the membership functions used in the rules and, finally, a reasoning mechanism to carry out the inference procedure on the rules and given facts. fig 2 shows reasoning procedures of derive conclusion based on information aggregation from all the rules. The steps of fuzzy reasoning performed by fuzzy inference systems are:

The first step is called fuzzification. The input variables compare with the membership functions on the premise part to obtain the membership values of each linguistic label.

The second step combines the membership values on the premise part to get firing. To present ANFIS architecture, two fuzzy if-then rules based on Takagi and Sugeno model are considered with two input variables and one

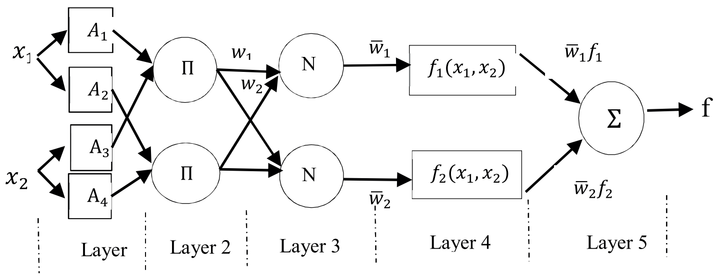

Fig. 3: ANFIS architecture.

output y.

$$\text{Rule 1 : IF } x_1 \text{ is } A_1 \text{ and } x_2 \text{ is } A_3 \text{ then } f_1 = p_0^1 + p_1^1 x_1 + p_2^1 x_2. \quad (29)$$

$$\text{Rule 1 : IF } x_1 \text{ is } A_2 \text{ and } x_2 \text{ is } A_4 \text{ then } f_2 = p_0^2 + p_1^2 x_1 + p_2^2 x_2. \quad (30)$$

The ANFIS model has five levels that the layers' architecture are shown in Fig 3, in which a circle indicates a fixed node without parameters and a square indicates an adaptive node with parameters. where x_1, x_2 and $y \in R$ are input and output variables, respectively. A_r's are fuzzy sets and f_k represents system output due to rule R_k ($k = 1, 2,\ r = 1, \ldots, 4$). In the following, the five layers of the system are explained that have two-dimensional input and one output. In the first layer, all the nodes are adaptive nodes. They generate membership grades of the inputs. The node functions are given by:

$$o_{1,k} = \mu_{A_r}(x_1) \ , \qquad k, r = 1, 2, \quad (31)$$

$$o_{1,k} = \mu_{A_r}(x_2) \ , \qquad k, r = 3, 4, \quad (32)$$

where x_1, x_2 are inputs and μ_{A_r}'s are appropriate membership functions. ol,k is the output of the k^{th} node of the layer l. In the second layer, the nodes are also fixed which multiply the base of the inputs on incoming the output of the first layer and send the product out. The outputs of this layer can be calculated as:

$$o_{2,k} = w_k = \mu_{A_r}(x_1) \cdot \mu_{A_r}(x_2) \ , \qquad r = 1, \ldots, 4. \quad (33)$$

In the third layer, the nodes are fixed nodes. It calculates the ratio of a rule's firing of all the rules. Of the output of this layer can be calculated as:

$$o_{3,k} = \overline{w}_k = \frac{w_k}{w_1 + w_2} \ , \qquad k = 1, 2. \quad (34)$$

This is called normalized firing strength.

In the fourth layer, node is an adaptive node. The node function associated in the level 4 is a linear function. The output of this layer can be represented as below:

$$o_{4,k} = \overline{w}_k f_k = \overline{w}_k (p_0^k + p_1^k x_1 + p_2^k x_2) \ , \qquad k = 1, 2, \quad (35)$$

In this work, p_i^k will be assumed to be a triangular fuzzy number for $k = 1, 2$ and $i = 0, 1, 2$.

In the fifth layer, the single node carries out the sum of inputs of all the layers. The overall output of the structure is expressed as:

$$o_{5,k} = \sum_{k=1}^{n} \overline{w}_k f_k = \overline{w}_1 f_1 + \overline{w}_2 f_2. \tag{36}$$

By substituting the fuzzy if-then rules in to Eq. (36), the following can be obtained:

$$\begin{aligned} o_{5,k} = \overline{Y} &= \overline{w}_1(p_0^1 + p_1^1 x_1 + p_2^1 x_2) + \overline{w}_2(p_0^2 + p_1^2 x_1 + p_2^2 x_2) \\ &= \sum_{k=1}^{2}(\overline{w}_k p_0^k) + \sum_{k=1}^{2}\sum_{i=1}^{2}(\overline{w}_k p_i^k) x_i. \end{aligned} \tag{37}$$

This equation is the same form as the following linear equation:

$$Y = p_0 + p_1 x_1 + p_2 x_2, \tag{38}$$

that the parameters p_i $(i = 0, 1, 2)$ are fuzzy numbers. Therefore, the estimated output value Y is the fuzzy number. What should be understood when reviewing the above layers are mainly three different types of components that can be adapted as follows [22]:

1. Premise parameters are nonlinear parameters in the input membership functions.
2. Consequent parameters are as linear parameters in the rules of consequents (output weights).
3. Rule structure needs to be optimized to achieve a better linguistic interpretability.

In the following, the ANFIS method is extended with applications to fuzzy nonparametric regression.

3 Extension of the ANFIS method in fuzzy regression

In Eq. (37), assume that consequence parameter p_i^k is a symmetric triangular fuzzy number and is represented by $p_i^k = (b_i^k, \alpha_i^k)$, $i = 0, \ldots, p$, $k = 1, \ldots, m$. Also, Y_j and $\hat{Y}_j$ .. are symmetric triangular fuzzy numbers. They are represented by $Y_j = (a_j, \beta_j)$ and $\hat{Y}_j = (\hat{a}_j, \hat{\beta}_j)$, $j = 1, \ldots, n$, where n is the number of data points, a_j is the center value and β_j is the spread value of Y_j, and $\hat{a}_j$ is the center value and $\hat{\beta}_j$ is the spread value of $\hat{Y}_j$. From the above definitions, using fuzzy arithmetic and by substituting p_i^k into Eq. (37), the output $\hat{Y}$, for two inputs x_1 and x_2, can be expressed as:

$$\begin{aligned} \hat{Y} &= (b_0^1, \alpha_0^1)\overline{w}_1 + (b_1^1, \alpha_1^1)\overline{w}_1 x_1 + (b_2^1, \alpha_2^1)\overline{w}_1 x_2 \\ &+ (b_0^2, \alpha_0^2)\overline{w}_2 + (b_1^2, \alpha_1^2)\overline{w}_2 x_1 + (b_2^2, \alpha_2^2)\overline{w}_2 x_2 \\ &= \sum_{k=1}^{2}\sum_{i=0}^{2} b_i^k \overline{w}_k x_i + \sum_{k=1}^{2}\sum_{i=0}^{2} \alpha_i^k \overline{w}_k x_i. \end{aligned} \tag{39}$$

where $\overline{w}_k$ is known.

Consider the following the fuzzy regression model as:

$$Y_j = p_0 + p_1 x_{j1} + p_2 x_{j2} + \cdots + p_p x_{jp} = PX_j \ , \quad j = 1, \ldots, n, \tag{40}$$

where n is the number of data points, $x_j = (1, x_{j1}, x_{j2}, \ldots, x_{jp})$ is an p-dimensional input vector of values of the independent variables at the j^{th} observations, also, $P = (p_0, p_1, \ldots, p_p)$ is vector of unknown fuzzy parameters and Y_j is the j^{th} observed value of the dependent variables. P can be denoted in vector form as $P = \{b, \alpha\}$ where $b = (b_0^k, b_1^k, \ldots, b_p^k)$ and $\alpha = (\alpha_0^k, \alpha_1^k, \ldots, \alpha_p^k)$, $k = 1, \ldots, m$ where b_i^k is the center value, and α_i^k is the spread value of p_i, $i = 0, \ldots, p$. So, from the above definitions, by using fuzzy arithmetic and with Eq. (39), $\hat{a}_j$ and $\hat{\beta}_j$ can be expressed as:

$$\hat{a}_j \sum_{k=1}^{m} \sum_{i=0}^{p} b_i^k \overline{w}_k x_{ji}, \tag{41}$$

and

$$\hat{\beta}_j \sum_{k=1}^{m} \sum_{i=0}^{p} \alpha_i^k \overline{w}_k x_{ji}. \tag{42}$$

3.1 Methodology of the proposed method

The learning algorithm of the ANFIS method is a hybrid algorithm. It combines Least-squares method (forward pass) and the gradient descent method (backward pass) to reduce error (For more details, see [20,21]). In this paper, the fuzzy Least-squares method is used to optimize the consequent parameters and the gradient descent method is used to compute the hider layers (2, 3, 4).

3.2 Premise parameters

By using the back propagation, the training algorithm of the premise parameters is updated. The purpose of training the premise parameters is to optimize the adjustment of the position and the shape of the associated membership function. The center of the observed fuzzy outputs are trained and

$$e = \sum_{j=1}^{n} e_j^2 = \sum_{j=1}^{n} (a_j - \hat{a}_j)^2, \tag{43}$$

is used for calculating the back propagation error. The influences of the spread are ignored. The obtained optimal premise parameters are applied to obtain a consequence parameters using the fuzzy least squares problem based on Diamond's distance that will explain in the following.

So, the back propagation error for each layer is [3,20]:

$$e_{l,k} = \sum_{r=1}^{M_{l+1}} e_{l+1,r} \frac{\partial A_{l+1,r}}{\partial o_{l,k}} \tag{44}$$

$e_{l,k}$ is the back propagation error of the k^{th} node of the layer l. $A_{l+1,r}$, is the node function of the r^{th} node of $(l+1)^{th}$ layer, $o_{l,k}$ represents the output k^{th} node of the layer l and M_{l+1} is the total number of nodes in the $(l+1)^{th}$ layer. So, error of the final output node is calculated as:

$$e_{5,1} = \frac{e_j^2}{\partial \hat{y}_j} = -(a_j - \hat{a}_j). \tag{45}$$

So, the gradient vector is defined as the error measure derivatives with respect to each parameter. The derivative of the overall error measure e with respect to parameter δ is:

$$\frac{\partial e}{\partial \delta} = \frac{1}{n} \sum_{j=1}^{n} \frac{\partial e_j^2}{\partial \delta} = \sum_{j=1}^{n} e_{l,k} \frac{\partial o_{l,k}}{\partial \delta}. \tag{46}$$

Therefore, the updating formula for δ is:

$$\delta = \vartheta \frac{\partial e}{\partial \delta}, \tag{47}$$

where ϑ is the learning rate.

3.3 Consequence parameters

In this section, consequence parameters are derived using the fuzzy least squares problem. First, they are derived for the bivariate fuzzy nonparametric regression model. Then, the derivations are extended to a multiple fuzzy nonparametric regression model.

3.4 Bivariate fuzzy nonparametric regression model

In the fuzzy regression model Eq. (3), the error measure is defined as:

$$e_j = Y_j \{-\} \hat{Y}_j, \tag{48}$$

where Y_j is the j^{th} output, $\hat{Y}_j$ is the network output of the j^{th} input, $x = (1, x_1, x_2, \ldots, x_n)$ and $\{-\}$ is an operator, whose definition depends on the fuzzy ranking method used. This performance measure is based on the calculation of the distance or difference between two fuzzy numbers. Kim and Bishu [51] pointed out that the membership function of an estimated fuzzy output must be close to the observed output as possible. There are various distances for obtaining the closeness between fuzzy numbers. The individual difference e_j can be obtained using Diamond's distance as a measure of the fit. In this paper, we optimize the consequent parameters for crisp inputs and symmetric fuzzy outputs using the fuzzy least squares based on Diamond's distance. Let the observed values $Y_j = (l_{y_j}, a_{y_j}, r_{y_j})$ and the predicted values $\hat{Y}_j = (\hat{l}_{y_j}, \hat{a}_{y_j}, \hat{r}_{y_j})$ are asymmetric triangular fuzzy numbers $(j = 1, \ldots, n)$. Where l_{y_j}, a_{y_j} and r_{y_j} are the lower, the center, and the upper limits of the observed fuzzy outputs. Also, $\hat{l}_{y_j}$, $\hat{a}_{y_j}$ and $\hat{r}_{y_j}$ are the lower, the center, and the upper limits of the predicted fuzzy outputs. In the following, the fuzzy leastsquares problem based on Diamond 'distance is formulated. That is, minimize

$$\sum_{j=1}^{n}(Y_j - \hat{Y}_j)^2 = \sum_{j=1}^{n}\Big((l_{y_j} - \hat{l}_{y_j})^2 + (a_{y_j} - \hat{a}_{y_j})^2 + (r_{y_j} - \hat{r}_{y_j})^2\Big) \tag{49}$$

with respect $\hat{l}_{y_j}$, $\hat{a}_{y_j}$ and $\hat{r}_{y_j}$ Suppose $Y_j = (a_{y_j}, \beta_{y_j})$ and $Y_j = (\hat{a}_{y_j}, \hat{\beta}_{y_j})$ are two symmetric fuzzy numbers, where a_{y_j} and β_{y_j} are the center, and the spread of the observed fuzzy outputs, $\hat{a}_{y_j}$ and $\hat{\beta}_{y_j}$ are the estimated center, and spread of the predicted fuzzy outputs, $l_{y_j} = a_{y_j} - \beta_{y_j}$, $r_{y_j} = a_{y_j} + \beta_{y_j}$, $\hat{l}_{y_j} = \hat{a}_{y_j} - \hat{\beta}_{y_j}$, and $\hat{r}_{y_j} = \hat{a}_{y_j} + \hat{\beta}_{y_j}$. By substituting $\hat{l}_{y_j}$, $\hat{a}_{y_j}$ and $\hat{r}_{y_j}$ in Eq. (49), the fuzzy least squares problem can be rewritten as:

$$\begin{aligned}
&\text{ERROR} = \sum_{j=1}^{n}(Y_j - \hat{Y}_j)^2 \\
&= \sum_{j=1}^{n}\Big((a_{y_j} - \beta_{y_j})(\hat{a}_{y_j} - \hat{\beta}_{y_j})\Big)^2 + (a_{y_j} - \hat{a}_{y_j})^2 + \Big((a_{y_j} + \beta_{y_j})(\hat{a}_{y_j} + \hat{\beta}_{y_j})\Big)^2 \\
&= \sum_{j=1}^{n}\Big((a_{y_j} + \hat{a}_{y_j})^2 + 2(\beta_{y_j} + \hat{\beta}_{y_j})^2\Big) \\
&= \sum_{j=1}^{n}\left(3\left(a_{y_j} - \sum_{k=1}^{m}\sum_{i=0}^{1} b_i^k \overline{w}_{jk} x_{ji}\right)^2 + 2\left(\beta_{y_j} - \sum_{k=1}^{m}\sum_{i=0}^{1} \alpha_i^k \overline{w}_{jk} x_{ji}\right)^2\right)
\end{aligned} \tag{50}$$

where, $x_{j0} = 1$.

It is observed that the objective function in Eq. (50) is the summation of two parts with two different groups of unknown parameters. The consequent parameters can be determined by minimizing error function with respect to the unknown parameters b_i^k and α_i^k. In order to derive the error function with respect to the unknown parameters set the derivations to zero, and solve for the unknown parameters.

In the following, we derive Eq. (50) with respect b_i^k and α_i^k respectively.

$$\frac{\partial \text{ERROR}}{\partial b_0^1} = \sum_{j=1}^{n} 6\overline{w}_{j1} \left(a_{y_j} - \sum_{k=1}^{m} \sum_{i=0}^{1} b_i^k \overline{w}_{jk} x_{ji} \right) = 0 \tag{51}$$

$$\frac{\partial \text{ERROR}}{\partial b_1^1} = \sum_{j=1}^{n} 6\overline{w}_{j1} x_{j1} \left(a_{y_j} - \sum_{k=1}^{m} \sum_{i=0}^{1} b_i^k \overline{w}_{jk} x_{ji} \right) = 0 \tag{52}$$

$$\vdots$$

$$\frac{\partial \text{ERROR}}{\partial b_0^m} = \sum_{j=1}^{n} 6\overline{w}_{jm} \left(a_{y_j} - \sum_{k=1}^{m} \sum_{i=0}^{1} b_i^k \overline{w}_{jk} x_{ji} \right) = 0 \tag{53}$$

$$\frac{\partial \text{ERROR}}{\partial b_1^m} = \sum_{j=1}^{n} 6\overline{w}_{jm} x_{j1} \left(a_{y_j} - \sum_{k=1}^{m} \sum_{i=0}^{1} b_i^k \overline{w}_{jk} x_{ji} \right) = 0 \tag{54}$$

After rearranging the terms of the equations, Eqs. (51), (52), (53) and (54) can be written as follows:

$$\sum_{j=1}^{n} \sum_{k=1}^{m} \sum_{i=0}^{1} \overline{w}_{k1} b_i^k \overline{w}_{jk} x_{ji} = \sum_{j=1}^{n} \overline{w}_{j1} a_{y_j} \tag{55}$$

$$\sum_{j=1}^{n} \sum_{k=1}^{m} \sum_{i=0}^{1} \overline{w}_{j1} x_{j1} b_i^k \overline{w}_{jk} x_{ji} = \sum_{j=1}^{n} \overline{w}_{j1} x_{j1} a_{y_j} \tag{56}$$

$$\vdots$$

$$\sum_{j=1}^{n} \sum_{k=1}^{m} \sum_{i=0}^{1} \overline{w}_{jm} b_i^k \overline{w}_{jk} x_{ji} = \sum_{j=1}^{n} \overline{w}_{jm} a_{y_j} \tag{57}$$

$$\sum_{j=1}^{n} \sum_{k=1}^{m} \sum_{i=0}^{1} \overline{w}_{jm} x_{j1} b_i^k \overline{w}_{jk} x_{ji} = \sum_{j=1}^{n} \overline{w}_{jm} x_{j1} a_{y_j} \tag{58}$$

Eqs. (55), (56), (57) and (58) can be written as the following matrix:

$$
\begin{aligned}
&[b_0^1 \quad b_0^2 \quad \cdots \quad b_0^m \quad b_1^1 \quad b_1^2 \quad \cdots \quad b_1^n]
\begin{bmatrix}
\overline{w}_{11} & \overline{w}_{21} & \cdots & \overline{w}_{n1} \\
\overline{w}_{12} & \overline{w}_{22} & \cdots & \overline{w}_{n2} \\
\vdots & & & \\
\overline{w}_{1m} & \overline{w}_{2m} & \cdots & \overline{w}_{nm} \\
\overline{w}_{11}x_{11} & \overline{w}_{21}x_{21} & \cdots & \overline{w}_{n1}x_{n1} \\
\overline{w}_{12}x_{11} & \overline{w}_{22}x_{21} & \cdots & \overline{w}_{n2}x_{21} \\
\vdots & & & \\
\overline{w}_{1m}x_{11} & \overline{w}_{2m}x_{21} & \cdots & \overline{w}_{nm}x_{n1}
\end{bmatrix} \\
&\times \begin{bmatrix}
\overline{w}_{11}\overline{w}_{12}\cdots\overline{w}_{1m}\overline{w}_{11}x_{11}\cdots\overline{w}_{1m}x_{11} \\
\overline{w}_{21}\overline{w}_{22}\cdots\overline{w}_{2m}\overline{w}_{21}x_{21}\cdots\overline{w}_{2m}x_{21} \\
\vdots \\
\overline{w}_{n1}\overline{w}_{n2}\cdots\overline{w}_{nm}\overline{w}_{n1}x_{n1}\cdots\overline{w}_{nm}x_{n1}
\end{bmatrix} \\
&= \begin{bmatrix}
\overline{w}_{11} & \overline{w}_{21} & \cdots & \overline{w}_{n1} \\
\overline{w}_{12} & \overline{w}_{22} & \cdots & \overline{w}_{n2} \\
\vdots & & & \\
\overline{w}_{1m} & \overline{w}_{2m} & \cdots & \overline{w}_{nm} \\
\overline{w}_{11}x_{11} & \overline{w}_{21}x_{21} & \cdots & \overline{w}_{n1}x_{n1} \\
\overline{w}_{12}x_{11} & \overline{w}_{22}x_{21} & \cdots & \overline{w}_{n2}x_{21} \\
\vdots & & & \\
\overline{w}_{1m}x_{11} & \overline{w}_{2m}x_{21} & \cdots & \overline{w}_{nm}x_{n1}
\end{bmatrix}
\begin{bmatrix} a_{y_1} \\ a_{y_2} \\ \vdots \\ a_{y_n} \end{bmatrix}
\end{aligned} \tag{59}
$$

Also,

$$\frac{\partial \text{ERROR}}{\partial \alpha_0^1} = \sum_{j=1}^{n} 4\overline{w}_{j1} \left(\beta_{y_j} - \sum_{k=1}^{m}\sum_{i=0}^{1} \alpha_i^k \overline{w}_{jk} x_{ji} \right) = 0 \tag{60}$$

$$\frac{\partial \text{ERROR}}{\partial \alpha_1^1} = \sum_{j=1}^{n} 4\overline{w}_{j1} x_{j1} \left(\beta_{y_j} - \sum_{k=1}^{m}\sum_{i=0}^{1} \alpha_i^k \overline{w}_{jk} x_{ji} \right) = 0 \tag{61}$$

$\vdots$

$$\frac{\partial \text{ERROR}}{\partial \alpha_0^m} = \sum_{j=1}^{n} 4\overline{w}_{jm} \left(\beta_{y_j} - \sum_{k=1}^{m}\sum_{i=0}^{1} \alpha_i^k \overline{w}_{jk} x_{ji} \right) = 0 \tag{62}$$

$$\frac{\partial \text{ERROR}}{\partial \alpha_1^m} = \sum_{j=1}^{n} 4\overline{w}_{jm} x_{j1} \left(\beta_{y_j} - \sum_{k=1}^{m}\sum_{i=0}^{1} \alpha_i^k \overline{w}_{jk} x_{ji} \right) = 0 \tag{63}$$

After rearranging the terms of the equations, Eqs. (60), (61), (62) and (63) can be written as follows:

$$\sum_{j=1}^{n}\sum_{k=1}^{m}\sum_{i=0}^{1}\overline{w}_{k1}\alpha_i^k\overline{w}_{jk}x_{ji}=\sum_{j=1}^{n}\overline{w}_{j1}\beta_{y_j} \tag{64}$$

$$\sum_{j=1}^{n}\sum_{k=1}^{m}\sum_{i=0}^{1}\overline{w}_{j1}x_{j1}\alpha_i^k\overline{w}_{jk}x_{ji}=\sum_{j=1}^{n}\overline{w}_{j1}x_{j1}\beta_{y_j} \tag{65}$$

$$\vdots$$

$$\sum_{j=1}^{n}\sum_{k=1}^{m}\sum_{i=0}^{1}\overline{w}_{jm}\alpha_i^k\overline{w}_{jk}x_{ji}=\sum_{j=1}^{n}\overline{w}_{jm}\beta_{y_j} \tag{66}$$

$$\sum_{j=1}^{n}\sum_{k=1}^{m}\sum_{i=0}^{1}\overline{w}_{jm}x_{j1}\alpha_i^k\overline{w}_{jk}x_{ji}=\sum_{j=1}^{n}\overline{w}_{jm}x_{j1}\beta_{y_j} \tag{67}$$

Eqs. (64), (65), (66) and (67) can be written as the following matrix:

$$\begin{aligned}
&[\alpha_0^1 \quad \alpha_0^2 \quad \cdots \quad \alpha_0^m \quad \alpha_1^1 \quad \alpha_1^2 \quad \cdots \quad \alpha_1^n]\begin{bmatrix} \overline{w}_{11} & \overline{w}_{21} & \cdots & \overline{w}_{n1} \\ \overline{w}_{12} & \overline{w}_{22} & \cdots & \overline{w}_{n2} \\ \vdots & & & \\ \overline{w}_{1m} & \overline{w}_{2m} & \cdots & \overline{w}_{nm} \\ \overline{w}_{11}x_{11} & \overline{w}_{21}x_{21} & \cdots & \overline{w}_{n1}x_{n1} \\ \overline{w}_{12}x_{11} & \overline{w}_{22}x_{21} & \cdots & \overline{w}_{n2}x_{21} \\ \vdots & & & \\ \overline{w}_{1m}x_{11} & \overline{w}_{2m}x_{21} & \cdots & \overline{w}_{nm}x_{n1} \end{bmatrix} \\
&\times \begin{bmatrix} \overline{w}_{11}\overline{w}_{12}\cdots\overline{w}_{1m}\overline{w}_{11}x_{11}\cdots\overline{w}_{1m}x_{11} \\ \overline{w}_{21}\overline{w}_{22}\cdots\overline{w}_{2m}\overline{w}_{21}x_{21}\cdots\overline{w}_{2m}x_{21} \\ \vdots \\ \overline{w}_{n1}\overline{w}_{n2}\cdots\overline{w}_{nm}\overline{w}_{n1}x_{n1}\cdots\overline{w}_{nm}x_{n1} \end{bmatrix} \\
&= \begin{bmatrix} \overline{w}_{11} & \overline{w}_{21} & \cdots & \overline{w}_{n1} \\ \overline{w}_{12} & \overline{w}_{22} & \cdots & \overline{w}_{n2} \\ \vdots & & & \\ \overline{w}_{1m} & \overline{w}_{2m} & \cdots & \overline{w}_{nm} \\ \overline{w}_{11}x_{11} & \overline{w}_{21}x_{21} & \cdots & \overline{w}_{n1}x_{n1} \\ \overline{w}_{12}x_{11} & \overline{w}_{22}x_{21} & \cdots & \overline{w}_{n2}x_{21} \\ \vdots & & & \\ \overline{w}_{1m}x_{11} & \overline{w}_{2m}x_{21} & \cdots & \overline{w}_{nm}x_{n1} \end{bmatrix}\begin{bmatrix} \beta_{y_1} \\ \beta_{y_2} \\ \vdots \\ \beta_{y_n} \end{bmatrix}
\end{aligned} \tag{68}$$

By solving these two groups of linear equations, these parameters estimation can be obtained for the fuzzy regression model as follows:

$$(\hat{b}_i^k)^T=(X^TX)^{-1}X^TA_Y, \tag{69}$$

$$(\hat{\alpha}_i^k)^T=(X^TX)^{-1}X^T\beta_Y. \tag{70}$$

where,

$$X = \begin{bmatrix} \overline{w}_{11}\overline{w}_{12}\cdots\overline{w}_{1m}\overline{w}_{11}x_{11}\cdots\overline{w}_{1m}x_{11} \\ \overline{w}_{21}\overline{w}_{22}\cdots\overline{w}_{2m}\overline{w}_{21}x_{21}\cdots\overline{w}_{2m}x_{21} \\ \vdots \\ \overline{w}_{n1}\overline{w}_{n2}\cdots\overline{w}_{nm}\overline{w}_{n1}x_{n1}\cdots\overline{w}_{nm}x_{n1} \end{bmatrix}$$

$$A_Y = \begin{bmatrix} a_{y_1} \\ a_{y_2} \\ \vdots \\ a_{y_n} \end{bmatrix}, \ \alpha_Y = \begin{bmatrix} \beta_{y_1} \\ \beta_{y_2} \\ \vdots \\ \beta_{y_n} \end{bmatrix}, \ (\hat{b}_i^k)^T = \begin{bmatrix} \hat{b}_0^1 \\ \hat{b}_0^m \\ \vdots \\ \hat{b}_1^1 \\ \hat{b}_1^m \end{bmatrix}, \ (\hat{\alpha}_i^k)^T = \begin{bmatrix} \hat{\alpha}_0^1 \\ \hat{\alpha}_0^m \\ \vdots \\ \hat{\alpha}_1^1 \\ \hat{\alpha}_1^m \end{bmatrix} \quad (71)$$

And $x_{j0} = 1$, the symbol "T" is the mean transpose of a matrix.

In the proposed method, the consequence parameters are obtained by Eqs. (69) and (70). So, Ishibushi and Tanaka [19] have suggested that the support of the estimated values from the regression model includes the support of the observed values in α-level $(0 \leq \alpha \leq 1)$.

Also, for this propose, one of the following two constraints must be established:

$$\sum_{k=1}^{m}\sum_{i=0}^{1} b_i^k \overline{w}_{jk} x_{ji} - (1-\alpha)\sum_{k=1}^{m}\sum_{i=0}^{1} \alpha_i^k \overline{w}_{jk} x_{ji} \leq a_{y_j} + (1-\alpha)\beta_{y_j}, \quad (72)$$

or

$$\sum_{k=1}^{m}\sum_{i=0}^{1} b_i^k \overline{w}_{jk} x_{ji} - (1-\alpha)\sum_{k=1}^{m}\sum_{i=0}^{1} \alpha_i^k \overline{w}_{jk} x_{ji} \leq a_{y_j} - (1-\alpha)\beta_{y_j},$$

and

$$\sum_{k=1}^{m} \alpha_i^k \overline{w}_{jk} \geq 0 , \qquad i = 0, 1 , \quad j = 1, \ldots, n$$

The aforementioned constraints are applied for the model parameters estimation in the learning algorithm of the proposed model.

After obtaining the premise and consequence parameters $\hat{Y}_j$ can be obtained as follows:

$$\hat{Y}_j = (\hat{a}_j, \hat{\beta}_j) = \sum_{k=1}^{m} b_0^k \overline{w}_{jk} + \left(\sum_{k=1}^{m} b_1^k \overline{w}_{jk}\right) x_{j1} + \sum_{k=1}^{m} \alpha_0^k \overline{w}_{jk} + \left(\sum_{k=1}^{m} \alpha_1^k \overline{w}_{jk}\right) x_{j1}. \quad (73)$$

3.5 Extension to multivariate input

The above discussions only focus on the case of univariate input. It is straightforward to the extent the proposed method to the case of multivariate input. In fact, let $x_j = (1, x_{j1}, x_{j2}, \ldots, x_{jp})$ is the j^{th} input vector and $Y_j = (a_{y_j}, \beta_{y_j})$ and $\hat{Y}_j = (\hat{a}_{y_j}, \hat{\beta}_{y_j})$ are symmetric triangular fuzzy output. Based on Diamond distance and by using Eq. refeq50, the following weighted least-squares problem is formulated. That is, minimize

$$\begin{aligned} \text{ERROR} &= \sum_{j=1}^{n} (Y_j - \hat{Y}_j)^2 \\ &= \sum_{j=1}^{n} \left(3 \left(a_{y_j} - \sum_{k=1}^{m} \sum_{i=0}^{p} b_i^k \overline{w}_{jk} x_{ji} \right)^2 + 2 \left(\beta_{y_j} - \sum_{k=1}^{m} \sum_{i=0}^{p} \alpha_i^k \overline{w}_{jk} x_{ji} \right)^2 \right) \end{aligned} \tag{74}$$

with respect to the unknown parameters b_i^k and α_i^k. In order to derive the error function with respect to the unknown parameters, set the derivations to zero and solve for the unknown parameters. The unknown parameters estimation b_i^k and α_i^k can be expressed by the same expressions in Eqs. (69) and (70) with X, $(\hat{b}_i^k)^T$ and $(\hat{\alpha}_i^k)^T$ in (69) and (70), respectively replaced by:

$$X = \begin{bmatrix} \overline{w}_{11}\overline{w}_{12}\cdots\overline{w}_{1m}\overline{w}_{11}x_{11}\cdots\overline{w}_{1m}x_{11}\cdots\overline{w}_{11}x_{1p}\cdots\overline{w}_{1m}x_{1p} \\ \overline{w}_{21}\overline{w}_{22}\cdots\overline{w}_{2m}\overline{w}_{21}x_{21}\cdots\overline{w}_{2m}x_{21}\cdots\overline{w}_{21}x_{2p}\cdots\overline{w}_{2m}x_{2p} \\ \\ \vdots \\ \overline{w}_{n1}\overline{w}_{n2}\cdots\overline{w}_{nm}\overline{w}_{n1}x_{n1}\cdots\overline{w}_{nm}x_{n1}\cdots\overline{w}_{n1}x_{np}\cdots\overline{w}_{nm}x_{np} \end{bmatrix}$$

and

$$(\hat{b}_i^k)^T = \begin{bmatrix} \hat{b}_0^1 \\ \hat{b}_0^m \\ \vdots \\ \hat{b}_p^1 \\ \hat{b}_p^m \end{bmatrix}, \quad (\hat{\alpha}_i^k)^T = \begin{bmatrix} \hat{\alpha}_0^1 \\ \hat{\alpha}_0^m \\ \vdots \\ \hat{\alpha}_p^1 \\ \hat{\alpha}_p^m \end{bmatrix} \tag{75}$$

So, one of the following two constraints must be established:

$$\sum_{k=1}^{m} \sum_{i=0}^{1} b_i^k \overline{w}_{jk} x_{ji} - (1-\alpha) \sum_{k=1}^{m} \sum_{i=0}^{p} \alpha_i^k \overline{w}_{jk} x_{ji} \leq a_{y_j} + (1-\alpha)\beta_{y_j}, \tag{76}$$

or

$$\sum_{k=1}^{m} \sum_{i=0}^{p} b_i^k \overline{w}_{jk} x_{ji} - (1-\alpha) \sum_{k=1}^{m} \sum_{i=0}^{p} \alpha_i^k \overline{w}_{jk} x_{ji} \leq a_{y_j} - (1-\alpha)\beta_{y_j},$$

and

$$\sum_{k=1}^{m} \alpha_i^k \overline{w}_{jk} \geq 0 \ , \qquad i = 0, 1, \ldots, p \ , \quad j = 1, \ldots, n$$

After obtaining the premise and consequence parameters, $\hat{Y}_j = (\hat{a}_j, \hat{\beta}_j)$ can be calculated as follows:

$$\begin{aligned}
&\hat{Y}_j = (\hat{a}_j, \hat{\beta}_j) \\
&= \sum_{k=1}^{m} b_0^k \overline{w}_{jk} + \left(\sum_{k=1}^{m} b_1^k \overline{w}_{jk} \right) x_{j1} + \cdots + \left(\sum_{k=1}^{m} b_p^k \overline{w}_{jk} \right) x_{jp} + \sum_{k=1}^{m} \alpha_0^k \overline{w}_{jk} \\
&+ \cdots + \left(\sum_{k=1}^{m} \alpha_1^k \overline{w}_{jk} \right) x_{j1} + \cdots + \left(\sum_{k=1}^{m} \alpha_p^k \overline{w}_{jk} \right) x_{jp} \\
&= \sum_{k=1}^{m} \sum_{i=0}^{p} b_p^k \overline{w}_{jk} x_{ji} + \sum_{k=1}^{m} \sum_{i=0}^{p} b_p^k \overline{w}_{jk} x_{ji} \ , \quad x_{j0} = 1. \qquad (77)
\end{aligned}$$

Thus, by using the Diamond's distance, the final error function is defined as:

$$\text{ERROR} = \frac{1}{n} \sum_{j=1}^{n} \left(3(a_j - \hat{a}_j)^2 + 2(\beta_j - \hat{\beta}_j)^2 \right) \qquad (78)$$

where n number is the pairs of training data.

The proposed method for the fuzzy regression function prediction may be summarized in Fig. 4. MATLAB software is used for codding.

3.6 V-fold cross validation technique

V-fold cross-validation technique is used to estimate the quality of the proposed method [41,43,52]. This technique divides all the samples into V equal-size groups (if possible), called folds. V-1 folds are used for training, and the fold left out is used for testing. After training V folds, the average errors of test across all V training are calculated and the results of the best net will be shown in the output. V-fold validation technique reduces variability by averaging over V different partitions.

3.7 Modelling Performance Criterion

A quantity is defined to measure bias between the observed values, $Y_j = (l_j, a_j, r_j)$, and the predicted values, $\hat{Y}_j = (\hat{l}_j, \hat{a}_j, \hat{r}_j)$, for all $X_j s$ $(j = 1, \ldots, n)$ where l_j, a_j, r_j, $\hat{l}_j$, $\hat{a}_j$ and $\hat{r}_j$ are the lower, the center, and the upper limits of the observed fuzzy outputs and the lower, the center, and the upper limits of the estimated fuzzy regression function. Therefore, the goodness of fit error

(GOF) rate based on Diamond's distance (2), that is the same error function ERROR, can be defined as [51]:

$$\mathrm{GOF} = \frac{1}{n}\sum_{j=1}^{n} d^2(Y_j, \hat{f}_j(x_j)) = \frac{1}{n}\sum_{j=1}^{n}\Big((l_j-\hat{l}_j)^2+(a_j-\hat{a}_j)^2+(r_j-\hat{r}_j)^2\Big) \quad (79)$$

where n is the number of pairs of observations. The large value of this quantity indicates lack-of-fit and too small value reflects over-fit for the observed fuzzy outputs. However, GOF value cannot efficiently reflect the closeness between the underlying fuzzy nonparametric regression function $f(x)$ and its estimate because of the error term in model Eq. (3). Thus, a quantity is used for measuring the bias between the underlying fuzzy regression function and its estimate which is called BIAS. It can be expressed as [2]:

$$\begin{aligned}\mathrm{BIAS} &= \frac{1}{n}\sum_{j=1}^{n} d^2(f(x), \hat{f}_j(x)) \\ &= \frac{1}{n}\sum_{j=1}^{n}\Big((l(x)-\hat{l}(x))^2+(a(x)-\hat{a}(x))^2+(r(x)-\hat{r}(x))^2\Big) \qquad (80)\end{aligned}$$

In practical applications, BIAS is not computable because the function $f(x)$ is certainly unknown. This quantity will be reported for the performance evaluation of the different methods in our simulations.

In the V-fold cross-validation method, the dataset is divided into v roughly equal parts. Each time one of the v subsets is used as the test dataset (TED) and, the other $v-1$ subsets are considered as a train dataset (TRD). For each $V = 1, \ldots, v$, TRD is trained by using the methodology of ANFIS and different membership functions. The membership function with the least errors in testing is selected as initial membership. After training V folds, the average errors of test across all V training is computed and results of the best net will be shown in output. The most common values for number of folds are 5 or 10. In our work, we assume $V = 5$. So, we showed $V = 5$ in flow chart. In $V = 5$, terminate the training of network and Report results of net with the least errors in output. In flow chart, $I = 1, \ldots, 4$ shows different memberships such as Triangular, Trapezoidal, Gaussian and Gbell. Data training has been performed up to 4 times for different functions and we have reported the results obtained with the membership function that has the least error. In this paper, Gaussian function is selected as the initial membership function with number of MF equal 5 for input where parameters τ and σ represent the center and the width, respectively.

4 Examples

To illustrate the proposed technique and smoothing methods, the following examples are analyzed. The fuzzy nonparametric regression function is estimated and the results are compared. For this purpose, error values of GOF

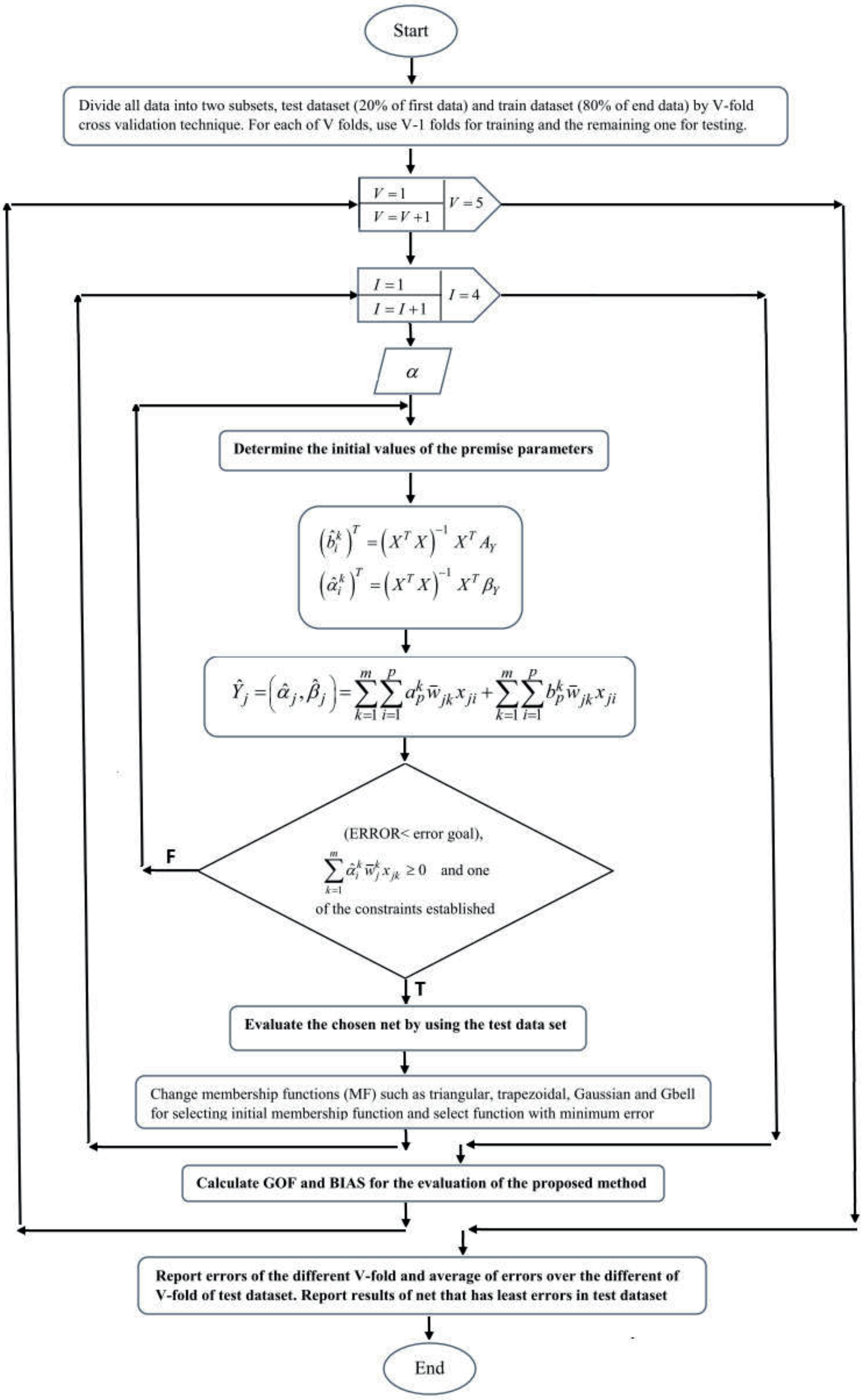

Fig. 4: Flow chart of the fuzzy regression function prediction by the proposed method

and BIAS, and their charts are used. In our simulations, two datasets are generated in the same way as that in Cheng and Lee [2] except that error exerted on the spread of fuzzy is taken on the interval $[-0.25, 0.25]$ instead of $[0, 1]$. In this paper, Gaussian function is selected as the initial membership function with number of MF equal 5 for input where parameters .. and .. represent the center and the width, respectively.

Example 1 Consider the following function:

$$f(x) = \frac{1}{5}x^2 + 2\exp\left(\frac{x}{10}\right),$$

and $x_j = 0.1j$ $(j = 1, 2, \ldots, 100)$ is uniformly generated within interval $[0, 10]$. 100 pairs of sample data are generated for function $f(x)$. Let $Y_j = (a_j, \beta_j)$ is symmetric triangular fuzzy output so that

$$\begin{cases} a_j = f(x_j) + \text{rand}[-0.5, 0.5] \\ \beta_j = (1/4)f(x_j) + \text{rand}[-0.25, 0.25] \end{cases} \quad j = 1, \ldots, 100$$

where rand $[a, b]$ denotes a random number between a and b for each j. The proposed method with 5-fold cross-validation technique and smoothing methods, such as the local linear smoothing, the kernel smoothing and the k-nearest neighbor methods, are applied to fit the regression model. The performance of these methods is compared by using error values of GOF and BIAS. The obtained results from different methods are summarized in Table 2. The obtained premise and consequence parameters of the fifth fold $(V = 5)$ that has the least error in the proposed method test are shown in Table 1. For making a graphical comparison, the observed values and the predicted values of the fifth fold $(V = 5)$ in the proposed method are depicted for train and test data in Figs. 5 and 6. So, the results of KS, k-NN and LLS methods are shown in Figs. 7, 8 and 9. These results can be compared by using Figures and Table 2. Results show that the proposed method works quite well not only in producing a satisfactory estimate of the fuzzy regression function, but also in reducing the boundary effect significantly. The convergence behavior of the training of this method is plotted in Fig. 15 that ERROR is defined in Eq. 78 and an 'epoch' means a complete presentation of the entire set of the training data.

Example 2 Consider the following function:

$$g(x) = 10 + 5\sin(0.25\pi(1 - x)^2),$$

and $x_j = 0.1$ $(j = 1, 2, \ldots, 100)$ is uniformly generated on interval $[0, 10]$. 100 pairs of sample data are generated for function $g(x)$. Let the observed fuzzy outputs are symmetric triangular fuzzy numbers as $Y_j = (a_j, \beta_j)$, where

$$\begin{cases} a_j = g(x_j) + \text{rand}[-0.5, 0.5] \\ \beta_j = (1/3)g(x_j) + \text{rand}[-0.25, 0.25] \end{cases} \quad j = 1, \ldots, 100$$

Table 1: The obtained premise and consequence parameters of the proposed method for Example 1.

V	(τ_r, σ_r)	(b_0^k, α_0^k)	(b_1^k, α_1^k)
1	(0.4508, 2.4816)	(2.7231, 0.8732)	(-0.4052,- 0.1040)
2	(2.3666, 2.2040)	(1.8143, 0.6716)	(1.5509, 0.3836)
3	(5.3432, 2.1505)	(0.4792, 0.3274)	(1.2556, 0.3090)
4	(7.8415, 2.3521)	(0.2105, 0.2710)	(1.4341, 0.3518)
5	(9.1400, 3.0563)	(-0.3051, 0.2649)	(3.5011, 0.8655)

Table 2: The obtained results of the proposed method using 5-fold cross validation technique and the different smoothing methods for Example 1.

proposed method						
Number of folds	GOF test	GOF train	GOF total	BIAS test	BIAS train	BIAS total
1	3.9e-08	6.23e-08	5.8e-08	7.8e-07	7.2e-08	6.8e-08
2	1.78e-07	6.1e-08	8.4e-08	7.87e-07	7.2e-08	6.1e-08
3	1.7e-07	5.98e-08	8.19e-08	7.9e-07	7.3e-08	5.1e-08
4	6.97e-08	6.18e-08	6.3e-08	6.6e-07	6.8e-08	5.3e-08
5*	7.71e-08	6.056e-08	6.4e-08	5.9e-07	7.9e-08	5.5e-08
average errors	1.1e-07	6.1e-08	7.03e-08	7.2e-07	7.3e-08	5.8e-08

Smoothing methods				
Methods	kernel	Smoothing parameter	GOF	BIAS
LLS	Gauss	0.51	0.2544	0.0317
	Epanechnikov	1.2	0.2607	0.0328
LLS	Gauss	0.19	0.2212	0.0628
	Epanechnikov	1.4	0.2302	0.0606
k-NN		5	0.2547	0.0928

The smoothing methods and the proposed method with 5-fold cross validation technique are applied to fit a regression model. The obtained premise and consequence parameters of the fifth fold ($V = 5$) that has the least error in the test are shown in Table 3. The convergence behavior of the training of this method is plotted in Figure 16. The error values of GOF and BIAS are numerically used to evaluate the performance of different methods. The smoothing parameter value, in k-NN, KS and LLS methods, is determined by cross-validation method. For numerical comparison, the obtained results from different smoothing methods in Table 4. Also, the regression results are depicted for different methods in Figs. 10-14. Table 4 can be used to compare these results. Like the previous example, it can be observed from Table 4, GOF and BIAS error values of the proposed method are lower than the other methods, which point to the accuracy of the proposed method. As it is seen, the proposed approach reduces the boundary effect significantly.

Table 3: The obtained premise and consequence parameters of the proposed method for Example 2.

V	(τ_r, σ_r)	(b_0^k, α_0^k)	(b_1^k, α_1^k)
1	(0.4482, 1.8077)	(11.6014, 3.7736)	(-1.3374, -0.4469)
2	(2.2061, 1.8813)	(6.7824, 2.1771)	(2.0249, 0.6736)
3	(5.7465, 1.5239)	(1.0712, 0.2686)	(2.9498, 0.9830)
4	(8.3850, 1.7026)	(4.8607, 1.5305)	(-0.6883, -0.2295)
5	(11.1514, 1.2760)	(1.3321, 0.3492)	(1.9486, 0.6500)

Table 4: The obtained results from the proposed method using 5-fold cross validation technique and the different smoothing methods for Example 2.

proposed method						
Number of folds	GOF test	GOF train	GOF total	BIAS test	BIAS train	BIAS total
1	1.81e-05	1.44e-05	1.5e-05	1.49e-04	1.84e-05	1.26e-05
2	5.51e-05	8.93e-06	1.8e-05	9.46e-05	1.15e-05	1.03e-05
3	3.25e-05	1.25e-05	1.7e-05	1.67e-04	1.48e-05	1.08e-05
4	2.16e-05	1.36e-05	1.5e-05	1.31e-04	1.57e-05	1.25e-05
5*	8.99e-06	1.55e-05	1.4e-05	6.7e-05	1.87e-05	1.18e-05
average errors	2.73e-05	1.29e-05	1.58e-05	1.22e-04	1.58e-05	1.16e-05

Smoothing methods				
Methods	kernel	Smoothing parameter	GOF	BIAS
LLS	Gauss	0.21	0.2252	0.0520
	Epanechnikov	0.52	0.2544	0.0552
LLS	Gauss	0.15	0.2080	0.0821
	Epanechnikov	0.34	0.2337	0.0848
k-NN		5	0.2997	0.01489

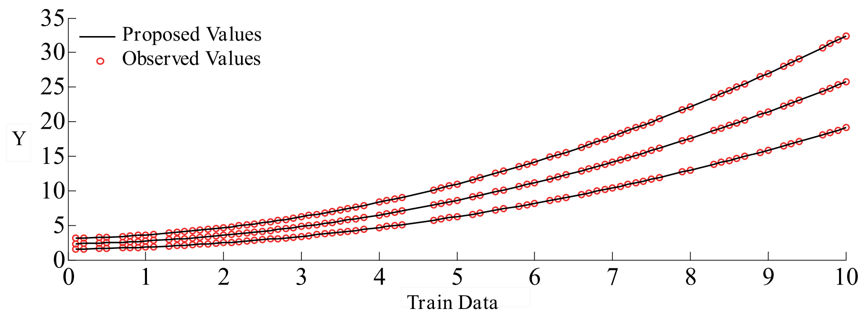

Fig. 5: The obtained regression results of fold $V = 5$ by the proposed method for Example 1 with train data.

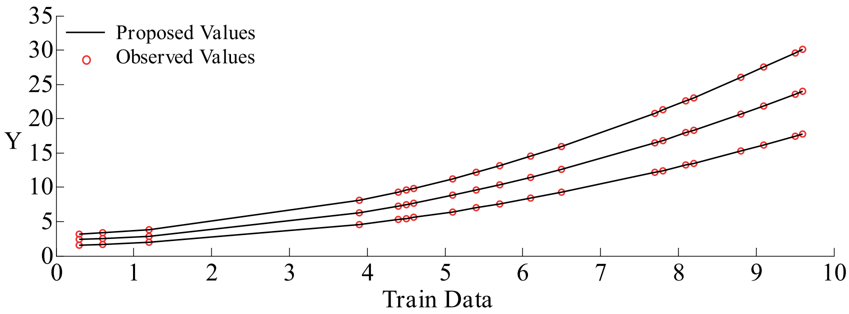

Fig. 6: The obtained regression results of fold V=5 by the proposed method for Example 1 with test data.

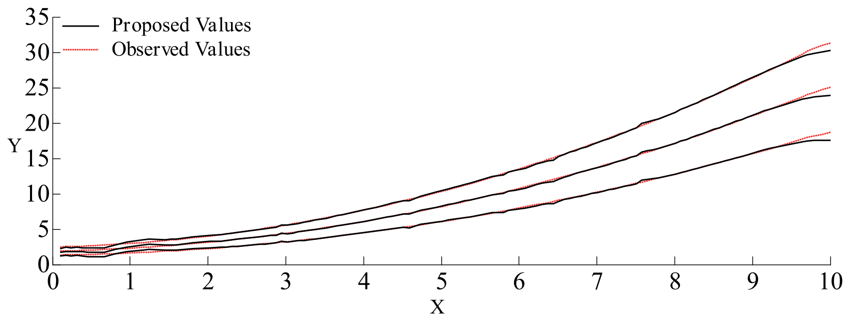

Fig. 7: The obtained regression results for Example 1 by using the fuzzy KS method with = 0.19.

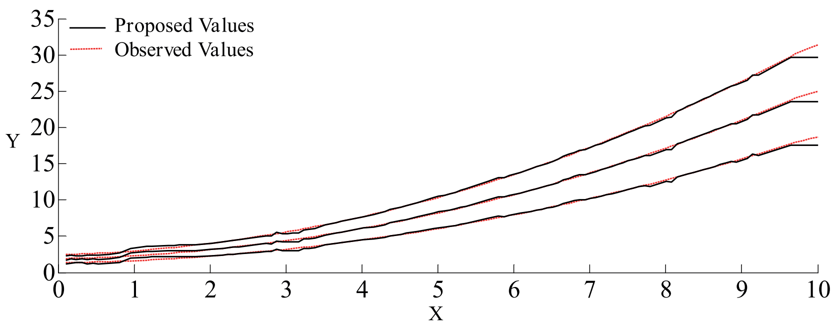

Fig. 8: The obtained regression results for Example 1 by using the fuzzy k-NN method with $k = 5$.

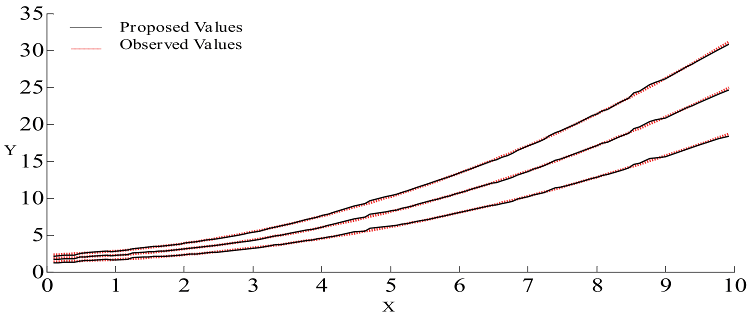

Fig. 9: The obtained regression results by the LLS method for Example 1 with = 0.51.

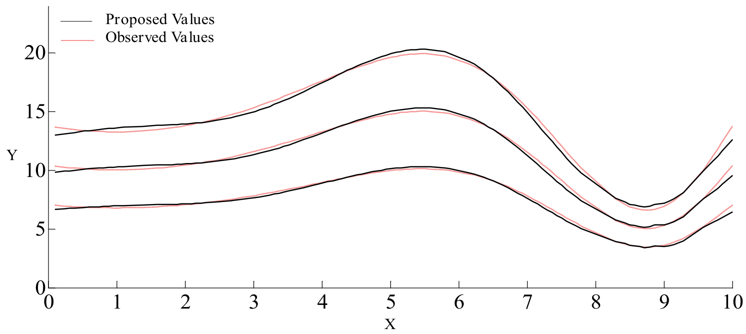

Fig. 10: The obtained regression results for Example 2 by using the KS method with = 0.15.

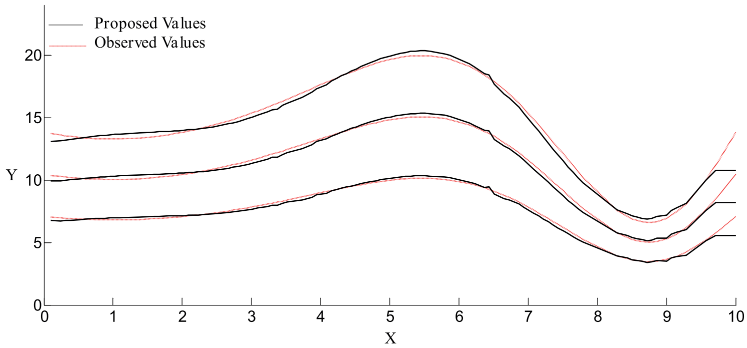

Fig. 11: The obtained regression results for Example 2 by using the fuzzy k-NN method with k=5.

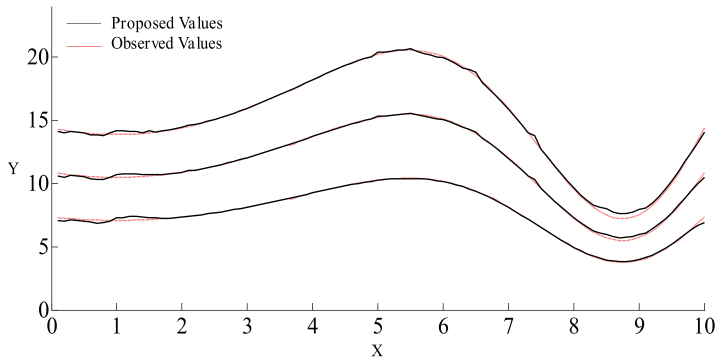

Fig. 12: The obtained regression results by the LLS method for Example 2 with =0.21.

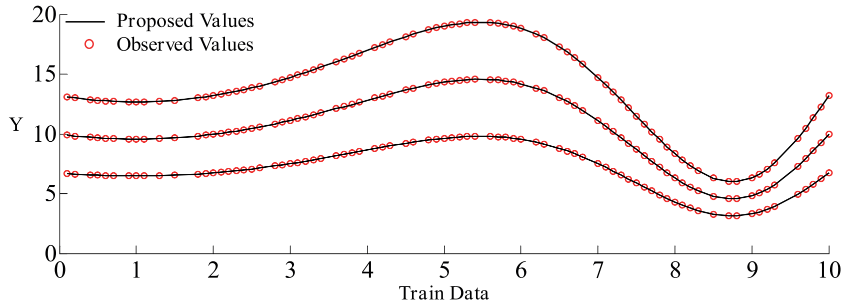

Fig. 13: The obtained regression results fold V=5 by the proposed method for Example 2 with train data.

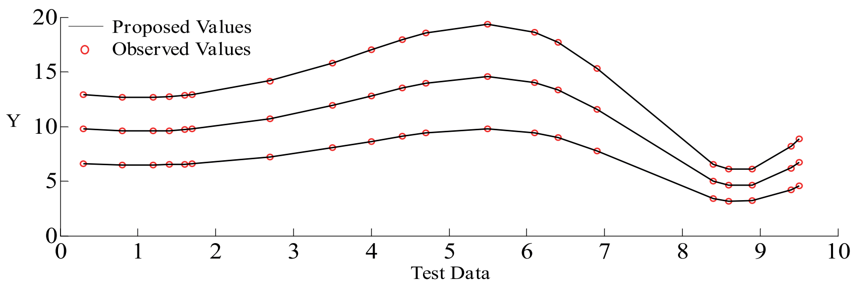

Fig. 14: The obtained regression results fold V=5 by the proposed method for Example 2 with test data.

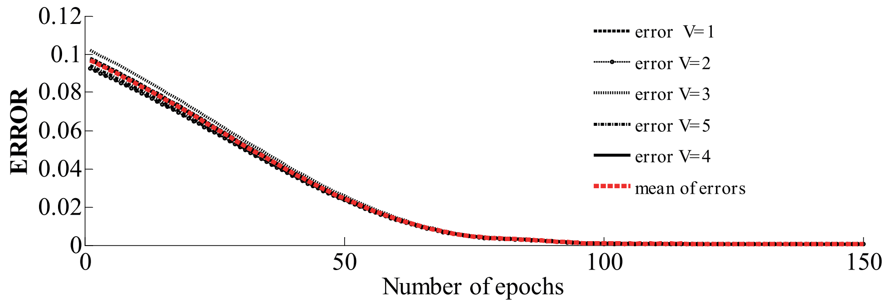

Fig. 15: Convergence behavior for Example 1.

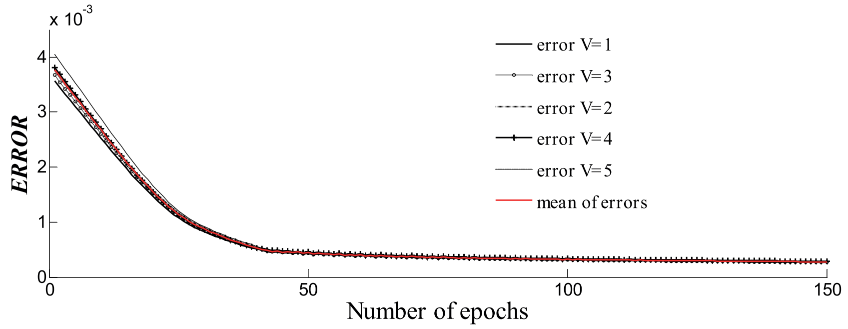

Fig. 16: Convergence behavior for Example 2.

Example 3 Consider the following function:

$$\begin{cases} a(x_1, x_2) = 5, \\ l(x_1, x_2) = \frac{4}{625}(25 - (5 - x_1)^2)(25 - (5 - x_2)^2), \\ r(x_1, x_2) = 10 - l(x_1, x_2), \end{cases}$$

where the domain of $X = (x_1, x_2)$ is $D = [0, 10]^2$. This function is depicted in Fig. 18. A set of data is generated the same way as that in [51] and in the following manner.

The crisp inputs of the independent variables x_1 and x_2 are equidistantly taken from 0 to 10 with increment 0.5. These values form the lattice points of size $n = 441$. These lattice points are ordered in such a way that their Cartesian coordinates can be expressed as:

$$(x_{j1}, x_{j2}) = (0.5\text{mod}(j - 1, 21), 0.5\text{int}(j - 1, 21)) \ , \quad j = 1, 2, \ldots, n,$$

where $\text{mod}(a, b)$ is the remainder and $\text{int}(a, b)$ is the integer part of a divided by b. Let output $Y_j = (a_j, \beta_j)$ $(j = 1, 2, \ldots, n)$ is a symmetric fuzzy number and it is generated by:

$$\begin{cases} a_j = a(x_{j1}, x_{j2}) + \epsilon_j, \\ \beta_j = a(x_{j1}, x_{j2}) - l(x_{j1}, x_{j2}) + \alpha_j, \end{cases} \quad j = 1, 2, \ldots, n,$$

Table 5: The obtained regression results from the proposed method using 5-fold cross validation technique and the smoothing methods for Example 3.

proposed method				
Number of folds	GOF test	GOF train	GOF total	BIAS total
1	0.3121	0.1189	0.1575	0.0028
2*	0.3411	0.1114	0.1572	0.00027
3	0.2341	0.1253	0.1471	0.0202
4	0.2670	0.1231	0.1518	0.0010
5	0.5296	0.1189	0.2009	0.0105
average errors	0.3337	0.1195	0.1622	0.0070
Smoothing methods				
Methods	kernel	Smoothing parameter	GOF	BIAS
LLS	Gauss	0.7	0.7360	0.0654
KS	Gauss	0.55	0.7007	0.1037

where $\epsilon_1, \epsilon_2, \ldots, \epsilon_n$ and $\alpha_1, \alpha_2, \ldots, \alpha_n$, are the observation errors of the centers and the spreads of the observed fuzzy outputs, respectively. Also, they are independently generated from the normal distributions $N(0, 0.5^2)$ and $N(0, 0.25^2)$, respectively. The dataset (x_{j1}, x_{j2}, Y_j) $(j = 1, 2, \ldots, n)$ is used to obtain the fuzzy regression function. Moreover, the proposed method with 5-fold cross validation technique, the fuzzy kernel smoothing and the local linear smoothing methods with Gaussian kernel are applied to fit the simulated dataset and the obtained regression results are shown in Table 3. Gaussian function is selected as the initial membership function with number of MF equal 5 for each input variable. The obtained premise and consequence parameters are not reported here in consideration of the limited space. The convergence behavior of the proposed method is plotted in Fig. 19. The estimated fuzzy nonparametric regression function of the proposed method is depicted in Fig. 18. By using Figs. 17, 18 and Table 5, it is observed that the proposed method still produces a quite satisfactory estimate of the underlying fuzzy regression function in the case of two-dimensional input. In the following, the results of different methods are compared with each other.

It can be seen from Tables 2, IV and V that the fuzzy k-NN method produces a less satisfactory estimates of the fuzzy regression function. So, the fuzzy k-NN method has the values of both GOF and BIAS larger than the corresponding values of the other smoothing methods and the proposed method. Also, the value of GOF in the kernel smoothing method is always less than that in the local linear smoothing method that indicates the kernel smoothing method tends to produce such estimates that are closer to their respective observations. In contrast, the value of BIAS in the local linear smoothing method is always smaller than the value of BIAS in the kernel smoothing method, which indicates that the local linear smoothing method gives less bi-

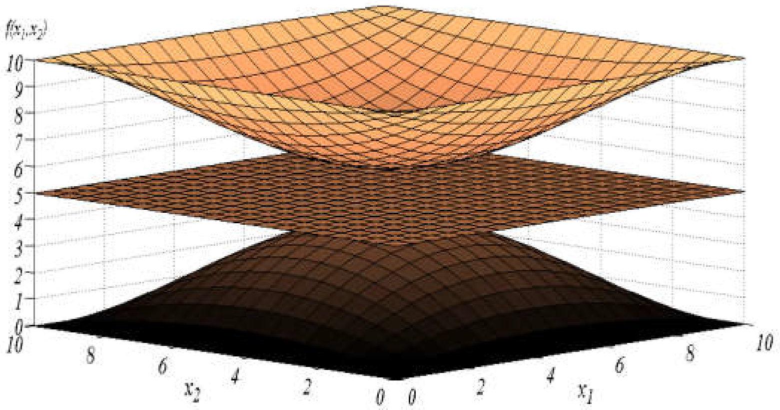

Fig. 17: The center, the lower and the upper limit lines of the real function for Example 3.

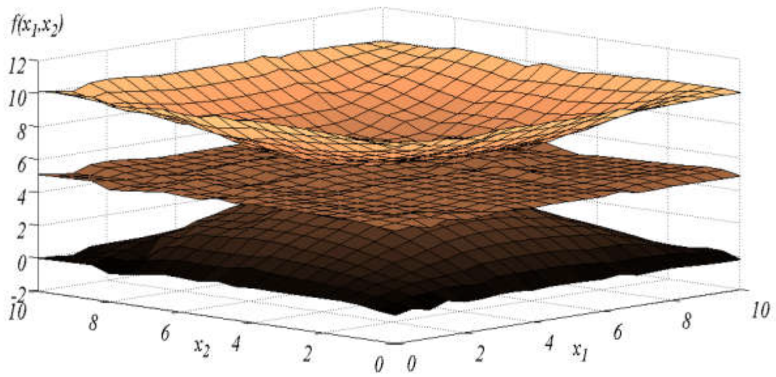

Fig. 18: The estimated ones by the proposed method for Example 3.

ased estimates of the center, the lower and upper limits of the underlying fuzzy regression function. Also, from Figs. 6-14, we see that, the kernel smoothing and k-NN methods produce more fluctuating estimates and suffer from more serious boundary effects. In contrast, the LLS method not only gives quite smooth estimates of the centerline, the lower and upper limit lines of the underlying regression function but also reduce the boundary effect significantly. So, it can be seen from Tables 2, 4 and 5 that, in each case, the error values of the proposed method are always smaller than that in the smoothing methods and this method decreases the values of both GOF and BIAS, simultaneously. In summary, these results show that the proposed method works quite well

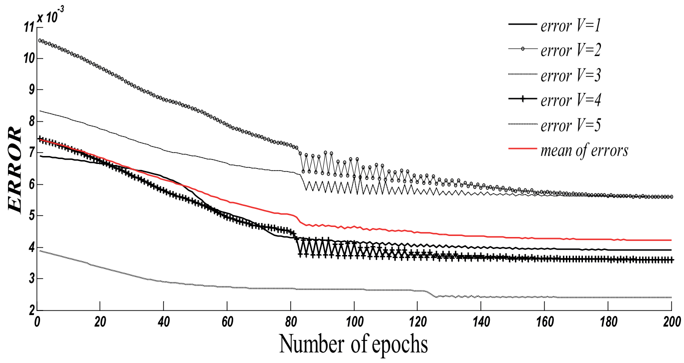

Fig. 19: Convergence behavior of Example 3.

not only in producing a satisfactory estimate of the fuzzy regression function but also in reducing the boundary effect significantly.

5 Conclusions

In this paper, the fuzzy nonparametric regression function is considered with crisp inputs and triangular fuzzy output. The adaptive neuro-fuzzy inference system method is formulated by the fuzzy least-squares based on Diamond's distance in section consequence parameters. So, the ANFIS method is fuzzified to fit the fuzzy regression function. V-fold cross-validation technique is used to estimate the quality and stability of the proposed method and avoid overfitting. It can be seen that convergence behaviors of the performance for datasets of the three examples are stable. Also, other forecasting techniques, such as LLS, KS and k-NN methods are used to fit the fuzzy regression function. The effectiveness of various methods is demonstrated by different simulation examples. By considering the obtained results of different methods and figures, it can be determined that the boundary effect for k-NN and KS methods have more severity than both the proposed and the LLS methods, and the performance of the proposed method is evidently better than the various smoothing methods. Based on Example 3, it should be pointed out that the performance of the fuzzy nonparametric regression function prediction can be significantly enhanced by using the proposed method in case twodimensional input. The proposed method is especially useful for practical problems, which involve uncertainty in the output observed data. Using the results, the conducted simulation experiments are shown that the performance of the proposed method is better than the smoothing methods, which reduces the CV. In the proposed approach, when the observation numbers are increased, the accuracy is increased, in comparison with the existing smoothing methods. These advantages would make our algorithm an acceptable one to generate nonparametric regression functions. Thus current proposed method has reduced the fuzziness of the system and it has faster adaptation.

Acknowledgements We would like to thank wish to thank Mrs. A. Mohammad Rahim, Mr. M. Vahabi and Dr. M. R. Gholamian for their supports and their help in doing our study.

References

1. A. Celmins, Multidimensional least-squares fitting of fuzzy models, Fuzzy sets and systems, 22, 245-253 (1997).
2. C. B. Cheng and E. S. Lee, Nonparametric fuzzy regression k-NN and kernel smoothing techniques, Computers and Mathematics with Applications, 38, 239-251 (1999).
3. C. B. Cheng, E. S. Lee, Applying Fuzzy Adoptive Network to Fuzzy Regression Analysis, Computers and Mathematics with Applications, 38, 123-140 (1999).
4. C. B. Cheng, E.S. Lee, Fuzzy regression with radial basis function networks, Fuzzy Sets and Systems, 119, 291-301 (2001).
5. T. E. Dalkilic and T. Apaydin, A fuzzy adaptive network approach to parameter estimation in cases where independent variables come from an exponential distribution, Journal of Computational and Applied Mathematics, 233, 36-45 (2009).
6. T. E. Dalkilic, T. Apaydin, Parameter Estimation by ANFIS in Cases Where Outputs are Non-Symmetric Fuzzy Numbers, International Journal of Applied Science and Technology, 92-103 (2014).
7. M. Danesh, S. Danesh, A Combinatorial Algorithm for Fuzzy Parameter Estimation with Application to Uncertain Measurements, Journal of AI and Data Mining, 8, 525-533 (2020).
8. M. Danesh, S. Danesh, K. khalili, Multi-Sensory Data Fusion System for Tool Condition Monitoring Using Optimized Artificial Fuzzy Inference System, Aerospace Mechanics Journal, 15, 103-118 (2020).
9. P. Diamond, Fuzzy least squares, Information Sciences, 46, 141-157 (1988).
10. V.A. Epanechnikov, non-parametric estimation of a multivariate probability density, Theory Probab. Appl, 14, 153-158 (1969).
11. J. Fan, I. Gijbels, Local Polynomial Modelling and Its Applications, Chapman & Hall, London (1996).
12. R. Farnoosh, J. Ghasemian, O.S fard, A modification on ridge estimation for fuzzy nonparametric regression, Iranian Journal of Fuzzy System, 9, 75-88 (2012).
13. L.V. Fausett, Fundamentals of neural networks: architectures, algorithms, and applications. Prentice Hall (1994).
14. M.H. Fazel Zarandi, I.B. Turksen, J. Sobhani, A.A. Ramezanianpour, Fuzzy polynomial neural networks for approximation of the compressive strength of concrete, Applied Soft Computing, 8, 488-498 (2008).
15. J.D. Hart, Nonparametric Smoothing and Lack-of-Fit Tests, Springer-Verlag, New York (1997).
16. D.H. Hong, C. Hwang and C. Ahn, Ridge estimation for regression models with crisp inputs and Gaussian fuzzy output, Fuzzy Sets and Systems, 142, 307-319 (2004).
17. H. Ishibuchi, K. Kwon and H. Tanaka, A learning algorithm of fuzzy neural networks with triangular fuzzy weights, Fuzzy Sets and Systems, 71, 277-293 (1995).
18. H. Ishibushi, H. Tanaka, Fuzzy neural networks with interval weights and its application to fuzzy regression analysis, Fuzzy Sets and Systems, 57, 27-39 (1993).
19. H. Ishibushi, H. Tanaka, Fuzzy regression analysis using neural networks, Fuzzy Sets and systems, 50, 257-265 (1992).
20. J.S.R. Jang. ANFIS: adaptive-network-based fuzzy inference system, IEEE Trans Syst. Man Cyber, 23, 665-685 (1993).
21. J.S.R. Jang, Self-learning fuzzy controllers based on temporal back-propagation, IEEE Transactions on Neural Network , 3, 714-723 (1992).
22. A. Jayadevsa, S.A. Rahman, A neural network with O(N) neurons for ranking N numbers in O(1/N) time. IEEE Transactions on Circuits and Systems I: Regular Papers, 51, 2044-2051 (2004).

23. A. Kenar Koohi, H. Soleimanjahi, Sh. Falahi, The application of the new intelligent system (ANFIS) in prediction of human papilloma virus oncogenicity potency, arak medical university Journal (Amuj, 13, 95-105 (2010).
24. B. Kim, R. R. Bishu, Evaluation of fuzzy linear regression models by comparing membership functions, Fuzzy Sets and Systems, 100, 343-351 (1998).
25. K.J. Kim, H.R. Chen, a comparison of fuzzy and non-parametric linear regression, Computers Ops Res, 24, 505-519 (1997).
26. H. Lee, H. Tanaka, Fuzzy approximations with non-symmetric fuzzy parameters in fuzzy regression analysis, Journal of the Operations Research Society of Japan, 42, 98-112 (1999).
27. D.O. Loftsgaarden and G.P. Quesenberry, A nonparametric estimate of a multivariate density function, Annals of Mathematical Statistics, 36, 1049-1051 (1965).
28. M. Ming, M. Friedman and A. Kandel,, General fuzzy least squares, Fuzzy Sets and Systems, 88, 107-118 (1997).
29. M. Modarres, E. Nasrabadi and MM. Nasrabadi, A mathematical-programming approach to fuzzy linear regression analysis, Applied Mathematics and Computation, 163, 977-989 (2005).
30. M. Mosleh, M. Otadi, S. abbasbandy, Evaluation of fuzzy regression model by fuzzy neural networks, Journal of Computational and Applied Mathematics, 234, 825-834 (2010).
31. E. A. Nadaraya, on estimating Regression, Theory Probab. Appl, 9, 141-142 (1964).
32. E. A. Nadaraya, Estimation of a bivariate probability density, Soobshch. Akad. Nauk Gruz. SSR, 36, 267-268 (1964).
33. M.M. Nasrabadi, E. Nasrabadi, A mathematical-programming approach to fuzzy linear regression analysis, Applied Mathematics and Computation, 155, 873-881 (2004).
34. M. Otadi, Fully fuzzy polynomial regression by fuzzy neural networks, Neurocomputing, 142, 486-493 (2014).
35. E. Pasha, T. Razzaghnia, T. Allahviranloo, Gh. Yari. and H. R. Mostafaei, A new mathematical programming approach in fuzzy linear regression models, Applied Mathematical Sciences, 35, 1715-1721 (2007).
36. W. Pedrycz, F. Gomide and D.A. Savic, An introduction to fuzzy sets: analysis and design, Fuzzy sets and systems, 39, 51-63 (1991).
37. T. Razzaghnia, Regression parameters prediction in data set with outliers using neural network, Hacettepe journal of mathematics and statistics, 48, 1170-1184 (2019).
38. T. Razzaghnia, S. Danesh & A. Maleki, Hybrid fuzzy regression with trapezoidal fuzzy data, Proceeding of the SPIE, 834901-834921 (2011).
39. T. Razzaghnia, S. Danesh, Nonparametric Regression with Trapezoidal Fuzzy Data", International Journal on Recent and Innovation Trends in Computing and Communication (IJRITCC), 3, 3826-3831 (2015).
40. M. Rezal, J. A. Ghani, M. Z. Nuawi and C. H. C. Haeon, Online tool wear prediction system in the turning process using an adaptive neuro-fuzzy inference system, Applied Soft Computing, 13, 1960- 1968 (2013).
41. C. L. Sabharwal, An implementation of hybrid approach to indexing image data bases, ACM Symp. On Applied Computing, 2, 421-426 (1999).
42. A. Shapiro, The merge of neural networks, fuzzy logic, and genetic algorithms, Insurance: Mathematic, 31, 115-131 (2002).
43. M. Stone, Cross validation choice and assessment of statistical predictions, Journal of the Royal Statistical Society, 36 (series B), 111-147 (1974).
44. T. Takagi, M. Sugeno, Fuzzy identification of systems and its application to modelling and control, IEEE Transactions on Systems, Man and Cybernetics , 15, 16-132 (1985).
45. A. Talei, L. Chua, and T. Wong, "Evaluation of rainfall and discharge inputs used by Adaptive Network based on Fuzzy Inference Systems (ANFIS) in rainfall-runoff modeling", Journal of Hydrology, 391, 248-262 (2010).
46. S. Tamer, M.A. Kamel, M. Hassan, ANFIS For Fault Classification in the Transmission Lines, The Online Journal on Electronics and Electrical Engineering (OJEEE), 2 (2008).
47. H. Tanaka, Fuzzy data analysis by possibilistic linear models, Fuzzy Sets and Systems, 24, 363-375 (1987).
48. H. Tanaka, S. Uejima, K. Asia, Linear regression analysis with fuzzy model, IEEE Transactions on Systems, Man and Cybernetics, 12, 903-907 (1982).

49. H. Tanaka, I. Hayashi, J. Watada, Possibility linear regression analysis for fuzzy data, European Journal of Operational Research , 40, 389-396 (1989).
50. H. Tanaka, J. Watada, Possibilistic linear systems and their application to the linear regression mode, Fuzzy Sets and System, 27, 275-289 (1988).
51. N. Wang, W. X. Zhang, C. L. Mei, Fuzzy nonparametric regression based on local linear smoothing technique, Information Sciences, 177, 3882-3900 (2007).
52. P. Zhang, Model selection via multifold cross validation, Annals of Statistics, 21, 299-313 (1993).