Information Sciences 364–365 (2016) 222–240

Contents lists available at ScienceDirect

Information Sciences

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

Fuzzy nonlinear regression analysis using a random weight network

Yu-Lin He, Xi-Zhao Wang*, Joshua Zhexue Huang*

Big Data Institute, College of Computer Science & Software Engineering, Shenzhen University, Shenzhen 518060, Guangdong, China

ARTICLE INFO

Article history:
Received 28 May 2015
Revised 11 December 2015
Accepted 12 January 2016
Available online 28 January 2016

Keywords:
α-cut set
Fuzzy-in fuzzy-out
Fuzzy nonlinear regression
Random weight network
Triangular fuzzy number

ABSTRACT

Modeling a fuzzy-in fuzzy-out system where both inputs and outputs are uncertain is of practical and theoretical importance. Fuzzy nonlinear regression (FNR) is one of the approaches used most widely to model such systems. In this study, we propose the use of a Random Weight Network (RWN) to develop a FNR model called FNR_{RWN}, where both the inputs and outputs are triangular fuzzy numbers. Unlike existing FNR models based on back-propagation (BP) and radial basis function (RBF) networks, FNR_{RWN} does not require iterative adjustment of the network weights and biases. Instead, the input layer weights and hidden layer biases of FNR_{RWN} are selected randomly. The output layer weights for FNR_{RWN} are calculated analytically based on a derived updating rule, which aims to minimize the integrated squared error between α-cut sets that correspond to the predicted fuzzy outputs and target fuzzy outputs, respectively. In FNR_{RWN}, the integrated squared error is solved approximately by Riemann integral theory. The experimental results show that the proposed FNR_{RWN} method can effectively approximate a fuzzy-in fuzzy-out system. FNR_{RWN} obtains better prediction accuracy in a lower computational time compared with existing FNR models based on BP and RBF networks.

1. Introduction

Fuzzy regression analysis is a powerful method for forecasting the fuzzy outputs of an uncertain system. According to Wang and Tsaur in their study entitled "*Insight of A Fuzzy Regression Model*", *fuzzy regression can be quite useful in estimating the relationships among variables where the available data are very limited and imprecise, and variables are interacting in an uncertain, qualitative, and fuzzy way* [40]. During recent decades, fuzzy regression has found many applications in various areas, such as business management, engineering, economics, sociology, and biological science. Fuzzy regression can be divided into two categories: fuzzy linear regression (FLR) and fuzzy nonlinear regression (FNR).

FLR was first studied by Tanaka et al. [35] in 1982, who used a fuzzy linear function with crisp inputs, fuzzy outputs, and fuzzy coefficients to approximate an uncertain system. Subsequently, their FLR model was extended by [35] to handle fuzzy regression tasks with fuzzy inputs and fuzzy outputs. Improvements were made in two areas: constructing fuzzy linear functions with crisp coefficients [2,7,8,12,20,21] and making fuzzy linear functions with fuzzy coefficients [8,15,30,31,38,42,43,45]. Linear programming and least squares are used widely to solve the crisp or fuzzy coefficients for FLR models. However, for a crisp-in fuzzy-out or fuzzy-in fuzzy-out system, the relationship between the inputs and outputs is usually nonlinear in

* Corresponding authors
E-mail addresses: csylhe@126.com, yulinhe@szu.edu.cn (Y.-L. He), xzwang@szu.edu.cn (X.-Z. Wang), zx.huang@szu.edu.cn (J. Z.-X. Huang).

http://dx.doi.org/10.1016/j.ins.2016.01.037

many applications, i.e., the fuzzy outputs cannot be expressed simply as the weighted sum of the crisp or fuzzy inputs; thus, a more sophisticated FNR approach is required. Due to their better capacity for approximating nonlinear functions, feed-forward neural networks are usually employed for constructing complex FNR models.

FNR models based on feed-forward neural networks include the following two categories:

1. Back-propagation (BP) network-based FNR models. In 1992, Ishibuchi and Tanaka [17] proposed an FNR model ($FNR_{BP\text{-}I}$) that uses two BP networks to fit the upper and lower bounds of interval-valued fuzzy numbers. These two BP networks with crisp weights and biases are trained using the standard BP algorithm [29]. Both the inputs and outputs of $FNR_{BP\text{-}I}$ are interval-valued fuzzy numbers. In 1993, Ishibuchi and Tanaka proposed another BP network-based FNR model ($FNR_{BP\text{-}II}$) [18] that handles the FNR problem using crisp inputs and fuzzy outputs. Unlike $FNR_{BP\text{-}I}$, there is only one BP network in $FNR_{BP\text{-}II}$. The weights and biases in this BP network are interval-valued fuzzy numbers. In 1995, Ishibuchi et al. [19] proposed the use of a BP network with triangular fuzzy number (TFN) weights to conduct the fuzzy regression analysis ($FNR_{BP\text{-}III}$), where the inputs and outputs are interval-valued fuzzy numbers.
2. Radial basis function (RBF) network-based FNR models. In 2001, Cheng and Lee [6] used an RBF network to design an FNR model ($FNR_{RBF\text{-}I}$). In $FNR_{RBF\text{-}I}$, the inputs and centers of the input layer nodes are crisp, whereas the outputs and output layer weights are TFNs. Compared with previous BP network-based FNR models, $FNR_{RBF\text{-}I}$ obtained a faster convergence rate. A fuzzified RBF network-based FNR model ($FNR_{RBF\text{-}II}$) was described by Zhang et al. [49] in 2005. In $FNR_{RBF\text{-}II}$, the inputs, outputs, centers, deviations, and weights are Left–Right (L–R) fuzzy numbers. $FNR_{RBF\text{-}II}$ can serve as a universal function approximation for any continuous fuzzy function defined on a compact set.

The main problems with these FNR models based on BP and RBF networks include their high training complexity, local minima, and complex parameter tuning (e.g., learning rate, learning epochs, and stopping criteria). Optimization of the neural network parameters, e.g., the weights, biases, node centers, and node deviations in $FNR_{BP\text{-}I}$, $FNR_{BP\text{-}II}$, $FNR_{BP\text{-}III}$, $FNR_{RBF\text{-}I}$, and $FNR_{RBF\text{-}II}$, is based on gradient descent approaches where the learning speed is relatively slow. Moreover, the gradient-based methods may readily converge to a local minimum. In addition, there are no widely accepted methods for determining the optimal learning rate, learning epochs, and stopping criteria for BP and RBF networks at present. Thus, trial-and-error methods are often used to select these parameters. These methods require large periods of computational time to establish BP and RBF network-based FNR models.

Thus, in the present study, we develop a new FNR learning algorithm that is faster with high generalization performance, as well as avoiding many of the difficulties that affect the traditional gradient-based FNR models. In contrast to the conventional training algorithms for BP and RBF networks, Random Weight Networks (RWNs) [3,5,34,51] do not require iterative adjustments of the network weights and there is no learning parameter to determine. Thus, the training speed of RWNs can be thousands of times faster than traditional gradient descent algorithms. In addition, the good generalization capacity of RWNs has been demonstrated in recent studies [3,5,51]. Therefore, we propose a RWN-based FNR model called FNR_{RWN} in the present study. FNR_{RWN} is a single hidden layer feed-forward neural network, where the inputs and outputs are TFNs. The input layer weights and hidden layer biases of FNR_{RWN} are selected randomly. In order to analytically calculate the output layer weights, we define a new computational paradigm to minimize the integrated squared error between α-cut sets that correspond to the predicted fuzzy outputs and target fuzzy outputs. Our simulation results indicate that FNR_{RWN} has better generalization performance as well as requiring less training time compared with $FNR_{BP\text{-}III}$ and $FNR_{RBF\text{-}II}$. Overall, our results demonstrate that FNR_{RWN} can effectively approximate a fuzzy-in fuzzy-out system.

The remainder of this paper is organized as follows. In Section 2, we provide a brief introduction to TFNs. In Section 3, we describe the BP network and RBF network-based FNR models. The RWN-based FNR model (FNR_{RWN}) is presented in Section 4. In Section 5, we report experimental comparisons that demonstrate the feasibility and effectiveness of FNR_{RWN}. Finally, we give our conclusions and suggestions for further research in Section 6.

2. TFNs and their mathematical operations

2.1. Definition of a TFN

Definition 1 [13,23]**.** A fuzzy number A is defined as a fuzzy set on the domain of real numbers $\Re$, which satisfies the following conditions:

1. $\exists\, x_0 \in \Re,\ \mu_A(x_0) = 1,$ where $\mu_A(x)$ is the membership function of fuzzy set A;
2. $\forall\, \alpha \in (0, 1]$, $A_\alpha = \{x | x \in \Re, \mu_A(x) \geq \alpha\}$ is a finite closed interval.

The TFN A is the most popular fuzzy number, which is represented by two endpoints a_1 and a_3 and one peak-point a_2

$$A = (a_1, a_2, a_3).$$

It can be interpreted as a membership function