第 31 卷 第 4 期 石家庄铁道大学学报(自然科学版) Vol. 31 No. 4
2018 年 12 月 Journal of Shijiazhuang Tiedao University(Natural Science Edition) Dec. 2018

梯形模糊数据基于期望值简约的模糊非线性回归

赵士欣， 陈惜源， 王荣荣

(石家庄铁道大学 数理系，河北 石家庄 050043)

摘要：模糊非线性回归是研究数据不确定性的一种有效方法。借鉴 2-型模糊数学理论中的模糊简约思想，提出了一种利用期望值简约技术处理梯形模糊数据的模糊非线性回归模型。首先将梯形模糊输入输出利用其期望值简约为清晰输入输出，然后利用经典随机赋权神经网络对其进行学习，最后利用目标输出模糊变量的宽度矩阵将网络实际清晰值输出还原为梯形模糊输出。与已有模型的对比实验表明，提出的模型具有更高的学习准确度和更好的扩展能力。

关键词：期望值简约；梯形模糊数；随机赋权网络；模糊非线性回归

中图分类号：TP183 文献标志码：A 文章编号：2095-0373(2018)04-0102-07

0 引言

模糊非线性回归(Fuzzy nonlinear regression，FNR) [1]是预测不确定系统中模糊输出的一种有效方法。近年来已在诸多领域得到了应用，例如化工厂的故障诊断[2]，降解过程的可靠性分析[3]，识别顾客需求的补偿程度[4]等。随着大数据时代的到来，模糊非线性回归也展示出了其在大规模不确定数据领域的巨大潜力[5]。

在已有模糊非线性回归模型分析中，神经网络是最常用且有效的机器学习算法，其处理的模糊数据主要有以下 3 种。(1)针对区间模糊数的 FNR。Ishibuchi 和 Tanaka 在 1992 和 1993 年分别提出两种模糊非线性模型[6-7]，这两个模型都针对区间型模糊数据[8]，其训练过程均采用 BP (Back-Propagation)算法。2014 年，Zhao 和 Wang 提出了一种处理区间模糊输入输出的模型[9]，该模型利用极速学习机算法，具有很快的学习速度。(2)针对三角模糊数的 FNR。1995 年，Ishibuchi et al[10]提出了一种模型处理三角型模糊输入输出的算法，其训练过程依然采用了 BP 算法，连接权值均为三角型模糊数。Cheng et al[11]在 2001 年利用 RBF(Radial Basis Function)网络设计了一种模糊非线性回归模型，该模型中输出和输出层连接权值均为三角型模糊数(FNRRBF)。He et al[12]在 2016 年提出了一种基于随机赋权网络(Random Weight Network，RWN)的模糊非线性回归模型(FNRRWN) 。无需反复迭代调整参数的 RWN[13-15]使得这一模型取得了比 FNRBP 和 FNRRBF 快得多的训练速度。(3)针对梯形模糊数的 FNR。1994 年，Ishibuchi et al[16]设计了一种利用梯形模糊数为权的 BP 算法的模糊非线性回归模型。该模型中的输入输出均为梯形模糊数，而训练采用了传统的 BP 算法。

由上可知，模糊非线性回归模型中采用的算法主要有 3 种，BP 算法，RBF 算法和 RWN 算法，FNRBP 和 FNRRBF 的训练精度尚可接受，但其计算复杂度过高导致训练时间太长。而 FNRRWN 克服了这一缺点，不仅拥有很快的训练速度，并且精度也可以接受。同时也发现了 FNRRWN 的缺陷，即 FNRRWN 模型中采用了两处近似运算，这在一定程度上降低了该模型的学习准确度。为了描述方便将 He et al 提出的思想用于梯形模糊数据的模型记为 TraFNRRWN-I。基于以上考虑，本文提出了一种针对梯形模糊

收稿日期：2017-06-14 责任编辑：车轩玉 DOI：10.13319/j.cnki.sjztddxxbzrb.2018.04.17

基金项目：国家自然科学基金(11501379)；河北省自然科学基金(A2015210103)；河北省教育厅青年基金(QN2016140)；石家庄铁道大学大学生创新创业计划(201610107008)。

作者简介：赵士欣(1978—)，女，讲师，研究方向为不确定信息处理，机器学习。E-mail：cssxzhao@163.com

赵士欣，陈惜源，王荣荣 .梯形模糊数据基于期望值简约的模糊非线性回归[J].石家庄铁道大学学报：自然科学版，2018，31(4)：102-108.

数据借助于模糊变量期望值简约的随机赋权模糊非线性回归模型(TraFNRRWN-II)。该模型借鉴了2-型模糊数学理论中的模糊简约思想,利用模糊变量的期望值对梯形模糊变量进行简约,然后利用经典的随机赋权网络对其进行训练,最后再根据网络的原始目标输出的宽度矩阵将网络实际的清晰值输出还原为模糊值输出。由于期望值这一概念是模糊变量本身固有的数字特征,无需加入人工干预成分,在相当程度上保留了数据的最大信息,从而能更好地对数据做出预测。

1　相关背景知识

1.1　梯形模糊数

模型针对梯形模糊输入输出设计,故先对梯形模糊数简单介绍如下。

定义 1.1 符合以下隶属函数的模糊数 $A=(a_1,a_2,\cdots,a_4)$ 是一个梯形模糊数

$$\mu_A(x)\begin{cases}\dfrac{x-a_1}{a_2-a_1}, & a_1\leqslant x<a_2\\ 1, & a_2\leqslant x<a_3\\ \dfrac{a_4-x}{a_4-a_3}, & a_3\leqslant x<a_4\\ 0, & x<a_1,x\geqslant a_4\end{cases}\tag{1}$$

它的 α-截集为

$$A_\alpha=[A_\alpha^L,A_\alpha^U]=[\alpha\alpha_2+(1-\alpha)\alpha_1,\alpha\alpha_3+(1-\alpha)\alpha_4]\tag{2}$$

1.2　期望值简约

不确定性简约,即模糊变量简约,是解决2-型模糊问题的重要方法。所谓简约,即用一个特殊的值对模糊变量进行代替,简约既是一种舍弃,更是一种保留[17]。不确定性简约的方法有很多,期望值简约是一种常用且有效的方法。模糊变量期望的定义有很多种[18-22],采用了 Liu B et al (2002) 中给出的定义[22]。

定义 1.2 假设 ξ 是定义在可信性空间(Γ,A,Cr)上的一个模糊变量,则 ξ 的期望定义为

$$E(\xi)=\int_0^{\infty}Cr\{\xi\geqslant r\}\mathrm{d}r-\int_{-\infty}^{0}Cr\{\xi\leqslant r\}\mathrm{d}r\tag{3}$$

由定义 1.2 可知,当 ξ 是一个梯形模糊变量$(\alpha_1,\alpha_2,\cdots,\alpha_4)$时,其期望值为

$$E(\xi)=\frac{1}{4}(2\alpha_1+2\alpha_2-\alpha_3+\alpha_4)\tag{4}$$

1.3　针对梯形模糊数据的基于随机赋权网络的模糊非线性回归(TraFNRRWN-I)

He et al 在 2016 年提出了一种新的基于随机赋权网络的模糊非线性回归模型[12],该模型的处理对象为三角型模糊数据,但也可用于梯形模糊数据,根据文献[12]中思想,介绍相对应的针对梯形模糊数据的模型记为 TraFNRRWN-I。

TraFNRRWN-I 处理的数据均为梯形模糊数,即输入输出均为梯形模糊数。其网络结构为一个单隐含层前馈神经网络,网络的输入和输出为梯形模糊变量的 α-截集。输入层权值和隐含层偏置被随机赋值为[0,1]之间的数,而输出层的权值则通过最小化所有样例的目标模糊输出和实际模糊输出各 α-截集的误差之和训练而得,具体训练过程简介如下,详细参见文献[12]。

第 n 组数据$(X_{n1},X_{n2},\cdots,X_{nD})$对应第 j 个隐含节点的输入和输出为

$$R_{nj}=(r_{nj1},r_{nj2},r_{nj3},r_{nj4})=(\sum_{i=1}^{D}\omega_{ji}x_{nj1}+b_j,\sum_{i=1}^{D}\omega_{ji}x_{nj2}+b_j,\sum_{i=1}^{D}\omega_{ji}x_{nj3}+b_j,\sum_{i=1}^{D}\omega_{ji}x_{nj4}+b_j)\tag{5}$$

$$H_{nj}=(h_{nj1},h_{nj2},h_{nj3},h_{nj4})=\left[\frac{1}{1+\exp(-r_{nj1})},\frac{1}{1+\exp(-r_{nj2})},\frac{1}{1+\exp(-r_{nj3})},\frac{1}{1+\exp(-r_{nj4})}\right]\tag{6}$$

H_{nj} 的 α-截集为

$$[H_{nj}]\alpha=[[H_{nj}]_\alpha^L[H_{nj}]_\alpha^U]=\left[\frac{1}{1+\exp[-[\alpha r_{nj2}+(1-\alpha)r_{nj1}]]},\frac{1}{1+\exp[-[\alpha r_{nj2}+(1-\alpha)r_{nj3}]]}\right]\tag{7}$$

则网络实际模糊输出的 α-截集为

$$[T_n]_\alpha[[T_n]_\alpha^L,[T_n]_\alpha^U]=\Big[\sum_{j=1}^{K}\boldsymbol{\beta}_j[H_{nj}]_\alpha^L,\sum_{j=1}^{K}\boldsymbol{\beta}_j[H_{nj}]_\alpha^U\Big],\quad \boldsymbol{\beta}>0 \tag{8}$$

式中,$\boldsymbol{\beta}$ 待确定,为输出层权值矩阵。

则定义在总体数据集 D 上的误差为

$$E=\sum_{n=1}^{N}E_n=\sum_{n=1}^{N}\int_0^1\|[T_n]_\alpha[Y_n]_\alpha\|^2\mathrm{d}\alpha。 \tag{9}$$

$\mathrm{TraFNR}_{\mathrm{RWN\text{-}I}}$ 通过最小化式(9)来训练网络输出权值 β。因为式(9)中用到了积分,故导致直接从中推导 β 的更新规则将十分困难,故文献[12]采用了近似误差求解 β 的方法。

由于无需反复迭代,He et al 提出的 TraFNRRWN-I 模型取得了较之 $\mathrm{TraFNR}_{\mathrm{BP}}$ 和 $\mathrm{TraFNR}_{\mathrm{RBF}}$ 更快的训练速度,但也有其缺陷。①经过隐含层之后的模糊变量已经变成非梯形,但 $\mathrm{TraFNR}_{\mathrm{RWN\text{-}I}}$ 模型用这个非梯形的模糊变量去逼近目标梯形模糊变量,学习精确度必将受到影响;②权值学习过程采用了近似误差,也直接导致该模型精确度降低。

2 基于期望值简约的梯形模糊数据非线性回归(TraFNRRWN-II)

考虑到 TraFNRRWN-I 模型的上述缺点,提出一种基于期望值简约思想的模糊非线性回归模型,因为网络的训练仍然采用了随机赋权网络,故而模型简记为 TraFNRRWN-II。该模型处理的也是梯形模糊输入输出数据。在该模型中,将梯形模糊输入输出用其期望值进行代替,从而将模糊数据转化为清晰数据,随后利用经典随机赋权网络对输出层权值进行学习。最后再通过一个预先定义的目标模糊输出的宽度矩阵,将网络清晰值输出还原为梯形模糊输出。其网络结构基本与 $\mathrm{TraFNR}_{\mathrm{RWN\text{-}I}}$ 相同,也是一个单隐含层前馈神经网络,但网络的输入输出均为模糊输入输出的期望值,具体过程如下。

首先对输入 $D_n=(X_{n1},X_{X2},\cdots,X_{nD}),X_{nd}=(x_{nd1},x_{nd2},x_{nd3},x_{nd4}),n=1,2,\cdots,N,d=1,2,\cdots,D$, 以及输出 $Y_n=(y_{n1},y_{n2},y_{n3},y_{n4})$,分别计算其期望值

$$[D_n]_E=([x_{n1}]_E,[x_{n2}]_E,\cdots,[x_{nD}]_E) \tag{10}$$

$$[Y_n]_E=\frac{1}{4}(2y_{n1}+2y_{n2}-y_{n3}+y_{n4}) \tag{11}$$

其中

$$[x_{nd}]_E=\frac{1}{4}(2x_{nd1}+2x_{nd2}-x_{nd3}+x_{nd4}) \tag{12}$$

然后用输入输出的期望值代替原输入输出。

则第 n 组数据$(X_{n1},X_{n2},\cdots,X_{nD})$对应第 j 个隐含节点的输入和输出分别为

$$R_{nj}=\sum_{i=1}^{D}\omega_{ji}[X_{nj}]_E \tag{13}$$

$$H_{nj}=\frac{1}{1+\exp(-R_{nj}-b_j)} \tag{14}$$

式中,ω_{ji} 和 b_j 分别为输入层权值和隐含层偏置,被设定为区间[0,1]上的随机数。

网络输出层权值计算如下

$$\beta=[H]_E^{+}[Y]_E \tag{15}$$

式中,$[H]_E^{+}$ 为隐含层输出矩阵$[H]_E$ 的摩尔—彭罗斯广义逆矩阵;$[Y]_E$ 为目标输出的期望值简约矩阵。

$$[Y]_E=[[Y_1]_E,[Y_2]_E,\cdots,[Y_N]_E]^{\mathrm{T}} \tag{16}$$

则网络的实际清晰输出为

$$[T]_E=[H]_E\beta \tag{17}$$

最后,定义一个记录目标模糊输出变量宽度信息的矩阵 $\boldsymbol{P}_Y$,用于还原网络的实际清晰值输出$[T]_E$ 为模糊值输出 $\boldsymbol{T}$,

$$T=\begin{bmatrix}T_1\\T_2\\\vdots\\T_n\end{bmatrix}\begin{bmatrix}t_{11}&t_{12}&t_{12}&t_{14}\\t_{21}&t_{22}&t_{23}&t_{24}\\\vdots&\vdots&\vdots&\vdots\\t_{N1}&t_{N2}&t_{N3}&t_{N4}\end{bmatrix}\tag{18}$$

定义 2.1 对于梯形模糊向量 $\boldsymbol{Y}=[\boldsymbol{Y}_1,\boldsymbol{Y}_2,\cdots,\boldsymbol{Y}_N]^{\mathrm{T}}$,$Y_n=(y_{n1},y_{n2},y_{n3},y_{n4})$,$n=1,2,\cdots,N$,定义其宽度矩阵为

$$\boldsymbol{P}_Y=\begin{bmatrix}y_{12}-y_{11}&y_{13}-y_{12}&y_{14}-y_{13}\\y_{22}-y_{21}&y_{23}-y_{22}&y_{24}-y_{23}\\\vdots&\vdots&\vdots\\y_{N2}-y_{N1}&y_{N3}-y_{N2}&y_{N4}-y_{N3}\end{bmatrix}\tag{19}$$

假设实际模糊输出具有和目标模糊输出相同的宽度,则有下列方程组成立

$$\begin{cases}\dfrac{1}{4}(2t_{n1}+2t_{n2}-t_{n3}+t_{n4})=[T_n]_E\\y_{n2}-y_{n1}=t_{n2}-t_{n1}\\y_{n3}-y_{n2}=t_{n3}-t_{n2}\\y_{n4}-y_{n3}=t_{n4}-t_{n3}\end{cases}\tag{20}$$

故可计算网络的实际模糊值输出

$$\boldsymbol{T}=[4[T_n]_E,P_Y]\begin{bmatrix}2&-1&0&0\\2&1&-1&0\\-1&0&1&-1\\1&0&0&1\end{bmatrix}^{-1}\tag{21}$$

3　仿真实验

将提出的 $\mathrm{TraFNR}_{\mathrm{RWN-II}}$ 与 He et al 提出的 $\mathrm{TraFNR}_{\mathrm{RWN-I}}$ 进行对比实验,两者皆在联想 Y410P,windows 7 操作系统下进行,英特尔处理器 i5-4200M,主频 2.50 GHz,4 GB 内存。比对的内容为算法学习准确度和学习时间,学习准确度的含义是在不同 α 水平下目标输出的 α-截集与实际输出的 α-截集的均方误差

$$E(\alpha)=\frac{\sum_{n=1}^{N}[([T_n]_\alpha^L-[Y_n]_\alpha^L)^2+([T_n]_\alpha^U[Y_n]_\alpha^U)^2]}{N}\tag{22}$$

实验所用数据为来源于 UCI 和 KEEL 中的 12 个真实数据集，参见表 1。数据模糊化的方法采用了 He 在文献[12]中的方法,详细过程参见该文献。$\mathrm{TraFNR}_{\mathrm{RWN-I}}$ 模型中采用了文献[12]中最优的参数,$C=2^{-9}$,两个模型中隐含层节点的个数均为 100。实验采用十次十折交叉验证运算,取 100 次实验结果的平均值作为最后的结果,详见表 2,表 3。

表 2 的结果表明,提出的 $\mathrm{TraFNR}_{\mathrm{RWN-II}}$ 基本和 He et al 提出的 $\mathrm{TraFNR}_{\mathrm{RWN-I}}$ 保持了相当的学习速度。表 3 的结果表明,较之 $\mathrm{TraFNR}_{\mathrm{RWN-I}}$,$\mathrm{TraFNR}_{\mathrm{RWN-II}}$ 具有更好的学习准确度,采用 $\mathrm{TraFNR}_{\mathrm{RWN-I}}$ 模型得到的实验结果其均方误差基本保持在 10^{-2} 量级,而采用 $\mathrm{TraFNR}_{\mathrm{RWN-II}}$ 模型得到的实验结果其均方误差的量级达到了 10^{-34}。

若数据变为其它类型的模糊变量,甚至混合类型的模糊变量,则 TraFNRRWN-I 推导起来将十分困难,而 TraFNRRWN-II 则很容易适用于这种复杂环境,只需要计算各种模糊变量的期望值即可,即 TraFNRRWN-II 还具有更好的扩展能力。

表1 仿真实验数据描述

序号	数据集	属性个数	样例个数	来源
1	Concrete compressive strength	8	1 030	KEEL
2	Daily Electricy Energy(DEE)	6	365	KEEL
3	Electrical-maintenance(ELe-2)	4	1 056	KEEL
4	Energy efficiency	8	768	UCI
5	Friedman	5	1 200	KEEL
6	Housing	14	506	UCI
7	Laser generated	4	993	KEEL
8	Mortgage	15	1 049	KEEL
9	Stock prices	9	950	KEEL
10	Treasury	15	1 049	KEEL
11	Weather Ankara	9	1 609	KEEL
12	Weather lzmir	9	1 461	KEEL

表2 $\text{TraFNR}_{\text{RWN-I}}$和$\text{TraFNR}_{\text{RWN-II}}$的训练和测试时间 s

参数	序号	$\text{TraFNR}_{\text{RWN-I}}$					$\text{TraFNR}_{\text{RWN-II}}$				
		$\alpha=0.90$	$\alpha=0.70$	$\alpha=0.50$	$\alpha=0.30$	$\alpha=0.10$	$\alpha=0.90$	$\alpha=0.70$	$\alpha=0.50$	$\alpha=0.30$	$\alpha=0.10$
训练时间	1	5.41	6.07	5.38	5.18	6.30	1.06	0.28	0.22	0.14	0.28
	2	1.93	1.78	1.58	2.14	2.09	0.11	0.08	0.06	0.08	0.08
	3	3.82	4.15	3.99	3.73	3.93	0.14	0.16	0.20	0.19	0.20
	4	4.40	3.93	4.70	4.12	2.45	0.17	0.17	0.11	0.11	0.11
	5	5.27	4.57	5.60	5.44	5.34	0.31	0.14	0.25	0.22	0.16
	6	3.88	4.40	4.49	4.49	4.52	0.14	0.08	0.12	0.16	0.11
	7	2.42	2.85	3.71	3.54	3.21	0.08	0.12	0.11	0.22	0.12
	8	9.59	11.33	10.28	11.51	11.15	0.20	0.20	0.14	0.23	0.39
	9	5.94	6.74	5.79	6.46	5.79	0.22	0.22	0.22	0.31	0.16
	10	10.78	10.14	9.77	10.62	9.69	0.28	0.17	0.20	0.19	0.36
	11	2.26	1.75	2.00	2.17	2.29	0.05	0.06	0.08	0.05	0.08
	12	10.14	8.64	9.64	7.91	8.91	0.28	0.19	0.22	0.17	0.19
测试时间	1	2.32	1.93	2.07	2.37	2.20	0.02	0.08	0.08	0.05	0.05
	2	0.59	0.62	0.75	0.51	0.53	0.02	0.02	0.02	0.05	0.01
	3	0.97	0.89	0.75	0.70	0.66	0.06	0.02	0.03	0.02	0.05
	4	0.83	0.86	0.86	0.98	0.86	0.03	0.03	0.02	0.03	0.02
	5	0.61	0.66	0.61	0.75	0.81	0.05	0.05	0.05	0.03	0.05
	6	0.80	0.80	0.83	0.80	0.84	0.05	0.02	0.05	0.05	0.03
	7	0.64	0.64	0.73	0.67	0.72	0.02	0.05	0.05	0.05	0.02
	8	1.03	1.06	1.11	1.06	1.05	0.06	0.02	0.03	0.05	0.02
	9	0.95	1.75	2.73	2.96	2.79	0.06	0.06	0.16	0.16	0.11
	10	2.20	3.42	2.70	3.20	3.49	0.06	0.09	0.09	0.20	0.09
	11	0.97	0.76	0.64	1.06	0.69	0.02	0.02	0.02	0.02	0.01
	12	2.37	2.29	1.90	1.26	3.01	0.16	0.11	0.03	0.02	0.14

表 3　$TraFNR_{RWN-I}$ 和 $TraFNR_{RWN-II}$ 的训练和测试误差

集合	序号	$TraFNR_{RWN-I}$（均值 $\times e^{-2}$）					$TraFNR_{RWN-II}$（均值 $\times e^{-34}$）				
		$\alpha=0.90$	$\alpha=0.70$	$\alpha=0.50$	$\alpha=0.30$	$\alpha=0.10$	$\alpha=0.90$	$\alpha=0.70$	$\alpha=0.50$	$\alpha=0.30$	$\alpha=0.10$
训练集	1	10.11	10.57	9.90	10.20	10.05	2.19	2.29	0.99	2.40	2.17
	2	11.27	11.03	11.11	10.63	11.18	1.72	2.37	0.74	2.46	1.61
	3	8.93	8.98	8.69	9.04	8.33	0.68	0.91	0.40	1.06	0.64
	4	14.07	14.05	14.04	13.98	14.11	2.05	0.51	0.51	0.00	1.03
	5	9.76	9.71	9.62	9.51	9.41	2.02	2.91	1.22	2.79	2.12
	6	9.35	9.27	9.34	9.30	9.24	0.02	0.06	0.02	0.06	0.02
	7	18.22	11.05	14.18	11.55	10.19	0.17	0.30	0.15	0.41	0.26
	8	9.61	9.23	9.03	8.92	8.79	0.86	0.83	0.47	0.74	0.59
	9	11.17	11.06	11.01	10.92	10.92	2.14	3.29	1.17	3.00	2.31
	10	12.81	12.63	12.48	12.41	12.45	1.19	1.33	1.04	1.35	1.19
	11	12.14	11.96	11.77	12.31	11.71	2.46	2.17	0.96	2.36	1.77
	12	13.25	12.96	12.61	13.29	12.77	2.22	3.02	1.50	3.18	2.15
测试集	1	10.73	10.87	10.38	10.29	10.14	1.69	2.91	1.05	3.32	3.53
	2	12.19	11.98	11.83	11.70	11.71	2.04	3.41	1.20	3.10	2.28
	3	7.96	7.89	8.51	9.07	8.00	0.25	0.16	0.08	0.25	0.36
	4	13.96	13.85	14.07	13.87	13.87	4.04	5.49	4.04	4.30	3.34
	5	9.71	9.48	9.36	9.26	9.30	2.05	2.82	1.08	2.96	1.94
	6	8.95	8.82	8.81	8.80	8.44	0.06	0.22	0.09	0.22	0.07
	7	13.49	14.57	9.74	10.60	12.65	0.37	0.69	0.11	0.78	0.42
	8	9.96	9.63	9.37	9.24	9.11	1.37	3.62	1.18	3.83	1.33
	9	11.32	11.37	11.25	10.90	10.84	2.21	3.88	0.76	2.89	1.78
	10	13.11	12.88	12.82	12.63	12.73	1.22	1.80	1.10	2.09	1.68
	11	12.88	12.91	12.93	12.94	13.07	1.30	2.16	0.74	1.23	1.54
	12	13.09	13.08	12.97	12.46	12.60	2.40	2.79	1.23	2.97	1.77

4　结论

借鉴 2-型模糊理论中的简约思想，提出了一种基于期望值简约的模糊非线性回归模型。本模型利用了模糊变量的期望值对梯形模糊输入输出进行简约，从而将模糊的数据转化为清晰数据，故而可以利用经典的随机赋权网络对其进行训练。最后再通过一个目标输出数据的宽度信息矩阵将清晰输出还原为模糊输出。因无需引入数据以外的其它信息，较之文献[12]中的模型，此模型具有更好的客观性，从而具有更高的学习准确度。又因为期望值计算的简便性，此模型还具有很好的应用范围。

参　考　文　献

[1]Wang H F, Tsaur R C. Insight of a fuzzy regression model[J]. Fuzzy Sets and Systems, 2000, 112(3): 355-369.

[2]Kimura D, Nii M, Yamaguchi T, et al. Fuzzy nonlinear regression analysis using fuzzified neural networks for fault diagnosis of chemical plants[J]. Journal of Advanced Computational Intelligence and Intelligent Informatics, 2011, 15(3): 336-344.

[3]Gonzalez-Gonzalez D S, Praga Alejo R J, Can-Sifuentes M, et al. A non-linear fuzzy regression for estimating reliability in a degradation process[J]. Applied Soft Computing, 2014, 16: 137-147.

[4]Liu Y Y, Zhou J, Chen Y Z. Using fuzzy non-linear regression to identify the degree of compensation among customer requirements in QFD[J]. Neurocomputing, 2014, 142: 115-124.

[5]Wang X Z. Learning from big data with uncertainty[J]. Journal of Intelligent & Fuzzy Systems, 2015, 28(5): 2329-2330.

[6]Ishibuchi H, Tanaka H. Fuzzy regression analysis using neural networks[J]. Fuzzy Sets and Systems,1992, 50(3): 257-265.

[7]Rumelhart D E, Hinton G E, Williams R J. Learning representations by back-propagating errors[J]. Nature,1986, 323: 533-536.

[8]Ishibuchi H, Tanaka H. An architecture of neural networks with interval weights and its application to fuzzy regression analysis[J]. Fuzzy Sets and Systems,1993, 57(1): 27-39.

[9]Zhao S X, Wang X Z. Extreme learning machine for interval-valued data[C]//Proceedings of 2014 International Conference on Machine Learning and Cybernetics. Netherlands: Springer, 2014:388-399.

[10]Ishibuchi H, Kwon K, Tanaka H. A learning algorithm of fuzzy neural network with triangular fuzzy weights[J]. Fuzzy Sets and Systems,1995, 71(3): 277-293.

[11]Cheng C B, Lee S. Fuzzy regression with radial basis function network[J]. Fuzzy Sets and Systems,2001, 119(2): 291-301.

[12]He Y L, Wang X Z, Huang J Z X. Fuzzy nonlinear regression analysis using a random weight network[J]. Information Sciences, 2016, 364/365: 222-240.

[13]Schmidt W F, Kraaijveld M A, Duin R P W. Feedfirward neural networks with random weights[C]//Proceedings of 11th IAPR International Conference on Pattern Recognition, Conference B: Pattern Recognition Methodology and Systems. Netherlands: IEEE, II, 1992:1-4.

[14]Huang G B, Slew C K. Extreme learning machine: RBF network case[C]//Proceedings of the 8th International Conference on Control, Automation, Robotics and Vision, Xiamen: IEEE, 2004, 2: 1029-1036.

[15]Huang G B, Zhu Q Y, Slew C K. Extreme learning machine: theory and application[J]. Neurocomputing, 2006, 70 (1-3): 489-501.

[16]Ishibuchi H, Morioka K, Tanaka H. A fuzzy neural network with trapezoid fuzzy wights[C]//Proceedings of the Third IEEE conference on Fuzzy Systems. Orlando: IEEE, 1994:228-233.

[17]刘彦奎,陈艳菊.模糊优化方法与应用[M].北京:科学出版社,2013:145.

[18]Campos L, González A. A subjective approach for ranking fuzzy numbers[J]. Fuzzy Sets and Systems, 1989, 29(2): 145-153.

[19]Dubois D, Prade H. The mean value of a Fuzzy number[J]. Fuzzy Sets and Systems, 1987, 24(3): 279-300.

[20]González A. A study of the ranking function approach through mean values[J]. Fuzzy Sets and Systems, 1990, 35: 29-41.

[21]Heilpern S. The expectd value of a fuzzy number[J]. Fuzzy Sets and Systems, 1992, 47(1): 81-86.

[22]Liu B, Liu Y K. Expected value of fuzzy variable and fuzzy expected value models[J]. IEEE transactions on Fuzzy Systems. 2002, 10(4): 445-450.

Expectation Reduction-based Fuzzy Nonlinear Regression for Trapezoidal Fuzzy Number Data

Zhao Shixin, Chen Xiyuan, Wang Rongrong

(Department of Mathematics and Physics, Shijiazhuang Tiedao University, Shijiazhuang 050043, China)

Abstract: Fuzzy nonlinear regression (FNR) analysis is an effective method for research of uncertainty data. Inspired by the idea of reduction in 2-fuzzy mathematics theory, a fuzzy nonlinear regression using expectation reduction for trapezoidal fuzzy number data is proposed. In this model, the trapezoidal fuzzy inputs and outputs are replaced by their expectations, respectively, which can be trained with classical random weight neural network. Finally, the crisp real outputs are recovered to trapezoidal fuzzy outputs by using a width matrix of original target outputs. The experiment results show that the proposed model obtains better prediction accuracy and wider extension.

Key words: expectation reduction; trapezoidal fuzzy number; random weight network; fuzzy nonlinear regression