

折线模糊神经网络的共轭梯度算法

何 英, 王贵君

(天津师范大学数学科学学院, 天津 300387)

摘 要: 为了近似实现模糊数的非线性运算及提高神经网络的逼近精度, 引入折线模糊数和折线模糊神经网络, 并依据折线模糊数的扩展运算对经典共轭梯度算法进行改进, 使该算法在迭代过程中通过一维非精确 Armijo-Goldstein 线性搜索方法获得优化学习常数, 进而在折线模糊神经网络环境下设计了折线模糊共轭梯度算法. 最后, 通过模拟实例说明了该算法具有计算复杂度低、收敛速度快等特性.

关键词: 折线模糊数; 折线模糊神经网络; 共轭梯度算法; A-G 算法

中图分类号: TP183; O159 **文献标识码:** A **文章编号:** 0372-2112 (2012)10-2079-06

电子学报 URL: <http://www.ejournal.org.cn>

DOI: 10.3969/j.issn.0372-2112.2012.10.029

Conjugate Gradient Algorithm of the Polygonal Fuzzy Neural Networks

HE Ying, WANG Gui-jun

(School of Mathematics Sciences, Tianjin Normal University, Tianjin 300387, China)

Abstract: In order to realize an nonlinear operations between fuzzy numbers and raise the accuracy of the approximation of fuzzy neural networks, the polygonal fuzzy numbers and a polygonal fuzzy neural network are introduced, and according to their extension operations to improve the classic conjugate gradient algorithm, in the iterative process, the optimal constants of the algorithm are obtained through the one-dimensional inexactitude A-G linear search method, and then a polygonal fuzzy conjugate gradient algorithm is designed under environment of polygonal fuzzy neural networks. Finally, utilizing a simulation examples, some good characteristics of this algorithm are illustrated, for example, low computational complexity, fast convergence and so on.

Key words: polygonal fuzzy numbers; polygonal fuzzy neural networks; conjugate gradient algorithm; A-G algorithm

1 引言

自从 1994 年 Buckley^[1]首次提出模糊神经网络的泛逼近性问题以来, 关于网络的泛逼近性及其学习算法的研究与日俱增^[2,3]. 事实上, 模糊神经网络的逼近性能直接由网络的连接权决定, 因此, 确定适宜的连接权是网络学习算法的主要目标之一. 2009 年, Aliev^[4]针对经常性模糊神经网络提出了一种差分进化优化 (DEO) 学习算法, 并且通过模拟基准预测和识别问题说明了该方法的有效性; Castro^[5]等人采用模糊 If-Then 规则来处理信息的不确定性和不精确性, 并用混合学习算法来训练间隔型的模糊神经网络. 2010 年, 文献[6]通过分析非线性函数各输入参数对函数值的影响, 提出了一种基于扩展傅立叶振幅灵敏度方法的神经网络隐含层神经元快速修剪算法. 2011 年, 文献[7]中提出了基于自适应动

态调整惯性权重的粒子群算法的 BP 网络算法, 算法根据适应度值的改变情况来调整惯性权重. 然而, 无论是常规的模糊 BP 算法^[8], 还是改进后的模糊 BP 算法, 在学习常数和动量常数的选择上都没有可依据的规则, 从而具有很大的盲目性. 如果单从训练方法上来说, 文献[9]给出的共轭梯度算法将会从一定程度上减少这种训练的盲目性. 当然, 这些有益结果可进一步见之于文献[10~12].

2002 年, 刘普寅教授首次通过引入折线模糊数概念提出了折线模糊神经网络^[13], 文献[14]通过引入 K-积分模研究了该网络的泛逼近性, 但对其学习算法并不是涉及很多. 因此, 本文将根据折线模糊数及其扩张运算, 结合 A-G 线性搜索方法来获得动态的学习常数, 设计折线模糊神经网络的共轭梯度算法, 通过算法对网络的权值和阈值进行优化计算, 结果显示该算法具有收敛

速度快,稳定性好等特点.

2 折线模糊数

模糊数的应用一直是模糊理论中一个重要问题,但不幸的是模糊算术运算极其复杂,即使最简单的三角模糊数或梯形模糊数运算起来也很困难,究其原因主要是依据 Zadeh 扩展原理的四则运算不满足封闭性,这固然提出如何近似地实现一般模糊数之间的非线性运算的问题.2011 年,文献[14]采取等距划分的方法对文献[13]的“ n -对称折线模糊数”定义做了改进,并利用折线模糊数扩展原理研究了折线模糊神经网络对一类 μ -可积函数的泛逼近性,这为进一步探索广义推理网络和模糊系统奠定了基础.

设 \mathbf{R} 表示实数集, \mathbf{R}^+ 表示非负实数集, \mathbf{N} 表自然数集, $\text{sign}(\cdot)$ 表示符号算子, $F_0(\mathbf{R})$ 表示 \mathbf{R} 上全体模糊数构成的集合, $\|\cdot\|$ 指欧氏范数.

定义 1^[13,14] 设 $\tilde{A} \in F_0(\mathbf{R})$, 给定 $n \in \mathbf{N}$, 在 y -轴将区间 $[0, 1]$ 分成 n 等份, 分点 $x_i = i/n, i = 1, 2, \dots, n-1$, 若存在一组有序实数 $a_0^1, a_1^1, \dots, a_n^1, a_n^2, \dots, a_1^2, a_0^2 \in \mathbf{R}$ 且 $a_0^1 \leq a_1^1 \leq \dots \leq a_n^1 \leq a_n^2 \leq \dots \leq a_1^2 \leq a_0^2, \tilde{A}(a_i^q) = \frac{i}{n}, q = 1, 2$, 使 $\tilde{A}(x)$ 在每个对应区间 $[a_{i-1}^1, a_i^1]$ 和 $[a_i^2, a_{i-1}^2]$ 上取直线段, 即, $\forall x \in \mathbf{R}, i = 1, 2, \dots, n$, 界定

$$\tilde{A}(x) = \begin{cases} \frac{i-1}{n} + \frac{x - a_{i-1}^1}{n(a_i^1 - a_{i-1}^1)}, & x \in [a_{i-1}^1, a_i^1] \\ 1, & x \in [a_n^1, a_n^2] \\ \frac{i-1}{n} + \frac{a_{i-1}^2 - x}{n(a_{i-1}^2 - a_i^2)}, & x \in [a_i^2, a_{i-1}^2] \\ 0, & \text{否则} \end{cases}$$

则称 \tilde{A} 为 \mathbf{R} 上一个 n -折线模糊数, 简记为 $\tilde{A} = (a_0^1, a_1^1, \dots, a_n^1, a_n^2, \dots, a_1^2, a_0^2)$, 参看图 1.

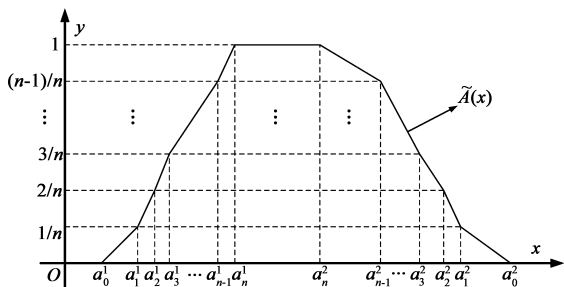


图1 n -折线模糊数的图像

显然, 一个 n -折线模糊数 \tilde{A} 可由 x -轴上有限个点 $a_0^1, a_1^1, \dots, a_n^1, a_n^2, \dots, a_1^2, a_0^2$ 完全确定. 记 $F_{oc}^m(\mathbf{R})$ 为 \mathbf{R} 上全体 n -折线模糊数构成的集.

定义 2^[13] 给定 $n \in \mathbf{N}$, 若 $\tilde{A}, \tilde{B} \in F_{oc}^m(\mathbf{R})$, 设 $\tilde{A} = (a_0^1, a_1^1, \dots, a_n^1, a_n^2, \dots, a_1^2, a_0^2)$, 并且 $\tilde{B} = (b_0^1, b_1^1, \dots, b_n^1, b_n^2, \dots, b_1^2, b_0^2)$, 界定 n -折线模糊数的扩张运算如下

$$\begin{aligned} (1) \tilde{A} + \tilde{B} &= (a_0^1 + b_0^1, a_1^1 + b_1^1, \dots, a_n^1 + b_n^1, a_n^2 + b_n^2, \dots, a_1^2 + b_1^2, a_0^2 + b_0^2); \\ (2) \tilde{A} - \tilde{B} &= (a_0^1 - b_0^2, a_1^1 - b_1^2, \dots, a_n^1 - b_n^2, a_n^2 - b_n^1, \dots, a_1^2 - b_1^1, a_0^2 - b_0^1); \\ (3) \tilde{A} \cdot \tilde{B} &= (c_0^1, c_1^1, \dots, c_n^1, c_n^2, \dots, c_1^2, c_0^2); \end{aligned}$$

其中 $c_i^1 = a_i^1 b_i^1 \wedge a_i^1 b_i^2 \wedge a_i^2 b_i^1 \wedge a_i^2 b_i^2$, 且 $c_i^2 = a_i^1 b_i^1 \vee a_i^1 b_i^2 \vee a_i^2 b_i^1 \vee a_i^2 b_i^2, i = 0, 1, \dots, n$;

$$(4) k \geq 0 \text{ 时, } k \cdot \tilde{A} = (ka_0^1, ka_1^1, \dots, ka_n^1, ka_n^2, \dots, ka_1^2, ka_0^2).$$

此外, 若转移函数 $\sigma: \mathbf{R} \rightarrow \mathbf{R}$ 是单调的, 则 σ 可作如下扩展 $\sigma: F_{oc}^m(\mathbf{R}) \rightarrow F_{oc}^m(\mathbf{R})$, 即

$$\sigma(\tilde{A}) = \begin{cases} (\sigma(a_0^1), \dots, \sigma(a_n^1), \sigma(a_n^2), \dots, \sigma(a_0^2)), & \text{若 } \sigma \text{ 不减} \\ (\sigma(a_0^2), \dots, \sigma(a_n^2), \sigma(a_n^1), \dots, \sigma(a_0^1)), & \text{若 } \sigma \text{ 不增} \end{cases}$$

显然, $F_{oc}^m(\mathbf{R})$ 关于运算“ \pm, \cdot ”和 σ 是封闭的, 其运算要比基于 Zadeh 扩展原理的相应运算简单的多, 而且保持了线性运算性质, 这无疑是引进折线模糊数的关键所在.

注 1 为方便对误差函数进行求偏导数运算及表示, 在 $F_{oc}^m(\mathbf{R})$ 中引进一种等价度量 D_E , 亦即 $\forall \tilde{X} = (x_0^1, x_1^1, \dots, x_n^1, x_n^2, \dots, x_1^2, x_0^2), \tilde{Y} = (y_0^1, y_1^1, \dots, y_n^1, y_n^2, \dots, y_1^2, y_0^2) \in F_{oc}^m(\mathbf{R})$, 界定

$$D_E(\tilde{X}, \tilde{Y}) = \left(\sum_{i=0}^n ((x_i^1 - y_i^1)^2 + (x_i^2 - y_i^2)^2) \right)^{1/2}.$$

3 折线模糊神经网络

折线模糊神经网络(简称折线 FNN)是指网络的连接权及阈值在 $F_{oc}^m(\mathbf{R})$ 中取值, 而内部运算基于 n -折线模糊数扩张运算的一类网络系统, 它通过确定折线模糊数的有限个点来完成模糊信息处理. 因此, 应用 n -折线模糊数的线性运算来近似替代模糊数的运算和度量, 进而实现折线 FNN 的近似表示有其重要意义.

本文以单输入单输出(SISO)的三层前向折线 FNN 为研究对象, 该系统的结构如下: 设系统的输入、输出神经元是线性的, 而隐含层神经元有转移函数 $\sigma: \mathbf{R} \rightarrow \mathbf{R}$, 其中输入信号 \tilde{X} , 模糊连接权 \tilde{U}_j, \tilde{V}_j 以及隐含层模糊阈值 $\tilde{\Theta}_j$ 均在 $F_{oc}^m(\mathbf{R})$ 中取值. 下文中设转移函数 σ 是连续递增 Sigmoidal 型函数, 且处处可微, 则三层前向折线 FNN 系统的输入输出(I/O)关系表示为

$$\begin{aligned} \tilde{Y} &= F_m(\tilde{X}) = \sum_{j=1}^p \tilde{V}_j \cdot \sigma(\tilde{U}_j \cdot \tilde{X} + \tilde{\Theta}_j) \\ p &\in \mathbf{N}, \tilde{X}, \tilde{V}_j, \tilde{U}_j, \tilde{\Theta}_j \in F_{oc}^m(\mathbf{R}). \end{aligned}$$

以下为讨论方便, 设 $(\tilde{X}(1), \tilde{\Theta}(1)), (\tilde{X}(2), \tilde{\Theta}(2)), \dots, (\tilde{X}(L), \tilde{\Theta}(L))$ 是用于训练的折线模糊模式对, 其中 $\tilde{X}(l), \tilde{\Theta}(l) \in F_{oc}^m(\mathbf{R}^+)$, $\tilde{X}(l)$ 表示网络的输入, $\tilde{\Theta}(l)$ 表示网络的期望输出, 网络实际输出为 $\tilde{Y}(l)$, 即 $\tilde{Y}(l) =$

$F_m(\tilde{X}(l))$, 其中 $l=1, 2, 3, \dots, L$. 不妨记

$$\begin{aligned}\tilde{X}(l) &= (x_0^1(l), x_1^1(l), \dots, x_n^1(l), x_n^2(l), \dots, x_1^2(l), \\ &x_0^2(l)), \\ \tilde{Y}(l) &= (y_0^1(l), y_1^1(l), \dots, y_n^1(l), y_n^2(l), \dots, y_1^2(l), \\ &y_0^2(l)), \\ \tilde{O}(l) &= (o_0^1(l), o_1^1(l), \dots, o_n^1(l), o_n^2(l), \dots, o_1^2(l), \\ &o_0^2(l)).\end{aligned}$$

根据注 1 给出的度量, 我们定义折线 FNN 的误差函数 E 如下

$$\begin{aligned}E &= \frac{1}{2} \sum_{l=1}^L D_E(\tilde{O}(l), \tilde{Y}(l))^2 \\ &= \frac{1}{2} \cdot \sum_{l=1}^L \left(\sum_{i=0}^n ((o_i^1(l) - y_i^1(l))^2 + (o_i^2(l) - y_i^2(l))^2) \right)\end{aligned}\quad (1)$$

注 2 事实上, 折线 FNN 系统的自身结构表达式就是一个集折线模糊数加法和乘法的运算体系, 每个 n -折线模糊数由 $(2n+2)$ 个参数唯一决定. 因此, 对输入信号 $\tilde{X}(l)$, 可以通过学习逐渐调节模糊连接权 \tilde{U}_j, \tilde{V}_j 及模糊阈值 $\tilde{\Theta}_j$, 使实际输出 $\tilde{Y}(l)$ 近似于 $\tilde{O}(l)$.

此外, 为方便记忆和表示, 将折线 FNN 所有可调参数 $u_i^q(j), v_i^q(j), \theta_i^q(j)$ ($i=0, 1, 2, \dots, n; j=1, 2, \dots, p; q=1, 2$) 写成一个参数向量形式

$$\mathbf{W} = (u_0^1(1), \dots, u_0^2(1), \dots, u_0^1(p), \dots, u_0^2(p), v_0^1(1), \dots, v_0^2(1), \dots, v_0^1(p), \dots, v_0^2(p), \theta_0^1(1), \dots, \theta_0^2(1), \dots, \theta_0^1(p), \dots, \theta_0^2(p)) = (w_1, w_2, \dots, w_{S_N}).$$

故由式(1)定义的误差函数 E 可表示为 $E(\mathbf{W})$.

引理 1^[15] 设 $E(\mathbf{W})$ 是如式(1)定义的误差函数, 则 $E(\mathbf{W})$ 在 \mathbf{R}^{S_N} 中几乎处处可微, 且其偏导数 $\frac{\partial E}{\partial v_i^q(j)}$, $\frac{\partial E}{\partial \theta_i^q(j)}$, $\frac{\partial E}{\partial u_i^q(j)}$ 均存在, $i=0, 1, \dots, n; j=1, 2, \dots, p; q=1, 2$. (这里, 关于偏导数的详细表达式可参见文献[15]).

设 $\nabla E(\mathbf{W}[t])$ 表示误差函数 $E(\mathbf{W})$ 的梯度向量, 则 $\nabla E(\mathbf{W}[t])$ 可表为如下形式

$$\nabla E(\mathbf{W}[t]) = \left(\frac{\partial E(\mathbf{W})}{\partial w_1}, \frac{\partial E(\mathbf{W})}{\partial w_2}, \frac{\partial E(\mathbf{W})}{\partial w_3}, \dots, \frac{\partial E(\mathbf{W})}{\partial w_{S_N}} \right).$$

4 折线模糊共轭梯度算法

因折线 FNN 的输入、连接权及阈值都取值于折线模糊数, 所以依折线模糊数设计连接权及阈值的学习算法要比以往模糊神经网络简单明了. 对此, 刘普寅^[15]曾提出了一种变学习常数变动量因子模糊 BP 算法, 但该算法有收敛速度低且易于陷入局部极小点等弱点. 事实上, 由于共轭梯度法仅需利用一阶导数信息, 算法

简单且易于编程, 并比较适合于求解具有大规模可调参数的权值优化问题. 此外, 依据文献[15]中定理 2.7 及引理 1 给出的误差函数 $E(\mathbf{W})$ 的偏导数计算公式, 只要在算法迭代过程中保证迭代向量 $\mathbf{W}[t+1] \in W_m$, 可以在非零空间 W_m 中计算误差函数 $E(\mathbf{W})$ 的梯度向量 $\nabla E(\mathbf{W})$, 其中 $W_m = \{\mathbf{W} \in \mathbf{R}^N | \mathbf{W} \neq \mathbf{0}\}$.

本文将针对折线模糊神经网络, 根据传统共轭梯度算法以及 n -折线模糊数的扩张运算来设计下述折线模糊共轭梯度算法.

Step 1 按照 n -折线模糊数的序关系初始化参数向量 $\mathbf{W} = \mathbf{W}[0] = (w_1, w_2, \dots, w_{S_N}) \in W_m$, 并令 $t=0$, 给定精度 $\epsilon > 0$.

Step 2 令方向向量 $\mathbf{h}[t] = -\nabla E(\mathbf{W}[t])$, 如果 $\|\nabla E(\mathbf{W}[t])\| < \epsilon$, 转 Step 8, 否则转下一步.

Step 3 通过一维搜索求 $\eta[t] \in \mathbf{R}^+$, 使得 $E(\mathbf{W}[t] + \eta[t] \cdot \mathbf{h}[t]) = \min_{\lambda > 0} \{E(\mathbf{W}[t] + \lambda \cdot \mathbf{h}[t])\}$ (2)

Step 4 令参数向量

$$\mathbf{W}[t+1] = \mathbf{W}[t] + \eta[t] \cdot \mathbf{h}[t]$$

调整参数向量 $\mathbf{W}[t+1]$, 使之每个分量都非零.

Step 5 对任意 $j=1, \dots, p$, 判断 $\tilde{U}_j[t+1], \tilde{V}_j[t+1], \tilde{\Theta}_j[t+1] \in F_{oc}^m(\mathbf{R})$? 若是转 Step 6; 否则用选择排序的方法将 $\tilde{U}_j[t+1], \tilde{V}_j[t+1], \tilde{\Theta}_j[t+1]$ 中的元素按升序排列, 转下一步.

Step 6 如果 $\|\nabla E(\mathbf{W}[t+1])\| < \epsilon$, 则转 Step 8; 否则转下一步.

Step 7 令搜索方向向量 $\mathbf{h}[t+1]$ 为,

$$\mathbf{h}[t+1] = -\nabla E(\mathbf{W}[t+1]) + \beta[t] \cdot \mathbf{h}[t]$$

$t=t+1$, 转 Step 3, 其中参量 $\beta[t]$ 的选取采用 Fletcher-Reeves 公式

$$\beta[t] = \frac{\|\nabla E(\mathbf{W}[t+1])\|^2}{\|\nabla E(\mathbf{W}[t])\|^2}.$$

Step 8 输出参数向量 $\mathbf{W}[t+1]$ 的值.

注 3 上述算法中 $\mathbf{W}[t], \mathbf{h}[t]$ 为第 t 步所对应的参数向量, $\eta[t], \beta[t]$ 表示第 t 步所对应的实参数. 此外, 算法中 Step 4 ~ Step 5 是针对折线模糊神经网络可调参数做的设计, $\tilde{V}_j[t+1], \tilde{U}_j[t+1], \tilde{\Theta}_j[t+1]$ 表示隐含层神经元 j 在第 $(t+1)$ 步迭代所对应的连接权和阈值.

值得注意的是: 实施上述学习算法并适当排序后, 可调实参数须满足下述不等式

$$\begin{aligned}u_0^1(j)[t+1] &\leq u_1^1(j)[t+1] \leq \dots \leq u_n^1(j)[t+1] \leq \\ u_0^2(j)[t+1] &\leq \dots \leq u_1^2(j)[t+1] \leq u_0^2(j)[t+1]; \\ v_0^1(j)[t+1] &\leq v_1^1(j)[t+1] \leq \dots \leq v_n^1(j)[t+1] \leq \\ v_0^2(j)[t+1] &\leq \dots \leq v_1^2(j)[t+1] \leq v_0^2(j)[t+1]; \\ \theta_0^1(j)[t+1] &\leq \theta_1^1(j)[t+1] \leq \dots \leq \theta_n^1(j)[t+1] \leq \\ \theta_0^2(j)[t+1] &\leq \dots \leq \theta_1^2(j)[t+1] \leq \theta_0^2(j)[t+1].\end{aligned}$$

因此,参数向量 $\mathbf{W}[t]$ 中元素每隔 $(2n+2)$ 个分量必须满足上述由小到大的序关系。

注4 由上述算法,若设动量常数 $\alpha[t] = (\eta[t] \cdot \beta[t-1]) / \eta[t-1]$, 那么迭代参数向量 $\mathbf{W}[t+1] = \mathbf{W}[t] - \eta[t] \cdot \nabla E(\mathbf{W}[t]) + \alpha[t] \cdot (\mathbf{W}[t] - \mathbf{W}[t-1])$, 且动量常数 $\alpha[t]$ 随着学习常数 $\eta[t]$ 在每一步都进行调节,故实现该算法的关键因素之一是如何求得学习常数 $\eta[t]$ 。

下面,我们应用 Armijo-Goldstein(A-G)线性搜索来求得 Step 3 中的 $\eta[t]$, 由于在迭代的每一步都会求得相应的 $\eta[t]$, 故不考虑步数 t , 令 $\mathbf{W}[t] = \mathbf{W}$, $\eta[t] = \eta$, $\mathbf{h}[t] = \mathbf{h}$. 设 $\rho \in (0, 1/2)$, 记相应的 A-G 准则如下

$$\begin{aligned} & \text{(i)} \quad E(\mathbf{W} + \eta\mathbf{h}) \leq E(\mathbf{W}) + \eta\rho \cdot \langle \mathbf{h}, \nabla E(\mathbf{W}) \rangle, \\ & \text{(ii)} \quad E(\mathbf{W} + \eta\mathbf{h}) \geq E(\mathbf{W}) + \\ & \quad \eta(1-\rho) \cdot \langle \mathbf{h}, \nabla E(\mathbf{W}) \rangle \quad (3) \end{aligned}$$

其中对任意 $\eta > 0$, 记 $\varphi(\eta) = E(\mathbf{W} + \eta\mathbf{h})$, 且设 $\varphi'(0) = \langle \mathbf{h}, \nabla E(\mathbf{W}) \rangle$, 下面我们给出应用(A-G)不精确线性搜索方法求 $\eta[t]$ 的具体步骤:

Step 1 选取初始数据,在搜索区间 $[0, +\infty)$ 中取定初始点 η_0 , 计算 $\varphi(0), \varphi'(0)$, 给出 $t > 1, \rho \in (0, \frac{1}{2})$, 令 $a_0 = 0, b_0 = +\infty, k = 0$.

Step 2 检测式(3)的标准(i), 计算 $\varphi(\eta_k)$, 若 $\varphi(\eta_k) \leq \varphi(0) + \rho\eta_k\varphi'(0)$, 转 Step 3; 否则, 令 $a_{k+1} = a_k, b_{k+1} = \eta_k$, 转 Step 4.

Step 3 检测式(3)的标准(ii), 若 $\varphi(\eta_k) \geq \varphi(0) + (1-\rho)\eta_k\varphi'(0)$, 停止迭代, 输出 η_k , 否则, 令 $a_{k+1} = \eta_k, b_{k+1} = b_k$, 若 $b_{k+1} < +\infty$, 转 Step 4; 否则令 $\eta_{k+1} = t\eta_k, k = k+1$, 转 Step 2.

Step 4 选取 $\eta_{k+1} = (a_{k+1} + b_{k+1})/2$ 为新探索点, 令 $k = k+1$, 转 Step 2.

注5 在求学习常数 $\eta[t]$ 的过程中, 设参数 $\rho \in (0, 1/2)$ 是必须的. 事实上, 如若不采用此限制, 将会影响 A-G 算法的超线性收敛, 甚至抑制该算法的有效性. 另外, 虽然 A-G 准则中(ii)可能把 η 的极小值排斥, 但实际上这种情形很少出现, 因此, 它还是一个时常被采用的准则。

5 模拟实例

本节将利用折线 FNN 来模拟一个单输入单输出(SISO)模糊推理模型, 该模型可应用于汽车速度的自动控制、集装箱起重机的自动操作等实际问题. 推理规则库由 L 条模糊推理规则组成, 其中, 前件折线模糊集为 $\tilde{X}(l)$, 后件折线模糊集为 $\tilde{D}(l)$. 设 $L = 5, n = 3$, 给定 $\tilde{X}(l)$ 和 $\tilde{D}(l)$ 的隶属函数曲线分别如图 2 和图 3 所示.

现取精度 $\varepsilon = 0.001$, 用 I/O 关系表示模糊推理的前

件和后件, 给出应用于折线 FNN 训练的 3-折线模糊数模式对 $(\tilde{X}(1), \tilde{D}(1)), (\tilde{X}(2), \tilde{D}(2)), \dots, (\tilde{X}(5), \tilde{D}(5))$ 如表 1 和表 2.

取隐含层转移函数为 $\sigma(x) = 0, x < 0; \sigma(x) = \frac{x^2}{1+x^2}, x \geq 0$, 取定前向折线 FNN 隐含层个数为 $p = 10$.

表 1 折线 FNN 的实际输入

l 值	实际输入 $\tilde{X}(l)$
1	(0,0,0,0,0,0,2,0,5,1)
2	(0,0,5,0,8,1,1,1,2,1,5,2)
3	(1,1,5,1,8,2,2,2,2,2,5,3)
4	(2,2,5,2,8,3,3,3,2,3,5,4)
5	(3,3,5,3,8,4,4,4,4,4,4)

表 2 折线 FNN 的期望输出

l 值	期望输出 $\tilde{D}(l)$
1	(0,0,0,0,0,0,0,0,5,0,15,0,4,0,5)
2	(0,3,0,4,0,65,0,75,0,85,0,95,1,2,1,3)
3	(1,1,1,2,1,45,1,55,1,65,1,75,2,2,1)
4	(1,9,2,2,25,2,35,2,45,2,55,2,8,2,9)
5	(2,7,2,8,3,05,3,15,3,2,3,2,3,2,3,2)

现在, 我们应用上述折线模糊共轭梯度算法, 经过迭代 225 步后, 通过学习的折线 FNN 对应于网络输入 $\tilde{X}(l) (l = 1, 2, 3, 4, 5)$ 的实际输出如图 4 和表 3 所示.

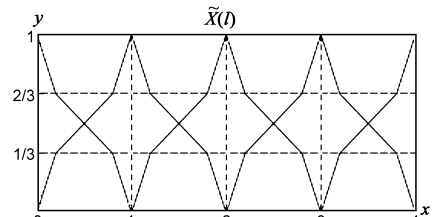


图 2 模糊推理前件 3-折线模糊数

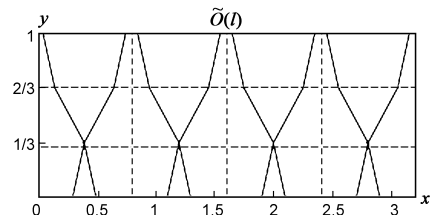


图 3 模糊推理后件 3-折线模糊数

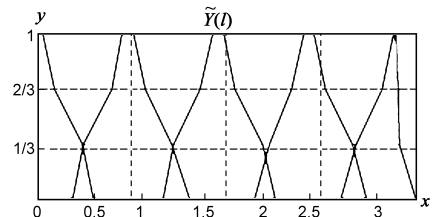


图 4 实际输出 3-折线模糊数

比较实际输出 $\tilde{Y}(l)$ 与期望输出 $\tilde{D}(l)$ 可知, 该折线 FNN 可以在较高精度下实现上述 SISO 函数关系. 下面图 5 给出了误差函数 E 与迭代步数 t 之间的关系曲线, 其中误差曲线的微小波动主要是由于在迭代的每一步

中,通过 A-G 算法找到的只是合适的学习常数,并非最优的学习常数.

表 3 3-折线 FNN 的实际输出

l 值	实际输出 $\hat{Y}(l)$
1	(0.0017, 0.0023, 0.0032, 0.0045, 0.0491, 0.1551, 0.4007, 0.4982)
2	(0.3118, 0.4069, 0.6600, 0.7483, 0.8531, 0.9559, 1.1903, 1.3500)
3	(1.1123, 1.2056, 1.4398, 1.5514, 1.6510, 1.7487, 1.9932, 2.1078)
4	(1.9203, 2.0531, 2.2489, 2.3573, 2.4482, 2.5493, 2.7884, 2.9142)
5	(2.6831, 2.8205, 3.0501, 3.1511, 3.1714, 3.1921, 3.2035, 3.3506)

事实上,由图 5 可见,在折线模糊神经网络中,折线模糊共轭梯度算法的误差函数 E 在整体上按一定的速度下降,并且在迭代 200 步之后就达到了较高的精度,这比文献[15]所给的变学习常数变动力因子模糊 BP 算法的收敛速度要快,同时,该折线模糊共轭梯度算法也能收敛到误差函数的全局最小点,并且具有较好的稳定性,这为进一步合理地设计学习算法奠定了一定的理论基础.

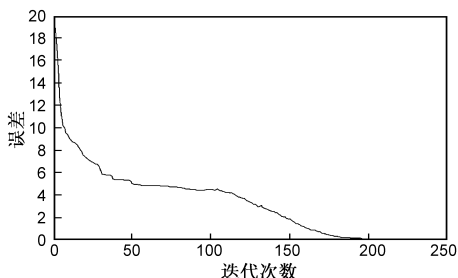


图 5 迭代 225 步误差函数曲线

6 结论

本文利用 n -折线模糊数的扩张运算及传统共轭梯度算法设计了折线模糊神经网络的学习算法,并通过模拟实例验证了该算法的有效性.事实上,该算法中采用 A-G 准则求动态的学习常数,但有时 A-G 准则有可能将最优的学习常数排除在搜索区间之外,造成算法损伤.因此,与此相关的进一步研究是如何为折线模糊神经网络设计更多简单实用的学习算法,例如,如果限定若干条件选择更合适的转移函数,或结合遗传算法来设计折线模糊神经网络的权值优化算法.此外,多输入单输出折线模糊神经网络的学习算法也是值得进一步研究的问题.

参考文献

[1] Buckley J J, Hayashi Y. Can fuzzy neural nets approximate continuous fuzzy functions [J]. *Fuzzy Sets and Systems*, 1994, 61(1): 43 - 51.

[2] Liu P Y. Universal approximation of continuous analyses fuzzy valued functions by multi-layer regular fuzzy neural networks [J]. *Fuzzy Sets and Systems*, 2001, 119(2): 303 - 311.

[3] Aliev R A, Fazlollahi B, Vahidov R. Genetic algorithm based

learning of fuzzy neural networks. Part 1: feed forward fuzzy neural networks [J]. *Fuzzy Sets and Systems*, 2001, 118(2): 351 - 358.

[4] Aliev R A, Guirimov B G, Aliev R R, et al. Evolutionary algorithm-based learning of fuzzy neural networks. Part 2: Recurrent fuzzy neural networks [J]. *Fuzzy Sets and Systems*, 2009, 160(17): 2553 - 2566.

[5] Castro J R, Castillo O, Melin P, et al. A hybrid learning algorithm for a class of interval type-2 fuzzy neural networks [J]. *Information Sciences*, 2009, 179(13): 2175 - 2193.

[6] 乔俊飞,李森,刘江.一种神经网络快速修剪算法[J].*电子学报*, 2010, 38(4): 830 - 834.
Qiao Jun-fei, Li Miao, Liu Jiang. A fast pruning algorithm for neural network [J]. *Acta Electronica Sinica*, 2010, 38(4): 830 - 834. (in Chinese)

[7] 张丹,韩胜菊,李建等.基于改进粒子群算法的 BP 算法的研究[J].*计算机仿真*, 2011, 28(2): 147 - 150.
Zhang Dan, Han Sheng-ju, Li Jian, et al. BP algorithm based on improved particle swarm optimization [J]. *Computer Simulation*, 2011, 28(2): 147 - 150. (in Chinese)

[8] 桂延宁,焦李成,张福顺.基于小波和 BP 神经网络的无线电探测目标识别技术[J].*电子学报*, 2003, 31(12): 1811 - 1814.
Gui Yan-ning, Jiao Li-cheng, Zhang Fu-shun. A technique of recognition for radio detected target based on wavelet and neural network theory [J]. *Acta Electronica Sinica*, 2003, 31(12): 1811 - 1814. (in Chinese)

[9] 陈朝阳,行小帅,李 ■. 共轭梯度算法在 Matlab7.0 中的实现[J].*现代电子技术*, 2009, (18): 125 - 127.
Chen Zhao-yang Y, Xing Xiao-shuai, Li Yue. Implement of conjugate gradient BP algorithm in Matlab 7.0 [J]. *Modern Electronics Technique*, 2009, (18): 125 - 127. (in Chinese)

[10] 丁世飞,贾伟宽,许新征等.基于 PLS 的 Elman 神经网络算法研究[J].*电子学报*, 2010, 38(2A): 71 - 75.
Ding Shi-fei, Jia Wei-kuan, Xu Xin-zheng, et al. Elman neural network algorithm based on PLS [J]. *Acta Electronica Sinica*, 2010, 38(2A): 71 - 75. (in Chinese)

[11] 郭业才,王丽华.模糊神经网络控制的混合小波神经网络盲均衡算法[J].*电子学报*, 2011, 39(4): 975 - 980.
Guo Ye-cai, Wang Li-hua. A hybrid wavelet neural network blind equalization algorithm based on fuzzy controlling [J]. *Acta Electronica Sinica*, 2011, 39(4): 975 - 980. (in Chinese)

[12] Roh S B, K Oh S, Pedrycz W. Design of fuzzy radial basis function-based polynomial neural networks [J]. *Fuzzy Sets and Systems*, 2011, 185(1): 15 - 37.

[13] 刘普寅.一种新的模糊神经网络及其逼近性能[J].*中国科学(E 辑)*, 2002, 32(1): 76 - 86.
Liu Pu-yin. A new fuzzy neural network and its approximation capability [J]. *Science in China (Ser E)*, 2002, 32(1): 76 -

86. (in Chinese)

- [14] 王贵君, 李晓萍. K-积分模意义下折线模糊神经网络的泛逼近性[J]. 中国科学. 信息科学, 2012, 42(3): 362 - 378.

Wang Gui-jun, Li Xiao-ping. Universal approximation of polygonal fuzzy neural networks in sense of K-integral norms

[J]. Science China Information Science, 2011, 54(11): 2307 - 2323.

- [15] 刘普寅. 模糊神经网络理论及其应用研究[D]. 北京: 北京师范大学, 2002.

Liu Pu-yin. Fuzzy Neural Network Theory and Applications [D]. Beijing: Beijing Normal University, 2002. (in Chinese)

作者简介



何 英 女, 1987 年 2 月出生, 湖北恩施人, 硕士研究生. 研究方向: 模糊神经网络与模糊系统.

E-mail: hyzdr2006@163.com



王贵君(通信作者) 男, 1962 年 7 月出生, 吉林通化人, 教授, 中国模糊系统与模糊数学学会理事, 天津市数学会理事. 研究方向: 模糊神经网络、模糊系统分析、模糊测度与模糊积分.

E-mail: tjwgj@126.com