

基于自适应小波神经网络的第二类 Fredholm 积分方程数值解法*

姜 微, 韩惠丽, 李风军

(宁夏大学 数学统计学院, 银川 750021)

摘要: 该文构造了一类三层前馈自适应小波神经网络,将小波分析中平移因子和伸缩因子的拟合设置为输入层到隐层的权值与阈值,采用小波基函数作为隐层激活函数,并根据梯度下降算法自适应地调整参数.应用自适应小波神经网络数值求解第二类 Fredholm 积分方程,通过数值算例验证了该方法的可行性和有效性.

关键词: 自适应小波神经网络; 第二类 Fredholm 积分方程; 数值解

中图分类号: TP183

文献标志码: A

DOI: 10.21656/1000-0887.400029

引 言

第二类 Fredholm 积分方程在工程等领域具有很好的应用背景^[1],由于其解析解难以获得,所以其高效的数值解法一直是研究的热点问题之一.一般地,第二类 Fredholm 积分方程形如

$$u(x) = f(x) + \lambda \int_{\Gamma} K(x, s) u(s) ds, \quad \Gamma = [a, b],$$

其中 $K(x, s)$, $f(x)$ 是已知函数, λ 为参数, $u(x)$ 是未知函数.第二类 Fredholm 积分方程的数值解法主要有小波基方法^[2]、配置法^[3]、同伦扰动法^[4]、迭代配置法^[5]、无网格方法^[6]等.这些方法在求解 Fredholm 积分方程的数值解时均将其与线性代数相结合,把复杂的积分问题转换为简单化的线性方程,并能应用于某些领域.但是这些方法存在计算量较大、运算速度较慢、精度不高等问题.近年来,国内外学者采用神经网络方法求解积分方程^[7-10],较好地解决了上述问题.但其构造的 BP (back propagation) 人工神经网络和 RBF (radial basis function) 神经网络出现了结构的盲目性、神经元规模有限的情况下训练精度瓶颈、训练过程不平稳及频繁出现系统误差跳变等问题^[11].

Zhang 等^[12]将小波分析、BP 神经网络和 RBF 神经网络三种方法的优点相结合,构建出具有更快收敛性、更高分辨精度、能克服局部极小问题的小波神经网络 (wavelet neural network,

* 收稿日期: 2019-01-10; 修订日期: 2019-10-30

基金项目: 国家自然科学基金(61662060;11762016);宁夏自然科学基金(2019AAC03037);宁夏高等学校自然科学研究项目(NGY2017018)

作者简介: 姜微(1994—),女,硕士生(E-mail: 281192409@qq.com);

韩惠丽(1972—),女,教授,博士(通讯作者.E-mail: nxhan@126.com);

李风军(1973—),男,教授,博士.

WNN).为了使小波基的选取更加灵活,根据数据本身的特征选取合适的小波基,称为“自适应方法”.本文将“自适应方法”融入小波神经网络,提出了一种基于自适应小波神经网络模型,其基本思想是将小波分析中平移因子和伸缩因子的拟合设置为输入层到隐层的权值与阈值,并将小波函数设置为网络隐层的激活函数,通过梯度下降算法自适应调整合适的小波基.然后,将该模型应用于第二类 Fredholm 积分方程的数值求解问题,数值算例也验证了此方法的可行性和有效性.

1 自适应小波神经网络模型的构建

本文以三层 BP 神经网络拓扑结构为基础,将小波基函数作为隐层激活函数构建小波神经网络模型.在此神经网络结构中每改变一次尺度因子,仅有一个平移因子与之对应,这与真正的小波变换存在一定差异.严格意义上说,该小波神经网络不是以小波的数学分析理论为基础的,其实质是利用某种形式的小波组合来逼近函数^[13].

在隐层中,每一个节点都有一个母小波函数与之对应.本文以 Mexican hat 小波函数作为母小波函数,实践证明该小波函数在不同的应用中效果较好^[14].每个节点的母小波形式如下:

$$\varphi(x) = (1 - x^2) \exp\left(-\frac{x^2}{2}\right). \quad (1)$$

1.1 输入-输出关系

设构造的小波神经网络输入节点数为 r , 隐层节点数为 n , 输出节点数为 m . 向量 $\mathbf{X} = (x_i) (i = 1, 2, \dots, r)$ 为输入向量, $a_{ij}, b_{ij} (i = 1, 2, \dots, r; j = 1, 2, \dots, n)$ 分别表示输入分量 x_i 到隐层第 j 个神经元对应的伸缩因子和 平移因子, $w_{ij}, b_{ij}^w (i = 1, 2, \dots, r; j = 1, 2, \dots, n)$ 分别表示输入分量 x_i 到隐层第 j 神经元对应的权值和 阈值.其拓扑结构如图 1 所示.

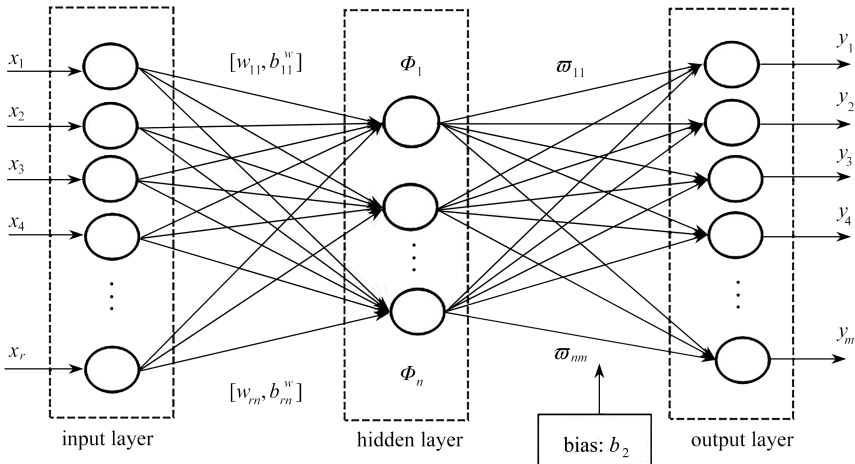


图 1 小波神经网络的拓扑结构

Fig. 1 The algorithm structure of the wavelet neural network

样本的自适应设计将根据数据本身的结构特征选取及调整合适的小波基,灵活改变 a_{ij}, b_{ij} 数值的大小,其中自适应机制的建立过程如下^[15].

计算自适应向量参数:

$$\begin{cases} P = \max(\mathbf{X}), \\ Q = \min(\mathbf{X}). \end{cases}$$

则平移因子自适应初始值为 $a_{ij} = P - Q$, 伸缩因子自适应初始值变换为 $b_{ij} = Q (i = 1, 2, \dots, r; j = 1, 2, \dots, n)$.

设定

$$w_{ij} = \frac{1}{a_{ij}}, \quad b_{ij}^w = -\frac{b_{ij}}{a_{ij}} \quad (i = 1, 2, \dots, r; j = 1, 2, \dots, n) . \quad (2)$$

输入层

$$o_{ij} = w_{ij}x_i + b_{ij}^w = \frac{1}{a_{ij}}x_i + \left(-\frac{b_{ij}}{a_{ij}}\right) = \frac{x_i - b_{ij}}{a_{ij}} \quad (i = 1, 2, \dots, r; j = 1, 2, \dots, n); \quad (3)$$

隐藏层

$$\Phi_j(\mathbf{X}) = \prod_{i=1}^r \varphi(o_{ij}) \quad (j = 1, 2, \dots, n); \quad (4)$$

输出层

$$y_k(\mathbf{X}) = \sum_{j=1}^n \varpi_{jk} \Phi_j(\mathbf{X}) + b_2 \quad (k = 1, 2, \dots, m), \quad (5)$$

其中, $\varphi(x)$ 是 Mexican hat 小波函数, y_k 为输出值, ϖ_{jk}, b_2 分别是隐层到输出层的权值与阈值.

1.2 参数调整

为了求解第二类 Fredholm 积分方程, 本文使用复化的 Simpson 公式, 将区间 $[a, b]$ 分成 r 个小区间 $\Delta_i = [x_{i-1}, x_i]$ ($i = 1, 2, \dots, r$), 并且在每个小区间内取得一个节点 $t_i \in \Delta_i$ ($i = 1, 2, \dots, r$), 代入公式:

$$\int_b^a g(s) ds \approx \frac{h}{6} \left[g(a) + 4 \sum_{i=0}^{r-1} g(t_{i+1/2}) + 2 \sum_{i=0}^{r-1} g(t_i) + g(b) \right], \quad (6)$$

从而求得 Fredholm 积分方程中积分项 $\int_r K(x, s)u(s) ds$ 的值, 其中 $g(s) = K(x, s)u(s)$. 调整小波神经网络的相关参数, 将问题转化为误差平方和最小化问题, 再根据梯度下降法自适应调节相关参数, 具体过程如下:

$$\begin{aligned} E_i &= \frac{1}{2} \sum_{k=1}^m \left[f(x_i) - \left(b_2 + \sum_{j=1}^n \left(\varpi_{jk} \prod_{i=1}^r \left(1 - \left(\frac{x_i - b_{ij}}{a_{ij}} \right)^2 \right) e^{-((x_i - b_{ij})/a_{ij})^2/2} \right) - \right. \right. \\ &\quad \left. \left. \lambda \int_{\tau_1} K(x_i, s) y_k(s) ds \right) \right]^2 = \\ &= \frac{1}{2} \sum_{k=1}^m \left[f(x_i) - \left(y_k - \lambda \int_{\tau_1} K(x_i, s) y_k(s) ds \right) \right]^2 = \frac{1}{2} \sum_{k=1}^m [f(x_i) - f_1(x_i)]^2 = \\ &= \frac{1}{2} \sum_{k=1}^m [e(x_i)]^2 \quad (i = 1, 2, \dots, r), \end{aligned} \quad (7)$$

其中 $f_1(x_i) = y_k + \lambda \int_{\tau_1} K(x_i, s) y_k(s) ds$. 分别对 $a_{ij}, b_{ij}, \varpi_{jk}, b_2$ 求偏导, 得

$$\frac{\partial E_i}{\partial a_{ij}} = - \sum_{k=1}^m (f(x_i) - f_1(x_i)) \varpi_{jk} \Phi_j'(o_{ij}) \left(\frac{b_{ij} - x_i}{a_{ij}^2} \right), \quad (8)$$

$$\frac{\partial E_i}{\partial b_{ij}} = - \sum_{k=1}^m (f(x_i) - f_1(x_i)) \varpi_{jk} \Phi_j'(o_{ij}) \left(-\frac{1}{a_{ij}} \right), \quad (9)$$

$$\frac{\partial E_i}{\partial \varpi_{jk}} = - \sum_{k=1}^m (f(x_i) - f_1(x_i)) \Phi_j, \quad (10)$$

$$\frac{\partial E_i}{\partial b_2} = - \sum_{k=1}^m (f(x_i) - f_1(x_i)) . \quad (11)$$

采用自适应梯度下降法进行调整,得到

$$a_{ij}(t+1) = a_{ij}(t) - \eta \frac{\partial E_i}{\partial a_{ij}} + \alpha [a_{ij}(t) - a_{ij}(t-1)] , \quad (12)$$

$$b_{ij}(t+1) = b_{ij}(t) - \eta \frac{\partial E_i}{\partial b_{ij}} + \alpha [b_{ij}(t) - b_{ij}(t-1)] , \quad (13)$$

$$\omega_{ij}(t+1) = \frac{1}{a_{ij}(t+1)} , \quad (14)$$

$$b_{ij}^w(t+1) = - \frac{b_{ij}(t+1)}{a_{ij}(t+1)} , \quad (15)$$

$$\omega_{jk}(t+1) = \omega_{jk}(t) - \eta \frac{\partial E_i}{\partial \omega_{jk}} + \alpha [\omega_{jk}(t) - \omega_{jk}(t-1)] , \quad (16)$$

$$b_2(t+1) = b_2(t) - \eta \frac{\partial E_i}{\partial b_2} + \alpha [\Delta b_2(t) - \Delta b_2(t-1)] , \quad (17)$$

其中 η 为学习率, α 为动量系数.

1.3 算法说明

隐层的激励函数 $\Phi(\mathbf{X})$, 伸缩因子 \mathbf{a} , 平移因子 \mathbf{b} , 输入值 \mathbf{X} , 隐层到输出层的权值 $\boldsymbol{\omega}$ 分别定义如下:

$$\Phi(\mathbf{X}) = [\Phi_1(\mathbf{X}), \Phi_2(\mathbf{X}), \dots, \Phi_n(\mathbf{X})]^T ,$$

$$\mathbf{a} = [a_{ij}] , \mathbf{b} = [b_{ij}] , \mathbf{X} = [x_{ij}] , \boldsymbol{\omega} = [\omega_{jk}] ,$$

$$i = 1, 2, \dots, r; j = 1, 2, \dots, n; k = 1, 2, \dots, m .$$

前面推导的公式是自适应小波神经网络算法的基础,下面给出自适应小波神经网络算法过程.

步骤 1: 初始化

对参数 \mathbf{a}, \mathbf{b} 自适应设定初始值, 参数 $\boldsymbol{\omega}, b_2$ 赋随机数, 误差 E_i 设置为 0, 学习率 η 和动量系数 α 设为 0~1 内的小数, 网络训练后达到的精度 E_{\min} 设为一个正的小数.

步骤 2: 隐层输出

用当前样本对输入向量 \mathbf{X} 赋值, 并代入式(5)计算神经网络输出值.

步骤 3: 计算网络输出误差

采用误差平方和 $E_i = (1/2) \sum_{k=1}^m [f(x_i) - f_1(x_i)]^2$ 作为网络的总误差, 最后结果可表示为式(7).

步骤 4: 调整权值

结合式(8)~(11)计算 $\mathbf{a}, \mathbf{b}, \boldsymbol{\omega}, b_2$ 的偏导, 采用梯度下降法自适应进行调整, 如式(12)~(17)所示.

步骤 5: 检查网络误差是否达到精度要求

当 E_{RME} 作为网络总误差时, 若满足 $E_{\text{RME}} < E_{\min}$, 训练结束. 否则返回步骤 2.

具体的算法流程图如图 2 所示.

1.4 收敛性分析

定理 1 设 N 为隐层神经元个数, 若学习率 $0 < \eta < 2/(N+1)$ 时, 则自适应小波神经网络

络模型的学习算法收敛.

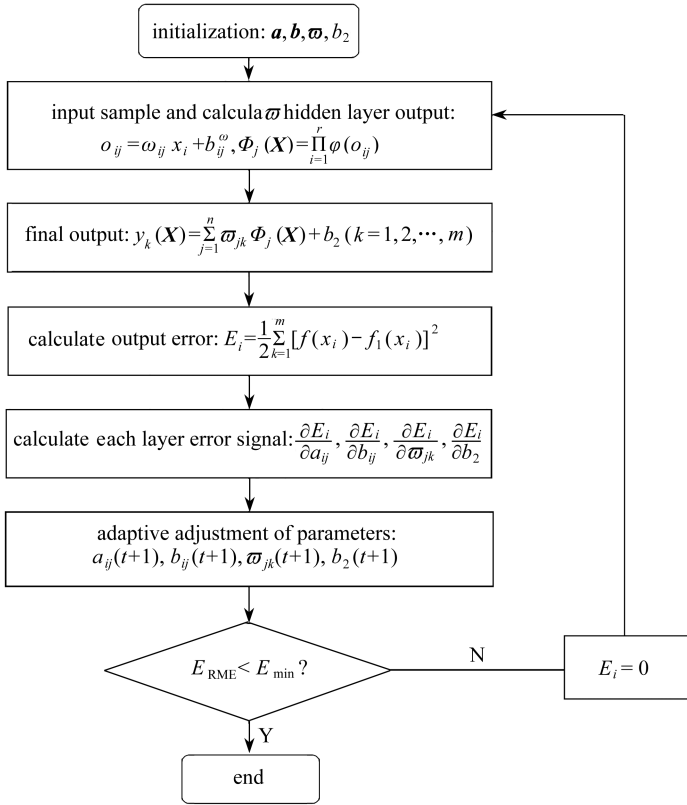


图 2 自适应小波神经网络算法流程图

Fig. 2 The adaptive wavelet neural network algorithm flow chart

证明 取 Lyapunov 函数 $V(t) = e^2(t)/2$, 则

$$\Delta V(t) = \frac{1}{2} e^2(t+1) - \frac{1}{2} e^2(t),$$

其中 $e(t+1) = e(t) + \Delta e(t)$, 而

$$\begin{aligned} \Delta e(t) &= \left[\frac{\partial e(t)}{\partial \mathbf{a}} \right]^T \Delta \mathbf{a} + \left[\frac{\partial e(t)}{\partial \mathbf{b}} \right]^T \Delta \mathbf{b} + \left[\frac{\partial e(t)}{\partial \boldsymbol{\omega}} \right]^T \Delta \boldsymbol{\omega} + \frac{\partial e(t)}{\partial b_2} \Delta b_2 = \\ &= \eta e(t) \left[\frac{\partial e(t)}{\partial \mathbf{a}} \right]^T \frac{\partial e(t)}{\partial \mathbf{a}} - \eta e(t) \left[\frac{\partial e(t)}{\partial \mathbf{b}} \right]^T \frac{\partial e(t)}{\partial \mathbf{b}} - \\ &= \eta e(t) \left[\frac{\partial e(t)}{\partial \boldsymbol{\omega}} \right]^T \frac{\partial e(t)}{\partial \boldsymbol{\omega}} - \eta e(t) \frac{\partial e(t)}{\partial b_2} \frac{\partial e(t)}{\partial b_2} = \\ &= \eta e(t) \left(\left\| \frac{\partial e(t)}{\partial \mathbf{a}} \right\|_2^2 + \left\| \frac{\partial e(t)}{\partial \mathbf{b}} \right\|_2^2 + \left\| \frac{\partial e(t)}{\partial \boldsymbol{\omega}} \right\|_2^2 + \left\| \frac{\partial e(t)}{\partial b_2} \right\|_2^2 \right), \end{aligned}$$

其中 $\| \cdot \|_2^2 = \sum | \cdot |^2$ 称为 Euclid 范数的平方, 故

$$\begin{aligned} \Delta V(t) &= \frac{1}{2} e^2(t+1) - \frac{1}{2} e^2(t) = \\ &= \frac{1}{2} [e(t) + \Delta e(t)]^2 - \frac{1}{2} e^2(t) = \Delta e(t) \left[e(t) + \frac{1}{2} \Delta e(t) \right] = \end{aligned}$$

$$\begin{aligned}
& -\eta e(t) \left[\left\| \frac{\partial e(t)}{\partial \mathbf{a}} \right\|_2^2 + \left\| \frac{\partial e(t)}{\partial \mathbf{b}} \right\|_2^2 + \left\| \frac{\partial e(t)}{\partial \boldsymbol{\omega}} \right\|_2^2 + \left\| \frac{\partial e(t)}{\partial b_2} \right\|_2^2 \right] \times \\
& \left[e(t) + \frac{1}{2} \eta e(t) \left(\left\| \frac{\partial e(t)}{\partial \mathbf{a}} \right\|_2^2 + \left\| \frac{\partial e(t)}{\partial \mathbf{b}} \right\|_2^2 + \left\| \frac{\partial e(t)}{\partial \boldsymbol{\omega}} \right\|_2^2 + \left\| \frac{\partial e(t)}{\partial b_2} \right\|_2^2 \right) \right] = \\
& e^2(t) \left[\left\| \frac{\partial e(t)}{\partial \mathbf{a}} \right\|_2^2 + \left\| \frac{\partial e(t)}{\partial \mathbf{b}} \right\|_2^2 + \left\| \frac{\partial e(t)}{\partial \boldsymbol{\omega}} \right\|_2^2 + \left\| \frac{\partial e(t)}{\partial b_2} \right\|_2^2 \right] \times \\
& \left[-\eta + \frac{1}{2} \eta^2 \left(\left\| \frac{\partial e(t)}{\partial \mathbf{a}} \right\|_2^2 + \left\| \frac{\partial e(t)}{\partial \mathbf{b}} \right\|_2^2 + \left\| \frac{\partial e(t)}{\partial \boldsymbol{\omega}} \right\|_2^2 + \left\| \frac{\partial e(t)}{\partial b_2} \right\|_2^2 \right) \right].
\end{aligned}$$

由上式可知,要使神经网络收敛,则

$$-\eta + \frac{1}{2} \eta^2 \left[\left\| \frac{\partial e(t)}{\partial \mathbf{a}} \right\|_2^2 + \left\| \frac{\partial e(t)}{\partial \mathbf{b}} \right\|_2^2 + \left\| \frac{\partial e(t)}{\partial \boldsymbol{\omega}} \right\|_2^2 + \left\| \frac{\partial e(t)}{\partial b_2} \right\|_2^2 \right] < 0,$$

即

$$0 < \eta < \frac{2}{\left\| \frac{\partial e(t)}{\partial \mathbf{a}} \right\|_2^2 + \left\| \frac{\partial e(t)}{\partial \mathbf{b}} \right\|_2^2 + \left\| \frac{\partial e(t)}{\partial \boldsymbol{\omega}} \right\|_2^2 + \left\| \frac{\partial e(t)}{\partial b_2} \right\|_2^2}.$$

由式(8)~(11)可得

$$\frac{\partial e(t)}{\partial \mathbf{a}} = \boldsymbol{\omega} \boldsymbol{\Phi}' \frac{\mathbf{b} - \mathbf{X}}{a^2}, \quad \frac{\partial e(t)}{\partial \mathbf{b}} = \boldsymbol{\omega} \boldsymbol{\Phi}' \left(-\frac{1}{a} \right), \quad \frac{\partial e(t)}{\partial \boldsymbol{\omega}} = \boldsymbol{\Phi}, \quad \frac{\partial e(t)}{\partial b_2} = 1.$$

则

$$\begin{aligned}
\left\| \frac{\partial e(t)}{\partial \mathbf{a}} \right\|_2^2 &= \left\| \boldsymbol{\omega} \boldsymbol{\Phi}' \frac{\mathbf{b} - \mathbf{X}}{a^2} \right\|_2^2 = \sum_{j=1}^n \left(\boldsymbol{\omega} \boldsymbol{\Phi}'_j \frac{\mathbf{b} - \mathbf{X}}{a^2} \right)^2 \leq 2.25N \sum_{j=1}^n \left(\boldsymbol{\omega} \frac{\mathbf{b} - \mathbf{X}}{a^2} \right)^2, \\
\left\| \frac{\partial e(t)}{\partial \mathbf{b}} \right\|_2^2 &= \left\| \boldsymbol{\omega} \boldsymbol{\Phi}' \left(-\frac{1}{a} \right) \right\|_2^2 = \sum_{j=1}^n \left(\boldsymbol{\omega} \boldsymbol{\Phi}'_j \left(-\frac{1}{a} \right) \right)^2 \leq 2.25N \sum_{j=1}^n \left(\boldsymbol{\omega} \left(-\frac{1}{a} \right) \right)^2, \\
\left\| \frac{\partial e(t)}{\partial \boldsymbol{\omega}} \right\|_2^2 &= \|\boldsymbol{\Phi}\|_2^2 = \sum_{j=1}^n (\boldsymbol{\omega})^2 \leq N, \\
\left\| \frac{\partial e(t)}{\partial b_2} \right\|_2^2 &= 1.
\end{aligned}$$

取

$$\left\| \frac{\partial e(t)}{\partial \boldsymbol{\omega}} \right\|_2^2 = \|\boldsymbol{\Phi}\|_2^2 = \sum_{j=1}^n \boldsymbol{\omega}^2 = N,$$

有

$$\begin{aligned}
N + 1 &\leq \left\| \frac{\partial e(t)}{\partial \mathbf{a}} \right\|_2^2 + \left\| \frac{\partial e(t)}{\partial \mathbf{b}} \right\|_2^2 + \left\| \frac{\partial e(t)}{\partial \boldsymbol{\omega}} \right\|_2^2 + \left\| \frac{\partial e(t)}{\partial b_2} \right\|_2^2 \leq \\
& 2.25N \sum_{j=1}^n \left(\boldsymbol{\omega} \frac{\mathbf{b} - \mathbf{X}}{a^2} \right)^2 + 2.25N \sum_{j=1}^n \left(\boldsymbol{\omega} \left(-\frac{1}{a} \right) \right)^2 + N + 1,
\end{aligned}$$

即

$$0 < \eta < \frac{2}{\left\| \frac{\partial e(t)}{\partial \mathbf{a}} \right\|_2^2 + \left\| \frac{\partial e(t)}{\partial \mathbf{b}} \right\|_2^2 + \left\| \frac{\partial e(t)}{\partial \boldsymbol{\omega}} \right\|_2^2 + \left\| \frac{\partial e(t)}{\partial b_2} \right\|_2^2} < \frac{2}{N + 1}.$$

故当学习率 η 满足 $0 < \eta < 2/(N + 1)$ 时,有 $\Delta V(t) < 0$,从而本文讨论的自适应小波神经网络

络算法收敛.

2 数值算例

为验证本文构造的自适应小波神经网络模型及算法的可靠性、可行性及有效性,并能与已有算法进行比较,在此引用文献[16]中的例子加以讨论.

例 1 考虑 Fredholm 积分方程:

$$u(x) = f(x) + \lambda \int_{\Gamma} xsu(s) ds, \quad \Gamma = [0, 1],$$

其中 $f(x) = 5x/6, \lambda = 1/2$, 精确解为 $u(x) = x$.

表 1 三种数值解与精确解(例 1)

Table 1 Three numerical solutions and exact solutions(case 1)

x	numerical solution of the adaptive wavelet neural network	numerical solution of the BP wavelet neural network	numerical solution of the gradual approach method	exact solution
0.0	0.000 000 000 259 275	0.000 000 000 105 942	0.000 000 000 000 000	0.000 000 000 000 000
0.1	0.100 000 971 289 362	0.100 011 000 010 000	0.100 372 942 386 831	0.100 000 000 000 000
0.2	0.200 001 942 285 015	0.200 152 000 020 001	0.200 745 884 773 663	0.200 000 000 000 000
0.3	0.300 002 914 424 274	0.300 003 000 030 001	0.301 118 827 160 494	0.300 000 000 000 000
0.4	0.400 003 885 233 542	0.400 774 000 040 001	0.401 491 769 547 325	0.400 000 000 000 000
0.5	0.500 004 855 703 966	0.500 845 000 050 002	0.501 864 711 934 156	0.500 000 000 000 000
0.6	0.600 005 827 244 334	0.600 026 000 060 002	0.602 237 654 320 988	0.600 000 000 000 000
0.7	0.700 006 797 900 472	0.700 997 000 070 002	0.702 610 596 707 819	0.700 000 000 000 000
0.8	0.800 007 768 072 160	0.800 218 000 080 002	0.802 983 539 094 650	0.800 000 000 000 000
0.9	0.900 008 739 216 096	0.900 009 000 090 003	0.903 356 481 481 482	0.900 000 000 000 000
1.0	1.000 009 710 280 087	1.000 110 000 100 004	1.003 729 423 868 313	1.000 000 000 000 000

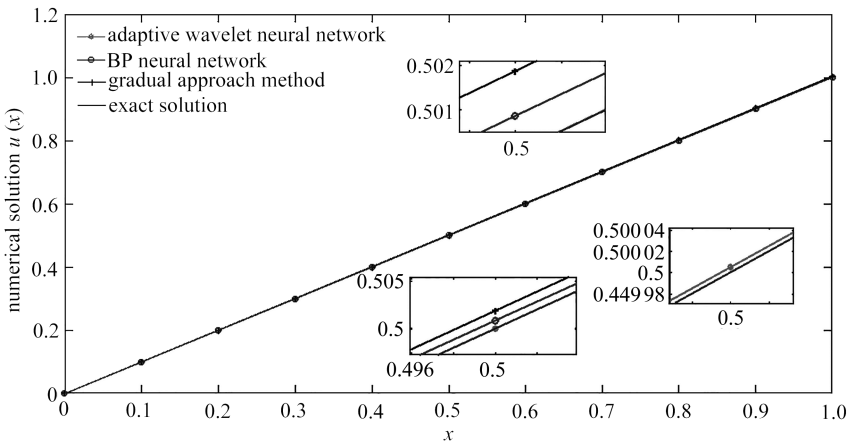


图 3 数值解及精确解的比较(例 1)

Fig. 3 Comparison of numerical results after training(case 1)

初始化参数 ω, b_2 及自适应向量参数 a, b , 利用 MATLAB 7 优化工具箱及式(2)~(5), 求得小波神经网络的输出值, 使用复化的 Simpson 公式将积分项转化为多项式并代入到式(7)中, 根据梯度下降算法自适应调节相关参数, 将误差函数最小化, 最终得到高精度的数值解. 其

中结合公式 $n = \sqrt{mr} + \alpha$ (r 为输入节点数, n 为隐层节点数, m 为输出节点数, α 为 0~1 之间的常数) 选取合适的隐层节点数. 本文神经网络隐层是由 15 个节点组成. 如果想要获得更精确的结果 (特别是在非线性的情况下), 可以使用更多的神经元或训练节点. 表 1 展现了自适应小波神经网络数值解、测试结果 BP 神经网络数值解、逐次逼近法数值解及精确解的具体值, 结果对比见图 3.

为了说明上述方法具有普遍性, 本文引入例 2, 将其推广到更一般的情况.

例 2 考虑第二类非线性 Fredholm 积分方程:

$$u(x) = f(x) + \lambda \int_{\Gamma} (1 - x \cos(xs)) u(s) ds, \quad \Gamma = [0, 1],$$

其中 $f(x) = \sin x, \lambda = 1$, 精确解为 $u(x) = 1$.

采用同例 1 相同的处理方法, 求得数值解和精确解见表 2, 结果对比见图 4.

表 2 三种数值解与精确解(例 2)

Table 2 Three numerical solutions and exact solutions(case 2)

x	numerical solution of the adaptive wavelet neural network	numerical solution of the BP wavelet neural network	numerical solution of the degenerated kernel approximate substitution method	exact solution
0.0	1.000 004 131 080 228	1.000 354 126 969 967	1.002 850 000 000 000	1.000 000 000 000 000
0.1	1.000 003 727 142 497	1.000 113 623 442 875	1.004 072 616 646 828	1.000 000 000 000 000
0.2	1.000 003 377 882 555	1.000 003 374 591 250	1.007 646 130 795 061	1.000 000 000 000 000
0.3	1.000 003 136 265 966	1.006 303 133 378 357	1.012 583 006 661 340	1.000 000 000 000 000
0.4	1.000 003 051 869 213	1.000 773 049 378 408	1.017 915 542 308 651	1.000 000 000 000 000
0.5	1.000 003 169 272 955	1.000 343 167 169 846	1.022 705 538 604 203	1.000 000 000 000 000
0.6	1.000 003 526 558 976	1.000 905 524 832 310	1.026 053 673 395 035	1.000 000 000 000 000
0.7	1.000 004 153 942 611	1.000 004 152 579 071	1.027 108 487 237 691	1.000 000 000 000 000
0.8	1.000 005 072 570 146	1.000 785 071 554 453	1.025 074 890 899 523	1.000 000 000 000 000
0.9	1.000 006 293 507 818	1.001 066 292 822 844	1.019 222 109 627 483	1.000 000 000 000 000
1.0	1.000 007 816 945 793	1.000 007 816 572 693	1.008 890 984 807 897	1.000 000 000 000 000

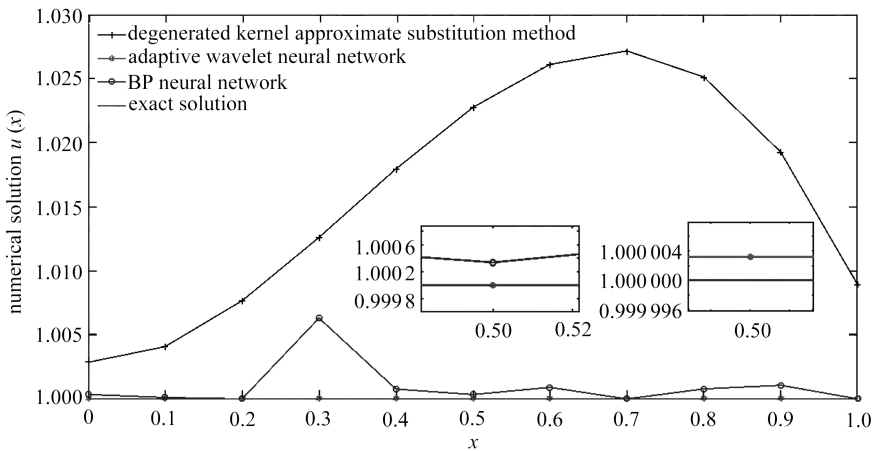


图 4 数值解及精确解的比较(例 2)

Fig. 4 Comparison of numerical results after training(case 2)

由表 1、2 和图 3、4 可以看出, 对于一般的第二类 Fredholm 积分方程, 自适应小波神经网络

络算法的数值精度高、适应性强、误差稳定。

3 总 结

本文构造了一种自适应小波神经网络模型及相关学习算法,并将其用于求解第二类 Fredholm 积分方程数值解,该方法将自适应学习方法和人工神经网络相结合,把小波基拟合成人工神经网络模型中权值和阈值的候选者,再利用自适应学习方法选取最优的权值与阈值,以充分发挥自适应学习方法、小波分析与人工神经网络的优点,克服彼此的缺点,实例分析验证了本文所构造模型及学习算法的可行性、可靠性与优越性。

参考文献 (References):

- [1] 郭钊, 郭子涛, 易玲艳. 多裂纹问题计算分析的本征 COD 边界积分方程方法[J]. 应用数学和力学, 2019, **40**(2): 200-209. (GUO Zhao, GUO Zitao, YI Lingyan. Analysis of multicroack problems with eigen COD boundary integral equations[J]. *Applied Mathematics and Mechanics*, 2019, **40**(2): 200-209. (in Chinese))
- [2] KIRAN M S. Particle swarm optimization with a new update mechanism[J]. *Applied Soft Computing*, 2017, **60**: 670-678.
- [3] 仪明旭, 陈一鸣, 魏金侠, 等. 应用 Haar 小波求解非线性分数阶 Fredholm 积分微分方程[J]. 河北师范大学学报(自然科学版), 2012, **36**(5): 452-455. (YI Mingxu, CHEN Yiming, WEI Jinxia, et al. Haar wavelet method for solving nonlinear Fredholm integro-differential equations of fractional order[J]. *Journal of Hebei Normal University(Natural Science)*, 2012, **36**(5): 452-455. (in Chinese))
- [4] BIAZAR J, GHAZVINI H. Convergence of the homotopy perturbation method for partial differential equations[J]. *Nonlinear Analysis: Real World Applications*, 2009, **10**(5): 2633-2640.
- [5] DONG Chunhuan, CHEN Zhong, JIANG Wei. A modified homotopy perturbation method for solving the nonlinear mixed Volterra-Fredholm inequal equation[J]. *Journal of Computational and Applied Mathematics*, 2013, **239**: 359-366.
- [6] BRUNNER H. Iterated collocation methods and their discretizations for Volterra integral equations[J]. *SIAM Journal on Numerical Analysis*, 1984, **21**(6): 1132-1145.
- [7] 张志刚, 赵新泉. 利用 BP 神经网络计算 Fredholm- II 型积分方程的近似解[J]. 中南民族大学学报(自然科学版), 2002, **21**(4): 79-81. (ZHANG Zhigang, ZHAO Xinquan. Solving Fredholm- II integral equations by using BP neural network[J]. *Journal of South-Central University for Nationalities(Natural Science Edition)*, 2002, **21**(4): 79-81. (in Chinese))
- [8] 王小华, 何怡刚. 三角基函数神经网络算法在数值积分中的应用研究[J]. 电子与信息学报, 2004, **26**(3): 394-399. (WANG Xiaohua, HE Yigang. Numerical integration study based on triangle basis neural network algorithm[J]. *Journal of Electronics & Information Technology*, 2004, **26**(3): 394-399. (in Chinese))
- [9] JAFARIAN A, MEASOOMY NIA S. Utilizing feed-back neural network approach for solving linear Fredholm integral equations system [J]. *Applied Mathematical Modelling*, 2013, **37**(7): 5027-5038.
- [10] 闫丽娜, 王珂. 基于神经网络的数值积分改进算法[J]. 应用数学与计算数学学报, 2017, **30**(4): 520-525. (YAN Lina, WANG Ke. Improved numerical integration algorithm based on neural network[J]. *Communication on Applied Mathematics and Computation*, 2017, **30**(4): 520-525. (in Chinese))
- [11] 刘经纬, 赵辉, 周瑞, 等. 高精度自适应小波神经网络人工智能方法探索[J]. 计算机科学与技

- 术边界学报, 2016, **10**(8): 1122-1132. (LIU Jingwei, ZHAO Hui, ZHOU Rui, et al. Exploration of high-precision adaptive wavelet neural network artificial intelligence method[J]. *Journal of Frontiers of Computer Science and Technology*, 2016, **10**(8): 1122-1132. (in Chinese))
- [12] ZHANG Q, BMVENLSTE A. Wavelet networks[J]. *IEEE Transactions on Neural Networks*, 1992, **3**(6): 889-898.
- [13] 黄同成. 基于小波神经网络理论的 VOCR 与 HOCR 技术研究[D]. 博士学位论文. 上海: 上海大学, 2008. (HUANG Tongcheng. Research on VOCR and HOCR technology based on wavelet neural network theory[D]. PhD Thesis. Shanghai: Shanghai University, 2008. (in Chinese))
- [14] ALEXANDRIDIS A K, ZAPRANIS A D. Wavelet neural networks; a practical guide[J]. *Neural Networks*, 2013, **42**:1-27.
- [15] 刘经纬, 王普. 基于自适应小波神经网络的复杂系统模式识别方法[J]. 北京工业大学学报, 2014, **40**(6): 843-850. (LIU Jingwei, WANG Pu. Complex system pattern recognition method based on adaptive wavelet neural network[J]. *Journal of Beijing University of Technology*, 2014, **40**(6): 843-850. (in Chinese))
- [16] 李星. 积分方程[M]. 北京: 科学出版社, 2008. (LI Xin. *Integral Equation*[M]. Beijing: Science Press, 2008. (in Chinese))

Numerical Solution to the Second Kind of Fredholm Integral Equation Based on the Adaptive Wavelet Neural Network

JIANG Wei, HAN Huili, LI Fengjun

(School of Mathematics and Statistics, Ningxia University,
Yinchuan 750021, P.R.China)

Abstract: A 3-layer feedforward adaptive wavelet neural network model was constructed. The fitting of the translation factor and the scaling factor were combined in the wavelet analysis. The result of combination was set as the weight and bias of the hidden layer. The wavelet basis function was used as the hidden layer activation function and the parameters could be adaptively adjusted according to the gradient descent algorithm. Numerical solution to the second kind of Fredholm integral equation was solved with the adaptive wavelet neural network, and the feasibility and validity of the method were verified through numerical examples.

Key words: adaptive wavelet neural network; second kind of Fredholm integral equation; numerical solution

Foundation item: The National Natural Science Foundation of China(61662060;11762016)

引用本文/Cite this paper:

姜微, 韩惠丽, 李风军. 基于自适应小波神经网络的第二类 Fredholm 积分方程数值解法[J]. 应用数学和力学, 2019, **40**(12): 1399-1408.

JIANG Wei, HAN Huili, LI Fengjun. Numerical solution to the second kind of Fredholm integral equation based on the adaptive wavelet neural network[J]. *Applied Mathematics and Mechanics*, 2019, **40**(12): 1399-1408.