

一种带修剪的增量极速学习模糊神经网络

胡蓉^{1,2} 徐蔚鸿^{1,3}

(南京理工大学计算机科学与技术学院 南京 210000)¹ (长沙航空职业技术学院 长沙 410014)²
(长沙理工大学计算机与通信工程学院 长沙 410000)³

摘要 由 Huang 提出的 extreme learning machine(ELM)批量学习算法在获得与其他算法相当的性能的同时显示了极快的学习速度。为了实现在线增量学习,扩展了 ELM 方法,提出了一种带修剪的极速学习模糊神经网络。首先随机产生模糊神经网络前件参数和规则数量,然后使用 SVD 将规则按照重要性能排序,再使用留一法 leave-one-out(LOO)选出最佳的模糊规则数,最后分析计算模糊规则的后件参数。在学习过程中无须保存过去的的数据,真正实现了增量学习。当新的数据到来时,无须重新训练网络。通过仿真实验对该方法与其他算法进行了验证和比较,结果表明,在获得与其他算法类似的性能的情况下,该算法能够获得更加简洁的结构。

关键词 极速学习机(ELM),增量学习,模糊神经网络,径向基函数

中图分类号 TP306.1 文献标识码 A

Pruned Incremental Extreme Learning Machine Fuzzy Neural Network

HU Rong^{1,2} XU Wei-hong^{1,3}

(School of Computer Science and Technology, Nanjing University of Science and Technology, Nanjing 210000, China)¹

(Changsha Aeronautical Vocational and Technical College, Changsha 410014, China)²

(College of Computer and Communications Engineering, Changsha University of Science & Technology, Changsha 410000, China)³

Abstract ELM for batch learning developed by Huang et al has been shown to be extremely fast with generalization performance better than other batch training methods. In order to learn online, a pruned incremental extreme learning algorithm was developed for fuzzy neural network. Based on the previously proposed ELM method, we proposed a pruned incremental extreme learning algorithm. First a set of simple antecedents and random values for the parameters of input membership functions were randomly generated. Then SVD was used to rank the fuzzy basis functions. Then the best number of fuzzy rules was selected by performing a fast computation of the leave-one-out validation error. Finally, the consequents parameters were determined analytically. During the procedure of learning, it is not need to memory the last date. It is a real incremental learning method. When a new data arrives, it is not need to retrain the network. A comparison was performed against well known neuro-fuzzy methods. It is shown that the method proposed is robust and competitive in terms of accuracy and speed.

Keywords Extreme learning machine(ELM), Incremental learning algorithm, Fuzzy neural network(FNN), Radial basis function

1 引言

模糊神经网络(FNN)由于结合了模糊规则系统的语义透明性和神经网络的学习能力而被广泛研究^[1-3,15,16]。模糊神经网络的参数学习算法被分为两类:批量学习和增量学习。在批量学习中,假设训练数据在训练之前全部到达,训练过程通常是周期训练数据直至误差达到设定的目标。增量学习是数据一个接一个地到达,每个数据被训练后就丢弃,因此整个过程不存在周期学习。

近来,一些真正的增量连续学习的模糊算法被提出来。它们包括动态进化神经模糊接口系统(DENFIS)^[4]、进化 T-S

模型^[5]、简化 eTS 算法^[6]、连续适应模糊接口系统(SAFIS)^[7]、进化模糊优化修剪极值学习机器^[8]。

ELM^[9]是训练单层前馈人工神经网络(SLFNS)的一个简单而有效的学习算法,理论和实验都表明 ELM 准确而快速。事实上,ELM 是一个非常快速的方法,而且能够达到其他计算复杂的算法同样的泛化性能。RONG^[7]证明了 SLFNS 能够被认为与 FIS 等价。基于 ELM 提出了一些优化修剪极速学习机模型^[10-12]。

由 Mich^[10]提出的 OP-ELM 可改善 ELM 算法的鲁棒性。它通过 3 个阶段来实现:首先构建一个 ELM,然后对神经元进行适当的排序,最后采用合适的方法决定多少神经元被修

到稿日期:2012-07-11 返修日期:2012-11-25 本文受国家自然科学基金(61163040),湖南省教育厅科研项目(11C0009)资助。

胡蓉(1974-),女,博士生,副教授,主要研究方向为模式识别、机器学习、人工智能,E-mail:410907140@qq.com;徐蔚鸿(1963-),男,教授,博士生导师,主要研究方向为人工智能、模式识别、图像处理、软件工程。

剪掉。这些阶段执行都是非常快速而且准确的。OP-ELM 相比其他方法更加快速准确而且结构简单。

本文在 ELM 的基础上提出一种根据模糊规则的影响来修剪规则的模糊神经网络 (A pruned extreme incremental leaning fuzzy neural network)。PEIL-FNN 初始时,随机产生 L 个模糊规则,模糊规则的前件参数也是随机分配的,然后,对模糊规则使用 SVD 按照性能进行排序,并采用 LOO 选择出合适的规则,最后分析得出后件参数值。在学习过程中无须保存所有过去的的数据。

可以看出 PEIL-FNN 是真正意义上的增量连续学习算法。为了评估该算法的性能,我们对混沌时序预测 (Mackey-Glass) 等一些典型问题进行了仿真。实验表明该算法快速而准确,具有很好的泛化性能。

2 PIEL-FNN 模型

根据 TakagSugeno-Kang(TSK) FISs 提出 PIEL-FNN 的模型,TSK 模糊模型由以下模糊规则组成:

Rule j : if (x_1 is A_{1j}) AND (x_2 is A_{2j}) AND \dots AND (x_r is A_{rj}), then (y_1 is β_{j1}) \dots (y_m is β_{jm}) (1)

式中, A_{ij} ($i=1,2,\dots,r,j=1,2,\dots,L$) 是第 j 个规则的 i 个输入变量 x_i 的模糊规则集, r 是数据变量的维数 ($x=[x_1,\dots,x_r]$), m 是输出向量的维数 ($y=[y_1,\dots,y_m]^T$), L 是模糊规则数。 β_{ik} ($k=1,2,\dots,m,i=1,2,\dots,L$) 是输入变量的线性加权值 i. e., $\beta_{ik} = q_{ik,0} + q_{ik,1}x_1 + \dots + q_{ik,n}x_n$ 。

在先前描述的 TSK 模糊模型中,给定第 i 个输入变量 x_i 满足第 j 个规则 A_{ij} 的隶属度函数 (MF) $\mu_{ij}(x_j)$ 。每一个 MF 的形式如下:

$$u_{ij} = \exp\left[-\frac{(x_i - c_{ij})^2}{2\sigma_{ij}^2}\right], i=1,2,\dots,r, j=1,2,\dots,L \quad (2)$$

式中, u_{ij} 是第 j 个规则的第 i 个隶属度函数; c_{ij} 是第 j 个规则的第 i 个隶属度函数的中心; σ_{ij} 是第 j 个规则的第 i 个隶属度函数的宽度; r 是输入变量的维数; u 是神经元的个数。

第 j 个规则的输出为:

$$\phi(x; c, \sigma) = \exp\left[-\frac{\sum_{i=1}^r (x_i - c_{ij})^2}{2\sigma_{ij}^2}\right], j=1,2,\dots,L \quad (3)$$

归一化为:

$$\phi(x; c, \sigma) = \frac{\phi(x; c, \sigma)}{\sum_{k=1}^L \phi(x; c, \sigma)} = \frac{\exp\left[-\frac{\sum_{i=1}^r (x_i - c_{ij})^2}{2\sigma_{ij}^2}\right]}{\sum_{k=1}^L \exp\left[-\frac{\sum_{i=1}^r (x_i - c_{ik})^2}{2\sigma_{ik}^2}\right]} \quad (4)$$

系统的输出是所有归一化后输出的加权和,因此对于给定的 x 系统的输出 \hat{y} 定义为:

$$\hat{y} = \frac{\sum_{j=1}^L \beta_j \phi(x; c, \sigma)}{\sum_{j=1}^L \phi(x; c, \sigma)} = \sum_{j=1}^L \beta_j \phi(x; c, \sigma) \quad (5)$$

式中, $\beta_j = (\beta_{j1}, \beta_{j2}, \dots, \beta_{jm})$ 代表输出权值向量, $\phi(\cdot)$ 代表隐含层节点的输出函数。

Huang 和 Chen^[13] 理论证明了具有激励函数 $\phi(\cdot)$ 的单层模糊神经网络,隶属度函数 ϕ 随机产生,能够逼近任意连续目标函数。

对任意 N 个训练样本对 (x_i, t_i) , $x_i = [x_{i1}, x_{i2}, \dots, x_{ir}]^T$

$\in R^n$, $t_i = [t_{i1}, t_{i2}, \dots, t_{im}]^T \in R^m$, PIEL-FNN 具有 L 个模糊规则和相应的参数 β_j, c_j, σ_j , 其数学模型如下:

$$f_L(x_i) = \sum_{j=1}^L \beta_j \phi(x_i; c, \sigma) = t_i, j=1, \dots, N \quad (6)$$

相应参数是输入变量的线性方程如下:

$$\beta_j = x_i^T q_j \quad (7)$$

式中, x_i 是对 x_i 的一个扩展即 $[1, x_i^T]^T$, q_j 是 TSK 模型中对应的第 i 个规则的参数矩阵,如下:

$$q_j = \begin{bmatrix} q_{j1,0} & \dots & q_{jm,0} \\ \vdots & \dots & \vdots \\ q_{j1,n} & \dots & q_{jm,n} \end{bmatrix}_{(n+1) \times m} \quad (8)$$

因此,式(6)的输出转变为如下形式:

$$f_L(x_i) = \sum_{j=1}^L x_i^T q_j \phi(x_i; c, \sigma) = t_i, j=1, \dots, N \quad (9)$$

等式可以进一步写成如下更加紧凑的形式:

$$H\beta = T \quad (10)$$

式中, H 是隐含层输出矩阵, β 是 TSK 模型的参数矩阵,分别为:

$$\begin{aligned} H(c, \dots, c_L, \sigma_1, \dots, \sigma_L; x_1, \dots, x_N) \\ = [f(x_1, c_1, \sigma_1) \dots f(x_1, c_L, \sigma_L)] \\ = [x_1^T \phi(x_1; c_1, \sigma_1), \dots, x_1^T \phi(x_1; c_L, \sigma_L)] \end{aligned} \quad (11)$$

$$\beta = \begin{bmatrix} q_1 \\ \vdots \\ q_L \end{bmatrix} \quad (12)$$

在 ELM 方法中,隐含层输出矩阵 H 是随机产生的,独立于训练集数据,输出权值 Q 是分析决定的。输出权值的估计计算如下:

$$\hat{\beta} = H^\dagger T \quad (13)$$

式中, H^\dagger 是 H 的广义逆矩阵或者自然逆。通常通过奇异值分解获得。近来, Huang^[13] 指出, ELM 不需要进行参数调整,这是对传统学习算法的挑战。

有理论表明函数逼近的所有隐含层参数可以不需要专家知识而随机产生。Huang^[13] 证明了隐含层的输出权值矩阵可以通过获得期望逼近误差来计算。当然这是在假设激励函数可微的前提下。ELM 的一致逼近能力已经被证明,因此, ELM 是一种极快的方式,在达到与传统方法同样的准确率的情况下,其快了近 10 倍。

3 修剪 ELM

3.1 按照规则的性能排列模糊规则

最佳修剪 ELM(OP-ELM)^[10] 是一个基于 ELM 的方法。OP-ELM 通过 3 个阶段实现。首先,构建 ELM,然后对隐含层利用 MRSR(multi-response sparse regression)对神经元精确排名,最后通过留一法 leave-one-out 错误估计来确定哪些神经元需要删除。

本文采用奇异值分解来对规则库中的规则的性能进行排序。矩阵 $H \in R^{n \times l}$ 可以用 SVD 方法分解为下式:

$$H = USV^T \quad (14)$$

式中, $U = (u_1, u_2, \dots, u_n) \in R^{n \times n}$ 和 $V = (v_1, v_2, \dots, v_l) \in R^{l \times l}$ 为正交矩阵, $S = \text{diag}(\xi_1, \xi_2, \dots, \xi_l) \in R^{n \times l}$ 为对角矩阵,同时 $\xi_1 \geq \xi_2 \geq \dots \geq \xi_l \geq 0$ 。矩阵 S 的对角元通常称为矩阵 H 的奇

异值。

SVD的一个重要特性就是它反映出了矩阵 H 的秩,即 $rank(H)=rank(S)$,因此,非零奇异值的个数就是 H 的秩。假设 $b=rank(H)$,则式(13)可以表示成如下形式:

$$H = \sum_{i=1}^b \xi_i u_i v_i^T \quad (15)$$

式中, $\xi_1, \xi_2, \dots, \xi_b$ 是矩阵 H 的 b 个非零奇异值,每个奇异值都相应地与一个模糊规则相关,同时它在 H 中的位置意味着它在规则库中相关规则的位置。因此利用该方法来对规则进行排序。

3.2 模型选择

一旦规则库中的规则的重要性按顺序排列好后,模型最佳的神经元个数就可以选择出来了,LOO(leave-one-out)^[8] 常用来进行选择。

$$\epsilon_i^{press} = \frac{t_i - h_i b_i}{1 - h_i p h_i^T} \quad (16)$$

式中, i 代表第 i 个隐层节点, h_i 是输出矩阵 H 的排序后的第 i 列, b_i 是输出权值, p 定义为 $p = (H^T H)^{-1}$,最佳神经元个数的选择可以通过估计不同节点数量的 LOO 误差来获得,选择最小化误差 L 个神经元:

$$L = \operatorname{argmin}_{j \in \{1, \dots, N\}; n_i=1} \sum \epsilon_i^{press} \quad (17)$$

可以看出通过前面的排序阶段有两个好处:一是收敛速度加快,二是所需要的神经元个数是最低的。

3.3 在线增量 ELM

原始的 ELM 方法是用于离线模型的,然而,由于计算简单,后来扩展为有效的在线方式,即提出了 OS-ELM^[14]。算法首先计算标准的 ELM 模型:初始化训练集。对于输出矩阵 T_0 、隐层节点矩阵 H_0 ,以及解 $\beta^0 = (H_0^T H_0)^{-1} H_0^T T_0$,使用 H 的摩尔逆来求解。定义 $K_0 = H_0^T H_0$ 。然后,对一个新的数据或者数据块,模型通过以下形式来更新:

$$\beta^{k+1} = \beta^k + P_{k+1} H_{k+1}^T (T_{k+1} - H_{k+1} \beta^k) \quad (18)$$

$$P_{k+1} = P_k - P_k H_{k+1}^T (I + H_{k+1} P_k H_{k+1}^T)^{-1} H_{k+1} P_k \quad (19)$$

式中, $P_{k+1} = K_{k+1}^{-1}$ 。

作为一种特殊情况,当数据是一个单个样本 (x_{k+1}, t_{k+1}) 而不是数据块时,按照如下简化形式来更新模型。

$$\beta^{k+1} = \beta^k + P_{k+1} h_{k+1}^T (t_{k+1} - h_{k+1} \beta^k) \quad (20)$$

$$P_{k+1} = P_k - \frac{P_k h_{k+1}^T P_k}{I + h_{k+1} P_k h_{k+1}^T} \quad (21)$$

式中, $h_{k+1} = [f(x_{k+1}, c_1, \sigma_1) \dots f(x_{k+1}, c_N, \sigma_N)]$ 。

3.4 PIEL-FNN 完整算法

在 PIEL-FNN 模型中,规则库的结构能够随着新来的数据而改变。因此模型的输出可以表达如下:

$$\Phi(j, x_j) = \sum_{i=1}^{L(j)} x_i^T \beta_i(j) f(x_j, c_i(j), \sigma_i(j)) = t_j \quad (22)$$

这样模糊规则的前件和后件都得到了进化。进一步得到式(22)的压缩形式如下:

$$H(j) \beta(j) = T(j) (j > 1) \quad (23)$$

将 x_j 作为最新的输入, $H(j)$ 是进化矩阵, $L(j)$ 是规则数,这里规则库完全进化,因此隐含层输出矩阵采用如下形式:

$$H(j) = \begin{bmatrix} F(x_k, c_1(j), \sigma_1(j)) & \dots & F(x_k, c_{L(j)}(j), \sigma_{L(j)}(j)) \\ \vdots & \ddots & \vdots \\ F(x_j, c_1(j), \sigma_1(j)) & \dots & F(x_j, c_{L(j)}(j), \sigma_{L(j)}(j)) \end{bmatrix} \quad (24)$$

$1 \leq k \leq j$ 定义了一个新输入数据的滑动窗口, $L(j)$ 是规则的进化数, $c_i(j) \in C$ 和 $\sigma_i(j) \in A$ 是输入隶属度函数的参数,是随机产生的参数值。

综上所述,PIEL-FNN 完整算法描述如下:

初始阶段,规则的前件 IF 部分随机地产生参数集 C 和 A 。

1. 给定 M 个观测样本 $(x_j, t_j) \in \mathbb{R}^r \times \mathbb{R}^m$ 的初始序列,通过应用 N 个模糊基函数式(11)创建一个 $M \times N$ 矩阵, $H_0(j), j = M$;

2. 使用 SVD 对模糊基函数的输出 $H_0(j)$ 的规则按照性能进行排序,按照式(17)选择最佳的规则数。获得结果 $H(j), j = M$,根据式(13)计算 $\beta(j), j = M$,根据式(9)获得初始进化模型 $\Phi(\cdot)$;

3. 对每一个新的输入数据 $(x_j, t_j), j > M$ do;

3.1 使用新的输入数据更新 $H_0(j)$,使用 SVD 对 $H_0(j)$ 进行排序,然后按照式(17)选择最佳规则数得到 $H(j)$;

3.2 找到分析解 $\beta(j)$,计算模型输出 $\hat{t}_j + 1$ 。

4 仿真实验

这一节,通过仿真实验来验证 PIEL-FNN 的性能,并与其他系统做比较分析。

4.1 实验 1

非线性动态系统的辨识

$$y_n = \frac{y(n-1)y(n-2)(y(n-1)+2.5)}{1+y^2(n-1)+y^2(n-2)} + u(n-1) \quad (25)$$

该对象被很多学者都研究过。本文用 PIEL-FNN 来逼近非线性函数,训练用数据 $u(n)$ 是通过在 $[-2, 2]$ 均匀分布获得的,测试用输入数据是由 $u(n) = \sin(2\pi n/25)$ 得到的。从训练数据中随机取 5000 个样本数据,从测试数据中随机取 200 个样本。将均匀分布噪声 $[-0.2, 0.2]$ 加到训练样本中,而测试样本无噪声加入。选择 $[y(n-1), y(n-2), u(n-1)]$ 和 $y(n)$ 作为 PIEL-FNN 的输入和输出,方程等式如下:

$$\hat{y}(n) = \hat{f}(y(n-1), y(n-2), u(n-1)) \quad (26)$$

式中, \hat{f} 是 PIEL-FNN 的逼近, \hat{y} 是 PIEL-FNN 的输出,表 1 给出了 PIEL-FNN, ANFIS, SAFIS, eTS, Simpl_eTS 和 DENFIS 的比较。从表中可以看出,PIEL-FNN 能够获得比其他算法类似甚至更好的测试准确率,而且,也可以看出 PIEL-FNN 需要的规则数量比 eTS, Simpl_eTS 和 DENFIS 更少,而且比其他算法所用的训练时间也更少。从表 1 还可以看出,PIEL-FN 获得的标准差更小,这说明随机选择隶属度函数的前件参数不影响测试性能。

表 1 非线性系统辨识的性能比较

algorithms	Training(RMSE)		Testing(RMSE)		Training Time(s)	# Rules
	Mean	SD	Mean	SD		
PIEL-FNN	0.1162	0.0060	0.0481	0.0067	15.362	30
SAFIS	0.1493	0.0092	0.0533	0.0103	22.473	30
DENFIS	0.1398	0.0067	0.0763	0.0023	25.234	29
eTS	0.1620	0.154	0.0608	0.0109	29.250	31

4.2 实验 2

这里考虑另一个函数

$$y = (1 + x_1^{-2} + x_2^{-1.5})^2 \quad (27)$$

其中 $1 \leq x_1, x_2 \leq 5$ 。我们采用 50 个输入输出对作为训练样

本,50 个样本对作为测试样本。不同学习方法的性能和泛化能力比较如表 2 所列。

表 2 与其他算法的泛化性能比较

algorithms	RMSE	Time(S)
PIEL-FNN	0.0021	10.785
ANFIS	0.0020	319.876
SAFIS	0.0015	401.027
eTS	0.0017	425.371

从表 2 可以看出,PIEL-FNN 在达到与其他算法类似的均方误差的情况下,需要的训练时间比其他算法短很多。

4.3 实验 3

时间序列预测在解决许多实际问题时是非常重要的。Mackey-Glass 混沌时间序列是 Mackey 等人在 1977 年所提出的一种模型。它可以用来描述复杂的生理过程。这里,为了比较,使用一个理论实例,微分方程为:

$$x(t) = \frac{0.2x(t-\tau)}{1+x^{10}(t-\tau)} - 0.1x(t) \quad (28)$$

初始条件 $x(0)=1.2, x(t)=0$, 其中 $t < 0, \tau=17$, 在 $(x(t+85))$ 前面的 85 个系列值基于 4 个输入进行建模: $x(t), x(t-6), x(t-12), x(t-18)$, 预测结果如图 1 所示。

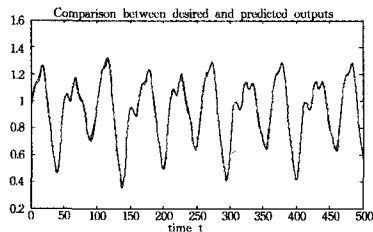


图 1 Mackey-Glass 时间序列; 预测了 500 个样本

与其他算法的性能比较如表 3 所列。

表 3 Mackey-Glass 混沌时间序列性能比较

algorithms	Training(RMSE)		Testing(RMSE)		Training Time(s)	# Rules
	Mean	SD	Mean	SD		
PIEL-FNN	0.1230	0.0050	0.0452	0.0056	6.03e-01	27
ANFIS	0.2320	0.0066	0.0065	0.0076	9.60e-01	50
SAFIS	0.1325	0.0083	0.0455	0.0230	8.70e-01	34
DENFIS	0.2430	0.0077	0.0530	0.0324	8.30e-01	35
eTS	0.2310	0.0234	0.0562	0.0109	8.60e-01	38

从表 3 可以看出,PIEL-FNN 能够获得比较好的准确率,甚至比其他算法更好,并且由于通过了修剪,所需要的规则数相对其他算法来说很少,因此训练时间也是最少的,这是因为该算法的初始结构是随机产生的。规则前件参数在训练过程中没有进行调整。

结束语 本文提出了一种带修剪的增量极速学习模糊神经网络。为了能在线增量学习,对 ELM 进行了扩展。该算法中,模糊规则的前件参数是随机产生的,初始规则数量也是随机产生的。然后使用 SVD 对规则按照重要性排序,通过 LOO 选择出最佳的模糊规则数,最后,模糊规则的后件参数通过分析计算得出。仿真实验结果表明,该算法与其他算法相比,具有较好的泛化性能和比较简单的结构,计算简单,能够达到快速学习,而且在学习过程中无须保存过去的数

据,节约了存储空间,是真正的在线增量学习。

参考文献

- [1] Li H X, Liu Z. A probabilistic neural-fuzzy learning system for stochastic modeling [J]. IEEE Transaction on Fuzzy System, 2008, 16(4): 898-908
- [2] Zhao Y, Gao H J, Mou S S. Asymptotic stability analysis of neural networks with successive time delay components[J]. Neurocomput, 2007, 71(13-15): 2848-2856
- [3] Huang H, Ho D W C, Cao J D. Analysis of global exponential stability and periodic solutions of neural networks with time-varying delays[J]. Neural Netw, 2005, 18(2): 161-170
- [4] Kasabov N K, Song Q. DENFIS: Dynamic Evolving Neural Fuzzy Inference System and Its Application for Time-Series Prediction[J]. IEEE Trans. Fuzzy Syst., 2002, 10(2): 144-154
- [5] Angelov P P, Filev D P. An approach to online identification of Takagi-Sugeno fuzzy models[J]. IEEE Trans. Syst. Man Cybern. B, 2004, 34(1): 484-498
- [6] Angelov P, Filev D. Simpl_eTS: A simplified method for learning evolving Takagi-Sugeno fuzzy models [C] // Proc. 14th IEEE Int. Conf. Fuzzy Syst. 2005: 1068-1073
- [7] Rong H-J, Sundararajan N, Huang G-B, et al. Sequential Adaptive Fuzzy Inference System(SAFIS) for nonlinear system identification and prediction[J]. Fuzzy Sets Syst., 2006, 157(9): 1260-1275
- [8] Pouzols F M, Lendasse A. Evolving fuzzy optimally pruned extreme learning machine for regression problems [J]. Evolving Systems, 2010, 1(1): 43-58
- [9] Huang G B, Zhu Q Y, Siew C K. Extreme learning machine: Theory and applications [J]. Neurocomputing, 2006, 70(1-3): 489-501
- [10] Miche Y, Sorjamaa A, Bas P, et al. OP-ELM: Optimally pruned extreme learning machine [J]. IEEE Trans. Neural Network, 2010, 21(1): 158-162
- [11] Miche Y, Sorjamaa A, Lendasse A. OP-ELM: Theory, Experiments and a Toolbox [C] // Proc. Int. Conf. Artif. Neural Netw., ser. Lect. Notes Comput. Sci. 2008, 5163: 145-154
- [12] Sorjamaa A, Miche Y, Weiss R, et al. Long-Term Prediction of Time Series using NNE-based Projection and OP-ELM [C] // Proc. Inter. Jt. Conf. Neural Netw., Hong Kong, China, June 2008: 2675-2681
- [13] Huang G-B, Chen L, Siew C-K. Universal approximation using incremental constructive feedforward networks with random hidden nodes [J]. IEEE Trans. Neural Netw., 2006, 17(4): 879-892
- [14] Liang N-Y, Huang G-B, Huang G-B, et al. A fast and accurate online sequential learning algorithm for feedforward networks [J]. IEEE Trans. Neural Netw., 2006, 17(6): 1411-1423
- [15] 许哲万, 李昌皎, 王爱侠, 等. 一种基于改进 T-S 模糊推理的模糊神经网络学习算法 [J]. 计算机科学, 2011, 38(11): 196-199, 219
- [16] 朱彦卿, 何怡刚. 一种基于模糊神经网络的模拟电路故障诊断方法 [J]. 计算机科学, 2010, 37(12): 280-28