

文章编号: 1001-0920(2005)03-0303-05

限定记忆的前向神经网络在线学习算法研究

盛守照, 王道波, 黄向华

(南京航空航天大学 自动化学院, 江苏 南京 210016)

摘要: 从理论上分析了隐含层激励函数满足Mercer条件的前向神经网络的数学本质, 给出了网络学习的指导方向。提出3种网络在线学习算法, 它们通过动态调整网络结构和权值来提高网络在线预测性能。算法完全符合统计学习理论提出的结构风险最小化原则, 具有较快的学习收敛速度和良好的抗噪声能力。最后通过具体数值实验验证了上述算法的可行性和优越性。

关键词: 前向神经网络; 在线学习; 统计学习理论; 机器学习

中图分类号: TP183 **文献标识码:** A

Online learning algorithm for feedforward neural networks with moving range

SHEN G Shou-zhao, WANG D ao-bo, HUANG X iang-hua

(College of Automation Engineering, Nanjing University of Aeronautics and Astronautics, Nanjing 210016, China)

Correspondent: SHENG Shou-zhao, E-mail: shengsz@163.com

Abstract: The mathematic essence of feedforward neural networks whose activation function of hidden neurons satisfies Mercer condition is analyzed. And the guideline for feedforward neural networks learning is given. Then three online learning algorithms are proposed, which can improve the online prediction performance of the networks by adjusting its architecture and connection weights dynamically. Those algorithms with global convergence and good antinnoise performance correspond to the principle of structural risk minimization. Their reliability and advantage are illustrated through concretely test.

Key words: feedforward neural networks; online learning; statistical learning theory; machine learning

1 引言

前向神经网络学习可分为离线学习和在线学习两种网络构成方式^[1]。一般说, 在线学习受到多方面条件的限制, 加上网络本身的复杂性以及学习收敛慢等因素, 传统的网络学习方法大多采用离线的方式。离线学习虽然能使神经网络在系统工作之前预先形成网络结构, 但实际系统往往存在着参数不可测性和时变性, 以及学习样本不可能覆盖系统所有的工作范围, 使得离线学习甚至没有有效的历史观测值可供利用, 造成直接运用离线学习得到的模型结构用于在线预测时, 结果往往与期望输出值差异

很大。究其原因, 主要在于系统实时变化的因素未在网络结构中反映出来^[2-6]。

本文主要探讨前向神经网络在线学习方法, 着重分析一类隐含层激励函数满足Mercer^[7,8]条件的前向神经网络的数学本质, 从而给出网络学习的指导方向。文中提出3种合理的网络在线学习算法, 它们通过动态调整网络结构和权值提高网络的在线预测性能, 并且具有较快的学习收敛速度和良好的抗噪声能力。

2 前向神经网络数学分析与设计理论

考虑MISO型三层前向神经网络, 设 $X \subseteq$

收稿日期: 2004-04-29; 修回日期: 2004-08-12

基金项目: 航空科学基金项目(01C52015)。

作者简介: 盛守照(1977—), 男, 安徽庐江人, 博士生, 从事机器学习、智能控制等研究; 王道波(1957—), 男, 河北易县人, 教授, 博士生导师, 从事先进无人机飞行控制、机电模拟等研究。

\mathbf{R}^p), $W (\subseteq \mathbf{R}^p)$, $\Phi (\subseteq \mathbf{R}^q)$ 和 $Y (\subseteq \mathbf{R})$ 分别为网络输入向量定义域、隐含层权值向量定义域、隐含层输出空间和网络输出域; p 和 q 分别为网络输入向量维数和隐含层神经元数; $\mathcal{Q}_x(\omega)$ 为隐含层激励函数, 输出层为线性激励函数; $\tilde{\omega} = \{\omega\}_{i=1}^q, \alpha = \{a_i\}_{i=1}^q$ 分别为网络隐含层权值向量集和输出层权值集

2.1 网络数学分析

设 Φ 为网络输入空间 X 到隐含层输出空间 Φ 的映射

$$\Phi_x, \tilde{\omega} = [\mathcal{Q}_x(\omega), \dots, \mathcal{Q}_x(\omega)]^T \quad (1)$$

对于常用的 Sigmoid 和 RBF 等类型的前向神经网络, 隐含层激励函数 $\mathcal{Q}_x(\omega)$ 满足 Mercer 条件^[8], 存在从空间 \mathbf{R}^p 到高维 Hilbert 空间 H 的非线性映射 $\Psi: \mathbf{R}^p \rightarrow H$, 使得

$$\mathcal{Q}_x(\omega) = \Psi^T(\omega) \Psi(x) \quad (2)$$

式(2)代入式(1), 得

$$\Phi_x, \tilde{\omega} = [\Psi(\omega), \dots, \Psi(\omega)]^T \Psi(x) \quad (3)$$

记 $\Gamma(\tilde{\omega}) = [\Psi(\omega), \dots, \Psi(\omega)]^T$, 则式(3)可改写为

$$\Phi_x, \tilde{\omega} = \Gamma(\tilde{\omega}) \Psi(x) \quad (4)$$

式中 $q = \dim(H)$. 式(4)表明, 整个隐含层通过非线性映射 Ψ 和线性降维映射 Γ , 实现了从空间 X 到空间 H 的升维映射后再降维映射到空间 Φ

设 f 为隐含层输出空间 Φ 到网络输出域 Y 的映射

$$f(x, \tilde{\omega}, \alpha) = \sum_{i=1}^q a_i \mathcal{Q}_x(\omega) \quad (5)$$

记 $a = [a_1, a_2, \dots, a_q]^T$, 则式(5)可改写为

$$f(x, \tilde{\omega}, \alpha) = a^T \Gamma(\tilde{\omega}) \Psi(x) \quad (6)$$

输出层最终将隐含层输出向量线性降维映射到输出域, 整个网络映射结构如图 1 所示

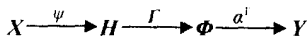


图 1 前向神经网络映射示意

从上述分析不难看出, 隐含层权值向量集 $\tilde{\omega}$ 是网络设计的关键

2.2 网络设计理论

已知系统的有限观测样本集

$$Z_n = \{(x_i, y_i) \mid y_i = g(x_i) + \epsilon_i\}_{i=1}^n \quad (7)$$

其中: $g(x_i)$ 为未知系统的实际输出, ϵ_i 为服从独立同分布零均值白噪声. 寻找合适的网络结构参数 $\tilde{\omega}$ 和 α 使得下式定义的预测风险:

$$J(\tilde{\omega}, \alpha) = \frac{1}{n} \sum_{i=1}^n [f(x_i, \tilde{\omega}, \alpha) - g(x_i)]^2 \quad (8)$$

最小

记 $X_n = \{x_i\}_{i=1}^n$ 为 Z_n 的输入部分, 形如 $H(X) (\subseteq H)$ 为 $x \in X (\subseteq \mathbf{R}^p)$ 在非线性映射 Ψ 下的像空

间, 则有如下结论:

定理 1 如果 $\tilde{\omega}$ 满足条件

$$H(\tilde{\omega}) \supseteq H(X_n), \quad (9)$$

$\exists \alpha \in \mathbf{R}$, 则使 $J(\tilde{\omega}, \alpha) = \min_{\tilde{\omega} \in \mathbf{R}^p, \alpha \in \mathbf{R}} J(\tilde{\omega}, \alpha)$ 成立

证明 若 $\tilde{\omega}$ 满足式(9), 则对于任意给定的 $\tilde{\omega}$, 存在相应的矩阵 C , 使得 $\Gamma(\tilde{\omega}) [\Psi(x_1), \dots, \Psi(x_n)] = C^T \Gamma(\tilde{\omega}) [\Psi(x_1), \dots, \Psi(x_n)]$. 由此不难得出上述结论

定理 1 只是网络最优化设计的一个充分条件, 而不是必要条件. 为简化网络结构, 实际应用时应尽可能选择少量的隐含层神经元, 使得式(8)定义的性能指标满足要求

定理 2 若下列任一情况成立:

- 1) $H(W) \supseteq H(X)$;
- 2) $W = X$;
- 3) $W = X_n$.

$\exists \tilde{\omega} \in W$, 则使式(9)成立

取 $W = X_n$, 不仅能大大缩小隐含层权值向量的搜索范围, 提高网络学习效率, 而且能找到使式(8)最小的隐含层权值向量集 $\tilde{\omega}$

另一方面, 为降低网络的复杂性, 要求 $H(\tilde{\omega})$ 中各向量相互独立. 记

$$\Phi(\tilde{\omega}) = \Gamma(\tilde{\omega}) \Gamma^T(\tilde{\omega}) = \begin{bmatrix} \mathcal{Q}(\omega_1, \omega_1) & \mathcal{Q}(\omega_1, \omega_2) & \dots & \mathcal{Q}(\omega_1, \omega_q) \\ \mathcal{Q}(\omega_2, \omega_1) & \mathcal{Q}(\omega_2, \omega_2) & \dots & \mathcal{Q}(\omega_2, \omega_q) \\ \vdots & \vdots & \ddots & \vdots \\ \mathcal{Q}(\omega_q, \omega_1) & \mathcal{Q}(\omega_q, \omega_2) & \dots & \mathcal{Q}(\omega_q, \omega_q) \end{bmatrix}$$

也就是要求 $\tilde{\omega}$ 满足

$$\text{rank}[\Phi(\tilde{\omega})] = q \quad (10)$$

上述定理及式(10)为选择隐含层权值向量集 $\tilde{\omega}$ 提供了理论指导

记

$$B(\tilde{\omega}) = [\Phi_{x_1}, \tilde{\omega}, \dots, \Phi_{x_n}, \tilde{\omega}]^T,$$

$$K(\tilde{\omega}) = B(\tilde{\omega}) [B^T(\tilde{\omega}) B(\tilde{\omega})]^{-1} B^T(\tilde{\omega}),$$

$$y = [y_1, y_2, \dots, y_n]^T,$$

$$y^* = [g(x_1), \dots, g(x_n)]^T,$$

$$y(\tilde{\omega}, \alpha) = [f(x_1, \tilde{\omega}, \alpha), \dots, f(x_n, \tilde{\omega}, \alpha)]^T,$$

$$\epsilon = [\epsilon_1, \epsilon_2, \dots, \epsilon_n]^T,$$

$$J_\epsilon(\tilde{\omega}, \alpha) = \frac{1}{n} K(\tilde{\omega}) \epsilon - \epsilon^2.$$

若 $\tilde{\omega}$ 满足式(10), 则

$$a = [B^T(\tilde{\omega}) B(\tilde{\omega})]^{-1} B^T(\tilde{\omega}) y^*, \quad (11)$$

$$y(\tilde{\omega}, \alpha) = K(\tilde{\omega}) y^*, \quad (12)$$

$$J(\tilde{\omega}, \alpha) = \frac{1}{n} K(\tilde{\omega}) y^* - y^*^2. \quad (13)$$

然而, 困难在于无法从样本观测输出 y_i 中分离出系

统的真实输出 $g(x_i)$, 若以系统观测输出 y_i 替代系统实际输出 $g(x_i)$, 则式(11) ~ (13) 的估计分别为

$$\hat{a} = [B^T(\tilde{\omega})B(\tilde{\omega})]^{-1}B^T(\tilde{\omega})y, \quad (14)$$

$$\hat{y}(\tilde{\omega}|\alpha) = K(\tilde{\omega})y, \quad (15)$$

$$\hat{J}(\tilde{\omega}|\alpha) = \frac{1}{n} K(\tilde{\omega})y - y^2. \quad (16)$$

考虑到 y_i^* 与 ϵ 的无关性以及 $E(\epsilon) = 0$, 有

$$\text{var}(\hat{a}) = [B^T(\tilde{\omega})B(\tilde{\omega})]^{-1}\text{var}(\epsilon), \quad (17)$$

$$\text{var}[\hat{y}(\tilde{\omega}|\alpha)] = K(\tilde{\omega})\text{var}(\epsilon), \quad (18)$$

$$E[\hat{J}(\tilde{\omega}|\alpha)] = E[J(\tilde{\omega}|\alpha)] + E[J_\epsilon(\tilde{\omega}|\alpha)] \quad (19)$$

为降低样本噪声对网络结构参数以及预测输出的影响, 要求 $\tilde{\omega}$ 满足如下约束条件:

$$\text{trace}\{[B^T(\tilde{\omega})B(\tilde{\omega})]^{-1}\} \leq m \text{ in}, \quad (20)$$

$$\text{trace}[K(\tilde{\omega})] \leq m \text{ in}. \quad (21)$$

考虑到 $K(\tilde{\omega})$ 为 n 阶 Hermite 非负定矩阵, 且存在正交矩阵 $U(\tilde{\omega})$, 使得

$$U^T(\tilde{\omega})K(\tilde{\omega})U(\tilde{\omega}) = \begin{bmatrix} I_q & 0 \\ 0 & 0 \end{bmatrix},$$

则

$$\text{trace}[K(\tilde{\omega})] = q,$$

$$E[J_\epsilon(\tilde{\omega}|\alpha)] = (1 - q/n)\text{var}(\epsilon).$$

式(21) 可简化为如下约束要求:

$$q \ll n. \quad (22)$$

此时

$$E[\hat{J}(\tilde{\omega}|\alpha)] \approx E[J(\tilde{\omega}|\alpha)] + \text{var}(\epsilon).$$

从统计意义上说, 可以近似认为

$$\hat{J}(\tilde{\omega}|\alpha) \approx J(\tilde{\omega}|\alpha) + \text{var}(\epsilon).$$

根据上述理论分析, 隐含层权值向量集 $\tilde{\omega}$ 的选择, 可近似为在满足式(10), (20) 和(22) 的约束条件下求解如下问题:

$$\arg \min_{\tilde{\omega} \in W} J(\tilde{\omega}|\alpha) < \delta \quad (23)$$

其中: $\tilde{\omega} \in W = X_n$, 门限 δ 应选择在系统观测噪声方差 $\text{var}(\epsilon)$ 附近, 输出层权值集 α 由式(14) 估计.

3 限定记忆的前向神经网络在线学习算法

设 (x_k, y_k) 为连续时变系统第 k 步输入输出;

$$Z_n^k = \begin{cases} \{(x_i, y_i)\}_{i=k-n+1}^k, & k = n \\ \{(x_i, y_i)\}_{i=0}^k, & k < n \end{cases}$$

为系统第 k 步限定记忆长度 n 的输入输出序列集;

$$X_n^k = \begin{cases} \{(x_i)\}_{i=k-n+1}^k, & k = n \\ \{(x_i)\}_{i=0}^k, & k < n \end{cases}$$

为对应的输入序列集; $\tilde{\omega}, \alpha, q_k$ 为第 k 步学习后网络结构参数. 网络在线学习算法的关键是如何利用上述已知信息快速修正网络结构参数, 提高网络的预测性能

算法 1

1) 初始化限定记忆长度 n , 性能指标门限 δ , 最

大隐含层权值向量数 $q_{\max} (\ll n)$, 网络结构参数 $\tilde{\omega}_0, \alpha_0 = \emptyset$, 在线学习步数 $k = 0$

2) 在线学习步数 $k = k + 1$, 通过已知网络结构参数给出系统第 k 步预测输出 $f(x_k, \tilde{\omega}_{k-1}, \alpha_{k-1})$.

3) 采样系统第 k 步输入输出序列 (x_k, y_k) , 更新学习序列集 Z_n^k 和 X_n^k .

4) 置 $\tilde{\omega} = \emptyset, \alpha = \emptyset, W = X_n^k$.

5) 下列任意情况出现时, 转到第 7) 步; 否则, 转到第 6) 步:

$$J(\tilde{\omega}, \alpha) > \delta$$

$$q_k > q_{\max};$$

在 W 中任意权值向量 $\omega \in \tilde{\omega}$ 加入 $\tilde{\omega}$, 不满足式(10).

6) 从 W 中选择某权值向量 $\omega \in \tilde{\omega}$ 加入 $\tilde{\omega}$, 在满足式(10) 的条件下, 使得 $\min_{\tilde{\omega} = \tilde{\omega} \cup \omega} J(\tilde{\omega}, \alpha)$ 成立, 转到第 5) 步.

7) 下列任意情况出现时, 转到第 8) 步; 否则, 转到第 9) 步:

从 $\tilde{\omega}$ 中删除某权值向量 ω , 满足 $J(\tilde{\omega}, \alpha) > \delta$

$$q_k > q_{\max}$$

8) 从 $\tilde{\omega}$ 中删除某权值向量 ω , 在优先(适合情况 7) 之), 但不一定适合情况 7) 之) 满足 $J(\tilde{\omega}, \alpha) < \delta$ 的条件下, 使得 $\min_{\tilde{\omega} = \tilde{\omega} - \omega} \text{trace}\{[B^T(\tilde{\omega})B(\tilde{\omega})]^{-1}\}$ 成立, 转到第 7) 步.

9) 按式(14) 重新计算输出层权值集 α , 并转到第 2) 步继续下次学习.

下面根据上述算法给出两种改进型算法, 以便加快网络的学习速度

算法 2 替换算法 1 第 4) 步如下(其他各步同上):

4) 置 $\tilde{\omega} = \tilde{\omega}_{k-1}, \alpha = \alpha_{k-1}, W = X_n^k$ 当 $k > n$ 时, 若存在下标 i 满足 $x_{k-n} = \omega_i(\tilde{\omega})$, 则从 $\tilde{\omega}$ 中删除隐含层权值向量 ω , 从 α 中删除与之对应的输出层权值 a_i .

算法 3 替换算法 2 第 5) 步和第 6) 步如下(其他各步同上):

5) 对当前输入 x_k 加入 $\tilde{\omega}$.

6) 若式(10) 成立, 则转到第 7) 步; 否则, 从 $\tilde{\omega}$ 中删除某权值向量 ω , 且 $\min_{\tilde{\omega} = \tilde{\omega} - \omega} \text{cond}(\Phi(\tilde{\omega}))$ 成立

现对上述 3 种算法的共同点说明如下:

1) 算法第 2) 步通过上次学习获得的网络结构参数给出系统一步预测输出, 并利用 Z_n^k 对网络的结构参数进行动态修正. 考虑到提高在线学习算法的快速性要求, 均取 $W = X_n^k$.

2) 为提高网络结构参数的可靠性, 算法中将对式(10)的约束转化为 $\text{cond}[\Phi(\tilde{\omega})] < c$, 其中 $c > 0$ 为预设的常数

3) 算法第7)步和第8)步在优先满足性能指标要求的前提下, 应尽可能提高网络输出层权值集 α 的置信度, 即使式(20)成立, 并要求满足 $q_k \leq q_{\max}$. 这样一方面有利于简化网络结构, 删除隐含层的冗余神经元, 提高网络的抗噪声和泛化性能; 另一方面它还约束了算法中计算矩阵逆、矩阵秩以及第7)和第8)步的优化时间, 从而提高网络在线学习效率

4) 算法采用限定记忆长度不仅能使网络适应于时变系统的在线预测, 而且可大大减少网络在线学习时间

5) 算法每次学习获得的隐含层权值向量集 $\tilde{\omega}$, 类似于统计学习理论的支持向量机所获得的支持向量集, 算法完全符合统计学习理论提出的结构风险最小化原则. 另外, 它既克服了传统的基于梯度下降的网络学习算法可能存在欠学习或过学习以及局部最小点问题, 又具有动态优化网络结构参数的优点, 从而提高其在线预测性能

上述3种算法的不同点仅在第4)步~第6)步, 现对它们在性能和快速性上的优缺点比较如下:

1) 算法1每次学习时完全抛弃上次学习获得的网络结构参数 $\tilde{\omega}_{k-1}$ 和 α_{k-1} , 在学习上的优点是能最优化性能指标, 提高网络的泛化性能, 适合大时变系统的预测学习; 其缺点是完全抛弃已有的网络结构参数信息, 大大降低了网络学习速度. 值得指出的是, 这种算法完全可以用于离线学习

2) 算法2每次学习时继承了上次学习获得的网络结构参数 $\tilde{\omega}_{k-1}$ 和 α_{k-1} 的几乎全部信息, 仅去除 $\tilde{\omega}_{k-1}$ 中不属于 $w = X^k$ 的元素和 α_{k-1} 中对应的元素. 这样既有较好的泛化性能, 又进一步提高了网络在线学习速度

3) 算法3试图将当前输入 x_k 直接加入 $\tilde{\omega}$, 使网络隐含层尽可能包含新信息, 具有跟踪时变系统的能力, 从而大大提高了网络在线学习速度. 该算法适合于快速采样控制系统的在线预测

4 数值实验

下面通过数值实验验证上述算法的可行性和有效性. 设实验系统为

$$\begin{cases} c(t) = \exp(-0.5t) \sin(2\pi t), \\ x(t) = 1.0 \sin(\pi t) + 0.1 \sin(2\pi t), \\ v(t) = 2\pi x(t), \\ g(t) = c(t) \exp(-0.05v^2(t)) \sin v(t), \\ y(t) = g(t) + \epsilon \end{cases}$$

输入 $x(t)$, 实际输出 $g(t)$, 观测输出 $y(t)$, 观测噪声

方差 $\text{var}(\epsilon) = 0.0004$, 取 $\mathcal{Q}(x, \omega) = \exp\left[-\frac{x - \omega}{\sigma^2}\right]$, 其中 $\sigma^2 = 0.1$.

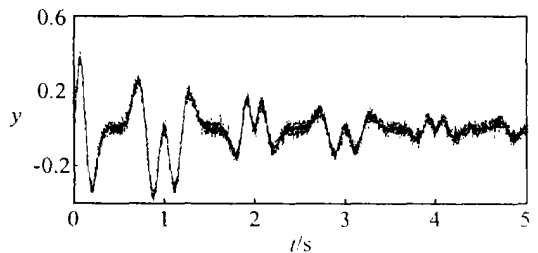
初始化参数 $n = 100, \delta = 0.0004, q_{\max} = 10$, 采样时间 $T = 0.001 \text{ s}$. 记

$$\begin{aligned} x_k &= x(T \times k), y_k = y(T \times k), \\ g_k &= g(T \times k), \\ e_k &= f(x_k, \tilde{\omega}_{k-1}, \alpha_{k-1}) - y_k, \\ e_k &= f(x_k, \tilde{\omega}_{k-1}, \alpha_{k-1}) - g_k, \\ \hat{J}_p &= \frac{1}{l} \sum_{k=1}^l e_k^2, J_p = \frac{1}{l} \sum_{k=1}^l e_k^2, \\ q_p &= \frac{1}{l} \sum_{k=1}^l q_k, t_p = \frac{1}{l} \sum_{k=1}^l t_k \end{aligned}$$

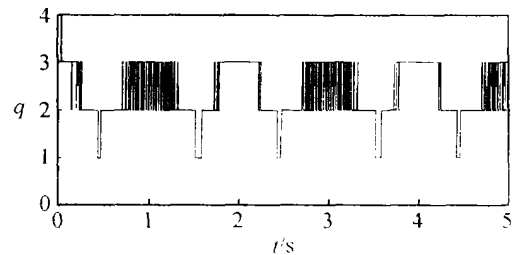
其中: t_k 为第 k 步学习时间, l 为在线学习总次数. 仿真软件为 Matlab 6.1, 操作系统为 Windows 2000, CPU 主频 666 MHz. 3种算法的实验结果如表1所示; 算法3的部分实验曲线如图2所示.

表1 3种算法实验结果比较

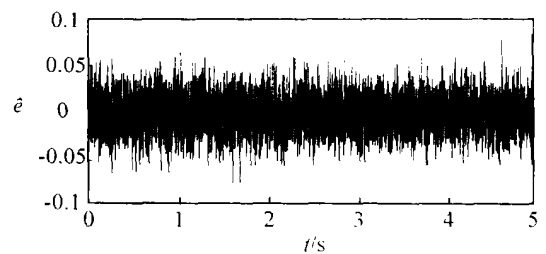
算法	$\hat{J}_p/10^{-4}$	$J_p/10^{-5}$	q_p	$t_p/10^{-3} \text{ s}$
算法1	4.0285	1.0597	3.9672	56.3679
算法2	4.0317	1.0638	3.8515	10.8437
算法3	4.0323	1.0643	2.3710	1.5116



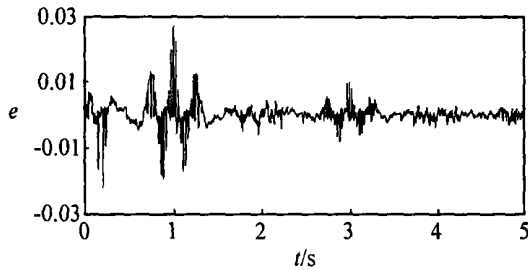
(a) 观测序列和网络预测序列



(b) 隐含层神经元数序列



(c) 网络预测对于观测输出误差序列



(d) 网络预测对于实际输出误差序列

图2 算法3部分实验结果

上述试验结果取得了令人满意的效果。3种算法完全克服了传统的学习算法可能存在的欠学习或过学习以及局部最小点问题, 具有较快的在线学习速度和良好的抗噪声和预测性能, 网络预测输出完全能跟踪上述快速时变系统的实际输出。在这方面, 传统的算法还少有较好的应用示例。

从表1和图2可以看出, 算法3不仅在线学习速度快, 而且具有良好的预测性能, 它尤其适合实时控制等领域。

5 结 论

传统的前向神经网络在线学习算法, 大多将注意力集中于改进误差反向传播算法和提高网络学习速度, 以满足网络在线学习的要求, 理论方面进展并不大, 大多源于梯度下降的思想; 如果搜索步长选择不当, 不仅降低了网络学习的收敛速度, 而且容易陷入局部最小点, 从而降低了网络的预测性能。本文提出的算法从理论上确保了网络学习性能的优化, 通过动态调整网络结构和权值来提高网络的在线预测性能, 从而使其具有快速跟踪时变系统的能力。本文算法具有较好的理论基础和应用价值, 可以推广应用用于其他机器学习领域。

参考文献 (References)

- [1] Martin T Hagan, Howard B Demuth, Mark Beale. *Neural network design* [M]. Boston: PWS Publishing Company, 1996: 60-97.
- [2] Engozinger S, Tom sen E. An accelerated learning algorithm for multilayer perceptrons: Optimization layer by layer [J]. *IEEE Trans on Neural Networks*, 1995, 6(1): 31-42.
- [3] Robert S Scalero, Nazif Tepedelenioglu. A fast new algorithm for training feedforward networks [J]. *IEEE Trans on Signal Processing*, 1992, 40(1): 202-210.
- [4] 李冬梅, 王正欧. 提高前向神经网络泛化性能和实时性能的新算法 [J]. *电机与控制学报*, 2002, 6(3): 241-264.
(Li D M, Wang Z O. A new algorithm for improving the generalization performance and real-time ability of feedforward neural networks [J]. *Electric Machines and Control*, 2002, 6(3): 241-264.)
- [5] 叶军, 张新华. 多层前向神经网络的快速学习算法及其应用 [J]. *控制与决策*, 2002, 17(增刊): 817-819.
(Ye J, Zhang X H. High speed learning algorithm for a multilayer feedforward neural network and its application [J]. *Control and Decision*, 2002, 17(S): 817-819.)
- [6] Chng E S, Chen S, Mulgrew B. Gradient radial basis function networks for nonlinear and nonstationary time series prediction [J]. *IEEE Trans on Neural Networks*, 1996, 7(1): 190-194.
- [7] 张铃. 基于核函数的SVM与三层前向神经网络的关系 [J]. *计算机学报*, 2002, 25(7): 696-700.
(Zhang L. The relationship between kernel functions based SVM and three-layer feedforward neural networks [J]. *Chinese J of Computers*, 2002, 25(7): 696-700.)
- [8] 边肇祺, 张学工. *模式识别* [M]. 北京: 清华大学出版社, 1999: 300-303.

(上接第302页)

- [9] 蔡自兴, 陈海燕, 魏世勇. 智能控制工程研究的进展 [J]. *控制工程*, 2003, 10(1): 1-6.
(Cai Z X, Chen H Y, Wei S Y. Recent advances in research of intelligent control engineering [J]. *Control Engineering of China*, 2003, 10(1): 1-6.)
- [10] Kolodner J L. *Case-based reasoning* [M]. New York:

Morgan Kaufmann, 1993.

- [11] 谭明皓. 基于案例推理的热轧层流冷却过程建模与控制研究 [D]. 沈阳: 东北大学, 2004.
- [12] 单旭沂. 宝钢2050mm热轧层流冷却控制系统改造开发 [A]. *中国科技年会* [C]. 杭州, 1999: 1-4.