

高斯激活函数特征值分解修剪技术的 D-FNN 算法研究*

何正风¹, 张德丰², 孙亚民³

- (1. 佛山科学技术学院理学院, 广东 佛山 528000;
2. 佛山科学技术学院计算机系, 广东 佛山 528000;
3. 南京理工大学计算机科学与技术学院, 江苏 南京 210094)

摘要: 提出了一种 D-FNN 的新算法。其算法的最主要特点是: D-FNN 选择高斯函数作为网络的激活函数和模糊系统的隶属函数, 该算法不仅具有强大的全局映射泛化能力, 而且在细化局部方面也有效; 使用特征值分解修剪技术使得网络结构不会持续增长, 可获得更为紧凑的 D-FNN 结构, 避免了过拟合现象。最后通过对 Hermite 多项式逼近能力来验证所提方案的有效性。仿真结果表明使用特征值分解修剪技术和高斯激活函数的 D-FNN 具有良好的性能。

关键词: 动态模糊神经网络; 模糊规则; 修剪技术; 特征值分解

中图分类号: TP183 **文献标志码:** A **文章编号:** 0529-6579 (2013) 01-0034-06

Research on Adaptive Dynamic Fuzzy Neural Network Algorithms with Gauss Activation Function and Eigenvalue Decomposition Pruning Technologies

HE Zhengfeng¹, ZHANG Defeng², SUN Yamin³

- (1. School of Science, Foshan University, Foshan 528000, China;
2. Department of Computer, Foshan University, Foshan 528000, China;
3. School of Computer and Technology, Nanjing University of Science and Technology, Nanjing 210094, China)

Abstract: A new algorithm, which uses D-FNN Gauss function as a network activation function and fuzzy membership function, is proposed. The algorithm obtains strong global mapping generalization ability and effectiveness in local refinement, and adopts eigenvalue decomposition pruning technology to enables more compact D-FNN structure to avoid the phenomenon of over fitting. Finally, the algorithm is confirmed through the Hermite polynomial approximation to approach ability of validity. The simulation results show that the eigenvalue decomposition of pruning techniques and Gauss activation function of D-FNN has good performance.

Key words: dynamic fuzzy neural network (D-FNN); fuzzy rule; pruning technology; eigenvalue decomposition (ED)

从神经网络的角度来看, 由于神经网络是一个黑箱, 不具有明确的物理意义, 因此, 专家对研究对象的部分知识无法应用到神经网络中。但是, 一

个模糊神经网络却有明确的物理意义, 专家知识很容易结合到模糊神经网络中。因此, 模糊神经网络不仅吸取了模糊逻辑和神经网络二者的优点, 还克

* 收稿日期: 2012-07-30

基金项目: 广东省自然科学基金资助项目 (S2011020002719)

作者简介: 何正风 (1957 年生), 男, 讲师; E-mail: hzf09@126.com

服了各自具有的缺点^[1]。正因如此, 模糊神经网络成为当前研究的一个热点, 并形成了一个较为完善的体系。本文研究的是模糊神经网络混合系统, 是模糊逻辑和神经网络一种比较高级的结合方法。在这种系统中, 模糊逻辑和神经网络各自发挥自己的功能, 利用各自的优势为共同的目标而实现功能结合与互补。

1 动态模糊神经网络的理论分析

1.1 激活函数与隶属函数的选择

一般来说, 只要是连续可微的函数都可以作为神经网络的激活函数。在实际应用中, 通常取如下 S 型函数:

$$f(x) = \frac{1}{1 + \exp(-\beta x)} \quad \beta > 0 \quad (1)$$

实际上, 任何绝对可积的激活函数都能够逼近基于傅里叶级数展开的任意函数。那么, 我们要问: 非 S 型激活函数对神经网络的实现和训练有什么影响? 由于神经网络的非线性映射能力, 在神经网络的实现和训练方面, 是否非 S 型函数能够更好地取代 S 型激活函数^[2]?

为了回答这个问题, 我们提出了一个判别函数, 该函数由有限个势函数的加权来表示:

$$\phi(x) = \sum_{i=1}^u c_i k(x, x_i) \quad (2)$$

其中, $k(x, x_i)$ 是 x 的第 i 个势函数, c_i 是实常数, u 是势函数个数。这点可以与传统的三层前馈网络的输出进行比较

$$\phi(x) = \sum_{i=1}^u c_i f\left(\sum_{j=1}^l \lambda_{ij} x_j, p_i\right) \quad (3)$$

实验研究表明: 在 S 型函数、高斯函数与正弦函数之中, 正弦激活函数提供最佳映射能力, 高斯激活函数次之, S 型函数最差。但是激活函数的泛化能力很大程度上依赖于某种映射的局部特性。这种情况下, 希望得到一个不仅具有强大的全局映射泛化能力, 而且在细化局部方面也有效, 并且在已学习的映射上无须太多改动的激活函数。这使得高斯函数激活函数在神经网络的应用方面成为一个不错的选择。由于高斯函数具有各项同性, 因此以高斯函数作为激活函数的神经网络称为径向基 (RBF) 神经网络。

一个具有 r 个输入和一个输出的 RBF 神经网络如图 1 所示。该网络可以看成如下形式的映射 $f: R^r \rightarrow R^s$:

$$y(X) = w_0 + \sum_{i=1}^u w_i R_i(\|X - C_i\|) \quad (4)$$

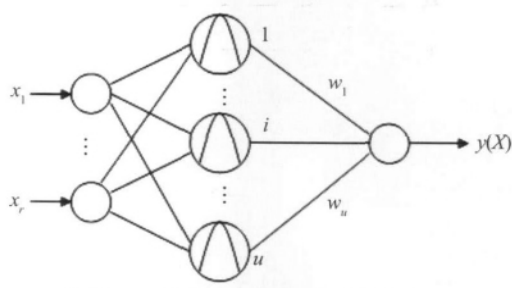


图 1 RBF 神经网络

Fig. 1 RBF neural network

其中, $X \in R^r$ 是输入向量, $R_i(\cdot)$ 是基函数, $\|\cdot\|$ 表示输入空间上的欧氏范数, $w_i (0 \leq i \leq u)$ 是权值, $C_i \in R^r$ 是 RBF 的中心, u 是 RBF 的单元数。尽管为了定义上的简单, 这里仅仅考虑了单输出的情形, 但很容易推广到多输出的情形^[3]。

如果采用高斯函数而不考虑偏置量, 则式 (4) 可以写为如下的函数:

$$y(X) = \sum_{i=1}^u w_i R_i\left(-\frac{\|X - C_i\|^2}{\sigma^2}\right) \quad (5)$$

如果把各高斯函数的输出归一化, 则 RBF 网络可以产生如下的归一化的输出响应^[4]:

$$y(X) = \frac{\sum_{i=1}^u w_i R_i\left(-\frac{\|X - C_i\|^2}{\sigma^2}\right)}{\sum_{i=1}^u R_i\left(-\frac{\|X - C_i\|^2}{\sigma^2}\right)} \quad (6)$$

模糊系统提供了一种方法, 即把基于专家知识的语言规则转换成自动的控制行为。这种方法明确地提供了一种用数学模型表达不确定性的方式。把隶属的程度定义为隶属函数 (MF), 通常该函数是一个位于 0 到 1 之间的值。通常来说, 两个用的最广泛的隶属描述为三角函数和高斯函数, 这两种类型的隶属函数分别如图 2、图 3 所示。

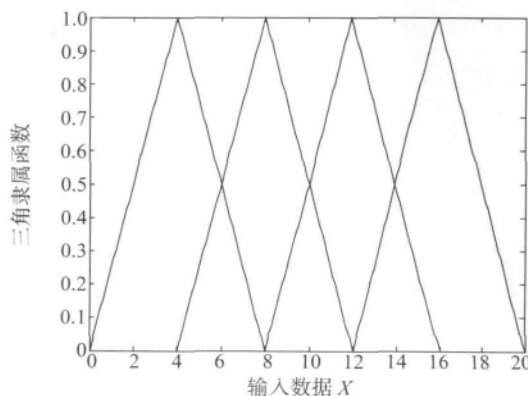


图 2 三角隶属函数

Fig. 2 Triangular MF

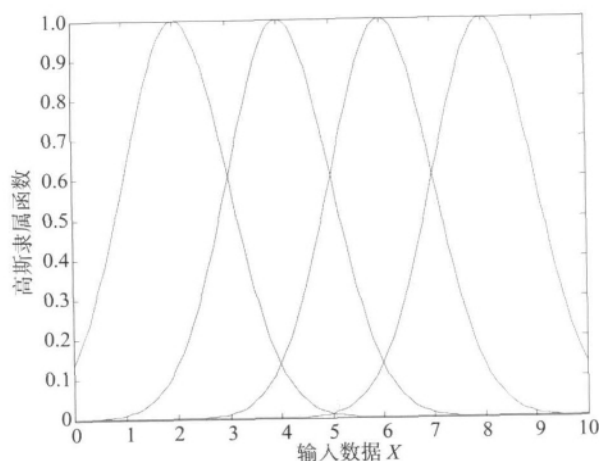


图3 高斯隶属函数

Fig. 3 Gaussian MF

1.2 动态模糊神经网络的结构

本文提出了一种动态模糊神经网络 D-FNN 结构及其学习算法, 该模糊神经网络的结构基于扩展的径向基神经网络。其学习算法的最大特点是参数的调整和结构的辨识同时进行, 且学习速度快, 可用于实时建模与控制。

本文所研究的动态模糊神经网络的结构如图 4 所示, 共 5 层。在图 4 中, x_1, x_2, \dots, x_r 是输入的语言变量, y 是系统的输出, MF_{ij} 是第 i 个输入变量的第 j 个隶属函数, R_j 表示第 j 条模糊规则, N_j 是第 j 个归一化节点, w_j 是第 j 个规则的结果参数或者连接权, u 指系统总的规则数^[5]。

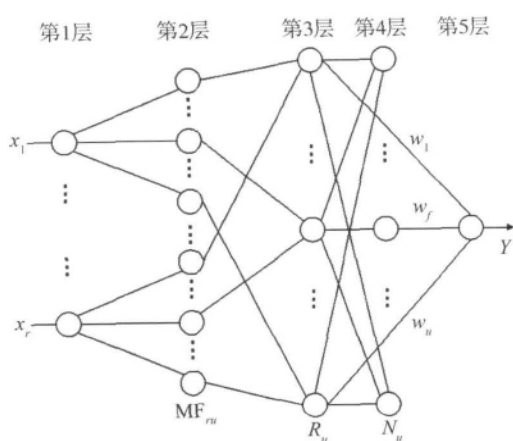


图4 D-FNN 的结构

Fig. 4 D-FNN structure

第 1 层称为输入层, 第 2 层称为隶属函数层, 第 3 层称为 T-范数层, 第 4 层称为归一化层, 第 5 层称为输出层。第 5 层中的每个节点分别表示一

个输出变量, 该输出是所有输入信号的叠加:

$$y(X) = \sum_{k=1}^u w_k \cdot \psi_k \quad (7)$$

其中, y 是变量的输出, w_k 是 THEN-部分或者第 k 个规则的连接权^[6]。

1.3 特征值分解 (ED) 修剪技术

修剪技术对于动态时变非线性系统的辨识是非常必要的。如果在学习进行时, 检测到不活跃的模糊规则并加以剔除, 则可获得更为紧凑的 D-FNN 结构。本文采用特征值分解 (ED) 方法作为一种修剪技术来选择重要的规则。

特征值分解 (ED) 的目的是为了构建简化的 RBF 神经网络以及 Volterra 级数 (一种泛函级数) 多项式。该方法使用如下的正则方程^[7]:

$$\Phi_{HH} \theta = \Phi_{HD} \quad (8)$$

其中, $\Phi_{HH} = H^T H \in \mathbf{R}^{v \times v}$ 和 $\Phi_{HD} = H^T D \in \mathbf{R}^v$ 分别称为相关矩阵和互相关向量。计算 Φ_{HH} 的特征值分解可以得到:

$$\Phi_{HH} = B \Lambda B^T \quad (9)$$

其中, $\Lambda = \text{diag}(\lambda_1, \lambda_2, \dots, \lambda_v) \in \mathbf{R}^{v \times v}$ 为对角矩阵, 矩阵的对角元为 Φ_{HH} 的特征值, 同时 $B = [b_1, b_2, \dots, b_v] \in \mathbf{R}^{v \times v}$ 是正交矩阵, 该矩阵的列为相应的特征向量。由于 Φ_{HH} 是个对称非负矩阵, 特征值 $\lambda_1, \lambda_2, \dots, \lambda_v$ 都是非负实数。因此, 该分解可以通过调整使得特征值呈降序排列即 $\lambda_1 \geq \lambda_2 \geq \dots \geq \lambda_v \geq 0$ 。

如果矩阵 Φ_{HH} 包含零或者接近零的特征值, 则表明 Φ_{HH} 是一个非满秩矩阵, 从而可以去掉多余的或者不重要的规则^[8]。定义如下测量指数向量 $I_{ED} \in \mathbf{R}^v$

$$\begin{aligned} I_{ED} &= [I_{ED_1}, I_{ED_2}, \dots, I_{ED_v}]^T \\ &= \text{diag}(B_b B_b^T) \\ &= \text{diag}(b_1 b_1^T + b_2 b_2^T + \dots + b_b b_b^T) \end{aligned} \quad (10)$$

其中, $B_b = [b_1, b_2, \dots, b_b] \in \mathbf{R}^{v \times b}$ 包含矩阵 B 的前 b 列。 I_{ED} 中最大的指数 b 预示着 θ 中的应当保留下来的元素位置。同时 I_{ED} 中保留的 $v-b$ 指数预示着 θ 应当去掉的元素的位置。

假设得到的 b 维参数向量定义为 $\theta^{(b)} = [\theta_1, \theta_2, \dots, \theta_b]^T$, 那么, 它可以通过求解简化的正则方程

$$\Phi_{HH}^{(b)} \theta^{(b)} = \Phi_{HD}^{(b)} \quad (11)$$

得到。其中, $\Phi_{HH}^{(b)} \in \mathbf{R}^{b \times b}$ 是通过除去 I_{ED} 中与 $v-b$ 个最小指数相关的行和列后得到的, $\Phi_{HD}^{(b)} \in \mathbf{R}^b$ 是除去 Φ_{HD} 的 $v-b$ 个列得到的。

ED 算法的主要步骤归纳如下^[9]:

1) 由式 (9) 计算 Φ_{HH} 的特征值分解得到 Λ 和 B 。通过检查包含在 Λ 中的特征值确定保留的模糊规则的规则数。最前面 b 个最大的特征值的数量预示了应当保留的模糊规则数。

2) 把 B 分解成 $B = [B_b \ B_{v-b}]$, 其中 B_b 和 B_{v-b} 分别对应于 B 的前 b 列和后 $v - b$ 列。

3) 用式 (10) 计算测量指数向量 I_{ED} 。 I_{ED} 中的最大的 b 的预示了应当保留的 Φ_{HH} 的行和列的位置以及 Φ_{HD} 的行的位置, 即规则库的 b 个最重要规则的位置。

4) 基于选择的 b 个最重要的模糊规则, 构建简化的矩阵 $H^{(b)} \in \mathbf{R}^{n \times b}$ 并构造 $\Phi_{HH}^{(b)} \in \mathbf{R}^{b \times b}$ 和 $\Phi_{HD}^{(b)} \in \mathbf{R}^b$ 。

5) 用 LLS 或者 SVD 方法求解式 (11) 得到 $\theta^{(b)}$ 。

2 实验结果与分析

为了验证特征值分解修剪技术的 D-FNN 算法有效性, 我们把本文的算法与其他学习算法进行了比较, 并深入探讨了这些算法之间的相互关系。实验表明使用了特征值分解修剪技术的 D-FNN 具有简洁的结构和优良的性能。

研究人员已经证明一个模糊系统或神经网络是一个通用的逼近器。但是, 这只是个存在性结论, 并没有给出如何才能找到这样的模糊系统或模糊神经网络^[9]。

当一个神经网络用于函数逼近问题时, 一般的形式如下:

$$f \approx \sum_{i=1}^u w_i R_i(X; \mu_i) \quad (12)$$

其中, w_i 是权值, $R_i(X; \mu_i)$ 是基函数, u 是基函数的个数。一旦 $R_i(X; \mu_i)$ 被选定, 权值的选择就可以用线性代数中的标准方法求解。

使用特征值分解修剪技术的 D-FNN 仿真实验。被逼近的函数为如下的埃尔米特 (Hermite) 多项式^[10]:

$$f(x) = 1.1(1 - x + 2x^2) \exp\left(-\frac{x^2}{2}\right) \quad (13)$$

为了学习这个被逼近的函数, 使用区间 $[-4 \ 4]$ 内的随机样本函数来产生 200 个输入输出数据作为训练集。预先确定的参数如下, 且跟文献 [11] 和文献 [12] 所选一样。

$$d_{\max} = 2 \quad d_{\min} = 0.2 \quad \gamma = 0.977 \quad \beta = 0.9 \quad \sigma_0 = 2, \\ e_{\max} = 1.1 \quad e_{\min} = 0.02 \quad k = 1.1 \quad k_{err} = 0.0015.$$

表 1 不同算法的结构和性能对比

Table 1 Contrast of the different algorithms in the structure and properties

	D-FNN	OLS	M-RAN	RANEKF
模糊规则数	6	7	8	13
可调参数数量	24	22	25	40
训练次数	200	7	200	200
均方根误差	0.0053	0.009	0.0437	0.0261

仿真结果如图 5 所示。表 1 列出了用均方根误差估计的结构和性能方面的对比。图 5a 所示, 经过 46 个训练数据, 规则数保持稳定。需要指出, 这并不意味着规则是保持不变的。实际上, 当新的样本数据加入后, 前提参数和结论参数一直保持更新, 而且, 一条新规则的产生可能会修剪掉已有的规则而同样保持系统的精确。在本例中, 当第 148 个数据达到时, 产生了一条新的规则, 这时, 原来已有的一条规则就显得不重要而被修剪掉。这在规则增长图中是看不出来的, 但在题 5c 中有反映。在这种情况下, 产生了一个陡峭的均方根误差。图 5b 显示训练时实际输出误差。

为了评估 D-FNN 产生的规则是否合理, 由 D-FNN 产生的规则分布示于图 6a 中。为了便于比较, 画出 OLS 产生的规则如图 6b 所示。由 D-FNN 和 OLS (正交最小二乘网络) 产生的 RBF 单元的具体参数列于表 2 中。

表 2 RBF 单元的分布

Table 2 RBF unit distribution

规则	D-FNN		OLS	
	中心	宽度	中心	宽度
1	0.5163	3.8971	-1.0229	1.2012
2	-1.5805	4.0864	1.5924	1.2012
3	2.7702	4.8312	-1.0191	1.2012
4	-3.8251	3.9764	2.0756	1.2012
5	1.6810	1.1980	-0.6863	1.2012
6	-0.4198	1.0298	2.3174	1.2012

由图 5d 及表 2 可以看出, 由 D-FNN 产生的规则分布在输出变化大的地方, 并根据变化的大小和快慢, 以不同的高斯宽度来实现, 从而使得可容纳的局部域达到最大。与之不同, OLS 方法中, 每个 RBF 的宽度是固定的, 当被逼近的函数在不同点出现缓慢不同变化时, 基函数的自适应性就差, 在变化剧烈的地方, 可能会要求有多个 RBF 产生。如本例中, 由 OLS 产生的 RBF 节点就有两个非常

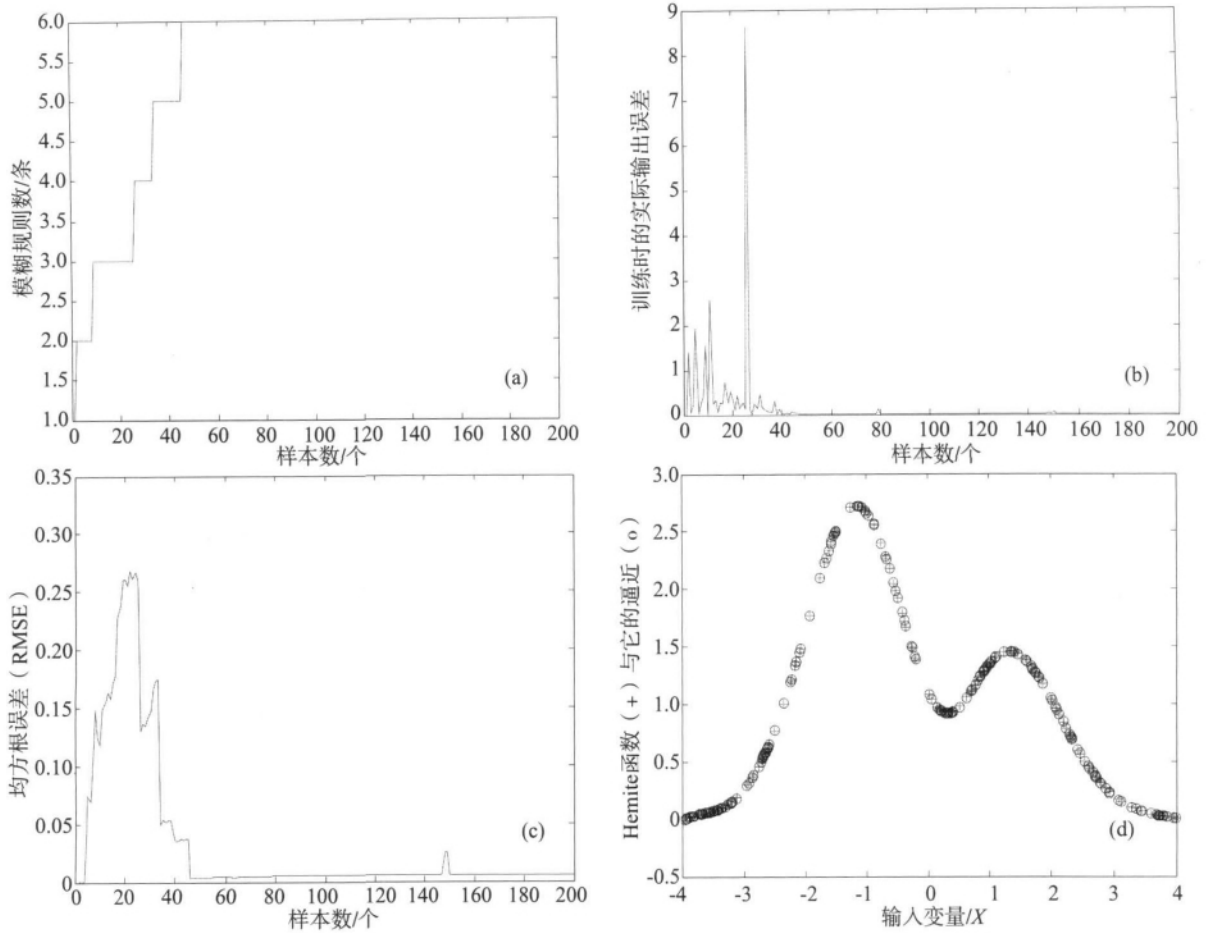


图 5 Hermite 函数逼近

Fig. 5 Hermite function approximation

(a) 模糊规则的产生; (b) 训练时实际输出误差; (c) 均方根误差; (d) Hermite 函数 (+) 与它的逼近 (o)

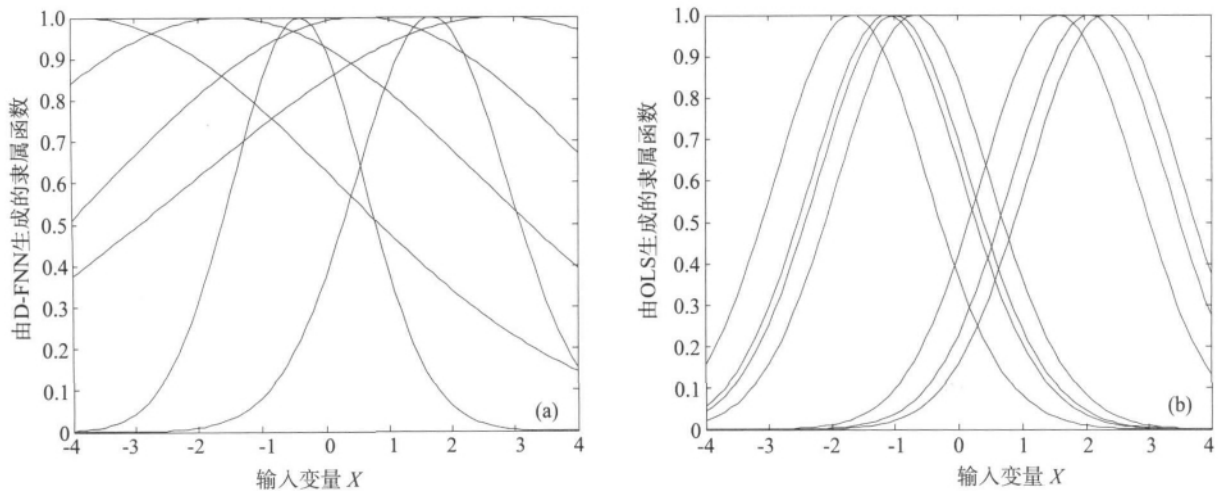


图 6 由 D-FNN 和 OLS 产生的规则分布对比

Fig. 6 Produce rules distribution contrast by D-FNN and OLS

(a) 由 D-FNN 生成的隶属函数; (b) 由 OLS 生成的隶属函数

接近的, 即 $C1 = -1.0229$ 及 $C3 = -1.0191$ 。所选高斯宽度越小, 这个问题越突出。同时, 也可以看到, OLS 方法确定的规则在整个输入变量空间中并不是均匀分布的, 所以, 所需规则数多, 即使如此, 精度还是不如 D-FNN 高。

3 结 论

修剪技术对于动态时变非线性系统的辨识逼近是非常必要的, 在学习进行时, 检测到不活跃的模糊规则并加以剔除, 所得到的模糊神经网络具有结构小的特点, 避免了出现过拟合的现象, 因而给使用者带来了很大的方便。实验研究表明选择高斯函数作为激活函数的新算法, 不仅具有强大的全局映射泛化能力, 而且在细化局部方面也有效。虽然高斯宽度在学习时可以自适应地调整, 但学习规则却很简单。仿真结果表明, 由于使用了高斯激活函数特征值分解的自适应系统使得 D-FNN 具有紧凑的系统结构、强大的泛化能力以及快速的学习速度。

参考文献:

- [1] CHEN S, COWAN C F N, GRANT P M. Orthogonal least squares algorithm for radial basis function network [J]. IEEE Trans. Neural Networks, 2010, 20(2): 302 - 310.
- [2] LEONTARITIS I J, BILLINGS S A. Input-output parametric models for nonlinear systems, Part 1: Deterministic nonlinear systems [J]. Int J Contr, 2009, 41(2): 303 - 344.
- [3] CHELLAPPA R, WILSON C L, SIROHEY S. Human and machine recognition of faces: A survey [J]. Proc IEEE, 2011, 93(2): 705 - 740.
- [4] BRUNELLI R, POGGIO T. Face recognition: features versus templates [J]. IEEE Trans Pattern Analysis and Machine Intelligence, 2011, 25(10), 1042 - 1053.
- [5] CHAO C T, CHEN Y J, TENG C C. Simplification of fuzzy neural systems using similarity analysis [J]. IEEE Trans Syst, Man, Cybern, Part B: Cybern, 2009, 35(2): 344 - 354.
- [6] WU S Q, ER M J. Dynamic fuzzy neural networks: a novel approach to function approximation [J]. IEEE Trans Syst, Man, Cybern Part B 2000, 30: 358 - 364.
- [7] ER M J, WU S Q. A fast learning algorithm for parsimonious fuzzy neural systems [J]. Fuzzy Sets and Systems, 2002, 126: 337 - 351.
- [8] 马莉, 张德丰, 马子龙. 滑动窗与修剪技术的动态模糊神经网络方法研究 [J]. 中山大学学报: 自然科学版, 2010, 49(1): 48 - 52.
- [9] Platt J. A resource-allocating network for function interpolation [J]. Neural Computation, 2008, 19(2): 213 - 225.
- [10] JANG J-S R. ANFIS: Adaptive-network-based fuzzy inference system [J]. IEEE Trans Syst, Man, Cybern, 1993, 23(3): 665 - 684.
- [11] HONG Z. Algebraic feature extraction of image for recognition [J]. Patt Recog, 2011, 24(2): 211 - 219.
- [12] LU Y, SUNDARARAJAN N, SARATCHANDRAN P. A sequential learning scheme for function approximation by using minimal radial basis function networks [J]. Neural Computation, 2007, 9: 461 - 478.
- [13] KADIRKAMANATHAN V, NIRANJAN M. A function estimation approach to sequential learning with neural networks [J]. Neural Computation 2003, 5: 954 - 975.
- [14] 任爱红. 模糊随机过程函数列均方差一致 Henstock 积分的可积性 [J]. 中山大学学报: 自然科学版, 2010, 51(4): 41 - 44.
- [15] TURK M A, PENTLAND A P. Eigenfaces for Recognition [J]. Cognitive Neuroscience, 2006, 3(1): 71 - 86.