

文章编号: 1001-0920(2001)03-277-05

一类新的 RBF 神经网络在非线性系统建模中的应用

刘妹琴, 沈 轶, 廖晓昕

(华中科技大学 控制科学与工程系, 湖北 武汉 430074)

摘 要: 提出一种基于输入集分类函数的新的距离度量方法, 它与前传回归的正交最小二乘法相结合, 不仅可以学习分类超平面的参数, 而且可以选择重要的输入节点。这种结构的 RBFNN 特别适用于非线性动力学系统的辨识(建模)和控制。将改进的 RBFNN 用于化工中的聚合反应过程建模, 结果表明该方法是有效而适用的。

关键词: 径向基函数神经网络(RBFNN); 分类超平面; 正交最小二乘法(OLS); 聚合反应过程
中图分类号: TP 183 文献标识码: A

Application of a Class of New RBF Neural Networks to Modelling Nonlinear Systems

L I U M ei-qin, S H E N Yi, L I A O X iao-xin

(Department of Control Science & Engineering, Huazhong University of Science & Technology, Wuhan 430074, China)

Abstract: A new distance metric based on a classification function of the set of input vectors is advanced. Combining the new measure with the forward regression orthogonal least square (OLS), not only the parameters of the classification hyperplane, but also the important input nodes can be obtained. The improved RBFNNs are suitable to identifying (modelling) or controlling nonlinear dynamic systems. Experimental results show that the improved RBFNN has successfully modelled the polymeric reaction process in the chemical industry.

Key words: radial basis function neural network (RBFNN); classification hyperplane; orthogonal least square (OLS); polymeric reaction process

1 引 言

径向基函数神经网络(RBFNN)以其诸多优点而广泛应用于模式识别、函数逼近、信号处理、系统建模和控制等领域。基本的 RBFNN 结构如图 1 所示。不失一般性, 假设输出层只有一个节点, 但是这种结构很容易扩展到多输出节点的情形。

网络输入与输出之间可认为是一种映射关系

$y(X): R^n \rightarrow R$ (n 为输入节点数), 即

$$y(X) = w_0 + \sum_{i=1}^{n_c} w_i \phi\left(X - C_i, \sigma_i\right) = w_0 + \sum_{i=1}^{n_c} w_i \exp\left[-\frac{\|X - C_i\|^2}{2\sigma_i^2}\right]$$

收稿日期: 2000-01-24; 修回日期: 2000-09-14

基金项目: 国家自然科学基金项目(69874016, 60074008)

作者简介: 刘妹琴(1972—), 女, 福建福州人, 博士后人员, 从事神经网络、信号处理等研究; 廖晓昕(1938—), 男, 湖南新化人, 教授, 博士生导师, 从事非线性控制系统、神经网络等研究。

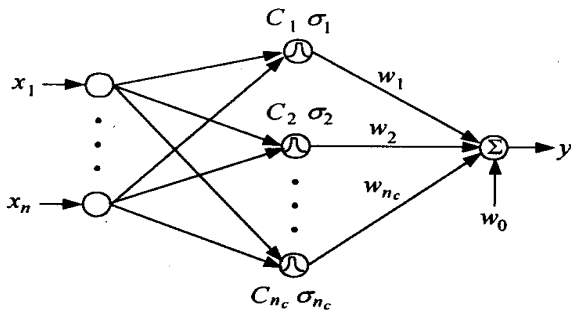


图1 RBFNN 结构

其中, n_c 是隐含层节点数, σ_i 是欧几里德范数, $X \in R^n$ 为输入向量, $C_i \in R^n$ 为第 i 个隐节点的中心, $\sigma_i \in R$ 为第 i 个隐节点的宽度, w_i 为第 i 个基函数与输出节点的连接权值, w_0 为调整输出的偏移量。由上式可知, 传统的 RBFNN 隐节点的响应是径向对称的。

在基于 RBFNN 的实际系统辨识器或控制器设计中, 系统过去的输入和输出作为 RBFNN 的输入, 这些输入和输出是自相关的; 而用欧几里德范数作为 RBF 的距离度量, 严格说只在输入不相关时才适用^[1]。这是因为输入的相关性导致 RBF 中心的相关性, 从而使基函数无法从输入集中抽取有效的信息, 这将使网络存在许多不必要的隐节点, 增加网络的复杂性, 而网络的泛化能力却很差。

传统的 RBFNN 用于系统辨识和控制, 其输入节点要根据具体问题或先验知识事先选定。如何决定过去哪些输入和输出作为神经网络的输入是相当重要的。有些输入变量与系统辨识或控制无关, 所以有可能使网络性能恶化^[2]; 有些输入变量是可有可无的, 这些变量会增加网络的复杂度并需要更多的计算时间, 但又不能改善网络的性能。Zheng 和 Billings^[3] 用互信息准则来选择输入节点, 但互信息的求得是较为困难且费时的。针对传统 RBFNN 的上述缺点, 本文采用基于分类函数的距离度量方法与传统学习方法相结合以克服其缺陷, 从而构成一类新的 RBFNN。

2 基于分类函数的距离度量方法

新的距离度量方法的基本思想源于判别式分析 (DA)。在判别式分析中, 能得到一种由类中所有变量的线性组合构成的分类函数, 它可用于决定新的向量是否落在类中^[1], 因此该分类函数可以实现多维向量到一维标量的映射, 并且包含数据集分布和判别式空间维数的信息。一个向量的分类函数值

表明该向量与数据集分布之间的相似性, 所以可用类似的方法来度量 RBFNN 中的向量距离。根据训练集中的输入向量构造一个分类函数, 其目的是为了找到一个多维输入向量空间中的分类超平面, 且输入向量到该超平面的均方距离最小。输入向量 $X(t)$ 的分类函数定义为

$$d[X(t)] = X(t)^T A \quad (1)$$

式中, $A = [a_1, a_2, \dots, a_n]^T \in R^n$ 为分类函数的加权向量。由此可得到分类超平面

$$d = X^T A \quad (2)$$

式中, d 为常数, $X \in R^n$ 表示超平面上的任何向量。式(2)说明落在超平面上的向量具有相同的分类函数值 d_0 。从几何意义上看, 超平面是由与 A 向量垂直、距离为 d/A 的所有向量构成, 如图 2 所示。其中, $X(t_0)$ 和 X 为超平面上的两个向量, n 为超平面单位法向量, $X(t)$ 为输入向量, $D_{X(t)}$ 为输入向量到超平面的距离, D_0 为原点到超平面的距离。

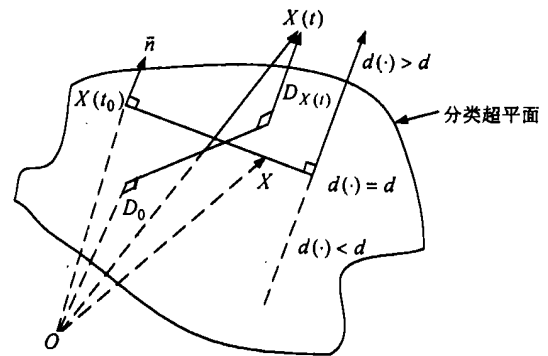


图2 输入向量分类函数的几何解释
式(2)的超平面方程可写成下列形式

$$[X - X(t_0)]^T n = 0, \quad n = A / \|A\| \quad (3)$$

将式(3)代入式(2), 得

$$X(t)^T n = d / \|A\| \quad (4)$$

在图 2 中, 原点到超平面的距离为

$$D_0 = X(t_0)^T n \quad (5)$$

从图 2 可以看出, 输入向量 $X(t)$ 到超平面的距离 $D_{X(t)}$ 与 D_0 有如下关系

$$X(t)^T n = D_0 + D_{X(t)} \quad (6)$$

将式(3)~(5)代入式(6), 得

$$D_{X(t)} = \frac{X(t)^T A - d}{\|A\|} = \frac{d[X(t)] - d}{\|A\|} \quad (7)$$

假设训练集中有两个自相关的输入向量 $X(t_1)$ 和 $X(t_2)$, 由于分类函数包含数据集分布信息, 所以它们与训练数据集分布的一致程度反映在相应的分

类函数值与常数 d 的偏差上(即它们到超平面的距离)。由此可知, $X(t_1)$ 与 $X(t_2)$ 之间的距离可以定义为它们各自到超平面的距离之差 $D_{X(t_1)} - D_{X(t_2)}$ 。如果分类函数的加权向量定义为单位向量, 即 $A = 1$, 则利用式(7) 可得

$$d[X(t_1)] - d[X(t_2)] \quad (8)$$

这样, 两个输入向量之间的距离就等于相应分类函数值之差。可以看出, 基于分类函数的距离度量方法使得隐节点的响应不再是径向对称的, 而是与 A 正交的超平面对称。

3 分类函数的学习以及 RBFNN 输入节点的选择

本节讨论用前传正交最小二乘法(OLS) 学习分类函数的加权向量 A 、分类函数值 d 和选择 n 个输入节点。由于所有训练集中的数据到超平面的均方距离最小, 又因一般由 RBFNN 辨识的系统有如下形式

$$y(t) = \text{RBFNN}[y(t-1), y(t-2), \dots, y(t-n_y), u(t-1), u(t-2), \dots, u(t-n_u)]$$

式中, $t = 1, 2, \dots, N$ 为采样时刻, n_y, n_u 为系统阶次。所以使下式

$$X(t) = [y(t-1), y(t-2), \dots, y(t-n_y), u(t-1), u(t-2), \dots, u(t-n_u)]^T$$

$$\frac{1}{N - n_y - n_u + 1} D_{X(t)}^2 = \frac{1}{N - n_y - n_u + 1} \frac{[X(t)^T A - d]^2}{A^2}$$

最小, 即可求得分类函数参数和对网络输出有重大贡献的输入节点。

为使上式最小, A 由 OLS 求得, 并将 A 单位化; d 的初始值可取任意非负常数, 并根据 A 规范化。这种方法求得的 A 和 d 是次优解, 因为它在每步最大化中由于输入节点的增加而引起 d 的增量, 而不是全过程优化, 但它可根据误差压缩比(ERR)^[4] 来选择重要的输入节点, 简化网络的结构。

考虑如下线性回归模型

$$D = XA + E \quad (9)$$

其中, $D = [d, \dots, d]^T$ R^{N-n_y} 表示分类函数值向

量, 而

$$X = \begin{bmatrix} y(n_y) & \dots & y(1) & u(n_y) & \dots & u(n_y + 1 - n_u) \\ y(n_y + 1) & \dots & y(2) & u(n_y + 1) & \dots & u(n_y + 2 - n_u) \\ \vdots & & \vdots & \vdots & & \vdots \\ y(N - 2) & \dots & y(N - n_y - 1) & u(N - 2) & \dots & u(N - n_u - 1) \\ y(N - 1) & \dots & y(N - n_y) & u(N - 1) & \dots & u(N - n_u) \end{bmatrix}$$

X $R^{(N-n_y) \times (n_y+n_u)}$ 称为回归矩阵。 X 表示成列向量的形式为

$$X = [X_1 \dots X_i \dots X_{n_y+n_u}]$$

$$X_i = [x_i(1) \ x_i(2) \ \dots \ x_i(N - n_y)]^T$$

X 表示成行向量的形式为

$$\begin{cases} X = [X(n_y + 1) \ X(n_y + 2) \ \dots \ X(N)]^T \\ X(t) = [x_1(t) \ x_2(t) \ \dots \ x_{n_y+n_u}(t)]^T \\ t = n_y + 1, n_y + 2, \dots, N \end{cases} \quad (10)$$

误差矩阵 $E = [e(n_y + 1), e(n_y + 2), \dots, e(N)]^T$, 假设它与回归矩阵不相关, 即 $X^T E = 0$ 。

将回归矩阵进行正交-三角分解, 有

$$X = Z\Gamma \quad (11)$$

式中, Γ 是一个 $(n_y + n_u) \times (n_y + n_u)$ 的单位上三角矩阵, 且

$$\Gamma = \begin{bmatrix} 1 & \gamma_{12} & \gamma_{13} & \dots & \dots & \gamma_{1, (n_y+n_u)} \\ 0 & 1 & \gamma_{23} & & & \gamma_{2, (n_y+n_u)} \\ 0 & & 1 & & & \vdots \\ \vdots & & & \ddots & & \vdots \\ \vdots & & & & 1 & \gamma_{(n_y+n_u-1), (n_y+n_u)} \\ 0 & 0 & 0 & \dots & 0 & 1 \end{bmatrix}$$

$R^{(n_y+n_u) \times (n_y+n_u)}$

Z 为 $(N - n_y) \times (n_y + n_u)$ 矩阵。 Z 表示成行向量的形式为

$$\begin{cases} Z = [Z(n_y + 1) \ \dots \ Z(n_y + i) \ \dots \ Z(N)]^T \\ Z(t) = [z_1(t) \ z_2(t) \ \dots \ z_{n_y+n_u}(t)]^T \\ t = n_y + 1, n_y + 2, \dots, N \end{cases} \quad (12)$$

Z 表示为列向量的形式为

$$Z = [Z_1 \ \dots \ Z_i \ \dots \ Z_{n_y+n_u}]$$

其各列 $Z_i (i = 1, 2, \dots, n_y + n_u)$ 正交, 且

$$Z^T Z = H = \text{diag}\{h_1, \dots, h_i, \dots, h_{(n_y+n_u)}\}$$

$$h_i = Z_i^T Z_i$$

由这组正交基向量 Z_i 张成的空间与 X_i 张成的空间相同, 故可将式(9) 写成 $D = ZQ + E$ 。正交 LS 解为

$$Q = (Z^T Z)^{-1} Z^T D = H^{-1} Z^T D$$

或

$$\begin{cases} Q = [q_1 \ \dots \ q_i \ \dots \ q_{(n_y+n_u)}]^T \\ q_i = Z_i^T D / (Z_i^T Z_i) \end{cases} \quad (13)$$

Q 和 A 满足如下三角方程组

$$\Gamma A = Q \tag{14}$$

式(14) 两边同除以 A, 得

$$\Gamma \frac{A}{A} = \frac{Q}{A} \tag{15}$$

将式(2) 两边同除以 A, 得

$$\frac{d}{A} = X^T \frac{A}{A} \tag{16}$$

$$\text{令 } \frac{A}{A} = A, \quad \frac{Q}{A} = Q, \quad d = \frac{d}{A}$$

这样, A 便为单位向量, Q 和 d 根据 A 规范化. 所以式(15) 和(16) 形式上与未作变化的式(14) 和(2) 相同.

将式(10) 和(12) 代入式(11), 得

$$X(t)^T = Z(t)^T \Gamma \\ t = n_y + 1, n_y + 2, \dots, N \tag{17}$$

由式(1), (14) 和(17) 可得输入向量的分类函数值为

$$d[X(t)] = X(t)^T A = Z(t)^T \Gamma A = Z(t)^T Q \tag{18}$$

由式(1) 和(14), 可得 RBFNN 中心向量 C_i 的分类函数值

$$d(C_i) = C_i^T A = C_i^T \Gamma^{-1} Q \tag{19}$$

$$\text{令 } B_i^T = C_i^T \Gamma^{-1} = [b_{i1} \quad b_{i2} \quad \dots \quad b_{i(n_y+n_u)}]$$

则式(19) 可重写为

$$d(C_i) = B_i^T Q \tag{20}$$

由式(8), (18) 和(20), 可得输入向量与第 i 个中心之间的距离为

$$d[X(t)] - d(C_i) = \\ Z(t) - B_i^T - Q = \\ \left\| \sum_{j=1}^{n_y+n_u} q_j(z_j(t) - b_{ij}) \right\| \tag{21}$$

由式(13) 知, 一般 $q_j = 1$. B_i 为 RBFNN 新的中心. 由式(21) 可知, RBFNN 隐节点的响应不再是径向对称的, 而在正交基 $Z_j (j = 1, 2, \dots, n_y + n_u)$ 上有不同的投影.

在实际问题中, 哪些过去输入和输出作为 RBFNN 的输入一般无法准确地确定. 为能充分反映系统的动态特性, n_y 和 n_u 取得较大, 这样 RBFNN 的结构便很复杂. 所以应从众多的过去输入和输出中选择对系统输出贡献最大的作为 RBFNN 的输入节点, 才能得到满意的网络结构. 对于 $i = j, Z_i$ 和 Z_j 正交, $X^T E = 0, D$ 的能量为

$$D^T D = \sum_{i=1}^{n_y+n_u} q_i^2 Z_i^T Z_i + E^T E$$

据此, 可以定义一个由 Z_i 带来的误差压缩比^[4]

$$ERR_i = \frac{q_i^2 Z_i^T Z_i}{D^T D}$$

ERR_i 值代表第 i 个正交基对于分类函数值的能量分布. 由于在每一正交基上的投影不同, 因此分类函数值的整体能量分布不是径向对称的. 通过减少不重要的正交基, 可以实现维数上的压缩, 所以实际上输入节点 $n = n_y + n_u$. OLS 的具体步骤参见文献[5].

4 RBFNN 隐节点的选择和参数的学习

由第 3 节选定 n 个输入节点后, Q 为下列形式

$$Q = [q_1 \quad q_2 \quad \dots \quad q_n]$$

而 RBFNN 可表示为如下形式

$$y(t) = w_0 + \sum_{i=1}^{n_c} w_i \Phi(v_i(t), \sigma_i) \\ v_i(t) = \left\| \sum_{j=1}^n q_j(z_j(t) - b_{ij}) \right\|$$

提供训练样本集 S_{tr} 和(或) 测试样本集 S_{te} 后, 可用前传回归方式的 OLS 算法或其它方法来确定 RBFNN 隐节点个数 n_c , 学习隐含层参数 B_i 和输出层权值 w_i ^[6].

5 用 RBFNN 进行非线性系统建模

化工中四氟乙烯的分散聚合反应是在带搅拌的夹套反应釜内进行的, 其聚合反应过程数学模型如下^[7]

$$\begin{aligned} \dot{x}_1 &= -\alpha_1 \exp\left[-\frac{\alpha_2}{273.15 + x_2}\right] x_1^{1.65} - \alpha_3 x_1 + \alpha_4 K_H P \\ \dot{x}_2 &= \alpha_5 \alpha_6 \exp\left[-\frac{\alpha_6}{273.15 + x_2}\right] x_1^{1.65} - \frac{\alpha_5}{V_R} \left\{ x_2 - \frac{x_3 + T_{j0}}{2} \right\} \\ \dot{x}_3 &= \alpha_6 \left\{ x_2 - \frac{x_3 + T_{j0}}{2} \right\} - \frac{1}{V_j} (x_3 - T_{j0}) u \\ y &= x_2 \end{aligned}$$

其中, x_1 为反应浓度, x_2 为反应温度, x_3 为夹套内冷却水出口温度, K_H 为亨利系数, P 为釜内气相压力, V_R 为反应体积, V_j 为夹套体积, T_{j0} 为冷却水进口温度, u 为冷却水流量即控制变量, $\alpha_1 \sim \alpha_6$ 为工艺参数.



由于上述系统本身存在严重的非线性, 使得建模相当困难, 也给系统的控制增加了难度。这里采用改进的 RBFNN 进行建模, 其结构如图 3 所示, 其中 θ 为系统的滞后。象这样一个高度非线性系统, 试图从整个空间辨识它的特性是不可能的。本文从实际应用角度出发, 对其额定工作点附近 (如 $\pm 10\%$) 进行建模。为了训练 RBFNN, 实际系统的输入由伪随机码、均匀分布的随机信号和正常工作状态下的输入信号三部分复合而成。在此输入下, 实际系统的输出充分反映了其在额定工作点附近的特性。

学习时从系统响应中采样 (采样周期为 15s), 得到 500 组输入 - 输出数据。系统参量初始值 $n_y = 5, n_u = 4, \theta = 1, d$ 的初始值取为 6, 选取初始中心点数 $n_c = 30$, 中心点的初始值取为 $[0, 13]$ 之间均匀分布的随机数。采用前传回归式的 OLS 学习 A, d, n, n_c, B_i 和 w_i , 输入节点 $n = 5$, 隐节点 $n_c = 20$ 。图 4 为实际系统 (实线)、改进的 RBFNN 模型 (虚线) 和传统的 RBFNN 模型 (点划线) 的阶跃响应。可以看出, 改进的 RBFNN 模型输出更接近于实际系统的输出, 其均方误差 $MSE = 0.006911$; 而传统的 RBFNN

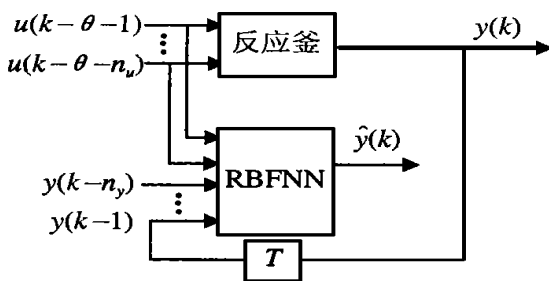


图 3 RBFNN 学习框图

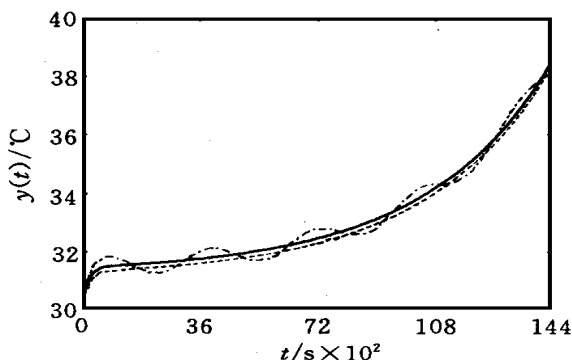


图 4 系统的阶跃响应 ($u = 0.415$)

模型的输出出现振荡, 这是由于其输入节点和隐节点都较多的缘故, 其输入节点 $n = 9$, 隐节点 $n_c = 25, MSE = 0.010812$ 。

6 结 论

本文提出一种基于分类函数的新的距离度量方法。在该度量方式下, 不但可以克服训练集中输入向量自相关带来的缺陷, 而且可用前传 OLS 方法来选择输入节点, 去除某些不必要的输入节点, 改善网络的性能。在新的距离度量方式下, RBFNN 隐节点的响应不再是径向对称的, 但输出层还是线性的。RBFNN 的学习可分为两个过程: 学习分类函数的参数, 选择 RBFNN 的输入节点; 选择 RBFNN 的隐节点个数, 学习参数。虽然前一过程计算较复杂, 增加了计算时间, 但是输入节点的减小可以简化网络结构, 加速第二过程的学习, 所以整体计算时间并未显著增加。本文提供的方法特别适用于非线性动力学系统的辨识 (建模) 和控制。

参考文献:

- [1] Kleinbaum D G, Kupper L L, Muller K E. Applied regression analysis and other multivariable methods [M]. Second Edition. Boston: PWS-KENT Publishing Company, 1987.
- [2] Billings S A, Jamaluddin H B, Chen S. Properties of neural networks with applications to modelling dynamic systems [J]. Int J of Control, 1992, 55(1): 193-224.
- [3] Zheng G L, Billings S A. Radial basis function network configuration using mutual information and the orthogonal least squares algorithm [J]. Neural Networks, 1996, 9(9): 1619-1637.
- [4] Billing S A, Chen S, Korenberg M J. Identification of MIMO nonlinear systems using a forward-regression orthogonal estimator [J]. Int J of Control, 1989, 49(6): 2157-2189.
- [5] 徐秉铮, 张百灵, 韦岗. 神经网络理论与应用 [M]. 广州: 华南理工大学出版社, 1994. 320-321.
- [6] Chen S, Cowan C F N, Grant P M. Orthogonal least squares learning algorithm for radial basis function networks [J]. IEEE Trans on Neural Networks, 1991, 2(2): 302-309.
- [7] 谢森, 王玉. 非线性系统完全线性化方法的研究 [J]. 控制理论与应用, 1997, 14(1): 139-143.