第 25 卷第 5 期
2008 年 10 月

控 制 理 论 与 应 用
Control Theory & Applications

Vol. 25 No. 5
Oct. 2008

文章编号: 1000−8152(2008)05−0827−04

基于快速回归算法的RBF神经网络及其应用

杜大军, 费敏锐, 李力雄

(上海大学 机电工程与自动化学院 上海市电站自动化技术重点实验室, 上海 200072)

摘要: 针对径向基神经网络(RBFNN)中存在的径向基函数中心的数目及其位置难以确定的问题, 提出了一种新型的基于快速回归算法(FRA)的RBFNN. 采用快速回归算法, 不但能够确定RBF的中心和中心个数, 而且能够求出隐含层到输出层的权重. 通过一元函数拟合和Mackey-Glass混沌时间序列预测的仿真, 验证了该网络的有效性与实用性.

关键词: 径向基神经网络(RBFNN); 快速回归算法; 正交最小二乘; 混沌时间序列

中图分类号: TP183　　　**文献标识码**: A

Radial-basis-function neural network based on fast recursive algorithm and its application

DU Da-jun, FEI Min-rui, LI Li-xiong

(Shanghai Key Laboratory of Power station Automation Technology, School of Mechatronical Engineering & Automation, Shanghai University, Shanghai 200072, China)

Abstract: Considering the difficulty in selecting the numbers and determining the locations of the centers of radial basis functions (RBF) in the RBF neural network (RBFNN), a novel RBFNN is proposed based on the fast recursive algorithm (FRA). Using FRA, we can determine the numbers and locations of the centers, and derive the weights between the hidden layer and the output layer. The new RBFNN is used to fit a single-variable function curve and predict the Mackey-Glass chaotic time series. The simulation results demonstrate the effectiveness and practicability.

Key words: radial basis function neural network(RBFNN); fast recursive algorithm(FRA); orthogonal least squares; chaotic time series

1 引言(Introduction)

RBF 神经网络是一种单隐层、前馈网络, 其具有网络结构简单、非线性逼近能力强、收敛速度快以及全局收敛等优点, 已广泛应用于模式识别、函数逼近、自适应滤波、非线性时间序列预测等方面[1~3].

RBF神经网络的非线性映射能力体现在隐层基函数上, 其特性主要由基函数的中心确定. 基函数中心的确定方法主要有以下几种: 正交最小二乘学习算法[2] (OLS) 是目前常用的一种基函数中心确定方法, 但存在运算量大和运算速度慢的缺点; 递推正交最小二乘算法[3] (ROLS)利用训练后正交矩阵中的信息, 采用后向选择方法, 逐步去掉那些使网络残差增加最小的中心, 与OLS相比, 在网络结构上有一定的简化; 自组织学习法通过自组织学习基函数中心位置, 根据输出是否参与聚类过程, 分为两种: 一种是输入聚类法, 常用的是k-均值聚类法[4], 另一种是输入输出聚类法[5], 采用聚类算法必须事先确定径向基函数中心个数, 由于同一种输入样本模式可能对应多个聚类, 往往通过聚类算法得到的隐节点数目仍很大, 从而产生由于隐节点数目过大而出现的过学习的问题; 进化优选算法[6] (ESA) 利用进化策略在解空间内对选择路径进行多点随机搜索, 以找到全局最优解, 但其存在计算量大, 精度不高的缺点.

快速回归算法[7,8] (FRA) 是一种不但能够选择模型结构而且能够估计模型参数的算法, 同OLS相比较, 不需要进行矩阵分解, 计算复杂度小, 数字稳定性更强, 已经被成功地应用于非线性模型的辨识. 因此, 本文将FRA引入到RBFNN中来选择RBF的中心和确定中心个数, 再结合FRA中模型参数估计算法来确定网络输出权值, 通过一元函数的拟合和混沌时间序列的预测, 来验证新型网络的有效性.

收稿日期: 2007−01−16; 收修改稿日期: 2007−12−22.
基金项目: 国家自然科学基金资助项目(60774059, 60834002); 上海市科委资助项目(061107031, 06ZR14131, 08XD14018); 上海市教委曙光计划跟踪项目(06GG10).

2 RBFNN 结构(RBFNN structure)

RBFNN在结构上通常由输入层、隐含层和输出层组成, 在网络特征上主要表现为输入层与隐含层单元之间采用直接映射, 而隐含层与输出层之间则采用加权线性求和的映射模式如图1所示[2]. 其完成如下非线性映射:

$$f_r(x) = w_0 + \sum_{i=1}^{n_r} w_i \phi(||x - c_i||), \tag{1}$$

其中: $x \in \mathbb{R}_n$是输入向量; $\phi(\cdot)$是一个$\mathbb{R}^+ \to \mathbb{R}$的径向基函数; w_i为权值; c_i为RBF的中心, $0 \leqslant i \leqslant n_r$; n_r为中心的个数.

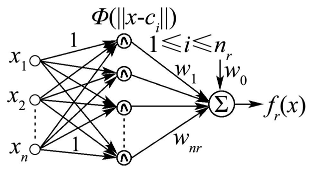

图 1 RBFNN 结构

Fig. 1 RBFNN structure

RBFNN中待定的参数有两类: 一类是基函数的中心c_i和中心数目n_r; 另一类为隐层与输出层之间的连接权重. 当基函数中心c_i以及n_r确定之后, 网络输出层权参数是线性的, 可以采用最小二乘算法、BP网络等求得. 因此c_i和n_r 的确定是建立RBFNN的关键.

3 基 于FRA的RBFNN学 习 算 法(RBFNN learning algorithm based on FRA)

3.1 RBFNN 模型(RBFNN model)

RBFNN从隐含层到输出层是一个线性变换过程, 将(1)看作是线性回归模型的一个特例[2], 则:

$$y(t) = \sum_{i=1}^{M} p_i(t)\theta_i + e(t), \tag{2}$$

其中: $y(t)$为期望输出; θ_i为模型参数(权值); $p_i(t)$为回归因子(隐含层输出); $e(t) = y(t) - f_r(x)$, 并且假设它与回归因子$p_i(t)$无关.

设$t = 1, \cdots, N$, 则式(2)可表示成

$$y = P\Theta + E, \tag{3}$$

其中:

$$y = [y(1), \cdots, y(N)]^{\mathrm{T}},\ P = [p_1, \cdots, p_M],$$
$$p_i = [p_i(1), \cdots, p_i(N)]^{\mathrm{T}}, 1 \leqslant i \leqslant M,$$
$$\Theta = [\theta_1, \cdots, \theta_M]^{\mathrm{T}},\ E = [e(1), \cdots, e(N)]^{\mathrm{T}}.$$

在进行基函数中心选择时, 回归因子p_i即作为候选项, 文献[2]中通过OLS算法把候选项p_i组成的集合P变换成一个正交基向量的集合W, 求出每个基向量对输出误差下降所作的贡献大小, 然后根据候选项p_i对网络的误差下降贡献来选择基函数的中心. 本文采用FRA[7,8]求出每个候选项p_i对价值函数的贡献大小, 然后根据贡献大小来选择候选项p_i作为基函数的中心.

3.2 快速回归算法(Fast recursive algorithm)

首先, 定义一个矩阵$\phi_k \in \mathbb{R}^{N\times k}$, 它包含式(3)中矩阵$P$的前$k$列, 则:

$$y = \phi_k \Theta_k + e, \tag{4}$$

其中:

$$y = [y(1), \cdots, y(N)]^{\mathrm{T}},\ \phi_k = [p_1, \cdots, p_k],$$
$$\Theta = [\theta_1, \cdots, \theta_k]^{\mathrm{T}},\ e = [e(1), \cdots, e(N)]^{\mathrm{T}}.$$

一个价值函数定义为:

$$J_k = \sum_{t=1}^{N} (y - \sum_{i=1}^{k} p_i\theta_i)^2 = (y - \phi_k\Theta_k)^{\mathrm{T}}(y - \phi_k\Theta_k). \tag{5}$$

如果矩阵ϕ_k 是一个满秩阵, 则式(4) 中回归系数的最小二乘估计为:

$$\hat{\Theta}_k = (\phi_k^{\mathrm{T}}\phi_k)^{-1}\phi_k^{\mathrm{T}} y. \tag{6}$$

为了快速地选择候选项, 定义矩阵$M_k \triangleq \phi_k^{\mathrm{T}}\phi_k$, 则式(6) 化简为:

$$\hat{\Theta}_k = (M_k)^{-1}\phi_k^{\mathrm{T}} y. \tag{7}$$

由式(5) (7) 可得:

$$J_k = y^{\mathrm{T}}y - \hat{\Theta}_k\phi_k^{\mathrm{T}} y. \tag{8}$$

定义一个递归阵$R_k \in \mathbb{R}^{N\times N}$, $k = 1, \cdots, M$,

$$R_k \triangleq I - \phi_k M_k^{-1}\phi_k^{\mathrm{T}}, \tag{9}$$

其中$\phi_k, k = 1, \cdots, M$是一个满秩阵, 并且$R_0 \triangleq I$.

矩阵R_k是一个递归阵, 假设矩阵P中的$p_i, i = 1, \cdots, M$是线性无关的, 则其具有下列性质[7,8]:

1) $$R_{k+1} = R_k - \frac{R_k p_{k+1} p_{k+1}^{\mathrm{T}} R_k^{\mathrm{T}}}{p_{k+1}^{\mathrm{T}} R_k p_{k+1}}, k = 0, 1, \cdots, M-1, \tag{10}$$

2) $$R_k^{\mathrm{T}} = R_k, (R_k)^2 = R_k, \tag{11}$$

3) $$R_k R_j = R_j R_k = R_k, k \geqslant j, \tag{12}$$

4) $$R_k P_i = 0, \forall i \in \{1, \cdots, k\}. \tag{13}$$

把式(9) 代入式(8) 得:

$$J_k = y^{\mathrm{T}}y - \hat{\Theta}_k\phi_k^{\mathrm{T}} y = y^{\mathrm{T}} R_k y, \tag{14}$$

由式(10)(14) 可得:

$$\begin{cases} J_{k+1} = y^{\mathrm{T}} R_{k+1} y = J_k - \dfrac{y^{\mathrm{T}} R_k p_{k+1} p_{k+1}^{\mathrm{T}} R_k^{\mathrm{T}} y}{p_{k+1}^{\mathrm{T}} R_k p_{k+1}}, \\ J_0 = y^{\mathrm{T}} y, \end{cases} \tag{15}$$

此外, 定义:

$$p_i^{(k)} \triangleq R_k p_i, p_i^{(0)} \triangleq R_0 p_i = p_i,$$
$$i = 1, \cdots, M, k = 0, 1, \cdots, M. \tag{16}$$

考虑式(11)和(16), 式(15) 变为:

$$\delta J_{k+1} = -(y^{\mathrm{T}} p_{k+1}^{(k)})^2 / ((p_{k+1}^{(k)})^{\mathrm{T}} p_{k+1}^{(k)}),$$
$$k = 0, \cdots, M-1. \tag{17}$$

方程(17) 展示了候选项p_{k+1} 对价值函数的贡献. 为了进一步简化计算, 定义两个新的变量:

$$\begin{cases} a_{k,i} \triangleq (p_k^{(k-1)})^{\mathrm{T}} p_i^{k-1}, a_{1,i} \triangleq p_1^{\mathrm{T}} p_i, \\ a_{k,y} \triangleq (p_k^{(k-1)})^{\mathrm{T}} y, a_{1,y} \triangleq p_1^{\mathrm{T}} y, \\ i = k, \cdots, M, k = 1, \cdots, M. \end{cases} \tag{18}$$

应用R_k的性质和式(16)中定义的$p_i^{(k)}$, 式(18)化简为:

$$\begin{cases} a_{k,i} = p_k^{\mathrm{T}} p_i - \sum\limits_{j=1}^{k-1} (a_{j,k} a_{j,i}) / a_{j,j}, \\ a_{k,y} = p_k^{\mathrm{T}} y - \sum\limits_{j=1}^{k-1} (a_{j,k} a_{j,y}) / a_{j,j}, \end{cases} \tag{19}$$

应用式(19), 可得:

$$\begin{cases} y^{\mathrm{T}} p_i^{(k)} = y^{\mathrm{T}} p_i - \sum\limits_{j=1}^{k} (a_{j,y} a_{j,i}) / a_{j,j}, \\ (p_i^{(k)})^{\mathrm{T}} p_i^{(k)} = (p_i)^{\mathrm{T}} p_i - \sum\limits_{j=1}^{k} (a_{j,i}^2) / a_{j,j}, \end{cases} \tag{20}$$

最后, 把式(20)代入式(17), 候选项p_{i+1}, $k = 0, \cdots, M-1$对价值函数的贡献能够明确地表示为:

$$\delta J_{k+1} = -\frac{(y^{\mathrm{T}} P_{k+1} - \sum\limits_{j=1}^{k} (a_{j,y} a_{j,k+1} / a_{j,j}))^2}{(P_{k+1})^{\mathrm{T}} P_{k+1} - \sum\limits_{j=1}^{k} (a_{j,k+1}^2 / a_{j,j})}. \tag{21}$$

根据式(21)计算出所有项的δJ_{k+1}, 并按从大到小的顺序排成一列, 然后从队列的第一个开始选取基函数中心, 直到输出误差小于事先给定的网络训练精度ρ, ρ的选取可参考文献[2]. 基函数中心选取的终止条件也可根据FPE或AIC 准则[9]来确定.

3.3 RBFNN权值的求取(Computation of the weights of RBFNN)

在RBF网络的中心确定以后, 即需求解隐层与输出层之间的连接权值. 首先, 在方程(7)两边同乘ϕ_k可得:

$$\phi_k \hat{\Theta}_k = \phi_k (M_k)^{-1} \phi_k^{\mathrm{T}} y = y - R_k y. \tag{22}$$

其次, 在(22) 两边同乘$(p_j^{(j-1)})^{\mathrm{T}}$并利用(13) 可得:

$$\sum_{i=j}^{k} \hat{\theta}_i (p_j^{(j-1)})^{\mathrm{T}} p_i = (p_j^{(j-1)})^{\mathrm{T}} y; j = 1, \cdots, k. \tag{23}$$

最后, 利用式(19), 式(23)可化简为:

$$\hat{\theta}_j = (a_{j,y} - \sum_{i=j+1}^{k} \hat{\theta}_i a_{j,i}) / a_{j,j}, j = k, \cdots, 1. \tag{24}$$

通过(24)可求出输出层与隐层之间的连接权值.

3.4 基于FRA的RBFNN的实现(Implementation of RBFNN based on FRA)

基于FRA的RBFNN的实现方法归纳如下：

1) 采集训练样本;

2) 利用FRA求出每个候选项对价值函数的δE, 并按从大到小的顺序排成一列;

3) 从队列的第一个开始选取基函数中心, 直到输出误差小于事先给定的网络训练精度ρ, 此时选取的基函数中心即构成RBF神经网络的中心并且确定中心的个数;

4) 利用步骤3)求出的基函数中心, 再根据(24) 计算输出层与隐层之间的连接权值.

4 仿真(Simulation)

4.1 用径向基神经网络作一元函数的拟合(Fitting one element function curve by RBFNN)

$$y = 0.4 + \sin c(4x) + 1.1 \sin c(4x+2) +$$
$$0.8 \sin c(6x-2) + 0.7 \sin c(6x-4), \tag{25}$$

式中$\sin c(x) = \begin{cases} 1, & x = 0, \\ \dfrac{\sin(\pi x)}{\pi x}, & x \neq 0. \end{cases}$

式(25)是一个典型的多峰函数, 选取步长为0.1, 训练样本81个, 网络的误差精度$\rho = 0.02$, 分别采用OLS算法[2]、k-mean聚类算法[4]和FRA训练RBF网络, 训练结果见表1. 从中可以看出, FRA的隐单元数比OLS法少, 并且训练速度远比OLS快; 若选择k-mean聚类算法和FRA的隐单元数一样, k-mean聚类算法产生的误差远大于FRA法的误差.

表 1　训练结果比较

Table 1　Comparison of training results

	训练时间/s	隐单元数目/个	误差平方和
OLS法	5.1570	56	0.0163
FRA法	1.6090	53	0.0161
k-mean法	2.1160	53	0.4808

4.2 Mackey-Glass混沌时间序列(Mackey-Glass chaotic time series)

选用Mackey-Glass混沌模型[10]

$$\frac{\mathrm{d}x(t)}{\mathrm{d}t} = \frac{ax(t-\tau)}{1 + x^{10}(t-\tau)} - bx(t), \tag{26}$$

其中: τ为时滞参数, $\tau > 17$时呈现混沌性, τ值越大, 混沌程度越高. 参数选取: $\tau = 17$, $a = 0.2$,

$b = 0.1$. 为了获取每一积分时间点的时间序列值, 用四阶龙格－库塔法寻找(26)的数值解, 时间步长为0.1 s, 这样经过对数值积分, 就得到时间序列$x(t)$. 在$t = 1$和$t = 50$之间以及$t = 150$和$t = 199$之间采集100对输入输出数据作为训练样本, 即$[x(t-4), x(t-3), x(t-2), x(t-1); x(t)]$, 用于训练集RBFNN; 在$t = 1501$和$t = 7500$之间采集6000对输入输出数据用来验证模型的准确性. 利用训练样本, 网络的误差精度$\rho = 0.02$, 分别采用OLS算法[2]、k-mean聚类算法[4]和FRA训练RBF网络, 并利用训练好的RBF神经网络来预测校验样本的输出, 网络训练和预测结果见表2. 从中可以看出, FRA算法训练速度最快, 并且预测误差平方和最小. 图2展示了系统的期望输出和3种算法的预测值曲线, 3种算法模型各点的预测误差Error分布如图3所示, 从图2和图3可以看出, 基于FRA的RBF神经网络预测精度最高.

5 结论(Conclusion)

本文采用FRA来设计一种新型的RBF神经网络, 不但能够确定FRA网络的中心和网络中心数, 而且能够求出隐含层到输出层的权重. 将其运用到一元函数的拟合和Mackey-Glass混沌时间序列的预测, 通过比较, 结果表明了该新型网络的快速性和有效性. 因此, 本文提出的基于FRA的RBF神经网络, 是研究混沌时间序列预测和复杂非线性动力系统辨识、控制的一种有效方法.

表 2 网络训练和预测结果比较

Table 2 Comparison of network training and predictive results

	OLS法	FRA	k-mean聚类法
隐单元数目/个	3	4	10
训练时间/s	15.8440	14.3430	15.2030
预测误差平方和	7.4159	5.695	7.6906

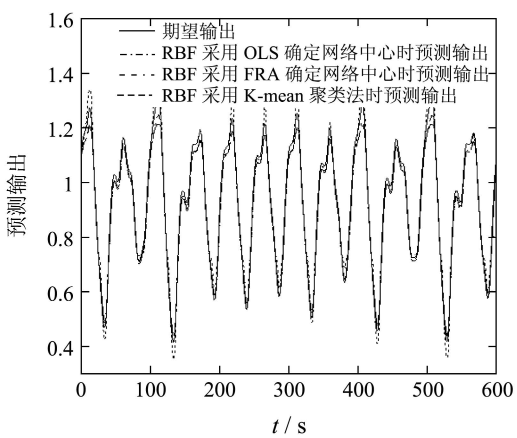

图 2 Mackey-Glass时间序列期望值和预测值

Fig. 2 The desire value and predictive value of Mackey-Glass chaotic time series

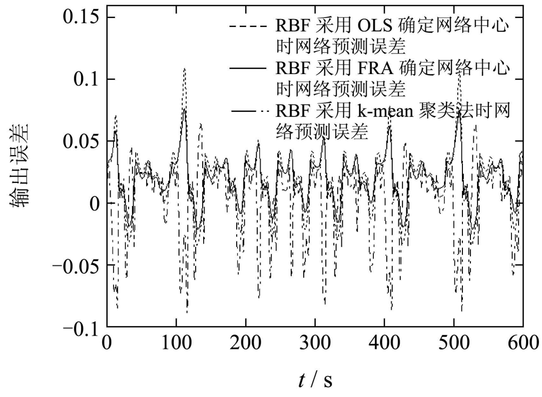

图 3 Mackey-Glass模型的预测误差(Error分布)

Fig. 3 Predictive errors of Mackey-Glass chaotic time series

参考文献(References):

[1] HO S L, FEI M R, FU W N, et al. Integrated RBF network based estimation strategy of the output characteristics of brushless DC motors[J]. *IEEE Transactions on Magnetics*, 2002, 38(2): 1033 – 1036.

[2] CHEN S. Orthogonal least square learning algorithm for radial basis function networks[J]. *IEEE Transactions on Neural Networks*, 1992, 2(2): 302 – 309.

[3] YU D L, YU D W, GOMM J B, et al. Adaptive RBF model for model-based control[C] //*Proceedings of the 5th World Congress on Intelligent Control and Automation*. China: IEEE Press, 2004: 78 – 82.

[4] XU L, KRZYZAK A, YUILLE A. On radial basis function nets and kernel regression: statistical consistency, convergence rates, and receptive field size[J]. *Neural Networks*, 1994, 7(4): 609 – 628.

[5] UYKAN Z, GUZELIS C, CELEBI M E, et al. Analysis of input-output clustering for determining centers of RBFN[J]. *IEEE Transactions on Neural Networks*, 2000, 11(4): 851 – 858.

[6] 魏海坤, 徐嗣鑫, 宋文忠. RBF网学习的进化优选算法[J]. 控制理论与应用, 2000, 17(4): 604 – 608.
(WEI Haikun, XU Sixin, SONG Wenzhong. An evolutionary selecting algorithm for the learning of RBF nets[J]. *Control Theory & Applications*, 2000, 17(4): 604 – 608.)

[7] LI K, PENG J X, IRWIN G W. A fast nonlinear model identification method[J]. *IEEE Transactions on Automatic Control*, 2005, 50(8): 1211 – 1216.

[8] LI K, PENG J X, BAI E W. A two-stage algorithm for identification of nonlinear dynamic systems[J]. *Automatica*, 2006, 42(7): 1189 – 1197.

[9] BURSHTEIN D, WEINSTEIN E. Some Relations between the various criteria for autoregressive model order determination[J]. *IEEE Transactions on Acoustics, Speech, and Signal Processing*, 1985, 33(4): 1017 – 1019.

[10] LO J C, YANG C H. Heuristic error-feedback learning algorithm for fuzzy modeling[J]. *IEEE Transactions on Systems, Man, and Cybernetics Part A: Systems and Humans*, 1999, 29(6): 686 – 691.

作者简介:

杜大军 (1978—), 男, 上海大学自动化系博士研究生, 目前研究方向为神经网络、网络学习控制等, E-mail: ddj559@163.com;

费敏锐 (1961—), 男, 上海大学自动化系教授, 目前研究方向为智能控制、网络学习控制以及无线传感网络等;

李力雄 (1975—), 男, 上海大学自动化系讲师, 目前研究方向为自适应控制、网络学习控制以及基于Web的远程监控等.