

自构造 RBF 神经网络及其参数优化

兰天鸽, 方勇华, 熊伟, 孔超

(中国科学院安徽光学精密机械研究所遥感研究室, 合肥 230031)

摘要: 径向基函数神经网络的构造需要确定每个 RBF 的中心、宽度和数目。该文利用改进的聚类算法自动构造 RBFN, 考虑样本的类别属性, 根据样本分布自动计算 RBF 的中心和宽度, 并确定 RBF 的数目。所有的网络参数采用非线性优化算法来优化。通过 IRIS 分类问题和混沌时间序列预测评价自构建 RBFN 的性能, 验证参数优化效果。结果表明, 自构造 RBFN 不但能够自动确定网络结构, 而且具有良好的模式分类和函数逼近能力。通过对网络参数的非线性优化, 该算法明显改善了网络性能。

关键词: 径向基函数; 自构造网络; 参数优化; 模式识别; 混沌时间序列

Self-growing RBF Neural Networks and Parameters Optimization

LAN Tian-ge, FANG Yong-hua, XIONG Wei, KONG Chao

(Remote Sensing Lab, Anhui Institute of Optics and Fine Mechanics, Chinese Academy of Sciences, Hefei 230031)

【Abstract】 Construction of Radial Basis Function Neural(RBFN) networks involves computation of centers, widths of each RBF, and number of RBF in the middle layer. The modified clustering algorithm is used to construct RBFN automatically. The algorithm considers the class membership of training samples, can automatically compute RBF centers and widths, and determines the number of Radial Basis Function(RBF) units based on the distribution of samples. Parameters are optimized with nonlinear optimization technique. The performance of the self-growing RBFN and effects of the optimization are estimated with IRIS classification problem and chaotic time series prediction. The results confirm that self-growing RBFN determines networks structure automatically, and has good performance in pattern recognition and function approximation. Better performance can be observed after nonlinear optimization of networks parameters.

【Key words】 Radial Basis Function(RBF); self-growing networks; parameter optimization; pattern recognition; chaotic time series

1 概述

近年来, RBFN 在模式识别、系统建模、信号处理等^[1-2]领域得到了广泛的应用。RBFN 是 3 层前馈式网络, 其中, 输入层和输出层由线性神经元组成, 中间层节点为 RBF 单元, RBF 对输入矢量产生局部响应, 输出节点对 RBF 的输出进行线性加权, 从而实现输入空间到输出空间的映射, 使整个网络达到分类或函数逼近的目的。RBFN 的输出可用式(1)表达:

$$\hat{y}_i = \sum_{k=1}^p w_{ik} R_k(x) \quad (1)$$

其中, x 是输入向量; \hat{y}_i 为输出层第 i 个输出, $i=1, 2, \dots, n$, n 是输出向量的维数; w_{ik} 为第 k 个 RBF 与输出层第 i 个节点的连接权值; $k=1, 2, \dots, p$, p 为 RBF 的个数; R_k 为第 k 个 RBF, 通常采用高斯函数:

$$R_k(x) = \exp\left(-\frac{\|x - c_k\|^2}{\sigma_k^2}\right) \quad (2)$$

其中, c_k 为第 k 个高斯函数的中心向量; σ_k 为与第 k 个高斯函数的宽度有关的变量; $\|\cdot\|$ 为 2 范数, 表示 x 与 c_k 的欧拉距离。RBF 的最大输出为 1, 输入与中心的向量距离越接近, 则 RBF 的输出越大, 对于给定的输入 x 只有靠近中心部分才被激活。

RBF 的数目、RBF 中心及其宽度是 RBFN 的主要控制参数。如果 RBF 数目太大, 尽管有可能使 RBFN 收敛到满意的精度, 但会使输入数据过拟合, 降低网络的泛化能力, 而且

网络结构过于庞大, 造成训练和测试时间过长。相反, 如果 RBF 数目太小, 网络则可能无法收敛。

K-均值聚类法是最常用的选择 RBF 中心和宽度的算法。但是, 该算法必须先主观地凭经验确定 RBF 的数目; 没有考虑训练样本的类别属性, 在不同类别样本的重叠区域可能导致错误聚类。而且这样的网络并不能在任何时候达到满意的效果, 所以, 在结果不够理想时, 就需要对网络的参数进行优化, 本文利用改进的聚类算法来自动构造 RBFN。

2 RBFN 构造及网络参数优化

2.1 RBFN 自构造算法

RBFN 自构造算法(self-growing algorithm)^[2-3]是根据训练样本分布计算 RBF 的中心和宽度, 并确定 RBF 的数目。不同类别的训练样本分别进行考虑。对每一个样本, 计算它到同类已有 RBF 中心的最小距离, 如果超出一定距离则增加新的 RBF 单元。这个距离由参数 λ 控制(步骤(3))。假设共有 ν 类样本, 则

(1) 对于 ν 类样本, 计算每类的均值向量 \bar{x} 和平均距离 \bar{d} 。

(2) 初始化 RBF 单元数目为 ν , 每个单元的中心为 \bar{x} , 宽

作者简介: 兰天鸽(1981—), 男, 博士研究生, 主研方向: 数字信号处理, 模式识别, 神经网络; 方勇华, 研究员、博士生导师; 熊伟, 博士研究生; 孔超, 硕士研究生

收稿日期: 2007-06-30 **E-mail:** l3t2g1@163.com

度为平均距离的倍数 $\eta \bar{d}$ ， η 可以调整。

(3)对训练集中的样本 x_i ，找出 x_i 与它同类别各 RBF 中心的最小距离 d_{\min} ；如果 d_{\min} 小于 $\lambda \bar{d}$ ，则用几何平均的方法修正该距离所对应的 RBF 单元的中心和宽度：

$$c^{m+1} = \frac{t \cdot c^m + x_i}{t+1} \quad (3)$$

$$d^{m+1} = \frac{t \cdot d^m + d_{\min}}{t+1} \quad (4)$$

其中， t 为该参数被修正的次数；否则，增加一个新的 RBF 单元，中心为样本 x_i ；这样，确定了 RBF 的数目、中心以及宽度以后，就可以计算中间层和输出层之间的连接权值了。式(1)改写为矩阵形式： $Y = W \times R$ ，所以，权值矩阵 W 就可以用伪逆法计算

$$W = Y \times R^+ = Y \times (R^T R)^{-1} R^T \quad (5)$$

对于函数逼近问题只要令 $\nu = 1$ 即可。

2.2 网络参数优化

RBFN 的构造和参数学习已经完成。但这样的网络并不能在任何时候达到满意的效果，所以，就需要对网络的参数进行优化。把已有的网络参数作为初始值，用非线性优化方法对其进行调整。

定义 RBFN 的性能指标为误差平方和，即

$$E = \frac{1}{2} \sum_{i=1}^n \sum_{j=1}^{ns} (y_{ij} - \hat{y}_{ij})^2 \quad (6)$$

其中， y_{ij} 和 \hat{y}_{ij} 分别为第 j 个样本的第 i 个输出的期望输出和实际输出； ns 为样本的个数； n 为输出向量的维数。若将所有的参数，即网络的 RBF 中心 c 、宽度 σ 、连接权 w ，构成一个参数集合 $U = \{c, w, \sigma\}$ ，将性能指标作为参数优化的目标函数，则

$$U^{(r+1)} = U^{(r)} - \eta \nabla E(U^{(r)}) \quad (7)$$

其中， ∇E 为目标函数 E 下降最快的梯度方向； η 则是参数调整的步长。

对于应用高斯函数作为 RBF 单元的网络，RBF 中心 c 、宽度 σ 、连接权 w 的学习可以依据：

$$\nabla w_{i,k} = \sum_{j=1}^{ns} R_k(x^j) (y_i^j - \hat{y}_i^j) \quad (8)$$

$$\nabla c_{\mu,k} = \sum_{j=1}^{ns} R_k(x^j) \frac{x_{\mu}^j - c_{\mu,k}}{\sigma_k^2} \sum_{i=1}^n w_{i,k} (y_i^j - \hat{y}_i^j) \quad (9)$$

$$\nabla \sigma_k = \sum_{j=1}^{ns} R_k(x^j) \frac{(x_{\mu}^j - c_{\mu,k})^2}{\sigma_k^3} \sum_{i=1}^n w_{i,k} (y_i^j - \hat{y}_i^j) \quad (10)$$

选择适当的参数修正步长或学习速率 η 在神经网络训练中是相当重要。如果 η 选择不合适，就可能使网络收敛得很慢，或者陷入局部最小。在梯度下降法中引入了可变学习步长，以保证目标函数的收敛趋势。关于可变学习步长的计算，可以参考文献[4-5]，这里不再赘述。

3 应用与讨论

用 IRIS 分类问题和混沌时间序列预测对自构建 RBFN 的性能进行评价以及对参数优化效果进行验证。

3.1 IRIS 分类问题

IRIS 是一个经典的模式识别问题^[6]，对 3 种花型(setosa, versicolor, virginica)进行分类，按照花瓣长、宽，花萼长、宽采样产生了 150 个 4 维样本。将其平分为训练集和预测集，用训练集构造网络并用非线性优化技术优化这些网络参数，然后用预测集对网络性能评价。网络期望输出为 0, 1 编码的

向量，对应某类的分量为 1，其余分量为 0。对于给定的样本，只有当网络的实际输出中类别属性对应的分量大于 0.5，而其他分量都小于 0.5 时，才认为该样本被正确分类。

表 1 是取不同 λ 值时，RBF 数目以及训练方均根误差(RMSE)、错分样本数目(NMCS)和预测方均根误差(RMSEP)、错分样本数目(NMCS)，其中，参数 $\eta = 3$ 。可以看到， $\lambda = 1$ 时，有 38 个 RBF 单元，RMSE 很小，只有 0.0816。错分样本数为 0。但是预测集 RMSEP 却较大，错分样本数为 4。这是过拟合的表现，说明 RBF 的数目太多。比较 $\lambda = 1.5$ 时，RBF 的数目是前者的 1/2，只有 19 个，虽然 RMSE 增大了，但是 NMCS 仍为 0；预测集 RMSEP 则减小，NMCS 也只有 1 个。随着 λ 的增大，RBF 数目不断地减少，训练精度和预测精度也随之下降。 $\lambda = 4.0$ 时，仅有 3 个 RBF，样本只对应一个 RBF。尽管 RMSEP 要比 RMSE 稍小，但预测错分数目却有 7 个。图 1 是 $\lambda = 4.0$ 时，网络参数的优化过程，RMSE 随迭代次数的变化。经过对网络参数的迭代优化以后，RMSE 和 RMSEP 分别为 0.1339 和 0.1583，而训练样本错分数为 0，预测错分数为 2。说明通过网络参数优化，明显改善了网络性能，可以用十分精简的网络结构实现分类功能。

表 1 不同的 λ 对模式识别网络($\eta = 3$)的影响

λ	RBFs	训练方均根误差	错分样本数目	预测方均根误差	错分样本数目
1.0	38	0.0816	0	0.20667	4
1.5	19	0.1262	0	0.15370	1
2.0	9	0.1401	2	0.15790	4
3.0	5	0.1449	2	0.15930	4
4.0	3	0.1913	4	0.19010	7

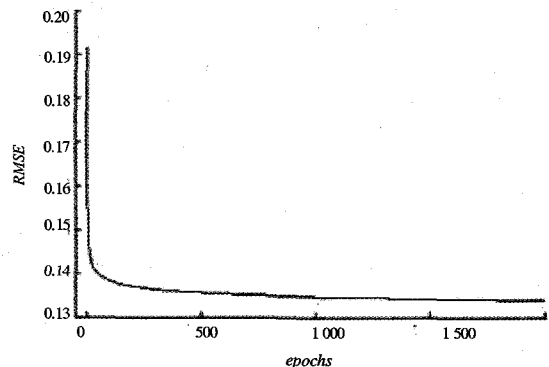


图 1 模式识别网络($\lambda = 4.0$)参数优化曲线

在图 2 中，RMSE 在最初迅速下降，500 次迭代以后，就没有太多改善，这是因为受到 RBF 数目的限制，所以只有增加 RBF 才能得到改善。

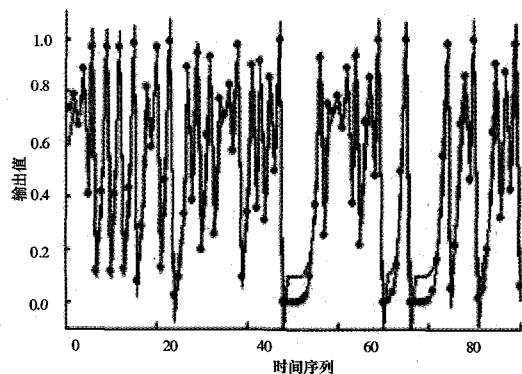


图 2 对混沌时间序列模型的预测结果 1

3.2 混沌时间序列预测

这里用一个混沌时间序列来检验所提算法的性能^[3]。该

混沌模型定义为

$$y(t+1) = 4y(t)(1-y(t))$$

其中, $y(0)$ 取值在 0 和 1 之间。 $y(0)=0.4$ 由上式产生 200 个样本, 前 100 个样本用于训练, 其余样本用于模型验证。在这里令样本类别 $\nu=1$ 。

λ 取不同值时, 表 2 显示了对应的 RBF 数目以及训练方均根误差(RMSE)和预测方均根误差(RMSEP), 其中, $\eta=3$ 。从表中可以看到, 随着 λ 的增大, RBF 的数目不断减少, 相应的训练精度和预测精度也随之下降。 $\lambda=1$ 时, RBF 的数目为 16, 而 $\lambda=2$ 时 RBF 单元数目则降为 1/4, 相应的训练误差和预测误差增大了 20 倍左右。图 2 是 $\lambda=2$ 时的期望输出和实际输出, 可以看出两者差异明显。训练和预测结果都不能令人满意。图 3 是 $\lambda=2$ 时, 网络参数的优化过程。可以看到, RMSE 在最初迅速下降, 经过 110 次迭代达到 $\lambda=1.7$ 时的训练精度, 经过 290 次迭代达到 $\lambda=1.3$ 时的训练精度, 经过 439 次迭代就已经达到 $\lambda=1.0$ 时的训练精度。在 2 000 次迭代结束时, 训练方均根误差为 0.001 07, 预测方均根误差为 0.001 05。图 4 是 $\lambda=2$ 时的网络参数经过优化以后的期望输出和实际输出, 可以看出两者差异很微小, 预测效果很好。这说明经过网络参数优化, 可以明显提高网络性能, 简化网络结构。 $\lambda=2$, 对混沌时间序列模型的预测结果见图 2; 经过参数优化的网络后, 混沌时间序列模型的预测结果见图 4。其中, 实线表示预测输出, 点线表示期望输出。

表 2 不同 λ 对混沌时间序列预测网络($\eta=3$)的影响

λ	RBFs	训练方均根误差	预测方均根误差
1.0	16	0.003 65	0.003 16
1.3	11	0.004 14	0.004 35
1.7	8	0.007 34	0.006 98
2.0	4	0.071 34	0.073 32

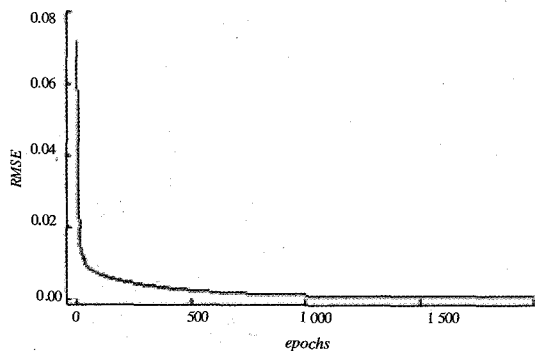


图 3 混沌时间序列预测网络参数优化曲线($\lambda=2$)

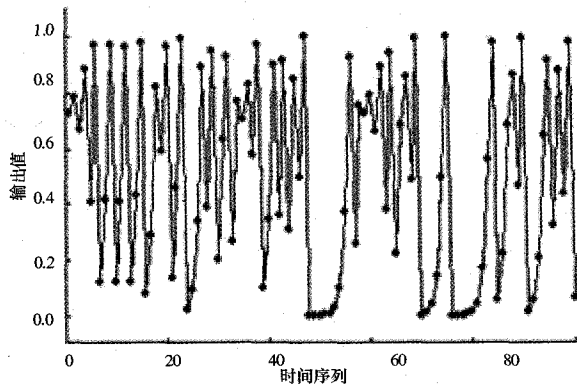


图 4 对混沌时间序列模型的预测结果 2

4 结束语

本文利用改进的聚类算法来确定 RBFN 的结构。该算法考虑了样本的类别属性, 根据样本分布自动计算 RBF 中心和宽度, 并确定 RBF 的数目; 中间层到输出层的连接权则用伪逆法计算; 把已有的网络参数作为初始值, 用迭代非线性优化算法来优化所有的网络参数。结果表明, 自构造 RBFN 不但能够自动确定网络结构, 而且具有良好的模式分类和函数逼近能力; 通过对网络参数的非线性优化, 明显提高了网络性能, 说明可以用更精简的网络结构来完成模式识别和函数逼近任务。

参考文献

- [1] 刘 昆, 颜钢锋. 基于模糊 RBF 神经网络的函数逼近[J]. 计算机工程, 2001, 27(2): 70-71.
- [2] Wan Chuanhao. Self-configuring Radial Basis Function Neural Networks for Chemical Pattern Recognition[J]. Journal Chem. Inf. Comput. Sci., 1999, 39(6): 1049-1056.
- [3] Choi S W, Lee D, Park J H. Nonlinear Regression Using RBFN with Linear Submodels[J]. Chemometrics and Intelligent Laboratory Systems, 2003, 65(2): 191-208.
- [4] Schwenker F, Kestler H A, Palm G. Three Learning Phases for Radial-basis-function Networks[J]. Neural Networks, 2001, 14(4/5): 439-458.
- [5] Magoulas G D, Vrahatis M N, Androulakis G S. Effective Backpropagation Training with Variable Stepsize[J]. Neural Networks, 1997, 10(1): 69-82.
- [6] 赵温波, 黄德双. 全结构遗传优化径向基神经网络[J]. 红外与毫米波学报, 2004, 23(2): 113-118.

(上接第 199 页)

参考文献

- [1] Kaufman L, Rousseeuw P J. Finding Groups in Data: An Introduction to Cluster Analysis[M]. New York, USA: John Wiley & Sons, 1990.
- [2] Wang Kaijun. Cluster Validation Toolbox for Estimating the Number of Clusters[Z]. [2007-05-01]. <http://www.mathworks.com/matlabcentral/fileexchange/loadAuthor.do?objectType=author&objectId=1095267>.
- [3] Dimitriadou E, Dolnicar S, Weingessel A. An Examination of Indexes for Determining the Number of Cluster in Binary Data Sets[J]. Psychometrika, 2002, 67(1): 137-160.
- [4] Ben-Hur A, Elisseeff A, Guyon I A. Stability Based Method for Discovering Structure in Clustered Data[C]//Proc. of Pacific Symposium on Biocomputing. Hawaii, USA: [s. n.], 2002.
- [5] Wang Kaijun. Estimating the Number of Clusters via System Evolution for Cluster Analysis of Gene Expression Data[D]. Xi'an: Xidian University, 2007.
- [6] Medvedovic M, Yeung K Y, Bumgarner R E. Bayesian Mixture Model Based Clustering of Replicated Microarray Data[J]. Bioinformatics, 2004, 20(8): 1222-1232.