

文章编号: 1001-0920(2007)10-1091-06

粗糙集意义下的一种 RBF 神经网络设计方法

王耀南¹, 张东波^{1,2}, 黄辉先², 易灵芝²

(1. 湖南大学 电气与信息工程学院, 长沙 410082; 2. 湘潭大学 信息工程学院, 湖南 湘潭 411105)

摘要: 提出一种集成粗糙集理论的 RBF 网络设计方法. 由布尔逻辑推理方法进行属性离散化, 得到初始决策模式集, 通过差异度对初始决策模式的相似度进行衡量并实现聚类, 以聚类决策模式构造 RBF 网络. 为加快训练速度, 分别对隐层参数和输出权值采用 BP 算法和线性最小二乘滤波法进行训练. 实验结果表明, 该方法设计的 RBF 网络结构简洁, 泛化性能良好, 混合学习算法的收敛速度优于单纯的 BP 算法.

关键词: 粗糙集; RBF 神经网络; 聚类; 模式识别

中图分类号: TP391.4 **文献标识码:** A

A method of designing RBF neural network based on rough sets

WANG Yaonan¹, ZHANG Dongbo^{1,2}, HUANG Hui-xian², YI Ling-zhi²

(1. College of Electrical and Information Engineering, Hu 'nan University, Changsha 410082, China; 2. Institute of Information Engineering, Xiangtan University, Xiangtan 411105, China. Correspondent: ZHANG Dong-bo, E-mail: zhadonbo@yahoo.com.cn)

Abstract: A method of designing RBF neural network, which integrates rough sets theory, is proposed. Continuous attributes are discretized by using Boolean reasoning algorithm and original decision modes are extracted. Similarities among original decision modes can be measured by dissimilarity degree and original decision modes can be clustered. Clustered decision modes are used to construct RBF neural network. To increase the training speed, a hybrid training algorithm is introduced, in which the parameters of hidden layer and weights of output layer are tuned by using back propagation algorithm and linear least squares filtering, respectively. Experiment results show that the designed RBF neural network has refined structure and good generalization ability. The convergence speed of hybrid training algorithm is superior to the single back propagation algorithm.

Key words: Rough sets; Radial basis function neural network; Clustering; Pattern recognition

1 引言

径向基函数(RBF)网络作为一种单隐层前馈神经网络,具有结构简单、非线性逼近能力强、收敛速度快、全局收敛等优点,在模式识别、函数逼近、信号处理、系统建模等领域得到广泛的应用^[1-3]. RBF 网络的结构设计主要集中在如何选取适当的隐层神经元数、径向基函数的中心向量和宽度等参数上. 通常,隐层神经元数越多,网络训练精度越高,但往往会产生过拟合,导致网络泛化能力降低,同时不可避免地造成大量的结构冗余. 径向基函数的中心和宽度决定基函数的位置和形状,合适的参数与 RBF 网络的泛化能力有直接关系.

RBF 网络的设计主要采用聚类方法和正则化方法^[4]. 由于网络的隐层对输入样本有聚类作用, RBF 网络通常利用聚类算法进行设计,常用的有 C-均值^[5]和 Kohonen^[6]聚类算法. 聚类数决定隐层神经元数,各聚类均值可作为基函数中心向量,类内类间距离可为基函数宽度选取提供依据. 尽管基于聚类的方法可设计出合适的 RBF 网络,但聚类算法的实现往往需要预先给定聚类数,或者控制聚类结果的参数或准则,这通常由实验确定或专家给定,带有一定的盲目性和主观性. 在输入特征空间进行的聚类,往往仅考虑样本在输入特征空间的相似性,而没有考虑样本本身的输出类别信息,生成的聚类不能

收稿日期: 2006-06-28; 修回日期: 2006-10-12.

基金项目: 国家自然科学基金项目(60775047); 湖南省自然科学基金项目(06JJ5112); 湖南省教育厅科研基金项目(05C093).

作者简介: 王耀南(1957—),男,昆明人,教授,博士生导师,从事智能控制、图像处理等研究; 张东波(1973—),男,湖南邵阳人,讲师,博士生,从事粗糙集、神经网络等研究.

完全反映输入输出之间的映射关系。

粗糙集理论^[7]作为处理不确定、不完整、不精确知识的有力工具,具有强大的知识约简和定性分析能力,利用其知识约简能力,可获得分类性能不变的精简的粗糙集数据模型;利用其定性分析能力,可获得输入与输出决策之间的映射关系描述。因此本文提出一种基于不可分辨划分的有监督粗糙集聚类算法,并以此为基础进行RBF网络的设计。

2 基于粗糙集划分的监督聚类算法

2.1 属性的离散化和初始决策模式的提取

为进行粗糙集数据分析,首先必须对连续属性进行离散化。常用的离散化方法有等间隔法、等频法^[8]、布尔逻辑推理^[9]、聚类方法^[10]等。

等间隔法和等频法需要预先给定分割的区间数,在实际中往往比较困难。合适的区间数通常需要经过反复实验才能找到,如果区间数设置不当,则在离散决策表中容易产生不一致决策规则。布尔逻辑推理在保证分类能力不变的前提下,可找到最少数目的分割点集,实现精细的不可分辨等价类划分,生成的离散决策表不存在不一致决策规则,其缺点是计算量大。聚类方法通过聚类的方式来寻找合适的离散区间分割点集,在只考虑输入条件属性相似性的条件下,仍难以避免不一致决策规则的生成。

人们希望得到精细的分类性能有保障的初始决策模式(每条决策规则都可看作一种决策模式),因此本文选择基于布尔逻辑推理的离散化方法。

属性离散化后,根据离散分割点集生成决策表,通过合适的属性约简算法^[11]找到保证分类性能不变的最小属性集,由约简属性构成的约简决策表提取初始决策模式集。

2.2 有监督的初始决策模式集的聚类

在构造RBF网络时,为保证分类性能,构建隐层神经元的同一决策模式类的对象应尽可能具有相同的输出类别,而相似性大的对象应尽可能划分到同一模式类,这样既可降低冗余,又能提高系统的推广泛化能力。由于采用了基于布尔逻辑推理的离散化方法,得到的初始决策模式类已是相容的一致决策模式类集,但由布尔逻辑推理得到的离散区间比较精细,容易造成初始决策模式类规模过大,含有较大的冗余。如果能对决策模式的相似性进行衡量,并根据相似性和输出类别信息,在不降低整体分类性能的前提下,将具有相同输出类别的相似性大的决策模式进行聚类,则可极大地减少系统冗余。

2.2.1 初始决策模式的差异度计算

对离散决策表提取得到的决策模式是一种符号向量形式,不适合直接采用欧氏距离进行相似性度

量。本文定义一种差异度来衡量模式之间的相似度,差异度既考虑到模式之间的距离,又考虑到决策模式每个属性对分类的重要度,给予不同的权重。在对象的值约简^[7]处理中,每个对象可能有一个或多个属性是不必要的,而对整个论域的值约简进行统计,出现频率越高的属性对分类越重要,因此可以此为基础定义属性的权值。

若 $z_i (i = 1, 2, \dots, n)$ 是得到的初始决策模式, a_k 是第 k 个属性, v_i 是模式 z_i 的值约简, $v_i(a_k)$ 是属性 a_k 关于值约简 v_i 的取值, l 是所有初始决策模式类的值约简数,则属性 a_k 的权值定义为

$$w_k = 1 - \frac{v_i(a_k)}{l}, \quad (1)$$

其中

$$v_i(a_k) = \begin{cases} 1, & a_k \text{ 是 } v_i \text{ 中必要的;} \\ 0, & a_k \text{ 是 } v_i \text{ 中不必要的。} \end{cases}$$

在获得各属性的权值后,模式之间的差异度定义为

$$d(z_i, z_j) = \sum_{k=1}^m (w_k \times \frac{k}{ij}), \quad (2)$$

其中

$$\frac{k}{ij} = \begin{cases} 0, & a_k(z_i) = a_k(z_j); \\ \text{abs}(a_k(z_i) - a_k(z_j)), & a_k(z_i) \neq a_k(z_j). \end{cases}$$

式中: $a_k(z_i)$ 是模式 z_i 的属性 a_k 取值,abs是绝对值运算。根据定义,差异度满足

$$d(z_i, z_j) = d(z_j, z_i) \quad 0, d(z_i, z_i) = 0.$$

2.2.2 基于差异度的等价关系及相似模式类划分

给定差异度阈值 TH_d ,可以定义基于差异度的等价关系 $\{R_i\}, i = 1, 2, \dots, n$,将初始决策模式集划分为相似模式类 $[z_i]_{R_i}$ 和非相似模式类 $\overline{[z_i]_{R_i}}$,即

$$U/R_i = \{[z_i]_{R_i}, \overline{[z_i]_{R_i}}\}, i = 1, 2, \dots, n. \quad (3)$$

其中

$$\begin{aligned} [z_i]_{R_i} &= \{z_j \mid d(z_i, z_j) \leq TH_d\}, \\ \overline{[z_i]_{R_i}} &= \{z_j \mid d(z_i, z_j) > TH_d\}, \\ j &= 1, 2, \dots, n. \end{aligned} \quad (4)$$

2.2.3 差异度阈值的选取方法

差异度阈值是决定初始决策模式聚类质量的重要参数。根据定义,差异度阈值应是一个大于等于零的实数。差异度阈值为零时,依据等价关系 $\{R_i\}$,划分的相似模式类 $[z_i]_{R_i}$ 只包含模式 z_i 。随着差异度阈值增大,相似模式类 $[z_i]_{R_i}$ 包含的相似模式也越多。基于不可分辨关系 $\text{ind}(R) = \prod_i R_i$ 进行模式聚类,能被聚成一类的模式增多,但当差异度阈值超过一定的上限时,具有不同输出类别的初始模式将被聚成一类。这显然会增大决策难度,降低分类性能。因此在设置差异度阈值时,应在保证分类性能不

变的前提下,尽可能选择大的差异度阈值.基于不可分辨关系 $\text{ind}(R) = \prod_i R_i$,初始决策模式的聚类可由下式计算:

$$[z]_{\text{ind}(R)} = \bigcap_{R_i \in R} [z]_{R_i}. \quad (5)$$

假设由式(5)得到的模式聚类为 $\{Y_1, Y_2, \dots, Y_m\}$, $m \leq n$,每个 Y_j 可能由一个或几个 z_i 组成,衡量聚类后的决策模式分类能力的指标定义为

$$= \left| \bigcup_{X \in U/D} \{Y_j \mid Y_j \subseteq X\} \right| / n. \quad (6)$$

其中: U/D 是决策属性 D 对论域 U 的等价划分, n 是初始决策模式数目, $| \cdot |$ 表示能被唯一确定分类类别的模式在全部决策模式中的比例.

本文采用基于布尔逻辑推理的离散化方法,所形成的初始决策模式 $\{z_1, z_2, \dots, z_n\}$ 是相容的一致决策模式集,因此初始决策模式的分类能力指标为

$$= \left| \bigcup_{X \in U/D} \{z_i \mid z_i \subseteq X\} \right| / n = 1. \quad (7)$$

搜索差异度阈值上限的算法如下:

- 1) 参数初始化, $\text{TH}_d = 0$, 设置合适的搜索步长 Step (通常在 0.01 ~ 0.03 之间取值);
- 2) $\text{TH}_d = \text{TH}_d + \text{Step}$, 由式(3) ~ (5) 求取模式聚类;
- 3) 由式(6) 计算当前分类性能指标, 若 $\geq \theta$, 则输出当前 TH_d , 否则转 2).

2.2.4 基于不可分辨等价关系的有监督模式聚类

在模式聚类的处理过程中,采用的是一种有监督的模式聚类.它既考虑了模式间的相似性,又考虑了模式的输出类别信息.即一方面要使相似性大(差异度小)的具有相同输出类别的模式尽可能形成聚类,另一方面要根据输出类别信息避免不同输出类别的模式被聚为一类.因此,在确定了差异度阈值后,只需根据差异度定义的不可分辨等价关系,对相同输出类别的初始决策模式进行聚类.在这种情况下,基于差异度的等价关系 $\{R_i\}$, 对初始决策模式集的划分变为

$$U/R_i = \{ [z_i]_{R_i}, [z_i]_{R_i} \}, i = 1, 2, \dots, n. \quad (8)$$

其中

$$\begin{cases} [z_i]_{R_i} = \{ z_j \mid d(z_i, z_j) \leq \text{TH}_d, \\ \text{dec}(z_j) = \text{dec}(z_i) \}, \\ [z_i]_{R_i} = \bigcap_i z_i - [z_i]_{R_i}. \end{cases} \quad (9)$$

式中 $\text{dec}(z_i)$ 表示模式 z_i 的输出类别.以全部等价关系 $\{R_i\}$ 的交可构成不可分辨关系 $\text{ind}(R)$, 对初始决策模式的聚类划分为

$$[z]_{\text{ind}(R)} = \bigcap_{R_i \in R} [z]_{R_i}. \quad (10)$$

3 粗糙集意义下的 RBF 神经网络设计与训练

3.1 粗糙集意义下的 RBF 神经网络结构

在完成初始决策模式类的聚类后,可得到最终的决策模式集,每个决策模式都代表一定数量的样本集,表示输入特征属性空间不可分的相似聚类.如果每个决策模式用一个隐层神经元来表示,则在粗糙集意义下的 RBF 神经网络如图 1 所示.

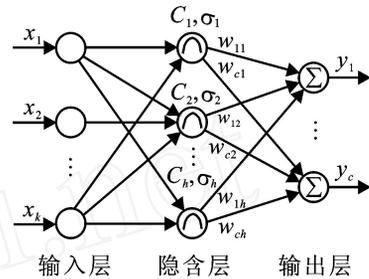


图 1 粗糙集意义下的 RBF 神经网络结构

图 1 中, x_1, x_2, \dots, x_k 是输入特征属性, y_1, y_2, \dots, y_c 是输出类别, 隐含层的每个神经元对应一个决策模式, 激励函数取高斯径向基函数

$$R_i(x) = \exp(-\frac{\|x - C_i\|^2}{2\sigma_i^2}), \quad i = 1, 2, \dots, h. \quad (11)$$

其中: h 为隐含层神经元的数目, $x = (x_1, x_2, \dots, x_k)$ 是 k 维输入向量, C_i 和 σ_i 分别表示第 i 个隐层神经元基函数的中心向量和宽度, $\|x - C_i\|$ 表示输入向量 x 与中心 C_i 之间的距离. RBF 神经网络的输出为

$$y_j = \sum_{i=1}^h w_{ji} R_i(x), \quad i = 1, 2, \dots, h, j = 1, 2, \dots, c. \quad (12)$$

3.2 RBF 神经网络参数初始化

为加快 RBF 神经网络的训练速度,对其参数进行合适的初始化是非常必要的. RBF 网络线性输出权值 w_{ji} 初始化为 $[0, 1]$ 区间的随机值, 隐层神经元的中心由对应聚类模式所代表的训练样本的均值进行估算. 隐层基函数的宽度与 RBF 网络的泛化能力有直接关系, 宽度的选择既要考虑类内样本分散程度, 又要考虑类间距离. 类内样本分散程度大或类间距离大, 基函数宽度应大些, 反之应小些. 在 RBF 网络中, 基函数的局部响应特性就是对同一类样本产生尽可能大的输出, 对其他类别的样本则产生尽可能小的输出.

基于上述考虑, 给出初始化基函数宽度 σ_i 的方法^[12] 如下:

- 1) 计算各聚类模式的类内分散程度 $d_i^{\text{inner}} = \sqrt{S_i^2}, S_i^2 = \frac{1}{|Y_i|} \sum_{x \in Y_i} \|x - C_i\|^2$ 表示第 i 个聚类模

式的样本到类中心的平均距离;

2) 计算类间距离,若 $d_{ij} = |C_i - C_j| = \min_{l \neq m} |C_l - C_m|$, 则称 C_j 为 C_i 的最近邻聚类;

3) 若 C_j 为 C_i 的最近邻聚类, 则第 i 个聚类对应的隐层节点的基函数宽度初始化为 $\sigma_i = d_{ij} - d_{ij}^{inner}$.

3.3 RBF 神经网络的训练算法

基于聚类方法设计 RBF 神经网络时, 网络的训练通常分成两步混合学习算法来实现^[13,14]: 第 1 步为自组织学习阶段, 根据 C 均值聚类或其他聚类算法, 找到隐层节点的中心和宽度参数; 第 2 步为监督学习阶段, 根据最小二乘法或最小均方算法 (LMS)^[4] 估计输出层权值. 上述方法的不足之处在于: 一是普通的自组织聚类学习对初始值和样本输入次序具有敏感性; 二是采用监督方法训练输出权值的过程中, 隐层神经元的中心和宽度参数固定不变, 这不利于提高网络的泛化能力, 也不能充分反映模式分类的动态自适应过程. 文献[15]指出, 广义 RBF 网络的中心位置、基函数宽度和权值均采用监督学习, 可在一定程度上克服过学习, 明显提高网络的泛化能力.

对 RBF 网络采取的学习过程可作如下分析: 网络输出单元相连的线性权值与隐层单元的非线性基函数相比, 是在不同时间尺度上的更新, 当隐层基函数根据某种非线性优化策略缓慢更新时, 输出权值根据线性最优策略进行快速调整. 重要的是, RBF 网络的不同层起着不同的作用, 因此对隐层和输出层采用不同的优化策略是合理的^[16]. 基于以上考虑, 为提高 RBF 网络的泛化能力及训练速度, 本文提出一种对隐层中心和宽度采用 BP 算法, 而对输出层权值采用线性最小二乘滤波方法^[4], 从而进行动态调整的 RBF 网络学习算法.

径向基函数的中心向量 C_i 和宽度参数 σ_i , 均由反向传播 BP 算法进行训练, 取误差性能指标函数

$$J = \frac{1}{2} \sum_{n=1}^N \sum_{j=1}^c (d_j(n) - y_j(n))^2. \quad (13)$$

其中: $d_j(n)$ 和 $y_j(n)$ 分别表示第 j 个输出神经元第 n 个样本的期望输出和实际输出, N 为训练样本数, c 为输出类别数.

计算误差性能指标函数对 σ_i 和 c_{il} ($i = 1, 2, \dots, h, l = 1, 2, \dots, k$) 的误差偏导数

$$\begin{aligned} \frac{\partial J}{\partial \sigma_i} = & \sum_{n=1}^N \sum_{j=1}^c \frac{\partial J}{\partial y_j(n)} \frac{\partial y_j(n)}{\partial R_i(x(n))} \frac{\partial R_i(x(n))}{\partial \sigma_i} = \\ & \sum_{n=1}^N \sum_{j=1}^c [- (d_j(n) - y_j(n))] w_{ji} \times \\ & \exp[- (x(n) - C_i)^2 / 2 \sigma_i^2] \times \end{aligned}$$

$$\begin{aligned} & (- (x(n) - C_i)^2 / \sigma_i^3)], \quad (14) \\ \frac{\partial J}{\partial c_{il}} = & \sum_{n=1}^N \sum_{j=1}^c \frac{\partial J}{\partial y_j(n)} \frac{\partial y_j(n)}{\partial R_i(x(n))} \frac{\partial R_i(x(n))}{\partial c_{il}} = \\ & \sum_{n=1}^N \sum_{j=1}^c [- (d_j(n) - y_j(n))] w_{ji} \times \\ & \exp[- (x(n) - C_i)^2 / 2 \sigma_i^2] (x_l(n) - c_{il}) / \sigma_i^2]. \quad (15) \end{aligned}$$

沿梯度下降方向, σ_i 和 c_{il} ($i = 1, 2, \dots, h, l = 1, 2, \dots, k$) 的训练算法如下:

$$\sigma_i(t+1) = \sigma_i(t) - \eta_1 \frac{\partial J}{\partial \sigma_i}, \quad (16)$$

$$c_{il}(t+1) = c_{il}(t) - \eta_2 \frac{\partial J}{\partial c_{il}}. \quad (17)$$

其中: t 表示迭代次数, η_1 和 η_2 分别为 σ_i 和 c_{il} 的学习速率.

输出层权值采用线性最小二乘滤波方法进行, 具体推导过程可参见文献[4]. 每个输出神经元的权值表示成向量的形式, 即

$$W_j = [w_{j1}, w_{j2}, \dots, w_{jh}]^T, 1 \leq j \leq c. \quad (18)$$

全部 N 个样本在 j 个输出神经元上形成的误差向量及期望响应向量为

$$E_j = [e_j(1), e_j(2), \dots, e_j(N)]^T, 1 \leq j \leq c; \quad (19)$$

$$D_j = [d_j(1), d_j(2), \dots, d_j(N)]^T, 1 \leq j \leq c. \quad (20)$$

隐层输出 R 为 $N \times h$ 的矩阵

$$\begin{cases} R = [R_1, R_2, \dots, R_h], \\ R_i = [R_i(1), R_i(2), \dots, R_i(N)]^T. \end{cases} \quad (21)$$

根据 Gauss-Newton 法, 权值的计算公式为

$$W_j = (R^T R)^{-1} R^T D_j, 1 \leq j \leq c. \quad (22)$$

综合起来, RBF 神经网络的训练算法如下:

- 1) 分别对 C_i , σ_i 和输出权值 w_{ji} 进行合适的初始化, 给定期望误差 ϵ , 学习速率 η_1 和 η_2 ;
- 2) 根据当前 C_i 和 σ_i 计算隐层输出矩阵 R ;
- 3) 根据式(14)和(15)分别计算 σ_i 和 c_{il} ($i = 1, 2, \dots, h, l = 1, 2, \dots, k$) 的误差偏导数;
- 4) 由式(16)和(17)调整隐层中心向量 C_i 和宽度参数 σ_i ;
- 5) 根据式(22)计算输出层权值;
- 6) 如果实际误差小于期望误差 ϵ , 则训练结束, 否则转 2) 继续训练.

4 实验结果与分析

4.1 实验数据

在 UCI 机器学习数据库中选择 iris 数据样本, 这是一个广泛使用的鸢尾属花的分类问题. 刚毛鸢尾花、变色鸢尾花和弗吉尼亚鸢尾花这 3 种花, 由花

萼长、花萼宽、花瓣长和花瓣宽 4 个输入变量来描述,样本总数 150 个,每类样本均为 50 个.选择其中 70% 作为训练样本,其余 30% 作为测试样本.

从 UCI 机器学习数据库中选择 wine 数据样本,该数据样本是一个酒类识别数据.3 种不同品种葡萄酒酿造的葡萄酒由 13 种化学成分的含量进行描述,样本总数 178 个.随机选取其中 89 个作为训练样本,其余 89 个作为测试样本.

另外,还选择一个相对比较复杂的非线性分类问题.文献[17]中有一个两类别人工合成的具有随机分布特性的样本集 rand,但是样本之间的分界面比较复杂,是一个典型的非线性可分样本集.其中训练样本 200 个,测试样本 1 000 个,测试样本远多于训练样本.该样本数据可以测试分类器的泛化能力.

4.2 模式分类实验

为验证本文方法的性能,将本文方法与广泛采用的 RBF 神经网络设计方法进行比较.该方法由 Matlab 神经网络工具箱的函数 newrb 来实现,其调用格式为

$$net = newrb(P, T, GOAL, SPREAD, MN, DF).$$

其中: P 是 $R \times Q$ 输入矩阵, R 是输入变量数, Q 是训练样本数; T 是 $S \times Q$ 目标矩阵, S 是输出类别数; $GOAL$ 是期望均方误差和; $SPREAD$ 是隐层基函数宽度; MN 是隐层最大允许神经元数,应不超过样本数; DF 是迭代过程中每次增加的隐层神经元数,一般取 1.

训练过程中,当误差小于期望误差或隐层神经元数达到最大允许值时,训练结束.在设计过程中,隐层基函数宽度 $SPREAD$ 需要多次实验才能找到较好的设置值,同时需要预先给出 $GOAL$ 或 MN 参数.在实际应用时,既要保证性能又不能使网络规模过大,事先给出合适的 $GOAL$ 或 MN 参数是比较困难的.

为比较 newrb 方法和本文方法的性能,可对以下两种情况进行实验:第 1 种情况是比较同等网络规模下二者的性能,即以本文方法得到的数据样本的最终聚类数作为 newrb 方法的 MN 参数,此时两种方法可以得到隐层神经元数相同的 RBF 神经网络;第 2 种情况是以本文方法训练后的误差作为 newrb 方法的 $GOAL$ 参数,比较此时 RBF 网络的规模和泛化能力.表 1 给出了不同方法及情况下参数的取值情况,其中差异度阈值 TH_d 是由差异度阈值选取方法求出的.

两种方法对 3 个数据样本集的分类测试性能比较如表 2 所示.在同等网络规模下(第 1 种情况),本文

表 1 不同方法及情况下参数取值

数据集	newrb 方法				本文方法	
	第 1 种情况		第 2 种情况		1	2
	SPREAD	MN	SPREAD	GOAL		
iris	1	11	1	4.07	1.4	0.001
wine	300	15	300	12.04	2.0	0.1
rand	1	24	1	8.76	0.5	0.001

表 2 newrb 方法与本文方法的分类性能比较

数据集	newrb 方法				本文方法	
	第 1 种情况		第 2 种情况		隐层神经元数	分类精度 / %
	隐层神经元数	分类精度 / %	隐层神经元数	分类精度 / %		
iris	11	97.3	25	98.0	11	98.0
wine	15	93.3	17	96.6	15	95.5
rand	24	90.9	32	92.3	24	93.3

方法的分类性能明显优于 newrb 方法;newrb 方法要达到本文方法的训练精度(第 2 种情况),需要更多的隐层神经元;二者对测试样本的分类精度不相上下.实验结果表明,本文方法设计的 RBF 网络不仅冗余较少,而且具有良好的泛化性能.

4.3 混合学习算法的收敛性能

本文针对 RBF 网络的特点,给出一种对隐层中心向量和宽度参数采用 BP 算法,而对输出层权值采用线性最小二乘滤波的混合学习算法.与单纯采用 BP 算法对所有参数进行调节的训练算法相比,该方法可以大大加快网络的训练速度.

图 2 ~ 图 4 给出了 iris, wine 和 rand 3 个数据集的 BP 算法和本文混合学习算法的学习误差曲线,其中实线为 BP 算法学习曲线,虚线为本文算法学习

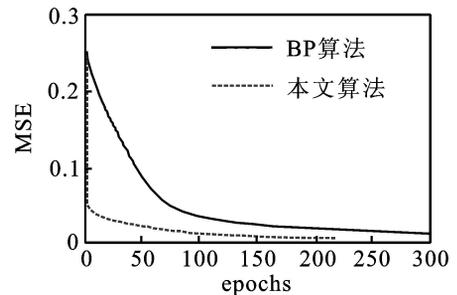


图 2 iris 学习误差曲线

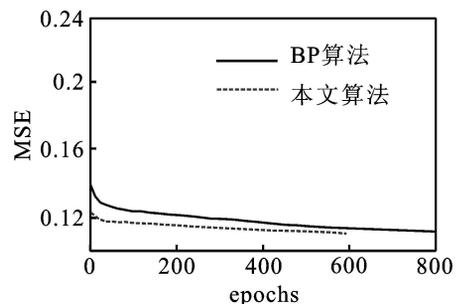


图 3 wine 学习误差曲线

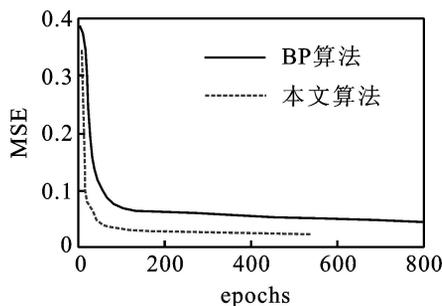


图4 rand学习误差曲线

曲线。从学习曲线可以看出,本文训练算法优于单纯的BP训练算法,特别是在训练的前期,本文算法收敛速度明显快于BP算法。

5 结论

本文提出一种粗糙集与RBF网络集成的RBF神经网络设计方法。该方法设计的RBF网络具有以下特点:

1) 算法易于实现,而且不需提供任何先验参数,唯一需要确定的差异度阈值也有合适的算法可供选取,不需人为给定,从而避免了设计过程中的主观盲目性;

2) 本文方法设计的RBF网络不仅结构简洁,而且具有良好的泛化能力;

3) RBF网络的隐层神经元表示具体的决策模式,因而具有明确的语义,从而使网络具有较强的语义解释性;

4) 对隐层参数和输出权值采用BP算法和线性最小二乘滤波的混合学习算法,收敛速度优于单纯的BP算法,尤其在训练的前期阶段更加明显。

参考文献(References)

- [1] Whitehead B A. Cooperative competitive genetic evolution of radial basis function centers and widths for time series prediction [J]. IEEE Trans on Neural Networks, 1996, 7(4): 869-880.
- [2] Lu Y W, Sundararajan N, Saratchandran P. Performance evaluation of sequential minimal radial basis function (RBF) neural network learning algorithm[J]. IEEE Trans on Neural Networks, 1998, 9(6): 308-317.
- [3] 王旭东, 邵惠鹤. RBF神经网络理论及其在控制中的应用[J]. 信息与控制, 1997, 26(4): 272-284. (Wang Xu-dong, Shao Hui-he. The theory of RBF neural network and its application in control [J]. Information and Control, 1997, 26(4): 272-284.)
- [4] Haykin S. Neural networks: A comprehensive foundation[M]. 2nd ed. Beijing: China Machine Press, 2004:183-220.
- [5] Bezdek J C. Pattern recognition with fuzzy objective function algorithm[M]. New York: Plenum, 1981.

- [6] Kohonen T. Self organization and associative memory [M]. 3rd ed. Berlin: Springer-Verlag, 1989.
- [7] Pawlak Z. Rough set theory and its application to data analysis[J]. Cybernetics and Systems, 1998, 29(9): 661-688.
- [8] Dougherty J, Kohavi R, Shami M. Supervised and unsupervised discretization of continuous features [C]. Proc of 12th Int Conf on Machine Learning. Morgan Kaufmann, 1995: 194-202.
- [9] Nguyen H S, Skowron A. Quantization of real-valued attributes [C]. Second Int Joint Conf on Information Sciences. Wrightsville Beach, 1995: 34-37.
- [10] 席静, 欧阳为民. 基于聚类的连续值属性最佳离散化算法[J]. 小型微型计算机系统, 2000, 21(10): 1025-1027. (Xi Jing, Ouyang Wei-min. Clustering based algorithm for bested discretizing continuous valued attributes[J]. Mini-micro Computer System, 2000, 21(10): 1025-1027.)
- [11] Wroblewski J. Finding minimal reducts using genetic algorithms[C]. Proc of the Int Workshop on Rough Sets Soft Computing at Second Annual Joint Conf on Information Sciences. Wrightsville Beach, 1995: 186-189.
- [12] 孙健, 申瑞民, 韩鹏. 一种新颖的径向基函数(RBF)网络学习算法[J]. 计算机学报, 2003, 26(11): 1562-1567. (Sun Jian, Shen Rui-min, Han Peng. An original RBF network learning algorithm [J]. Chinese J of Computers, 2003, 26(11): 1562-1567.)
- [13] Moody J, Darken C J. Fast learning in networks of locally-tuned processing units[J]. Neural Computing, 1989, 1(2): 281-294.
- [14] Chen S, Mulgrew B, McLaughlin S. Adaptive Bayesian feedback equalizer based on radial basis function network[C]. Int Conf on Communications. Chicago, 1992, 3: 1267-1271.
- [15] Berthold M R, Diamond J. Boosting the performance of RBF networks with dynamic decay adjustment[J]. Advances in Neural Information Processing Systems, 1995, 7(3): 521-528.
- [16] Lowe D. What have neural networks to offer statistical pattern processing [C]. Proc of the SPIE Conf on Adaptive Signal Processing. San Diego, 1991: 460-471.
- [17] Mitra S, Kuncheva L I. Improving classification performance using fuzzy MLP and two-level selective partitioning of the feature space [J]. Fuzzy Sets and Systems, 1995, 70(1): 1-13.