

基于可拓聚类的极限学习机神经网络

罗庚合*

(西安航空学院 机械工程系, 西安 710077)

(*通信作者电子邮箱 Luogenghe@126.com)

摘要:针对极限学习机(ELM)算法随机选择输入层权值的问题,借鉴第2类型可拓神经网络(ENN-2)聚类的思想,提出了一种基于可拓聚类的 ELM(EC-ELM)神经网络。该神经网络是以隐含层神经元的径向基中心向量作为输入层权值,采用可拓聚类算法动态调整隐含层节点数目和径向基中心,并根据所确定的输入层权值,利用 Moore-Penrose 广义逆快速完成输出层权值的求解。同时,对标准的 Friedman#1 回归数据集和 Wine 分类数据集进行测试,结果表明,EC-ELM 提供了一种简便的神经网络结构和参数学习方法,并且比基于可拓理论的径向基函数(ERBF)、ELM 神经网络具有更高的建模精度和更快的学习速度,为复杂过程的建模提供了新思路。

关键词:可拓聚类;极限学习机;径向基函数;回归;分类

中图分类号: TP18 **文献标志码:** A

Extension clustering-based extreme learning machine neural network

LUO Genghe*

(Department of Mechanical Engineering, Xi'an Aeronautical University, Xi'an Shaanxi 710077, China)

Abstract: During the construction process of Extreme Learning Machine (ELM), its input weights are randomly generated, and these parameters are non-optimized and contain no prior knowledge of the inputs. To solve these problems, combining the clustering method of Extension Neural Network type 2 (ENN-2), an extension clustering based extreme learning machine (EC-ELM) neural network was proposed. In EC-ELM neural network, the radial basis function centers of hidden neurons were firstly taken as the input weights, then extension clustering method was used to adaptively adjust the hidden neurons number and center vectors, and this well-adjusted information was trained by Moore-Penrose generalized inverse to obtain the output weights. Meanwhile, the effectiveness of this network was tested by the Friedman#1 dataset and the Wine dataset. The results indicate that EC-ELM provides a simple and convenient way to train the structure and parameters of neural network, and it is of higher modeling accuracy and faster learning speed than Extension theory based Radial Basis Function (ERBF) or ELM, which will provide a new way to apply the EC-ELM to complex process modeling.

Key words: extension clustering; extreme learning machine; radial basis function; regression; classification

0 引言

2006年,Guang-Bin Huang^[1-2]等提出了一类针对单隐含层前馈神经网络学习算法,称为极限学习机(Extreme Learning Machine, ELM)算法,该算法可以在随机选择输入层权值前提下,利用 Moore-Penrose 广义逆解析求出输出层权值,同时又保证了网络的泛化能力。与传统的误差反传(Back Propagation, BP)神经网络、径向基函数(Radial Basis Function, RBF)神经网络相比,ELM 学习算法具有学习速度快、可调参数少等许多优良特性。但是 ELM 的输入层权值是随机选择的,这些权值并没有包含任何训练样本的先验知识,因此可能会降低神经网络的拟合精度。

为了克服 ELM 学习算法的缺点,本文引入了可拓理论。可拓理论^[3-4]是一门由中国科学家提出的新型人工智能技术,该技术采用形式化模型讨论事物拓展的可能性和开拓创新的规律与方法,近几年来,在经济^[5]、管理^[6]、控制^[7]和决策^[8]中开展了研究和应用。其中,可拓神经网络(Extension Neural Network, ENN)是以 Mang-Hui Wang 为代表的多位专家融合可拓理论和神经网络优点后提出的一种新型神经网络。该类型网络对分类和聚类问题特别有效,分为监督学习 ENN-

1 型^[9]和非监督学习 ENN-2 型^[10],其中 ENN-2 型无需事先确定聚类数目,可通过可拓距离自动对训练样本进行聚类,同时又保证了聚类过程中的稳定性和收敛性。而 ELM 神经网络正是可以通过聚类提取训练样本的先验知识,优化网络的输入层权值。基于此,本文将 ENN-2 型神经网络可拓聚类思想和 ELM 学习算法相结合,在建立输入变量基元模型和隐含层径向基函数中心向量基元模型的基础上,通过可拓变换实现训练样本在输入变量空间中的聚类,将聚类后的聚类数目、聚类中心分别作为 ELM 神经网络的隐含层节点数目和输入层参数,并利用 ELM 学习算法快速求出输出层参数,最终探索出了一种基于可拓聚类的 ELM (Extension Clustering based Extreme Learning Machine, EC-ELM)神经网络。同时采用 UCI 标准数据库中的 Friedman#1 数据集和 Wine 数据集进行测试,结果验证了该 EC-ELM 神经网络在处理回归和分类问题中的可行性和有效性,为神经网络的实际应用提供了帮助。

1 ELM 学习算法

ELM 学习网络为前馈式网络,只有一个隐含层,假设 ELM 网络中包括了 U 个不同的训练样本,其集合表示为:

$$\Omega = \{(X_u, Y_u) | u = 1, 2, \dots, U\}$$

$$\begin{aligned} \mathbf{X}_u &= [x_{u1}, \dots, x_{uN}] \in \mathbf{R}^N; \\ \mathbf{Y}_u &= [y_{u1}, \dots, y_{uV}] \in \mathbf{R}^V \end{aligned}$$

则具有 L 个隐含层神经元的 ELM 输出表示为:

$$f(\mathbf{X}_u) = \sum_{l=1}^L \beta_l G(\mathbf{a}_l, b_l, \mathbf{X}_u) \quad (1)$$

其中: $\beta_l = [\beta_{l1}, \beta_{l2}, \dots, \beta_{lV}]^T$ 为第 l 个隐含层神经元与输出层神经元的连接权值向量; $G(\mathbf{a}_l, b_l, \mathbf{X}_u)$ 为与输入 \mathbf{X}_u 对应的第 l 个隐含层神经元的输出,且隐含层神经元可以分为加性神经元、RBF 神经元和复杂神经元三种类型。当隐含层神经元为 RBF 神经元时,其相应的输出为:

$$G(\mathbf{a}_l, b_l, \mathbf{X}_u) = g(b_l \|\mathbf{X}_u - \mathbf{a}_l\|) = \exp\left(-\frac{\|\mathbf{X}_u - \mathbf{a}_l\|^2}{2b_l^2}\right)$$

其中 \mathbf{a}_l 和 b_l 分别为第 l 个径向基函数的中心和影响宽度。

在 ELM 理论中,对于给定的 L 个 RBF 隐含层神经元,存在 β_l, \mathbf{a}_l 和 b_l ,使得该单隐含层前馈神经网络能够以零误差逼近样本期望 \mathbf{Y}_u 。因此,由式(1)可得一个含有 U 个方程的线性方程组:

$$\mathbf{H}\beta = \mathbf{Y} \quad (2)$$

其中:

$$\mathbf{H} = \begin{bmatrix} G(\mathbf{a}_1, b_1, \mathbf{X}_1) & \dots & G(\mathbf{a}_L, b_L, \mathbf{X}_1) \\ \vdots & & \vdots \\ G(\mathbf{a}_1, b_1, \mathbf{X}_U) & \dots & G(\mathbf{a}_L, b_L, \mathbf{X}_U) \end{bmatrix}_{U \times L}$$

$$\beta = [\beta_1^T, \dots, \beta_L^T]^T_{L \times V}$$

$$\mathbf{Y} = [\mathbf{Y}_1^T, \dots, \mathbf{Y}_U^T]^T_{U \times V}$$

由此,隐含层神经元与输出层神经元的连接权值 β 可以由式(2)的极小 2-范数最小二乘解得:

$$\hat{\beta} = \mathbf{H}^+ \mathbf{Y} \quad (3)$$

其中 \mathbf{H}^+ 为 \mathbf{H} 的 Moore-Penrose 广义逆。因此,在给定一个训练样本集 Ω ,隐含层神经元个数 L 后,ELM 学习算法步骤为:

1) 随机设定隐含层神经元的径向基中心 \mathbf{a}_l 和影响宽度 $b_l, l=1, 2, \dots, L$;

2) 计算隐含层输出矩阵 \mathbf{H} ;

3) 计算隐含层神经元与输出层神经元的连接权值: $\hat{\beta} = \mathbf{H}^+ \mathbf{Y}$ 。

2 基于可拓聚类的 ELM 神经网络

由于 ELM 学习算法随机选择隐含层神经元的中心和宽度,虽然提高了神经网络的训练速度,但是这些中心和宽度相对于输入数据来说,不是最优的选择。在实际应用中,为了使得神经网络有较好的泛化性能,可能会需要较多的隐含层神经元,因而增加了网络的复杂度。为了减少隐含层神经元个数、提高网络的泛化性能,引入可拓聚类算法,通过对训练样本分类,动态调整隐含层节点数目和径向基中心,形成基于可拓聚类的 ELM 神经网络。

图 1 展示了所提出的 EC-ELM 神经网络结构,其中输入层权值经过可拓聚类算法获得,输出层权值经过 Moore-Penrose 广义逆解析获得。可拓聚类算法作为一种新型的解决聚类和分类问题方法,包括了采用基元模型实现待聚类对象的统一深层次描述,采用关联函数计算各对象与已存在聚类中心的最小可拓距离,以此判断对象是否属于当前已存在的类,采用可拓变换调整聚类中心,保证所有的聚类过程收敛。因此,将可拓聚类算法用于 ELM 神经网络中,可以对训练样本进行初步分析,指导神经网络向正确的方向训练。

具体的可拓聚类方法步骤如下:

1) 对于 ELM 神经网络的 U 个训练样本,构造每个样本

的基元模型,并找到每个特征的最大和最小值。其中,第 u 个训练样本的基元模型表示如下:

$$\mathbf{R}_{\mathbf{X}_u} = \begin{bmatrix} N_{\mathbf{X}_u} & c_{x_{u1}} & x_{u1} \\ \vdots & \vdots & \vdots \\ c_{x_{uN}} & x_{uN} \end{bmatrix}; \quad u = 1, 2, \dots, U \quad (4)$$

$$x_{n, \max} = \max\{x_{1n}, \dots, x_{Un}\}, x_{n, \min} = \min\{x_{1n}, \dots, x_{Un}\}; \\ n = 1, 2, \dots, N \quad (5)$$

其中: $N_{\mathbf{X}_u}$ 表示第 u 个训练样本的名称, $c_{x_{un}}$ 为 $N_{\mathbf{X}_u}$ 的第 n 个特征, x_{un} 为第 u 个训练样本关于第 n 个特征的值, $x_{n, \max}$ 和 $x_{n, \min}$ 为所有训练样本中关于第 n 个特征的最大、最小值。

2) 样本特征归一化。在神经网络训练过程中,由于样本中每个特征的描述对象不同,量纲不统一,将导致各特征的贡献不同,因此在引入神经网络训练算法之前,采用删减变换 T_1 和缩小变换 T_2 组合,实现训练样本基元中每个特征的归一化:

$$\begin{aligned} x'_{un} &= T_1 x_{un} = x_{un} - x_{n, \min} \\ \bar{x}_{un} &= T_2 x'_{un} = x'_{un} / (x_{n, \max} - x_{n, \min}) \\ \mathbf{R}_{\bar{\mathbf{X}}_u} &= \begin{bmatrix} N_{\bar{\mathbf{X}}_u} & c_{\bar{x}_{u1}} & \bar{x}_{u1} \\ \vdots & \vdots & \vdots \\ c_{\bar{x}_{uN}} & \bar{x}_{uN} \end{bmatrix} = T_1 T_2 \mathbf{R}_{\mathbf{X}_u} = \\ &= \begin{bmatrix} N_{\bar{\mathbf{X}}_u} & c_{\bar{x}_{u1}} & (x_{u1} - x_{1, \min}) / (x_{1, \max} - x_{1, \min}) \\ \vdots & \vdots & \vdots \\ c_{\bar{x}_{uN}} & (x_{uN} - x_{N, \min}) / (x_{N, \max} - x_{N, \min}) \end{bmatrix} \end{aligned} \quad (6)$$

其中: $\mathbf{R}_{\bar{\mathbf{X}}_u}$ 表示归一化后的第 u 个训练样本, $N_{\bar{\mathbf{X}}_u}$ 表示基元 $\mathbf{R}_{\bar{\mathbf{X}}_u}$ 的名称。

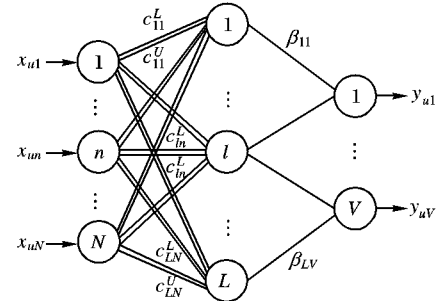


图 1 基于可拓聚类的 ELM 神经网络结构

3) 聚类参数初始化。定义聚类中样本的数目 $H=0$, 第 h 个聚类包括的训练样本数目为 M_h , 并根据样本分布等先验知识或实验法, 确定聚类中心与期望边界的距离阈值 θ 。

4) 读取第 1 ($u=1$) 个归一化后的训练样本 $\mathbf{R}_{\bar{\mathbf{X}}_1}$, 创建第 1 ($h=1$) 个类, 则参数 $H=1, M_h=1$, 对基元 $\mathbf{R}_{\bar{\mathbf{X}}_1}$ 采用复制变换 $T_{\bar{\mathbf{X}}_u \rightarrow \mathbf{C}_h}$ ($u=1, h=1$), 获得第 1 个聚类中心基元 $\mathbf{R}_{\mathbf{C}_1}$:

$$\mathbf{R}_{\mathbf{C}_h} = \begin{bmatrix} N_{\mathbf{C}_h} & c_{c_{h1}} & c_{h1} \\ \vdots & \vdots & \vdots \\ c_{c_{hN}} & c_{hN} \end{bmatrix} = T_{\bar{\mathbf{X}}_u \rightarrow \mathbf{C}_h} \mathbf{R}_{\bar{\mathbf{X}}_u} = \begin{bmatrix} N_{\mathbf{C}_h} & c_{c_{h1}} & \bar{x}_{u1} \\ \vdots & \vdots & \vdots \\ c_{c_{hN}} & \bar{x}_{uN} \end{bmatrix} \quad (7)$$

其中: \mathbf{C}_h 表示第 h 个聚类中心集合; c_{hn} 表示第 h 个聚类关于第 n 个特征的聚类中心, 并根据所定义的距离阈值参数 θ , 可知特征 c_{hn} 所对应的经典域为 $\mathbf{V}_{c_{hn}} = \langle c_{hn}^L, c_{hn}^U \rangle$, 其中 L 表示经典域的下限, U 表示经典域的上限, $c_{hn}^L = c_{hn} - \theta$, $c_{hn}^U = c_{hn} + \theta$ ($n=1, 2, \dots, N$)。

5) 读取下一个训练样本 (即 $u=u+1$), 计算该输入属性与已存在的 H 个聚类中心的最小可拓距离。定义第 u 个训

训练样本 \bar{X}_u 与第 h 个聚类中心集合 C_h 的可拓距离公式为:

$$d(\bar{X}_u, C_h) = \sum_{n=1}^N \alpha_n \rho(\bar{x}_{un}, \langle c_{hn}^L, c_{hn}^U \rangle) = \sum_{n=1}^N \alpha_n [|\bar{x}_{un} - c_{hn}| - (c_{hn}^U - c_{hn}^L)/2] \quad (8)$$

其中: c_{hn}^L 和 c_{hn}^U 分别表示已存在的第 h 个聚类中心 c_{hn} 的经典域下限值和上限值, $h=1, 2, \dots, H$; α_n 为关于第 n 个样本的权重系数, 用以衡量样本的重要性程度, 这里采用平均分配权重, 即选取 $\alpha_n = 1/N$ 。根据计算出的可拓距离集合 $\{d(\bar{X}_u, C_1), d(\bar{X}_u, C_2), \dots, d(\bar{X}_u, C_H)\}$, 采用式(9)查找与第 u 个训练样本基元 R_{X_u} 距离最近的聚类编号:

$$d_m = \min \{d(\bar{X}_u, C_1), d(\bar{X}_u, C_2), \dots, d(\bar{X}_u, C_H)\} \quad (9)$$

其中: d_m 表示第 u 个训练样本基元 R_{X_u} 与所有已存在聚类的最小可拓距离, m 表示可拓距离集合 $\{d(\bar{X}_u, C_1), d(\bar{X}_u, C_2), \dots, d(\bar{X}_u, C_H)\}$ 中值最小的元素编号。

6) 如果 $d_m > 0$, 表示第 u 个训练样本不属于已经存在的所有聚类, 增加一个新聚类, 即 $H = H + 1, M_H = 1$, 并对归一化后的第 u 个训练样本基元 R_{X_u} 采用式(7)的复制变换 $T_{\bar{X}_u \rightarrow C_H}$, 获得第 H 个聚类中心基元 R_{C_H} ; 反之, 如果 $d_m \leq 0$, 表示第 u 个训练样本属于第 m 个聚类, 则第 m 个聚类所包括的训练样本数目 $M_m = M_m + 1$, 并采用置换变换 $T_{\bar{X}_u \cup C_m \rightarrow C_m}$ 更新第 m 个聚类中心:

$$R_{C_m} = \begin{bmatrix} N_{C_m} & c_{c_{m1}} & c_{m1} \\ \vdots & \vdots & \vdots \\ c_{c_{mN}} & c_{mN} & \end{bmatrix} = T_{\bar{X}_u \cup C_m \rightarrow C_m} R_{X_u} R_{C_m} = \begin{bmatrix} N_{C_m} & c_{c_{m1}} & \frac{c_{m1} \times (M_m - 1) + \bar{x}_{u1}}{M_m} \\ \vdots & \vdots & \vdots \\ c_{c_{mN}} & c_{mN} & \frac{c_{mN} \times (M_m - 1) + \bar{x}_{uN}}{M_m} \end{bmatrix} \quad (10)$$

7) 重新判断第 u 个训练样本之前所有的训练 ($k=1, 2, \dots, u-1$) 所属的聚类是否发生跳转。若其中第 k 个训练样本从聚类 p 变化到聚类 q , 则 $M_q = M_q + 1, M_p = M_p - 1$, 采用置换变换 $T_{\bar{X}_k \cup C_q \rightarrow C_q}$ 更新第 q 个聚类中心 (即式(10)), 同时采用置换变换 $T_{\bar{X}_k \cup C_p \rightarrow C_p}$ 更新第 p 个聚类中心 (即式(11))。如此循环, 直到前 u 个训练样本的所有聚类中心不发生变动。

$$R_{C_p} = \begin{bmatrix} N_{C_p} & c_{c_{p1}} & c_{p1} \\ \vdots & \vdots & \vdots \\ c_{c_{pN}} & c_{pN} & \end{bmatrix} = T_{\bar{X}_k \cup C_p \rightarrow C_p} R_{X_k} R_{C_p} = \begin{bmatrix} N_{C_p} & c_{c_{p1}} & \frac{c_{p1} \times (M_p + 1) - \bar{x}_{k1}}{M_p} \\ \vdots & \vdots & \vdots \\ c_{c_{pN}} & c_{pN} & \frac{c_{pN} \times (M_p + 1) - \bar{x}_{kN}}{M_p} \end{bmatrix} \quad (11)$$

8) 判断 $u \geq U$, 如果成立, 表示训练样本聚类过程完成, 则进入下一步; 否则返回第 5) 步, 判断下一个训练样本所属的类。

9) 设置隐含层节点数目 L 等于聚类的数目 H , 将每个中心基元作为径向基中心向量, 即 $a_h = [c_{h1}, c_{h2}, \dots, c_{hN}]$ ($h=1, 2, \dots, H$), 每个径向基中心 a_h 最近邻中心距离的 1~2 倍作为 a_h 对应的影响宽度 b_h 。根据所确定的 a_h 和 b_h , 建立隐含层输出矩阵 H , 采用 Moore-Penrose 广义逆求取隐含层神经元与输出层神经元的连接权值 $\hat{\beta}$, 由此完成可拓 ELM 神经网络的构建。

通过以上步骤, 基于可拓聚类的 ELM 神经网络实现流程

如图 2 所示。

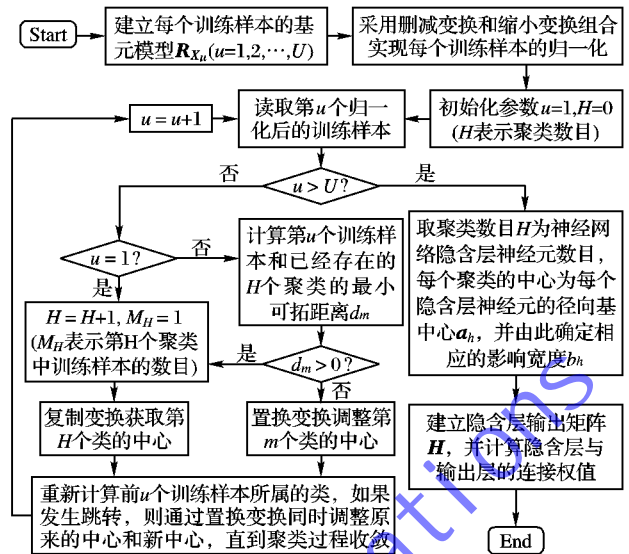


图 2 EC-ELM 神经网络实现流程

3 实验测试

为验证该基于可拓聚类的 ELM 神经网络的有效性和稳定性, 分别选取 UCI 数据库中的 Wine 数据集和 Friedman#1 数据集进行测试, 其各自信息如表 1 所示。其中在 UCI 数据库中, Friedman#1 数据集是由 5 个 $[0, 1]$ 区间均匀分布的输入变量和 1 个服从正态分布的 $N(0, 1)$ 的噪声变量通过以下函数关系式产生的:

$$y = 10 \sin(\pi x_1 x_2) + 20(x_3 - 0.5)^2 + 10x_4 + 5x_5 + \varepsilon; \quad x_i \sim U[0, 1], \varepsilon \sim N(0, 1) \quad (12)$$

表 1 测试数据集

测试数据集	样本数目 (训练/泛化)	输入属性	输出属性	用途
Friedman#1	1400 (934/466)	5	1	回归
Wine	178 (119/59)	13	1	分类

3.1 Friedman#1 实验测试

在可拓聚类中, 距离阈值 θ 是一个非常重要的参数, 其选取直接决定了所聚类的类数目, 进而影响 ELM 神经网络的结构和拟合精度。选取 Friedman#1 数据集进行实验测试, 表 2 列出了 EC-ELM 网络各项性能指标随参数 θ 的变化情况。

表 2 Friedman#1 数据集上距离阈值 θ 与 EC-ELM 网络性能的关系

θ	隐含层节点数目	回想误差/%	回想标准差	泛化误差/%	泛化标准差
0.10	302	1.07	0.1101	1.84	0.2297
0.15	85	2.98	0.3184	3.89	0.4335
0.20	34	7.32	0.7276	7.76	0.7978
0.25	20	10.92	1.1711	10.92	1.0985

从表 2 可以看出, 随着距离阈值 θ 增大, 隐含层节点数目减少, 回想和泛化误差、标准差增大, 精度变低、稳定性变差。因此 θ 的选取需要综合权衡网络复杂度和精度, 即在达到精度的要求下尽可能使网络结构更为简单。以此为目标, 对于 Friedman#1 数据集, 设置回想误差的精度标准为 3%, 此时所选取距离阈值参数 θ 为 0.15。

为验证 EC-ELM 神经网络同时具有带知识的输入权值和较快的训练速度的特征, 在 Friedman#1 数据集上, 将 EC-ELM 神经网络与标准的 ELM 神经网络、文献[11]提出的基于可拓理论的 RBF (Extension theory based RBF, ERBF) 神经网络的

性能进行比较,比较结果如表3所示。

表3 Friedman#1 数据集上 ERBF、ELM 和 EC-ELM 测试结果对比

隐含层节点数目	神经网络	训练迭代次数	回想误差/%	泛化误差/%
302	ERBF	>100	4.31	6.72
	ELM	50	1.10	1.89
	EC-ELM	1	1.07	1.84
85	ERBF	243	6.55	7.76
	ELM	50	3.07	3.99
	EC-ELM	1	2.98	3.89
34	ERBF	>100	10.14	11.29
	ELM	50	8.52	9.56
	EC-ELM	1	7.32	7.76
20	ERBF	>100	18.31	17.66
	ELM	50	12.70	13.21
	EC-ELM	1	10.92	10.92

从表3可以看出,相比于 ERBF 和 ELM 神经网络,无论隐含层节点数目为 302,85,34 或 20 时,EC-ELM 都具有较小的回想误差和泛化误差,反映出 EC-ELM 通过可拓聚类获得的中心向量可以使所训练的神经网络更接近真实值。同时,ERBF 神经网络使用的是误差反馈型训练方法,而 ELM 神经网络由于受输入权值的影响,其网络输出值是 50 次 ELM 学习算法反复实验后的平均值。因此,相比于 ERBF 和 ELM 神经网络,EC-ELM 只需要一次迭代,训练速度更快,输出稳定性更高。

3.2 Wine 实验测试

从 UCI 标准数据库中选取分类数据集 Wine,通过可拓聚类对训练样本分类,确定径向基中心向量,并利用 Moore-Penrose 广义逆直接求出输出层权值,由此完成 EC-ELM 在 Wine 数据集上的建模。表4列出了 EC-ELM 网络各项性能指标与可拓聚类时所设置的距离阈值参数 θ 的关系。

表4 Wine 数据集上距离阈值 θ 与 EC-ELM 网络性能的关系

θ	隐含层节点数目	回想分类正确率/%	泛化分类正确率/%
0.15	18	99.16	100.00
0.20	8	97.48	94.92
0.25	4	97.48	98.31
0.30	2	68.91	69.49

综合权衡网络复杂度和精度,距离阈值参数 θ 可取 0.25。表4中, $\theta = 0.3$ 对应的回想和泛化分类正确率分别只有 68.91%,69.49%,这是因为当 $\theta = 0.3$ 时,可拓聚类只找到了 2 个隐含层节点,而 Wine 数据集的目标输出为 3 个类(即神经网络的期望输出为 1,2,3),隐含层节点数目不能与期望输出形成一一映射,故此时的回想和泛化分类正确率明显偏低。

为验证 EC-ELM 神经网络用于分类数据集时的有效性,在标准的分类数据集 Wine 上,将 EC-ELM 神经网络与 ERBF 神经网络、标准的 ELM 神经网络进行比较,表5记录了在不同的隐含层节点数目下,三种算法的训练迭代次数、回想分类准确率和泛化分类准确率。

从表5可以看出,相比于 RBF 神经网络,EC-ELM 不但回想和泛化分类准确率高,而且只需要一次迭代,训练速度更快。相比于 ELM 神经网络,EC-ELM 具有更高的回想和泛化分类准确率,尤其是当隐含层节点数目为 4 时,EC-ELM 通过对输入权值加入先验知识以提高训练和泛化精度的优势非常明显,而随着隐含层节点数目的增加,可调的输出层权值数目

增多,ELM 与 EC-ELM 训练和泛化精度的差距将越来越小。

表5 Wine 数据集上 RBF、ELM 和 EC-ELM 测试结果对比

隐含层节点数目	神经网络	训练迭代次数	回想分类准确率/%	泛化分类准确率/%
18	ERBF	1000	93.28	93.22
	ELM	50	94.47	89.29
	EC-ELM	1	99.16	100.00
8	ERBF	1000	91.60	91.53
	ELM	50	83.48	81.09
	EC-ELM	1	97.48	94.92
4	ERBF	1000	91.60	91.53
	ELM	50	67.56	65.39
	EC-ELM	1	97.48	98.31
2	ERBF	1000	67.23	50.85
	ELM	50	52.13	49.90
	EC-ELM	1	68.91	69.49

4 结语

针对极限学习及随机选择输入层权值的缺点,本文借鉴第2类型可拓神经网络(ENN-2)聚类的思想,提出了一种基于可拓聚类的 ELM 神经网络,该网络运用可拓聚类算法确定 ELM 神经网络的结构和输入层参数,运用 ELM 学习方法解析求出网络的输出层参数,优化了神经网络的泛化能力。通过对 UCI 标准数据库中的回归数据集 Friedman#1 和分类数据集 Wine 的测试,结果表明,EC-ELM 神经网络不仅通过样本先验知识优化了网络的结构和输入层权值,而且其学习速度和网络拟合精度都要优于 ERBF 和 ELM 神经网络。将所提出的 EC-ELM 神经网络应用于复杂过程的建模,可以大大提高建模的效率和准确率。

参考文献:

- [1] HUANG G B, ZHU Q Y, SIEW C-K. Extreme learning machine: theory and applications[J]. Neurocomputing, 2006, 70(1/2/3): 489-501.
- [2] HUANG G B, CHEN L. Enhanced random search based incremental extreme learning machine[J]. Neurocomputing, 2008, 71(16/17/18): 3460-3468.
- [3] CAI W. Extension theory and its application[J]. Chinese Science Bulletin, 1999, 44(17): 1538-1548.
- [4] 蔡文,杨春燕.可拓学的应用研究、普及与推广[J].数学的实践与认识,2010,40(7):214-220.
- [5] 赵赛,陈炳发,王体春.基于可拓关联规则推理的产品方案设计[J].机械设计与制造,2012,259(9):251-253.
- [6] CAI W. Extension management engineering and applications[J]. International Journal of Operations and Quantitative Management, 1999, 5(1): 59-72.
- [7] YE J. Application of extension theory in misfire fault diagnosis of gasoline engines[J]. Expert Systems with Applications, 2009, 36(2): 1217-1221.
- [8] 何斌.基于可拓-集对的经济策略生成方法[J].系统工程,2004,22(4):91-94.
- [9] WANG M H, HUANG C P. Extension neural network and its applications[J]. Neural Networks, 2003, 16(5/6): 779-784.
- [10] WANG M H. Extension neural network - type 2 and its applications[J]. IEEE Transactions on Neural Networks, 2005, 16(6): 1352-1361.
- [11] 徐圆,冯晶,朱群雄.基于可拓理论的 RBF 神经网络研究及应用[J].控制与决策,2011,26(11):1721-1725.