

基于蚁群优化的极限学习机选择性集成学习算法

杨 菊 袁玉龙 于化龙

(江苏科技大学计算机科学与工程学院 镇江 212003)

摘 要 针对现有极限学习机集成学习算法分类精度低、泛化能力差等缺点,提出了一种基于蚁群优化思想的极限学习机选择性集成学习算法。该算法首先通过随机分配隐层输入权重和偏置的方法生成大量差异的极限学习机分类器,然后利用一个二叉蚁群优化搜索算法迭代地搜寻最优分类器组合,最终使用该组合分类测试样本。通过 12 个标准数据集对该算法进行了测试,该算法在 9 个数据集上获得了最优结果,在另 3 个数据集上获得了次优结果。采用该算法可显著提高分类精度与泛化性能。

关键词 极限学习机,蚁群优化,集成学习,选择性集成

中图分类号 TP181 **文献标识码** A **DOI** 10.11896/j.issn.1002-137X.2016.10.050

Selective Ensemble Learning Algorithm of Extreme Learning Machine Based on Ant Colony Optimization

YANG Ju YUAN Yu-long YU Hua-long

(School of Computer Science and Engineering, Jiangsu University of Science and Technology, Zhenjiang 212003, China)

Abstract This paper proposed a novel selective ensemble learning algorithm of extreme learning machine (ELM) based on the idea of ant colony optimization. The algorithm can overcome the drawbacks of the existing ensemble learning algorithms of ELM, such as low classification accuracy and generalization ability. Firstly, the proposed algorithm generates lots of ELM classifiers by the strategy of randomly assigning input weights and biases of the hidden layer. It then uses a binary ant colony optimization algorithm to search the optimal combination of ELMs. At last, it uses the extracted combination of classifiers to classify test instances. The experimental results on 12 baseline data sets show that the proposed algorithm has acquired the best performance on nine data sets and the second best performance on the three remaining data sets. Adopting the proposed algorithm can obviously help to improve the classification accuracy and generalization ability.

Keywords Extreme learning machine, Ant colony optimization, Ensemble learning, Selective ensemble

1 引言

极限学习机(Extreme Learning Machine, ELM)是 Huang 等人^[1]于 2006 年提出的一种新的单隐层前馈神经网络(Single-hidden Layer Feed forward Neural Networks, SLFNs)学习算法。与传统的训练误差反传(Back Propagation, BP)算法^[2]不同,ELM 无需对网络权重与偏置进行迭代调整,而是随机指定隐层的输入权重和偏置,并通过最小二乘法求取隐层的输出权重,故其在很大程度上提升了神经网络的训练速度,同时增强了其泛化能力^[3]。鉴于 ELM 上述的两个优点,其已被广泛应用于图像分类^[4]、人体动作识别^[5]、销售预测^[6]、价格预测^[7]、疾病诊断^[8]以及生物信息学^[9,10]等领域。

前人研究表明,尽管 ELM 具有训练速度快、泛化能力强等优点,但由于其隐层的输入权重与偏置完全是随机生成的,因此 ELM 具有稳定性差的缺点^[11]。同时,由于 ELM 在训

练过程中总是试图去最大限度地拟合训练样本,因此在一定程度上仍会降低其泛化能力,从而产生过适应的现象^[12]。

针对上述问题,研究者已从集成学习的角度出发提出了几种有效的解决方案。Lan 等人^[11]通过整合多个在线序列极限学习机(Online Sequential Extreme Learning Machine, OS-ELM)的输出均值,有效地提升了单个 OS-ELM 的稳定性。Sun 等^[6]采用相同的思想提出了一种集成学习算法 ELME,并将其应用于销售预测领域,取得了比单个 ELM 更优的分类精度。特别需要说明的是,ELME 可被看作是一种 ELM 的加权 Bagging^[13]集成学习算法。2002 年, Zhou 等人^[14]提出了选择性集成的概念,即在大量基分类器中选出那些精确且具有较大差异的个体进行集成,可能比集成全部的分类器性能更优。借鉴这一思想, Liu 和 Wang^[12]提出了一种集成学习算法 EN-ELM。该算法同时考虑了个体 ELM 的分类精度与泛化能力,并选取性能最好的半数 ELM 进行集

到稿日期:2015-09-21 返修日期:2015-12-16 本文受国家自然科学基金(61305058),江苏省自然科学基金(BK20130471),中国博士后特别资助计划项目(2015T80481),中国博士后科学基金(2013M540404),江苏省博士后基金(1401037B),江苏省 2015 年普通高校研究生实践创新计划项目(1132161503)资助。

杨 菊(1990-),女,硕士生,主要研究方向为机器学习、主动学习;袁玉龙(1988-),男,硕士生,主要研究方向为机器学习、数据挖掘;于化龙(1982-),男,博士,副教授,主要研究方向为机器学习、数据挖掘、生物信息学等。

成。特别地,EN-ELM通过引入内部交叉验证法来计算个体ELM的分类精度,通过计算隐层输出矩阵的L2范数来评估个体ELM的泛化能力。实验结果表明,该算法通常可获得更好的泛化性能。陆慧娟等人^[10]提出了一种基于输出不一致测度的ELM选择性集成学习算法D-D-ELM,并将其应用于DNA微阵列数据分类领域。该算法首先以输出不一致测度为标准对多个ELM模型进行相异性评估,然后根据ELM的平均分类精度迭代地剔除掉相应的模型,最后对筛选后的模型用多数投票法进行集成。实验结果表明,该方法能够以较少的模型数量达到稳定的分类精度。

尽管上述各类集成学习算法均在一定程度上提升了单个ELM的分类精度与泛化能力,但其分类性能仍有较大的提升空间。本文借鉴蚁群优化的思想^[15-17],提出了一种新的极限学习机选择性集成学习算法。该算法首先通过随机分配隐层输入权重与偏置的方式生成大量差异的ELM,然后利用一种二叉蚁群优化搜索算法^[16,17]从中选取最优的分类器组合。特别需要说明的是,在对蚁群个体进行适应度评估时,算法同时考虑了个体的分类精度与泛化能力,力求选取少而精的分类器进行集成。在Keel数据仓库^[18]中随机选取了12个基准数据集对本文算法的性能进行了验证。实验结果表明,与前人提出的几种集成学习算法相比,所提算法可获得更好的分类性能。

2 方法

2.1 极限学习机

2006年,新加坡南洋理工大学教授Huang等人提出了单隐层前馈神经网络SLFNs的极限学习概念,并介绍了其基本原理^[1]。不同于传统的BP学习算法,极限学习机无需对网络权重与偏置进行迭代调整,通过最小二乘法直接学习即可得到,故极大地提升了网络的学习速度,并在一定程度上降低了网络陷入过拟合的概率^[1,3]。图1给出了SLFNs的基本结构。

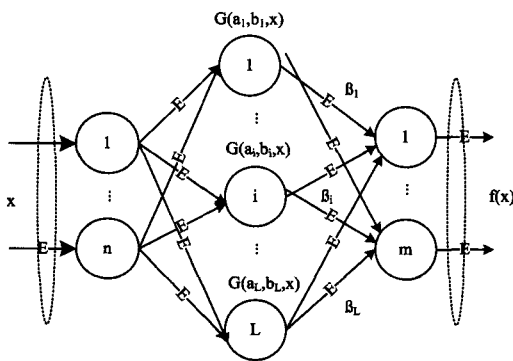


图1 单隐层前馈神经网络结构图

不失一般性,不妨设训练样本集为 $\mathcal{N} = \{(x_i, t_i) \in \mathbb{R}^n \times \mathbb{R}^m \mid i=1, 2, \dots, N\}$,其中, x_i 代表样本集中第*i*个训练样本, t_i 为其类别标记, n 和 m 分别代表特征与类别的个数,而 N 为训练样本的数量。若一个含有 L 个隐节点的SLFNs能以零误差拟合上述 N 个训练样本,则意味着存在 a_i, b_i 和 β_i 令下式成立:

$$f_L(x_j) = \sum_{i=1}^L \beta_i G(a_i, b_i, x_j) = t_j, j=1, 2, \dots, N \quad (1)$$

其中, a_i 和 b_i 分别代表第*i*个隐节点的输入权重和偏置, β_i 为第*i*个隐节点的输出权重, $G(a_i, b_i, x_j)$ 为激活函数,用于计算第*j*个样本在第*i*个隐节点上的输出值。则式(1)可以简化为如下形式:

$$H\beta = T \quad (2)$$

其中:

$$H(a_1, \dots, a_L, b_1, \dots, b_L, x_1, \dots, x_N) = \begin{bmatrix} G(a_1, b_1, x_1) & \dots & G(a_L, b_L, x_1) \\ \dots & \dots & \dots \\ G(a_1, b_1, x_N) & \dots & G(a_L, b_L, x_N) \end{bmatrix} \quad (3)$$

$$\beta = \begin{bmatrix} \beta_1^T \\ \vdots \\ \beta_L^T \end{bmatrix}_{L \times m} \quad \text{且} \quad T = \begin{bmatrix} t_1^T \\ \vdots \\ t_N^T \end{bmatrix}_{N \times m} \quad (4)$$

其中, H 为神经网络中隐层的输出矩阵,其每一行代表同一样本在不同隐节点的输出值,而每一列则代表不同样本在同一隐节点的输出值。

对于SLFNs而言,隐节点数 L 通常小于训练样本数 N ,故网络的训练误差不会保证收敛到完全为0,而是具有一个相对较小的训练误差矩阵 ξ ,则式(2)可重写为:

$$H\beta + \xi = T \quad (5)$$

其中,训练误差矩阵表示如下:

$$\xi = \begin{bmatrix} \epsilon_1^T \\ \vdots \\ \epsilon_N^T \end{bmatrix}_{N \times m} \quad (6)$$

其中, ϵ_i^T 表示网络对第*i*个样本的训练误差。BP学习算法通过迭代的方式不断下调 ξ 的L2范数,直至均方误差低于一个预先给定的阈值。而ELM则直接为隐节点的输入权重 a_i 和偏置 b_i 分别在 $(-1, 1)$ 和 $(0, 1)$ 范围内分配随机值,并通过下式近似求得隐层输出权重矩阵 β :

$$\hat{\beta} = H^\dagger T \quad (7)$$

其中, H^\dagger 为隐层输出矩阵 H 的Moore-Penrose广义逆,若 $H^T H$ 非奇异,则 $H^\dagger = (H^T H)^{-1} H^T$;若 HH^T 非奇异,则 $H^\dagger = H^T (HH^T)^{-1}$ 。 $\hat{\beta}$ 是对 β 的近似,其为式(2)的最小范数最小二乘解^[1]。

ELM算法可描述如下。

算法1 ELM算法

- 步骤1 为每个隐节点在 $(-1, 1)$ 范围内随机分配输入权重 a_i ,在 $(0, 1)$ 范围内随机分配偏置 b_i ;
- 步骤2 计算隐层输出矩阵 H ;
- 步骤3 使用式(7)计算隐层输出权重矩阵 β 。

前人工作发现:采用ELM算法训练神经网络,可较好地拟合任意的连续目标函数,其泛化能力通常优于BP算法,且训练时间则较BP算法有了大幅的降低^[1,3]。

2.2 二叉蚁群优化搜索算法

1991年,意大利学者Colormi等人借鉴蚁群在觅食行为中所体现出的智能行为提出了蚁群优化搜索算法,并将其成功应用于解决旅行商问题(Travelling Salesman Problem, TSP)^[15]。近年来,该算法已成为集群智能算法中重要的一员,并在诸多领域得到了实际应用。

2009年,Yu等人^[16,17]提出了二叉蚁群优化搜索算法,并

用其解决复杂的组合优化问题,如特征选择^[16]、数据集采样^[17]等。该算法继承了原始蚁群优化搜索算法的核心思想,即利用不同路径上的信息素浓度来指导蚁群寻优。图2为二叉蚁群优化搜索算法搜索过程的示意图。

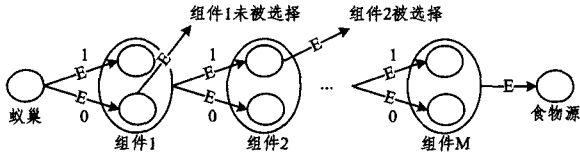


图2 二叉蚁群优化搜索算法搜索过程示意图

从图2可以看出,对于一个具有 M 个组件的组合优化问题,可按照生成顺序将其排列在蚁巢与食物源之间,每两个相邻组件间有两条路径相连,若蚂蚁选择路径1,则代表对应组件被选择;若选择路径0,则代表对应组件未被选择。蚂蚁每到达一个节点,对下一步的路径选择可通过式(8)进行计算。

$$p_{i,j} = \frac{\tau_{i,j}}{\sum_{j=0}^1 \tau_{i,j}} \quad (8)$$

其中, $\tau_{i,j}$ 表示在第 i 个组件在第 j 条路径上的信息素浓度, $p_{i,j}$ 表示蚂蚁在第 i 个组件上选择第 j 条路径的概率, $i \in \{1, 2, \dots, M\}$, $j \in \{0, 1\}$ 。

在每一轮寻优过程中,蚁群均从蚁巢出发,按照概率随机选择各个组件上的路径,直至到达食物源为止,从而形成问题的多个解。然后,根据问题的适应度函数 $fitness$ 对各个解的优劣进行打分并排序,选出一定比例的较优解,并将它们的路径存储于集合 E 中。在下一轮寻优前,首先使用蒸发因子 ρ 对所有路径上的信息素浓度进行等比例稀释:

$$\tau_{i,j}(t+1) = (1-\rho) \times \tau_{i,j}(t) \quad (9)$$

此后,利用所选出那部分较优解对其所对应路径上的信息素进行更新:

$$\Delta\tau_{i,j} = \begin{cases} fitness / (\lambda \times ant_n), & \text{如该路径} \in E \\ 0, & \text{如该路径} \notin E \end{cases} \quad (10)$$

其中, ant_n 为蚁群规模, λ 为所选较优解在蚁群中所占比例, $\Delta\tau_{i,j}$ 为对应路径上新增的信息素浓度。该算法迭代运行,不同路径上的信息素浓度差也将持续增大,直至收敛为止。二叉蚁群优化搜索算法流程如算法2所示。

算法2 二叉蚁群优化搜索算法

- 步骤1 为所有路径分配相应的初始信息素浓度值,标注目前最优解适应度为0;
- 步骤2 利用式(8)为每只蚂蚁进行路径选择,从而构成问题的多个可行解;
- 步骤3 评估每个解的适应度,并对其降序排列;
- 步骤4 对本代最优解与目前最优解的适应度值进行比较,若本代最优解适应度更大,则用其对目前最优解进行替换;
- 步骤5 选出一定比例高适应度的个体,记录其所经过的路径;
- 步骤6 利用式(9)与式(10)对所有路径上的信息素浓度进行更新;
- 步骤7 判断是否达到预设的最大迭代次数,若未达到,则转至步骤2继续执行;若已达到,则退出并输出目前最优解。

2.3 本文算法

本文拟采用2.2节描述的二叉蚁群优化搜索算法解决ELM的选择性集成问题,即在大量随机生成的ELM中选出部分个体进行集成,以达到比集成全部ELM更好的分类效

果。这是一个典型的组合优化问题,设初始ELM数为 M ,对于选择性集成而言共有 $2^M - 1$ 个备选解,当 M 较大时,备选解个数将是一个天文数字,此时通过枚举法获取最优解的做法将是不切实际的,而前文所提到的二叉蚁群优化搜索算法对解决此类问题却较为适用。

首先,需要考虑如何设计适应度函数,即对所选取的ELM子集优劣的评价问题。从分类精度和泛化能力两方面同时入手来设计该适应度函数。对于分类精度,拟借鉴文献[12]的思想,采用内部 N 折交叉验证的方法对其进行评估。首先将原始训练集划分为数量相等且互不相交的 N 个子集,每次用 $N-1$ 个子集进行训练,剩余的1个子集进行测试,当每一子集都进行了一次验证后,以平均分类精度来近似该ELM子集的分类精度。这样做的目的在于:在保证各基质分类器网络参数,即特性不变的前提下,可更有效地反映出它们各自的真实分类能力,其结果往往比在原始训练集上直接测试分类精度要更为真实客观。而对于泛化能力,则采用子集中各ELM的隐层输出权重矩阵 β 的L2范数均值 $\|\beta\|$ 来加以刻画。故对于一个特定的ELM子集 i 而言,其适应度函数 $fitness_i$ 可描述为如下形式:

$$fitness_i = \frac{Acc_i}{\|\beta\| / \|\beta_A\|} \quad (11)$$

其中, Acc_i 为子集中各ELM的平均分类精度, $\|\beta\|$ 和 $\|\beta_A\|$ 分别为子集中的ELM及全部备选ELM隐层输出权重矩阵的L2范数均值。从式(11)中不难看出,一个子集的平均分类精度 Acc 越高, $\|\beta\|$ 越小,则其适应度就越大。

其次,需考虑如何为不同路径赋予相应的信息素初值。众所周知,选择性集成的目的是以尽可能少的基质分类器来最大化分类的性能。故对于本文算法而言,需为所有的路径0赋予一个较大的初始信息素初值,而为路径1分配一个相对小的信息素初值。

此外,为防止算法过早收敛,从而陷入局部最优,本文也沿用了文献[16,17]中的做法:通过为信息素分配最小与最大值使算法具有持续发现新的更优解的能力,避免早熟现象的出现。即在对各路径上的信息素进行更新时,若更新后的信息素浓度小于预先设定的最小值 ph_{min} 或大于预先设定的最大值 ph_{max} ,则以对应的最小或最大值来加以替换。

本文算法的具体流程描述如算法3所示。

算法3 基于蚁群优化的ELM选择性集成学习算法

- 步骤1 调用算法1在原始训练集上随机生成 M 个基质ELM。其中,隐层权重与偏置分别对应于 $(-1,1)$ 与 $(0,1)$ 范围内的随机数,偏置从连续区间 $(0,1)$ 内随机选取,并对这两组参数及学习得到的网络输出权重矩阵 β 进行记录,计算全部 M 个ELM的网络输出权重矩阵的L2范数均值 $\|\beta_A\|$ 。
- 步骤2 将原始训练集等分为互不相交的 N 折,维持步骤1中所记录各基质分类器的权重与偏置不变,并执行 N 折交叉验证,从而得到对各基质分类器的近似评估精度。
- 步骤3 调用算法2搜寻最优的ELM子集。其中,通过步骤2中已得到的各基质分类器的近似评估精度来计算各子集的平均分类精度,并以步骤1中记录各基质分类器的网络输出权重矩阵计算各子集的 $\|\beta\|$,进而通过式(11)计算得到各子集的适应度。

步骤4 使用步骤3中获得的最优ELM子集对测试集进行验证,得到最终的测试分类精度。

特别需要指出以下两点:1)在保证基质分类器差异性方面,本文沿用了文献[10,12,13]的做法,即仅利用权重与偏置在生成时的随机性来保证基质分类器间的差异性,而并未采用传统的Bootstrap方法;2)本文在进行最终决策时,沿用了文献[6,12]的方法,即通过计算各ELM对应输出节点的均值而非多数投票来判别样本的类别。这种做法的优点在于可以最大限度地保留基质ELM中的原始信息。

3 实验结果与讨论

3.1 数据集

本文从Keel数据仓库^[16]中随机选取了12个数据集对各类算法的性能进行了实验比较,有关这些数据集的详细信息如表1所列。

表1 本文所用数据集

数据集	样本数	特征数	类别数	隐层节点数
hayes-roth	160	4	3	30
monk-2	432	6	2	30
bupa	345	6	2	30
heart	270	13	2	30
tae	151	5	3	30
sonar	208	60	2	30
pima	768	8	2	50
balance	625	4	3	50
wdbc	569	30	2	50
vowel	990	13	11	100
titanic	2201	3	2	100
segment	2310	19	7	100

从表1可以看出,所选取的数据集既包括了两类问题,也

涵盖了多类问题。此外,考虑到不同数据集之间的样本规模相差较大,特为小样本数据集赋予相对小的隐层节点数 L ,而为样本数较多的数据集分配相对大的 L 值。

3.2 实验初始参数设置

为体现本文算法的优越性,分别与几种前人提出的算法(包括ELM,ELME,EN-ELM及D-D-ELM)进行实验比较。为保证比较实验的公正性,对几类算法共有的参数做出了相同的设置:隐层均使用Sigmoid作为激活函数,基质分类器的规模 M 均预设为100,实验验证方法均使用外部5折交叉验证。同时,考虑到本文算法与EN-ELM方法均使用了内部 N 折交叉验证,故在此将 N 统一预设为5。

此外,通过大量实验为本文算法所独有的参数预设了较优的初值,如表2所列。

表2 本文算法的初始参数设置

参数	初值
种群规模 ant_n	50
迭代次数 cycle_n	20
路径1上的信息素浓度初值 $\tau_{i,1}$	1.0
路径0上的信息素浓度初值 $\tau_{i,0}$	2.0
蒸发因子 ρ	0.2
较优解比例 λ	0.2
信息素浓度下限 ph_{min}	0.5
信息素浓度上限 ph_{max}	4.0

3.3 实验结果与分析

考虑到外部交叉验证划分的随机性,本文采用10次随机划分,以均值±标准差的形式给出各方法的分类精度。各类方法的实验结果如表3所列,其中粗体表示各个数据集上得到的最优结果,标注下划线为次优结果,而斜体则表示最差结果。

表3 各类方法的分类精度比较

数据集	ELM	ELME	EN-ELM	D-D-ELM	本文算法
hayes-roth	0.7413 ±0.0348	<u>0.7738±0.0266</u>	0.7638±0.0325	0.7707±0.0206	0.7764±0.0269
monk-2	0.9149 ±0.0160	0.9321±0.0023	0.9442±0.0121	<u>0.9569±0.0033</u>	0.9685±0.0041
bupa	0.7072 ±0.0094	0.7183±0.0069	0.7113±0.0091	<u>0.7207±0.0077</u>	0.7258±0.0095
heart	0.8215 ±0.0206	0.8307±0.0084	0.8330±0.0075	0.8376±0.0091	<u>0.8352±0.0099</u>
tae	0.5186 ±0.0279	0.5360±0.0248	0.5385±0.0228	0.5346±0.0292	<u>0.5384±0.0256</u>
sonar	0.7249 ±0.0312	0.7523±0.0180	<u>0.7801±0.0215</u>	0.7752±0.0144	0.7879±0.0157
pima	0.7630 ±0.0070	0.7684±0.0066	<u>0.7748±0.0067</u>	0.7690±0.0058	0.7914±0.0078
balance	0.9573 ±0.0053	0.9715±0.0041	0.9698±0.0040	<u>0.9757±0.0034</u>	0.9783±0.0041
wdbc	0.9601 ±0.0069	0.9640±0.0038	0.9643±0.0061	0.9698±0.0022	<u>0.9686±0.0030</u>
vowel	0.7938 ±0.0079	0.8267±0.0134	<u>0.8334±0.0128</u>	0.8328±0.0098	0.8493±0.0082
titanic	<u>0.7885±0.0018</u>	0.7883±0.0011	0.7857±0.0037	0.7879±0.0018	0.7887±0.0011
segment	0.9356 ±0.0024	<u>0.9422±0.0019</u>	0.9412±0.0022	0.9402±0.0009	0.9444±0.0012

从表3可以看出,与ELM相比,各类ELM集成学习算法均有助于提升分类的性能,在除titanic以外的11个数据集上,ELM均得到了最低的分类精度。而与ELME算法相比,3种选择性集成学习算法也均在分类精度上有了或多或少的提高,表明仅选取一些较优的分类器进行集成要优于集成全部的分类器。

从表3的结果也可以看出,本文算法展现出了比EN-ELM和D-D-ELM更好的分类性能。具体而言,本文算法在9个数据集上得到了最优的分类结果,而在其它3个数据集上也得到了次优的分类结果。与之相对的是,EN-ELM和D-D-ELM分别只取得了1个和2个最优结果,并各取得了3个次优的结果。探究其原因不难发现:EN-ELM只孤立地考查

了个体分类器的性能,并没有考虑到分类器之间的内在联系,所选出的个体尽管均较优,但组合在一起却未必能形成合力;而D-D-ELM尽管在一定程度上考查了分类器之间的关联性,但其仅从分类精度的角度进行考虑,这种做法极有可能牺牲掉分类器的泛化性能,故易出现过适应的结果;而本文算法以子集为单位对集成的分类效果进行考查,充分考虑到了基质分类器之间的相互关系,同时从分类精度和泛化能力两方面入手来评估子集的优劣,这不但有助于提高整体的分类精度,而且可较好地保证集成学习的泛化性能。经过上述分析可知,本文算法能取得比EN-ELM和D-D-ELM更好的分类结果。

本文采用非参数的假设检验方法检验了本文算法的

优势是否在统计上显著,有关非参数假设检验的统计学知识可参考文献[19-21]。另,非参数的标准化测试软件可从以下网址获取: <http://sci2s.ugr.es/sicidm>。表4给出了各类算法在12个数据集上的平均排序(Average Ranking, AR)结果。

表4 各类算法的平均排序

算法	AR
本文算法	1.2500
ELM	4.7500
ELME	3.3333
EN-ELM	3.0000
D-D-ELM	2.6667

从表4可以看出,本文算法的平均排序值只有1.2500,是所有算法中最低的,而D-D-ELM的排序值则略低于EN-ELM和ELME,ELM是全部算法中排序最高的,故说明与单个ELM相比,各集成学习算法均可在一定程度上提高分类的性能。同时,Friedman检验返回了 3.9324×10^{-6} 的 p 值,故说明各类算法等同的假设是应该被拒绝的。因此,在 $\alpha=0.05$ 显著性水平下做了 $1 \times N$ Holm post-hoc检验,以调查本文算法是否在统计上显著优于其它算法,结果如表5所列。

表5 $\alpha=0.05$ 显著性水平下的Holm假设检验结果

i	算法	Z	p	Holm	假设 ($\alpha=0.05$)
4	ELM	5.4222	5.8878×10^{-8}	0.0125	拒绝
3	ELME	3.2275	0.0012	0.0167	拒绝
2	EN-ELM	2.7111	0.0067	0.025	拒绝
1	D-D-ELM	2.1947	0.0282	0.05	拒绝

从表5可以看出,本文算法在统计学上是显著优于其它4种比较算法的。为了进一步对各类算法的性能进行比较,采用Shaffer post-hoc检验对各算法进行了两两比较分析,结果如表6所列。在该表中,“+”代表行算法优于列算法,“-”则表示列算法优于行算法,“=”代表两算法在统计上无显著差异,括号中的数字代表调整后的 p 值。

表6 各算法比较的Shaffer假设检验结果

	ELM	ELME	EN-ELM	D-D-ELM	本文算法
ELM	×	=(0.1127)	-(0.0402)	-(0.0075)	-(5.89×10^{-7})
ELME	=(0.1127)	×	=(1.2112)	=(0.9051)	-(0.0075)
EN-ELM	+(0.0402)	=(1.2112)	×	=(1.2112)	-(0.0402)
D-D-ELM	+(0.0075)	=(0.9051)	=(1.2112)	×	=(0.1127)
本文算法	+(5.89×10^{-7})	+(0.0075)	+(0.0402)	=(0.1127)	×

从表6可以看出,本文算法在统计上明显优于ELM、ELME及EN-ELM算法,而与D-D-ELM算法相比,其统计上的差异并不显著。相比之下,ELME、EN-ELM及D-D-ELM等3种算法在进行两两比较时,均不具有显著的差异,证明这3种算法性能大体相当。相比之下,ELM明显劣于除ELME以外的其它算法。

此外,对4种集成学习算法在最终决策时所采用的基分类器个数进行了统计,结果如表7所列。其中,本文算法给出的是10次外部5折交叉验证的平均结果。

表7 各算法最终集成基分类器的个数

数据集	ELME	EN-ELM	D-D-ELM	本文算法
hayes-roth	100	50	30	19.37
monk-2	100	50	30	26.58
bupa	100	50	30	23.49
heart	100	50	30	17.56
tae	100	50	30	32.81
sonar	100	50	30	22.27
pima	100	50	30	14.98
balance	100	50	30	25.06
wdbc	100	50	30	18.78
vowel	100	50	30	21.54
titanic	100	50	30	20.39
segment	100	50	30	27.96

从表7可以看出,与其它几种集成学习算法相比,本文算法通常可以通过选取并集成更少的基分类器来得到更优的分类性能,从而表明了该算法的有效性 with 优越性。然而,本文算法也有其固有的缺点,即时间复杂度较高。以monk-2数据集为例,各类算法的运行时间如图3所示。

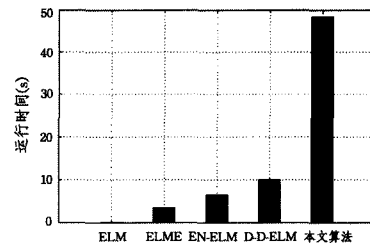
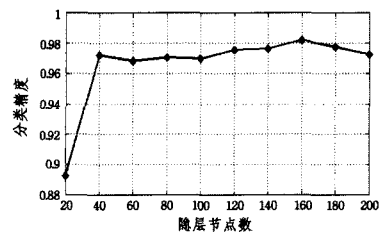
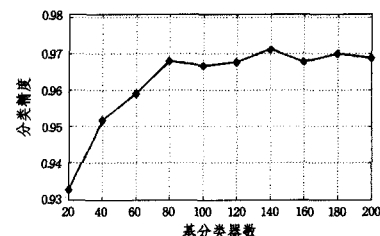


图3 各类算法在monk-2数据集上的运行时间比较

从图3可以看出,本文算法的运行时间远多于其它几种算法的,其主要原因在于本文算法搜索最优解要经历一个迭代的过程。蚁群规模、迭代次数及集成规模等参数均会对本文算法的运行时间产生较大的影响,故选择合理的参数对保证该算法的运行效率显得尤为重要。同时,在实验中发现,当上述参数均维持不变的情况下,随着训练样本规模的增长,本文算法与EN-ELM及D-D-ELM算法的运行时间几乎保持一定比例不变,但与ELM及ELME算法相比,运行时间会比会急剧增大,故可知本文算法对分类性能的提升是通过支付更多的时空开销实现的。



(a)分类精度随隐层节点数的变化



(b)分类精度随集成规模的变化

图4 本文算法性能随两个主要参数的变化

最后,考查了两个关键参数,即隐层节点数 L 及集成规

模 M 对本文算法性能的影响。仍以 monk-2 数据集为例,在保持其它参数不变的情况下,分别调整这两个参数的取值,取值范围为 20~200,以 20 为增量。实验结果如图 4 所示。

从图 4 可以看出,这两个参数对分类性能均有较大影响。同时,这两个参数对分类性能的影响也呈现出相似的规律,即当参数取值过小时,均难以取得较好的分类性能;而随着参数取值的增大,性能也呈现增长的趋势直至趋于稳定。考虑到无论是增加隐层节点数还是备选基质分类器的个数都会在很大程度上增加时空的开销,故设置合理的参数就显得尤为重要。在实际应用中,可以考虑首先采用内部 N 折交叉验证的方法对参数进行预选,然后再调用本文算法搜索最优的分类器组合。

结束语 选择性集成是近年来集成学习领域的研究热点,其优点在于不但可以进一步提升分类的性能,而且可大幅降低对学习模型的存储开销。本文以极限学习机的选择性集成成为研究对象,将其视为一个组合优化问题,并借鉴二叉蚁群优化搜索算法的思想,设计了相应的选择性集成学习算法。本文算法的优点在于:1)考查了所选取的分类器子集中各基质分类器间的相互作用;2)同时考虑用分类精度和泛化能力这两个重要指标对所选分类器子集的性能进行评价。通过与前人提出的几种算法在 12 个基准数据集上进行的比较实验表明:本文算法能有效提升分类的精度与泛化能力,且与前人算法相比,其优越性在统计方面是显著的。

当然,本文算法也具有一些缺点,如时间复杂度过高、需要调整的参数较多等。在接下来的工作中,拟进一步考查各种选择性集成策略,并结合极限学习机自身的特点,设计出时间复杂度更低、泛化能力更强的选择性集成学习算法,并用其来解决实际应用问题。

参 考 文 献

- [1] Huang G B, Zhu Q Y, Siew C K. Extreme learning machine: theory and applications[J]. Neurocomputing, 2006, 70(1-3): 489-501
- [2] Rumelhart D E, Hinton G E, Williams R J. Learning representations by back-propagation errors[J]. Nature, 1986, 323(9): 533-536
- [3] Huang G B, Wang D H, Lan Y. Extreme learning machine: a survey[J]. International Journal of Machine Learning and Cybernetics, 2011, 2(2): 107-122
- [4] Wu Jun, Wang Shi-tong, Zhao Xin. Positive and negative fuzzy rule system, extreme learning machine and image classification [J]. Journal of Image and Graphics, 2011, 11(8): 1408-1417 (in Chinese)
吴军, 王士同, 赵鑫. 正负模糊规则系统、极限学习机与图像分类 [J]. 中国图像图形学报, 2011, 11(8): 1408-1417
- [5] Minhas R, Baradaran A, Seifzadeh S, et al. Human action recognition using extreme learning machine based on visual vocabularies[J]. Neurocomputing, 2010, 73(10-12): 1906-1917
- [6] Sun Z L, Choi T M, Au K F, et al. Sales forecasting using extreme learning machine with applications in fashion retailing[J]. Decision Support Systems, 2008, 46(1): 411-419
- [7] Yan Qi-sheng, Wang Shi-tong, Zhang Yan-fei, et al. Uranium resource price based on empirical mode decomposition and extreme learning machine[J]. Control and Decision, 2014, 29(7): 1187-1192 (in Chinese)
颜七笙, 王士同, 张延飞, 等. 基于经验模式分解和极限学习机的铀资源价格预测方法[J]. 控制与决策, 2014, 29(7): 1187-1192
- [8] Li L N, Ouyang J H, Chen H L, et al. A Computer Aided Diagnosis System for Thyroid Disease Using Extreme Learning Machine[J]. Journal of Medical Systems, 2012, 36(5): 3327-3337
- [9] Zhang R, Huang G B, Sundararajan N, et al. Multicategory Classification Using an Extreme Learning Machine for Microarray Gene Expression Cancer Diagnosis [J]. IEEE/ACM Transactions on Computational Biology and Bioinformatics, 2007, 4(3): 485-495
- [10] Lu Hui-juan, An Chun-lin, Ma Xiao-ping, et al. Disagreement measure based ensemble of extreme learning machine for gene expression data classification [J]. Chinese Journal of Computers, 2013, 36(2): 341-348 (in Chinese)
陆慧娟, 安春霖, 马小平, 等. 基于输出不一致测度的极限学习机集成的基因表达数据分类[J]. 计算机学报, 2013, 36(2): 341-348
- [11] Lan Y, Soh Y C, Huang G B. Ensemble of online sequential extreme learning machine [J]. Neurocomputing, 2009, 72(13): 3391-3395
- [12] Liu N, Wang H. Ensemble based extreme learning machine[J]. IEEE Signal Processing Letters, 2010, 17(8): 754-757
- [13] Breiman L. Bagging Predictors[J]. Machine Learning, 1996, 24(2): 123-140
- [14] Zhou Z H, Wu J X, Tang W. Ensembling neural networks; Many could be better than all[J]. Artificial Intelligence, 2002, 137(1/2): 239-263
- [15] Colnari A, Dorigo M, Maniezzo V. Distributed optimization by ant colonies[C]// Proceedings of the 1st European Conference on Artificial Life, 1991. Paris, France, Cambridge, MA: MIT Press, 1991: 134-142
- [16] Yu H L, Gu G C, Liu H B, et al. A modified ant colony optimization algorithm for tumor marker gene selection[J]. Genomics, Proteomics & Bioinformatics, 2009, 7(4): 200-208
- [17] Yu H L, Ni J, Zhao J. ACO Sampling, an ant colony optimization-based undersampling method for classifying imbalanced DNA microarray data[J]. Neurocomputing, 2013, 101(3): 309-318
- [18] Alcalá-Fdez J, Fernández A, Luengo J, et al. KEEL data-mining software tool; Data set repository, integration of algorithms and experimental analysis framework[J]. Journal of Multiple-Valued Logic and Soft Computing, 2011, 17(2/3): 255-287
- [19] Demsar J. Statistical comparisons of classifiers over multiple data sets[J]. Journal of Machine Learning Research, 2006, 7(1): 1-30
- [20] Garcia S, Fernandez A, Luengo J, et al. Advanced non-parametric tests for multiple comparisons in the design of experiments in computational intelligence and data mining; experimental analysis of power[J]. Information Sciences, 2010, 180(10): 2044-2064
- [21] Garcia S, Herrera F. An extension on statistical comparisons of classifiers over multiple data sets for all pairwise comparisons [J]. Journal of Machine Learning Research, 2008, 9(12): 2677-2694