融合改进二元萤火虫算法和互补性测度的集成剪枝方法

朱旭辉[①②] 倪志伟[*①②] 倪丽萍[①②] 金飞飞[①②] 程美英[③] 李敬明[④]

[①](合肥工业大学管理学院 合肥 230009)

[②](过程优化与智能决策教育部重点实验室 合肥 230009)

[③](湖州师范学院商学院 湖州 313000)

[④](安徽财经大学管理科学与工程学院 蚌埠 233030)

摘 要：差异性和平均精度是提高分类器集成性能的两个重要指标。增加差异性势必会降低平均精度，增大平均精度一定会减小差异性。故在差异性和平均精度之间存在一个平衡状态，使得集成性能最优。为了寻找该平衡状态，该文提出融合改进二元萤火虫算法和互补性测度的集成剪枝方法。首先，采用 bootstrap 抽样方法独立训练出多个基分类器，构建原始基分类器池。其次，采用互补性测度对原始基分类器池进行预剪枝。接着，通过改进萤火虫的移动方式和搜索过程，引入重新初始化机制和跳跃行为，提出改进二元萤火虫算法。最后，采用改进二元萤火虫算法对预剪枝后的基分类器，进行进一步剪枝，选择出集成性能最优的基分类器子集合。在 5 个 UCI 数据集上的实验结果表明，较其他方法，使用较少的基分类器，获得了更优的集成性能，具有良好的有效性和显著性。

关键词：萤火虫算法；互补性测度；集成剪枝

中图分类号： TP391 **文献标识码**： A **文章编号**：1009-5896(2018)07-1643-09

DOI: 10.11999/JEIT170984

Improved Binary Glowworm Swarm Optimization Combined with Complementarity Measure for Ensemble Pruning

ZHU Xuhui[①②] NI Zhiwei[①②] NI Liping[①②] JIN Feifei[①②] CHENG Meiying[③] LI Jingming[④]

[①](*School of Management*, *Hefei University of Technology*, *Hefei* 230009, *China*)

[②](*Key Laboratory of Process Optimization and Intelligent Decision-making*, *Ministry of Education*, *Hefei* 230009, *China*)

[③](*Business School*, *Huzhou University*, *Huzhou* 313000, *China*)

[④](*School of Management Science and Engineering*, *Anhui University of Finance and Economics*, *Bengbu* 233030, *China*)

Abstract: The key to the success of an ensemble system are the diversity and the average accuracy of base classifiers. The increase of diversity among base classifiers will lead to the decrease of the average accuracy, and vice versa. So there exists a tradeoff between the diversity and the average accuracy, which makes the ensemble perform the best with respect to ensemble pruning. To find the tradeoff, Improved Binary Glowworm Swarm Optimization combined with Complementarity measure for Ensemble Pruning (IBGSOCEP) is proposed. Firstly, an initial pool of classifiers is constructed through training independently some base classifiers using bootstrap sampling. Secondly, the classifiers in the initial pool are pre-pruned using complementarity measure. Thirdly, Improved Binary Glowworm Swarm Optimization (IBGSO) is proposed by improving moving way, searching processes of glowworm, introducing re-initialization, and leaping behaviors. Finally, the optimal sub-ensemble is achieved from the base classifiers after pre-pruning using IBGSO. Experimental results on 5 UCI datasets demonstrate that IBGSODSEN can achieve better results than other approaches with less number of base classifiers, and that its effectiveness and significance.

Key words: Glowworm Swarm Optimization (GSO); Complementarity measure; Ensemble pruning

收稿日期：2017-10-23；改回日期：2018-04-02；网络出版：2018-05-10

*通信作者：倪志伟 zhwnelson@163.com

基金项目：国家自然科学基金(91546108, 71271071, 71490725, 71301041)，国家重点研发计划(2016YFF0202604)，过程优化与智能决策教育部重点实验室开放课题

Foundation Items: The National Natural Science Foundation of China (91546108, 71271071, 71490725, 71301041), The National Key Research and Development Plan (2016YFF0202604), Open Research Fund Program of Key Laboratory of Process Optimization and Intelligent Decision-making

1 引言

集成学习一直是模式识别、机器学习和数据挖掘领域一个具有挑战性的研究方向[1]。集成学习构建分类器主要分为两步：第 1 步，生成多个具有差异性的基分类器；第 2 步，综合所有基分类器的结果进行决策。大量文献表明集成学习已经成为决策的重要工具[2]。集成学习已被广泛应用到人脸识别[1]、年龄识别[3]、生物信息学[4]等实际模式分类问题中，并且取得了良好的效果。集成学习成功的关键是集成系统中的基分类器具有差异性，研究表明差异性和集成性能之间存在一定的关系[5]。至于采用何种标准衡量基分类器间的差异性，仍然是有待于解决的难题[4]。

集成学习已经取得了较大的成功，但为了提高集成性能，不断增加基分类器的规模，致使集成系统中存在大量的冗余基分类器，影响了集成的性能，增加了计算复杂度[6]。故需进行剪枝，集成部分基分类器比集成全部，可以获得更优的集成结果[7]。对于一个包含 M 个基分类器的集成系统，其拥有个 2^M-1 非空子集，这已经被证明是一个 NP 难问题[8]。为了解决该难题，学者们提出了众多集成剪枝方法，集成剪枝方法主要分为 4 类[6]：排序集成、优化集成、聚类集成和其他方法。

(1)排序集成方法：主要是通过采用某种测度，对基分类器进行排序，达到预定阈值的基分类器将会被选择，此类方法的关键在于构造衡量测度。Martínez-Muñoz 等人[8]按照差异性测度进行排序集成，通过实验表明，其较 Bagging 可以取得更优的集成效果；基于 Kappa 测度的集成剪枝方法(Kappa pruning, Kappa)被提出[8]，其集成部分 Kappa 测度较小的基分类器；Guo 等人[9]提出了基于边界测度的集成剪枝方法(Margin-based Ordered AGgregation, MOAG)，选择了一些边界测度较大的基分类器；Dai 等人[10]基于 RE(Reduce Error) 提出了 RRE，采用 RE 选择了部分分类器，然后充分利用了被遗弃的基分类器，进行综合集成。

(2)优化集成方法：主要是基于启发式算法对基分类器进行集成剪枝。较为经典的是 Zhou 等人[7]提出的 GASEN(Genetic Algorithm based Selective ENsemble)，使用 GA 优化基分类器的权重，并选择出泛化误差最小的子集成；Rokach 等人[11]提出了 CAP(Collective Agreement-based ensemble Pruning)，统计基分类器间预测结果的一致性，然后选择性能较优且一致性较低的基分类器，进行集成。

(3)聚类集成方法：主要思想源自聚类技术，该类方法主要分为两步：第 1 步，将集成系统中的基分类器分为不同的簇，则在同一簇内的基分类器的预测结果必然类似；来自不同簇的基分类器间的差异性较大。鉴于此，常用的聚类技术主要有：K-means[12]，凝聚层次聚类技术[13]，确定性退火技术[14]等被应用于集成剪枝领域中。第 2 步，为了增加差异性，分别剪枝每个聚类簇中的基分类器，如 Bakker 等人[14]使用在每个聚类簇几何中心的基分类器，构成剪枝后的基分类器集合。

(4)其他集成方法：不属于上述 3 类的其他剪枝方法。Martínez-Muñoz 等人[8]采用 Adaboosting 剪枝基于 Bagging 生成的基分类器；Lu 等人[4]提出了基于双错测度的极限学习机(Extreme Learning Machine, ELM)集成方法(DF-D)，剔除双错测度不在置信区间中的基 ELM，并进行集成；Zhou 等人[15]提出了 EP-FP(Ensemble Pruning algorithm based on Frequent Patterns)，挖掘基分类器的布尔矩阵中的频繁模式，最后进行选择性集成；Cavalcanti 等人[16]提出了 DivP(combining Diversity measures for ensemble Pruning)，采用遗传算法(Genetic Algorithm, GA)优化不同差异性测度，形成组合测度，并使用图着色方法进行集成剪枝；Ykhlef 等人[6]提出了 SCG-P(ensemble Pruning based on Simple Coalitional Games)，通过计算每个基分类器的班扎夫权力系数，计算出最小赢得联盟，从而得到集成性能最优的子集成。

上述各类方法均单独采用差异性测度或元启发式搜索算法，进行集成剪枝。单独采用差异性测度，结合不同的策略进行剪枝，只能剪枝部分冗余基分类器，无法实现精确剪枝；单独采用元启发式搜索算法进行剪枝，无法从数目十分庞大的子集中，穷尽搜索出集成性能较优的子集成。为了克服上述缺陷，本文将差异性测度和元启发式算法结合起来，进行集成剪枝。先采用差异性测度对集成系统中基分类器池进行预剪枝，在保留具有较大差异性的基分类器情况下，大幅剔除冗余基分类器，减小集成规模，极大限度地降低二次剪枝的复杂度；再采用元启发式搜索算法进行进一步剪枝。在选择差异性测度方面，考虑到互补性测度在衡量基分类器差异性方面的性能较为突出[8]，故选择互补性测度进行预剪枝；在选择元启发式搜索算法方面，考虑到萤火虫算法(Glowworm Swarm Optimization, GSO) 具有考参数少、易实现、鲁棒性强等优点[17]，故采用 GSO 作为搜索策略。因此，提出了融合改进二元萤火虫算法和互补性测度的集成剪枝方法。

本文的主要贡献如下：(1)提出了一种融合改进二元萤火虫算法和互补性测度的集成剪枝方法(Improved Binary GSO combined with Complementarity measure Ensemble Pruning, IBGSOCEP)，通过采用互补性测度进行预剪枝，再使用改进二元萤火虫算法进行进一步精确剪枝，从而显著提高了集成性能。(2)发现了互补性测度可以有效地剔除集成系统中冗余的基分类器，保留性能较好、差异性较大的基分类器，可以应用于基分类器的预剪枝。(3)提出了改进二元萤火虫算法(IBGSO)，通过改进萤火虫的移动方式和搜索过程，引入重新初始化机制和跳跃行为，较为显著地提高了算法在二元离散型空间中的搜索性能。(4)在 5 个 UCI 数据集上的结果表明了，提出的 IBGSOCEP，较其他集成剪枝方法，使用较少的基分类器，获得了更优的集成性能，进一步表明了 IBGSOCEP 的有效性和显著性。(5)进一步通过实验表明了，提出的 IBGSO，相较于其他二元启发式算法，具有更优的收敛速度和精度。

2　互补性测度

基分类器间的差异性在集成学习中是至关重要的。若基分类器间无差异，则集成的结果必然相同，则集成学习失去了实际意义。增加基分类器间存在较大的差异性，则必然会降低单个基分类器的性能[5]。换言之，需降低单个基分类器的性能，以提高差异性。如何找到差异性和性能之间的平衡？这是一个有待于解决的问题[4]。鉴于差异性的决定性作用，需找一种有效的衡量测度。下面介绍一种互补性测度[8]，以保证所选择的基分类器具有较大的差异性。

Martínez-Muñoz 等人[8]提出了互补性测度。首先选择性能最优的基分类器；再不断地挑选出与当前已选择的基分类器集合之间互补性最大的基分类器；最后将其加入到已选择的基分类器集合中，以实现所选择的基分类器具有较大的差异性。互补性测度选择基分类器的过程如下：

假设有 N 个样本 $X=\{x_1,x_2,\cdots,x_N\}$，其实际类别 $Y=\{y_1,y_2,\cdots,y_N\}$，所有样本类别 $C=\{c_1,c_2,\cdots,c_l\}$，M 个基分类器集合 $F=\{f_1,f_2,\cdots,f_M\}$，采用多数投票法，对 M 个基分类器集合 F 进行集成过程为

$$E_F(x)=\arg\max_{c\in C}\sum_{i=1}^{M}I\left(f_i(x)=c\right) \tag{1}$$

其中，x 表示样本，$c(c\in C)$ 表示所有样本类别 C 中的一种，$E_F(x)$ 表示对样本 x 的集成结果，$f_i(x)$ 表示基分类器 f_i 样本 x 上的分类结果，$I(\bullet)$ 为指示函数(若 $\bullet$ 为真，则 $I(\bullet)=1$；否则，$I(\bullet)=0$)，$\mathrm{argmax}(f(x))$ 表示使得 $f(x)$ 最大的 x。

假设当前所选择的基分类器集合有 $u(1\le u\le M)$ 个基分类器 S_u，下面选择第 $u+1$ 个基分类器加入到 S_u 中，构成 S_{u+1}，如式(2)：

$$S_{u+1}=\mathrm{argmax}\sum_{(x,y)\in(X,Y)}I\left(f_k(x)=y\cap E_{S_u}(x)\neq y\right) \tag{2}$$

其中，$f_k\in F\setminus S_u$，$f_k\in F\cap f_k\notin S_u$。

采用互补性测度所选择的基分类器集合，既保证了基分类器间具有较大的差异性，又剔除了性能相似的基分类器。因此，本文采用互补性测度，对基分类器池进行预剪枝。预剪枝大幅度减少了基分类器的规模，显著降低了集成剪枝的计算复杂度，提高了二元启发式搜索算法的搜索效率。

3　改进二元萤火虫算法

萤火虫算法是通过模仿自然界中萤火虫的发光行为，而提出的元启发式算法[18]。在 GSO 中，随机分布在整个搜索空间中的萤火虫，自身带有一定量的荧光素，萤火虫个体在自视野范围内，不断地向比自身更亮的萤火虫靠拢，进而实现群体优化，最终收敛到全局最优解上。其参数包括：种群规模 g，荧光素挥发因子 ρ，荧光素更新率 γ，动态决策域更新率 β，在第 t 次迭代萤火虫 $\boldsymbol{X}_i(t)$ 决策域内的萤火虫 $N_i(t)$，领域内所含萤火虫数目阈值 n_t，移动步长 s，萤火虫个体的感知半径 r_s。其具体执行步骤见文献[17]。

3.1 IBGSO

为了使得 GSO 适合处理二元空间中的离散型问题，首先，为了使得 GSO 可以在二元离散型空间中进行搜索，对萤火虫的移动方式进行改进；其次，为了避免算法的早熟收敛现象，引入重新初始化机制和跳跃行为；最后，为了提高算法的收敛速度和精度，对算法的搜索过程进行改进。因此，提出了 IBGSO。

3.1.1 重新初始化机制　萤火虫算法在迭代一定次数后，易陷入局部最优，影响算法的收敛性，出现早熟现象。为了防止算法陷入局部最优，每迭代 T 次数后，重新初始化萤火虫种群。

3.1.2 移动方式的改进　萤火虫在二元离散型空间中搜索，固定步长并不适用，为了简化萤火虫的移动方式，提高萤火虫搜索效率，向目标萤火虫方向，以一定的概率更新当前萤火虫的每一维[17]。在萤火虫算法的第 t 次迭代中，当前萤火虫 $\boldsymbol{X}_i(t)=[x_{i1}\ x_{i2}\cdots x_{in}]$，目标萤火虫 $\boldsymbol{X}_j(t)=[x_{j1}\ x_{j2}\cdots x_{jn}]$，其中 n 为萤火虫解空间的维数，则当前萤火虫位置

更新方式如式(3)：

$$x_{ik}(t+1)=\begin{cases}x_{ik}(t), & r\le p_1\\ x_{jk}(t), & p_1<r<p_2\\ \text{round}(\text{rand}), & r\ge p_2\end{cases}\tag{3}$$

其中，p_1,p_2 为选择参数，$0<p_1<p_2<1$，r 为(0,1)间的随机数，$k=1,2,\cdots,n$。

3.1.3 搜索过程的改进 为了提高算法的收敛速度和精度，改进对萤火虫的搜索过程。在第 t 次迭代中，当前萤火虫 $\boldsymbol{X}_i(t)$ 向公告板内的全局最优位置、决策域内的最优萤火虫和决策域中的某一随机位置，按照式(3)分别移动 1 次，移动后的位置分别记为 $\boldsymbol{X}_i^{'}(t+1),\boldsymbol{X}_i^{''}(t+1),\boldsymbol{X}_i^{'''}(t+1)$。将 $\boldsymbol{X}_i^{'}(t+1),\boldsymbol{X}_i^{''}(t+1),\boldsymbol{X}_i^{'''}(t+1)$ 中的最优位置赋值给 $\boldsymbol{X}_i(t+1)$。

3.1.4 跳跃行为 为了避免算法陷入局部最优，增加种群的多样性，引入跳跃行为，使得萤火虫可以探索领域内的其他位置。在每次迭代前，判断每只萤火虫的目标函数值是否与公告板最优值相等。若相等，则随机搜索其决策领域内的任一位置；若不等，则不处理。

3.1.5 公告板 为了记录萤火虫算法搜索过程中的最优值，故而增设公告板。萤火虫算法每一次迭代后，若种群中最优萤火虫目标函数值优于公告板中最优值，则更新公告板。

3.2 IBGSO 步骤

IBGSO 算法示于表 1。

表 1 IBGSO 算法

输入：萤火虫参数
输出：全局最优值
步骤 1 初始化参数；
步骤 2 随机初始化种群，并将种群中最优萤火虫的目标函数值和位置赋值给公告板；
步骤 3 判断 MOD(t,T)=0 是否成立。若成立，则重新初始化种群；否则，不处理；
步骤 4 判断个体萤火虫目标函数值是否与公告板最优值相等。若相等，则执行跳跃行为；
步骤 5 更新个体萤火虫荧光素值和动态决策域半径；
步骤 6 选择当前萤火虫领域集内荧光素值最大的萤火虫；
步骤 7 当前萤火虫分别向公告板最优位置、荧光素值最大的萤火虫和决策域内的随机位置，按照式(3)更新位置，并将其中最优位置赋值当前萤火虫；
步骤 8 若当前萤火虫优于公告板，则更新公告板；
步骤 9 循环步骤 3−步骤 8，若满足循环结束条件，转步骤 10；否则，继续；
步骤 10 输出公告板，即全局最优值。

3.3 IBGSO 复杂度分析

萤火虫算法的收敛性问题，在文献[17]中已证明。下面对 IBGSO 的时间和空间复杂度进行分析，从理论上分析算法的可行性和有效性。

3.3.1 时间复杂度分析 在实际应用中，通常采用时间复杂度的渐进法，估算算法执行的效率。在分析算法的时间复杂度时，把算法的关键操作，例如：加、减、乘、除、比较等操作，指定为基本操作，通常把算法执行基本操作的次数定义为算法的复杂度。假设 IBGSO 初始时刻种群规模为 g，每只萤火虫的维数为 n，根据 3.2 节算法的步骤分析算法的时间复杂度。具体步骤示于表 2。

表 2 分析算法时间复杂度步骤

步骤 1	初始化 g 只萤火虫的时间复杂度为 $O(g)$；
步骤 2	初始化 g 只萤火虫，需计算 $g\times n$ 次，时间复杂度为 $O(g\times n)$；
步骤 3	重新初始化 g 只萤火虫最多需计算 $g\times n$ 次，时间复杂度最高为 $O(g\times n)$；
步骤 4	g 只萤火虫跳跃行为最多需计算 $g\times n$ 次，时间的复杂度为 $O(g\times n)$；
步骤 5	每只萤火虫荧光素值更新需计算 1 次，动态决策域半径更新需计算 g 次，g 只萤火虫的时间复杂度为 $O\left(g^2\right)$；
步骤 6	每只萤火虫确定其决策域需计算 g 次，选择目标萤火虫最多需计算 g 次，g 只萤火虫的时间复杂度为 $O\left(g^2\right)$；
步骤 7	每只萤火虫分别向全局最优位置、决策域内的最优萤火虫、随机位置移动 1 次均需计算 $g\times n$ 次，时间复杂度为 $O(g\times n)$；
步骤 8	当前萤火虫与公告板判断比较 1 次，更新 1 次，时间复杂度为 $O(1)$。

综上所述：IBGSO 每次迭代后的时间复杂度为 $O\left(g^2\right)$，经过 $T_{\max}$ 次迭代后，整个算法的时间复杂度为 $O\left(T_{\max}\times g^2\right)$。

3.3.2 空间复杂度分析 存储每只萤火虫的荧光素值、决策域半径所需空间均为 g；存储长度为 n 的实数位置所需空间 $g\times n$；公告板存储全局最优值和最优个体所需空间为 n；存储其他参数所需空间为常数；综上所述，整个计算过程所需存储空间为 $O(g\times n)$。

4 IBGSOCEP

4.1 基分类器的生成

本文采用 Bagging 中 bootstrap 抽样方法抽取 M 次训练样本，构成 M 个的训练样本，分别采用基分类器独立训练，则可以获得 M 个基分类器，即构成原始基分类器池。

4.2 预剪枝

对于一个包含 M 个基分类器池，其有 2^M-1 个非空子集，这也是一个 NP 难问题。当基分类器的规模较大时，则采用 IBGSO 很难搜索到基分类器最优子集合。因此，需对基分类器池进行预剪枝，剔除大量性能差、差异性小的基分类器，减少基分类器的数目，以显著提高 IBGSO 的搜索效率。本文采用互补性测度，对基分类器池进行预剪枝。关于采用互补性测度保留的基分类器数目 M' 的确定，见第 5.2 节分析。预剪枝后，采用 IBGSO 进行进一步剪枝，搜索出集成精度最优的基分类器集合。

4.3 二次剪枝

4.3.1 编码方式　基分类器池中的 M 个基分类器 $F=\{f_1,f_2,\cdots,f_M\}$，在预剪枝后，保留了 M' 个基分类器，记为 $F'=\left\{f_1^{'},f_2^{'},\cdots,f_{M'}^{'}\right\}$。本文采用二进制编码中 1 或 0 表示基分类器的选取与否。萤火虫 $\boldsymbol{X}=\left[x_1\ x_2\cdots x_{M'}\right]$，$x_i=1$ 表示选择第 i 个基分类器；否则，不选择。例如：$\boldsymbol{X}=[1\,0\,1\,1\,0]$ 表示选择第 1, 3, 4 个基分类器。

4.3.2 适应度函数构造　集成剪枝的适应度函数构造，如式(4)所示：

$$\mathrm{Fn}=A \tag{4}$$

其中，A 表示集成精度，$A=\dfrac{1}{m}\sum_{j=1}^{m}\mathrm{Acc}\left(\hat{y}_j,y_j\right)$，$\mathrm{Acc}\left(\hat{y}_j,y_j\right)=\begin{cases}1, & \hat{y}_j=y_j\\ 0, & \hat{y}_j\neq y_j\end{cases}$，$m$ 表示测试样本的数目，$\hat{y}_j$ 和 y_j 分别表示在第 j 个测试样本上的集成结果和实际类别。适应度值越高，则表示集成精度越高。

4.3.3 不可行解处理方式　IBGSO 在搜索最优解过程中，可能会出现解元素全为 0 或全为 1 的情况（表示所有基分类器均不选择或均选择），则认为其为不可行解，本文通过重新初始化，以处理该不可行解。

4.4 IBGSOCEP 的步骤

IBGSOCEP 的基本步骤如下：

输入：训练集，测试集，IBGSO 参数。

输出：集成剪枝结果。

步骤 1　采用 bootstrap 抽样方法训练多个基分类器，构造原始基分类器池；

步骤 2　运用互补性测度对原始基分类器池进行预剪枝；

步骤 3　采用 IBGSO 对预剪枝后的基分类器集合进行二次剪枝；

步骤 4　输出公告板，即集成剪枝结果。

5　实验结果及分析

为了验证 IBGSOCEP 的有效性，从 UCI 数据库中选取 5 个数据集进行测试，如表 3。鉴于 ELM 具有不稳定和学习效率极快的特性[4]，适合作为基分类器[16]，本文将 ELM 作为基分类器。采用 5-折交叉验证技术将数据集随机分成 5 份，其中 4 份作为训练集，1 份作为测试集。

表 3　UCI 数据集

数据集	实例个数	属性个数	类别
Column	310	6	2
Forest	523	27	4
Wineq-r	1599	11	6
Segment	2310	19	7
Landsat	6435	36	6

5.1 实验环境和参数设置

本文实验环境所涉及的代码均采用 Matlab R2012a 软件编写，编译运行的 PC 机参数为：32 位 Windows 7 操作系统、Intel(R) Core(TM)2 E7500 2.93 GHz CPU, 4.00 GB 内存。IBGSO 算法参数设置[17]：荧光素挥发因子 $\rho=0.4$，荧光素更新率 $\gamma=0.6$，动态决策域更新率 $\beta=0.08$，领域阈值 $n_t=5$，$p_1=0.15$, $p_2=0.85$，其余参数将在 5.4 节中进行分析。为了增强实验结果的稳定性，实验结果均取独立重复 30 次试验的平均值。

5.2 预剪枝

为了确定采用互补性测度保留的基分类器数目 M'，本文通过实验分析，在 Column 数据集上，不同规模的基分类器(规模为 100, 200)条件下，运用互补性测度保留的基分类器数目与集成精度之间的变化趋势，如图 1 所示(但由于文章篇幅的限制，本文仅在 Column 上进行展示)。通过图 1 可以看出，随着所保留的基分类器数目的增加，集成精度先上升，后下降。集成精度先上升的原因是：集成初期由于基分类器数目太少，集成系统缺乏差异性，影响了集成精度；后下降的原因是：当基分类器达到一定数目后，继续增加基分类器的数目，使得集成系统中存在大量的冗余基分类器，造成集成精度下降。同时也可以看出，当基分类器数目达到 25 以前，集成精度可以达到最高值；当基分类器数目达到 25 以后，集成精度下降趋势较为明显。因此，采用互补性测度保留的基分类器数目取 25，即 $M'=25$。最后再采用 IBGSO 对所保留的基分类器进行二次剪枝。

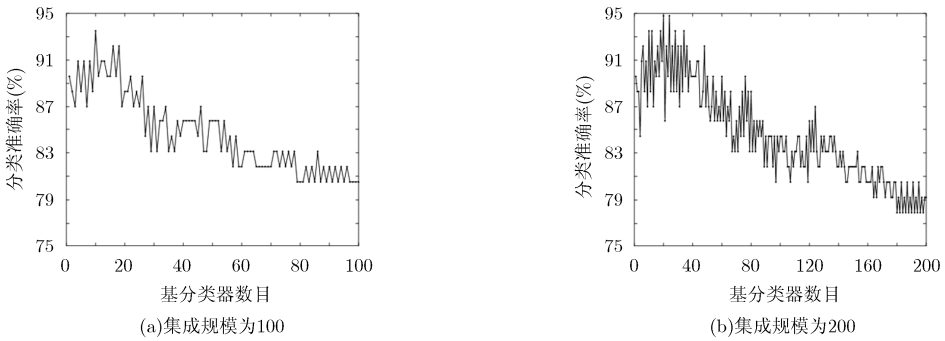

图 1 在 Column 数据集上不同规模基分类器按互补性测度排序集成性能分析

5.3 实验结果分析

表 4，表 5 呈现了 IBGSOCEP 在不同规模基分类器(50, 100, 150, 200, 250, 300)下的集成结果。从表 4，表 5 看出，IBGSOCEP 在 5 个 UCI 数据集上，集成精度均明显高于基分类器池中的最优和平均精度。表 6，表 7 对 IBGSOCEP 与 Bagging[19]在不同规模基分类器下，集成结果及所选择的基分类器数目进行了对比分析，其中 'n' 表示算法所选取的基分类器的数目，表明了 IBGSOCEP 相较于 Bagging，使用了更少数目的基分类器，获得了更高的集成精度。

为了验证 IBGSOCEP 的集成性能，下面与 DF-D[4], GASEN[7], RRE[10], DivP[16], SCG-P[6]进行对比分析。DF-D 剔除双侧测度在置信区间之外的基 ELM; GASEN 采用 GA 优化基分类器的权重，使得集成误差最小化； RRE 充分利用了集成剪枝中，未被选择的基分类器；DivP 通过 GA 对多个差异性测度进行组合优化，并采用图着色方法进行剪枝；SCG-P 根据基分类器的班扎夫权力系数，得到最小赢得联盟。上述方法在实验中均按照文献中描述的步骤进行。

通过表 6，表 7 可以看出，IBGSOCEP 在基分类器规模达到 200 时，其集成性能较好，再继续增加基分类器的数目，集成精度增加的幅度很小。因此，建议基分类器池的规模取 200。因此，本文在基分类器规模为 200 的条件下，与其他方法在集成精度和所选择的基分类器数目方面进行对比分析，如表 8 所示。从表 8 可以看出，本文方法的集成性能均优于其他方法，所选取的基分类器数目明显少于其他方法(除了 DivP)。

Bagging 在集成时，集成系统中存在大量冗余基分类器，影响了集成性能；DF-D, RRE 和 SCG-P 在集成剪枝，仅采用了不同的差异性测度，进行了集成剪枝，其仅能剪枝部分冗余基分类器，无法实现精确剪枝；GASEN 在集成剪枝时，由于基分类器池的规模较大，其子集合数目过于庞大，GA 无

表 4 IBGSOCEP 在不同规模基分类器(50, 100, 150)下的集成精度(%)

数据集	50				100				150			
		最高	平均	最低		最高	平均	最低		最高	平均	最低
Column	**92.86**	87.27	78.13	66.36	**93.90**	87.73	77.86	64.61	**94.87**	88.31	77.97	63.64
Forest	**94.36**	88.75	80.26	66.79	**95.26**	88.19	78.29	62.56	**95.61**	88.69	78.99	62.99
Wineq-r	**62.09**	58.84	54.41	49.94	**62.88**	58.56	54.46	49.34	**63.88**	59.13	54.40	49.00
Segment	**90.30**	82.80	76.07	69.10	**90.97**	83.27	75.82	68.50	**91.80**	83.80	75.88	67.50
Landsat	**81.89**	73.38	66.02	53.99	**82.80**	74.77	66.77	53.88	**82.99**	74.93	66.60	52.47

表 5 IBGSOCEP 在不同规模基分类器(200, 250, 300)下的集成精度(%)

数据集	200				250				300			
		最高	平均	最低		最高	平均	最低		最高	平均	最低
Column	**95.71**	88.57	78.05	63.51	**95.71**	88.57	78.07	62.79	**95.91**	88.90	78.02	62.21
Forest	**94.89**	88.78	78.13	61.83	**95.42**	89.45	78.96	60.05	**96.29**	90.19	79.41	58.94
Wineq-r	**64.22**	59.50	54.50	49.13	**64.81**	59.63	54.42	49.19	**64.69**	59.69	54.45	49.19
Segment	**92.00**	83.47	75.98	66.70	**92.27**	84.07	75.90	66.27	**92.33**	84.17	75.98	66.60
Landsat	**84.53**	75.94	67.26	49.94	**83.32**	75.65	66.72	48.29	**84.61**	76.28	67.03	49.10

表 6 在不同规模基分类器(50, 100, 150)下 IBGSOCEP 与 Bagging 对比分析

数据集	50				100				150			
	Bagging	n	IBGSOCEP	n	Bagging	n	IBGSOCEP	n	Bagging	n	IBGSOCEP	n
Column	80.91	50	**92.86**	**11**	80.39	100	**93.90**	**12**	80.65	150	**94.87**	**12**
Forest	89.09	50	**94.36**	**12**	88.31	100	**95.26**	**13**	89.18	150	**95.61**	**14**
Wineq-r	54.84	50	**62.09**	**9**	55.34	100	**62.88**	**12**	55.19	150	**63.88**	**11**
Segment	85.43	50	**90.30**	**12**	85.43	100	**90.97**	**13**	85.40	150	**91.80**	**13**
Landsat	77.98	50	**81.89**	**15**	78.96	100	**82.80**	**16**	78.55	150	**82.99**	**16**

表 7 在不同规模基分类器(200, 250, 300)下 IBGSOCEP 与 Bagging 对比分析

数据集	200				250				300			
	Bagging	n	IBGSOCEP	n	Bagging	n	IBGSOCEP	n	Bagging	n	IBGSOCEP	n
Column	80.52	200	**95.71**	**13**	80.26	250	**95.71**	**13**	79.94	300	**95.91**	**13**
Forest	88.10	200	**94.89**	**13**	88.75	250	**95.42**	**14**	89.45	300	**96.29**	**13**
Wineq-r	55.13	200	**64.22**	**11**	55.13	250	**64.81**	**11**	55.25	300	**64.69**	**10**
Segment	85.90	200	**92.00**	**14**	85.83	250	**92.27**	**14**	85.80	300	**92.33**	**13**
Landsat	80.11	200	**84.53**	**16**	78.90	250	**83.32**	**13**	79.92	300	**84.61**	**15**

表 8 与其他方法在集成精度和集成规模方面对比分析

数据集	IBGSOCEP	n	DF-D	n	GASEN	n	RRE	n	DivP	n	SCG-P	n
Column	**95.71**	13	87.19	107	83.38	49	92.16	22	92.68	**9**	90.46	67
Forest	**94.89**	13	89.86	106	91.06	92	93.94	24	92.37	**7**	89.21	40
Wineq-r	**64.22**	11	56.90	104	58.46	76	62.19	26	61.60	**8**	60.83	61
Segment	**92.00**	14	88.10	105	87.58	87	90.85	23	89.63	**11**	86.95	67
Landsat	**84.53**	16	80.77	113	79.98	99	84.35	36	81.64	**8**	79.96	88

法搜索到性能较优的集成子集合；DivP 集成时，DivP 采用图着色理论，剔除了过多的基分类器，使得集成系统中的基分类器数目太少，以至于缺乏多样性，影响了最终的集成精度。

5.4 参数分析

在 IBGSOCEP 中，采用了 IBGSO 进行优化，为了提高 IBGSO 的性能，对 IBGSO 的主要参数进行分析，包括重新初始化迭代次数、最大迭代次数、种群规模、决策域半径初始值和最大值。由于文章篇幅限制，本节以 Column 数据集为例，在 IBGSOCEP 中，原始基分类器池的规模为 200，预剪枝后剩下基分类器数目为 25，然后采用 IBGSO 进行优化，分析结果(独立重复 30 次取均值)如下。

为了验证本文提出的 IBGSO 的性能，将 IBGSO 分别与 IDGSO(Improved Discrete Glowworm Swarm Optimization)[17], BAFSA (Binary Artificial Fish Swarm Algorithm)[20] 和 GA[11]，进行对比分析，如图 2(a)所示。图 2(a)展示了在 Column 数据集上，对原始规模为 200 的基分类器池预剪枝后，分别采用不同的二元启发式算法进行集成剪枝的结果分析。大多种算法的种群规模均为 25，其余参数均按照文献中的描述设置。由图 2(a)可知，二元启发式算法的性能均与迭代次数成正相关，同时可以看出，IBGSO 较其他方法，具有更优的收敛速度和精度。同时也可以看出，当迭代次数达到 500 后，再继增加迭代次数，集成精度提升幅度较小，性能趋于平稳，建议最大迭代次数取 500。

图 2(b)所示，对 IBGSO 的种群规模进行分析，当种群规模达到 25 后，算法性能较为平稳，若继续增加种群规模，算法性能提升幅度较小，反而会大幅增加算法的时间和空间复杂度。建议种群规模取 25。

图 2(c)分析了 IBGSO 的初始决策域半径，预剪枝后的基分类器数目为 25，故初始决策域半径变化范围为 [1,25]。若初始决策域半径较小时，决策领域内的萤火虫数目较少，若其较大时，易陷入局部最优，均影响算法性能。从图 2(c)可知，当初始决策域半径为 11 时，算法性能最佳。建议初始决策域半径取 11。

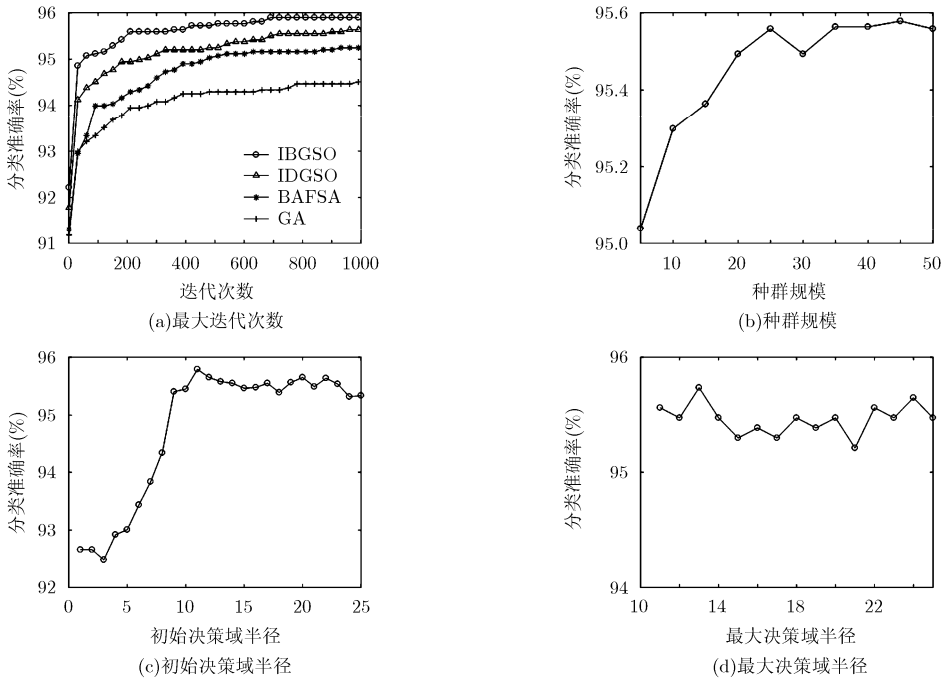

图 2 参数对算法性能影响分析

图 2(d)分析了 IBGSO 的最大决策域半径。最大决策域半径需大于初始半径，故最大决策域半径变化范围为[11,25]。从图 2(d)可知，当最大决策域半径取 13 时，IBGSO 的性能最优。建议最大决策域半径取 13。

6 结束语

基分类器间的差异性和平均精度是影响集成性能的两个重要指标。在集成系统中存在大量冗余的基分类器，且集成剪枝问题又是一个 NP 难问题。本文通过融合互补性测度和 IBGSO，提出了 IBGSOCEP。本文采用互补性测度进行预剪枝，大幅剔除了冗余的基分类器；改进了 GSO 的移动方式和搜索过程，引进了重新初始化机制、跳跃行为和公告板，提出了 IBGSO，提升了算法的收敛速度和精度；融合了互补性测度和 IBGSO，进行集成剪枝，显著提高了集成剪枝的性能。在 5 个 UCI 数据集上的实验结果，表明了算法的有效性和显著性，为集成剪枝领域提供了新的研究思路。下一步研究的工作是研究其他差异性测度，并应用于预剪枝，为基于元启发式搜索算法的集成剪枝技术提供性能好、差异性大的基分类器。

参 考 文 献

[1] BASHBAGHI S, GRANGER E, SABOURIN R, *et al*. Dynamic ensembles of exemplar-SVMs for still-to-video face recognition[J]. *Pattern Recognition*, 2017, 69(C): 61–81. doi: 10.1016/j.patcog.2017.04.014.

[2] 刘家辰, 苗启广, 曹莹, 等. 基于混合多样性生成与修剪的集成单类分类算法[J]. 电子与信息学报, 2015, 37(2): 386–393. doi: 10.11999/JEIT140161.
LIU Jiachen, MIAO Qiguang, CAO Ying, *et al*. Ensemble one-class classifiers based on hybrid diversity generation and pruning[J]. *Journal of Electronics & Information Technology*, 2015, 37(2): 386–393. doi: 10.11999/JEIT140161.

[3] LI Kai, XING Junliang, HU Weiming, *et al*. D2C: Deep cumulatively and comparatively learning for human age estimation[J]. *Pattern Recognition*, 2017, 66(6): 95–105. doi: 10.1016/j.patcog.2017.01.007.

[4] LU Huijuan, AN Chunlin, ZHENG Enhui, *et al*. Dissimilarity based ensemble of extreme learning machine for gene expression data classification[J]. *Neurocomputing*, 2014, 128(5): 22–30. doi: 10.1016/j.neucom.2013.02.052.

[5] 杨春, 殷绪成, 郝红卫, 等. 基于差异性的分类器集成: 有效性分析及优化集成[J]. 自动化学报, 2014, 40(4): 660–674. doi: 10.3724/SP.J.1004.2014.00660.
YANG Chun, YIN Xucheng, HAO Hongwei, *et al*. Classifier ensemble with diversity: Effectiveness analysis and ensemble optimization[J]. *Acta Automatica Sinica*, 2014, 40(4): 660–674. doi: 10.3724/SP.J.1004.2014.00660.

[6] YKHLEF H and BOUCHAFFRA D. An efficient ensemble pruning approach based on simple coalitional games[J]. *Information Fusion*, 2017, 34(C): 28–42. doi: 10.1016/j.inffus.2016.06.003.

[7] ZHOU Zhihua, WU Jianxin, and TANG Wei. Ensembling neural networks: many could be better than all[J]. *Artificial Intelligence*, 2002, 137(1): 239–263. doi: 10.1016/S0004-

3702(02)00190-X.

[8] MARTÍNEZ-MUÑOZ G, HERNANDEZ LOBATO D, and SUAREZ A. An analysis of ensemble pruning techniques based on ordered aggregation[J]. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2009, 31(2): 245–259. doi: 10.1109/TPAMI.2008.78.

[9] GUO Li and BOUKIR S. Margin-based ordered aggregation for ensemble pruning[J]. *Pattern Recognition Letters*, 2013, 34(6): 603–609. doi: 10.1016/j.patrec.2013.01.003.

[10] DAI Qun, ZHANG Ting, and LIU Ningzhong. A new reverse reduce-error ensemble pruning algorithm[J]. *Applied Soft Computing*, 2015, 28(3): 237–249. doi: 10.1016/j.asoc.2014.10.045.

[11] ROKACH L. Collective-agreement-based pruning of ensembles[J]. *Computational Statistics and Data Analysis*, 2009, 53(4): 1015–1026. doi: 10.1016/j.csda.2008.12.001.

[12] LAZAREVIC A and OBRADOVIC Z. Effective pruning of neural network classifier ensembles[C]. International Joint Conference on Neural Networks, Washington DC, USA, 2001: 796–801. doi: 10.1109/IJCNN.2001.939461.

[13] GIACINTO G, ROLI F, and FUMERA G. Design of effective multiple classifier systems by clustering of classifiers[C]. International Conference on Pattern Recognition, Barcelona, Spain, 2000: 160–163. doi: 10.1109/ICPR.2000.906039.

[14] BAKKER B and HESKES T. Clustering ensembles of neural network models[J]. *Neural Network*, 2003, 16(2): 261–269. doi: 10.1016/S0893-6080(02)00187-9.

[15] ZHOU Hongfang, ZHAO Xuehan, and WANG Xiao. An effective ensemble pruning algorithm based on frequent patterns[J]. *Knowledge-Based Systems*, 2014, 56(C): 79–85. doi: 10.1016/j.knosys.2013. 10.024.

[16] CAVALCANTI G D C, OLIVEIRA L S, NOURA T J M, *et al*. Combining diversity measures for ensemble pruning[J]. *Pattern Recognition Letters*, 2016, 74(C): 38–45. doi: 10.1016/j.patrec.2016.01.029.

[17] 倪志伟, 张琛, 倪丽萍. 基于萤火虫群优化算法的选择性集成雾霾天气预测方法[J]. 模式识别与人工智能, 2016, 29(2): 143–153. doi: 10.16451/j.cnki.issn1003-6059.201602006.
NI Zhiwei, ZHANG Chen, and NI Liping. Haze forecast method of selective ensemble based on glowworm swarm optimization algorithm[J]. *Pattern Recognition and Artificial Intelligence*, 2016, 29(2): 143–153. doi: 10.16451/j.cnki.issn1003-6059.201602006.

[18] MARINAKI M and MARINAKI Y. A glowworm swarm optimization algorithm for the vehicle routing problem with stochastic demands[J]. *Expert Systems with Applications*, 2016, 46(C): 145–163. doi: 10.1016/j.eswa.2015.10.012.

[19] BREIMAN L. Bagging predictors[J]. *Machine Learning*, 1996, 24(2): 123–140. doi: 10.1023/A:1018054314350.

[20] SINGHAL P K, NARESH R, and SHARMA V. Binary fish swarm algorithm for profit-based unit commitment problem in competitive electricity market with ramp rate constraints [J]. *IET Generation, Transmission & Distribution*, 2015, 9(13): 1697–1707. doi: 10.1049/iet-gtd.2015.0201.

朱旭辉：男，1991 年生，博士生，研究方向为进化计算和机器学习.

倪志伟：男，1963 年生，教授，研究方向为人工智能、机器学习和云计算.

倪丽萍：女，1981 年生，副教授，研究方向为分形数据挖掘、人工智能和机器学习.

金飞飞：男，1988 年生，博士生，研究方向为智能决策和智能计算.

程美英：女，1983 年生，讲师，研究方向为智能计算和数据挖掘.

李敬明：男，1979 年生，讲师，研究方向为智能计算和数据挖掘.