控制与决策
Control and Decision

基于评分机制的类贪心森林优化特征选择算法

王霞, 张珊, 王勇, 王卓然

引用本文:
王霞, 张珊, 王勇, 等. 基于评分机制的类贪心森林优化特征选择算法[J]. 控制与决策, 2025, 40(2): 517-527.

在线阅读 View online: https://doi.org/10.13195/j.kzyjc.2023.1545

基于评分机制的类贪心森林优化特征选择算法

王 霞[1,2†]，张 珊[1,2]，王 勇[1,2]，王卓然[1,2]

(1. 云南民族大学 电气信息工程学院， 昆明 650504；
2. 云南民族大学 云南省无人自主系统重点实验室，昆明 650504)

摘 要: 森林优化特征选择算法 (FSFOA) 具有良好的分类性能和维度缩减能力, 但其初始化森林的质量参差不齐, 局部播种和全局播种的随机性较大, 且适应度评估代价较高导致计算效率较低. 针对上述问题, 提出一种基于评分机制的类贪心森林优化特征选择算法 (FSGLFOA-SM). 首先, 以每维决策变量的分类精度为其得分构建评分机制, 提出类贪心初始化策略以生成较优质的初始化森林; 其次, 提出基于评分比较的类贪心局部播种策略, 使评分相对较高的决策变量获得更大的局部播种概率; 然后, 在全局播种阶段提出类贪心遗传算子播种策略, 对候选森林择优重建并进行遗传、类贪心交叉和变异操作, 以保留评分较高的特征维度, 有利于提高全局播种阶段的分类准确率; 最后, 为解决昂贵适应度评估带来的计算效率低下问题, 建立历史数据库, 在适应度评估前先进行库内查找, 减少了重复解个体的计算量. 实验结果表明, 相比 9 个对比算法, FSGLFOA-SM 在 16 个 UCI 数据集上的分类精度和维度缩减率更加优越.

关键词: 特征选择森林优化算法；评分机制；类贪心；初始化；播种策略；计算效率

中图分类号: TP18 **文献标志码:** A

DOI: 10.13195/j.kzyjc.2023.1545

引用格式: 王霞, 张珊, 王勇, 等. 基于评分机制的类贪心森林优化特征选择算法 [J]. 控制与决策, 2025, 40(2): 517-527.

Feature selection using greedy-like forest optimization algorithm based on scoring mechanism

WANG Xia[1,2†]，*ZHANG Shan*[1,2]，*WANG Yong*[1,2]，*WANG Zhuo-ran*[1,2]

(1. School of Electrical and Information Technology， Yunnan Minzu University， Kunming 650504， China；
2. Yunnan Key Laboratory of Unmanned Autonomous System， Yunnan Minzu University， Kunming 650504， China)

Abstract: Feature selection using forest optimization algorithm(FSFOA) has well classification performance and dimensional reduction ability, but it has variable quality of initialised forest, larger randomness of local seeding and global seeding, and the low computational efficiency caused by expensive fitness evaluation. To solve the above problems, this paper proposes a feature selection using greedy-like forest optimization algorithm based on scoring mechanism(FSGLFOA-SM). Firstly, a scoring mechanism is constructed by using the classification accuracy of each dimensional decision variable as its score. From this, a greedy-like initialization strategy is proposed to generate an initialised forest with better quality, and a greedy-like local seeding strategy is proposed based on the comparison of scores, so that the decision variables with relatively higher scores can get a larger probability of local seeding. Then, a greedy-like genetic operator seeding strategy is proposed in the global seeding stage. The candidate forest is obtained by optimal selection and reconstruction, on which genetic, greedy-like crossover and mutation are carried out. So that the feature dimensions with higher scores are more likely to be retained and the classification accuracy of global seeding stage can be improved. Finally, a historical database is established, which reduces the calculation of duplicate solutions through accessing the database before fitness evaluation. The experimental results show that FSGLFOA-SM has superior classification accuracy and dimension reduction on 16 UCI datasets compared to the nine feature selection algorithms.

Keywords: feature selection using forest optimization algorithm(FSFOA)； scoring mechanism； greedy-like； initialization；seeding strategy；computational efficiency

收稿日期: 2023-11-07；**录用日期:** 2024-03-18.

基金项目: 云南省科技厅基础研究专项-面上项目 (202201AT070021)；国家自然科学基金项目 (61963038)；云南省教育厅科学研究基金项目 (2022J0439).

责任编委: 陈家伟.

†通讯作者. E-mail: wangxiacsu@163.com.

本文附带电子附录文件,可登录本刊官网该文“资源附件”区自行下载阅览.

0 引 言

特征选择 (feature selection, FS) 是机器学习和数据挖掘中的热门领域之一[1], 旨在从一组特征中选择部分有效特征, 使得分类效果达到与特征选择前近似甚至更好, 从而降低数据维度[2]. 特征选择过程包含 4 个部分:子集搜索机制、子集评价机制、停止准则和验证方法[3]. 根据子集评价机制的不同, 特征选择方法可以分为:过滤式、包裹式和嵌入式[4]. 过滤式方法将特征与类别之间的相关性作为判断特征贡献程度的重要标准[5]. 嵌入式方法利用特征选择方法模型完成训练得到特征子集[6]. 包裹式方法则直接把最终将要使用的学习器性能作为评价特征子集优劣的准则[7], 相较于过滤式和嵌入式, 包裹式特征选择方法更能充分考虑特征之间的相互影响.

子集搜索技术一般分为 3 种:完全搜索、启发式搜索、基于智能优化算法的搜索[8]. 随着特征数量的增加, 完全搜索方法的搜索空间呈指数型增长, 导致特征选择面临困难[9]. 启发式搜索, 如贪婪爬山算法, 容易陷入局部最优的问题中. 而近几年, 基于智能优化算法的搜索方法成为学者们争相研究的对象, 并且在包裹式特征选择领域取得了较好的成果. 文献 [10] 引入莱维飞行策略改进秃鹰搜索优化参数, 提高局部搜索能力, 并利用种群位置解提高全局寻优能力. 文献 [11] 提出了一种二进制沙猫群优化特征选择算法, 利用 KNN 分类器验证了算法的有效性. 文献 [12] 利用量子旋转门增强 WOA 算法的探索能力, 引入改进的突变算子和交叉算子, 提高了分类性能.

2014 年, Ghaemi 等[13] 提出了一种基于树木播种的演化算法 —— 森林优化算法 (forest optimization algorithm, FOA), 用于解决连续优化问题, 具有较好的全局搜索的能力; 之后,又将 FOA 算法应用于特征选择问题中, 提出特征选择的森林优化算法 (feature selection using forest optimization algorithm, FSFOA), 将森林优化算法的适用范围扩展到了离散型优化问题中, 在离散型数据集上实验并取得了良好的效果[14]. 文献 [15] 提出了基于森林优化特征选择算法的改进算法, 通过将向前和向后选择相结合的方法进行森林初始化, 采用极度贪婪策略进行局部播种, 并将维度更小、分类精度更好的新树添加到森林中, 提高了分类准确率和维度缩减率. 文献 [16] 采用皮尔逊相关和 $L1$ 正则化的方法完成森林初始化, 候选种群阶段采用优劣树分开和差额补足的方法解决优劣树不完备问题, 算法具有较好的分类性能. 文献 [17] 采用信息增益方法为初始化森林提供较为优秀的树木, 提出基于模拟退火算法的自适应全局播种策略, 并利用贪心策略更新最优树; 此外, 还设计了新的适应度函数, 综合考量了分类准确率和维度缩减率. 从以上研究可以看出, 目前研究者们对 FSFOA 算法的改进主要集中在 3 个方面:初始化策略的改进, 候选森林更新策略的改进和适应度函数的改进. 然而, FSFOA 算法除了初始化具有随机性, 局部播种和全局播种均采用随机播种策略, 这可能会导致错失最优特征子集, 降低了算法的搜索效率.

针对以上不足之处, 本文提出一种基于评分机制的类贪心森林优化特征选择算法 (feature selection using greedy-like forest optimization algorithm based on scoring mechanism, FSGLFOA-SM). 充分考虑到每个特征对分类精度的贡献差异, 对其进行量化评分, 并将评分应用于初始化、局部播种和全局播种阶段, 以提高算法的搜索效率; 同时, 针对昂贵适应度计算的问题, 建立历史数据库, 提高算法的计算效率. 实验结果表明, 相较于其他几种特征选择算法, FSGLFOA-SM 算法具有更好的分类性能及维度缩减能力.

1 FSFOA 算法及其不足之处

1.1 FSFOA 算法

FSFOA 算法以分类准确率 (classification accuracy, CA) 作为适应度函数,其值越大,表示所选特征有较高的分类精度,则当前树木的质量越高. FSFOA 算法主要由初始化森林、局部播种、森林规模限制、全局播种和更新最优树 5 部分组成,其主要实现步骤为:

1) 初始化森林:随机产生 N 棵年龄为 0 的树木并形成森林$\boldsymbol{T}$. 每棵树木包含D 维决策变量和树龄, 即 $\boldsymbol{T}=\{\boldsymbol{T}_1,\boldsymbol{T}_2,\dots,\boldsymbol{T}_i,\dots,\boldsymbol{T}_N\}$, $\boldsymbol{T}_i=[x_{i,1},x_{i,2},\dots,x_{i,j},\dots,x_{i,D},\text{Age}]$. 其中: $\boldsymbol{T}_i(i=1,2,\dots,N)$ 表示第 i 棵树, $x_{i,j}$表示第 i 棵树的第 j 维变量,与第 j 个特征相对应, $x_{i,j}$ 的取值为“0”或“1”, 分别代表对应特征未被选中或被选中.

2) 局部播种:对于 Age 为 0 的树, 根据参数 LSC (local seeding change) 值, 每棵树随机选取 LSC 个特征变量分别取反,进而产生 LSC 棵新树. 新树的树龄为 0, 被添加到森林中,旧树年龄加 1.

3) 森林规模限制: 先将年龄超过上限值的树木放入候选森林, 再将树木按照适应度降序排列,超过区域上限值的树木放入候选森林.

4) 全局播种: 从候选森林中随机选取若干棵树,

每棵树均随机选择 GSC (global seeding change) 个决策变量全部取反, 从而生成一棵树龄为 0 的新树, 并放入森林中, 同时, 旧树年龄加 1.

5) 更新最优树: 选取适应度值最大的树作为最优树, 并将其年龄置 0, 重新放入森林中.

1.2 FSFOA 算法的不足之处

相较于其他算法, FSFOA 算法在分类准确率和维度缩减率 (dimension reduction, DR) 方面虽已有较好的实验结果, 但仍存在以下几点不足之处:

1) FSFOA 算法采用随机策略进行森林初始化, 特征选择具有盲目性, 没有考虑各特征对分类精度的贡献具有差异性, 从而导致森林质量差.

2) 局部播种阶段, FSFOA 算法采用随机播种策略, 选择特征时盲目性较大, 不利于对优质特征的选择和劣质特征的删除, 增大了搜索难度.

3) FSFOA 算法在全局播种阶段依旧采用随机播种策略, 容易错失最优特征子集, 降低搜索效率.

4) 随着演化的进行, 森林中会出现大量特征重复的树木, 若每棵树木都进行适应度评估, 将会大大消耗计算资源和时间成本, 降低算法的运行效率.

2 FSGLFOA-SM 算法

针对 FSFOA 算法的不足之处, 本文提出基于评分机制的类贪心森林优化特征选择算法 (FSGLFOA-SM), 设计了基于评分机制的类贪心初始化策略、基于评分比较的类贪心局部播种策略、类贪心遗传算子播种策略和历史数据库以改善 FSFOA 算法在初始化阶段、局部播种阶段、全局播种阶段和适应度评估方面的不足. 下面分别对 4 个改进策略进行详细说明.

2.1 基于评分机制的类贪心初始化策略

初始化阶段作为算法的第 1 步, 应该为局部播种和全局播种提供优质的初始化森林. FSFOA 算法在初始化阶段并未考虑所选特征是否对提高分类准确率有利, 导致生成的森林分类准确率较低, 不利于后续阶段的寻优. 为使森林在该阶段生成较多的优质树木, 定义每个特征的分类精度为该特征的评分, 基于评分提出类贪心初始化策略, 使得建立的初始化森林能选中分类精度较高的特征而又不失随机性. 以下是决策变量评分机制和类贪心初始化策略的具体实施细节.

1) 决策变量评分机制.

首先,生成 $D \times D$ 维的单位矩阵 Q, 其中 D 为决策变量维度, 也即特征维度. Q 的每行代表森林中的一棵树,每棵树中仅有一个决策变量设置为 1, 其他的 $D-1$ 个变量均为 0, 且每棵树被选中的变量维度各不相同. 然后,对于矩阵 Q, 将其每一行作为特征选择问题的一个可能解,即一个特征子集, 通过分类器计算其分类精度, 作为各维决策变量的评分, 存储在变量 **Score** 中. **Score** $=[s_1, s_2, \ldots, s_j, \ldots, s_D]$, $j=1,2,\ldots,D$, s_j表示第 j 维决策变量的分值. 分类精度越高评分越高,则该决策变量被选中的概率越高.

2) 类贪心初始化策略.

对于初始化中待生成的每一棵树 $\boldsymbol{T_i}$, $i=1,2,\ldots,N$, 通过以下方式完成一次维度选择和初始化赋值: 在$[1, D]$间随机选取两个整数k_1和k_2, 若决策变量k_1的评分高于决策变量k_2的评分, 则$\boldsymbol{T_i}$的第k_1维度的值x_{i,k_1}为 1, 反之,则$\boldsymbol{T_i}$的第k_2维度的值x_{i,k_2}为 1.

在特征选择问题中, 维度缩减率也是优化过程中要考虑的主要指标. 因此, 在初始化时, 不宜选中过多的特征. 初始化每棵树时, 都生成一个 [0,1] 中的均匀分布随机数 r, 每棵树将进行 $r \times D$ 次的维度选择和初始化赋值.由于 r 的数学期望为 0.5, 且实施“有放回”地选择 $r \times D$ 个元素, 同一棵树中选择设定为 1 的决策变量可能会重复,所以最终整个森林 $\boldsymbol{T}$ 中“1”元素的占比会低于 50 % ,保证初始化的森林树木不会选中太多特征.

2.2 基于评分比较的类贪心局部播种策略

FSFOA 算法采用完全随机的方式进行局部播种, 这种方式盲目性较大, 不利于分类准确率的提高. 为了探索评分较高的特征对分类精度的贡献程度, 提出基于评分比较的类贪心局部播种策略, 使评分相对较高的决策变量获得更大的改变几率.

局部播种过程中生成子代树木的数量由 LSC 决定,例如, 当 LSC = 2 时将生成两个子代树. 图 1 所示为 LSC = 2 时基于评分比较的类贪心局部播种过程. 其具体步骤为:

1) 因 LSC = 2, 所以首先从父代 $\boldsymbol{T_i}$ 中随机选择 2 个决策变量 x_{i,m_1} 和 $x_{i,n_1}, m_1, n_1 \in [1, D]$;

2) 再从父代中随机选择两个决策变量x_{i,m_2}和x_{i,n_2}. 其中: $m_2, n_2 \in [1, D]$, $m_2 \neq m_1, n_2 \neq n_1$, 且$m_1, m_2, n_1, n_2$之间没有必然联系,都是随机选取的变量维度;

3) 决策变量x_{i,m_1}和x_{i,n_1}、x_{i,m_2}和x_{i,n_2}分别进行评分对比, 将分值大的变量值取反. 同时, 将子代树的树龄置 0, 父代树的树龄加 1, 放入森林中.

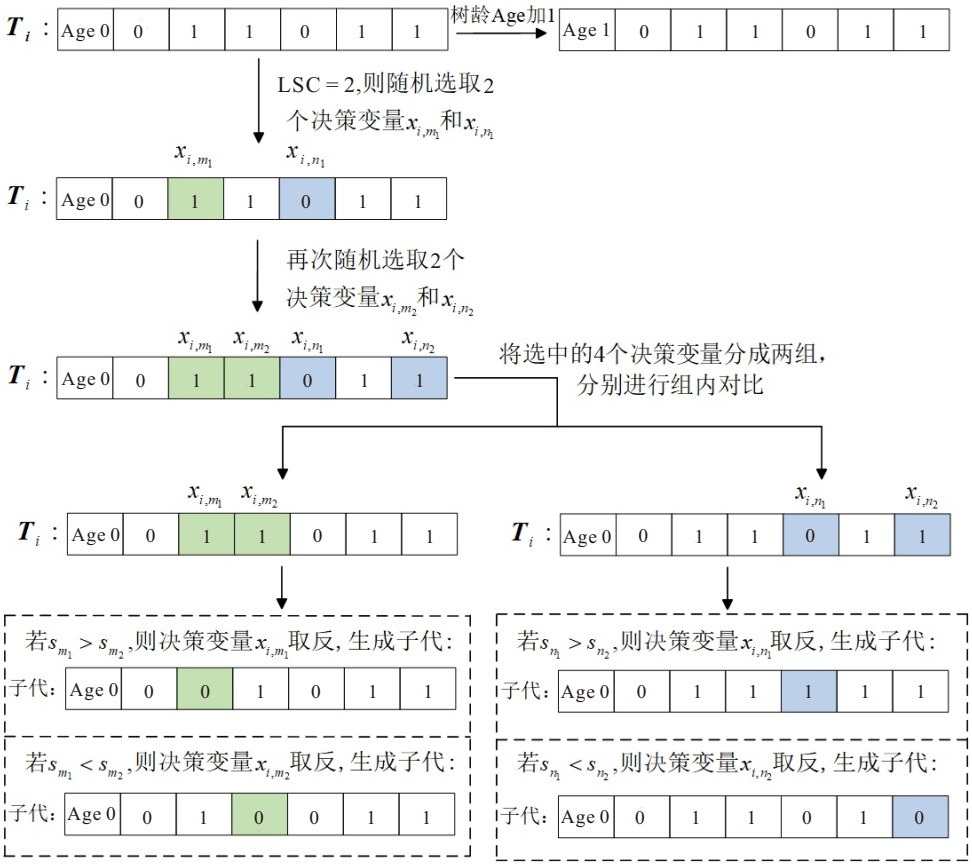

图1 基于评分比较的类贪心局部播种过程 (LSC = 2)

2.3 类贪心遗传算子播种策略

由 FSFOA 算法原理可知, 候选森林中主要由两类树构成: 超过年龄上限的树和超过区域上限的树, 若从中随机选取若干棵树, 则被选中的树木中可能包含适应度函数极低的树木, 不利于全局播种阶段分类准确率的提高. 针对这一问题, 首先, 将候选森林中的树进行择优, 对于筛选出的较优质的树, 给与其更多的播种机会; 然后, 提出类贪心遗传算子播种策略, 选出两棵优质树中在相同维度上取值相同的变量, 作为优秀基因在子代树中继承下去; 最后, 对于取值不同的变量维度, 通过类贪心交叉操作保留评分较高的特征维度, 从而有利于保留优质树中的优良基因. 以下是类贪心遗传算子播种策略的具体实施过程.

1) 类贪心择优: 每次从候选森林中随机选取两棵树, 对比其适应度值, 将值较大的树选出. 为使较优质的树获得更高的播种机会, 采用有放回的选取, 最终选定与候选森林规模相同的优质树.

2) 遗传与类贪心交叉: 类贪心择优选出的优质树木相比于候选森林中的树木有较好的适应度值,随机选取两个父本树, 它们可能在若干变量维度上的值相同,意味着该类特征对于分类准确率的提高会具有一定的帮助,通过遗传操作保留两特征子集相同维度中变量取值相同的特征. 而相同维度中变量取值不同的特征,则采取类贪心交叉操作, 选择并保留评分较高的特征维度. 图 2 为全局播种阶段的遗传与类贪心交叉过程. 其中, $\boldsymbol{T_p}$ 和 $\boldsymbol{T_q}$ 是交叉池的两个随机父本, 两父本生成一个子代个体 $\boldsymbol{T_o}$, 随机选取的维度变量 $m_1, m_2, n_1, n_2 \in [1, D]$. 交叉播种规则如下:

①生成随机数 r, 若 $r < 0.5$, 则执行 $\boldsymbol{T_p}\&\overline{\boldsymbol{T_q}}$ 选出 $\boldsymbol{T_p}$ 中被选择但 $\boldsymbol{T_q}$ 中被删除的特征维度, 利用评分机制从此类变量维度中随机选择两个变量维度比较其评分, 筛选出评分小的变量维度,将其置 0;

②若随机数 $r > 0.5$, 则执行 $\overline{\boldsymbol{T_p}}\&\boldsymbol{T_q}$, 选出 $\boldsymbol{T_q}$ 中被选择但 $\boldsymbol{T_p}$ 中被删除的特征维度, 利用评分机制从此类变量维度中随机选择两个变量维度比较其评分, 筛选出评分大的变量, 将其置 1.

通过遗传与类贪心交叉操作, 两个父本中相同的优良基因遗传给子代, 部分不同基因进行交叉, 保证了子代树木的优良性和多样性.

3) 变异: 因类贪心择优采取有放回选取, 适应度较高的树木被选中的可能性更大, 被选出的优质树中存在大量适应度值相同且特征子集相同的树木. 在进行交叉操作时, 若选中的两个父代特征子集完全相同, 则无论 $\boldsymbol{T_p}\&\overline{\boldsymbol{T_q}}$ 还是 $\overline{\boldsymbol{T_p}}\&\boldsymbol{T_q}$ 操作, 均无法选出有差异的特征维度, 无法完成交叉. 此时, 变异操

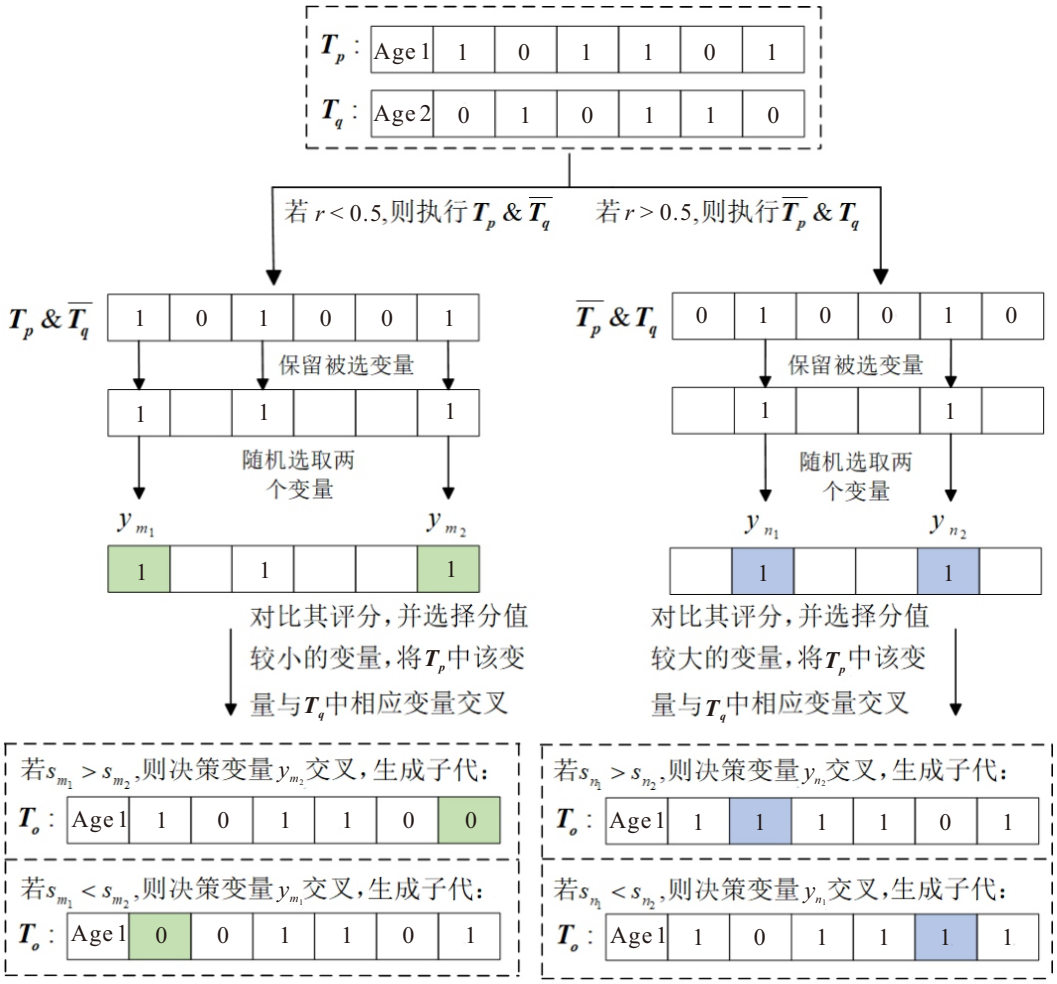

图2　类贪心遗传算子播种策略的遗传与类贪心交叉过程

作便显得尤为重要. 变异播种规则如下:

① 生成随机数 r, 若 $r<0.5$, 则随机选取两个值为“1”的变量维度, 通过评分筛选出分值小的变量维度进行值取反;

②若随机数 $r>0.5$, 则随机选取的两个值为“0”的变量, 通过评分筛选出分值大的变量维度进行值取反.

通过变异,使分值较高的变量维度被选中的几率更大. 图 3 为变异操作过程. m_1, m_2, n_1, n_2 为随机选取的决策变量维度,得到的子代 T_o 有 4 种可能性结果.

2.4　历史数据库

与 FSFOA 算法相同, 本文以分类准确率 CA 作为适应度函数, 依据每棵树选中的特征将数据集划

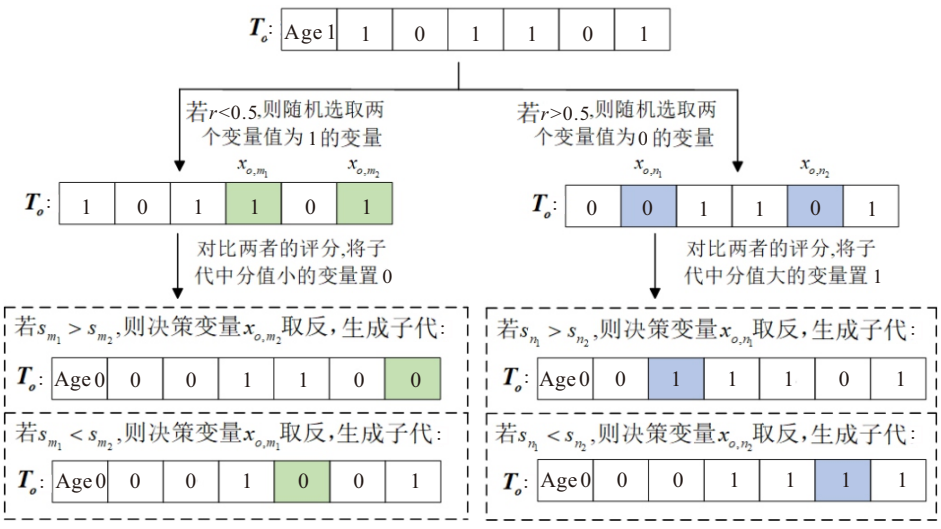

图3　类贪心遗传算子播种策略的变异操作过程

分为训练集和测试集, 经过分类器训练模型并对测试集进行分类测试实验. 由此可知, 计算每棵树的适应度值都需要进行分类测试, 耗费较多的计算资源和时间成本. 然而, 特征选择问题中的决策变量取值只有“0”和“1”两种可能状态, 在算法进化迭代过程中难免会出现大量重复解个体, 它们所选中的特征完全相同. 为减少重复解个体的计算量, 提高算法运行效率, 本文提出建立历史数据库的方法, 为每棵不同的树赋予一个唯一的 Id 值, 每次计算树的适应度时, 先在历史数据库中查询相应树的 Id 值, 若该 Id 已经存在于历史库中, 则直接将这棵树的适应度值从历史数据库中读取出; 否则, 先计算这棵树的适应度值, 再将其存入历史数据库中. Id 值按下式计算:

$$\mathrm{Id}=\sum_{j=1}^{D}x_{i,j}2^{j-1}. \tag{1}$$

2.5 FSGLFOA-SM 算法流程

FSGLFOA-SM 算法的具体流程如下.

1) 初始化阶段: 构建决策变量评分机制, 利用基于评分机制的类贪心初始化策略, 对每棵树木完成 $r\times D$次的初始化赋值, 并将树木年龄置 0;

2) 局部播种: 利用基于评分比较的类贪心局部播种策略对于年龄为 0 的树木进行局部播种, 每棵父代树木生成 LSC 棵子代树木, 子代树木年龄为 0, 父代树木年龄加 1.

3) 森林规模限制: 先将树龄超过 life time 的树木放入候选森林, 再将树木按照适应度值降序排列, 超过 area limit 的树木放入候选森林.

4) 全局播种: 随机性选择类贪心遗传算子播种策略和随机播种策略完成全局播种. 随机播种策略与 FSFOA 算法中的全局播种策略相同; 类贪心遗传算子播种策略通过从候选森林中择取优质树木进行遗传、类贪心交叉和变异操作.

5) 更新最优树:将森林中适应度值最大的树年龄置 0, 重新放入森林中.

3 实验分析

所有实验均在 DELL 机上执行, 编程语言为 Matlab, 该机器的配置是 AMD Ryzen 5 5600H with Radeon Graphics (3.30 GHz), 16 G 内存.

3.1 数据集及对比算法

实验使用了 16 个 UCI 数据集, 其参数信息如表 1 所示. 若数据集中的特征数量是 [1, 19], [20, 49], [50, ∞], 则对应的数据集规模分别是低维、中维和高维数据集. 选取 9 个对比算法进行对比实验, 如表 2 所示.

表1 数据集及参数信息

数据集	特征数	样本数	LSC	GSC	类别	数据集维度
Wine	13	178	3	6	3	低维
Cleveland	13	303	3	6	5	
Heart-statlog	13	270	3	6	2	
Segmentation	19	2310	4	9	7	
Glass	9	214	2	4	7	
Vehicle	18	846	4	9	4	
Ionosphere	34	351	7	15	2	中维
Dermatology	34	366	7	15	6	
Glioma Grading Clinical and Mutation Features	23	839	5	12	2	
Forest Type Mapping	27	523	5	14	4	
Breast Cancer Wisconsin (Diagnostic)	30	569	6	15	2	
Sonar	60	208	12	30	2	高维
Semeion Handwritten Digit	256	1593	51	128	10	
LSVT Voice Rehabilitation	310	126	62	155	2	
ISOLET	617	7797	123	308	26	
Musk (Version 1)	166	476	33	83	2	

表2 对比算法的详细信息

序号	算法名称	数据集划分	描述/发表年份
1	FSFOA	70 ～ 30, 10-fold, 2-fold	森林优化特征选择算法[14]/2016
2	FSIFOA	70 ～ 30, 10-fold, 2-fold	改进的森林优化特征选择算法[16]/2020
3	DAFSFOA	70 ～ 30, 10-fold, 2-fold	基于重复度分析的森林优化特征选择算法[18]/2022
4	IFSFOA	70 ～ 30, 10-fold, 2-fold	基于森林优化特征选择算法的改进算法[15]/2018
5	BGWOPSO	70 ～ 30, 10-fold, 2-fold	基于混合灰狼优化的二元优化特征选择算法[19]/2019
6	Rc-BBFA	70 ～ 30	基于收益成本的萤火虫算法的特征选择算法[20]/2017
7	SFS	70 ～ 30, 50 ～ 50	序列正向选择[21]/2010
8	SBS	70 ～ 30, 50 ～ 50	序列反向选择[21]/2010
9	SFFS	70 ～ 30, 50 ～ 50	序列浮动向前选择[21]/2010

注: 70 ～ 30 表示对数据集的划分采用 70 % 作为训练集, 30 % 作为测试集; 50 ～ 50 表示 50 % 作为训练集, 50 % 作为测试集; 10-fold 表示十折交叉验证; 2-fold 表示二折交叉验证

由于 FSGLFOA-SM 算法是针对森林优化特征选择算法 (FSFOA) 做的改进算法, 且适用于特征选择问题, 表 1 中的对比算法选择依据如下:

1) 与 FSFOA 做对比, 即表 2 中的第 1 个算法;

2) 与其他 FSFOA 改进算法做对比, 即表 2 中的第 2 ～ 第 4 个算法;

3) 与其他基于启发式算法的特征选择算法做对比, 即表 2 中的第 5 和第 6 个算法;

4) 与经典的特征选择算法做对比, 即表 2 中的第 7 ～ 第 9 个算法.

3.2 实验参数设置

本文实验参数设置包括分类器参数设置、数据集划分方式、算法参数设置, 设置依据如下.

1) 分类器参数设置:本文分类器包括 KNN 和 SVM. 因文献 [14-16] 和文献 [19] 中, KNN 分类器参数采用了 K =1、3、5, SVM 分类器采用了 Rbf 核函数, 为分析 FSGLFOA-SM 算法与对比算法在分类器上的性能差异, 本文分类器参数设置与文献 [14-16]、文献 [19] 参数相同.

2) 数据集划分方式:本文数据集划分方式与文献 [14-16]、文献 [19] 相同, 即 70 % 作为训练集、30 % 作为测试集, 十折交叉验证, 二折交叉验证.

3) 算法参数设置: 本文涉及到的参数分别是树木的年龄上限 (life time)、森林规模上限 (area limit)、全局播种比例 (transfer rate)、局部播种变化个数 (LSC) 和全局播种变化个数 (GSC). 为保证算法对比结果的公平性, 本文算法参数设置与文献 [14-16]、[19] 中的参数设置相同, 即 "life time" =15, "area limit" = 50, "transfer rate" = 5 %. LSC 和 GSC 的数值分别为各数据集维度的 1/5 和 1/2[14].

3.3　算法对比实验结果及分析

将 FSGLFOA-SM 算法与表 2 中的 9 种算法进行比较, 表 3 ～表 5 中分别总结了以上算法在低维、中维、高维数据集的实验结果. 在每个表中, 对于最高分类准确率和维度缩减率均加粗显示.

由表 3 ～表 5 可以看出:

1) FSGLFOA-SM 算法在 6 个低维数据集的 14 次分类对比情况中, CA 排名第 1 的次数为 6 次, 排名第 2 的次数为 4 次, DR 排名第 1 的次数为 6 次, 排名第 2 的次数为 6 次. 其中, 在 Wine 数据集的 1-NN 分类器、Rbf-svm 分类器以及 Vehicle 数据集的 1-NN 分类器中, FSGLFOA-SM 算法的 CA 与 DR 并列第 1. 综合 CA 和 DR 的排名可以看出, FSGLFOA-SM 算法在低维数据集上获得的分类精度和维度缩减率, 优于表 3 中的所有对比算法.

2) FSGLFOA-SM 算法在 5 个中维数据集的 16 次分类对比情况中, CA 排名第 1 的次数为 9 次, 排名第 2 的次数为 3 次.其中, 在 Glioma Grading Clinical and Mutation Features 数据集 5-NN 和 3-NN 分类情况下, FSGLFOA-SM 算法的 CA 虽排名第 2, 但与 CA 排名第 1 的算法相比, FSGLFOA-SM 算法的 DR 均高出其 10 % 左右, 而 CA 仅比排名第 1 的算法分别低 1.6 % 和 0.19 %. FSGLFOA-SM 算法的 DR 排名第 1 的次数为 13 次, 第 2 的次数为 2 次, 其中, 在 Ionosphere 数据集的 Rbf-svm 和 Breast Cancer Wisconsin (Diagnostic) 数据集的 1-NN 分类情况下, 与 DR 排名第 2 的算法相比, FSGLFOA-SM 算法的 DR 分别高出其 9 % 和 14 % 左右.可以看出, FSGLFOA-SM 算法在中维数据集上维度缩减优势较明显, 牺牲较少的 DR 可以获得 CA 的较大提升.

3) FSGLFOA-SM 算法在 5 个高维数据集的 10 次分类对比情况下, CA 排名第 1 的次数为 9 次, 排名第 2 的次数为 1 次. 其中, 在 Sonar 数据集的 5-NN 和 Musk (Version 1) 的 1-NN 分类情况下, 与 CA 排名第 2 的算法相比, FSGLFOA-SM 算法的 CA 分别高出其 12 % 和 4 %、DR 分别高出其 12 % 和 19 %. FSGLFOA-SM 算法的 DR 排名第 1 的次数为 8 次, 排名第 2 的次数为 1 次. 可见, FSGLFOA-SM 算法在高维数据集中的分类准确率和维度缩减率优势更为明显.

综合以上实验结果, 可以得出如下结论:

FSGLFOA-SM 算法在低维、中维和高维数据集上均具有良好的分类性能, 尤其是维度缩减率的优势较为明显. 这得益于本文提出的评分机制对维度缩减率的贡献, 以及类贪心策略对收敛精度的提高. 首先, 评分机制实施 "有放回" 的选择$r \times D$个元素, 使初始化森林中 "1" 元素的占比低于 50 %, 在初始化阶段就较好地降低了特征维度, 为维度缩减奠定了基础; 同时, 基于每个特征的评分来选取特征, 能更好地选出对分类精度提高有贡献的特征, 对维度缩减和分类精度的提高有极大的帮助. 其次, 在后续的森林更新中, 类贪心局部播种策略使得树木中评分较高的特征获得更大的改变几率, 类贪心遗传算子播种策略使得优质树木中的优良基因得以保留, 降低了特征选择的盲目性, 抑制了随机选择特征引发的维度灾难现象, 为分类精度和维度缩减率的提高提供保障.

评价一个算法的优劣, 需同时考虑 CA 和 DR 两个指标. 由于表 3 ～ 表 5 中的数据结果较多, 为直观体现 FSGLFOA-SM 算法与其他算法的对比情况, 统计表 3 ～ 表 5 中 FSGLFOA-SM 算法相较于其他算法在所有数据集和分类器情况下, 分别胜利、失败和不相上下的次数, 示于表 6 中. 以 "$\surd$" 表示与当前算法相比, FSGLFOA-SM 算法获胜, 即 FSGLFOA-SM 算法的 CA 和 DR 值均优于当前算法或其中一个指标数值相等, 另一指标数值比当前算法高; 以 "×" 表示与当前算法相比, FSGLFOA-SM算法失败, 即 FSGLFOA-SM 算法的 CA 和 DR 值均劣于当前算法或其中一个指标数值相等、另一指标数值比当前算法低; "O" 表示与当前算法相比, FSGLFOA-SM 算法与其不相上下, 即 FSGLFOA-SM 算法的其中一个指标数值低于当前算法, 而另一指标数值则高于当前算法. 由表 6 可直观地看出, 与其他 9 种对比算法

表3 低维数据集中 FSGLFOA-SM 及其对比算法的分类准确率及维度缩减率

Wine	CA / %	DR / %	Classifier
FSGLFOA-SM	98.11(70 ～ 30)	61.54	5-NN
FSFOA	**99.2(70 ～ 30)**	30.76	
FSIFOA	95.7(70 ～ 30)	38.46	
DAFSFOA	97.69(70 ～ 30)	**70**	
IFSFOA	98.07(70 ～ 30)	46.15	
BGWOPSO	91.02(70 ～ 30)	17.08	
FSGLFOA-SM	95.49(10-fold)	61.54	3-NN
FSFOA	98.87(10-fold)	42.58	
FSIFOA	95.61(10-fold)	61.54	
DAFSFOA	98.33(10-fold)	53.85	
IFSFOA	**99.99(10-fold)**	**79.23**	
BGWOPSO	88.67(10-fold)	37.08	
FSGLFOA-SM	**99.99(70 ～ 30)**	**69.23**	1-NN
FSFOA	98.07(70 ～ 30)	50	
FSIFOA	95.61(70 ～ 30)	61.54	
DAFSFOA	98.16(70 ～ 30)	**69.23**	
IFSFOA	96.15(70 ～ 30)	**69.23**	
BGWOPSO	89.55(70 ～ 30)	17.54	
SFS	97.69(70 ～ 30)	35.38	
SBS	94.77(70 ～ 30)	46.15	
SFFS	96.56(70 ～ 30)	36.92	
Rc-BBFA	99.66(70 ～ 30)	38.46	
FSGLFOA-SM	**98.31(2-fold)**	**76.92**	Rbf-svm
FSFOA	96.06(2-fold)	37.17	
DAFSFOA	96.07(2-fold)	53.85	
IFSFOA	**98.31(2-fold)**	57.69	
BGWOPSO	88.8(2-fold)	54.23	

Cleveland	CA / %	DR / %	Classifier
FSGLFOA-SM	**64.44(70 ～ 30)**	69.23	1-NN
FSFOA	55.55(70 ～ 30)	**71.42**	
FSIFOA	62.22(70 ～ 30)	61.54	
DAFSFOA	62.64(70 ～ 30)	69.23	
IFSFOA	59.77(70 ～ 30)	61.53	
BGWOPSO	53.31(70 ～ 30)	38.62	
SFS	51.79(70 ～ 30)	47.7	
SBS	54.8(70 ～ 30)	38.5	
SFFS	49.55(70 ～ 30)	53.8	

Segmentation	CA / %	DR / %	Classifier
FSGLFOA-SM	**97.1(10-fold)**	57.9	3-NN
FSFOA	96.2(10-fold)	30	
FSIFOA	96.88(10-fold)	52.63	
DAFSFOA	**97.1(10-fold)**	**68.16**	
IFSFOA	96.32(10-fold)	65.26	
BGWOPSO	95.54(10-fold)	36.74	
FSGLFOA-SM	97.69(70 ～ 30)	**42.11**	1-NN
FSFOA	96.51(70 ～ 30)	36.84	
BGWOPSO	96.93(70 ～ 30)	24.79	
Rc-BBFA	**98.27(70 ～ 30)**	36.84	

Glass	CA / %	DR / %	Classifier
FSGLFOA-SM	75(70 ～ 30)	**66.67**	1-NN
FSFOA	71.88(70 ～ 30)	40	
FSIFOA	75.38(70 ～ 30)	55.56	
DAFSFOA	**77.23(70 ～ 30)**	42.77	
IFSFOA	74.28(70 ～ 30)	48.88	
BGWOPSO	72.56(70 ～ 30)	19	
SFS	72.24(70 ～ 30)	26.66	
SFFS	71.77(70 ～ 30)	37.77	
FSGLFOA-SM	70.09(2-fold)	55.56	Rbf-svm
FSFOA	68.22(2-fold)	**60**	
FSIFOA	68.69(2-fold)	33.33	
DAFSFOA	66.83(2-fold)	55.55	
IFSFOA	**71.03(2-fold)**	44.44	
BGWOPSO	68.3(2-fold)	39.33	

Vehicle	CA / %	DR / %	Classifier
FSGLFOA-SM	77.08(70 ～ 30)	**66.67**	5-NN
FSFOA	73.98(70 ～ 30)	50	
FSIFOA	76.77(70 ～ 30)	55.56	
DAFSFOA	**77.17(70 ～ 30)**	51.11	
IFSFOA	75.39(70 ～ 30)	50	
BGWOPSO	70.46(70 ～ 30)	28.94	
FSGLFOA-SM	**76.68(70～30)**	**66.67**	1-NN
FSFOA	73.81(70 ～ 30)	61.11	
BGWOPSO	69.18(70 ～ 30)	24.56	
Rc-BBFA	75.79(70 ～ 30)	61.11	
FSGLFOA-SM	**76.12(2-fold)**	55.56	Rbf-svm
FSFOA	62.41(2-fold)	47.22	
FSIFOA	69.03(2-fold)	66.67	
DAFSFOA	69.38(2-fold)	66.67	
IFSFOA	69.62(2-fold)	**75**	
BGWOPSO	73.65(2-fold)	27.78	

Heart-statlog	CA / %	DR / %	Classifier
FSGLFOA-SM	84.07(10-fold)	69.23	3-NN
FSFOA	85.18(10-fold)	35.71	
FSIFOA	83.33(10-fold)	53.85	
DAFSFOA	85.92(10-fold)	60	
IFSFOA	**91.85(10-fold)**	**70**	
BGWOPSO	80.28(10-fold)	32.39	
FSGLFOA-SM	82.22(2-fold)	61.54	Rbf-svm
FSFOA	84.07(2-fold)	50	
FSIFOA	84.81(2-fold)	**76.92**	
DAFSFOA	**85.22(2-fold)**	53.85	
IFSFOA	84.44(2-fold)	57.69	
BGWOPSO	76.43(2-fold)	49.31	
—	—	—	—
—	—	—	—
—	—	—	—
—	—	—	—

表4　中维数据集中 FSGLFOA-SM 及其对比算法的分类准确率及维度缩减率

Ionosphere	CA / %	DR / %	Classifier
FSGLFOA-SM	93.47(10-fold)	**91.18**	5-NN
FSFOA	89.43(10-fold)	54.28	
FSIFOA	93.23(10-fold)	79.41	
DAFSFOA	95(10-fold)	79.7	
IFSFOA	**98.86(10-fold)**	87.65	
BGWOPSO	88.08(10-fold)	39.65	
FSGLFOA-SM	94.31(10-fold)	82.35	3-NN
FSFOA	92.3(10-fold)	61.76	
FSIFOA	93.83(10-fold)	76.47	
DAFSFOA	95.28(10-fold)	77.06	
IFSFOA	**99.42(10-fold)**	**87.35**	
BGWOPSO	89.56(10-fold)	44.12	
FSGLFOA-SM	98.1(70 ～ 30)	70.59	1-NN
FSFOA	89.52(70 ～ 30)	54.28	
FSIFOA	95.16(70 ～ 30)	61.76	
DAFSFOA	**98.44(70 ～ 30)**	74.56	
IFSFOA	96.85(70 ～ 30)	**79.11**	
BGWOPSO	91.57(70 ～ 30)	43.94	
SFS	87.75(50 ～ 50)	65.88	
SBS	84.61(50 ～ 50)	77.64	
SFFS	88.32(50 ～ 50)	75.29	
Rc-BBFA	96.18(70 ～ 30)	58.82	
FSGLFOA-SM	94.87(2-fold)	**79.41**	Rbf-svm
FSFOA	94.58(2-fold)	57.14	
FSIFOA	95.16(2-fold)	58.82	
DAFSFOA	**97.27(2-fold)**	67.65	
IFSFOA	95.73(2-fold)	70.58	
BGWOPSO	91.43(2-fold)	52.59	

Dermatology	CA / %	DR / %	Classifier
FSGLFOA-SM	98.17(70 ～ 30)	63.64	1-NN
FSFOA	97.27(70 ～ 30)	45.71	
FSIFOA	99.07(70 ～ 30)	58.82	
DAFSFOA	99.4(70 ～ 30)	**67.06**	
IFSFOA	**99.79(70 ～ 30)**	51.96	
BGWOPSO	93.81(70 ～ 30)	45.46	
SFS	94.02(70 ～ 30)	44.7	
SBS	91.78(70 ～ 30)	58.23	
SFFS	93.7(70 ～ 30)	62.35	

Glioma Grading Clinical and Mutation Features	CA / %	DR / %	Classifier
FSGLFOA-SM	88.11(70 ～ 30)	**54.35**	5-NN
FSFOA	**89.71(70 ～ 30)**	44.93	
BGWOPSO	84.07(70 ～ 30)	32.92	
FSGLFOA-SM	87.52(70 ～ 30)	**60.14**	3-NN
FSFOA	**87.71(70 ～ 30)**	50.73	
BGWOPSO	82.55(70 ～ 30)	28.24	
FSGLFOA-SM	**82.62**(10-fold)	**58.7**	1-NN
FSFOA	82.46(10-fold)	51.45	
BGWOPSO	79.83(10-fold)	32.44	

Forest Type Mapping	CA / %	DR / %	Classifier
FSGLFOA-SM	**92.74(70 ～ 30)**	**59.26**	5-NN
FSFOA	91.99(70 ～ 30)	52.47	
BGWOPSO	88.73(70 ～ 30)	35.06	
FSGLFOA-SM	**91.94(10-fold)**	**56.18**	3-NN
FSFOA	91.78(10-fold)	47.53	
BGWOPSO	90.07(10-fold)	28.95	
FSGLFOA-SM	**93.38(70 ～ 30)**	**59.26**	1-NN
FSFOA	92.2(70 ～ 30)	50	
BGWOPSO	85.99(70 ～ 30)	34.25	
FSGLFOA-SM	**90.55(2-fold)**	**79.26**	Rbf-svm
FSFOA	90.21(2-fold)	77.04	
BGWOPSO	84.02(2-fold)	67.59	

Breast Cancer Wisconsin (Diagnostic)	CA / %	DR / %	Classifier
FSGLFOA-SM	**95.78(70 ～ 30)**	**66.11**	5-NN
FSFOA	95(70 ～ 30)	49.44	
BGWOPSO	93.47(70 ～ 30)	35.45	
FSGLFOA-SM	**96.28(70 ～ 30)**	**61.11**	3-NN
FSFOA	94.61(70 ～ 30)	52.22	
BGWOPSO	91.44(70 ～ 30)	42.28	
FSGLFOA-SM	**96.08(70 ～ 30)**	**64.45**	1-NN
FSFOA	94.41(70 ～ 30)	50	
BGWOPSO	92(70 ～ 30)	37.17	
FSGLFOA-SM	**97.68(2-fold)**	**83.33**	Rbf-svm
FSFOA	97.65(2-fold)	78.67	
BGWOPSO	51.22(2-fold)	51.22	
—	—	—	—
—	—	—	—
—	—	—	—

表5　高维数据集中 FSGLFOA-SM 及其对比算法的分类准确率及维度缩减率

Sonar	CA / %	DR / %	Classifier
FSGLFOA-SM	**98.39(70 ～ 30)**	76.67	5-NN
FSFOA	86.98(70 ～ 30)	44.26	
FSIFOA	74.6(70 ～ 30)	68.33	
DAFSFOA	95.40(70 ～ 30)	75.67	
IFSFOA	87.54(70 ～ 30)	**86.33**	
BGWOPSO	85.16(70 ～ 30)	37.64	
FSGLFOA-SM	96.77(70 ～ 30)	70	1-NN
FSFOA	85.43(70 ～ 30)	57.37	
FSIFOA	76.19(70 ～ 30)	76.67	
DAFSFOA	**98.33(70 ～ 30)**	76.58	
IFSFOA	91.96(70 ～ 30)	**79.33**	
BGWOPSO	86.98(70 ～ 30)	46.83	
SFS	66.43(50 ～ 50)	61.33	
SBS	62.2(50 ～ 50)	45.33	
SFFS	64.55(50 ～ 50)	61.33	
Rc-BBFA	95.57(70 ～ 30)	53.33	
FSGLFOA-SM	**89.9(2-fold)**	**88.33**	Rbf-svm
FSFOA	65.86(2-fold)	54.09	
FSIFOA	75.94(2-fold)	78.33	
DAFSFOA	88.51(2-fold)	62.83	
IFSFOA	78.84(2-fold)	85	
BGWOPSO	56.07(2-fold)	58.1	
—	—	—	
—	—	—	

ISOLET	CA / %	DR / %	Classifier
FSGLFOA-SM	**92.06(10-fold)**	**57.5**	3-NN
FSFOA	91.6(10-fold)	54.62	
BGWOPSO	88.13(10-fold)	25.35	
FSGLFOA-SM	**92.13(70 ～ 30)**	**62.56**	1-NN
FSFOA	91.02(70 ～ 30)	50.24	
BGWOPSO	88.46(70 ～ 30)	33.98	

LSVT Voice Rehabilitation	CA / %	DR / %	Classifier
FSGLFOA-SM	**91.89(70 ～ 30)**	**98.39**	1-NN
FSFOA	86.49(70 ～ 30)	79.68	
BGWOPSO	62.08(70 ～ 30)	40.35	

Musk (Version 1)	CA / %	DR / %	Classifier
FSGLFOA-SM	**92.66(10-fold)**	**59.04**	5-NN
FSFOA	91.82(10-fold)	54.82	
BGWOPSO	89.84(10-fold)	31.45	
FSGLFOA-SM	**97.89(70 ～ 30)**	**54.22**	3-NN
FSFOA	96.48(70 ～ 30)	53.61	
BGWOPSO	92.03(70 ～ 30)	37.88	
FSGLFOA-SM	**95.18(10-fold)**	**69.88**	1-NN
FSFOA	91.4(10-fold)	50	
BGWOPSO	88.78(10-fold)	38.15	

Semeion Handwritten Digit	CA / %	DR / %	Classifier
FSGLFOA-SM	**93.29(70 ～ 30)**	**58.98**	5-NN
FSFOA	91.4(70 ～ 30)	46.09	
BGWOPSO	92.1(70 ～ 30)	15.15	

相比, FSGLFOA-SM 算法分类性能有明显的优势. 此外, 根据数据集维度不同, 进一步分别统计低维、中维和高维数据集中 FSGLFOA-SM 算法与其他算法的对比结果, 如表 7 所示. 可以看出, 与其他算法相比, FSGLFOA-SM 算法在低维、中维和高维数据中, “√”的个数总是高于“O”和“×”的个数, FSGLFOA-SM 算法的分类性能更为优越.

表6 FSGLFOA-SM 算法与其他算法对比的结果统计

对比算法	个数		
	√	O	×
FSFOA	32	8	0
FSIFOA	12	5	2
DAFSFOA	6	10	4
IFSFOA	7	10	3
BGWOPSO	40	0	0
SFS	6	0	0
SBS	4	1	0
SFFS	5	1	0
Rc-BBFA	4	1	0

表7 不同数据集维度下 FSGLFOA-SM 算法与其他算法对比的结果统计

对比算法	不同维度“√”、“O”和“×”的个数								
	低维			中维			高维		
	√	O	×	√	O	×	√	O	×
FSFOA	8	6	0	14	2	0	10	0	0
FSIFOA	7	2	2	3	2	0	2	1	0
DAFSFOA	4	7	1	0	3	2	2	0	1
IFSFOA	6	4	2	0	4	1	1	2	0
BGWOPSO	14	0	0	16	0	0	10	0	0
SFS	3	0	0	2	0	0	1	0	0
SBS	2	0	0	1	1	0	1	0	0
SFFS	3	0	0	1	1	0	1	0	0
Rc-BBFA	2	1	0	1	0	0	1	0	0

4 结 论

本文针对 FSFOA 算法不足之处, 提出了 4 个改进策略, 形成了新算法 FSGLFOA-SM. 提出基于评分机制的类贪心初始化策略, 生成了具有较高适应度值的初始化森林, 为后续特征搜索奠定了良好的基础; 提出基于评分比较的类贪心局部播种策略, 使评分相对较高的决策变量获得更多的局部播种机会; 在全局播种阶段提出类贪心遗传算子播种策略, 引入遗传、交叉和变异的方法, 弥补了随机播种策略的不足; 设计历史数据库, 极大地提高了算法的计算效率. 将 FSGLFOA-SM 算法在 16 个数据集和 9 个特征选择算法上进行测试和比较, 实验结果表明, 本文所提出的 FSGLFOA-SM 算法在低维、中维和高维数据集分类问题中均有较好的分类性能, 尤其是对于中维和高维的分类问题, FSGLFOA-SM 算法在分类精度和维度缩减率上的优势更为明显. FSGLFOA-SM 算法可用于组织病理学图像分类、情感分类、可再生能源预测、抗生素耐药性预测、恶意软件识别等现实特征选择问题, 尤其是高维特征选择问题. 虽然现阶段的工作获得了较理想的结果, 但仍有很多地方值得进一步地深入研究:

1) 现阶段的工作主要针对 UCI 数据库提供的标准数据集进行了测试, 下一步的工作可将算法应用于实际问题中, 以检验算法在实际应用问题中的求解性能, 扩展算法的适用领域;

2) 在局部播种和全局播种阶段, 由于 LSC 参数是数据集维度的 1/5, GSC 参数是数据集维度的 1/2, 导致高维数据集生成大量子代, 时间成本增加, 因此, 可考虑设计更为合理的 LSC 和 GSC 参数.

参考文献 (References)

[1] Diao R, Chao F, Peng T X, et al. Feature selection inspired classifier ensemble reduction[J]. IEEE Transactions on Cybernetics, 2014, 44(8): 1259-1268.

[2] 姚旭, 王晓丹, 张玉玺, 等. 特征选择方法综述[J]. 控制与决策, 2012, 27(2): 161-166.
(Yao X, Wang X D, Zhang Y X, et al. Summary of feature selection algorithms[J]. Control and Decision, 2012, 27(2): 161-166.)

[3] Liu H, Motoda H. Feature selection for knowledge discovery and data mining[M]. New York: Kluwer Academic Publishers, 1998.

[4] 何杜博, 孙胜祥, 梁新, 等. 基于自适应图学习的多目标特征选择算法[J]. 控制与决策, 2024, 39(7): 2295-2304.
(He D B, Sun S X, Liang X, et al. Multi-target feature selection algorithm based on adaptive graph learning[J]. Control and Decision, 2024, 39(7): 2295-2304.)

[5] Bommert A, Sun X D, Bischl B, et al. Benchmark for filter methods for feature selection in high-dimensional classification data[J]. Computational Statistics & Data Analysis, 2020, 143: 106839.

[6] Wei J W, Wang F Y, Zeng W X, et al. An embedded feature selection framework for control[C]. Proceedings of the 28th ACM SIGKDD Conference on Knowledge Discovery and Data Mining. New York, 2022: 1979-1988.

[7] Kohavi R, John G H. Wrappers for feature subset selection[J]. Artificial Intelligence, 1997, 97(1/2): 273-324.

[8] 赵玲, 龚加兴, 黄大荣, 等. 基于 Fisher Score 与最大信息系数的齿轮箱故障特征选择方法[J]. 控制与决策, 2021, 36(9): 2234-2240.
(Zhao L, Gong J X, Huang D R, et al. Fault feature selection method of gearbox based on Fisher Score and maximum information coefficient[J]. Control and

Decision, 2021, 36(9): 2234-2240.)

[9] Guyon I, Elisseeff A. An introduction to variable and feature selection[J]. Journal of Machine Learning Research, 2003, 3: 1157-1182.

[10] 贾鹤鸣, 姜子超, 李瑶. 基于改进秃鹰搜索算法的同步优化特征选择[J]. 控制与决策, 2022, 37(2): 445-454.
(Jia H M, Jiang Z C, Li Y. Simultaneous feature selection optimization based on improved bald eagle search algorithm[J]. Control and Decision, 2022, 37(2): 445-454.)

[11] Seyyedabbasi A. Binary sand cat swarm optimization algorithm for wrapper feature selection on biological data[J]. Biomimetics, 2023, 8(3): 310.

[12] Agrawal R K, Kaur B, Sharma S. Quantum based whale optimization algorithm for wrapper feature selection[J]. Applied Soft Computing, 2020, 89: 106092.

[13] Ghaemi M, Feizi-Derakhshi M R. Forest optimization algorithm[J]. Expert Systems with Applications, 2014, 41(15): 6676-6687.

[14] Ghaemi M, Feizi-Derakhshi M R. Feature selection using forest optimization algorithm[J]. Pattern Recognition, 2016, 60: 121-129.

[15] 初蓓, 李占山, 张梦林, 等. 基于森林优化特征选择算法的改进研究[J]. 软件学报, 2018, 29(9): 2547-2558.
(Chu B, Li Z S, Zhang M L, et al. Research on improvements of feature selection using forest optimization algorithm[J]. Journal of Software, 2018, 29(9): 2547-2558.)

[16] Xie Q, Cheng G G, Zhang X, et al. Feature selection using improved forest optimization algorithm[J]. Information Technology and Control, 2020, 49(2): 289-301.

[17] 刘兆赓, 李占山, 王丽, 等. 森林优化特征选择算法的增强与扩展[J]. 软件学报, 2020, 31(5): 1511-1524.
(Liu Z G, Li Z S, Wang L, et al. Enhancement and extension of feature selection using forest optimization algorithm[J]. Journal of Software, 2020, 31(5): 1511-1524.)

[18] 冀若含, 董红斌. 基于重复度分析的森林优化特征选择算法[J]. 智能系统学报, 2022, 17(6): 1113-1122.
(Ji R H, Dong H B. Feature selection using forest optimization algorithm based on duplication analysis[J]. CAAI Transactions on Intelligent Systems, 2022, 17(6): 1113-1122.)

[19] Al-Tashi Q, Abdul Kadir S J, Rais H M, et al. Binary optimization using hybrid grey wolf optimization for feature selection[J]. IEEE Access, 2019, 7: 39496-39508.

[20] Zhang Y, Song X F, Gong D W. A return-cost-based binary firefly algorithm for feature selection[J]. Information Sciences, 2017, 418: 561-574.

[21] Moustakidis S P, Theocharis J B. SVM-FuzCoC: A novel SVM-based feature selection method using a fuzzy complementary criterion[J]. Pattern Recognition, 2010, 43(11): 3712-3729.

作者简介

王霞 (1985−), 女, 副教授, 博士, 硕士生导师, 主要研究方向为智能优化算法, E-mail: wangxiacsu@163.com;

张珊 (1997−), 女, 硕士生, 主要研究方向为智能优化算法、机器学习, E-mail: zs18713963512@163.com;

王勇 (1999−), 男, 硕士生, 主要研究方向为机器学习、脑电信号处理, E-mail: 13253617767@163.com;

王卓然 (1999−), 女, 硕士生, 主要研究方向为智能优化算法, E-mail: 2417143098@qq.com.