计算机系统应用 ISSN 1003-3254, CODEN CSAOBN
Computer Systems & Applications,2022,31(4):204−212 [doi: 10.15888/j.cnki.csa.008437]

E-mail: csa@iscas.ac.cn
http://www.c-s-a.org.cn
Tel: +86-10-62661041

基于量子进化算法的包装式特征选择方法①

雷华军, 蒋　强

(乐山师范学院 电子与材料工程学院, 乐山 614000)
通信作者: 雷华军, E-mail: leihuajun2010@163.com

摘　要: 针对监督分类中的特征选择问题, 提出一种基于量子进化算法的包装式特征选择方法. 首先分析了现有子集评价方法存在过度偏好分类精度的缺点, 进而提出基于固定阈值和统计检验的两种子集评价方法. 然后改进了量子进化算法的进化策略, 即将整个进化过程分为两个阶段, 分别选用个体极值和全局极值作为种群的进化目标. 在此基础上, 按照包装式特征选择遵循的一般框架设计了特征选择算法. 最后, 通过 15 个 UCI 数据集分别验证了子集评价方法和进化策略的有效性, 以及新方法相较于其它 6 种特征选择方法的优越性. 结果表明, 新方法在 80% 以上的数据集上取得相似甚至更好的分类精度, 在 86.67% 的数据集上选择了特征个数更小的子集.

关键词: 监督分类; 特征选择; 特征子集评价; 进化策略; 量子进化算法; 机器学习

引用格式: 雷华军,蒋强.基于量子进化算法的包装式特征选择方法.计算机系统应用,2022,31(4):204–212. http://www.c-s-a.org.cn/1003-3254/8437.html

Wrapper Method for Feature Selection Based on Quantum-inspired Evolutionary Algorithm

LEI Hua-Jun, JIANG Qiang

(School of Electronics and Materials Engineering, Leshan Normal University, Leshan 614000, China)

Abstract: This study proposes a wrapper method based on a quantum-inspired evolutionary algorithm for feature selection in supervised classification. Firstly, it analyzes the shortcoming of excessively preferring classification accuracy in existing subset evaluation methods and then puts forward two new subset evaluation methods respectively based on a fixed threshold and a statistical test. Second, some improvements are made to the evolutionary strategy of the quantum-inspired evolutionary algorithm. More specifically, its whole evolutionary process is divided into two phases, in which individual and global extrems are selected as the evolutionary target of population respectively. On this basis, a feature selection algorithm is designed in accordance with the general wrapper framework. Finally, 15 UCI datasets are used to validate the effectiveness of the subset evaluation methods and the evolutionary strategy, as well as the superiority of the proposed method over other 6 feature selection methods. The results show that the new wrapper method achieves similar or even better classification accuracy in more than 80% of the datasets and selects feature subset with less number of features in 86.67% of the datasets.

Key words: supervised classification; feature selection; feature subset evaluation; evolutionary strategy; quantum-inspired evolutionary algorithm (QEA); machine learning

随着信息技术的飞速发展, 高维数据的监督分类问题在生产、生活和科研活动中大量涌现. 理论上讲, 增加特征的个数可以提供更多的分辨能力, 但研究表明: 数据维数的线性增加将使得问题的假设空间成指数增长, 特征每增加一维, 为确保学习算法的性能不变, 样本量亦需成倍增加. 另一方面, 在样本量一定的情况

① 收稿时间: 2021-06-20; 修改时间: 2021-07-14; 采用时间: 2021-08-13; csa 在线出版时间: 2022-03-22

下, 由于无关或冗余特征的存在, 过多的特征不仅降低学习算法的效率, 还可能导致过拟合, 致使模型的泛化能力下降[1]. 特征选择是解决前述问题的有效手段之一, 它根据某种评价标准, 从原始特征集中挑选一些最有效的特征以达到降低特征空间维数的目的, 并取得与原始特征集相似甚至更好的分类性能. 特征选择已成功应用在文本分类[2]、癌症识别[3]、设备故障检测[4]、贷款风险预测[5] 等领域.

纵观现有特征选择方法, 均包含特征子集生成和子集评价两个构成要素[6]. 前者采用一定的策略搜索特征空间中的全部或部分子集, 常用的方法有穷举搜索、启发式搜索和随机搜索. 后者使用某一标准评价搜索到的子集, 依据该标准是否独立于后续的学习算法, 可将特征选择方法分为过滤式 (filter)、包装式 (wrapper) 和嵌入式 (embedded) 三类. 相比过滤式, 包装式选择的子集通常拥有更优的分类性能, 因其直接使用特征子集的分类精度作为评价标准[7].

近年来, 国内外学者对基于随机搜索策略的包装式特征选择开展了广泛的研究, 大量新颖的群智能优化算法被用于特征选择[8], 其中较为典型的方法有粒子群优化[9]、蜻蜓算法[10]、灰狼优化[11]、鲸鱼优化[12]、引力搜索算法[13] 等. 这些方法大多从搜索策略的角度对原算法进行了有益的改进, 而有关子集评价的研究则较少.

量子进化算法 (quantum-inspired evolutionary algorithm, QEA) 是一种基于量子计算原理的概率进化算法, 具有种群规模小、收敛速度快和全局寻优能力强等优点[14]. 目前, 仅检索到一篇与本文主题密切相关的文献 [15], 该文将 QEA 作为特征子集的搜索策略, 并使用 Fisher 比评价其性能, 因而它属于一种过滤式的特征选择方法. 综上所述, 将 QEA 与包装式特征选择结合的相关研究尚处于空白, 本文将从子集评价和搜索策略两个方面开展工作, 力图提出一种具有竞争力的包装式特征选择方法.

1 问题描述

为结构完整性, 给出特征选择的形式化描述. 假定 $\boldsymbol{D}=\{(\boldsymbol{s}_1,y_1),(\boldsymbol{s}_2,y_2),\cdots,(\boldsymbol{s}_m,y_m)\}$ 为原始数据集, $\boldsymbol{S}=(\boldsymbol{s}_1,\boldsymbol{s}_2,\cdots,\boldsymbol{s}_m)^{\mathrm{T}}$ 为样本矩阵, 行和列分别代表不同的样本和特征, $\boldsymbol{Y}=(y_1,y_2,\cdots,y_m)^{\mathrm{T}}$ 为各样本对应的真实类别. $\boldsymbol{s}_i=(s_{i1},s_{i2},\cdots,s_{in})$ 是 n 维特征空间 χ 的一个向量, 即 $\boldsymbol{s}_i\in\chi$, $\chi=span\{f_1,f_2,\cdots,f_n\}$, 其中 f_j 表示第 j 个特征, 元素 s_{ij} $(1\leqslant i\leqslant m, 1\leqslant j\leqslant n)$ 为样本 $\boldsymbol{s}_i$ 在特征 f_j 上的取值. 类别 $y_i\in\boldsymbol{C}$, $\boldsymbol{C}=\{c_1,c_2,\cdots,c_k\}$ 为类别集合. m、n 和 k 分别为样本、特征和类别的个数.

基于上述定义, 监督分类的任务是: 利用给定的学习算法从 $\boldsymbol{D}$ 中学习一个分类模型 h: $\chi\rightarrow\boldsymbol{C}$, 它对采样自 χ 的新样本具有良好的泛化能力. 然而, 并不是所有特征都对分类有贡献, 为提高模型的训练效率和泛化能力, 特征选择从原始特征集 $\boldsymbol{F}=\{f_1,f_2,\cdots,f_n\}$ 中找出一个子集 $\boldsymbol{X}$, $\boldsymbol{X}\subseteq\boldsymbol{F}$, 它使得评价函数 $J(\boldsymbol{X})$ 尽量大, $\boldsymbol{X}$ 包含的特征数尽量少. $J(\boldsymbol{X})$ 量化了 $\boldsymbol{X}$ 分类性能的好坏, 不失一般性, 假定 $J(\boldsymbol{X})$ 值越大, $\boldsymbol{X}$ 越优. 不难看出, 分类性能和特征个数是该问题最重要的两个优化目标.

基于现有研究, 归纳出包装式特征选择的一般框架如图 1 所示.

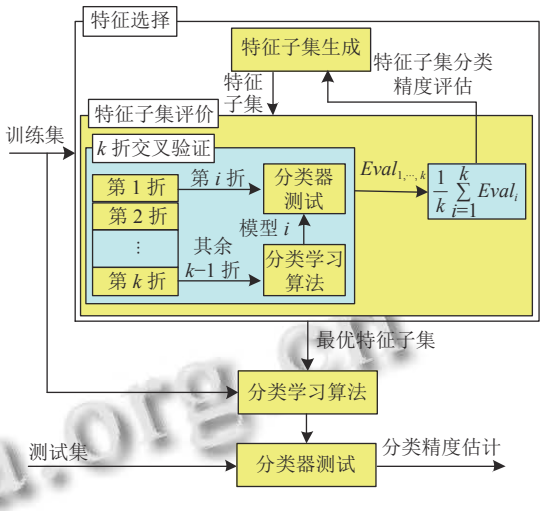


图 1 包装式特征选择的一般框架

由图 1 可以看出, 数据集被分为训练集和测试集, 特征选择在训练集上进行, 测试集不参与该过程. 特征选择结束, 需验证最优子集在测试集上的分类精度.在特征选择过程中, 子集的分类性能通过 K 折平均分类精度来衡量, 分类器被视为一个封闭的黑盒.

2 量子进化算法

基本量子进化算法包含种群初始化、随机观测、种群进化等 3 个主要步骤[14,16]. 它采用量子比特编码, 一个量子比特可表示为:

$$|\psi\rangle=\alpha|0\rangle+\beta|1\rangle \tag{1}$$

其中, $|0\rangle$ 和 $|1\rangle$ 表示两个不同的量子态, α 和 β 分别表示

状态|0⟩和|1⟩的概率幅. 一个量子比特可能处于|0⟩或|1⟩两个基态, 或者两基态的任意叠加态, $|\alpha|^2$和$|\beta|^2$表示该量子比特处于|0⟩和|1⟩的概率大小.

设第 t 代的种群为$\boldsymbol{Q}(t)=\left\{\boldsymbol{q}_1^t,\boldsymbol{q}_2^t,\cdots,\boldsymbol{q}_G^t\right\}$, 其中$G$为种群规模, $\boldsymbol{q}_j^t$为第 j $(1\leqslant j\leqslant G)$ 个采用量子比特编码的染色体, 其表达式如下:

$$\boldsymbol{q}_j^t=\begin{bmatrix}\alpha_{j1}^t & \alpha_{j2}^t & \cdots & \alpha_{jl}^t\\ \beta_{j1}^t & \beta_{j2}^t & \cdots & \beta_{jl}^t\end{bmatrix} \tag{2}$$

式中, $\left|\alpha_{jk}^t\right|^2+\left|\beta_{jk}^t\right|^2=1, 1\leqslant k\leqslant l$, l 为染色体长度. 采用量子比特编码, 一个长度为 l 的染色体能够同时表达 2^l 个状态的叠加, 因而更容易保持种群的多样性.

通过对$\boldsymbol{Q}(t)$的随机观测, 得到一组二进制状态$\boldsymbol{P}(t)=\left\{\boldsymbol{x}_1^t,\boldsymbol{x}_2^t,\cdots,\boldsymbol{x}_G^t\right\}$, 其中$\boldsymbol{x}_j^t=\left(x_{j1}^t,x_{j2}^t,\cdots,x_{jl}^t\right)$为观察$\boldsymbol{q}_j^t$的结果. 具体过程为: 对$\boldsymbol{q}_j^t$的每个量子比特$\left[\alpha_{jk}^t,\beta_{jk}^t\right]^{\mathrm{T}}$, 产生一个随机数 Rnd, $Rnd\in(0, 1)$, 若$Rnd<\left|\beta_{jk}^t\right|^2$, 则 x_{jk}^t=1, 否则, x_{jk}^t=0.

$\boldsymbol{P}(t)$通常代表问题的一组解, 计算每个$\boldsymbol{x}_j^t$的适应度$f\left(\boldsymbol{x}_j^t\right)$, 然后使用量子旋转门更新种群$\boldsymbol{Q}(t)$. 设$\boldsymbol{B}(t)=\left\{\boldsymbol{b}_1^t,\boldsymbol{b}_2^t,\cdots,\boldsymbol{b}_G^t\right\}$, 其中$\boldsymbol{b}_j^t=\left(b_{j1}^t,b_{j2}^t,\cdots,b_{jl}^t\right)$为$\boldsymbol{q}_j^t$截至第$t$代搜索到的个体最优解, 整个种群搜索到的全局最优解记为$\boldsymbol{b}^t=\left(b_1^t,b_2^t,\cdots,b_l^t\right)$. 对于$\boldsymbol{q}_j^t$, 第$k$ $(1\leqslant k\leqslant l)$ 位量子比特的更新方式如下:

$$\begin{bmatrix}\alpha_{jk}^{t+1}\\ \beta_{jk}^{t+1}\end{bmatrix}=\begin{bmatrix}\cos(\Delta\theta_k) & -\sin(\Delta\theta_k)\\ \sin(\Delta\theta_k) & \cos(\Delta\theta_k)\end{bmatrix}\begin{bmatrix}\alpha_{jk}^t\\ \beta_{jk}^t\end{bmatrix} \tag{3}$$

式中, $\left[\alpha_{jk}^{t+1},\beta_{jk}^{t+1}\right]^{\mathrm{T}}$为$q_j^{t+1}$的第$k$位量子比特, $\Delta\theta_k$为旋转角, 其取值由式 (4) 决定[16]:

$$\Delta\theta_k=\begin{cases}\theta\times\left\{\mathrm{sign}\left(\alpha_{jk}^t\beta_{jk}^t\right)\left(b_k^t-x_{jk}^t\right)\right\}, & f\left(\boldsymbol{x}_j^t\right)<f\left(\boldsymbol{b}^t\right)\\ 0, & \text{otherwise}\end{cases} \tag{4}$$

其中, $\mathrm{sign}(\cdot)$为符号函数, θ为$\Delta\theta_k$的旋转角幅值大小, 其取值一般为 0.001π~0.05π. 按照式 (3) 和式 (4) 更新$\boldsymbol{q}_j^t$的所有量子比特即得到$\boldsymbol{q}_j^{t+1}$. 重复上述步骤, 实现整个种群的更新.

3 特征选择方法

3.1 种群初始化

算法开始时, 设置进化代数 t=0, 初始化种群$\boldsymbol{Q}(0)=\left\{\boldsymbol{q}_1^0,\boldsymbol{q}_2^0,\cdots,\boldsymbol{q}_G^0\right\}$, 其中$\boldsymbol{q}_j^0$ $(1\leqslant j\leqslant G)$ 为第 j 个量子染色体, 其形式如式 (2), 设置染色体的长度为原始数据集的特征个数 n, 且其中的α_{jk}^0和$\beta_{jk}^0(1\leqslant k\leqslant n)$ 均初始化为$1/\sqrt{2}$. 按照这种初始化方法, 意味着 t=0 时每种特征组合以相同的概率 $(1/2^n)$ 叠加, 随机观测时, 每种特征组合出现的概率亦相等.

3.2 特征子集评价和性能比较

第 t 代, 通过对种群的随机观测, 得到 $\boldsymbol{P}(t)=\left\{\boldsymbol{x}_1^t,\boldsymbol{x}_2^t,\cdots,\boldsymbol{x}_G^t\right\}$, 则$\boldsymbol{x}_j^t$代表一个特征子集, 若$x_{ji}^t=1$, 表示特征$f_i(1\leqslant i\leqslant n)$ 选中, 否则f_i未选中. 利用$\boldsymbol{x}_j^t$和学习算法在训练集上进行 K 折交叉验证, 其结果记为$\boldsymbol{R}_j^t=(ac_1,ac_2,\cdots,ac_K)$, 其中$ac_k$为第 k 折的分类精度. 定义$\boldsymbol{x}_j^t$的 K 折平均分类精度Ave_j^t和特征个数N_j^t分别如式 (5) 所示:

$$Ave_j^t=\frac{1}{K}\sum_{i=1}^{K}ac_i \tag{5}$$

$$N_j^t=\sum_{i=1}^{n}x_{ji}^t \tag{6}$$

设$f\left(\boldsymbol{x}_j^t\right)$为$\boldsymbol{x}_j^t$的适应度函数, $\boldsymbol{x}_i^t$为不同于$\boldsymbol{x}_j^t$的任意一个特征子集, 为评价和比较$\boldsymbol{x}_i^t$和$\boldsymbol{x}_j^t$的优劣, 现有文献采用的主要方法如下:

方法 1[17]: $f\left(\boldsymbol{x}_j^t\right)=Ave_j^t$, 仅考虑特征子集的分类精度. 若$f\left(\boldsymbol{x}_j^t\right)>f\left(\boldsymbol{x}_i^t\right)$, 则$\boldsymbol{x}_j^t$优于$\boldsymbol{x}_i^t$.

方法 2[18]: $f\left(\boldsymbol{x}_j^t\right)=\left(Ave_j^t,N_j^t\right)$, 由$Ave_j^t$和$N_j^t$两个分量构成, 但前者的优先级更高. 若$Ave_j^t>Ave_i^t$, 或者$Ave_j^t=Ave_i^t$, 且$N_j^t<N_i^t$, 则$\boldsymbol{x}_j^t$优于$\boldsymbol{x}_i^t$.

方法 3[9−13]: $f\left(\boldsymbol{x}_j^t\right)=\alpha\cdot\left(1-Ave_j^t\right)+\beta\cdot N_j^t/n$, 通过加权的方式兼顾分类精度和特征个数, α和β为两者的权重, 取值分别为 0.99 和 0.01. 若$f\left(\boldsymbol{x}_j^t\right)<f\left(\boldsymbol{x}_i^t\right)$, 则$\boldsymbol{x}_j^t$优于$\boldsymbol{x}_i^t$.

通过实验发现: 相比方法 1, 在一些数据集上, 后两种方法确能得到分类精度高且特征个数更少的子集, 但仍旧偏好分类精度高的特征子集. 若$Ave_j^t>Ave_i^t$, 方法 2 直接判定$\boldsymbol{x}_j^t$优于$\boldsymbol{x}_i^t$; 根据α和β的取值, 方法 3 大多情况下也会得到相同的结论. 考虑这样的情形, 虽然$Ave_j^t>Ave_i^t$, 但两者并无显著的差异, 这时仅仅根据这种严格的大小关系就判定$\boldsymbol{x}_j^t$优于$\boldsymbol{x}_i^t$并不合理, 因为这种微小的差异可能是由 K 的取值及 K 折数据集的划分方式等因素造成, 而不是$\boldsymbol{x}_j^t$和$\boldsymbol{x}_i^t$分类性能差异的一种真实反映. 在两者分类精度无显著差异的情况下, 选择特

征个数更小的子集有助于减小分类模型的复杂度, 增强其泛化能力, 更好的兼顾分类精度和特征个数这两个优化目标.

根据以上分析, 提出两种新的子集评价方法:

方法 4: $f\left(\boldsymbol{x}_j^t\right)=\left(Ave_j^t, N_j^t\right)$, 若$Ave_j^t-Ave_i^t>\varepsilon$, 或者$abs\left(Ave_j^t-Ave_i^t\right)\leqslant\varepsilon$且$N_j^t<N_i^t$, 则$\boldsymbol{x}_j^t$优于$\boldsymbol{x}_i^t$, 其中$abs(\cdot)$为绝对值函数.

该方法采用固定阈值ε来判断两个子集的分类精度是否有显著差异. 当Ave_j^t和Ave_i^t的差值大于ε, 则认为$\boldsymbol{x}_j^t$和$\boldsymbol{x}_i^t$的分类精度有显著差异, 直接判定$\boldsymbol{x}_j^t$优于$\boldsymbol{x}_i^t$; 当Ave_j^t和Ave_i^t之差的绝对值不大于ε, 则认为$\boldsymbol{x}_j^t$和$\boldsymbol{x}_i^t$的分类精度相似, 特征个数越小的子集更优.

方法 5: $f\left(\boldsymbol{x}_j^t\right)=\left(\boldsymbol{R}_j^t, N_j^t\right)$, 若$Ave_j^t>Ave_i^t$, 且$p<delta$或者$p\geqslant delta$, 且$N_j^t<N_i^t$, 则$\boldsymbol{x}_j^t$优于$\boldsymbol{x}_i^t$, 其中 p 为$\boldsymbol{R}_j^t$和$\boldsymbol{R}_i^t$进行威尔科克森秩和检验 (Wilcoxon rank-sum test) 的 P 值, $delta$ 为显著性水平.

该方法采用统计检验来判断两个子集的分类精度是否有显著差异. 第一种情况表示Ave_j^t显著的大于Ave_i^t, 第二种情况表示Ave_j^t和Ave_i^t无显著差异, 但$N_j^t<N_i^t$, 满足这两种情况均判定$\boldsymbol{x}_j^t$优于$\boldsymbol{x}_i^t$. 注意, 此处引入非参数统计检验的目的是提出一种比较子集分类性能的启发式准则, 并非一般意义上的统计检验.

3.3 种群进化策略

为避免 QEA 早熟, 文献 [15] 采取随机初始化、两次观测取优和动态调整旋转角幅值等措施. 文献 [16] 指出: 在 QEA 中, 使用$\boldsymbol{b}^t$作为各个量子染色体的进化目标, 虽能使整个种群快速地向全局最优子集靠拢, 但容易陷入局部最优. 受此启发, 将整个算法的进化过程分为两个阶段, 前一阶段随机选择一个更优的个体极值作为进化目标, 以维持种群的多样性; 后一阶段使用$\boldsymbol{b}^t$作为所有个体的进化目标, 加速算法收敛.

(1) 随机选择更优个体极值的种群更新

Step 1. 对量子染色体$\boldsymbol{q}_j^t$ $(1\leqslant j\leqslant G)$ 进行随机观测, 得到二进制状态$\boldsymbol{x}_j^t$, 并计算$\boldsymbol{R}_j^t$, 再由式 (5) 和式 (6) 计算Ave_j^t和N_j^t.

Step 2. 按照第 3.2 节的方法 4 或 5 更新$\boldsymbol{b}_j^t$和$\boldsymbol{b}^t$.

Step 3. 使用方法 4 或 5 依次比较$\boldsymbol{x}_j^t$与所有的个体极值$\boldsymbol{b}_j^t$ $(1\leqslant j\leqslant G)$, 记性能优于$\boldsymbol{x}_j^t$的个体极值构成的集合为 temp.

Step 4. 从 temp 中随机选择一个个体极值$\boldsymbol{b}_j^t$, 作为$\boldsymbol{x}_j^t$的进化目标, 按照式 (3) 和式 (4) 更新$\boldsymbol{q}_j^t$.

Step 5. 重复 Step 1~Step 4, 直至完成 1 次种群进化.

(2) 基于全局极值的种群更新

Step 1. 按照前述相同的方法观测$\boldsymbol{q}_j^t$, 得到$\boldsymbol{x}_j^t$, 并计算$\boldsymbol{R}_j^t$、Ave_j^t和N_j^t.

Step 2. 更新$\boldsymbol{b}_j^t$和$\boldsymbol{b}^t$.

Step 3. 选择$\boldsymbol{b}^t$作为进化目标, 更新$\boldsymbol{q}_j^t$.

Step 4. 重复 Step 1~Step 3, 直至完成 1 次种群进化.

(3) 旋转角幅值的确定

在 QEA 中, θ为一个固定值, 不能很好的兼顾全局和局部搜索. 针对该问题, 提出如下方法:

$$\theta=\theta_{\max}-(\theta_{\max}-\theta_{\min})\times\frac{t}{T} \tag{7}$$

其中, $\theta_{\max}$和$\theta_{\min}$分别为θ的最大和最小可能取值, T 为算法的最大进化代数. 算法前期, θ取值较大, 使得算法专注于全局范围的搜索; 算法后期, θ取值较小, 使得算法集中于$\boldsymbol{b}^t$邻域内的搜索. 经过多次测试, 实验中$\theta_{\max}$和$\theta_{\min}$的取值分别为 0.04π和 0.0025π.

3.4 整个算法流程

综合第 3.2 节和第 3.3 节的改进措施和图 1, 得到整个特征选择方法的流程如图 2 所示.

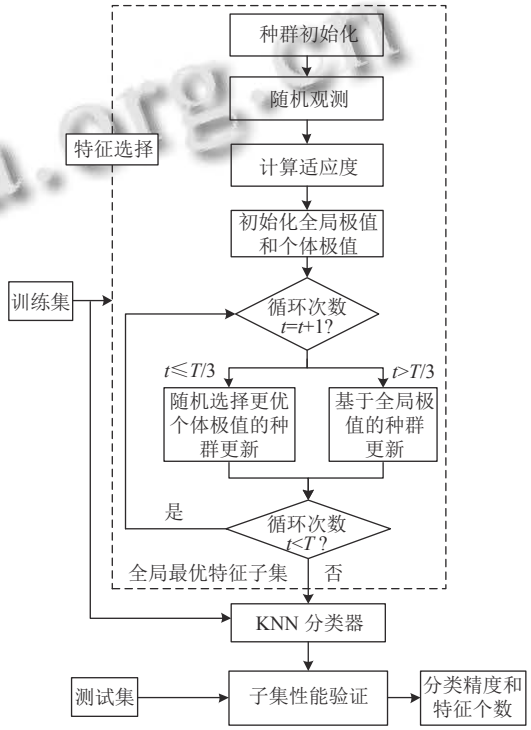


图 2 特征选择方法的流程

由图 2 可知, 实验中利用分层采样将数据集分为训练集和测试集. 在特征选择过程中, 两种进化策略分别执行 $T/3$ 和 $2T/3$ 代.

4 实验与结果分析

4.1 实验方案

选取 UCI[19] 中的 15 个数据集作为实验数据, 其中大多已被用于验证各种特征选择方法, 关于这些数据集的详细信息如表 1 所示. 注意, 前 11 个数据集以一个整体的形式提供, 实验中利用分层采样将其分为 70% 的训练集和 30% 的测试集; 后 4 个加“*”号的数据集以训练集和测试集的形式提供, 实验中直接使用训练集做特征选择, 测试集用于验证, 不再进行划分.

表 1 实验数据集

编号	数据集	样本数	特征数	类别数
1	biodegradation	1 055	41	2
2	breast	569	30	2
3	dermatology	366	34	6
4	waveform	5 000	40	3
5	sonar	208	60	2
6	movementlibras	360	90	15
7	ionosphere	351	34	2
8	spambase	4 601	57	2
9	semeion	1 593	256	2
10	LSVT	126	310	2
11	musk1	476	166	4
12	spect*	267	44	2
13	hillvalley*	606	100	2
14	segmentation*	2 310	19	7
15	landsat*	6 435	36	6

包装式特征选择方法使用分类器来评价特征子集, 后续实验均使用 K 近邻 (K-nearest neighbor, KNN) 学习算法, 这有利于算法之间公平的比较[9–13]; KNN 的距离度量使用欧氏距离, 近邻个数为 5. 所有实验均在同一台 PC 机上进行, 配置为: Intel(R) Core(TM) i7-9700 CPU@ 3.00 GHz, 8 GB 内存, Win10 64 位操作系统, 编程环境为 Matlab 9.9 (R2020b). 为了结果的客观性, 若无特别说明, 对于每个数据集, 算法均独立运行 20 次, 以便评估算法的性能, 表中所示分类精度和特征个数是 20 次独立试验后的平均值.

4.2 重要参数的选择

子集评价方法 4 和 5 分别依赖于参数ε和*delta*的取值, 它们直接影响显著性的判定. 选取表 1 中的前 5 个数据集进行参数的讨论, 假定ε的可能取值为{0.001, 0.005, 0.010, 0.050, 0.100, 0.500}, *delta*的可能取值为{0.05, 0.10, 0.15, 0.20}. 为选择一组合适的参数, 进化策略采用基本 QEA, 将其分别与子集评价方法 4 和 5 相结合构成 2 种特征选择算法. 其它的算法参数为: 种群规模 G=10, 最大进化代数 T=50, 交叉验证次数 K=10, 旋转角幅值θ=0.01π. 不同参数取值对应的平均分类精度和特征个数如图 3 所示.

从图 3(a) 可以看出, 除了 waveform, 在其它数据集上平均特征个数随ε增大而减小, ε的值越小越好; 考察分类精度, 其值亦呈现相同的趋势, 但前 3 种ε值对应的分类精度在 5 个数据集上均无显著差异, 后 3 种ε值对应的分类精度则显著下降. 综合考虑, 选取ε的值为 0.010. 同理分析图 3(b), 平均特征个数随*delta*增大而增大, *delta* 的值越小越好; 从分类精度的角度, 仅有 *delta*=0.10 时对应的分类精度在 5 个数据集上与*delta*的其余 4 种取值的最优分类精度均无显著差异, 故选取 *delta* 值为 0.10.

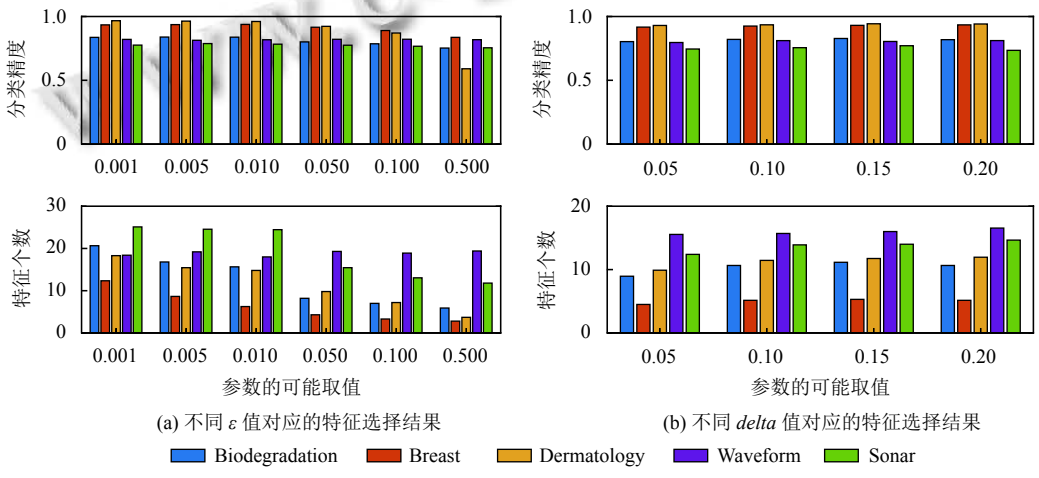


图 3 重要参数的结果对比

4.3 子集评价方法对比

为对比不同的子集评价方法, 搜索策略采用基本 QEA, 将其分别与 5 种子集评价方法相结合构成 5 种特征选择算法, 算法参数同第 4.2 节. 表 2 为 5 种特征选择方法的运行结果, 其中 P_{ij} 表示对方法 i (i=1, 2, 3) 和方法 j (j=4, 5) 的 20 次分类精度进行威尔科克森秩和检验的 P 值, 显著性水平为 0.05. “+”和“–”分别表示前者的分类精度显著好于和差于后者, 而“=”则表示两者的分类精度相似, 没有显著差异, 表中最后一行统计了对应列“+”“–”和“=”的个数.

表 2 5 种子集评价方法的分类精度和特征个数

编号	方法1				方法2				方法3				方法4		方法5	
	分类精度(%)	特征个数	P_{14}	P_{15}	分类精度(%)	特征个数	P_{24}	P_{25}	分类精度(%)	特征个数	P_{34}	P_{35}	分类精度(%)	特征个数	分类精度(%)	特征个数
1	**84.15**	20.75	0.371	0.005	83.54	20.75	0.935	0.028	83.77	21.55	0.715	0.015	83.56	15.50	81.69	**9.75**
2	94.06	14.45	0.643	0.237	**94.15**	13.80	0.902	0.226	93.97	11.95	0.774	0.258	94.12	7.75	93.12	**4.65**
3	95.69	20.35	0.013	0.001	96.51	19.10	0.590	0.000	96.79	18.00	0.889	0.000	**96.83**	16.40	92.98	**10.05**
4	82.04	23.80	1.000	**0.001**	81.82	24.40	0.525	0.002	**82.05**	22.65	1.000	0.000	82.03	19.00	80.47	**16.30**
5	76.53	29.15	0.145	0.496	76.37	26.35	0.205	0.521	77.50	28.50	0.568	0.259	**78.87**	24.65	75.56	**13.00**
6	70.19	44.25	0.839	0.497	**71.34**	41.65	0.355	0.187	70.42	44.60	0.881	0.704	70.28	37.70	69.91	**22.10**
7	86.00	11.65	0.693	0.523	86.29	12.30	0.957	0.765	85.29	12.25	0.399	0.321	86.67	8.85	**86.86**	**4.05**
8	**91.37**	34.80	0.005	0.000	91.14	31.75	0.058	0.000	91.02	31.00	0.176	0.001	90.69	25.95	89.63	**21.15**
9	95.89	129.25	0.734	0.004	**96.05**	130.85	0.455	0.001	96.04	129.70	0.356	0.002	95.77	107.55	94.79	**92.85**
10	**63.92**	153.70	0.913	0.816	61.62	154.35	0.556	0.670	63.65	153.40	0.763	0.945	63.78	149.15	**63.92**	**113.00**
11	84.79	82.30	0.568	0.222	84.93	84.75	0.924	0.255	84.68	82.95	0.694	0.266	**85.07**	74.15	83.20	**54.10**
12	60.32	21.05	0.049	0.073	**61.79**	18.35	0.734	0.828	61.50	17.90	0.673	0.463	61.63	17.55	61.68	**7.95**
13	52.29	50.20	0.010	0.015	52.42	47.85	0.015	0.023	**53.51**	46.95	0.776	0.423	53.42	39.60	53.19	**26.95**
14	90.95	9.60	1.000	0.000	90.98	8.05	0.924	0.000	**91.06**	8.10	0.488	0.000	91.00	6.25	88.28	**3.85**
15	**90.14**	22.75	0.000	0.000	90.12	21.35	0.000	0.000	89.96	20.15	0.000	0.000	89.17	12.15	88.45	**10.35**
–/=/+			3/10/2	1/7/7			1/13/1	1/7/7			0/14/1	0/8/7				

由表 2 可知, 相比子集评价方法 1、方法 2 和方法 3, 方法 4 至少在 13 个数据集上取得相似甚至更好的分类精度; 而方法 5 仅在 8 个数据集上表现出同样的性能, 在其余 7 个数据集上分类精度要显著的差于方法 1、方法 2 和方法 3. 从特征个数的角度, 不难看出方法 5 总是取得最优的结果, 其次是方法 4. 为更具体的说明, 以 breast 数据集为例, 给出其 20 次实验中的分类精度和特征个数的箱线图, 如图 4 所示.

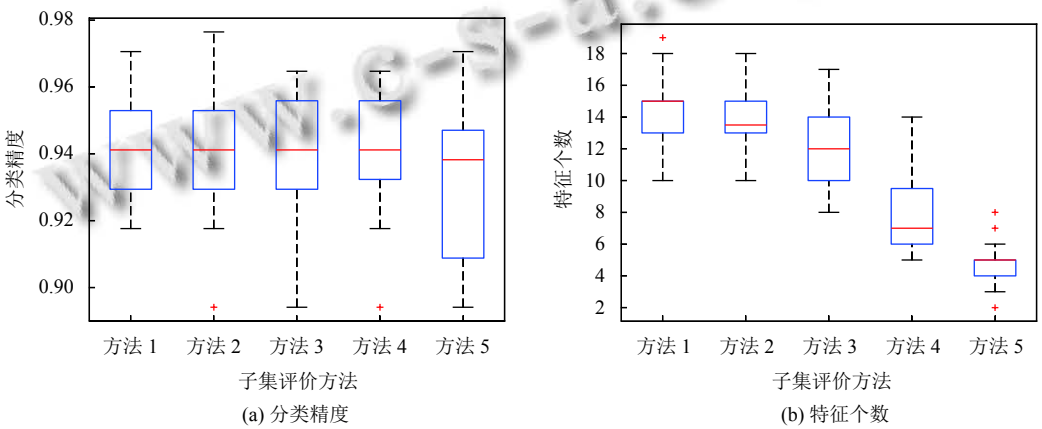


图 4 子集评价方法在 breast 数据集上的对比结果

从图 4(a) 可以看出, 前 4 种方法对应分类精度的中位值和平均值大致相当, 难以直观判断哪种方法更优, 但均优于方法 5 的分类精度; 从图 4(b) 可以很容易看出方法 5 选择的特征个数最少, 其次是方法 4. 综合

两者, 方法 4 能在确保分类精度无显著差异的情况下, 选择特征个数更小的子集, 表明针对子集评价方法的改进是有效的, 后续实验均采用方法 4 作为子集评价方法.

4.4 进化策略的有效性验证

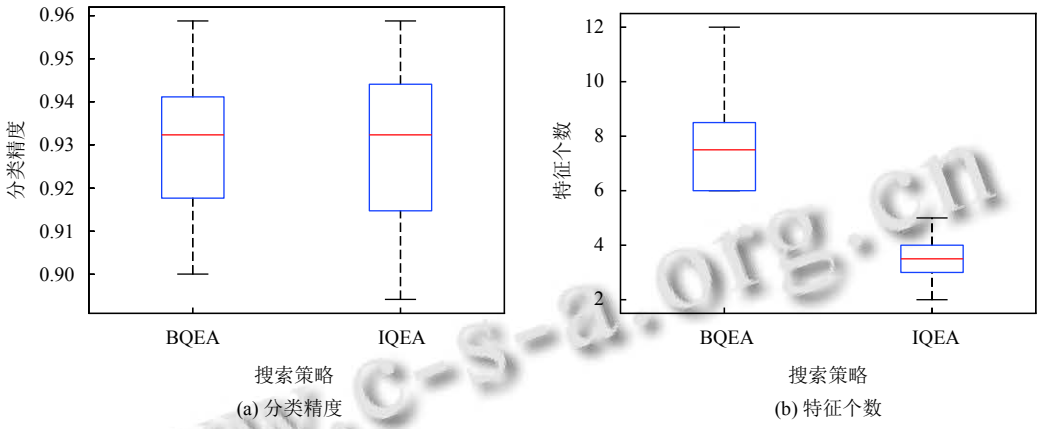
采用子集评价方法 4, 将其分别与基本 QEA 和改进 QEA 相结合构成 2 种特征选择方法, 算法参数同第 4.2 节. 为叙述方便, 分别简记为 BQEA 和 IQEA, 其运行结果如表 3 所示, 其中P为针对分类精度的威尔科克森秩和检验的 P 值, “+”和“–”分别表示 BQEA 的分类精度显著的好于和差于 IQEA, 而“=”则表示两者的分类精度相似.

由表 3 可以看出, 在 Semeion 和 Landsat 数据集上, BQEA 的分类精度显著优于 IQEA, 在 Ionosphere 和 Hillvalley 数据集上的结论正好相反, 在其余数据集上两者的分类精度均相似. 从特征个数来看, IQEA 的结果总是优于 BQEA 对应的结果. 同样以 Breast 数据集为例, 其分类精度和特征个数的箱线图如图 5 所示.

由图 5(a) 可以看出, BQEA 和 IQEA 的分类精度大致相当, 无法直观判断哪种方法更优; 从图 5(b) 却可以直观的看出 IQEA 选择的特征个数小于 BQEA 对应的结果. 由实验的设置可知, BQEA 和 IQEA 的差异仅在于进化策略的不同, 表明针对进化策略的改进措施是有效的.

表 3 BQEA 和 IQEA 的运行结果

编号	BQEA			IQEA	
	分类精度 (%)	P	特征个数	分类精度 (%)	特征个数
1	**83.09**	0.645	15.55	82.64	**11.55**
2	**93.09**	0.967	7.75	92.97	**3.50**
3	**96.83**	0.496	16.85	96.56	**13.00**
4	**81.94**	0.685	18.65	81.79	**10.15**
5	76.53	0.157	25.10	**78.79**	**16.50**
6	69.91	0.797	40.05	**69.95**	**33.25**
7	86.57	0.009	9.15	**89.43**	**3.80**
8	90.80	0.401	25.80	**90.99**	**23.05**
9	**95.83**	0.029	110.60	94.95	**69.00**
10	**61.22**	0.978	148.10	60.68	**145.45**
11	86.34	0.192	76.85	**84.65**	**61.35**
12	**62.25**	0.616	17.85	**62.25**	**16.55**
13	53.02	0.009	40.80	**54.18**	**33.50**
14	90.99	0.497	7.25	**91.04**	**5.95**
15	**89.18**	0.003	12.75	88.44	**9.30**
–/=/+		2/11/2			



(a) 分类精度 (b) 特征个数

图 5 BQEA 和 IQEA 在 breast 数据集上的对比结果

4.5 特征选择方法的对比

选取近年来新提出的包装式和过滤式特征选择方法作为对比, 包括 HPSOSSM[9]、SBDA[10]、ABGWO[11]、HMNWOA[12]、HGSA[13] 和 FS-IQEA[15], 其中前 5 种方法属于包装式, 后 1 种方法为过滤式. 为比较的公平性, 设置所有算法的种群规模 G=20、进化代数 T=60, 交叉验证次数 K=10, 每种方法其余的参数按照对应文献给定的值设置. 这些方法连同 IQEA 的运行结果如表 4 所示, 其中P_i (i=1, 2, 3, 4, 5, 6) 分别表示 6 种对比方法和 IQEA 的分类精度做威尔科克森秩和检验的 P 值, “+”“–”和“=”的含义同前.

由表 4 可知, 分类精度方面, IQEA 显著优于 HPSOSSM、SBDA、ABGWO、HMNWOA、HGSA 和 FS-IQEA 的数据集占比分别为 26.67%、20.00%、26.67%、13.33%、20.00% 和 66.67%; 取得相似分类精度的数据集占比分别为 53.33%、66.67%、60.00%、73.33%、66.67% 和 33.33%. 综合两者, IQEA 与比较方法取得相似甚至更好分类性能的数据集占比分别为 80.00%、86.67%、86.67%、86.67%、86.67% 和 100%.

表 4 IQEA 与其它方法的对比结果

编号	HPSOSSM			SBDA			ABGWO			HMNWOA			HGSA			FS-IQEA			IQEA	
	分类精度(%)	P_1	特征个数	分类精度(%)	P_2	特征个数	分类精度(%)	P_3	特征个数	分类精度(%)	P_4	特征个数	分类精度(%)	P_5	特征个数	分类精度(%)	P_6	特征个数	分类精度(%)	特征个数
1	83.04	0.935	20.35	83.80	0.356	20.10	82.90	0.635	21.60	83.20	0.807	19.50	**83.92**	0.218	19.65	69.62	0.000	**2.60**	83.20	10.80
2	92.97	0.521	10.30	**93.65**	0.784	13.85	92.97	0.549	13.85	93.53	0.881	12.05	93.44	1.000	12.90	88.74	0.000	**1.10**	93.26	3.05
3	96.42	0.418	21.80	96.33	0.583	19.55	96.47	0.481	20.60	96.70	0.501	21.15	95.32	0.021	17.55	48.85	0.000	**1.60**	**96.97**	12.45
4	81.13	0.015	32.55	80.88	0.004	22.80	81.68	0.482	25.75	81.75	0.357	25.10	81.23	0.081	22.85	54.97	0.000	**2.45**	**81.88**	10.40
5	76.77	0.445	22.65	78.23	0.913	27.45	77.10	0.512	31.15	78.31	0.817	24.45	**78.63**	0.807	28.85	71.21	0.004	**6.90**	78.15	16.60
6	71.11	0.871	44.30	71.44	0.913	44.15	70.69	0.654	48.25	69.91	0.271	39.20	70.65	0.683	43.75	68.70	0.069	**15.45**	**71.53**	34.20
7	86.86	0.034	5.50	84.48	0.000	12.15	82.95	0.000	16.40	86.00	0.020	7.35	84.71	0.001	12.20	81.90	0.002	**1.80**	**88.76**	4.00
8	89.34	0.020	38.55	90.38	0.021	30.95	90.13	0.238	35.20	**91.24**	0.110	35.10	91.18	0.129	31.10	62.35	0.000	**6.20**	90.85	21.60
9	95.51	0.001	148.20	**95.97**	0.000	133.00	95.74	0.000	135.25	95.96	0.000	138.25	95.50	0.000	127.70	93.16	0.059	75.70	94.14	**65.40**
10	**72.30**	0.000	130.15	63.65	0.396	152.80	61.49	1.000	154.75	62.70	0.556	131.60	62.43	1.000	152.95	63.38	0.390	**95.90**	61.76	143.55
11	**84.86**	0.881	81.75	83.77	0.597	82.25	83.80	0.542	86.30	84.72	0.935	82.75	83.94	0.645	79.30	80.21	0.004	**39.85**	84.30	52.10
12	62.91	0.228	14.10	60.61	0.003	18.25	58.26	0.000	22.70	61.34	0.015	14.30	61.28	0.012	17.90	61.60	0.107	**2.90**	**63.85**	12.20
13	53.65	0.551	18.75	53.23	0.081	48.40	52.80	0.004	50.90	53.40	0.144	42.15	53.32	0.107	48.15	53.87	0.725	**17.30**	**54.04**	31.75
14	90.17	0.000	9.10	90.87	0.935	8.70	90.05	0.000	9.95	90.89	0.807	8.30	90.83	0.724	8.35	58.02	0.000	**1.50**	**90.93**	5.20
15	90.01	0.000	24.20	89.97	0.000	19.65	89.90	0.000	21.20	**90.05**	0.000	20.20	89.85	0.000	18.85	55.19	0.000	**1.55**	88.63	9.20
–/=/+		4/8/3			3/10/2			4/9/2			2/11/2			3/10/2			10/5/0			

从特征个数来看, 除了 Semeion 数据集, FS-IQEA 选择的子集总是包含最少的特征, 但其对应的平均分类精度亦小于包装式方法对应的结果, 从而证实了“相比过滤式, 包装式选择的子集通常拥有更优的分类性能”的一般结论. 若仅考察包装式方法, 不难看出, 除了 LSVT 和 hillvalley 数据集 , IQEA 在其余数据集上选择的特征个数均最少, 占总数据集的 86.67%.

综合考虑分类精度和特征个数, 相比现有包装式特征选择方法, 可以看出 IQEA 在大多数情况下能取得相似甚至更好的分类性能, 并且选择的特征个数更少, 降维效果更加明显.

5 结论与展望

子集评价和搜索策略是构成包装式特征选择方法的两个关键要素, 论文从这两个角度出发, 分别提出了改进的子集评价方法和搜索策略, 进而设计了一种

基于量子进化算法的包装式特征选择方法. 大量的实验结果表明, 相比现有特征选择方法, 本文方法在大多数情况下能搜索到更小的特征子集, 并能取得相似甚至更好的分类精度, 降维效果突出. 同时表明, 要提高包装式特征选择方法的性能, 除了现有文献广泛关注的搜索策略, 子集评价方法也应重点关注.

需要指出, 随着样本量和特征个数的增加, 包装式特征选择方法的计算量将显著增加, 甚至根本不适用. 实验中所用数据集的特征个数均不超过 1 000, 因此如何进一步提高算法的性能, 使其适用于更高维 (特征个数>1000) 数据的场合, 例如基因选择, 将是下一步要研究的内容.

参考文献

1 Xue B, Zhang MJ, Browne WN, *et al*. A survey on evolutionary computation approaches to feature selection. IEEE Transactions on Evolutionary Computation, 2016, 20(4): 606–626. [doi: 10.1109/TEVC.2015.2504420]

2 Ghareb AS, Bakar AA, Hamdan AR. Hybrid feature selection based on enhanced genetic algorithm for text categorization. Expert Systems with Applications, 2016, 49: 31–47. [doi: 10.1016/j.eswa.2015.12.004]

3 Jain I, Jain VK, Jain R. Correlation feature selection based improved-binary particle swarm optimization for gene selection and cancer classification. Applied Soft Computing, 2018, 62: 203–215. [doi: 10.1016/j.asoc.2017.09.038]

4 韩金鹏, 李冬梅, 王嵩. 基于 PSO_RF 双向特征选择和 LightGBM 设备故障检测. 计算机系统应用, 2020, 29(7): 228–232. [doi: 10.15888/j.cnki.csa.007479]

5 张涛, 范博. 基于 CLPSO-CatBoost 的贷款风险预测方法. 计算机系统应用, 2021, 30(4): 222–226. [doi: 10.15888/j.cnki.csa.007866]

6 Dash M, Liu H. Feature selection for classification. Intelligent Data Analysis, 1997, 1(1–4): 131–156. [doi: 10.1016/S1088-467X(97)00008-5]

7 李郅琴, 杜建强, 聂斌, 等. 特征选择方法综述. 计算机工程

与应用, 2019, 55(24): 10–19. [doi: 10.3778/j.issn.1002-8331.1909-0066]

8 Nguyen BH, Xue B, Zhang MJ. A survey on swarm intelligence approaches to feature selection in data mining. Swarm and Evolutionary Computation, 2020, 54: 100663. [doi: 10.1016/j.swevo.2020.100663]

9 Chen K, Zhou FY, Yuan XF. Hybrid particle swarm optimization with spiral-shaped mechanism for feature selection. Expert Systems with Applications, 2019, 128: 140–156. [doi: 10.1016/j.eswa.2019.03.039]

10 Hammouri AI, Mafarja M, Al-Betar MA, *et al*. An improved dragonfly algorithm for feature selection. Knowledge-Based Systems, 2020, 203: 106131. [doi: 10.1016/j.knosys.2020.106131]

11 Hu P, Pan JS, Chu SC. Improved binary grey wolf optimizer and its application for feature Selection. Knowledge-Based Systems, 2020, 195: 105746. [doi: 10.1016/j.knosys.2020.105746]

12 Guo WY, Liu T, Dai F, *et al*. An improved whale optimization algorithm for feature selection. Computers, Materials & Continua, 2020, 62(1): 337–354. [doi: 10.32604/cmc.2020.06411]

13 Taradeh M, Mafarja M, Heidari AA, *et al*. An evolutionary gravitational search-based feature selection. Information Sciences, 2019, 497: 219–239. [doi: 10.1016/j.ins.2019.05.038]

14 Han KH, Kim JH. Quantum-Inspired evolutionary algorithm for a class of combinatorial optimization. IEEE Transactions on Evolutionary Computation, 2002, 6(6): 580–593. [doi: 10.1109/TEVC.2002.804320]

15 周丹, 吴春明. 基于改进量子进化算法的特征选择. 计算机工程与应用, 2018, 54(1): 146–152.

16 雷华军, 秦开宇. 测试不可靠条件下基于量子进化算法的测试优化选择. 电子学报, 2017, 45(10): 2464–2472. [doi: 10.3969/j.issn.0372-2112.2017.10.022]

17 Kashef S, Nezamabadi-Pour H. An advanced ACO algorithm for feature subset selection. Neurocomputing, 2015, 147: 271–279. [doi: 10.1016/j.neucom.2014.06.067]

18 Xue B, Zhang MJ, Browne WN. Particle swarm optimisation for feature selection in classification: Novel initialisation and updating mechanisms. Applied Soft Computing, 2014, 18: 261–276. [doi: 10.1016/j.asoc.2013.09.018]

19 Dua D, Graff C. UCI machine learning repository [Technical report]. University of California. http://archive.ics.uci.edu/ml, 2019.