



文章编号:1001-9081(2018)11-3156-05

DOI:10.11772/j.issn.1001-9081.2018041358

基于多目标骨架粒子群优化的特征选择算法

张翠军¹, 陈贝贝¹, 周冲^{1*}, 尹心歌²

(1. 河北地质大学 信息工程学院, 石家庄 050031; 2. 天津工业大学 管理学院, 天津 300387)

(*通信作者电子邮箱 zc0315@foxmail.com)

摘要:针对在分类问题中,数据之间存在大量的冗余特征,不仅影响分类的准确性,而且会降低分类算法执行速度的问题,提出了一种基于多目标骨架粒子群优化(BPSO)的特征选择算法,以获取在特征子集个数与分类精确度之间折中的最优策略。为了提高多目标骨架粒子群优化算法的效率,首先使用了一个外部存档,用来引导粒子的更新方向;然后通过变异算子,改善粒子的搜索空间;最后,将多目标骨架粒子群算法应用到特征选择问题中,并利用K近邻(KNN)分类器的分类性能和特征子集的个数作为特征子集的评价标准,对UCI数据集以及基因表达数据集的12个数据集进行实验。实验结果表明,所提算法选择的特征子集具有较好的分类性能,最小分类错误率最大可以降低7.4%,并且分类算法的执行时间最多能缩短12 s,能够有效提高算法的分类性能与执行速度。

关键词:特征选择;K近邻分类器;骨架粒子群优化算法

中图分类号:TP391.4 **文献标志码:**A

Feature selection algorithm based on multi-objective bare-bones particle swarm optimization

ZHANG Cuijun¹, CHEN Beibei¹, ZHOU Chong^{1*}, YIN Xinge²

(1. School of Information Engineering, Hebei GEO University, Shijiazhuang Hebei 050031, China;

2. School of Management, Tianjin Polytechnic University, Tianjin 300387, China)

Abstract: Concerning there are a lot of redundant features classified in data which not only affect the classification accuracy, but also reduce classification speed, a feature selection algorithm based on multi-objective Bare-bones Particle Swarm Optimization (BPSO) was proposed to obtain the tradeoff between the number of feature subsets and the classification accuracy. In order to improve the efficiency of the multi-objective BPSO, firstly an external archive was used to guide the update direction of the particle, and then the search space of the particle was improved by a mutation operator. Finally, the multi-objective BPSO was applied to feature selection problems, and the classification performance and the number of selected features of the K Nearest Neighbors (KNN) classifier were used as feature selection criteria. The experiments were performed on 12 datasets of UCI datasets and gene expression datasets. The experimental results show that the feature subset selected by the proposed algorithm has better classification performance, the maximum error rate of the minimum classification can be reduced by 7.4%, and the maximum execution speed of the classification algorithm can be shortened by 12 s at most.

Key words: feature selection; K Nearest Neighbor (KNN) classifier; Bare-bones Particle Swarm Optimization (BPSO)

0 引言

特征选择是数据挖掘以及模式识别中一种重要的数据预处理步骤^[1-2]。高维数据特征往往包含大量冗余特征、不相关的和噪声特征。在数据分类问题中,分类性能常常被冗余的、不相关的和噪声特征影响,并增加了计算成本。特征选择的目的是选择出最少的、相关的和有用的特征,以便提高分类性能和降低计算成本。

根据特征子集的选择策略,特征选择方法可以分为两类:过滤式特征选择方法^[3]和封装式特征选择方法^[4-5]。过滤式特征选择的评价标准从数据集本身的内部特性获得,与学习算法无关,通常选择与类别相关度大的特征或特征子集;封装式特征选择方法是利用分类算法的性能来评价特征子集的优劣,采用搜索策略来寻找最优子集。过滤式特征选择方法由

于特征选择的标准与学习算法无关,不需要分类器的训练步骤,因此其通用性比封装式特征选择强,复杂度比封装式特征选择方法低,但是过滤式特征子集在分类准确率方面比封装式特征选择方法低。

优化技术,像进化算法(Evolutionary Algorithm, EA)、遗传算法(Genetic Algorithm, GA)、粒子群优化(Particle Swarm Optimization, PSO)算法、蚁群优化(Ant Colony Optimization, ACO)算法、人工蜂群(Artificial Bee Colony, ABC)算法和差分进化(Differential Evolution, DE)算法等,有很多已经成功应用到各种应用领域的特征选择问题^[6-12]中。

大多数特征选择问题都是针对单目标来解决的,仅考虑了算法的分类准确率,没有考虑选择特征子集的大小。虽然在一些单目标特征选择研究^[13-16]中,通过聚合函数形成一个

收稿日期:2018-04-30;修回日期:2018-06-08;录用日期:2018-06-29。 基金项目:国家自然科学基金资助项目(61402481);河北省青年拔尖人才支持计划项目(冀字[2013]17号);河北省教育厅科学技术研究重点项目(ZD2018083)。

作者简介:张翠军(1968—),女,河北石家庄人,教授,硕士,主要研究方向:智能计算、机器学习; 陈贝贝(1993—),女,河北石家庄人,硕士研究生,主要研究方向:智能计算、机器学习; 周冲(1989—),男,河北邯郸人,讲师,博士,主要研究方向:智能计算、多目标优化; 尹心歌(1999—),女,河北石家庄人,主要研究方向:信息管理、信息系统。



目标, 达到同时考虑分类性能和选择的特征子集大小的目的, 但是这样引入了新的聚合参数, 导致算法不灵活、设置参数等问题, 因此, 本文将特征选择作为多目标优化问题。特征选择有两个主要目标, 即最大化分类准确率(最小化分类错误率)和最小化特征选择的个数, 由此, 特征选择问题可以表示为两个目标的最小化问题。一般来说, 这两个目标函数是相互矛盾的, 需要在它们之间进行折中来寻找最优策略。Xue 等^[17]首次将改进的多目标 PSO 算法应用到特征选择中, 提出了一种基于 PSO 算法的帕累托(Pareto)前沿面特征选择方法, 并通过拥挤距离、变异操作以及支配集寻找 Pareto 前沿面。实验结果表明, 该方法分类性能优于其他特征选择方法。Xue 等的算法中将融合多种策略的多目标 PSO 算法应用到特征选择中, 提高了算法的性能, 但其存在着大量需要调节的参数, 导致收敛速度过慢、计算时间长的问题, 使算法的性能下降。

为了达到选择的特征子集个数最小, 且算法的分类错误率最小的目的, 本文设计了一种基于多目标骨架粒子群(Multi-Objective Bare-bones Partivle Swarm Optimization, MOBPSO)算法用于特征选择方法中, 并通过约束机制和变异算子来提高 BPSO 算法的搜索能力, 在算法的执行过程中, 通过外部集合记录与维护 Pareto 最优解, 并通过 K 最近邻(K Nearest Neighbor, KNN)(K=3)方法来评估粒子的适应度值。

1 骨架粒子群算法

2003 年 Kennedy^[18]提出了一种变异的骨架粒子群(Bare-bones Particle Swarm Optimization, BPSO)算法, 它移除了速度更新公式, 在搜索过程中仅使用粒子的位置信息。BPSO 更新位置信息时, 所有的粒子都从粒子个体历史最优值和全局最优值进行学习。BPSO 位置信息更新公式如式(1)所示:

$$x_{id}^{t+1} = N\left(\frac{Pbest_{id}^t + Gbest_d^t}{2}, |Pbest_{id}^t - Gbest_d^t|\right) \quad (1)$$

其中, 位置信息根据高斯分布随机产生, 高斯分布的均值为 $(Pbest + Gbest)/2$, 方差为 $|Pbest - Gbest|$ 。Kennedy 还提出了一种 BPSO 的变异版本 BPSOE(Exploiting Bare-bones PSO), 这种方法更倾向于搜索到历史最优位置。位置信息更新公式如式(2)所示:

$$x_i^d(t+1) = \begin{cases} N(\mu_i^d(t), \sigma_i^d(t)), & R < 0.5 \\ Pbest_i^d(t), & \text{其他} \end{cases} \quad (2)$$

其中 R 是 $[0, 1]$ 区间的随机数。在 BPSOE 中粒子有 50% 的机会跳到自己的最优位置。

由于 BPSO 算法无需调参, 比 PSO 算法更简单, 而且目前还没有将多目标 BPSO 算法应用到特征选择问题中, 因此, 本文提出将多目标 BPSO 算法应用到特征选择方法中。

2 基于多目标骨架粒子群的特征选择算法

本文提出了一种基于多目标骨架粒子群算法特征选择算法, 该算法结合 KNN($K=3$)分类器在 12 个数据集上进行测试。

2.1 种群的编码、解码与评价

种群中每个粒子对应特征选择的一个解, 粒子采用实数编码, 如式(3)所示, 每个解 X 中包含 n 个实数, n 为对应数据

集特征的总特征数, 其中每一维 x_i 代表是否选择该特征。

$$X = [x_1, x_2, \dots, x_n] \quad (3)$$

为了形成特征子集, 需要在解码前进行解码处理。粒子的位置可以转换成如下的特征子集:

$$A_d = \begin{cases} 1, & x_d > 0.5 \\ 0, & \text{其他} \end{cases}$$

其中, A_d 代表从每个解第 d 维解码出来的特征子集。 A_d 可以根据粒子第 d 维特征的值 x_d 选择其值为 0 或者是 1: 如果 $A_d = 1$, 代表第 d 维特征被选择; 否则, 该维特征不被选择。

将每个粒子解码转换为数据集中所有特征的选择方案, 然后采用 KNN($K=3$)分类器对每个粒子对应的选择方案的分类性能进行测试, 并按照式(4)计算该种选择方案对应的分类错误率:

$$ER = \frac{FP + FN}{FP + FN + TP + TN} \quad (4)$$

其中: TP 表示正样例被分类正确的个数, TN 表示负样例被分类正确的个数, FP 表示正样例被分类错误的个数, FN 表示负样例被分类错误的个数。 ER 越小, 表示当前特征子集的选择方案对应的错误率越小。将分类错误率结合每个粒子的选择的特征数作为衡量每个粒子适应度值的两个指标。

2.2 更新外部存档

外部存档的作用是保存搜索过程中发现的所有 Pareto 最优解。在每次迭代过程中, 都会产生非支配解, 这些新产生的非支配解将会与外部存档中的解进行比较: 如果外部存档是空的, 当前所有的非支配解都会加入到外部存档中; 如果外部存档中存在一个个体, 能够支配当前新产生的非支配解, 则将会丢弃这个新产生的非支配解, 否则, 将当前新产生的非支配解存储到外部存档中; 如果外部存档中存在一个个体, 由新产生的非支配解支配, 则将该个体从外部存档中删除; 如果外部存档中的个体数量超过最大允许的容量, 则调用自适应网格搜索, 根据粒子的密度信息选择外部存档中的个体。粒子所在网格中包含的粒子数越多, 则粒子的密度值越大; 反之越小。为了保证解的多样性, 粒子的密度值越小, 选择当前粒子的概率就越大; 反之概率越小。

2.3 更新 $Pbest$ 和 $Gbest$

个体最优位置 $Pbest$ 保存的是到目前为止粒子本身在搜索空间中找到的最优解。本文采用基于支配的策略更新每个粒子 $Pbest$ 的位置信息: 如果一个粒子在迭代过程中的解受历史保存的粒子最优解支配, 则保存 $Pbest$ 的位置信息不变; 否则, 将 $Pbest$ 的位置信息修改为当前的解。

全局最优位置 $Gbest$ 保存的是到目前为止种群在搜索空间中找到的最优位置。在单目标问题中, 种群只有一个 $Gbest$; 而对于多目标问题, 由于各个目标之间是相互矛盾的, $Gbest$ 有很多个, 但是每个粒子只能选择一个作为 $Gbest$, 然后对粒子的位置信息进行更新。为了解决这个问题, 本文应用自适应网格法, 根据粒子的密度信息从外部存档中随机选择一个作为 $Gbest$, 粒子的密度值越小, 选择该粒子作为 $Gbest$ 的概率就越大。

2.4 变异操作

本文采用的变异率计算公式如式(5)所示:

$$pm = \left(1 - \frac{it - 1}{MaxIt - 1}\right)^{1/mu} \quad (5)$$



其中: it 为当前的迭代次数, $MaxIt$ 为最大迭代次数, mu 为给定的 $[0,1]$ 区间的值。

在执行变异操作时,随机产生一个 $[0,1]$ 的数,如果小于变异率,则进行变异操作;否则,不进行变异操作。在粒子进行变异时,会根据新产生的粒子与当前粒子的支配关系进行选择:若新产生的粒子受当前粒子支配时,则粒子不变;若当前粒子受新产生的粒子支配时,则修改为新产生的粒子;若两个粒子互相非支配,则随机选择一个粒子。

2.5 MOBPSO 算法描述

```

输入 粒子群大小  $N$ ,最大迭代次数  $MaxIt$ ;
输出 非支配的外部存档解集  $Archive$ 。
1)  $t \leftarrow 0$ ,初始化粒子群  $S_0$  和外部存档  $Archive$ ;
2) for  $i \leftarrow 1$  to  $N$  do
3)   初始化  $S_0$  中粒子  $p_i$  的位置;
4) end
5) 评价粒子群  $S_0$ ,计算每个粒子的目标值,分类错误率和选择
    的特征个数;
6) 计算非支配解,并对外部存档进行更新;
7) while  $t < MaxIt$  do
8)   for  $i \leftarrow 1$  to  $N$  do
9)     更新粒子的历史最优位置  $Pbest$ ;
10)    更新全局最优位置  $Gbest$ ;
11)    根据式(1)更新粒子的位置;
12)   end
13)  对每个粒子执行变异操作;
14)  评价粒子群  $S_{t+1}$ ;
15)  计算非支配解,更新外部存档  $Archive$ ;
16)   $t \leftarrow t + 1$ ;
17) end
18) 输出外部存档  $Archive$  中的非支配解。

```

3 性能仿真和分析

3.1 数据集以及实验参数设置

本文采用的数据集均来自于 UCI 数据集^[19]以及基因表达数据集^[20],包括 zoo、wine、musk1、Ionosphere、lungcancer、parkinsons、dermatology、Alizadeh、wdbc、Hillvalley、Prostate 以及 9_Tumors 数据集,这些数据集类别数包括两类或多类,特征数从 13 到 10 509,样本数从 32 到 1212。这 12 个数据集具有一定的代表性,不仅能反映算法在特征数较少或较多情况下去除冗余特征的能力,还能够检测算法在样本数不足以及样本数非常大的情况下的优化性能,具体信息如表 1 所示。实验仿真在主频为 3.40 GHz,内存为 8.0 GB 的 PC 机上实现,软件环境为 Matlab R2016a。为了验证算法的性能,将本文算法与两种经典的多目标特征选择方法在上述数据集上进行对比,分别是 Coello 等^[21]在 2004 年提出来的多目标粒子群优化(Multi-Objective Particle Swarm Optimization, MOPSO)算法和 Deb 等^[22]在 2002 年提出的快速非支配排序遗传算法(Non-dominated Sort Genetic Algorithm II, NSGA II)。算法的实验参数设置如下,种群大小 N 设置为 20,外部存档的最大容量 $nrep$ 设置为 20,最大的迭代次数 $MaxIt$ 设置为 30。MOPSO 算法中,粒子的最大速度 v_{max} 为 1.0,惯性权重 w 为 0.5,加速因子 c_1 为 2, c_2 为 1。NSGA II 中,变异概率 $mrate$ 为 $1/n$, n 为种群个数,交叉概率 $crate$ 设置为 0.9。由于采用封装式的特征选择方法,在训练过程中需要一个学习算法来评

价分类性能,本文采用最简单的 KNN($K=3$)算法当作学习算法。

表 1 数据集描述

Tab. 1 Description of datasets

数据集	样本个数	特征个数	分类种类
Alizadeh	34	366	6
dermatology	100	1 212	2
Hillvalley	34	351	2
Ionosphere	56	32	3
lungcancer	166	476	2
musk1	22	195	2
parkinsons	30	569	2
wdbc	13	178	3
wine	16	101	7
zoo	34	366	6
Prostate	102	10 509	2
9_Tumors	60	5 726	9

本文实验中,设置的训练集为整个数据集的 70%,测试集为 30%。在训练过程中,根据粒子的位置,确定选择特征子集的个数,通过 KNN($K=3$)分类器并采用十折交叉验证,计算特征子集的错误率,将得到的特征子集的个数和错误率作为粒子的适应度函数值;在测试过程中,对选出的每个特征子集进行测试,并将测试结果作为算法的最终结果。

3.2 实验结果分析

实验中,分别将 MOBPSO 算法与 MOPSO 算法以及 NSGA II 在 12 个数据集上进行测试。MOBPSO 算法与 MOPSO 算法以及 NSGA II 三种多目标特征选择方法运行结果如图 1 所示。为了比较算法的计算时间,这三个算法分别运行了 10 次,计算了每一个算法的平均运行时间,比较结果如表 2 所示。

表 2 三种算法在 12 个数据集上运行时间比较

Tab. 2 Time comparison between MOBPSO and

compared algorithms on twelve datasets

数据集	MOBPSO	MOPSO	NSGA II
Alizadeh	33.88	34.14	41.98
dermatology	40.54	44.02	40.18
Hillvalley	43.81	44.99	40.55
Ionosphere	43.52	44.45	40.72
lungcancer	36.04	43.92	40.16
musk1	40.44	42.73	42.78
parkinsons	41.23	44.17	39.35
wdbc	34.36	35.08	39.45
wine	40.99	41.27	41.48
zoo	34.17	39.49	44.60
Prostate	86.97	92.16	89.19
9_Tumors	85.84	99.28	98.02

在 Ionosphere 数据集上,样本数为 351,本文算法与 NSGA II 都能够找到最小的特征个数,但是应用 MOBPSO 算法得到的分类错误率比 NSGA II 得到的分类错误率小。在得到的所有解中,应用 MOBPSO 算法特征数为 10 时可以得到最小的错误率,比整个数据集特征的个数少 24 个,去除了较多的冗余特征,并且在特征数为 8、9、10、和 11 时,MOBPSO 算法得到的分类错误率比 MOPSO 算法和 NSGA II 低。在 Hillvalley 数据集上,数据集有足够的样本,为 1212,在该数据



集上,本文算法在特征数为 24 时,可以找到最小分类错误率;在 parkinsons 数据集上,本文算法不能找到使分类错误率比其他两种算法小的特征子集;在 wdbc 数据集上,MOBPSO 可以获得最小的分类错误率,需要的特征子集大小为 16,其他

两个比较的算法的分类错误率较高;在 Prostate 数据集上,特征数最多,为 10 509,MOBPSO 可以获得最小的分类错误率,需要的特征子集大小为 1 079,其他两个比较的算法的分类错误率较高。

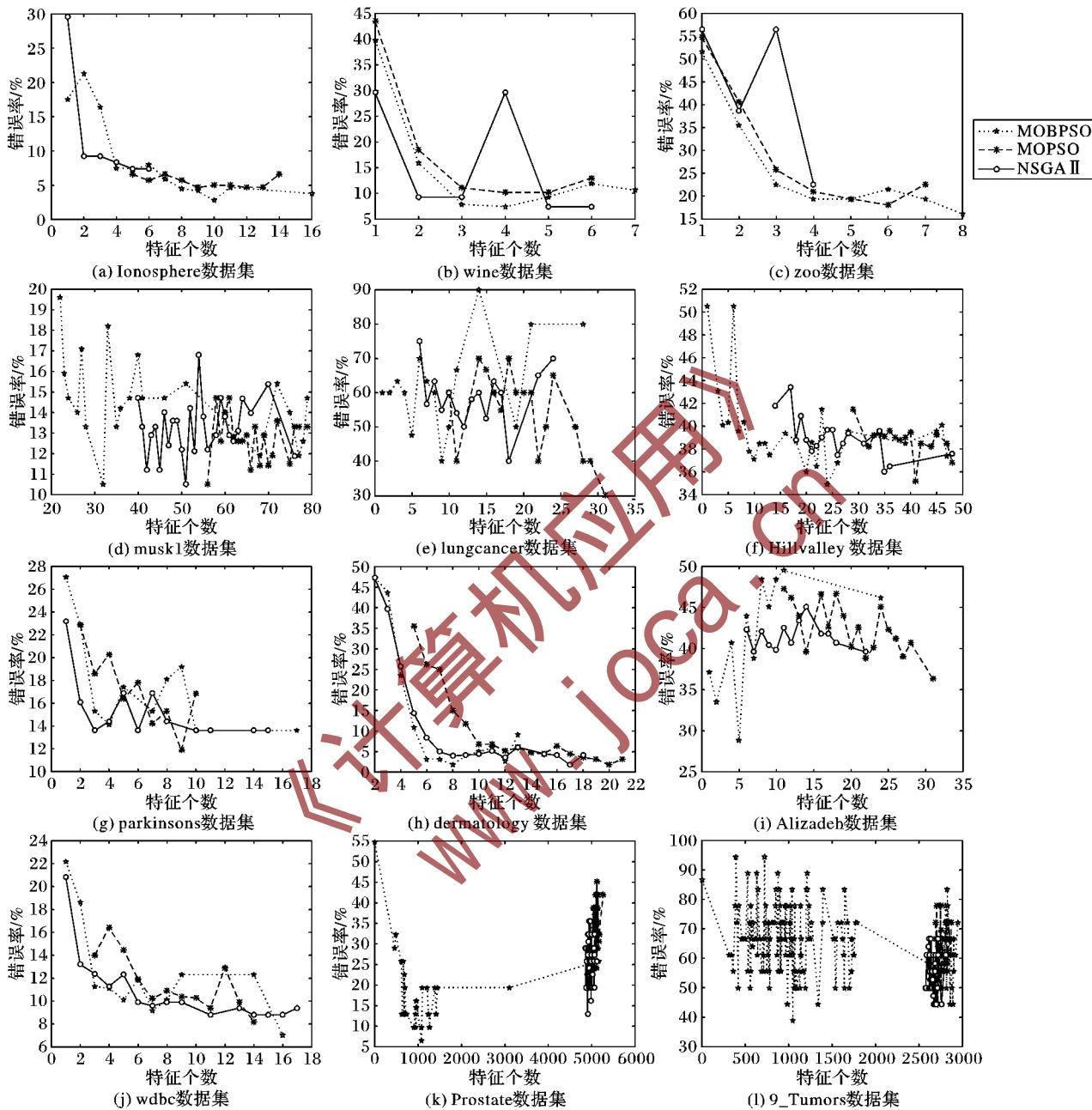


Fig. 1 MOBPSO 与对比算法在 12 个数据集上的对比结果

Fig. 1 Results comparison between MOBPSO and compared algorithms on twelve datasets

在 wine 数据集上,该数据集的特征数较少,仅为 13 个,虽然 NSGA II 在特征数为 5 时,得到的最小分类错误率与本文算法的最小分类错误率相等,但是本文算法选择的特征数为 4,且选择特征数的分类错误率均低于 MOPSO 算法,但是,特征数为 5 以及更多时,分类错误率比 NSGA II 的分类错误率高;在 musk1 数据集上,该数据集特征数为 166,特征数较多,本文算法能够找到最小的特征子集,但是分类错误率较高,三个算法均能够找到使错误率最小的特征子集,应用本文算法仅需要 32 个特征,而另两个算法分别需要 51 和 56 个,说明本文算法去除冗余特征能力比另外两种算法强,但是,本文算

法在其他特征子集上的分类错误率较高。因此,本文算法还需要进一步改进。

在多分类数据集 zoo 上,本文算法在特征数为 8 时,达到最小错误率,且在特征数为 1、2、3 和 4 时,得到的分类错误率均小于其他两种算法;在 lungcancer 数据集上,本文算法不能找到使分类错误率比其他两种算法小的特征子集;在 dermatology 数据集上,本文算法在特征数为 8 时,可以得到最小分类错误率,与其他两种算法相比,在特征数为 17 和 20 时得到相同的最小错误率时,本文算法使用的特征数更少;在 9_Tumors 数据集上,该数据集由 9 类,特征数为 5 726,MOBPSO



可以获得最小的分类错误率,需要的特征子集大小为 1046,其他两个比较的算法的分类错误率较高;在 Alizadeh 数据集上,MOBPSO 算法在 7 个特征子集上,获得了最小的分类错误率,相比其他两个算法,该算法有明显的优势。

对上述 12 个数据集分析可知,本文算法在部分数据集上去除冗余特征的能力比其他两种算法强,但是,还存在一些数据集,本文算法不能够达到很好的分类效果,还需要进一步的改进。表 3 对应三个算法在 12 个测试集上的最小错误率,其中加粗的代表在每个数据集上三种算法得到的最小错误率。由表 3 可知,本文算法在 7 个数据集上可以找到分类最小错误率,在其余数据集上虽然得到的不是三个算法中的最小错误率,但是与另外一种算法得到的最小错误率相等。从运行时间上来分析,MOBPSO 在 8 个数据集上,计算时间最少,占比 67%。因此,本文算法在去除冗余特征,得到最小特征子集,提高分类效果和降低计算时间上,具有显著优势。

表 3 三种算法最小错误率比较

Tab. 3 Comparison of minimum error rate of three algorithms %

数据集	MOBPSO	MOPSO	NSGA II
Alizadeh	28.8	36.2	39.6
dermatology	1.8	1.8	1.8
Hillvalley	34.9	35.2	36.0
Ionosphere	2.8	4.7	7.4
lungcancer	40.0	30.0	40.0
musk1	10.5	10.5	10.5
parkinsons	13.6	11.9	13.6
wdbc	7.0	8.2	8.8
wine	7.4	10.2	7.4
zoo	16.1	18.1	22.6
Prostate	6.5	25.8	12.9
9_Tumors	38.9	44.4	44.4

4 结语

针对单目标解决特征选择问题中存在的缺陷,本文把特征选择问题作为一个多目标优化问题,即选择分类错误率和特征子集大小作为两个目标。针对粒子群算法中调参问题,多目标骨架粒子群算法是一个少参数的算法,并且该算法还没有应用到特征选择中,针对这些问题,本文提出基于多目标骨架粒子群的特征选择算法。在 12 个 UCI 数据集上得到的结果显示,本文算法在大多数数据集上可以去除大部分冗余特征,提高算法分类效果和降低计算成本,但是,仍然存在一些数据集,不能够取得最小分类错误率,仍需要进一步作改进。

参考文献 (References)

- [1] KUNCHEVA L I. Combining Pattern Classifiers: Methods and Algorithms [M]. New York: Wiley-Interscience, 2004: 290–325.
- [2] DUDA R O, HART P E, STORK D G. Pattern Classification [M]. 2nd ed. New York: Wiley, 2001: 55–88.
- [3] SU C T, LIN H C. Applying electromagnetism-like mechanism for feature selection [J]. Information Sciences, 2011, 181(5): 972–986.
- [4] WANG L, ZHOU N, CHU F. A general wrapper approach to selection of class-dependent features [J]. IEEE Transactions on Neural Networks, 2008, 19(7): 1267–1278.
- [5] DATTA A, GHOSH S, GHOSH A. Self-adaptive differential evolution for feature selection in hyperspectral image data [J]. Applied Soft Computing, 2013, 13(4): 1969–1977.
- [6] XUE B, ZHANG M, BROWNE W N, et al. A survey on evolutionary computation approaches to feature selection [J]. IEEE Transactions on Evolutionary Computation, 2016, 20(4): 606–626.
- [7] ZHANG Z, YANG P. An ensemble of classifiers with genetic algorithm based feature selection [J]. IEEE Intelligent Informatics Bulletin, 2008, 9(1): 18–24.
- [8] ABD-ALSABOUR N, RANDALL M. Feature selection for classification using an ant colony system [C]// Proceedings of the 2010 6th IEEE International Conference on E-Science Workshops. Washington, DC: IEEE Computer Society, 2014: 86–91.
- [9] SHUNMUGAPRIYA P, KANMANI S, DEVIKPRIYA S, et al. Investigation on the effects of ACO parameters for feature selection and classification [C]// CNC 2012: International Conference on Advances in Communication, Network, and Computing. Berlin: Springer, 2012: 136–145.
- [10] CHUANG L Y, TSAI S W, YANG C H. Catfish binary particle swarm optimization for feature selection [EB/OL]. [2018-03-20]. <https://pdfs.semanticscholar.org/222d/8b2803f9cedf0da0b454c061c0bb46384722.pdf>.
- [11] 齐秀琴, 李炜. 人工蜂群算法优化的特征选择方法 [J/OL]. (2018-01-19) [2018-04-30]. 计算机科学与探索, <http://kns.cnki.net/kcms/detail/11.5602.TP.20180206.1345.012.html>. (CHAO X Q, LI W. A feature selection method optimized by artificial bee colony algorithm [J/OL]. (2018-01-19) [2018-04-30]. Journal of Frontiers of Computer Science and Technology, <http://kns.cnki.net/kcms/detail/11.5602.TP.20180206.1345.012.html>.)
- [12] AL-ANI A, ALSUKKER A, KHUSHABA R N. Feature subset selection using differential evolution and a wheel based search strategy [J]. Swarm & Evolutionary Computation, 2013, 9: 15–26.
- [13] EMARY E, ZAWBAA H M, HASSANIEN A E. Binary grey wolf optimization approaches for feature selection [J]. Neurocomputing, 2016, 172(C): 371–381.
- [14] MAFARJA M M, MIRJALILI S. Hybrid whale optimization algorithm with simulated annealing for feature selection [J]. Neurocomputing, 2017, 260: 302–312.
- [15] SHUNMUGAPRIYA P, KANMANI S. A hybrid algorithm using ant and bee colony optimization for feature selection and classification (AC-ABC Hybrid) [J]. Swarm and Evolutionary Computation, 2017, 36: 27–36.
- [16] 张梦林, 李占山. 基于 SAC 的特征选择算法 [J]. 计算机科学, 2018, 45(2): 63–68. (ZHANG M L, LI Z S. Feature selection algorithm using SAC algorithm [J]. Computer Science, 2018, 45(2): 63–68.)
- [17] XUE B, ZHANG M, BROWNE W N. Particle swarm optimization for feature selection in classification: a multi-objective approach [J]. IEEE Transactions on Cybernetics, 2013, 43(6): 1656–1671.
- [18] KENNEDY J. Bare bones particle swarms [C]// Proceedings of the 2003 IEEE Swarm Intelligence Symposium. Piscataway, NJ: IEEE, 2003: 80–87.

(下转第 3166 页)



- ceedings of the 2016 European Conference on Computer Vision. Berlin: Springer, 2016: 791–808.
- [12] GARCIA J, MARTINEL N, MICHELONI C, et al. Person re-identification ranking optimisation by discriminant context information analysis [C]// Proceedings of the 2015 IEEE International Conference on Computer Vision. Piscataway, NJ: IEEE, 2015: 1305–1313.
- [13] YE M, CHEN J, LENG Q, et al. Coupled-view based ranking optimization for person re-identification [C]// Proceedings of the 2015 International Conference on Multimedia Modeling. Berlin: Springer, 2015: 105–117.
- [14] YE M, LIANG C, YU Y, et al. Person reidentification via ranking aggregation of similarity pulling and dissimilarity pushing [J]. *IEEE Transactions on Multimedia*, 2016, 18(12): 2553–2566.
- [15] XIAO T, LI H, OUYANG W, et al. Learning deep feature representations with domain guided dropout for person re-identification [C]// Proceedings of the 2016 IEEE Computer Vision and Pattern Recognition. Piscataway, NJ: IEEE, 2016: 1249–1258.
- [16] CHUM O, PHILBIN J, SIVIC J, et al. Total recall: automatic query expansion with a generative feature model for object retrieval [C]// Proceedings of the 2007 IEEE International Conference on Computer Vision. Piscataway, NJ: IEEE, 2007: 1–8.
- [17] HE K, ZHANG X, REN S, et al. Deep residual learning for image recognition [C]// Proceedings of the 2016 IEEE Conference on Computer Vision and Pattern Recognition. Piscataway, NJ: IEEE, 2016: 770–778.
- [18] ZHENG L, BIE Z, SUN Y, et al. Mars: a video benchmark for large-scale person re-identification [C]// Proceedings of the 2016 European Conference on Computer Vision. Berlin: Springer, 2016: 868–884.
- [19] SRIVASTAVA N, HINTON G, KRIZHEVSKY A, et al. Dropout: a simple way to prevent neural networks from overfitting [J]. *Journal of Machine Learning Research*, 2014, 15(1): 1929–1958.
- [20] FELZENZWALB P F, GIRSHICK R B, MCALLESTER D, et al. Object detection with discriminatively trained part-based models [J]. *IEEE Transactions on Pattern Analysis & Machine Intelligence*, 2014, 47(2): 6–7.
- [21] SU C, ZHANG S, XING J, et al. Deep attributes driven multi-camera person re-identification [C]// Proceedings of the 2016 European Conference on Computer Vision. Berlin: Springer, 2016: 475–491.
- [22] USTINOVA E, GANIN Y, LEMPITSKY V. Multi-region bilinear convolutional neural networks for person re-identification [C]// Proceedings of the 2017 Advanced Video and Signal Based Surveillance. Washington, DC: IEEE Computer Society, 2017: 1–6.
- [23] ZHANG L, XIANG T, GONG S. Learning a discriminative null space for person re-identification [C]// Proceedings of the 2016 IEEE Conference on Computer Vision and Pattern Recognition. Piscataway, NJ: IEEE, 2016: 1239–1248.
- [24] ZHENG L, HUANG Y, LU H, et al. Pose invariant embedding for deep person re-identification [J/OL]. arXiv Preprint, 2017, 2017: arXiv: 1701. 07732 [2017-01-26]. <http://en.arxiv.org/abs/1701.07732>.
- [25] BARBOSA I B, CRISTANI M, CAPUTO B, et al. Looking beyond appearances: synthetic training data for deep CNNs in re-identification [J]. *Computer Vision and Image Understanding*, 2017, 167: 50–62.
- [26] ZHENG L, ZHANG H, SUN S, et al. Person re-identification in the wild [C]// Proceedings of the 2017 IEEE Conference on Computer Vision and Pattern Recognition. Washington, DC: IEEE Computer Society, 2017: 3346–3355.
- [27] LIAO S, HU Y, ZHU X, et al. Person re-identification by local maximal occurrence representation and metric learning [C]// Proceedings of the 2015 IEEE Conference on Computer Vision and Pattern Recognition. Piscataway, NJ: IEEE, 2015: 2197–2206.
- [28] VARIOR R R, SHUAI B, LU J, et al. A siamese long short-term memory architecture for human re-identification [C]// Proceedings of the 2016 European Conference on Computer Vision. Berlin: Springer, 2016: 135–153.

This work is partially supported by the National Natural Science Foundation of China (61572085).

CHEN Shoubing, born in 1991, M. S. candidate. His research interests include computer vision.

WANG Hongyuan, born in 1960, Ph. D., professor. His research interests include computer vision.

JIN Cui, born in 1994, M. S. candidate. Her research interests include computer vision.

ZHANG Wei, born in 1980, M. S., lecturer. Her research interests include machine vision.

This work is partially supported by the National Natural Science Foundation of China (61402481), the Top-notch Talent Support Program for Youth of Hebei Province ([2013]17), the Key Project of Education Department of Hebei Province (ZD2018083).

ZHANG Cuijun, born in 1968, M. S., professor. Her research interests include intelligent computation, machine learning.

CHEN Beibei, born in 1993, M. S. candidate. Her research interests include intelligent computation, machine learning.

ZHOU Chong, born in 1989, Ph. D., lecturer. His research interests include intelligent computation, multi-objective optimization.

YIN Xinge, born in 1999. Her research interests include information management, information system.

(上接第3160页)

- [19] HETTICH S, BLAKE C, MERZ C. UCI repository of machine learning databases [DB/OL]. [2017-07-08]. <http://www.ics.uci.edu/mlearn/MLRepository.html>.
- [20] STATNIKOV A, ALIFERIS C F, TSAMARDINOS I. Gene expression datasets [DB/OL]. [2017-08-10]. <http://www.gems-system.org>.
- [21] COELLO C A C, PULIDO G T, LECHUGA M S. Handling multiple objectives with particle swarm optimization [J]. *IEEE Transactions on Evolutionary Computation*, 2004, 8(3): 256–279.
- [22] DEB K, PRATAP A, AGARWAL S, et al. A fast and elitist multiobjective genetic algorithm: NSGA-II [J]. *IEEE Transactions on Evolutionary Computation*, 2002, 6(2): 182–197.