计算机科学
COMPUTER SCIENCE

混合改进的花授粉算法与灰狼算法用于特征选择

康雁，王海宁，陶柳，杨海潇，杨学昆，王飞，李浩

引用本文

康雁，王海宁，陶柳，杨海潇，杨学昆，王飞，李浩. 混合改进的花授粉算法与灰狼算法用于特征选择[J]. 计算机科学，2022, 49(6A): 125-132.

KANG Yan, WANG Hai-ning, TAO Liu, YANG Hai-xiao, YANG Xue-kun, WANG Fei, LI Hao. Hybrid Improved Flower Pollination Algorithm and Gray Wolf Algorithm for Feature Selection[J]. Computer Science, 2022, 49(6A): 125-132.

计算机科学 COMPUTER SCIENCE
http://www.jsjkx.com
DOI:10.11896/jsjkx.210600135

混合改进的花授粉算法与灰狼算法用于特征选择

康 雁 王海宁 陶 柳 杨海潇 杨学昆 王 飞 李 浩
云南大学软件学院 昆明 650500
(kangyan@ynu.edu.cn)

摘 要 特征选择在数据预处理阶段中极为重要。特征选择的优劣不仅影响着神经网络训练的时间长短,更影响神经网络性能的好坏。灰狼改进花授粉算法(Grey Wolf Improved Flower Pollination Algorithm,GIFPA)是一种基于花授粉算法(Flower Pollination Algorithm,FPA)框架与灰狼优化算法融合的混合算法,将其应用于特征选择问题,既可以保留原始特征的内涵信息,又可以最大化分类特征的准确率。GIFPA 算法在花授粉算法的异花授粉阶段中加入了最差个体信息,并用作全局搜索,将灰狼优化算法中的狩猎过程作为局部搜索,并且通过转换系数来调节二者的搜索过程。同时,为了克服群智能算法易陷入局部最优的问题,首次采用数据挖掘领域中的 ReliefF 算法,通过 ReliefF 算法过滤出高权重特征并用于改进最佳个体信息。为了验证算法的性能,实验选取 UCI 数据库中 21 个领域的经典数据集进行测试,利用 K 近邻(KNN)分类器进行分类测评,以适应度值和准确率作为评价标准,并通过 K-折交叉验证来克服过拟合问题。实验选择了包括 FPA 算法在内的多种经典算法和先进算法进行比较,结果表明 GIFPA 算法在特征选择问题上有很强的竞争力。

关键词:特征选择;FPA 算法;灰狼算法;ReliefF;优化器
中图法分类号 TP391

Hybrid Improved Flower Pollination Algorithm and Gray Wolf Algorithm for Feature Selection

KANG Yan,WANG Hai-ning,TAO Liu,YANG Hai-xiao,YANG Xue-kun,WANG Fei and LI Hao
School of Software,Yunnan University,Kunming 650500,China

Abstract Feature selection is very important in the stage of data preprocessing. The quality of feature selection not only affects the training time of the neural network but also affects the performance of the neural network. Grey Wolf improved Flower pollination algorithm(Grey Wolf improved Flower pollination algorithm,GIFPA) is a hybrid algorithm based on the fusion of flower pollination algorithm framework and gray wolf optimization algorithm. When it is applied to feature selection,it can not only retain the connotation information of the original features but also maximize the accuracy of classification features. The GIFPA algorithm adds the worst individual information to the FPA algorithm,uses the cross-pollination stage of the FPA algorithm as the global search,uses the hunting process of the gray wolf optimization algorithm as the local search,and adjusts the search process of the two through the conversion coefficient. At the same time,to overcome the problem that swarms intelligence algorithm is easy to fall into local optimization,this paper uses the ReliefF algorithm in the field of data mining to improve this problem and uses the ReliefF algorithm to filter out high weight features and improve the best individual information. To verify the performance of the algorithm,21 classical data sets in the UCI database are selected for testing,*k*-nearest neighbor(KNN) classifier is used for classification and evaluation,fitness value and accuracy are used as evaluation criteria,and K-fold crossover verification is used to overcome the over-fitting problem. In the experiment,a variety of classical algorithms and advanced algorithms,including the FPA algorithm,are compared. The experimental results show that the GIFPA algorithm has strong competitiveness in feature selection.

Keywords Feature selection,FPA,GWO,ReliefF,Optimizer

1 引言

分类在机器学习分类器的应用中是一个非常重要的领域,然而分类器的性能主要受放入特征的影响,当放入分类器的特征含有较多噪音时分类结果往往会受到有影响。因此,特征选择的作用是在特征放入分类器前选择出信息量较大的特征,并舍弃不相关或者有噪声的特征。目前特征选择的方法众多,如使用主成分分析法来降低特征维度,或使用模糊推理来降低特征维度,从而降低空间复杂度,减少分类器实际训练时间。特征选择方法主要分为两个步骤:搜索子集及质量

基金项目:云南省科技厅重大专项(2019ZE001-1,202002AB080001-6)
This work was supported by the Yunnan Provincial Science and Technology Department Major Special Projects(2019ZE001-1,202002AB080001-6).
通信作者:李浩(493895015@qq.com)

评价[1]。搜索子集是指使用一定的搜索策略在数据高维特征中选择出特征子集,而质量评价则是将上一个步骤选择出来的特征放入分类器中评价该搜索策略的质量。

群智能优化算法是元启发式算法的一种,由于其性能优良,近些年受到了广泛的关注,越来越多的研究人员利用元启发式算法解决各领域问题,并且取得了很好的效果。这些算法能够利用群体的有用信息来寻找最优解[2],如将改进的蜉蝣优化算法用于轴承故障诊断[3],将改进的蝙蝠算法用于Android的恶意软件侦测[4],或利用帝王蝶优化算法解决01背包问题[5],Salehi将果蝇优化算法与广义回归神经网络结合用于榛叶细胞紫杉醇合成预测[6]等。

此外,许多研究者将元启发式算法中的群智能算法用于特征选择问题,如动态樽海鞘算法[7]、二进制蝴蝶优化[8]、蜉蝣算法[9]、量子蚱蜢优化[10]和鲸鱼优化算法[11]。但是由于群智能算法都具有一定的局限性,如原始Jaya算法由于位置更新方法单一,不能很好地平衡全局搜索和局部搜索的过程,因此易陷入局部最优解。目前有多种对群智能算法改进的措施,如将完全随机的种群初始化变为带一定限制的种群初始化以增加种群初始多样化,或优化位置更新策略等。如Faris等利用混沌理论优化樽海鞘算法的位置更新策略,并将其应用于特征选择问题[12]。同时越来越多的研究者选择结合不同算法的特点形成新的混合算法,如将二元化学反应优化与禁忌搜索结合用于高维生物医学数据的特征选择[13],利用禁忌搜索算法来减少遗传算法过早收敛的问题并用于特征选择[14]。因此,混合不同特点的群智能算法已经被证明在特征选择领域相对于单一的群智能算法具备更好的性能。

选用FPA算法和GWO算法是因为它们之间有可以互补的优点。FPA算法具有对迭代初始猜测不敏感的特点,与其他算法相比在全局搜索能力上有较强的优势,无需设置过多的参数,并且易于实现[15]。在FPA算法中,将生物(交叉传粉)和非生物(自花传粉)传粉过程分别建模为全局搜索和局部搜索,其中任意一种传粉过程的发生都采用开关概率 p 来控制[16]。然而,FPA算法的全局搜索策略只考虑了最佳个体的位置信息,这会导致FPA算法在稳定性上有所欠佳,从而陷入局部最优解。而GWO算法具有较强的空间搜索能力,它能够利用灰狼种群制度的特点更快地向目标点靠拢,但GWO算法在后期搜索能力有所减弱。为了解决这些问题,本文结合了原始FPA和GWO的优点并做出两点改进,提出了GIFPA算法。新提出的GIFPA算法可以改善易陷入局部最优解的问题,并且在特征选择问题上有很好的效果。

在本研究中,主要贡献总结如下:

(1)提出了一种利用灰狼来改进FPA的混合算法GIFPA,它基于FPA算法框架做了4个方面的改进。

1)首次将数据挖掘领域中的ReliefF算法与包装器方法结合,用于改进最佳个体,从而改善易陷入局部最优解的问题。

2)在种群初始化阶段将后一半的个体与前一半的个体位置信息取反,这一改进增加了种群个体多样性。

3)在FPA算法的异花授粉阶段加入了最差个体信息,这一改进提升了算法的稳定性。

4)使用转换系数在原始FPA异花授粉阶段和灰狼狩猎的过程中进行选择更新个体位置,这一改进可以提高算法解的多样性。

(2)将新提出的GIFPA算法的包装模式用于特征选择发展。

(3)将GIFPA算法在来自UCI数据库的21个基准数据集上进行测试和评估,并且与其他7种启发式算法(FPA,BA,MVO,BCSA,bWOA,NLPSO,TMGWO)进行了比较。从实验结果来看,这些改进有效地改进了FPA算法的性能,并且性能上优于其他启发式算法。

本文第2节介绍了一些相关的工作;第3节介绍了原始Jaya算法的简要背景及所提出的GIFPA算法的细节;第4节介绍了所进行实验的细节以及这些实验的结果比较;最后总结全文。

2 相关工作

2.1 特征选择

特征选择(Feature Selection,FS)也叫属性选择,是指在大规模数据中选出对结果有重大价值的特征。FS在机器学习中是一项非常重要的数据预处理工作,若不进行特征选择,当数据规模较大时会耗费大量的时间及CPU运行效能,若在特征选择中舍弃了对结果价值较高的特征则会影响结果的精度。因此,如何在大量的特征数据中挖掘出有价值的特征子集成了一项困难却又不得不解决的问题。特征选择技术可以应用于回归[17]和分类算法[18],本文主要研究利用群智能优化算法在分类问题上进行特征选择。

特征选择技术通常分为过滤、包装和嵌入方法[19]。过滤方法较为简单,直接在数据上进行过滤,而不考虑将其用于从中提取知识的机器学习技术,因此,它们通常被用作预处理步骤。它们根据从数据中计算的得分函数对所有特征进行排名,并过滤掉排名较低的特征。过滤方法虽然计算速度快且简单,但在学习方法的交互性上有所欠佳。

包装方法的目的是找到一个变量子集,给出最佳的预测性能值。包装方法中执行两个步骤。首先,选择特征的子集;其次,根据基于机器学习问题的评分函数来评估这种子集的质量。重复这个过程,直到满足某个停止标准。在这里,学习机器就像一个黑匣子,但它以某种方式指导最终的结果。我们提出的GIFPA方法用于特征选择就是一种包装方法。

嵌入式方法同时训练模型和执行特征选择,即同时执行学习部分和特征选择部分。数据经过使用的算法模型训练后会得到不同特征的权重系数,权重越高的特征越被优先选择。因为它与预测模型相互作用,所以其更为高效,同时它在时间复杂度上也会比包装器低。Asunción等使用非线性的SVM进行特征选择就是一种嵌入式方法[20]。

2.2 灰狼优化算法

灰狼优化算法是由澳大利亚格里菲斯大学学者Mirjalili等于2014年提出来的一种群智能优化算法,详细算法过程请参照文献[21]。GWO通过模仿灰狼种群严格的种群制度来实现捕获猎物的过程。灰狼种群中按社会层级分为4类狼,分别为 α,β,δ 和 ω 狼,α,β 和 δ 狼在灰狼算法中代表着适应度值最佳的3只狼,α 狼适应度值最佳,除了 α,β 和 δ 狼,其他都

为 ω 狼。每一头狼 $\vec{X_i}$ 在位置更新时依据式(1)－式(4)计算出它与 3 头适应度值最佳狼的距离，然后使用式(5)更新其位置。

$$\vec{A}=2a\cdot\vec{r_1}-a \tag{1}$$

$$\vec{B}=2\vec{r_2} \tag{2}$$

$$\begin{cases}\vec{D_\alpha}=|\vec{B}\cdot\vec{X_\alpha}-\vec{X_i}|\\ \vec{D_\beta}=|\vec{B}\cdot\vec{X_\beta}-\vec{X_i}|\\ \vec{D_\delta}=|\vec{B}\cdot\vec{X_\delta}-\vec{X_i}|\end{cases} \tag{3}$$

$$\begin{cases}\vec{X_1}=\vec{X_\alpha}-\vec{A}\cdot\vec{D_\alpha}\\ \vec{X_2}=\vec{X_\beta}-\vec{A}\cdot\vec{D_\beta}\\ \vec{X_3}=\vec{X_\delta}-\vec{A}\cdot\vec{D_\delta}\end{cases} \tag{4}$$

$$\vec{X_i}(t+1)=\frac{\vec{X_1}+\vec{X_2}+\vec{X_3}}{3} \tag{5}$$

其中，t 表示当前迭代次数；$\vec{X_i}$，$\vec{X_\alpha}$，$\vec{X_\beta}$ 和 $\vec{X_\delta}$ 分别代表当前第 i 头狼和 α，β 和 δ 狼的位置向量；$\vec{X_i}(t+1)$ 表示当前狼更新的新位置向量；$\vec{D_\alpha}$，$\vec{D_\beta}$，$\vec{D_\delta}$ 分别表示当前第 i 头狼与 α，β 和 δ 狼之间的距离向量；$\vec{r_1}$ 和 $\vec{r_2}$ 为两个位于[0,1]之间的随机向量；$\vec{A}$ 和 $\vec{B}$ 为两个系数向量。a 的计算公式如下：

$$a=2-2*l/Max\_iter \tag{6}$$

其中，l 表示当前的迭代次数，Max_iter 为最大迭代次数。

3　GIFPA 算法

本节介绍了 FPA 算法的简要背景以及它们的数学表示，并概述了将 FPA 算法与 GWO 算法二阶段融合的过程。

3.1　原始花授粉算法

花授粉算法由 Yang 于 2012 年提出，是一种受开花植物的花授粉过程启发的超启发式算法，在寻找全局最优解上效果明显。

FPA 算法对生物异花授粉则是将其建模为传粉者通过 Levy 飞行进行全局授粉，即全局搜索过程；而非生物自花授粉则建模为局部授粉，即局部搜索过程。FPA 算法的全局搜索公式如式(7)所示，局部搜索式如式(9)所示，所采用的 Levy 飞行的数学公式如式(8)所示。

如果转换概率 $p>r$ 条件成立，则进行全局搜索，按式(7)对当前个体进行更新。若转换概率 $p<r$ 条件成立，则进行局部搜索，按式(9)对当前个体进行更新。

$$X_i^{t+1}=X_i^t+L(g^*-X_i^t) \tag{7}$$

$$L\sim\frac{\lambda\Gamma(\lambda)\sin\left(\frac{\pi\lambda}{2}\right)}{\pi s^{1+\lambda}},s\gg s_0\gg 0 \tag{8}$$

$$X_i^{t+1}=X_i^t+\varepsilon(X_j^t-X_k^t) \tag{9}$$

其中，X_i^{t+1}，X_i^t 分别表示个体 i 第 $t+1$ 代和第 t 代的解，g^* 表示当前最优个体信息，L 为步长，$\lambda=1.5$，$\Gamma(\lambda)$ 为标准的伽马函数。原始 FPA 算法的伪代码如算法 1 所列。

算法 1　FPA 算法

1. Initialize wolf pack individuals X_i(i=1,2,…,n)
2. While(l<Max_iters)：
3.　Traverse the population to find the best fitness individuals X_α.
4. for(every individual X_i)：
5. Calculate thep value according to equation(10).
6. if p < rand()：
7. Calculate X_i^{t+1} according to equation (7), and get $fitness_i^{t+1}$ by X_i^{t+1}.
8. else：
9. Calculate X_i^{t+1} according to equation (9), and get $fitness_i^{t+1}$ by X_i^{t+1}.
10.　Update current position and fitness value.
11. end if
12. end for
13. l=l+1
14. end while

目前花授粉算法被用在众多领域并取得了很好的效果，如将花授粉算法用于蛋白质与蛋白质的对接[22]，但将其用在特征选择领域的研究不多，主要原因是花授粉算法效果不稳定导致效果不佳。

3.2　GIFPA 算法

本节概述了将 FPA 算法与 GWO 算法二阶段融合的过程，GIFPA 算法的整体伪代码如算法 2 所示。首先对种群进行随机初始化，每个个体的每个特征位置信息均在(0,1)之间，为使种群多样化，前 $N/2$ 个个体的位置信息和后 $N/2$ 个个体的位置信息相反。在随机初始化种群后遍历种群得到适应度最佳个体和最差个体，同时也需要找到适应度前三的 3 个个体分别作为灰狼种群中的 α 狼、β 狼和 δ 狼，其中 α 狼即为适应度值最佳的个体，因此总共需要关注并记录 4 种个体，即适应度值前三的和最差的个体。在种群迭代过程中，每次迭代初始化计数器(improvement_best)为 0，并且根据式(10)计算出转换系数 p，在对整个种群每个个体进行位置更新时首先生成一个位于(0,1)的随机数，若该随机数小于 p，则用加入最差个体信息的 FPA 异花授粉阶段(式(11))进行全局搜索，否则利用灰狼算法中式(3)－式(5)进行全局搜索，将每个个体经过位置更新产生的新个体放入基于 KNN 分类器的适应度函数进行评估，并按照 FPA 算法的框架更新当前种群，即若新个体优于当前个体的位置则更新当前个体的位置。同理判断是否更新最佳个体位置和 β，δ 狼位置以及最差个体的位置。若最佳个体的位置得到优化，则计数器变量 $improvement\_best$ 加 1。然后判断是否陷入局部最优，当 $improvement\_best\leqslant 2$ 时认为陷入局部最优，使用 ReliefF 算法找出高权重的特征并选择，从而优化最佳个体，进而优化整个种群。如此反复迭代直至迭代次数达到最大迭代次数时返回最佳个体的二进制向量作为最佳解，二进制向量中为 1 的特征则为筛选出来的特征。

$$p=e^{0.08\frac{Max\_iter}{l}-1} \tag{10}$$

其中，Max_iter 表示最大迭代次数，l 表示当前迭代次数，e 为欧拉数。

算法 2　GIFPA

1. Initialize wolf pack individuals X_i(i=1,2,…,n).
2. While(l<Max_iters)：
3. Traverse the population to find the top three and worst individuals with fitness X_α, X_β, X_δ and X_w.
4.　Calculate thep value according to equation(10).
5. improvement_best=0.

```
6. for(every individual X_i):
7. if rand()<p:
8.         Calculate X_i^{t+1} according to equation(11).
9. else:
10.        Calculate X_i^{t+1} according to equation(3)(4) and(5).
11. end if
12. Get fitness_i^{t+1} by X_i^{t+1}.
13.    if fitness^{t+1}i< fitness_α:
14. improvement_best+=1
15. Update current position and fitness value.
16. if improvement_best<=2:
17. call RelifF on α wolf to improve its fitness value.
18. end if
19. end for
20. l=l+1.
21. end while
```

3.2.1 种群初始化

种群初始化在群智能算法中至关重要，好的种群初始化策略不仅能够加快寻优的速度，还能找出更优的特征。本文采用的种群初始化策略即将后 $N/2$ 个个体的位置信息与前 $N/2$ 个个体的位置信息取反，其目的一方面是保证种群位置信息多样化，能够更快地发现全局最优解，二是避免陷入局部最优解中。

3.2.2 局部搜索与全局搜索

原始 FPA 算法在搜索最优解时利用式(7)进行全局搜索，尽管采用 Levy 飞行会提升全局探索能力，但同时也加大了搜索过程的随机性，使算法整体不够稳定。特征选择问题实际可以建模为二进制问题，故最差解的信息也可以用来反向找出高质量的特征。为了使算法稳定性更强，在异花授粉阶段加入最差解，如式(11)所示。

$$X_i^{t+1}=X_i^t+L(X_i^t-g^*)+L(X_i^t+d^*) \tag{11}$$

其中，除了 d^* 表示最差个体的位置信息以外，其他符号与式(7)中的符号含义一样。

而灰狼优化算法的狩猎阶段在局部搜索上有很强的优势，因此用灰狼算法的狩猎阶段来代替 FPA 算法的自花授粉阶段。GIFPA 算法为了改善这一问题，综合 FPA 算法和灰狼算法的特点，提高解决方案的多样性。通过式(10)计算转换系数 p 来很好地平衡全局搜索与局部搜索，显然转换系数 p 会随着迭代次数 l 的增加而减小，因此前几次迭代大概率按式(11)进行全局搜索，而后几次的迭代则大概率按灰狼算法中式(3)－(5)计算进行局部搜索。这主要是利用式(11)代替了灰狼算法中包围猎物的阶段，并且有效结合了 FPA 算法在全局搜索上有良好性能的特点，加快了包围猎物的过程，让 α 狼的位置得到很好的改善，即高效优化了整个种群中最佳个体的位置。但易陷入局部最优解的问题，仍然显著。

3.2.3 RelifF 算法

RelifF 是数据挖掘领域中由经典算法 Relif 演化而来的算法，运行效率很高。本文另一个改进是结合了过滤式算法来改进种群，以往的研究者往往是将过滤式方法与包装器方法分开，而本文选用 RelifF 算法来过滤出具有较大分类意义的特征，从而改善种群中的最优个体。引入 RelifF 算法主要是为了改善群智能算法易陷入局部最优的问题，利用变量(*improvement_best*)记录下每次迭代最佳个体进化的次数，若 *improvement_best*≤2 则认为陷入了局部最优，将进行 RelifF 算法尝试突破局部最优解。

群智能算法易陷入局部最优解的一个主要原因是最佳个体得不到改进。尽管利用灰狼优化算法的狩猎阶段可以在一定程度上改善最佳个体的位置，但灰狼算法本身也存在一定的局限性，因此使用 RelifF 算法改进 α 狼，即改进最佳个体，促进整个种群进化。首先在每次迭代开始添加一个新的计数器变量(*improvement_best*)，用于记录最佳个体进化的次数，当 *improvement_best*≤2 时则调用 RelifF 算法，RelifF 算法每次在训练集中随机选取一个样本 R，并分别选择出与样本 R 同类和不同类的近邻样本各 K 个，然后利用下列公式更新特征权重。

$$W(A)=W(A)-\sum_{j=1}^{k}\frac{diff(A,R,H_j)}{mk}+\sum_{C\notin class(R)}\left[\frac{p(C)}{1-p(Class(R))}\sum_{j=1}^{k}diff(A,R,M_j(C))\right]/(mk) \tag{12}$$

其中，$W(A)$表示特征 A 的权重，$diff(A,R_1,R_2)$表示样本R_1与样本R_2在特征 A 上的差，$M_j(C)$表示在类别 C 中第 j 个紧邻样本。$diff(A,R_1,R_2)$的计算公式如下：

$$diff(A,R_1,R_2)=\begin{cases}\dfrac{|R_1[A]-R_2[A]|}{\max(A)-\min(A)}, & \text{if } A \text{ is continuous}\\ 0, & \text{if } A \text{ is discrete and } R_1[A]=R_2[A]\\ 1, & \text{if } A \text{ is discrete and } R_1[A]\neq R_2[A]\end{cases} \tag{13}$$

通过 RelifF 算法可以快速找出对分类结果影响可能较大的特征，将选出的高权重特征与最佳个体进行对比，在最佳个体中按权重大小依次选择对应特征来优化位置信息。

4 实验结果与讨论

4.1 数据集与参数设置

本文的代码均采用 python 语言进行编写，为了验证 GIFPA 算法的性能，将数据维度小于 30 定义为小型数据集，大于等于 30 小于 70 定义为中型数据集，大于 70 为大型数据集。本文选用 UCI 数据库中 21 个不同领域的经典数据集进行特征选择实验，其中小型数据集 13 个，中型数据集 3 个，大型数据集 5 个。表 1 列出了 21 个数据集的基本信息，包括总的特征维度(dim)和总的样本数量。

在所进行的实验中，特征选择包装器模式采用 KNN 分类器，其中 $K=5$(折叠数)。此外，每个数据集都使用适应度值和分类准确率作为评价指标。将数据集划分为训练集和测试集，并使用交叉验证保证了实验结果的有效性。K 倍交叉验证是将数据集按比例随机划分成多部分，分别用于训练、测试和验证，并且会将特征随机打散，当算法初始化时则会随机选择一些特征产生初始解。为了使实验结果更具说服力，对每个数据集都运行了 20 次再取平均值作为每个数据集的结果，我们选取了适应度值、分类准确率和选择的特征数作为目标值。实验设置的参数如表 2 所列。

表 1　数据集介绍

Table 1　Introduction to datasets

No.	Dataset	dim	Number of samples
1	Australian	14	690
2	Breast cancer tissue	9	106
3	Climate	20	540
4	HeartEW	13	270
5	IonosphereEW	34	351
6	Lymphography	18	148
7	Page blocks	10	5473
8	Robot-failures-lp1	90	88
9	Robot-failures-lp2	90	47
10	Robot-failures-lp3	90	47
11	Robot-failures-lp4	90	117
12	Robot-failures-lp5	90	164
13	Segment	19	2310
14	SonarEW	60	208
15	SpectEW	22	267
16	Stock	9	950
17	Vehicle	18	846
18	Vowel	10	990
19	WineEW	13	178
20	WDBC	30	569
21	Zoo	16	101

表 2　参数设置

Table 2　Parameters setting

Parameter	Value
Population number	20
Max_iteration	20
dim	Number of features
K-neighbors	5
K-fold cross-validation	10
α	0.01
β	0.99

4.2　评价函数

如上所述，使用 KNN 分类器进行评价包装器特征选择，并以准确率和选择特征数量作为最后的评价指标。由于选择的特征数量并非越少越好，要保证重要的特征不被舍弃，因此适应度函数需要有效综合准确率和选择特征数量，其函数须满足两点要求：1）尽可能少的选择特征；2）尽可能高的分类准确率。

为了平衡以上两点要求，设计适应度函数如下：

$$fitness=\alpha*\frac{s}{dim}+\beta*error(X_i) \tag{14}$$

其中，$\alpha\in(0,1)$，$\beta=1-\alpha$，α 和 β 表示错误率和选择特征比的权重系数；S 表示选择特征的数量；而 dim 则为数据集的特征维度；$error(X_i)$表示当前个体放入 KNN 分类器中得到的错误率。

因此，$fitness$ 的值在很大程度上可以说明 GIFPA 算法在特征选择上的效能。通过将每个数据集连续运行 20 次取平均的 $fitness$ 值作为 GIFPA 算法在该数据集上特征选择的性能指标，并且与其他元启发式方法进行比较可以很好地证明 GIFPA 算法的竞争力。同时，将 GIFPA 算法在每个数据集上连续迭代 20 次，效果如图 1 所示。可以看到，GIFPA 算法的初始 $fitness$ 值都较低，说明具有很强的收敛性；在前 5 次迭代中能够迅速找出一个较优的解，并且在后 15 次迭代中 $fitness$ 变化次数较多，说明在全局搜索上 GIFPA 算法具有很强的全局探索能力。可见，将 FPA 算法与灰狼算法融合可以很好地结合二者的特点。

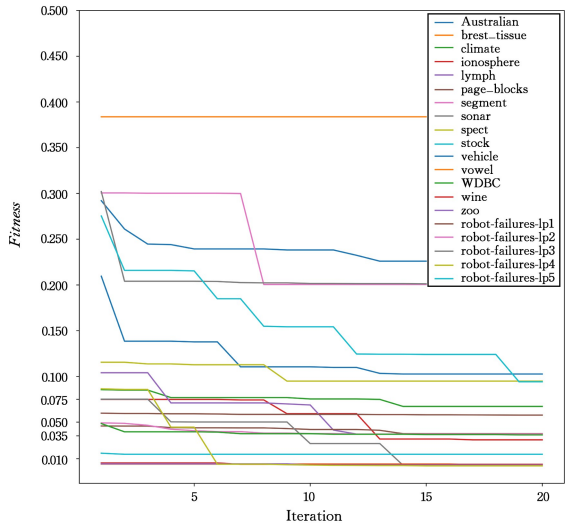

图 1　fitness 迭代图

Fig. 1　Fitness iteration diagram

4.3　结果比较与分析

表 3 中列出了 GIFPA 算法对于每个数据集选择的特征数量（$\sharp s$）及其占总特征维度比例的情况，从表中可以看出 GIFPA 算法在绝大部分数据集上的降维程度超过了 50%，具有很好的降维效果。同时，由于选择的特征数并不是越少越好，因此还需要考虑分类准确率，我们主要针对分类准确率与适应度值进行了比较。将 GIFPA 算法与一些传统的群智能优化算法和新提出的算法在以上 21 个数据集上进行了实验比较，以每个数据集的平均准确率和平均适应度值作为评价指标。

选取的对比算法如下。

（1）FPA[23]：花授粉算法是一种受开花植物自花授粉和异花授粉过启发的新算法。

（2）BA[24]：二进制蝙蝠算法是模仿蝙蝠的回声定位行为进行全局优化的启发式算法。

（3）MVO[25]：多向优化器算法是一种非受自然启发的元启发式算法，将 3 个自然天体虫洞、白洞和黑洞进行数学建模后分别用于局部搜索、勘探和全局搜索。。

（4）BCSA[26]：乌鸦搜索算法（CSA）是近些年提出的一种模拟乌鸦觅食行为的生物启发式算法，BCSA 是将 CSA 算法用于特征选择的二进制形式算法。

（5）bWOA[27]：二进制鲸鱼优化算法是一种将鲸鱼优化算法离散化后的新的元启发式算法，在特征选择中具有较好的性能。

（6）NLPSO[28]：非线性粒子群算法是一种基于时变惯性权重策略的二进制粒子群优化算法，并用于特征选择问题中。

（7）TMGWO[29]：二阶段灰狼算法是一种改进的灰狼优化算法，它利用二阶段变异改善了灰狼优化算法的性能，并将其用于特征选择问题中。

表 3 连续运行 20 次平均选择的特征数量

Table 3 Average number of features selected for 20 consecutive runs

Datasets	#s	Proportion/%	Datasets	#s	Proportion/%	Datasets	#s	Proportion/%
Australian	4.00	28.57	Breast tissue	4.50	50.00	Climate	4.90	24.50
HeartEW	4.80	36.92	IonosphereEW	8.10	23.82	Lymphography	6.75	37.50
Page blocks	4.10	41.00	Robot-failures-lp1	19.65	21.83	Robot-failures-lp2	19.10	21.22
Robot-failures-lp3	15.65	17.39	Robot-failures-lp4	21.45	23.83	Robot-failures-lp5	33.35	37.06
Segment	6.35	33.42	SonarEW	21.60	36.00	SpectEW	7.75	35.23
Stock	4.30	47.78	Vehicle	9.45	52.50	Vowel	3.90	39.00
WineEW	4.25	32.69	WDBC	5.10	17.00	Zoo	5.20	32.50

在大多数数据集中，GIFPA 算法优于其他算法。表 4 的数值描述了 8 种算法经过 20 次训练后的平均分类精度。实验数据表明，本文提出的 GIFPA 算法具有较高的准确率，弥补了以往算法的不足，为特征选择提供了一种很好的方法。

表 4 连续运行 20 次与其他算法的准确率比较

Table 4 Comparison of the accuracy of 20 consecutive runs with other algorithms

No.	Dataset	FPA	BA	MVO	BCSA	bWOA	NLPSO	TMGWO	GIFPA
1	Australian	0.8318	0.8312	0.8318	0.8304	0.8391	0.8304	0.8463	**0.8736**
2	Breast cancer tissue	0.3300	0.3300	0.3300	0.3300	0.3200	0.3300	0.3300	**0.5905**
3	Climate	0.9277	0.9184	0.9277	**0.9416**	0.9222	**0.9416**	0.9314	0.9347
4	HeartEW	0.8407	0.8259	0.8296	0.8259	0.8148	0.8259	0.8407	**0.8861**
5	IonosphereEW	0.9057	0.8765	0.9057	0.9085	0.8942	0.9085	0.9314	**0.9536**
6	Lymphography	0.8571	0.8775	0.8857	0.8785	0.8714	0.8785	0.9000	**0.9467**
7	Page blocks	0.9612	0.9163	0.9610	0.9559	0.9643	0.9570	0.9639	**0.9692**
8	Robot-failures-lp1	0.8750	0.8924	0.9125	0.8875	0.8750	0.8875	0.9125	**0.9194**
9	Robot-failures-lp2	0.7500	0.7500	**0.7750**	**0.7750**	0.7250	**0.7750**	**0.7750**	0.7650
10	Robot-failures-lp3	0.7000	0.7250	0.7500	0.7500	0.7250	0.7500	**0.7750**	0.7300
11	Robot-failures-lp4	0.8909	0.8750	0.9272	0.8818	0.8818	0.8818	0.9363	**0.9667**
12	Robot-failures-lp5	0.6437	0.6374	0.6875	0.6250	0.6312	0.6250	0.6750	**0.8106**
13	Segment	0.9709	0.9714	**0.9744**	0.9679	0.9679	0.9679	**0.9744**	0.9694
14	SonarEW	0.6600	0.6550	0.7300	0.6799	0.6600	0.6799	0.7400	**0.9631**
15	SpectEW	0.7436	0.7423	0.7576	0.7346	0.7230	0.7346	0.7615	**0.9056**
16	Stock	0.9326	0.9326	0.9326	0.9326	0.9326	0.9326	0.9326	**0.9784**
17	Vehicle	0.7273	0.7261	0.7345	0.7321	0.7309	0.7321	0.7380	**0.7642**
18	Vowel	0.9888	0.9888	0.9888	0.9534	0.9672	0.9756	0.9888	**1.0000**
19	WineEW	0.9411	0.9411	0.9470	0.9325	0.9411	0.9352	0.9470	**0.9889**
20	WDBC	0.9482	0.9482	0.9482	0.9464	0.9482	0.9464	0.9482	**0.9605**
21	Zoo	0.9600	0.9600	0.9600	0.9600	0.9600	0.9600	0.9600	**0.9750**

为了验证提出的式(11)能增强 FPA 全局探索的稳定性，我们对比了 FPA 和 GIFPA 在各个数据集上连续平均运行 20 次的方差情况，结果如图 2—图 4 所示。可以发现，GIFPA 在 16 个数据集上的标准差明显低于 FPA，其他数据集标准差相差不多。

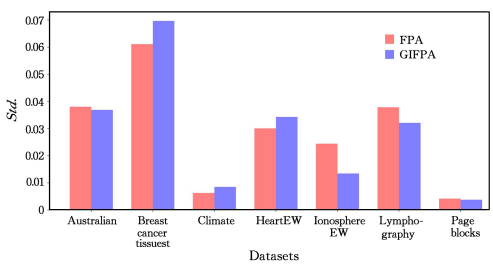

图 2 FPA 与 GIFPA 标准差比较(a)

Fig. 2 Comparison of standard deviation between FPA and GIFPA(a)

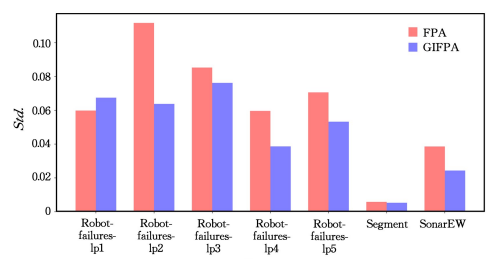

图 3 FPA 与 GIFPA 标准差比较(b)

Fig. 3 Comparison of standard deviation between FPA and GIFPA(b)

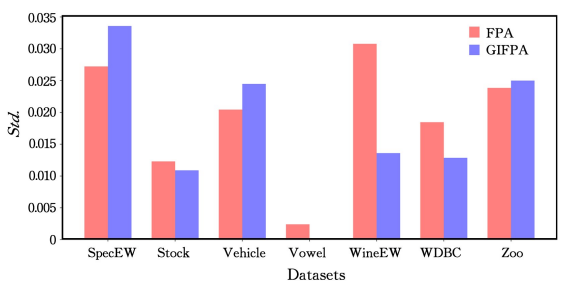

图 4 FPA 与 GIFPA 标准差比较(c)

Fig. 4 Comparison of standard deviation between FPA and GIFPA(c)

可以看出，GIFPA 算法在分类准确率上表现很好，并且效果较为稳定，特别是在 SpecteEW 和 SonarEW 数据集上，GIFPA 算法的分类效果明显优于其他算法，在 Vowel 数据集上虽然各类算法的分类准确率都较高，但是本文算法能保持多次连续分类准确率达到 1，可见 GIFPA 算法的稳定性较强。与原始 FPA 算法相比，GIFPA 算法在绝大部分数据集上分类准确率都明显更高，说明我们提出的改进策略能够很好地改善 FPA 算法。GIFPA 在 Climate 数据集上的分类准确率略低于 BCSA 算法，但比其他算法都要高。在 Robot-failures-lp2 和 Robot-failures-lp3 数据集上，GIFPA 算法的分类准确率低于 TMGWO 算法，而原始 FPA 算法在这两个数据集上效果也不理想，说明基于 FPA 算法在这两个数据集上

效果并不好，但分类准确率较原始 FPA 算法却有了明显改进，说明提出的改进策略能够很好地改进 FPA 算法。GIFPA 算法在 Segment 数据集上的分类准确率仅比最高的 TMGWO 算法差 0.005。总的来看，GIFPA 算法尽管在个别数据集上的分类准确率略低于一些算法，但平均分类准确率明显高于其他算法，因此 GIFPA 算法与其他优化算法相比拥有一定的竞争力。

表 5 给出了 *improvement_best* 分别为 1，2，3 和 4 时的平均分类准确率。可以看到，当 *improvement_best* 值为 2 时平均分类准确率最优，这是由于这一变量较小对最优个体的优化不足导致分类准确率不高，较大时加大了随机性，从而导致分类准确率下降。表 6 为 FPA 算法与 GIFPA 算法超 20 次时总的平均运行时间及平均方差比较。可以看到，加入了 RelifF 算法后在平均运行时间上只增加了 1 s，但却有效地提高了降维能力和分类准确率。

表 5　*improvement_best* 分别为 1，2，3 和 4 时的平均分类准确率

Table 5　Average accuracy of the algorithm when *improvement_best* is 1，2，3，and 4 respectively

improvement_best	*Average accuracy*
1	0.8865
2	**0.8977**
3	0.8879
4	0.8857

表 6　FPA 算法与 GIFPA 算法的平均运行时间及平均方差

Table 6　Average time and standard deviationof FPA algorithm and GIFP Aalgorithm

Algorithm	*Average time*/s	*Std*
FPA	2.9118	0.0658
GIFPA	3.2195	0.0986

根据适应度定义函数可知，适应度值不仅取决于分类准确率，还要受选择特征数量的影响，适应度值越低代表效果越好。对每个数据集独立运行 20 次取平均值，以查看算法的稳定性。表 7 为在 21 个数据集上 GIFPA 算法与其他 7 种算法连续运行 20 次所得到的平均适应度值，图 5 为 GIFPA 算法与其他 7 种算法在所有数据集上平均适应度值的比较。从图 5 可以看出，GIFPA 的 *fitness* 值要明显低于其他算法，可见 GIFPA 算法在特征选择领域具备强力的竞争性。可以从表 7 中看出，GIFPA 算法在大部分数据集上的适应度值要明显优于其他算法，而原始的 FPA 算法本身性能并不突出，可见提出的改进策略对 FPA 算法在性能上有很大改进。GIFPA 算法原本在 Climate 和 Robot-failures-lp2 数据集上的分类准确率略低于 BCSA 算法，但 GIFPA 算法能够选择更少的特征，因此在适应度值上要优于 BCSA 算法。同分类准确率实验一样，GIFPA 算法在 Robot-failures-lp3 数据集上的适应度值差于 TMGWO 算法，这一方面是因为分类准确率较低，二是因为选择的特征数较多。在 Segment 数据集上，GIFPA 的效果略差于 TMGWO 算法，不过相差不大。同样地，GIFPA 的总适应度值平均值明显优于其他算法，这足以证明 GIFPA 算法不仅能够选择出重要的特征信息，还能保证分类准确率。

表 7　连续运行 20 次与其他算法的 fitness 值比较

Table 7　Comparison of 20 consecutive runs with the fitness values of other algorithms

No.	Dataset	FPA	BA	MVO	BCSA	bWOA	NLPSO	TMGWO	GIFPA
1	Australian	0.1819	0.1781	0.1759	0.2222	0.1882	0.1813	0.1686	**0.1280**
2	Breast cancer tissue	0.6844	0.6955	0.6788	0.7313	0.7007	0.7014	0.6817	**0.4104**
3	Climate	0.0827	0.0849	0.0844	0.0774	0.0845	0.0861	0.0804	**0.0671**
4	HeartEW	0.2047	0.1988	0.1955	0.2318	0.2089	0.2009	0.1760	**0.1164**
5	IonosphereEW	0.1146	0.1257	0.1118	0.1202	0.1190	0.1029	0.0906	**0.0483**
6	Lymphography	0.1577	0.1587	0.1441	0.1796	0.1531	0.1502	0.1400	**0.0417**
7	Page blocks	0.0472	0.0458	0.0455	0.0509	0.0485	0.0485	0.0455	**0.0355**
8	Robot-failures-lp1	0.1412	0.1464	0.1138	0.2014	0.1487	0.1448	0.0909	**0.0819**
9	Robot-failures-lp2	0.2759	0.2739	0.2489	0.2903	0.2835	0.2671	0.2415	**0.2348**
10	Robot-failures-lp3	0.3058	0.3026	0.2716	0.3255	0.3061	0.2902	**0.2493**	0.2690
11	Robot-failures-lp4	0.1405	0.1464	0.0912	0.1563	0.1400	0.1296	0.0826	**0.0354**
12	Robot-failures-lp5	0.3849	0.3860	0.3460	0.4043	0.3866	0.3858	0.3430	**0.1912**
13	Segment	0.0413	0.0381	0.0311	0.0464	0.0419	0.0396	**0.0284**	0.0337
14	SonarEW	0.4055	0.3957	0.3347	0.3914	0.3876	0.3616	0.3337	**0.0401**
15	SpectEW	0.2876	0.2858	0.2727	0.3053	0.2880	0.2826	0.2644	**0.0970**
16	Stock	0.0786	0.0776	0.0723	0.0919	0.0775	0.0747	0.0730	**0.0261**
17	Vehicle	0.2868	0.2885	0.2808	0.3074	0.2833	0.2830	0.2731	**0.2387**
18	Vowel	0.0109	0.0109	0.0109	0.0437	0.0313	0.0109	0.0109	**0.0032**
19	WineEW	0.0742	0.0824	0.0706	0.1211	0.0800	0.0788	0.0638	**0.0143**
20	WDBC	0.0604	0.0602	0.0559	0.0700	0.0598	0.0605	0.0535	**0.0408**
21	Zoo	0.0518	0.0541	0.0488	0.0570	0.0517	0.0515	0.0485	**0.0280**

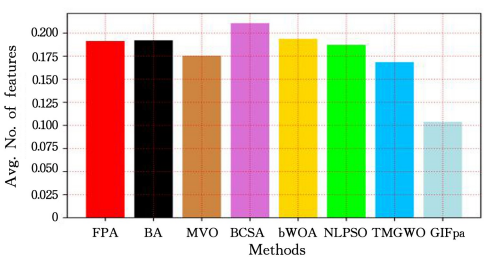

图 5　平均 fitness 值比较

Fig. 5　Average fitness value comparison

结束语　本文提出了一种新的将 FPA 优化算法框架与灰狼优化算法融合的新型混合算法。GIFPA 算法不仅利用了 FPA 算法在全局搜索问题上简单高效的特点，也利用了灰狼算法局部搜索能力强的优势，并且引入了转换系数 p 来平衡全局搜索与局部搜索的关系。为了改善易陷入局部最优问题，首次将数据挖掘领域的 RelifF 算法融于包装器方法中，通过记录下 α 狼进化的次数来判断是否陷入了局部最优，若陷入局部最优则尝试用 RelifF 算法来突破局部最优解。在 21 个不同领域的经典数据集上进行实验，并且与 7 种先进算法进行比较，使用 10 倍交叉验证来减少过拟合问题。实验结果说明了 GIFPA 算法无论是在分类准确率还是适应度值上都是优于其他算法，可见其在特征选择问题上具有很好的性能。

参 考 文 献

[1] ZAWBAA H M, EMARY E, GROSAN C. Feature Selection via Chaotic Antlion Optimization[J]. Plos One, 2016, 11(3): e0150652.

[2] JIN X M, HUA W Q. Resource Management for Mobile Cloud Computing Energy Consumption Optimization[J]. Computer Science, 2020, 47(6): 253-257.

[3] LIU Y, CHAI Y, LIU B, et al. Bearing Fault Diagnosis Based on Energy Spectrum Statistics and Modified Mayfly Optimization Algorithm[J]. Sensors, 2021, 21(6): 2245.

[4] RAVI K, MALLIDI S, SANTOSH J K, et al. Bat optimization algorithm for wrapper-based feature selection and performance improvement of android malware detection[J]. IET Networks, 2021: 1-10.

[5] FENG Y, WANG G G, DEB S, et al. Solving 0-1 knapsack problem by a novel binary monarch butterfly optimization[J]. Neural Computing and Applications, 2017, 28(7): 1-16.

[6] SALEHI M, FARHADI S, MOIENI A, et al. A hybrid model based on general regression neural network and fruit fly optimization algorithm for forecasting and optimizing paclitaxel biosynthesis in Corylus avellana cell culture[J]. Plant Methods, 2021, 17(1): 13.

[7] TUBISHAT M, JA'AFAR S, ALSWAITTI M, et al. Dynamic Salp Swarm Algorithm for Feature Selection[J]. Expert Systems with Applications, 2020, 147: 113873.

[8] ARORA S, ANAND P. Binary butterfly optimization approaches for feature selection[J]. Expert Systems with Application, 2019, 116(FEB.): 147-160.

[9] BHATTACHARYYA T, CHATTERJEE B, SINGH P K, et al. Mayfly in Harmony: A New Hybrid Meta-Heuristic Feature Selection Algorithm[J]. IEEE Access, 2020, 8: 195929-195945.

[10] WANG D, CHEN H, LI T, et al. A novel quantum grasshopper optimization algorithm for feature selection[J]. International Journal of Approximate Reasoning, 2020, 127: 33-53.

[11] CHEN H W, HU Z, HAN L, et al. A Spark-based Distributed Whale Optimization Algorithm for Feature Selection[C]// The 10th IEEE International Conference on Intelligent Data Acquisition and Advanced Computing Systems: Technology and Applications. IEEE, 2019: 70-74.

[12] FARIS H, MAFARJA M M, HEIDARI A A, et al. An Efficient Binary Salp Swarm Algorithm with Crossover Scheme for Feature Selection Problems[J]. Knowledge-Based Systems, 2018, 154(Aug. 15): 43-67.

[13] YAN C, MA J, LUO H, et al. A hybrid algorithm based on binary chemical reaction optimization and tabu search for feature selection of high-dimensional biomedical data[J]. Tsinghua Science and Technology, 2018, 23(6): 733-743.

[14] SHI L, WAN Y C, GAO X J, et al. Feature Selection for Object-Based Classification of High-Resolution Remote Sensing Images Based on the Combination of a Genetic Algorithm and Tabu Search[J]. Computational Intelligence and Neuroscience, 2018, 2018.

[15] WANG M, LIN J, YUE L, et al. Compensation for mobile carrier magnetic interference in a SQUID-based full-tensor magnetic gradiometer using the flower pollination algorithm[J]. Measurement Science and Technology, 2021, 32(8): 085010.

[16] POA B, SC A, CYT A, et al. Prediction of tea theanine content using near-infrared spectroscopy and flower pollination algorithm-ScienceDirect[J]. Spectrochimica Acta Part A: Molecular and Biomolecular Spectroscopy, 2021, 255.

[17] ANDERSEN C M, BRO R. Practical aspects of PARAFAC modeling of fluorescence excitation-emission data[J]. Journal of Chemometrics, 2010, 17(4): 200-215.

[18] JUNG D. Distributed Feature Selection for Multi-Class Classification Using ADMM[J]. IEEE Control Systems Letters, 2020, 5(3): 821-826.

[19] CHANDRASHEKAR G, SAHIN F. A survey on feature selection methods[J]. Computers & Electrical Engineering, 2014, 40(1): 16-28.

[20] JIMÉNEZ-CORDERO A, MORALES J M, PINEDA S. A novel embedded min-max approach for feature selection in nonlinear Support Vector Machine classification[J]. European Journal of Operational Research, 2021, 293(1): 24-35.

[21] SM A, SMM B, AL A. Grey Wolf Optimizer[J]. Advances in Engineering Software, 2014, 69: 46-61.

[22] SUNNY S, JAYARAJ P B. FPDock: Protein-Protein Docking Using Flower Pollination Algorithm[J]. Computational Biology and Chemistry, 2021, 93(2): 107518.

[23] RAO R V. Jaya: A simple and new optimization algorithm for solving constrained and unconstrained optimization problems [J]. International Journal of Industrial Engineering Computations, 2016, 7(1934): 19-34.

[24] YANG X S. Flower Pollination Algorithm for Global Optimization[C]// International Conference on Unconventional Computing and Natural Computation. Berlin: Springer, 2012: 240-249.

[25] MIRJALILI S, MIRJALILI S M, YANG X S. Binary bat algorithm[J]. Neural Computing & Applications, 2014, 25(3/4): 663-681.

[26] MIRJALILI S, MIRJALILI S M, HATAMLOU A. Multi-Verse Optimizer: a nature-inspired algorithm for global optimization [J]. Neural Computing and Applications, 2015, 27(2): 495-513.

[27] SOUZA R, COELHO L, MACEDO C, et al. A V-Shaped Binary Crow Search Algorithm for Feature Selection[C]// 2018 IEEE Congress on Evolutionary Computation(CEC). IEEE, 2018: 1-8.

[28] HUSSIEN A G, HASSANIEN A E, HOUSSEIN E H, et al. S-shaped Binary Whale Optimization Algorithm for Feature Selection[M]// Recent Trends in Signal and Image Processing. Singapore: Springer, 2019: 79-87.

[29] MAFARJA M, JARRAR R, AHMAD S, et al. Feature Selection Using Binary Particle Swarm Optimization with Time Varying Inertia Weight Strategies[C]// International Conference on Future Networks & Distributed Systems. 2018: 1-9.

[30] ABDEL-BASSET M, EL-SHAHAT D, EL-HENAWY I, et al. A new fusion of grey wolf optimizer algorithm with a two-phase mutation for feature selection[J]. Expert Systems with Application, 2020, 139(Jan.): 112824. 1-112824. 14.

KANG Yan, born in 1972, postgraduate supervisor, is a member of China Computer Federation. Her main research interests include machine learning and software engineering.

LI Hao, born in 1970, postgraduate supervisor, is a member of China Computer Federation. His main research interests include machine learning and software engineering.