

2 FSFOA 算法

森林优化算法(forest optimization algorithm,简称 FOA)^[27]是 Ghaemi 于 2014 年提出的一种仿生类进化算法,用于解决单目标非线性连续搜索空间问题.FSFOA 算法试图将 FOA 算法用在离散搜索空间的问题上,即特征选择,并取得了不错的效果.

在 FSFOA 算法中,每棵树代表问题的一个可能解,即一个特征子集.树中的每个“1”表示相应的特征被选择参与机器学习过程,每个“0”表示在学习过程中相应特征被排除.FSFOA 算法包含 5 个部分:初始化森林、局部播种、形成候选森林、全局播种(global seeding)、更新最优树(update the best tree).该算法以初始化森林为起始点,经过局部播种、形成候选区,全局播种这 3 个主要步骤产生新树,更新森林,经过 FSFOA 算法的多次迭代,从而得到最优特征子集.具体流程如图 1 所示.

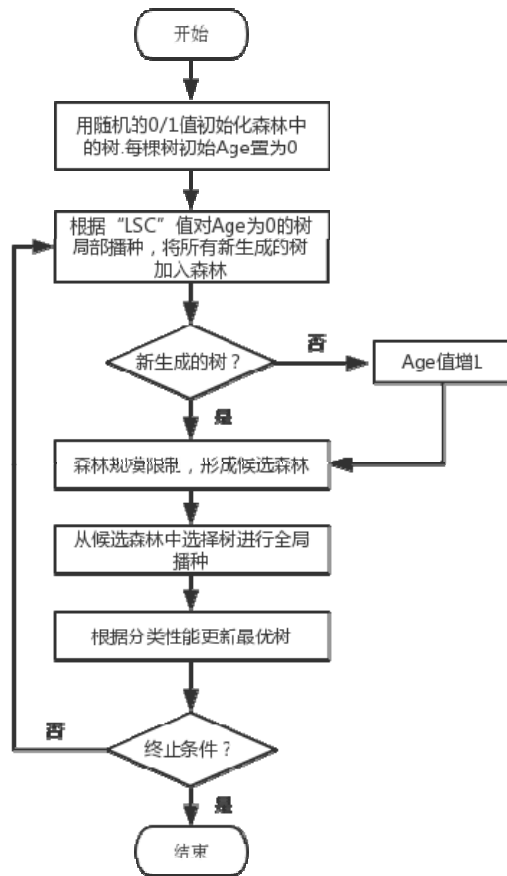


Fig.1 Flowchart of FSFOA

图 1 FSFOA 流程图

初始化阶段,随机产生一定数量的树形成森林,树中除了有 0/1 表示的相应变量值外,还有一个表示“年龄”的属性 Age,将所有刚刚生成的树的 Age 设置为 0 后,进入算法的局部播种阶段.如图 1 所示:在初始化阶段,FSFOA 算法只是采用随机的方式(即,在 n 维特征中随机选取若干个特征)来完成初始化,具有一定的盲目性,而算法在局部播种和全局播种阶段的寻优又依赖于初始点的选择,因此,FSFOA 算法在初始点选择不理想时很容易错失最优特征子集.

在局部播种阶段,将 0-Age 树的临近树加入森林.即对 Age 为 0 的树随机选择一些变量(LSC 参数决定选择

变量的个数),然后将这些被选择变量的值从 0 改为 1,反之亦然.经过局部播种后,新产生的树的 Age 置 0,其余旧树的 Age 加 1,将新树添加到森林中.局部播种的具体过程如图 2 所示.FSFOA 算法模拟局部播种的过程,将其用于算法的局部搜索.在图 1 中的局部播种阶段,FSFOA 算法并未考虑新树的优劣情况,而是将所有的新树添加到了森林中,这样做会使森林中存在较多的劣质树,增大搜索难度.

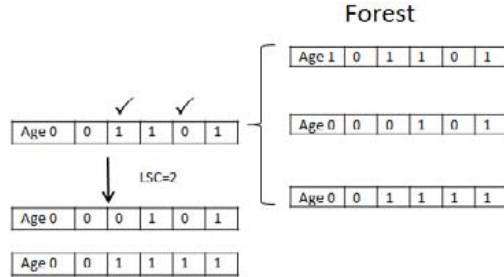


Fig.2 An example of local seeding operation on one tree with LSC=2

图 2 LSC=2,树局部播种示例

由于局部播种阶段的操作,森林中树的数量不断增加.为限制森林规模,将以下两种情况的树从森林中移除,放入候选森林中(即图 1 的形成候选森林阶段):首先,将森林中 Age 值大于年龄上限值 *life time* 的树从森林中移除,放入候选森林;如果原森林中剩余树的数量仍超出区域上限值 *area limit*,则根据树的适应度值(分类准确率)从小到大依次移除多余的树,并将这些移除的树放到候选森林中.

在全局播种阶段,根据预定义参数 *transfer rate* 值从候选区中随机选出若干棵树(即图 1 的全局播种阶段),对每棵树随机选一些变量(*GSC* 参数决定选择变量的个数),然后将这些被选择变量的值从 0 改为 1,反之亦然.但这一阶段是同时添加或删除一些特征,而不是一次只改变一个特征的取舍.如图 3 所示.FSFOA 算法模拟全局播种过程为算法提供全局搜索方法,一定程度上克服因局部播种而可能陷入局部最优解的缺点.

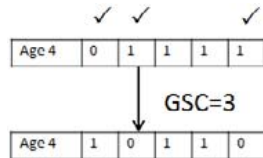


Fig.3 An example of global seeding operation on one tree with GSC=3

图 3 GSC=3,树全局播种示例

最后在更新阶段(即图 1 的更新最优树阶段),FSFOA 算法只选取适应度值最大的树作为最优树,并将其 Age 置 0,重新放入森林中.更新阶段迭代多次,直至满足停止条件为止.很明显,该算法在更新策略上具有传统更新机制的潜在局限性,即在更新过程中只考虑了分类准确率并未考虑维度缩减的问题.

3 IFSFOA 算法

本节就当前 FSFOA 算法性能的不足之处提出 3 点改进,并通过实验证实改进后的 IFSFOA 算法具有更好的性能.

3.1 初始化策略

新的初始化策略的灵感是源于两种传统特征选择方法:前向选择^[28]和后向选择^[29].前向选择的特征子集从空集开始,通常选择一个特征数量较小的特征子集,因此很容易错失特征数量较多的最优特征子集;而后向选择的特征子集从全集开始,通常选择一个特征数量较大的特征子集,这样做不仅容易错失特征数量小的特征子集,而且计算上的时间成本也较前向选择高很多.

因此,我们采用一种新的初始化策略,利用前向选择和后向选择的优点,摒弃其缺点,形成双向选择的策略.在这个新的策略中,树初始化使用少量的特征,从而确保算法从搜索小的特征子集的解空间开始,这样做不仅可以加大维度缩减的力度,还能够降低计算成本.但是,如果所有的树初始化都是用少量的特征,FSFOA 算法在寻优过程中很可能错失具有更好分类性能、特征数量较多的特征子集.为此,在新的初始化策略中,大部分树初始化使用少量的特征(模拟前向选择)而其余树的初始化使用较多的特征(模拟后向选择).经过新策略初始化后的森林通过与局部播种及全局播种的交互,选取性能最优的特征子集.

3.2 更新机制

在特征选择中,传统的更新机制具有潜在的局限性.如在 FSFOA 算法中,若新树的分类性能与旧树相同,但特征数目更小,即新树对应更好的特征子集,根据传统的更新机制,新树将不被更新,因为它们具有相同的分类性能.

为了克服这个限制,我们采用新的更新机制.在新的更新机制中,分类性能仍然是最主要的,但我们也将维度缩减问题纳入考虑范围.具体更新策略如下:首先,如果新树的分类性能比旧树效果好,忽略维度问题,更新新树,取代旧树;其次,如果新树的分类性能和原树效果一样,比较维度,若新树维度更小,更新新树,取代旧树,否则不予更新.

3.3 极度贪婪策略

如引言所述,为避免 FSFOA 算法在局部播种阶段将过多的劣质树添加到森林中,增加搜索难度,影响分类性能,我们让 IFSFOA 算法在局部播种阶段采用极度贪婪策略^[5],即:对于每棵新树,只将比旧树更优秀的新树添加到森林中,舍弃劣质树,这样做可以在保留下来的新树的基础上产生更优秀的新树.但为避免因极度贪婪策略导致算法陷入局部最优问题,在 IFSFOA 算法中,我们将全局播种阶段作用的对象由侯选森林改为由侯选森林和森林中所有 Age 为 0 的树共同确定,但改变前后用在全局播种阶段的树的数目并未改变.这样就使得同一棵 0-Age 树既可以局部播种,又可以全局播种,一定程度上解决了因极度贪婪策略带来的易陷入局部最优解问题.

极度贪婪策略具体如下:先复制一棵 Age 为 0 的树作为临时树,用和 FSFOA 一样的方法对临时树的某一变量做改变:若新生成的树比临时树更优,那么将新树添加到森林中,并将新树作为新的临时树;否则淘汰新树,临时树不变.重复以上操作,直至生成 LSC 棵新树为止.算法 1 给出了 IFSFOA 算法的伪代码,并用粗体标出了改进的地方.

算法 1. IFSFOA (*life time, LSC, GSC, transfer rate, area limit*).

Input: *life time, LSC, GSC, transfer rate, area limit*;

Output: The best feature set with the highest fitness.

```

1*:   Procedure IFSFOA
2*:   Initialize forest with simulating FS, BS methods
3*:   Forest←simForward(iniFo,fn,vn)
4*:   Forest←simBackward(iniFo,bn,vn)
5*:   Each tree in forest is a  $(D+1)$ -dimensional vector
      x ( $D$  is the number of all features)
6*:   The "Age" of each tree is initially zero
      While stop condition is not satisfied do
1:   Perform local seeding on trees with age 0
2:   tt←copy(t)
3:   For  $i=1$ :"LSC" do
4:     t'←reverseLocal(tt,Attri[i])
5:     if compOpt(t',tt) is true

```

```

6:      Forest←t'
7:      tt←t'
8:      else Forest←tt, tt is still used as a temporary tree.
9:  End for
10: increase the Age of all trees by 1
11: population limiting
12: Global seeding
13: Choose “transfer rate” percent of the candidate population
14: For each selected tree do
15: Choose “GSC” variables of the selected tree randomly
    change from 0 to 1 vice versa
16: End for
17: Update the best so far tree
18:  Sort trees according to their fitness value and
    dimensionality reduction
19:  optima←m[0]
20:  For i=1:len(m)
21:    if classifiers(m[i],flag)>classifiers(optima,flag)
22:      optima←m[i]
23:    else if classifiers(m[i],flag)=classifiers(optima,flag)
    and |m[i]|<|optima|
24:      optima←m[i]
25:    else No update
26:  End For
27: Set the Age of the best tree to 0
End while
7*: End procedure

```

Return the best tree which shows the best selected feature subset

在初始化阶段, $simForward(iniFo,fn,vn)/simBackward(iniFo,bn,vn)$ 是结构体数组,存储模拟前向选择/后向选择时的特征子集.其中 fn/bn 表示模拟前向选择/后向选择时树的个数,详情见第4.2节的实验参数设置. vn 表示树中‘1’的个数.值得注意的是: $simBackward$ 中的 vn 不是常数,是介于1/2特征数和特征总数之间的随机数.

在局部播种阶段, $copy(t)$ 函数用来复制Age为0的树形成临时树; $reverseLocal(tt,Attri[i])$ 对树实现局部播种,其中: tt 为临时树; $Attri$ 是一个数组,记录树中需要反转属性的索引值; $compOpt(t',tt)$ 比较两者的适应度值,为真说明新树优于临时树.

在更新阶段, m 是一个数组,记录每次循环中需要进行择优的树,其长度由参数 $area\ limit,transfer\ rate$ 决定; $optima$ 是临时变量,用来记录当前循环中的最优解(同时考虑准确率和维度缩减率). $classifiers(m[i],flag)$ 表示使用不同的分类器计算 $m[i]$ 的适应度值, $flag$ 标记具体使用的分类器; $|m[i]|$ 表示树中‘1’的个数.

在时间上,改进后的IFSFOA算法与FSFOA算法具有相同的时间复杂度,但在时间开销上会略高于FSFOA,高出的时间主要源于局部播种阶段的贪婪策略和更新机制中特征子集维度的比较;在空间上,与FSFOA相比,IFSFOA算法需要多开辟一段内存用于局部播种阶段临时树的存储,但这并未提高空间复杂度,仍在可接受范围内.

4 实验结果与分析

在实验中,我们使用公开的机器学习工具包 scikit-learn.编程语言为 python.所有实验均在 DELL 机上执行,该机器的配置是 Intel Core i5 CPU(3.20GHz),8G 内存.

4.1 数据集

我们在 11 个数据集上对 IFSFOA 算法的性能予以测试,涵盖了 6 个小维数据集(Glass,Wine,Heart,Cleveland,Vehicle,Segmentation)、2 个中维数据集(Dermatology,Ionosphere)和 3 个高维数据集(Sonar,SRBCT,Arcene).根据文献[30],在特征子集问题中,如果数据集中的特征数量是 $[0,19]$ 、 $[20,49]$ 、 $[50,\infty]$,那么对应数据集的规模分别是小维数据集、中维数据集、高维数据集.关于数据集的具体信息由表 2 给出.

Table 2 Specific information of the selected datasets

表 2 数据集详细信息

Dataset	#Feature	#Instance	class
Heart-statlog	13	270	2
Vehicle	18	846	4
Cleveland	13	303	5
Dermatology	34	366	6
Ionosphere	34	351	2
Sonar	60	208	2
Glass	9	214	7
Wine	13	178	3
Segmentation	19	2 310	7
SRBCT	2 308	63	4
Arcene	10 000	900	2

根据对比算法,我们对每个数据集的划分采用 70%作为训练集 30%作为测试集、10-折交叉验证、2-折交叉验证这 3 种方式.测试集的分类准确率 CA(classification accuracy)和维度缩减能力 DR(dimensionality reduction)作为 IFSFOA 算法及对比算法的评价准则.

- 分类准确率的具体定义如公式(1):

$$CA = N_{CC} / N_{AS} \quad (1)$$

其中, N_{CC} (number of correct classification)是正确分类的实例数, N_{AS} (number of all samples)是数据集实例总数;

- 维度缩减率的定义如公式(2):

$$DR = 1 - (N_{SF} / N_{AF}) \quad (2)$$

其中, N_{SF} (number of selected features)是被选择的特征数, N_{AF} (number of all features)是数据集的特征总数.

4.2 实验参数设置

为保证实验的公平性,IFSFOA 算法的参数设置与 FSFOA 算法完全相同.IFSFOA 对参数的设置如下.在参数中,“life time”,“area limit”,“transfer rate”的取值不依赖于数据集的大小,置为常数,“life time”=15,“area limit”=50,“transfer rate”=5%.由于 LSC 和 GSC 的取值依赖于每个数据集的特征数,因此这些参数值由表 3 分别给出.

根据文献[27]的实验,我们将 LSC 的值设为每个数据集特征维数的 1/5.初始化时,森林中 2/3 的树使用 10% 的特征数,其他 1/3 使用 K 个特征数(K 是介于 1/2 特征数和特征总数之间的随机数).

4.3 实验结果对比分析

我们将 IFSFOA 算法与其他经典特征选择算法进行比较.对比算法详情已由表 1 给出.表 4 总结了全部实验结果,并用粗体标出了不同分类器对应的表现最佳的分类准确率和维度缩减率.

Table 3 Value of *LSC* and *GSC* parameters

表 3 各数据集 *LSC, GSC* 参数值

Dataset	#Feature	<i>LSC</i>	<i>GSC</i>
Heart-statlog	13	3	6
Vehicle	18	4	9
Cleveland	13	3	6
Dermatology	34	7	15
Ionosphere	34	7	15
Sonar	60	12	30
Glass	9	2	4
Wine	13	3	6
Segmentation	19	4	9
SRBCT	2 308	460	700
Arcene	10 000	2 000	4 100

Table 4 Classification accuracy and dimension reduction of IFSFOA and compared methods

表 4 IFSFOA 及其对比算法的分类准确率和维度缩减率

Ionosphere	<i>Accuracy</i> (%)	<i>DR</i> (%)	Classifier	Wine	<i>Accuracy</i> (%)	<i>DR</i> (%)	Classifier
IFSFOA	98.86(10-fold)	87.65	5-NN	IFSFOA	98.07(70%~30%)	46.15	5-NN
FSFOA	89.43(10-fold)	54.28	5-NN	FSFOA	99.2(70%~30%)	30.76	5-NN
PSO(4-2)	87.27(70%~30%)	90.41	5-NN	PSO(4-2)	95.26(10-fold)	51.6	5-NN
IFSFOA	99.42(10-fold)	87.35	3-NN	IFSFOA	99.99(10-fold)	79.23	3-NN
FSFOA	92.3(10-fold)	61.76	3-NN	FSFOA	98.87(10-fold)	42.58	3-NN
NSM	92(10-fold)	88.23	3-NN	NSM	98(10-fold)	53.84	3-NN
IFSFOA	96.85(70%~30%)	79.11	1-NN	IFSFOA	96.15(70%~30%)	69.23	1-NN
FSFOA	89.52(70%~30%)	54.28	1-NN	FSFOA	98.07(70%~30%)	50	1-NN
SVM-FuzCoc	89.46(70%~30%)	88.23	1-NN	SVM-FuzCoc	97.12(70%~30%)	53.84	1-NN
SFS	87.75(50%~50%)	65.88	1-NN	SFS	97.69(70%~30%)	35.38	1-NN
SBS	84.61(50%~50%)	77.64	1-NN	SBS	94.77(70%~30%)	46.15	1-NN
SFFS	88.32(50%~50%)	75.29	1-NN	SFFS	96.56(70%~30%)	36.92	1-NN
IFSFOA	95.73(2-fold)	70.58	Rbf-svm	IFSFOA	98.31(2-fold)	57.69	Rbf-svm
FSFOA	94.58(2-fold)	57.14	Rbf-svm	FSFOA	96.06(2-fold)	37.17	Rbf-svm
HGAFS	92.76(2-fold)	82.35	Rbf-svm	HGAFS	98.31(2-fold)	53.85	Rbf-svm
IFSFOA	97.81(70%~30%)	60.29	J48	IFSFOA	96.15(70%~30%)	66.67	J48
FSFOA	95.12(70%~30%)	47.05	J48	FSFOA	96(70%~30%)	57.14	J48
UFSACO	88.61(70%~30%)	11.17	J48	UFSACO	95.08(70%~30%)	61.53	J48
IFSFOA	99.14(10-fold)	78.23	J48	IFSFOA	98.85(10-fold)	79.23	J48
FSFOA	93.16(10-fold)	68.57	J48	FSFOA	96.06(10-fold)	21.42	J48
FS-NEIR	92.59(10-fold)	82.35	J48	FS-NEIR	95.04(10-fold)	61.53	J48
Cleveland	<i>Accuracy</i> (%)	<i>DR</i> (%)	Classifier	Sonar	<i>Accuracy</i> (%)	<i>DR</i> (%)	Classifier
IFSFOA	59.77(70%~30%)	61.53	1-NN	IFSFOA	87.54(70%~30%)	86.33	5-NN
FSFOA	55.55(70%~30%)	71.42	1-NN	FSFOA	86.98(70%~30%)	44.26	5-NN
SVM-FuzCoc	61.01(70%~30%)	46.1	1-NN	PSO(4-2)	78.16(70%~30%)	81.26	5-NN
SFS	51.79(70%~30%)	47.7	1-NN	IFSFOA	91.96(70%~30%)	79.33	1-NN
SBS	54.8(70%~30%)	38.5	1-NN	FSFOA	85.43(70%~30%)	57.37	1-NN
SFFS	49.5(70%~30%)	53.8	1-NN	SVM-FuzCoc	73.17(70%~30%)	68.33	1-NN
SRBCT	<i>Accuracy</i> (%)	<i>DR</i> (%)	Classifier	SFS	66.43(50%~50%)	61.33	1-NN
IFSFOA	95.55(70%~30%)	89.66	1-NN	SBS	62.2(50%~50%)	45.33	1-NN
FSFOA	94.73(70%~30%)	49.06	1-NN	SFFS	64.55(50%~50%)	61.33	1-NN
SVM-FuzCoc	98.88(70%~30%)	98.57	1-NN	IFSFOA	78.84(2-fold)	85	Rbf-svm
Segmentatin	<i>Accuracy</i> (%)	<i>DR</i> (%)	Classifier	FSFOA	65.86(2-fold)	54.09	Rbf-svm
IFSFOA	96.32(10-fold)	65.26	3-NN	HGAFS	87.02(2-fold)	75	Rbf-svm
FSFOA	96.2(10-fold)	30	3-NN	IFSFOA	93.78(10-fold)	81.66	J48
NSM	95(10-fold)	63.15	3-NN	FSFOA	82.69(10-fold)	52.45	J48
Vehicle	<i>Accuracy</i> (%)	<i>DR</i> (%)	Classifier	FS-NEIR	75.97(10-fold)	91.66	J48
IFSFOA	75.39(70%~30%)	50	5-NN	Heart_statlog	<i>Accuracy</i> (%)	<i>DR</i> (%)	Classifier
FSFOA	73.98(70%~30%)	50	5-NN	IFSFOA	91.85(10-fold)	70	3-NN
PSO(4-2)	85.3(70%~30%)	68.4	5-NN	FSFOA	85.18(10-fold)	35.71	3-NN
IFSFOA	69.62(2-fold)	75	Rbf-svm	NSM	84(10-fold)	69.23	3-NN
FSFOA	62.41(2-fold)	47.22	Rbf-svm	IFSFOA	84.44(2-fold)	57.69	Rbf-svm
HGAFS	76.36(2-fold)	38.99	Rbf-svm	FSFOA	84.07(2-fold)	50	Rbf-svm
IFSFOA	83.57(10-fold)	42.22	J48	HGAFS	82.59(2-fold)	76.92	Rbf-svm
FSFOA	73.04(10-fold)	31.57	J48	IFSFOA	93.70(10-fold)	61.53	J48
FS-NEIR	70.98(10-fold)	50	J48	FSFOA	85.15(10-fold)	48.07	J48
				FS-NEIR	79.86(10-fold)	46.15	J48

Table 4 Classification accuracy and dimension reduction of IFSFOA and compared methods (Continued)**表 4** IFSFOA 及其对比算法的分类准确率和维度缩减率(续)

Dermatology	<i>Accuracy (%)</i>	<i>DR (%)</i>	Classifier	Glass	<i>Accuracy (%)</i>	<i>DR (%)</i>	Classifier
IFSFOA	99.79(70%~30%)	51.96	1-NN	IFSFOA	74.28(70%~30%)	48.88	1-NN
FSFOA	97.27(70%~30%)	45.71	1-NN	FSFOA	71.88(70%~30%)	40	1-NN
SVM-FuzCoc	94.11(70%~30%)	64.7	1-NN	SFS	72.24(70%~30%)	26.66	1-NN
SFS	94.02(70%~30%)	44.7	1-NN	SFFS	71.77(70%~30%)	37.77	1-NN
SBS	91.78(70%~30%)	58.23	1-NN	IFSFOA	71.03(2-fold)	44.44	Rbf-svm
SFFS	93.7(70%~30%)	62.35	1-NN	FSFOA	68.22(2-fold)	60	Rbf-svm
IFSFOA	99.17(10-fold)	69.17	J48	HGAFS	65.51(2-fold)	44.44	Rbf-svm
FSFOA	96.99(10-fold)	21.42	J48	IFSFOA	77.72(10-fold)	54.44	J48
FS-NEIR	93.95(10-fold)	70.58	J48	FSFOA	75.7(10-fold)	50	J48
IFSFOA	98.55(70%~30%)	61.11	J48	FS-NEIR	68.53(10-fold)	22.22	J48
FSFOA	90.09(70%~30%)	44.11	J48				
UFSACO	95.28(70%~30%)	26.47	J48				

Arcene	<i>Accuracy (%)</i>	<i>DR (%)</i>	Classifier
IFSFOA	76.85(70%~30%)	95.4	J48
FSFOA	73.69(70%~30%)	77.67	J48
UFSACO	67.40(70%~30%)	99.8	J48

从表 4 可以直观地看到:

- 当指导学习的分类器和数据集划分方法相同时,对比其他所有算法,IFSFOA 在 Ionosphere,Glass, Heart-statlog,Dermatology,Arcene 数据集上,其分类准确率都是最高的;
- 而在 Sonar,Cleveland 数据集中,分类准确率虽不能持续最高,但也达到了次高水平,其中,在 Sonar 数据集中,当分类器为 1NN 时,IFSFOA 分类准确率比 SBS 高出 30%;
- 在 Vehicle 数据集中,IFSFOA 虽仅优于众多方法中的一个(不包括 FSFOA),但在 J48 分类器下,其准确率要比 FS-NEIR 高出 13%;
- 而在 SRBCT 数据集中,与 SVM-FuzCoc 相比,IFSFOA 的性能并不理想.其原因可能是该数据集特征的数量(2308)远远超出了实例的数量(63),这使得算法很难选择适当的特征进行预测;但与 FSFOA 算法相比,IFSFOA 算法虽然在准确率上仅提高 1%左右,但在维度缩减率上却显著提高了 50%左右;
- 在 Arcene 数据集中,IFSFOA 算法的分类准确率比 UFSACO 高出 13%左右,但在维度缩减率上低于 UFSACO 算法,IFSFOA 算法以牺牲 4%左右的维度换取了更高的分类准确率.

IFSFOA 总体来说,除在数据集 Wine 中,当分类器是 1NN,5NN 时,分类性能略低于 FSFOA 算法(但维度缩减却显著提高了 18%左右)以及 Cleveland,Glass 数据集中维度削减低于 FSFOA 算法外,对于其他 8 个数据集,IFSFOA 算法在准确率和维度缩减上均高于 FSFOA.这说明改进后的 IFSFOA 算法确有成效.

比较表 4 中各算法的 DR 值,很明显,FSFOA 的效果并不好.除在 Glass 数据集中 DR 值最高外,在其他数据集中,DR 值几乎是最差的.其原因主要是因为 FSFOA 在更新过程中只使用分类准确率作为评价函数,并未考虑到在特征选择过程中维度缩减情况.改进的 IFSFOA 采用新的更新策略,克服传统更新机制的局限性.很明显,对比 FSFOA,IFSFOA 的 DR 值有了显著提高,甚至在 Wine 数据集 10-fold 划分条件下,分类器为 J48 时,IFSFOA 要比 FSFOA 高出 58%;对比其他所有算法(不包括 FSFOA),IFSFOA 的 DR 值仍具有很强的竞争力,在 Sonar 数据集 70%~30%划分条件下,分类器为 1NN 时,IFSFOA 要比效果最好的 SVM-FuzCoc 算法高出 11%.

本文针对 FSFOA 算法提出了 3 点改进,其中,初始化策略和在局部播种阶段加入贪婪策略用来提高分类准确率,而新的更新机制则是用来提高维度缩减率.因此在 3 点改进中,只有新的更新机制对实验结果中的 DR 值有影响.对实验结果中分类准确率影响更大的则是初始化策略,因为新的初始化策略为整个算法的寻优奠定了良好的基础.在局部播种阶段加入贪婪策略对准确率的提高起到了辅助作用,因为算法每次循环时只会产生 LSC 棵新树,而 LSC 又是一个比较小的值,同时,新树在形成候选森林阶段又会面临着被移除森林的风险.

5 总 结

本文针对 FSFOA 算法的不足之处,提出了 3 点改进:在 IFSFOA 的初始化阶段,我们采用新的初始化策略,利用前向选择和后向选择的优点,摒弃其缺点,形成双向选择的策略;在更新机制上,克服传统更新机制的局限性,将维度缩减问题也纳入考虑范围;在算法的局部播种阶段,为避免森林中存在过多劣质树,增加搜索难度,影响分类性能,我们采用了极度贪婪的策略.将 IFSFOA 算法在 11 个数据集和 10 个特征选择算法上进行测试、比较,实验结果表明,IFSFOA 普遍提高了数据集的分类准确率和维度缩减能力.

References:

- [1] Zhou ZH. Machine Learning. Beijing: TsingHua University Press, 2016. 247–261 (in Chinese).
- [2] Guyon I, Elisseeff A. An introduction to variable and feature selection. *Journal of Machine Learning Research*, 2003,3(6): 1157–1182.
- [3] Ghaemi M, Feizi-Derakhshi MR. Feature selection using forest optimization algorithm. *Pattern Recognition*, 2016,60:121–129.
- [4] Xue B, Zhang M, Browne WN. Novel initialisation and updating mechanisms in PSO for feature selection in classification. In: *Proc. of the European Conf. on Applications of Evolutionary Computation*. Springer-Verlag, 2013. 428–438.
- [5] Nie DG. Research on improvements and discretization of forest optimization algorithm [MS. Thesis]. Lanzhou: Lanzhou University, 2016 (in Chinese with English abstract).
- [6] Tabakhi S, Moradi P, Akhlaghian F. An unsupervised feature selection algorithm based on ant colony optimization. *Engineering Applications of Artificial Intelligence*, 2014,32(6):112–123.
- [7] Zhu W, Si G, Zhang Y, *et al.* Neighborhood effective information ratio for hybrid feature subset evaluation and selection. *Neurocomputing*, 2013,99:25–37.
- [8] Xue B, Zhang M, Browne WN. Particle swarm optimisation for feature selection in classification. *Applied Soft Computing*, 2014, 18(C):261–276.
- [9] Hu Q, Che X, Zhang L, *et al.* Feature evaluation and selection based on neighborhood soft margin. *Neurocomputing*, 2010, 73(10–12):2114–2124.
- [10] Moustakidis SP, Theocharis JB. SVM-FuzCoC: A novel SVM-based feature selection method using a fuzzy complementary criterion. *Pattern Recognition*, 2010,43(11):3712–3729.
- [11] Huang J, Cai Y, Xu X. A hybrid genetic algorithm for feature selection wrapper based on mutual information. *Pattern Recognition Letters*, 2007,28(13):1825–1844.
- [12] Hall MA. Correlation-Based Feature Selection for Machine Learning. 1999. 19.
- [13] Hall MA. Correlation-Based feature selection for discrete and numeric class machine learning. In: *Proc. of the 17th Int'l Conf. on Machine Learning*. Morgan Kaufmann Publishers Inc., 2000. 359–366.
- [14] Kira K, Rendell LA. The feature selection problem: Traditional methods and a new algorithm. In: *Proc. of the 10th National Conf. on Artificial Intelligence*. AAAI Press, 1992. 129–134.
- [15] Almuallim H, Dietterich TG. Efficient algorithms for identifying relevant features. In: *Proc. of the Canadian Conf. on Artificial Intelligence*. Oregon State University, 1992. 38–45.
- [16] Kohavi R, John GH. Wrappers for Feature Subset Selection. Elsevier Science Publishers Ltd., 1997.
- [17] Liu H, Setiono R. Feature selection and classification—A probabilistic wrapper approach. In: *Proc. of the Int'l Conf. on Industrial & Engineering Applications of AI & ES*. 1996. 419–424.
- [18] Efron B, Hastie T, Johnstone I, *et al.* Least angle regression. *Annals of Statistics*, 2004,32(2):407–451.
- [19] Almuallim H, Dietterich TG. Learning Boolean concepts in the presence of many irrelevant features. *Artificial Intelligence*, 1994, 69(1–2):279–305.
- [20] Pudil P, Novovičová J, Kittler J. Floating search methods in feature selection. *Pattern Recognition Letters*, 1994,15(11):1119–1125.
- [21] Fujarewicz K, Wiench M. Selecting differentially expressed genes for colon tumor classification. *International Journal Of Applied Mathematics and Computer Science*, 2003,13(3):327–335.

- [22] Guyon I, Weston J, Barnhill S. Gene selection for cancer classification using support vector machines. *Machine Learning*, 2002, 46(1-3):389–422.
- [23] Mao Y, Zhou XB, Xia Z, *et al.* A survey for study of feature selection algorithms. *Pattern Recognition and Artificial Intelligence*, 2007,20(2):211–218 (in Chinese with English abstract).
- [24] Kabir MM, Shahjahan M, Murase K. A new hybrid ant colony optimization algorithm for feature selection. *Expert Systems with Applications*, 2012,39(3):3747–3763.
- [25] Hamdani TM, Won JM, Alimi AM, *et al.* Hierarchical genetic algorithm with new evaluation function and bi-coded representation for the selection of features considering their confidence rate. *Applied Soft Computing*, 2011,11(2):2501–2509.
- [26] Tan KC, Teoh EJ, Yu Q, *et al.* A hybrid evolutionary algorithm for attribute selection in data mining. *Expert Systems with Applications*, 2009,36(4):8616–8630.
- [27] Ghaemi M, Feizi-Derakhshi MR. Forest optimization algorithm. *Expert Systems with Applications*, 2014,41(15):6676–6687.
- [28] Whitney AW. A direct method of nonparametric measurement selection. *IEEE Trans. on Computers*, 1971,20(9):1100–1103. [doi: 10.1109/T-C.1971.223410]
- [29] Marill T, Green DM. On the effectiveness of receptors in recognition systems. *IEEE Trans. on Information Theory*, 1963,9(1): 11–17. [doi: 10.1109/TIT.1963.1057810]
- [30] Tahir MA, Bouridane A, Kurugollu F. Simultaneous feature selection and feature weighting using hybrid tabu search/*K*-nearest neighbor classifier. *Pattern Recognition Letters*, 2007,28(4):438–446.

附中文参考文献:

- [1] 周志华.机器学习.北京:清华大学出版社,2016.247–261.
- [5] 聂大千.森林优化算法的改进及离散化研究[硕士学位论文].兰州:兰州大学,2016.
- [23] 毛勇,周晓波,夏铮,等.特征选择算法研究综述.模式识别与人工智能,2007,20(2):211–218.



初蓓(1990—),女,吉林桦甸人,硕士生,主要研究领域为机器学习.



张梦林(1991—),男,硕士生,主要研究领域为机器学习.



李占山(1966—),男,博士,教授,博士生导师,CCF 专业会员,主要研究领域为约束优化与约束求解,机器学习,基于模型的诊断,智能规划与调度.



于海鸿(1975—),男,博士,讲师,主要研究领域为约束优化与约束求解,大数据与数据挖掘,智能规划与调度.