计算机科学 COMPUTER SCIENCE
http://www.jsjkx.com
DOI:10.11896/jsjkx.190700089

基于遗传实例和特征选择的K近邻训练集优化方法

董明刚[1,2]　黄宇扬[1]　敬　超[1,2]
1 桂林理工大学信息科学与工程学院　广西 桂林 541004
2 广西嵌入式技术与智能系统重点实验室　广西 桂林　541004
(d2015mg@qq.com)

摘　要　K近邻的分类性能依赖于训练集的质量。设计高效的训练集优化算法具有重要意义。针对传统的进化训练集优化算法效率较低、误删率较高的不足,提出了一种遗传训练集优化算法。该算法采用基于最大汉明距离的高效遗传算法,每次交叉保留父代并生成两个新的具有最大汉明距离的子代,既提高了效率,又保证了种群多样性。该算法将局部的噪声样本删除策略与特征选择策略相结合。首先使用决策树算法确定噪声样本存在的范围,然后使用遗传算法精准删除此范围内的噪声样本和全局的噪声特征,降低了误删率,提高了效率。该算法采用基于最近邻规则的验证集选择策略,进一步提高了遗传算法实例选择和特征选择的准确度。在15个标准数据集上,该方法相较于协同进化实例特征选择算法IFS-CoCo、加权协同进化实例特征选择算法CIW-NN、进化特征选择算法EIS-RFS、进化实例选择算法PS-NN、K近邻算法KNN,在分类精度上分别平均提升了2.18%,2.06%,5.61%,4.06%和4.00%。实验结果表明,所提方法的分类精度和优化效率优于当前的进化训练集优化算法。

关键词:遗传算法;K近邻;实例选择;特征选择;噪声样本;决策树
中图法分类号　TP181

K-Nearest Neighbor Classification Training Set Optimization Method Based on Genetic Instance and Feature Selection

DONG Ming-gang[1,2],HUANG Yu-yang[1] and JING Chao[1,2]
1 College of Information Science and Engineering,Guilin University of Technology,Guilin,Guangxi 541004,China
2 Guangxi Key Laboratory of Embedded Technology and Intelligent System,Guilin,Guangxi 541004,China

Abstract　The classification performance of K-Nearest Neighbor depends on the quality of training set. It is significant to design an efficient training set optimization algorithm. Two major drawbacks of traditional evolutionary training set optimization algorithm are low efficiency and removing the non-noise samples and features by mistake. To address these issues,this paper proposes a genetic training set optimization algorithm. The algorithm uses the efficient genetic algorithm based on the maximum Hamming distance. Each cross preserves the parent and generates two new children with the largest Hamming distance,which not only improves the efficiency but also ensures the population diversity. In the proposed algorithm,the local noise sample deletion strategy is combined with the feature selection strategy. Firstly,the decision tree is used to determine the range of noise samples. Then the genetic algorithm is used to remove the noise samples in this range and select the features simultaneously. It reduces the risk of mistaken and improves the efficiency. At last,the 1NN-based selection strategy of validation set is used to improve the instance and feature selection accuracy of the genetic algorithm. Compared with co-evolutionary instance feature selection algorithm (IFS-CoCo),weighted co-evolutionary instance feature selection algorithm (CIW-NN),evolutionary feature selection algorithm (EIS-RFS,evolutionary instance selection algorithm (PS-NN) and traditional KNN,the average improvement of the proposed algorithm in classification accuracy is 2.18%,2.06%,5.61%,4.06%,4.00%,respectively. The experiments results suggest that the proposed method has higher classification accuracy and optimization efficiency.

Keywords　Genetic algorithm,K-nearest neighbor,Instance selection,Feature selection,Noise sample,Decision tree

到稿日期:2019-07-15　返修日期:2019-10-11　本文已加入开放科学计划(OSID),请扫描上方二维码获取补充信息。
基金项目:国家自然科学基金(61563012,61802085,61203109);广西自然科学基金(2014GXNSFAA118371,2015GXNSFBA139260);广西嵌入式技术与智能系统重点实验室基金(2018A-04)
This work was supported by the National Natural Science Foundation of China (61563012,61802085,61203109),Guangxi Natural Science Foundation (2014GXNSFAA118371,2015GXNSFBA139260) and Guangxi Key Laboratory of Embedded Technology and Intelligent Systems (2018A-04).
通信作者:敬超(jingchao@glut.edu.cn)

1 引言

分类是机器学习中重要的任务,K 近邻(K-Nearest Neighbor,KNN)分类算法是机器学习中最经典的分类方法之一[1],它具有简单性和有效性,已经被成功应用到实际的分类问题中[2-4]。KNN 通过训练集中周围的样本来预测新样本的类,因此其分类精度和分类效率由训练集的样本和样本特征决定。目前,KNN 分类方法还存在以下不足:1)训练集维度较高时,其分类效率较低[5-6];2)训练集存在噪声特征或噪声样本时,分类精度会被严重影响[7-9]。

对 KNN 训练集进行优化,可以有效弥补以上不足。实例选择(Instance Selection,IS)[10]和特征选择(Feature Selection,FS)[11]是训练集优化的主要方式。IS 通过精简训练集来提升分类效率,并且通过移除噪声样本来提高分类精度。与 IS 不同,FS 是通过寻找训练效果较好的代表特征子集用于分类,以提高分类精度。两种方法都降低了原始训练集的维度,删除了噪声数据,改善了 KNN 的分类效果。

进化算法[12]是一种具有高鲁棒性、灵活性的优化方法,能有效地进行 IS 和 FS。文献[13]将 IS 问题看作一个优化问题,通过自然选择选出最优的训练样本集进行训练以收获较优的分类效果。文献[14]使用遗传算法对样本特征进行选择,排除了噪声特征,有效提高了 KNN 的分类精度。文献[15]提出了一种混合的遗传算法,并用它来进行 FS,提高了 FS 的准确度。文献[16]使用遗传算法同时进行 IS 与 FS 协同选择(Instance and Feature Selection,IFS),然后将它们的分类效果进行对比,使用其中最优方式选出的训练集进行训练,以进一步提升 KNN 分类算法的分类效果。以上方法虽对 KNN 性能有一定的提升,但还存在如下问题:1)需要较大的种群规模和较多的遗传迭代次数才能选出训练效果较优的训练集,算法效率较低;2)大范围地删减训练集中的噪声样本和噪声特征,存在误删除的风险。

针对以上不足,本文设计了一种新的基于遗传实例特征选择的 KNN 分类训练集优化方法(Genetic Instance and Feature Selection algorithm,GIFS)。本文主要的创新性工作如下:

1)提出基于最大汉明距离的高效遗传算法(Efficient Genetic Algorithm based on Maximum Hamming Distance,EGA),选出优秀的个体作为父代;每次交叉保留父代,并随机选择交叉位置交叉产生两个具有最大汉明距离的子代,每次交叉能产生 4 个子代,提高了遗传算法的效率。基于最大汉明距离的交叉规则保证了种群的多样性。

2)提出将局部的噪声样本删除策略与特征选择策略相结合。首先,利用决策树来判定噪声的范围,然后使用遗传算法精准删除该范围内的噪声样本,同时进行特征选择。该策略在缩减实例选择范围的同时能降低误删率,与采用整个训练集进行优化的算法[16-19]相比,该算法的效率和精度得到了提升。将实例选择与特征选择相结合,能进一步提高分类精度。

2 相关工作

2.1 实例与特征协同选择

IS 与 FS 均能有效地提高 KNN 的分类精度。一些研究为了进一步提高分类精度,将实例选择与特征选择相结合进行协同选择,即 IFS。文献[20]提出了使用遗传算法对 KNN 的训练集同时进行 IS 和 FS,以获取训练效果较优的训练样本集及样本特征。

近年来,将 FS,IS,IFS 结合进行协同进化成为了研究的热点。文献[16]较早地采用了这种协同进化的思想来优化 KNN 的训练集,有效地提高了 KNN 的分类性能。文献[17]赋予样本和样本特征不同的权重,使用稳态遗传算法同时进行 FS,IS 和 IFS,选出最优的训练集进行训练,大幅提升了 KNN 的分类精度。

2.2 遗传算法与训练集优化

遗传算法是进化算法中进行 IS,FS 和 IFS 最主要的方法。通过遗传算法能有效删除训练集中的噪声样本和噪声特征,选出优秀的训练样本集和样本特征。为了使遗传算法更有效地进行训练集优化,研究者相继提出了与之相关的新的遗传算法以及策略。

文献[8]提出了一种改进的遗传算法,采用了基于聚类的交叉策略以及一种快速智能的突变机制。该遗传算法相较于传统的遗传算法,在实例选择上有着较高的准确度。文献[21]提出的一种自适应的搜索策略(CHC Adaptive Search Algorithm,CHC)具有较高的鲁棒性、有效性,它的交叉策略能有效地保证种群的多样性。使用基于 CHC 的遗传算法进行 IS,FS 和 IFS,能有效提升选择的准确度。

3 GIFS 算法

3.1 EGA 算法

3.1.1 算法流程

传统的遗传算法每次交叉只产生两个新的子代。为了提高遗传算法进行实例选择和特征选择的效率和稳定性,本文使用轮盘赌法随机选择优秀的当代个体(个体适应度优于平均适应度)作为父代。每次交叉将保留父代并产生两个具有最大汉明距离的子代,共产生 4 个个体。用 $\boldsymbol{P}_1$ 和 $\boldsymbol{P}_2$ 分别表示父代的两个基因,用 $\boldsymbol{C}_1$,$\boldsymbol{C}_2$,$\boldsymbol{C}_3$ 和 $\boldsymbol{C}_4$ 分别表示产生的 4 个子代,整体交叉过程如图 1 所示。首先,复制父代基因 $\boldsymbol{P}_1$ 和 $\boldsymbol{P}_2$ 作为优秀的染色体保留在子代中,分别用 $\boldsymbol{C}_1$ 和 $\boldsymbol{C}_2$ 表示。其次,由 $\boldsymbol{P}_1$ 和 $\boldsymbol{P}_2$ 交叉产生 $\boldsymbol{C}_3$ 和 $\boldsymbol{C}_4$,其过程为:逐位对比 $\boldsymbol{P}_1$ 与 $\boldsymbol{P}_2$ 交叉位置的基因,若该位基因不同,则双方基因进行交换;若该位基因相同,则对 $\boldsymbol{P}_1$ 该位基因进行取反,$\boldsymbol{P}_2$ 该位基因保持不变,变换后的 $\boldsymbol{P}_1$ 为子代 $\boldsymbol{C}_4$,变换后的 $\boldsymbol{P}_2$ 为子代 $\boldsymbol{C}_3$。在遗传算法的交叉过程中,我们只能改变交叉片段的基因,因此交叉片段的汉明距离大小决定了整条染色体的汉明距离大小。该交叉方式保证了 $\boldsymbol{C}_3$ 和 $\boldsymbol{C}_4$ 在交叉片段中具有最大的汉明距离(该距离为交叉片段的长度),从而使这两条染色体具有最大的汉明距离。$\boldsymbol{C}_3$ 和 $\boldsymbol{C}_4$ 的产生以图 2 为例,$\boldsymbol{P}_1$ 和 $\boldsymbol{P}_2$ 的交叉位置为基因片段 3—8,交叉片段长度为 6。在交叉片段中,$\boldsymbol{P}_1$ 和 $\boldsymbol{P}_2$ 的第 3,4,5,7,8 位基因相同,第 6 位基因不同。首先,对 $\boldsymbol{P}_1$ 的 3,4,5,7,8 位基因取反,依次为 0,0,0,1,1,其他位基因保持不变;其次,$\boldsymbol{P}_1$ 和 $\boldsymbol{P}_2$ 交换第 6 位基因,从而生成对应的子代 $\boldsymbol{C}_4$ 和 $\boldsymbol{C}_3$。交叉产生的子代 $\boldsymbol{C}_3$ 和 $\boldsymbol{C}_4$ 在交叉片段的汉明距离为 6,整体的汉明距离为 10。若利用传统的交叉方式,$\boldsymbol{C}_3$ 和 $\boldsymbol{C}_4$ 的染色体分别为 0000001111 和 1100011100,交叉片段的汉明距离为 1,整体的汉明距离为 5。

本文算法的具体流程如算法 1 所示。

算法 1 基于最大汉明距离的高效遗传算法

输入:初始种群

输出:最优个体

```
1. 评估初始种群;
2. while 未达到遗传迭代次数
3.    while 子代数量未达到种群规模
4.       轮盘赌法选择优秀的个体作为父代;
5.       将父代作为子代保留;
6.       随机选择父代双方的交叉片段(两交叉点);
7.       for i=交叉起始位置:交叉结束位置
8.          if 父代双方第 i 位基因不相同
9.             父代双方交换第 i 位基因;
10.         else
11.            父代的第一个个体的第 i 位基因取反;
12.         end if
13.            保留交叉产生的子代;
14.      end for
15.   end while
16.   将二进制突变应用到整个种群中;
17.   评估子代;
18. end while
```

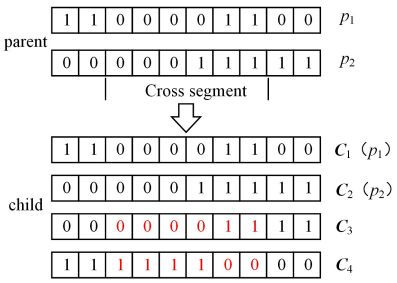

图 1 交叉过程

Fig. 1 Crossing operator

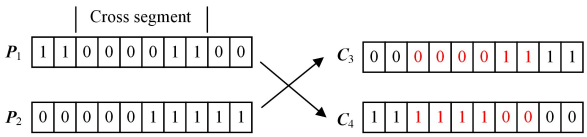

图 2 C_3 和 C_4 的产生

Fig. 2 Production of C_3 and C_4

3.1.2 算法推论

定义 1 模式是一个相同的构型,它描述的是一个数字串的子集合,在这个集合中的所有数字串在某些位置上是相同的,一般用大写字母 H 表示。模式 H 的定义长度用 $\delta(H)$ 表示;模式 H 的阶用 $O(H)$ 表示;数字串长用 l 表示。模式的相关概念参见文献[22]。

推论 1 EGA 能有效保证种群的多样性。

在交叉操作中,EGA 在保留原父代(C_1 和 C_2)的同时,采用两点片段交叉产生两个新的子代(C_3 和 C_4)。C_1 和 C_2 分别保留了父代两个个体的模式,因此它们继承模式的生存概率均为 1。设模式 H 第一个常数的位置为 a,最后一个常数的位置为 b,则 $b-a=\delta(H)$。在不考虑父母双方交叉片段某些常数位相同的情况下,为了使来自父代的模式不被破坏,两个交叉点必须同时落在 a 点以前或 b 点以后。因此,C_3 和 C_4 从 P_1 和 P_2 继承的模式生存概率为 $\frac{a^2}{l^2}+\frac{(l-b)^2}{l^2}$(若父母交叉片段某些常数位相同,即使进行交叉,某些模式也不会被破坏)。设 p_l 为幸存率,表示父代交叉后实际保留的模式数量和不考虑父母双方某些常数位相同的情况下保留模式数量的比值($p_l \geqslant 1$)。EGA 的交叉机制使 C_4 的继承模式的幸存率最低,即 $p_l=1$。设交叉的概率为 p_c,模式生存的概率为 p_ε,综合以上分析得出 EGA 模式生存定理,如式(1)所示:

$$p_\varepsilon=\begin{cases}1, & C_1, C_2\\ p_c \cdot p_l \cdot \dfrac{a^2+(l-b)^2}{l^2}, & C_3\\ p_c \cdot \dfrac{a^2+(l-b)^2}{l^2}, & C_4\end{cases} \tag{1}$$

每次交叉,C_1 和 C_2 将保留父代双方的模式,不生成新的模式。考虑到幸存率 p_l,C_3 将生成 $p_c \cdot \frac{2^{\delta(H)}}{p_l}$ 种新的模式。C_4 的 $p_l=1$,它生成的新模式最多,数量为 $p_c \cdot 2^{\delta(H)}$。设每一代生成新模式的数量为 S,EGA 的模式生成定理如式(2)所示。由式(1)、式(2)可知,EGA 在保留父母原有模式(C_1 和 C_2)的基础上,会生成大量的新模式(C_3 和 C_4),保证了种群的多样性,从而可证推论 1 成立。

$$S=\begin{cases}0, & C_1, C_2\\ p_c \cdot \dfrac{2^{\delta(H)}}{p_l}, & C_3\\ p_c \cdot 2^{\delta(H)}, & C_4\end{cases} \tag{2}$$

3.2 实例与特征选择策略

在进行实例选择与特征选择前,使用决策树将训练集分成几个样本子集。根据样本子集的主类占比 p 和阈值 α($0 \leqslant \alpha \leqslant 1$)确定该子集是否是噪声样本大量存在的样本子集(每个样本子集的 p 为子集中最大同类样本数量和总样本数量的比值)。若该样本子集 p 小于或等于 $1-\alpha$,则该样本子集为噪声样本子集,否则为非噪声样本子集。噪声样本子集中的样本将全部放入 T_{noise},随后使用遗传算法在 T_{noise} 中精确删除噪声样本,保留非噪声样本。非噪声样本子集中的样本全部保留在训练集中。样本子集的处理遵循式(3):

$$\begin{cases}\text{reserve}, & p>1-\alpha\\ T_{\text{noise}}, & p \leqslant 1-\alpha\end{cases} \tag{3}$$

α 决定非噪声样本子集的数量,若 α 很小(如 0),几乎所有样本都被视为噪声样本,会使遗传算法难以在庞大的噪声样本中高效、准确地进行实例特征选择;若 α 很大(如 1),几乎所有样本都被视为非噪声样本,相当于直接使用 KNN 对原始训练集进行分类,将无法优化训练集。因此,建议将 α 设置在 0.1~0.5 之间,以准确、有效地进行子集划分。为了使用遗传算法删除 T_{noise} 中的噪声样本,同时对训练集的样本特征进行选择,本文将 T_{noise} 里的样本和训练集的样本特征形式化为一条染色体,如图 4 所示。N 为所有 T_{noise} 中包含的样本总量,M 为训练集样本特征的数量。染色的前 N 位是 T_{noise} 基因片段,每一位代表 T_{noise} 里的每一个样本。如果前 N 位的某位基因为 1,则代表的相应样本为非噪声样本,将保留在训练集中;反之将从训练集中删除。染色体的后 M 位是样本特征片段,每一位代表样本的一种特征。如果后 M 位中的某位基因为 0,则代表相应特征为噪声特征,将从特征集中删除;反

之将保留在特征集中。本文将使用 EGA 获取最优的染色体，如图 5 所示。

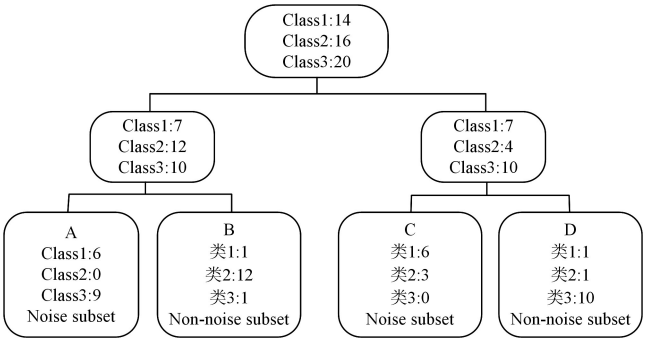

图 3 样本子集的处理

Fig. 3 Processing of sample subsets

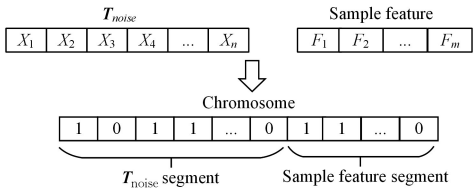

图 4 $\boldsymbol{T}_{\text{noise}}$ 和样本特征的基因

Fig. 4 Genes of $\boldsymbol{T}_{\text{noise}}$ and sample feature

3.3 验证集选择策略

验证集用于辅助模型的构建。传统的验证集是从训练集中随机选出与测试集等量的样本组合而成[8]。文献[23]采用最近邻规则复制训练集中与测试集最邻近的样本来组成验证集，该验证集选择策略使验证集的特征更接近测试集。本文算法能通过这些验证集，自适应地构造出更有效的训练集，得到更优秀的分类效果和更精准的分类精度。

3.4 分类精度惩罚函数

得到最优训练集的重点在于遗传算法的适应度计算函数的设计。文献[8]提出了更为有效和稳定的基于均方误差的分类精度惩罚函数，如式(4)所示：

$$F=\frac{1}{NC}\sum_{n=1}^{N}\sum_{i=1}^{C}\left(\frac{k_{n[i]}}{k}-\delta[i-c_n]\right)^2 \tag{4}$$

其中，k 代表 K 邻近值，$\frac{k_n[i]}{k}$ 代表验证集中第 n 个样本被分类为第 i 类的概率；c_n 为该样本的实际类别；N 表示验证集中样本的数量；C 是样本集的类别数。

3.5 GIFS 算法流程

GIFS 算法包括三大步骤：1)采用决策树来判定噪声样本存在的范围，即 $\boldsymbol{T}_{\text{noise}}$；2)使用 3.1 节提出的 EGA 对 $\boldsymbol{T}_{\text{noise}}$ 和样本特征进行选择，构造优化的训练集；3)采用构造的训练集来训练 KNN 分类器。算法的具体流程如算法 2 所示。

算法 2 GIFS 算法

1. 构造验证集。采用最近邻验证集选择策略选择与测试集最邻近的样本。
2. 预分类训练集。采用 C4.5 决策树将训练集划分为几个样本子集 $\mathbf{T}_1$，$\mathbf{T}_2$，$\mathbf{T}_3$，…，并计算样本子集的主类占比 p_1，p_2，p_3，…。
3. for i=1：样本子集的数目
4. $\mathbf{T}_i$ 的主类占比大于 1－α 则不做处理，否则将该子集的样本加入 $\mathbf{T}_{\text{noise}}$。
5. 根据 $\mathbf{T}_{\text{noise}}$ 总的特征数量 M 和样本数量 N，初始化 10 个 M+N 位二进制向量个体的种群，其中 9 个随机产生，剩余 1 个二进制向量每一位都为 1。
6. end for
7. 使用 EGA 同时进行局部的实例选择和全局的特征选择，获取最优的个体，其中遗传算法的适应度通过式(4)计算。
8. 利用 KNN 算法，基于全局最优个体对应的 **Tr** 对 **Te** 中的所有样本进行类别标号。
9. 输出标号后的数据集 $\mathbf{T}_e$。

步骤 5 将每一位都为 1 的二进制向量作为初代个体之一。算法的流程如图 5 所示。

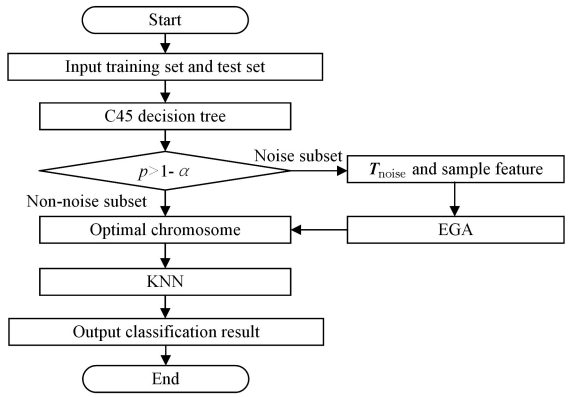

图 5 GIFS 算法的流程

Fig. 5 Flow of GIFS

4 实验方案

4.1 实验设置

本文利用 Keel 软件进行实验。对本文算法 GIFS、协同进化算法 IFS-CoCo[16]、加权协同进化算法 CIW-NN[17]、特征选择算法 EIS-RFS[18] 和 PHGA[15](PHGA 仅适用二分类问题)、实例选择算法 PS-NN[13]、传统的 KNN 算法[1] 这 7 个算法进行对比。实验数据是 15 个公开的数据集，其中 6 个数据集只包含两种类别的样本，9 个数据集包含两种以上类别的样本。数据集的基本信息如表 1 所列。

表 1 数据集信息

Table 1 Information of data sets

Dataset	Examples	Attributes	Classes
glass	214	9	7
vowel	990	13	11
segment	2310	19	7
new-thyroid	215	5	3
automobie	150	25	6
hayes	160	4	3
contrace	1473	9	3
cleveland	297	13	5
breast	277	9	2
ionosphere	351	33	2
sonar	208	60	2
saheart	462	9	2
spectfheart	267	44	2
bupa	345	6	2
page	5472	10	5

为了更好地验证 GIFS 的稳定性和有效性，使用 10 折交叉验证，实验结果取 3 次(共 30 次)的平均值。在优化训练集

后，训练集中的噪声样本数量得以削减，此时选择相对较大的 k 值可以减小噪声样本对 K 近邻分类的影响，以获得较优的分类效果，因此设置 7 种算法的 $k=7$；GIFS 的 $\alpha=0.2$；遗传迭代 50 次，GIFS，IFS-CoCo，CIW-NN 和 EIS-RFS 初始化 10 个个体；7 种算法的其余参数为相应的默认值。

4.2 评价标准

本文采用以下两种评价指标来评价实验结果：1）分类精度，它是衡量一个分类器分类性能最常用的指标之一；2）Kappa 系数，它是用于评价分类与完全随机分类相比分类错误率减少的比例。

5 实验结果

5.1 分类精度结果及分析

在 4.1 节的实验条件下，7 种算法的分类精度对比结果如表 2（分类精度±标准差，其中最优分类精度用黑体表示）和图 6 所示。由实验结果可知：

1）GIFS 在最优分类精度占比（10/15）和平均分类精度方面都高于其他对比算法；

2）GIFS 平均分类精度高于 PS-NN 4.6%，提高范围为 0.75%～22.96%；

3）GIFS 平均分类精度高于 CIW-NN 2.06%，提高范围为 0.12%～17.56%；

4）GIFS 平均分类精度高于 IFS-CoCo 2.18%，提高范围为 0.22%～12.34%；

5）GIFS 平均分类精度高于 EIS-RFS 5.61%，提高范围为 0.11%～40.10%；

6）GIFS 平均分类精度高于 KNN 4.00%，提高范围为 0.16%～26.66%；

7）GIFS 在二类数据集上的平均分类精度优于专门处理二类问题的 PHGA，提高范围为 1.33%～2.88%。

表 2 7 种算法的分类精度

Table 2 Classification accuracy of seven algorithms

（单位：%）

Alg	GIFS	PS-NN	CIW-NN	IFS-CoCo	EIS-RFS	PHGA	KNN
glass	**68.12±11.43**	62.61±8.08	64.99±10.51	63.94±11.99	59.67±12.50	—	66.82±11.24
vowel	**88.84±3.32**	65.88±10.29	74.74±6.65	80.90±5.40	48.74±4.69	—	88.68±2.58
segment	**95.83±1.37**	94.13±0.81	93.31±1.97	94.97±1.54	92.75±1.46	—	94.76±1.40
new-thyroid	**95.10±5.36**	92.29±5.36	94.44±4.52	91.06±7.51	92.13±6.11	—	93.06±6.00
automobie	**70.04±9.21**	52.65±7.75	52.48±9.79	57.70±10.29	55.47±11.62	—	54.25±10.90
hayes	55.41±13.70	50.62±9.91	63.95±12.50	**64.16±12.99**	53.12±12.67	—	28.75±11.24
contrace	**49.35±4.69**	46.90±1.77	47.92±3.36	48.94±3.26	46.41±4.03	—	47.65±3.31
cleveland	**57.86±6.17**	57.11±4.14	56.64±6.83	56.35±5.42	57.67±5.09	—	57.11±6.61
breast	73.35±5.55	75.54±4.42	73.23±6.77	72.88±5.42	73.24±5.99	**77.82±4.15**	72.53±6.61
ionosphere	85.66±3.77	84.60±4.80	**92.89±6.16**	84.52±4.50	85.18±5.58	88.59±5.00	84.32±4.81
sonar	**80.56±7.82**	71.00±6.25	78.30±9.21	73.98±8.07	73.42±5.58	77.68±8.33	79.21±9.94
saheart	68.87±6.43	69.73±5.10	66.7±6.86	68.44±5.85	**71.11±8.10**	66.93±3.90	67.28±3.58
spectfheart	77.15±8.08	**78.56±6.13**	77.01±5.61	76.93±5.60	78.54±4.96	75.02±7.05	74.49±8.08
bupa	**64.84±6.38**	61.48±6.09	63.95±8.12	64.05±8.21	61.03±5.01	63.51±7.55	62.52±6.46
page	**95.89±0.86**	94.76±0.65	95.49±0.72	95.38±0.79	94.20±0.80	—	95.38±0.63
Avg.	**75.12±6.28**	70.52±5.43	73.06±6.64	72.94±6.44	69.51±6.40	—	71.12±6.22

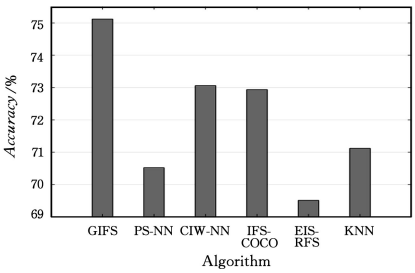

图 6 平均分类精度的实验结果

Fig. 6 Experiment results of average classification accuracy

GIFS 与其他 5 种对比算法在分类精度上威尔科克森符号秩检验的结果如表 3 所列。

表 3 分类精度威尔科克森符号秩检验

Table 3 Wilconx signed-rank test for accruacy

Alg	R^+	R^-	*P-value*
GIFS vs CIW-NN	95	25	0.0479
GIFS vs PS-NN	106	14	0.0067
GIFS vs IFS-Coo	106	14	0.0067
GIFS vs EIS-RFS	110	10	0.0004
GIFS vs KNN	120	0	0.00006

可以看出，GIFS 的 R^+ 值都远大于 R^- 值，其 *P-value* 都小于常规的显著水平（0.05）。这证明在这组实验的分类精度上，相对于其他对比算法，GIFS 表现得更优秀。

5.2 Kappa 系数结果及分析

在 4.1 节的实验条件下，5 种算法的 Kappa 系数（Kappa 系数±标准差，每个数据集的最优 Kappa 系数用黑体表示）对比结果如表 4 和图 7 所示。由实验数据得：

1）GIFS 的最优 Kappa 系数占比（8/15）和 Kappa 系数均值都比其他算法好；

2）GIFS 的平均 Kappa 系数高于 PS-NN 6.88%，提高范围为 2.01%～24.71%；

3）GIFS 的平均 Kappa 系数高于 CIW-NN 4.66%，提高范围为 0.88%～24.26%；

4）GIFS 的平均 Kappa 系数高于 IFS-CoCo 4.12%，提高范围为 0.04%～11.58%；

5）GIFS 的平均 Kappa 系数高于 EIS-RFS 13.46%，提高范围为 0.58%～52.65%；

6）GIFS 的平均 Kappa 系数高于 KNN 6.39%，提高范围为 1.16%～44.01%。

7）GIFS 在二类数据集上的平均 Kappa 系数优于专门处理二类问题的 PHGA，提高范围为 0.36%～9.85%。

表 4 7 种算法的 Kappa 系数

Table 4 Kappa of seven algorithms

（单位：%）

Alg	GIFS	PS-NN	CIW-NN	IFS-CoCo	EIS-RFS	PHGA	KNN
glass	**56.11±18.04**	47.12±1.06	50.21±15.57	47.48±11.97	45.46±13.05	—	53.66±14.6
vowel	87.18±4.10	62.47±1.13	69.40±6.77	79.55±4.15	40.73±4.59	—	**87.54±2.84**
segment	**95.16±1.41**	93.15±0.95	92.09±2.66	93.79±2.00	91.55±1.82	—	93.88±1.63
new-thyroid	**88.04±12.98**	80.70±13.94	87.63±9.41	77.42±21.90	77.84±20.46	—	82.72±16.64
automobie	**55.39±8.93**	36.89±10.95	37.04±11.53	43.81±14.91	35.55±18.16	—	38.97±14.90
hayes	27.84±23.06	18.34±21.24	**42.82±22.41**	38.64±18.75	26.74±20.50	—	-16.17±19.00
contrace	**20.58±5.80**	17.20±2.71	16.41±5.13	17.88±5.84	16.00±5.98	—	19.42±10.47
cleveland	26.36±8.85	27.23±6.73	25.48±9.17	22.99±9.32	25.67±6.03	—	**27.78±4.97**
breast	27.04±18.10	30.42±12.13	21.94±18.52	24.66±16.21	26.46±17.83	**35.69±14.07**	25.38±14.32
ionosphere	68.46±10.34	63.79±11.71	**85.07±11.89**	63.68±12.31	15.81±17.60	73.98±10.45	62.88±21.19
sonar	**63.83±15.79**	41.26±12.80	56.63±20.24	53.86±17.37	45.08±16.13	53.98±16.85	57.31±7.60
saheart	24.78±12.26	27.43±1.30	21.18±15.22	20.57±16.02	**29.41±12.44**	21.97±10.62	22.33±7.60
spectfheart	24.26±19.85	**28.22±22.34**	0±8.44	24.22±23.83	8.88±11.66	21.45±22.11	18.53±21.09
bupa	**23.33±14.87**	19.51±12.53	18.86±14.65	23.01±19.50	13.38±14.26	22.97±15.94	22.09±13.41
page	**76.42±4.21**	67.80±4.66	70.08±6.28	71.37±6.14	64.28±5.96	—	72.56±4.74
Avg.	**50.98±11.95**	44.10±9.07	46.32±11.85	46.86±13.35	37.52±12.43	—	44.59±11.95

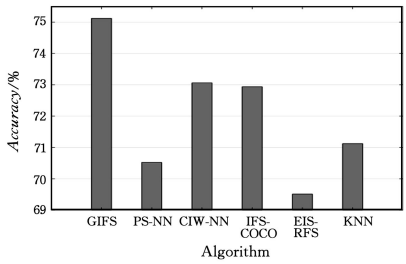

图 7 平均 Kappa 系数的实验结果

Fig. 7 Experiment results of average Kappa

GIFS 与其他 4 种对比算法在 Kappa 系数上的实验结果和威尔科克森符号秩检验结果如表 5 所列。GIFS 的 *P-value* 都小于常规的显著水平(0.05)，这说明在 Kappa 系数上，相对于其他比较算法，GIFS 表现更优秀。

表 5 Kappa 威尔科克森符号秩检验

Table 5 Signed-rank test for kappa

Alg	R^+	R^-	*P-value*
GIFS vs CIW-NN	97	23	0.0353
GIFS vs PS-NN	105	15	0.0088
GIFS vs IFS-CoCo	106	14	0.0067
GIFS vs EIS-RFS	114	6	0.0008
GIFS vs KNN	114	6	0.0008

6 讨论

6.1 EGA 的有效性讨论析

为了验证 EGA 实例选择和特征选择的效果，本文将在 GIFS 中分别使用 EGA 传统的遗传算法 GA 和 CHC[21] 进行对比实验，对实验得出的分类精度以及时间复杂度进行有效性讨论。其中，CHC 只采用它的交叉方式 HUX[16-18]。实验的平均分类精度对比如图 8 所示。

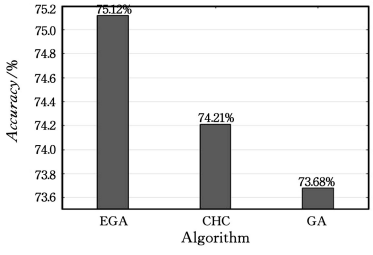

图 8 EGA 的有效性验证

Fig. 8 Validity verification of EGA

由实验结果可得，GIFS 使用 EGA 进行实例选择和特征选择时的分类精度高于使用 GA 和 CHC 进行实例选择和特征选择的分类精度。GA 在保持种群多样性上不如 EGA 和 CHC，导致它在进行实例选择和特征选择时的准确度和稳定性偏低。相较于 CHC，EGA 在保持种群多样性的同时，提高了实例选择和特征选择的稳定性，获得了最优的分类精度。

在主要的时间复杂度(交叉的时间复杂度)上，GA 为 $\frac{u}{2}O(z\cdot z)$，CHC 为 $\frac{u}{2}O(v\cdot v)$，EGA 为 $\frac{u}{4}O(z\cdot z)$。其中，u 为种群规模，v 为染色体长度，z 为随机选择的交叉基因片段的长度($z\leqslant v$)。EGA 一次交叉能产生 4 个子代，使其时间复杂度小于 GA 和 CHC，算法的效率更高。

综合以上分析，EGA 在保持种群多样性的同时，提高了算法的效率。相较于 CA 和 CHC，EGA 在进行实例选择和特征选择时，在准确度和效率上均具有优势。

6.2 实例与特征选择策略的有效性讨论

为了验证局部的噪声删除策略与特征选择策略的有效性，本文在 GIFS 中分别进行局部的噪声样本删除(G-IS)、特征选择(G-FS)，并将 G-IS 与 G-FS 相结合(GIFS)。将这 3 种训练集选择方式的分类精度结果进行对比，如图 9 所示。

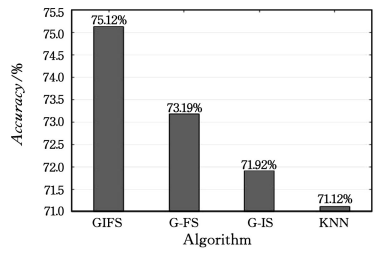

图 9 选择策略的有效性验证

Fig. 9 Validity verification of select strategy

由实验结果得，G-IS，G-FS，GIFS 的分类精度均高于 KNN；G-FS 的分类精度高于 G-IS，将 G-IS 与 G-FS 结合(GIFS)获得的分类精度最高。GIFS 相较于 G-IS 和 G-FS，在分类精度和分类稳定性上均具有优势。

结束语 本文提出了一种基于遗传实例特征选择的 KNN 分类训练集优化方法 GIFS，以提高 KNN 的分类效果；并且设计了一种基于最大汉明距离的高效遗传算法 EGA，其

先利用决策树来判定噪声样本存在的范围，然后使用 EGA 算法删除噪声样本并进行特征选择。新方法能在较小的种群规模和较少的遗传迭代次数下获得训练效果优良的训练集，有效地提高了 KNN 的分类精度。EGA 能高效、准确地进行小范围的特征选择和实例选择。本文使用最近邻策略选择验证集并将基于均方误差的分类精度惩罚函数作为 EGA 的目标函数，进一步提高了 EGA 进行实例选择和特征选择的稳定性和准确度。经验证，GIFS 的综合性能优于 PS-NN，CIW-NN，IFS-COCO，EIS-RFS，PHGA，KNN 等算法。后续将研究将 GIFS 扩展到优化其他分类器的训练集上，而不局限于 KNN 的训练集。

参 考 文 献

[1] COVER T, HART P. Nearest neighbor pattern classification [J]. IEEE Transactions on Information Theory, 1967, 13(1): 21-27.

[2] ZHAN Y, DAI S, MAO Q, et al. A Video Semantic Analysis Method Based on Kernel Discriminative Sparse Representation and Weighted KNN[J]. Computer Journal, 2018, 58(6): 1360-1372.

[3] WANG Y, YANG Y W. KNN Similarity Graph Algorithm Based on Heap and Neighborhood Coexistence[J]. Computer Science, 2018, 45(5): 196-200, 227.

[4] FENG G L, ZHOU W G. Spark-based Parallel Outlier Detection Algorithm of K-nearest Neighbor[J]. Computer Science, 2018, 45(S2): 349-352, 366.

[5] DENG Z, ZHU X, CHENG D, et al. Efficient kNN classification algorithm for big data[J]. Neurocomputing, 2016, 195(C): 143-148.

[6] ZHANG S, LI X, MING Z, et al. Efficient kNN Classification With Different Numbers of Nearest Neighbors[J]. IEEE Transactions on Neural Networks & Learning Systems, 2017, PP(99): 1-12.

[7] ZHANG S, LI X, ZONG M, et al. Learning k, for kNN Classification[J]. Acm Transactions on Intelligent Systems & Technology, 2017, 8(3): 43.

[8] GILPITA R, YAO X. Evolving edited k-nearest neighbor classifiers[J]. International Journal of Neural Systems, 2009, 18(6): 459-467.

[9] XIE H, LIANG D, ZHANG Z, et al. A Novel Pre-Classification Based kNN Algorithm[C]// IEEE International Conference on Data Mining Workshops. New Orleans, America, 2017: 1269-1275.

[10] LIU H, MOTODA H. Instance Selection and Construction for Data Mining[M]. Springer International, 2001: 448-454.

[11] WETTSCHERECK D, AHA D W, MOHRI T. A Review and Empirical Evaluation of Feature Weighting Methods for a Class of Lazy Learning Algorithms[J]. Artificial Intelligence Review, 1997, 11(1/2/3/4/5): 273-314.

[12] EIBEN A E, SCHOENAUER M. Evolutionary Computing[J]. Soft Computing, 1998, 82(1): 1-6.

[13] ACAMPORA G, TORTORA G, VITIELLO A. Applying SPEA2 to prototype selection for nearest neighbor classification[C]// IEEE International Conference on Systems, Man, and Cybernetics. Montreal, Canada, 2017: 003924-003929.

[14] SASIKALA S, APPAVU A B S, GEETHA S. A novel adaptive feature selector for supervised classification[J]. Information Processing Letters, 2017, 117: 25-34.

[15] KHIABANI A, SABBAGHI A. PHGA: Proposed hybrid genetic algorithm for feature selection in binary classification [C]// International Conference on Information and Knowledge Technology. Tehran, Iran, 2017: 147-154.

[16] DERRAC J, GARCÍA S, HERRERA F. Ifs-CoCo: Instance and Feature Selection Based on Cooperative Coevolution With Nearest Neighbor Rule[J]. Pattern Recognition, 2010, 43(6): 2082-2105.

[17] DERRAC J, TRIGUERO I, GARCIA S, et al. Integrating instance selection, instance weighting, and feature weighting for nearest neighbor classifiers by coevolutionary algorithms[J]. IEEE Transactions on Systems Man & Cybernetics Part B, 2012, 42(5): 1383-1397.

[18] DERRAC J, CORNELIS C, GARCÍA S, et al. Enhancing evolutionary instance selection algorithms by means of fuzzy rough set based feature selection[J]. Information Sciences, 2012, 186(1): 73-92.

[19] DERRAC J, VERBIEST N, GARCÍA S, et al. On the use of evolutionary feature selection for improving fuzzy rough set based prototype selection[J]. Soft Computing, 2013, 17(2): 223-238.

[20] BRAHIM A B, LIMAM M. A hybrid feature selection method based on instance learning and cooperative subset search ☆[J]. Pattern Recognition Letters, 2016, 69: 28-34.

[21] ESHELMAN L J. The CHC Adaptive Search Algorithm: How to Have Safe Search When Engaging in Nontraditional Genetic Recombination[J]. Foundations of Genetic Algorithms, 1991, 1: 265-283.

[22] GOLDBERG D E, SASTRY K. A Practical Schema Theorem for Genetic Algorithm Design and Tuning[C]// Genetic and Evolutionary Computation Conference. San Francisco, America, 2001: 328-335.

[23] HUANG Y Y, DONG M G, JING C. Genetic instance selection algorithm for k-nearest neighbor classifier[J]. Journal of Computer Applications, 2018, 38(11): 3112-3118.

DONG Ming-gang, born in 1977, Ph.D, professor, is a member of China Computer Federation. His main research interests include intelligent computing and its applications, machine learning.

JING Chao, born in 1983, Ph.D, associate professor, is a member of China Computer Federation. His main research interests are intelligent computing, deep reinforcement learning and optimization approaches.