计算机系统应用 ISSN 1003-3254, CODEN CSAOBN
Computer Systems & Applications,2022,31(3):275−281 [doi: 10.15888/j.cnki.csa.008373]

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

基于集成特征选择的 FSSD 算法①

张　崟, 何振峰

(福州大学 数学与计算机科学学院, 福州 350108)
通信作者: 张　崟, E-mail: 1143047418@qq.com

摘　要: FSSD (fast and efficient subgroup set discovery) 是一种子群发现算法, 旨在短时间内提供多样性模式集, 然而此算法为了减少运行时间, 选择域数量少的特征子集, 当特征子集与目标类不相关或者弱相关时, 模式集质量下降. 针对这个问题, 提出一种基于集成特征选择的 FSSD 算法, 它在预处理阶段使用基于 ReliefF (Relief-F) 和方差分析的集成特征选择来获得多样性和相关性强的特征子集, 再使用 FSSD 算法返回高质量模式集. 在 UCI 数据集、全国健康和营养调查报告 (NHANES) 数据集上的实验结果表明, 改进后的 FSSD 算法提高了模式集质量, 归纳出更有趣的知识. 在 NHANES 数据集上, 进一步分析模式集的特征有效性和阳性预测值.

关键词: 子群发现; 集成特征选择; ReliefF; 方差分析

引用格式: 张崟,何振峰.基于集成特征选择的 FSSD 算法.计算机系统应用,2022,31(3):275–281. http://www.c-s-a.org.cn/1003-3254/8373.html

FSSD Algorithm Based on Ensemble Feature Selection

ZHANG Yin, HE Zhen-Feng

(College of Mathematics and Computer Science, Fuzhou University, Fuzhou 350108, China)

Abstract: Fast and efficient subgroup set discovery (FSSD) is a subgroup discovery algorithm that aims to provide a diverse set of patterns in a short period of time. However, in order to reduce the running time, this algorithm selects a feature subset with a small number of domains. When the feature subset is irrelevant or weakly related to the target class, the quality of the pattern set decreases. To solve this problem, this study proposes a FSSD algorithm based on ensemble feature selection. In the preprocessing stage, it uses ensemble feature selection based on ReliefF (Relief-F) and analysis of variance to obtain feature subset with diversity and strong correlation, and then uses FSSD algorithm to return high-quality pattern set. The experimental results on the UCI datasets and the National Health and Nutrition Examination Survey (NHANES) dataset show that the improved FSSD algorithm improves the quality of the pattern set, thereby summarizing more interesting knowledge. Furthermore, the feature validity and positive predictive value of the pattern set are further analyzed on the NHANES dataset.

Key words: subgroup discovery; ensemble feature selection; ReliefF; analysis of variance

子群发现是一种数据挖掘技术, 它基于给定目标特征, 挖掘不同特征间的有趣关联[1]. 提取的模式通常用规则表示, 规则定义为:

$$R: Cond \rightarrow Target_{\text{value}} \quad (1)$$

$Cond$ 是由一组属性键值对组成, 称为规则前件; $Target_{\text{value}}$是目标特征值, 称为规则后件.

尽管子群发现算法在过去得到了足够的发展, 但仍存在一些不足. 传统子群发现算法通常是将连续型数值数据直接离散化, 导致了精度损失和次优结果; 并且是基于局部模式挖掘, 导致模式集缺乏多样性[2]. 对

① 基金项目: 福建省自然科学基金 (2018J01794)
收稿时间: 2021-05-19; 修改时间: 2021-06-14; 采用时间: 2021-06-30; csa 在线出版时间: 2022-01-24

于这些问题, Millot 等人[3] 提出 OSMIND (optimal subgroup discovery in purely numerical data) 算法, 利用闭合间隔模式避免连续型数值数据离散化, 但是忽略了模式集挖掘. Bosc 等人[4] 提出 MCTS4DM (Monte Carlo tree search for pattern mining) 算法, 在使用闭合间隔模式的基础上, 增加相似性度量函数, 对 MCTS (Monte Carlo tree search) 输出的候选模式集进行后处理, 从而获得多样性的模式集, 但是需要消耗大量内存来存储候选模式集. Belfodil 等人[5] 提出 FSSD 算法, 使用闭合间隔模式和子群集质量度量函数, 后者用来评估模式集的多样性, 同时在每次迭代过程中只存储最优模式, 避免了消耗大量内存. 但是 FSSD 算法为了减少运行时间, 选择特征域数量少的特征子集, 此特征选择方法没有考虑数据集中的监督信息, 属于无监督特征选择, 当选择的特征子集与目标类不相关或弱相关时, 模式集质量下降, 因此研究如何选择 FSSD 算法的特征子集具有重要意义.

特征选择是根据某些特征选择标准从原始特征集中获取特征子集的过程[6]. 现有特征选择主要用在分类、聚类上, 用在子群发现上的研究较少. 在子群发现上的特征选择只找到文献 [7], 它提出基于相关性约束的过滤式特征选择, 将特征-值对的覆盖关系作为相关性约束, 当特征的所有特征-值对满足相关性约束时, 才被定义为不相关并且去除, 在最坏的情况下, 每个特征都存在一个特征-值对不满足相关性约束, 此时无法去除任何特征, 同时单一的特征选择方法生成的特征子集缺少多样性和稳定性[8].

分类、聚类特征选择与子群发现特征选择的原则有些不同: (1) 前者是将类和类区分开, 后者是将目标类和非目标类区分开, 在多类情况下, 两者差别比较明显; (2) 前者需要去除不相关和冗余特征, 而后者只去除不相关特征, 不去除冗余特征, 这点在医学领域尤其明显, 在文献 [9] 中, 使用子群发现挖掘初次使用合成阿片类药物后, 出现不良后果的患者特征, 挖掘结果认为有慢性疼痛病史的患者, 会增加药物上瘾风险, 与合成阿片类药物处方指南相吻合. 由于慢性病与疼痛病存在强相关, 如果考虑去除冗余特征, 子群发现挖掘结果会变成有慢性病史或者疼痛病史的患者, 会增加药物上瘾风险.

为了解决 FSSD 算法的特征子集选择问题, 本文引入基于 ReliefF 和方差分析的集成特征选择算法, 获得具有多样性、稳定性以及与目标类相关性强的特征子集. 第 2 节介绍 FSSD 算法, 第 3 节介绍改进的 FSSD 算法, 第 4 节对改进的 FSSD 算法进行实验并且对实验结果做分析, 第 5 节对所做的工作进行总结和未来进一步的研究方向.

1 FSSD 算法简介

FSSD 算法旨在使用少量内存和短时间内提供多样性的模式集[5]. FSSD 是基于穷举搜索的子群发现算法, 穷举搜索的运行时间与模式数量成正比, 当搜索空间很大时, 运行时间变长, 所以在文献 [5] 中, 先选择特征子集, 再使用 FSSD 算法来提取模式集.

1.1 特征子集选择

在运行 FSSD 算法前, 数据集的特征先按照名义型、数值型排序, 再按照特征域数量从小到大排序, 然后根据用户给定的特征子集数量选择特征子集, 此特征选择方法是为了尽可能选择特征域数量少的特征子集, 随着特征子集的域数量减少, 模式数量就减少, 运行时间就缩短. 然而此方法忽视了数据集中的监督信息, 没有考虑特征与目标类的相关性, 特征与目标类的相关性就完全取决于数据集分布情况, 当特征子集与目标类相关性不佳时, 模式集质量下降, 当特征子集与目标类相关性强时, 模式集质量上升, 模式集质量随着特征子集与目标类的相关性而变化, 让模式集质量充满不确定性.

1.2 FSSD 算法

在 FSSD 算法中, 子群 s 是子群集合 $\mathbb{S}=ext[D]=\{ext(d)|d\in D\}$ 的任何子集, 模式 $d\in D$ 提供了对子群的描述. 模式结构是三元组 $(G,(D,\sqsubseteq),\delta)$ 形式, G 是样本集, D 是模式集, $\sqsubseteq$ 是偏序关系, 模式集 D 的模式通过偏序关系 $\sqsubseteq$ 从最普通限制到最严格限制进行排序, 即 $c\sqsubseteq d\Leftrightarrow ext(d)\subseteq ext(c)$, 表示模式 d 覆盖的子群是模式 c 覆盖的子群的子集. δ 将样本映射到最严格模式上, 例如: $\delta(g)$ 是样本 g 的最严格模式, 称为闭合间隔模式, 只要修改模式 $\delta(g)$, 至少会丢弃一个样本. 与模式结构相关的两个映射: (1) $ext: D\to\wp(G), ext(d)=\{g\in G|\forall d\in D, d\sqsubseteq\delta(g)\}$, $\wp(G)$ 是样本集 G 的幂集, $ext(d)$ 表示模式 d 覆盖的子群; (2) $int:\wp(G)\to D$, $int(E)=\prod_{g\in E}\delta(g)$, 其中 $E\subseteq G$, $int(E)$ 表示覆盖样本子集 E 的闭合间隔模式.

给定子群 s 和子群集 $S\subseteq\mathbb{S}$, 计算子群 s 给 S 带来的质量增益可以表示为:

$$WRAcc_S(s)=\alpha\cdot(1-\alpha)\cdot[TPR(s,G_S,G)-FPR(s,G_S,G)] \tag{2}$$

$$TPR(s,G_S,G)=tpr(G_S,G)\cdot tpr(s,G_S) \tag{3}$$

$$FPR(s,G_S,G)=fpr(G_S,G)\cdot fpr(s,G_S) \tag{4}$$

$$tpr(s,G_X)=\frac{|s\cap G^+\cap G_X|}{|G^+\cap G_X|} \tag{5}$$

$$fpr(s,G_X)=\frac{|s\cap G^-\cap G_X|}{|G^-\cap G_X|} \tag{6}$$

$\alpha=|G^+|/|G|$是正样本比例, $G_S=G\setminus\cup S$是未被子群集S覆盖的样本集, $TPR(s,G_S,G)$和$FPR(s,G_S,G)$分别表示在G中未覆盖样本G_S的子群s的真阳性率和假阳性率. G^+,G^-分别表示样本集G中的正、负样本, G_X是样本集G中的非空子集, $tpr(s,G_X)$和$fpr(s,G_X)$分别表示在G_X中子群s的真阳性率和假阳性率.

与式 (2) 对应的严格乐观估计:

$$WRAcc_S^{oe}(s)=\alpha\cdot(1-\alpha)[TPR(s,G_S,G)] \tag{7}$$

FSSD 算法框架如算法 1.

算法 1. FSSD(G,A_n,k)

输入: 数据集(G,A_n), 子群数量k
输出: 子群集S

1) $S=\phi$, $G_S=G$ // G_S是当前样本集
2) while $|S|<k$
3) 在模式结构 $(G_S,(D,\sqsubseteq),\delta)$ 中寻找子群s^*使得增益最大$WRAcc_S(ext(int(s^*)))$
4) if $WRAcc_S(ext(int(s^*)))\leqslant 0$ then break
5) $S=S\cup\{ext(int(s^*))\}$, $G_S=G_S\setminus s^*$
6) end while

对于算法 1, 在第 1) 行中初始化子群集S为空和当前样本集G_S为G; 第 2) 行和第 4) 行是控制迭代次数, 当$|S|=k$或者最大增益小于 0 时, 算法终止并返回子群集S; 在每次迭代过程中, 第 3) 行是在未覆盖样本集G_S的模式结构 $(G_S,(D,\sqsubseteq),\delta)$ 中选择最大化增益$WRAcc_S(ext(int(s^*)))$的子群s^*, 由于$s^*\subseteq G_S$但不一定$s^*\in\mathbb{S}$, 所以通过映射关系使得$ext(int(s^*))\in\mathbb{S}$, 在寻找子群$s^*$过程中, 使用严格乐观估计式 (7) 进行剪枝, 从而缩小搜索空间; 在第 5) 行中更新子群集S和当前样本集G_S, $ext(int(s^*))$添加到子群集S并且将s^*从当前样本集G_S中去除.

2 基于集成特征选择的 FSSD 算法

当 FSSD 算法的特征子集与目标类相关性不强时, 模式集质量下降, 而单一特征选择方法获得的特征子集缺乏多样性和稳定性. 为此, 本文提出基于集成特征选择的 FSSD 算法, 简称 FSSD++算法. FSSD++算法通过集成 ReliefF 和方差分析方法获得多样性、稳定性以及与目标类相关性强的特征子集, 再使用 FSSD 算法返回高质量子群集. 在设计集成特征选择时, 需要确定以下几个方面: (1) 集成方法; (2) 确定特征选择器的返回形式和使用的特征选择器; (3) 组合方式.

2.1 集成方法

集成特征选择的集成方法主要有两种: 同构集成和异构集成, 如果使用相同的基本特征选择器, 称为同构集成, 否则称为异构集成. 在同构集成特征选择中, 使用相同的特征选择器和不同的数据子集, 可以使用自举法抽样得到数据子集; 在异构集成特征选择中, 使用不同的特征选择器和相同的数据, 如图 1 所示. 前者用于大规模数据集, 通过在并行节点中处理数据来缩短计算时间; 后者确保稳定、鲁棒的特征选择[10], 由于本文的实验数据集规模不大, 所以采用异构集成方法.

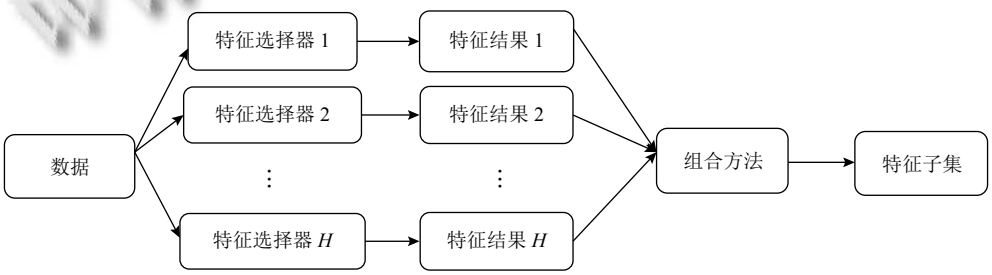

图 1 异构集成特征选择

2.2 特征选择器

特征选择器的返回形式分为特征子集和特征排名, 特征子集是先根据预先定义的搜索策略生成特征子集, 再使用最优原则对特征子集进行评估, 最终获得局部

最优特征子集; 特征排名是根据特征相关性或者重要性返回所有特征排名, 再使用阈值确定最终特征子集[10]. 返回特征子集的特征选择器需要搜索所有特征子集, 计算量大, 为了避免此问题, 本文选择返回特征排名的特征选择器. 此外, 为了缩短运行时间和有效利用监督信息, 所选的特征选择器还应该是基于过滤式和监督的特征选择器.

本文选择 ReliefF[6] 和方差分析[11] 作为基本特征选择器, 它们既是返回特征排名, 又是基于过滤式、监督的特征选择器. 同时它们都是单独评估每个特征与目标类的相关性, 并且不去除冗余特征[6,11], 这与子群发现特征选择原则相符合. 下面具体介绍 ReliefF 和方差分析.

ReliefF 是根据特征区分不同类别样本的程度, 对特征进行加权, 它为每个特征分配一个相关权重来表示特征与目标类的相关性. 假设在 n 个样本中随机选择l个样本, 每个特征f_i的计算公式:

$$ReliefF\_score(f_i)=\frac{1}{l}\sum_{j=1}^{l}W(j,i) \tag{8}$$

$$W(j,i)=-d_i(H(x_j,y_j),x_j)+\sum_{y\neq y_j}\left[\frac{p(y)}{1-p(y_j)}d_i(H(x_j,y),x_j)\right] \tag{9}$$

$$d_i(H(x_j,y_j),x_j)=\frac{1}{|H(x_j,y_j)|}\sum_{r=1}^{|H(x_j,y_j)|}|x_{ji}-x_{ri}| \tag{10}$$

$$d_i(H(x_j,y),x_j)=\frac{1}{|H(x_j,y)|}\sum_{r=1}^{|H(x_j,y)|}|x_{ji}-x_{ri}| \tag{11}$$

c是类别数量, y_j是样本x_j的类别, $W(j,i)$是样本x_j与同类样本和不同类样本在特征f_i上的距离之和, $d_i(S,x_j)$是样本x_j与样本集S在特征f_i上的距离, $H(x_j,t)$是样本x_j的近邻t类样本集, $|H(x_j,t)|$是$H(x_j,t)$样本集的数量, $p(y)$是y类样本的比例.

方差分析是研究每个特征对目标类是否产生显著影响, 它使用 F 统计的值作为每个特征得分, 得分越高, 类间特征均值差异越大, 说明特征变化引起了目标类变动. 每个特征f_i的计算公式:

$$F\_score(f_i)=\frac{\sum_{j=1}^{c}n_j(\mu_{ij}-\mu_i)^2/(c-1)}{\sum_{j=1}^{c}\sum_{k=1}^{n_j}(x_{ij}^k-\mu_{ij})^2/(n-c)} \tag{12}$$

c是类别数量, n是样本总数量, n_j是j类别的样本数量, μ_{ij}是j类样本中特征f_i的平均值, μ_i是全部样本中特征f_i的平均值, x_{ij}^k是j类样本中第k个样本在特征f_i上的值.

2.3 组合方式

参考文献 [12] 中的组合方式, 使用 min 函数获取每个特征在所有特征选择器的特征排名中的最小排名, 排名越小, 特征越重要, 重复此过程获得所有特征的最小排名, 再将所有特征的最小排名按进行二次排序: 先按排名从小到大排序, 再按特征域数量从小到大排序, 最后得到所有特征最终排名. 假设 ReliefF 返回所有特征排名$Rank_1$, 方差分析返回所有特征排名$Rank_2$, 首先获取特征a_j在所有特征选择器的排名$Rank_{*,j}=\{Rank_{1,j}, Rank_{2,j}\}$, 然后使用 min 函数获取特征$a_j$在所有特征选择器的最小排名, 即特征$a_j$的最终排名$min_a_j=\min(Rank_{*,j})$, 重复此步骤直到获得所有特征的最小排名$A'=\{min_a_j|j=1,\cdots,J\}$, 再将排名先按从小到大排序, 再按特征域数量从小到大排序得到所有特征最终排名A''.

2.4 算法

FSSD++算法框架如算法 2.

算法 2. FSSD++算法

输入: 数据集(G,A), 子群数量k, 特征子集大小n
输出: 子群集S

1) *for* h=1 *to* H //H 个特征选择器
2) $Rank_h$=第 h 个特征选择器获得特征集 A 的排名
3) *end for*
4) *for* j=1 *to* J //组合所有特征排名, J 是特征集 A 的数量
5) $Rank_{*,j}=\{Rank_{h,j}|h=1,\cdots,H\}$
//获取特征$a_j,a_j\in A$在所有特征选择器中最小的特征排名
6) $min\_a_j=min(Rank_{*,j})$
7) $A'=A'\cup min\_a_j$
8) *end for*
9) A''=对A'进行二次排序, 先按排名从小到大排序, 再按特征域数量从小到大排序
10)A''_n=从A''中取前 n 个特征
11) FSSD(G,tA''_n,k)

对于算法 2, 在第 1)–3) 行中获取每个特征选择器返回的特征排名; 在第 4)–8) 行中使用 min 函数获取在所有特征选择器中每个特征的最小排名, 直到获得全部特征的最小排名A'; 在第 9) 行中对组合排名A'先按排名升序, 再按特征域数量升序获得最终排名A''; 在第 10) 行中根据用户给定数量n, 从A''中获取前n个特征子集A''_n.

3 实验与分析

3.1 实验数据集

实验选择 7 个 UCI 数据集和 1 个 NHANES (national health and nutrition examination survey) 数据集, 如表 1 所示. abalone、adult、autos、credit、dermatology、mushrooms 和 sonar 来自 UCI 数据集, 1999_2004_Audiometry 来自 NHANES 数据集.

NHANES 是一项连续的横断面健康访问和调查研究, 旨在评估美国人民的健康和功能状况. 该研究每两年周期收集一次数据, 本文重点分析 1999–2004 年参加听力检测和听力问卷调查的 20–69 岁人群的数据, 研究在自我报告听力损失人群中, 不同特征之间的关联. 根据测试者编码 (SEQN) 连接听力检测数据 (audiometry examination data)、听力问卷数据 (audiometry questionnaire data)、糖尿病数据 (diabetes questionnaire data)、高血压数据 (blood pressure questionnaire data) 和人口统计数据 (demographics data) 生成 5 417 条数据. 样本排除标准如下: (1) 变量数据缺失, (2) 在第 1 次 1 kHz 和第 2 次 1 kHz 频率下, 听力阈值的差值超过 10 dB, (3) 自我报告听力损失程度为耳聋, (4) 血糖水平介于正常和糖尿病之间的数据, 根据上述标准排除 280 条数据, 最终纳入 5 137 条数据, 被命名为 1999_2004_Audiometry. 同时作者将自我报告听力损失程度 (good、little of trouble hearing、a lot of trouble hearing) 重新分类: good 表示未自我报告听力损失, little 、a lot of trouble hearing 表示自我报告听力损失. 1999_2004_Audiometry 数据集的特征有性别、年龄、种族、国家、学历、24 小时内有没有听音乐、糖尿病、高血压、在 0.5、1、2、3、4、6、8 kHz 下左右耳的听力阈值, 目标变量是自我报告听力损失, 它的取值范围是{Yes, No}, Yes 表示自我报告听力损失, No 表示未自我报告听力损失.

3.2 实验参数

FSSD++算法实验选择 FSSD 算法实验中特征子集数量与特征总数量不同的 7 个 UCI 数据集, 并在此基础上增加 1999_2004_Audiometry 数据集. 特征子集数量与特征总数量相同的数据集, 无法突显出特征选择的意义, 所以在此不做实验对比. 在 FSSD++算法对比实验中, FSSD++算法的参数有最大规则数量k =5、特征子集数量 n、最大搜索深度$Depth_{\max}$ = 8. 除了 1999_2004_Audiometry 数据集的特征子集数量n = 6 由作者给定外, 其他数据集的特征子集数量与文献 [5] 中 FSSD 算法实验给定特征子集数量一致, FSSD 算法实验的最大规则数量k =5和最大搜索深度$Depth_{\max}$ = 8, 所以 FSSD++算法实验也选择此值. MCTS4DM 算法的参数除了最大规则数量k =5、特征子集数量 n 外, 还有最大迭代次数nb_{iter}=5 000, 最大冗余$maxRedundancy$ =0.25. 每个数据集指定特征子集数量 n 后, 表示为数据集-特征子集数量, 例如: abalone-5, 具体如表 2 所示. 表 2 中“–”表示 WRAcc 质量小于 0.

表 1 UCI 的 7 个数据集以及 NHANES 的 1 个数据集

数据集	行数	目标变量值	正样本/总样本	特征(名义/数值)数量
abalone	4 177	M	0.37	(0/8)
adult	32 561	⩾ 50 K	0.24	(8/6)
autos	195	3	0.12	(11/14)
credit	666	+	0.45	(9/6)
dermatology	358	3	0.20	(0/34)
mushrooms	8 124	p	0.48	(22/0)
sonar	208	R	0.47	(0/60)
1999_2004_Audiometry	5 137	Yes	0.21	(7/17)

表 2 FSSD++算法与 MCTS4DM、FSSD 算法的 WRAcc 对比

数据集-属性数量	MCTS4DM	FSSD	FSSD++
abalone-5	0.045	0.068	**0.069**
adult-10	—	0.109	**0.115**
autos-10	—	0.071	**0.091**
credit-10	—	0.187	**0.188**
dermatology-10	0.004	0.158	0.158
mushrooms-10	—	0.235	**0.239**
sonar-10	0.081	0.165	**0.191**
1999_2004_Audiometry-6	—	0.035	**0.061**

3.3 实验分析

表 2 是 FSSD++算法与 FSSD 算法、MCTS4DM 算法的对比实验结果, 使用 WRAcc 作为评估指标. 对比 FSSD++和 FSSD 算法, 在大部分数据集中 FSSD++提供更优的 WRAcc 质量, 除了数据集 dermatology-10 提供相等 WRAcc 质量, 这个结果表明使用集成特征选择的 FSSD++算法能够提高 WRAcc 质量, 验证了 FSSD++算法的有效性. 同时 FSSD++和 FSSD 算法都优于 MCTS4DM 算法.

表 3 是在自我报告听力损失中, FSSD 和 FSSD++算法的特征子集和阳性预测值对比. 表 4 对表 3 中特征子集进行了具体描述, 从表 3 可以看出, FSSD 和 FSSD++算法的特征子集都有 DIQ010 (糖尿病)、BPQ020 (高血压) 和 AUQ030 (24 小时内有没有听音乐), FSSD 算法是根据特征域数量来获取特征, 而 FSSD++算法是根据特征与目标变量的相关性来获取特征, FSSD++算法认为自我报告听力损失与 3 kHz 、4 kHz、6 kHz、糖尿病、高血压、24 小时内有没有听音乐有关. 文献 [13] 验证了 4 kHz 是自我报告听力损失最重要的个体频率, 但是目前还没有文献表明自我报告听力损失与 3 kHz 、6 kHz 相关, FSSD++算法为研究自我报告听力损失的相关知识提供了新思路. 文献 [14] 和文献 [15] 表明糖尿病和高血压与听力损失有关, 听力损失与自我报告听力损失存在中等一致性 [16], 所以糖尿病和高血压与自我报告听力损失有关, 进一步说明 FSSD++算法挖掘自我报告听力损失人群特征的有效性.

表 3 在自我报告听力损失中 FSSD 和 FSSD++算法的特征子集和阳性预测值对比

	FSSD	FSSD++
特征子集	['BPQ020', 'DMDCITZN', 'AUQ030', 'DIQ010', 'RIAGENDR', 'DMDBORN']	['BPQ020', 'AUXU3KL', 'DIQ010', 'AUXU4KL', 'AUXU6KL', 'AUQ030']
n(WE4FA >25 or WEHFA >35)/n(class = Yes)	0.578	**0.669**

表 4 FSSD 和 FSSD++算法的特征子集对比

FSSD		FSSD++	
特征	描述	特征	描述
BPQ020	高血压	BPQ020	高血压
DMDCITZN	国家	AUXU3KL	在3 kHz下左耳听力阈值
AUQ030	24小时内有没有听音乐	DIQ010	糖尿病
DIQ010	糖尿病	AUXU4KL	在4 kHz下左耳听力阈值
RIAGENDR	性别	AUXU6KL	在6 kHz下左耳听力阈值
DMDBORN	国家	AUQ030	24小时内有没有听音乐

在文献 [13] 中听力损失定义: 在 0.5、1、2、4 kHz 下较差耳朵的纯音平均听力阈值>25 dB (WE4FA > 25 dB) 或者在 4、6、8 kHz 下较差耳朵的纯音平均听力阈值>35 dB (WEHFA >35 dB). 表 3 中 n(WE4FA > 25 or WEHFA >35)/n(class= Yes) 表示模式集覆盖人群中 WE4FA >25 dB 或者 WEHFA >35 dB 的数量占自我报告听力损失总数量的比, 即对于 WE4FA >25 dB 或者 WEHFA >35 dB, 自我报告听力损失的阳性预测值, n(class =Yes) 是自我报告听力损失总数量. 与 FSSD 算法相比, FSSD++算法挖掘自我报告听力损失的阳性预测值更高, WE4FA >25 dB 或者 WEHFA > 35 dB 的人数也更多, 因为 FSSD++算法的特征子集包含 4 kHz, 4 kHz 听力下降是导致自我报告听力损失的重要因素, 所以 FSSD++算法挖掘自我报告听力损失人群在 4 kHz 的听力损失比较高, WE4FA >25 dB 或者 WEHFA >35 dB 的人数就增多, 自我报告听力损失的阳性预测值也就变高.

4 总结与展望

针对 FSSD 算法选择特征域数量较少的特征子集, 导致模式集质量下降的问题, 提出一种基于集成特征选择的 FSSD 算法. 该算法在预处理阶段, 根据 min 函数组合 ReliefF 和方差分析的输出结果, 对组合结果先按排名升序, 再按特征域数量升序, 最后获得前 n 个特征. 此特征子集作为 FSSD 算法的输入参数, 从而获取更优的模式集. 在 UCI 数据集和 NHANES 数据集中进行实验, 对比 FSSD++、FSSD 和 MCTS4DM 算法的 WRAcc; 对比 FSSD 和 FSSD++算法的自我报告听力损失的特征子集和阳性预测值. 实验结果表明, 与 MCTS4DM 和 FSSD 算法相比, FSSD++算法归纳的模式集 WRAcc 值更优; 与 FSSD 算法相比, FSSD++算法挖掘自我报告听力损失人群的特征有效性和阳性预测值更高. 在未来工作中将侧重研究如何自适应选择特征数量.

参考文献

1 Helal S. Subgroup discovery algorithms: A survey and

empirical evaluation. Journal of Computer Science and Technology, 2016, 31(3): 561–576. [doi: 10.1007/s11390-016-1647-1]

2 Meeng M, Knobbe A. For real: A thorough look at numeric attributes in subgroup discovery. Data Mining and Knowledge Discovery, 2021, 35(1): 158–212. [doi: 10.1007/s10618-020-00703-x]

3 Millot A, Cazabet R, Boulicaut JF. Optimal subgroup discovery in purely numerical data. Pacific-Asia Conference on Knowledge Discovery and Data Mining. Singapore: Springer, 2020. 112–124.

4 Bosc G, Boulicaut JF, Raïssi C, *et al*. Anytime discovery of a diverse set of patterns with Monte Carlo tree search. Data Mining and Knowledge Discovery, 2018, 32(3): 604–650. [doi: 10.1007/s10618-017-0547-5]

5 Belfodil A, Belfodil A, Bendimerad A, *et al*. FSSD—A fast and efficient algorithm for subgroup set discovery. 2019 IEEE International Conference on Data Science and Advanced Analytics (DSAA). Washington, DC: IEEE, 2019. 91–99.

6 Li JD, Cheng KW, Wang SH, *et al*. Feature selection: A data perspective. ACM Computing Surveys, 2017, 50(6): 1–45.

7 Lavrač N, Gamberger D. Relevancy in constraint-based subgroup discovery. In: Boulicaut JF, De Raedt L, Mannila H, eds. Constraint-based Mining and Inductive Databases. Berlin: Springer, 2006. 243–266.

8 Bolón-Canedo V, Alonso-Betanzos A. Ensembles for feature selection: A review and future trends. Information Fusion, 2019, 52: 1–12. [doi: 10.1016/j.inffus.2018.11.008]

9 Nagpal C, Wei D, Vinzamuri B, *et al*. Interpretable subgroup discovery in treatment effect estimation with application to opioid prescribing guidelines. Proceedings of the ACM Conference on Health, Inference, and Learning. New York: ACM, 2020. 19–29.

10 Seijo-Pardo B, Porto-Díaz I, Bolón-Canedo V, *et al*. Ensemble feature selection: Homogeneous and heterogeneous approaches. Knowledge-Based Systems, 2017, 118: 124–139. [doi: 10.1016/j.knosys.2016.11.017]

11 Bommert A, Sun XD, Bischl B, *et al*. Benchmark for filter methods for feature selection in high-dimensional classification data. Computational Statistics & Data Analysis, 2020, 143: 106839.

12 Willett P. Combination of similarity rankings using data fusion. Journal of Chemical Information and Modeling, 2013, 53(1): 1–10. [doi: 10.1021/ci300547g]

13 Swanepoel DW, Eikelboom RH, Hunter ML, *et al*. Self-reported hearing loss in baby boomers from the Busselton Healthy Ageing Study: Audiometric correspondence and predictive value. Journal of the American Academy of Audiology, 2013, 24(6): 514–521. [doi: 10.3766/jaaa.24.6.7]

14 Mishra A, Poorey VK. Clinical and audiometric assessment of hearing loss in diabetes mellitus. Indian Journal of Otolaryngology and Head & Neck Surgery, 2019, 71(2): 1490–1494.

15 Marchiori LLDM, Filho EDAR, Matsuo T. Hypertension as a factor associated with hearing loss. Brazilian Journal of Otorhinolaryngology, 2006, 72(4): 533–540. [doi: 10.1016/S1808-8694(15)31001-6]

16 Gomez MI, Hwang SA, Sobotova L, *et al*. A comparison of self-reported hearing loss and audiometry in a cohort of New York farmers. Journal of Speech, Language and Hearing Research, 2001, 44(6): 1201–1208. [doi: 10.1044/1092-4388(2001/093)]