



## 面向置换检验的冗余对比模式过滤算法

吴 军, 欧阳艾嘉, 张 琳

(遵义师范学院 信息工程学院, 贵州 遵义 563000)

**摘 要:** 置换检验方法在进行对比模式挖掘时, 返回结果中存在许多冗余对比模式。利用 Charm 方法挖掘样本集合中的对比模式, 提出基于固定属性置换的 FSPRP 和 FEPRP 算法, 依次为不同长度的对比模式构建零分布, 从而过滤冗余对比模式。FSPRP 算法通过生成一定数量的置换样本集合构建零分布, FEPRP 算法则通过计算每个模式的对比性度量值分布合并建立零分布。实验结果表明, FSPRP 和 FEPRP 算法相较于比较约束法能够过滤较多数量的冗余对比模式, 并且 FEPRP 算法生成的零分布更接近精确零分布。

**关键词:** 数据挖掘; 对比模式挖掘; 置换检验; 冗余对比模式过滤; 固定属性置换

开放科学(资源服务)标志码(OSID):



中文引用格式: 吴军, 欧阳艾嘉, 张琳. 面向置换检验的冗余对比模式过滤算法[J]. 计算机工程, 2022, 48(1): 75-84.

英文引用格式: WU J, OUYANG A J, ZHANG L. Redundant contrast pattern filtering algorithm for permutation testing[J]. Computer Engineering, 2022, 48(1): 75-84.

## Redundant Contrast Pattern Filtering Algorithm for Permutation Testing

WU Jun, OUYANG Aijia, ZHANG Lin

(School of Information Engineering, Zunyi Normal University, Zunyi, Guizhou 563000, China)

**[Abstract]** The existing contrast pattern mining algorithms based on permutation testing suffer from redundant contrast patterns. To address the problem, two algorithms are proposed, which use the fixed attribute permutation procedure (including FSPRP and FEPRP) for filtering redundant contrast patterns. The Charm method is used to mine the contrast patterns in the sample set, and to construct the null distributions for the contrast patterns of different lengths. The FSPRP algorithm constructs the null distributions by generating a number of permuted data sets, whereas the FEPRP algorithm constructs the null distributions by calculating the contrast measure distribution of each pattern. The experimental results show that the FSPRP algorithm and the FEPRP algorithm can successfully filter out a certain number of redundant contrast patterns than the comparison constraint methods. Additionally, the null distributions generated by the FEPRP algorithm are closer to the exact null distributions.

**[Key words]** data mining; contrast pattern mining; permutation testing; redundant contrast pattern filtering; fixed attribute permutation

DOI: 10.19678/j.issn.1000-3428.0060307

### 0 概述

分析不同类型数据样本之间的差异性对于分类、特征选择、突变点检测等研究具有重要意义。在数据挖掘领域中, 对比模式发现任务是为了找到在不同类别标签的数据样本集合中出现频率差异显著的模式<sup>[1]</sup>。目前, 对比模式被广泛应用在医学(如探索不同外科手术之间的联系<sup>[2]</sup>)、生物学(如发现蛋白质中的磷酸化基序<sup>[3]</sup>)、音乐学(如找到不同类型曲目的旋律差别<sup>[4]</sup>)等领域。

传统的对比模式挖掘算法根据数据类型不同分为面向序列数据和面向非序列数据的对比模式挖掘算法<sup>[5-6]</sup>; 根据策略的不同分为基于阈值约束和 TOP-K 差异约束的算法<sup>[7-8]</sup>。这些算法的不同之处主要体现在候选模式生成方式、对比性度量、剪枝方式、搜索方式和数据结构方面。

由于传统的对比模式挖掘算法将注意力放在如何快速有效地找到满足自定义约束的模式上, 而算法返回报告的结果中存在一定数量的假阳性模式<sup>[9]</sup>。假阳性模式是指模式单纯偶然地满足对比模

**基金项目:** 国家自然科学基金(61662090); 贵州省教育厅青年科技人才成长项目(黔教合 KY 字[2017]250); 贵州省教育厅工程研究中心项目(黔教合 KY 字[2016]018)。

**作者简介:** 吴 军(1990—), 男, 讲师, 主研方向为数据挖掘、深度学习; 欧阳艾嘉, 教授、博士; 张 琳, 副教授、硕士。

**收稿日期:** 2020-12-16 **修回日期:** 2021-02-14 **E-mail:** wujun.myway@gmail.com

式挖掘算法定义的对比性度量约束,而没有真实地表现不同类别数据样本集合的差异特征。由于假阳性模式提供了错误的信息,因此有必要对挖掘到的模式进行质量评估。统计显著性检验是一种常用的模式质量评估方法<sup>[10]</sup>。该方法首先构建任务相关零假设,随后计算相应的统计显著性度量值决定是否拒绝零假设。如果多个零假设被同时检验,则称为多重假设检验。在对比模式发现任务中,常用的统计显著性检验方法有直接计算方法<sup>[11]</sup>和置换检验方法<sup>[12]</sup>。无论在序列数据还是非序列数据中,置换检验方法的效率都优于直接计算方法<sup>[13-14]</sup>。

在直接计算方法和置换检验方法返回的结果中存在许多的冗余对比模式<sup>[15]</sup>。冗余对比模式是受子模式统计显著性的影响而呈现出统计显著性的超模式。因此,冗余对比模式实际上没有提供新的信息,且携带的额外信息还会对后续任务产生一定的干扰。除闭频繁模式以外<sup>[13]</sup>,置换检验方法中常用的冗余对比模式过滤方法还有比较约束法<sup>[14]</sup>。该方法基于的假设是超模式的统计显著性只有大于子模式的统计显著性,才会提供额外有用的信息。实际上,统计显著的超模式和子模式之间并不一定具备这样的关系,然而比较约束法能够过滤掉一定数量的冗余对比模式和许多非冗余对比模式。

本文设计 FSPRP 和 FEPRP 算法用于过滤置换检验方法中的冗余对比模式。通过在置换过程中固定子模式属性列的方式,打破子模式和超模式在置换样本集合中的联系,从而计算不受子模式统计显著性影响的超模式  $p$  值。FSPRP 算法依据标准置换检验原理,通过生成一定次数的置换样本集合构建零分布,而 FEPRP 算法基于精确置换检验原理,通过计算每个模式的对比性度量值分布构建零分布。

## 1 问题定义

### 1.1 对比模式挖掘

令  $D$  为一个数据样本集合,其中每条样本  $s$  均能被属性集  $A = \{A_1, A_2, \dots, A_{|A|}\}$  表示。假设  $a$  是  $A_j$  的一个值,则  $A_j = a$  被称作一个项  $t$ 。单个或多个项构成的集合,即  $\{t_1, t_2, \dots, t_k\}$ , 被称为模式,用  $x$  表示,同时该模式的长度被定义为项的个数,用  $k$  表示。对比模式给定模式  $x_1$  和  $x_2$ , 如果  $x_1$  中所有的项均被  $x_2$  包含,则  $x_1$  被称作  $x_2$  的子模式,  $x_2$  被称作  $x_1$  的超模式。

如果一条样本  $s$  的  $A_j$  属性的值是  $a$ , 那么称样本  $s$  包含  $A_j = a$  这个项。如果一条样本  $s$  含有模式  $x$  所有的项,则称样本  $s$  包含模式  $x$ , 表示为  $x \subseteq s$ 。  $x$  在  $D$  中的支持度被定义为  $D$  中包含  $x$  的样本数量, 即  $\text{sup}(x, D) = |\{s | s \in D \wedge x \subseteq s\}|$ 。如果  $x$  的支持度超过一个自定义阈

值  $\theta_{\text{sup}}$ , 则  $x$  被认为是频繁模式。近年来, 研究人员提出许多频繁模式挖掘算法, 例如, FP-growth 算法<sup>[16]</sup>、Charm 算法<sup>[17]</sup>等。

如果  $A_{\text{cla}}$  是一个类别属性, 那么  $A_{\text{cla}}$  的每种值都被称为一个类别标签。在包含类别属性的  $D$  中, 如果一些模式在不同类别标签的样本中支持度呈现较大差异, 这样的模式就被称为对比模式。该差异可以由不同的对比性度量进行量化, 即如果一个模式的对比性度量值超过一个自定义的阈值  $\theta_{\text{dis}}$ , 那么该模式就是对比模式。为便于描述后续方法, 假定  $A_{\text{cla}}$  仅包含两种值  $c_1$  和  $c_2$ , 从而  $D$  根据该属性划分为  $D_1$  和  $D_2$ 。传统的对比模式挖掘算法通常包含两个步骤: 首先, 挖掘样本集合  $D_1$  中的频繁模式; 其次, 在这些频繁模式中找到所有满足阈值  $\theta_{\text{dis}}$  的对比模式。

### 1.2 冗余对比模式

如果一个模式本身是非统计显著的模式, 但其被错误地认定为统计显著的模式并用于后续决策, 这样的模式被认为假阳性模式。传统的对比模式挖掘算法报告结果中存在大量的假阳性模式, 这些假阳性模式提供的错误信息会干扰后续决策。本文采用统计显著性检验方法过滤这些假阳性模式。在统计显著性检验中, 模式的统计显著性由  $p$  值度量, 其定义是在零假设为真的前提下, 找到一个至少同样极端的模式的概率。  $p$  值越小则表明统计显著性越强。

常用的统计显著性检验方法分为直接计算方法<sup>[11]</sup>和置换检验方法<sup>[12]</sup>。直接计算方法将模式服从的分布当作零分布, 直接计算得到模式的  $p$  值; 置换检验方法通过置换类别标签生成新的置换样本集合, 再从置换样本集合中构建零分布计算模式的  $p$  值。然而, 在直接计算方法和置换检验方法返回的结果中均存在一定数量的冗余对比模式。冗余对比模式是给定超模式  $x_2$  和子模式  $x_1$ , 如果  $x_2$  的统计显著性来源于  $x_1$ , 那么  $x_2$  就是冗余对比模式。冗余对比模式没有提供新的信息, 保留它们会造成后续不必要的计算开销, 甚至还会干扰后续决策。

在直接计算方法中, 文献[18]提出使用模式集合挖掘策略过滤冗余对比模式, 规定同一个集合中的模式必须满足相异性约束才能被保留。文献[19]设计了3种冗余度量, 并提出一个启发式搜索的方法找到非冗余的对比模式组。文献[20]在局部约束的基础上额外定义基于子模式的全局约束, 要求模式必须同时满足2个约束才是非冗余对比模式。文献[15]设计 CP-tree 和 PDP-tree 两种树形结构用于计算和比较超模式和子模式的统计显著性, 从而过滤不满足约束阈值的冗余对比模式。

无论在序列数据还是非序列数据的对比模式发现任务中, 置换检验方法的检验效力均强于直接计算方法<sup>[13-14]</sup>。在置换检验方法中, 保留闭频繁模式是最基本的冗余对比模式过滤方法<sup>[13]</sup>。闭频繁模式是如果 $x_1$ 不存在一个超模式 $x_2$ , 使得 $x_2$ 和 $x_1$ 具有相同的支持度, 那么 $x_1$ 就是闭频繁模式, 但这种方法只能过滤掉少量冗余对比模式。在置换检验方法的基础上, 冗余对比模式通常还采用比较约束法进一步过滤<sup>[14]</sup>。比较约束法将子模式和超模式的 $p$ 值进行直接比较, 如果超模式的 $p$ 值大于子模式的 $p$ 值, 则超模式被认定为冗余对比模式。实际上, 统计显著的超模式的 $p$ 值和子模式的 $p$ 值并不一定具备这种关系, 因此, 比较约束法在过滤冗余对比模式的同时, 也会过滤许多非冗余的统计显著的超模式。

### 1.3 固定属性置换过程

面向对比模式发现的置换检验方法无论使用精确零分布还是近似零分布, 都遵循标准随机置换过程。该过程通过随机交换类别标签得到置换样本集合。标准随机置换过程如图1所示。例如, 假定 $D$ 包含8条样本, 每条样本由6个普通属性和1个类别属性构成, 类别属性的值由 $c_1$ 和 $c_2$ 表示, 如图1(a)所示。根据样本的编号, 样本集合随机生成一个置换序列:  $s_7, s_6, s_2, s_8, s_4, s_1, s_3, s_5$ 。随后根据该置换序列指定样本的类别标签, 将样本 $s_1$ 的类别标签 $c_1$ 指派给样本 $s_7$ , 并将其放在原来 $s_1$ 的位置, 样本 $s_2$ 的类别标签 $c_1$ 指派给样本 $s_6$ , 并将其放到原来 $s_2$ 的位置, 以此类推, 就得到了标准随机置换过程的置换样本集合, 如图1(b)所示。

	$A_1$	$A_2$	$A_3$	$A_4$	$A_5$	$A_6$	$A_c$
$s_1$	$t_{11}$	$t_{21}$	$t_{31}$	$t_{42}$	$t_{51}$	$t_{63}$	$c_1$
$s_2$	$t_{11}$	$t_{23}$	$t_{31}$	$t_{41}$	$t_{52}$	$t_{62}$	$c_1$
$s_3$	$t_{12}$	$t_{21}$	$t_{32}$	$t_{42}$	$t_{53}$	$t_{62}$	$c_1$
$s_4$	$t_{11}$	$t_{22}$	$t_{31}$	$t_{41}$	$t_{54}$	$t_{61}$	$c_1$
$s_5$	$t_{11}$	$t_{23}$	$t_{32}$	$t_{42}$	$t_{53}$	$t_{61}$	$c_1$
$s_6$	$t_{12}$	$t_{21}$	$t_{32}$	$t_{41}$	$t_{51}$	$t_{63}$	$c_2$
$s_7$	$t_{11}$	$t_{21}$	$t_{31}$	$t_{42}$	$t_{52}$	$t_{62}$	$c_2$
$s_8$	$t_{12}$	$t_{22}$	$t_{32}$	$t_{42}$	$t_{52}$	$t_{61}$	$c_2$

(a) 原始样本集合

	$A_1$	$A_2$	$A_3$	$A_4$	$A_5$	$A_6$	$A_c$
$s'_1$	$t_{11}$	$t_{21}$	$t_{31}$	$t_{42}$	$t_{52}$	$t_{62}$	$c_1$
$s'_2$	$t_{12}$	$t_{21}$	$t_{32}$	$t_{41}$	$t_{51}$	$t_{63}$	$c_1$
$s'_3$	$t_{11}$	$t_{23}$	$t_{31}$	$t_{41}$	$t_{52}$	$t_{62}$	$c_1$
$s'_4$	$t_{12}$	$t_{22}$	$t_{32}$	$t_{42}$	$t_{52}$	$t_{61}$	$c_1$
$s'_5$	$t_{11}$	$t_{22}$	$t_{31}$	$t_{41}$	$t_{54}$	$t_{61}$	$c_1$
$s'_6$	$t_{11}$	$t_{21}$	$t_{31}$	$t_{42}$	$t_{51}$	$t_{63}$	$c_2$
$s'_7$	$t_{12}$	$t_{21}$	$t_{32}$	$t_{42}$	$t_{53}$	$t_{62}$	$c_2$
$s'_8$	$t_{11}$	$t_{23}$	$t_{32}$	$t_{42}$	$t_{53}$	$t_{61}$	$c_2$

(b) 置换样本集合

图1 标准随机置换过程

Fig.1 The standard random permutation process

从图1可以看出, 标准随机置换过程并不会打破子模式和超模式对应样本的联系, 即置换样本集合中包含超模式和子模式的样本数量等于原始样本集合中包含它们的数量, 例如, 给定子模式 $x_1=\{t_{11}\}$ 和超模式 $x_2=\{t_{11}, t_{31}\}$ , 原始样本集合中有4条样本 $\{s_1, s_2, s_4, s_7\}$ 包含 $x_1$ 和 $x_2$ , 经过置换后, 置换样本集合中仍有4条样本 $\{s'_1, s'_3, s'_5, s'_6\}$ 包含 $x_1$ 和 $x_2$ 。因此, 在标准随机置换计算得到的结果中, 如果 $x_1$ 和 $x_2$ 均是统计

显著的, 那么 $x_2$ 的统计显著性很有可能是由 $x_1$ 的统计显著性导致, 即 $x_2$ 很大概率是冗余对比模式。

一种可行的过滤冗余对比模式原理是在置换过程中打破超模式与子模式对应样本的联系。本文提出一个固定属性的置换过程。在该置换过程中, 如果一个子模式是统计显著的, 则固定该子模式所有项对应的属性列, 仅置换余下的属性列。例如, 假定原始样本集合与图1(a)中的原始样本集合相同, 且子模式 $x_1=\{t_{11}\}$ 是统计显著的。同样地, 令随机生成的置换序列为 $s_7, s_6, s_2, s_8, s_4, s_1, s_3, s_5$ 。固定属性置换过程首先将 $t_{11}$ 对应的属性列 $A_1$ 固定, 余下的属性 $A_2 \sim A_6$ 列根据置换序列置换。固定属性置换过程如图2所示。固定属性置换过程将样本 $s_1$ 的类别标签 $c_1$ 指派给不包含 $A_1$ 属性的样本 $s_7$ , 并将其放在样本 $s_1$ 原来的位置; 将样本 $s_2$ 的类别标签 $c_1$ 指派给不包含 $A_1$ 属性的样本 $s_6$ , 并将其放在样本 $s_2$ 原来的位置, 以此类推, 得到了置换样本集合。

从图2(b)可以看出, 置换样本集合中仅有2条样本 $\{s'_1, s'_5\}$ 包含 $x_1$ 和 $x_2$ , 说明固定属性置换过程打破了子模式和超模式对应样本的联系。因此, 在固定属性置换过程生成的置换样本集合包含超模式和子模式的样本数量不一定等于原始样本集合中包含超模式和子模式的样本数量。在该置换过程中, 每条样本非固定的项称为可置换样本, 固定的项称为固定样本。对于 $s_2$ 而言,  $\{t_{23}, t_{31}, t_{41}, t_{52}, t_{62}\}$ 是可置换样本,  $\{t_{11}\}$ 是固定样本。基于上述置换过程, 本文提出两个新的面向置换检验的冗余对比模式过滤算法: FSPRP算法和FEPRP算法。

	$A_1$	$A_2$	$A_3$	$A_4$	$A_5$	$A_6$	$A_c$
$s_1$	$t_{11}$	$t_{21}$	$t_{31}$	$t_{42}$	$t_{51}$	$t_{63}$	$c_1$
$s_2$	$t_{11}$	$t_{23}$	$t_{31}$	$t_{41}$	$t_{52}$	$t_{62}$	$c_1$
$s_3$	$t_{12}$	$t_{21}$	$t_{32}$	$t_{42}$	$t_{53}$	$t_{62}$	$c_1$
$s_4$	$t_{11}$	$t_{22}$	$t_{31}$	$t_{41}$	$t_{54}$	$t_{61}$	$c_1$
$s_5$	$t_{11}$	$t_{23}$	$t_{32}$	$t_{42}$	$t_{53}$	$t_{61}$	$c_1$
$s_6$	$t_{12}$	$t_{21}$	$t_{32}$	$t_{41}$	$t_{51}$	$t_{63}$	$c_2$
$s_7$	$t_{11}$	$t_{21}$	$t_{31}$	$t_{42}$	$t_{52}$	$t_{62}$	$c_2$
$s_8$	$t_{12}$	$t_{22}$	$t_{32}$	$t_{42}$	$t_{52}$	$t_{61}$	$c_2$

(a) 原始样本集合

	$A_1$	$A_2$	$A_3$	$A_4$	$A_5$	$A_6$	$A_c$
$s'_1$	$t_{11}$	$t_{21}$	$t_{31}$	$t_{42}$	$t_{52}$	$t_{62}$	$c_1$
$s'_2$	$t_{11}$	$t_{21}$	$t_{32}$	$t_{41}$	$t_{51}$	$t_{63}$	$c_1$
$s'_3$	$t_{12}$	$t_{23}$	$t_{31}$	$t_{41}$	$t_{52}$	$t_{62}$	$c_1$
$s'_4$	$t_{11}$	$t_{22}$	$t_{32}$	$t_{42}$	$t_{52}$	$t_{61}$	$c_1$
$s'_5$	$t_{11}$	$t_{22}$	$t_{31}$	$t_{41}$	$t_{54}$	$t_{61}$	$c_1$
$s'_6$	$t_{12}$	$t_{21}$	$t_{31}$	$t_{42}$	$t_{51}$	$t_{63}$	$c_2$
$s'_7$	$t_{11}$	$t_{21}$	$t_{32}$	$t_{42}$	$t_{53}$	$t_{62}$	$c_2$
$s'_8$	$t_{12}$	$t_{23}$	$t_{32}$	$t_{42}$	$t_{53}$	$t_{61}$	$c_2$

(b) 置换样本集合

图2 固定属性置换过程

Fig.2 The fixed attribute permutation process

## 2 FSPRP算法

FSPRP算法首先使用Charm方法挖掘 $D_1$ 中的闭频繁模式<sup>[17]</sup>, 随后计算每个闭频繁模式的对比性度量 $G_{\text{Growth Rate}}$ 值<sup>[7]</sup>, 如式(1)所示:

$$G_{\text{Growth Rate}}(x, D) = \frac{s_{\text{sup}}(x, D_1)}{s_{\text{sup}}(x, D_2)} \quad (1)$$

在这些模式中, 满足阈值 $\theta_{\text{dis}}$ 的模式被认定为候选



对比模式。FSPRP算法根据固定属性置换过程,由短到长为各长度的模式生成一定数量的置换样本集合(长度为1的模式除外),并进行对比模式挖掘;利用这些对比模式的对比性度量值构建各个长度相应的零分布 $N_k$ 。每个候选对比模式被赋予一个 $p$ 值度量其统计显著性,其定义是发现一个至少与该对比模式对比性度量值相同的对比模式概率。最后,将某个候选对比模式 $x$ 的对比性度量值放置到其长度对应的零分布 $N_k$ 中就能计算出其 $p$ 值,如式(2)所示:

$$p_{\text{value}}(x, N_k) = \frac{|\{n | G_{\text{GrowthRate}}(x, D) \leq n \wedge n \in N_k\}|}{|N_k|} \quad (2)$$

上述零分布 $N_k$ 是通过固定属性置换过程建立的,因此由 $N_k$ 计算得到 $k$ 长度候选对比模式的 $p$ 值去除了统计显著子模式的影响。最后,FSPRP算法采用错误发现率( $F_{\text{FDR}}$ )度量约束 $k$ 长度候选对比模式假阳性结果的数量<sup>[21]</sup>,如式(3)所示:

$$F_{\text{FDR}}(X_k, \alpha) = \left\{ x | p_{\text{value}}(x) \leq \frac{\alpha \times o_{\text{order}}(x)}{|X_k|} \wedge x \in X_k \right\} \quad (3)$$

其中: $\alpha$ 为统计显著水平; $X_k$ 为 $k$ 长度候选对比模式的集合, $o_{\text{order}}(x)$ 为根据 $x$ 的 $p$ 值在 $X_k$ 中从小到大的排序位置。详细的FSPRP算法步骤见算法1。

**算法1** FSPRP( $D, \theta_{\text{sup}}, \theta_{\text{dis}}, h, z, \alpha$ )

输入 数据样本集合 $D=\{D_1, D_2\}$ ;支持度阈值 $\theta_{\text{sup}}$ ;对比性度量阈值 $\theta_{\text{dis}}$ ,固定个数 $h$ ,置换次数 $z$ ,统计显著水平 $\alpha$

输出 统计显著的对比模式集合 $E$

1.  $X \leftarrow \text{cp\_mine}(D_1, \theta_{\text{sup}}, \theta_{\text{dis}})$

2.  $X_1, X_2, \dots, X_{\text{maxl}(X)} \leftarrow \text{group}(X)$

3. for  $i = 1$  to  $z$  do

4.  $D' \leftarrow \text{sr\_permute}(D)$

5.  $X' \leftarrow \text{cp\_mine}(D', \theta_{\text{sup}}, \theta_{\text{dis}})$

6.  $N_1 \leftarrow N_1 \cup \text{mv\_extract}(X', 1)$

7. end for

8.  $E \leftarrow \text{EUsig\_patterns}(X_1, N_1, \alpha)$

9. for  $k = 2$  to  $\text{maxl}(X)$  do

10. for  $i = 1$  to  $z$  do

11.  $D' \leftarrow \text{af\_permute}(D, E, h)$

12.  $X' \leftarrow \text{cp\_mine}(D', \theta_{\text{sup}}, \theta_{\text{dis}})$

13.  $N_k \leftarrow N_k \cup \text{mv\_extract}(X', k)$

14. end for

15.  $E \leftarrow \text{EUsig\_patterns}(X_k, N_k, \alpha)$

16. end for

FSPRP算法步骤主要分为3步:

1)使用 $\text{cp\_mine}()$ 方法挖掘 $D_1$ 中的候选对比模式。随后,使用 $\text{group}()$ 方法根据模式的长度进行分组(第1、2步)。

2)运用标准随机置换过程 $\text{sr\_permute}()$ 对 $D$ 执行 $z$ 次置换,并挖掘这些置换样本集合 $D'_1$ 中的对比模式。利用 $\text{mv\_extract}()$ 方法提取1长度模式的对比性度量

值,并用其构建1长度模式的零分布 $N_1$ ,由于它们不存在统计显著的子模式(第3~7步),因此对长度为1的模式运用标准随机置换过程。通过 $N_1$ 计算1长度候选对比模式的 $p$ 值,并用错误发现率约束找到统计显著的1长度对比模式,即 $\text{sig\_patterns}()$ 。最后,将所有统计显著的1长度对比模式放入保存统计显著的对比模式集合 $E$ 中(第8步)。

3)对于长度大于1的模式,从短到长依次运用固定属性置换过程 $\text{af\_permute}()$ 执行 $z$ 次置换。为提升效率,置换过程每次同时固定 $h$ 个统计显著的子模式对应的属性列,随后,对置换样本集合进行挖掘,并计算其中模式的对比性度量值用于构建 $k$ 长度模式的零分布 $N_k$ 。从 $N_k$ 中计算得到 $k$ 长度候选对比模式的 $p$ 值,并用错误发现率约束得到统计显著的 $k$ 长度对比模式。这些 $k$ 长度对比模式 $p$ 值的计算均考虑了统计显著的子模式的影响。最后,所有统计显著的 $k$ 长度对比模式放入集合 $E$ 中。 $E$ 的最终结果是过滤了冗余对比模式的统计显著的对比模式(第9~16步)。

### 3 FEPRP 算法

FSPRP算法具有以下3个缺点:1)候选对比模式的 $p$ 值可能为0, $p$ 值为0是一个非常极端的近似值,它表示该对比模式的统计显著性无穷大;2)多次运行FSPRP算法得到的统计显著对比模式可能不同;3)由于FSPRP算法需要使用固定属性置换过程生成一定次数的置换样本集合,随后还需要对样本集合进行对比模式挖掘,因此FSPRP算法计算开销较大。

FEPRP算法采用标准置换检验中通过生成一定置换次数的置换样本集合构建零分布的策略,因此产生上述缺点。由于每次运行FSPRP算法生成置换样本集合不同,从而导致构建的零分布也不相同。文献[14]利用精确置换检验解决标准置换检验中的问题。精确置换检验的基本原理是独立计算每个对比模式的对比性度量值分布,然后合并得到相应的精确零分布。根据精确置换检验计算生成零分布的原理,本文提出使用固定属性置换过程的FEPRP算法。

FEPRP算法计算得到对比模式 $x$ 的对比性度量值分布,首先要清楚 $x$ 在固定属性置换过程生成的置换样本集合中的数量分布, $x$ 的数量分布分为 $x$ 不存在统计显著的子模式和 $x$ 存在一个或多个统计显著的子模式。在 $x$ 不存在统计显著的子模式没有固定的属性列, $x$ 在固定属性置换过程生成的置换样本集合中的数量分布等价于 $x$ 在标准随机置换过程生成的置换样本集合中的数量分布。设 $x$ 在置换样本集合 $D'_1$ 中的数量为 $v$ ,其分布情况如表1所示。

表1 在标准随机置换过程中对比模式 $x$ 的  
样本集合数量分布

Table 1 The number of sample set distribution of contrast  
mode  $x$  in the standard random permutation process

样本集合	$D'_1$	$D'_2$
包含 $x$ 的样本数量	$v$	$s_{\sup}(x, D) - v$
不包含 $x$ 的样本数量	$ D_1  - v$	$ D_2  - s_{\sup}(x, D) + v$

在对比模式 $x$ 存在一个或多个统计显著的子模式中, 设对比模式 $x$ 某一个统计显著的子模式为 $x_u$ ,  $x^*$ 表示 $x$ 除去 $x_u$ 剩下的项, 即 $x - x_u$ 。在固定属性置换过程中,  $x_u$ 对应的属性列不随置换序列而改变, 从而当一条包含 $x^*$ 的可置换样本被置换到包含 $x_u$ 的固定样本的位置时, 才能在置换样本集合中形成一条包含 $x$ 的样本, 即包含 $x^*$ 的可置换样本与包含 $x_u$ 的固定样本相结合得到一条包含 $x$ 的样本。 $x$ 在

表2 在固定属性置换过程中 $x^*$ 的样本集合数量分布

Table 2 The number of sample set distribution of  $x^*$  in the fixed attribute permutation process

样本集合	$D_1^*$	$D_2^*$	$D^-$
包含 $x^*$ 的样本数量	$q$	$r$	$s_{\sup}(x^*, D) - q - r$
不包含 $x^*$ 的样本数量	$s_{\sup}(x_u, D_1) - q$	$s_{\sup}(x_u, D_2) - r$	$ D  - s_{\sup}(x_u, D) - s_{\sup}(x^*, D) + q + r$

$x$ 的对比性度量值分布由对比性度量值及其在置换样本集合中出现的次数构成。对比性度量值可由 $x$ 的每种数量分布计算得到, 其对应的次数可以通过模拟该数量分布对应的置换样本集合生成得到。

在 $x$ 不存在统计显著的子模式中,  $x$ 在固定属性置换过程生成的置换样本集合中的数量分布等价于标准随机置换过程生成的置换样本集合中的数量分布。因此,  $v$ 的最小值 $L(v)$ 为 $\max\{\theta_{\sup}, |D_1| + s_{\sup}(x, D_1) - |D|\}$ , 最大值 $U(v)$ 为 $\min\{s_{\sup}(x, D), |D_1|\}$ 。由于每个 $v$ 对应一个对比性度量值, 因此可以通过从小到大递增 $v$ 得到所有的对比性度量值。至于每个对比性度量值对应的次数, 可通过模拟 $D'_1$ 中恰好有 $v$ 条样本包含 $x$ 的置换样本集合生成计算得到, 先从包含 $x$ 的样本 $s_{\sup}(x, D)$ 中选取 $v$ 条放入 $D'_1$ 中, 然后从不包含 $x$ 的样本 $|D| - s_{\sup}(x, D)$ 中选取 $|D_1| - v$ 条放入 $D'_1$ 中, 最后将余下的样本放入 $D'_2$ 中。因此, 该置换样本集合的数量如式(4)所示:

$$n_{\text{set}_1}^p(x, v) = \binom{s_{\sup}(x, D)}{v} \binom{|D| - s_{\sup}(x, D)}{|D_1| - v} \quad (4)$$

在 $x$ 存在一个或多个统计显著的子模式中,  $x$ 的数量分布由 $x^*$ 的数量分布决定, 即每个 $q$ 和 $r$ 决定了 $x$ 在固定属性置换过程生成的置换样本集合 $D'_1$ 和 $D'_2$ 中的支持度。 $q$ 的最小值 $L(q)$ 为 $\max\{\theta_{\sup}, s_{\sup}(x_u, D_1) + s_{\sup}(x^*, D) - |D|\}$ , 最大值 $U(q)$ 为 $\min\{s_{\sup}(x^*, D),$

置换样本集合中的数量分布取决于有多少条包含 $x^*$ 的可置换样本被置换到包含 $x_u$ 的固定样本的位置。因此,  $x$ 的数量分布实际上是由 $x^*$ 的数量分布决定。

$x^*$ 可以被置换到3个部分: 与 $D_1$ 中包含 $x_u$ 的固定样本结合(表示为 $D_1^*$ ), 与 $D_2$ 中包含 $x_u$ 的固定样本结合(表示为 $D_2^*$ ), 与 $D_1$ 和 $D_2$ 中不包含 $x_u$ 的固定样本结合(表示为 $D^-$ )。当且仅当包含 $x^*$ 的可置换样本被置换到 $D_1^*$ 和 $D_2^*$ 中时, 才能生成相应的包含 $x$ 的样本。设 $D_1^*$ 中 $x^*$ 的数量为 $q$ ,  $D_2^*$ 中 $x^*$ 的数量为 $r$ , 那么 $x^*$ 在固定属性置换过程生成的置换样本集合中的数量分布如表2所示。从表2可以看出,  $q$ 和 $r$ 决定着 $x^*$ 在置换样本集合中的分布情况, 对于每个 $q$ 会存在多个与之匹配的 $r$ , 反之亦然。当 $q$ 和 $r$ 确定后,  $x$ 在置换样本集合中的数量分布也随之确定。

$s_{\sup}(x_u, D_1)\}$ 。 $r$ 的最小值 $L(r)$ 为 $\max\{0, s_{\sup}(x^*, D) - q - |D| + s_{\sup}(x_u, D_2)\}$ ,  $r$ 的最大值 $U(r)$ 为 $\min\{s_{\sup}(x^*, D) - q, s_{\sup}(x_u, D_2)\}$ 。由于 $q$ 和 $r$ 对应 $x$ 在固定属性置换过程生成的置换样本集合 $D'_1$ 和 $D'_2$ 中的支持度, 从而通过一对 $q$ 和 $r$ 计算得出 $x$ 的一个对比性度量值。

每个对比性度量值对应的次数也可以通过模拟 $D'_1$ 和 $D'_2$ 中分别有 $q$ 和 $r$ 条样本包含 $x$ 的置换样本集合计算得到, 先从包含 $x^*$ 可置换样本中取出 $q$ 条放入 $D_1^*$ 中, 再从不包含 $x^*$ 可置换样本中取出 $s_{\sup}(x_u, D_1) - q$ 条放入 $D_1^*$ 中。所有可能的 $D_1^*$ 的组成数量如式(5)~式(7)所示:

$$s_{c1}(x^*, q) = \binom{s_{\sup}(x^*, D)}{q} \quad (5)$$

$$s_{c1}(x^*, q) = \binom{|D| - s_{\sup}(x^*, D)}{s_{\sup}(x_u, D_1) - q} \quad (6)$$

$$n_{d1}(x^*, q) = s_{c1}(x^*, q) s_{c2}(x^*, q) \quad (7)$$

FEPRP算法从余下的包含 $x^*$ 可置换样本中取出 $r$ 条放入 $D_2^*$ 中, 再从余下的不包含 $x^*$ 可置换样本中取出 $s_{\sup}(x_u, D_2) - r$ 条放入 $D_2^*$ 中。所有可能 $D_2^*$ 的组成数量如式(8)~式(10)所示:

$$s_{c3}(x^*, q, r) = \binom{s_{\sup}(x^*, D) - q}{r} \quad (8)$$

$$s_{c4}(x^*, q, r) = \binom{|D| - s_{\sup}(x^*, D) - s_{\sup}(x_u, D_1) + q}{s_{\sup}(x_u, D_2) - r} \quad (9)$$

$$n_{d2}(x^*, q, r) = s_{c3}(x^*, q, r) s_{c4}(x^*, q, r) \quad (10)$$

最后将余下的可置换样本放入 $D^-$ 中,且 $D^-$ 可能的组成数量为1,模拟了固定属性置换过程中 $x$ 在 $D_1'$ 和 $D_2'$ 中支持度分别为 $q$ 和 $r$ 的置换样本集合的生成,且该置换样本集合的数量如式(11)所示:

$$n_{\text{set}_2}^p(x^*, q, r) = n_{d_1}(x^*, q) n_{d_2}(x^*, q, r) \quad (11)$$

由于上述对比性度量值分布的计算是基于固定属性置换过程得到的,因此其打破了统计显著的子模式对超模式的影响。在计算每个超模式 $x$ 的对比性度量分布时,如果考虑其所有统计显著的子模式 $x_u$ ,则能够计算得到固定属性置换过程对应的精确零分布。但在实际应用中,一个较长的模式可能包含许多统计显著的子模式,考虑到算法的实用性,FEPRP算法只去除了统计显著性最强的 $m$ 个子模式的影响,详细的FEPRP算法步骤见算法2。

#### 算法2 FEPRP( $D, \theta_{\text{sup}}, \theta_{\text{dis}}, m, \alpha$ )

输入 数据样本集合 $D=\{D_1, D_2\}$ ;支持度阈值 $\theta_{\text{sup}}$ ;对比性度量阈值 $\theta_{\text{dis}}$ ,子模式个数 $m$ ,统计显著水平 $\alpha$

输出 统计显著的对比模式集合 $E$

1.  $X_1, X_2, \dots, X_{\text{maxl}(X)} \leftarrow \text{group}(\text{cp\_mine}(D_1, \theta_{\text{sup}}, \theta_{\text{dis}}))$

2.  $X'_1, X'_2, \dots, X'_{\text{maxl}(X)} \leftarrow \text{group}(\text{cp\_mine}(D_2, \theta_{\text{sup}}, \theta_{\text{dis}}))$

3. for  $k = 1$  to  $\text{maxl}(X)$  do

4. for each  $x$  in  $X'_k$

5. if  $x$  not exists  $x_u$

6.  $B \leftarrow \text{brcd\_construct}(x, \theta_{\text{sup}}, \theta_{\text{dis}})$

7. else

8.  $B \leftarrow \text{facd\_construct}(x, E, m, \theta_{\text{sup}}, \theta_{\text{dis}})$

9.  $N_k \leftarrow N_k \cup B$

10. end for

11.  $E \leftarrow E \cup \text{sig\_patterns}(X_k, N_k, \alpha)$

12. end for

1)使用 $\text{cp\_mine}()$ 方法挖掘 $D_1$ 中的候选对比模式和 $D$ 中所有可能在置换样本集合中出现的对比模式,随后使用 $\text{group}()$ 方法根据模式的长度进行各自分组(第1~2步)。

2)从长度为1的模式开始,由短到长依次为不同长度的模式构建零分布 $N_k$ 。具体而言,对于每个长度为 $k$ 的模式 $x$ ,如果其不存在统计显著的子模式 $x_u$ ,使用 $\text{brcd\_construct}()$ 方法计算其对比性度量值分布 $B$ ,即用式(1)计算每个的对比性度量值,用式(4)计算每个对比性度量值相应的置换样本集合的数量(第5~6步);如果 $x$ 存在统计显著的子模式 $x_u$ ,使用 $\text{facd\_construct}()$ 方法计算其对比性度量值分布 $B$ ,即先找到 $x$ 统计显著性最强的 $m$ 个子模式,随后将每个子模式视作 $x_u$ ,用式(1)计算每对 $q$ 和 $r$ 分布的对比性度量值,用式(11)计算每个对比性度量值对应的置换样本集合的数量(第7~8步)。计算得到所有 $k$ 长度模式的对比性度量值分布后,将其合并就能得到 $k$ 长度模式的零分布 $N_k$ (第9步)。

3)利用 $\text{sig\_patterns}()$ 方法计算每个 $k$ 长度模式的 $p$ 值,并使用错误发现率约束得到统计显著的 $k$ 长

度对比模式,再将这些模式放到集合 $E$ 中。迭代完成后, $E$ 的最终结果即是过滤了冗余对比模式的统计显著的对比模式(第11步)。

## 4 实验

为验证FSPRP和FEPRP算法过滤冗余对比模式的效率,在4个不同类型的数据样本集合上进行实验。实验对比算法为DA算法<sup>[1]</sup>、SP算法<sup>[13]</sup>、SPRF算法<sup>[12]</sup>和IEPCSP算法<sup>[14]</sup>,其中DA和SP算法分别使用直接计算方法和标准置换检验方法找到统计显著的对比模式,且这两个算法都未考虑冗余对比模式问题。SPRF和IEPCSP算法分别使用标准置换检验和精确置换检验挖掘统计显著的对比模式,此外还采用比较约束方法过滤结果中的冗余对比模式。在实验中,如无特殊说明FEPRP算法仅考虑8个统计显著性最强的子模式,FSPRP算法的置换次数为1 000。为了进行统一比较,每个对比算法都采用Charm方法挖掘候选对比模式,且使用错误发现率作为多重假设检验约束。所有实验均在一台配置为2.40 GHz CPU和12 GB内存的设备上运行。

### 4.1 实验数据

实验采用4个不同类型的真实数据样本集合:即german、hypo、gamma和adult。这4个数据样本集合均源自于UCI machine learning repository数据库<sup>[22]</sup>,其中,german是银行客户信用特征样本集;hypo是甲状腺疾病患者特征样本集;gamma是 $\gamma$ -粒子成像特征样本集;adult是成人收入特征样本集。实验样本集合信息如表3所示,其中对连续的属性值进行了离散化。

表3 实验样本集合信息

Table 3 Information of experimental sample set

样本集合	$ A $	$ D $
german	21	1 000
hypo	26	3 163
gamma	11	19 020
adult	15	48 842

### 4.2 实验结果

#### 4.2.1 返回的模式数量

不同算法的统计显著对比模式数量如图3所示,其中所有算法参数相同,即 $\theta_{\text{sup}}$ 、 $\theta_{\text{dis}}$ 和 $\alpha$ 相同。从图3可以看出,在各个样本集合中DA和SP算法返回的统计显著模式数量远远大于其他算法。基于直接计算方法和标准随机置换过程方法返回的结果中保留了大量的冗余对比模式。此外,在4种使用冗余对比模式过滤的算法中,FSPRP和FEPRP算法返回的模式数量多于SPRF和IEPCSP算法。其原因是FSPRP和FEPRP算法采用的固定属性置换过程是在考虑子模式影响的条件下计算得出的超模式 $p$ 值。



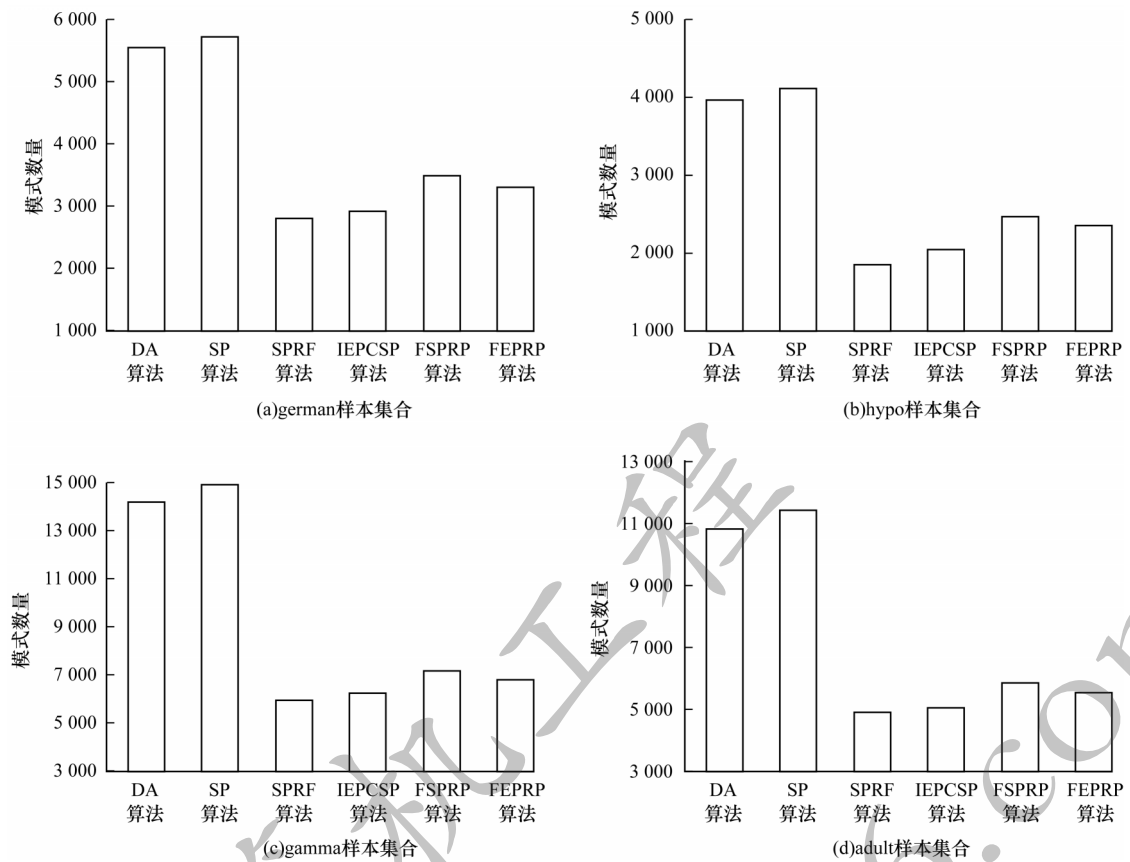


图3 不同算法的统计显著的对比模式数量

Fig.3 The number of statistically significant contrast patterns of different algorithms

为进一步分析冗余对比模式过滤情况,SPRF、IEPCSP、FSPRP和FEPRP算法在hypo样本集合上不同长度模式数量 $l$ (省略了8长度模式以后的信息)如表4所示。从表4可以看出,长模式过滤的比率高于短模式过滤的比率,这是因为模式越长包含的子模式越多,从而受子模式统计显著性影响的可能性就越大。SPRF和IEPCSP算法针对长模式过滤的力度大于FSPRP和FEPRP算法,这表明比较约束方法较固定属性置换过程对长模式的要求更加苛刻,从而更易保留较短的模式。

表4 在hypo样本集合上各算法不同长度模式数量对比

Table 4 The number of patterns with different lengths comparison of each algorithms on hypo sample set

算法	$l=2$	$l=3$	$l=4$	$l=5$	$l=6$	$l=7$	$l=8$
未过滤冗余对比模式算法	30	129	311	508	695	892	799
SPRF算法	24	78	186	270	281	343	301
IEPCSP算法	25	80	205	296	307	367	328
FSPRP算法	28	89	236	351	404	489	391
FEPRP算法	28	84	224	338	387	465	372

4.2.2 分类预测结果

为体现冗余对比模式对后续任务的影响以及过滤的好处,将各算法报告的模式作为数据样本集

中提取的特征进行分类预测实验。本文将每个统计显著的对比模式作为一个特征,利用每条样本与所有模式的包含关系生成特征向量。统计显著的对比模式作为特征是因为其本身反映了不同类别标签样本集合的差异性<sup>[23]</sup>。考虑到分类模型自身因素影响,实验采用3种不同机制的模型,即决策树分类模型<sup>[24]</sup>、逻辑回归分类模型<sup>[24]</sup>和随机森林分类模型<sup>[24]</sup>。此外,为了避免偶然性,本文使用十折交叉验证的平均正确率作为预测结果,决策树分类模型下不同算法的分类准确率对比如表5所示,逻辑回归分类模型不同算法的分类准确率如表6所示,随机森林分类模型不同算法的分类准确率如表7所示。

表5 在决策树分类模型下不同算法的分类准确率对比

Table 5 The classification accuracy comparison among different algorithms under decision tree classifier model %

算法	german	hypo	gamma	adult
DA算法	67.01	92.13	76.85	78.86
SP算法	67.24	92.47	77.04	79.02
SPRF算法	67.89	93.09	79.18	81.21
IEPCSP算法	69.04	93.63	80.33	82.26
FSPRP算法	70.21	94.67	81.96	83.57
FEPRP算法	70.42	94.93	82.48	83.89

表 6 在逻辑回归分类模型下不同算法的分类准确率对比

Table 6 The classification accuracy comparison among different algorithms under logistic resupession

classifier model				%
算法	german	hypo	gamma	adult
DA 算法	74.22	93.06	79.24	83.41
SP 算法	74.41	93.18	79.39	83.67
SPRF 算法	75.89	94.36	81.87	85.03
IEPCSP 算法	76.77	94.98	83.05	86.15
FSPRP 算法	77.61	95.53	84.42	87.72
FEPRP 算法	77.84	95.78	85.03	88.28

表 7 在随机森林分类模型下不同算法的分类准确率

Table 7 The classification accuracy comparison among different algorithms under random forest classifier model

				%
算法	german	hypo	gamma	adult
DA 算法	73.84	92.84	78.71	83.52
SP 算法	73.98	93.12	78.89	83.69
SPRF 算法	75.22	94.12	81.63	85.37
IEPCSP 算法	76.03	94.71	82.92	86.18
FSPRP 算法	77.09	95.59	84.28	87.92
FEPRP 算法	77.27	95.84	84.89	88.43

从表5~表7可以看出,DA和SP算法的分类准确率低于其他4个算法,虽然DA和SP算法使用统计显著性检验去除一定数量的假阳性模式,但其结果还保留着大量的冗余对比模式,尤其是在gamma和adult样本集合返回的结果中。这些冗余对比模式提供的额外无用信息对分类模型产生了一定干扰。本文使用比较约束方法或者固定属性置换过程都过滤了一定数量的冗余对比模式,从而提高准确率。

结合图3可知,虽然SPRF和IEPCSP算法过滤的模式数量多于FSPRP和FEPRP算法,但其相应的准确率却低于FSPRP和FEPRP算法,其原因是SPRF和IEPCSP算法利用比较约束法要求超模式的统计显著性必须大于其子模式才能予以保留,但事实上超模式和子模式的统计显著性并不一定具备这样的关系,从而导致SPRF和IEPCSP算法在过滤冗余对比模式的同时,也过滤了许多非冗余的统计显著模式,从而丢失许多差异特征。而FSPRP和FEPRP算法从冗余对比模式的本质出发,使用固定属性置换过程打破超模式和子模式在置换样本集合中的联系,真正过滤了受子模式统计显著性影响的冗余对比模式,因此达到了更高的预测准确率。

冗余对比模式会给后续任务决策带来干扰,使用固定属性置换过程的FSPRP和FEPRP算法能够过滤掉置换检验中一定数量的冗余对比模式,且比

使用比较约束法的SPRF和IEPCSP算法效果更优。

4.2.3 FSPRP算法和FEPRP算法的讨论

上述实验结果表明,FEPRP算法略优于FSPRP算法,这体现在FEPRP算法报告的模式数量更少但分类准确率更高。虽然FSPRP和FEPRP算法建立的都是近似零分布,但是FSPRP算法采用生成固定属性置换样本集合的方式,而FEPRP算法通过计算对比模式对比性度量值分布的方式。为探究2种算法的构建方式差别,实验通过计算得到german和hypo样本集合的精确零分布分别为3 259和6 748,并对比不同置换次数 $z$ 的FSPRP算法和不同子模式数量 $m$ 的FEPRP算法返回的模式数量与精确零分布返回的模式数量。在german和hypo样本集合上不同算法的近似零分布对比如表8所示。

表 8 在german和hypo样本集上不同算法的近似零分布对比

Table 8 Approximate null distributions comparison among different algorithms on german and hypo sample sets

算法	参数	german	hypo
FEPRP 算法	$m=4$	3 458	7 029
	$m=6$	3 387	6 915
	$m=8$	3 312	6 804
	$m=10$	3 287	6 771
FSPRP 算法	$z=500$	3 591	7 316
	$z=1\ 000$	3 517	7 208
	$z=1\ 500$	3 463	7 146
	$z=2\ 000$	3 423	6 993

从表8中可以得出,增加FSPRP算法中 $z$ 和增加FEPRP算法中 $m$ 均能得到各自更接近精确零分布返回的结果。FEPRP算法在 $m=10$ 时近似零分布与精确零分布的结果已相差不大,但FSPRP算法在 $m=2\ 000$ 时近似零分布与精确零分布的结果还存在一定差距,表明FEPRP算法相较于FSPRP算法更易得到一个接近精确零分布的近似零分布。

由于FSPRP算法零分布的建立方式依据标准置换检验方法,其存在标准置换检验中的 $p$ 值可能为0、结果不唯一、计算开销大3个缺点;FEPRP算法虽然没有建立精确零分布,但其使用精确置换检验中计算对比性度量值分布建立零分布的策略,在 $m$ 确定的情况下,FEPRP算法构造的零分布也是确定的。本文进行对比实验以验证FEPRP算法不存在FSPRP算法中3个缺点。

FSPRP算法在各样本集合中 $p$ 值为0的对比模式数量对比如图4所示。从图4可以看出,FSPRP算法在每个样本集合中均存在一定数量 $p$ 值为0的对比模式,FEPRP算法在构建每个模式的对比性度量



值分布时, 总能找到和该模式对比性度量值相等或更大的模式, 所以 FEPRP 算法不存在  $p$  值为 0 的模式。FSPRP 和 FEPRP 算法在 adult 样本集合上各运行 100 次返回的模式数量如图 5 所示。从图 5 可以看出, FSPRP 算法返回的结果不唯一, 由于每次使用固定属性置换过程生成的置换样本集合均不一样, 因此计算得到的对比模式的  $p$  值也不相同。FEPRP 算法在  $m$  给定的情况下, 每次构建的零分布都是相同的, 因此其结果始终相同。

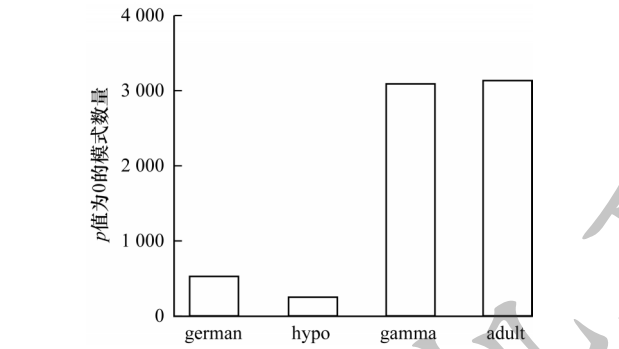


图 4 在不同样本集合上 FSPRP 算法的  $p$  值对比  
Fig.4 The  $p$  values comparison of FSPRP algorithm on different sample sets

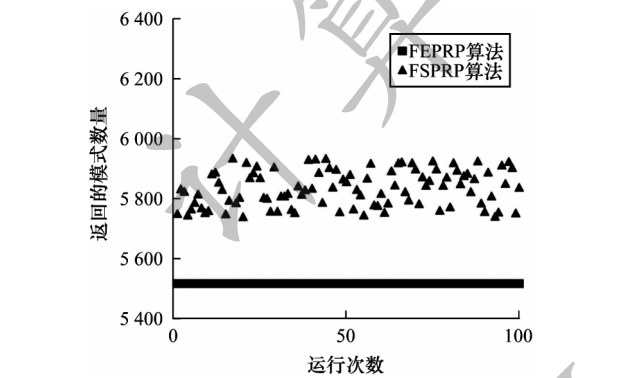


图 5 FSPRP 和 FEPRP 算法运行 100 次的模式数量  
Fig.5 The number of patterns of FSPRP and FEPRP algorithms in the 100 runs

FSPRP 算法的运行时间主要受置换次数  $z$  的影响, 而 FEPRP 算法的运行时间主要受子模式个数  $m$  的影响。在 gamma 样本集合上 FSPRP 和 FEPRP 算法的运行时间对比如图 6 所示。从图 6(a) 可以看出, FSPRP 算法在各个置换次数下的运行时间均超过 FEPRP 算法在各个子模式数量下的运行时间, 说明 FSPRP 算法的计算开销大于 FEPRP 算法。其主要原因是置换样本集合的生成和对比模式的挖掘都是计算开销较大的操作。FSPRP 算法在生成零分布的过程中需要实际使用固定属性置换过程生成置换样本集合, 随后再对这些样本集合进行对比模式挖

掘; 而 FEPRP 算法在对样本集合进行一次挖掘后, 只需要模拟固定属性置换过程生成置换样本集合就能够计算得到零分布, 而不用实际生成置换样本集合, 这样就大幅减少了运行时间。

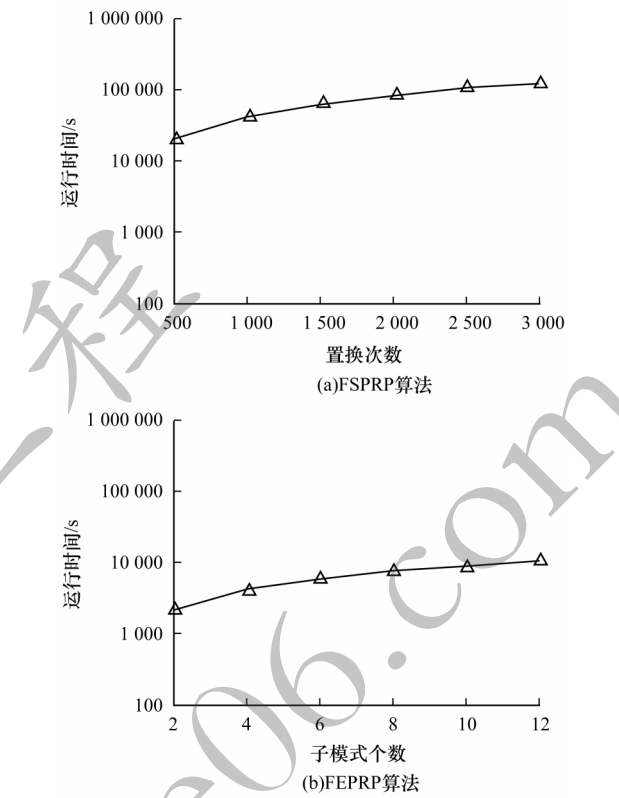


图 6 在 gamma 样本集合上 FSPRP 和 FEPRP 算法的运行时间对比  
Fig.6 The running time comparison of the FSPRP and FEPRP algorithms on gamma sample set

因此, FEPRP 算法整体性能优于 FSPRP 算法, 其更适用于置换检验中过滤冗余对比模式。

因此, FEPRP 算法整体性能优于 FSPRP 算法, 其更适用于置换检验中过滤冗余对比模式。

5 结束语

本文提出 2 个使用固定属性置换过程的冗余对比模式过滤算法 (FSPRP 和 FEPRP), 分别利用生成一定数量的置换样本集合和计算模式对比性度量值分布的方式建立零分布。实验结果表明, FSPRP 和 FEPRP 算法能够过滤置换检验中一定数量的冗余对比模式, 且相较于比较约束法的效果更优。与 FSPRP 算法相比, FEPRP 算法分类准确率较高, 更适用于过滤置换检验中的冗余对比模式。由于在固定属性置换过程中每次得到的样本都会发生改变, 因此后续将设计适用于 FSPRP 算法的一次数据挖掘方法, 以减少运行时间。此外, 将固定属性置换检验方法应用于解决序列数据中冗余对比模式的过滤问题, 也将是下一步研究的重点方向。

## 参考文献

- [1] HÄMÄLÄINEN W, WEBB G I. A tutorial on statistically sound pattern discovery[J]. *Data Mining and Knowledge Discovery*, 2019, 33(2): 325-377.
- [2] FORESTIER G, PETITJEAN F, SENIN P, et al. Finding discriminative and interpretable patterns in sequences of surgical activities[J]. *Artificial Intelligence in Medicine*, 2017, 82(1): 11-19.
- [3] CHENG A, GRANT C E, NOBLE W S, et al. MoMo: discovery of statistically significant post-translational modification motifs[J]. *Bioinformatics*, 2019, 35(16): 2774-2782.
- [4] NEUBARTH K, SHANAHAN D, CONKLIN D. Supervised descriptive pattern discovery in native American music [J]. *Journal of New Music Research*, 2018, 47(1): 1-16.
- [5] FOURNIER P, LIN J C, KIRAN R U, et al. A survey of sequential pattern mining [J]. *Data Science and Pattern Recognition*, 2017, 1(1): 54-77.
- [6] KOH Y S, RAVANA S D. Unsupervised rare pattern mining: a survey [J]. *ACM Transactions on Knowledge Discovery from Data*, 2016, 10(4): 1-29.
- [7] FANG G, PANDEY G, WANG W, et al. Mining low-support discriminative patterns from dense and high-dimensional data [J]. *IEEE Transactions on Knowledge and Data Engineering*, 2010, 24(2): 279-294.
- [8] 杨皓, 段磊, 胡斌, 等. 带间隔约束的Top-k对比序列模式挖掘 [J]. *软件学报*, 2015, 26(11): 2994-3009.
- YANG H, DUAN L, HU B, et al. Mining Top-k distinguishing sequential patterns with gap constraint [J]. *Journal of Software*, 2015, 26(11): 2994-3009. (in Chinese)
- [9] WEBB G I, VREEKEN J. Efficient discovery of the most interesting associations [J]. *ACM Transactions on Knowledge Discovery from Data*, 2014, 8(3): 1-31.
- [10] ZHANG A S, SHI W Z, WEBB G I. Mining significant association rules from uncertain data [J]. *Data Mining and Knowledge Discovery*, 2016, 30(4): 928-963.
- [11] HE Z Y, ZHANG S M, GU F Y, et al. Significance-based discriminative sequential pattern mining [J]. *Expert Systems with Applications*, 2019, 122(1): 54-64.
- [12] LLINARES-LÓPEZ F, SUGIYAMA M, PAPAXANTHOS L, et al. Fast and memory-efficient significant pattern mining via permutation testing [C]//*Proceedings of the 21st ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. New York, USA: ACM Press, 2016: 725-734.
- [13] LIU G M, ZHANG H J, WONG L. Controlling false positives in association rule mining [C]//*Proceedings of the VLDB Endowment*. New York, USA: ACM Press, 2011: 145-156.
- [14] 吴军, 欧阳艾嘉, 张琳. 面向对比序列模式发现的独立精确置换检验算法 [J]. *计算机工程*, 2021, 47(8): 45-53, 61.
- WU J, OUYANG A J, ZHANG L. Discovering contrast sequential patterns based on independent exact permutation testing [J]. *Computer Engineering*, 2021, 47(8): 45-53, 61. (in Chinese)
- [15] HE Z Y, GU F Y, ZHAO C. Conditional discriminative pattern mining: concepts and algorithms [J]. *Information Sciences*, 2017, 375(1): 1-15.
- [16] 高权, 万晓冬. 基于负载均衡的并行FP-Growth算法 [J]. *计算机工程*, 2019, 45(3): 32-35, 40.
- GAO Q, WAN X D. Parallel FP-growth algorithm based on load balance [J]. *Computer Engineering*, 2019, 45(3): 32-35, 40. (in Chinese)
- [17] ZAKI M J, HSIAO C J. Charm: an efficient algorithm for closed itemset mining [C]//*Proceedings of the 2nd SIAM International Conference on Data Mining*. Arlington, USA: SIAM Press, 2002: 457-473.
- [18] GUNS T, NIJSSEN S, RAEDT L. K-pattern set mining under constraints [J]. *IEEE Transactions on Knowledge and Data Engineering*, 2013, 25(2): 402-418.
- [19] VAN L M, KNOBBE A. Diverse subgroup set discovery [J]. *Data Mining and Knowledge Discovery*, 2012, 26(2): 208-242.
- [20] LIU X Q, WU J, GONG H P, et al. Mining conditional phosphorylation motifs [J]. *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, 2014, 11(5): 915-927.
- [21] ZHU Y L, GUO W. Family-wise error rate controlling procedures for discrete data [EB/OL]. [2020-11-12]. <https://arxiv.org/abs/1711.08147>.
- [22] DUA D, GRAFF C. UCI machine learning repository [EB/OL]. [2020-11-10]. <http://archive.ics.uci.edu/ml>.
- [23] LOYOLA G O, MONROY R, RODRÍGUEZ J, et al. Contrast pattern-based classification for bot detection on twitter [J]. *IEEE Access*, 2019, 7: 45800-45817.
- [24] HAN J W, KAMBER M. *Data mining: concepts and techniques* [M]. San Francisco, USA: Morgan Kaufmann Publishers, 2000.

编辑 薛晋栋