# 利用粒计算的符号型数据分组算法

杨 烽
(西南石油大学计算机科学学院 成都 610500)

**摘 要** 在数据挖掘领域,基于符号型数据分组的数据预处理是一个极富挑战性的问题,它给人们提供了一种更加简化的数据表现形式。在已往的研究中,相关学者提出了许多解决方案,例如,运用粗糙集的方法来解决这一问题。文中提出了一种基于粒计算的符号型数据分组算法,主要分为粒度生成和粒度选择两个阶段。在粒度生成阶段,对于每一条属性,以对应属性值的聚类为叶子节点,自底向上以二进制树的形式构建粒层,形成属性树森林。在粒度选择阶段,以信息增益为基础,对每棵树进行全局考虑,选取最优的粒层,选层结果就是符号型数据的分组结果。实验结果表明,本算法呈现出比已有算法更加平衡的层次结构和更加优秀的压缩效率,具有较好的应用价值。

**关键词** 粒计算,信息增益,符号型,数据分组
**中图法分类号** TP311 **文献标识码** A

## Symbolic Value Partition Algorithm Using Granular Computing

YANG Feng
(School of Computer Science, Southwest Petroleum University, Chengdu 610500, China)

**Abstract** In the field of data mining, data preprocessing based on symbolic data packets is a very challenging issue. It provides people with a more simplified representation of data. In the past research, researchers proposed many solutions, such as using rough set approach to solve this problem. In this paper, a symbolic data grouping algorithm based on grain computing was proposed, which is divided into two stages: granularity generation and granularity selection. At the stage of particle size generation, for each attribute, the tree is constructed from the bottom of the leaf with the cluster of corresponding attribute values as a binary tree, forming a forest of attribute trees. In the stage of granularity selection, each tree is globally considered on the basis of information gain, and the optimal grain layer is selected. The result of layer selection is the grouping result of symbolic data. Experimental results show that compared with the existing algorithms, this algorithm presents a more balanced hierarchy and more excellent compression efficiency, and has better application value.

**Keywords** Granular computing, Information gain, Symbol, Value partition

## 1 引言

符号型数据分组也被称为基于属性值的分组,旨在压缩属性值的个数和属性域的大小,类似于连续的属性值离散和属性约减,它是数据挖掘领域中一种非常重要的数据预处理技术[1-2]。相对于属性值离散和属性约减,它更具一般性和挑战性。

许多粗糙集的方法被用于解决此类问题。Bazan 等通过构建新型的差别矩阵[3],将符号型数据分组转换为图形着色问题,然而许多实例证明该方法并不可行。2008 年,闵帆在其研究基础上将此问题转换成一系列属性约减问题,并证明了算法的最优子结构性质,但没能验证贪心选择的有效性[4],因此该算法不能保障分组结果为最佳。

本次研究将粒计算的思想及粒度构建与粒度选择应用到符号型数据分组中。在粒度构建阶段,自底向上以树的形式对每一条条件属性建立属性值的层次结构。每棵树在构造上基于哈夫曼树的形式,但区别于哈夫曼树以最小加权路径为构建依据,本算法中叶子节点均为向量,叶子节点的聚类以向量间的相似度为标准,向量所表示的是条件属性值与决策属性值之间的关系。在粒度选择阶段,算法采用信息增益作为选层标准。计算每棵属性树中每一层之间的信息增益,通过横向、纵向来全局比较属性树层与层之间的信息增益,每次选取信息增益最大的一层,重复该步骤直到最终生成的决策表与最初的决策表信息熵一致。因此,本算法能在压缩决策表属性值数量与属性域的同时保证决策的正确性。

实验结果表明,本算法不仅在单一数据集上有出色的发挥,还能够帮助大多数的数据集完成压缩属性值数目(及减小规则条数)和属性域大小的预处理工作。

## 2 相关技术

本节描述了基于粒计算的符号型数据分组算法所涉及的相关技术基础,包括决策系统[5]、哈夫曼原理[6]、信息熵的基本概念及具体应用[7-8]。

### 2.1 决策系统

决策系统是有监督学习(supervised learning)的基础数据模型,定义如下。

**定义 1** 决策系统是一个五元组:

$$S=(U,C,D,V=\{Va\mid a\in C\cup D\},I=\{I_a\mid a\in C\cup D\}) \quad (1)$$

杨 烽 男,硕士生,主要研究领域为粒计算、推荐系统等。

其中，$U$ 为对象的集合，也称论域；$C$ 为条件属性集合；$D$ 为决策属性集合；$V_a$ 为属性 $a$ 的值域；$I_a: U \rightarrow V_a$ 为信息函数。

大多数决策系统中仅有一个决策属性，记为 $D=\{d\}$。表 1 给出了一些决策表实例。

表 1 决策表实例

| Patient | Mcv | Alkphos | Sgpt | Sgot | Gammagt | Drinks | Selector |
|---|---|---|---|---|---|---|---|
| $x_1$ | Normal | Low | Low | Normal | Normal | Normal | Yes |
| $x_2$ | Abnormal | Normal | Normal | High | Normal | Normal | Yes |
| $x_3$ | Abnormal | Normal | Normal | Low | High | Normal | Yes |
| $x_4$ | Normal | Normal | Normal | Low | High | Abnormal | No |
| $x_5$ | Normal | High | High | Normal | Low | Normal | No |
| $x_6$ | Normal | High | High | Normal | Low | Abnormal | No |

其中，$U=\{x_1, x_2, \cdots, x_6\}$，$C=\{$Mcv, Alkphos, Sgpt, Sgot, Gammagt, Drinks$\}$，$D=\{$Selector$\}$。

### 2.2 类哈夫曼树

给定 $n$ 个权值作为 $n$ 的叶子节点，构造一棵二叉树，称带权路径长度达到最小的二叉树为最优二叉树，也称为哈夫曼树(Huffman tree)[9]。哈夫曼树是带权路径长度最短的树，权值较大的节点离根较近。

如果二叉树中的叶节点都具有一定的权值，则可将这一概念推广。设二叉树有 $n$ 个带权值的叶节点，那么二叉树带权路径长度应记为：

$$WPL=\sum_{k=1}^{n} W_k \times L_k \tag{2}$$

其中，$W_k$ 为第 $k$ 个叶节点的权值，$L_k$ 为第 $k$ 个叶节点的路径长度。

但是，本算法中所构建的树与传统意义上的哈夫曼树有一定的区别：以叶子节点之间的相似度为依据来对叶子节点进行聚类，将相似度较高的叶子节点聚类在一起。因此，定义所构建的树为类哈夫曼树。

### 2.3 信息熵与信息增益

熵的概念源于热物理学，用以表示系统的稳定状态。熵值越大，表示系统越稳定[10]。信息熵由 Shannon 于 1948 年在论文“A Mathematical Theory of Communication”中正式提出，用于表示信息中排除了冗余后的平均信息量，以描述给定信息的不确定度；并给出其计算公式：

$$H(X)=-\sum P(a_i) IbP(a_i) \tag{3}$$

Jaynes 于 1957 年在论文“Information theory and statistical mechanics”中提出了最大信息熵原理，并定义熵为：

$$H(X)=-k\sum_{i=1}^{n} P_i \log P_i \tag{4}$$

将信息熵应用到本算法时，需要重新对熵的概念进行定义。

令 $S=\{U, C, \{d\}\}$ 为一个决策系统，$P, Q \subset C \cup \{d\}$。根据 $P$ 和 $Q$ 对 $U$ 进行分组，结果分别为 $x=\{x_1, x_2, x_3, x_4, x_5, \cdots, x_n\}$ 和 $y=\{y_1, y_2, y_3, y_4, y_5, \cdots, y_n\}$，$P$ 和 $Q$ 可以被视为论域 $U$ 中的随机变量，其概率分布表示如下。

**定义 2** $P$ 和 $Q$ 在论域 $U$ 中的概率分布为：

$$[\chi : p]=\begin{bmatrix} X_1 & X_2 & \cdots & X_n \\ p(X_1) & p(X_2) & \cdots & p(X_n) \end{bmatrix}$$

$$[\gamma : p]=\begin{bmatrix} Y_1 & Y_2 & \cdots & Y_m \\ p(Y_1) & p(Y_2) & \cdots & p(Y_m) \end{bmatrix}$$

其中，$P(X_i)=\dfrac{|X_i|}{|U|}$，$i=1,2,3,\cdots,n$；$P(Y_j)=\dfrac{|Y_j|}{|U|}$，$j=1,2,3,\cdots,m$。

**定义 3** $P$ 的信息熵：

$$H(P)=-\sum_{i=1}^{n} P(X_i)\log(P(X_i)) \tag{5}$$

**定义 4** $P$ 在特征值 $Q$ 下的条件信息熵：

$$H(Q|P)=-\sum_{i=1}^{n} P(X_i)\sum_{j=1}^{m} P(Y_j|X_i)\log(P(Y_j|X_i)) \tag{6}$$

其中，$P(Y_j|X_i)=|Y_j \cup X_i|/X_i$，$i=1,2,3,\cdots,n$；$j=1,2,3,\cdots,m$。

信息增益是指期望信息或信息熵的有效减少量，以及对一个特征而言，系统有它和没它时信息量所发生的变化。信息增益能被用于确定在不同层次上选择相应变量进行分类。

例如计算特征值 $Q$ 对决策系统 $P$ 及其分类带来的信息增益时，首先分别计算决策系统 $P$ 的信息熵，再计算利用特征值 $Q$ 对 $P$ 进行聚类或者分类时所产生的决策系统的信息熵，两者相减所得的值就为决策系统 $P$ 根据特征值 $Q$ 进行分类或者聚类时产生的信息增益。

**定义 5** 信息增益 $IG(P)$：

$$IG(P)=H(P)-H(P/Q) \tag{7}$$

$IG(Q)$越大，表示对特征 $T$ 进行聚类或分类时系统的区分度越大。

**定义 6** 条件信息增益：

$$IG(P_i)=H(P/Q_i)-H((P/Q_i)/Q_{i+1}) \tag{8}$$

### 2.4 符号型数据分组

为了更好地表述本算法，本节将给出符号型数据分组的一些定义。为方便表达，将举例进行说明。

首先，将对一个属性下的属性值分组进行讨论。对属性值 $v_{l1}$ 分组将被表示为 $P_i$。举例说明：设用 $l_1$ 表示表 2 中的条件属性 Occupation，则 $v_{l1}=\{$Student, Doctor, Nurse, Lawyer, Teacher$\}$。而 $P_i=\{\{$Student, Teacher$\}, \{$Doctor, Nurse$\}, \{$Lawyer$\}\}$所表示的是属性 $l_1=$Occupation 的属性值的分组。

**定义 7** 分组方法 $P_i$ 运行后产生的一类属性值为：

$$v_{si}=v_{li} \rightarrow P_i \tag{9}$$

### 2.5 基于粗糙集的符号型数据分组

闵帆[4]于 2008 年在“基于粗糙集的符号型数据分组”一文中利用粗糙集的思想解决了符号型数据的分组问题，其主要思想是将符号型数据分组转换为一系列的属性约简问题。通过对符号型数据分组的研究发现，通过选择最优约减和次优约减，能够得到一个次优约减分区，这个分区往往也是符号型数据分组的一个次优解。

举例说明：定义一个决策系统，如表 2 所列。

表 2　示例决策系统

| $U$ | Occupation | Temperature | Cough | Sars |
|---|---|---|---|---|
| $x_1$ | Student | Low | Yes | Suspicious |
| $x_2$ | Doctor | High | No | Yes |
| $x_3$ | Nurse | High | Yes | Yes |
| $x_4$ | Nurse | Normal | Yes | Yes |
| $x_5$ | Teacher | Normal | No | Suspicious |
| $x_6$ | Teacher | Normal | Yes | Suspicious |
| $x_7$ | Lawyer | Normal | Yes | No |
| $x_8$ | Student | Normal | No | No |
| $x_9$ | Student | High | No | No |

通过发现各条件属性的属性值与决策属性之间的关系，对当前决策表进行第一轮转换，如表 3 所列。

其中，$(O,s)$代表条件属性 Occupation 下的 Student 属性值，$(O,d)$代表条件属性 Occupation 下的 Doctor 属性值，$(O,n)$代表条件属性 Occupation 下的 Nurse 属性值，$(O,t)$代表条件属性 Occupation 下的 Teacher 属性值$(O,l)$代表条件属性 Occupation 下的 Lawyer 属性值。以此类推，$(T,l)$表示条件属性 Temperature 下的 Low 属性值，$(C,y)$表示条件属性 Cough 下的 Yes 属性值。表中的 0 和 1 表示属性值的当前状态，例如：规则 $x_1$ 表示当患者的 Occupation 为 Student、Temperature 为 Low、Cough 为 No 时，决策表所给出的诊断意见为 Suspicious。

表 3　第一轮转换

| $U$ | $(O,s)$ | $(O,d)$ | $(O,n)$ | $(O,t)$ | $(O,l)$ | $(T,l)$ | $(T,h)$ | $(T,n)$ | $(C,y)$ | $(C,n)$ | $d$ |
|---|---|---|---|---|---|---|---|---|---|---|---|
| $x_1$ | 1 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 1 | Suspicious |
| $x_2$ | 0 | 1 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 1 | Yes |
| $x_3$ | 0 | 0 | 1 | 0 | 0 | 0 | 1 | 0 | 1 | 0 | Yes |
| $x_4$ | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 1 | 1 | 0 | Yes |
| $x_5$ | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 1 | 0 | 1 | Suspicious |
| $x_6$ | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 1 | 1 | 0 | Suspicious |
| $x_7$ | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 1 | 1 | 0 | No |
| $x_8$ | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 1 | No |
| $x_9$ | 1 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 1 | No |

通过表 3，我们能够发现相同状态的部分条件属性所对应的决策属性值是相同的，若当前条件属性下并未出现决策信息不一致或信息丢失等情况，则在此轮后形成新的决策系统，如表 4 所列。

表 4 中属性 $O^{P^1}$ 中的 1 所代表{StudentLlawyer}，$T^{P^1}$ 中的 1 代表{HighNnormal}，$C^{P^1}$ 中的 1 代表{Yes,No}。通过此表可以发现，$C^{P^1}$ 这一属性对决策属性值的判定没有任何影响，可以约减掉。然后，将此表进行第二轮转换，如表 5 所列。

表 4　第一轮聚类结果

| $U$ | $O^{P^1}$ | $T^{P^1}$ | $C^{P^1}$ | $d$ |
|---|---|---|---|---|
| $x_1$ | 1 | Low | 1 | Suspicious |
| $x_2$ | Doctor | 1 | 1 | Yes |
| $x_3$ | Nurse | 1 | 1 | Yes |
| $x_5$ | Teacher | 1 | 1 | Suspicious |
| $x_7$ | 1 | 1 | 1 | No |

表 5　第二轮转换

| $U$ | $(O^{P^2},1)$ | $(O^{P^2},2)$ | $(O^{P^2},t)$ | $(T^{P^2},1)$ | $(T^{P^2},2)$ | $(C^{P^2},1)$ | $d$ |
|---|---|---|---|---|---|---|---|
| $u_1$ | 1 | 0 | 0 | 0 | 1 | 1 | Suspicious |
| $u_2$ | 0 | 1 | 0 | 1 | 0 | 1 | Yes |
| $u_5$ | 0 | 0 | 1 | 1 | 0 | 1 | Suspicious |
| $u_7$ | 1 | 0 | 0 | 1 | 0 | 1 | No |

重复以上步骤，直至决策表中属性值的数量和属性域的大小不能再被压缩，算法得出的最终结果如表 6 所列。

表 6　第二轮聚类结果

| $U$ | $O^{P^3}$ | $T^{P^3}$ | $C^{P^3}$ | $d$ |
|---|---|---|---|---|
| $u_1$ | {Student, Lawyer} | {Low} | {Yes,No} | Suspicious |
| $u_2$ | {Doctor, Nurse} | {Normal, High} | {Yes,No} | Yes |
| $u_5$ | {Teacher} | {Normal, High} | {Yes,No} | Suspicious |
| $u_7$ | {Student, Lawyer} | {Normal, High} | {Yes,No} | No |

通过这种方法，将属性值分类问题转换为了一系列属性约简问题。但此方法也存在一些问题，如因为没有限制条件，最后得出的决策表与原决策表在判定精度上会有出入，而且不能保证在过程中产生的子结构的平衡性。这些都是本算法需要克服的缺点。

## 3　算法框架

本节将讨论所设计算法的具体框架。首先将对最佳属性值分组问题进行定义；然后从两个阶段来解决本问题，第一阶段为粒度生成，第二阶段为粒度选择，每一阶段的具体工作均将通过理论结合实例的方式完整呈现。

### 3.1　问题定义

属性值分组问题的目的是压缩决策系统。为达到这个目的，需要定义属性和决策系统的区间。特征选取相关问题可以被理解为对相关属性进行约束。将最佳属性值分组问题定义如下：

Input：A nominal decision system S.

Output：A partition scheme P.

Constraint：P is consistent wrt. S.

Optimization objective：min rank(SP).

通过这种方法，可以避免很多不必要的工作，如闵帆提出的“基于粗糙集的符号型数据分组”中对分区约减的定义，从而促使本算法使用更一般的方法解决问题。还要注意的是，本算法处理的是医疗系统中的海量数据，因此需要将启发式的方法应用于本算法。

算法主要采用的粒计算技术为粒度生成和粒度构建。在粒度构建阶段，对每一条属性构建二进制树，这些树可以看作

是对属性的分类,因此这些树被称为属性的分类树。在粒度选择阶段,通过遍历比较整个属性树森林,从而获得一个属性值的分组方案。这两部分工作都将在 3.2 节和 3.3 节中详细介绍。为了更好地说明本算法,本节构造了一个具体的五元组决策系统,如表 7 所列。

表 7 实验用决策表

| $U$ | Occupation | Temperature | Cough | Sars |
|---|---|---|---|---|
| $x_1$ | Teacher | High | No | Suspicious |
| $x_2$ | Doctor | High | No | Yes |
| $x_3$ | Student | Low | Yes | Suspicious |
| $x_4$ | Student | Normal | No | No |
| $x_5$ | Student | High | No | No |
| $x_6$ | Teacher | High | Yes | Yes |
| $x_7$ | Lawyer | Normal | No | No |
| $x_8$ | Doctor | Low | Yes | Yes |
| $x_9$ | Teacher | Normal | No | Suspicious |
| $x_{10}$ | Student | High | Yes | Yes |
| $x_{11}$ | Doctor | High | Yes | Yes |
| $x_{12}$ | Nurse | Low | Yes | Yes |
| $x_{13}$ | Nurse | Low | No | Suspicious |
| $x_{14}$ | Lawyer | Low | No | No |
| $x_{15}$ | Doctor | Normal | No | Suspicious |
| $x_{16}$ | Nurse | High | Yes | Yes |
| $x_{17}$ | Teacher | Normal | Yes | Suspicious |
| $x_{18}$ | Doctor | Low | No | Suspicious |
| $x_{19}$ | Nurse | Normal | Yes | Yes |
| $x_{20}$ | Nurse | Normal | No | Suspicious |
| $x_{21}$ | Lawyer | High | Yes | Yes |
| $x_{22}$ | Lawyer | Low | Yes | Suspicious |
| $x_{23}$ | Doctor | Normal | Yes | Yes |
| $x_{24}$ | Lawyer | Normal | Yes | No |
| $x_{25}$ | Teacher | Low | No | No |
| $x_{26}$ | Nurse | High | No | Yes |

其中,$U=\{x_1,x_2,x_3,x_4,\cdots,x_{26}\}$,$C=\{$Occupation,Temperature,Cough$\}$,$D=\{$Sars$\}$。

### 3.2 粒度生成

本节就怎样根据决策表构建粒度进行介绍。为了让本算法能更好地被理解,将在理论分析之后对构造的实例进行具体说明。

首先,算法在原决策表的基础上,根据每一条属性与之对应的决策属性生成子表(见表 8)。这样做,是为了挖掘出每一条属性的属性值与决策属性值之间的关系。属性值 Occupation 与决策属性值 Sars 之间的关系如表 8 所列。

表 8 属性值 Occupation 与决策属性值 Sars 之间的关系

(a)Teacher

| $U$ | Occupation | Sars |
|---|---|---|
| $x_1$ | Teacher | Suspicious |
| $x_6$ | Teacher | Yes |
| $x_9$ | Teacher | Suspicious |
| $x_{17}$ | Teacher | Suspicious |
| $x_{25}$ | Teacher | No |

(b)Lawyer

| $U$ | Occupation | Sars |
|---|---|---|
| $x_7$ | Lawyer | No |
| $x_{14}$ | Lawyer | No |
| $x_{21}$ | Lawyer | Yes |
| $x_{22}$ | Lawyer | Suspicious |
| $x_{24}$ | Lawyer | No |

(c)Doctor

| $U$ | Occupation | Sars |
|---|---|---|
| $x_3$ | Doctor | Yes |
| $x_8$ | Doctor | Yes |
| $x_{11}$ | Doctor | Yes |
| $x_{15}$ | Doctor | Suspicious |
| $x_{18}$ | Doctor | Suspicious |
| $x_{23}$ | Doctor | Yes |

(d)Nurse

| $U$ | Occupation | Sars |
|---|---|---|
| $x_{12}$ | Nurse | Yes |
| $x_{13}$ | Nurse | Yes |
| $x_{16}$ | Nurse | Yes |
| $x_{19}$ | Nurse | Suspicious |
| $x_{20}$ | Nurse | Suspicious |
| $x_{26}$ | Nurse | Yes |

(e)Student

| $U$ | Occupation | Sars |
|---|---|---|
| $x_2$ | Student | Suspicious |
| $x_4$ | Student | No |
| $x_5$ | Student | No |
| $x_{10}$ | Student | Yes |

在此基础上,为了将挖掘的关系用数学形式表现出来,给出如下关系向量的定义。

**定义 8** 关系向量:

$$v(a,v_{li})=[ct_1,ct_2,ct_3,ct_4,\cdots,ct_k] \tag{10}$$

其中,$a$ 表示条件属性,$v_{li}$ 表示条件属性值,$k$ 表示决策属性值的种类,$ct_k(1\leqslant i\leqslant k)$则表示当前条件属性的属性值对应的种类决策属性值的数量。

从表 2 中,我们能够得到条件属性 Occupation 相对于决策属性 Sars 的关系向量:

$v$(Occuption;Teacher)=[1;1;3],$v$(Occuption;Lawyer)=[1;3;1],$v$(Occuption;Doctor)=[4;0;2],$v$(Occuption;Nurse)=[4;0;2]和 $v$(Occuption;Student)=[1;2;1]。用同样的方法,可以得到其他条件属性与决策属性的关系向量 $v$(Temperatrue;High)=[7;1;1],$v$(Temperature;Lawyer)=[2;2;4],$v$(Temperature;Normal)=[2;3;4],$v$(Cough;Yes)=[2;6;5]和 $v$(Cough;No)=[9;1;3]。

根据定义 7,聚类后的属性值集为 $v_s$。

**定义 9** 聚类后的 $v_s$ 属性值集向量:

$$v(a,v_{si})=\sum_{v_l\in v_s}v(a,v_{li}) \tag{11}$$

式(11)表示的是属性值的聚类过程。其中,$a$ 表示条件属性,$v_l$ 表示条件属性值,$v_s$ 表示能够进行聚类的条件属性值集。

本节需解决的主要问题是如何对条件属性值进行聚类。通过挖掘,发现在同一条件属性下,若不同条件属性值与决策属性值的关系向量之间的相似度越高,则表示这两种条件属性值越能够聚类。这里引入相似度的计算公式:

$$Sim(X,Y)=\cos\theta=\frac{X\cdot Y}{|X|\cdot|Y|} \tag{12}$$

其中,$X$ 和 $Y$ 代表同一条件属性下的不同属性值向量或不同属性值聚类后的向量。例如:

$Sim(v$(Occupation,Nurse),$v$(Occupation,Doctor))=1

通过相似计算,算法发现虽然一些属性值似乎并不影响

决策属性的决策,但它们往往能聚类在一起,例如:

$Sim(v(\text{Occupation},\text{Nurse}),v(\text{Occupation},\text{Doctor}))=1$

属性值为 Doctor 和 Nurse 时对于病情的决策完全一致,于是不用区分属性值 Doctor 和 Nurse。从语义的角度,Doctor 和 Nurse 可以被一个属性值代替。通过这种方法,可以建立一个初始的分类法:

$P_1=[\{(D,N),(S,L),(T)\},\{(L,N),(H)\},\{(Y,N)\}]$

该式表示的是通过第一轮相似度计算后,所有属性下的属性值的聚类情况。通过第一次聚类之后,发现聚类后的属性值之间仍然存在紧密联系,因此再次对聚类后的属性值向量进行相似度计算。例如:

$Sim(v(\text{Occupation},(\text{Doctor},\text{Nurse})),v(\text{Occupation},(\text{Sudent},\text{Lawyer})))=0.96$

因此属性值能够进一步进行聚类。重复此步骤,直到属性域的大小不能再被压缩。

为了保证算法具有一个平衡的子结构,将从下至上地基于哈夫曼树原理构造模型。具体模型如图 1－图 3 所示。

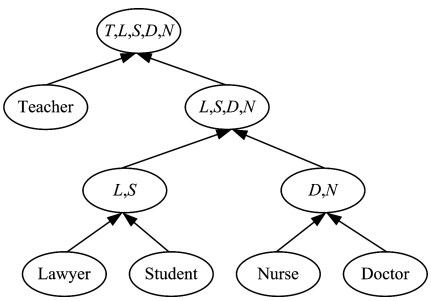


图 1　属性树 Occupation 的构建

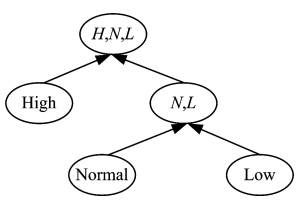


图 2　属性树 Temperature 的构建

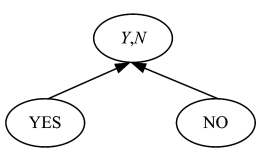


图 3　属性树 Cough 的构建

### 3.3　粒度选择

本节主要介绍在构建粒度模型之后,如何进行粒层的选择。因为在完成粒度构建后,每一棵树的顶层相当于对每一条件属性的属性值并没有区分,但是最终生成决策表的过程中会发现,根据当前决策表进行判断并不能得出正确的判断结果,则我们的算法仅仅起到了压缩决策表的作用,不能真正应用到实际中。于是,需要对构建的粒度进行粒度的选择,使算法能够在减少属性值个数和压缩属性域的同时不影响决策的正确率。

因为每次对每一条件属性的属性值进行聚类之后都会形成一张新的决策表,而信息熵所表示的是当前决策系统的混乱程度,所以通过计算每一棵属性值树中每一层的信息熵也就是计算每一次聚类后生成的新的决策表的信息熵,可以借助其判断本次分类是否正确。根据定义 3 对信息熵的计算,我们能够对构建的粒度进行量化分析,这里将通过本节所构造的实例模型进行详细介绍,如图 4 所示。

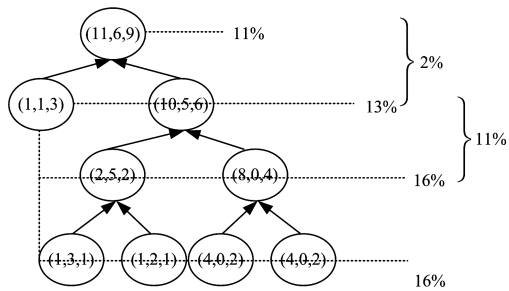


图 4　对属性树 Occupation 的选层结果

这是对属性树 Occupation 所进行的量化分析以及针对其每一层形成的决策表计算信息熵。根据定义 4,可将信息熵记为 $H(\text{Occupation}/i)$,这里的 $i$ 代表属性树的层数。同理计算出属性树 Temperature 和属性树 Cough 的每一层所形成决策表的信息熵,分别记为 $H(\text{Temperature}/i)$ 和 $H(\text{Cough}/i)$,如图 5、图 6 所示。

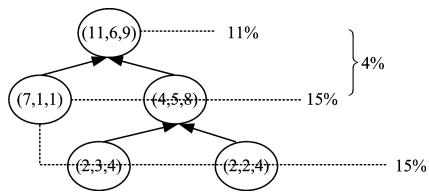


图 5　对属性树 Temperature 的选层结果

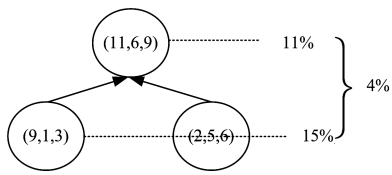


图 6　对属性树 Cough 的选层结果

根据定义 6,计算出层与层之间的信息增益,如图 4 所示,在属性树 Occupation 中第一层与第二层之间的信息增益为 $IG(\text{Occupation}_1)=2\%$。

在计算完所有属性树层与层之间的信息增益之后,采用全局策略进行比较;并且算法并不是单一地进行横向比较或纵向比较,而是采用两种比较相结合的方式。具体是指,首先纵向比较每棵树中哪两层之间的信息增益最大,根据实例(图 4－图 6)中的计算结果,能够发现属性树 Occupation 中层与层之间的最大信息增益为 $IG(\text{Occupation}_2)=3\%$,Temperature 中层与层之间的最大信息增益为 $IG(\text{Temperature}_1)=4\%$,Cough 中层与层的最大信息增益 $IG(\text{Cough}_1)=4\%$;然后再进行横向比较,结果为:$IG(\text{Occupation}_2)<IG(\text{Temperature}_1)=IG(\text{Cough}_1)$。

这里根据先来先得的原则,基于第一次选层的结果属性树 Temperature 的第二层形成新的决策表,如表 9 所列。

但是通过表 9 发现,其中有两条规则在它们条件属性值一样时决策属性却不一样,这在算法中则体现为比较当前决策表的信息熵与初始系统的信息熵。如果熵值差距大,则表示决策不一致。本次选层错误,继续选层。

表 9 第一次选择的结果

| $U$ | Occupation | Temperature | Cough | Sars |
|---|---|---|---|---|
| $x_1$ | Teacher | High | No | Suspicious |
| $x_2$ | Doctor | High | No | Yes |
| $x_3$ | Student | LN | Yes | Suspicious |
| $x_4$ | Student | LN | No | No |
| $x_5$ | Student | High | No | No |
| $x_6$ | Teacher | High | Yes | Yes |
| $x_7$ | Lawyer | LN | No | No |
| $x_8$ | Doctor | LN | Yes | Yes |
| $x_9$ | Teacher | LN | No | Suspicious |
| $x_{10}$ | Student | High | Yes | Yes |
| $x_{11}$ | Doctor | High | Yes | Yes |
| $x_{12}$ | Nurse | LN | Yes | Yes |
| $x_{13}$ | Nurse | LN | No | Suspicious |
| $x_{14}$ | Lawyer | LN | No | No |
| $x_{15}$ | Doctor | LN | No | Suspicious |
| $x_{16}$ | Nurse | High | Yes | Yes |
| $x_{17}$ | Teacher | LN | Yes | Suspicious |
| $x_{18}$ | Doctor | LN | No | Suspicious |
| $x_{19}$ | Nurse | LN | Yes | Yes |
| $x_{20}$ | Nurse | LN | No | Suspicious |
| $x_{21}$ | Lawyer | High | Yes | Yes |
| $x_{22}$ | Lawyer | LN | Yes | Suspicious |
| $x_{23}$ | Doctor | LN | Yes | Yes |
| $x_{24}$ | Lawyer | LN | Yes | No |
| $x_{25}$ | Teacher | LN | No | No |
| $x_{26}$ | Nurse | High | Yes | Yes |

因为第一轮选层时，属性树 Occupation 的第三层与 Cough 的第二层并未被选取，所以此次计算出 Temperature 第二层与第三层的信息增益 $IG(\text{Temperature}_2)=0$，将其与 $IG(\text{Cough}_1)$ 和 $IG(\text{Occupation}_2)$ 进行横向比较：$IG(\text{Temperature}_2)<IG(\text{Occupation}_2)<IG(\text{Cough}_1)$。

第二轮选层所选取的是属性树 Cough 的第二层，生成的决策表如表 10 所列。

表 10 第二次的选层结果

| $U$ | Occupation | Temperature | Cough | Sars |
|---|---|---|---|---|
| $x_1$ | Teacher | High | Y,N | Suspicious |
| $x_2$ | Doctor | High | Y,N | Yes |
| $x_3$ | Student | Low | Y,N | Suspicious |
| $x_4$ | Student | Normal | Y,N | No |
| $x_5$ | Student | High | Y,N | No |
| $x_6$ | Teacher | High | Y,N | Yes |
| $x_7$ | Lawyer | Normal | Y,N | No |
| $x_8$ | Doctor | Low | Y,N | Yes |
| $x_9$ | Teacher | Normal | Y,N | Suspicious |
| $x_{10}$ | Student | High | Y,N | Yes |
| $x_{11}$ | Doctor | High | Y,N | Yes |
| $x_{12}$ | Nurse | Low | Y,N | Yes |
| $x_{13}$ | Nurse | Low | Y,N | Suspicious |
| $x_{14}$ | Lawyer | Low | Y,N | No |
| $x_{15}$ | Doctor | Normal | Y,N | Suspicious |
| $x_{16}$ | Nurse | High | Y,N | Yes |
| $x_{17}$ | Teacher | Normal | Y,N | Suspicious |
| $x_{18}$ | Doctor | Low | Y,N | Suspicious |
| $x_{19}$ | Nurse | Normal | Y,N | Yes |
| $x_{20}$ | Nurse | Normal | Y,N | Suspicious |
| $x_{21}$ | Lawyer | High | Y,N | Yes |
| $x_{22}$ | Lawyer | Low | Y,N | Suspicious |
| $x_{23}$ | Doctor | Normal | Y,N | Yes |
| $x_{24}$ | Lawyer | Normal | Y,N | No |
| $x_{25}$ | Teacher | Low | Y,N | No |
| $x_{26}$ | Nurse | High | Y,N | Yes |

通过表 10，发现依然有两条规则在它们条件属性值一样时决策属性不一样，说明这种选层方案依然失败。此时仅剩属性树 Occupation 的第三层和属性树 Temperature 的第三层未被选取，则比较信息增益 $IG(\text{Temperature}_2)=0$ 和信息增益 $IG(\text{Occupation}_2)=3\%$：$IG(\text{Temperature}_2)<IG(\text{Occupation}_2)$。

第三轮的选层方案为属性树 Occupation 的第三层，所得决策表如表 11 所列。

表 11 第三次选层的结果

| $U$ | Occupation | Temperature | Cough | Sars |
|---|---|---|---|---|
| $x_1$ | Doctor,Nurse | High | No | Yes |
| $x_2$ | Doctor,Nurse | High | Yes | Yes |
| $x_3$ | Doctor,Nurse | Low | Yes | Yes |
| $x_4$ | Doctor,Nurse | Low | No | Suspicious |
| $x_5$ | Doctor,Nurse | Normal | Yes | Yes |
| $x_6$ | Doctor,Nurse | Normal | No | Suspicious |
| $x_7$ | Student,Lawyer | High | No | No |
| $x_8$ | Student,Lawyer | High | Yes | Yes |
| $x_9$ | Student,Lawyer | Low | Yes | Suspicious |
| $x_{10}$ | Student,Lawyer | Normal | No | No |
| $x_{11}$ | Student,Lawyer | Low | No | No |
| $x_{12}$ | Student,Lawyer | Normal | Yes | No |
| $x_{13}$ | Teacher | High | No | Suspicious |
| $x_{14}$ | Teacher | High | Yes | Yes |
| $x_{15}$ | Teacher | Low | No | No |
| $x_{16}$ | Teacher | Normal | Yes | Suspicious |
| $x_{17}$ | Teacher | Normal | No | Suspicious |

经过本次选层，并未发现两条规则在它们条件属性值一样时决策属性却不一样的情况，证明当前的选层方案是正确的；且此时剩下的属性树 Temperature 的第三层和属性树 Cough 的第二层均为属性树的最底层，相当于并未对属性值进行分组，不用继续向下选层。因此，当前决策表为选层的最终结果，也是“基于粒计算的符号型数据分组”算法的最终结果。

通过观察表 11 和表 1 两张决策表，能够发现当前决策表中的规则数为 17 条，而初始表中的规则数为 26 条，压缩了 9 条规则，成功起到了压缩数据集的作用。

总的来说，粒度选择这一阶段可以概括为：对每一棵属性树均进行量化分析，首先纵向比较信息增益，得出最大值，再横向比较得出所有属性树中的最大值，从而选取层次，得到新的决策表；验证新的决策表是否出现在相同条件属性下决策不一致的情况（及比较当前决策表与初始决策表的信息熵），未出现则代表选层成功，此时只需在剩下的属性值中继续选层，如果失败，则根据信息增益值从大到小的顺序，选取下一个信息增益与其他属性树的当前最大信息增益进行比较，重复此过程，直至所得决策表在保持决策一致的情况下决策表不能再被压缩为止。

## 4 实验结果比较

本节将根据实验来判定算法的性能，将比较本算法和“基于粗糙集的符号型数据分组”实现后所压缩的属性条目和产生决策表的精确度。为了简化对这两种算法的标识，将本算法和“基于粗糙集的符号型数据分组”分别简记为“GSVP”和“RSVP”。

### 4.1 数据集

本节将从机器学习数据库中（UCI 数据集）[10] 选取 3 个

数据集对算法进行测试:

1)Mushroom 数据集(蘑菇数据集)[11];

2)Tic-tac-toe 数据集(井字游戏数据集)[13];

3)car 数据集(天气情况统计)[12]。

这些数据集的具体信息如表 12 所列,其中$|C|$是属性数量,$|U|$是实例数量,$|D|$是决策名称,$|V_d|$是决策属性值的数量。

表 12　实验数据集信息

| Dataset | $\|C\|$ | $\|U\|$ | $\|D\|$ | $\|V_d\|$ |
|---|---|---|---|---|
| Mushroom | 21 | 8124 | Classes | 2 |
| Tic-tac-toe | 10 | 1956 | Classes | 2 |
| Car | 6 | 1728 | Classes | 4 |

### 4.2　实验进度

由于有些数据中包含有连续属性,因此需要首先对这些数据集进行离散化。为了简单起见,将离散化后的数据集记为"$S$"。

在离散化后的数据集上使用算法,算法运行过程中将不断产生新的数据集。根据分组的效果,这些数据集有不同的等级,将最佳分组结果记为 $R$,最差的分组结果记为($S^R$),其中 $S^R=(U,R,\{d\})$。

鉴于一些数据集含有制定的训练集,实验结果产生的规则数是对应于训练集的;而其他数据集产生的规则数是基于整个数据集的。实验结果将体现在对整个决策表的压缩效率和分类精度上。

### 4.3　实验结果分析

本算法和"基于粗糙集的符号型数据分组"算法所产生的决策表是不一样的,且面向不同规模的数据集所起到的效果也不同。本节将从压缩后所产生的规则数和压缩后产生决策表的精确度两个角度进行比较。

进行实验时,测试算法分别选取每一个数据集的 10%,20%,30%,40%,50%,60%作为训练集,余下部分作为测试集来对新生成的决策表进行测试,每一种数据集测试 100 次,Aaccuracy 表示这 100 次实验所得准确率的平均值,而 Instance 则表示运行算法后所得到的压缩数据集中的实例数。采用本算法和"基于粗糙集的符号型数据分组"算法所产生的实验结果如表 13 所列。

表 13　实验结果对比表

| Dataset | Accuracy | | Instance | |
|---|---|---|---|---|
| | GSVP | RSVP | GSVP | RSVP |
| Mushroom | 0.987 | 0.974 | 48 | 43 |
| Tic-tac-toe | 0.842 | 0.754 | 1744 | 1687 |
| Car | 0.886 | 0.785 | 1296 | 1324 |

根据表 13 得出了以下结论。

结论 1　采用 Mushroom 数据集时,实验取得了很好的结果,当采用 GSVP 算法[14]时将 8124 条实例压缩至了 48 条,且分类精度达到了 0.987;采用 RSVP 算法[15]时将 8124 条实例压缩至了 43 条,分类精度达到 0.974,而采用其他数据集时的效果较差。通过对数据集进行分析发现,相对于其他数据集而言,Mushroom 条件属性值规整,规则易于提取,因此取得了较好的结果。而 Tic-tac-toe 中的条件属性值排列复杂度高,规则不易提取,因此实验结果较差。

结论 2　根据实验结果发现,采用 GSVP 算法和 RSVP 算法对数据集的压缩结果影响不大。在数据集 Tic-tac-toe 中,采用 GSVP 算法和 RSVP 算法所减少的实例数分别为 212 条和 269 条;在数据集 Car 中,采用 GSVP 算法和 RSVP 算法所减少的实例条数分别为 432 条和 404 条数据集 Mushroom 中,实验结果差别很小,分别为 8076 条和 8081 条。

结论 3　实验结果显示,采用 GSVP 算法和 RSVP 算法所产生决策表的分类精度差别较大,而这在数据集 Mushroom 中体现得并不明显,采用这两种算法所得到的决策表的分类精度相差较小,分别为 0.987 和 0.974;但在数据集 Tic-tac-toe 和数据集 Car 上体现得非常明显,数据集 Tic-tac-toe 中采用两种算法所产生的分类精度分别为 0.842 和 0.754,数据集 Car 则为 0.886 和 0.785。总的来说,采用本算法的最终生成的决策表的分类精度高于 RSVP 算法分类精度。

综上所述,本算法能够在保证精度的情况下压缩决策表的属性值数目和属性域大小。压缩的效果取决于原始决策表的条件属性值是否规整和规则提取的难易程度,能对部分数据集取得很好的应用效果,例如在 Mushroom 数据集上,同时采用不同算法对决策表的压缩效率相差不大,但对分类精度影响较大,且采用本算法所生成的决策表的精度普遍高于 RSVP 的精度。

**结束语**　本文提出了用粒计算的方法,从粒度的构建和选择的角度考虑、解决符号型数据分组的问题。实验结果表明,本算法从压缩比和精确度上全面超越了理论计算值;同时,本算法不仅能够在单一数据集上有出色的发挥,还能够帮助大多数的数据集完成压缩属性值数目(及减小规则条数)和属性域大小的预处理工作。

## 参 考 文 献


[1]　王齐,钱宇华,李飞江. 基于空间结构的符号数据仿射传播算法[J]. 模式识别与人工智能,2016,29(12):1132-1139.

[2]　党红恩,赵尔平,刘炜,等. 利用数据变换与并行运算的闭频繁项集挖掘方法[J]. 湘潭大学自然科学学报,2018,40(1):119-122.

[3]　BAZAN J G,NGUYEN H S,NGUYEN S H,et al. Rough Set Algorithms in Classification Problem[C]// Rough set methods and applications. Physica-Verlag GmbH,2000:49-88.

[4]　MIN F,LIU Q,FANG C. Rough sets approach to symbolic value partition[J]. International Journal of Approximate Reasoning,2008,49(3):689-700.

[5]　沈思倩,毛宇光,江冠儒. 不完全数据集的差分隐私保护决策树研究[J]. 计算机科学,2017,44(6):139-143.

[6]　HOSSAIN M M,HABIB A,RAHMAN M S. Transliteration Based Bengali Text Compression using Huffman principle[C]// International Conference on Informatics,Electronics & Vision. IEEE,2014:1-6.

[7]　朱淑芹,李俊青,葛广英. 基于一个新的四维离散混沌映射的图像加密新算法[J]. 计算机科学,2017,44(1):188-193.

[8]　孙艳歌,王志海,原继东,等. 基于信息熵的数据流自适应集成分类算法[J]. 中国科学技术大学学报,2017,47(7):575-582.

[9]　XU Y,CHEN B Z,HU Z C. Research for multi-sensor data fusion based on Huffman tree clustering algorithm in greenhouses [J]. International Journal of Embedded Systems,2016,8(1):34.

[10]　曹鹏,栗伟,赵大哲. 面向不均衡数据集的 ARSGOS 算法[J]. 小型微型计算机系统,2014,35(4):818-823.

[11] FALANDYSZ J. Review:On published data and methods for selenium in mushrooms[J]. Food Chemistry,2013,138(1):242-250.

[12] YANG L,LUO P,CHEN C L,et al. A large-scale car dataset for fine-grained categorization and verification[C]// Computer Vision and Pattern Recognition. IEEE,2015:3973-3981.

[13] SHASHA D. Open Field Tic-Tac-Toe[J]. Communications of the Acm,2017,60(1):112.

[14] JONAS A. DieGSVP-Agenturen als Forschungsobjekt[M]// Das Governance-System der GSVP: Die Rolle des EU-Satellitenzentrums und der Europäischen Verteidigungsagentur. Nomos Verlagsgesellschaft mbH & Co. KG,2015:133-177.

[15] 赵继军,郭昆,冯楠,等. 基于 RSVP—TE 的有向泛洪 IRWA 算法研究[J]. 光通信研究,2013(5):8-11.

(上接第 426 页)

[2] STRANG K. How student behavior and reflective learning impact grades in online business courses[J]. Journal of Applied Research in Higher Education,2016,8(3):390-410.

[3] PRIOR D D,MAZANOV J,MEACHEAM D,et al. Attitude, digital literacy and self efficacy:Flow-on effects for online learning behavior[J]. Internet & Higher Education, 2016,29:91-97.

[4] BUTCHER K R,SUMNER T. How Does Prior Knowledge Impact Students' Online Learning Behaviors? [J]. International Journal of Cyber Behavior Psychology & Learning,2011,1(4):1-18.

[5] YANG C,HSIEH T. Regional differences of online learning behavior patterns[J]. Electronic Library,2013,31(2):167-187.

[6] PARK Y,YU J H,JO I H. Clustering blended learning courses by online behavior data:A case study in a Korean higher education institute[J]. Internet & Higher Education,2016,29:1-11.

[7] SHIMADA A,OKUBO F,YIN C,et al. Informal Learning Behavior Analysis Using Action Logs and Slide Features in E-Textbooks[C]// International Conference on Advanced Learning Technologies. IEEE,2015:116-117.

[8] HWANG W Y,SHADIEV R,WANG C Y,et al. A pilot study of cooperative programming learning behavior and its relationship with students' learning performance[J]. Computers & Education,2012,58(4):1267-1281.

[9] TOUYA K,FAKIR M. Mining Students' Learning Behavior in Moodle System[J]. Journal of Information Technology Research (JITR),2014,7(4):12-26.

[10] YE C,KINNEBREW J S,SEGEDY J R,et al. Learning Behavior Characterization with Multi-Feature, Hierarchical Activity Sequences[C]// Proceedings of the 8th International Conference on Educational Data Mining. 2015:380-383.

[11] LINAN L C,ANGEL ALEJANDRO JUAN PEREZ. Educational data mining and learning analytics: differences, similarities and time evolution[J]. Ruse Revista De Universidad Y Sociedad Del Conocimiento,2015,12(3):98-112.

[12] DURKSEN T L,CHU M W,AHMAD Z F,et al. Motivation in a MOOC:a probabilistic analysis of online learners' basic psychological needs[J]. Social Psychology of Education,2016, 19(2):241-260.

[13] FITOUSSI J P,VELUPILLAI K. Technology for Mining the Big Data of MOOCs[J]. Research & Practice in Assessment, 2014,9:29-37.

[14] MAC CALLUM K,JEFFREY L. Factors Impacting Teachers' Adoption of Mobile Learning[J]. Journal of Information Technology Education Research,2014,13(13):141-162.

[15] 樊超,宗利永. MOOC 在线学习行为的人类动力学分析[J]. 开放教育研究,2016,22(2):53-58.

[16] 宗阳,孙洪涛,张享国,等. MOOCs 学习行为与学习效果的逻辑回归分析[J]. 中国远程教育,2016,36(5):14-22.

[17] 肖建忠,陈小娟,贾秀险. 高等教育评估多元化研究[J]. 高教探索,2013(1):13-15.

[18] O'CONNOR M C,PAUNONEN S V. Big Five personality predictors of post-secondary academic performance[J]. Personality & Individual Differences,2007,43(5):971-990.

[19] POROPAT A E. A meta-analysis of the five-factor model of personality and academic performance[J]. Psychological Bulletin, 2009,135(2):322-328.

[20] VEDEL A. The Big Five and tertiary academic performance: A systematic review and metaanal-ysis[J]. Personality & Individual Differences,2014,71(2):66-76.

[21] KONTOYIANNIS I,ALGOET P H,SUHOV Y M,et al. Nonparametric entropy estimation for stationary processes and random fields,with applications to English text[J]. IEEE Transactions on Information Theory,1998,44(3):1319-1327.

[22] CAO Y,GAO J,LIAN D,et al. Orderness Predicts Academic Performance: Behavioral Analysis on Campus Lifestyle[J]. eprint arXiv:1704.04013.

[23] TOKTAROVA V I,PANTUROVA A A. Learning and Teaching Style Models in Pedagogical Design of Electronic Educational Environment of the University[OL]. http://www.mcser.org/journal/index.php/mjss/article/view/6874.

[24] 倍智人才研究院. 大五人格心理学:The big five[M]. 北京:企业管理出版社,2015.

[25] PERRY T W. 16-Cattle Finishing Systems[OL]. http://doi.org/10.1016/B978-012552052-2150019-6.

[26] 王晨煜,管明辉,殷传涛,等. 基于 FelderS-ilverman 学习风格模型的网络学习风格研究[J]. 重庆理工大学学报,2017,31(2):102-109.

[27] FREUND Y, MASON L. The Alternating Decision Tree Learning Algorithm[C]// Machine Learning:Sixteenth International Conference. 1999:124-133.

[28] MOZINA M,DEMSAR J,KATTAN M,et al. Nomograms for Visualization of Bayesian Classifier[C]// European Conference on Principles of Data Mining & Knowledge Discovery. 2004: 337-348.