DOI：10.3969/j.issn.2096-8248.2025.01.0007
引用格式：范译文，廖淑娇，吴迪 . 区间值决策表中测试代价敏感的属性约简方法［J］. 江苏海洋大学学报（自然科学版），2025，34（1）：51-61.

区间值决策表中测试代价敏感的属性约简方法

范译文[a, b, c, d]，廖淑娇[a, b, c, d]，吴　迪[a, b, c, d]

（闽南师范大学　a. 数学与统计学院；b. 福建省粒计算及其应用重点实验室；
c. 数字福建气象大数据研究所；d. 数据科学与统计重点实验室，福建 漳州　363000）

摘　要：在当前的大数据时代，数据处理至关重要。代价敏感学习是机器学习、数据挖掘等领域研究热点之一，而测试代价是一种重要的代价，数据处理往往要考虑到测试代价。但是，目前较少有基于测试代价去考虑区间值数据的属性约简。针对该情况，讨论了区间值决策表中测试代价敏感的属性约简问题，创建了相应的粗糙集理论模型，提出了测试代价相关的加权属性重要度函数，并设计了测试代价敏感属性约简的回溯算法和启发式算法。最后在多个 UCI 数据集上进行实验，检验了所提出算法的有效性。该算法相较于现有的两个属性约简算法，在降低总测试代价方面具有显著的优势。回溯算法总是可以得到最优约简，而启发式算法能较高效率地得到最优或次优的约简。

关键词：区间值；决策表；属性约简；测试代价；不一致对象
中图分类号：TP391　　**文献标志码：**A　　**文章编号：**2096-8248（2025）01-0051-11

Test-cost-sensitive attribute reduction method in interval-valued decision tables

FAN Yiwen[a, b, c, d], LIAO Shujiao[a, b, c, d], WU Di[a, b, c, d]

（a. School of Mathematics and Statistics; b. Fujian Key Laboratory of Granular Computing and Application;
c. Institute of Meteorological Big Data-Digital Fujian; d. Key Laboratory of Data Science and Statistics,
Minnan Normal University, Zhangzhou 363000, China）

Abstract: In the current era of big data, data processing is very critical. Cost sensitive learning is one of the current research hot spots in machine learning, data mining and other fields. Test cost is an important type of cost, and data processing often needs to take test costs into account. However, there are few attribute reduction methods that consider interval-valued data based on test costs. In response to this situation, this paper discusses the problem of test-cost-sensitive attribute reduction in interval-valued decision tables, creates a corresponding rough set theory model, presents a test-cost-related weighted attribute significance function, and designs a backtracking algorithm and a heuristic algorithm for test-cost-sensitive attribute reduction. Finally, experiments are conducted on multiple UCI datasets to verify the effectiveness of the proposed algorithms. Compared with the two existing attribute reduction algorithms, the proposed algorithms have great advantages in reducing the total test cost. The backtracking algorithm can always obtain the optimal reduction, and the heuristic algorithm can obtain the optimal or sub-optimal reduction efficiently.

收稿日期：2024-05-16；**修订日期：**2024-08-22
基金项目：国家自然科学基金资助项目（12101289）；福建省自然科学基金面上项目（2024J01800）
作者简介：范译文（1997—），女，硕士研究生，研究方向为粗糙集、粒计算和数据挖掘，（E-mail）1483426583@qq.com。
通信作者：廖淑娇（1981—），女，教授，博士，研究方向为粒计算、数据挖掘和人工智能，（E-mail）sjliao2011@163.com。

Key words: interval value; decision table; attribute reduction; test cost; inconsistent object

0 引言

在当前的大数据时代，面对复杂的海量数据，数据处理变得非常重要。粗糙集是一种刻画不完整和不确定性问题的数学工具，可以有效地对数据进行分析和处理[1]。属性约简是粗糙集研究的重要内容，其核心思想是在保持原始数据特性不变的前提下，选择必要的属性和删除冗余的属性。区间值是一种重要的数据，人们在描述和理解某些数据时，主观上更倾向于一个区间范围，而不是一个确定的数。区间数在现实生活中应用较为广泛，研究区间值数据的属性约简问题具有重要意义。对于区间值数据的属性约简问题，学者们也取得了诸多的成果[2-17]。例如，郭庆等[3]通过计算得到相似度矩阵和等价矩阵，并在区间值的信息系统中进行约简；Du等[4]研究了不一致区间有序决策表中的属性约简方法，提出近似分布约简的概念，并运用辨识矩阵方法列举所有相关约简；Dai等[8]提出一种基于概率分布的优势关系，在这种关系下创建区间值决策信息系统中的粗糙集模型，并进行属性约简；Shu等[9]在不确定度量下研究有缺失值的区间值数据的属性约简问题，并提出有效的属性约简算法；鲍迪等[13]在区间值决策系统中提到单增量和组增量，并给出相关的约简算法。总地来说，这些已有的对区间值数据的属性约简方法主要是在非代价敏感环境下进行的，很少考虑到代价因素。

代价敏感学习是数据挖掘和机器学习等领域中极具研讨性的问题之一[18]，其研究在近十几年来取得重要进展。它是指建立数据挖掘或机器学习模型时，以获得最少的代价函数值为目标。代价一般可以分为测试代价和分类代价等，其中测试代价是指为获取数据值而消耗的时间、钱财，储存数据所需要的贮存容量和传递数据而耗费的带宽等，通常在获取和测试数据过程中都会产生测试代价。因此在现实应用中，也需要关注数据的测试代价。只有在属性约简过程中考虑到测试代价，算法才能更适应实际问题[19]。Min等[20]对名词型数据的测试代价敏感属性约简问题进行了研究，之后一系列相关工作随之展开[21-30]。例如，Min等[21]进一步研究了数值型数据的测试代价敏感属性约简问题，提出了相应的回溯算法和启发式算法；鞠恒荣等[23]为了解决在不完备信息系统中涉及测试代价的属性约简问题，在可变精度分类粗糙集模型中考虑了测试代价，给出新的启发式约简算法；刘偲等[26]在传统正区域约简中考虑到测试代价，采用模拟退火算法得到测试代价总和最少的正区域约简结果；吴迪等[29]研究了多尺度决策系统中测试代价敏感的属性与尺度同步选择，基于测试代价给出属性－尺度重要度函数，并提出属性与尺度同步选择的启发式算法。Lu等[30]采用区间值形式来表示测试费用的可能范围，用区间排序方法构建了区间值型测试代价敏感的属性约简的理论模型，建立了一个与风险态度相关的优化问题，并设计回溯算法和启发式算法来处理该优化问题。总地来说，目前对测试代价敏感属性约简的研究中，有针对数值型数据的，也有针对名词型数据的，但是较少涉及区间值数据。

针对这一现象，本文在区间值决策表中考虑测试代价敏感的属性约简。首先，创建了相应的粗糙集理论模型，并基于测试代价构建了加权的属性重要度函数。其次，在区间值决策表中提出了基于测试代价的属性约简算法，包括回溯算法和启发式算法。最后，通过在数个UCI数据集上进行实验，对两种算法的有效性进行了验证，并与两个相关算法进行了比较。实验结果表明，相较于两个对比算法，本文算法可以很大程度地降低总测试代价，并且本文算法也具有较高的计算效率。

1 理论模型

首先回顾关于区间值决策表的一些粗糙集理论知识；其次，提出不一致对象等相关定义，并探究它们的性质；最后，在区间值决策表中讨论测试代价敏感的属性约简的理论知识。

定义1[2] 设$S=(U, C, D, V=\{V_a|a\in C\cup D\}, I=\{I_a|a\in C\cup D\})$为一个区间值决策表，其中$U=\{x_1, x_2, \cdots, x_{|U|}\}$是非空有限对象的集合，$C=\{a_1, a_2, \cdots, a_m\}$为条件属性集，$D$为决策属性集，$V_a$是属性$a$的值域，$I_a: U\to V_a$。

如表1所示，列举一个区间值决策表的例子。

表 1　一个区间值决策表[31]

Table 1　An interval-valued decision table[31]

x	a_1	a_2	a_3	D
x_1	[22.5, 35.5]	[0, 2]	[0.15, 0.3]	1
x_2	[19, 32]	[0.6, 3.2]	[0.15, 0.24]	1
x_3	[25.5, 63]	[0.84, 4.97]	[0.14, 0.22]	1
x_4	[19.2, 31]	[0.14, 7]	[0.09, 0.17]	2
x_5	[13.7, 25]	[1, 15]	[0.05, 0.28]	2
x_6	[13, 20.5]	[0, 0.4]	[0, 0]	2

定义 2[2]　给定两个区间数 $I_1=[l_1, r_1]$ 和 $I_2=[l_2, r_2]$，定义 I_1 大于 I_2 的概率为

$$P_{12}=\min\left\{1, \max\left\{\frac{r_1-l_2}{(r_1-l_1)+(r_2-l_2)}, 0\right\}\right\}。\quad (1)$$

根据这一度量，给定 I_1 和 I_2 之间的相似性度量，即

$$s(I_1, I_2)=1-|P_{12}-P_{21}|,\quad (2)$$

则可以得到相应的邻域。

定义 3[2]　设 $S=(U, C, D, V, I)$ 为一个区间值决策表，$B\subseteq C$，$\delta\in(0, 1]$，则 $\forall x\in U$ 关于 B 和 δ 的邻域为

$$N_B^\delta(x)=\{y\in U: s_a(x, y)\geqslant\delta, \forall a\in B\},\quad (3)$$

其中 $s_a(x, y)$ 与公式(2)中的含义相同，$\delta\in(0, 1]$ 被称为相似水平。

区间值决策表中一些重要的粗糙集概念如下。

定义 4[2]　设 $S=(U, C, D, V, I)$ 为一个区间值决策表，$U/D=\{D_1, D_2, \cdots, D_m\}$，$B\subseteq C$，$\delta\in(0, 1]$，$D_j\in U/D$，则 D_i 关于 B 和 δ 有以下上近似和下近似的定义为

$$\overline{R_B^\delta}(D_i)=\{x\in U: N_B^\delta(x)\cap D_i\neq\varnothing\},\quad (4)$$

$$\underline{R_B^\delta}(D_i)=\{x\in U: N_B^\delta(x)\subseteq D_i\},\quad (5)$$

其中 $j=1, 2, \cdots, m$。

定义 5[2]　设 $S=(U, C, D, V, I)$ 为一个区间值决策表，$U/D=\{D_1, D_2, \cdots, D_m\}$，$B\subseteq C$，$\delta\in(0, 1]$，则 D 关于 B 和 δ 的上、下近似为

$$\overline{R_B^\delta}(D)=\cup_{i=1}^m\overline{R_B^\delta}(D_i),\quad (6)$$

$$\underline{R_B^\delta}(D)=\cup_{i=1}^m\underline{R_B^\delta}(D_i)。\quad (7)$$

定义正区域为

$$\mathrm{POS}_B^\delta(D)=\underline{R_B^\delta}(D)。\quad (8)$$

边界区域为

$$\mathrm{BR}_B^\delta(D)=\overline{R_B^\delta}(D)-\underline{R_B^\delta}(D)。\quad (9)$$

这些概念的关系如下：

(1) $\overline{R_B^\delta}(D)=U$；

(2) $\mathrm{POS}_B^\delta(D)\cap\mathrm{BR}_B^\delta(D)=\varnothing$；

(3) $\mathrm{POS}_B^\delta(D)\cup\mathrm{BR}_B^\delta(D)=U$。

通过这些关系可以得到

$$\mathrm{BR}_B^\delta(D)=U-\mathrm{POS}_B^\delta(D)。\quad (10)$$

事实上，正区域中对象的邻域可以区分为一类，而边界区域中对象的邻域区分为两类或者更多类，也就是说，后者的邻域中存在决策值不同的对象，由此提出下面这个定义。

定义 6　设 $S=(U, C, D, V, I)$ 为一个区间值决策表，$x\in U$，$B\subseteq C$，$\delta\in(0, 1]$，对于任意 $y\in N_B^\delta(x)$，若 $D(y)\neq D(x)$，则 y 被称为一个不一致对象，不一致对象的集合为

$$\mathrm{ic}_B^\delta(x)=\{y\in N_B^\delta(x)\,|\,D(y)\neq D(x)\}。\quad (11)$$

其中不一致对象的数量为 $|\mathrm{ic}^\delta{}_B(x)|$。

定义 7[2]　设 $S=(U, C, D, V, I)$ 为一个区间值决策表，$B\subseteq C$，$\delta\in(0, 1]$，D 关于 B 和 δ 的分类质量为

$$\gamma_B^\delta(D)=|\mathrm{POS}_B^\delta(D)|/|U|。\quad (12)$$

因为 $\mathrm{POS}^\delta{}_B(D)\subseteq U$，则有 $\gamma^\delta{}_B(D)\in(0, 1]$，而当 $\gamma_B^\delta(D)=1$ 时，则称 D 完全依赖于 B。

例 1　设区间值决策表如表 1 所示。可以看出 $U=\{x_1, x_2, \cdots, x_6\}$，$C=\{a_1, a_2, a_3\}$，则依据决策属性集对论域 U 进行划分，分为 $U/D=\{x_1, x_2\}$，$X_1=\{x_1, x_2, x_3\}$，$X_2=\{x_4, x_5, x_6\}$，令 $\delta=0.65$，依据定义 3 计算出对象关于不同属性子集的邻域，表 2 列举了部分结果。

表 2　不同测试集上对象的邻域

Table 2　Neighborhoods of objects on different test sets

X	$\{a_1, a_2\}$	$\{a_1, a_3\}$	$\{a_2, a_3\}$
x_1	$\{x_1, x_3, x_5, x_6\}$	$\{x_1, x_3, x_6\}$	$\{x_1, x_3, x_4, x_6\}$
x_2	$\{x_2, x_3, x_5, x_6\}$	$\{x_2, x_6\}$	$\{x_2, x_4, x_6\}$
x_3	$\{x_1, x_2, x_3, x_4, x_5, x_6\}$	$\{x_1, x_3, x_4, x_6\}$	$\{x_1, x_3, x_4, x_6\}$
x_4	$\{x_3, x_4, x_6\}$	$\{x_3, x_4, x_6\}$	$\{x_1, x_2, x_3, x_4, x_6\}$
x_5	$\{x_1, x_2, x_3, x_5\}$	$\{x_5\}$	$\{x_5, x_6\}$
x_6	$\{x_1, x_2, x_3, x_4, x_6\}$	$\{x_1, x_2, x_3, x_4, x_6\}$	$\{x_1, x_2, x_3, x_4, x_5, x_6\}$

同样地，根据定义 6 可以得到不同测试集上不一致对象的集合，表 3 列举了部分结果。

表 3　不同测试集上的不一致对象集

Table 3　Inconsistent object sets on different test sets

X	$\{a_1, a_2\}$	$\{a_1, a_3\}$	$\{a_2, a_3\}$
x_1	$\{x_5, x_6\}$	$\{x_6\}$	$\{x_4, x_6\}$
x_2	$\{x_5, x_6\}$	$\{x_6\}$	$\{x_4, x_6\}$
x_3	$\{x_4, x_5, x_6\}$	$\{x_4, x_6\}$	$\{x_4, x_6\}$
x_4	$\{x_3\}$	$\{x_3\}$	$\{x_1, x_2, x_3\}$
x_5	$\{x_1, x_2, x_3\}$	$\varnothing$	$\varnothing$
x_6	$\{x_1, x_2, x_3\}$	$\{x_1, x_2, x_3\}$	$\{x_1, x_2, x_3\}$

再根据定义4可以得到不同测试集上对象子集 的近似集,表4列举了相应的结果。

表4 不同测试集上对象子集的近似集

Table 4 Approximate sets of object subsets on different test sets

近似集		B		
	X	$\{a_1, a_2\}$	$\{a_1, a_3\}$	$\{a_2, a_3\}$
$\underline{R_B^\delta}(X)$	X_1	$\varnothing$	$\varnothing$	$\varnothing$
	X_2	$\varnothing$	$\{x_5\}$	$\{x_5\}$
$\overline{R_B^\delta}(X)$	X_1	$\{x_1, x_2, x_3, x_4, x_5, x_6\}$	$\{x_1, x_2, x_3, x_4, x_6\}$	$\{x_1, x_2, x_3, x_4, x_6\}$
	X_2	$\{x_1, x_2, x_3, x_4, x_5, x_6\}$	$\{x_1, x_2, x_3, x_4, x_5, x_6\}$	$\{x_1, x_2, x_3, x_4, x_5, x_6\}$

从表4中可以得出,在不同测试集上,决策属性集D关于条件属性子集B和相似水平δ的正区域和边界区域以及分类质量分别为$\mathrm{POS}^\delta_{\{a_1, a_2\}}(D)=\varnothing$,$\mathrm{POS}^\delta_{\{a_1, a_3\}}(D)=\{x_5\}$,$\mathrm{POS}^\delta_{\{a_2, a_3\}}(D)=\{x_5\}$,$\mathrm{BR}^\delta_{\{a_1, a_2\}}(D)=U$,$\mathrm{BR}^\delta_{\{a_1, a_3\}}(D)=\mathrm{BR}^\delta_{\{a_2, a_3\}}(D)=\{x_1, x_2, x_3, x_4, x_6\}$,$\gamma^\delta_{\{a_1, a_2\}}(D)=0$,$\gamma^\delta_{\{a_1, a_3\}}(D)=\gamma^\delta_{\{a_2, a_3\}}(D)=\frac{1}{6}$。

例1表明了以上所涉及的概念之间的关系,不难得到以下性质。

性质8 全集U中所有不一致对象集的并集为边界区域,即

$$\mathrm{BR}_B^\delta(D)=U_{x\in U}\mathrm{ic}_B^\delta(x)。\tag{13}$$

设$S=(U, C, D, V, I)$为一个区间值决策表,$B\subseteq C$,则关于$\mathrm{POS}_C(D)$的不一致对象总数可表示为

$$\mathrm{pc}_B(S)=\sum_{x\in\mathrm{POS}_C(D)}|\mathrm{ic}_B(x)|。\tag{14}$$

约简是粗糙集中的基本概念,文献[2]中给出了区间值决策表中约简的定义如下。

定义9[2] 设$S=(U, C, D, V, I)$为一个区间值决策表,$B\subseteq C$,$\delta\in(0, 1]$,如果B满足以下两个条件,则称B为一个约简。

(1) $\mathrm{POS}_B^\delta(D)=\mathrm{POS}_C^\delta(D)$;

(2) $\mathrm{POS}_{B-\{a\}}^\delta(D)\neq\mathrm{POS}_C^\delta(D)$,$\forall a\in B$。

条件(1)为充分条件,说明属性子集B和属性全集C具有一样的分辨能力,此时,B是一个超级约简;条件(2)为必要条件,表明B中每个属性对于保持分辨能力都不可缺少。并且可以知道,$\forall B\subseteq C$,$x\in\mathrm{POS}_B^\delta(D)$当且仅当$\mathrm{ic}_B^\delta(x)=\varnothing$。由此能够得到下面的定义。

定义10 设$S=(U, C, D, V, I)$为一个区间值决策表,当$B\subseteq C$,$\delta\in(0, 1]$,满足以下条件时,B为一个约简。

(1) $\forall x\in\mathrm{POS}_C^\delta(D)$,$\mathrm{ic}_B^\delta(x)=\varnothing$;

(2) $\forall a\in B$,$\exists x\in\mathrm{POS}_C^\delta(D)$,s.t. $\mathrm{ic}_B^\delta(x)\neq\varnothing$。

其中,当第一个条件满足的时候,B是一个超级约简,第二个条件表明B中每个属性对于保持分辨能力都不可缺少。还可以得到下面这个定理。

定理11 设$S=(U, C, D, V, I)$为一个区间值决策表,$B\subsetneq C$,$a\in C-B$,$\delta\in(0, 1]$,如果$\forall x\in\mathrm{POS}_C^\delta(D)-\mathrm{POS}_B^\delta(D)$,$\mathrm{ic}_B^\delta(x)=\mathrm{ic}_{B\cup\{a\}}^\delta(x)$,则$\forall B'$,当$B\cup\{a\}\subseteq B'\subseteq C$时,$B'$不是一个约简。

证明 给定B'满足$B\cup\{a\}\subseteq B'\subseteq C$,当$\mathrm{POS}_{B'}^\delta(D)\neq\mathrm{POS}_C^\delta(D)$时,$B'$不是一个约简。现在假定$\mathrm{POS}_{B'}^\delta(D)=\mathrm{POS}_C^\delta(D)$,令$x\in\mathrm{POS}_{B'}^\delta(D)$,可得$\mathrm{ic}_{B'-\{a\}}^\delta(x)=\mathrm{ic}_{B'-\{a\}-B}^\delta(x)\cap\mathrm{ic}_B^\delta(x)=\mathrm{ic}_{B'-\{a\}-B}^\delta(x)\cap\mathrm{ic}_{B\cup\{a\}}^\delta(x)=\mathrm{ic}_{B'}^\delta(x)$,根据定义10,得出$B'$不是约简。

根据定义6以及文献[32-33]可以得到所介绍的这些概念的单调性,如以下两个定理所示。

定理12(关于属性子集的单调性) 设$S=(U, C, D, V, I)$为一个区间值决策表,$U/D=\{D_1, D_2, \cdots, D_m\}$,$X\subseteq U$,$B_1\subseteq B_2\subseteq C$,$\delta\in(0, 1]$,则有

(1) $\forall x\in U$,$N_{B_1}^\delta(x)\supseteq N_{B_2}^\delta(x)$,$\mathrm{ic}_{B_1}^\delta(x)\supseteq\mathrm{ic}_{B_1}^\delta(x)$;

(2) $\forall X\subseteq U$,$\underline{R_{B_1}^\delta}(X)\subseteq\underline{R_{B_2}^\delta}(X)$,$\overline{R_{B_1}^\delta}(X)\supseteq\overline{R_{B_2}^\delta}(X)$;

(3) $\underline{R_{B_1}^\delta}(D)\subseteq\underline{R_{B_2}^\delta}(D)$,$\overline{R_{B_1}^\delta}(D)\supseteq\overline{R_{B_1}^\delta}(D)$;

(4) $\mathrm{POS}_{B_1}^\delta(D)\subseteq\mathrm{POS}_{B_2}^\delta(D)$,$\gamma_{B_1}^\delta(D)\leqslant\gamma_{B_2}^\delta(D)$;

(5) $\mathrm{BR}_{B_1}^\delta(D)\supseteq\mathrm{BR}_{B_2}^\delta(D)$。

定理13(关于相似水平的单调性) 设$S=(U, C, D, V, I)$为一个区间值决策表$U/D=\{D_1, D_2, \cdots, D_m\}$,$X\subseteq U$,$B\subseteq C$,$\delta_1\leqslant\delta_2$,则

(1) $\forall x\in U$,$N_B^{\delta_1}(x)\supseteq N_B^{\delta_2}(x)$,$\mathrm{ic}_B^{\delta_1}(x)\supseteq\mathrm{ic}_B^{\delta_2}(x)$;

(2) $\forall X\subseteq U$,$\underline{R_B^{\delta_1}}(X)\subseteq\underline{R_B^{\delta_2}}(X)$,$\overline{R_B^{\delta_1}}(X)\supseteq\overline{R_B^{\delta_2}}(X)$;

(3) $\underline{R_B^{\delta_1}}(D)\subseteq\underline{R_B^{\delta_2}}(D)$,$\overline{R_B^{\delta_1}}(D)\supseteq\overline{R_B^{\delta_2}}(D)$;

(4) $\mathrm{POS}_B^{\delta_1}(D)\subseteq\mathrm{POS}_B^{\delta_2}(D)$,$\gamma_B^{\delta_1}(D)\leqslant\gamma_B^{\delta_2}(D)$;

(5) $BR_B^{\delta_1}(D) \supseteq BR_B^{\delta_2}(D)$。

前面提到的区间值决策表的相关知识中没有考虑到测试代价, 而 Min 等[20]定义了测试代价敏感的属性约简问题, 给出下面的定义。

定义 14[20]　称 $S=(U, C, D, V, I, \mathrm{tc})$ 为一个测试代价敏感的决策表, 其中, U, C, D, V, I 如定义 1 中所示, tc 为测试代价向量, 而向量中的元素 $\mathrm{tc}(a)$ 就为属性 a 的测试代价。

如表 5 所示, 列举一个测试代价向量的例子。

表 5　测试代价向量
Table 5　Test cost vector

a	a_1	a_2	a_3
tc	\$2	\$14	\$40

一个区间值决策表和相应的测试代价向量就组成一个测试代价敏感的区间值决策表。

令 $\mathrm{TTC}(R)$ 表示属性子集 R 对应的总测试代价 (total test cost, 简写为 TTC)。

$$\mathrm{TTC}(B)=\sum_{a_i\in B}\mathrm{tc}(a_i)。\tag{15}$$

定义 λ 加权的属性重要度函数为

$$\mathrm{sig}(B, a, \mathrm{tc}(a))=(|\mathrm{pc}_B(S)|-|\mathrm{pc}_{B\cup\{a\}}(S)|)[\mathrm{tc}(a)]^{\lambda},\tag{16}$$

其中 $\mathrm{pc}_B(S)$ 如公式 (14) 所示, $\mathrm{tc}(a)$ 为属性 a 的测试代价, $\lambda\leqslant 0$ 是一个用户指定的参数, 当 λ=0 时, 测试费用本质上可以不去考虑, 当 λ<0 时, 测试代价较低具有较大意义。使用 λ 加权的属性重要度函数, 使得属性约简比较倾向于选中分辨能力较大且测试代价较小的属性。

2　算法设计和评价指标

针对区间值决策表的测试代价敏感属性约简问题, 设计回溯算法和启发式算法两种算法, 并介绍算法的多个评判指标和启发式算法的竞争策略。

2.1　算法设计

算法 1 和算法 2 分别显示了回溯算法和启发式算法的框架。

算法 1 为回溯算法, 算法开始之前先初始化几个全集变量 $B=\varnothing$, $R=C$, $l=0$, $U'=\mathrm{POS}_C^{\delta}(D)$。

第一步, 判断属性 a_i 的测试代价是否过高, 过高时就进行剪枝, 如算法 1 的第 2 行到第 4 行所示。

第二步, 根据 U' 中每个对象关于属性子集 $B\cup\{a_i\}$ 的不一致对象集的情况, 计算当前的边界对象集 $\mathrm{BR}^{\delta}_{B\cup\{a_i\}}(D)$, 如算法 1 的第 5 行到第 14 行所示。

第三步, 判断添加属性后 a_i 是否使不一致对象减少。如果没有减少则进行剪枝, 如果减少则继续判断 $B\cup\{a_i\}$ 是否是超级约简。如果 $B\cup\{a_i\}$ 相应的边界对象集为空集, 则得到一个超级约简, 再进行下一层回溯, 如算法 1 的第 15 行到第 24 行所示。

算法 1: 回溯约简算法

输入: 测试代价敏感的区间值决策表 $S=(U, C, D, V, I, \mathrm{tc})$ 选择的测试属性集 B, 上一层的边界对象集 U', 当前层的属性指标下界 l。

输出: 一个最优约简 R。

```
1: for (i=l; i<|C|; i++) do
2:   if (tc(B∪{a_i}) ≥ tc(R)) then
3:     continue; /* 剪枝, 摒弃测试代价过高的属性 */
4:   end if
5:   lessInconsistent=false;
6:   BR^δ_{B∪{a_i}}(D)=∅; /*BR^δ_{B∪{a_i}}(D)为当前的边界对象集 */
7:   for (each x∈U') do
8:   if (ic^δ_{B∪{a_i}}(x) ⊂ ic^δ_B(x)) then
9:     lessInconsistent=true;
10:    if (ic^δ_{B∪{a_i}}(x) ≠ ∅) then
11:      BR^δ_{B∪{a_i}}(D)=BR^δ_{B∪{a_i}}(D) ∪ {x};
12:    end if
13:  end if
14:  end for
15:  if (lessInconsistent==false) then
16:    continue; /*a_i 没办法使不一致对象减少, 剪枝 */
17:  else
18:    if (BR^δ_{B∪{a_i}}(D) == ∅) then
19:      R=B∪{a_i};
20:    continue; /* 得到一个更好的超级约简, 剪枝 */
21:    else
22:      backtrack(BR^δ_{B∪{a_i}}(D), B∪{a_i}, i+1); /* 下一层回溯 */
23:    end if
24:  end if
25: end for
```

算法 1 中使用了 4 次剪枝技术, 提升了算法效

率。首先,如行1所示,回溯算法的搜索路径中,属性指标的初始值不都是从0开始,而是随着回溯算法的进行不断增加,这样能够减少搜索的次数;其次,如行2至行4所示,对测试代价太高的属性进行剪枝;再次,如行15至行16所示,当添加属性 a_i 不能使不一致对象减少时进行剪枝;最后,如行18至行20所示,当 $B\cup\{a_i\}$ 是一个超级约简时则进行剪枝。这4次剪枝技术明显使得算法的效率得到了提高。

算法2是启发式算法,在算法2中运用了添加-删除策略,主要有3个步骤。

第一步,初始化约简 $B=\varnothing$,如算法2的第1行所示。

第二步,添加阶段,在属性子集 B 中添加属性重要度最大的待选属性,如算法2的第2行到第7行所示。

第三步,删除阶段,将冗余的属性删除,最后得到一个约简,如算法2的第8行到第14行所示。

算法2: 启发式约简算法

输入:测试代价敏感的区间值决策表 $S=(U, C, D, V, I, \mathrm{tc})$

输出:一个约简 B

1: $B=\varnothing$; /* 添加步骤 */
2: $CA=C$;
3: while $(\gamma_B^\delta(D)<\gamma_C^\delta(D))$ do
4: 计算每个属性 $a\in CA$ 的 $\mathrm{sig}(B, a, tc)$;
5: 选择属性 a',使其满足 $\mathrm{sig}(B, a', \mathrm{tc})=\max_{a\in CA}\{\mathrm{sig}(B, a, \mathrm{tc})\}$
6: $B=B\cup\{a'\}$; $CA=CA-\{a'\}$;
7: end while; /* 删除步骤 */
8: $CD=B$;
9: while $(CD\neq\varnothing)$ do
10: $CD=CD-\{a'\}$;
11: if $(\mathrm{POS}_{B-\{a'\}}^\delta(D)=\mathrm{POS}_B^\delta(D))$ then
12: $B=B-\{a'\}$;
13: end if
14: end while
15: return B;

算法2中,关键步骤为行4和行5,可以通过设置不同的 λ 值得到不同的约简结果。由于在算法2中基于加权的属性重要度函数采取了贪婪的添加属性策略,又在此基础上删除了冗余的属性,这使得启发式算法能较高效率地得到最优或次优的约简。

这里分析两个算法的时间复杂度。算法1中,每次选取属性平均可以将 $|U|/|B|$ 个对象划分到正区域,随着算法1中第1行for循环的进行,

$$|U'|=|U|\left(1-\frac{i}{|B|}\right)=$$
$$|U|\frac{|B|-i}{|B|}\ (i=0, 1, 2, \cdots, |B|-1, |B|)。$$

故算法1的时间复杂度为

$$O\left(|C||U|\cdot1\cdot4+|C||U|\frac{|B|-1}{|B|}\cdot2\cdot4+\right.$$
$$\left.|C||U|\frac{|B|-2}{|B|}\cdot3\cdot4+\cdots+|C||U|\frac{1}{|B|}\cdot|B|\cdot4\right)=$$
$$O\left(\frac{|C||U|}{|B|}\sum_{i=0}^{|B|-1}(i+1)(|B|-i)\right)。$$

算法2的主要步骤为添加步骤,删除步骤的计算复杂度可忽略不计。在添加步骤中,随着while循环的进行,属性子集 B 中的属性逐渐增多,CA 属性逐渐减少。故算法2的计算复杂度为

$$O\left((|C|-1)|U|\cdot1\cdot6+(|C|-2)|U|\cdot2\cdot6+\right.$$
$$\left.\cdots+(|C|-|B|)|U||B|\cdot6\right)=O\left(|U|\sum_{i=0}^{|B|}i(|C|-i)\right)。$$

2.2 评判指标

为了衡量所设计算法的性能,在后文的实验中用了多个评判指标,部分介绍如下。

首先,定义约简的属性占比(attribute proportion of reduct)为

$$\mathrm{ap}(R)=\frac{|R|}{|C|},\tag{17}$$

其中 R 表示约简,C 表示属性全集。

定义约简的测试代价占比(test cost proportion of reduct)为

$$\mathrm{tcp}(R)=\frac{\mathrm{tc}(R)}{\mathrm{tc}(C)},\tag{18}$$

其中 R 表示约简,C 表示属性全集。

此外,由于回溯算法总是可以得到最优约简,而启发式算法未必,所以,为了衡量启发式算法的性能,Min等[20]提出3种不同的统计评价指标,如下所示。

寻优因子(finding optimal factor,简写为FOF)定义为

$$\mathrm{FOF}=\frac{k}{K},\tag{19}$$

其中,K 为测试的次数,k 是启发式算法搜索得到最优约简的次数。

假设 R' 和 R 分别是一个数据集在一组测试代价下得到的最优约简和约简，则相对于 R'，R 的超出因子为

$$\mathrm{ef}(R)=\frac{\mathrm{tc}(R)-\mathrm{tc}(R')}{\mathrm{tc}(R')}, \tag{20}$$

表明当一个约简不是最优时，它的总测试代价超出最小值的程度。显然，当 R 为最优约简时，超出因子值为零。

假设实验次数为 L，第 $j(1 \leqslant j \leqslant L)$ 次实验得到的约简为 R_j，则最大超出因子（maximal exceeding factor，简写为 MEF）为

$$\max_{1\leqslant j\leqslant L} \mathrm{ef}(R_j) \text{ 。} \tag{21}$$

平均超出因子（average exceeding factor，简写为 AEF）为

$$\frac{\sum_{j=1}^{L}\mathrm{ef}(R_j)}{L} \text{ 。} \tag{22}$$

总地来说，当 FOF 越大，同时 MEF 和 AEF 越小时，启发式算法在最小化总测试代价方面的性能更好。

2.3 竞争策略

为了获得更好的约简结果，Min 等[20]对启发式算法提出了配套的竞争策略。具体地，令 R_λ 为启发式算法在指数 λ 下得到的约简，L 为一组用户指定的 λ 值，则最小总测试代价为

$$\mathrm{TTC}_L=\min_{\lambda\in L}\ \mathrm{TTC}(R_\lambda)。 \tag{23}$$

即启发式算法在不同的 λ 值下共运行 $|L|$ 次，得到相应 λ 值下约简的总测试代价，再通过比较得到总测试代价最小的约简。

虽然在使用竞争策略过程中，对启发式算法进行了 $|L|$ 次运行，但是因为该算法的运行速度较快，所以可以接受相对较小的 $|L|$ 次。

3 实验结果与分析

通过实验回答以下几个问题：① 本文算法是否能解决区间值数据的最小测试代价约简问题？② 两种算法的效率和效果怎么样？③ 竞争策略能否提高启发式算法结果的质量？④ 相比已有的算法，本文算法是否具有优越性？

3.1 实验准备

为了验证所提出的约简算法的效果，在 11 个数据集上进行了实验检测，数据集是从 UCI 数据库[34]中获得。表 6 展示了实验数据的情况。实验中令相似水平 δ=0.65。

表 6　实验数据介绍

Table 6　Introduction of experimental data

数据集	领域	样本数量	属性个数	类数
Iris	Zoology	150	4	3
Wine	Agriculture	178	13	3
Glass	Manufacture	214	9	7
Iono	Physics	351	34	2
Sonar	Physics	208	60	2
Diab	Clinic	768	8	2
Liver	Clinic	345	6	2
Wdbc	Clinic	569	30	2
Wpbc	Clinic	198	33	2
QSAR	Biology	1 055	41	2
Image	Graphics	2 310	18	7

由于实验数据集是实值型数据，所以在实验前，首先用公式

$$a'(x)=[a(x)-\xi^*\sigma, a(x)+\xi^*\sigma]。 \tag{24}$$

对实数数据进行区间化，其中，$a(x)$ 为对象 x 关于属性 a 的属性值，σ 为属性 a 的属性值的标准差，ξ 是一个参数。取 ξ=0.1，转换对象 x 在属性 a 下的属性值，从而得到区间数 $a'(x)$。

由于大部分 UCI 数据不带有测试代价，出于实验需要，Min 等[20]根据现实情况在均匀分布下生成数据的测试代价，取值范围为 [1, 100]，且取整数。

规定 C_u 表示生成的测试代价，则 C_u=p 概率为

$$P(C_u=p)=\frac{1}{Q-P+1}, \tag{25}$$

其中 $p\in[P, Q]$。令 x 为 (0, 1) 上满足均匀分布的任意值，则测试代价为

$$C_u(P, Q, x)=P+[(Q-P+1)x]。 \tag{26}$$

3.2 代表性结果与分析

如表 7 所示，给出 11 个数据集分别运用两种算法得到的代表性结果。从表 7 中可以看出，所有数据集在两个算法下得到的约简都具有较低的属性占比值（约简后的属性个数 / 属性全集基数）和测试代价占比值（约简后的总测试代价 / 属性全集的总测试代价），有的值甚至低至 0.1 以下。这说明不管是回溯算法还是启发式算法，都能很好地起到降低属性维度和减少总测试代价的效果。

表7 两种算法分别的代表性结果
Table 7 Representative results of the two algorithms

数据集	算法	约简	属性占比值	测试代价占比值
Iris	回溯	{1, 2, 4}	0.750 0	0.654 7
	启发式	{1, 2, 4}	0.750 0	0.654 7
Wine	回溯	{9, 10}	0.153 8	0.067 8
	启发式	{5, 10, 12}	0.230 8	0.086 8
Glass	回溯	{3, 4, 5}	0.333 3	0.239 7
	启发式	{3, 4, 5}	0.333 3	0.239 7
Iono	回溯	{10, 11, 12, 27}	0.117 6	0.031 1
	启发式	{10, 11, 13, 24, 27}	0.147 1	0.035 5
Sonar	回溯	{44, 52, 57}	0.050 0	0.006 4
	启发式	{44, 52, 57}	0.050 0	0.006 4
Diab	回溯	{3, 6, 7, 8}	0.500 0	0.292 7
	启发式	{3, 6, 7, 8}	0.500 0	0.292 7
Liver	回溯	{1, 2, 3}	0.500 0	0.397 8
	启发式	{1, 2, 3}	0.500 0	0.397 8
Wdbc	回溯	{3, 17, 19, 29}	0.133 3	0.025 4
	启发式	{3, 17, 19, 29}	0.133 3	0.025 4
Wpbc	回溯	{4, 15, 27}	0.090 9	0.024 2
	启发式	{4, 15, 27}	0.090 9	0.024 2
QSAR	回溯	{1, 13, 22, 30, 34}	0.122 0	0.088 5
	启发式	{2, 13, 18, 22}	0.097 6	0.092 3
Image	回溯	{3}	0.052 6	0.054 7
	启发式	{1, 6, 11, 14, 18}	0.263 2	0.146 9

3.3 两种方法的比较

对两种方法进行比较，这两种方法都是基于算法2的。第一种方法被称为非加权方法，通过设置$\lambda=0$来实现；第二种方法就是“2.3”节介绍的竞争策略。令$L=\{0, -0.25, \cdots, -1.75, -2.0\}$，两种方法下的结果如表8所示。

表8 $\lambda=0$和$\lambda\in L$下分别的结果
Table 8 Results under $\lambda=0$ and $\lambda\in L$

数据集	寻优因子		最大超出因子		平均超出因子	
	$\lambda=0$	$\lambda\in L$	$\lambda=0$	$\lambda\in L$	$\lambda=0$	$\lambda\in L$
Iris	0.24	1.00	1.71	0	0.31	0
Wine	0.12	0.92	13.75	0.28	1.94	0.01
Glass	0.11	0.84	3.12	0.41	0.72	0.02
Iono	0	0.40	255.00	73.00	15.38	2.31
Sonar	0	0.58	11.19	0.20	3.55	0.03
Diab	0.05	0.78	5.23	0.24	0.59	0.02
Liver	0.40	0.87	1.66	0.23	0.25	0.01
Wdbc	0	0.77	23.44	0.37	5.69	0.03
Wpbc	0	0.93	31.40	0.26	8.80	0.01
QSAR	0	0.35	2.20	0.35	0.90	0.08
Image	0	0	193.00	64.00	19.37	9.07

通过表8可以看出：① 非加权的启发式约简方法较少能得到最优约简，这是因为它的重要度函数没有考虑到测试代价，违背了最小化总测试代价的初衷。② 竞争策略显著提高了启发式算法约简结果的质量，尤其是当算法较难获得最优约简的时候。例如Wine数据集，表8中可以看出$\lambda\in L$时，FOF为0.92，比$\lambda=0$时增加了0.80，大大提高了FOF，同时大大减小了MEF和AEF。这是因为$\lambda\in L$时考虑了测试代价，并运用了竞争策略，而$\lambda=0$时没有考虑到测试代价，更没有考虑到竞争策略。

3.4 两个算法的效率

用两个指标说明回溯算法的效率，一个是程序执行的次数，即回溯算法的调用次数，该指标能够用于研究剪枝技术的效果；另一个是算法之间运行时间的比较，在本文中将两种算法的运行时间进行比较，其中λ设置为$[-2, 0]$。

先考察第一个指标。在11个数据集上都采取100组不同的测试代价进行属性约简，算法1中状态空间树的大小和回溯算法的调用次数在表9中表示。从表9中可以看出回溯算法的步数远远小于状态空间树的大小，这验证了剪枝技术的有效性。

表9 算法1中状态空间树的大小和回溯算法的调用次数
Table 9 Size of the state space tree in algorithm 1 and the invoking number of the backtracking algorithm

数据集	状态空间树的大小	最小步数	最大步数	平均步数
Iris	2^4	7	11	9.47
Wine	2^{13}	7	54	21.45
Glass	2^9	10	90	40.5
Iono	2^{34}	2	1 000	188.25
Sonar	2^{60}	9 048	205 819	44 710.13
Diab	2^8	24	118	72.96
Liver	2^6	8	41	24.43
Wpbc	2^{33}	947	1 563	1 038.67
Wdbc	2^{30}	18	209	64.66
QSAR	2^{41}	4 205	1 002 967	126 227.4
Image	2^{18}	3	2 522	201.73

第二个指标是两个算法的运行时间的比较。如图1所示，绘制出$\lambda=-1$时给定的11个数据集在100组不同的测试代价下的平均运行时间和最大运行时间。

从图1可以看出，对于部分数据集来说，算法1比算法2的运行时间高，而其他数据集中算法2的运行时间反而比较高。这是因为在算法1即回溯算法中运用了4次剪枝技术，从而较大程度地减少了该算法运行的时间。此外，虽然在小数据集上算法2即启发式算法的效率优势不明显，但是在属性比较多的数据集例如Sonar和QSAR等数据集上该算法的效率优势就很明显。

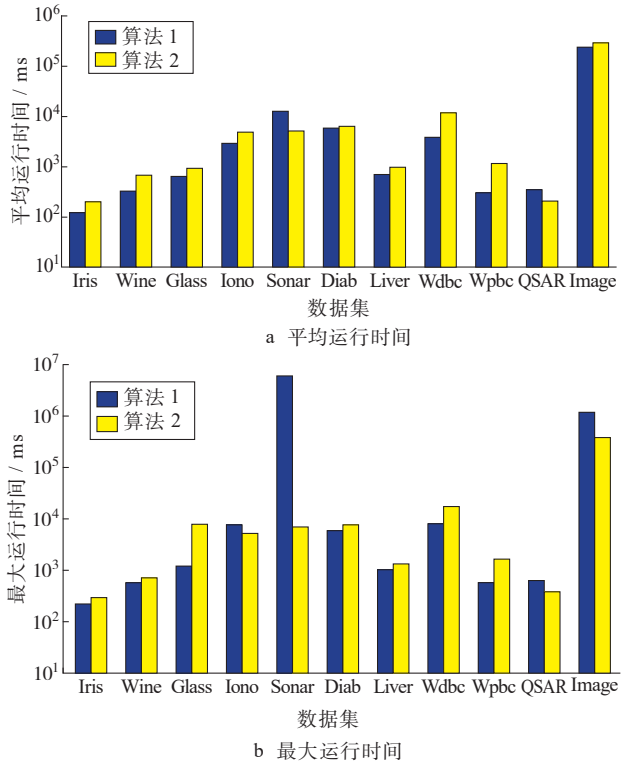

图 1　运行时间比较

Fig. 1　Run-time comparison

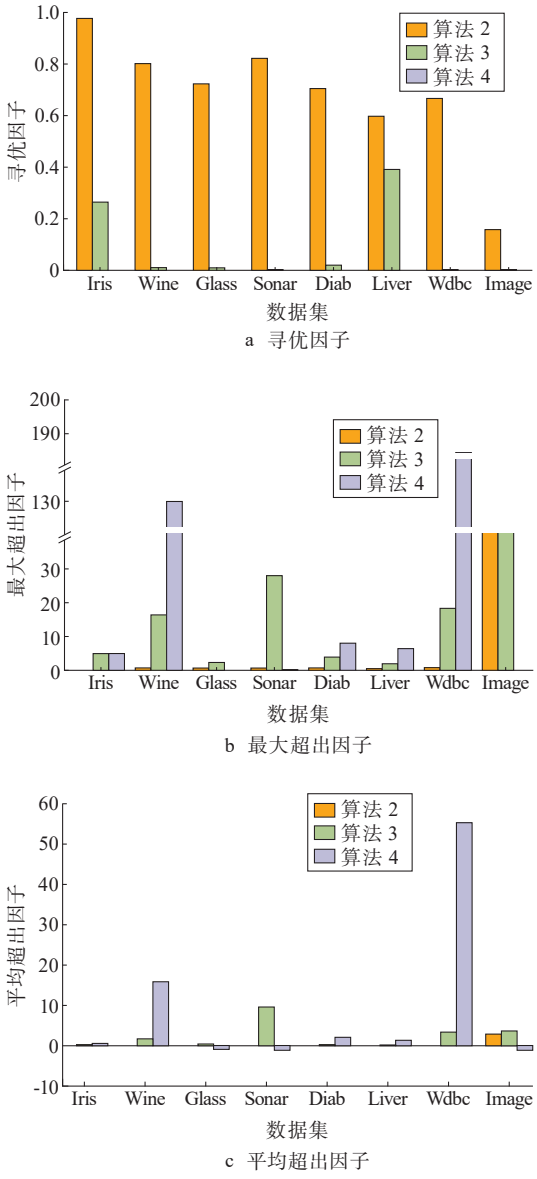

图 2　3 个指标的比较

Fig. 2　Comparison of three metrics

总地来说，回溯算法总是可以得到最优约简，即最小化总测试代价的约简，从而能够运用回溯算法评判启发式算法的性能；而启发式算法能较高效率地得到最优或次优的约简。

3.5　与两种相关算法的比较

为了更客观地评价本文算法，选取两个在决策信息系统下分别基于属性质量度和条件熵进行属性约简的启发式算法进行对比试验。这两个算法分别选自文献[11]和[35]，把本文中的回溯算法记为算法 1，文中的启发式算法记为算法 2，把文献[11]中的算法记为算法 3（算法 3 通过定义属性质量度来判断属性是否能纳入约简），把文献[35]中的算法记为算法 4（算法 4 通过定义条件熵，引入属性重要度概念进行属性约简）。在对比实验中，比较了本文算法 2 和算法 3 以及算法 4 中的 FOF，MEF 和 AEF 的结果；还比较了 4 个算法的运行时间，约简长度的大小。

在对比实验中，算法 2 中 $\lambda = -1$，算法 3 中 $P = \{0.6, 0.7, 0.8, 0.9\}$，$\alpha = 0.7$。算法 2、算法 3 和算法 4 在 8 个数据集上的 FOF，MEF 和 AEF 的结果如图 2 所示。

从图 2 可以看出，本文算法 2 中 FOF 的结果最大，其次是算法 3，最小的是算法 4。对于 MEF 和 AEF 的结果来说，一般情况下，算法 2 的 MEF 和 AEF 最小，其次是算法 3，最大的为算法 4。而算法 4 中有时会出现约简为空集的情况，故而此时算法 4 中的 AEF 结果为负数。

图 3 绘制了 4 个算法的运行时间的比较结果，给出了 4 个算法分别在 8 个数据集上运行 100 次的平均时间。

图 4 绘制了 4 个算法的属性占比值，给出了 4 个算法分别在 8 个数据集上运行 100 次后得到的平均属性占比值。

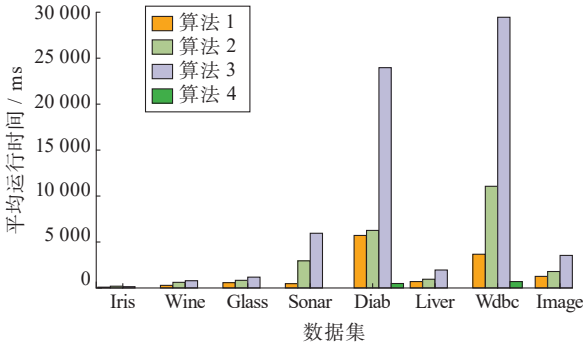

图 3 平均运行时间
Fig. 3 Average run-time

从图 3 可以看出，算法 4 的运行时间最小，其次是算法 1 和算法 2，最大的是算法 3。而在约简长度上(见图 4)，算法 1 和算法 2 以及算法 3 的约简长度差不多，都比较小，而算法 4 的约简长度则较大。综上，相较于对比算法，本文算法可以很大程度地减小总测试代价，且在运行效率上也具有一定的优势。

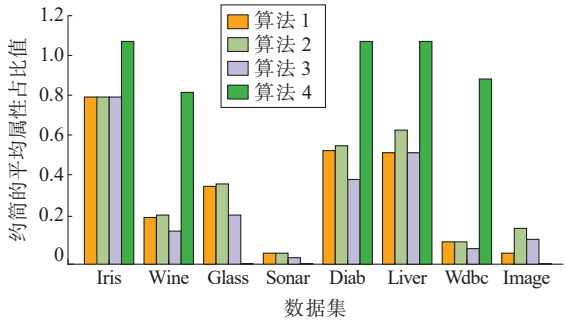

图 4 约简的平均属性占比值
Fig. 4 Average attribute proportion of reductions

4 结论

本文研究了区间值决策表中测试代价敏感的属性约简方法，介绍了相关的粗糙集理论知识，并设计了回溯算法和启发式算法。通过在多个 UCI 数据集上进行实验，验证了两种算法的有效性。未来将基于更多种类型的代价继续研究区间值决策表中代价敏感的属性约简方法。

参考文献：

[1] 谢小军. 测试代价敏感粗糙集中属性约简算法的研究[D]. 桂林: 广西师范大学, 2016.

[2] 陈华峰, 龙建武, 瞿先平. 区间值决策信息系统中基于正域的属性约简[J]. 重庆理工大学学报(自然科学版), 2019, 33(11): 130-136.

[3] 郭庆, 刘文军, 焦贤发, 等. 一种基于模糊聚类的区间值属性约简算法[J]. 模糊系统与数学, 2013, 27(1): 149-153.

[4] DU Wensheng, HU Baoqing. Approximate distribution reducts in inconsistent interval valued ordered decision tables[J]. Information Sciences, 2014, 271(11): 93-114.

[5] 尹继亮, 张楠, 赵立威, 等. 区间值决策系统的局部属性约简[J]. 计算机科学, 2018, 45(7): 178-185.

[6] 焦玉清, 张勇. 基于区间值信息系统的信息熵增量式属性约简算法[J]. 绥化学院学报, 2021, 41(9): 141-147.

[7] 胡明礼, 李林丽. 基于 α-β 优势关系的区间值信息系统属性约简方法[J]. 河南师范大学学报(自然科学版), 2014, 42(1): 151-156.

[8] DAI Jianhua, ZHENG Guojie, HAN Huifeng, et al. Probability approach for interval-valued ordered decision systems in dominance-based fuzzy rough set theory[J]. Journal of Intelligent and Fuzzy Systems, 2017, 32(1): 703-710.

[9] SHU Wenhao, QIAN Wenbin, XIE Yonghong, et al. An efficient uncertainty measure-based attribute reduction approach for interval-valued data with missing values[J]. International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems, 2019, 27(6): 931-947.

[10] 黄丽萍. 区间序信息系统在向量相似度下的优势关系及属性约简[J]. 齐齐哈尔大学学报(自然科学版), 2015, 31(6): 1-4.

[11] 徐伟华, 李思琪. 可变精度邻域区间值决策表的属性约简[J]. 西北大学学报(自然科学版), 2022, 52(5): 737-744.

[12] 唐鹏飞, 莫智文, 谢鑫. 区间值决策表中基于相对知识粒度的属性约简[J]. 重庆理工大学学报(自然科学版), 2021, 35(11): 286-292.

[13] 鲍迪, 张楠, 童向荣, 等. 区间值决策表的正域增量式属性约简算法[J]. 计算机应用, 2019, 39(8): 2288-2296.

[14] CHEN Yiying, LI Zhaowen, ZHANG Gangqiang. Attribute reduction in an incomplete interval-valued decision information system[J]. IEEE Access, 2021, 35(99): 1-13.

[15] LIU Xiaofeng, DAI Jianhua, CHEN Jiaolong, et al. A fuzzy α-similarity relation-based attribute reduction approach in incomplete interval-valued information systems[J]. Applied Soft Computing, 2021, 109(5): 107-113.

[16] 肖满红, 陶志, 胡柳 . 区间值优势关系系统中属性约简及规则提取方法[J]. 模糊系统与数学, 2022, 57(3): 36-41.

[17] 张晓雨, 李同军 . 不协调区间值决策系统中的 α-上、下近似约简[J]. 山东大学学报(理学版), 2022, 57(5): 8-12.

[18] YANG Qiang, WU Xindong. 10 challenging problems in data mining research[J]. International Journal of Information Technology & Decision Making, 2006, 5(4): 597-604.

[19] 张欣蕊, 万仁霞, 岳晓冬, 等 . 基于测试代价的三支邻域属性约简算法[J]. 计算机应用研究, 2024, 41(3): 1-8.

[20] MIN Fan, HE Huaping, QIAN Yuhua, et al. Test-cost-sensitive attribute reduction[J]. Information Sciences, 2011, 181(22): 4928-4942.

[21] MIN Fan, ZHU William. Attribute reduction of data with error ranges and test costs[J]. Information Sciences, 2012, 211(36): 48-67.

[22] XU Zilong, ZHAO Hong, MIN Fan, et al. Ant colony optimization with three stages for independent test cost attribute reduction[J]. Mathematical Problems in Engineering, 2013, 2013(1): 389-405.

[23] 鞠恒荣, 马兴斌, 杨习贝, 等 . 不完备信息系统中测试代价敏感的可变精度分类粗糙集[J]. 智能系统学报, 2014, 9(2): 5-11.

[24] 徐苏平, 杨习贝, 范霁月, 等 . 基于测试代价敏感的多粒度模糊粗糙集模型[J]. 电子设计工程, 2014, 22(7): 4-10.

[25] 谢小军, 徐章艳, 乔丽娟, 等 . 基于测试代价敏感的不完备决策系统属性约简算法[J]. 计算机应用与软件, 2016, 33(9): 6-13.

[26] 刘偲, 秦亮曦 . 测试代价敏感的决策粗糙集正域约简[J]. 计算机科学与探索, 2017, 11(6): 7-13.

[27] TAN Anhui, WU Weizhi, TAO Yuzhi. A set-cover-based approach for the test-cost-sensitive attribute reduction problem[J]. Soft Computing, 2017, 31(2): 11-20.

[28] 谢小军, 俞春强, 王博, 等 . 基于免疫量子粒子群优化的测试代价敏感属性约简算法[J]. 计算机工程与科学, 2017, 39(7): 8-15.

[29] 吴迪, 廖淑娇, 范译文 . 协调多尺度决策系统中基于测试代价的属性与尺度选择[J]. 模式识别与人工智能, 2023, 36(5): 433-447.

[30] LU Yaqian, LIAO Shujiao, YANG Wenyuan, et al. Interval-valued test cost sensitive attribute reduction related to risk attitude[J]. International Journal of Machine Learning and Cybernetics, 2024, 40(8): 1-20.

[31] 刘琼, 代建华, 陈姣龙 . 区间值数据的代价敏感特征选择[J]. 南京大学学报(自然科学版), 2021, 57(1): 121-129.

[32] 唐鹏飞, 莫智文, 谢鑫 . 区间值决策表中基于相对知识粒度的属性约简[J]. 重庆理工大学学报(自然科学版), 2021, 4(3): 30-37.

[33] 唐鹏飞 . 区间集决策信息系统的不确定性度量与属性约简[D]. 成都: 四川师范大学, 2022.

[34] BLAKE C L, KEOGH E, MERZ C J. UCI repository of machine learning databases[J/OL]. [2024-06-01]. http: //www.ics.uci.edu/~mlearn/MLRepository.html.

[35] 张晓燕, 匡洪毅 . 区间值序决策表的条件熵属性约简[J]. 山西大学学报(自然科学版), 2023, 46(1): 101-107.

(责任编辑：褚金红)