面向多尺度的属性约简加速器

姜泽华[1] 王怡博[2] 徐 刚[3] 杨习贝[1] 王平心[4]

(江苏科技大学计算机学院 江苏 镇江 212003)[1] (东南大学计算机科学与工程学院 南京 211189)[2]

(江苏科技大学船舶与海洋工程学院 江苏 镇江 212003)[3] (江苏科技大学理学院 江苏 镇江 212003)[4]

摘 要 邻域粗糙集,采用半径的方式度量样本之间是否相似,因而不同大小的半径自然地构成了不同尺度意义下的粗糙近似。基于邻域粗糙集的属性约简问题往往需要在多个不同半径上求解约简,其目的是找到具有较好泛化性能的属性子集,或探讨不同尺度意义下约简性能的变化趋势。但值得注意的是,利用传统的启发式算法在多个半径所对应的多尺度意义下进行约简求解时,往往需要在所有尺度上逐一重复执行这一算法,时间消耗较大,特别是尺度个数较多的情况下,时间消耗会变得更高。为解决这一问题,借助半径的变化,文中提出了面向多尺度的约简求解加速策略。这一策略在分别考虑半径从小到大和从大到小的变化趋势的情况下,同时缩小了样本和属性的遍历规模,将当前半径下约简的求解过程建立在上一个半径所求得约简的基础上,利用启发式搜索进行正向或逆向的属性增加及删除操作。为验证所提加速策略的有效性,实验选取 8 个 UCI 数据集,采用十折交叉验证的方法求取 20 个半径下的约简,对比不同方法求解约简的时间消耗和分类性能。实验结果表明,与利用传统的启发式算法在每一个尺度意义下单独求解约简的方法相比较,文中所提出的正向或逆向加速搜索方法可以在保持分类性能不发生显著变化的情况下,极大地降低多尺度意义下求解约简的时间消耗,并且有效地降低过拟合的程度。

关键词 属性约简,启发式搜索,多尺度,邻域粗糙集

中图法分类号 TP181 **文献标识码** A **DOI** 10.11896/jsjkx.181102031

Multi-scale Based Accelerator for Attribute Reduction

JIANG Ze-hua[1] WANG Yi-bo[2] XU Gang[3] YANG Xi-bei[1] WANG Ping-xin[4]

(School of Computer Science, Jiangsu University of Science and Technology, Zhenjiang, Jiangsu 212003, China)[1]

(School of Computer Science and Engineering, Southeast University, Nanjing 211189, China)[2]

(School of Naval Architecture and Ocean Engineering, Jiangsu University of Science and Technology, Zhenjiang, Jiangsu 212003, China)[3]

(School of Science, Jiangsu University of Science and Technology, Zhenjiang, Jiangsu 212003, China)[4]

Abstract The neighborhood rough set measures the similarity between samples by radius, consequently, different radii naturally construct the rough approximations with different scales. Traditional attribute reduction based on neighborhood rough set is frequently executed over multi-radius. The aims are to select an attribute subset with better generalization performance or to explore the trend line of the performances of the reducts in terms of different scales. However, it should be emphasized that the process should be repeatedly executed for each scale, if the traditional algorithm based on heuristic searching is used to compute the multiple scale reducts. The time consumption of computing reducts is too high to be accepted, especially the number of the scale is more, the time consumption will be more. To solve such a problem, the multi-scale based accelerated strategy for attribute reduction was proposed by means of the changing of radius. This strategy considers two trends of changing radius, from smaller radius to greater radius and from greater radius to smaller radius. Moreover, the traversal size of samples and attributes is reduced. Therefore, the current process to find reduct is executed based on the reduct related to the previous radius, which follows that the forward or backward heuristic searching for adding and deleting attributes can be realized. To validate the effectiveness of the accelerated strategy, 8 UCI data sets, 10-fold cross-validation and 20 radii were employed to conduct the experiment, and the time consumption of computing different reducts and the classification of different reducts were compared. The experimental results over 8 UCI data sets show that the proposed forward or backward accelerated searching can significantly reduce the time consumptions of finding reducts if the case of multi-scale is considered. Moreover, the proposed approach will

到稿日期:2018-11-04 返修日期:2019-03-11 本文受国家自然科学基金项目(61572242,61502211,61503160)资助。

姜泽华(1997—),女,硕士生,主要研究方向为粗糙集、粒计算;**王怡博**(1999—),男,主要研究方向为粗糙集、粒计算;**徐 刚**(1981—),男,博士,副教授,主要研究方向为智能信息处理;**杨习贝**(1980—),男,博士后,副教授,主要研究方向为粒计算、粗糙集与机器学习,E-mail:jsjxy_yxb@just.edu.cn(通信作者);**王平心**(1980—),男,博士后,副教授,主要研究方向为三支决策、粒计算。

not significantly decrease the classification performance, and can reduce the degree of over-fitting.

Keywords　Attribute reduction, Heuristic searching, Multi-scale, Neighborhood rough set

1　引言

作为一种刻画不确定与不精确问题的数学工具，粗糙集理论与方法[1]近年来在模式识别[2-3]、数据挖掘[4]、粒计算[5-6]、知识发现[7-8]及机器学习[9-11]等领域得到了广泛应用。在粗糙集领域中，属性约简[12-16]一直是众多学者关注的焦点。这主要是因为在实际工程应用所收集到的数据中往往存在大量冗余属性，这为诸如分类、回归以及推理等学习问题带来了严峻的挑战。作为一种数据预处理手段，属性约简不仅能够从数据中删除冗余属性，提高学习性能，而且具有简洁明了的度量意义，因而相较于其他的特征选择方法来说，所选中的属性往往具备一定的可解释性。

近年来，在粗糙集理论中，穷举法[17-18]和启发式搜索方法已经被公认为是求解约简的两大基本策略。但是，以分辨矩阵[19-21]为代表的穷举法时间复杂度较高，并不适用于较大规模的数据集；而建立在适应度函数基础上的启发式搜索则具备快速、高效的特性，所以受到了粗糙集学者的广泛关注。值得注意的是，无论是穷举法还是启发式搜索方法，在求解约简时，都仅仅是在一个尺度意义下进行的。然而，以粒计算的观点来看，从多层次、多视角以及多尺度[22]来分析与求解问题，才能达到人类智能在不同粒度意义下进行变换的目的[23]。鉴于此，笔者将主要讨论多尺度属性约简问题。

实际上，从粗糙集模型的构造方法来看，多尺度这一概念是广泛存在的。例如，以半径为基础的邻域粗糙集模型[24-25]利用半径来判定两个样本之间是否相似，所以不同大小的半径就恰好对应了不同尺度意义下的信息粒化结果；相应地，不同大小的半径也可以构造出不同尺度意义下的粗糙近似。以邻域粗糙集为基础，利用启发式算法进行约简求解，往往需要指定一个半径，也就是说这种约简求解方式是建立在固定尺度的基础上的。但是，若仅以一个半径求得约简，所得结果并不能适用于参数(半径)选择问题，并且也不能反映出约简性能在不同半径下的变化趋势。因此，已有关于邻域粗糙集约简问题的研究成果，往往是给定一组半径，并在这一组半径上分别求得约简。但是，随着半径个数的增加，这一过程往往非常耗时。为了解决这一问题，可以考虑半径由小到大和由大到小的变化趋势，设计基于正向和逆向的属性约简加速策略，以期降低建立在半径意义下的多尺度约简求解的时间消耗。

本文从多个半径上求解约简的角度出发，以半径为度量构造多尺度，并对面向多尺度的属性约简的概念进行定义；进一步地，针对传统的启发式算法计算多尺度约简的时间消耗较大的问题，设计加速策略。

本文第 2 节简要介绍邻域粗糙集的基本概念和基于集成选择的属性约简；第 3 节提出基于正向和逆向的约简加速算法；第 4 节进行实验分析；最后总结全文。

2　基础知识

2.1　邻域粗糙集

在粗糙集理论中，一个决策系统可以表示为二元组：$DS=\langle U, AT\cup\{d\}\rangle$。其中，$U=\{x_1, x_2, \cdots, x_n\}$是由 n 个样本组成的非空有限集合，称为论域；AT 是所有条件属性的集合；d 是决策属性。$U/\mathrm{IND}(\{d\})=\{X_1, X_2, \cdots, X_k\}$表示由决策属性 d 所诱导出论域上的划分，$\forall X_p\in U/\mathrm{IND}(\{d\})$，$X_p$ 表示具有相同标记的样本所构成的第 p 个决策类。

给定一个决策系统 DS，$\forall x_i, x_j\in U$，令 r_{ij}^B 表示利用条件属性集合 B 所得到样本 x_i 与 x_j 之间的距离。$\forall x_i\in U$，集合 $\delta_B(x_i)=\{x_j\in U\mid r_{ij}^B\leqslant\delta\}$表示 x_i 的 δ 邻域，其中 $\delta\in[0,1]$称为半径。为解决因半径过小而导致邻域所描述的信息不够充分这一问题，文献[26]给出了一种邻域区间的表示方法，即 $\forall x_i\in U$，x_i 的邻域区间为：

$$\mathrm{Int}(x_i)=\min_{1\leqslant j\leqslant n, j\neq i} r_{ij}^B+\delta\times\left(\max_{1\leqslant j\leqslant n, j\neq i} r_{ij}^B-\min_{1\leqslant j\leqslant n, j\neq i} r_{ij}^B\right) \tag{1}$$

其中，$\min\limits_{1\leqslant j\leqslant n, j\neq i} r_{ij}^B$表示 $U-\{x_i\}$中所有样本与 x_i 之间的距离的最小值，$\max\limits_{1\leqslant j\leqslant n, j\neq i} r_{ij}^B$表示 $U-\{x_i\}$中所有样本与 x_i 之间的距离的最大值。采用这种邻域区间，$\forall x_i\in U$，其邻域为：

$$\delta_B'(x_i)=\{x_j\in U\mid r_{ij}^B\leqslant\mathrm{Int}(x_i)\} \tag{2}$$

定义 1　给定一个决策系统 DS，$\delta\in[0,1]$，$\forall B\subseteq AT$，决策属性 d 关于 B 的下、上近似集定义为：

$$\underline{N_B^\delta}(d)=\bigcup_{p=1}^{k}\underline{N_B^\delta}(X_p) \tag{3}$$

$$\overline{N_B^\delta}(d)=\bigcup_{p=1}^{k}\overline{N_B^\delta}(X_p) \tag{4}$$

其中，$\forall X_p(U/\mathrm{IND}(\{d\})$，决策类 X_p 的下、上近似集分别为：

$$\underline{N_B^\delta}(X_p)=\{x_i\in U\mid\delta_B'(x_i)\subseteq X_p\} \tag{5}$$

$$\overline{N_B^\delta}(X_p)=\{x_i\in U\mid\delta_B'(x_i)\cap X_p\neq\varnothing\} \tag{6}$$

定义 2　给定一个决策系统 DS，$\delta\in[0,1]$，$\forall B\subseteq AT$，决策属性 d 相对于 B 的近似质量表示为 $\gamma^\delta(B,d)$：

$$\gamma^\delta(B,d)=\frac{|\underline{N_B^\delta}(d)|}{|U|} \tag{7}$$

其中，$|X|$表示集合 X 的基数。

2.2　基于集成选择的属性约简

属性约简的本质是在满足一定约束条件的前提下，删除条件属性集合中冗余的属性。在粗糙集理论中，不同的约束条件往往是依据不同的度量准则而构造的。近年来，根据不同的应用需求，已有大量的学者提出了多种不同类型的度量准则，如近似质量/依赖度[27]、条件熵[28-30]、决策错误率[31]、鉴别指数[32]等。笔者选取了由 Pawlak 提出的近似质量这种应用最为广泛的度量，作为构造约简的约束条件。

定义 3　给定一个决策系统 DS，$\delta\in[0,1]$，$\forall B\subseteq AT$，B 被称为 AT 的一个近似质量约简，当且仅当：

(1) $\gamma^\delta(B,d)\geqslant\gamma^\delta(AT,d)$；

(2) $\forall B'\subset B$，$\gamma^\delta(B',d)<\gamma^\delta(B,d)$。

定义 3 给出的约简是一个能够保持近似质量至少不会降低的最小属性子集。为了求得这样一个属性子集，已有很多学者给出了各种形式的搜索策略，其中启发式搜索因具有较快的求解速度而受到了广大粗糙集学者的青睐。值得注意的是，在启发式算法框架下，适应度函数是一个重要的概念，它的主要作用是描述属性的重要度。

定义 4　给定一个决策系统 DS，$\delta\in[0,1]$，$\forall B\subseteq AT$，$\forall a\in AT-B$，属性 a 相对于 B 的重要度定义为：

$$Sig^{\delta}(a,B,d)=\gamma^{\delta}(B\cup\{a\},d)-\gamma^{\delta}(B,d) \quad (8)$$

在定义 4 中，Sig 这一适应度函数是用来描述将属性 a 加入到属性集合 B 后，近似质量的变化情况。显然，如果 $\gamma^{\delta}(B\cup\{a\},d)<\gamma^{\delta}(B,d)$，那么就说明：属性 a 的加入没有带来近似质量的增长，此时 a 可以被视为冗余属性；如果 $\gamma^{\delta}(B\cup\{a\},d)\geqslant\gamma^{\delta}(B,d)$，那么就说明：属性 a 的加入可以为近似质量的提升带来帮助。

显然，在启发式算法框架下，利用 Sig 函数，可以在每次迭代过程中找到一个为近似质量的提升带来最大帮助的属性，直到找出的属性子集满足约束条件。然而，Yang 与 Yao[33] 指出，这一搜索策略仅仅采用了一个适应度函数，所求得的约简并不能满足多视角需求，同时也不具有普适性。因此，他们在启发式算法流程中，利用多个适应度函数提出了一类集成选择器。利用这一集成选择器求解定义 3 所示的约简，具体流程如算法 1 所示。

算法 1 基于集成选择的约简求解

输入：决策系统 DS，半径 δ

输出：约简 B

步骤 1 B=∅。

步骤 2 计算 U/IND({d})={X_1,X_2,…,X_k}，计算 γ(AT,d)。

步骤 3 T=∅（T 为暂时存放所选出属性的集合）。

For p=1 to k

(1) ∀a∈AT−B，计算 $Sig^{\delta}(a,B,X_p)=|\underline{N^{\delta}_{B\cup\{a\}}}(X_p)-\underline{N^{\delta}_{B}}(X_P)|$；

(2) 选择 b，满足 $Sig^{\delta}(b,B,X_p)=\max\{Sig^{\delta}(a,B,X_p):\forall a\in AT-B\}$；

(3) T=T∪{b}；

End For

步骤 4 计算 T 中每个不同属性出现的频次。

步骤 5 采用投票策略，在 T 中选择一个出现频次最高的属性，并将其加入 B。

步骤 6 计算 $\gamma^{\delta}(B,d)$，若 $\gamma^{\delta}(B,d)\geqslant\gamma^{\delta}(AT,d)$，则转步骤 7；否则转步骤 3。

步骤 7 While |B|≥2 且 $\max\{\gamma^{\delta}(B-\{c\},d):\forall c\in B\}\geqslant\gamma^{\delta}(B,d)$

∀c∈B，计算 $\gamma^{\delta}(B-\{c\},d)$，若 $\gamma^{\delta}(B-\{c\},d)\geqslant\gamma^{\delta}(B,d)$，则 B=B−{c}。

End While

步骤 8 输出 B。

一般来说，分类学习系统中往往存在多个不同的决策类，依据不同的决策类，可以为每个条件属性计算出多个适应度函数的值。因此，算法 1 考虑不同的决策类，利用 k 个与决策类相关的适应度函数 $Sig^{\delta}(a,B,X_p)(1\leqslant p\leqslant k)$ 进行属性的集成选择。

但值得注意的是，上述关于邻域粗糙集的属性约简[34-35]往往需要在一组而非单个半径上进行问题的求解，以期能够选择出具有较好泛化性能的约简。在多个半径下进行约简的求解势必增加时间消耗，为了解决这一问题，下一节将借助半径由小到大和由大到小的变化趋势，分别设计基于正向和逆向的属性约简加速策略。

3 多尺度属性约简

定义 5 给定一个决策系统 DS，若 $\delta_1,\cdots,\delta_s\in[0,1]$ 且 $\delta_1\leqslant\cdots\leqslant\delta_s$，则 $\{B_1,\cdots,B_s\}$ 被称为 DS 的一个多尺度约简，当且仅当对于任意的半径 $\delta_t(1\leqslant t\leqslant s)$，$B_t$ 是半径 δ_t 意义下的一个近似质量约简。

不同于定义 3 给出的近似质量约简，定义 5 给出的约简 $\{B_1,\cdots,B_s\}$ 不再是一个属性集合，而是一组约简所构成的集合。若利用算法 1 进行定义 5 给出的约简求解，则需要针对所给定的所有半径，逐个地利用算法 1 进行约简求解。为了降低求解定义 5 给出的多尺度约简的时间消耗，下文将从正向与逆向两个角度分别构建加速算法。

3.1 基于正向的属性约简

给定一个决策系统 DS，对于任意的半径 δ_1 和 δ_2，如果 $\delta_1\leqslant\delta_2$，那么就有 $\underline{N^{\delta_2}_{B}}(d)\subseteq\underline{N^{\delta_1}_{B}}(d)$ 成立。据此，当利用半径 δ_2 进行约简求解时，可以将 $\underline{N^{\delta_1}_{B}}(d)$ 视为一个新的论域，然后在此基础上进行计算。

此外，当半径从 δ_1 变化为 δ_2 时，约简 B_2 中属性的搜索不是从 AT 开始进行，而是在 B_1 的基础上，考查 $AT-B_1$ 中的每一个属性，并利用集成选择的思想选取重要度最大的属性加入 B_2，直至满足约束条件。

显然，上述策略可以在考虑半径变化的情况下，同时缩小样本和属性的遍历规模，以达到减少时间消耗的目的。根据以上描述，借助半径由小到大的变化趋势，算法 2 构建了一个基于正向的约简求解的流程。

算法 2 基于正向的约简求解

输入：决策系统 DS，$\delta_1,\cdots,\delta_s\in[0,1]$ 且 $\delta_1\leqslant\cdots\leqslant\delta_s$

输出：多尺度约简 $\{B_1,\cdots,B_s\}$

步骤 1 利用算法 1 计算 B_1。

步骤 2 对于半径 $\delta_t(t=2,\cdots,s)$，执行步骤 3。

步骤 3 (1) 令 $B_t=B_{t-1}$；若 $\underline{N^{\delta_{t-1}}_{B_{t-1}}}(d)\neq\emptyset$，则 $U'=\underline{N^{\delta_{t-1}}_{B_{t-1}}}(d)$。

(2) 计算 U′/IND({d})={X_1,X_2,…,X_m}。

(3) T=∅。

For p=1 to m

1) ∀a∈AT−B_t，计算 $Sig^{\delta_t}(a,B_t,X_p)=|\underline{N^{\delta_t}_{B_t\cup\{a\}}}(X_p)-\underline{N^{\delta_t}_{B_t}}(X_p)|$；

2) 选择 b，满足 $Sig^{\delta_t}(b,B_t,X_p)=\max\{Sig^{\delta_t}(a,B_t,X_p):\forall a\in AT-B_t\}$；

3) T=T∪{b}。

End For

(4) 计算 T 中不同属性出现的频次。

(5) 采用投票策略，在 T 中选择一个出现频次最高的属性，并将其加入 B_t。

(6) 计算 $\gamma^{\delta_t}(B_t,d)$，若 $\gamma^{\delta_t}(B_t,d)\geqslant\gamma^{\delta_t}(AT,d)$，则转步骤(7)；否则转步骤(3)。

(7) While $|B_t|\geqslant2$ 且 $\max\{\gamma^{\delta_t}(B_t-\{c\},d):\forall c\in B_t\}\geqslant\gamma^{\delta_t}(B_t,d)$

∀c∈B_t，计算 $\gamma^{\delta_t}(B_t-\{c\},d)$，若 $\gamma^{\delta_t}(B_t-\{c\},d)\geqslant\gamma^{\delta_t}(B_t,d)$，则 $B_t=B_t-\{c\}$；

End While

步骤 4 输出 $\{B_1,\cdots,B_s\}$。

3.2 基于逆向的属性约简

给定一个决策系统 DS，对于任意的半径 δ_1 和 δ_2，如果

$\delta_1 \leqslant \delta_2$，那么就有 $\underline{N_B^{\delta_2}}(d) \subseteq \underline{N_B^{\delta_1}}(d)$ 成立。可以看出，半径为 δ_1 时的下近似集包含半径为 δ_2 时的下近似集，因此在利用半径 δ_1 进行下近似计算时，无须重复计算 $\underline{N_B^{\delta_2}}(d)$ 中的样本，可以以 $\underline{N_B^{\delta_2}}(d)$ 为基础，仅搜索出 $U-\underline{N_B^{\delta_2}}(d)$ 中的样本。

此外，与算法 2 中遍历属性的方法类似，当半径从 δ_2 变化为 δ_1 时，约简 B_1 中属性的搜索不是从 AT 开始进行，而是在 B_2 的基础上，考查 $AT-B_2$ 里的每一个属性，并利用集成选择的思想选取重要度最大的属性加入 B_1，直至满足约束条件。

显然，上述策略也可以在考虑半径变化的情况下，同时缩小样本和属性的遍历规模，以达到减少时间消耗的目的。根据以上描述，借助半径由大到小的变化趋势，算法 3 构建了一个基于逆向的约简求解的流程。

算法 3　基于逆向的约简求解

输入：决策系统 DS，$\delta_1,\cdots,\delta_s \in [0,1]$ 且 $\delta_1 \leqslant \cdots \leqslant \delta_s$

输出：多尺度约简 $\{B_1,\cdots,B_s\}$

步骤 1　利用算法 1 计算 B_s。

步骤 2　对于半径 δ_t（t=s−1,…,1），执行步骤 3。

步骤 3　(1)令 $B_t=B_{t+1}$；$U'=U-\overline{N_{B_{t+1}}^{\delta_{t+1}}}(d)$。

(2)$U'/IND(\{d\})=\{X_1,X_2,\cdots,X_m\}$。

(3)$T=\varnothing$。

For p=1 to m

1) $\forall a \in AT-B_t$，计算 $Sig^{\delta_t}(a,B_t,X_p)=|\underline{N_{B_t\cup\{a\}}^{\delta_t}}(X_p)-\underline{N_{B_t}^{\delta_t}}(X_p)|$；

2)选择 b，满足 $Sig^{\delta_t}(b,B_t,X_p)=\max\{Sig^{\delta_t}(a,B_t,X_p): \forall a \in AT-B_t\}$；

3)$T=T\cup\{b\}$。

End For

(4)计算 T 中不同属性出现的频次。

(5)采用投票策略，在 T 中选择一个出现频次最高的属性，并将其加入 B_t。

(6)计算 $\gamma^{\delta_t}(B_t,d)$，若 $\gamma^{\delta_t}(B_t,d) \geqslant \gamma^{\delta_t}(AT,d)$，则转步骤(7)；否则转步骤(3)。

(7)While $|B_t| \geqslant 2$ 且 $\max\{\gamma^{\delta_t}(B_t-\{c\},d): \forall c \in B_t\} \geqslant \gamma^{\delta_t}(B_t,d)$

$\forall c \in B_t$，计算 $\gamma^{\delta_t}(B_t-\{c\},d)$，若 $\gamma^{\delta_t}(B_t-\{c\},d) \geqslant \gamma^{\delta_t}(B_t,d)$，则 $B_t=B_t-\{c\}$。

End While

步骤 4　输出 $\{B_1,\cdots,B_s\}$。

4　实验分析

为了验证上述所提正向和逆向算法的有效性，笔者从 UCI 数据集中选取了 8 组数据，数据的基本描述如表 1 所列。其中，属性个数是指所有条件属性的个数。

表 1　数据集描述

Table 1　Description of data sets

ID	数据集	样本个数	属性个数	决策类个数
1	Congressional Voting	435	16	2
2	Dermatology	366	34	6
3	German	1000	24	2
4	Ionosphere	351	34	2
5	Lymphography	98	18	3
6	Page-blocks	5473	10	5
7	Statlog (Heart)	270	13	2
8	Wine	178	13	3

实验选择了 20 个不同的半径：0.025，0.05，0.075，0.10，…，0.50。为了对比新提出算法的有效性，实验采用了十折交叉验证的方法。其具体过程是：将数据中的样本平均分为 10 份，即 $U_1,U_2,\cdots,U_{10}$。第 1 次将 $U_2\cup U_3\cup\cdots\cup U_{10}$ 作为训练集，分别利用 3 种算法求得 20 个不同半径下的约简；第 2 次将 $U_1\cup U_3\cup\cdots\cup U_{10}$ 作为训练集，分别利用 3 种算法求得 20 个不同半径下的约简；以此类推，第 10 次将 $U_1\cup U_2\cup\cdots\cup U_9$ 作为训练集，分别利用 3 种算法求得 20 个不同半径下的约简。此外，实验中选取近似质量作为度量准则。

4.1　时间消耗的对比

在此实验中，3 种算法均按照以下步骤进行：利用十折交叉验证在对应的训练集上分别求取 10 个半径下的约简结果，并计算求取约简的时间，然后计算每一半径下时间消耗的平均值。此外，对比 3 种算法的时间消耗（单位为 s），如图 1 所示。

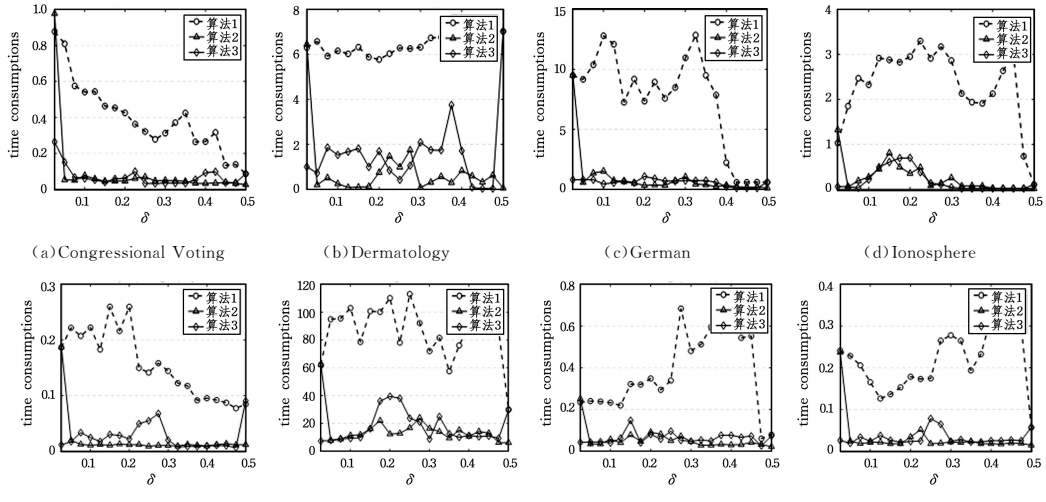


图 1　时间消耗在不同约简下的对比

Fig. 1　Comparison of time consumption among different reducts

观察图 1 可以看出，在 20 个不同半径下，算法 2 在最小半径下计算约简的时间消耗与算法 1 的时间消耗接近，而在其他半径下计算约简的时间消耗远远小于算法 1 的时间消耗。这主要是因为算法 2 在求解最小半径下的约简时采用算法 1 的求解策略，而在求解其余半径下的约简时采用基于正向的属性约简加速策略。

算法 3 在最大半径下的时间消耗与算法 1 的时间消耗接近，但是在其他半径下计算约简的时间消耗远远小于算法 1 的时间消耗。这主要是因为算法 3 在求解最大半径下的约简时采用算法 1 的求解策略，而在求解其余半径下的约简时采用基于逆向的属性约简加速策略。

综上所述，在每一半径下，使用加速器求解约简的时间消耗均明显小于用原始方法求解约简的时间消耗。由此不难得到，在这一组半径下，使用加速器求解约简的时间消耗之和也必然小于使用原始方法求解约简的时间消耗之和，即使用加速器求解这一组约简的时间消耗明显小于用原始方法求解约简的时间消耗。因此，在求解多尺度约简时，相比于算法 1，算法 2 和算法 3 在时间消耗上具备明显的优势。

为了检验新提出的算法 2、算法 3 与原有算法 1 在求取约简过程中的耗时差异性，选取 K-S 检验进行统计比较，并将显著性水平设置为 0.05，所求得的 p 值如表 2 所列。

表 2 时间消耗在不同约简下 K-S 检验的 p 值

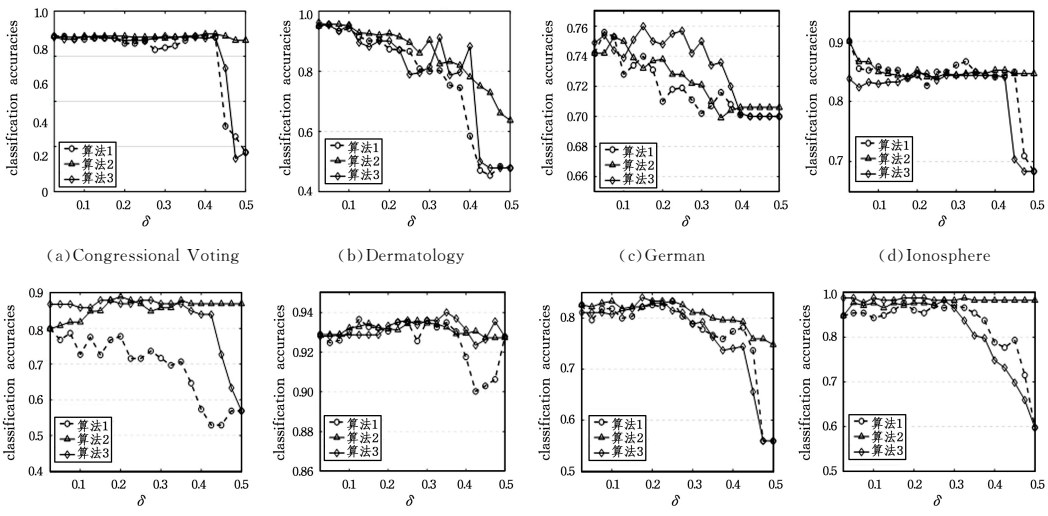
Table 2 p-value of K-S test for comparing time consumption of different reducts

ID	算法 1 & 算法 2	算法 1 & 算法 3
1	2.49×10^{-7}	5.55×10^{-10}
2	4.74×10^{-9}	5.55×10^{-10}
3	4.74×10^{-9}	5.55×10^{-10}
4	4.74×10^{-9}	5.55×10^{-10}
5	4.74×10^{-9}	5.55×10^{-10}
6	4.74×10^{-9}	5.55×10^{-10}
7	5.55×10^{-10}	5.55×10^{-10}
8	4.74×10^{-9}	4.74×10^{-9}

观察表 2 不难看出，在 8 组数据下，算法 2 和算法 1 比较的 p 值都明显小于 0.05，算法 3 和算法 1 比较的 p 值也明显小于 0.05，这说明算法 2、算法 3 与算法 1 相比，在时间消耗上具有显著的统计差异。

4.2 分类精度的对比

在此实验中，采取 SVM 分类器进行分类。3 种算法均按照以下步骤进行：利用十折交叉验证在对应的训练集上分别求取 10 个半径下的约简结果，利用求得的约简结果对测试集上的数据进行分类，并计算分类精度，最后计算每一半径下分类精度的平均值。对比 3 种算法所求得约简的分类精度，如图 2 所示。



(a)Congressional Voting (b)Dermatology (c)German (d)Ionosphere

(e)Lymphography (f)Page-blocks (g)Statlog(Heart) (h)Wine

图 2 分类精度在不同约简下的对比

Fig. 2 Comparison of classification accuraly among different reducts

观察图 2 可以发现，在大多数情况下，算法 2 所求得的约简能够提供略高于算法 1 的分类精度，而算法 3 与算法 1 所求得的约简具有相似的分类能力。

为了验证算法 2 和算法 3 能够避免过拟合现象，实验选取 KNN，SVM，CART 3 种分类器进行分类精度的对比，实验结果如表 3－表 5 所列。

表中，“精度”表示在所有半径上所得约简结果的分类精度的最大值，“半径”表示在取得最大精度时对应的半径，“属性个数”表示所有条件属性的个数，“约简长度”表示相应半径下计算所得约简结果的长度。

表 3 KNN 分类器下的精度最大值和属性长度

Table 3 Maximal value of accuracy based on KNN classifier and length of attributes

ID	原始数据		算法 2			算法 3		
	精度	属性长度	精度	半径	约简长度	精度	半径	约简长度
1	0.92	16.0	0.95	0.245	4.2	0.95	0.090	6.5
2	0.96	34.0	0.96	0.090	8.0	0.96	0.020	13.1
3	0.72	24.0	0.72	0.170	8.4	0.72	0.245	8.6
4	0.85	34.0	0.90	0.200	8.8	0.88	0.050	6.2
5	0.86	18.0	0.78	0.410	4.7	0.74	0.100	5.4
6	0.96	10.0	0.96	0.005	5.3	0.96	0.010	5.1
7	0.81	13.0	0.81	0.275	6.2	0.81	0.225	6.7
8	0.95	13.0	0.98	0.040	6.2	0.97	0.240	5.2

表 4　SVM 分类器下的精度最大值和属性长度

Table 4　Maximal value of accuracy based on KNN classifier and length of attributes

ID	原始数据		算法 2			算法 3		
	精度	属性长度	精度	半径	约简长度	精度	半径	约简长度
1	0.93	16.0	0.95	0.140	5.3	0.95	0.055	6.6
2	0.97	34.0	0.95	0.020	13.1	0.97	0.010	14.4
3	0.76	24.0	0.75	0.190	9.4	0.77	0.020	9.8
4	0.91	34.0	0.87	0.210	7.6	0.88	0.190	11.9
5	0.86	18.0	0.81	0.005	6.8	0.78	0.160	4.7
6	0.94	10.0	0.94	0.005	5.3	0.94	0.005	5.3
7	0.84	13.0	0.83	0.270	6.9	0.83	0.225	6.7
8	0.98	13.0	0.97	0.040	6.2	0.97	0.225	5.2

表 5　CART 分类器下的精度最大值和属性长度

Table 5　Maximal value of accuracy based on CART classifier and length of attributes

ID	原始数据		算法 2			算法 3		
	精度	属性长度	精度	半径	约简长度	精度	半径	约简长度
1	0.94	16.0	0.95	0.320	4.4	0.96	0.060	6.6
2	0.94	34.0	0.96	0.010	14.4	0.96	0.005	10.5
3	0.71	24.0	0.72	0.060	10.7	0.73	0.015	9.5
4	0.89	34.0	0.89	0.040	7.8	0.89	0.230	7.4
5	0.73	18.0	0.80	0.135	5.0	0.80	0.005	6.8
6	0.96	10.0	0.96	0.025	5.0	0.96	0.005	5.3
7	0.76	13.0	0.82	0.170	5.5	0.83	0.145	5.5
8	0.88	13.0	0.92	0.370	4.5	0.90	0.255	5.1

观察表 3－表 5 可以看出，在 3 种分类器上，对于大多数数据来说，相比于原始数据，利用算法 2 和算法 3 所选择出的约简能够提供略好或相同的分类性能，有效地避免了过拟合现象；对比约简结果的长度可以看出，利用算法 2 和算法 3 所得约简结果的长度明显小于原始数据中所有条件属性的个数，约简过程去除了冗余属性。

4.3　稳定性分析

下面考虑不同半径下所求得约简的相似性程度，并将其作为一种多尺度约简的稳定性度量指标。因此，给出如下计算方法。

定义 6　给定一个决策系统 DS，若 $\delta_1,\cdots,\delta_s\in[0,1]$ 且 $\delta_1\leqslant\cdots\leqslant\delta_s$，则稳定性定义为：

$$Sta=\sum_{t=1}^{s-1}\frac{|B_t\bigcap B_{t+1}|}{|B_t\bigcup B_{t+1}|} \tag{9}$$

其中，B_t 是半径为 δ_t 时获得的约简。

由式(9)可以看出，$Sta\in[0,1]$。若对于任意两个不同半径下的约简 B_t 和 B_{t+1}，都有 $B_t\bigcap B_{t+1}=\varnothing$，则 Sta 取得最小值，此时可知半径的变化对约简结果的影响是非常巨大的；若对于任意两个不同半径下的约简 B_t 和 B_{t+1}，都有 $B_t=B_{t+1}$，则 Sta 取得最大值，此时可知半径的变化对约简结果没有任何影响。表 6 对比了 3 种算法所求得约简的稳定性。

观察表 6 可以看出，相比于利用算法 1 所求得的约简，利用算法 2 所求得的约简的稳定性更高；利用算法 3 所求得的约简的稳定性也明显高于利用算法 1 所求得的约简的稳定性。

表 6　约简稳定性的对比

Table 6　Comparison of stabilities among different reducts

ID	算法 1	算法 2	算法 3
1	0.4067	**0.7224**	0.4829
2	0.3610	**0.4305**	0.4034
3	0.4127	**0.6603**	0.4836
4	0.7178	0.8639	**0.9433**
5	0.4314	**0.8618**	0.6138
6	0.3153	**0.6127**	0.4120
7	0.7034	0.7126	**0.7935**
8	0.5803	**0.8507**	0.7899

综合图 2 和表 6 可以得到以下结论：算法 2 和算法 3 在大多数半径下的约简都能提供较高的分类精度，虽然在某些半径下约简的分类精度不如算法 1，但其所得约简的稳定性都明显高于算法 1 所得约简的稳定性。

结束语　面向多尺度进行约简求解时，传统基于适应度函数的启发式搜索的时间消耗会随着半径数量的增加而不断增长。鉴于此，考虑半径由小到大和由大到小的变化趋势，面向多尺度，设计了正向和逆向约简求解的加速策略。通过实验对比分析可以得出，所提方法能够显著地降低求取多尺度约简的时间消耗，提高约简结果的稳定性，并且利用新方法，约简的分类性能并不逊色于利用传统单尺度算法所得到的约简的分类性能。因此，正向和逆向约简求解加速方法在获取以半径为基准的多尺度约简时是有效的。在这一工作的基础上，笔者后期将就以下问题进行进一步的探讨。

(1)文中仅仅利用了多个不同的半径，构建了多尺度。未来可以考虑其他一些多尺度的构造方式，如由层次属性所形成的多尺度结构。

(2)文中进行集成属性选择时并未考虑不同适应度函数的权重问题。未来可以从决策类样本规模的角度对不同的适应度函数进行权重赋值，并以此为基础，进一步探讨多尺度约简求解加速策略的有效性。

参 考 文 献

[1] PAWLAK Z. Rough Sets[J]. International Journal of Computer & Information Sciences, 1982, 11(5): 341-356.

[2] DAI J H, GAO S C, ZHENG G J. Generalized Rough Set Models Determined by Multiple Neighborhoods Generated from a Similarity Relation[J]. Soft Computing, 2018, 22(7): 2081-2094.

[3] QIAN Y H, LIANG X Y, WANG Q, et al. Local Rough Set: A Solution to Rough Data Analysis in Big Data[J]. International Journal of Approximate Reasonging, 2018, 97(1): 38-63.

[4] HU Q H, AN S, YU D R. Soft Fuzzy Rough Sets for Robust Feature Evaluation and Selection[J]. Information Sciences, 2010, 180(22): 4384-4400.

[5] MIAO D Q, GAO C, ZHANG N, et al. Diversereduct Subspaces Based Co-training for Partially Labeled Data[J]. International Journal of Approximate Reasoning, 2011, 52(8): 1103-1117.

[6] CHEN H M, LI T R, LUO C, et al. A Decision-theoretic Rough Set Approach for Dynamic Data Mining[J]. IEEE Transactions on Fuzzy Systems, 2015, 23(6): 1958-1970.

[7] TAO F, MI J S. Variable Precision Multigranulation Decision-theoretic Fuzzy Rough Sets[J]. Knowledge-Based Systems, 2016, 91(1): 93-101.

[8] QIAN Y H,CHENG H H,WANG J T,et al. Grouping Granular Structures in Human Granulation Intelligence[J]. Information Sciences,2017,382/383:150-169.

[9] CHEN Y F,YUE X D,FUJITA H. Three-way Decision Support for Diagnosis on Focal Liver Lesions[J]. Knowledge-Based Systems,2017,127(C):85-99.

[10] LIU D,LIANG D C,WANG C C. A Novel Three-way Decision Model Based on Incomplete Information System[J]. Knowledge-Based Systems,2016,91(C):32-45.

[11] HU Q H,PAN W W,ZHANG L,et al. Feature Selection for Monotonic Classification[J]. IEEE Transactions on Fuzzy Systems,2012,20(1):69-81.

[12] JIA X Y,LIAO W H,TANG Z M,et al. Minimum Cost Attribute Reduction in Decision-theoretic Rough Set Models[J]. Information Sciences,2013,219(1):151-167.

[13] MIN F,ZHU W. Attribute Reduction of Data with Error Ranges and Test Costs[J]. Information Sciences,2012,211:48-67.

[14] SONG J J,TSANG E C C,CHEN D G,et al. Minimal Decision Cost Reduct in Fuzzy Decision-theoretic Rough Set Model[J]. Knowledge-Based Systems,2017,126(C):104-112.

[15] JU H R,LI H X,YANG X B,et al. Cost-sensitive Rough Set:A Multi-granulation Approach[J]. Knowledge-Based Systems,2017,123(1):137-153.

[16] YAO Y Y,ZHANG X Y. Class-specific Attribute Reducts in Rough Set Theory[J]. Information Sciences,2017,418/419:601-618.

[17] MIN F,HE H P,QIAN Y H,et al. Test-cost-sensitive Attribute Reduction[J]. Information Sciences,2011,181(22):4928-4942.

[18] YANG X B,QI Y S,SONG X N,et al. Test Cost Sensitive Multigranulation Rough Set:Model and Minimal Cost Selection[J]. Information Sciences,2013,250(11):184-199.

[19] CHEN D G,YANG Y Y,DONG Z. An Incremental Algorithm for Attribute Reduction with Variable Precision Rough Sets[J]. Applied Soft Computing,2016,45(1):129-149.

[20] FAN J,JIANG Y L,LIU Y. Quick Attribute Reduction with Generalized Indiscernibility Models[J]. Information Sciences,2017,397/398:15-36.

[21] YAO Y Y,ZHAO Y. Discernibility Matrix Simplification for Constructing Attribute Reducts[J]. Information Sciences,2009,179(7):867-882.

[22] WU W Z,QIAN Y H,LI T J,et al. On Rule Acquisition in Incomplete Multi-scale Decision Tables[J]. Information Sciences,2017,378(C):282-302.

[23] ZHANG L,ZHANG B. Theory of Fuzzy Quotient Space(Methods of Fuzzy Granular Computing)[J]. Journal of Software,2003,14(4):770-776. (in Chinese)
张铃,张拔. 模糊熵空间理论(模糊粒度计算方法)[J]. 软件学报,2003,14(4):770-776.

[24] SUN L,PAN J F,ZHANG X Y,et al. Multi-label-specific Feature Selection Method Based on Neighborhood Rough Set[J]. Computer Science,2018,45(1):173-178. (in Chinese)
孙林,潘俊方,张霄雨,等. 一种基于邻域粗糙集的多标记专属特征选择方法[J]. 计算机科学,2018,45(1):173-178.

[25] XU S P,YANG X B,YU H L,et al. Neighborhood Collaborative Representation Based Classification Method[J]. Computer Science,2017,44(9):234-238. (in Chinese)
徐苏平,杨习贝,于化龙,等. 一种基于邻域协同表达的分类方法[J]. 计算机科学,2017,44(9):234-238.

[26] HU Q H,YU D R,XIE Z X. Neighborhood Classifiers[J]. Expert Systems with Applications,2008,34(2):866-876.

[27] YANG X B,LIANG S C,YU H L,et al. Pseudo-label Neighborhood Rough Set:Measures and Attribute Reductions[J]. International Journal of Approximate Reasoning,2019,105(1):115-129.

[28] WEI W,WEI Q,WANG F. Comparative Study of Uncertainty Measure in Rough Set[J]. Journal of Nanjing University(Natural Sciences),2015,51(4):714-722. (in Chinese)
魏巍,魏琪,王锋. 粗糙集的不确定性度量比较研究[J]. 南京大学学报(自然科学版),2015,51(4):714-722.

[29] ZHANG X,MEI C L,CHEN D G,et al. Feature Selection in Mixed Data:A Mehhod Using a Novel Fuzzy Rough Set-based Information Entropy[J]. Pattern Recognition,2016,56(1):1-15.

[30] GAO Y,CHEN X J,YANG X B,et al. Neighborhood Attribute Reduction:A Multicriterion Strategy Based on Sample Selection[J]. Information,2018,9(11):282-302.

[31] HU Q H,PEDRYCZ W,YU D R,et al. Selecting Discrete and Continuous Features Based on Neighborhood Decision Error Minimization[J]. IEEE Transactions on Systems Man and Cybernetics Part B,2009,40(1):137-150.

[32] WANG C Z,HU Q H,WANG X Z,et al. Feature Selection Based on Neighborhood Discrimination Index[J]. IEEE Transactions on Neural Networks and Learning Systems,2018,29(1):2986-2999.

[33] YANG X B,YAO Y Y. Ensemble Selector for Attribute Reduction[J]. Applied Soft Computing,2018,70(1):1-11.

[34] LI Z Y,YANG X B,XU S P,et al. Attribute Reduction Approach to Neighborhood Decision Agreement[J]. Journal of Henan Normal University(Natural Science Edition),2017,45(5):68-73. (in Chinese)
李智远,杨习贝,徐苏平,等. 邻域决策一致性的属性约简方法研究[J]. 河南师范大学学报(自然科学版),2017,45(5):68-73.

[35] LI Z Y,YANG X B,CHEN X J,et al. Attribute Reduction Constrained by Class-specific Approximate Quality[J]. Journal of Henan Normal University(Natural Science Edition),2018,46(3):112-118. (in Chinese)
李智远,杨习贝,陈向坚,等. 类别近似质量约束下的属性约简方法研究[J]. 河南师范大学学报(自然科学版),2018,46(3):112-118.