

基于邻域关系的一种高效属性约简算法

李长达

烟台大学计算机与控制工程学院, 山东 烟台

收稿日期: 2025年3月22日; 录用日期: 2025年4月22日; 发布日期: 2025年4月29日

摘要

属性约简是粗糙集理论中的重要研究内容之一。近年来得到了快速发展, 诸多学者已经做出了大量的优秀成果。基于粗糙集的属性约简通过约简目标的约束构建属性重要度函数, 迭代剔除冗余属性, 从而获得求解结果, 这种方法往往具有较高的时间开销。针对上述问题, 本文在邻域关系下提出一种求解约简的高效算法。通过设计逐层收缩的正域迭代机制, 去除已确定的正域对象, 从而减少每次迭代过程中论域的基数, 降低局部的时间复杂度。在UCI数据集下对算法的运行结果以及运行时间进行比较。实验结果表明, 该算法在保证约简结果一致性的前提下, 显著降低了求解属性约简的时间开销。

关键词

粗糙集, 邻域关系, 属性约简, 高效算法

An Efficient Attribute Reduction Algorithm under Neighborhood Relations

Changda Li

School of Computer and Control Engineering, Yantai University, Yantai Shandong

Received: Mar. 22nd, 2025; accepted: Apr. 22nd, 2025; published: Apr. 29th, 2025

Abstract

Attribute reduction, as a pivotal research focus in rough set theory, has undergone significant advancement in recent years with substantial contributions from scholars. The rough set-based attribute reduction methodology constructs attribute significance functions constrained by reduction criteria and iteratively removes redundant attributes to derive reducts, which typically entails considerable computational complexity. Addressing this challenge, this paper proposes an efficient reduct computation algorithm under neighborhood relations. The core innovation involves a progressively contracted positive-region iteration mechanism that reduces universe cardinality during iterations by eliminating confirmed positive-region objects, thereby decreasing localized time complexity. Comparative evaluations of algorithm performance and execution time on UCI datasets

demonstrate that the proposed approach significantly reduces computational overhead while preserving reduct consistency.

Keywords

Rough Set, Neighborhood Relations, Attribute Reduction, Efficient Algorithm

Copyright © 2025 by author(s) and Hans Publishers Inc.

This work is licensed under the Creative Commons Attribution International License (CC BY 4.0).

<http://creativecommons.org/licenses/by/4.0/>



Open Access

1. 引言

粗糙集中的属性约简[1], 作为单决策表数据分析的核心降维方法之一, 已广泛应用于知识发现、模式识别及智能决策等领域, 并在实际场景中展现出显著优势。因其基于二元关系的数学建模特性, 该方法通过严格保持数据分类能力的一致性, 能够自适应地剔除冗余属性, 尤其适用于高维、噪声数据的特征选择。

多数数据集的高维特征中混杂着冗余和噪声属性, 而属性约简可以有效缓解这一问题, 并显著提高模型的学习效率。针对属性约简中诸多问题, 例如求解约简效率低, 学者们提出了多种有效的技术来解决这些问题。Chen [2]等人扩展了优势邻域粗糙集(DNRS)模型以支持数值与分类属性间的偏序关系建模, 并提出一种基于并行计算的属性约简方法。Yong [3]等人采用基于欧氏距离的分桶机制与邻域局部搜索优化方法, 构建了分桶优化的邻域粗糙集模型, 并提出正域快速迭代算法以及快速属性约简框架。Wang [4]等人用基于类标签标准差加权的样本质量评估方法, 提出了加权 K 近邻邻域粗糙集模型(WKNRS)和结合前向贪心搜索策略的加权 K 近邻特征选择算法(WKNFS), 有效改善了传统邻域粗糙集对噪声敏感的问题。Qian [5]等人引入基于粗糙集理论的正向近似理论框架, 设计了一种加速启发式属性约简过程的通用算法, 该框架能够提升现有启发式算法的计算效率。Wang [6]等人提出基于邻域关系基数的邻域区分指数及其变体(联合、条件、互区分指数), 结合邻域半径参数分析实值数据的区分能力, 并基于此定义了特征显著性度量, 设计了一种前向贪心特征选择算法。Hu [7]等人通过引入基于属性与决策相关性的动态权重分配机制, 提出了加权邻域粗糙集模型(WNRS), 设计融合贪心搜索与等距阈值优化的属性约简算法。

上述工作中分别针对不同的问题提出了相对应的解决措施, 一定程度上缓解了混合数据、噪声敏感以及大规模数据上存在的一些问题, 推动属性约简技术向高效、鲁棒、自适应发展。本文进一步聚焦时间效率优化问题, 提出基于逐层收缩正域迭代机制的邻域粒度熵约简算法: 通过动态排除已确定正域对象降低论域基数, 结合邻域粒度熵量化分类不确定性, 在 UCI 数据集实验中验证了算法在保证约简一致性的同时, 显著降低时间开销。

2. 基本概念

定义 1: 四元组 $ISDT = (U, F = C \cup D, V, \phi)$ 表示一个决策信息表, 其中 $U = \{o_1, o_2, \dots, o_n\}$ 表示非空有限的样本集合, 也称为论域; $C = \{f_1, f_2, \dots, f_l\}$ 表示条件属性集; $D = \{d\}$ 表示决策属性集; V 为属性集合 F 的值域; ϕ 为映射关系, 对于 $\forall att \in F, \forall o \in U, \phi(att, o)$ 表示样本 o 在属性 att 下的取值。

定义 2 [8]: 给定决策信息表 $ISDT = (U, F = C \cup D, V, \phi)$, 对于 $\forall o \in U$, γ 表示邻域半径, $B \subseteq C$ 是一个条件属性子集, 则对象 o 在论域 U 下关于条件属性集 B 的邻域定义为:

$$\delta_B^U(o) = \{u \in U \mid dis_B(o, u) \leq \gamma\} \quad (1)$$

其中 $dis_B(o, u)$ 表示对象 o 和 u 之间的距离。

定理 1: 对于任意的条件属性子集 $P \subseteq Q \subseteq C$, $\forall o \in U$ 满足 $\delta_Q^U(o) \subseteq \delta_P^U(o)$ 。

证明: 易证, 此处省略。

定义 3: 给定四元组 $ISDT = (U, F = C \cup D, V, \phi)$ 为决策信息表, $\pi_D^U = \{\pi_D^U(o_1), \pi_D^U(o_2), \dots, \pi_D^U(o_n)\}$ 表示论域 U 在决策属性下的划分, π_D 是基于二元等价关系的划分, 即对于任意对象属于且仅属于一个决策类内。

定义 4 [9]: 四元组 $ISDT = (U, F = C \cup D, V, \phi)$ 表示一个决策信息表, 条件属性子集 $B \subseteq C$ 对于任意的非空集合 $X \subseteq U$ 的下近似和上近似分别定义为:

$$\underline{B}^U(X) = \{u \in U \mid \delta_B^U(u) \subseteq X\} \quad (2)$$

$$\overline{B}^U(X) = \{u \in U \mid \delta_B^U(u) \cap X \neq \emptyset\} \quad (3)$$

具体地, 条件属性子集 B 关于决策属性 D 的下近似和上近似分别定义为:

$$\underline{B}^U(D) = \{u \in U \mid \delta_B^U(u) \subseteq \pi_D^U(u)\} \quad (4)$$

$$\overline{B}^U(D) = \{u \in U \mid \delta_B^U(u) \cap \pi_D^U(u) \neq \emptyset\} \quad (5)$$

其中 $\pi_D(u)$ 表示样本 u 的决策类, 即 $u \in \pi_D^U(u) \in \pi_D^U$ 。

定义 5 [10]: 在决策信息表 $ISDT = (U, F = C \cup D, V, \phi)$ 中, 对于任意的条件属性子集 $B \subseteq C$ 相对于决策属性集 D 在论域 U 下的条件粒度熵定义为:

$$CGH^U(D|B) = \frac{1}{|U|} \sum_{u \in U} \left(\frac{|\delta_B^U(u) \cap (\pi_D^U(u))^c|}{|U|} \right) \quad (6)$$

其中 $(\pi_D^U(u))^c$ 表示 $\pi_D^U(u)$ 在论域 U 下的补集。

定理 2: 对于任意的条件属性子集 $P \subseteq Q \subseteq C$, 满足 $CGH^U(D|Q) \leq CGH^U(D|P)$ 。

证明: 易证, 此处省略。

定义 6: 假设四元组 $ISDT = (U, F = C \cup D, V, \phi)$ 表示一个多标记决策表, 对于已选条件属性子集 $S \subseteq C$, $s \in (F - S)$ 表示候选属性, 则候选属性 s 在论域 U 下的外部属性重要度定义为:

$$Sig_{out}^U(s, S, D) = CGH^U(D|S) - CGH^U(D|S \cup s) \quad (7)$$

定义 7: 假设四元组 $ISDT = (U, F = C \cup D, V, \phi)$ 表示一个多标记决策表, 对于任意内部属性 $s \in C$ 在论域 U 下的内部属性重要度定义为:

$$Sig_{in}^U(s, C, D) = CGH^U(D|S - s) - CGH^U(D|S) \quad (8)$$

定义 8: 给定四元组 $ISDT = (U, F = C \cup D, V, \phi)$, 对于条件属性子集 $B \subseteq C$, B 称为四元组 $ISDT$ 的一个约简如果满足:

- 1) $CGH^U(D|B) = CGH^U(D|C)$;
- 2) 对 $\forall b \in B$, $CGH^U(D|B - b) \neq CGH^U(D|C)$ 。

3. 基于正域逼近的高效属性约简算法

定义 9 [5]: 四元组 $ISDT = (U, F = C \cup D, V, \phi)$ 表示一个决策信息表, $P_i = \{F_1, F_2, \dots, F_i\}$ 表示一个逐

渐递增的属性集合簇，即 $F_1 \succ F_2 \succ \dots \succ F_i$ 。 F_i 表示 P_i 内属性划分最精细的属性集合。则 P_i 关于集合 $X \subseteq U$ 的下近似定义为：

$$\underline{P}_i(X) = \bigcup_{k=1}^i \underline{P}_k(X_k) \quad (9)$$

其中 $X_1 = X$ ， $X_k = X - \bigcup_{j=1}^{k-1} \underline{P}_j(X_j)$ 。

根据定义 9，可以通过减少每次迭代所需要遍历的对象数目从而减少计算约简结果所需要的时间开支，但需保证在第 k 次迭代时所选择的属性与经典算法相比是相同的。

定理 3： 四元组 $ISDT = (U, F = C \cup D, V, \phi)$ 表示一个决策信息表， U_k 表示第 k 次迭代的论域，对于已选属性集合 $S \subseteq C$ 以及任意的候选属性 $a, b \in (F - S)$ ，如果在论域 U 上 $Sig_{out}^U(a, S, D) > Sig_{out}^U(b, S, D)$ 成立，则在论域 U_k 上 $Sig_{out}^{U_k}(a, S, D) > Sig_{out}^{U_k}(b, S, D)$ 也成立。

其中 $U_k = U - \bigcup_{i=1}^k \underline{P}_i(X)$ 。

证明： 当第 k 次迭代时， $s \in (F - S)$ 表示候选属性，原始可化为：

$$\begin{aligned} Sig_{out}^U(s, S, D) &= CGH^U(D|S) - CGH^U(D|S \cup s) \\ &= \frac{1}{|U|} \sum_{u \in U} \left(\frac{|\delta_s^U(u) \cap (\pi_D^U(u))_U^C|}{|U|} - \frac{|\delta_{S \cup s}^U(u) \cap (\pi_D^U(u))_U^C|}{|U|} \right) \\ &= \frac{1}{|U|^2} \sum_{u \in U} \left| (\delta_s^U(u) - \delta_{S \cup s}^U(u)) \cap (\pi_D^U(u))_U^C \right| \\ &= \frac{1}{|U|^2} \sum_{u \in U_k} \left| (\delta_s^U(u) - \delta_{S \cup s}^U(u)) \cap (\pi_D^U(u))_U^C \right| + \sum_{u \in U - U_k} \left| (\delta_s^U(u) - \delta_{S \cup s}^U(u)) \cap (\pi_D^U(u))_U^C \right| \end{aligned}$$

对于任意的 $o \in U - U_k$ ，易知： $(\delta_s^U(o) - \delta_{S \cup s}^U(o)) \cap (\pi_D^U(o))_U^C = \emptyset$ ；

此外当 $o \in \delta_s^U(u) \wedge o \notin \delta_{S \cup s}^U(u) \wedge o \notin \pi_D^U(u)$ 时，根据对称性可知 $u \in \delta_s^U(o)$ ，又因为 $o \notin \pi_D^U(u)$ ，可以推导出： $\delta_s^U(o) \not\subset \pi_D^U(o)$ 与 $o \in U - U_k$ 相矛盾；其余情况易知： $o \notin (\delta_s^U(u) - \delta_{S \cup s}^U(u)) \cap (\pi_D^U(u))_U^C$ ，即 U 删除的对象并不影响计算 $u \in U$ ， $(\delta_s^U(u) - \delta_{S \cup s}^U(u)) \cap (\pi_D^U(u))_U^C$ 。

原始可化为：

$$\begin{aligned} Sig_{out}^U(s, S, D) &= CGH^U(D|S) - CGH^U(D|S \cup s) \\ &= \frac{1}{|U|^2} \sum_{u \in U_k} \left| (\delta_s^U(u) - \delta_{S \cup s}^U(u)) \cap (\pi_D^U(u))_U^C \right| + \sum_{u \in U - U_k} \left| (\delta_s^U(u) - \delta_{S \cup s}^U(u)) \cap (\pi_D^U(u))_U^C \right| \\ &= \frac{1}{|U|^2} \sum_{u \in U_k} \left| (\delta_s^{U_k}(u) - \delta_{S \cup s}^{U_k}(u)) \cap (\pi_D^{U_k}(u))_{U_k}^C \right| \\ &= \frac{|U_k|^2}{|U|^2} \frac{1}{|U_k|^2} \sum_{u \in U_k} \left| (\delta_s^{U_k}(u) - \delta_{S \cup s}^{U_k}(u)) \cap (\pi_D^{U_k}(u))_{U_k}^C \right| \\ &= \frac{|U_k|^2}{|U|^2} Sig_{out}^{U_k}(s, S, D) \end{aligned}$$

因为二者比率等于两个论域的比率为固定值。即在删除对象迭代的过程中，在迭代过程中所选择的属性与经典算法相比是相同的。

根据以上的描述，在邻域关系下提出了基于正域逼近的高效属性约简算法，其具体算法内容如表 1 所示。

Table 1. Positive region approximation-based efficient attribute reduction algorithm

表 1. 基于正域逼近的高效属性约简算法

输入： 多标记决策表 $ISDT$ ，邻域半径 γ 。

输出： $MLDS$ 关于新的论域 U' 的一个约简 Red 。

1: 初始化 $B \leftarrow \emptyset$ ， $i = 0$ ， P ；

2: 根据邻域半径 γ 更新每个样本的邻域；

3: 对于 $\forall ak \in C$ ：

3.1: 计算在论域 U' 上的内部属性重要度 $Sig_m(ak, B, L)$ ；

3.2: 选择 $Sig_m(ak, S, D) > 0$ 执行 $B \leftarrow B - ak$ ；

4: 当 $CGH^{U_i}(D|B) \neq CGH^{U_i}(D|C)$ 时，重复执行：

4.1: $i += 1$ ；对于候选属性 $\forall ak \in F - B$ ，计算其外部属性重要度 $Sig_{out}^{U_i}(ak, B, L)$ ；

4.2: 选择 $ak' = \max\{Sig_{out}^{U_i}(ak, B, L)\}$ 执行 $B \leftarrow B \cup \{ak'\}$ ，更新 P ；

4.3: $U_i = U - \bigcup_{k=1}^i P_k(D)$

5: $Red \leftarrow B$ ；

6: 返回关于论域 U 的约简 Red 。

4. 实验分析

在本节中，我们进行了一系列实验。实验在 Windows 11 操作系统、Intel Core i7-13700H CPU 和 16GB DDR5 内存的环境下执行，所有算法均使用 Python 3.11 实现。

基于属性重要性测度的保序性，所有改进型属性约简算法与原算法生成的属性约简结果具有理论一致性。因此，本实验聚焦于验证算法在约简结果等价性与计算效率两个维度的性能差异，分类精度不作为比较指标，主要针对约简结果一致性以及约简效率进行对比分析。数据集的详细信息如表 2 所示，所有数据集均从真实透明的 UCI 数据集下载获得。

Table 2. Datasets

表 2. 数据集

序号	数据集	样本数	属性数
1	Wine_red	1599	11
2	BreastCancer	699	8
3	Sonar	208	60
4	Lonosphere	351	32
5	Image segment	210	19
6	Absenteeism at work	741	21

4.1. 算法约简结果一致性验证

为了严谨验证本文提出的加速策略在保持算法核心功能方面的有效性，本研究设计了系统性对比验证方案。实验通过控制变量方法，在相同实验环境下对比了加速算法与原始非加速算法在属性约简任务中的表现。具体而言，实验设置中严格保持了两类算法的核心参数配置与终止条件，仅针对加速策略引入的优化模块进行差异性测试。这种设计确保了实验结果差异的唯一来源为加速机制本身，从而能够准确评估其是否改变算法的本质行为。

实验结果表明，加速算法在各项对比维度中均展现出与原始算法的高度一致性。这种一致性不仅体现在约简集的静态属性构成上，更反映在动态任务场景中的功能等价性。值得注意的是，加速策略通过引入创新的计算优化机制，在不影响约简过程收敛方向的前提下，显著减少了冗余计算路径的探索。理论分析表明，该策略通过减少每次迭代所需遍历样本数量，有效降低了特征评估过程中的时间复杂度，这一优势在高维数据处理场景中尤为突出。本验证工作从实践层面证实了加速策略在算法加速与结果保真之间的平衡能力，为相关优化方法的研究提供了可复现的验证范式。其具体约简结果以及参数在表 3 中得以体现。

Table 3. Comparative analysis of reduction consistency
表 3. 约简一致性对比

序号	PRAR		AR	
	约简结果	半径	约简结果	半径
1	{10, 2, 5, 9}	0.01	{10, 2, 5, 9}	0.01
2	{3, 0, 4, 2, 6}	0.05	{3, 0, 4, 2, 6}	0.05
3	{20, 36, 29, 53, 55}	0.1	{20, 36, 29, 53, 55}	0.1
4	{28, 25, 21, 2, 3, 16}	0.1	{28, 25, 21, 2, 3, 16}	0.1
5	{0, 1, 5, 11, 13, 14, 15, 16, 17, 18, 10, 8}	0.1	{0, 1, 5, 11, 13, 14, 15, 16, 17, 18, 10, 8}	0.1
6	{1, 2, 3, 4, 9, 10, 0, 6}	0.1	{1, 2, 3, 4, 9, 10, 0, 6}	0.1

表 3 所展示的实验结果表明了 PRAR 加速算法与 AR 非加速算法相比，在相同的半径下以及在所有实验的数据集上均取得了相同的约简结果。六组实验中，PRAR 与 AR 的约简集合元素完全重合，特征编号及数量严格一致，在多种异构数据集下两类算法的结果同步性未受参数影响；即使是含 12 个特征的大规模约简结果(序号 5)，二者仍保持完全相同的复杂集合结构该结果直接证明了加速策略在优化计算效率的过程中，完整保留了原始算法的核心逻辑与结果生成机制。

4.2. 算法效率对比试验

在本小节中主要针对算法的效率即算法运行时间的结果展开对比，与上一小节约简结果一致性验证的实验环境相同，本小节展示了不同的属性约简算法得到其约简结果的运行时间。其具体的实验结果见表 4。实验结果表明，PRAR 加速算法具有普适性：在半径参数从 0.01 (数据集 1)到 0.1 (数据集 3~6)的变化范围内，PRAR 算法始终展现出显著的时间优势。例如在数据集 1 中，PRAR 将计算耗时从 264.72 秒缩短至 205.63 秒，效率提升 22.3%；而在数据集 6 中，运行时间从 63.29 秒降至 38.49 秒，加速比达 1.64 倍。最后通过两个小节的实验验证结果表明，加速算法在保证约简结果相同的同时与非加速算法相比均有明显的加速效果。由此得出本文提出的加速算法具备有效性和高效性。

Table 4. Algorithm efficiency comparison**表 4.** 算法效率对比

序号	PRAR		AR	
	算法运行时间/s	半径	算法运行时间/s	半径
1	205.6317	0.01	264.7153	0.01
2	19.0512	0.05	32.5395	0.05
3	27.3375	0.1	37.5631	0.1
4	32.2824	0.1	50.9737	0.1
5	3.6416	0.1	5.1651	0.1
6	38.4943	0.1	63.2857	0.1

5. 结论

本文针对属性约简技术在高维数据处理中的时间效率瓶颈问题，提出了一种基于邻域粗糙集理论的高效优化算法。通过设计逐层收缩的正域迭代机制，动态剔除已确认的正域样本以缩减论域规模，有效降低局部计算复杂度；同时引入邻域粒度熵作为分类不确定性的量化指标，优化特征评估准则。在 UCI 标准数据集上的实验表明，该算法在不影响约简结果一致性的前提下，显著缩短了计算耗时。相较于传统方法，其主要差异在于：一方面通过动态论域降维策略减少冗余计算路径，另一方面借助邻域粒度熵的敏感性提升分类边界判定效率。

基金项目

本文受烟台市科技计划项目(编号：2022XDRH016)的资助。

参考文献

- [1] Pawlak, Z. (1982) Rough Sets. *International Journal of Computer & Information Sciences*, **11**, 341-356. <https://doi.org/10.1007/bf01001956>
- [2] Chen, H., Li, T., Cai, Y., Luo, C. and Fujita, H. (2016) Parallel Attribute Reduction in Dominance-Based Neighborhood Rough Set. *Information Sciences*, **373**, 351-368. <https://doi.org/10.1016/j.ins.2016.09.012>
- [3] Yong, L., Wenliang, H., Yunliang, J. and Zhiyong, Z. (2014) Quick Attribute Reduct Algorithm for Neighborhood Rough Set Model. *Information Sciences*, **271**, 65-81. <https://doi.org/10.1016/j.ins.2014.02.093>
- [4] Wang, N. and Zhao, E. (2024) A New Method for Feature Selection Based on Weighted k-Nearest Neighborhood Rough Set. *Expert Systems with Applications*, **238**, Article 122324. <https://doi.org/10.1016/j.eswa.2023.122324>
- [5] Qian, Y., Liang, J., Pedrycz, W. and Dang, C. (2010) Positive Approximation: An Accelerator for Attribute Reduction in Rough Set Theory. *Artificial Intelligence*, **174**, 597-618. <https://doi.org/10.1016/j.artint.2010.04.018>
- [6] Wang, C., Hu, Q., Wang, X., et al. (2017) Feature Selection Based on Neighborhood Discrimination Index. *IEEE Transactions on Neural Networks and Learning Systems*, **29**, 2986-2999. <https://doi.org/10.1109/TNNLS.2017.2710422>
- [7] Hu, M., Tsang, E.C.C., Guo, Y., Chen, D. and Xu, W. (2021) A Novel Approach to Attribute Reduction Based on Weighted Neighborhood Rough Sets. *Knowledge-Based Systems*, **220**, Article 106908. <https://doi.org/10.1016/j.knosys.2021.106908>
- [8] Hu, Q., Yu, D., Liu, J. and Wu, C. (2008) Neighborhood Rough Set Based Heterogeneous Feature Subset Selection. *Information Sciences*, **178**, 3577-3594. <https://doi.org/10.1016/j.ins.2008.05.024>
- [9] Raza, I., Jamal, M.H., Qureshi, R., Shahid, A.K., Vistorte, A.O.R., Samad, M.A., et al. (2024) Adaptive Neighborhood Rough Set Model for Hybrid Data Processing: A Case Study on Parkinson's Disease Behavioral Analysis. *Scientific Reports*, **14**, Article No. 7635. <https://doi.org/10.1038/s41598-024-57547-4>
- [10] Dai, J., Huang, W., Wang, W. and Zhang, C. (2023) Semi-Supervised Attribute Reduction Based on Label Distribution and Label Irrelevance. *Information Fusion*, **100**, Article 101951. <https://doi.org/10.1016/j.inffus.2023.101951>