Knowledge-Based Systems 91 (2016) 204–218

Contents lists available at ScienceDirect

Knowledge-Based Systems

journal homepage: www.elsevier.com/locate/knosys

Generalized attribute reduct in rough set theory

Xiuyi Jia[a], Lin Shang[b,*], Bing Zhou[c], Yiyu Yao[d]

[a] School of Computer Science and Engineering, Nanjing University of Science and Technology, Nanjing 210094, China
[b] State Key Laboratory for Novel Software Technology, Nanjing University, Nanjing 210023, China
[c] Department of Computer Science, Sam Houston State University, Huntsville 77341, USA
[d] Department of Computer Science, University of Regina, Regina S4S 0A2, Canada

ARTICLE INFO

Article history:
Received 18 December 2014
Revised 13 April 2015
Accepted 16 May 2015
Available online 21 May 2015

Keywords:
Attribute reduction
Rough set
Generalized definition

ABSTRACT

Attribute reduction plays an important role in the areas of rough sets and granular computing. Many kinds of attribute reducts have been defined in previous studies. However, most of them concentrate on data only, which result in the difficulties of choosing appropriate attribute reducts for specific applications. It would be ideal if we could combine properties of data and user preference in the definition of attribute reduct. In this paper, based on reviewing existing definitions of attribute reducts, we propose a generalized attribute reduct which not only considers the data but also user preference. The generalized attribute reduct is the minimal subset which satisfies a specific condition defined by users. The condition is represented by a group of measures and a group of thresholds, which are relevant to user requirements or real applications. For the same data, different users can define different reducts and obtain their interested results according to their applications. Most current attribute reducts can be derived from the generalized reduct. Several reduction approaches are also summarized to help users to design their appropriate reducts.

1. Introduction

Attribute reduction is a key concept in rough set theory [28]. It plays an important role in many areas including machine learning, data mining, and knowledge representation. Specifically, rough sets can be used to construct granular structures in the area of granular computing [35,58]. Attribute reduction has been drawing broad attention in recent years, which can be classified into two groups: One group concentrates on seeking quick reduction algorithms to compute the reducts efficiently [4,6,8,9,11,12,23,17,18,22,25,27,31,32,44,52,54,60,67]; The other group focuses on the definition of reduct to find appropriate reducts for different applications [7,13,14,16,20,21,24,26,28,33,36,40,45–47,56,64–66]. In this paper, we aim to conduct an investigation on the definition of attribute reduct.

Why do we have so many different definitions of attribute reduct? In real applications, for the same data, different users may have different learning tasks, leading to the fact that the learned results are possibly different. Thus, many kinds of attribute reducts have been defined to meet different needs.

What are the differences among these reducts? Generally speaking, an attribute reduct can be interpreted as a minimal set of attributes that can preserve or improve one or several criteria. Different attribute reducts were defined based on different criteria.

Although so many different reducts have been studied, they still cannot be applied directly in some simple applications with user requirement. For example, the rules derived based on the Pawlak's reduct are all certain rules, which is the result of requiring the positive region remain unchanged. Assuming a typical situation, a rule is acceptable to users if its confidence is greater than 80%, then the Pawlak's reduct is no longer suitable, and we cannot find any appropriate reduct from existing definitions for this application. For the sake of a better understanding of this problem, we convert it to the following question.

How to choose or define appropriate reducts for different users in different applications?

In general, previous studies on definition of attribute reduct focus on selecting what kinds of criteria or properties of data to keep unchanged or to extend, such as distribution of objects, quality of classification, and so on. However, all these attribute reducts are relevant to the data only, but irrelevant to the application problem.

* Corresponding author.
E-mail addresses: jiaxy@njust.edu.cn (X. Jia), shanglin@nju.edu.cn (L. Shang), zhou@shsu.edu (B. Zhou), yyao@cs.uregina.ca (Y. Yao).

http://dx.doi.org/10.1016/j.knosys.2015.05.017
0950-7051/

In this paper, we argue that the criteria for definition of reduct should be connected with both the data and the user requirements. Actually, the requirements represent the user's preference and demand for the problem. Based on reviewing existing studies, we propose a generalized attribute reduct which considers both properties of data and user requirements in real applications. The contribution of the generalized reduct is that it can instruct users to define appropriate attribute reducts to meet their requirements.

By considering the user preference on learning data, the generalized attribute reduct is interpreted as a minimal subset of attributes which satisfies a specific condition. The condition is represented by a set of measures and a set of thresholds. Most existing attribute reducts can be derived from our definition through constructing corresponding measures and thresholds. Moreover, the measures and the thresholds can be provided by users or domain experts, which indicate the user requirements in real applications.

The rest of the paper is organized as follows. Section 2 summarizes existing attribute reducts and compares them through experiments. In Section 3, we introduce the definition of our generalized attribute reduct, and discuss some properties of the definition. In Section 4, we briefly review some measures based on rule induction and give a detailed illustration to show how to define an appropriate reduct that users really want. Section 5 derives the existing reducts from our definition. Section 6 introduces several reduction approaches. Section 7 concludes the paper.

2. Summary of existing definitions of attribute reducts

In this section, we summarize existing definitions of attribute reducts, and compare these definitions through experiments.

2.1. Existing definitions of attribute reducts

To summarize existing different definitions of attribute reducts, we classify these definitions into two groups from a decision perspective. One group contains those definitions that are decision-independent. Zhang et al. [64] proposed the theory of attribute reduction in the concept lattice and examined the judgement theorems of consist sets. Wu [47] discussed the attribute reduct in incomplete information systems and incomplete decision systems based on Dempster–Shafer theory of evidence, and introduced the plausibility reduct and belief reduct. Quafafou [33] defined an α-reduct in *Alpha Rough Set Theory* based on α-dependency, which preserves the dependency relation unchanged. Chen [3] introduced a concept of part reduct to describe the minimal description of a definable set by attributes of the given information system.

The other group contains definitions that are decision-dependent. These definitions are usually applied in classification problems. This group can be further divided into two categories. In the first category, the purpose of attribute reduction is to obtain a minimal subset of attributes that has the same classification power as the entire set of condition attributes. Pawlak [30] defined a quantitative reduct which ensures the classification ability unchanged, in which γ is used to represent the quality of classification. Śle¸zak proposed a concept of reduct to find the majority decision rules, he also defined the attribute reduct that keeps the class membership distribution unchanged for all objects by using the membership distribution function [41,42]. Zhang et al. [63] proposed the notions of the distribution reduct and maximum distribution reduct. Mi et al. [24] introduced the concepts of β lower distribution reduct and β upper distribution reduct based on variable precision rough sets. Their reducts preserve the lower distribution and the upper distribution of the decision class unchanged. These attribute reducts are summarized as a minimal subset of attributes that has the same classification power in terms of generalized decision, majority decision, or maximum distribution for all objects in the universe, they concentrated on the decision class or classes to which an equivalence class belongs [66].

In the second category, the reduct is interpreted as a minimal subset of attributes that keeps the positive, boundary and negative regions of decision classes, or other criteria unchanged or extended [66]. Pawlak defined a reduct of knowledge be the essential part which suffices to define all basic concepts. The Pawlak's reduct keeps the positive region unchanged [28,30]. Miao et al. [25] studied the mutual information as the reduction criterion, which is actually a kind of conditional entropy. Wang et al. [45] discussed attribute reduct from an algebra viewpoint and an information viewpoint. In the algebra view, a reduct is a minimal subset of attributes that keeps the positive region unchanged. In the information view, a reduct is a minimal subset of attributes that keeps the conditional entropy unchanged. Hu et al. [13] defined the consistency based attribute reduct, considered the distribution of each decision class under the precondition of keeping positive region unchanged. Jiang and Lu [16] gave two new definitions of reducts based on two concepts: mean decision power and decision information entropy. Xu and Sun [49] constructed a new conditional entropy based reduct to reflect the change of decision ability objectively in a decision table. For the definitions of reducts in Pawlak rough set model, the boundary region and the negative region were usually not considered. However, for the reducts defined in probabilistic rough sets, all three regions were considered with different decision region rules [56]. In decision-theoretic rough set model, Li et al. [20] proposed a positive region expanded attribute reduct, as the monotonicity of positive region does not always hold. Jia et al. [14,15] introduced a minimum cost attribute reduct which can minimize the decision cost.

Additionally, some researchers focused on the relationships between different definitions. Kryszkiewicz [19] compared several different reducts, analyzed the relationships between them, and generalized that a reduct is a minimal subset of attributes that satisfying some specific criteria. Miao et al. [26] discussed several definitions of reducts based on consistent and inconsistent data, and introduced corresponding algorithms. Some researchers also have done some works on the generality of reducts. Yao and Zhao [56] introduced a generalized reduct in probabilistic rough set models, which is a minimal subset of attributes that satisfying some criteria. Wang et al. [46] generalized the equivalence relation to a binary relation, and defined an attribute reduct based on the binary relation. Śle¸zak [40] also suggested that some measures can be the criteria for defining a reduct, such as information entropy.

Existing definitions of attribute reducts are summarized in Table 1.

2.2. Experimental comparisons of attribute reducts

In this section, we check the performances of different kinds of reducts on several criteria through some comparison experiments.

2.2.1. Comparison reducts

There are 22 different reducts in Table 1, they can be further grouped according to some criteria. A coarse and a fine approach to group all reducts are shown in Tables 2 and 3. Several typical reducts will be selected as the comparison algorithms.

The addition–deletion method[1] is applied to implement these definitions. The method starts with an empty set and uses inner significance to rank the attributes. As the fuzzy information relevant definition needs more expert opinions and the formal context relevant definition is not easy to apply the addition–deletion method, we

[1] The reduction approach will be explained in Section 6.

Table 1
Brief description of existing reduct definitions.

ID	Reduct definition	Brief description
1	Pawlak reduct [28]	Keep the positive region or the quality of classification unchanged
2	Reduct of generalized information [40]	Preserve the value of generalized inference measure function
3	ε-Reduct of generalized information [40]	Approximate reduct with tolerance condition
4	Distribution reduct [63]	Keep the membership distribution function unchanged
5	Maximum distribution reduct [63]	Preserve all maximum decision classes
6	Part reduct [3]	keep the collection of all the definable sets unchanged
7	β Lower approximate reduct [24]	Preserve the sum of objects in β lower approximations of all decision classes
8	β Upper approximate reduct [24]	Preserve the sum of objects in β upper approximations of all decision classes
9	β Lower distribute reduct [24]	Preserve the β lower approximations of all decision classes
10	β Upper distribute reduct [24]	Preserve the β upper approximations of all decision classes
11	α Reduct [33]	α Dependencies hold:
12	Reduct in formal contex [64]	A consistent set of the formal context
13	Belief reduct [47]	Keep the belief function value unchanged
14	Plausibility reduct [47]	Keep the plausibility function value unchanged
15	Positive region expanded reduct [20]	Expand the positive region
16	Mean decision power based reduct [16]	Increase the mean decision power
17	Decision information entropy based reduct [16]	Decrease the decision information entropy
18	New conditional information entropy based reduct [49]	Maximize the new conditional information entropy with an upper limit
19	Conditional entropy based reduct [25]	Keep the conditional entropy unchanged
20	Consistency based reduct [13]	Preserve the consistency
21	Generalized probabilistic reduct [56]	Keep the partial order relation unchanged based on multiple measures
22	Minimum cost attribute reduct [14]	Minimize the decision cost

Table 2
A coarse approach to group reduct definitions.

Criterion	Qualitative definitions	Quantitative definitions
ID	1, 2, 4, 5, 6, 12, 13, 14	3, 7, 8, 9, 10, 11, 15, 16, 17, 18, 19, 20, 21, 22

Table 3
A fine approach to group reduct definitions.

Criterion	Region	Distribution	Rule	Entropy
ID	1, 6, 7, 8, 13, 14, 15	2, 4, 5, 9, 10	16	3, 17, 18, 19
Criterion	Consistency	Cost	Fuzzy	Formal context
ID	20	22	11	12

do not consider them in our experiments. 6 typical definitions are compared and the ID numbers are: 1, 5, 16, 19, 20, 22. We use PRPAR (ID 1: Positive Region Preserved Attribute Reduction), MDPAR (ID 5: Maximum Distribution Preserved Attribute Reduction), MDPIAR (ID 16: Mean Decision Power Increased Attribute Reduction), CEPAR (ID 19: Conditional Entropy Preserved Attribute Reduction), CPAR (ID 20: Consistency Preserved Attribute Reduction) and MCAR (ID 22: Minimum Cost Attribute Reduction) to denote the corresponding algorithms.

Table 4
Brief description of data sets.

Dataset (abbreviation)	$\lvert U \rvert$	$\lvert C \rvert$	$\lvert D \rvert$
Congressional voting records (voting)	435	16	2
Credit approval (credit)	690	15	2
Hepatitis (hepatitis)	155	19	2
Ionosphere (ionosphere)	351	34	2
Blood transfusion service center (transfusion)	748	5	2
Wisconsin diagnostic breast cancer (wdbc)	569	30	2
Wisconsin prognostic breast cancer (wpbc)	198	33	2
Glass identification (glass)	214	9	6
Hayes-roth (hayes)	132	4	3
Iris (iris)	150	4	3
Balance scale (balance)	625	4	3
Breast tissue (breast)	106	9	6
Car evaluation (car)	1728	6	4
Contraceptive method choice (contraceptive)	1473	9	3
Annealing (annealing)	798	38	5
Splice-junction gene sequences (splice)	3190	60	4

2.2.2. Data sets and experimental setting

In our experiments, 16 UCI data sets [68] are used, with information shown in Table 4. $\lvert U \rvert$ denotes the number of objects, $\lvert C \rvert$ denotes the number of condition attributes and $\lvert D \rvert$ denotes the number of classes.

All algorithms are implemented in the WEKA (version 3.5) platform. As most algorithms can deal with nominal data only, and there exist missing values and obvious redundant attributes (e.g. id number) in some data sets, data preprocessing procedure has to be considered. We remove the id attributes from data sets first, including *wdbc*,*wpbc*,*glass*,*hayes* and *splice*. We also replace the missing values and discretize all continuous attributes by using WEKA filters.[2]

Three classifiers including Naive Bayes (NB), C4.5 and k-Nearest Neighbor algorithm (KNN) are selected to test the classification accuracies based on different reducts. All classifiers are implemented in WEKA with default parameters, and 10-fold cross validation is applied. For the minimum cost attribute reduction (MCAR), let $\lambda_{PN} = 5, \lambda_{NP} = 4, \lambda_{BN} = 1, \lambda_{BP} = 2$ and $\lambda_{PP} = \lambda_{NN} = 0$, which means $\alpha = 0.67, \beta = 0.33$ and $\gamma = 0.56$.

2.2.3. Experimental results

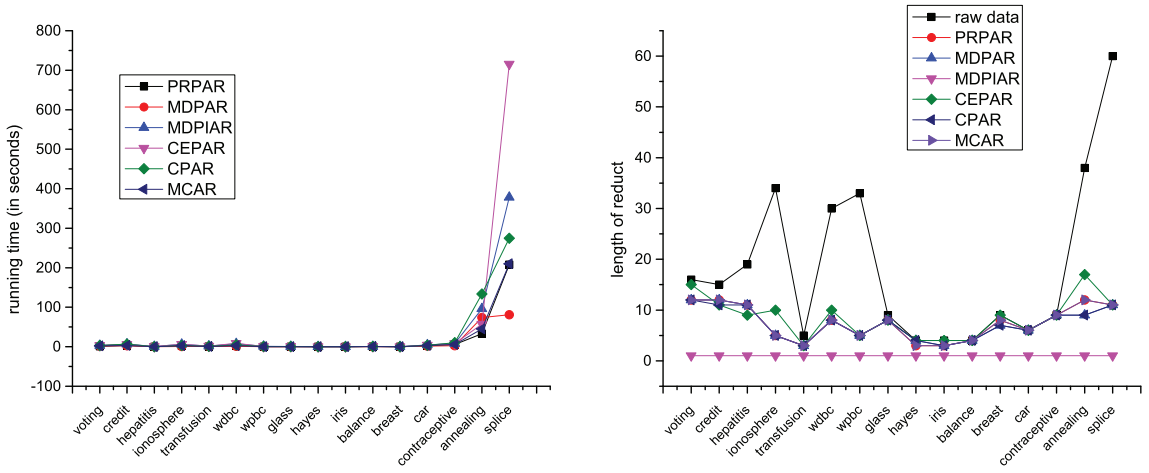
All experimental results are presented in Figs. 1–3 and Tables 8–15.[3] Fig. 1(a) gives the results of running time. Fig. 1(b) shows the comparison results of reduct lengths based on different reduct definitions. Fig. 2 are the comparison results of classification accuracies based on NB, C4.5 and KNN, respectively. Fig. 3 are the results of cardinalities of positive regions, entropies, and consistencies based on different reduct definitions. The detailed information can also be found in corresponding tables.

From these figures and tables, we can observe the following results:

- Mean decision power increased attribute reduction (MDPIAR) always obtains one attribute as the reduct. Based on the sole attribute, it is an expected result that MDPIAR gets the minimal classification accuracy. But surprisingly, the average accuracy still has an acceptable value: > 0.6.
- From the average results, we can see that conditional entropy preserved attribute reduction (CEPAR) obtains the longest reduct, and consistency preserved attribute reduction (CPAR) obtains

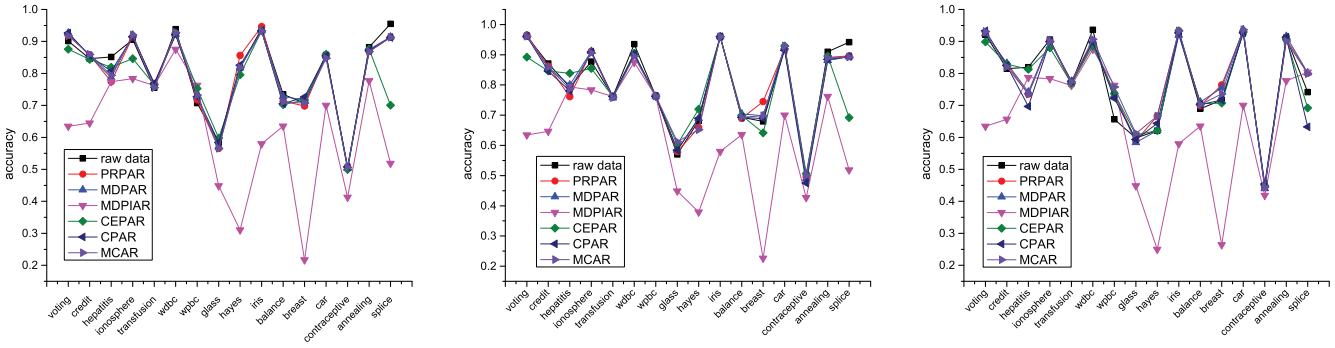
[2] The discretization is based on the unsupervised method and default parameters are adopted.

[3] Tables 8–15 are in Appendix.



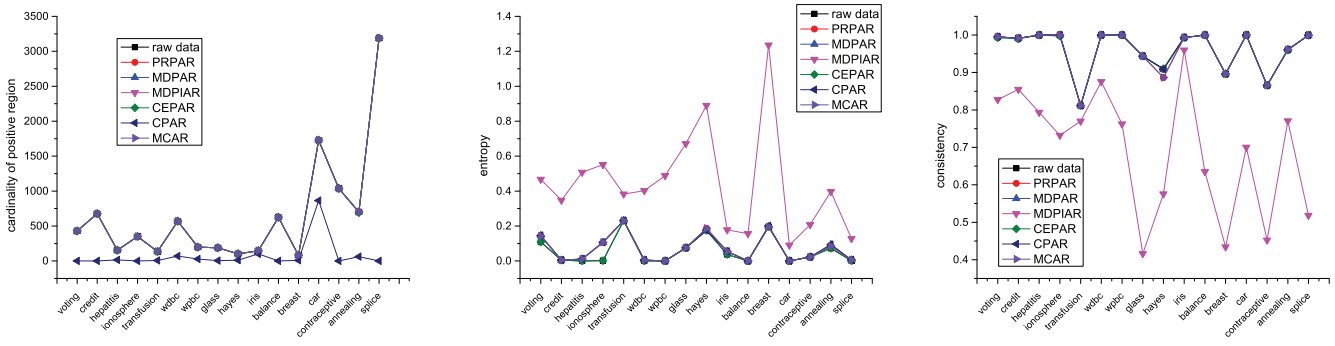
(a) running time (b) length of reduct

Fig. 1. Comparison of running time (in seconds) and lengths of reducts based on different reduct approaches.



(a) accuracy of NB (b) accuracy of C4.5 (c) accuracy of KNN

Fig. 2. Comparison of classification accuracies based on NB, C4.5 and KNN.



(a) cardinality of positive region (b) entropy (c) consistency

Fig. 3. Comparison of cardinalities of positive regions, entropies and consistencies based on different reducts.

the shortest reduct (except MDPIAR). MDPIAR obtains the slowest result and maximum distribution preserved attribute reduction (MDPAR) obtains the fastest result on running time measure.

- It is hard to say which attribute reduction can obtain the best classification accuracy based on different classifiers. Even for MDPIAR, it can still obtain the best accuracy on some data sets.
- Different classifiers prefer different reducts. Compared to NB and C4.5, KNN prefers short reducts. Based on the raw data, NB and C4.5 can obtain the best accuracy, but KNN does not. Based on the sole attribute provided by MDPIAR, KNN obtains the average accuracy value 0.6336, which is greater than NB (0.6148) and C4.5 (0.6209). For other 5 reducts, NB and C4.5 show a better performance rather than KNN.

- Positive region preserved attribute reduction (PRPAR) induces the largest positive region. Conditional entropy preserved attribute reduction (CEPAR) obtains the minimal entropy. Consistency preserved attribute reduction (CPAR) has the maximal consistency.

2.3. Remarks on existing definitions of different attribute reducts

Based on the above analysis on existing definitions and experimental results, we have following remarks on these attribute reducts:

- Classification accuracy is a common criterion to measure attribute reduct. Besides reduct, classifier also has influence on this measure. The experimental results show that no reduct is optimal among these reducts. All 6 compared reducts were not defined based on the classification accuracy measure.
- We argue that if a criterion is applied to define a reduct and to measure all other reducts, the corresponding reduct defined based on the criterion will obtain the best value. The experimental results on the cardinalities of positive regions, entropies, consistencies, and misclassification costs [14] can support this argument.

In order to answer the last question proposed in the introduction, based on reviewing the existing different definitions of attribute reducts and the comparison experiments, a generalized attribute reduct is proposed and discussed in the rest of this paper.

3. Generalized attribute reduct

Information and knowledge are usually represented in an information table, where a set of objects are described by a set of attributes [28].

Definition 1 (Information table)**.** An information table is the following tuple:

$$IS = (U, At, \{V_a | a \in At\}, \{I_a | a \in At\}), \quad (1)$$

where U is a finite nonempty set of objects, we call it the universe. At is a finite nonempty set of attributes. $At = C \cup D, C$ is a set of condition attributes and D is a set of decision attributes, $C \cap D = \varnothing$. This kind of table is called a *decision table* in classification problems. V_a is a nonempty set of values for an attribute $a \in At$, and $I_a : U \to V_a$ is an information function.

Example 1. Table 5 is a decision table, while $U = \{x_1, x_2, x_3, x_4\}, C = \{c_1, c_2, c_3\}$, and D is the decision class.

The goal of the generalized attribute reduct is to find the minimal subset of attributes that satisfies a *specific condition*, which is connected with the application and the data, usually the condition is defined by the user or domain experts.

3.1. Definition of generalized attribute reduct

Let S denote the specific condition defined by the user and S_A denote the condition under the set of attributes $A \subseteq At$, then the generalized attribute reduct is defined as following:

Definition 2 (Semantic definition of the generalized reduct)**.** In an information table, S is the specific condition defined by the user, a set of attributes $R \subseteq At$, we call R is an attribute reduct of the table if the following conditions are satisfied:

(i) S_R is satisfied;
(ii) for any subset $R' \subset R, S_{R'}$ is not satisfied.

The condition S is represented by a set of measures and a set of thresholds. We call S is satisfied when the learning result from the given data meets the thresholds requirement on the corresponding measures. Otherwise, S is not satisfied.

For the condition S, we can define appropriate measures and thresholds to quantify. For example, if a user wants to find a reduct which requires the quality of classification is greater than 90%, the measure is the "quality of classification" and the threshold is "90%". Definition 2 expresses the semantic of the reduct and it can be seen as a qualitative definition. For the sake of a better understanding of this definition, we consider a typical situation and give a formal definition. Assuming S means the values of all measures on the reduct are greater than or equal to the corresponding thresholds.

First, some symbols and notions are introduced. $M = \{m_1, m_2, \ldots, m_n\}$ is a set of measures defined by users or experts. $T(M) = \{t_1(m_1), t_2(m_2), \ldots, t_n(m_n)\}$ is a set of variables with respect to the set of measures M, it can be shorten as $T = \{t_1, t_2, \ldots, t_n\}$, where $t_i \in T$ is the threshold of the measure $m_i \in M$, which can be provided by users or learned from data. $E_A(M) = \{e_A^1(m_1), e_A^2(m_2), \ldots, e_A^n(m_n)\}$ is a set of variables, it can be shorten as $E_A = \{e_A^1, e_A^2, \ldots, e_A^n\}$, while $e_A^i \in E_A$ is the value of $m_i \in M$ on the data based on the set of attributes A. $T, E_A \subseteq \mathbb{R}^n$. Then S can be denoted by a set of propositions: $\{e_A^i \geq t_i\}, m_i \in M$. For $\forall m_i \in M$, proposition $e_A^i \geq t_i$ holds, then S is satisfied. Otherwise, S is not satisfied.

A mapping function $L : 2^{\mathbb{R}^n} \times 2^{\mathbb{R}^n} \to \{0,1\}$ is defined to represent S. $L(E_A, T) = 1$ denotes $e_A^i \geq t_i$ holds for $\forall m_i \in M$, which means S is satisfied, $L(E_A, T) = 0$ means S is not satisfied.

Now the generalized attribute reduct is redefined as following:

Definition 3 (Generalized reduct definition)**.** In an information table, a set of attributes $R \subseteq At$, we call R is an attribute reduct of the table if the following conditions are satisfied:

(i) $L(E_R, T) = 1$;
(ii) for any subset $R' \subset R, L(E_{R'}, T) = 0$.

Example 2. In Table 5, a user may want to find the reduct R which keeps the conditional entropy unchanged [45]. Under this situation, the condition S can be formalized as $H(D \mid R) = H(D \mid C)$, then we can let $M = \{H(D \mid A)\}, T = \{H(D \mid C)\}$. By computing the conditional entropy based on different condition attributes, we have $H(D \mid C) = 0, E_{\{c_1,c_2\}} = \{H(D \mid \{c_1, c_2\})\} = \{0\}, E_{\{c_1\}} = \{1\}, E_{\{c_2\}} = \{1\}$. Then $L(E_{\{c_1,c_2\}}, T) = 1, L(E_{\{c_1\}}, T) = 0, L(E_{\{c_2\}}, T) = 0$, we have the conclusion that $\{c_1, c_2\}$ is an attribute reduct of Table 5 meeting the user's requirement.

Table 5
An information table.

Object	c_1	c_2	c_3	D
x_1	1	0	1	1
x_2	1	1	1	0
x_3	0	0	0	1
x_4	0	1	1	1

3.2. Analysis on the definition

3.2.1. Generalized operators

Besides E_R and T are two variables of function L, there exists a set of comparison operators and a set of logical operators in L implicitly. In Section 3.1, S is denoted by a set of propositions: $\{e_A^i \geq t_i\}$. To simplify our definition, $\geqslant$ is the unique comparison operator between the value of each measure e_A^i and the threshold t_i, and $\wedge$ is the unique logical operator among the propositions $(e_A^1 \geq t_1) \wedge (e_A^2 \geq t_2) \wedge \cdots \wedge (e_A^n \geq t_n)$. Generally speaking, one may consider other types of comparison and logical operators to define the attribute reduct according to the requirement of the application problem. For example, $=, \neq, <, >, \leq, \geq$ and other comparison operators can be combined to apply, $\vee, \wedge$ and $\neg$ can be used as the logical operator.

3.2.2. Jointly sufficient constraint of reduct

A definition of an attribute reduct usually contains two constraints, one is jointly sufficient constraint and the other one is individually necessary constraint [56]. In a traditional definition of an attribute reduct, there always exists a subset of attributes $R \subseteq At$ that satisfies the jointly sufficient constraint. Since the traditional attribute reduct is a minimal set of attributes that provides a specific ability as the entire set of attributes At, in the worst situation, $R = At$. Consequently, the jointly sufficient constraint satisfaction guarantees the existence of a reduct.

For the generalized attribute reduct, it is the minimal set of attributes satisfying a specific condition S. If S is too strict, the jointly sufficient constraint may be violated, that means we cannot find a set of attributes R to satisfy the condition defined by users, even the entire set of attributes At. The reduct does not exist.

Example 3. In a decision table M, if there exist at least two objects that have the same values on C but different values on D, this kind of table is called an inconsistent decision table. If users want to find a reduct to induce all ***correct*** rules to cover ***all*** objects, the reduct does not exist, because there is no such a correct rule which can distinguish the inconsistent objects, even the rule is constructed based on the entire set $C \cup D$.

3.2.3. Individually necessary constraint of reduct

The individually necessary constraint guarantees that reduct is a minimal set of attributes. Any subset of reduct will violate the jointly sufficient constraint. The monotonicity of measures are usually analyzed to discuss the individually necessary constraint.

Based on the description in Section 3.1, a measure m_i is monotonic, if for any $A_1, A_2 \subseteq At$ such that $A_1 \subseteq A_2$ one has $e^i_{A_1} \leq e^i_{A_2}$. Obviously, the amalgamation of measures is monotonic if all measures have the same monotonicity. Assume the amalgamation of measures is monotonic, consider two subsets of attributes $A_1, A_2 \subseteq At$ with $A_1 \subseteq A_2$, then we have $L(E_{A_1}, T) \leq L(E_{A_2}, T)$. That is:

Property 1. *If* $L(E_{A_1}, T) = 1$, *then* $L(E_{A_2}, T) = 1$.

Property 2. *If* $L(E_{A_2}, T) = 0$, *then* $L(E_{A_1}, T) = 0$.

If the monotonicity of the amalgamation of measures holds, the condition (ii) in Definition 3 can be rewritten as:

(ii$*$) for any attribute $r \in R, L(E_{(R-\{r\})}, T) = 0$.

This condition is useful in reduction algorithms, which can reduce the time complexity for some deletion methods [54].

Followed by the monotonicity property, a theorem about the existence of reduct can be drawn as the following:

Theorem 1. If the amalgamation of measures is monotonic and the condition S can be satisfied by the entire set of attributes At, there exists at least one reduct. If the amalgamation of measures is monotonic and the condition S cannot be satisfied by At, no reduct exists.

Proof. The condition S can be satisfied by the entire set of attributes At, which means $L(E_{At}, T) = 1$. According to Property 1, we can find a subset $R \subseteq At$, which satisfies the Definition 3. The condition S cannot be satisfied by At, which means $L(E_{At}, T) = 0$. According to Property 2, for any subset $R \subseteq At, L(E_R, T) = 0$, no reduct exists. □

If the amalgamation of the measures is non-monotonic, we may have $L(E_{A_1}, T) \geq L(E_{A_2}, T)$ when $A_1 \subseteq A_2$. In order to get the minimal subset of attributes, the condition (ii) cannot be replaced by (ii$*$). We can induce another conclusion that for an amalgamation of the measures, which is non-monotonic and it cannot be satisfied by the entire set of attributes At, there may exist a reduct $R \subseteq At$.

4. Measures in rule induction

In this section, we introduce some common measures. For same data, different users may have different learning tasks, a typical one of them is rule induction, which is a common data mining method in classification problems.

4.1. Expression of a rule

A generalized information table was defined in [55], in which a family of binary relations were considered. In this paper, without loss of generality we only consider one special case of the binary relations: the equivalence relation.

In an information table $IS = \{U, At, \{V_a\}, \{I_a\}\}$, an equivalence relation with respect to $A \subseteq At$ is denoted as IND_A, $IND_A = \{(x,y) \in U \times U \mid \forall a \in A, (I_a(x) = I_a(y))\}$. Two objects in U satisfy IND_A if and only if they have the same values on all attributes in A. An equivalence relation is reflexive, symmetric and transitive.

The pair (U, IND_A) is called an approximation space defined by the set of attributes A. The equivalence relation IND_A induces a partition of U, denoted by U/IND_A or π_A. The equivalence class of U/IND_A containing x is given by $[x]_{IND_A} = [x]_A = \{y \in U \mid (x,y) \in IND_A\}$. Let $\pi_D = \{D_1, D_2, \ldots, D_n\}$ be a partition of the universe U, defined by the decision attribute set D, representing n classes.

Typically, a rule is expressed in the form of $[x]_A \rightarrow D_i$ or $[x] \rightarrow D_i$, stating that an object with description $[x]_A$ would be in the decision class D_i.

The confidence and generality of the learned rules are common used in rule induction as evaluation measures. Many quantitative measures associated with rules have been studied [1,5,53,56]. On rule evaluation, the authors also studied the connections between single rule and rule set evaluation measures [56,57]. Micro evaluation is based on single rule which can be measured by the common empirical measures, like *confidence*, *generality*, *coverage*, and so on. Macro evaluation is based on rule sets, depending on the relationships between rules in the set. In this paper, for simplicity's sake, we only consider confidence and generality here, the details of other measures can be found in [56,57].

4.2. Micro evaluation of a single rule

Micro evaluation focuses on a single rule and usually be treated as a criterion to choose the appropriate rules from rule set in many algorithms. For example, in algorithm PRISM [2], the rule which only covers instances of objective class is generated first; in ID3 [34], the leaf node means all instances of that are classified to correct classes. The correctness or confidence of a rule equals 1 is considered in these algorithms. Besides the confidence of a rule, the generality, coverage and cost are usually considered in different algorithms.

4.2.1. Confidence of a single rule

The confidence of the rule $[x]_A \rightarrow D_i$ can be defined as:

$$confidence([x]_A \rightarrow D_i) = \frac{|[x]_A \cap D_i|}{|[x]_A|} = P(D_i|[x]_A), \tag{2}$$

where $0 \leq confidence([x]_A \rightarrow D_i) \leq 1$. In classification problems, the confidence means the ratio of the number of objects in equivalence class $[x]_A$ that are correctly classified as decision class of D_i. The higher the confidence, the stronger the rule is.

4.2.2. Generality of a single rule

The generality of the rule $[x]_A \to D_i$ can be defined as:

$$generality([x]_A \to D_i) = \frac{|[x]_A|}{|U|}, \tag{3}$$

where $0 \le generality([x]_A \to D_i) \le 1$. The generality indicates the relative size of the subset $[x]_A$. A rule is more general if it covers more objects of the universe.

The confidence of a single rule is non-monotonic and the generality is monotonically decreasing, for any two subsets $A_1 \subseteq A_2 \subseteq C$, we have $[x]_{A_2} \subseteq [x]_{A_1}$, then $generality([x]_{A_2} \to D_i) \le generality([x]_{A_1} \to D_i)$.

The confidence and generality are not independent of each other, in many cases, one may find a trade-off between the confidence and the generality. A rule with higher confidence may have a lower generality, while a rule with higher generality may have a lower confidence. In most situations, decreasing the confidence of the rule followed by increasing the generality of the rule, this is exactly the idea of pruning an over-fitted rule to a more general rule by dropping off some descriptions of the rule [56].

4.3. Macro evaluation of a rule set

Micro evaluation can be used to evaluate a single rule for a rule learning algorithm, but in sometime it cannot evaluate the learned rule set correctly as it is only a local evaluation for a rule set. Macro evaluation focus on the performance of the rule set instead of the performance of each single rule.

4.3.1. Confidence of a rule set.

The confidence of the rule set *RS* can be interpreted as the ratio of number of correctly classified objects and the number of classified objects covered by all the rules in the set,

$$confidence(RS) = \frac{\#\text{ of correctly classified objects by } RS}{\#\text{ of classified objects by } RS}. \tag{4}$$

4.3.2. Generality of a rule set.

The generality of the rule set *RS* can be interpreted as the ratio of number of objects covered by the rule set and the number of objects in universe *U*,

$$generality(RS) = \frac{\#\text{ of objects covered by } RS}{\#\text{ of objects in } U}. \tag{5}$$

The confidence and generality of a rule set are not equal to the sum of the confidence and generality of each rule in most cases, because one object can be covered by two or more rules. The objects are covered by the rule set, not partitioned by the rule set. The detail of relation of the micro evaluation and macro evaluation can be found in [56,57].

4.4. Thresholds

Followed by appropriate measures, corresponding thresholds need to determined before defining an attribute reduct. Two kinds of strategies are usually applied to set threshold values. One is a subjective method that the values are provided by users or domain experts. For example, 90% for the confidence of each rule and 80% for the generality of the rule set, both real values express user's requirement for the learning result. The other strategy is learned from data, which is an objective method. The values of the given measures based on the entire set of attributes *At* are usually adopted in the reduct, like $H(D \mid C)$ in Example 2.

Table 6
A decision table.

Object	c_1	c_2	c_3	c_4	D
o_1	1	1	1	1	M
o_2	1	0	1	0	M
o_3	0	0	1	1	Q
o_4	1	1	1	0	Q
o_5	1	0	1	0	F
o_6	0	0	0	0	F
o_7	1	0	1	0	F

4.5. An illustration

In many real problems, noise refers to random errors in the data, either due to incorrect class labels or errors in object descriptions. The $confidence = 1$ for positive rules and γ measure for the Pawlak's reduct are very strict conditions. To conquer the over-fitting problem and obtain more general rules, the attribute reduct with tolerance of errors needs to be considered.

Table 6 is a decision table, where $U = \{o_1, \ldots, o_7\}, C = \{c_1, c_2, c_3, c_4\}$ is the set of condition attributes and D is the decision attribute. In this table, $POS_{\pi_C}(\pi_D) = \{o_1, o_3, o_4, o_6\}$. According to the definition of the Pawlak's reduct, $\{c_2, c_3, c_4\}$ is an attribute reduct of the table, then the partition $\pi_{\{c_2,c_3,c_4\}}$ induces the following positive rules:

$$\{o_1\} \to_P M, (confidence = 1);$$
$$\{o_3\} \to_P Q, (confidence = 1);$$
$$\{o_4\} \to_P Q, (confidence = 1);$$
$$\{o_6\} \to_P F, (confidence = 1).$$

$\{o_2, o_5, o_7\}$ is the boundary region of the table, where $\{o_2, o_5, o_7\} \to_B M$ and $\{o_2, o_5, o_7\} \to_B F$ are two conflict boundary rules. We just use the maximum-confidence criterion to solve the conflict. The confidence of $\{o_2, o_5, o_7\} \to_B M$ is 0.33 and the confidence of $\{o_2, o_5, o_7\} \to_B F$ is 0.67, then only a boundary rule is generated: $\{o_2, o_5, o_7\} \to_B F, confidence = 0.67$.

Assume a user wants to obtain an attribute reduct induced the rule set with its confidence is greater than 0.8, then $\{c_2, c_4\}$ is an attribute reduct of the table according to Definition 2, the partition $\pi_{\{c_2,c_4\}}$ induces the following rule set *RS*:

$$\{o_1\} \to_P M, (confidence = 1);$$
$$\{o_3\} \to_P Q, (confidence = 1);$$
$$\{o_4\} \to_P Q, (confidence = 1);$$
$$\{o_2, o_5, o_6, o_7\} \to_B F, (confidence = 0.75).$$

For the derived rule set from the reduct $\{c_2, c_4\}, confidence(RS) = 0.86$. We can say that $\{c_2, c_4\}$ is an attribute reduct meeting the user's requirement.

In the example, the threshold 0.8 is provided by the user, and different thresholds may obtain different reducts.

5. Derivation of existing reducts

In this section, most common attribute reducts are studied and explained as the special cases of our proposed generalized attribute reduct.

5.1. Reduct in Pawlak rough set model

5.1.1. Basic notions of Pawlak rough set model

Based on the information table IS which has been defined in Section 3, for a subset $X \subseteq U$, the lower and upper approximations of X with respect to π_A are defined as [28]:

$$\begin{aligned} \underline{apr}_{\pi_A}(X) &= \{x \in U | [x]_A \subseteq X\}, \\ \overline{apr}_{\pi_A}(X) &= \{x \in U | [x]_A \cap X \neq \varnothing\}. \end{aligned} \tag{6}$$

Three disjoint regions of X can be defined by:

$$\begin{aligned} \mathrm{POS}_{\pi_A}(X) &= \underline{apr}_{\pi_A}(X), \\ \mathrm{BND}_{\pi_A}(X) &= \overline{apr}_{\pi_A}(X) - \underline{apr}_{\pi_A}(X), \\ \mathrm{NEG}_{\pi_A}(X) &= U - \mathrm{POS}_{\pi_A}(X) \cup \mathrm{BND}_{\pi_A}(X) \\ &= U - \overline{apr}_{\pi_A}(X). \end{aligned} \tag{7}$$

where positive region $\mathrm{POS}_{\pi_A}(X)$ indicates the union of all the equivalence classes defined by π_A, that each for sure can induce the decision class X; the boundary region $\mathrm{BND}_{\pi_A}(X)$ indicates the union of all the equivalence classes defined by π_A, that each can induce a partial decision of X; and the negative region $\mathrm{NEG}_{\pi_A}(X)$ which is the union of all equivalence classes that for sure cannot induce the decision class X.

Given a partition $\pi_D = \{D_1, D_2, \ldots, D_n\}$ defined by the decision attribute set D, the lower and upper approximations of the partition π_D with respect to π_A are the families of the lower and upper approximations of all the equivalence classes of π_D. That is [29],

$$\begin{aligned} \underline{apr}_{\pi_A}(\pi_D) &= (\underline{apr}_{\pi_A}(D_1), \underline{apr}_{\pi_A}(D_2), \ldots \underline{apr}_{\pi_A}(D_n)), \\ \overline{apr}_{\pi_A}(\pi_D) &= (\overline{apr}_{\pi_A}(D_1), \overline{apr}_{\pi_A}(D_2), \ldots \overline{apr}_{\pi_A}(D_n)). \end{aligned} \tag{8}$$

For this n-class problem, $\mathrm{POS}_{\pi_A}(\pi_D)$ indicates the union of all the equivalence classes defined by π_A, that each for sure can induce a decision. $\mathrm{BND}_{\pi_A}(\pi_D)$ indicates the union of all the equivalence classes defined by π_A, that each can induce a partial decision. Formally, we have [29]:

$$\begin{aligned} \mathrm{POS}_{\pi_A}(\pi_D) &= \bigcup_{1 \le i \le n} \mathrm{POS}_{\pi_A}(D_i), \\ \mathrm{BND}_{\pi_A}(\pi_D) &= \bigcup_{1 \le i \le n} \mathrm{BND}_{\pi_A}(D_i), \\ \mathrm{NEG}_{\pi_A}(\pi_D) &= U - \mathrm{POS}_{\pi_A}(\pi_D) \cup \mathrm{BND}_{\pi_A}(\pi_D). \end{aligned} \tag{9}$$

In Pawlak rough set, $\mathrm{POS}_{\pi_A}(\pi_D) \cup \mathrm{BND}_{\pi_A}(\pi_D) = U$, then we have:

$$\mathrm{NEG}_{\pi_A}(\pi_D) = \varnothing. \tag{10}$$

Based on the notions of positive and boundary regions, positive rules and boundary rules are introduced:

1. Positive rule: If $[x]_A \subseteq \mathrm{POS}_{\pi_A}(\pi_D)$, the induced rule is a positive rule, denoted as:

   $[x]_A \to_P (D_i)$, where $D_i \in \pi_D$ and $[x]_A \subseteq \mathrm{POS}_{\pi}(D_i)$.

2. Boundary rule: If $[x]_A \subseteq \mathrm{BND}_{\pi_A}(\pi_D)$, the induced rule is a boundary rule, denoted as:
   $[x]_A \to_B (D_i)$, where $D_i \in \pi_D$ and $[x]_A \subseteq \mathrm{BND}_{\pi_A}(D_i)$.

5.1.2. Attribute reduct based on positive region

A classical Pawlak's reduct $R \subseteq C$, more precisely a relative reduct with respect to the decision attribute D, is defined by requiring that the positive region of π_D is unchanged [28].

Definition 4 (Pawlak's reduct)**.** Given a decision table $IS = (U, At = C \cup D, \{V_a\}, \{I_a\})$, a subset of condition attributes $R \subseteq C$ is called a Pawlak's reduct of C with respect to D, if R satisfies the following conditions:

(i) $\mathrm{POS}_{\pi_R}(\pi_D) = \mathrm{POS}_{\pi_C}(\pi_D)$;
(ii) for any $a \in R, \mathrm{POS}_{\pi_{R-\{a\}}}(\pi_D) \neq \mathrm{POS}_{\pi_C}(\pi_D)$.

As Definition 4 can be seen as a kind of qualitative definition, an equivalent quantitative definition of Pawlak's reduct were usually applied in the real applications [1,12,28,45]. The quantitative definition is based on the following measure: *quality of classification*.

$$\gamma(\pi_D | \pi_A) = \frac{|\mathrm{POS}_{\pi_A}(\pi_D)|}{|U|}. \tag{11}$$

The positive region has the following monotonicity [56]:

$$A_1 \subseteq A_2 \Rightarrow \mathrm{POS}_{\pi_{A_1}}(\pi_D) \subseteq \mathrm{POS}_{\pi_{A_2}}(\pi_D),$$

where $A_1 \subseteq A_2 \subseteq C$.

The monotonicity of the γ measure can also be induced as:

$$A_1 \subseteq A_2 \Rightarrow \gamma(\pi_D | \pi_{A_1}) \le \gamma(\pi_D | \pi_{A_2}).$$

By monotonicity, the condition (i) in Definition 4 can be re-expressed as:

$$\gamma(\pi_D | \pi_R) = \gamma(\pi_D | \pi_C). \tag{12}$$

According to Eq. (5), the *generality* of a Pawlak positive rule set PRS_A can be expressed as:

$$generality(\mathrm{PRS}_A) = \frac{|\mathrm{POS}_{\pi_A}(\pi_D)|}{|U|} = \gamma(\pi_D | \pi_A). \tag{13}$$

In Pawlak rough set, the *generality* of the positive rule set is the same as the γ measure.

As we mentioned above, the *confidence* of each positive rule in Pawlak rough set equals 1, according to Eqs. (2) and (4), the *confidence* of the positive rule set PRS_A in Pawlak rough set is computed as:

$$confidence(\mathrm{PRS}_A) = \frac{|\mathrm{POS}_{\pi_A}(\pi_D)|}{|\mathrm{POS}_{\pi_A}(\pi_D)|} = 1. \tag{14}$$

If $\mathrm{POS}_{\pi_A}(\pi_D) = \varnothing$, we assume $confidence(\mathrm{PRS}_A) = 1$. From the equation we can find that the *confidence* of the positive rule set is always 1, not relative to the γ measure.

The Pawlak's reduct can be re-expressed by the γ measure as following:

Definition 5 (Quantitative representation of Pawlak's reduct)**.** Given a decision table $IS = (U, At = C \cup D, \{V_a\}, \{I_a\})$, a subset of condition attributes $R \subseteq C$ is called a Pawlak's reduct of C with respect to D, if R satisfies the following conditions:

(i) $\gamma(\pi_D | \pi_R) = \gamma(\pi_D | \pi_C)$;
(ii) for any $a \in R, \gamma(\pi_D | \pi_{R-\{a\}}) \neq \gamma(\pi_D | \pi_C)$.

Now, $\gamma(\pi_D \mid \pi_A)$ can be seen as the measure and $\gamma(\pi_D \mid \pi_C)$ the threshold in our generalized attribute reduct with the equality operator. The generalized representation of Pawlak's reduct can be defined as:

Definition 6 (Generalized representation of Pawlak's reduct)**.** Given a decision table $IS = (U, At = C \cup D, \{V_a\}, \{I_a\})$, a subset of condition attributes $R \subseteq C$ is called a Pawlak's reduct of C with respect to D, if R satisfies the following conditions:

(i) $L(\{\gamma(\pi_D \mid \pi_R)\}, \{\gamma(\pi_D \mid \pi_C)\}) = 1$;
(ii) for any $a \in R, L(\{\gamma(\pi_D \mid \pi_{R-\{a\}})\}, \{\gamma(\pi_D \mid \pi_C)\}) = 0$.

In Pawlak rough set, if the reduct is based on the positive region, the form of positive rules induced from the reduct R can be written as: $[x]_R \rightarrow_P (D_i)$. Based on the reduct, the *confidence* of the rule set equals 1 and the value of *generality* is a constant. Similarly, the reduct can be defined equally by using another measure: the *generality* of the positive rule set. Consequently, it can be concluded that the Pawlak's reduct is a special case of our generalized attribute reduct.

5.2. Attribute reduct in probabilistic rough set models

Compared to Pawlak rough set model, probabilistic rough set models consider the statistical information of the overlap of an equivalence class and a set, while the overlap can be interpreted as making acceptance or rejection decisions for more objects with some tolerance of error. It is expressed by a pair of thresholds (α,β) with $\alpha > \beta$.

After giving or computing the two parameters α and β, the probabilistic lower and upper approximations can be defined by:

$$\begin{aligned} \underline{apr}_{\pi_{(\alpha,\beta)}}(X) &= \{x \in U | P(X|[x]) \geq \alpha\}, \\ \overline{apr}_{\pi_{(\alpha,\beta)}}(X) &= \{x \in U | P(X|[x]) > \beta\}. \end{aligned} \tag{15}$$

The probabilistic positive, boundary and negative regions are defined by:

$$\begin{aligned} \mathrm{POS}_{\pi_{(\alpha,\beta)}}(X) &= \underline{apr}_{\pi_{(\alpha,\beta)}}(X), \\ \mathrm{BND}_{\pi_{(\alpha,\beta)}}(X) &= \overline{apr}_{\pi_{(\alpha,\beta)}}(X) - \underline{apr}_{\pi_{(\alpha,\beta)}}(X), \\ \mathrm{NEG}_{\pi_{(\alpha,\beta)}}(X) &= U - \mathrm{POS}_{\pi_{(\alpha,\beta)}}(X) \cup \mathrm{BND}_{\pi_{(\alpha,\beta)}}(X) \\ &= U - \overline{apr}_{\pi_{(\alpha,\beta)}}(X). \end{aligned} \tag{16}$$

The probabilistic regions of a single decision to a partition π_D can be defined by:

$$\begin{aligned} \mathrm{POS}_{\pi_{(\alpha,\beta)}}(\pi_D) &= \bigcup_{1 \leq i \leq n} \mathrm{POS}_{\pi_{(\alpha,\beta)}}(D_i), \\ \mathrm{BND}_{\pi_{(\alpha,\beta)}}(\pi_D) &= \bigcup_{1 \leq i \leq n} \mathrm{BND}_{\pi_{(\alpha,\beta)}}(D_i), \\ \mathrm{NEG}_{\pi_{(\alpha,\beta)}}(\pi_D) &= U - \mathrm{POS}_{\pi_{(\alpha,\beta)}}(\pi_D) \cup \mathrm{BND}_{\pi_{(\alpha,\beta)}}(\pi_D), \end{aligned} \tag{17}$$

where the decision attributes $D = \{D_1, \ldots, D_n\}$.

In [56], two kinds of attribute reducts were defined, being parallel to Pawlak's reduct, one is a probabilistic attribute reduct definition by requiring that the probabilistic positive region of π_D is unchanged.

Definition 7 (Positive region based reduct in probabilistic rough set models)**.** Given a decision table $IS = \{U, At = C \cup D, \{V_a\}, \{I_a\}\}$, a subset of condition attributes $R \subseteq C$ is a reduct of C with respect to D if it satisfies the following two conditions:

(i) $\mathrm{POS}_{\pi_{R_{(\alpha,\beta)}}}(\pi_D) = \mathrm{POS}_{\pi_{C_{(\alpha,\beta)}}}(\pi_D)$;
(ii) for any $a \in R, \mathrm{POS}_{\pi_{R-\{a\}_{(\alpha,\beta)}}}(\pi_D) \neq \mathrm{POS}_{\pi_{C_{(\alpha,\beta)}}}(\pi_D)$.

The other is a general definition by considering multiple criteria. For one criterion, a particular measure is used as its indicator. The measure, roughly denoted as $e: 2^C \rightarrow (L, \succeq)$, maps a set of condition attributes to an element of a poset L, which is equipped with the partial order relation $\succeq$. Based on the partial order relation, the evaluation of a reduct $R \subseteq C$ with respect to e is the same or superior to $e(C)$, and the evaluation of any subset of R with respect to e is inferior to $e(C)$. By considering multiple criteria and multiple measures, the general definition of an attribute reduct is described as follows [56].

Definition 8 (Generalized reduct in probabilistic rough set models)**.** Given a decision table $IS = \{U, At = C \cup D, \{V_a\}, \{I_a\}\}$. Suppose, we can evaluate the properties of M by a set of measures $E = \{e_1, e_2, \ldots\}$. A subset $R \subseteq C$ is a reduct of C with respect to D if it satisfies the following two conditions:

(i) $e(\pi_D \mid \pi_R) \succeq e(\pi_D \mid \pi_C)$ for all $e \in E$;
(ii) for any subset $R' \subset R, e(\pi_D \mid \pi_{R'}) \prec e(\pi_D \mid \pi_C)$ for all $e \in E$.

The authors explained the general definition through three criteria, *decision-monotonicity*, *generality* and *cost*, the detail can be found in [56].

Compared with our proposed generalized attribute reduct in Definitions 3 and 8 also considers the combination of different measures, and the main difference is how to define the threshold of the measure. In Definition 8, the threshold is the corresponding value of the measure under the entire attribute set C. In our proposed generalized reduct, the threshold can be provided by users or learned from data, and the value of the measure under C is just one choice. In general, Definition 8 is a special case of Definition 3. If users do not know how to define the threshold, Definition 8 provides a good solution.

5.3. Distribution reduct and maximum distribution reduct

In [40], Śle¸zak presented an attribute reduct that preserves the class membership distribution for all objects in universe. In [63], the authors proposed an attribute reduct called a maximum distribution reduct, which preserves all maximum decision classes.

In a decision table $IS = \{U, At = C \cup D, \{V_a\}, \{I_a\}\}, U/E_D = \{D_1, \ldots, D_n\}$, a membership distribution function $\mu_A : U \rightarrow [0,1]^n$ is defined as follows:

$$\mu_A(x) = (D(D_1/[x]_A), \ldots, D(D_n/[x]_A)), x \in U, \tag{18}$$

where

$$D(D_j/[x]_A) = \frac{|D_j \cap [x]_A|}{|[x]_A|}. \tag{19}$$

The $\mu_A(x)$ is a probability distribution on U/E_D. $m_A(x)$ is the degree of confidence of uncertainty rule (or boundary rule): $[x]_A \rightarrow_B D_{j_i}$, defined by:

$$m_A(x) = \max\{D(D_j/[x]_A) : j \leq n\} = D(D_{j_i}/[x]_A), x \in U. \tag{20}$$

The maximum decision function γ_A is defined as follows:

$$\gamma_A(x) = \{D_{j_i} : D(D_{j_i}/[x]_A) = m_A(x)\}, x \in U. \tag{21}$$

Then the distribution reduct and maximum distribution reduct were defined as following [63]:

Definition 9 (Distribution reduct)**.** In a decision table IS, a subset $R \subseteq C$, if $\forall x \in U, \mu_R(x) = \mu_C(x)$, we say that R is a distribution consistent set of IS. If R is a distribution consistent set and no proper subset of R is distribution consistent, then R is referred to as a distribution reduct of IS.

Definition 10 (Maximum distribution reduct)**.** In a decision table IS, a subset $R \subseteq C$, if $\forall x \in U, \gamma_R(x) = \gamma_C(x)$, we say that R is a maximum distribution consistent set of IS. If R is a maximum distribution consistent set and no proper subset of R is maximum distribution consistent, then R is referred to as a maximum distribution reduct of IS.

Table 7
Using generalized reduct to derive existing 22 reduct definitions.

ID	$e_R(m) \in E_R$	Operator	$t(m) \in T$
1	$\gamma(\pi_D \mid \pi_R)$	$=$	$\gamma(\pi_D \mid \pi_C)$
2	$G_n^A(R)$	$=$	$G_n^A(At)$
3	$\mid G_n^A(R) - G_n^A(At) \mid$	$\leqslant$	ε
4	$\mu_R(x)$	$=$	$\mu_C(x)$
5	$\gamma_R(x)$	$=$	$\gamma_C(x)$
6	$D(U,R)$	$=$	$D(U,At)$
7	σ_R^β	$=$	σ_C^β
8	λ_R^β	$=$	λ_C^β
9	L_R^β	$=$	L_C^β
10	H_R^β	$=$	H_C^β
11	$IND(R,\alpha,\beta)$	$=$	$IND(At,\alpha,\beta)$
12	$L(U,R,I_R))$	$\cong$	$L(U,At,I)$
13	$Bel_R(X)$	$=$	$Bel_{At}(X)$
14	$Pl_R(X)$	$=$	$Pl_{At}(X)$
15	$\mid \mathrm{POS}_R^{(\alpha,\beta)}(D) \mid$	$\geqslant$	$\mid \mathrm{POS}_C^{(\alpha,\beta)}(D) \mid$
16	σ_R	$\geqslant$	σ_C
17	$H(R \to D)$	$\leqslant$	$H(C \to D)$
18	$HG(RD;R)$	$\leqslant$	$HG(RD;C)$
19	$H(D \mid R)$	$=$	$H(D \mid C)$
20	$\delta_R(D)$	$=$	$\delta_C(D)$
21	$e(\pi_D \mid \pi_R)$	$\succeq$	$e(\pi_D \mid \pi_C)$
22	COST_R	$\leqslant$	COST_C

Both the distribution reduct and the maximum distribution reduct are special cases of the generalized reduct, where the measures are membership distribution function μ_R and the maximum decision function γ_R, and the thresholds are μ_C and γ_C, respectively.

5.4. Derivation of other existing reduct definitions

In this subsection, most of current existing reduct definitions are summarized. Table 1 shows the brief descriptions of different reduct definitions. Table 7 tells us how to use generalized reduct to derive the existing reducts. In Table 7, the column "$e_R(m) \in E_R$" means the measure used in each reduct definition, the column "$t(m) \in T$" denotes the corresponding threshold value, and the column "Operator" gives the comparison relation between the measure and the threshold.

6. Reduction approaches

Besides different definitions of attribute reduct, several reduction approaches were also studied by many researchers. The purpose of all these studies is to obtain the correct reduct efficiently. In this section, we summarize the existing reduction approaches and help users design appropriate reduction algorithms to compute their reducts.

By reviewing the existing algorithms, we can classify these algorithms into two groups by their purpose: getting all reducts and getting one reduct.

6.1. Getting all reducts

Getting all reducts is an NP-hard problem [38], one of the most important algorithms is the discernibility matrix algorithm. As a beautiful theoretical result, discernibility matrix algorithm for attribute reduction was developed by Skowron and Rauszer [39]. In a discernibility matrix, both the rows and columns correspond to the objects, and the element of matrix is the set of all attributes that distinguish the corresponding object pairs. Then the set of attribute reducts are the set of prime implicants of the reduced disjunctive form of the discernibility function [39]. Many researchers studied reduction algorithms by using the discernibility matrix [9,44,51,59].

A classical discernibility matrix given by Skowron and Rauszer was defined as following.

Definition 11 (Discernibility matrix)**.** Given a decision table $IS = \{U,At = C \cup D,\{V_a\},\{I_a\}\}$, its discernibility matrix $DM = (DM(x,y))$ is a $\mid U \mid \times \mid U \mid$ matrix, in which $DM(x,y)$ for an object pair (x,y) is:

$$DM(x,y) = \{a \in C | [I_a(x) \neq I_a(y)] \wedge [I_D(x) \neq I_D(y)]\} \tag{22}$$

Based on the discernibility matrix, one can get the Pawlak's reduct through the following discernibility function.

$$f(DM) = \bigwedge \left\{ \bigvee (DM(x,y)) | \forall x,y \in U, DM(x,y) \neq \varnothing \right\} \tag{23}$$

The expression $\bigwedge\{\bigvee (DM(x,y))\}$ is the conjunction of all $\bigvee (DM(x,y))$, while $\bigvee (DM(x,y))$ is the disjunction of all condition attributes in $DM(x,y)$. The discernibility function can be transformed to a reduced disjunctive form, and each conjunctor of the reduced disjunctive form is a Pawlak's reduct [39].

6.2. Getting one reduct

As finding all reducts is an NP-hard problem, heuristic approaches are usually applied to speed up the reduction process and to get one reduct. A heuristic approach is composed of two parts: attribute reduction heuristics and search strategy. Attribute reduction heuristics is the fitness function of a heuristic approach, and search strategy is the control structure. Based on the definition of generalized reduct, we can find that heuristics are relative to the criteria and the corresponding thresholds. For a determined reduct, different heuristic algorithms usually have different search strategies with a same fitness function.

There are two basic search strategies in heuristic approaches, one is directional search strategy and the other is nondirectional search strategy. Directional search strategy contains three kinds of methods: deletion method, addition method, and addition–deletion method. Nondirectional search strategy is usually applied in evolutionary algorithms, swarm algorithms, and some other search algorithms for optimization problems.

Yao et al. [54] discussed the directional search strategy and proposed the generalized deletion method, addition method and addition–deletion method for reduct construction. The deletion method starts with the entire attribute set, and checks all attributes for deletion. A reduct will be gotten by deleting redundant attribute one by one. The addition method usually starts with the core or an empty set, and consequently adds attributes until it becomes a reduct. The addition–deletion method is a combination of the addition method and the deletion method. Adding attributes and deleting attributes are both applied in the reduct construction procedure.

For directional search strategy, the order of attributes for addition or deletion is essential for reduct construction. Different fitness functions determine different orders of attributes, and result in different reducts [54]. Usually, the significance of each attribute is applied. There are two kinds of definitions of significance, one is called inner significance and the other is outer significance.

Definition 12 (Inner Significance)**.** Given a decision table $IS = \{U,At = C \cup D,\{V_a\},\{I_a\}\}, B \subseteq C$ and $a \in B$. The inner significance of a in B based on measure e is defined as

$$Sig_{inner}(a,B,D) = \frac{|e_B(D) - e_{B-\{a\}}(D)|}{|e_B(D)|}. \tag{24}$$

Definition 13 (Outer Significance)**.** Given a decision table $IS = \{U,At = C \cup D,\{V_a\},\{I_a\}\}, B \subseteq C$ and $a \in C - B$. The outer significance of

a in B based on measure e is defined as

$$Sig_{outer}(a,B,D) = \frac{|e_{B\cup\{a\}}(D) - e_B(D)|}{|e_{B\cup\{a\}}(D)|}. \quad (25)$$

Compared to directional search strategy, nondirectional search strategy does not change the number of candidate attributes monotonically. The reduction procedure is usually represented as solving an optimization problem. The optimal result or an approximate result will be outputted as the reduct.

An evolutionary algorithm (EA) is a heuristic optimization algorithm using techniques inspired by mechanisms from organic evolution such as mutation, recombination, and natural selection to find an optimal configuration for a specific system within specific constraints. Genetic algorithm is the most popular type of evolutionary algorithm and has been studied for reduction by many researchers [14,22,37,48,50]. Swarm algorithms is a class of algorithms modeled on biological systems, such as ant colony optimization [10,18,61], particle swarm optimization [6,60], and so on. Some other search algorithms were also used to get reducts, including simulated annealing algorithm [14], tabu search [11], hill climbing [17], and so on.

6.3. Remarks on reduction approaches

Based on our proposed generalized attribute reduct, we find that the main difference of different reduct definitions lies in their criteria. According to the criteria, current definitions of different reducts can be classified into two categories: qualitative attribute reducts based on qualitative criteria and quantitative attribute reducts based on quantitative criteria. By studying the relationship of reduct definitions and reduction approaches, we have following conclusions:

* As the discernibility matrix represents the discernibility attributes between two objects, the discernibility matrix based approaches are appropriate for computing qualitative attribute reducts. One drawback of this kind of algorithm is that it is not easy to represent the statistical information and it is not suitable for getting quantitative attribute reducts. Usually, it also has a higher time complexity rather than heuristic approaches. The theoretical advantage of discernibility matrix based reduction approach is that it can get all reducts, and the practical advantage of it would be its easy implementability.
* Efficiency is the most important reason for choosing heuristic approaches to attribute reduction. As the various fitness functions, heuristic approaches are suitable for getting quantitative attribute reducts. Besides the quantitative attribute reducts, qualitative attribute reducts can also be obtained from heuristic algorithms. The reason is that qualitative attribute reducts can be transformed to corresponding quantitative attribute reducts. The shortcoming of heuristic approaches is that it is not easy to get all reducts at once. The worse thing is that sometimes it cannot guarantee the result is a reduct by violating the individually necessary constraint.
* There is no strict distinction between two reduction approaches. For example, discernibility matrix can be applied in heuristic approaches [8,54]. For a given reduct definition, we should choose appropriate heuristics (fitness function) and search strategy to design a correct and efficient reduction algorithm.
* In the era of big data, the classical reduction approaches are becoming more challenging from both data and various user requirements. Some parallel methods [31,62], incremental learning methods [43] and online learning methods can be considered in the reduction procedure.

7. Conclusion

In this paper, we focus on the problem of how to choose or define appropriate reducts for different users in different applications. To answer this question, we introduced a generalized framework of attribute reduct. Under this generalized framework, most existing attribute reducts can be derived by applying different conditions.

Formally, the generalized reduct is represented by a group of measures and a group of thresholds, while the measures and thresholds are relevant to user requirements on real applications. Besides the reduct definition, reduction approaches were also discussed. Discernibility matrix based algorithms for qualitative attribute reducts and heuristic algorithms for quantitative attribute reducts are mainly two kinds of reduction approaches. Users can design appropriate reduction algorithm based on their proposed definition of attribute reduct.

This study suggests that the attribute reduction should be associated with real applications and data. We hope the result can help users effectively define appropriate reducts meeting their requirements. In the future, defining different generalized attribute reducts in different granularity, and studying their relationships will be an important and interesting work for granular computing.

Acknowledgments

We would like to acknowledge the support for this work from the National Natural Science Foundation of China (Grant Nos. 61403200 and 61170180), Natural Science Foundation of Jiangsu Province (Grant No. BK20140800).

Appendix A. Experimental results

Tables 8–15

Table 8
Comparison of reduct lengths based on different reduct approaches.

Data sets	PRPAR	MDPAR	MDPIAR	CEPAR	CPAR	MCAR
Voting	12	12	1	15	12	12
Credit	12	12	1	11	11	12
Hepatitis	11	11	1	9	11	11
Ionosphere	5	5	1	10	5	5
Transfusion	3	3	1	3	3	3
wdbc	8	8	1	10	8	8
wpbc	5	5	1	5	5	5
Glass	8	8	1	8	8	8
Hayes	3	4	1	4	4	3
Iris	3	3	1	4	3	3
Balance	4	4	1	4	4	4
Breast	8	7	1	9	7	8
Car	6	6	1	6	6	6
Contraceptive	9	9	1	9	9	9
Annealing	12	9	1	17	9	12
Splice	11	11	1	11	11	11
Average length	7.5	7.3125	1	8.4375	7.25	7.5
Average rank	2.25	2.1875	1	2.5626	2.1875	2.25

Table 9
Comparison of running times based on different reduct approaches (in milliseconds).

Data sets	PRPAR	MDPAR	MDPIAR	CEPAR	CPAR	MCAR
Voting	1261	1861	3073	3246	3016	2276
Credit	3010	2638	6581	5897	6877	3067
Hepatitis	152	59	390	408	242	159
Ionosphere	2597	429	6269	6060	3013	2433
Transfusion	802	640	1613	1387	1282	820
wdbc	2712	947	7854	8500	4487	2646
wpbc	329	62	951	736	573	310
Glass	308	260	627	589	524	339
Hayes	18	11	32	28	25	20
Iris	32	34	110	106	74	48
Balance	374	178	491	419	374	368
Breast	30	16	31	32	28	22
Car	1973	1471	4479	4086	3566	2961
Contraceptive	5507	2646	8327	8206	10,291	5632
Annealing	33,012	73,947	96,194	57,338	133,753	45,363
Splice	207,951	80,949	378,472	715,651	274,902	210,023
Average time	16254.25	10384.25	32218.38	50793.06	27689.19	17280.44
Average rank	2.1875	1.3125	5.5	5	4.25	2.625

Table 10
Comparison of classification accuracies based on NB. The best results are highlighted by boldface.

Data sets	Raw data	PRPAR	MDPAR	MDPIAR	CEPAR	CPAR	MCAR
Voting	0.9011	0.9195	**0.9264**	0.6345	0.8759	**0.9264**	0.9172
Credit	0.8464	**0.8580**	0.8551	0.6449	0.8438	0.8551	0.8594
Hepatitis	0.8516	0.7742	0.7871	0.7742	**0.8194**	0.8065	0.7935
Ionosphere	0.9060	0.9145	**0.9202**	0.7835	0.8462	0.9145	**0.9202**
Transfusion	0.7553	**0.7687**	0.7620	0.7602	0.7647	**0.7687**	0.7553
wdbc	0.9385	0.9262	0.9228	0.8752	0.9262	0.9209	**0.9279**
wpbc	0.7071	0.7172	0.7273	**0.7626**	0.7525	0.7273	0.7323
Glass	0.5841	0.5654	0.5701	0.4486	**0.5981**	0.5841	0.5654
Hayes	0.8182	**0.8561**	0.8182	0.3106	0.7955	0.8258	0.8182
Iris	0.9400	**0.9467**	0.9333	0.5800	0.9333	0.9333	0.9333
Balance	0.7344	0.7136	**0.7280**	0.6352	0.7024	0.7056	0.7104
Breast	0.7075	0.6981	0.7170	0.2170	0.7170	**0.7264**	0.7075
Car	0.8576	0.8553	0.8542	0.7002	**0.8594**	0.8547	0.8472
Contraceptive	0.5058	0.5071	0.5037	0.4128	0.4997	**0.5078**	0.5058
Annealing	0.8822	0.8684	0.8747	0.7769	**0.8759**	0.8722	0.8684
Splice	0.9551	0.9137	**0.9140**	0.5187	0.7007	0.9121	0.9128
Average	0.8059	0.8002	0.8009	0.6148	0.7819	**0.8026**	0.7984

Table 11
Comparison of classification accuracies based on C4.5. The best results are highlighted by boldface.

Data sets	Raw data	PRPAR	MDPAR	MDPIAR	CEPAR	CPAR	MCAR
Voting	0.9609	**0.9655**	0.9632	0.6345	0.8920	0.9609	0.9609
Credit	0.8710	**0.8623**	0.8565	0.6464	0.8464	0.8464	0.8594
Hepatitis	0.7935	0.7613	0.8000	0.7935	**0.8387**	0.7806	0.7871
Ionosphere	0.8775	**0.9117**	**0.9117**	0.7835	0.8547	0.9088	0.9060
Transfusion	0.7620	**0.7620**	0.7580	**0.7620**	**0.7620**	**0.7620**	0.7580
wdbc	0.9350	0.8998	0.9033	0.8752	**0.9051**	0.9033	0.8946
wpbc	0.7626	**0.7626**	**0.7626**	**0.7626**	**0.7626**	**0.7626**	**0.7626**
Glass	0.5701	0.5794	0.5841	0.4486	0.6028	0.5841	**0.6122**
Hayes	0.6818	0.6591	0.6894	0.3788	**0.7197**	0.6894	0.6515
Iris	0.9600	**0.9600**	**0.9600**	0.5800	**0.9600**	**0.9600**	**0.9600**
Balance	0.6928	0.6896	**0.7040**	0.6352	0.6992	0.6928	0.6928
Breast	0.6792	**0.7453**	0.6981	0.2264	0.6415	0.6887	0.6981
Car	0.9248	0.9178	0.9259	0.7002	**0.9294**	0.9149	**0.9294**
Contraceptive	0.4942	0.4976	0.4895	0.4270	**0.5064**	0.4752	0.5003
Annealing	0.9098	0.8910	0.8810	0.7619	**0.8960**	0.8847	0.8897
Splice	0.9416	**0.8965**	0.8933	0.5187	0.6919	0.8936	0.8924
Average	0.8011	0.7976	**0.7988**	0.6209	0.7818	0.7943	0.7972

Table 12
Comparison of classification accuracies based on KNN. The best results are highlighted by boldface.

Data sets	Raw data	PRPAR	MDPAR	MDPIAR	CEPAR	CPAR	MCAR
Voting	0.9218	0.9287	0.9241	0.6345	0.8989	**0.9333**	0.9310
Credit	0.8145	0.8275	**0.8319**	0.6565	0.8290	0.8246	0.8203
Hepatitis	0.8194	0.7355	0.7419	0.7871	**0.8129**	0.6968	0.7355
Ionosphere	0.9060	**0.9031**	**0.9031**	0.7835	0.8803	0.9003	0.9003
Transfusion	0.7687	0.7660	**0.7754**	0.7620	0.7687	**0.7754**	0.7741
wdbc	0.9367	0.8963	0.8998	0.8752	0.8858	0.8981	**0.9069**
wpbc	0.6566	0.7374	0.7323	**0.7626**	0.7374	0.7222	0.7576
Glass	0.5981	0.5888	0.5841	0.4486	0.6075	0.5935	**0.6122**
Hayes	0.6212	**0.6667**	0.6212	0.2500	0.6212	0.6439	**0.6667**
Iris	0.9267	**0.9333**	0.9200	0.5800	**0.9333**	0.9267	**0.9333**
Balance	0.6896	0.6992	**0.7104**	0.6352	0.7056	0.7024	0.7024
Breast	0.7170	**0.7642**	0.7547	0.2642	0.7075	0.7170	0.7358
Car	0.9282	0.9334	**0.9369**	0.7002	0.9294	0.9271	0.9363
Contraceptive	0.4467	0.4447	0.4399	0.4182	0.4487	**0.4542**	0.4447
Annealing	0.9123	0.9060	**0.9160**	0.7769	0.9060	0.9135	0.9010
Splice	0.7415	0.8011	0.8020	**0.8036**	0.6919	0.6326	0.7989
Average	0.7753	0.7832	0.7809	0.6336	0.7728	0.7664	**0.7848**

Table 13
The sizes of positive regions based on different reducts.

Data sets	Raw data	PRPAR	MDPAR	MDPIAR	CEPAR	CPAR	MCAR
Voting	430	430	430	0	427	430	430
Credit	677	677	677	0	675	676	677
Hepatitis	155	155	155	16	155	155	155
Ionosphere	351	351	351	0	349	351	351
Transfusion	135	135	135	6	135	135	135
wdbc	569	569	569	71	569	569	569
wpbc	198	198	198	29	198	198	198
Glass	188	188	188	5	188	188	188
Hayes	102	102	102	12	102	102	102
Iris	146	146	146	101	146	146	146
Balance	625	625	625	0	625	625	625
Breast	80	80	80	10	80	80	80
Car	1728	1728	1728	864	1728	1728	1728
Contraceptive	1037	1037	1037	0	1037	1037	1037
Annealing	702	702	694	61	700	694	702
Splice	3188	3188	3188	0	3188	3188	3188
Average	644.44	644.44	643.94	73.44	643.81	643.81	644.44

Table 14
The comparison of entropies based on different reducts.

Data sets	Raw data	PRPAR	MDPAR	MDPIAR	CEPAR	CPAR	MCAR
Voting	0.1092	0.1428	0.1449	0.4669	0.1092	0.1449	0.1428
Credit	0.0046	0.0046	0.0046	0.3477	0.0046	0.0053	0.0046
Hepatitis	0	0.0111	0.0111	0.5073	0	0.0111	0.0111
Ionosphere	0.0020	0.1061	0.1061	0.5510	0.0020	0.1061	0.1061
Transfusion	0.2310	0.2310	0.2310	0.3827	0.2310	0.2310	0.2310
wdbc	0	0.0056	0.0056	0.4025	0	0.0056	0.0056
wpbc	0	0	0	0.4880	0	0	0
Glass	0.0756	0.0756	0.0756	0.6710	0.0756	0.0756	0.0756
Hayes	0.1752	0.1863	0.1752	0.8905	0.1752	0.1752	0.1863
Iris	0.0370	0.0551	0.0551	0.1785	0.0370	0.0551	0.0551
Balance	0	0	0	0.1560	0	0	0
Breast	0.1955	0.1955	0.1955	1.2358	0.1955	0.2005	0.1955
Car	0	0	0	0.0901	0	0	0
Contraceptive	0.0230	0.0230	0.0230	0.2074	0.0230	0.0230	0.0230
Annealing	0.0729	0.0854	0.0934	0.3971	0.0729	0.0934	0.0854
Splice	0.0013	0.0059	0.0059	0.1279	0.0013	0.0059	0.0059
Average	0.0580	0.0705	0.0704	0.4438	0.0580	0.0708	0.0705

Table 15
The comparison of consistencies based on different reducts.

Data sets	Raw data	PRPAR	MDPAR	MDPIAR	CEPAR	CPAR	MCAR
Voting	0.9954	0.9954	0.9954	0.8276	0.9931	0.9954	0.9954
Credit	0.9913	0.9913	0.9913	0.8551	0.9899	0.9913	0.9913
Hepatitis	1	1	1	0.7935	1	1	1
Ionosphere	1	1	1	0.7322	0.9972	1	1
Transfusion	0.8115	0.8115	0.8115	0.7701	0.8115	0.8115	0.8115
wdbc	1	1	1	0.8752	1	1	1
wpbc	1	1	1	0.7626	1	1	1
Glass	0.9439	0.9439	0.9439	0.4159	0.9439	0.9439	0.9439
Hayes	0.9091	0.8864	0.9091	0.5758	0.9091	0.9091	0.8864
Iris	0.9933	0.9933	0.9933	0.96	0.9933	0.9933	0.9933
Balance	1	1	1	0.6352	1	1	1
Breast	0.8962	0.8962	0.8962	0.4330	0.8962	0.8962	0.8962
Car	1	1	1	0.7002	1	1	1
Contraceptive	0.8656	0.8656	0.8656	0.4521	0.8656	0.8656	0.8656
Annealing	0.9612	0.9612	0.9612	0.7707	0.9599	0.9612	0.9612
Splice	0.9997	0.9997	0.9997	0.5182	0.9994	0.9997	0.9997
Average	0.9605	0.9590	0.9605	0.6924	0.9599	0.9605	0.9590

References

[1] T. Beaubouef, F.E. Petry, G. Arora, Information-theoretic measures of uncertainty for rough sets and rough relational databases, Inf. Sci. 109 (1–4) (1998) 185–195.
[2] J. Cendrowska, PRISM: an algorithm for inducing modular rules, Int. J. Man Mach. Stud. 27 (4) (1987) 349–370.
[3] D.G. Chen, The part reductions in information systems, in: Proc. RSCTC, 2004, pp. 477–482.
[4] D.G. Chen, W.L. Liu, X. Zhang, S. Kwong, Evidence-theory-based numerical algorithms of attribute reduction with neighborhood-covering rough sets, Int. J. Approx. Reason. 55 (2014) 908–923.
[5] P. Clark, S. Matwin, Using qualitative models to guide induction learning, in: Proc. ICML, 1993, pp. 49–56.
[6] J.H. Dai, W.D. Chen, H.Y. Gu, Y.H. Pan, Particle swarm algorithm for minimal attribute reduction of decision data tables, in: Proc. IMSCCS, 2006, pp. 572–575.
[7] J.H. Dai, W.T. Wang, H.W. Tian, L. Liu, Attribute selection based on a new conditional entropy for incomplete decision systems, Knowl.-Based Syst. 39 (2013) 207–213.
[8] A.K. Das, S. Chakrabarty, S.K. Pati, A.H. Sahaji, Applying restrained genetic algorithm for attribute reduction using attribute dependency and discernibility matrix, in: Proc. ICIP, 2012, pp 299–308.
[9] D.Y. Deng, H.K. Huang, A new discernibility matrix and function, in: Proc. RSKT, 2006, pp. 114–121.
[10] T.K. Deng, C.D. Yang, Y.T. Zhang, X.X. Wang, An improved ant colony optimization applied to attributes reduction, Fuzzy Inf. Eng. 1 (2009) 1–6.
[11] A.R. Hedar, J. Wang, M. Fukushima, Tabu search for attribute reduction in rough set theory, Soft Comput. 12 (9) (2008) 909–918.
[12] Q.H. Hu, D.R. Yu, Z.X. Xie, Information-preserving hybrid data reduction based on fuzzy-rough techniques, Pattern Recogn. Lett. 27 (5) (2006) 414–423.
[13] Q.H. Hu, H, Zhao, Z.X. Xie, D.R. Yu, Consistency based attribute reduction, in: Proc. PAKDD, 2007, pp. 96–107.
[14] X.Y. Jia, W.H. Liao, Z.M. Tang, L. Shang, Minimum cost attribute reduction in decision-theoretic rough set models, Inf. Sci. 219 (2013) 151–167.
[15] X.Y. Jia, Z.M. Tang, W.H. Liao, L. Shang, On an optimization representation of decision-theoretic rough set model, Int. J. Approx. Reason. 55 (1) (2014) 156–166.
[16] S.Y. Jiang, Y.S. Lu, Two new reduction definitions of decision table, Mini-Micro Syst. 27 (3) (2006) 512–515 (in Chinese).
[17] R. Jensen, S. Qiang, Semantics-preserving dimensionality reduction: rough and fuzzy-rough-based approaches, IEEE Trans. Knowl. Data Eng. 16 (12) (2004) 1457–1471.
[18] L.J. Ke, Z.R. Feng, Z.G. Ren, An efficient ant colony optimization approach to attribute reduction in rough set theory, Pattern Recogn. Lett. 29 (9) (2008) 1351–1357.
[19] M. Kryszkiewicz, Comparative study of alternative types of knowledge reduction in inconsistent systems, Int. J. Intell. Syst. 16 (1) (2001) 105–120.
[20] H.X. Li, X.Z. Zhou, J.B. Zhao, D. Liu, Attribute reduction in decision-theoretic rough set model: a further investigation, in: Proc. RSKT, 2011, pp. 466–475.
[21] J.Y. Liang, J.R. Mi, W. Wei, F. Wang, An accelerator for attribute reduction based on perspective of objects and attributes, Knowl.-Based Syst. 44 (2013) 90–100.
[22] B.X. Liu, F. Liu, X. Cheng, An adaptive genetic algorithm based on rough set attribute reduction, in: Proc. BMEI, 2010, pp. 2880–2883.
[23] J.N.K. Liu, Y. Hu, Y. He, A set covering based approach to find the reduct of variable precision rough set, Inf. Sci. 275 (2014) 83–100.
[24] J.S. Mi, W.Z. Wu, W.X. Zhang, Approaches to knowledge reduction based on variable precision rough set model, Inf. Sci. 159 (3-4) (2004) 255–272.
[25] D.Q. Miao, G.R. Hu, A heuristic algorithm for reduction of knowledge, J. Comput. Res. Develop. 36 (6) (1999) 681–684 (in Chinese).
[26] D.Q. Miao, Y. Zhao, Y.Y. Yao, H.X. Li, F.F. Xu, Relative reducts in consistent and inconsistent decision tables of the Pawlak rough set model, Inf. Sci. 179 (24) (2009) 4140–4150.
[27] S.H. Nguyen, H.S. Nguyen, Some efficient algorithms for rough set methods, in: Proc. IPMU, 1996, pp. 1451–1456.
[28] Z. Pawlak, Rough sets, Int. J. Parallel Prog. 11 (5) (1982) 341–356.
[29] Z. Pawlak, Rough classification, Int. J. Man Mach. Stud. 20 (5) (1984) 469–483.
[30] Z. Pawlak, Rough Sets. Theoretical Aspects of Reasoning About Data, Kluwer Academic Publishers, Dordrecht, Boston, London, 1991.
[31] J. Qian, P. Lv, X.D. Yue, C.H. Liu, Z.J. Jing, Hierarchical attribute reduction algorithms for big data using MapReduce, Knowl.-Based Syst. 73 (2015) 18–31.
[32] Y.H. Qian, C. Li, J.Y. Liang, An efficient fuzzy-rough attribute reduction approach, in: Proc. RSKT, 2011, pp. 63–70.
[33] M. Quafafou, α-RST: a generalization of rough set theory, Inf. Sci. 124 (1–4) (2000) 301–316.
[34] J.R. Quinlan, Learning efficiency classification procedure and their chess endgames, in: Machine Learning: An Artificial Intelligence Approach, Tioga, Palo Alto, 1983, pp. 463–482.
[35] S. Salehi, A. Selamat, H. Fujita, Systematic mapping study on granular computing, Knowl.-Based Syst. (2015), doi:10.1016/j.knosys.2015.02.018.
[36] M.W. Shao, Y. Leung, Relations between granular reduct and dominance reduct in formal contexts, Knowl.-Based Syst. 65 (2014) 1–11.
[37] H. Shi, J.Z. Fu, A heuristic genetic algorithm of attribute reduction, in: Proc. ICMLC, 2006, pp. 2263–2267.
[38] A. Skowron, J. Stepaniuk, Towards an approximation theory of discrete problems: Part I, Fundam. Inform. 15 (2) (1991) 187–208.
[39] A. Skowron, C. Rauszer, The discernibility matrices and functions in information systems, Intelligent Decision Support Handbook of Applications and Advances of the Rough Sets Theory, Kluwer Academic Publishers, Dordrecht, 1992, pp. 331–362.
[40] D. Śle¸zak, Approximate reducts in decision tables, in: Proc. IPMU, 1996, pp. 1159–1164.
[41] D. Śle¸zak, Searching for dynamic reducts in inconsistent decision tables, in: Proc. IPMU, 1998, pp. 1362–1369.
[42] D. Śle¸zak, Normalized decision functions and measures for inconsistent decision tables analysis, Fundam. Inform. 44 (3) (2000) 291–319.
[43] F. Wang, J.Y. Liang, Y.H. Qian, Attribute reduction: a dimension incremental strategy, Knowl.-Based. Syst. 39 (2013) 95–108.
[44] J. Wang, J. Wang, Reduction algorithms based on discernibility matrix: the ordered attributes method, J. Comput. Sci. Technol. 16 (6) (2001) 489–504.
[45] G.Y. Wang, J. Zhao, J. Wu, A comparative study of algebra viewpoint and information viewpoint in attribute reduction, Foundam. Inform. 68 (3) (2005) 1–13.
[46] C.Z. Wang, C.X. Wu, D.G. Chen, A systematic study on attribute reduction with rough sets based on general binary relations, Inf. Sci. 178 (9) (2008) 2237–2261.
[47] W.Z. Wu, Attribute reduction based on evidence theory in incomplete decision systems, Inf. Sci. 178 (5) (2008) 1355–1371.
[48] L.Z. Xu, Z. Han, Y.N. Zhang, A Taxation attribute reduction based on genetic algorithm and rough set theory, in: Proc. ICSP, 2008, pp. 2884–2887.
[49] J.C. Xu, L. Sun, Research of knowledge reduction based on new conditional entropy, in: Proc. RSKT, 2009, pp. 144–151.
[50] Z.Y. Xu, D.Y. Gu, B. Yang, Attribute reduction algorithm based on genetic algorithm, in: Proc. ICICTA, 2009, pp. 169–172.

[51] Z.Y. Xu, B.R. Yang, W.B. Qian, W.H. Shu, An efficient algorithm for Pawlak reduction based on simplified discernibility matrix, Fuzzy Inf. Eng. 2 (4) (2010) 433–443.
[52] Y.Y. Yang, D.G. Chen, Z. Dong, Novel algorithms of attribute reduction with variable precision rough set model, Neurocomputing 139 (2014) 336–344.
[53] Y.Y. Yao, N. Zhong, An analysis of quantitative measures associated with rules, in: Proc. PAKDD, 1999, pp. 479–488.
[54] Y.Y. Yao, Y. Zhao, J. Wang, On reduct construction algorithms, in: Proc. RSKT, 2006, pp. 297–304.
[55] Y.Y. Yao, B. Zhou, Y.H. Chen, Interpreting low and high order rules: a granular computing approach, in: Proc. RSEISP, 2007, pp. 371–380.
[56] Y.Y. Yao, Y. Zhao, Attribute reduction in decision-theoretic rough set models, Inf. Sci. 178 (17) (2008) 3356–3373.
[57] Y.Y. Yao, B. Zhou, Micro and macro evaluation of classification rules, in: Proc. IEEE ICCI, 2008, pp. 441–448.
[58] Y.Y. Yao, Interpreting concept learning in cognitive informatics and granular computing, IEEE Trans. Syst. Man. Cybern. B 39 (4) (2009) 855–866.
[59] Y.Y. Yao, Y. Zhao, Discernibility matrix simplification for constructing attribute reducts, Inf. Sci. 179 (7) (2009) 867–882.
[60] D.Y. Ye, Z.J. Chen, J.K. Liao, A new algorithm for minimum attribute reduction based on binary particle swarm optimization with vaccination, in: Proc. PAKDD, 2007, pp. 1029–1036.
[61] H. Yu, G.Y. Wang, F.K. Lan, Solving the attribute reduction problem with ant colony optimization, in: Proc. RSCTC, 2008, pp. 242–251.
[62] J.B. Zhang, T.R. Li, D. Ruan, Z.Z. Gao, C.B. Zhao, A parallel method for computing rough set approximations, Inf. Sci. 194 (2012) 209–223.
[63] W.X. Zhang, J.S. Mi, W.Z. Wu, Approaches to knowledge reductions in inconsistent systems, Int. J. Intell. Syst. 18 (9) (2003) 989–1000.
[64] W.X. Zhang, L. Wei, J.J. Qi, Attribute reduction theory and approach to concept lattice, Sci. China Inf. Sci. 48 (6) (2005) 713–726.
[65] X.H. Zhang, B. Zhou, P. Li, A general frame for intuitionistic fuzzy rough sets, Inf. Sci. 216 (2012) 34–49.
[66] Y. Zhao, S.K.M. Wong, Y.Y. Yao, A note on attribute reduction in the decision-theoretic rough set model, in: Proc. RSCTC, 2008, pp. 61–70.
[67] J. Zhou, D.Q. Miao, Q.R. Feng, L.J. Sun, Research on complete algorithms for minimal attribute reduction, in: Proc. RSKT, 2009, pp. 152–159.
[68] UCI Machine Learning Repository. <http://archive.ics.uci.edu/ml/>.