Information Sciences 418–419 (2017) 601–618

Contents lists available at ScienceDirect

Information Sciences

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

Class-specific attribute reducts in rough set theory

Yiyu Yao[a,*], Xianyong Zhang[a,b]

[a] Department of Computer Science, University of Regina, Regina, Saskatchewan S4S 0A2, Canada
[b] College of Mathematics and Software Science, Sichuan Normal University, Chengdu 610068, China

ARTICLE INFO

Article history:
Received 7 February 2016
Revised 13 June 2017
Accepted 9 August 2017
Available online 9 August 2017

Keywords:
Classification-based attribute reduct
Class-specific attribute reduct
Rough set
Three-way classification
Three-way decision
Feature selection

ABSTRACT

The concept of attribute reducts plays a fundamental role in rough set analysis. There are at least two possibilities to define an attribute reduct. A classification-based or global attribute reduct is a minimal subset of condition attributes that preserves the positive region of the decision classification, namely, the positive regions of all decision classes, in a decision table. A class-specific, class-dependent, or local attribute reduct is a minimal subset of condition attributes that preserves the positive region of a particular decision class. While a classification-based reduct may not work equally well for every decision class, a class-specific attribute reduct is optimally tailored to a particular decision class. However, studies in rough set theory are dominated by classification-based reducts; there is very limited investigation on class-specific reducts. An objective of this paper is to draw attention to class-specific reducts. We systematically compare the two types of reducts and investigate their relationships with respect to both individual reducts and families of all reducts. It is possible to derive a class-specific reduct from a classification-based reduct and to derive a classification-based reduct from a family of class-specific reducts. The families of all class-specific reducts provide a pair of lower and upper bounds of the family of all classification-based reducts. Based on a three-way classification of attributes into the pair-wise disjoint sets of core, marginal, and nonuseful attributes, we examine relationships between the corresponding classes of classification-based and class-specific attributes. The union of the sets of class-specific core attributes is the set of classification-based core attributes. It is only possible to obtain an upper bound for the set of classification-based marginal attributes and a lower bound for the set of classification-based nonuseful attributes from the family of the class-specific corresponding sets of attributes.

1. Introduction

Rough set theory concerns analyzing and reasoning about data in a data table, in which rows are objects, columns are attributes, and each cell is the value of an object on an attribute [23,24]. A decision table is a special data table such that the set of attributes is the union of a set of condition attributes and a set of decision attributes. The notions of attribute reducts play a fundamental role in rough set analysis.

Pawlak [24] defined an attribute reduct of a decision table as a minimal subset of condition attributes that has the same classification ability as the entire set of condition attributes with respect to the set of decision attributes. Following

* Corresponding author.
E-mail addresses: yyao@cs.uregina.ca, yiyuyao@gmail.com (Y. Yao), xianyongzh@sina.com.cn (X. Zhang).

http://dx.doi.org/10.1016/j.ins.2017.08.038
0020-0255/

Pawlak, subsequent studies have been concentrated on such a classification-based definition of attribute reducts, for example, see references [8,9,12,16,20,27,28,36,37,40,44–46,48]. A decision classification consists of two or more decision classes. A classification-based attribute reduct must fit all individual decision classes, which typically requires a compromise between different decision classes. In other words, a classification-based attribute reduct is an “average” for all decision classes and may not necessarily be suitable or optimal for every decision class. The compromise made by a classification-based attribute reduct may not be desirable for many real-world applications. For example, in medical diagnosis, a set of condition attributes may consist of tests or symptoms. A doctor usually suggests different sets of tests, or looks for different symptoms, for different types of disease. It seems unreasonable to recommend the same set of tests, independent of a particular disease. In teaching, a set of condition attributes may be a set of concepts or notions. For different classes of students, an instructor may focus on different sets of concepts or notions. It may not be effective to consider the same set of concepts for different classes of students. This recognition of class-specific or class-dependent attributes offers a new point of view for studying attribute reducts.

The ideas of attribute reduction are closely related to feature selection in many other fields. In contrast to traditional methods that select a single feature subset (i.e., the counterpart of an attribute reduct in rough set theory) for all the classes, class-specific methods choose possibly different feature subsets for different classes. Class-specific feature selection has received much attention in recent years [1,25,38,47]. On the other hand, this is not the case in rough set theory. For example, a survey of a sample of the most recently published papers in Information Sciences journal [6,10,13–15,17–19,26,34,35,49] shows that, although there are extensive studies on the problem of attribute reduction, research on classification-based attribute reducts still dominates. The discrepancy between the dominated existing studies on classification-based reducts and a practical need for class-specific reducts calls for attention.

Chen and associates [3–5,30] introduced the notion of a local attribute reduct. Instead of considering all decision classes, they defined a local attribute reduct with respect to a particular subset of decision classes. Two special cases of local reducts need a careful study. In one special case, when the entire set of decision classes is used, a local attribute reduct becomes a global or classification-based reduct. In this paper, we consider another special case by considering only a single decision class. To explicitly show and emphasize the role of a single decision class, we introduce the notion of a class-specific attribute reduct.[1] While a classification-based reduct is defined with respect to all decision classes, a class-specific reduct is defined with respect to one particular class. The underlying ideas and philosophy of class-specific reducts are related to a widely used strategy of “one-versus-the-rest” that transforms a k-class classification problem into a family of two-class classification problem [2,7]. In particular, for a class D_i, one can build a classifier that separates objects in D_i and not in D_i. This may result in different classifiers for different classes. Similarly, class-specific reducts allows us to find possibly different attribute reducts for different decision classes. In order to gain more insights into class-specific attribute reducts, we investigate relationships between classification-based attribute reducts and class-specific attribute reducts.

To achieve the main goal of promoting further research on class-specific attribute reducts without possible distractions, we limit our scopes in two aspects. According to a recent study by Yao [43], there are two sides of the theory of rough sets. On one side, conceptual formulations focus on the meaning of various notions and, on the other side, computational formulations focus on methods to actually construct these notions. In this paper, we restrict our discussions to conceptual formulations. The algorithms given in the paper are intended for a conceptual understanding of various notions, rather than efficient methods for carrying out actual computations. Nevertheless, once we have a good grasp of conceptual formulations, it is relatively easy to build computational formulations. In general, we can define an attribute reduct as a subset of attributes that satisfies some criteria [43]. One can, in fact, define different categories of attribute reducts based on different criteria. For example, Susmaga [31–33] introduced the notions of intra-class reducts, inter-class reducts, and constructs. Ślęzak [28] introduced the notion of approximate reducts. Stawicki et al. [29] compared decision bireducts and decision reducts. These categories of reducts are related to indiscernibility reducts, discernibility reducts, and indiscernibility-and-discernibility reducts [34,50]. In this paper, we consider only the category of attribute reducts defined by the criterion of preserving the positive region as given by the entire set of attributes. It is important to realize that the same argument can be applied to study other categories of reducts. In other words, one can study the class-specific versions of other categories of reducts based on results given in this paper.

The rest of the paper is organized as follows. Section 2 gives a critical analysis of classification-based attribute reducts. Section 3 proposes class-specific attribute reducts. Section 4 analyzes relationships between classification-based and class-specific attribute reducts. Section 5 makes an analysis of attributes based on a three-way classification to probe connections between both types of attribute reducts. Finally, Section 6 concludes this paper. For readability of the paper, proofs of lemmas and theorems that do not easily follow from either definitions or discussions are given in Appendix A.

[1] There are two definitions of a local attribute reduct suggested by Chen and associates. One is defined with respect to the family of definable sets [3,4] and the other is defined with respect to the family of decision classes [5]. Since not every decision class is a definable set, one needs to change a decision class into a definable set by using its lower approximation or upper approximation. The two definitions are not equivalent in all aspects. Our definition of a class-specific reduct is defined with respect to a particular decision class and is related to the version of a local reduct with respect to the family of decision classes. In a paper published after the submission of our paper, Song et al. [30] called a class-specific reduct a local reduct. We choose to use the term “class-specific reduct” in order to explicitly show the physical meaning of such a reduct, which is also in line with the uses of “class-specific feature” or “label-specific feature” in machine learning [1,47]. Our notion of class-specific reducts, in fact, corresponds to the notion of a label-specific feature reduct used by Xu et al. [41], if we interpret the set of objects obtained from a particular label as a decision class.

2. A critical analysis of classification-based attribute reducts

In this section, we motivate the present study by pointing out blind spots of classification-based attribute reducts. To resolve these difficulties, we introduce the notion of a class-specific attribute reduct in Section 3.

2.1. Classification-based attribute reducts

Rough set analysis (RSA) focuses on data represented in a data table:

$$T = (OB, AT, \{V_a \mid a \in AT\}, \{I_a \mid a \in AT\}), \tag{1}$$

where OB is a finite nonempty set of objects called the universe, AT is a finite nonempty set of attributes, V_a is the domain of values for $a \in AT$, and $I_a : OB \rightarrow V_a$ is an information function. Each object x takes exactly one value $I_a(x)$ on attribute a. Given a subset of attributes $A \subseteq AT$, we define an equivalence relation by:

$$E_A = \{(x, y) \in OB \times OB \mid \forall a \in A (I_a(x) = I_a(y))\}. \tag{2}$$

This equivalence relation induces a partition of OB, denoted by $\pi_A = \{[x]_A \mid x \in OB\}$, where $[x]_A = \{y \mid yE_Ax\}$ is the equivalence class containing x. A partition of OB is also called a classification [23,24].

A decision table is a special type of data tables with $AT = C \cup D$ and $C \cap D = \emptyset$, where C and D are the sets of condition and decision attributes, respectively. For the set of condition attributes C, we have a partition π_C. For the set of decision attributes D, we have a partition $\pi_D = \{[x]_D \mid x \in OB\}$, which is called a decision classification of OB. In terms of set-inclusion relation $\subseteq$, we say that a decision table is consistent if $E_C \subseteq E_D$ and is inconsistent otherwise.

Given a set of attributes $A \subseteq C$ in a decision table, the family of equivalence classes in π_A serves as the building blocks to construct approximations of a subset of objects. According to Pawlak [24], one can approximate a subset $X \subseteq OB$ by either a pair of lower and upper approximations or three pair-wise disjoint positive, negative and boundary regions. In this paper, we adopt the three-region based definition [43] and the notation system introduced by Huang and Yao [11].

Definition 1. For a subset of objects $X \subseteq OB$, the positive, negative, and boundary regions of X given π_A are constructed from the equivalence classes in π_A as follows:

$$\begin{aligned}
\mathrm{POS}(X|\pi_A) &= \{x \mid [x]_A \subseteq X\},\\
\mathrm{NEG}(X|\pi_A) &= \{x \mid [x]_A \subseteq X^c\},\\
\mathrm{BND}(X|\pi_A) &= OB - \mathrm{POS}(X|\pi_A) \cup \mathrm{NEG}(X|\pi_A),
\end{aligned} \tag{3}$$

where $X^c = OB - X$ denotes the complement of X.

A decision classification π_D in a decision table usually has two or more classes and we denote it by $\pi_D = \{D_i \mid i = 1, \ldots, m\}, m \geq 2$. For a decision class D_i, its positive region is given by $\mathrm{POS}(D_i|\pi_A)$. By taking the union of the positive regions of all decision classes, Pawlak [24] introduces the positive region of a decision classification π_D.

Definition 2. The positive region of classification π_D given π_A is defined by:

$$\mathrm{POS}(\pi_D|\pi_A) = \bigcup_{i=1}^{m} \mathrm{POS}(D_i|\pi_A). \tag{4}$$

By definition, different subsets of attributes induce different positive regions of the decision classification π_D. It can be easily verified that positive regions are monotonic with respective to the set inclusion of attributes. That is, for $A, B \subseteq AT$, $X \subseteq OB$ and a decision classification π_D, we have:

$$\begin{aligned}
A \subseteq B &\Longrightarrow \mathrm{POS}(X|\pi_A) \subseteq \mathrm{POS}(X|\pi_B),\\
A \subseteq B &\Longrightarrow \mathrm{POS}(\pi_D|\pi_A) \subseteq \mathrm{POS}(\pi_D|\pi_B).
\end{aligned} \tag{5}$$

A subset of attributes normally produces a smaller positive region than its supersets. Let $\mathrm{POS}(\pi_D|\pi_C)$ denote the positive region of the decision classification given the partition π_C induced by the entire set of condition attributes. The notion of a classification-based attribute reduct is a minimal set of C that produces the same positive region of the decision classification [23,24].

Definition 3. A subset of condition attributes $R \subseteq C$ is called a classification-based (relative) attribute reduct of C if it satisfies the following conditions:

$$\begin{aligned}
&\text{(S)}\quad \mathrm{POS}(\pi_D|\pi_R) = \mathrm{POS}(\pi_D|\pi_C),\\
&\text{(N)}\quad \forall a \in R(\mathrm{POS}(\pi_D|\pi_{R-\{a\}}) \neq \mathrm{POS}(\pi_D|\pi_R)).
\end{aligned}$$

We denote the set of all classification-based attribute reducts of C by $\mathrm{RED}(\pi_D)$.

For an inconsistent table, $\mathrm{POS}(\pi_D|\pi_C) \subset OB$ holds and, for a consistent table, $\mathrm{POS}(\pi_D|\pi_C) = OB$. The definition of a classification-based reduct is applicable to both consistent and inconsistent tables. Condition (S) may be viewed as a joint

Table 1
Decision table I.

OB	c_1	c_2	c_3	d
o_1	0	1	0	1
o_2	0	1	0	1
o_3	0	1	1	1
o_4	0	0	1	2
o_5	0	0	1	2
o_6	1	0	1	2
o_7	1	0	0	3
o_8	1	0	0	3
o_9	1	1	0	3

Table 2
Decision table II.

OB	c_1	c_2	c_3	c_4	c_5	d
o_1	0	1	1	1	0	1
o_2	1	0	0	0	1	1
o_3	2	2	0	1	0	1
o_4	1	1	1	1	0	2
o_5	0	0	0	0	1	2
o_6	2	2	2	0	1	2
o_7	0	0	1	1	0	3
o_8	1	1	0	1	0	3
o_9	2	2	2	2	2	3

sufficiency condition in the sense that all attributes in R are jointly sufficient to preserve the positive region of π_D. Condition (N) may be viewed as an individual necessity condition in the sense that each attribute in R is necessary for preserving the positive region. The removal of any attribute in R would result in a smaller positive region of π_D, reflecting the minimality of a classification-based attribute reduct.

2.2. *Blind spots of classification-based attribute reducts*

According to Definition 2, the positive region $\mathrm{POS}(\pi_D|\pi_R)$ is the union of positive regions of all decision classes $\mathrm{POS}(D_i|\pi_R)$, $1 \le i \le m$. We may consider $\mathrm{POS}(\pi_D|\pi_C)$ to be a global classification-based approximation and $\mathrm{POS}(D_i|\pi_C)$, $1 \le i \le m$, to be class-specific local approximations. By definition, a classification-based attribute reduct is a minimal subset of condition attributes that preserves the positive region of π_D. A classification-based attribute reduct considers all decision classes collectively at the same time. An interesting question naturally arises: What would happen if one considers decision classes individually? It is surprising that this question has not been, at least not explicitly, asked in studies in rough set theory. An answer to this question will reveal blind spots of classification-based reducts. We provide a detailed analysis by considering a few simple examples.

Table 1 is a consistent decision table with $OB = \{o_1, \ldots, o_9\}$, $C = \{c_1, c_2, c_3\}$, and $D = \{d\}$. The decision classification π_D consists of three decision classes:

$$D_1 = \{o_1, o_2, o_3\}, \quad D_2 = \{o_4, o_5, o_6\}, \quad D_3 = \{o_7, o_8, o_9\}.$$

This decision table has only one reduct $R = C = \{c_1, c_2, c_3\}$. One cannot remove any attribute in order to preserve the positive region $\mathrm{POS}(\pi_D|\pi_C)$. If we consider decision classes individually, we have:

$$\begin{aligned}
\mathrm{POS}(D_1|\pi_C) &= \mathrm{POS}(D_1|\pi_{\{c_1,c_2\}}),\\
\mathrm{POS}(D_2|\pi_C) &= \mathrm{POS}(D_2|\pi_{\{c_2,c_3\}}),\\
\mathrm{POS}(D_3|\pi_C) &= \mathrm{POS}(D_3|\pi_{\{c_1,c_3\}}).
\end{aligned}$$

They suggest that we can further remove attributes when preserving the positive region of a particular decision class. For instance, in order to preserve the positive region $\mathrm{POS}(D_1|\pi_C)$, we only need to use attributes c_1 and c_2, rather than the entire set of condition attributes.

Although the decision table has one classification-based attribute reduct, it shows the first blind spot of a classification-based reduct. An attribute in a classification-based reduct is necessary for preserving the positive regions of some decision classes, but may not be necessary for every decision class. For instance, attribute c_1 is necessary for D_1 and D_3, but not necessary for D_2; attribute c_2 is necessary for D_1 and D_2, but not necessary for D_3. A classification-based attribute reduct, although being a minimal set of attributes when all decision classes are considered collectively, may not be a minimal set when a particular decision class is considered.

Table 3
Decision table III.

OB	c_1	c_2	c_3	c_4	c_5	d
o_1	0	0	3	0	0	1
o_2	0	0	0	1	1	1
o_3	0	3	3	2	2	1
o_4	1	1	1	0	1	2
o_5	1	1	1	1	2	2
o_6	1	2	2	2	0	2
o_7	1	2	1	0	2	3
o_8	1	2	1	1	0	3
o_9	1	1	2	2	1	3

Table 2 is a consistent decision table with $OB = \{o_1, \ldots, o_9\}$, $C = \{c_1, c_2, c_3, c_4, c_5\}$, and $D = \{d\}$. The decision classification π_D consists of three decision classes:

$$D_1 = \{o_1, o_2, o_3\}, \quad D_2 = \{o_4, o_5, o_6\}, \quad D_3 = \{o_7, o_8, o_9\}.$$

There are two classification-based reducts $\{c_1, c_2, c_3, c_4\}$ and $\{c_1, c_2, c_3, c_5\}$. To preserve the positive region of decision class D_1, we can use any one of the sets $\{c_1, c_2, c_3\}$, $\{c_1, c_2, c_4\}$, and $\{c_1, c_2, c_5\}$. To preserve the positive region of D_2, we can use either $\{c_1, c_3, c_4\}$ or $\{c_1, c_3, c_5\}$. Finally, to preserve the position region of D_3, we can use $\{c_2, c_3, c_4\}$ or $\{c_2, c_3, c_5\}$.

With two reducts, this example shows another blind spot of a classification-based reduct. With respect to a classification-based reduct, one uses the same reduct for all decision classes [24] or chooses different subsets of the same reduct for different decision subclasses as defined by classification/decision rules [44]. For example, for a classification-based reduct $\{c_1, c_2, c_3, c_4\}$, one can use either $\{c_1, c_2, c_3\}$ or $\{c_1, c_2, c_4\}$ for class D_1, use $\{c_1, c_3, c_4\}$ for D_2, and use $\{c_2, c_3, c_4\}$ for D_3. It is difficult to use $\{c_1, c_2, c_5\}$ for D_1, use $\{c_1, c_3, c_4\}$ for D_2, and use $\{c_2, c_3, c_4\}$ for D_3, since they are derived from different classification-based reducts. The first set is derived from classification-based reduct $\{c_1, c_2, c_3, c_5\}$ and the last two sets are derived from classification-based reduct $\{c_1, c_2, c_3, c_4\}$.

In practical applications, such as the selection of tests in medical diagnosis and the selection of concepts or notions in teaching mentioned in the introduction, it may be desirable to use different sets of attributes for different decision classes. Classification-based attribute reducts fail to consider this possibility.

Table 3 is a consistent decision table with $OB = \{o_1, \ldots, o_9\}$, $C = \{c_1, c_2, c_3, c_4, c_5\}$, and $D = \{d\}$. The decision classification π_D contains three decision classes:

$$D_1 = \{o_1, o_2, o_3\}, \quad D_2 = \{o_4, o_5, o_6\}, \quad D_3 = \{o_7, o_8, o_9\}.$$

There are three classification-based reducts, $\{c_2, c_3\}$, $\{c_2, c_4\}$, and $\{c_4, c_5\}$. It can be verified that the singleton set of attribute $\{c_1\}$ preserves the positive region of D_1. Attribute c_1 is not in any of the three classification-based reducts. It is therefore impossible to derive the set $\{c_1\}$ for class D_1 from classification-based reducts. This example reveals a third blind spot of classification-based reducts. That is, classification-based reducts may fail to include attributes that are useful for particular decision classes.

The blind spots of classification-based reducts, as demonstrated by the three simple examples, call for an in-depth investigation of the problem of attribute reduction when decision classes are considered independently. This motivates us to propose the notion of a class-specific attribute reduct.

3. Class-specific attribute reducts

When defining a classification-based attribute reduct, one considers all decision classes collectively. As a result, a classification-based attribute reduct, although a minimal set for the entire decision classification, may not be minimal for each decision class. In other words, such a global reduct is not necessarily a local reduct for all decision classes. By looking at a specific decision class independent of other classes, we can introduce the notion of class-specific attribute reducts.

Consider a decision class $D_i \in \pi_D$. The positive region of D_i given π_C is denoted by $\mathrm{POS}(D_i|\pi_C)$. By following a similar interpretation of a classification-based reduct, one may view a class-specific attribute reduct to be a minimal subset of C that produces the same positive region. By using the monotonicity given by Eq. (5), we give the following formal definition of a class-specific attribute reduct.

Definition 4. A subset of condition attributes $R \subseteq C$ is called a class-specific attribute reduct of C with respect to D_i if it satisfies the two conditions:

(s) $\mathrm{POS}(D_i|\pi_R) = \mathrm{POS}(D_i|\pi_C)$,

(n) $\forall a \in R(\mathrm{POS}(D_i|\pi_{R-\{a\}}) \neq \mathrm{POS}(D_i|\pi_R))$.

The set of all class-specific attribute reducts of C with respect to D_i is denoted by $\mathrm{RED}(D_i)$.

By using the positive region of a decision class, the definition of a class-specific reduct is applicable to both consistent and inconsistent decision tables. Condition (s) is a joint sufficiency condition; all attributes in R jointly preserve the positive region of D_i. Condition (n) is an individual necessity condition; every attribute in R is necessary for preserving the positive region. In defining a class-specific reduct, we focus on a particular decision class without considering other decision classes. This enables us to derive a reduct that is most suitable to the particular decision class.

Our definition of a class-specific reduct is related to a special case of a local attribute reduct introduced by Chen and Zhao [5] and corresponds to a label-specific reduct introduced by Xu et al. [41]. There is an alternative possible way to define class-specific reducts by making use of classification-based reducts.[2] Given a decision class D_i, it produces a partition $\{D_i, OB - D_i\}$. One can define a D_i-specific reduct as a classification-based reduct with respect to the classification $\{D_i, OB - D_i\}$, instead of the original classification π_D. This new definition takes into consideration of both D_i and $OB - D_i$. For a consistent decision table, the new definition is equivalent to Definition 4. An advantage of Definition 4 is that we do not need to consider explicitly $OB - D_i$. For an inconsistent decision table, the two definitions are not equivalent. Definition 4 only considers D_i and is different from the classification-based definition using $\{D_i, OB - D_i\}$ that considers both D_i and $OB - D_i$. As future research, it is interesting to study the new definition of class-specific reducts.

The introduction of the notion of class-specific reducts avoids the blind spots of classification-based reducts. First, for each decision class, by condition (n), every attribute in a class-specific reduct is necessary. Second, by considering decision classes in isolation, we are able to use class-specific reducts that are not necessarily subsets of the same classification-based reduct. Third, although the definition of a class-specific reduct only follows the general idea in defining a classification-based reduct, it does not make use of any particular classification-based reduct. Consequently, we will not miss any attribute that is useful for a particular class but not for a classification.

Consider the decision table given by Table 3. The set of classification-based reducts and the sets of class-specific reducts for individual decision classes are given, respectively, by:

$$\begin{aligned}
\mathrm{RED}(\pi_D) &= \{\{c_2, c_3\}, \{c_2, c_4\}, \{c_4, c_5\}\};\\
\mathrm{RED}(D_1) &= \{\{c_1\}, \{c_2\}, \{c_3\}, \{c_4, c_5\}\},\\
\mathrm{RED}(D_2) &= \{\{c_2, c_3\}, \{c_2, c_4\}, \{c_4, c_5\}\},\\
\mathrm{RED}(D_3) &= \{\{c_2, c_3\}, \{c_2, c_4\}, \{c_4, c_5\}\}.
\end{aligned}$$

We can immediately make the following observations. A class-specific reduct may be derived from a classification-based reduct and may be a proper subset of the latter. For instance, D_1-specific reduct $\{c_2\}$ can be derived from either classification-based reduct $\{c_2, c_3\}$ or $\{c_2, c_4\}$ and is a proper subset of the two classification-based reducts. We can use class-specific reducts derivable from different classification-based reducts. For instance, we can use D_1-specific reduct $\{c_3\}$, D_2-specific reduct $\{c_2, c_4\}$, and D_3-specific reduct $\{c_4, c_5\}$. They are derived from three different classification-based reducts. Finally, we can have a class-specific reduct that cannot be derived from any classification-based reduct. For instance, D_1-specific reduct $\{c_1\}$ cannot be derived from any of the three classification reducts.

The example shows that classification-based reducts and class-specific reducts are both related and different. It is worthwhile to study their relationships and differences.

4. Relationships between two types of reducts

A classification-based attribute reduct is a kind of "average" for all decision classes. A class-specific attribute reduct is adapted to a particular decision class. Different decision classes may have different attribute reducts. This section investigates relationships between classification-based and class-specific attribute reducts by focusing on three fundamental issues, namely, (a) the conditions under which a reduct of one type is also the other type, (b) the derivation of a reduct of one type from the other type, and (c) the connections between the set of classification-based reducts and the family of sets of class-specific reducts.

4.1. A comparative analysis

To study the connections between classification-based and class-specific reducts, we first look at the relationships between conditions for defining the two types of reducts in Definitions 3 and 4. By Definition 2, the positive region of a classification $\pi_D = \{D_1, \ldots, D_m\}$, $\mathrm{POS}(\pi_D|\pi_A)$, $A \subseteq C$, is the union of the positive regions of individual classes, that is,

$$\mathrm{POS}(\pi_D|\pi_A) = \bigcup_{i=1}^{m} \mathrm{POS}(D_i|\pi_A). \tag{6}$$

Accordingly, we can re-express the two conditions of classification-based reducts in terms of the positive regions of the individual decision classes.

[2] The authors are grateful to a reviewer of the paper for suggesting this definition of a class-specific attribute reduct.

Lemma 1. *The following equivalences hold:*

(i) $\mathrm{POS}(\pi_D|\pi_R) = \mathrm{POS}(\pi_D|\pi_C) \iff \forall 1 \le i \le m(\mathrm{POS}(D_i|\pi_R) = \mathrm{POS}(D_i|\pi_C))$,

(ii) $\forall a \in R(\mathrm{POS}(\pi_D|\pi_{R-\{a\}}) \ne \mathrm{POS}(\pi_D|\pi_R)) \iff$
$\forall a \in R, \exists 1 \le i \le m(\mathrm{POS}(D_i|\pi_{R-\{a\}}) \ne \mathrm{POS}(D_i|\pi_R))$.

Consider a classification-based reduct R. According to Lemma 1(i), the condition (S) of Definition 3 and the condition (s) of Definition 4, R is sufficient to preserve the positive region for all decision classes. On the other hand, Lemma 1(ii) suggests that any attribute a in a classification-based reduct is required by at least one decision class. Different attributes in R may be required by different decision classes. To preserve the positive region of a particular class, we may not need all attributes in R. Therefore, according to condition (n) of Definition 4, a classification-based reduct is not necessarily a class-specific reduct with respect to a decision class. Consider a class-specific reduct R' with respect to D_i. By condition (s) of Definition 4, R' preserves the positive region of D_i. There is no guarantee that R' would also preserve the positive region of other decision classes and may not be a classification-based reduct. In summary, we can conclude that in general the two types of reducts are different.

A classification-based reduct R preserves the positive regions of all decision classes. For R to be a class-specific reduct with respect to a decision class D_i, R only needs to satisfy the condition (n) of Definition 4. Recall that none of the attribute in a class-specific reduct R' of decision class D_i can be removed. If R' preserves the positive region of the classification π_D, it is also a classification-based reduct. We immediately have the following theorem.

Theorem 1. *Suppose $R \in \mathrm{RED}(\pi_D)$ and $R' \in \mathrm{RED}(D_i)$. The following conditions hold:*

(i) $R \in \mathrm{RED}(D_i) \iff \forall a \in R(\mathrm{POS}(D_i|\pi_{R-\{a\}}) \ne \mathrm{POS}(D_i|\pi_R))$,

(ii) $R' \in \mathrm{RED}(\pi_D) \iff \mathrm{POS}(\pi_D|\pi_{R'}) = \mathrm{POS}(\pi_D|\pi_C)$.

The theorem only shows the conditions under which a reduct of one type becomes a reduct of the other type. Condition $R \in \mathrm{RED}(\pi_D)$ implies the sufficiency condition (s) in Definition 4. Condition $R' \in \mathrm{RED}(D_i)$ implies the necessity condition (N) in Definition 3. We cannot define one type of reducts in terms of the other type.

By definition, a classification-based reduct may contain attributes that are not necessarily needed for a specific decision class. The next theorem shows that a classification-based reduct is a superset of at least one class-specific reduct with respect to any decision class.

Theorem 2. *Suppose $R \in \mathrm{RED}(\pi_D)$. For any $D_i \in \pi_D$, there exists a class-specific attribute reduct $R' \in \mathrm{RED}(D_i)$ such that $R' \subseteq R$.*

According to Theorem 2, a classification-based reduct must contain at least one class-specific reduct for any decision class. In order to derive a class-specific reduct R_i for D_i from a classification-based reduct R, one can sequentially test and delete, if necessary, unnecessary attributes based on condition (n) of Definition 4. This gives a deletion strategy based reduct construction algorithm [46], as shown by Algorithm CBR2CSR.

Algorithm CBR2CSR Construction of a family of class-specific reducts from a classification-based reduct

Input: A decision table T and a classification-based reduct $R \in \mathrm{RED}(\pi_D)$;
Output: A family of class-specific reducts $(R_1, \ldots, R_m)$, with $R_i \subseteq R$ and $R_i \in \mathrm{RED}(D_i)$.

1: **for** $i = 1$ to m **do**
2:     $R_i = R$;
3:     **for** each $a \in R$ **do**
4:         **if** $\mathrm{POS}(D_i|\pi_{R_i-\{a\}}) = \mathrm{POS}(D_i|\pi_{R_i})$ **then**
5:             $R_i = R_i - \{a\}$;
6:         **end if**
7:     **end for**
8: **end for**
9: **return** $(R_1, \ldots, R_m)$.

In Algorithm CBR2CSR, the outer **for** loop given by Line 1 controls the search for a classification-based reduct for each decision class. Line 2 starts with a classification-based reduct R. The inner **for** loop given by Line 3 sequentially checks if an attribute $a \in R$ is needed in preserving the positive region of D_i and deletes a if a is not needed. By checking all attributes in R sequentially, we ensure that R_i does not contain any unnecessary attributes and, hence, it is a class-specific reduct of D_i. The inner **for** loop given by Line 3 does not specify a particular sequence of attributes in R. By using different sequences of attributes, the algorithm may produce different class-specific reducts.

A question related to Theorem 2 is whether an arbitrary class-specific reduct is contained in a classification-based reduct. Unfortunately, the answer is negative. That is, an arbitrary class-specific reduct may not be a subset of any classification-based reduct. Consequently, we cannot obtain a classification-based reduct by starting with a class-specific reduct and adding more attributes to it.

A classification-based reduct preserves the positive regions of all decision classes, while a class-specific reduct preserves the positive region of a particular decision class. This suggests a possible way to study the relationship between a classification-based reduct and a family of class-specific reducts with each for one decision class. First, we construct the following Cartesian product of m sets of class-specific reducts:

$$\begin{aligned} \mathrm{CRED} &= \mathrm{RED}(D_1) \times \ldots \times \mathrm{RED}(D_m) \\ &= \{(R_1, \ldots, R_m) \mid R_i \in \mathrm{RED}(D_i), 1 \le i \le m\}. \end{aligned} \tag{7}$$

Since R_i preserves the positive region of decision class D_i, $\bigcup_{i=1}^{m} R_i$ will preserve the positive region of π_D. There is no guarantee that every attribute in $\bigcup_{i=1}^{m} R_i$ is necessary for preserving the positive regions of π_D. Thus, $\bigcup_{i=1}^{m} R_i$ must contain a classification-based reduct.

Theorem 3. *Suppose* $(R_1, \ldots, R_m) \in \mathrm{CRED}$. *There exists a classification-based reduct* $R \in \mathrm{RED}(\pi_D)$ *such that* $R \subseteq \bigcup_{i=1}^{m} R_i$.

Theorem 3 immediately suggests a deletion-based algorithm for constructing a classification-based reduct from a family of class-specific reducts, as given by Algorithm CSR2CBR.

Algorithm CSR2CBR Construction of a classification-based reduct from a family of class-specific reducts

Input: A decision table T and a family of class-specific reducts $(R_1, ..., R_m)$;
Output: A classification-based reduct $R \subseteq \bigcup_{i=1}^{m} R_i$.

$R = \bigcup_{i=1}^{m} R_i$;
for each $a \in \bigcup_{i=1}^{m} R_i$ **do**
  if $\mathrm{POS}(\pi_D|\pi_{R-\{a\}}) = \mathrm{POS}(\pi_D|\pi_R)$ **then**
    $R = R - \{a\}$;
  end if
end for
return R.

The combination of Algorithms CBR2CSR and CSR2CBR produces an interesting result. If we start with a classification-based reduct R, Algorithm CBR2CSR produces a family of class-specific reducts $(R_1, \ldots, R_m)$. By using $(R_1, \ldots, R_m)$ as input of Algorithm CSR2CBR, we will recover the original classification-based reduct R. This immediately leads to the following theorem.

Theorem 4. *For any classification-based reduct* $R \in \mathrm{RED}(\pi_D)$, *there exists a family of class-specific reducts* $(R_1, \ldots, R_m) \in \mathrm{CRED}$ *such that* $R = \bigcup_{i=1}^{m} R_i$.

In general, it is very difficult to find a family of class-specific reducts so that their union is a classification-based reduct. We must know a classification-based reduct to construct a family of class-specific reducts by using Algorithm CBR2CSR. Therefore, although Theorem 4 is very interesting, we cannot use it to help us to construct a classification-based reduct from a family of class-specific reducts.

The combination of Algorithms CBR2CSR and CSR2CBR produces an identity mapping in $\mathrm{RED}(\pi_D)$. On the other hand, the combination of Algorithms CSR2CBR and CBR2CSR does not necessarily produce an identity mapping in CRED. By starting with a family of class-specific reducts $(R_1, \ldots, R_m)$, Algorithm CSR2CBR finds a classification-based reduct R. By using R as the input of Algorithm CBR2CSR, we may not necessarily recover $(R_1, \ldots, R_m)$.

A class-specific reduct only preserves the positive region of a particular decision class and no attribute in it can be removed. With respect to the set of all decision classes in π_D, there may not exist a common class-specific reduct. If such a common class-specific reduct indeed exists, it preserves the positive regions of all decision classes and no attribute in it can be removed. By Lemma 1 and Definition 3, a common class-specific reduct must be a classification-based reduct, as stated in the next lemma.

Lemma 2. *If* R *is a common class-specific attribute reduct with respect to all decision classes, then* R *is a classification-based attribute reduct. That is, if for all decision classes* D_i $(1 \le i \le m)$, $R \in \mathrm{RED}(D_i)$, *then* $R \in \mathrm{RED}(\pi_D)$.

The set of all common class-specific reducts can be obtained by taking an intersection of the family of sets of class-specific reducts for all decision classes. Thus, Lemma 2 can be expressed equivalently by:

$$\bigcap \{\mathrm{RED}(D_i) \mid D_i \in \pi_D\} \subseteq \mathrm{RED}(\pi_D). \tag{8}$$

In this way, we have a lower bound for classification-based attribute reducts in terms of class-specific reducts. On the other hand, by the fact that a classification-based reduct may not be a class-specific reduct with respect to any decision class, we cannot obtain an upper bound of $\mathrm{RED}(\pi_D)$ in terms of the union of the families of class-specific reducts. That is, $\bigcup \{\mathrm{RED}(D_i) \mid D_i \in \pi_D\}$ is not necessarily an upper bound of the set of all classification-based reducts.

By Theorem 4, for any classification-based reduct $R \in \text{RED}(\pi_D)$, we can find a family of class-specific reducts $(R_1, \ldots, R_m) \in$ CRED such that $R = \bigcup_{i=1}^{m} R_i$. Therefore, we can derive an upper bound of $\text{RED}(\pi_D)$ through CRED as follows:

$$\text{RED}(\pi_D) \subseteq \left\{ \bigcup_{i=1}^{m} R_i \mid (R_1, \ldots, R_m) \in \text{CRED} \right\}. \tag{9}$$

By combining Eqs. (8) and (9), we arrive at the following theorem.

Theorem 5. *The set of classification-based reducts is bounded by the intersection and union of class-specific reducts as follows:*

$$\bigcap \{\text{RED}(D_i) \mid D_i \in \pi_D\} \subseteq \text{RED}(\pi_D) \subseteq \left\{ \bigcup_{i=1}^{m} R_i \mid (R_1, \ldots, R_m) \in \text{CRED} \right\}. \tag{10}$$

By Theorem 5, we establish a pair of a lower and an upper bound of the set of all classification-based reducts in terms of the family of sets of class-specific reducts with respect to all decision classes. In addition, we have a pair of algorithms for deriving one type of reducts from the other type.

4.2. An example

We illustrate notions and results discussed in this section by using the decision table given by Table 3.

Recall that the classification-based reducts and class-specific reducts are given, respectively, by:

$$\begin{aligned}
\text{RED}(\pi_D) &= \{\{c_2, c_3\}, \{c_2, c_4\}, \{c_4, c_5\}\};\\
\text{RED}(D_1) &= \{\{c_1\}, \{c_2\}, \{c_3\}, \{c_4, c_5\}\},\\
\text{RED}(D_2) &= \{\{c_2, c_3\}, \{c_2, c_4\}, \{c_4, c_5\}\},\\
\text{RED}(D_3) &= \{\{c_2, c_3\}, \{c_2, c_4\}, \{c_4, c_5\}\}.
\end{aligned}$$

It follows that $\text{RED}(D_1) \cap \text{RED}(D_2) \cap \text{RED}(D_3) = \{\{c_4, c_5\}\} \subseteq \text{RED}(\pi_D)$. That is, we have only one common class-specific reduct $\{c_4, c_5\}$ that is a classification-based reduct. Consider the classification-based reduct $\{c_2, c_3\}$. It contains two D_1-specific reducts $\{c_2\}$ and $\{c_3\}$ and is, in fact, a D_2-specific and a D_3-specific reduct. Similar observations can be made for other two classification-based reducts $\{c_2, c_4\}$ and $\{c_4, c_5\}$. By using Algorithm CBR2CSR, from the classification-based reduct $\{c_2, c_3\}$ we can derive two families of class-specific reducts $(\{c_2\}, \{c_2, c_3\}, \{c_2, c_3\})$ and $(\{c_3\}, \{c_2, c_3\}, \{c_2, c_3\})$. The D_1-specific reduct $\{c_2\}$ is obtained by Algorithm CBR2CSR under the testing sequence of (c_3, c_2) and $\{c_3\}$ under (c_2, c_3). From other classification-based reducts, we can obtain additional families of class-specific reducts. It is interesting to note that one cannot derive D_1-specific reduct $\{c_1\}$ from any classification-based reduct.

There are $4 \times 3 \times 3 = 36$ different families of class-specific reducts. The union of each family contains at least one classification-based reduct. For example, the union of the family $(\{c_1\}, \{c_2, c_3\}, \{c_2, c_4\})$ is $\{c_1, c_2, c_3, c_4\}$, which contains two classification-based reducts $\{c_2, c_3\}$ and $\{c_2, c_4\}$. Under the testing sequence (c_1, c_2, c_3, c_4) and (c_1, c_2, c_4, c_3), Algorithm CSR2CBR produces the classification-based reducts $\{c_2, c_4\}$ and $\{c_2, c_3\}$, respectively.

Consider a classification-based reduct $\{c_2, c_4\}$. Algorithm CBR2CSR may produce a family of class-specific reducts $(\{c_2\}, \{c_2, c_4\}, \{c_2, c_4\})$. By using this family as input to Algorithm CSR2CBR, we recover the classification-based reduct $\{c_2, c_4\}$. Consider now a family of class-specific reducts $(\{c_1\}, \{c_2, c_3\}, \{c_2, c_4\})$. Algorithm CSR2CBR can produce a classification-based reduct $\{c_2, c_4\}$. Algorithm CBR2CSR with classification-based reduct $\{c_2, c_4\}$ cannot produce the family of class-specific reducts $(\{c_1\}, \{c_2, c_3\}, \{c_2, c_4\})$.

For the bounds of $\text{RED}(\pi_D)$, one can easily verify the following relationships:

$$\begin{aligned}
\bigcap \{\text{RED}(D_i) \mid D_i \in \pi_D\} &= \{c_4, c_5\}\\
&\subseteq \{\{c_2, c_3\}, \{c_2, c_4\}, \{c_4, c_5\}\}\\
&= \text{RED}(\pi_D)\\
&\subseteq \{\{c_2, c_3\}, \{c_2, c_4\}, \{c_4, c_5\}, \ldots\}\\
&= \{\cup_{i=1}^{3} R_i \mid (R_1, R_2, R_3) \in \text{CRED}\},
\end{aligned}$$

where classification-based reducts $\{c_2, c_3\}$, $\{c_2, c_4\}$, and $\{c_4, c_5\}$ can be obtained from the families of class-specific reducts $(\{c_3\}, \{c_2, c_3\}, \{c_2, c_3\})$, $(\{c_2\}, \{c_2, c_4\}, \{c_2, c_4\})$, and $(\{c_4, c_5\}, \{c_4, c_5\}, \{c_4, c_5\})$, respectively.

5. An analysis of attributes based on a three-way classification

By using the family of all reducts of any one of the two types, one can divide the set of condition attributes into three classes [8,22,39,42]. This three-way classification provides another means to further study connections between the two types of reducts.

5.1. Three-way classification of attributes

Let RED denote a set of all reducts with respect to π_D or with respect to D_i, that is, $\mathrm{RED} = \mathrm{RED}(\pi_D)$ or $\mathrm{RED} = \mathrm{RED}(D_i)$. According to RED, we divide the set of condition attributes C into three classes through set intersection and union of all reducts in RED.

Definition 5. Given the set of all reducts RED, the set of condition attributes C can be divided into three pair-wise disjoint classes:

$$\begin{aligned} \mathrm{CORE} &= \bigcap \mathrm{RED}, \\ \mathrm{MARGINAL} &= \bigcup \mathrm{RED} - \bigcap \mathrm{RED}, \\ \mathrm{NONUSEFUL} &= C - \bigcup \mathrm{RED}. \end{aligned} \tag{11}$$

They are called the sets of core, marginal, and nonuseful attributes, respectively. Moreover, the sets of useful and noncore attributes are defined, respectively, as follows:

$$\begin{aligned} \mathrm{USEFUL} &= \bigcup \mathrm{RED}, \\ \mathrm{NONCORE} &= C - \bigcap \mathrm{RED}. \end{aligned} \tag{12}$$

We can easily establish connections between different classes of attributes, as summarized by the next theorem.

Theorem 6. *The following properties hold:*

(*a*) $\forall R \in \mathrm{RED}$, $\mathrm{CORE} \subseteq R \subseteq \mathrm{USEFUL}$;
(*b*) $\mathrm{CORE} = \mathrm{USEFUL} - \mathrm{MARGINAL}$,
$\mathrm{CORE} = C - \mathrm{NONCORE}$;
(*c*) $\mathrm{USEFUL} = \mathrm{CORE} \cup \mathrm{MARGINAL}$,
$\mathrm{USEFUL} = C - \mathrm{NONUSEFUL}$;
(*d*) $\mathrm{MARGINAL} = \mathrm{USEFUL} - \mathrm{CORE}$,
$\mathrm{MARGINAL} = C - (\mathrm{CORE} \cup \mathrm{NONUSEFUL})$;
(*e*) $\mathrm{NONUSEFUL} = C - \mathrm{USEFUL}$,
$\mathrm{NONUSEFUL} = C - (\mathrm{CORE} \cup \mathrm{MARGINAL})$.

The pair (CORE, USEFUL) constitutes a lower and an upper bounds of an arbitrary reduct. Although the set of core attributes is defined by the intersection of all reducts, one can have a simple method to determine if an attribute is a core attribute without computing all reducts (see discussion in the next subsection). Based on a result given by Moshkov, Skowron, and Suraj [21], although one can construct the set of useful attributes in polynomial time, it is practically a difficult task. Thus, the study of the class of useful attribute as an upper of a reduct is more of a theoretical value. The pairs (CORE, NONCORE) and (USEFUL, NONUSEFUL) are two complementary sets of attributes.

The three-way classification provides a qualitative three-level characterization of attributes, from important to marginally important and to unimportant. A core attribute is in all reducts and plays a primary role in a decision table. A marginal attribute is in at least one reduct, but not in all reducts and plays a secondary role. A nonuseful attribute is not in any reduct and does not play a useful role. Our definition of three-way classification is adopted from Wei, Li and Zhang [39] and Gao and Yao [8], but we use different names for some of the three classes. Wei, Li and Zhang [39] referred to the sets of core, marginal, and nonuseful as the sets of absolutely necessary, relatively necessary, and absolutely unnecessary attributes, respectively. Gao and Yao [8] called the sets of marginal and nonuseful the sets of useful noncore, and useless attributes, respectively. Some authors considered overlapping three-way classifications. Nguyen and Nguyen [22] called the sets of useful and nonuseful attributes the sets of reductive and redundant attributes, respectively.

There are two possible approaches to studying attributes. One method focuses on the pair of CORE and USEFUL that bounds any reduct. The other uses three pair-wise disjoint sets, namely, CORE, NONUSEFUL, and MARGINAL, which leads to a clearer analysis. Therefore, we analyze connections between the two types of reducts based on the three sets of core, nonuseful, and marginal attributes.

In order to have some hints for a systematic study, we summarize the main results of three-way classification of attributes for the three examples in Section 2.2. In Tables 4–6, we also provide the intersection and union of the family of sets of class-specific core, nonuseful and marginal attributes. With the help of these results, we are ready to examine relationships between three-way classifications with respect to the two types of reducts in the rest of this section.

5.2. Classification-based and class-specific core attributes

The set of core attributes consists of the most important attributes. One easily tests if an attribute is a core attribute based on the following lemma [24].

Table 4
Three-way classification of attributes in Table 1.

	RED	CORE	NONUSEFUL	MARGINAL
D_1	$\{\{c_1, c_2\}\}$	$\{c_1, c_2\}$	$\{c_3\}$	$\emptyset$
D_2	$\{\{c_2, c_3\}\}$	$\{c_2, c_3\}$	$\{c_1\}$	$\emptyset$
D_3	$\{\{c_1, c_3\}\}$	$\{c_1, c_3\}$	$\{c_2\}$	$\emptyset$
$\bigcap$	–	$\emptyset$	$\emptyset$	$\emptyset$
$\bigcup$	–	$\{c_1, c_2, c_3\}$	$\{c_1, c_2, c_3\}$	$\emptyset$
π_D	$\{\{c_1, c_2, c_3\}\}$	$\{c_1, c_2, c_3\}$	$\emptyset$	$\emptyset$

Table 5
Three-way classification of attributes in Table 2.

	RED	CORE	NONUSEFUL	MARGINAL
D_1	$\{\{c_1, c_2, c_3\}, \{c_1, c_2, c_4\}, \{c_1, c_2, c_5\}\}$	$\{c_1, c_2\}$	$\emptyset$	$\{c_3, c_4, c_5\}$
D_2	$\{\{c_1, c_3, c_4\}, \{c_1, c_3, c_5\}\}$	$\{c_1, c_3\}$	$\{c_2\}$	$\{c_4, c_5\}$
D_3	$\{\{c_2, c_3, c_4\}, \{c_2, c_3, c_5\}\}$	$\{c_2, c_3\}$	$\{c_1\}$	$\{c_4, c_5\}$
$\bigcap$	–	$\emptyset$	$\emptyset$	$\{c_4, c_5\}$
$\bigcup$	–	$\{c_1, c_2, c_3\}$	$\{c_1, c_2\}$	$\{c_3, c_4, c_5\}$
π_D	$\{\{c_1, c_2, c_3, c_4\}, \{c_1, c_2, c_3, c_5\}\}$	$\{c_1, c_2, c_3\}$	$\emptyset$	$\{c_4, c_5\}$

Table 6
Three-way classification of attributes in Table 3.

	RED	CORE	NONUSEFUL	MARGINAL
D_1	$\{\{c_1\}, \{c_2\}, \{c_3\}, \{c_4, c_5\}\}$	$\emptyset$	$\emptyset$	$\{c_1, c_2, c_3, c_4, c_5\}$
D_2	$\{\{c_2, c_3\}, \{c_2, c_4\}, \{c_4, c_5\}\}$	$\emptyset$	$\{c_1\}$	$\{c_2, c_3, c_4, c_5\}$
D_3	$\{\{c_2, c_3\}, \{c_2, c_4\}, \{c_4, c_5\}\}$	$\emptyset$	$\{c_1\}$	$\{c_2, c_3, c_4, c_5\}$
$\bigcap$	–	$\emptyset$	$\emptyset$	$\{c_2, c_3, c_4, c_5\}$
$\bigcup$	–	$\emptyset$	$\{c_1\}$	$\{c_1, c_2, c_3, c_4, c_5\}$
π_D	$\{\{c_2, c_3\}, \{c_2, c_4\}, \{c_4, c_5\}\}$	$\emptyset$	$\{c_1\}$	$\{c_2, c_3, c_4, c_5\}$

Lemma 3. *The sets of classification-based and class-specific core attributes can be computed, respectively, by:*

(i) $\mathrm{CORE}(\pi_D) = \{a \mid \mathrm{POS}(\pi_D|\pi_{C-\{a\}}) \neq \mathrm{POS}(\pi_D|\pi_C)\}$,
(ii) $\mathrm{CORE}(D_i) = \{a \mid \mathrm{POS}(D_i|\pi_{C-\{a\}}) \neq \mathrm{POS}(D_i|\pi_C)\}$.

Yao [43] argued that there are two sides of the theory of rough sets, concerning the conceptual and computational formulations and definitions, respectively. While a conceptual formulation focuses on the meaning of a concept or a notion, a computational formulation focuses on an effective method for constructing an instance of the concept. The argument can shed light on an understanding of the two definitions of the set of core attributes given by Definition 5 and Lemma 3.

By Lemma 3, an attribute is a core attribute if it is needed to preserve the positive region. This does not require computing the set of all reducts as defined by Definition 5. For this reason, Pawlak [24] defined the notion of a core attribute by the inequality condition of Lemma 3(i). However, such a definition does not provide an explicit link to the notion of a reduct. In contrast, Definition 5 provides an explicit link, which leads to a better understanding of reducts and core attributes. Instead of using Lemma 3 as a definition of a core attribute, we use definitions in terms of the intersection of the family of reducts. On the other hand, Definition 5 does not explicitly provide a computationally efficient method, as it requires the computation of the set of all reducts. In summary, in the context of attribute reducts, Definition 5 is a conceptual definition of a core attribute and Lemma 3 provides a computational definition. Working together, the two formulations offer both an in-depth understanding and a computationally efficient method.

By combining Lemma 1(ii) and Lemma 3(i), we immediately have another useful definition of the set of classification-based core attributes.

Lemma 4. *The set of classification-based core attributes can be equivalently defined by:*

$$\mathrm{CORE}(\pi_D) = \{a \mid \exists 1 \leq i \leq m (\mathrm{POS}(D_i|\pi_{C-\{a\}}) \neq \mathrm{POS}(D_i|\pi_C))\}. \tag{13}$$

By combining Lemma 3(i) and Lemma 4, we can establish a connection between the two types of core attributes.

Lemma 5. *For any decision class D_i, its set of class-specific core attributes is a subset of the set of classification-based core attributes, i.e.,*

$$\forall 1 \leq i \leq m(\mathrm{CORE}(D_i) \subseteq \mathrm{CORE}(\pi_D)). \tag{14}$$

The result of Lemma 5 can be equivalently expressed in terms of the union of the family of the sets of class-specific core attributes:

$$\bigcup_{i=1}^{m} \mathrm{CORE}(D_i) \subseteq \mathrm{CORE}(\pi_D). \tag{15}$$

That is, any class-specific core attribute is a classification-based core attribute.

Consider now the reverse direction of Eq. (15). Suppose an attribute is a classification-based core attribute, i.e., $a \in \mathrm{CORE}(\pi_D)$. By Lemma 4, there exists at least one decision class such that $\mathrm{POS}(D_i|\pi_{C-\{a\}}) \neq \mathrm{POS}(D_i|\pi_C)$. By Lemma 3(ii), it follows that $a \in \mathrm{CORE}(D_i)$. Immediately, this gives the following lemma.

Lemma 6. $\forall a \in \mathrm{CORE}(\pi_D), \exists 1 \leq i \leq m(a \in \mathrm{CORE}(D_i))$.

In terms of the union of the family of the sets of class-specific core attributes, the result of Lemma 6 can be equivalently expressed as:

$$\mathrm{CORE}(\pi_D) \subseteq \bigcup_{i=1}^{m} \mathrm{CORE}(D_i). \tag{16}$$

Different decision classes may have different sets of class-specific core attributes. Usually, a set of class-specific core attributes for a specific decision class may not include all classification-based core attributes.

By combining Eqs. (15) and (16), we can establish a connection between two types of core attributes.

Theorem 7. *The set of classification-based core attributes is the union of all sets of class-specific core attributes, namely,*

$$\mathrm{CORE}(\pi_D) = \bigcup_{i=1}^{m} \mathrm{CORE}(D_i). \tag{17}$$

Theorem 7 is a counterpart of Theorem 2 given by Chen and Tsang [4] regarding sets of core attributes in the study of local reducts. By the theorem, conceptually, we can construct the set of classification-based core attributes by taking the union of the family of class-specific core attributes. Computationally, we have a more efficient construction method. For an attribute, instead of constructing the positive region of π_D that is required by Lemma 3(i), we can sequentially construct the positive regions of decision classes and verify Lemma 3(ii). We usually may determine if an attribute is a core attribute before exhausting all decision classes.

5.3. Classification-based and class-specific nonuseful attributes

From Tables 4–6, we can make several observations regarding the relationships between the set of classification-based nonuseful attributes and the set of class-specific nonuseful attributes of a particular decision class. There may exist a decision class whose set of nonuseful attributes is not a subset of the set of classification-based nonuseful attributes. For instance, in Table 4 the set of D_1-specific nonuseful attributes is $\{c_3\}$ and the set of classification-based nonuseful attributes is the empty set $\emptyset$. There may exist a decision class whose set of nonuseful attributes is not a superset of the set of classification-based nonuseful attributes. For instance, in Table 6 the set of D_1-specific nonuseful attributes is the empty set $\emptyset$ and the set of classification-based nonuseful attributes is $\{c_1\}$. There may exist a decision class whose set of nonuseful attributes has a nonempty intersection with the set of classification-based nonuseful attributes. For instance, in Table 6 both the set of D_2-specific nonuseful attributes and the set of classification-based nonuseful attributes are $\{c_1\}$. Based on these observations, in contrast to the case of core attributes, it is impossible to study the relationships between two types of nonuseful attributes by using the set of classification-based nonuseful attributes and the set of class-specific nonuseful attributes of a particular decision class. Instead, we must focus on the family of the sets of class-specific nonuseful attributes of all decision classes.

A class-specific nonuseful attributes with respect to a particular class is not in any of its reducts. That is, the attribute is not useful for preserving the positive region of the decision class. A partition π_D is a family of pair-wise disjoint decision classes. Intuitively, if an attribute is not useful for all decision class, it is not useful for the partition.

Theorem 8. *The intersection of the family of the sets of class-specific nonuseful attributes is a subset of the set of classification-based nonuseful attributes, that is,*

$$\bigcap_{i=1}^{m} \mathrm{NONUSEFUL}(D_i) \subseteq \mathrm{NONUSEFUL}(\pi_D). \tag{18}$$

Table 7
Decision table IV.

OB	c_1	c_2	c_3	d
o_1	1	0	1	1
o_2	0	1	1	1
o_3	0	0	1	1
o_4	0	1	1	2
o_5	0	0	1	2
o_6	0	2	1	2
o_7	1	1	0	2

Table 8
Three-way classification of attributes in Table 7.

	RED	CORE	NONUSEFUL	MARGINAL
D_1	$\{\{c_1, c_2\}, \{c_1, c_3\}\}$	$\{c_1\}$	$\emptyset$	$\{c_2, c_3\}$
D_2	$\{\{c_1, c_2\}, \{c_2, c_3\}\}$	$\{c_2\}$	$\emptyset$	$\{c_1, c_3\}$
$\bigcap$	–	$\emptyset$	$\emptyset$	$\{c_3\}$
$\bigcup$	–	$\{c_1, c_2\}$	$\emptyset$	$\{c_1, c_2, c_3\}$
π_D	$\{\{c_1, c_2\}\}$	$\{c_1, c_2\}$	$\{c_3\}$	$\emptyset$

By Theorem 8, $\bigcap_{i=1}^{m}$ NONUSEFUL(D_i) is a lower bound of the set of classification-based nonuseful attributes. The converse of Theorem 8 is not necessarily true. That is, it is possible that,

$$\text{NONUSEFUL}(\pi_D) \nsubseteq \bigcap_{i=1}^{m} \text{NONUSEFUL}(D_i). \tag{19}$$

A classification-based nonuseful attribute may only be nonuseful for some decision classes, but not nonuseful for all decision classes. For some attributes, although they are nonuseful for the entire classification π_D, they are indeed useful for some specific classes. By looking at the classification, we may overlook some useful attributes for a particular decision class.

To establish an upper bound of NONUSEFUL(π_D), it is attempting to conclude that a classification-based nonuseful attribute is nonuseful for at least one decision class. That is, the union of the family of the sets of class-specific nonuseful attributes seems to provide an upper bound. Unfortunately, this is not the case, as demonstrated by a simple example. Table 7 is an inconsistent decision table with $OB = \{o_1, \dots, o_7\}$, $C = \{c_1, c_2, c_3\}$, and $D = \{d\}$. Decision classification π_D consists of two decision classes, $D_1 = \{o_1, o_2, o_3\}$ and $D_2 = \{o_4, o_5, o_6, o_7\}$. In the table, we have:

$$\begin{aligned}
\pi_C &= \{\{o_1\}, \{o_2, o_4\}, \{o_3, o_5\}, \{o_6\}, \{o_7\}\},\\
\text{POS}(\pi_D|\pi_C) &= \{o_1, o_6, o_7\},\\
\text{POS}(D_1|\pi_C) &= \{o_1\},\\
\text{POS}(D_2|\pi_C) &= \{o_6, o_7\}.
\end{aligned}$$

Table 8 provides the results of three-way classification of attributes. It can be seen that,

$$\text{NONUSEFUL}(\pi_D) = \{c_3\} \nsubseteq \emptyset = \text{NONUSEFUL}(D_1) \cup \text{NONUSEFUL}(D_2).$$

Therefore, the union of the family of sets of class-specific nonuseful attributes of all decision classes cannot be an upper bound of the set of classification-based nonuseful attributes.

This example in fact shows that a classification-based nonuseful attribute may in fact be useful for all decision classes. Such a counter-intuitive result gives us more insights into the differences regarding importance of attributes with respect to the entire classification and with respect to individual classes.

By Theorem 6(c), the set of useful attributes can be expressed as the set complement of the set of nonuseful attributes, namely,

$$\text{USEFUL} = C - \text{NONUSEFUL}. \tag{20}$$

We can therefore easily derive the relationships between the two types of useful attributes.

Theorem 9. *The set of classification-based useful attributes is a subset of the union of the family of sets of class-specific useful attributes, that is,*

$$\text{USEFUL}(\pi_D) \subseteq \bigcup_{i=1}^{m} \text{USEFUL}(D_i). \tag{21}$$

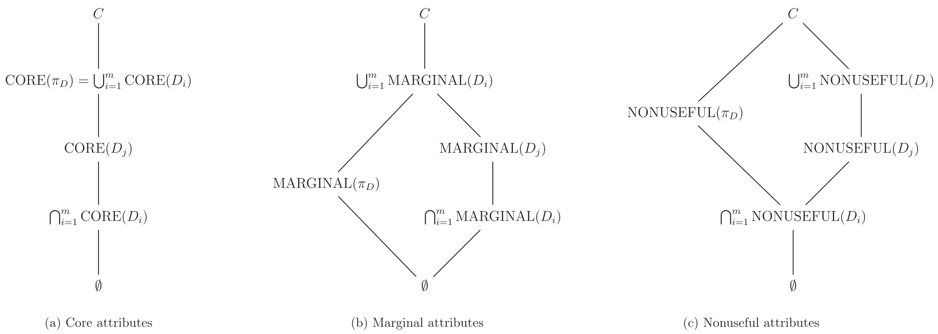

Fig. 1. Hasse diagrams of sets of attributes.

According to Theorem 9, a classification-based useful attribute is useful for at least one decision class. On the other hand, a class-specific useful attribute for a particular decision class may not be useful for the entire classification. Moreover, an attribute that is useful for all individual classes may not be useful for the classification. For instance, in the previous example we have:

$$\mathrm{USEFUL}(D_1) \cap \mathrm{USEFUL}(D_2) = \{c_1, c_2, c_3\} \not\subseteq \{c_1, c_2\} = \mathrm{USEFUL}(\pi_D).$$

Although c_3 is useful for both classes D_1 and D_2, it is not useful for $\pi_D = \{D_1, D_2\}$. The two interpretations of usefulness need further attention in a study of attribute reduction in rough set theory.

5.4. Classification-based and class-specific marginal attributes

By Theorem 6(d), we have:

$$\mathrm{MARGINAL} = \mathrm{USEFUL} - \mathrm{CORE}. \tag{22}$$

According to Theorems 7 and 9, we immediately establish the relationships between two types of marginal attributes.

Theorem 10. *The set of classification-based marginal attributes is a subset of the union of the family of sets of class-specific marginal attributes. That is,*

$$\mathrm{MARGINAL}(\pi_D) \subseteq \bigcup_{i=1}^{m} \mathrm{MARGINAL}(D_i). \tag{23}$$

According to Theorem 10, a classification-based marginal attribute is a marginal attribute for at least one decision class. However, the reverse of Theorem 10 is not necessarily true, which can be verified by results in Table 8:

$$\mathrm{MARGINAL}(D_1) \cup \mathrm{MARGINAL}(D_2) = \{c_1, c_2, c_3\} \not\subseteq \emptyset = \mathrm{MARGINAL}(\pi_D).$$

A marginal attribute of a particular decision class may not necessarily be a marginal attribute for the entire decision classification π_D. The intersection of the family of sets of class-specific marginal attributes, $\bigcap_{i=1}^{m} \mathrm{MARGINAL}(D_i)$, is not a lower bound of the set of classification-based marginal attributes. That is, a class-specific marginal attribute of all decision classes may not necessarily be a classification-based marginal attribute.

5.5. A summary of relationships between classes of attributes

In the last three subsections, we restrict the discussion of relationships with respect to the same category of attributes, namely, classification-based core, marginal, and nonuseful attributes versus the corresponding class-specific core, marginal, and nonuseful attributes. With respect to core attributes, there are possibly five different sets of attributes. With respect to marginal and nonuseful attributes, there are possibly six different sets of attributes. The inclusion relation of these sets of attributes are summarized by the three Hasse diagrams of Fig. 1.

We extend the discussion in this subsection by considering relationships between different categories. Three-way classification of attributes produces two fundamental nested sequences of sets of attributes:

$$\begin{aligned} &\mathrm{CORE} \subseteq \mathrm{USEFUL} \subseteq C, \\ &\mathrm{NONUSEFUL} \subseteq \mathrm{NONCORE} \subseteq C. \end{aligned} \tag{24}$$

where $\mathrm{NONCORE} = C - \mathrm{CORE}$. The differences between two adjacent sets result in the following connections:

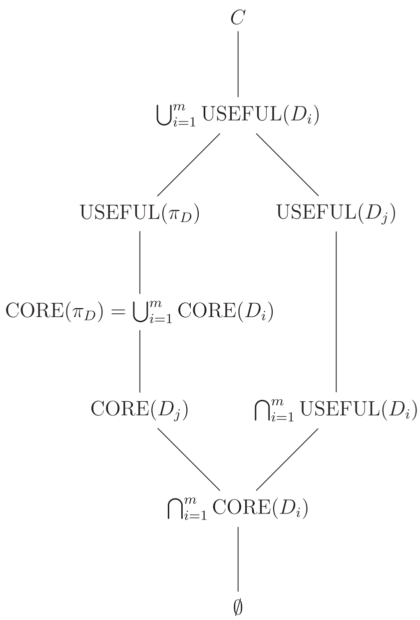

Fig. 2. Hasse diagram of sets of attributes.

$$\begin{aligned} \mathrm{MARGINAL} &= \mathrm{USEFUL} - \mathrm{CORE} \\ &= \mathrm{NONCORE} - \mathrm{NONUSEFUL}, \\ \mathrm{NONUSEFUL} &= C - \mathrm{USEFUL}, \\ \mathrm{CORE} &= C - \mathrm{NONCORE}. \end{aligned} \tag{25}$$

Thus, it is sufficient to focus on one of the two sequences. We consider the first sequence.

From the families of the sets of class-specific core and useful attributes, we obtain four classes of attributes through their intersection and union, i.e.,

$$\bigcap_{i=1}^{m} \mathrm{CORE}(D_i), \ \bigcup_{i=1}^{m} \mathrm{CORE}(D_i), \ \bigcap_{i=1}^{m} \mathrm{USEFUL}(D_i), \ \bigcup_{i=1}^{m} \mathrm{USEFUL}(D_i). \tag{26}$$

According to Eq. (24) and properties of set intersection and union, the first and the last are the minimum and maximum, respectively, with respect to the set inclusion relation. That is, $\bigcap_{i=1}^{m} \mathrm{CORE}(D_i)$ and $\bigcup_{i=1}^{m} \mathrm{USEFUL}(D_i)$ are a subset and a superset of each of the four classes. The relationship between the middle two sets needs further consideration. From Table 5, we can make several observations. In the table, we have $\mathrm{CORE}(D_1) \cup \mathrm{CORE}(D_2) \cup \mathrm{CORE}(D_3) = \{c_1, c_2, c_3\}$ and $\mathrm{USEFUL}(D_1) \cap \mathrm{USEFUL}(D_2) \cap \mathrm{USEFUL}(D_3) = \{c_3, c_4, c_5\}$. This suggests that in a decision table $\bigcup_{i=1}^{m} \mathrm{CORE}(D_i)$ may not be a subset nor a superset of $\bigcap_{i=1}^{m} \mathrm{USEFUL}(D_i)$. It also suggests that the two intersections of sets may not be empty. From the four sets of attributes, we can divide C into six pair-wise disjoint classes, and some of them may be empty.

By Theorems 7–10 and Eqs. (24) and (25), we can derive relationships between various classes of attributes. Consider the sets of classification-based core and useful attributes, class-specific core and useful attributes of a specific decision class D_j, and the intersection and union of the families of the sets of class-specific core and useful attributes. There is a total of nine possibly different classes, including the empty set and the entire set of condition attributes. Their relationships are summarized by two nest sequences of sets in the next theorem.

Theorem 11. *The following inclusion relationships hold:*

$$\begin{aligned} &\text{(i)}\ \emptyset \subseteq \bigcap_{i=1}^{m} \mathrm{CORE}(D_i) \subseteq \mathrm{CORE}(D_j) \subseteq \bigcup_{i=1}^{m} \mathrm{CORE}(D_i) = \mathrm{CORE}(\pi_D) \\ &\quad \subseteq \mathrm{USEFUL}(\pi_D) \subseteq \bigcup_{i=1}^{m} \mathrm{USEFUL}(D_i) \subseteq C, \\ &\text{(ii)}\ \emptyset \subseteq \bigcap_{i=1}^{m} \mathrm{CORE}(D_i) \subseteq \bigcap_{i=1}^{m} \mathrm{USEFUL}(D_i) \subseteq \mathrm{USEFUL}(D_j) \subseteq \bigcup_{i=1}^{m} \mathrm{USEFUL}(D_i) \subseteq C. \end{aligned} \tag{27}$$

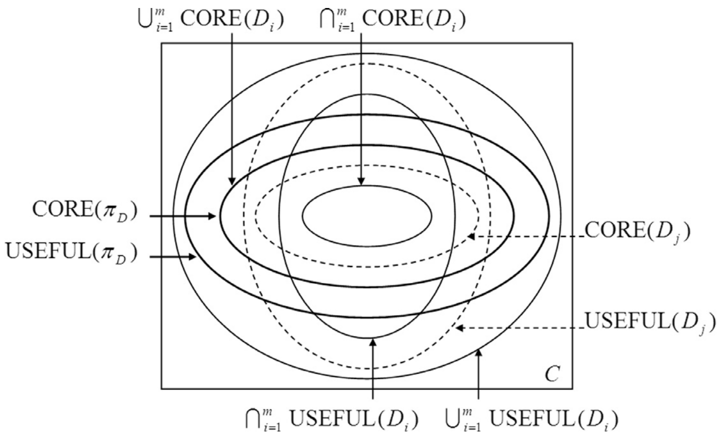

Fig. 3. Relationships between classes of attributes.

The two sequences have a common subsequence, namely,

$$\emptyset \subseteq \bigcap_{i=1}^{m} \mathrm{CORE}(D_i) \subseteq \bigcup_{i=1}^{m} \mathrm{USEFUL}(D_i) \subseteq C. \tag{28}$$

The set $\bigcap_{i=1}^{m} \mathrm{USEFUL}(D_i)$ is not necessarily a superset nor a subset of other three sets in Sequence (i). The same is true for the set $\mathrm{USEFUL}(D_j)$. A Hasse diagram of the nine possibly different sets is given by Fig. 2 and the Venn diagram is given by Fig. 3. In Fig. 3, the two bold lines indicate the sets of classification-based core and useful attributes. The two dash lines indicate the sets of class-specific core and useful attributes of a specific decision class D_j. The remaining four lines indicate the sets of attributes in Eq. (26).

6. Conclusions

Existing studies on attribute reduction in rough set theory have been dominated by classification-based attribute reducts. A classification-based reduct is defined with respect to all decision classes, representing a compromise, or an average, of all decision classes. A classification-based reduct may not work equally well for every decision class. Our notion of a class-specific attribute reduct considers only a particular decision class. A class-specific reduct is tailored to a decision class. Different decision classes may have possibly different attribute reducts.

The introduction of class-specific attribute reducts opens a new avenue of research. As a first step, we present a theoretical study on properties of class-specific reducts and the relationships between classification-based reducts and class-specific reducts. We can derive a class-specific reduct from a classification-based reduct and derive a classification-based reduct from a family of class-specific reducts. The families of all class-specific reducts provide a pair of lower and upper bounds of the family of all classification-based reducts.

We consider a three-way classification of attributes into the pair-wise disjoint sets of core, marginal, and nonuseful attributes. The set of classification-based core attributes is the union of the families of the sets of class-specific core attributes of all decision classes. It is only possible to obtain an upper of the set of classification-based marginal attributes and a lower bound of the set of classification-based nonuseful attributes based on the family of the class-specific corresponding sets of attributes.

This paper focuses on conceptual formulation. The theoretical results establish a basis for studying class-specific attribute reducts. On such a basis, it is possible to study computational issues in the design of new algorithms for constructing a classification-based reduct through a family of class-specific reducts. As future work, we need to evaluate experimentally the practical value of class-specific reducts in terms of the gain of the class-specific reducts over classification-based reducts. In addition, we can study class-specific versions of other categories of reducts, including, for example, approximate reducts [28], intra-class reducts, inter-class reducts, and constructs [31–33], indiscernibility reducts, discernibility reducts, and indiscernibility-and-discernibility reducts [34,50].

Acknowledgments

This work was supported in part by a Discovery Grant from NSERC, Canada and grants from the National Natural Science Foundation of China (61673285, 61203285). The authors thank Associate Editor and reviewers for their constructive comments and Professor Degang Chen for discussion on local attribute reducts.

Appendix A. Proofs of lemmas and theorems

Proof of Lemma 1. (i) According to Definition 2, $\mathrm{POS}(\pi_D|\pi_R) = \mathrm{POS}(\pi_D|\pi_C)$ is equivalent to

$$\bigcup_{i=1}^{m} \mathrm{POS}(D_i|\pi_R) = \bigcup_{i=1}^{m} \mathrm{POS}(D_i|\pi_C). \tag{29}$$

Thus, if $\forall D_i \in \pi_D(\mathrm{POS}(D_i|\pi_R) = \mathrm{POS}(D_i|\pi_C))$, then $\mathrm{POS}(\pi_D|\pi_R) = \mathrm{POS}(\pi_D|\pi_C)$. In a decision classification, decision classes are pair-wise disjoint. It follows that their positive regions are pair-wise disjoint. According to the monotonicity given by Eq. (5), if $\mathrm{POS}(\pi_D|\pi_R) = \mathrm{POS}(\pi_D|\pi_C)$, we must have $\forall D_i \in \pi_D(\mathrm{POS}(D_i|\pi_R) = \mathrm{POS}(D_i|\pi_C))$.

(ii) According to Definition 2, $\mathrm{POS}(\pi_D|\pi_{R-\{a\}}) \neq \mathrm{POS}(\pi_D|\pi_R)$ is equivalent to

$$\bigcup_{i=1}^{m} \mathrm{POS}(D_i|\pi_{R-\{a\}}) \neq \bigcup_{i=1}^{m} \mathrm{POS}(D_i|\pi_R). \tag{30}$$

By the fact that positive regions of decision classes are pair-wise disjoint and the monotonicity given by Eq. (5), if $\exists D_i \in \pi_D(\mathrm{POS}(D_i|\pi_{R-\{a\}}) \neq \mathrm{POS}(D_i|\pi_R))$, then $\mathrm{POS}(\pi_D|\pi_{R-\{a\}}) \neq \mathrm{POS}(\pi_D|\pi_R)$. The other direction can be easily proved based on the same observations. □

Proof of Lemma 3. In his book on rough sets, Pawlak [24] proved (i). We prove (ii) by following essentially the same argument.

($\subseteq$) Assume $b \in \mathrm{CORE}(D_i)$. By definition, $b \in R_i$ for all $R_i \in \mathrm{RED}(D_i)$. Now we can prove that $\mathrm{POS}(D_i|\pi_{C-\{b\}}) \neq \mathrm{POS}(D_i|\pi_C)$ by contradiction. Assume $\mathrm{POS}(D_i|\pi_{C-\{b\}}) = \mathrm{POS}(D_i|\pi_C)$. There must exists a D_i-specific reduct $R_i' \subseteq C - \{b\}$. This contradicts with $b \in R_i$ for all $R_i \in \mathrm{RED}(D_i)$. Therefore, $\mathrm{POS}(D_i|\pi_{C-\{b\}}) \neq \mathrm{POS}(D_i|\pi_C)$, which implies $b \in \{a \mid \mathrm{POS}(D_i|\pi_{C-\{a\}}) \neq \mathrm{POS}(D_i|\pi_C)\}$.

($\supseteq$) Assume $b \in \{a \mid \mathrm{POS}(D_i|\pi_{C-\{a\}}) \neq \mathrm{POS}(D_i|\pi_C)\}$. We have $\mathrm{POS}(D_i|\pi_{C-\{b\}}) \neq \mathrm{POS}(D_i|\pi_C)$. We prove by contradiction that $b \in R_i$ for all $R_i \in \mathrm{RED}(D_i)$. Assume that there exists a D_i-specific reduct R_i' such that $b \notin R_i'$. By definition, we have $\mathrm{POS}(D_i|\pi_{R_i'}) = \mathrm{POS}(D_i|\pi_C)$. Since $R_i' \subseteq C - \{b\}$, by the monotonicity given by Eq. (5), we have $\mathrm{POS}(D_i|\pi_{R_i'}) \subseteq \mathrm{POS}(D_i|\pi_{C-\{b\}})$. It follows that $\mathrm{POS}(D_i|\pi_{C-\{b\}}) = \mathrm{POS}(D_i|\pi_C)$, leading to a contradication. Therefore, $b \in R_i$ for all $R_i \in \mathrm{RED}(D_i)$, which is equivalent to $b \in \mathrm{CORE}(D_i)$. □

Proof of Theorem 8. Suppose $a \notin \mathrm{NONUSEFUL}(\pi_D)$, i.e., $a \in \mathrm{USEFUL}(\pi_D) = \bigcup \mathrm{RED}(\pi_D)$. Then, there exists $R \in \mathrm{RED}(\pi_D)$ such that $a \in R$. By Theorem 4, there exists a family of class-specific reducts $(R_1, \ldots, R_m) \in \mathrm{CRED}$ such that $R = \bigcup_{i=1}^{m} R_i$. Thus, there exists $1 \le i \le m$ such that $a \in R_i$ and $a \in \mathrm{USEFUL}(D_i)$, i.e., $a \notin \mathrm{NONUSEFUL}(D_i)$. Furthermore, $a \notin \bigcap_{i=1}^{m} \mathrm{NONUSEFUL}(D_i)$, so $\bigcap_{i=1}^{m} \mathrm{NONUSEFUL}(D_i) \subseteq \mathrm{NONUSEFUL}(\pi_D)$. □

Proof of Theorem 10. According to Theorems 6(d), 7 and 9, we have:

$$\begin{aligned}
\mathrm{MARGINAL}(\pi_D) &= \mathrm{USEFUL}(\pi_D) - \mathrm{CORE}(\pi_D) \\
&\subseteq \left(\bigcup_{i=1}^{m} \mathrm{USEFUL}(D_i)\right) - \left(\bigcup_{j=1}^{m} \mathrm{CORE}(D_j)\right) \\
&= \left(\bigcup_{i=1}^{m} \mathrm{USEFUL}(D_i)\right) \cap \left(\bigcup_{j=1}^{m} \mathrm{CORE}(D_j)\right)^c \\
&= \left(\bigcup_{i=1}^{m} \mathrm{USEFUL}(D_i)\right) \cap \left(\bigcap_{j=1}^{m} (\mathrm{CORE}(D_j))^c\right) \\
&= \bigcup_{i=1}^{m} \left(\mathrm{USEFUL}(D_i) \cap \left(\bigcap_{j=1}^{m} (\mathrm{CORE}(D_j))^c\right)\right) \\
&\subseteq \bigcup_{i=1}^{m} (\mathrm{USEFUL}(D_i) \cap (\mathrm{CORE}(D_i))^c) \\
&= \bigcup_{i=1}^{m} (\mathrm{USEFUL}(D_i) - \mathrm{CORE}(D_i)) \\
&= \bigcup_{i=1}^{m} \mathrm{MARGINAL}(D_i).
\end{aligned}$$

Thus, the inclusion in the theorem holds. □

References

[1] P.M. Baggenstoss, Class-specific feature sets in classification, IEEE Trans. Signal Process. 47 (1999) 3428–3432.
[2] C.M. Bishop, Pattern Recognition and Machine Learning, Springer, New York, 2006.
[3] D.G. Chen, Theory and Methods of Fuzzy Rough sets, Science Press, Beijing, 2013. (In Chinese)
[4] D.G. Chen, E.C.C. Tsang, On the local reduction of information system, in: D. Yeung, Z. Liu, X. Wang, H. Yan (Eds.), Proceedings of the International Conference on Machine Learning and Cybernetics, Lecture Notes in Computer Science, vol. 3930, 2005, pp. 588–594.
[5] D.G. Chen, S.Y. Zhao, Local reduction of decision system with fuzzy rough sets, Fuzzy Sets Syst. 161 (2010) 1871–1883.
[6] J.K. Chen, Y.J. Lin, G.P. Lin, J.J. Li, Z.M. Ma, The relationship between attribute reducts in rough sets and minimal vertex covers of graphs, Inf. Sci. 325 (2015) 87–97.
[7] R.O. Duda, P.E. Hart, Pattern Classification and Scene Analysis, Wiley, New York, 1973.
[8] C. Gao, Y.Y. Yao, An addition strategy for reduct construction, in: D. Miao, W. Pedrycz, D. Ślęzak, G. Peters, Q. Hu, R. Wang (Eds.), Proceedings of the International Conference on Rough Sets and Knowledge Technology, Lecture Notes in Computer Science, vol. 8818, 2014, pp. 535–546.
[9] Q.H. Hu, Z.X. Xie, D.R. Yu, Hybrid attribute reduction based on a novel fuzzy-rough model and information granulation, Pattern Recognit. 40 (2007) 3509–3521.
[10] A.P. Huang, H. Zhao, W. Zhu, Nullity-based matroid of rough sets and its application to attribute reduction, Inf. Sci. (NY) 263 (2014) 153–165.
[11] G.S. Huang, Y.Y. Yao, Region vector based attribute reducts in decision-theoretic rough sets, in: Y. Yao, Q. Hu, H. Yu, J. Grzymala-Busse (Eds.), Proceedings of the International Conference on Rough Sets, Fuzzy Sets, Data Mining, and Granular Computing, Lecture Notes in Computer Science, vol. 9437, 2015, pp. 355–365.
[12] X.Y. Jia, L. Shang, B. Zhou, Y.Y. Yao, Generalized attribute reduct in rough set theory, Knowl. Based Syst. 91 (2016) 204–218.
[13] R. Jensen, A. Tuson, Q. Shen, Finding rough and fuzzy-rough set reducts with SAT, Inf. Sci. (NY) 255 (2014) 100–120.
[14] M.S. Lazo-Cortés, J.F. Martínez-Trinidad, J.A. Carrasco-Ochoa, G. Sanchez-Diaz, On the relation between rough set reducts and typical testors, Inf. Sci. (NY) 294 (2015) 152–163.
[15] M. Li, C.X. Shang, S.Z. Feng, J.P. Fan, Quick attribute reduction in inconsistent decision tables, Inf. Sci. (NY) 254 (2014) 155–180.
[16] J.Y. Liang, F. Wang, C.Y. Dang, Y.H. Qian, A group incremental approach to feature selection applying rough set technique, IEEE Trans. Knowl. Data Eng. 26 (2014) 294–308.
[17] J.N.K. Liu, Y.X. Hu, Y.L. He, A set covering based approach to find the reduct of variable precision rough set, Inf. Sci. (NY) 275 (2014) 83–100.
[18] X.A. Ma, G.Y. Wang, H. Yu, T.R. Li, Decision region distribution preservation reduction in decision-theoretic rough set model, Inf. Sci. (NY) 278 (2014) 614–640.
[19] Z.Q. Meng, Z.Z. Shi, On quick attribute reduction in decision-theoretic rough set models, Inf. Sci. (NY) 330 (2016) 226–244.
[20] 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. (NY) 179 (2009) 4140–4150.
[21] M.J. Moshkov, A. Skowron, Z. Suraj, On covering attribute sets by reducts, in: M. Kryszkiewicz, J. Peters, H. Rybinski, A. Skowron (Eds.), Proceedings of the International Conference on Rough Sets and Intelligent Systems Paradigms, Lecture Notes in Computer Science, vol. 4585, 2007, pp. 175–180.
[22] L.G. Nguyen, H.S. Nguyen, Searching for reductive attributes in decision tables, in: Transactions on Rough Sets XIX, Lecture Notes in Computer Science, 2015, pp. 51–64.
[23] Z. Pawlak, Rough classification, Int. J. Man. Mach. Stud. 20 (1984) 469–483.
[24] Z. Pawlak, Rough Sets: Theoretical Aspects of reasoning About Data, Kluwer Academic Publishers, Dordrecht, 1991.
[25] B.B. Pineda-Bautista, J.A. Carrasco-Ochoa, J.F. Martínez-Trinidad, General framework for class-specific feature selection, Expert Syst. Appl. 38 (2011) 10018–10024.
[26] J. Qian, D.Q. Miao, Z.H. Zhang, X.D. Yue, Parallel attribute reduction algorithms using mapreduce, Inf. Sci. (NY) 279 (2014) 671–690.
[27] Y.H. Qian, J.Y. Liang, W. Pedrycz, C.Y. Dang, Positive approximation: an accelerator for attribute reduction in rough set theory, Artif. Intell. 174 (2010) 597–618.
[28] D. Ślęzak, Approximate entropy reducts, Fundam. Inf. 53 (2002) 365–390.
[29] S. Stawicki, D. Ślęzak, A. Janusz, S. Widz, Decision bireducts and decision reducts – a comparison, Int. J. Approx. Reason. 84 (2017) 75–109.
[30] J.J. Song, E.C.C. Tsang, D.G. Chen, X.B. Yang, Minimal decision cost reduct in fuzzy decision-theoretic rough set model, Knowl. Based Syst. 126 (2017) 104–112.
[31] R. Susmaga, Reducts versus constructs: an experimental evaluation, Electron. Notes Theor. Comput. Sci. 82 (2003) 239–250.
[32] R. Susmaga, Reducts and constructs in classic and dominance-based rough sets approach, Inf. Sci. (NY) 271 (2014) 45–64.
[33] R. Susmaga, R. Słowiński, Generation of rough sets reducts and constructs based on inter-class and intra-class information, Fuzzy Sets Syst. 274 (2015) 124–142.
[34] S.H. Teng, M. Lu, A.F. Yang, J. Zhang, Y.J. Nian, M. He, Efficient attribute reduction from the viewpoint of discernibility, Inf. Sci. (NY) 326 (2016) 297–314.
[35] C.Z. Wang, Q. He, D.G. Chen, Q.H. Hu, A novel method for attribute reduction of covering decision systems, Inf. Sci. (NY) 254 (2014) 181–196.
[36] G.Y. Wang, J. Zhao, J.J. An, Y. Wu, A comparative study of algebra viewpoint and information viewpoint in attribute reduction, Fundam. Inf. 68 (2005) 289–301.
[37] G.Y. Wang, X.A. Ma, H. Yu, Monotonic uncertainty measures for attribute reduction in probabilistic rough set model, Int. J. Approx. Reason. 59 (2015) 41–67.
[38] L.P. Wang, Y.L. Wang, Q. Chang, Feature selection methods for big data bioinformatics: a survey from the search perspective, Methods 111 (2016) 21–31.
[39] L. Wei, H.R. Li, W.X. Zhang, Knowledge reduction based on the equivalence relations defined on attribute set and its power set, Inf. Sci. (NY) 177 (2007) 3178–3185.
[40] W.Z. Wu, Attribute reduction based on evidence theory in incomplete decision systems, Inf. Sci. (NY) 178 (2008) 1355–1371.
[41] S.P. Xu, X.B. Yang, H.L. Yu, D.J. Yu, J.Y. Yang, E.C.C. Tsang, Multi-label learning with label-specific feature reduction, Knowl. Based Syst. 104 (2016) 52–61.
[42] Y.Y. Yao, Duality in rough set theory based on the square of opposition, Fundam. Inf. 127 (2013) 49–64.
[43] Y.Y. Yao, The two sides of the theory of rough sets, Knowl. Based Syst. 80 (2015) 67–77.
[44] Y.Y. Yao, R. Fu, The Concept of Reducts in Pawlak Three-Step Rough Set Analysis, in: Transactions on Rough Sets XVI, Lecture Notes in Computer Science, 2013, pp. 53–72.
[45] Y.Y. Yao, Y. Zhao, Attribute reduction in decision-theoretic rough set models, Inf. Sci. (NY) 178 (2008) 3356–3373.
[46] Y.Y. Yao, Y. Zhao, J. Wang, On reduct construction algorithms, in: Transactions on Computational Science II, Lecture Notes in Computer Science, 2008, pp. 100–117.
[47] M.L. Zhang, L. Wu, LIFT: multi-label learning with label-specific features, IEEE Trans. Pattern Anal. Mach. Intell. 37 (2015) 107–120.
[48] X.Y. Zhang, D.Q. Miao, Double-quantitative fusion of accuracy and importance: systematic measure mining, benign integration construction, hierarchical attribute reduction, Knowl. Based Syst. 91 (2016) 219–240.
[49] X.Y. Zhang, D.Q. Miao, An expanded double-quantitative model regarding probabilities and grades and its hierarchical double-quantitative attribute reduction, Inf. Sci. (NY) 299 (2015) 312–336.
[50] Y. Zhao, Y.Y. Yao, F. Luo, Data analysis based on discernibility and indiscernibility, Inf. Sci. (NY) 177 (2007) 4959–4976.