Quick attribute reduction in inconsistent decision tables

4 authors:

Min Li
Chinese Academy of Sciences
16 PUBLICATIONS **91** CITATIONS
SEE PROFILE

Changxing Shang
4 PUBLICATIONS **54** CITATIONS
SEE PROFILE

Shengzhong Feng
Chinese Academy of Sciences
33 PUBLICATIONS **305** CITATIONS
SEE PROFILE

Jianping Fan
Chinese Academy of Sciences
68 PUBLICATIONS **507** CITATIONS
SEE PROFILE

Information Sciences 254 (2014) 155–180

Contents lists available at ScienceDirect

Information Sciences

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

Quick attribute reduction in inconsistent decision tables

Min Li [a,b,c,*], Changxing Shang [a,b], Shengzhong Feng [a], Jianping Fan [a]

[a] Shenzhen Institutes of Advanced Technology, Chinese Academy of Sciences, Shenzhen, Guangdong 518055, PR China
[b] Graduate School of Chinese Academy of Sciences, Beijing 100080, PR China
[c] Nanchang Institute of Technology, Nanchang, Jiangxi 330099, PR China

ARTICLE INFO

Article history:
Received 28 November 2011
Received in revised form 5 May 2013
Accepted 19 August 2013
Available online 27 August 2013

Keywords:
Rough set
Attribute reduction
Inconsistent decision table
Assignment reduct
Distribution reduct
Maximum distribution reduct

ABSTRACT

This paper focuses on three types of attribute reducts in inconsistent decision tables: assignment reduct, distribution reduct, and maximum distribution reduct. It is quite inconvenient to judge these three types of reduct directly according to their definitions. This paper proposes judgment theorems for the assignment reduct, the distribution reduct and the maximum distribution reduct, which are expected to greatly simplify the judging of these three types of reducts. On this basis, we derive three new types of attribute significance measures and construct the Q-ARA (Quick Assignment Reduction Algorithm), the Q-DRA (Quick Distribution Reduction Algorithm), and the Q-MDRA (Quick Maximum Distribution Reduction Algorithm). These three algorithms correspond to the three types of reducts. We conduct a series of comparative experiments with twelve UCI (machine learning data repository, University of California at Irvine) data sets (including consistent and inconsistent decision tables) to evaluate the performance of the three reduction algorithms proposed with the relevant algorithm QuickReduct [9,34]. The experimental results show that QuickReduct possesses weak robustness because it cannot find the reduct even for consistent data sets, whereas our proposed three algorithms show strong robustness because they can find the reduct for each data set. In addition, we compare the Q-DRA (Quick Distribution Reduction Algorithm) with the CEBARKNC (conditional entropy-based algorithm for reduction of knowledge without a computing core) [43] because both find the distribution reduct by using a heuristic search. The experimental results demonstrate that Q-DRA runs faster than CEBARKNC does because the distribution function of Q-DRA has a lower calculation cost. Instructive conclusions for these reduction algorithms are drawn from the perspective of classification performance for the C4.5 and RBF-SVM classifiers. Last, we make a comparison between discernibility matrix-based methods and our algorithms. The experimental results indicate that our algorithms are efficient and feasible.

1. Introduction

Rough set theory is a powerful mathematical tool introduced by Pawlak [23] to address imprecise, incomplete or vague information. Many researchers have contributed to its development and applications [4,7,9,10,16,17,24–27,33–35,37,39–41,45]. One fundamental aspect of rough set theory is attribute reduction in information systems (ISs), which is selecting or reserving those attributes that provide the same information for classification purposes as the entire set of available attributes. There are many types of attribute reductions in the area of rough sets [1,5,6,8,13,15–22,33–35,38,42–44,46–54]. Pawlak proposes the classic attribute reduction, which is intended to preserve the deterministic information with respect to

[*] Corresponding author. Tel.: +86 13607002079.
E-mail address: liminghuadi@hotmail.com (M. Li).

0020-0255/$ - see front matter © 2013 Elsevier Inc. All rights reserved.
http://dx.doi.org/10.1016/j.ins.2013.08.038

decision attributes of a decision table and is therefore often applied in extracting deterministic decision rules from the decision tables. However, most of the decision tables are not consistent because of various factors such as noise in the data, lack of critical knowledge, compact representation, and prediction capability. There is no guarantee for such classic attribute reduction to preserve non-deterministic decision information in an inconsistent decision table. Regarding this concern, Kryszkiewicz presents two types of attribute reduction for inconsistent decision tables: assignment reduction and distribution reduction [11,12]. Assignment reduction maintains unchanged the possible decisions for an arbitrary object in an inconsistent decision table. In comparison, distribution reduction is a more complete knowledge reduction and is characterized by preserving the class membership distribution for all of the objects in an inconsistent decision table. In other words, the distribution reduction preserves not only all of the deterministic information but also the non-deterministic information of an inconsistent decision table. Yao thinks that the partition based on the membership distribution vector is finer and more complex, which allows the distribution reduction to preserve the quality of the decisions [46]. However, it could be a concern that the distribution reduction has strict requirements, and the decision rules derived from distribution reduction are usually less compact and more complicated. For this reason, Zhang et al. have proposed the maximum distribution reduction [51,52]. It maintains unchanged the maximum decision classes for all of the objects in a decision table, which is seen as a good compromise between the capability of preserving information with respect to decisions and the compactness of the derived rules. In Ref [47], Ye et al. have presented another extended type of attribute reduction, called M-reduct. M-reduct has stricter requirements than the maximum distribution reduction in that M-reduct rigidly retains the membership degree to the maximum decision class for each object of the decision table. This characteristic makes M-reduct more susceptible to noise contaminated data sets than the maximum distribution reduction because a small amount of noisy data can cause the M-reduct to include more attributes.

To find the three types of reducts (the assignment reduct, the distribution reduct and the maximum distribution reduct), Zhang et al. have utilized the discernibility matrixes with respect to those reducts and have obtained the corresponding Boolean function, called the discernibility function [36,51]. This function is reduced by using the distribution and the absorption laws. And then all of the reducts are generated by finding all of the prime implicants of the function. The attribute reduction based on the discernibility matrix has been extensively researched [2,13,19,21,28]. For example, a type of attribute reduction called a lower approximation reduct and an upper approximation reduct is presented [28]. This type of attribute reduction preserves the lower/upper approximation distribution of a target decision. In fact, the lower approximation reduct is equivalent to the relative positive region reduct, and the upper approximation reduct is equivalent to the assignment reduct in inconsistent (complete) decision tables. The discernibility matrixes associated with the two approximation reductions are examined as well. In paper [2], Chen et al. have constructed tolerance granules by defining a cover of the universe and have proposed a discernibility matrix-based reduction algorithm for computing the relative reducts of consistent and inconsistent covering decision systems. Similarly, the reduction approach based on discernibility matrixes is also found in [21], where the region preservation reduct (equivalent to the position region reduct), decision preservation reduct (equivalent to the assignment reduct) and relationship preservation reduct are discussed in detail. Then, three distinct definitions of discernibility matrixes are defined to find these types of reducts. More recently, a relatively systematic study of attribution reduction in inconsistent incomplete decision tables is presented in [19], where five types of discernibility function-based approaches are proposed to identify a specific type of reduct. Although discernibility matrix-based methods can find all of the reducts, the conversion from conjunction normal form to disjunction normal form constitutes an NP-hard problem. When the data set has many attributes, these discernibility matrix-based methods become not feasible because the matrix contains too many candidates. Therefore, heuristic methods are desirable. In [18], Meng and Shi developed a fast attribute reduction algorithm for incomplete decision systems based on tolerance relation rough sets. The complexity of this algorithm is no more than $O(|C|^2|U|\log|U|)$, which means that the algorithm can be used for attribute reduction in large-scale decision systems. However, the proposed approach is suitable only for computing the positive region-based reducts of an inconsistent incomplete decision system.

It is quite inconvenient to judge the three types of reducts, namely the assignment, the distribution and the maximum distribution reducts, directly according to their definitions because their definitions are rather complex. In this paper, we propose judgment theorems for the assignment reduct, the distribution reduct and the maximum distribution reduct, which are expected to greatly simplify the judging of these three types of reducts. On this basis, we derive three novel types of attribute significance measures and construct the Quick Assignment Reduction Algorithm, the Quick Distribution Reduction Algorithm, and the Quick Maximum Distribution Reduction Algorithm, so as to correspond to these three types of reducts. These three algorithms have similarities to the well-known reduction algorithm of QuickReduct [9,34] in a forward greedy search form, with respect to time and space complexities. However, QuickReduct only finds the positive region reducts, which are often applied in consistent decision tables. A series of comparative experiments with twelve UCI data sets (including consistent and inconsistent decision tables) show that the QuickReduct possesses weak robustness because it cannot find the reduct, even for consistent data sets, whereas our proposed three algorithms show strong robustness and they find the reduct for each data set. At the same time, some instructive conclusions for these reduction algorithms are drawn from the perspective of classification performance on C4.5 and RBF-SVM classifiers. The CEBARKNC [43] proves to be a reduction algorithm with efficiency because it is applied to find the distribution reduct for inconsistent decision tables. Additionally in our experiments, we compare the Quick Distribution Reduction Algorithm with the CEBARKNC because both find distribution reducts. The results show that both have a high degree of similarity: they find 9 matching reducts in the 12 data sets. Nevertheless, the experimental results prove that the simple calculation of our distribution function enables the

Quick Distribution Reduction Algorithm to run faster than CEBARKNC. Last, we make comparisons between discernibility matrix-based methods and our algorithms, and the experimental results further indicate that our algorithms are efficient and feasible.

The structure of the remainder of this paper is as follows. Section 2 presents preliminary notions related to the Pawlak rough set. Section 3 reviews the concepts for the assignment reduct, the distribution reduct and the maximum distribution reduct. Section 4 introduces the judgment theorem of the assignment reduct and presents the Quick Assignment Reduction Algorithm. Section 5 introduces the judgment theorem of the distribution reduct and presents the Quick Distribution Reduction Algorithm. Section 6 introduces the judgment theorem of the maximum distribution reduct and presents the Quick Maximum Distribution Reduction Algorithm. Section 7 conducts a series of comparative experiments to evaluate the performance of our proposed three reduction algorithms. Finally, Section 8 gives conclusions that are drawn from this study.

2. Theoretical foundations

The theory of the rough set begins with the notion of indiscernibility. Let $IS = (U, A, V, f)$ be an information table, where $U = \{x_1, x_2, \ldots, x_n\}$ is a set of finite and nonempty objects, called the universe. A is a nonempty finite set of attributes, and V is a set of values = $\cup_{q \in A} V_q$, where V_q is a value set of the attribute q; and $f : U \times A \to V$ is an information function that specifies the attribute value of $x_j \in U$. With any nonempty subset of attributes $P \subseteq A$, there is an associated equivalence relation $Ind(P) = \{(x, y) \in U^2 | \forall a \in P, a(x) = a(y)\}$, where $a(x)$ and $a(y)$ denote the values of objects x and y under a condition attribute a, respectively. This equivalence relation $Ind(P)$ divides the universe U into a family of disjoint classes, which are denoted by $U/Ind(P) = \{X_1, X_2, \ldots, X_S\}$(thereinafter denoted by U/P for simplicity), where X_i is an equivalence class induced by $Ind(P)$, $i = 1, 2, \ldots, S$. Obviously, any two objects that belong to the same equivalence class $X_i \in U/P$ are indiscernible according to the attribute set P.

Given an arbitrary set $X \subseteq U, X$ can be approximated using only the information that is contained within P by constructing two unions of elemental sets $\underline{P}(X) = \cup_{Xi \subseteq X} Xi$ and $\overline{P}(X) = \cup_{Xi \cap X \neq \phi} Xi$, where $\underline{P}(X)$ and $\overline{P}(X)$ are called P-lower and P-upper approximations of X in IS. These definitions state that object $x \in \underline{P}(X)$ belongs certainly to X, while object $x \in \overline{P}(X)$ could belong to X. Obviously, there is $\underline{P}(X) \subseteq X \subseteq \overline{P}(X)$. A set X is said to be definable if $\underline{P}(X) = \overline{P}(X)$; otherwise, X is said to be rough. The difference between $\underline{P}(X)$ and $\overline{P}(X)$ is called the P-boundary region of X, which is denoted as $BN_P(X) = \overline{P}(X) - \underline{P}(X)$.

One type of special information table is called a decision table, which is denoted as $DT = (U, A = C \cup D, V, f)$ with $C \cap D = \phi$, where an element of C is called a condition attribute, C is called condition attribute set, an element of D is called a decision attribute, and D is called decision attribute set. Given an arbitrary condition attribute subset $B \subseteq C$, then $POS_B(D) = \cup_{X \in U/D} \underline{B}(X)$ is called the positive region of D with respect to the condition attribute set B. Additionally, the positive region can also be expressed as $POS_B(D) = \cup_{Y \in U/B \wedge |Y/D|=1} Y$ ($|Y/D|$ denotes the number of equivalence classes in the partition Y/D), which is a direct and quick calculation method.

Definition 2.1. Let $DT = (U, C \cup D, V, f)$ be a decision table. We say that DT is consistent if $POS_C(D) = U$; otherwise, DT is inconsistent.

Definition 2.2. Let $DT = (U, C \cup D, V, f)$ be a decision table. We say that $a \in B \subseteq C$ is relatively dispensable in B if $POS_{B-\{a\}}(D) = POS_B(D)$; otherwise, a is said to be relatively indispensable in B. If every attribute in B is relatively indispensable, then we say that B is relatively independent in DT.

One of the major applications of rough set theory is attribute reduction, which means selecting or reserving those condition attributes that provide the same information for classification purposes as the entire set of available attributes. More precisely, a classic relative positive region reduct with respect to the decision D is defined as follows:

Definition 2.3. Let $DT = (U, C \cup D, V, f)$ be a decision table. We say that attribute set $B \subseteq C$ is a relative positive region reduct if $POS_B(D) = POS_C(D)$, and B is relatively independent in DT.

Definition 2.4. Let $DT = (U, C \cup D, V, f)$ be a decision table, with $X \subseteq U(X \neq \phi)$ and $B \subseteq C$. The approximation quality of X by the attribute set B is defined as:

$$r_B(X) = \frac{|\underline{B}(X)|}{|X|}, \tag{1}$$

where $|\bullet|$ denotes the cardinality of set.

Obviously, for any condition attribute set $B \subseteq C$, we have $r_B(X) \leqslant r_C(X)$. This relation means that the approximation quality increases monotonically when the condition attribute increases.

For most of the applications, only one relative reduct is required. Many methods in the literature [5–8,17,34,35,43] adopt a forward greedy algorithm to find a reduct on the basis of various significance measures of attributes. In rough set theory, one of the most widely applied attribute significance measures is defined as follows [45]:

$$SIG(a,B,D)=\gamma_B(D)-\gamma_{B-\{a\}}(D); \tag{2}$$

$$\gamma_B(D)=\frac{|POS_B(D)|}{|U|}=\frac{\sum_{Y\in U/B\wedge|Y/D|=1}|Y|}{|U|}. \tag{3}$$

Formula (2) expresses the significance degree of attribute a relative to the condition attribute set B with respect to the decision attribute set D. Formula (3) defines the dependency degree of D by the attribute set B(called the positive region dependency degree in this paper). The positive region dependency degree is the ratio of the number of objects included in the positive region to the total number of all of the objects in universe U. This value is also the sum of the weighted approximation quality of each decision class, as illustrated in Formula (4):

$$\gamma_B(D)=\sum_{X\in U/D}\frac{|\underline{B}(X)|}{|U|}=\sum_{X\in U/D}\frac{|X|}{|U|}r_B(X). \tag{4}$$

Formula (3) has been widely applied owing to having higher calculation efficiency than Formula (4).

The QuickReduct [9,34] algorithm listed below is typical of the algorithms that use a greedy search strategy to find a relative reduct. QuickReduct works as follows: according to Formula (2), a forward selection hill-climbing search starts with an empty set $red=\phi$, evaluates each attribute individually to find a single attribute that results in the max $SIG(a,red,D)$, and places the single attribute into the variable *red*. The search then tries each of the remaining attributes in conjunction with the attribute set *red* to find the best single attribute and places it into *red* once again. This process continues until $\gamma_B(D)=\gamma_C(D)$. However, the QuickReduct algorithm possesses weak robustness because all of the single attributes can zero the positive region dependency degree for many of the data sets, and it cannot make a good attribute choice when the significance degrees of each candidate attribute are the same. Furthermore, the stop criterion $\gamma_B(D)=\gamma_C(D)$ guarantees only the discovery of a positive region reduct, which is often applied in consistent decision tables.

Algorithm 1. QuickReduct [9] algorithm for finding the relative positive region reduct.

```
Input: Decision table $DT=(U,C\cup D,V,f)$.
Output: One relative reduct of $DT$.
$red \leftarrow \{\}$
do
  $T \leftarrow red$
  foreach  $a \in C-red$
    if $\gamma_{red\cup\{a\}}(D) > \gamma_T(D)$
      $T \leftarrow red \cup \{a\}$
  $red \leftarrow T$
until  $\gamma_{red}(D)=\gamma_C(D)$
return red
```

Another well-known attribute significance measure using conditional information entropy [32] is defined as follows [43]:

$$SIG^e(a,B,D)=H(D|B)-H(D|B\cup\{a\}); \tag{5}$$

$$H(D|B)=-\sum_{Y\in U/B}p(Y)\sum_{X\in U/D}p(X|Y)\log(p(X|Y)), \tag{6}$$

where $p(Y)=|Y|/|U|, p(X|Y)=|X\cap Y|/|Y|$. Formula (6) is the conditional information entropy of D with respect to B. Formula (5) expresses the attribute significance from the perspective of the information. On such a basis, Wang et al. developed the conditional entropy-based algorithm for the reduction of knowledge with a computing core (CEBARKCC) and the conditional entropy-based algorithm for reduction of knowledge without a computing core (CEBARKNC) [43]. The main difference lies in that the CEBARKCC algorithm computes core attributes, whereas CEBARKNC does not. The structure of the CEBARKNC algorithm is similar to the QuickReduct algorithm except that the attribute significance measure is used.

3. Concept of attribute reduction in an inconsistent decision table

For completeness, in this section, we briefly review the concepts used in the assignment reduct, the distribution reduct and the maximum distribution reduct, which can be found in Refs. [51,52].

With the attribute set B, the rough membership value of an object x that belongs to $X(X\subseteq U)$ is the probability of the object in X given that the object is in $[x]_B$. Concretely, the probabilistic interpretation of rough membership of an object x belonging to X is denoted as $\mu_X^B(x)$ and is computed as follows [27]:

$$\mu_X^B(x) = p(X|[x]_B) = \frac{|[x]_B \cap X|}{|[x]_B|}. \tag{7}$$

For a decision table $DT = (U, C \cup D, V, f)$, let $U/D = \{X_1, X_2, \ldots, X_S\}$ be the decision partition on U. Then, for an arbitrary object $x \in U$, the equivalence class $[x]_C$ could have an intersection with any of the decision equivalence class $X_i (1 \leqslant i \leqslant S)$, if DT is an inconsistent decision table.

Let $\delta_D^C(x) = \{X_j | X_j \cap [x]_C \neq \phi\}, (1 \leqslant j \leqslant S)$ be the family of decision equivalence classes that have an intersection with $[x]_C$; let $\mu_D^C(x) = [p(X_1|[x]_C), p(X_2|[x]_C), \ldots, p(X_S|[x]_C)]$ be the distribution of all of the rough memberships of object x by the condition attributes C relative to the each target decision equivalence class of partition U/D; and let $\gamma_D^C(x) = \{X_j | X_j = \arg\max_{1 \leqslant i \leqslant S} P(X_i|[x]_C)\}$ be the decision equivalence class that has the maximum rough membership of object x given in $[x]_C$ relative to all of the decision equivalence classes of partition U/D; then:

Definition 3.1. [51,52]. Let $DT = (U, C \cup D, V, f)$ be a decision table and $B \subseteq C$. Then we have:

(1) B is an assignment-consistent set of C if $\forall x \in U, \delta_D^B(x) = \delta_D^C(x)$. If B is an assignment-consistent set of C and any proper subset of B is not an assignment-consistent set, then B is an assignment reduct of C;
(2) B is a distribution-consistent set of C if $\forall x \in U, \mu_D^B(x) = \mu_D^C(x)$. If B is a distribution-consistent set of C and any proper subset of B is not a distribution-consistent set, then B is a distribution reduct of C;
(3) B is a maximum distribution-consistent set of C if $\forall x \in U, \gamma_D^B(x) = \gamma_D^C(x)$. If B is a maximum distribution-consistent set of C and any proper subset of B is not a maximum distribution-consistent set, then$B$ is a maximum distribution reduct of C.

Definition 3.1 shows that an assignment reduct is an independent subset of condition attributes that maintain unchanged the possible decision for an arbitrary object of U on full condition attributes; a distribution reduct B is an independent subset of condition attributes that maintain unchanged the distribution of rough membership values of all objects of U on full condition attributes; and a maximum distribution reduct B is an independent subset of condition attributes that maintain unchanged the maximum decision classes of all objects of U on full condition attributes.

From Definitions 2.3 and 3.1, we can obtain the inner links (shown in Fig. 1) among positive region reduct, assignment reduct, distribution reduct, and maximum distribution reduct. That is, a distribution reduct includes a maximum reduct and an assignment reduct; an assignment reduct, a distribution reduct, and a maximum distribution reduct include a relative position region reduct.

As mentioned above, it is quite inconvenient to judge and compute the assignment reduct, the distribution reduct, and the maximum distribution reduct directly according to their definitions. In Sections 4–6, we will present three quick heuristic algorithms that compute these three types of reducts.

4. Quick assignment reduction

In this section, we present a new algorithm for quickly finding an assignment reduct, which is called the Q-ARA (Quick Assignment Reduction Algorithm). We start with the concept of an assignment approximation set (Section 4.1). Then, we propose the judgment theorem of the assignment reduct and the attribute assignment significance, followed by the pseudo-code description of Q-ARA (Section 4.2).

4.1. Assignment approximation

The core of the rough set theory lies in lower and upper approximations. This approach means that, for any object subset $X \subseteq U$ and attributes $P \subseteq C$, we can use a lower approximation set $\underline{P}(X)$ and an upper approximation set $\overline{P}(X)$ to approximate X. In this section, we will introduce a new concept to approximate$X$: the assignment approximation set.

Theorem 4.1. *Let $DT = (U, C \cup D, V, f)$ be a decision table and $P \subseteq B \subseteq C$; then, $\forall x \in U, \delta_D^B(x) \subseteq \delta_D^P(x)$.*

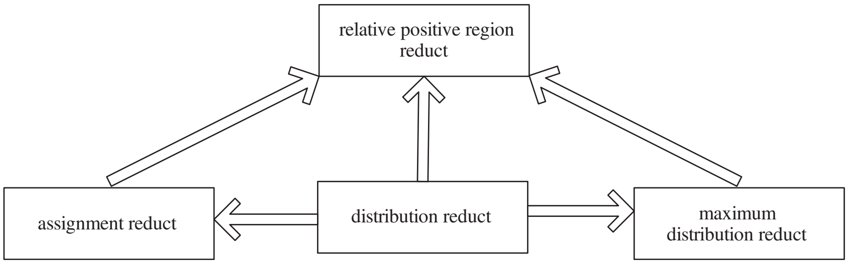

Fig. 1. Inner links among four types of reducts.

Proof. It is straightforward. □

Definition 4.2. Let $DT = (U, C \cup D, V, f)$ be a decision table, with $B \subseteq C$, and $U/D = \{X_1, X_2, \ldots, X_S\}$. The assignment approximation set of $X_j (1 \leqslant j \leqslant S)$ based on the attribute set$B$ is defined as:

$$\widetilde{B}^{As}(X_j) = \left\{ \frac{1/|\delta_D^B(x)|}{x} |x \in X_j \right\}. \tag{8}$$

Definition 4.2 shows that the assignment approximation set of X_j by the attributes set B is a fuzzy set of X_j, and the membership of each object x in this fuzzy set uses the reciprocal of the cardinality of the partition $[x]_B/D$.

For two attribute sets $P \subseteq C$ and $B \subseteq C$, we denote $\widetilde{P}^{As}(X_j) \subseteq \widetilde{B}^{As}(X_j)$, meaning $1/|\delta_D^P(x)| \leqslant 1/|\delta_D^B(x)|$ for any $x \in X_j$.

Theorem 4.3. *Let $DT = (U, C \cup D, V, f)$ be a decision table, $P \subseteq B \subseteq C$, and $U/D = \{X_1, X_2, \ldots, X_S\}$; then, $\forall X_j (1 \leqslant j \leqslant S), \widetilde{P}^{As}(X_j) \subseteq \widetilde{B}^{As}(X_j)$.*

Proof. Let X_j be an arbitrary equivalence class of partition U/D. For any $x \in X_j$, because $P \subseteq B$, according to Theorem 4.1, $\delta_D^B(x) \subseteq \delta_D^P(x)$ holds. Hence, $|\delta_D^B(x)| \leqslant |\delta_D^P(x)|$ is true. Therefore, $1/|\delta_D^P(x)| \leqslant 1/|\delta_D^B(x)|$ holds. Hence, the theorem is proven. □

Definition 4.4. Let $DT = (U, C \cup D, V, f)$ be a decision table, with $B \subseteq C$, and $U/D = \{X_1, X_2, \ldots, X_S\}$. The assignment approximation quality of $X_j \quad (1 \leqslant j \leqslant S)$ by the attribute set B is defined as:

$$r_B^{As}(X_j) = \frac{|\widetilde{B}^{As}(X_j)|}{|X_j|} = \frac{\sum_{x \in X_j} 1/|\delta_D^B(x)|}{|X_j|}. \tag{9}$$

Definition 4.4 shows that the assignment approximation quality of X_j based on the attribute set B is the sum of all of the membership values of those objects that are in the assignment approximation set $\widetilde{B}^{As}(X_j)$ divided by the total number of objects in X_j. Formula (9) is equivalent to Formula (10):

$$r_B^{As}(X_j) = \frac{\sum_{Y \in U/B} |Y \cap X_j| / |Y/D|}{|X_j|}, \tag{10}$$

where $|Y/D|$ denotes the number of equivalence classes in the partition Y/D. Obviously, $0 < r_B^{As}(X_j) \leqslant 1$. If $r_B^{As}(X_j) = 1$, then $\delta_D^B(x) = \{[x]_D\}$ holds for any $x \in X_j$, and X_j is definable by the attribute set B; otherwise, X_j is not definable by the attribute set B; Compared with Formula (9), Formula (10) is a direct and quick calculation method.

Theorem 4.5. *Let $DT = (U, C \cup D, V, f)$ be a decision table, $P \subseteq B \subseteq C$, and $U/D = \{X_1, X_2, \ldots, X_S\}$; then, $\forall X_j (1 \leqslant j \leqslant S)$, $r_B^{As}(X_j) \geqslant r_P^{As}(X_j)$.*

Proof. According to Formula (9) and Theorem 4.3, the conclusion is clearly established.

Theorem 4.5 shows that the assignment approximation quality increases monotonically with the increase in the condition attributes. □

4.2. Attribute assignment significance

Definition 4.6. Let $DT = (U, C \cup D, V, f)$ be a decision table, with $B \subseteq C$ and $U/D = \{X_1, X_2, \ldots, X_S\}$. The assignment positive region of U by the attribute set B is defined as:

$$POS_B^{As}(D) = \cup_{X_j \in U/D} \widetilde{B}^{As}(X_j). \tag{11}$$

Formula (11) can also be expressed as follows:

$$POS_B^{As}(D) = \left\{ \frac{1/|\delta_D^B(x)|}{x} |x \in U \right\}. \tag{12}$$

Definition 4.6 shows that the assignment positive region of U by the attribute set B is a fuzzy set of U, and the membership of each object x in this fuzzy set uses the reciprocal of the cardinality of the partition $[x]_B/D$.

Theorem 4.7. *Let $DT = (U, C \cup D, V, f)$ be a decision table and $P \subseteq B \subseteq C$; then, $POS_P^{As}(D) \subseteq POS_B^{As}(D)$.*

Proof. According to Definition 4.6 and Theorem 4.3, the conclusion is clearly established.

Theorem 4.7 shows the assignment positive region enlarges monotonically with the increase in the condition attributes. □

Definition 4.8. Let $DT = (U, C \cup D, V, f)$ be a decision table, with $B \subseteq C$ and $U/D = \{X_1, X_2, \ldots, X_S\}$. The assignment dependency degree of U based on the attribute set B is defined as:

$$\gamma_B^{As}(D) = \frac{|POS_B^{As}(D)|}{|U|} = \sum_{X_j \in U/D} \frac{|\widetilde{B}^{As}(X_j)|}{|U|}. \tag{13}$$

According to Formulas (14) and (15), the $\gamma_B^{As}(D)$ can also be expressed as:

$$\gamma_B^{As}(D) = \frac{|POS_B^{As}(D)|}{|U|} = \frac{\sum_{x \in U} 1/|\delta_D^B(x)|}{|U|}, \tag{14}$$

$$\gamma_B^{As}(D) = \frac{\sum_{Y \in U/B} |Y|/|Y/D|}{|U|}. \tag{15}$$

Formula (14) shows that the assignment dependency degree of U by the attribute set B is the sum of the reciprocal of the cardinality of the partition $[x]_B/D$ that is induced by each object of U, divided by the total number of objects in U. Obviously, $0 < \gamma_B^{As}(D) \leqslant 1$. If $\gamma_C^{As}(D) = 1$, then there must be $|\delta_D^C(x)| = 1$ for any $x \in U$, and DT is a consistent decision table; otherwise, $\exists x \in U$ such that $|\delta_D^C(x)| > 1$, and DT is an inconsistent decision table. Thus, $\gamma_B^{As}(D)$ reflects the integrative assignment consistent degree of all of the objects by the attribute set B.

Of Formulas (13)–(15), Formula (15) is a direct and quick calculation method. The reason is that we can easily form all of the equivalence classes of partition U/B and all of the equivalence classes of Y/D $(Y \in U/B)$ after sorting all of the objects of U in an ascending (or descending) order according to the attribute set B and $B \cup D$ respectively.

Theorem 4.9. *Let $DT = (U, C \cup D, V, f)$ be a decision table and $B \subseteq C$; then, $\gamma_B^{As}(D) \leqslant \gamma_C^{As}(D)$.*

Proof. According to Formula (13) and Theorem 4.7, the conclusion is clearly established. □

Theorem 4.10. *Let $DT = (U, C \cup D, V, f)$ be a decision table and $B \subseteq C$; then, B is an assignment-consistent set of C if and only if $\gamma_B^{As}(D) = \gamma_C^{As}(D)$.*

Proof. ⇒ BecauseB is an assignment-consistent set of C, $\delta_D^B(x) = \delta_D^C(x)$ holds for any $x \in U$. According to Formula (14), $\gamma_B^{As}(D) = \gamma_C^{As}(D)$ obviously holds.

⇐ If $\gamma_B^{As}(D) = \gamma_C^{As}(D)$, then, according to Formula (14), $\frac{1}{|U|}\sum_{x \in U} 1/|\delta_D^B(x)| = \frac{1}{|U|}\sum_{x \in U} 1/|\delta_D^C(x)|$ holds. Hence, $\sum_{x \in U} 1/|\delta_D^B(x)| = \sum_{x \in U} 1/|\delta_D^C(x)|$ holds. Furthermore, according to Theorem 4.1, there must be $\delta_D^B(x) = \delta_D^C(x)$ for any $x \in U$ (otherwise, $\sum_{x \in U} 1/|\delta_D^B(x)| < \sum_{x \in U} 1/|\delta_D^C(x)|$). Hence,B is an assignment-consistent set of C.

Theorem 4.10 provides a simple approach to judging whether a subset of the condition attributes is an assignment-consistent set. □

Theorem 4.11. *Let $DT = (U, C \cup D, V, f)$ be a decision table and $B \subseteq C$; then B is an assignment reduct of C if and only if: (1) $\gamma_B^{As}(D) = \gamma_C^{As}(D)$; and (2) $\gamma_{B'}^{As}(D) \neq \gamma_C^{As}(D), \forall B' \subset B$.*

Proof. It is straightforward. □

Theorem 4.12. *Let $DT = (U, C \cup D, V, f)$ be a decision table, with $B \subseteq C$ and $U/D = \{X_1, X_2, \ldots, X_S\}$. If B is an assignment consistent set of C, then $\underline{B}(X_j) = \underline{C}(X_j)$ and $\overline{B}(X_j) = \overline{C}(X_j), \forall X_j (1 \leqslant j \leqslant S)$.*

Proof. Let X be an arbitrary equivalence class of partition U/D.

For an arbitrary object $x \in U$, if $x \in \underline{C}(X_j)$, then $\delta_D^C(x) = \{X_j\}$. Because $\delta_D^B(x) = \delta_D^C(x)$ (because B is an assignment-consistent set of C), it follows that $\delta_D^B(x) = \{X_j\}$. Hence, $x \in \underline{B}(X_j)$ holds. Vice versa, we can prove: if $x \in \underline{B}(X_j)$, then $x \in \underline{C}(X_j)$ holds, for any $x \in U$. That is, $\underline{B}(X_j) = \underline{C}(X_j)$.

For an arbitrary object $x \in U$, if $x \in \overline{C}(X_j)$, then $X_j \in \delta_D^C(x)$. Because $\delta_D^B(x) = \delta_D^C(x)$ (because B is an assignment-consistent set of C), it follows that $X_j \in \delta_D^B(x)$. Hence, $x \in \overline{B}(X_j)$ holds. Vice versa, we can prove: if $x \in \overline{B}(X_j)$, then $x \in \overline{C}(X_j)$ holds, for any $x \in U$. That is, $\overline{B}(X_j) = \overline{C}(X_j)$.

This completes the proof. □

Theorem 4.12 implies that, for an inconsistent decision table, an assignment reduct B produces identical lower and upper approximations as the entire condition attribute set C concerning decision attributes D, which is quite different from a relative positive region reduct that only preserves original positive regions (only all lower approximations).

The assignment reducts have stricter requirements than the relative positive region reducts. However, for the widely used two-class classification decision tables ($|U/D| = 2$), we have Theorem 4.13.

Theorem 4.13. *Let $DT = (U, C \cup D, V, f)$ be a two-class classification decision table and $B \subseteq C$. If B is a non-empty relative positive region reduct of C, then B is also an assignment reduct of C.*

Proof. Let $U/D = \{X_1, X_2\}$.

If B is a non-empty relative positive region reduct of$C$, then we have $\underline{B}(X_1) = \underline{C}(X_1)$ and $\underline{B}(X_2) = \underline{C}(X_2)$. In addition, $\overline{C}(X_1) = U - \underline{C}(X_2)$ and $\overline{B}(X_1) = U - \underline{B}(X_2), \overline{C}(X_2) = U - \underline{C}(X_1)$ and $\overline{B}(X_2) = U - \underline{B}(X_1)$. Then clearly we have $\overline{B}(X_1) = \overline{C}(X_1)$, and $\overline{B}(X_2) = \overline{C}(X_2)$ holds. Therefore, $BN_B(X_1) = BN_C(X_1)$ and $BN_B(X_2) = BN_C(X_2)$ holds (with boundary regions unchanged).

Moreover, $U = \underline{B}(X_1) \cup \underline{B}(X_2) \cup BN_B(X_1)$ obviously holds. Then $\forall x \in U$, there can be only three cases.

Case 1: if $x \in \underline{B}(X_1)$, then $x \in \underline{C}(X_1)$; hence, $\delta_D^B(x) = \delta_D^C(x) = \{X_1\}$;
Case 2: if $x \in \underline{B}(X_2)$, then $x \in \underline{C}(X_2)$; hence, $\delta_D^B(x) = \delta_D^C(x) = \{X_2\}$;
Case 3: if $x \in BN_B(X_1)$, then $x \in BN_C(X_1)$; hence, $\delta_D^B(x) = \delta_D^C(x) = \{X_1, X_2\}$.

From the above statements, we have the conclusion: $\delta_D^B(x) = \delta_D^C(x)$ holds for any $x \in U$. That is, B is also an assignment reduct of C.

This completes the proof. □

Theorem 4.13 implies that for a two-class classification decision table, a non-empty relative positive region reduct is also an assignment reduct. Obviously, for a multi-class classification decision table ($|U/D| > 2$), the relative positive region reduct and the assignment reduct are not always equivalent.

It is worth noting that a singularity occurs in some decision tables, keeping the positive region unchanged without making any requirements for the condition attributes. These cases occur when the positive region of the universe U is the empty set with respect to the whole condition attribute set. In such extreme cases, the empty attribute set is the relative position region reduct, which will lead us to know nothing. However, for any decision tables, the empty attribute set will not be an assignment reduct. In this respect, the assignment reduct has more robustness than the relative positive region reduct does.

Definition 4.14. Let $DT = (U, C \cup D, V, f)$ be a decision table and $B \subseteq C$. The assignment significance of attribute a in attribute set B is defined as:

$$SIG^{As}(a, B, D) = \gamma_B^{As}(D) - \gamma_{B-\{a\}}^{As}(D). \tag{16}$$

This definition computes the increment of assignment discernibility power by introducing attribute a.

Based on the assignment significance of attribute a, a heuristic greedy attribute reduction algorithm is designed as Algorithm 2. Algorithm 2 starts with an empty attribute set and iteratively selects an attribute to add in, with assignment positive dependency being the largest until its maximum possible value is produced for the decision table. The pseudo-code is as follows:

Algorithm 2. Quick Assignment Reduction Algorithm (Q-ARA) for finding the assignment reduct.

```
Input: Decision table $DT = (U, C \cup D, V, f)$.
Output: One assignment reduct of $DT$.
$red \leftarrow \{\}$
$do$
  $T \leftarrow red$
  $flag \leftarrow true$
  $foreach\ a \in C - red$
    $if\ \ \gamma_{red \cup \{a\}}^{As}(D) > \gamma_T^{As}(D)$
    $T \leftarrow red \cup \{a\}$
  $if\ \ T \neq red$
  $red \leftarrow T$
     $flag \leftarrow false$
$until\ flag = true$
$return\ red$
```

If there are k condition attributes, then the time complexity for searching an assignment reduct would need to perform $(k^2 + k)/2$ evaluations of the assignment positive dependency function for the worst-case data set. This time complexity is as the same as that of the QuickReduct algorithm.

5. Quick distribution reduction

In this section, we present a new algorithm for quickly finding the distribution reduct, which is called Q-DRA (Quick Distribution Reduction Algorithm). We start with the concept of the distribution approximation set (Section 5.1). Then, we propose the judgment theorem of the distribution reduct and the attribute distribution significance, followed by the pseudo-code description of Q-DRA (Section 5.2).

5.1. Distribution approximation

In this section, we will introduce a new concept that is used to approximate any subset $X \subseteq U$: the distribution approximation set.

Definition 5.1. Let $DT = (U, C \cup D, V, f)$ be a decision table, with $B \subseteq C \cup D$, and $X \subseteq U(X \neq \phi)$. The distribution approximation set of X based on the attribute set B is defined as:

$$\widetilde{B}^d(X) = \left\{ \frac{\mu_X^B(x)}{x} | x \in X \right\}. \tag{17}$$

Definition 5.1 shows that the distribution approximation set of X based on the attribute set B is a probabilistic fuzzy set of X, and the membership of each object x in this fuzzy set uses its rough membership value in X, given that the object is in the equivalence class $[x]_B$.

Definition 5.2. Let $DT = (U, C \cup D, V, f)$ be a decision table, with $B \subseteq C$, and $X \subseteq U(X \neq \phi)$. The distribution approximation quality of X by the attribute set B is defined as:

$$r_B^d(X) = \frac{|\widetilde{B}^d(X)|}{|X|} = \frac{\sum_{x \in X} \mu_X^B(x)}{|X|}. \tag{18}$$

Definition 5.2 shows that the distribution approximation quality of X based on the attribute set B is the sum of all of the rough membership values of those objects in the approximation set $\widetilde{B}^d(X)$ divided by the total number of objects in X. Formula (18) is equivalent to Formulas (19) and (20).

$$r_B^d(X) = \frac{|\underline{B}(X)| + \sum_{x \in X - \underline{B}(X)} \mu_X^B(x)}{|X|}, \tag{19}$$

$$r_B^d(X) = \frac{1}{|X|} \sum_{Y \in U/B} \frac{|Y \cap X|^2}{|Y|}. \tag{20}$$

From Formula (19), we can see that the distribution approximation quality of X depends in the rough membership of those objects which possibly as well as definitely belong to the X. Thus, it is possible to compare the discernibility power of two condition attribute sets $P$and B in the case of $\underline{P}(X) = B(X)$. Among the Formulas (18)–(20), Formula (20) is a direct and quick calculation method.

Theorem 5.3. *Let $DT = (U, C \cup D, V, f)$ be a decision table and $P \subseteq B \subseteq C$; then, $\forall X \subseteq U, r_B^d(X) \geqslant r_P^d(X)$.*

Proof. According to Formula (20), $r_P^d(X) = \frac{1}{|X|} \sum_{Y \in U/P} \frac{|X \cap Y|^2}{|Y|}$. Because $P \subseteq B$, any equivalence class $Y \in U/P$ would be divided into smaller equivalence classes. Thus the $r_B^d(X)$ can be expressed as $r_B^d(X) = \frac{1}{|X|} \sum_{Y \in U/P} \sum_{Y_i \in Y/B} \frac{|X \cap Y_i|^2}{|Y_i|}$. Therefore, it is necessary only to prove that $\sum_{Y_i \in Y/B} \frac{|X \cap Y_i|^2}{|Y_i|} \geqslant \frac{|X \cap Y|^2}{|Y|}$ holds for any $Y \in U/P$.

For any equivalence class $Y \in U/P$, let $Y/B = \{Y_1, Y_2, \ldots, Y_m\}$, and let $b = |X \cap Y|, b_1 = |X \cap Y_1|, b_2 = |X \cap Y_2|, \ldots$, and $b_m = |X \cap Y_m|$; obviously, we have $b = b_1 + b_2 + \cdots + b_m$. Furthermore, let $a = |Y|, a_1 = |Y_1|, a_2 = |Y_2|, \ldots$, and $a_m = |Y_m|$; then, we have $a = a_1 + a_2 + \cdots + a_m$. According to Theorem A.1 (details in the appendix), $\frac{b_1^2}{a_1} + \frac{b_2^2}{a_2} + \cdots + \frac{b_m^2}{a_m} \geqslant \frac{(b_1 + b_2 + \cdots + b_m)^2}{a_1 + a_2 + \cdots + a_m}$ certainty holds. That is, $\sum_{1 \leqslant i \leqslant m} \frac{|X \cap Y_i|^2}{|Y_i|} \geqslant \frac{|X \cap Y|^2}{|Y|}$ holds. This completes the proof.

Theorem 5.3 shows that the distribution approximation quality increases monotonically along with the increase in the condition attributes. Notably, with the premise of $P \subseteq B \subseteq C$, $\widetilde{P}^d(X) \subseteq \widetilde{B}^d(X)$ (meaning that $\mu_X^P(x) \leqslant \mu_X^B(x)$ holds for any $x \in X$) does not necessarily hold. □

Theorem 5.4. *Let* $DT = (U, C \cup D, V, f)$ *be a decision table and* $B \subseteq C$ *; then,* $\forall Y \in U/B, \sum_{X \in U/D} \frac{|Y \cap X|^2}{|Y|} \leqslant \quad \sum_{Y_i \in Y/C} \sum_{X \in U/D} \frac{|Y_i \cap X|^2}{|Y_i|}$.

Proof. The proof is the same as the proof for Theorem 5.3; thus, it is omitted here. □

Definition 5.5. Let $DT = (U, C \cup D, V, f)$ be a decision table. The positive rough membership value of object $x \in U$ by the attribute set C is defined as:

$$\mu_{pos}^{C}(x) = p([x]_D | [x]_C) = \frac{|[x]_{C \cup D}|}{|[x]_C|}. \tag{21}$$

Definition 5.5 means that the positive rough membership value of x by the attribute set C is the ratio of the number of objects that have the same decision values as x among the equivalence class $[x]_C$ to the total number of objects in $[x]_C$. It is notable that, for the two condition attribute sets B and $P, P \subseteq B$, and the inequality$\mu_{pos}^{P}(x) \leqslant \mu_{pos}^{B}(x)$ might not be set up, as shown in Example 1.

5.2. Attribute distribution significance

Definition 5.6. Let $DT = (U, C \cup D, V, f)$ be a decision table and $B \subseteq C$. The distribution positive region of U based on attributes B is denoted by $POS_B^d(D)$ and is defined as:

$$POS_B^d(D) = \underset{X \in U/D}{\cup} \widetilde{B}^d(X). \tag{22}$$

Formula (22) can also be expressed as:

$$POS_B^d(D) = \left\{ \frac{\mu_{pos}^{B}(x)}{x} | x \in U \right\}. \tag{23}$$

Definition 5.6 shows that the distribution positive region of U by the attribute set B is a fuzzy set of U, and the membership of each object x in this fuzzy set uses its positive rough membership value by the attribute set B.

In addition, for any two condition attribute set P and B, $POS_P^d(D) \subseteq POS_B^d(D)$means that $\mu_{pos}^{P}(x) \leqslant \mu_{pos}^{B}(x)$ holds for any $x \in U$. $POS_P^d(D) = POS_B^d(D)$ means that $\mu_{pos}^{P}(x) = \mu_{pos}^{B}(x)$ holds for any $x \in U$.

Definition 5.7. Let $DT = (U, C \cup D, V, f)$ be a decision table and $B \subseteq C$. The distribution positive dependency degree of U based on the attribute set B is defined as:

$$\gamma_B^d(D) = \frac{|POS_B^d(D)|}{|U|} = \frac{\sum_{x \in U} \mu_{pos}^{B}(x)}{|U|}. \tag{24}$$

The $\gamma_B^d(D)$ can also be expressed as Formulas (25)–(27):

$$\gamma_B^d(D) = \sum_{X \in U/D} \frac{|\widetilde{B}^d(X)|}{|U|}, \tag{25}$$

$$\gamma_B^d(D) = \sum_{X \in U/D} \frac{|X|}{|U|} r_B^d(X), \tag{26}$$

$$\gamma_B^d(D) = \frac{1}{|U|} \sum_{Y \in U/B} \sum_{X \in U/D} \frac{|Y \cap X|^2}{|Y|}. \tag{27}$$

Formula (24) shows that the distribution positive dependency degree of U is the sum of the positive rough membership value of all of the objects in U divided by the total number of objects in U. Obviously, $0 < \gamma_B^d(D) \leqslant 1$; if $\gamma_B^d(D) = 1$, then there must be $\mu_{pos}^{C}(x) = 1$ for any $x \in U$, andDT is a consistent decision table; otherwise, $\exists x \in U$ such that $\mu_{pos}^{C}(x) < 1$, and DT is an inconsistent decision table. Therefore, $\gamma_B^d(D)$ reflects the integrative distribution consistent degree of all of the objects by the attribute set B. The smaller the value of $\gamma_B^d(D)$ is, the more conflicts there will be in the decision table.

The distribution positive dependency degree $\gamma_B^d(D)$ computes the discernibility power of attribute set B with respect to decision attributes D. Comparatively, the positive region dependency degree only embodies the discernibility power to the positive region of universe U, whereas the distribution positive dependency degree can reflect the discernibility power to all of the objects in universe U. Among the Formulas (24)–(27), Formula (27) is a direct and quick calculation method because we can easily form all of the equivalence classes of partition U/B and all of the equivalence classes of $Y \cap X$ ($Y \in U/B$ and $X \in U/D$) after sorting all of the objects of U in an ascending (or descending) order according to the attribute set B and $B \cup D$ respectively.

Theorem 5.8. *Let $DT = (U, C \cup D, V, f)$ be a decision table and $P \subseteq B \subseteq C$; then, $\gamma^d_B(D) \geqslant \gamma^d_P(D)$.*

Proof. According to Formula (26) and Theorem 5.3, the conclusion is clearly established. □

Theorem 5.9. *Let $DT = (U, C \cup D, V, f)$ be a decision table and $B \subseteq C$. B is a distribution-consistent set if and only if $POS^d_B(D) = POS^d_C(D)$.*

Proof. $\Rightarrow$ To prove that $POS^d_B(D) = POS^d_C(D)$ is established is equivalent to proving that $\mu^B_{pos}(x) = \mu^C_{pos}(x)$ holds for any $x \in U$.

According to the definition of a distribution-consistent set, we have $\mu^B_D(x) = \mu^C_D(x)$ for any $x \in U$. Hence, $\mu^B_{pos}(x) = \mu^C_{pos}(x)$ holds for any $x \in U$.

$\Leftarrow$ To prove that B is a distribution-consistent set, it is necessary only to prove that $\mu^B_D(x) = \mu^C_D(x)$ for any $x \in U$.

Let $U/D = \{X_1, X_2, \ldots, X_S\}$ be the partition of U that is induced by the decision attribute set D; then the distribution of rough membership for an arbitrary $x \in U$ by attributes C relative to each equivalence class of partition U/D is $\mu^C_D(x) = \left[\frac{|X_1 \cap [x]_C|}{|[x]_C|}, \frac{|X_2 \cap [x]_C|}{|[x]_C|}, \cdots, \frac{|X_S \cap [x]_C|}{|[x]_C|}\right]$. Suppose that $x \in X_i (1 \leqslant i \leqslant S)$; then, $\frac{|X_i \cap [x]_B|}{|[x]_B|} = \frac{|X_i \cap [x]_C|}{|[x]_C|}$ (owing to the premise of $POS^d_B(D) = POS^d_C(D)$) holds. Next, we will prove that $\frac{|X_j \cap [x]_B|}{|[x]_B|} = \frac{|X_j \cap [x]_C|}{|[x]_C|}$ also holds, $\forall X_j \in U/D \quad (j \neq i)$. There can be only three cases.

Case 1: If $X_j \cap [x]_C \neq \phi$, then $\forall y \in X_j \cap [x]_C, \frac{|X_j \cap [y]_B|}{|[y]_B|} = \frac{|X_j \cap [y]_C|}{|[y]_C|}$ (owing to the premise of $POS^d_B(D) = POS^d_C(D)$) holds. Furthermore, because there are $[y]_B = [x]_B$ and $[y]_C = [x]_C (y \in [x]_C$ and $B \subseteq C), \frac{|X_j \cap [x]_B|}{|[x]_B|} = \frac{|X_j \cap [x]_C|}{|[x]_C|}$ is established.

Case 2: If $X_j \cap [x]_C = \phi$ and $X_j \cap [x]_B = \phi$, then $|X_j \cap [x]_C| = 0$ and $|X_j \cap [x]_B| = 0$, and the equality $\frac{|X_j \cap [x]_B|}{|[x]_B|} = \frac{|X_j \cap [x]_C|}{|[x]_C|}$ is also established.

Case 3: If $X_j \cap [x]_C = \phi$ and $X_j \cap [x]_B \neq \phi$, then X_j has an intersection with $[x]_B$ but not with $[x]_C$. However, this construct is not possible. We now proceed to a proof by contradiction. Let $[x]_C/D = \{Y_1, Y_2, \cdots, Y_m\}$, where $Y_1 \subseteq X_1, Y_2 \subseteq X_2, \cdots, Y_m \subseteq X_m$; and let $[x]_B/D = \left\{Y'_1, Y'_2, \cdots, Y'_m, Y'_j\right\}$, where $Y_1 \subseteq Y'_1, Y_2 \subseteq Y'_2, \cdots, Y_m \subseteq Y'_m, Y'_j \subseteq X_j$ (without loss of generality, X_j is assumed to be the only decision class that has an intersection with $[x]_B$ but not with $[x]_C$); then, we have $\sum_{1 \leqslant l \leqslant m} \frac{|Y_l|}{|[x]_C|} = 1$ and $\sum_{1 \leqslant l \leqslant m} \frac{|Y'_l|}{|[x]_B|} + \frac{|Y'_j|}{|[x]_B|} = 1$, which implies $\sum_{1 \leqslant l \leqslant m} \frac{|Y_l|}{|[x]_C|} \neq \sum_{1 \leqslant l \leqslant m} \frac{|Y'_l|}{|[x]_C|}$. As a result, $\exists Y_l \in [x]_C/D$ such that $\frac{|Y_l|}{|[x]_C|} \neq \frac{|Y'_l|}{|[x]_B|}$, and therefore there exists $y \in Y_l \subset [x]_C$ such that $\mu^B_{pos}(y) \neq \mu^C_{pos}(y)$. This result is contradictory with the premise of $POS^d_B(D) = POS^d_C(D)$.

From the above statements, we can conclude that, on the premise of $POS^d_B(D) = POS^d_C(D), \frac{|X \cap [x]_B|}{|[x]_B|} = \frac{|X \cap [x]_C|}{|[x]_C|}$ holds, $\forall X \in U/D, \forall x \in U$, i.e., $\mu^B_D(x) = \mu^C_D(x), \forall x \in U$.

This completes the proof. □

Theorem 5.9 simplifies judging whether a subset of condition attributes is a distribution-consistent set.

Theorem 5.10. *Let $DT = (U, C \cup D, V, f)$ be a decision table and $B \subseteq C; B$ is a distribution reduct if and only if: (1) $POS^d_B(D) = POS^d_C(D)$; and (2) $POS^d_{B\prime}(D) \neq POS^d_C(D), \forall B' \subset B$.*

Proof. It is straightforward. □

In searching for a relative region reduct, the QuickReduct algorithm uses $\gamma_B(D) = \gamma_C(D)$ as the stopping criteria for a forward search process. This procedure means that, if $\gamma_B(D) = \gamma_C(D)$ (or $|POS_B(D)| = |POS_C(D)|$), then B is a positive region reduct. In other words, $\gamma_B(D) = \gamma_C(D)$ is equivalent to $POS_B(D) = POS_C(D)$. Obviously, this is established. The reason is that $POS_B(D) \subseteq POS_C(D)$ for any subset $B \subseteq C$, so $|POS_B(D)| \leqslant |POS_C(D)|$ is true; and therefore, $POS_B(D) = POS_C(D)$ holds if and only if $|POS_B(D)| = |POS_C(D)|$ holds. However, for an inconsistent decision table $DT = (U, C \cup D, V, f)$ and $B \subseteq C$, we do not always have that $POS^d_B(D) \subseteq POS^d_C(D)$ (meaning $\mu^B_{pos}(x) \leqslant \mu^C_{pos}(x), \forall x \in U$) holds under the premise of $\gamma^d_B(D) \leqslant \gamma^d_C(D)$ (or $|POS^d_B(D)| \leqslant |POS^d_C(D)|$). This result is exemplified as follows:

Example 1. Let $Y = \{x_1, x_2, x_3, x_4, x_5\}$ be one of the equivalence classes of the partition $U/B, Y/C = \{\{x_1, x_2, x_3\}, \{x_4, x_5\}\}$ $(B \subset C)$, and the decision values of these five objects are $D(x_1) = D(x_2) = d_0$ and $D(x_3) = D(x_4) = D(x_5) = d_1$ (d_0 and d_1 represent two decision values); then, we have $POS^d_C(D) = \left\{\frac{2/3}{x_1}, \frac{2/3}{x_2}, \frac{1/3}{x_3}, \frac{1/1}{x_4}, \frac{1/1}{x_5}\right\}$ and $POS^d_B(D) = \left\{\frac{2/5}{x_1}, \frac{2/5}{x_2}, \frac{3/5}{x_3}, \frac{3/5}{x_4}, \frac{3/5}{x_5}\right\}$. We note that $\mu^{B}_{pos}(x) \leqslant \mu^{C}_{pos}(x)$ for $x \in \{x_1, x_2, x_4, x_5\}$ but $\mu^{B}_{pos}(x_3) \geqslant \mu^{C}_{pos}(x_3)$. Thus arises an important question: whether $POS^{d}_{B}(D) = POS^{d}_{C}(D)$ actually holds under the premise of $\gamma^{d}_{B}(D) = \gamma^{d}_{C}(D)$ (or $|POS^{d}_{B}(D)| = POS^{d}_{C}(D)|$). For this question, we have Theorem 5.11:

Theorem 5.11. *Let $DT = (U, C \cup D, V, f)$ be a decision table and $B \subseteq C$; then, $POS^{d}_{B}(D) = POS^{d}_{C}(D)$ if and only if $\gamma^{d}_{B}(D) = \gamma^{d}_{C}(D)$.*

Proof. $\Rightarrow$ If $POS^{d}_{B}(D) = POS^{d}_{C}(D)$, then according to Formula (24), $\gamma^{d}_{B}(D) = \gamma^{d}_{C}(D)$ obviously holds.

$\Leftarrow$ To prove that $POS^{d}_{B}(D) = POS^{d}_{C}(D)$ is established, it is necessary only to prove that $\mu^{B}_{pos}(x) = \mu^{C}_{pos}(x)$ holds for any $x \in U$. According to Formula (27), $\gamma^{d}_{B}(D) = \frac{1}{|U|}\sum_{Y \in U/B}\sum_{X \in U/D}\frac{|Y \cap X|^2}{|Y|}$. Because $B \subseteq C$, any equivalence class $Y (Y \in U/B)$ would be divided into smaller equivalence classes by the attribute set C. Thus, $\gamma^{d}_{C}(D)$ can be expressed as $\gamma^{d}_{C}(D) = \frac{1}{|U|}\sum_{Y \in U/B}\sum_{Y_i \in Y/C}\sum_{X \in U/D}\frac{|X \cap Y_i|^2}{|Y_i|}$. In addition, according to Theorem 5.4, we have $\sum_{X \in U/D}\frac{|Y \cap X|^2}{|Y|} \leqslant \sum_{Y_i \in Y/C}\sum_{X \in U/D}\frac{|Y_i \cap X|^2}{|Y_i|}$ for any $Y \in U/B$. Thus $\gamma^{d}_{B}(D) = \gamma^{d}_{C}(D)$ holds if and only if $\sum_{X \in U/D}\frac{|Y \cap X|^2}{|Y|} = \sum_{Y_i \in Y/C}\sum_{X \in U/D}\frac{|Y_i \cap X|^2}{|Y_i|}$ holds for any $Y \in U/B$. Next, we will prove that $\mu^{B}_{pos}(x) = \mu^{C}_{pos}(x)$ holds, $\forall x \in Y_i, \forall Y_i \in Y/C, \forall Y \in U/B$.

According to Theorem 5.3, $\sum_{Y_i \in Y/C}\frac{|X \cap Y_i|^2}{|Y_i|} \geqslant \frac{|X \cap Y|^2}{|Y|}$ holds for any $X \in U/D$. Therefore, $\sum_{X \in U/D}\frac{|Y \cap X|^2}{|Y|} = \sum_{Y_i \in Y/C}\sum_{X \in U/D}\frac{|Y_i \cap X|^2}{|Y_i|}$ holds if and only if $\sum_{Y_i \in Y/C}\frac{|X \cap Y_i|^2}{|Y_i|} = \frac{|X \cap Y|^2}{|Y|}$ holds for any $X \in U/D$. For an arbitrary equivalence class $Y \in U/B$ and an arbitrary set $X \in U/D$, let $Y/C = \{Y_1, Y_2, \ldots, Y_m\}, b = |X \cap Y|, b_1 = |X \cap Y_1|, b_2 = |X \cap Y_2|, \ldots$, and $b_m = |X \cap Y_m|$. Obviously, we have $b = b_1 + b_2 + \cdots + b_m$. Furthermore, let $a = |Y|, a_1 = |Y_1|, a_2 = |Y_2|, \ldots$, and $a_m = |Y_m|$; then we have $a = a_1 + a_2 + \cdots + a_m$. According to Theorem A.1 (details in the Appendix), we have $\frac{b_1^2}{a_1} + \frac{b_2^2}{a_2} + \cdots + \frac{b_m^2}{a_m} \geqslant \frac{(b_1 + b_2 + \cdots + b_m)^2}{a_1 + a_2 + \cdots + a_m}$, and the equality holds if and only if $\frac{b_1}{a_1} = \frac{b_2}{a_2} = \cdots = \frac{b_m}{a_m}$ holds. That is, $\frac{|X \cap Y_i|}{|Y_i|} = \frac{|X \cap Y|}{|Y|}$ holds, $\forall X \in U/D, \forall Y_i \in Y/C, \forall Y \in U/B$. In addition, for an arbitrary object $x \in Y_i$ $(Y_i \in Y/C)$, there exists $X \in U/D$ such that $x \in X$; then we have $\mu^{C}_{pos}(x) = \mu^{C}_{X}(x) = \frac{|X \cap Y_i|}{|Y_i|}$ and $\mu^{B}_{pos}(x) = \mu^{B}_{X}(x) = \frac{|X \cap Y|}{|Y|}$. Therefore, $\mu^{B}_{pos}(x) = \mu^{C}_{pos}(x)$ holds, $\forall x \in Y_i, \forall Y_i \in Y/C, \forall Y \in U/B$. That is, $\mu^{B}_{pos}(x) = \mu^{C}_{pos}(x)$ holds $\forall x \in U$.

This completes the proof. □

Theorem 5.11 means that $\gamma^{d}_{B}(D) = \gamma^{d}_{C}(D)$ is equivalent to $POS^{d}_{B}(D) = POS^{d}_{C}(D)$. Therefore, we can use Theorem 5.11 to judge whether a subset of condition attributes is a distribution-consistent set or not. This result greatly simplifies the problem.

Definition 5.12. Let $DT = (U, C \cup D, V, f)$ be a decision table and $B \subseteq C$. The distribution significance of attribute a in attribute set B is defined as:

$$SIG^{d}(a, B, D) = \gamma^{d}_{B}(D) - \gamma^{d}_{B-\{a\}}(D). \tag{28}$$

This definition computes the increment of the distribution discernibility power by introducing the attribute a.

Based on the distribution significance of the attribute a, a heuristic greedy attribute reduction algorithm is designed as Algorithm 3. Algorithm 3 starts with an empty attribute set and iteratively selects an attribute to add in, with a distribution positive dependency being the largest until its maximum possible value is produced for the decision table. The pseudo code is as follows:

Algorithm 3. Quick Distribution Reduction Algorithm (Q-DRA) for finding the distribution reduct.

```
Input: Decision table $DT = (U, C \cup D, V, f)$.
Output: One distribution reduct of $DT$.
$red \leftarrow \{\}$
do
  $T \leftarrow red$
  $flag \leftarrow true$
  foreach $a \in C - red$
    if $\gamma^{d}_{red \cup \{a\}}(D) > \gamma^{d}_{T}(D)$
      $T \leftarrow red \cup \{a\}$
  if $T \neq red$
    $red \leftarrow T$
     $flag \leftarrow false$
until $flag = true$
return $red$
```

If there are k condition attributes, then the time complexity for searching a distribution reduct would need to perform $(k^2 + k)/2$ evaluations of the distribution positive dependency function for the worst-case data set. This time complexity is as the same as that of the QuickReduct algorithm.

6. Quick maximum distribution reduction

In this section, we present a new algorithm for quickly finding the maximum distribution reduct, called Q-MDRA (Quick Maximum Distribution Reduction Algorithm). We start with the concept of the maximum distribution approximation set (Section 6.1). Then we propose the judgment theorem of the maximum distribution reduct and the attribute maximum distribution significance, followed by the pseudo-code description of the Q-MDRA (Section 6.2).

6.1. Maximum distribution approximation

In this section, we will introduce a new concept that is used to approximate a decision equivalence class $X \in U/D$: the maximum distribution approximation set.

Definition 6.1. Let $DT = (U, C \cup D, V, f)$ be a decision table and $B \subseteq C$. We say that $x \in U$ is a compatible maximum decision distribution object by the attribute set B if $\gamma_D^B(x) = [x]_D$.

Definition 6.1 shows that those objects $y \in [x]_B$ are compatible maximum decision distribution objects if the number is the largest of the objects that belong to $[y]_B \cap [y]_D$ (obviously, $[y]_B = [x]_B$). In other words, those compatible maximum decision distribution objects in $[x]_B$ are an equivalence class of the partition $[x]_B/D$ with the largest number of objects. For example, let $Y = \{x_1, x_2, x_3, x_4, x_5\}$ be one of the equivalence classes of the partition U/B, and $Y/D = \{\{x_1, x_2, x_3\}, \{x_4, x_5\}\}$. Since $|\{x_1, x_2, x_3\}| > |\{x_4, x_5\}|, x_1, x_2, x_3$ are compatible maximum decision distribution objects by the attribute set B.

Definition 6.2. Let $DT = (U, C \cup D, V, f)$ be a decision table, with $B \subseteq C$, and $U/D = \{X_1, X_2, \ldots, X_S\}$; then the compatible maximum distribution approximation set of $X_j (1 \leqslant j \leqslant S)$ by the attribute set B is defined as follows:

$$\widetilde{B}^{Md}(X_j) = \{x | x \in X_j \wedge \gamma_D^B(x) = X_j\}. \tag{29}$$

Definition 6.2 shows that the compatible maximum distribution approximation set of X_j by the attribute set B are those compatible maximum decision distribution objects inX_j by the attribute set B. Note that $\widetilde{B}^{Md}(X_j)$ could be an empty set.

It is notable that under the premise of $B \subseteq C$, either $\widetilde{B}^{Md}(X_j) \subseteq \widetilde{C}^{Md}(X_j)$ or $|\widetilde{B}^{Md}(X_j)| \leqslant |\widetilde{C}^{Md}(X_j)|$ $(1 \leqslant j \leqslant S)$ does not necessarily hold.

Theorem 6.3. *Let $DT = (U, C \cup D, V, f)$ be a decision table and $P \subseteq B \subseteq C$; then, $\forall Y \in U/P, |\widetilde{P}^{Md}(Y)| \leqslant \sum_{Y_i \in Y/B} |\widetilde{B}^{Md}(Y_i)|$.*

Proof. Because $P \subseteq B$, any equivalence class $Y \in U/P$ could be divided into smaller equivalence classes by the attribute set B. Let $Y/B = \{Y_1, Y_2, \ldots, Y_m\}, U/D = \{X_1, X_2, \ldots, X_S\}, X_k = \arg\max_{1 \leqslant i \leqslant S} P(X_i|Y), h = |Y \cap X_k|, h_1 = |Y_1 \cap X_k|, \ldots$, and $h_m = |Y_m \cap X_k|$; then obviously $h = h_1 + \cdots + h_m$. Let $Y_1^{Md} = \{Y_1 \cap X_j | X_j = \arg\max_{1 \leqslant i \leqslant S} P(X_i|Y_1)\}\}$; and let more formal definitions of $Y_2^{Md}, \cdots$, and Y_m^{Md} be like Y_1^{Md}, and further let $l_1 = |Y_1^{Md}|, l_2 = |Y_2^{Md}|, \cdots, l_m = |Y_m^{Md}|$. Obviously, we have $l_1 \geqslant h_1, \cdots, l_m \geqslant h_m$. Hence, $l_1 + l_2 + \cdots + l_m \geqslant h$. That is, $|\widetilde{P}^{Md}(Y)| \leqslant \sum_{Y_i \in X/B} |\widetilde{B}^{Md}(Y_i)|$

Theorem 6.3 means that the number of compatible maximum distribution objects for any equivalence class $Y \in U/P$ can increase after Y is subdivided. □

Theorem 6.4. *Let $DT = (U, C \cup D, V, f)$ be a decision table, $B \subseteq C$, and $\forall Y \in U/B$. If $|\widetilde{B}^{Md}(Y)| = \sum_{Y_i \in Y/C} |\widetilde{C}^{Md}(Y_i)|$, then $\widetilde{B}^{Md}(Y) = \cup_{Y_i \in X/B} \widetilde{C}^{Md}(Y_i)$.*

Proof. Let $U/D = \{X_1, X_2, \ldots, X_S\}, Y/C = \{Y_1, Y_2, \ldots, Y_m\}, X_k = \arg\max_{1 \leqslant i \leqslant S} P(X_i|Y)$, and $h = |Y \cap X_k|$. Suppose that there is one equivalence class Y_j $(1 \leqslant j \leqslant m)$ that makes $X_r = \arg\max_{1 \leqslant i \leqslant S} P(X_i|Y_j)$ and $X_r \neq X_k$; then we have $|Y_j \cap X_r| \geqslant |Y_j \cap X_k|$. Obviously, this arrangement will cause $\sum_{Y_i \in Y/C} |\widetilde{C}^M(Y_i)| > |\widetilde{B}^M(Y)|$, which is in contradiction with the premise. Consequently, there is no equivalence class Y_j $(1 \leqslant j \leqslant m)$ such that $X_k \neq \arg\max_{1 \leqslant i \leqslant S} P(X_i|Y_j)$.

This completes the proof. □

Definition 6.5. Let $DT = (U, C \cup D, V, f)$ be a decision table, with $B \subseteq C$, and $U/D = \{X_1, X_2, \ldots, X_S\}$; then the compatible maximum distribution approximation quality of $X_j (1 \leqslant j \leqslant S)$ by the attribute set B is defined as:

$$r_B^{Md}(X_j) = \frac{|\widetilde{B}^{Md}(X_j)|}{|X_j|}. \tag{30}$$

Formula (30) is equivalent to Formula (31):

$$r_B^{Md}(X_j) = \sum_{Y \in U/B \wedge X_j = \arg\max_{1 \leqslant i \leqslant S} P(X_i|Y)} \frac{|X_j \cap Y|}{|X_j|}. \tag{31}$$

Definition 6.5 shows that the compatible maximum distribution approximation quality of X_j based on the attribute set B is the ratio of the number of objects that are included in the compatible maximum distribution approximation set $\widetilde{B}^{Md}(X_j)$ to the total number of all of the objects in X_j. Formula (31) makes a direct and quick calculation method.

6.2. Attribute maximum distribution significance

Definition 6.6. Let $DT = (U, C \cup D, V, f)$ be a decision table, with $B \subseteq C$, and $U/D = \{X_1, X_2, \ldots, X_S\}$; then the maximum distribution positive region of U by the attribute set B is defined as:

$$POS_B^{Md}(D) = \bigcup_{X_j \in U/D} \widetilde{B}^{Md}(X_j). \tag{32}$$

Formula (32) can also be expressed as follows:

$$POS_B^{Md}(D) = \bigcup_{Y \in U/B} \{Y \cap X_j | X_j = \arg\max_{1 \leqslant i \leqslant S} P(X_i|Y)\}. \tag{33}$$

Definition 6.6 shows that the maximum distribution positive region of U by the attribute set B is those compatible maximum decision distribution objects in U.

It is notable that, if $B \subseteq C$, the $POS_B^{Md}(D) \subseteq POS_C^{Md}(D)$ does not certainly hold, although $|POS_B^{Md}(D)| \leqslant |POS_C^{Md}(D)|$ certainty holds (according to Theorem 6.8).

Definition 6.7. Let $DT = (U, C \cup D, V, f)$ be a decision table, with $B \subseteq C$, and $U/D = \{X_1, X_2, \ldots, X_S\}$; then the maximum distribution positive dependency degree of U that is based on the attribute set B is defined as:

$$\gamma_B^{Md}(D) = \frac{|POS_B^{Md}(D)|}{|U|} = \frac{|\cup_{X_j \in U/D} \widetilde{B}^{Md}(X_j)|}{|U|}. \tag{34}$$

The value of $\gamma_B^{Md}(D)$ can also be expressed according to Formulas (35) and (36):

$$\gamma_B^{Md}(D) = \sum_{X_j \in U/D} \frac{|X|}{|U|} r_B^{Md}(X_j), \tag{35}$$

$$\gamma_B^{Md}(D) = \frac{\sum_{Y \in U/B} |Y \cap X_j : X_j = \arg\max_{1 \leqslant i \leqslant S} P(X_i|Y)|}{|U|}. \tag{36}$$

Among the Formulas (34)–(36), Formula (36) is a direct and quick calculation method because we can easily form all of the equivalence classes of partition U/B and all of the equivalence classes of $Y \cap X$ ($Y \in U/B$ and $X \in U/D$) after sorting all of the objects of Uin an ascending (or descending) order according to the attribute sets B and $B \cup D$ respectively.

Theorem 6.8. *Let $DT = (U, C \cup D, V, f)$ be a decision table and $P \subseteq B \subseteq C$; then $\gamma_B^{Md}(D) \geqslant \gamma_P^{Md}(D)$.*

Proof. According to Formula (36) and Theorem 6.3, the conclusion is clearly established. □

Theorem 6.9. *Let $DT = (U, C \cup D, V, f)$ be a decision table and $B \subseteq C$; then $POS_B^{Md}(D) = POS_C^{Md}(D)$ if and only if $\gamma_B^{Md}(D) = \gamma_C^{Md}(D)$.*

Proof. $\Rightarrow$ If $POS_B^{Md}(D) = POS_C^{Md}(D)$, then according to Formula (34), $\gamma_B^{Md}(D) = \gamma_C^{Md}(D)$ obviously holds.

$\Leftarrow$ That $\gamma_B^{Md}(D) = \gamma_C^{Md}(D)$ means $|POS_B^{Md}(D)| = |POS_C^{Md}(D)|$. According to Theorem 6.3, we have $|\widetilde{B}^M(Y)| = \sum_{Y_i \in Y/C} |\widetilde{C}^M(Y_i)|$ for any $Y \in U/B$ (otherwise, $|POS_B^{Md}(D)| < |POS_C^{Md}(D)|$). Then, according to Theorem 6.4 and Formula (33), $POS_B^{Md}(D) = POS_C^{Md}(D)$ clearly holds. □

Theorem 6.10. *Let $DT = (U, C \cup D, V, f)$ be a decision table and $B \subseteq C.B$ is a maximum distribution-consistent set of C if and only if $\gamma_B^{Md}(D) = \gamma_C^{Md}(D)$.*

Proof. $\Rightarrow$ To prove that $\gamma_B^{Md}(D) = \gamma_C^{Md}(D)$ is established, it is necessary only to prove that $|POS_B^{Md}(D)| = |POS_C^{Md}(D)|$.

For an arbitrary object $x \in U$, if $x \in POS_C^{Md}(D)$ (x is a compatible maximum decision distribution object by the attribute set C), then we have $\gamma_D^C(x) = [x]_D$. Because $\gamma_D^B(y) = \gamma_D^C(y)$ holds for any $y \in U$, it follows that $\gamma_D^B(x) = [x]_D$. This means that $x \in POS_B^{Md}(D)$ (x is also a compatible maximum decision distribution object by the attribute set B). Vice versa, we can prove if $x \in POS_B^{Md}(D)$, then $x \in POS_C^{Md}(D)$, for any $x \in U$. Hence, we have $POS_B^{Md}(D) = POS_C^{Md}(D)$, which ensures that $|POS_B^{Md}(D)| = |POS_C^{Md}(D)|$ certainly holds.

$\Leftarrow$ To prove that B is a maximum distribution consistent set of C, it is necessary only to prove that $\gamma_D^B(x) = \gamma_D^C(x)$ for any $x \in U$.

According to Theorem 6.9, $\gamma_B^{Md}(D) = \gamma_C^{Md}(D)$ means that $POS_B^{Md}(D) = POS_C^{Md}(D)$ holds. For an arbitrary object $x \in U$, there can only be two cases.

Case 1: if $x \in POS_C^{Md}(D)$ (x is a compatible maximum decision distribution object by the attribute set C), then $\gamma_D^C(x) = [x]_D$ holds. This means that $\gamma_D^B(x) = [x]_D$ (otherwise,$POS_B^{Md}(D) \neq POS_C^{Md}(D)$). Therefore, $\gamma_D^B(x) = \gamma_D^C(x)$ holds;

Case 2: if $x \notin POS_C^{Md}(D)$ (x is not a compatible maximum decision distribution object by attribute set C), then there must exist $X_i \in U/D$ ($X_i \neq [x]_D$) such that $\gamma_D^C(x) = X_i$. Furthermore, there must exist $y \in [x]_C$ such that $\gamma_D^C(y) = [y]_D = X_i$ (y is a compatible maximum decision distribution object by the attribute set C). Because $\gamma_D^B(y) = \gamma_D^C(y)$ (otherwise, $POS_B^{Md}(D) \neq POS_C^{Md}(D)$), it follows that $\gamma_D^B(y) = X_i$. Because $[x]_B = [y]_B$ (because $y \in [x]_C$ and $B \subseteq C$), then we have $\gamma_D^B(x) = X_i$. Therefore, $\gamma_D^B(x) = \gamma_D^C(x)$ holds.

Summarizing (1) and (2), we have the conclusion: $\gamma_D^B(x) = \gamma_D^C(x)$ holds for any $x \in U$.

This completes the proof. □

Theorem 6.10 greatly simplifies the judgment of whether or not a subset of condition attributes is a maximum distribution-consistent set.

Theorem 6.11. *Let $DT = (U, C \cup D, V, f)$ be a decision table and $B \subseteq C$.B is a maximum distribution reduct if and only if: (1) $\gamma_B^{Md}(D) = \gamma_C^{Md}(D)$; and (2) $\gamma_{B'}^{Md}(D) \neq \gamma_C^{Md}(D), \forall B' \subset B$.*

Proof. It is straightforward. □

Definition 6.12. Let $DT = (U, C \cup D, V, f)$ be a decision table and $B \subseteq C$. The maximum distribution significance of attribute a in attribute set B is defined as:

$$SIG^{Md}(a, B, D) = \gamma_B^{Md}(D) - \gamma_{B-\{a\}}^{Md}(D). \tag{37}$$

This definition computes the increment of the maximum distribution discernibility power by introducing the attribute a.

Based on the maximum distribution significance of attribute a, a heuristic greedy attribute reduction algorithm is designed as Algorithm 4. Algorithm 4 starts with an empty attribute set and iteratively selects an attribute to add, with the maximum distribution positive dependency being the largest until its maximum possible value is produced for the decision table. The pseudo code is as follows:

Algorithm 4. Quick Maximum Distribution Reduction Algorithm (Q-MDRA) for finding the maximum distribution reduct.

```
Input: Decision table $DT = (U, C \cup D, V, f)$.
Output: One maximum distribution reduct of $DT$.
$red \leftarrow \{\}$
do
  $T \leftarrow red$
  $flag \leftarrow true$
  foreach $a \in C - red$
    if $\gamma_{red \cup \{a\}}^{Md}(D) > \gamma_T^{Md}(D)$
      $T \leftarrow red \cup \{a\}$
  if $T \neq red$
    $red \leftarrow T$
    $flag \leftarrow false$
until $flag = true$
return $red$
```

If there are k condition attributes, then the time complexity for searching a maximum distribution reduct might need to perform $(k^2 + k)/2$ evaluations of the distribution positive dependency function for the worst-case data set. This time complexity is as the same as that of the QuickReduct algorithm.

7. Experimental results and analyses

To demonstrate the effectiveness of our proposed algorithms, we conduct experiments on the 12 UCI data sets outlined in Table 1. We then run all of the experiments on a PC equipped with a Windows 7 operating system, Intel core 2 quad 2.4 GHz CPU, and 2G memory. We compare our algorithms with the other three leading reduction algorithms from 7.1 to 7.3.

First, we have conducted a series of experiments to compare our three algorithms with QuickReduct because all of these algorithms adopt the same forward greedy algorithm framework. In these experiments, we have compared the execution time and the cardinalities of the reducts. In addition, we have compared the attribute sequences of reducts to discover the difference between them. Furthermore, we have compared the effect of four types of reducts and their prefixes, as determined by four algorithms, on the performance of C4.5 [31] and RBF-SVM (support vector machine with radial basis function kernel) [3] classifiers.

Second, we have compared the Quick Distribution Reduction Algorithm with the CEBARKNC [43] algorithm because both are highly relevant in using forward greedy search strategy to search a distribution reduct. In these experiments, a comparison focus is on the execution time, the cardinalities of the reducts, the attribute sequences of the reducts, and the effect of the classification performance on the C4.5 and RBF-SVM classifiers.

Last, we have compared our three algorithms with the corresponding discernibility matrix-based algorithms. In these experiments, the primary comparison is on the execution time of the algorithms.

To be fair, the same quick sorting technique is adopted in QuickReduct, in CEBARKNC, and in our three algorithms. All of the reduction algorithms involved in the experiments can address only discrete attributes; as a result, we employ here the CACM [14] discretization algorithm to transform the continuous data into discrete data. For each discretized data set in Table 1, the positive region dependency degree $\gamma_C(D)$ is listed in the second column of Table 2. As we know, when $\gamma_C(D) = 1$, the discretized data set is consistent; otherwise, it is inconsistent. Thus, Wpbc, Wine, and Sonar are consistent. Here, we take Sat, Segment, Wdbc, and Wave as consistent data sets because the value of $\gamma_C(D)$ for these data sets is very close to 1. The other 5 data sets (Vehicle, Ion, Glass, Heart, and Pid) are inconsistent.

7.1. Comparison with QuickReduct

Our three algorithms are all identical to the QuickReduct except for the attribute significance measures that are used. In fact, the ideas of our three proposed attribute significance measures originate from the positive region dependency that is used in QuickReduct. QuickReduct and our three algorithms search, respectively, the positive region reduct, the assignment reduct, the distribution reduct, and the maximum distribution reduct. In this section, we will describe the performance comparison of these four algorithms.

In Table 2, columns 3–6 present the numbers of attributes with four types of reduct that were found by QuickReduct and our three algorithms respectively, and columns 7–10 present the corresponding execution time.

It is found that in general, the number of attributes in the maximum distribution reduct (found by Q-MDRA) is equal to or less than the number of attributes in the assignment reduct (found by Q-ARA) and the distribution reduct (found by Q-DRA). The average value of the number of attributes in distribution reduct (found by Q-DRA) is the largest (12.5), and the average value of the number of attributes in the maximum distribution reduct is the smallest (11.2). With respect to the data sets of Wdbc, Heart, Pid, Wave, and Sat, no attributes are selected by the QuickReduct Algorithm because all of the single attributes get zero for the relative positive dependency degree, and no attribute can be selected in the first loop. However, our algorithms show strong robustness because they can find a reduct for every data set. Of four algorithms (QuickReduct, Q-ARA, Q-DRA, and Q-MDRA), the execution time for Q-MDRA is the least because of its simple calculation and the relatively small cardinalities of the reducts found by Q-MDRA.

Table 3 presents the attribute sequences with four types of reducts (found by QuickReduct, Q-ARA, Q-DRA, and Q-MDRA) for a specific data set. The integers listed in Table 3 represent the corresponding attributes in each specific data set. That is, 1

Table 1
Description of data sets (C: Continuous, N: Nominal).

	Data sets	Abbreviation	Samples	Attributes		Classes
1	Wisconsin prognostic breast cancer	Wpbc	198	34		2
2	Wine recognition	Wine	178	13C		3
3	Statlog project satellite image	Sat	6435	36C		6
4	Image Segmentation	Segment	2310	19C		7
5	Wisconsin diagnostic breast cancer	Wdbc	569	30C		2
6	Waveform	Wave	5000	21C		3
7	Vehicle Silhouettes	Vehicle	846	18C		4
8	Jhons Hopkins University Ionosphere	Ions	351	34C		2
9	Glass identification	Glass	214	9C		7
10	Statlog Heart	Heart	270	6C	7N	2
11	Sonar, Mines vs.Rocks	Sonar	208	60C		2
12	Pima Indians Diabetes	Pid	768	8C		2

Table 2
Positive region dependency; numbers of attribute with four types of reduct, and execution time.

Data set (discretized)	$\gamma_C(D)$	Number of attributes in reduct				Execution time (s)			
		QuickReduct	Q-ARA	Q-DRA	Q-MDRA	QuickReduct	Q-ARA	Q-DRA	Q-MDRA
Wpbc	1.000	11	11	9	10	1.295	1.328	1.296	1.429
Wine	1.000	5	6	5	5	0.161	0.182	0.163	0.162
Sat	0.993	0	31	33	26	–	82.812	82.816	72.218
Segment	0.991	9	8	9	7	1.919	2.180	2.262	1.906
Wdbc	0.989	0	14	13	13	–	2.226	2.238	2.129
Wave	0.981	0	21	21	21	–	16.466	16.785	16.340
Vehicle	0.946	15	15	15	15	1.691	1.778	1.781	1.786
Ion	0.940	12	12	11	9	2.093	2.211	2.216	1.737
Glass	0.937	7	7	7	7	0.118	0.118	0.119	0.118
Heart	0.935	0	12	11	10	–	0.275	0.276	0.261
Sonar	1.000	8	8	8	9	0.312	0.312	0.313	0.316
Pid	0.519	0	1	8	2	–	0.103	0.269	0.141
Average	0.936	–	12.2	12.5	11.2	–	9.166	9.211	8.212

denotes the first attribute, 2 the second attribute, and so on. The comparison results show that the attribute sequences are different, in spite of the fact that the cardinalities of the reducts that were generated by the four algorithms are close or even equivalent. This result is within expectation because these four algorithms adopt different attribute significance measures. We also note that the relative positive region reducts (found by QuickReduct) have identical attribute sequences with the assignment reducts (found by Q-ARA) for the three data sets of Wpbc, Sonar, and Ion, which account for three-sevenths of the reducts that were found by the QuickReduct Algorithm. This observation indicates that both the assignment reduct and the relative reduct have some similarity. With respect to the data sets Pid and Segment, the maximum distribution reduct (found by Q-MDRA) is the prefix of the distribution reduct (found by Q-DRA). For the many data sets, the first few attributes are identical to each other in both the maximum distribution reduct and the distribution reduct. Overall, the maximum distribution reduct and the distribution reduct have some similarity.

We also notice that the relative positive region dependency degree for the discretized data set Pid is 0.519, which means that the Pid includes more conflicting objects. For this data set Pid, reducts found by these four algorithms (QuickReduct, Q-ARA, Q-DRA, and Q-MDRA) show significant differences. That is, the QuickReduct algorithm cannot find a relative positive region reduct (because all of the single attributes get zero for the relative positive dependency degree, and no attribute can be selected in the first loop); the Q-ARA finds an assignment reduct that includes only one attribute; the Q-DRA finds a distribution reduct that includes eight attributes; and the Q-MDRA finds a maximum distribution reduct that includes two attributes.

To evaluate the effect of generated reducts on the performance of the classification algorithm, we use the corresponding reduced data sets that are generated by these four algorithms (QuickReduct, Q-ARA, Q-DRA, and Q-MDRA) to train the C4.5 and RBF-SVM classifiers. The numbers reported in Tables 4, 5 are the average classification accuracy over the 10 cross-validation runs ± the standard deviation of the accuracy obtained in those runs. The comparison results show that for the C4.5 classifier, the maximum distribution reduct (found by Q-MDRA) obtains the highest classification accuracy 8 times (two times identical with other reducts); the distribution reduct (found by Q-DRA) obtains the highest classification accuracy 3 times (one time identical with other reducts); the assignment reduct (found by Q-ARA) obtains the highest classification accuracy 3 times (two times identical with other reducts); and the relative positive region reduct (found by QuickReduct) obtains the highest classification accuracy only 2 times (one time identical with other reducts). In comparison, for the RBF-SVM classifier, the distribution reduct (found by Q-DRA) obtains the highest classification accuracy 6 times (one time identical with other reducts); the assignment reduct (found by Q-ARA) obtains the highest classification accuracy 4 times (one time identical with other reducts); the maximum distribution reduct (found by Q-MDRA) obtains the highest classification accuracy 3 times (one time identical with other reducts); and the relative positive region reduct (found by QuickReduct) obtains the highest classification accuracy only 1 time.

In Tables 4,5, the numbers in round brackets rank four types of reducts for each data set separately, with the best performing reduct ranked 1, the second best ranked 2, and so on; in case of a tie, the average ranks are assigned. We utilize the mean classification accuracy ranks rather than the average classification accuracies because the latter are susceptible to outliers (they allow a classifier's excellent performance on one data set to compensate for overall bad performance, or the opposite, a total failure in one domain can prevail over the acceptable results on most others). The direct comparison of these four types of reducts can be obtained by checking the mean ranks in the last row in Tables 4,5. The mean rank value is defined as each algorithm's rank for a specific data set among the four algorithms, which is averaged over the 12 data sets.

From the mean rank values of the classification accuracy, we have the following four findings: (1) Q-MDRA obtains the best classification performance on the C4.5 classifier. It can be understood that Q-MDRA finds the maximum distribution reducts, which have a relatively smaller number of attributes, but have a strong classification power. Consequently, the maximum distribution reducts effectively reduce the phenomenon of overfitting for the C4.5 classifier; (2) Q-DRA obtains the best classification performance on the RBF-SVM classifier. This result can be explained in that Q-DRA finds the distribution

Table 3
Attribute sequence with four types of algorithms.

Data set (discretized)	Attribute sequence of reduct			
	QuickReduct	Q-ARA	Q-DRA	Q-MDRA
Wpbc	10, 23, 20, 18, 13, 6, 3, 9, 26, 1, 8	10, 23, 20, 18, 13, 6, 3, 9, 26, 1, 8	18, 1, 7, 21, 13, 3, 8, 33, 10	18, 5, 3, 21, 1, 7, 8, 10, 26, 6
Wine	3, 11, 10, 7, 1	12, 1, 11, 5, 2, 3	7, 10, 13, 5, 1	7, 10, 13, 2, 5
Sat	–	18, 17, 20, 12, 27, 3, 35, 25, 22, 24, 7, 13, 31, 11, 19, 16, 1, 34, 4, 23, 9, 36, 33, 8, 15, 28, 5, 2, 21, 6, 14, 29	18, 17, 24, 16, 2, 25, 34, 11, 27, 35, 10, 12, 15, 31, 32, 19, 4, 1, 3, 23, 6, 20, 7, 21, 5, 28, 33, 36, 13, 8, 9, 29, 14	18, 17, 24, 15, 9, 30, 7, 27, 35, 2, 12, 22, 23, 31, 3, 4, 19, 5, 16, 8, 21, 25, 34, 6, 20, 33
Segment	11, 20, 3, 2, 7, 10, 15, 8, 16	13, 20, 3, 2, 9, 7, 16, 8	12, 3, 20, 2, 9, 7, 13, 8, 16	12, 3, 20, 2, 9, 7, 13
Wdbc	–	3, 4, 7, 21, 29, 16, 10, 25, 12, 11, 24, 31, 19, 20	23, 30, 4, 24, 10, 16, 28, 29, 21, 20, 19, 27, 11	23, 30, 24, 9, 4, 10, 13, 21, 14, 15, 18, 20, 25
Wave	–	1, 15, 16, 11, 13, 5, 12, 17, 19, 2, 3, 20, 10, 9, 4, 18, 7, 14, 21, 8, 6	7, 11, 15, 12, 9, 5, 14, 19, 2, 10, 3, 20, 1, 18, 4, 17, 13, 16, 21, 8, 6	7, 11, 9, 15, 12, 10, 14, 19, 3, 20, 2, 1, 18, 4, 5, 17, 13, 16, 21, 8, 6
Vehicle	9, 11, 6, 19, 16, 17, 18, 4, 2, 14, 7, 5, 15, 3, 8	9, 7, 6, 5, 17, 16, 19, 11, 14, 2, 18, 4, 15, 3, 8	13, 7, 11, 16, 17, 19, 6, 4, 15, 2, 14, 18, 5, 3, 8	8, 7, 19, 11, 16, 17, 6, 14, 2, 18, 3, 15, 4, 5, 9
Ion	4, 34, 20, 18, 9, 13, 35, 14, 12, 5, 15, 22	4, 34, 20, 18, 9, 13, 35, 14, 12, 5, 15, 22	35, 6, 18, 9, 26, 22, 28, 13, 20, 12, 14	35, 6, 9, 16, 10, 22, 5, 18, 15
Glass	10, 7, 6, 2, 5, 3, 4	4, 2, 7, 6, 5, 3, 10	2, 4, 5, 3, 10, 8, 7	2, 4, 5, 6, 3, 10, 7
Heart	–	2, 5, 13, 4, 12, 14, 8, 3, 6, 9, 10, 11	14, 13, 4, 12, 2, 6, 10, 3, 8, 11, 9	14, 4, 13, 12, 5, 2, 6, 9, 3, 8
Sonar	55, 8, 41, 24, 27, 12, 43, 51	55, 8, 41, 24, 27, 12, 43, 51	11, 27, 55, 29, 17, 51, 26, 10	11, 16, 26, 30, 27, 55, 25, 17, 6
Pid	–	2	3, 7, 9, 6, 5, 8, 4, 2	3, 7

Table 4
Comparison of classification accuracy using C4.5 classifier with four types of reducts (bolded values indicating best results).

Data set (discretized)	QuickReduct	Q-ARA	Q-DRA	Q-MDRA
Wpbc	0.7905 ± 0.0742 (3.5)	0.7905 ± 0.0742 (3.5)	0.7911 ± 0.0808 (2)	**0.8082** ± 0.0753 (1)
Wine	0.9549 ± 0.0456 (4)	**0.9607** ± 0.0476 (1)	0.9586 ± 0.0426 (2)	0.9580 ± 0.0417 (3)
Sat	0 (4)	0.8692 ± 0.0122 (2)	0.8606 ± 0.0069 (3)	**0.8696** ± 0.0180 (1)
Segment	**0.9551** ± 0.0168 (1)	0.9533 ± 0.0167 (3.5)	0.9540 ± 0.0165 (2)	0.9533 ± 0.0132 (3.5)
Wdbc	0 (4)	0.9520 ± 0.0281 (3)	0.9555 ± 0.0283 (2)	**0.9596** ± 0.0246 (1)
Wave	0 (4)	0.7782 ± 0.0171 (2)	0.7782 ± 0.0213 (2)	0.7782 ± 0.0164 (2)
Vehicle	0.7033 ± 0.0415 (3.5)	0.7033 ± 0.0415 (3.5)	**0.7040** ± 0.0407 (1)	0.7037 ± 0.0392 (2)
Ion	0.9040 ± 0.0418 (2.5)	0.9040 ± 0.0418 (2.5)	0.8994 ± 0.0400 (4)	**0.9097** ± 0.0445 (1)
Glass	0.7235 ± 0.0897 (2)	0.7235 ± 0.0897 (2)	0.7054 ± 0.0818 (4)	0.7235 ± 0.0897 (2)
Heart	0 (4)	0.8030 ± 0.0688 (2.5)	0.8030 ± 0.0688 (2.5)	**0.8148** ± 0.0698 (1)
Sonar	0.7969 ± 0.0862 (2.5)	0.7969 ± 0.0862 (2.5)	0.7865 ± 0.0940 (4)	**0.8006** ± 0.0862 (1)
Pid	0 (4)	0.6783 ± 0.0267 (3)	**0.7748** ± 0.0284 (1)	0.7708 ± 0.0546 (2)
Mean rank	3.25	2.58	2.46	**1.71**

Table 5
Comparison of classification accuracy using RBF-SVM classifier with four types of reducts (bolded values indicating best results).

Data set (discretized)	QuickReduct	Q-ARA	Q-DRA	Q-MDRA
Wpbc	0.7867 ± 0.0521 (2.5)	0.7867 ± 0.0521 (2.5)	0.7628 ± 0.0489 (4)	**0.7971** ± 0.0529 (1)
Wine	0.9674 ± 0.0368 (3)	0.9590 ± 0.0447 (4)	**0.9883** ± 0.0229 (1)	0.9709 ± 0.0385 (2)
Sat	0 (4)	0.**9094** ± 0.0130 (1)	0.9092 ± 0.0068 (2)	0.9054 ± 0.0094 (3)
Segment	**0.9587** ± 0.0158 (1)	0.9565 ± 0.0148 (3)	0.9579 ± 0.0182 (2)	0.9563 ± 1.48 (4)
Wdbc	0 (4)	**0.9724** ± 0.0191 (1)	0.9668 ± 0.0220 (2)	0.9554 ± 0.0256 (3)
Wave	0 (4)	0.8328 ± 0.0188 (2)	0.8328 ± 0.0188 (2)	0.8328 ± 0.0188 (2)
Vehicle	0.7373 ± 0.0387 (3.5)	0.7373 ± 0.0387 (3.5)	**0.7386** ± 0.0398 (1)	0.7379 ± 0.03.95 (2)
Ion	0.9319 ± 0.0372 (2.5)	0.9319 ± 0.0372 (2.5)	0.**9419** ± 0.0386 (1)	0.9174 ± 0.0381 (4)
Glass	0.7327 ± 0.0770 (3)	0.7327 ± 0.0770 (3)	**0.7402** ± 0.0877 (1)	0.7327 ± 0.0770 (3)
Heart	0 (4)	**0.8556** ± 0.0612 (1)	0.8374 ± 0.0716 (3)	0.8415 ± 0.0746 (2)
Sonar	0.7952 ± 0.0893 (3.5)	0.7952 ± 0.0893 (3.5)	0.**8275** ± 0.0864 (1)	0.8163 ± 0.0837 (2)
Pid	0 (4)	0.6783 ± 0.0267 (3)	0.7565 ± 0.0296 (2)	**0.7708** ± 0.0546 (1)
Mean rank	3.25	2.5	**1.83**	2.42

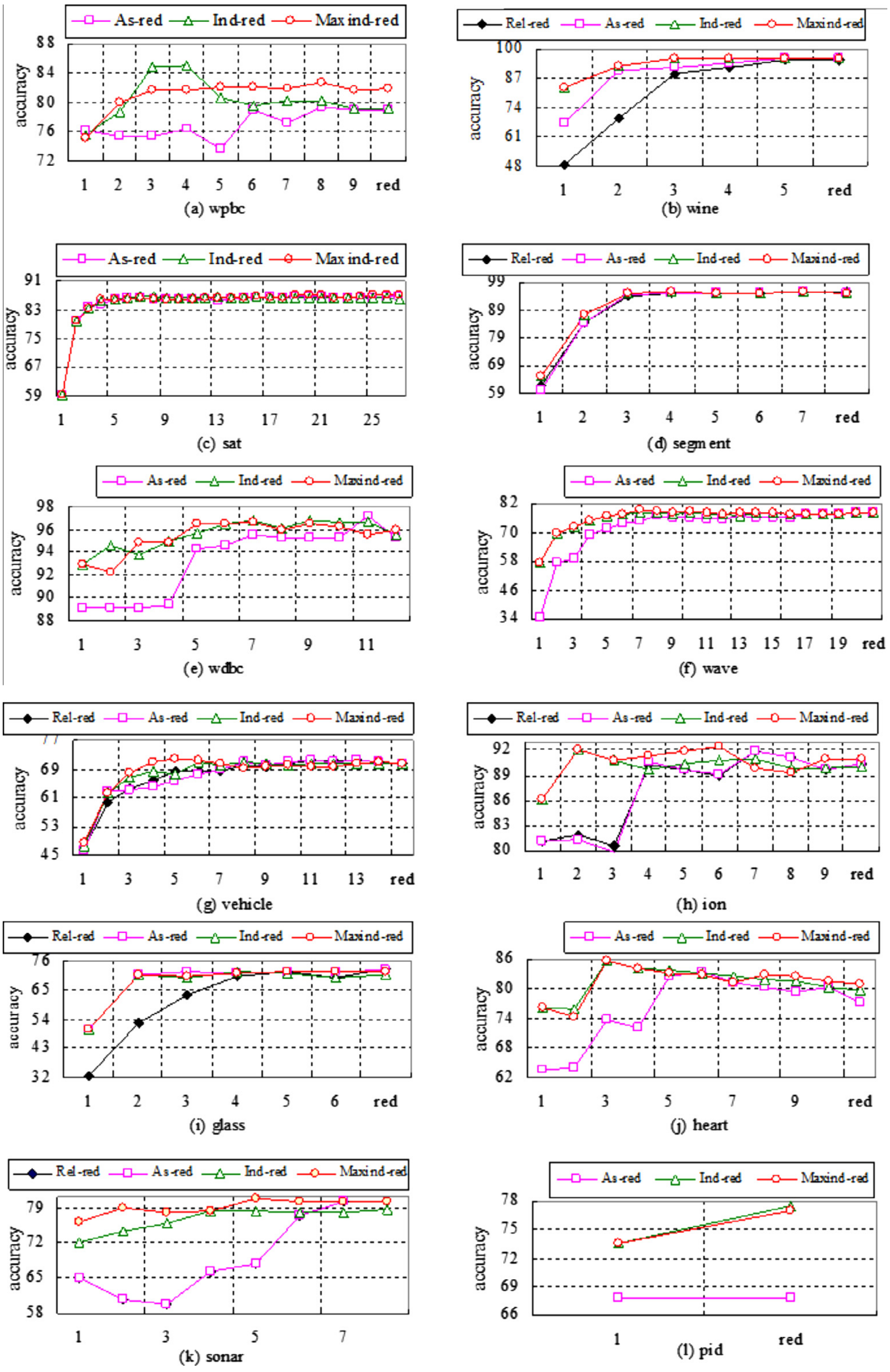

Fig. 2. Classification accuracy curves varying with attribute sequence of reduct using C4.5 classifier.

reducts, which are well suited to the RBF-SVM classifier because the distribution reducts have a relatively large number of attributes; (3) Q-ARA obtains a relatively poor level result for both the C4.5 and RBF-SVM classifiers; and (4) QuickReduct obtains the last mean ranks value.

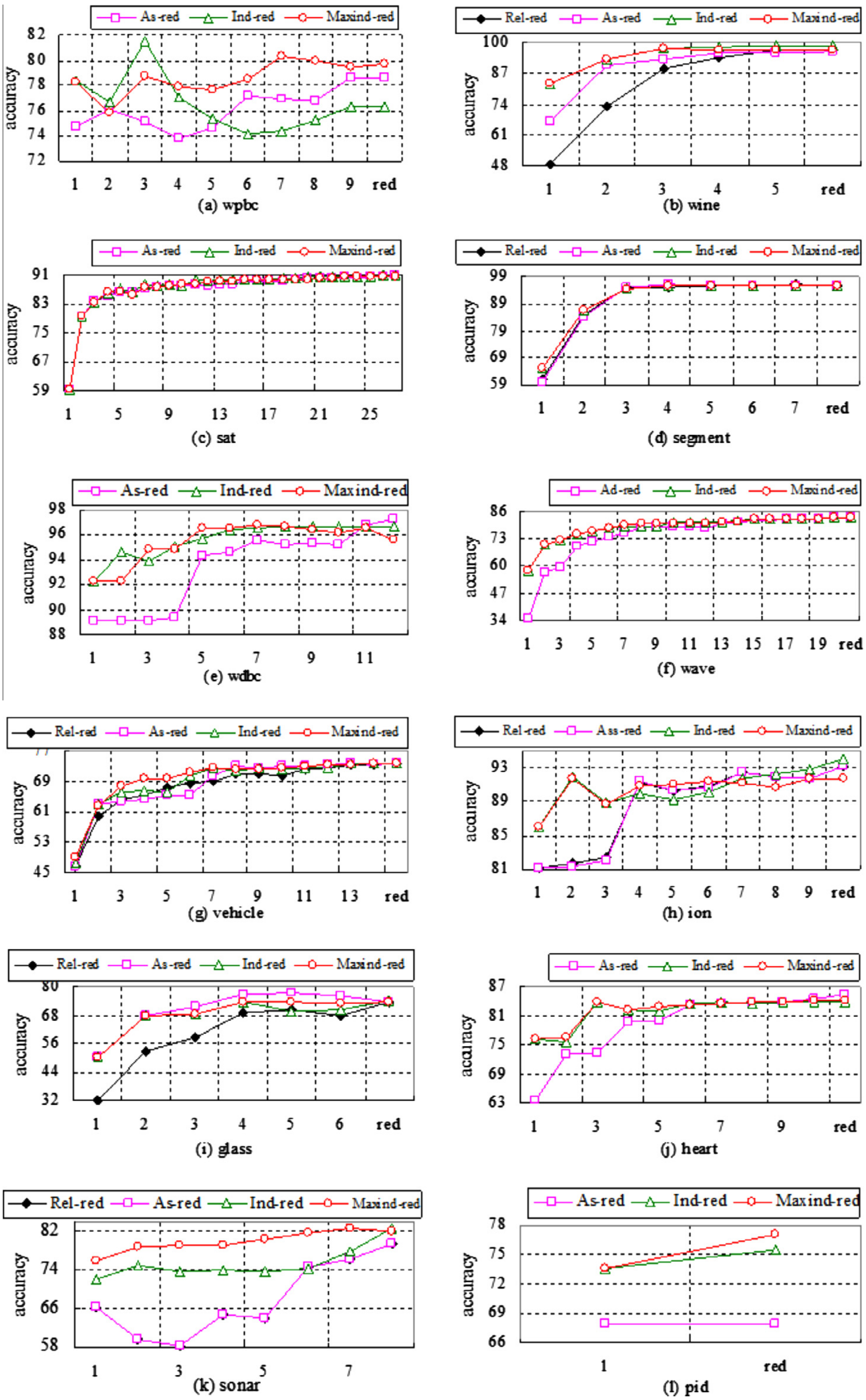

Fig. 3. Classification accuracy curves varying with attribute sequence of reduct using RBF-SVM classifier.

Perhaps some will argue that this comparison is not fair to the QuickReduct algorithm because the QuickReduct algorithm is derived for consistent decision tables. However, a check for 6 consistent data sets (the first 6 data sets in Tables 4 and 5) reveals that the QuickReduct algorithm cannot find the reduct for the 3 data sets of Wdbc, Wave, and Sat. In addition, for the

Table 6
Best classification accuracy results using C4.5 classifier with prefixes of four types of reducts (bolded values indicating best results).

Data set (discretized)	QuickReduct	Q-ARA	Q-DRA	Q-MDRA
Wpbc	0.7945 ± 0.0726 (3.5)	0.7945 ± 0.0726 (3.5)	**0.8502** ± 0.0690 (1)	0.8268 ± 0.0792 (2)
Wine	0.9549 ± 0.0456 (4)	0.9607 ± 0.0476 (3)	0.9620 ± 0.0421 (1.5)	0.9620 ± 0.0421 (1.5)
Sat	0 (4)	0.8692 ± 0.0122 (2.5)	0.8692 ± 0.0122 (2.5)	**0.8721** ± 0.0145 (1)
Segment	0.9555 ± 0.0173 (3)	0.9539 ± 0.0169 (4)	0.9557 ± 0.0162 (1.5)	0.9557 ± 0.0162 (1.5)
Wdbc	0 (4)	0.9587 ± 0.0277 (3)	0.9610 ± 0.0249 (2)	**0.9622** ± 0.0245 (1)
Wave	0 (4)	0.7782 ± 0.0171 (3)	0.7822 ± 0.0127 (2)	**0.7944** ± 0.0182 (1)
Vehicle	0.7133 ± 0.0424 (2.5)	0.7133 ± 0.0424 (2.5)	0.7106 ± 0.0453 (3)	**0.7180** ± 0.0356 (1)
Ion	0.9188 ± 0.0421 (3.5)	0.9188 ± 0.0421 (3.5)	0.9203 ± 0.0448 (2)	**0.9236** ± 0.0421 (1)
Glass	**0.7294** ± 0.0785 (1)	0.7235 ± 0.0897 (2)	0.7184 ± 0.0786 (3)	0.7218 ± 0.0904 (4)
Heart	0 (4)	0.8333 ± 0.0531 (3)	0.8556 ± 0.0679 (1.5)	0.8556 ± 0.0679 (1.5)
Sonar	0.8015 ± 0.0870 (2.5)	0.8015 ± 0.0870 (2.5)	0.7809 ± 0.0844 (4)	**0.8060** ± 0.0882 (1)
Pid	0 (4)	0.6783 ± 0.0267 (3)	**0.7748** ± 0.0284 (1)	0.7708 ± 0.0546 (2)
Mean rank	3.33	2.96	2.08	**1.54**

Table 7
Best classification accuracy results using RBF-SVM classifier with prefixes of four types of reducts (bolded values indicating best results).

Data set (discretized)	QuickReduct	Q-ARA	Q-DRA	Q-MDRA
Wpbc	0.7867 ± 0.0521 (3.5)	0.7867 ± 0.0521 (3.5)	**0.8153** ± 0.0580 (1)	0.8027 ± 0.0536 (2)
Wine	0.9674 ± 0.0368 (3)	0.9590 ± 0.0447 (4)	**0.9883** ± 0.0229 (1)	0.9721 ± 0.0371 (2)
Sat	0 (4)	**0.9094** ± 0.0130 (1)	0.9092 ± 0.0068 (2)	0.9063 ± 0.0099 (3)
Segment	**0.9622** ± 0.0149 (1)	0.9593 ± 0.0143 (2)	0.9579 ± 0.0182 (3)	0.9572 ± 0.0147 (4)
Wdbc	0 (4)	**0.9724** ± 0.0191 (1)	0.9673 ± 0.0250 (2.5)	0.9673 ± 0.0250 (2.5)
Wave	0 (4)	0.8328 ± 0.0188 (2)	0.8328 ± 0.0188 (2)	0.8328 ± 0.0188 (2)
Vehicle	0.7373 ± 0.0387 (3.5)	0.7373 ± 0.0387 (3.5)	**0.7386** ± 0.0398 (1)	0.7379 ± 0.0395 (2)
Ion	0.9319 ± 0.0372 (2.5)	0.9319 ± 0.0372 (2.5)	**0.9419** ± 0.0386 (1)	0.9174 ± 0.0381 (4)
Glass	0.7327 ± 0.0770 (4)	**0.7734** ± 0.0819 (1)	0.7402 ± 0.0877 (2)	0.7354 ± 0.0794 (3)
Heart	0 (4)	**0.8518** ± 0.0612 (1)	0.8396 ± 0.0699 (3)	0.8415 ± 0.0746 (2)
Sonar	0.7952 ± 0.0893 (3.5)	0.7952 ± 0.0893 (3.5)	**0.8275** ± 0.0809 (1)	0.8223 ± 0.0837 (2)
Pid	0 (4)	0.6783 ± 0.0267 (3)	0.7565 ± 0.0296 (2)	**0.7708** ± 0.0546 (1)
Mean rank	3.42	2.33	**1.79**	2.49

Table 8
Comparison of the reducts found by CEBARKNC and Q-DRA (bolded indicating the identical attribute reduct sequence and italic indicating the same reduct).

Data set (discretized)	Attribute sequence of reduct		Number of attributes in reduct		Execution time (s)	
	CEBARKNC	Q-DRA	CEBARKNC	Q-DRA	CEBARKNC	Q-DRA
Wpbc	**18, 1, 7, 21, 13, 3, 8, 33, 10**	**18, 1, 7, 21, 13, 3, 8, 33, 10**	9	9	1.371	1.296
Wine	**7, 10, 13, 5, 1**	**7, 10, 13, 5, 1**	5	5	0.198	0.163
Sat	18, 17, 20, 26, 12, 3, 35, 22, 27, 11, 5, 7, 10, 34, 1, 28, 19, 24, 31, 36, 23, 4, 15, 16, 2, 21, 13, 33, 8, 9, 29, 14	18, 17, 24, 16, 2, 25, 34, 11, 27, 35, 10, 12, 15, 31, 32, 19, 4, 1, 3, 23, 6, 20, 7, 21, 5, 28, 33, 36, 13, 8, 9, 29, 14	32	33	84.113	82.816
Segment	*12, 20, 3, 2, 9, 7, 13, 8, 16*	*12, 3, 20, 2, 9, 7, 13, 8, 16*	9	9	2.433	2.262
Wdbc	25, 30, 16, 4, 12, 27, 21, 5, 19, 11, 7, 20, 24	23, 30, 4, 24, 10, 16, 28, 29, 21, 20, 19, 27, 11	13	13	2.619	2.238
Wave	*15, 11, 6, 10, 7, 12, 14, 19, 3, 2, 20, 9, 18, 1, 4, 5, 17, 16, 13, 21, 8*	*7, 11, 15, 12, 9, 5, 14, 19, 2, 10, 3, 20, 1, 18, 4, 17, 13, 16, 21, 8, 6*	21	21	16.899	16.785
Vehicle	9, 7, 11, 16, 17, 19, 6, 4, 14, 2, 18, 5, 15, 3, 8	13, 7, 11, 16, 17, 19, 6, 4, 15, 2, 14, 18, 5, 3, 8	15	15	1.926	1.781
Ion	**35, 6, 18, 9, 26, 22, 28, 13, 20, 12, 14**	**35, 6, 18, 9, 26, 22, 28, 13, 20, 12, 14**	11	11	2.419	2.216
Glass	*4, 2, 5, 3, 10, 8, 7*	*2, 4, 5, 3, 10, 8, 7*	7	7	0.192	0.119
Heart	**14, 13, 4, 12, 2, 6, 10, 3, 8, 11, 9**	**14, 13, 4, 12, 2, 6, 10, 3, 8, 11, 9**	11	11	0.316	0.276
Sonar	**11, 27, 55, 29, 17, 51, 20, 16**	**11, 27, 55, 29, 17, 51, 20, 16**	8	8	0.349	0.313
Pid	*3, 9, 7, 6, 5, 8, 4, 2*	*3, 7, 9, 6, 5, 8, 4, 2*	8	8	0.275	0.269
Average	–	–	12.4	12.5	9.426	9.211

C4.5 classifier, QuickReduct algorithm obtains the mean ranks for Wpbc and Wine are 3.5 and 4 respectively. These results for QuickReduct are also not ideal.

All of Algorithms 1–4 adopt a forward greedy search strategy to find the reduct, and the differences lie only in the computing attribute significance measure. To show the differences, we can compare the classification power of all of the prefixes of four types of reducts. Concretely, for a specific data set, we can obtain four corresponding reduced data sets, which only include the first attribute found by the four algorithms (QuickReduct, Q-ARA, Q-DRA, and Q-MDRA), to train the C4.5 and RBF-SVM classifiers,. Then we compare the average classification accuracy over the 10 cross-validation runs. Accordingly, we compare the classification power for the first n (n is from 1 to the minimum number of attributes for four types of reducts for a specific data set) attributes found by these four algorithms. After these experiments, we obtain a classification accuracy for all of the prefixes with four types of reducts. Figs. 2 and 3 present the obtained results running on each reduced data set using the C4.5 and RBF-SVM classifiers respectively. In these two figures, the vertical axis represents the classification accuracy, and the horizontal axis represents the number of attributes in the prefixes for the four types of reducts.

For the C4.5 classifier, Fig. 2 shows that the classification accuracy does not always increase when the prefix enlarges. In most cases, the best classification accuracy is not obtained through using the whole reduct. This finding means that too many attributes can cause the phenomenon of overfitting for the C4.5 classifier. However, for the RBF-SVM classifier, Fig. 3 shows that the best classification accuracy is obtained by using the whole reduct in most cases. This observation means that to some extent, more attributes can help to raise the classification accuracy for the RBF-SVM classifier. For this reason, the maximum distribution reduct (found by Q-MDRA) has unsatisfactory classification accuracy (because it has relatively few attributes) for the RBF-SVM classifier. Furthermore, for each data set, Figs. 2 and 3 show that the classification accuracy curves have a fast convergence rate for both the maximum distribution reduct and the distribution reduct. The accuracy curves directly show that the first several attributes of the maximum distribution reduct and the distribution reduct have an apparent advantage in terms of classification power over the first few attributes of the assignment reduct and the relative positive region reduct. For the C4.5 classifier, Fig. 2 shows that the first few attributes for both the maximum distribution reduct and the distribution reduct are at or close to the highest accuracy. For the RBF-SVM classifier, Fig. 3 shows that the first few attributes for both the maximum distribution reduct and the distribution reduct are close to the highest accuracy. In most of the cases, the accuracy curves for both the assignment reduct and the relative positive region reduct are below the accuracy curves for both the distribution reduct and the maximum distribution reduct.

The best classification accuracy results of all of the prefixes for the four types of reducts for a specific data set are listed in Tables 6,7. Table 6 shows that for the C4.5 classifier, the maximum distribution reduct (found by Q-MDRA) obtains the best mean rank, and the distribution reduct (found by Q-DRA) obtains the second best mean rank. For the RBF-SVM classifier, the results listed in Tables 7 are relatively close to those in Table 5. The reason is that, in most of the cases, the best classification accuracy for the RBF-SVM classifier is obtained by using the whole reduct.

7.2. Comparison with CEBARKCNC

In [49], Yuan and Zhang have proven that the distribution reduct is actually equivalent to the reduct in the information view. Therefore, the reducts found by CEBARKCC and CEBARKNC are distribution reducts [43]. However, the principle of the Q-DRA (Quick Distribution Reduction Algorithm) differs from that of CEBARKNC (or CEBARKCC), although both algorithms compute the distribution reduct. The theoretical principle of the Q-DRA (Quick Distribution Reduction Algorithm) is a probabilistic rough set, but the theoretical principle of the CEBARKNC is conditional information entropy [32]. Next, we will directly compare the experimental results by using Q-DRA and CEBARKNC (both computing distribution reduct without a computing core).

For the data sets listed in Tables 1 and 8 demonstrates the attribute reduct sequences found by CEBARKNC and Q-DRA respectively. The experimental results show that the reducts found by CEBARKNC and Q-DRA exhibit a high degree of similarity: they generate the same attribute reducts for 9 data sets, and out of these 9 data sets, 5 (Wpbc, Wine, Heart, Sonar, and Ion) generate the identical attribute reduct sequences. We believe that the experimental results of Q-DRA and CEBARKNC exhibit similarities for the following two reasons: (1) both search out the distribution reduct, and (2) both use the forward greedy algorithm framework. CEBARKNC and Q-DRA in theory have the same time complexity. However, Q-DRA proves to

Table 9
Comparison of classification accuracy of C4.5 and RBF-SVM classifiers with reducts found by CEBARKNC and Q-DRA.

Data set (discretized)	C4.5		RBF-SVM	
	CEBARKNC	Q-DRA	CEBARKNC	Q-DRA
Wpbc	0.7911 ± 0.0808 (1.5)	0.7911 ± 0.0808 (1.5)	0.7628 ± 0.0489 (1.5)	0.7628 ± 0.0489 (1.5)
Wine	0.9586 ± 0.0426 (1.5)	0.9586 ± 0.0426 (1.5)	0.9883 ± 0.0229 (1.5)	0.9883 ± 0.0229 (1.5)
Sat	0.8659 ± 0.0162 (1)	0.8606 ± 0.0069 (2)	0.9082 ± 0.0085 (2)	0.9092 ± 0.0068 (1)
Segment	0.9540 ± 0.0165 (1.5)	0.9540 ± 0.0165 (1.5)	0.9579 ± 0.0182 (1.5)	0.9579 ± 0.0182 (1.5)
Wdbc	0.9501 ± 0.0272 (2)	0.9555 ± 0.0283 (1)	0.9587 ± 0.0255 (2)	0.9668 ± 0.0220 (1)
Wave	0.7782 ± 0.0213 (1.5)	0.7782 ± 0.0213 (1.5)	0.8328 ± 0.0188 (1.5)	0.8328 ± 0.0188 (1.5)
Vehicle	0.7033 ± 0.0415 (2)	0.7040 ± 0.0407 (1)	0.7373 ± 0.0387 (2)	0.7386 ± 0.0398 (1)
Ion	0.8994 ± 0.0400 (1.5)	0.8994 ± 0.0400 (1.5)	0.9419 ± 0.0386 (1.5)	0.9419 ± 0.0386 (1.5)
Glass	0.7054 ± 0.0818 (1.5)	0.7054 ± 0.0818 (1.5)	0.7402 ± 0.0877 (1.5)	0.7402 ± 0.0877 (1.5)
Heart	0.8030 ± 0.0688 (1.5)	0.8030 ± 0.0688 (1.5)	0.8374 ± 0.0716 (1.5)	0.8374 ± 0.0716 (1.5)
Sonar	0.7865 ± 0.0940 (1.5)	0.7865 ± 0.0940 (1.5)	0.8275 ± 0.0864 (1.5)	0.8275 ± 0.0864 (1.5)
Pid	0.7865 ± 0.0284 (1.5)	0.7748 ± 0.0284 (1.5)	0.7565 ± 0.0296 (1.5)	0.7565 ± 0.0296 (1.5)
Mean Rank	1.54	1.46	1.625	1.375

Table 10
Execution time of Algorithm_a and Q-ARA.

Data set (discretized)	Algorithm_a (finds assignment reduct)		Q-ARA
	Reduct numbers	Execution time (s)	Execution time (s)
Wpbc	17722	13134.551	1.328
Wine	72	1.372	0.182
Sat	*	*	82.812
Segment	11	17.693	2.180
Wdbc	3353	15911.062	2.226
Wave	1	169.598	16.466
Vehicle	6	4.012	1.778
Ions	*	*	2.211
Glass	5	0.486	0.118
Heart	1	0.593	0.275
Sonar	*	*	0.312
Pid	1	2.717	0.103

run faster than CEBARKNC because the execution time of Q-DRA is less than that of CEBARKNC on each data set. This finding can be explained by the fact that the computational costs of the distribution positive dependency degree in Q-DRA are relatively less than the computational costs of conditional information entropy in CEBARKNC.

Table 9 lists the effect of Q-DRA and CEBARKNCC on the performance of the C4.5 and RBF-SVM classifiers. The mean rank values of the classification accuracy for the C4.5 classifier obtained by Q-DRA and CEBARKNCC are 1.46 and 1.54, and the mean rank values obtained for the classification accuracy of the RBF-SVM classifier are 1.375 and 1.625. These results show that Q-DRA performs better than CEBARKNCC in terms of classification accuracy.

The concepts of information entropy and information gain are well known, and they have been successfully used in data mining [31,32]. In terms of finding a distribution reduct from decision tables, our experimental results demonstrate that the Q-DRA has stronger competitiveness, especially in computational efficiency.

7.3. Comparison with discernibility matrix-based algorithms

As mentioned in the Introduction, many discernibility matrix-based algorithms for reduction in inconsistent decision tables have been proposed. However, a lack is found in practical and effective algorithms for assignment reduction and maximum distribution reduction. This article aims to quickly find assignment reduct, distribution reduct, and maximum distribution reduct for large-scale inconsistent or consistent decision tables. To further demonstrate the practical value of our algorithms, we have used principles from paper [21] to design Algorithm_a for assignment reduction, and we have used principles from paper [52] to design Algorithm_b and Algorithm_c for distribution reduction and maximum distribution reduction respectively. These three algorithms are built on a discernibility matrix and a discernibility function. The time complexities of these three algorithms mainly depend on constructing a discernibility matrix and calculating a discernibility function. The time complexity of calculating a discernibility matrix is $O(|C||U|^2)$, and calculating a discernibility function constitutes an NP-hard problem. The execution time of Algorithm_a and Q-ARA is compared in Table 10, the execution time of Algorithm_b and Q-DRA is compared in Table 11, and the execution time of Algorithm_c and Q-MDRA is compared in Table 12. In Tables 10–12, we also list the respective numbers of reducts found by Algorithm_c, Algorithm_b and Algorithm_c, because these discernibility matrix-based algorithms enumerate all of the reducts in one decision table. Experimental results show that discernibility matrix-based algorithms can search all of the reducts for some of the data sets, but all are very time-consuming. We note: although Wave, Heart and Pid data sets include only one reduct, discernibility matrix-based algorithms consume much more computational time than our heuristic algorithms. Moreover, discernibility matrix-based methods

Table 11
Execution time of Algorithm_b and Q-DRA.

Data set (discretized)	Algorithm_b (finds distribution reduct)		Q-DRA
	Reduct numbers	Execution time (s)	Execution time (s)
Wpbc	17,722	13176.239	1.296
Wine	72	1.385	0.163
Sat	*	*	82.816
Segment	11	17.089	2.262
Wdbc	3353	17050.173	2.238
Wave	1	182.322	16.785
Vehicle	6	4.206	1.781
Ions	*	*	2.216
Glass	5	0.493	0.119
Heart	1	0.421	0.276
Sonar	*	*	0.313
Pid	1	4.2	0.269

Table 12
Execution time of Algorithm_c and Q-MDRA.

Data set (discretized)	Algorithm_c (finds maximum distribution reduct)		Q-MDRA
	Reduct numbers	Execution time (s)	Execution time (s)
Wpbc	17722	13107.057	1.429
Wine	72	1.325	0.162
Sat	*	*	72.218
Segment	25	26.782	1.906
Wdbc	3353	16120.701	2.129
Wave	1	171.138	16.340
Vehicle	6	3.949	1.786
Ions	*	*	1.737
Glass	8	0.507	0.118
Heart	1	0.581	0.261
Sonar	*	*	0.316
Pid	1	2.271	0.141

show insufficient memory and abort the execution after they run over 24 h for three data sets: Sat, Ions, and Sonar. We note that both Ions and Sonar include only hundreds of objects. This observation further confirms that the conversion from conjunction normal form to disjunction normal form constitutes an NP-hard problem. Thus, discernibility matrix-based methods are not feasible in real-world applications. For most of the applications in reality, only one reduct is required, and there is no need for all of the calculations that are involved in discovering the remainder. The experimental results further indicate that our three quick heuristic reduction algorithms are both efficient and feasible.

8. Conclusions

The contributions of this paper lie in the proposed judgment theorems for the assignment reduct, the distribution reduct and the maximum distribution reduct. These proposed theorems greatly simplify judging these three types of reducts. On this basis, we derive three novel types of attribute significance measures and construct the Quick Assignment Reduction Algorithm, the Quick Distribution Reduction Algorithm, and the Quick Maximum Distribution Reduction Algorithm. These three algorithms correspond to the three types of reducts. We have conducted a series of comparative experiments with twelve UCI data sets (including consistent and inconsistent decision tables) to evaluate the performance of the three reduction algorithms proposed in comparison with the relevant algorithm of QuickReduct. The experimental results show that the QuickReduct possesses weak robustness because it cannot find the reduct even for consistent data sets, whereas our three proposed algorithms show strong robustness because they can find a reduct for each data set. In addition, we have compared the Quick Distribution Reduction Algorithm with the CEBARKNC because both use a heuristic search to find the distribution reduct. The experimental results demonstrate that Quick Distribution Reduction Algorithm runs faster than CEBARKNC because the Quick Distribution Reduction Algorithm uses a simple heuristic function with a lower calculation cost. Although some discernibility matrix-based methods for reduction in inconsistent decision tables have been proposed [2,21,52], the conversion from conjunction normal form to disjunction normal form constitutes an NP-hard problem. We have compared discernibility matrix-based methods with our algorithms, and the experiments show that our algorithms are more efficient and feasible.

Furthermore, we also find that: (1) the maximum distribution reducts can effectively reduce the phenomenon of overfitting for the C4.5 classifier; (2) the distribution reducts are well suited to the RBF-SVM classifier; (3) assignment reducts and relative region reducts have relatively weak classification power; and (4) classification accuracy curves formed on all of the prefixes of both the maximum distribution reducts (found by Q-MDRA) and the distribution reducts (found by Q-DRA) show a fast convergence rate.

Our proposed three reduction algorithms have an identical framework with QuickReduct. While QuickReduct computes a positive region reduct, our algorithms compute assignment reduct, distribution reduct, and maximum distribution reduct. The time complexities of quick heuristic forward greedy algorithms are ideal. The experimental results show a high efficiency for all of our three algorithms. Nevertheless, how to further improve the efficiency of a quick heuristic forward reduction algorithm is not the focus of this paper. In fact, the major contribution of this paper lies in the proposal of three novel significance measures of attributes and the corresponding judgment theorems, which are the key issues for constructing quick heuristic forward greedy reduction algorithms for searching out assignment reduct, distribution reduct, and maximum distribution reduct. Although information entropy can be used to construct a forward greedy algorithm for searching distribution reduct, our proposed distribution function still involves novelty and has advantages in its calculation efficiency.

The time complexity of the heuristic forward greedy reduction algorithms largely depend upon the sorting technology that is adopted. The time complexity of our three algorithms are no more than $O(|C|^2|U|\log|U|)$ by using a quick sort algorithm, and the time complexity can be further decreased to $O(|C|^2|U|)$ by using a radix sort algorithm. Of course, to further improve the efficiency of the quick heuristic forward greedy reduction algorithms is worth studying, and it is a challenging task. In Refs. [29,30], Qian et al. have proposed an efficient accelerator to speed up the computation of heuristic forward greedy reduction algorithms. Obviously, this accelerator can also be applied in our algorithms. In addition, with other techniques such as cloud computing, we expect to develop some reduction algorithms that have higher performance, which will be our research work for the future.

Our proposed three quick reduction algorithms can be applied as a preprocessing technique for data mining in large-scale inconsistent or consistent data sets. It should be noted that these three reduction algorithms are suitable only for computing the three types of reduct in either inconsistent or consistent complete decision tables. In the tolerance relation-based rough set model, the concepts of the assignment reduct, the distribution reduct, and the maximum distribution reduct have been extended to inconsistent incomplete decision tables. In future work, we therefore intend to investigate the algorithms for the quick assignment reduction, for the quick distribution reduction, and for the quick maximum distribution reduction in inconsistent incomplete decision tables. Such research efforts are expected to enrich and improve the approach for intelligent data analysis in large-scale inconsistent incomplete decision tables.

Acknowledgements

We would like to thank all the referees for their critical and constructive comments and suggestions to improve the presentation of the paper. Research on this work was partially supported by the grants from the National Science Foundation of China (No. 61363047), by Jiangxi Provincial Natural Science Foundation (No. 2011ZBAB201006), by the funds from Jiangxi Education Department (No. GJJ13760), and by the funds from Science and Technology Support Foundation of Jiangxi Province (No. 20111BBE50008).

Appendix A

Theorem 8.1. *The inequality* $\frac{b_1^2}{a_1}+\frac{b_2^2}{a_2}+\cdots+\frac{b_m^2}{a_m}\geqslant\frac{(b_1+b_2+\cdots+b_m)^2}{a_1+a_2+\cdots+a_m}$ *is established if* $a_i>0\quad(1\leqslant i\leqslant m)$.

Proof (*using mathematical induction*). If $m=1$, then this inequality is obviously established.

If $m=2$, then the left is

$$\frac{b_1^2}{a_1}+\frac{b_2^2}{a_2}=\frac{(a_2b_1^2+a_1b_2^2)(a_1+a_2)}{a_1a_2(a_1+a_2)}=\frac{a_1a_2b_1^2+a_2^2b_1^2+a_1^2b_2^2+a_1a_2b_2^2}{a_1a_2(a_1+a_2)},$$

and the right is

$$\frac{(b_1+b_2)^2}{a_1+a_2}=\frac{a_1a_2(b_1+b_2)^2}{a_1a_2(a_1+a_2)}=\frac{a_1a_2b_1^2+2a_1a_2b_1b_2+a_1a_2b_2^2}{a_1a_2(a_1+a_2)}.$$

Subtract the right from the left, and then, the following expression is obtained: $\frac{a_2^2b_1^2-2a_1a_2b_1b_2+a_1^2b_2^2}{a_1a_2(a_1+a_2)}=\frac{(a_2b_1-a_1b_2)^2}{a_1a_2(a_1+a_2)}\geqslant 0$. Thus, $\frac{b_1^2}{a_1}+\frac{b_2^2}{a_2}\geqslant\frac{(b_1+b_2)^2}{a_1+a_2}$ is established.

Assume that when $m=k$, the inequality $\frac{b_1^2}{a_1}+\frac{b_2^2}{a_2}+\cdots+\frac{b_k^2}{a_k}\geqslant\frac{(b_1+b_2+\cdots+b_k)^2}{a_1+a_2+\cdots+a_k}$ is established. Let $g=b_1+b_2+\cdots+b_k$ and $h=a_1+a_2+\cdots+a_k$; then, when $m=k+1$, the left is:

$$\frac{b_1^2}{a_1}+\frac{b_2^2}{a_2}+\cdots+\frac{b_k^2}{a_k}+\frac{b_{k+1}^2}{a_{k+1}}\geqslant\frac{g^2}{h}+\frac{b_{k+1}^2}{a_{k+1}}=\frac{ha_{k+1}g^2+a_{k+1}^2g^2+h^2b_{k+1}^2+ha_{k+1}b_{k+1}^2}{ha_{k+1}(h+a_{k+1})},$$

and the right is:

$$\frac{(g+b_{k+1})^2}{(a+a_{k+1})}=\frac{ha_{k+1}g^2+2gha_{k+1}b_{k+1}+ha_{k+1}b_{k+1}^2}{ha_{k+1}(h+a_{k+1})}.$$

Subtract the right from the left; then the following expression is obtained:$\frac{(a_{k+1}g-b_{k+1}h)^2}{ha_{k+1}(h+a_{k+1})}\geqslant 0$. Thus, $\frac{b_1^2}{a_1}+\frac{b_2^2}{a_2}+\cdots+\frac{b_k^2}{a_k}+\frac{b_{k+1}^2}{a_{k+1}}\geqslant\frac{(b_1+b_2+\cdots+b_k+b_{k+1})^2}{a_1+a_2+\cdots+a_k+a_{k+1}}$ is also established.

This completes the proof. □

References

[1] M. Beynon, Reducts within the variable precision rough sets model: a further investigation, European Journal of Operational Research 134 (3) (2001) 592–605.

[2] D. Chen, C. Wang, Q. Hu, A new approach to attribute reduction of consistent and inconsistent covering decision systems with covering rough sets, Information Sciences 177 (17) (2007) 3500–3518.

[3] N. Cristianini, J. Shawe-Taylor, An Introduction to Support Vector Machines and Other Kernel-Based Learning Methods, Cambridge University Press, Cambridge, 2000.

[4] Y. Du, Q. Hu, P. Zhu, P. Ma, Rule learning for classification based on neighborhood covering reduction, Information Sciences 181 (24) (2011) 5457–5467.

[5] Q. Hu, D. Yu, Z. Xie, Hybrid attribute reduction based on a novel fuzzy-rough model and information granulation, Pattern Recognition 40 (2007) 3509–3521.

[6] Q. Hu, D. Yu, Z. Xie, Information-preserving hybrid data reduction based on fuzzy-rough techniques, Pattern Recognition Letters 27 (2006) 414–423.

[7] J. Jelonek, K. Krawiec, R. Slowinski, Rough set reduction of attributes and their domains for neural networks, Computational Intelligence 11 (2) (1995) 339–347.
[8] R. Jensen, Q. Shen, Semantics-preserving dimensionality reduction: rough and fuzzy-rough-based approaches, IEEE Transactions on Knowledge and Data Engineering 16 (12) (2004) 1457–1471.
[9] R. Jensen, Q. Shen, A rough set-aided system for sorting WWW bookmarks, in: N. Zhong et al. (Eds.), Web Intelligence: Research and Development, 2001, pp. 95–105.
[10] R. Jensen, Q. Shen, Fuzzy-rough attribute reduction with application to web categorization, Fuzzy Sets and Systems 141 (3) (2004) 469–485.
[11] M. Kryszkiewicz, Comparative studies of alternative type of knowledge reduction in inconsistent systems, International Journal of Intelligent Systems 16 (1) (2001) 105–120.
[12] M. Kryszkiewicz, Certain, generalized decision, and membership distribution reducts versus functional dependencies in incomplete systems, Proceedings of Rough Sets and Intelligent Systems Paradigms (2007) 162–174.
[13] J. Li, C. Mei, Y. Lv, Knowledge reduction in real decision formal contexts, Information Sciences 189 (2012) 191–207.
[14] M. Li, S.B. Deng, S.Z. Feng, J.P. Fan, An effective discretization based on class-attribute coherence maximization, Pattern Recognition Letters 32 (2011) 1962–1973.
[15] J.Y. Liang, Z.B. Xu, The algorithm on knowledge reduction incomplete information systems, International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems 10 (2002) 95–103.
[16] J. Liu, Q. Hu, D. Yu, A weighted rough set based method developed for class imbalance learning, Information Sciences 178 (4) (2008) 1235–1256.
[17] S.H. Liu, Q.J. Sheng, B. Wu, Z.Z. Shi, F. Hu, Research on efficient algorithms for rough set methods, Chinese Journal of Computers 26 (5) (2003) 24–529.
[18] Z.Q. Meng, Z.Z. Shi, A fast approach to attribute reduction in incomplete decision systems with tolerance relation-based rough sets, Information Sciences 179 (2009) 2774–2793.
[19] Z.Q. Meng, Z.Z. Shi, Extended rough set-based attribute reduction in inconsistent incomplete decision systems, Information Sciences 204 (2012) 44–66.
[20] J.S. Mi, W.Z. Wu, W.X. Zhang, Approaches to knowledge reduction based on variable precision rough set model, Information Sciences 159 (2004) 255–272.
[21] 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, Information Sciences 179 (24) (2009) 4140–4150.
[22] F. Min, H. He, Y. Qian, W. Zhu, Test-cost-sensitive attribute reduction, Information Sciences 181 (22) (2011) 4928–4942.
[23] Z. Pawlak, Rough sets, International Journal of Computer and Information Sciences 11 (5) (1982) 341–356.
[24] Z. Pawlak, Rough set approach to multi-attribute decision analysis, European Journal of Operational Research 72 (1993) 443–459.
[25] Z. Pawlak, Rough set theory and its applications in data analysis, Cybernetics and Systems 29 (1998) 661–688.
[26] Z. Pawlak, Rough sets and intelligent data analysis, Information Sciences 147 (2002) 1–12.
[27] Z. Pawlak, A. Skowron, Rough membership functions, in: R.R. Yager, M. Fedrizzi, J. Kacprzyk (Eds.), Advances in the Dempster–Shafer Theory of Evidence, John Wiley and Sons, New York, 1994, pp. 251–271.
[28] Y.H. Qian, J.Y. Liang, D.Y. Li, F. Wang, N. Ma, Approximation reduction in inconsistent incomplete decision tables, Knowledge-Based Systems 23 (5) (2010) 427–433.
[29] Y.H. Qian, J.Y. Liang, W. Pedrycz, C.Y. Dang, Positive approximation: an accelerator for attribute reduction in rough set theory, Artificial Intelligence 174 (2010) 597–618.
[30] Y.H. Qian, J.Y. Liang, W. Pedrycz, C.Y. Dang, An efficient accelerator for attribute reduction from incomplete data in rough set framework, Pattern Recognition 44 (2011) 1658–1670.
[31] J.R. Quinlan, C4.5 Programs for Machine Learning, Morgan Kaufman Publisher, San Mateo, CA, 1993.
[32] J.R. Quinlan, Induction of decision trees, Machine Learning 1 (1986) 81–106.
[33] Q. Shen, A. Chouchoulas, A rough-fuzzy approach for generating classification rules, Pattern Recognition 35 (11) (2002) 2438–2452.
[34] Q. Shen, A. Chouchoulas, A modular approach to generating fuzzy rules with reduced attributes for the monitoring of complex systems, Engineering Applications of Artificial Intelligence 13 (3) (2000) 263–278.
[35] Q. Shen, R. Jensen, Selecting informative features with fuzzy-rough sets and its application for complex systems monitoring, Pattern Recognition 37 (7) (2004) 1351–1363.
[36] A. Skowron, C. Rauszer, The discernibility matrices and functions in information systems, in: R. Słowiński (Ed.), Intelligent Decision Support, Handbook of Applications and Advances of the Rough Sets Theory, Kluwer, Dordrecht, 1992.
[37] A. Skowron, S.K. Pal, Rough Sets, Pattern Recognition, and Data Mining, Pattern Recognition Letters 24 (6) (2003) 829–933.
[38] D. Slezak, Approximate entropy reducts, Fundamenta Informaticae 53 (3–4) (2002) 365–390.
[39] D. Slezak, Normalized decision functions and measures for inconsistent decision tables analysis, Fundamenta Informaticae 44 (2000) 291–319.
[40] S. Tsumoto, Automated extraction of medical expert system rules from clinical databases based on rough set theory, Information Sciences 112 (1–4) (1998) 67–84.
[41] S. Tsumoto, Mining diagnostic rules from clinical databases using rough sets and medical diagnostic model, Information Sciences 162 (2) (2004) 65–80.
[42] G.Y. Wang, Rough reduction: in algebra view and information view, International Journal of Intelligent Systems 18 (6) (2003) 679–688.
[43] G.Y. Wang, H. Yu, D. Yang, Decision table reduction based on conditional information entropy, Chinese Journal of Computers 25 (7) (2002) 759–766.
[44] W.Z. Wu, M. Zang, H.Z. Li, J.S. Mi, Knowledge reduction in random information systems via Dempster–Shafer theory of evidence, Information Sciences 174 (2005) 143–164.
[45] H.H. Yang, T.C. Liu, Y.T. Lin, Applying rough sets to prevent customer complaints for IC packaging foundry, Expert Systems with Applications 32 (2007) 151–156.
[46] Y.Y. Yao, Y. Zhao, Attribute reduction in decision-theoretic rough set models, Information Sciences 178 (17) (2008) 3356–3372.
[47] D.Y. Ye, Z.J. Chen, A new type of attribute reduction for inconsistent decision tables and its computation, International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems 18 (2) (2010) 209–222.
[48] D.Y. Ye, Z.J. Chen, S.L. Ma, A novel and better fitness evaluation for rough set based minimum attribute reduction problem, Information Sciences 222 (2013) 413–423.
[49] X.J. Yuan, W.X. Zhang, Studies on equivalence of the distribution reduction and the strictly convex function based reduction in decision tables, Systems Engineering 21 (5) (2003) 5–7.
[50] W.X. Zhang, G.F. Qiu, W.Z. Wu, A general approach to attribute reduction in rough set theory, Science in China F: Information Sciences 50 (2) (2007) 188–197.
[51] W.X. Zhang, J.S. Mi, W.Z. Wu, Knowledge reduction in inconsistent information systems, Chinese Journal of Computers 26 (2003) 12–18.
[52] W.X. Zhang, J.S. Mi, W.Z. Wu, Approaches to knowledge reductions in inconsistent systems, International Journal of Intelligent Systems 18 (2003) 989–1000.
[53] W. Zhu, F.Y. Wang, Reduction and axiomization of covering generated rough sets, Information Sciences 152 (2003) 217–230.
[54] W. Ziarko, Variable precision rough set model, Journal of Computer System Science 46 (1993) 35–39.