Information Sciences 503 (2019) 533–550

Contents lists available at ScienceDirect

Information Sciences

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

PARA: A positive-region based attribute reduction accelerator

Peng Ni[a], Suyun Zhao[a,*], Xizhao Wang[b], Hong Chen[a], Cuiping Li[a]

[a] School of Information, Renmin University of China, Beijing, PR China
[b] Shenzhen University, Shenzhen, Guangdong, PR China

ARTICLE INFO

Article history:
Received 15 December 2018
Revised 27 May 2019
Accepted 8 July 2019
Available online 10 July 2019

Keywords:
Attribute reduction
Fuzzy rough techniques
Accelerator
Positive region

ABSTRACT

Attribute reduction, also known as feature selection, is a common problem by selecting a subset of relevant attributes (e.g. features) to reach efficient learning/mining. Many attribute reduction methods have been proposed however, quite often, these methods are still computationally time-consuming while handling large-scale data. To overcome this shortcoming, we present a novel accelerator based on the positive region, by deleting the learned/discernible instance pairs in the process of attribute reduction, which can avoid redundant computation and accelerate attribute reduction. Our experiments numerically demonstrate that the proposed accelerator can reach drastically faster computation than previous methods, especially on the datasets with a large number of instances.

1. Introduction

In recent years, we encounter databases in which the instances with high dimension become dramatically large. Hundreds, thousands or even millions of such instances are stored in many real-world application databases [2,6,15,19]. Storing and processing all instances might be computationally costly and impractical for machine learning and data mining [2,20]. Data mining based on granular computing is an emerging research field. A vast number of real-world problems are tackled using techniques encompassed in granular computing [26,47]. Attribute reduction based on fuzzy rough sets, also known as feature selection, is a significant and effective way to reduce the scale of data in data mining, machine learning and granular computing [2,20,21,23,26,30,36,47].

By using an evaluation measure to score every attribute and/or different attribute subsets, attribute reduction selects a subset of relevant attributes to speed up learning/mining and improve learning/mining quality [14]. Those attribute reduction methods are roughly split into three main categories: wrappers, filters and embedded algorithms [14]. In this paper, we focus on the filter method, which selects attributes regardless of the learning/mining model and often works as a preprocessing method.

Now, there are many known filter attribute reduction methods. For example, RELIEF [21] is a typical algorithm on the filter method, which could effectively estimate the quality of attributes by measuring the interactions among features. For another example, mRMR (maximum relevance minimum redundancy) [30] is a mutual information-based filter feature selection method for finding a set of relevant and complementary features. What is more, rough/fuzzy rough reduction is a useful attribute reduction method [1,3,4,5,16,18,28,37,38,39,43,48,49], whose characteristics are human understanding and non-need extra expert knowledge.

* Corresponding author.
E-mail addresses: nipeng@ruc.edu.cn (P. Ni), zhaosuyun@ruc.edu.cn (S. Zhao), xizhaowang@ieee.org (X. Wang), chong@ruc.edu.cn (H. Chen), cuiping_li@263.net (C. Li).

https://doi.org/10.1016/j.ins.2019.07.038
0020-0255/

Fuzzy rough sets (FRS) are the generalization of rough sets in fuzzy set framework, which assume that instances characterized by the same information are indiscernible (similar) in the view of the available information about them (with every instance in the Universe we associate some information) [9,22,28,29,45]. The fuzzy indiscernibility (similar/equivalence) relation generated in this way is the mathematical basis of fuzzy rough set [28,46], which makes fuzzy rough sets work well on some problems, but also limits the further application of fuzzy rough sets [5,17,38,45,43]. For example, fuzzy rough based attribute reduction algorithms are rather time-consuming, or even cannot work efficiently on the large-scale datasets. Because fuzzy rough sets must discern all the heterogeneous pairs in the universe. As a result, it is necessary to propose new ways into fuzzy rough sets which could reduce the huge computation.

To handle the problems of huge computations, some researchers have proposed a lot of heuristic attribute reduction algorithms including parallel and accelerated methods [7,24,27,31,32,33,34,35,40]. Some parallel methods have been proposed to speed up the computation of attribute significance by parallelizing the traditional attribute reduction process based on MapReduce mechanism [7,27,31,32]. Whereas the parallel methods still have to consider all the instances in the universe, and the redundant computation is not avoided. Then, some incremental attribute reduction methods have been proposed [11,25,41,42], which mainly focus on handling the dynamic datasets, such as incremental attributes, incremental instances and incremental attribute values.

To handle static huge data, some accelerators have been proposed [24,33,34,35,40]. From the perspective of instances, Qian et al. [33] proposed an accelerator called positive approximation for attribute reduction from complete data. Furthermore, Qian et al. [35] presented a theoretical framework called positive approximation to accelerate a heuristic process for attribute reduction from incomplete data. From the perspective of both instances and attributes, Liang et al. [24] introduced an accelerator which could remove insignificant attributes in the process of attribute reduction. More interestingly, Wei et al. [40] accelerated incremental attribute reduction by compacting a decision table. In these accelerators, they remove the redundant instances, which have been discerned by the current selected attributes, in the process of attribute reduction. Whereas fuzzy rough sets could not conduct this operation, Because the selected attributes just could discern some heterogeneous instance pairs, not some instances from all heterogeneous ones. This is one special difficulty for fuzzy rough accelerator.

In this paper, we propose an accelerator based on the positive region in the fuzzy rough set model. The basic definition to attribute reduction: the positive region which is obtained by the lower approximation, is really time-consuming. What is more, many heterogeneous instance pairs, which have been discerned in the process of attribute reduction, are still used in the subsequent calculation to find the new attributes. If those instance pairs, which are not key and redundant to calculate the positive region, are successively removed, then attribute reduction would be accelerated. Now, Qian et al. [34] already developed an extended version of the accelerator called forward approximation for accelerating fuzzy rough attribute reduction (shortened by FA-FPR). Actually, it can be seen as another kind of accelerator of rough sets, not a really accelerator based on the general fuzzy rough set model. This motivates us to propose a new fuzzy rough accelerator based on a generalized fuzzy rough set, which is a necessary supplement for the fuzzy rough accelerator.

In this paper, we choose Dübois and Prade's fuzzy rough model [13] to compute the fuzzy positive region, and then design our accelerated algorithm. The main contributions in this paper include:

1) Key Instance Set, which only contains the instances key to update the candidate of reduction, is proposed. The monotonicity of Key Instance Set makes our accelerator feasible.
2) Order preservation property of Key Instance Set ensures that the reduct found by the accelerator is consistent with the non-accelerated one.
3) Based on Key Instance Set, a positive region attribute reduction accelerator, called PARA, is proposed which avoids redundancy computation on the whole universe.

The remainder of this paper is organized as follows. In Section 2, two types of fuzzy rough set models are briefly reviewed. In Section 3, we propose an alternative version of fuzzy rough based attribute reduction algorithm, shortened by PAR. In Section 4, we propose a fuzzy rough based attribute reduction accelerator, shortened by PARA. In Section 5, numerical experiments on eight datasets are given to show our proposed accelerated algorithm outperforms the state-of-the-art, i.e., FA-FPR. Finally, we conclude this paper in Section 6.

2. Existing fuzzy rough based reductions and accelerators

In this section, we briefly review two types of existing fuzzy rough set models and some reduction concepts, such as fuzzy positive region, reducts and accelerator [13,17,34,38,45,43].

2.1. Some notations

The dataset usually is described as one decision table, denoted by $DT = (U, C \cup D)$. Let $U = \{x_1, x_2, \ldots, x_n\}$, called the Universe, be a nonempty set with a finite number of instances. Each instance in U is described by a non-empty finite set of attributes, denoted by $C \cup D$; C denotes the set of condition attributes and D denotes the set of decision attributes, $C \cap D = \emptyset$; When the attributes in $C \cup D$ are crisp, each attribute $r \in C \cup D$ corresponds to a $U \to V_r$ mapping, in which V_r is the value set of r over U. The Universe is split into q equivalence classes $U/C = \{X_1, X_2, \ldots, X_q\}$, where $U = \bigcup_{i=1}^{q} X_i$ and $X_i \cap X_j = \emptyset$ (for any

$i \neq j$). $\forall X \subseteq U, B \subseteq C, \underline{B}X = \{X_i | X_i \subseteq X \text{ and } X_i \in U/B\}$; $\bar{B}X = \{X_i | X_i \cap X \neq \emptyset \text{ and } X_i \in U/B\}$, the pair $(\underline{B}X, \bar{B}X)$ is called a rough set of X on the attribute set B. It is easy to find that the rough set theory is only suitable for crisp attributes.

Let A be a fuzzy subset on U, which is defined as a mapping A: $U \to [0, 1]$. $\forall x \in U$, $A(x) \in [0, 1]$ is a fuzzy membership degree of x belonging to fuzzy set A [46]. If the attributes are continuous, not crisp, each attribute $r \in C \cup D$ corresponds to a $U \to [0, 1]$ mapping. That is to say, each continuous attribute could be mapped into a fuzzy set. The decision table with continuous attributes is then called a Fuzzy Decision Table, shortly denoted by $FD = (U, C \cup D)$.

A map $\tilde{R} : U \times U \to [0, 1]$ is called a binary relation on U, denoted by $\tilde{R}(\cdot, \cdot)$. Moreover, given a triangular norm T: $[0, 1] \times [0, 1] \to [0, 1]$, we say that a fuzzy relation $\tilde{R}(\cdot, \cdot)$ on U is a fuzzy T-similarity relation induced by continuous attribute subset $B \subseteq C$ if it satisfies the following conditions: for any $x, y, z \in U$. $\tilde{R}_B(x, x) = 1$ (reflexivity); $\tilde{R}_B(x, y) = \tilde{R}_B(y, x)$ (symmetry);$\tilde{R}_B\ (x, y) \geq T(\tilde{R}_B\ (x, z), \tilde{R}_B(z, y))$ (T-transitivity).

In this paper, we take the bounded intersection (also called the Lukasiewicz T-norm) $T_L = max\{0, a + b - 1\}$ as a special case of triangular norm T.

Given a fuzzy decision table $FD = (U, C \cup D)$ and $B \subseteq C$. Then $\tilde{R}_B$, as the fuzzy similarity relation induced by the attribute subset B, satisfies:

1) $\tilde{R}_B = \cap_{a \in B} \tilde{R}_a$;
2) $\tilde{R}_a(x_i, x_j) = 1 - (\max(a(x_i), a(x_j)) - \min(a(x_i), a(x_j)))$;
3) $\tilde{R}_B \succcurlyeq \tilde{R}_C \Leftrightarrow \tilde{R}_B(x_i, x_j) \geq \tilde{R}_C(x_i, x_j)$.

If using the terms of granular computing, we denote the coarseness/fineness relationship between any two fuzzy similar relation $\tilde{R}_B \succcurlyeq \tilde{R}_C$ [34]. Here $a(x_i)$ represents the attribute value of x_i on attribute a.

Let $B_1 \subseteq B_2 \subseteq \ldots \subseteq B_n$, $\tilde{R}_{B_i}$ is the fuzzy similarity relation induced by the attribute subset B_i, for $i = 1, 2, \ldots, n$, then $\{\tilde{R}_{B_1}, \tilde{R}_{B_2}, \ldots, \tilde{R}_{B_i}\}$ represents a family of fuzzy similarity relations on $B_1, B_2, \ldots, B_i$ satisfying $\tilde{R}_{B_1} \succcurlyeq \tilde{R}_{B_2} \succcurlyeq \ldots \succcurlyeq \tilde{R}_{B_i}$.

2.2. Conceptual definitions of the theory of rough sets

A conceptual definition of rough sets focuses on the meaning, interpretation and inherent properties of rough sets and their generalizations, whereas a computational definition focuses on algorithms and methods for constructing their applications. The reference [44] already theoretically and systemically discusses the conceptual and computational sides of rough sets. And it is helpful to construct the reduction algorithm. As a result, in this Subsection, we briefly review some conceptual definitions of the theory of rough sets, which are helpful to interpret the meaning of rough sets/fuzzy rough sets and their applications of reductions. For more details, please kindly refer to [44].

A conceptual definition of rough set approximation, which provides a clear interpretation and a conceptual understanding of approximation in rough set theory.

Definition 2.1. Given a decision table $DT = (U, C \cup D)$, $X \subseteq U$, $B \subseteq C$, the lower and upper approximation of are defined by the following pair of definable sets,

(1) $\underline{apr}_B(X)$ = the greatest definable set in $\mathrm{DEF}_B(DT)$ contained by X;
(2) $\overline{apr}_B(X)$ = the least definable set in $\mathrm{DEF}_B(DT)$ containing X.

Where $\mathrm{DEF}_B(DT)$ denotes the family of all definable sets by using the attribute subset B in a decision table D. (With regard to the description of definable sets, please kindly refer to Definitions 1&2 in [44].)

Definition 2.1 is interpretable and helpful to design the computational definition of rough sets/fuzzy rough sets.

An alternative approach to define and interpret rough set approximations is based on three regions, namely, the positive, negative and boundary regions, as follows:

$POS_B(X)$ = the greatest definable set in $\mathrm{DEF}_B(DT)$ contained by X.
$NEG_B(X)$ = the greatest definable set in $\mathrm{DEF}_B(DT)$ contained by X^c.
$BND_B(X) = (POS_B(X) \cup NEG_B(X))^c$.

where $(\cdot)^c$ denotes the complement of a set.

The pair of lower and upper approximation and the three regions are two different, but mathematically equivalent, forms of rough set approximation [44].

In rough set theory, a subset may be considered as a reduct of the original set if and only if the subset can serve the same power or performance as that of the entire set. Suppose S is a finite set. Let $\mathbb{P}$ denote a unary predicate on subsets of S, that is, for $K \subseteq S$, $\mathbb{P}(K)$ stands for the statement that "subset K has property $\mathbb{P}$". The values of $\mathbb{P}$ are computed by an evaluation e with reference to certain available data or information. For a subset $K \subseteq S$, $\mathbb{P}_e(K)$ is true if K has property $\mathbb{P}$, otherwise, it is false. A conceptual definition of a reduct of a set is then defined as follows:

Definition 2.2. Given an evaluation e of $\mathbb{P}$, if a subset $R \subseteq S$ satisfies the following conditions:

1) Existence: $\mathbb{P}_e(S)$;
2) Sufficiency: $\mathbb{P}_e(R)$;

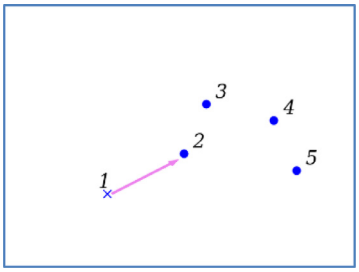


Fig. 2.1. The illustration of the positive region.

3) Minimization: $\forall B \subsetneq R$, $\neg \mathbb{P}_e(B)$.

We call R a reduct of S.

The three conditions reflect the fundamental characteristics of a reduct. The condition of existence requires that the whole set S must have the property $\mathbb{P}$. And it ensures that a reduct of S exists. The condition of sufficiency requires that a reduct R of S is sufficient for preserving property $\mathbb{P}$. The condition of minimization requires that a reduct is a minimal subset of S having the property.

2.3. One classical fuzzy rough set model

In most practical applications, condition attributes are usually continuous. This paper focuses on this type of real applications. And the fuzzy decision table with crisp decision attributes and continuous attributes is considered as the platform of this paper. To handle this type of applications, fuzzy sets [46] is introduced into rough sets and then fuzzy rough sets are proposed, which is one known generalization of rough sets supporting both continuous and crisp values [13,38,45]. The fuzzy rough set model was first introduced by Dübois and Prade [13].

As Dübois and Prade defined [13], if a crisp/fuzzy set is approached by a family of fuzzy sets in the same universe, then the corresponding lower/upper approximation pair is called a fuzzy rough set [34]. In this paper, our method is based on the fuzzy rough set proposed by Dübois and Prade, our discussions consider the membership of a single instance from the universe.

Definition 2.3. Given a fuzzy decision table $FD = (U, C \cup D)$ and for any fuzzy set A on U. For each $x \in U$, an ordered pair of lower and upper approximation operators of A on U is defined as

$$\begin{cases} \underline{\tilde{R}_B} A(x) = inf_{u \in U} \max\left\{1 - \tilde{R}_B(x,u), A(u)\right\} \\ \overline{\tilde{R}_B} A(x) = sup_{u \in U} \min\left\{\tilde{R}_B(x,u), A(u)\right\} \end{cases}$$

where $\tilde{R}_B$ is a fuzzy similarity relation induced by the attribute subset B on U.

Definition 2.3 presents the computational definition of fuzzy rough sets approximations, which is consistent with the conceptual definition of rough sets approximations. The meaning of $\underline{\tilde{R}_B} A(x)$ is the minimum distance of the instance x in the universe from its heterogeneous instances. That is to say, those instances, whose distances from x are less than or equal to $\underline{\tilde{R}_B} A(x)$, are definitely (greatest definable) belonging the same decision class with the instance x, which is consistent with Definition 2.1(1). The meaning of $\overline{\tilde{R}_B} A(x)$ is the similarity of the instance in Universe from the farthest homogeneous instances. That is to say, those instances, whose similarities are less than or equal to $\overline{\tilde{R}_B} A(x)$, are possibly (i.e., least definable) belonging to the same decision class with the instance x, which is consistent with Definition 2.1(2).

Definition 2.4. Given a fuzzy decision table $FD = (U, C \cup D)$ and for any fuzzy set A on U. For each $x \in U$, the fuzzy positive region of D relative to $B \subseteq C$ can be denoted as

$$POS_B^U(x) = sup_{X \in U/D} \underline{\tilde{R}_B} A(x); POS_B^U(D) = \{POS_B^U(x) | \forall x \in U\}.$$

The positive region of x is a value belonging to [0,1]. The relation between the fuzzy positive region of x and the lower approximation of x satisfies the following properties.

Proposition 2.1. *Given a fuzzy decision table $FD = (U, C \cup D)$, the fuzzy positive region of relative to $B \subseteq C$ can be simplified as follows.*

$$\forall x \in U, POS_B^U(x) = \underline{\tilde{R}_B}[x]_D(x) = \min_{\{u \in U, u \notin [x]_D\}} \left\{1 - \tilde{R}_B(x,u)\right\}.$$

Proposition 2.1 shows the computational method of the fuzzy positive region. The meaning of the positive region of x is the minimum distance of the instance x from its heterogeneous instances, just as Fig. 2.1.

In Fig. 2.1, the crossing points like "x" denote the positive class and the dot points like "•" denote the negative class. The purple line in Fig. 2.1 demonstrates that the positive region of the point "x" means that the distance between the positive class point "x" and its nearest negative class point "2".

What is more, Fig. 2.1and Proposition 2.1 illustrate that any instance, whose distance from x is less than or equal to $POS_B^U(x)$, is definitely belonging to the same decision class with x, and any instance whose distance from x is larger than $POS_B^U(x)$, may not belong to the same decision class with x. These show that the value of $POS_B^U(x)$ is the upper bound (i.e., greatest definable set) of those instances which definitely belonging to the same decision class with x, which is just consistent with the conceptual definition of the positive region.

The positive region for D is a fuzzy set on U. For any $x \in U$, the membership of x belonging to $POS_B^U(D)$ is $POS_B^U(x)$. Actually, $POS_B^U(D)$ measures all the discernibility information of the instances in the Universe.

Since the crisp set is a special case of fuzzy set, Definitions 2.3 and 2.4 also suit for the crisp set $X \subseteq U$. It is necessary to point out that no matter X is crisp or not, $POS_B^U(D)$ is always a fuzzy set.

Definition 2.5. Given a fuzzy decision table $FD = (U, C \cup D)$, the dependence degree of D on C, denoted by γ_C^U, is defined as

$$\gamma_C^U = \sum_{x \in U} POS_C^U(x)/|U|,$$

where $|\cdot|$ denotes the cardinality of a set. Here the dependence degree is a real value between 0 and 1.

By Definition 2.5, the dependence degree is the average of the fuzzy positive region. Its physical meaning could be seen as the average minimal distance of all instances from their heterogeneous class. That is to say, the dependence degree reflects the discernibility power of the attribute set.

Definition 2.6. Given a fuzzy decision table $FD = (U, C \cup D)$, $B \subseteq C$ and $\forall a \in C - B$. The significance of a in B is defined as

$$Sig_1(a, B, D, U) = \gamma_{B \cup \{a\}}^U - \gamma_B^U.$$

The significance of an attribute is the difference between the dependence degrees when a new attribute is added.

Let γ_C^U be the property $\mathbb{P}(C)$. By Definition 2.2, which is the conceptual definition of a reduct, the computational definition of a reduct could be defined as follows.

Definition 2.7. Given a fuzzy decision table $FD = (U, C\ D)$, $B \subseteq C$ is called a reduct of C with regard to D if B satisfies the following two statements:

(1) $\gamma_C^U = \gamma_B^U$;
(2) for any $r \in B$, $\gamma_C^U \neq \gamma_{B-\{r\}}^U$.

Similar to Definition 2.2, the statements in Definition 2.7 reflect the fundamental characteristics of a reduct. The first statement reflects the condition of sufficiency, which ensures that a reduct is sufficient for preserving dependence degree. The second statement reflects the condition of minimization, which ensures that the reduct is one minimal subset of attributes to keep the dependence degree unchanged.

To design an attribute reduction algorithm, it is necessary to know how dependence degree grows with the increasing attributes. And then Proposition 2.2 is given as follows.

Proposition 2.2. *Given a fuzzy decision table $FD = (U, C \cup D)$, if $B \subseteq C$, then (1) $POS_B^U(x) \leq POS_C^U(x)$; (2) $\gamma_B^U \leq \gamma_C^U$.*

Proposition 2.2 only holds for the fuzzy decision table with crisp decision attributes. Proposition 2.2 shows that both the dependence degree and positive region are monotonic with the increasing attributes. This result is the theoretical foundation to design the algorithm of attribute reduction.

By starting with the empty set, we can successively add elements until we have a set that is jointly sufficient for preserving the dependence degree (i.e., which satisfies the sufficiency of a reduct, $\gamma_C^U = \gamma_B^U$). Since such a set may contain redundant attributes, we need to delete the redundant attributes to satisfy the minimization of a reduct (i.e., for any $\in B$, $\gamma_C^U \neq \gamma_{B-\{r\}}^U$). The attribute reduction algorithm based on the dependence degree, shortened by DAR, is then described in Algorithm 2.1 [43].

DAR is a heuristic algorithm, which is designed based on the monotonicity of the dependence degree. DAR is classical and known as the dependence degree based fuzzy rough attribute reduction. Sometimes, we will replace $\gamma_B^U < \gamma_C^U$ by $\gamma_B^U + \alpha < \gamma_C^U$, $\alpha \in [0, 1)$ in Step 4, which means we allow the gap between γ_B^U and γ_C^U less than a threshold α. Meanwhile, $\gamma_{red-\{a_i\}}^U = \gamma_C^U$ in step 6 will be replaced by $\gamma_{red-\{a_i\}}^U + \alpha \geq \gamma_C^U$. Comparatively, few researchers pay attention to the attribute reduction algorithms based positive region.

2.4. Fuzzy rough sets based on the cut fuzzy similarity relation

There also exists another type of fuzzy rough sets which is based on a cut fuzzy similarity relation. Usually, we called it cut fuzzy rough sets if no confusion arising. Cut fuzzy rough set is applicable to deal with the hybrid data with both crisp

Algorithm 2.1 DAR.

Input: $FD = (U, C \cup D)$
Output: red
1. $B \leftarrow \emptyset$;
2. $lef \leftarrow C - B$;
3. **Calculate** γ_C^U;
4. **While** $\gamma_B^U < \gamma_C^U$ **Do**
   $a^* = arg(\max_{a \in lef}\{Sig_1(a, B, D, U)\})$,
   $B \leftarrow B \cup \{a^*\}$,
   $lef \leftarrow lef - \{a^*\}$;
5. **Let** $red \leftarrow B$, $i = 0$;
6. **While** $i < |B|$ **Do**
   Take the ith attribute a_i in B
   if $\gamma_{red-\{a_i\}}^U = \gamma_C^U$ **then** $red \leftarrow red - \{a_i\}$;
   $i = i + 1$;
7. **Output** red;

Algorithm 2.2 FA-FPR.

Input: $FD = (U, C \cup D)$
Output: red
1. $red \leftarrow \emptyset, i \leftarrow 1$, and $U_1 \leftarrow U$;
2. **While** $C\_\gamma_{red}^{U_i} < C\_\gamma_C^{U_i}$ **Do**
   Calculate $C\_POS_{B_i}^U(D)$,
   $U_{i+1} \leftarrow U - C\_POS_{B_i}^U(D)$,
   $i \leftarrow i + 1$,
   $lef \leftarrow C - red$,
   Select $a^* \in lef$ which satisfies
   $Sig_1(a^*, red, D, U_i) = \max_{a \in lef}\{Sig_1(a, red, D, U_i)\}$;
   If $Sig_1(a^*, red, D, U_i) > 0$, then $red \leftarrow red \cup \{a^*\}$.
3. **Output** red;

and real-values. Since the state-of-the-art fuzzy rough accelerator is designed based on them [24], we briefly review them as follows.

The fuzzy cut similarity degree between the instances x_i and x_j with respect to the numerical attribute a is computed as $\widetilde{C\_R}_a(x_i, x_j) = \begin{cases} 1 - |a(x_i) - a(x_j)|/\beta, & |a(x_i) - a(x_j)| \le \beta \\ 0, & otherwise \end{cases}$, where $\beta \in (0, 1]$.

Definition 2.8. Given a fuzzy decision table $FD = (U, C \cup D)$ and for any fuzzy set A on U. The lower and upper approximation operators of A on U are defined as

$$\begin{cases} \underline{\widetilde{C\_R}_B A} = \{x_i | [x_i]_{\widetilde{C\_R}_B} \subseteq A, \ x_i \in U\} \\ \overline{\widetilde{C\_R}_B A} = \left\{x_i | [x_i]_{\widetilde{C\_R}_B} \cap A \neq \emptyset, \ x_i \in U\right\} \end{cases},$$

where $[x_i]_{\widetilde{C\_R}_B} = \widetilde{C\_R}_B(x_i, x_1)/x_1 + \widetilde{C\_R}_B(x_i, x_2)/x_2 + \ldots + \widetilde{C\_R}_B(x_i, x_n)/x_n$ is the fuzzy neighborhood of x_i and $\widetilde{C\_R}_B(x_i, x_j)$ is the fuzzy cut similarity degree of x_i and x_j with respect to the attribute subset B.

In this model, the lower approximation and the upper approximation can be seen as the 1-cut/strong 0-cut of original counterparts in Dübois's model, respectively [34]. When A degenerates to a crisp set, Definition 2.8 is not a fuzzy set any more, but a crisp subset of the Universe. Or say, Definition 2.8 is not a real fuzzy rough set, but a cut fuzzy rough set.

Definition 2.9. Given a fuzzy decision table $FD = (U, C \cup D)$, $B \subseteq C$ and $U/D = \{X_1, X_2, \ldots, X_r\}$, $\widetilde{C\_R}_B$ is the cut fuzzy similarity relation induced by the attribute subset B. Then the lower and upper approximations of the decision attribute D are defined as

$$\begin{cases} \underline{\widetilde{C\_R}_B D} = \left\{\underline{\widetilde{C\_R}_B X_1}, \ \underline{\widetilde{C\_R}_B X_2}, \ldots, \ \underline{\widetilde{C\_R}_B X_r}\right\} \\ \overline{\widetilde{C\_R}_B D} = \left\{\overline{\widetilde{C\_R}_B X_1}, \overline{\widetilde{C\_R}_B X_2}, \ldots, \overline{\widetilde{C\_R}_B X_r}\right\} \end{cases}.$$

Definition 2.10. Given a fuzzy decision table $FD = (U, C \cup D)$, $B \subseteq C$ and $U/D = \{X_1, X_2, \ldots, X_r\}$, the positive region of D with respect to attribute set B is denoted by $C\_POS_B^U(D) = \bigcup_{i=1}^r \underline{\widetilde{C\_R}_B X_i}$. And the dependence degree of D with respect to attribute set B is defined as $C\_\gamma_B^U(D) = |C\_POS_B^U(D)|/|U|$.

Definition 2.10 shows the relation between the positive region and the lower approximation. It is easy to find that based on the cut fuzzy similarity relation, $C\_POS_B^U(D)$ is a crisp set which makes forward approximation proposed by Qian et al. [34] possible.

Theorem 2.1. *Let $B_1 \subseteq B_2 \subseteq \ldots \subseteq B_n$, then $\{\widetilde{C\_R}_{B_1}, \widetilde{C\_R}_{B_2}, \ldots, \widetilde{C\_R}_{B_n}\}$ be a family of fuzzy binary relations with $\widetilde{C\_R}_{B_1} \succcurlyeq \widetilde{C\_R}_{B_2} \succcurlyeq \ldots \succcurlyeq \widetilde{C\_R}_{B_n}$. we have*

$$C\_POS^U_{B_{i+1}}(D) = C\_POS^U_{B_i}(D) \cup^C \_POS^{U_{i+1}}_{B_{i+1}}(D),$$

where $U_1 = U$ and $U_{i+1} = U - C\_POS^U_{B_i}(D)$.

The equation $C\_POS^U_{B_{i+1}}(D) = C\_POS^U_{B_i}(D) \cup C\_POS^{U_{i+1}}_{B_{i+1}}(D)$ clearly shows the positive region $C\_POS^U_{B_{i+1}}(D)$, which denotes the positive region of the attribute subset B_{i+1} on U with respect to D, composes of two parts $C\_POS^U_{B_i}(D)$ and $C\_POS^{U_{i+1}}_{B_{i+1}}(D)$. Here $C\_POS^U_{B_i}(D)$ denotes the positive region of the attribute set B_i on U with respect to D; $C\_POS^{U_{i+1}}_{B_{i+1}}(D)$ denotes the positive region of the attribute set B_{i+1} on U_{i+1} with respect to D.

Theorem 2.1 shows that with the increment of attributes (from B_i to B_{i+1}), the positive region of B_{i+1} on U could be obtained by updating the positive region of B_i on U with the help of the positive region of B_{i+1} on U_{i+1}. That is to say, we just need to compute the positive region of B_{i+1} on a smaller instance subset (i.e., U_{i+1}) rather than the whole universe U. Thus, a large number of redundant computations are omitted. For more details, please kindly refer to [34].

According to Theorem 2.1, $C\_POS^U_B(D)$ could also be called forward approximation of positive region, just as Qian et al. [34].

Theorem 2.2. *Let $FD = (U, C \cup^D)$, $B \subseteq C$ and $U' = U - \{x | C\_POS^U_B(x) = 1, x \in U\}$. For $\forall a, b \in C - P$,*

If $Sig_1(a, B, D, U) \geq Sig_1(b, B, D, U)$, then $Sig_1(a, B, D, U') \geq Sig_1(b, B, D, U')$.

The improved attribute reduction accelerator based on the forward approximation is then presented as follows.

It is necessary to point out that FA-FPR may contain superfluous attributes because it does not eliminate redundant attributes. FA-FPR is a powerful tool to accelerate attribute reduction. Whereas it is regretted that FA-FPR is designed by using the fuzzy rough sets based on a cut fuzzy similarity relation. The cutting action makes much information, which hidden in the fuzzy attributes, lost. What is more, this kind of accelerator is not really a fuzzy rough accelerator. As a result, it is necessary to design an accelerator based on the classical fuzzy rough sets.

3. PAR: an alternative version of DAR

Our main motivation of this paper is to generalize DAR into an incremental framework. Whereas it is hard to directly accelerate DAR since DAR is designed based on the dependence degree which needs to calculate the average of the positive region of all the instances. As a result, an alternative version of DAR, attribute reduction based on the positive region, shortened by PAR, is designed in this section.

First, we present a theorem which shows that it is feasible by using the measure of the positive region to find a reduct.

Theorem 3.1. *In a fuzzy decision table $FD = (U, C \cup^D)$, $B \subseteq C$ is a reduct of C with regard to D if B satisfies the following two statements:*

(1) $\forall x \in U, POS^U_B(x) = POS^U_C(x)$;
(2) for any $a \in B, \exists x \in U,\ POS^U_{B-\{a\}}(x) \neq POS^U_C(x)$.

Proof.

(1) By Definition 2.7(1), if B is a reduct of R, then $\gamma^U_C = \gamma^U_B$. By the definition of dependence degree, we can get $\gamma^U_C = \gamma^U_B \Leftrightarrow \sum_{x\in U} POS^U_C(x)/|U| = \sum_{x\in U} POS^U_B(x)/|U|$. By Proposition 2.2, we can get if $B \subseteq C$, then $POS^U_B(x) \leq POS^U_C(x)$. If B is a reduct of C and $\exists x \in U$, $POS^U_B(x) < POS^U_C(x)$, then $\sum_{x\in U} POS^U_B(x)/|U| < \sum_{x\in U} POS^U_C(x)/|U|$, which contradicts Definition 2.8(1).

(2) By Definition 2.7(2) and Proposition 2.2, if B is a reduct of C, then for any $r \in B$, $\gamma^U_C \neq \gamma^U_{B-\{r\}} \Leftrightarrow \sum_{x\in U} POS^U_C(x)/|U| \neq \sum_{x\in U} POS^U_{B-\{r\}}(x)/|U| \Leftrightarrow \exists x \in U, POS^U_{B-\{r\}}(x) \neq POS^U_C(x)$. □

Let the positive region be set as the property $\mathbb{P}(C)$. The sufficiency of a reduct is satisfied by $\forall x \in U, POS^U_B(x) = POS^U_C(x)$. And the statement 'for any $a \in B, \exists x \in U,\ POS^U_{B-\{a\}}(x) \neq POS^U_C(x)$' ensures the minimization of a reduct. As a result, Theorem 3.1 verifies that the reduct defined by positive region is not only equal the computational definition of reduct, but also consistent with the conceptual definition of a reduct.

Based on Definition 2.7 and Theorem 3.1, we find that the reduct should satisfy $\gamma^U_C = \gamma^U_B$ or $\forall x \in U, POS^U_B(x) = POS^U_C(x)$, which is very hash and highly sensitive to noisy data. Usually, in real applications, this hash criterion is loosened by setting a threshold $\alpha \in [0,1)$, i.e., $\gamma^U_B + \alpha \geq \gamma^U_C$ or $\forall x \in U$, if $POS^U_B(x) + \alpha \geq POS^U_C(x)$. Then the following result is found.

Theorem 3.2. *If $\forall x \in U$, $POS^U_B(x) + \alpha \geq POS^U_C(x)$, then $\gamma^U_B + \alpha \geq \gamma^U_C$.*

Proof. $\forall x \in U, POS^U_B(x) + \alpha \geq POS^U_C(x) \Rightarrow \sum_{x\in U} (POS^U_B(x) + \alpha)/|U| \geq \sum_{x\in U} POS^U_C(x)/|U| \Rightarrow \sum_{x\in U} POS^U_B(x)/|U| + \alpha \geq \sum_{x\in U} POS^U_C(x)/|U| \Rightarrow \gamma^U_B + \alpha \geq \gamma^U_C$. □

Theorem 3.2 shows that if we make $POS_B^U(x) + \alpha \geq POS_C^U(x)$ for every instance in U, then it is certain that $\gamma_B^U + \alpha \geq \gamma_C^U$.

The significance of an attribute in DAR is measured by the increment of dependence degree, which considers the overall changes of the positive region. If considering each instance, we could obtain a novel way of measuring the significance of one attribute. The significance is defined as follows.

Definition 3.1. Given a fuzzy decision table $FD = (U, C \cup D)$, $B \subseteq C$, $\forall a \in C - B$, $\forall S \subseteq U$ and a threshold $\alpha \in [0, 1)$. We have

(1) $DIS_{B\cup\{a\}}^U(S) = \{x \in S \mid POS_{B\cup\{a\}}^U(x) + \alpha \geq POS_C^U(x)\}$ is called Discernible Instance Set of a in S with respect to B;
(2) The significance of a in B is defined as $Sig_2(a, B, D, S) = |DIS_{B\cup\{a\}}^U(S)|$.

By Definition 3.1, we know that the Discernible Instance Set of a in S is composed of those instances in S which could be distinguished by the attribute subset $B\cup\{a\}$ from all those heterogeneous instances. The significance could be seen as a measure which reflects the number of discernible instances in S when an attribute a is selected into the candidate attribute set. As a result, this significance could be used to select the attribute with the most discernible power.

Proposition 3.1. *Given $FD = (U, C \cup D)$, $B_i \subseteq C$, $S \subseteq U$. Let $B_1 \subseteq B_2 \subseteq \ldots \subseteq B_n$, we have*

(1) $DIS_{B_1}^U(S) \subseteq DIS_{B_2}^U(S) \subseteq \ldots \subseteq DIS_{B_i}^U(S), 1 \leq i \leq n$;
(2) $|DIS_{B_1}^U(S)| \leq |DIS_{B_2}^U(S)| \leq \ldots \leq |DIS_{B_i}^U(S)|, 1 \leq i \leq n$.

Proof. By Proposition 2.2, we can get $\forall x \in S \subseteq U$, $B_i \subseteq B_j \subseteq C$, $POS_{B_i}^U(x) + \alpha \geq POS_C^U(x) \Rightarrow POS_{B_j}^U(x) + \alpha \geq POS_C^U(x)$. □

Proposition 3.1 shows that by using the positive region of every instance, it is feasible to design the reduction construction algorithm by using the addition-deletion strategy [44]. By starting with the empty set, we could successively add attributes until we have a set that is jointly sufficient for preserving $\forall x \in U, POS_B^U(x) = POS_C^U(x)$. As a result, $\exists x \in U$, $POS_B^U(x) + \alpha < POS_C^U(x)$ is set as the stop criterion in our proposed algorithm based on positive region. Furthermore, to ensures the minimization of a reduct, deleting the redundant attribute should also be considered in the proposed algorithm based on positive region. That is, if one attribute satisfying $POS_{red-\{a_i\}}^U = POS_C^U$, it should be deleted.

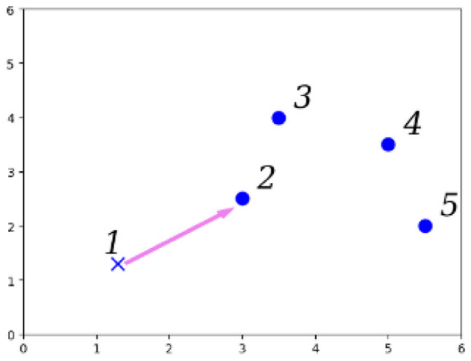
Based on Theorems 3.1, Definition 3.1 and Proposition 3.1, the attribute reduction algorithm based on the positive region, shortened by PAR, is then designed in Algorithm 3.1.

Algorithm 3.1 PAR.

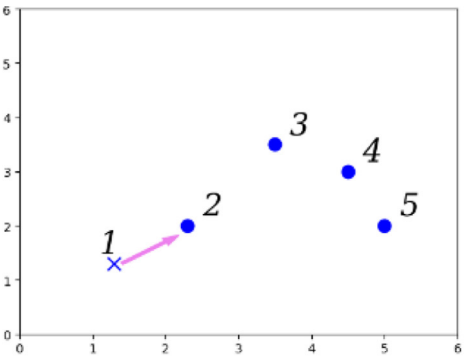
Input: $FD = (U, C \cup D)$
Output: *red*
1. $B \leftarrow \emptyset$;
2. $lef \leftarrow C - B$;
3. $red \leftarrow \emptyset$;
4. **Calculate** $POS_C^U(x)$, $x \in U$;
5. **While** $\exists x \in U$, $POS_B^U(x) + \alpha < POS_C^U(x)$ **Do**
   For each $a \in lef$, calculate $POS_{B\cup\{a\}}^U(x)$, $x \in U$;
   Select $a^* \in lef$ satisfying
   $Sig_2(a^*, B, D, U) = max_{a\in lef}\{Sig_2(a, B, D, U)\}$,
   If $Sig_2(a^*, B, D, U) = 0$, then $a^* = \arg(\max_{a\in lef}\{Sig_1(a, B, D, U)\})$;
   $B \leftarrow B\cup\{a^*\}$,
   $lef \leftarrow lef - \{a^*\}$;
6. **Let** $red \leftarrow B$, $i = 0$;
7. **While** $i < |B|$ **Do**
   Take the ith attribute a_i in B
   if $POS_{red-\{a_i\}}^U + \alpha \geq POS_C^U$ then $red \leftarrow red - \{a_i\}$;
   $i = i + 1$;
8. **Output** *red*;

DAR and PAR are based on the same idea to attribute reduction: to keep the discernibility information invariant. And they use the dependence degree and positive region as the discernibility information measure, respectively. When the threshold of DAR and PAR takes zero at the same time, the reduct of DAR and PAR is equivalent. Because the stop criteria ($\gamma_B^U = \gamma_C^U \Leftrightarrow \forall x \in U, POS_B^U(x) = POS_C^U(x)$ by Theorem 3.1 and Definition 2.7) of both are equivalent.

When the same threshold 'alpha' is not zero, DAR and PAR may find different attribute. because their significances of the attribute are different. PAR chooses $Sig_2(a, B, D, U) = |DIS_{B\cup\{a\}}^U(U)|$ which is the number of discernible instances in U when an attribute a is selected into the candidate attribute set. Whereas DAR chooses $Sig_1(a, B, D, U) = \gamma_{B\cup\{a\}}^U - \gamma_B^U$ which is the increment of dependence degree when an attribute a is selected into the candidate attribute set. When a threshold is added, the $Sig_2(a, B, D, U)$ is different from $Sig_1(a, B, D, U)$. As a result, no matter the threshold of DAR and PAR takes the same threshold $\alpha \in (0, 1)$ or not, the different attributes may be chosen by DAR and PAR. Hence, the reduct obtained by DAR and PAR may not be equivalent. However, it is necessary to point out that by Theorem 3.2 the α reduct found by PAR definitely meets the α reduct's requirement of DAR.



(a) Data distribution on condition attribute set C.



(b) Data distribution on the attribute subset $B_i \subseteq C$.

Fig. 4.1. The illustration of Key Instance Set.

PAR, as another form of DAR, has an obvious advantage that DAR does not have. That is, PAR considers the change of every instance's positive region. Based on this characteristic, it is feasible for PAR to design an attribute reduction accelerator.

4. Fuzzy rough based feature selection accelerator

By the review of FA-FPR in Section 2.4, it is easy to find that the existing fuzzy rough accelerator, i.e., FA-FPR, may make much information lost by cutting. In this section, we propose a novel fuzzy positive region-based accelerator of PAR.

4.1. Fuzzy rough forward approximation and Key Instance Set

The positive region is monotonic with the increment of attributes just as Proposition 4.1.

Proposition 4.1 (Monotonicity of Positive Region)**.** *Given* $B_1 \subseteq B_2 \subseteq \ldots \subseteq B_n$, *then* $POS^U_{B_1}(D) \subseteq POS^U_{B_2}(D) \subseteq \ldots \subseteq POS^U_{B_n}(D)$ *holds.*

Proof. According to Definition 2.4, we have $POS^U_{B_i}(D) = \{POS^U_{B_i}(x) | \forall x \in U\}$, $POS^U_{B_j}(D) = \{POS^U_{B_j}(x) | \forall x \in U\}$, $1 \le i \le j \le n$. By Proposition 2.1(1), we get that if $B_i \subseteq B_j$ then for every x in, we have $POS^U_{B_i}(x) \le POS^U_{B_j}(x)$. Thus, we have $POS^U_{B_i}(D) \subseteq POS^U_{B_j}(D)$. □

By Proposition 4.1, we find the positive region values become larger and larger with the increment of attributes. Furthermore, Proposition 4.1 denotes that B_j has a stronger forward approximation power than B_i, where $1 \le i \le j \le n$, n is the number of all condition attributes.

Definition 4.1. Let $FD = (U, C \cup D)$, $B_i \subseteq C$, a Key Instance Set of B_i on U is defined as $Key(B_i) = \{x \in U \ | POS^U_{B_i}(x) + \alpha < POS^U_C(x)\}$. Where α is a threshold, $\alpha \in [0, 1)$, if α is set to zero, then $Key(B_i) = \{x \in U \ | POS^U_{B_i}(x) < POS^U_C(x)\}$ is called a strong Key Instance Set, else $Key(B_i) = \{x \in U \ | POS^U_{B_i}(x) + \alpha < POS^U_C(x)\}$ is called a weak Key Instance Set.

In real applications, for the robustness of the algorithm, we usually use the weak Key Instance Set.

In Fig. 4.1, the crossing points like "x" denote the positive class and the dot points like "•" denote the negative class. Fig. 4.1(a) shows the instance distribution on C, Fig. 4.1(b) shows the instance distribution on B_i. Fig. 4.1 demonstrates that the minimal distance, i.e., the positive region value of point "1" on B_i maybe not equal to the minimal distance on C any more, but less than. That is to say, B_i cannot discern all instances just as C do. As a result, some new attributes should be supplemented to make the purple lines in Fig. 4.1(a)(b) equal. These explanations show the point "1" is key to select the new attribute. All such kinds of key points "1" make up the Key Instance Set.

Proposition 4.2. *Given* $FD = (U, C \cup D)$, $B_i \subseteq C$, *we have* $DIS^U_{B_i}(U) \cup Key(B_i) = U$;

Proof. By Definition 3.1, we can get $DIS^U_{B_i}(U) = \{x \in U \ | POS^U_{B_i}(x) + \alpha \ge POS^U_C(x)\}$. By Definition 4.1, we can get $Key(B_i) = \{x \in U \ | POS^U_{B_i}(x) + \alpha < POS^U_C(x)\}$. □

Proposition 4.2 shows that the complement of Key Instance Set is Discernible Instance Set.

Similarly, with the positive region's property, Key Instance Set is also monotonic with the increment of attributes. This property corresponds to Theorem 2.1 in Section 2.4. Both of them discuss the monotonicity.

Proposition 4.3 (Monotonicity of Key Instance Set). *Given* $FD = (U, C \cup D)$, $B_i \subseteq C$. *Let* $B_1 \subseteq B_2 \subseteq \ldots \subseteq B_n$, *we have* $Key(B_1) \supseteq Key(B_2) \supseteq \ldots \supseteq Key(B_i)$, $1 \le i \le n$.

Proof. By Proposition 2.2, we can get $POS^U_{B_i}(x) \le POS^U_{B_j}(x)$, $1 \le i \le j \le n$. By Definition 4.2, we can get $\forall x \in Key(B_j)$, $POS^U_{B_j}(x) + \alpha < POS^U_C(x) \Rightarrow POS^U_{B_i}(x) + \alpha < POS^U_C(x)$. This completes the proof. □

Proposition 4.3 presents the monotonicity of Key Instance Set. It shows that the Key Instance Set is gradually decreasing, which reflects the B_i's discernibility power to gradually forward approximate the discernibility ability of all condition attributes. Proposition 4.3 motivates us to propose a fuzzy rough forward approximation accelerator.

Definition 4.2. Let $FD = (U, C \cup D)$, $B_i \subseteq C$. Given $B_1 \subseteq B_2 \subseteq \ldots \subseteq B_i$, and $\alpha \in [0, 1)$, we define a Fuzzy Rough Forward Approximation of B_i on U as

(1) $FRFA(B_i) = \{x \mid POS^U_{B_i}(x) + \alpha \ge POS^U_C(x), x \in Key(B_{i-1})\}$.
(2) $U_{i+1} = U_i - FRFA(B_i)$, where $U_i = U$ and $U_n = \emptyset$.

Definition 4.2 reflects that the B_i could make some instances' fuzzy positive regions weakly reach the maximum, and these instances just belong to $Key(B_{i-1})$. Definition 4.2 also indicates that U_i is getting smaller and smaller with the increment of the attribute subset B_i, which reflects B_i make the fuzzy positive region gradually reach maximum. Actually, U_i is equal to the $Key(B_i)$. Theorem 2.2 in Section 2.4 reveals that FA-FPR also takes a similar action to remove some redundant instances. The key idea of both accelerators is consistent.

Definition 4.2 shows that the fuzzy rough forward approximation is complementary with Key Instance Set. What is more, we find that Key Instance Set has a more interesting property.

Theorem 4.1 (Order Preservation Property). *Given* $FD = (U, C \cup D)$, $B \subseteq C$, $U' = Key(B) = U - \{x \mid POS^U_B(x) + \alpha \ge POS^U_C(x), x \in U\}$ *with* $\alpha \in [0, 1)$. *For* $\forall a, b \in C - B$,

if $Sig_2(a, B, D, U) \ge Sig_2(b, B, D, U)$, then $Sig_2(a, B, D, U') \ge Sig_2(b, B, D, U')$.

Proof.

$$
\begin{aligned}
&Sig_2(a,B,D,U) \ge Sig_2(b,B,D,U) \Leftrightarrow |DIS^U_{B\cup\{a\}}(U)| \ge |DIS^U_{B\cup\{b\}}(U)| \\
&\Leftrightarrow \left|\left\{x \in U \ \middle|\ POS^U_{B\cup\{a\}}(x) + \alpha \ge POS^U_C(x)\right\}\right| \ge \left|\left\{x \in U \ \middle|\ POS^U_{B\cup\{a\}}(x) + \alpha \ge POS^U_C(x)\right\}\right| \\
&\Leftrightarrow \left|\left\{x \in U' \ \middle|\ POS^U_{B\cup\{a\}}(x) + \alpha \ge POS^U_C(x)\right\} \cup \left\{x \in U - U' \ \middle|\ POS^U_{B\cup\{a\}}(x) + \alpha \ge POS^U_C(x)\right\}\right| \\
&\ge \left|\left\{x \in U' \ \middle|\ POS^U_{B\cup\{b\}}(x) + \alpha \ge POS^U_C(x)\right\} \cup \left\{x \in U - U' \ \middle|\ POS^U_{B\cup\{b\}}(x) + \alpha \ge POS^U_C(x)\right\}\right| \\
&\Leftrightarrow \left|\left\{x \in U' \ \middle|\ POS^U_{B\cup\{a\}}(x) + \alpha \ge POS^U_C(x)\right\}\right| + \left|\left\{x \in U - U' \ \middle|\ POS^U_{B\cup\{a\}}(x) + \alpha \ge POS^U_C(x)\right\}\right| \\
&\ge \left|\left\{x \in U' \ \middle|\ POS^U_{B\cup\{b\}}(x) + \alpha \ge POS^U_C(x)\right\}\right| + \left|\left\{x \in U - U' \ \middle|\ POS^U_{B\cup\{b\}}(x) + \alpha \ge POS^U_C(x)\right\}\right| \\
&\Leftrightarrow Sig_2(a,B,D,U') \ge Sig_2(b,B,D,U')
\end{aligned}
$$

□

Actually, U' in Theorem 4.1 is the Key Instance Set. This theorem corresponds to Theorem 2.2 in Section 2.4. Both of them discuss the property of order preservation of significance measure.

Theorem 4.1 is an important result of Key Instance Set, called order preservation property. It clearly shows that the order of significance degrees of attribute computed on the whole universe is consistent with that one computed on the Key Instance Set. That is to say, only the instances in the Key Instance Set are valuable in the process of selecting significant attributes. As a result, it is enough by updating positive region just on Key Instance Set to find the reduct. More importantly, much redundant computation could be omitted and then the computational efficiency could be enhanced.

4.2. Positive-region based attribute reduction accelerator

By computing the significant degree just on the Key Instance Set, it is helpful to accelerate the process of attribute reduction. Whereas there may exist one special case. That is, for any $a \in C - B$, $Sig_2(a, B, D, Key(B))$ may be equal to 0. In such case, any attribute in $C - B$ cannot make the positive region of any instance in $Key(B)$ reaching the maximum. To handle this special case, it is reasonable to choose the attribute with maximal $\sum_{x \in Key(B)} POS^U_{B\cup\{a\}}(x)$ as the most significant one. By these analyses, in this subsection, an accelerator of PAR is designed based on Theorem 4.1 and Definitions 3.1, 4.1 and 4.2.

The accelerated algorithm is then designed in Algorithm 4.1.

Algorithm 4.1 PARA (Positive-region based Attribute Reduction Accelerator).

Input: $FD = (U, C \cup D)$
Output: *red*
1. $B \leftarrow \emptyset, i \leftarrow 1$, $lef \leftarrow C$, and $U_1 \leftarrow U$;
2. **Calculate** $POS_C^U(x)$, $x \in U_1$;
3. **While** $|U_i| \neq 0$ **Do**
   For each $a \in lef$, calculate $POS_{B\cup\{a\}}^U(x)$, $x \in U_i$;
   Select $a^* \in lef$ satisfying
   $Sig_2(a^*, B, D, U_i) = max_{a\in lef}\{Sig_2(a, B, D, U_i)\}$,
   If $Sig_2(a^*, B, D, U_i) = 0$,
   then $a^* = \arg(\max_{a\in lef}\{\sum_{x\in U_i} POS_{B\cup\{a\}}^U(x)\})$;
   $B = B \cup \{a^*\}$,
   $lef = lef - \{a^*\}$;
   $i \leftarrow i + 1$,
   $U_{i+1} \leftarrow U_i - FRFA(B)$;
4. **Let** $red = B$, $i = 0$;
5. **While** $i < |B|$ **Do**
   Take the ith attribute a_i in B
   if $\forall\, x \in U,\ POS_{red-\{a_i\}}^U(x) + \alpha \geq POS_C^U(x)$, then $red = red - \{a_i\}$;
   $i \leftarrow i + 1$;
6. **Output** *red*;

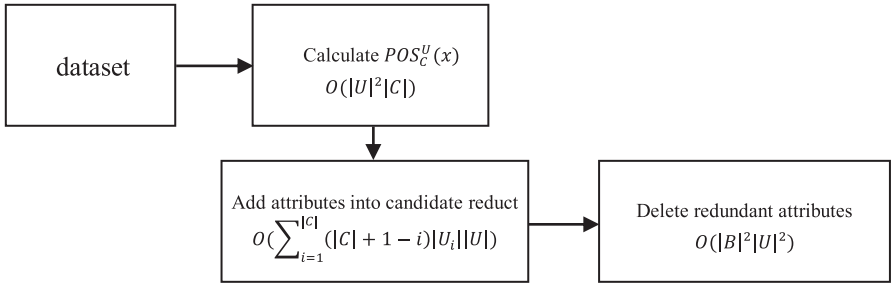


Fig. 4.2. Scalability analysis of PARA.

PARA could quickly find a reduct from a fuzzy decision table. It is likely that two or more attributes will have the same maximal significance degree in the PAR and PARA algorithms. When two or more attributes have the same maximal significance degree, it is feasible to choose any one of them. In this paper, we choose the one with the minimal attribute sequence number. Just as demonstrated in Fig. 4.2, The time complexity on other steps, except Step 5, of PARA is the same as that of PAR. In PARA, Step 5′s time complexity is $O(\sum_{i=1}^{|C|}(|C|+1-i)|U_i||U|)$. Whereas it is $O(\sum_{i=1}^{|C|}(|C|+1-i)|U|^2)$ in PAR, it is obvious that PARA could save much time than PAR.

4.3. Comparison with other algorithms

In this subsection, we detailly compare the four algorithms: DAR, PAR, FA-FPR, and PARA. Because they are closely related to the proposed accelerator in this paper. The similarities and differences between them are listed in Table 4.1.

5. Numerical experiments

In this section, the proposed accelerator, i.e., PARA is compared with the DAR, PAR and the state-of-the-art accelerator, i.e., FA-FPR. We conduct several numerical experiments to compare the performance of DAR and PAR. Furthermore, we compare PARA with the state-of-the-art accelerated algorithm FA-FPR [34].

Experimental setting

1) The selected datasets are described in Table 5.1. ‘stock 2’, ‘stock 3’ are stock data and not public, the rest of datasets in Table 5.1 could be downloaded from [12]. All these datasets are greatly different from instance size and feature number. They are all numerical and have been normalized into the interval [0, 1] with MinMaxScaler.
2) The K-nearest neighbor [10] (k is usually set as 3) and XGBoost [8] are chosen as the classifiers to measure the classification performance of reduction. And 5-fold cross validation is used to guarantee the stability and fairness of classification results.
3) All experiments are conducted on a computer with Ubuntu 16.04.4 LTS, Intel(R) Xeon(R) W-2145 CPU @ 3.70 GHz and 32GB memory. The programming language is C++.

Table 4.1
The comparison among DAR, PAR, FA-FPR and PARA.

	DAR	FA-FPR	PAR	PARA
An accelerator or not	Non-accelerator	Accelerator	Non-accelerator	Accelerator
The stop criterion	Dependence function	Dependence function	Positive region	Positive region
The reduct	–	Different to DAR	–	The same to PAR
The threshold setting place	The stop criterion	The similarity relation	The stop criterion	The stop criterion
The similarity relation	$\tilde{R}_B(x_i,x_j) = \cap_{a\in B}\tilde{R}_a = \min_{a\in B}\tilde{R}_a(x_i,x_j)$	$\widetilde{C\_}R_a(x_i,x_j) = \begin{cases} 1-\lvert a(x_i)-a(x_j)\rvert/\beta, & \lvert a(x_i)-a(x_j)\rvert \le \beta \\ 0, & otherwise \end{cases}$, where $\beta \in (0,1]$	The same to DAR	The same to DAR
The lower approximation	A fuzzy set	A crisp set	A fuzzy set	A fuzzy set
The positive region	A fuzzy set	A crisp set	A fuzzy set	A fuzzy set
Search strategy	Addition-deletion Strategy	Addition-deletion Strategy	Addition-deletion Strategy	Addition-deletion Strategy
Discernibility measure	Dependence function	Dependence function	Positive region	Positive region
The key idea to reduction	Keep the discernibility invariant	Keep the discernibility invariant	Keep the discernibility invariant	Keep the discernibility invariant

Table 5.1
Description of data sets.

	Data sets	Number of features	Number of instances	Number of classes
1	Waveform	21	5000	3
2	Letter	16	20,000	26
3	Credit	23	30,000	2
4	shuttle	9	58,000	7
5	Sensorless	48	58,509	11
6	stock 2	1350	2018	3
7	stock 3	1350	2018	3
8	FPS-5	3208	3600	6

4) DAR, PAR, PARA use the same similarity calculation method. For single attribute a, the similarity of two objects is calculated by $\tilde{R}_a(x_i,x_j) = 1 - (\max(a(x_i),a(x_j)) - \min(a(x_i),a(x_j)))$; For attribute set B, the similarity of two objects is calculated by $\tilde{R}_B(x_i,x_j) = \cap_{a\in B}\tilde{R}_a = \min_{a\in B}\tilde{R}_a(x_i,x_j)$; FA-FPR uses the similarity calculation method offered in the original paper [34]: $\widetilde{C\_}R_a(x_i,x_j) = \begin{cases} 1-\lvert a(x_i)-a(x_j)\rvert/\beta, & \lvert a(x_i)-a(x_j)\rvert \le \beta \\ 0, & otherwise \end{cases}$, where $\beta \in (0,1]$.
5) The size of reduct PARA is anti-monotonic with the increment of the threshold alpha, just as PAR, DAR, and FA-FPR. As a result, the setting of alpha may affect the size of reduct. With the increment of the threshold, the consumption time is decreasing. Whereas the classification performance isn't monotonic with the incremental threshold. To fairly compare with the existing algorithms, we appropriately choose the threshold which makes the reduct size of them comparable. Generally, we attempt many times and then choose a suitable threshold for each algorithm.

5.1. The comparison between DAR and PAR

In this subsection, DAR and PAR are numerically compared. Just as mentioned in Theorems 3.1 and 3.2, PAR is an alternative version of DAR. The stop criterion and attribute selection criterion of PAR is the sufficient condition of those of DAR. One difference between DAR and PAR is that DAR is designed based on dependence degree, whereas PAR is designed based on the positive region. The threshold in PAR and DAR usually takes 0.3 and 0.45 respectively. The comparison of DAR and PAR is presented in Table 5.2.

From Table 5.2, we could observe three facts. One is that the size of reduct is similar; the second is that the running time of both is close on some datasets, but on some other datasets (see the datasets with *) the running time of them are very long; the last one is that both the average ratios of reduct and the ratio of running time closely approximate to 1. The above facts show that PAR, as an alternative version of DAR, has similar results with DAR in most cases. These facts also show that it is urgent to accelerate PAR and DAR since they are too slow on some datasets with a high number of instances,

Table 5.2
The time and attribute selection of DAR and PAR.

Data sets	All attributes	DAR		PAR		Ratio: DAR/PAR	
		Reducts	Time (s)	Reducts	Time(s)	Reducts	Time
waveform	21	14	1059.89	14	937.22	1.0	1.13
letter	16	9	9226.72	9	8821.28	1.0	1.05
credit	23	9	16,386.90	9	13,540.68	1.0	1.21
shuttle	9	7	12,530.16	7	9308.99	1.0	1.35
Sensorless	48	*	>259,200 (>3 ds)	*	>259,200 (>3 ds)	*	*
stock 2	1350	17	20,559.18	17	21,164.93	1.0	0.97
stock 3	1350	18	26,089.95	18	31,442.18	1.0	0.83
FPS-5	3208	*	>259,200 (>3 ds)	*	>259,200 (>3 ds)	*	*
Average	753	12	75,531.60	12	75,451.91	1.0	1.09

* The running time of DAR and PAR on 'Sensorless' and 'FPS-5' is more than 3 days, we have to terminate them and then the results of them are not presented in Table 5.2.

Table 5.3
The reduction and speedup ratio.

Data sets	All attributes	Ratio: PAR/PARA		Ratio: DAR/FA-FPR		Ratio: FA-FPR/PARA	
		Reduct	Time	Reduct	Time	Reduct	Time
waveform	21	1	2.72	1.0	2.61	1.0	1.18
letter	16	1	6.81	0.8	2.13	1.3	3.34
credit	23	1	9.03	0.4	2.61	2.6	4.19
shuttle	9	1	4.75	0.9	2.20	1.1	2.90
Sensorless	48	1	7.67	*	>4.70	0.9	1.63
stock 2	1350	1	8.27	1.0	6.20	1.0	1.30
stock 3	1350	1	10.83	1.0	6.35	1.0	1.41
FPS-5	3208	1	3.50	*	>4.40	1.2	0.80
Average	753	1	6.70	0.8	>3.90	1.3	2.1

such as 'Sensorless'. By analyzing the characteristic of PAR just as mentioned in Section 3, we find that it is feasible for PAR to design an attribute reduction accelerator. And then in Section 4, we propose PARA: an accelerator of PAR.

5.2. Comparison of FA-FPR and PARA

Just as mentioned in Section 2, there exists one type of fuzzy rough attribute reduction accelerator, which is the state-of-the-art fuzzy rough based accelerator. As a result, in this subsection, we compare our proposed PARA with FA-FPR and demonstrate the efficiency of PARA.

Before numerical comparing PARA with FA-FPR, we briefly highlight the similarities and the differences between the Algorithms 4.1 and 2.2.

The similarities of them are listed as follows: (1) Both of them are fuzzy rough accelerators; (2) Both of them share the common key idea to remove the redundant computation; (3) Both of them adopt the forward approximation operator.

The main differences between these two algorithms are listed as follows, more details are listed in Table 4.1: (1) The lower approximation operators of them are different. (2) Forward approximation operators of them are different. These show that their power to remove the redundant instances are different. (3) The stop criteria for them are different. These different stop criteria obtained different results of reduct.

Both PARA and FA-FPR involve the threshold. In this paper, the threshold in FA-FPR is usually set as 0.01, but the threshold in PARA is set as 0.3.

To show the time efficiency of PARA, we divide each selected dataset into ten parts with equal size. The first part is viewed as the first dataset, the combination of the first part and the second part is regarded as the second dataset, and so on. Fig. 5.1. displays the detail change trends of PARA and FA-FPR with the increment of data size. The purple trends grow dramatically, which demonstrates FA-FPR consumes more and more time with the increment of data size, especially on the datasets with a high number of instances (i.e., 'letter', 'credit', 'shuttle'). Comparatively, the red trends grow more slowly than FA-FPR. These show that PARA could save some more time than FA-FPR.

In Table 5.3, we display the reduction and speedup ratios of two accelerated algorithms. In Table 5.3 we not only compare two accelerators, i.e., FA-FPR/PARA, but also compare them with their original counterparts, i.e., PAR/PARA and DAR/FA-FPR.

We observe that both PARA and FA-FPR could accelerate the original algorithm, but the speedup ratio of PAR/PARA is often bigger than that of DAR/FA-FPR. These demonstrate the acceleration power of PARA with regard to PAR is better than that of FA-FPR with regard to DAR. Of course, on some datasets, the speedup ratio of DAR/FA-FPR outperforms that of PAR/PARA. This is because DAR runs really too slow, not because FA-FPR runs very fast.

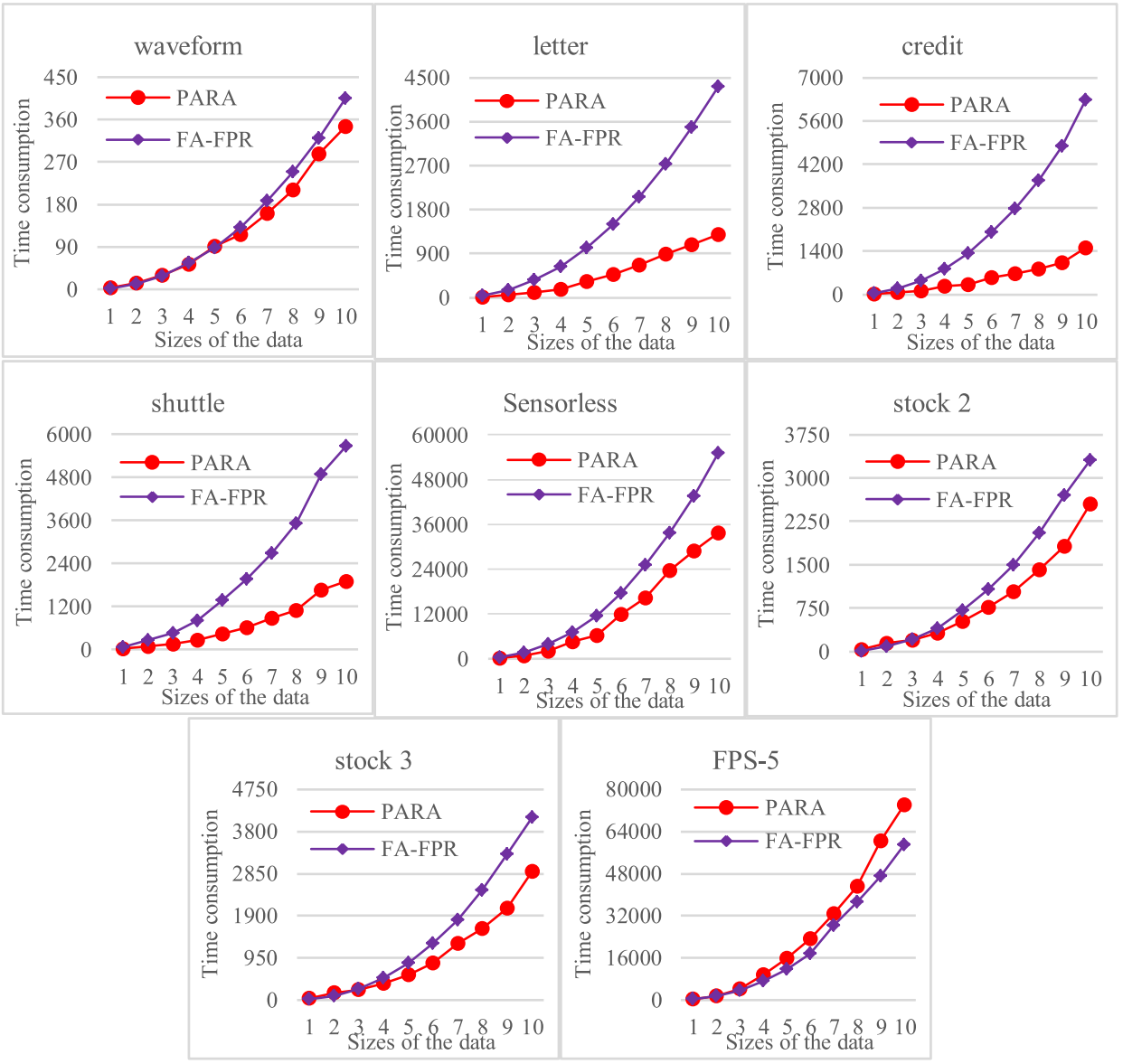


Fig. 5.1. Times of PARA and FA-FPR versus the size of data.

Table 5.3 also shows that the running time of PARA is just half of FA-FPR, which further demonstrates PARA possesses stronger acceleration power than FA-FPR.

Furthermore, 'waveform', 'shuttle', and 'Sensorless' are chosen as examples to demonstrate why PARA is faster than FA-FPR, please kindly see Tables 5.4–5.6. These tables indicate the number of instances, whose positive regions need to be updated within each loop of PARA and FA-FPR respectively.

From Table 5.4, we could observe that the number of instances is decreasing with iterations of PARA and FA-FPR, which indicates both of these two accelerators could reduce the number of instances with iterations. Most noteworthy, in the sixth iteration, the numbers of instances are 2889 and 3965, respectively. It demonstrates that PARA needs to update far fewer instances than FA-FPR in the following iterations. This is the reason why PARA is faster than FA-FPR.

Table 5.5 further displays the difference between PARA and FA-FPR on the dataset 'shuttle', which is with a high number of instances. In the second iteration, PARA just needs to update 759 instances, but FA-FPR needs to update 57,092 instances, which clearly shows the advantage of PARA over FA-FPR on the datasets with a high number of instances. Table 5.6 also demonstrates this fact. In the second iteration, PARA just needs to update 5105 instances, but FA-FPR needs to update 50,379 instances. Tables 5.4–5.6 demonstrate the superiority of PARA, especially on the datasets with a high number of instances.

Table 5.4
The changes of instances of Dataset 'waveform' in each iteration of Algorithm PARA and FA-FPR.

Loop no.	PARA Number of instances	FA-FPR Number of instances
1	5000	5000
2	4996	4999
3	4963	4991
4	4776	4929
5	4031	4620
6	2889	3965
7	1747	2938
8	881	1822
9	404	981
10	187	432
11	79	173
12	33	67
13	12	24
14	3	2
15	1	0

Table 5.5
The changes of instances of Dataset 'shuttle' in each iteration of Algorithm PARA and FA-FPR.

Loop no.	PARA Number of instances	FA-FPR Number of instances
1	58,000	58,000
2	759	57,092
3	160	53,681
4	82	13,879
5	43	4887
6	15	3392
7	3	3172
8	0	3165

Table 5.6
The changes of instances of Dataset 'Sensorless' in each iteration of Algorithm PARA and FA-FPR.

Loop no.	PARA Number of instances	FA-FPR Number of instances
1	58,509	58,509
2	5105	50,379
3	1167	27,459
4	56	7579
5	29	3163
6	22	1008
7	17	361
8	13	83
9	9	38
10	6	20
11	4	10
12	3	6
13	2	4
14	1	0
15	1	0

5.3. Comparison of classification performance

In this subsection, two classifiers are used to show the classification performance of PARA and FA-FPR. Figs. 5.2 and 5.3 display two columns on each dataset, which represents the result of PARA and FA-FPR. It is clear to see that two accelerator algorithms have similar classification performance and standard deviation, which indicates the reduct of PARA and FA-FPR is comparable from the aspect of classification performance.

Above all, the accelerator proposed in this paper can vastly decrease the execution time with no or less classification performance loss, especially on the datasets with a large number of instances. As a result, the proposed method could be used in the real application of feature selection with a large number of instances.

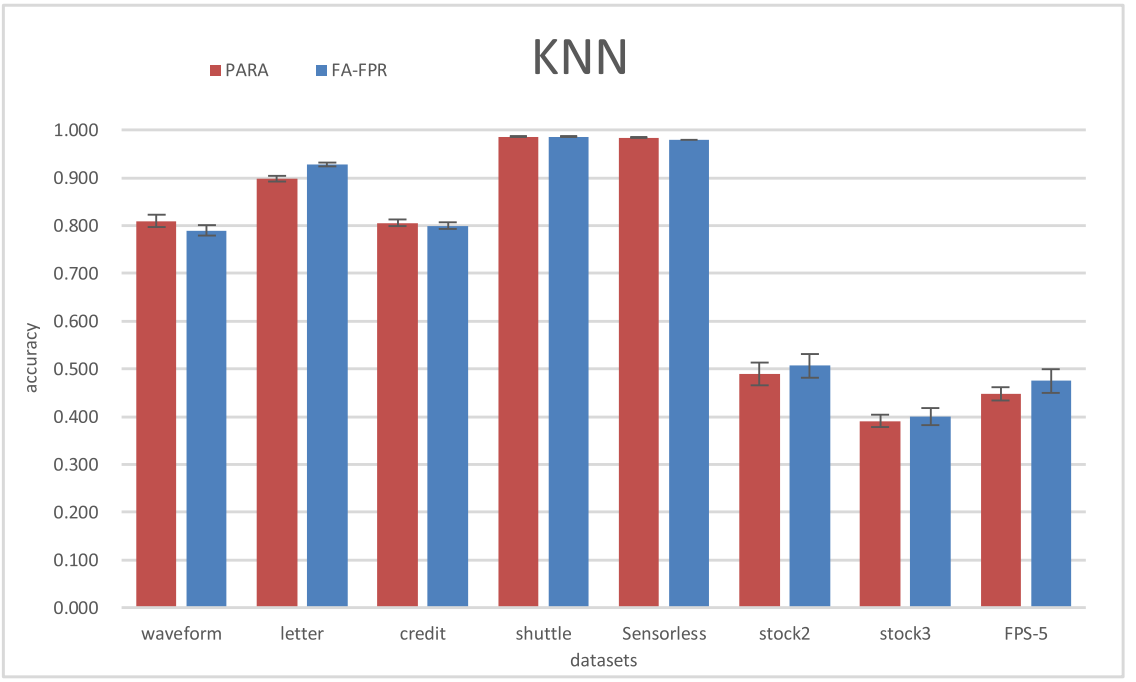


Fig. 5.2. The classification performance on KNN.

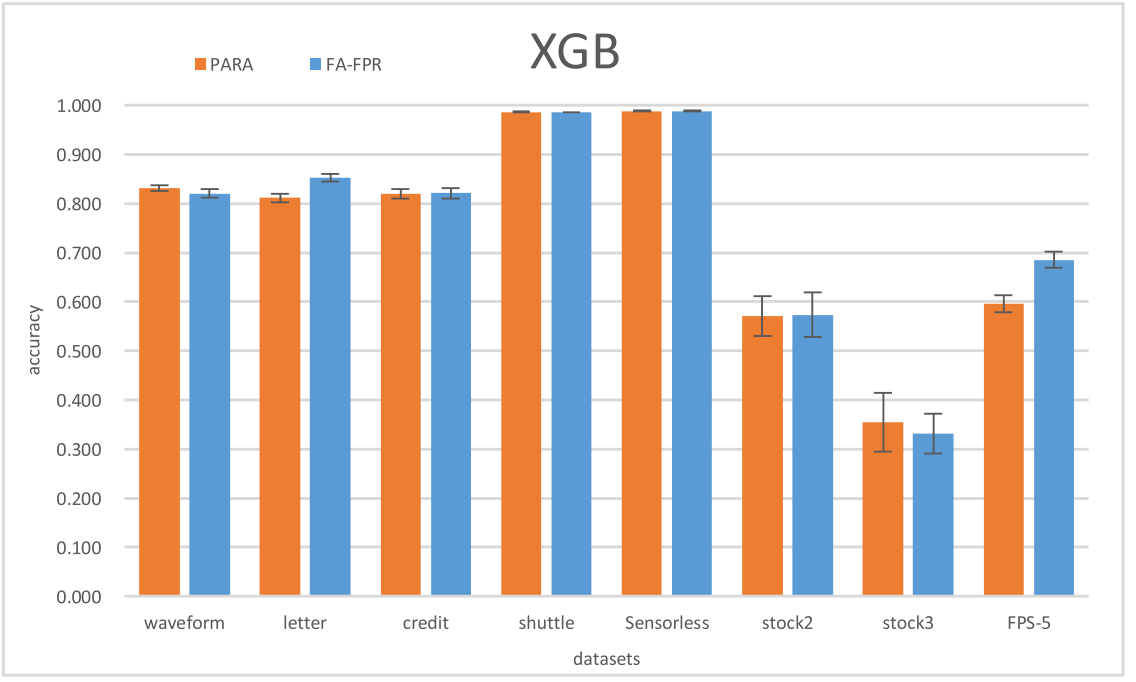


Fig. 5.3. The classification performance on XGBoost.

6. Conclusions

In this paper, we proposed an accelerator, based on fuzzy rough sets, for attribute reduction. The proposed accelerator is based on a strict mathematical foundation, since the monotonicity of Key Instance Set and order preservation property is verified by mathematical reasoning. All these make sure that the reduct found by the accelerator is consistent with the non-accelerated algorithm. Also, based on the experimental comparisons, it is easy to draw a conclusion that the accelerator

proposed in this paper can vastly decrease the execution time with no or less classification performance loss, especially on the datasets with a large number of instances.

Conflict of interest

The authors declared that they have no conflicts of interest to this work. We declare that we do not have any commercial or associative interest that represents a conflict of interest in connection with the work submitted.

Acknowledgments

This work is supported by National Key Research & Develop Program of China (2017YFB1400700), National Key Research & Develop Plan (2018YFB1004401, 2016YFB1000702), NSFC under the grant No. 61732006, 61532021, 61772536, 61772537, 61702522 and NSSFC (No.12\&ZD220), National Basic Research Program of China (973) (No. 2014CB340402), National High Technology Research and Development Program of China (863) (No. 2014AA015204) and the Fundamental Research Funds for the Central Universities, and the Research Funds of Renmin University of China (15XNLQ06). It was partially done when the authors worked in SA Center for Big Data Research in RUC. This Center is funded by a Chinese National 111 Project Attracting.

References

[1] R.B. Bhatt, M. Gopal, On fuzzy rough sets approach to feature selection, Pattern Recognit. Lett. 26 (7) (2005) 965–975.
[2] A.L. Bluma, P. Langley, Selection of relevant features and examples in machine learning, Artif. Intell. 97 (1–2) (1997) 245–271.
[3] D.G. Chen, E.C.C. Tsang, S.Y. Zhao, Attributes reduction with fuzzy rough sets, in: 2007 IEEE Internat. Conf. on Systems, Man, and Cybernetics, 1, 2007, pp. 486–491.
[4] D.G. Chen, Y.Y. Yang, Attribute reduction for heterogeneous data based on the combination of classical and fuzzy rough set models, IEEE Trans. Fuzzy Syst. 22 (5) (2014) 1325–1334.
[5] D.G. Chen, S.Y. Zhao, Local reduction of decision system with fuzzy rough sets, Fuzzy Sets Syst. 161 (13) (2010) 1871–1883.
[6] H.M. Chen, T.R. Li, D. Ruan, J.H. Lin, C.X. Hu, A rough-set based incremental approach for updating approximations under dynamic maintenance environments, IEEE Trans. Knowl. Data Eng. 25 (2013) 274–284.
[7] H.M. Chen, T.R. Li, Y. Cai, C. Luo, H. Fujita, Parallel attribute reduction in dominance-based neighborhood rough set, Inf. Sci. 373 (2016) 351–368.
[8] T.Q. Chen, G. Carlos, Xgboost: a scalable tree boosting system, in: Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, ACM, 2016, pp. 785–794.
[9] Y. Cheng, Dynamic maintenance of approximations under fuzzy rough sets, Int. J. Mach. Learn. Cybern. 9 (12) (2018) 2011–2026.
[10] D. Coomans, D.L. Massart, Alternative k-nearest neighbour rules in supervised pattern recognition: part 1. k-Nearest neighbour classification by using alternative voting rules, Anal. Chim. Acta 136 (1982) 15–27.
[11] A.K. Das, S. Sengupta, S. Bhattacharyya, a group incremental feature selection for classification using rough set theory based on genetic algorithm, Appl. Soft Comput. 65 (2018) 400–411.
[12] D. Dua, E.K. Taniskidou, UCI Machine Learning Repository, University of California, School of Information and Computer Science, Irvine, CA, 2018 https://archive.ics.uci.edu/ml/datasets.html/.
[13] D. Dübois, H. Prade, Rough fuzzy sets and fuzzy rough sets, Int. J. Gen. Syst. 17 (2–3) (1990) 191–209.
[14] I. Guyon, A. Elisseeff, An introduction to variable and feature selection, JMLR 3 (2003) 1157–1182.
[15] B. Hnich, R. Rossi, S.A. Tarim, S. Prestwich, Filtering algorithms for global chance constraints, Artif. Intell. 189 (2012) 69–94.
[16] Q.H. Hu, D.R. Yu, Z.X. Xie, Information-preserving hybrid data reduction based on fuzzy-rough techniques, Pattern Recognit. Lett. 27 (5) (2006) 414–423.
[17] Q.H. Hu, L. Zhang, S. An, D. Zhang, D.R. Yu, On robust fuzzy rough set models, IEEE Trans. Fuzzy Syst. 20 (4) (2012) 636–651.
[18] M.M. Javidi, S. Eskandari, Streamwise feature selection: a rough set method, Int. J. Mach. Learn. Cybern. 9 (4) (2018) 667–676.
[19] S. Joshi, C. Jermaine, Materialized sample views for database approximation, IEEE Trans. Knowl. Data Eng. 20 (3) (2008) 337–351.
[20] N.E. Karabadji, H. Seridi, I. Khelf, N. Azizi, R. Boulkroune, Improved decision tree construction based on attribute selection and data sampling for fault diagnosis in rotating machines, Eng. Appl. Artif. Intell. 35 (2014) 71–83.
[21] K. Kira, L.A. Rendell, A practical approach to feature selection, in: Proceedings of the 9th International Conference on Machine Learning, Morgan Kaufmann, Los Altos, CA, 1992, pp. 249–256.
[22] D.C. Li, W.Z. Wu, On the characterization of fuzzy rough sets based on a pair of implications, Int. J. Mach. Learn. Cybern. 9 (12) (2018) 2081–2092.
[23] K.W. Li, M.W. Shao, W.Z. Wu, A data reduction method in formal fuzzy contexts, Int. J. Mach. Learn. Cybern. 8 (4) (2017) 1145–1155.
[24] J.Y. Liang, J.R. Mi, W. Wei, F. Wang, An accelerator for attribute reduction based on perspective of objects and attributes, Knowl. Based Syst. 44 (1) (2013) 90–100.
[25] J.Y. Liang, F. Wang, C.Y. Dang, Y.H. Qian, Incremental approach to feature selection based on rough set theory, IEEE Trans. Knowl. Data Eng. 26 (2) (2014) 294–308.
[26] S.J. Liao, Q.X. Zhu, Y.H. Qian, G.P. Lin, Multi-granularity feature selection on cost-sensitive data with measurement errors and variable costs, Knowl. Based Syst. 158 (2018) 25–42.
[27] J.J. Niu, C.C. Huang, J.H. Li, M. Fan, Parallel computing techniques for concept-cognitive learning based on granular computing, Int. J. Mach. Learn. Cybern. 9 (11) (2018) 1785–1805.
[28] Z. Pawlak, Rough Sets: Theoretical Aspects of Reasoning about Data, Kluwer Academic Publishers, Boston, 1991.
[29] Z. Pawlak, J.W. Grzymala-Busse, R. Slowiski, W. Ziako, Rough sets, Commun. ACM 38 (11) (1995) 89–95.
[30] H. Peng, F. Long, C. Ding, Feature selection based on mutual information: criteria of max-dependency, max-relevance, and min-redundancy, IEEE Trans. Pattern Anal. Mach. Intell. 27 (8) (2005) 1226–1238.
[31] J. Qian, D.Q. Miao, Z.H. Zhang, X.D. Yue, Parallel attribute reduction algorithms using MapReduce, Inf. Sci. 279 (2014) 671–690.
[32] J. Qian, M. Xia, X.D. Yue, Parallel knowledge acquisition algorithms for big data using MapReduce, Int. J. Mach. Learn. Cybern. 9 (6) (2018) 1007–1021.
[33] Y.H Qian, J.Y. Liang, W. Pedrycz, C.Y. Dang, Positive approximation: an accelerator for feature reduction in rough set theory, Artif. Intell. 174 (9) (2010) 597–618.
[34] Y.H Qian, Q. Wang, H.H. Cheng, J.Y. Liang, C.Y. Dang, Fuzzy-rough feature selection accelerator, Fuzzy Sets Syst. 258 (C) (2015) 61–78.
[35] 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 Recognit. 44 (8) (2011) 1658–1670.
[36] M.W. Shao, K.W. Li, Attribute reduction in generalized one-sided formal contexts, Inf. Sci. 378 (1) (2017) 317–327.
[37] R.W. Swiniarski, A. Skowron, Rough set methods in feature selection and recognition, Pattern Recognit. Lett. 24 (6) (2003) 833–849.

[38] E.C.C. Tsang, D.G. Chen, D.S. Yeung, X.Z. Wang, J. Lee, Attributes reduction using fuzzy rough sets, IEEE Trans. Fuzzy Syst. 16 (5) (2008) 1130–1141.
[39] C.Z. Wang, Q. He, M.W. Shao, Q.H. Hu, Feature selection based on maximal neighborhood discernibility, Int. J. Mach. Learn. Cybern. 9 (11) (2018) 1929–1940.
[40] W. Wei, P. Song, J.Y. Liang, X.Y. Wu, Accelerating incremental attribute reduction algorithm by compacting a decision table, Int. J. Mach. Learn. Cybern. (2018) 1–19.
[41] Y.Y. Yang, D.G. Chen, H. Wang, E.C.C. Tsang, D.L. Zhang, Fuzzy rough set based incremental attribute reduction from dynamic data with sample arriving, Fuzzy Sets Syst. 312 (2017) 66–86.
[42] Y.Y. Yang, D.G. Chen, H. Wang, X.Z. Wang, Incremental perspective for feature selection based on fuzzy rough sets, IEEE Trans. Fuzzy Syst. 26 (3) (2018) 1257–1273.
[43] Y.Y. Yao, Y. Zhao, J. Wang, On reduct construction algorithms, Trans. Comput. Sci. 2 (2008) 100–117.
[44] Y.Y. Yao, The two sides of the theory of rough sets, Inf. Sci. 80 (2015) 67–77.
[45] D.S. Yeung, D.G. Chen, E.C.C. Tsang, J.W.T. Lee, X.Z. Wang, On the generalization of fuzzy rough sets, IEEE Trans. Fuzzy Syst. 13 (3) (2005) 343–361.
[46] L.A. Zadeh, Fuzzy sets, Inf. Control 8 (3) (1965) 338–353.
[47] H. Zhao, P. Wang, Q.H. Hu, Cost-sensitive feature selection based on adaptive neighborhood granularity with multi-level confidence, Inf. Sci. 366 (2016) 134–149.
[48] S.Y. Zhao, H. Chen, C.P. Li, M.Y. Zhai, X.Y. Du, RFRR: robust fuzzy rough reduction, IEEE Trans. Fuzzy Syst. 21 (5) (2013) 825–841.
[49] S.Y. Zhao, X.Z. Wang, D.G. Chen, E.C.C. Tsang, Nested structure in parameterized rough reduction, Inf. Sci. 248 (2013) 130–150.