Information Sciences 181 (2011) 5457–5467

Contents lists available at ScienceDirect

Information Sciences

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

Rule learning for classification based on neighborhood covering reduction

Yong Du, Qinghua Hu *, Pengfei Zhu, Peijun Ma

Harbin Institute of Technology, Harbin 150001, PR China

ARTICLE INFO

Article history:
Received 5 October 2010
Received in revised form 24 May 2011
Accepted 20 July 2011
Available online 28 July 2011

Keywords:
Rough set
Neighborhood covering
Covering reduction
Relative covering reduction
Rule learning

ABSTRACT

Rough set theory has been extensively discussed in the domain of machine learning and data mining. Pawlak's rough set theory offers a formal theoretical framework for attribute reduction and rule learning from nominal data. However, this model is not applicable to numerical data, which widely exist in real-world applications. In this work, we extend this framework to numerical feature spaces by replacing partition of universe with neighborhood covering and derive a neighborhood covering reduction based approach to extracting rules from numerical data. We first analyze the definition of covering reduction and point out its advantages and disadvantages. Then we introduce the definition of relative covering reduction and develop an algorithm to compute it. Given a feature space, we compute the neighborhood of each sample and form a neighborhood covering of the universe, and then employ the algorithm of relative covering reduction to the neighborhood covering, thus derive a minimal covering rule set. Some numerical experiments are presented to show the effectiveness of the proposed technique.

1. Introduction

In the last decade, we have witnessed great progress in rough set theory and its applications [7,20,22,25,33,34]. Since Pawlak's initiative work in the beginning of 1980s, this theory has become a powerful tool to deal with imperfect and inconsistent data, and extract useful knowledge from a given dataset [3,4,12,23,24,30,32].

The key idea of rough set theory is to divide the universe into a collection of subsets (called information granules or elemental concepts) according to the relation between objects, and then use these subsets to approximate arbitrary subsets of the universe. Thus there are two ways to generalize the model introduced by Pawlak: the way to generate information granules used in approximation and the way to approximate concepts in the universe.

First, Pawlak considered equivalence relations and generated a partition of the universe. The derived model can be used to deal with datasets described with nominal features. However, in real-world applications the relations between objects are much more complex than equivalence relations. Similarity relation, neighborhood relation and dominance relation are usually used in reasoning. Based on this observation, similarity relation rough sets [26], neighborhood rough sets [18,15], and dominance rough sets [1,10,11] were developed one by one. The family of subsets induced with these relations forms a covering of the universe, instead of a partition, thus all these models can be categorized into covering rough sets [41]. The general properties of covering rough sets were widely discussed in these years [44,45]. In addition, in the above models, the information granules used in approximation are crisp, whereas we usually employ fuzzy concepts to approximately describe a subset of objects. In this context, fuzzy information granules are preferred. As a result, fuzzy rough sets and rough fuzzy sets were developed [7]. As to rough fuzzy sets, crisp information granules are used to approximate fuzzy concepts. In the case of fuzzy rough sets we use fuzzy granules to approximate rough concepts. All the above models

* Corresponding author.
E-mail address: huqinghua@hit.edu.cn (Q. Hu).

0020-0255/$ - see front matter © 2011 Elsevier Inc. All rights reserved.
doi:10.1016/j.ins.2011.07.038

simulate a certain way in that human reasons and makes decisions. So these models satisfy some requirement of real-world applications.

Second, different approximating ways were developed in computing lower and upper approximations. In Pawlak's models, the information granules are grouped into the lower approximation of a set if all the elements in these granules belong to the same set and the information granules are grouped into the upper approximation of the set if one of the elements in these granules belongs to it. These definitions are too strict to bear the influence of noise, which leads to the algorithms developed on this model sensitive to noisy information. In order to overcome this problem, decision-theoretic rough sets [38,39], variable precision rough sets (VPRS) [46], Bayesian rough sets [27] and probabilistic rough sets [36,37] were designed. In the meanwhile, several fuzzy approximation operators were proposed based on different triangular norms, triangular conorms and implication operators [14,16,17,20,25,33,40,42].

Among these generalizations, covering rough sets and fuzzy rough sets have gained much attention from the domains of machine learning and uncertainty reasoning. In this work we will focus on the theoretic foundation of relative covering reduction for covering approximation spaces and develop a rule learning algorithm based on neighborhood covering reduction. In fact covering reduction has been extensively discussed in literature. Two classes of covering reduction have been reported so far. The first one is to reduce the redundant elements in a covering by search the minimal description of objects [35,44,45]. In this way, the redundant representation of a covering approximation space is reduced. The second one is to reduce the redundant covering in a family of coverings so as to reduce redundant attributes, where each attribute segments the universe into a covering of the universe and some coverings are redundant for computing lower and upper approximations [6,29]. In order to discern them, we call them covering element reduction and covering set reduction, respectively. Essentially, covering element reduction is designed for rule learning, while covering set reduction is developed for attribute reduction.

As to covering element reduction, the current work does not consider decision attributes. In order to keep the discernibility of a covering approximation space, one requires computing the minimal description of objects. However, if an decision attribute is considered, the objective of covering element reduction is to find the maximal description of each object, in the meanwhile, the lower and upper approximations of decisions keep invariant. In this way, the derived decision rules would have the greatest generalization power. We call this operation relative covering element reduction. This issue was studied by Hu and Wang in 2009 [13]. However, this work did not systematically discuss the properties of relative covering element reduction. It also did not show how to generate a covering from a given dataset. In addition, no empirical analysis was presented in their paper.

In this work, we will redefine the relative covering element reduction and analyze the difference between covering element reduction and relative covering element reduction in detail. These different definitions form distinct directions of applications. The objective of covering element reduction is to preserve the capability of the original covering in approximating any new information granule. However, relative covering element reduction is to derive a new compact covering such that the new covering has the same capability in approximating the classification. Based on this observation, we design a rule learning algorithm based on relative covering reduction. We compute the neighborhood of each sample to form a covering of the universe and the size of neighborhoods depends on the classification margin of samples [9]. And then we develop an algorithm to reduce the elements of neighborhood covering and generate a set of rules from data. Finally we show some numerical experiments to evaluate the proposed theoretic and algorithmic framework. In fact, Pawlak introduced a rule learning algorithm through covering reduction in [22]. However, that algorithm was developed for nominal data. It does work in numerical applications. Our algorithm can be used in more complex tasks.

The remainder of this paper is organized as follows. First, we will introduce the preliminaries of covering approximation spaces and covering reduction in Section 2.Then we will show the definition of relative covering element reduction and discuss its properties in Section 3. In Section 4, we design a rule learning algorithm based neighborhood covering reduction. The numerical experiments are presented in Section 5. Finally, conclusions and future work are given in Section 6.

2. Preliminaries of covering approximation spaces and covering reduction

In this section, we review the basic knowledge about covering rough sets and covering reduction.

Definition 1 [41]. Let U be a nonempty and finite set of objects, where $U = \{x_1,x_2,\ldots,x_n\}$ is called a universe of discourse. $C = \{X_1,X_2,\ldots,X_k\}$ is a family of nonempty subsets of U, and $\cup_{i=1}^{k} X_i = U$. We say C is a covering of U, X_i is a covering element, and the ordered pair $\langle U,C \rangle$ is a covering approximation space.

In Pawlak's approximation space, one condition is added to the family of subsets that $X_i \cap X_j = \emptyset$ if $i \neq j$, thus C forms a partition of the universe. However different subsets of objects usually overlap in real-world application. For example, if we consider the extensions of the concepts we use everyday, we can find that most of the concepts have overlapped extensions with other words, which tells us that the basis information granules we use in reasoning forms a covering of the objects in the world, instead of a partition. Neighborhood relations, similarity relations and order relations are widely used. These relations generate a covering of the universe, instead of a partition. Thus covering approximation spaces are more general than Pawlak's partition approximation spaces and can handle more complex tasks.

Definition 2 [5]. Let $\langle U,C\rangle$ be a covering approximation space, and $x \in U$. The family

$$Md(x) = \{X \in C : x \in X \wedge \forall S \in C(x \in S \wedge S \subseteq X \Rightarrow S = X)\}$$

is called the minimal description of x.

The minimal description of x is the subset of covering elements containing x and these elements are not contained by other covering elements. Thus these covering elements are the minimal ones associated with x.

Definition 3 (*[43–45]*). Let $\langle U,C\rangle$ be a covering approximation space. $X \subseteq U$ is an arbitrary subset of the universe. The covering lower and upper approximations of X are defined as

$$\begin{aligned}
&\underline{C}X = \cup\{X_i \in C | X_i \subseteq X\},\\
&\overline{C_1}X = \underline{C}X \cup \bigcup\{Md(x) | x \in X - \underline{C}X\},\\
&\overline{C_2}X = \cup\{X_i \in C | X_i \cap X \neq \emptyset\},\\
&\overline{C_3}X = \cup\{Md(x) | x \in X\},\\
&\overline{C_4}X = \underline{C}X \cup \bigcup\{X_i \in C | X_i \cap (X - \underline{C}X) \neq \emptyset\},
\end{aligned}$$

where $\underline{C}X$ is the covering lower approximation, and $\overline{C_1}X, \overline{C_2}X, \overline{C_3}X$ and $\overline{C_4}X$ are the first, the second, the third and the fourth types of upper approximations, respectively [45].

In 2008, Hu, Yu and Xie introduced a point-wise neighborhood covering and gave the definition of neighborhood rough sets based on neighborhood [18].

Definition 4. Let U be the universe of discourse, $\delta(x)$ is the neighborhood of x, where $\delta(x) = \{x_i : \Delta(x,x_i) \leqslant \delta\}$, Δ is a distance function. Then $\mathrm{N} = \{\delta(x_1),\delta(x_2),\ldots,\delta(x_n)\}$ forms a covering of U, we call $\langle U,\mathrm{N}\rangle$ a neighborhood covering approximation space. Let $X \subseteq U$ be a subset of the universe. The lower and upper approximations of X with respect to $\langle U,\mathrm{N}\rangle$ are defined as

$$\begin{aligned}
&\underline{\mathrm{N}}X = \{x \in U : \delta(x) \subseteq X\},\\
&\overline{\mathrm{N}}X = \{x \in U : \delta(x) \cap X \neq \emptyset\}.
\end{aligned}$$

As the elements of a covering overlap, some of them may be redundant for keeping the discernibility of the covering approximation space. In order to extract the essential information of the approximation space, it is expected to reduce the redundant elements of the covering.

Definition 5 [44]. Let $\langle U,C\rangle$ be a covering approximation space, $X \in C$. If $\exists X_1, X_2, \ldots, X_l \in C - \{X\}$, such that $X = \cup_{i=1}^{l} X_i$, then we say X is a reducible element of C; otherwise, we say X is a irreducible. In addition, if each element of C is irreducible, we say C is irreducible; otherwise, we say C is reducible.

Definition 6 [44]. Let $\langle U,C\rangle$ be a covering approximation space, C' is a covering derived from C by reducing the redundant elements, and C' is irreducible. We say C' is a reduct of C, denoted by $reduct(C)$.

In classification and regression learning, we are usually confronted with the task of approximating some concepts with a covering. The above definitions are not involved with any decision attribute. Now we consider the problem of using coverings to approximate a set of decision concepts.

Definition 7 [6]. Let $C = \{X_1,X_2,\ldots,X_k\}$ be a covering of U. For $\forall x \in U$, we set that $C_x = \cap\{X_j : X_j \in C, x \in X_j\}$. We say $Cov(C) = \{C_x : x \in U\}$ is the induced covering of C.

Definition 8 [6]. Let $\mathbb{C} = \{C_1, C_2, \ldots, C_m\}$ be a family of coverings of U. For $\forall x \in U$, we set $\mathbb{C}_x = \cap\{C_{ix} : C_{ix} \in Cov(C_i),\ x \in C_{ix}\}$. Then $Cov(\mathbb{C}) = \{\mathbb{C}_{\mathbb{x}} : \mathbb{x} \in \mathbb{U}\}$ is also a covering of U. We call it the induced covering of $\mathbb{C}$.

Definition 9 [6]. Let $\mathbb{C} = \{C_1, C_2, \ldots, C_m\}$ be a family of coverings of U, D be a decision attribute. U/D is a partition of U induced by D. We call $\langle U, \mathbb{C}, D\rangle$ a covering decision system. If $\forall x \in U$, $\exists D_j \in U/D$ such that $\mathbb{C}_x \subseteq D_j$, then decision system $\langle U, \mathbb{C}, D\rangle$ is consistent, denoted by $Cov(\mathbb{C}) \prec U/D$; otherwise, $\langle U, \mathbb{C}, D\rangle$ is inconsistent.

In [6], the authors introduced a new concept of the induced covering of a covering. In fact, if we derive a covering C of the universe according to the provided knowledge, then the knowledge has been used. We have no additional knowledge for generating the induced covering $Cov(C)$. The subsequent approximation operation should be processed on C, instead of $Cov(C)$.

Definition 10 [6]. Let $\langle U, \mathbb{C}, D\rangle$ be a consistent covering decision system. For $C_i \in \mathbb{C}$, if $Cov(\mathbb{C} - \{C_i\}) \prec U/D$, we say C_i is superfluous in $\mathbb{C}$ with respect to D; otherwise, C_i is said to be indispensable. If $\forall C \in \mathbb{C}$ is indispensable, we say $\mathbb{C}$ is independent. Assume $\mathbb{Q} \subseteq \mathbb{C}, Cov(\mathbb{Q}) \prec U/D$ and $\mathbb{Q}$ is independent, we say $\mathbb{Q}$ is a relative reduct of $\mathbb{C}$, denoted by $reduct_D(\mathbb{C})$.

Assume that we have N features $\{f_i\}_{i=1}^N$ describe the objects. According to the information provided by each feature, we can generate a covering of the universe. Then we can get N coverings $\mathbb{C} = \{C_i\}_{i=1}^N$. Among these attributes, some of them are not related to the decision and irrelevant. Thus removing the coverings induced with these features does not have impact on the approximation power of the system. The above definition shows us which attributes can be reduced. In classification learning, we require removing not only the redundant attributes, but also the redundant elements in a covering in order to obtain a concise description of classification rules.

Definition 11 [13]. Let $\langle U,C\rangle$ be a covering approximation space. $X \subseteq U$ and $X_i \in C$. If there exists $X_j \in C$, such that $X_i \subseteq X_j \subseteq X$, we say X_i is a relatively reducible element of C with respect to X; otherwise, X_i is relatively irreducible. If all the elements in C are relatively irreducible, we say C is relatively irreducible.

Comparing Definitions 5 and 11, we can see that in covering element reduction, the small covering elements are selected and the large one is reduced. However, the large covering elements are preserved and the small one is removed in relative covering element reduction. Why they are defined so. As we know, in covering element reduction, we should keep the finest granularity of the covering approximation space for keeping the approximation power of the original system. In this case, the small granules are preferred as they may give finer approximation of arbitrary subsets of the universe, whereas in the context of relative covering element reduction, the large information granules, namely, the large covering elements used to approximate decision classes are preferred because these granules have more powerful generalization ability, and the number of the derived rules would be less.

3. Relative neighborhood covering reduction

As to classification and regression learning, we are given a set of samples described with a $n \times m$ matrix $[x_{ij}]_{n\times m}$, where x_{ij} is the j'th feature value of sample x_i. In addition, each sample has an output y_i. Considering a classification task, y_i is the class label of Sample x_i. Formally, the data set can be written as $\langle U,A,D\rangle$, where $U = \{x_1,\ldots,x_n\}$, $A = \{a_1,\ldots,a_m\}$ is the set of attributes, $D = \{d\}$ is the decision attribute.

No matter the attributes are nominal, numerical or fuzzy, we can define a distance function between samples. For example, if the features are mixed with nominal and numerical features, the distance function can be defined as [31]

$$HEOM(x,y) = \sqrt{\sum_{i=1}^{m} w_{a_i} \times d_{a_i}^2(x_{a_i}, y_{a_i})},$$

where m is the number of attributes, w_{a_i} is the weight of attribute $a_i, d_{a_i}(x,y)$ is the distance between samples x and y with respect to attribute a_i, computed as

$$d_{a_i}(x,y) = \begin{cases} 1 \text{ if the attribute value of } x \text{ or } y \text{ is unknown}, \\ overlap_a(x,y), \quad \text{if } a \text{ is a nominal attribute}, \\ rn\_diff_a(x,y), \quad \text{if } a \text{ is a numerical attribute}. \end{cases}$$

Here $overlap(x,y) = \begin{cases} 0, & \text{if } x = y \\ 1, & \text{otherwise} \end{cases}$ and $rn\_diff_a(x,y) = \frac{|x-y|}{\max_a - \min_a}$. Now according to the closeness between the objects in feature spaces, the objects can be granulated into different subsets. The samples which are close to each other form an information granule, called a neighborhood granule, which is formulated as follows.

Definition 12. Given $x_i \in U$, the neighborhood $\mathbb{N}(x_i)$ of x_i in feature space is defined as

$$\mathbb{N}(x_i) = \{x_j \in U : \Delta(x_i, x_j) \leqslant \delta\},$$

where Δ is a distance function and δ is a parameter dependent on x_i.

Then the family of neighborhoods forms a covering of the universe. Note that δ is a constant in [18,19]. In this work, the sizes of neighborhoods of different samples are different and they are calculated according to the location of samples in feature spaces. Here we set δ as the classification margin of samples [9].

Definition 13 [9]. Given $\langle U,A,D\rangle$, $x \in U$. $NH(x) \in U - \{x\}$ is the nearest sample of x within the same class of x, called the nearest hit of x (Provided there is only one sample in the class of x, we set $NH(x) = x$). $NM(x)$ is the nearest sample of x out of the class of x, called the nearest miss of x. Then the classification margin of x is defined as

$$m(x) = \Delta(x, NM(x)) - \Delta(x, NH(x)).$$

The margin of a sample x reflects how much the features of x can be corrupted by noise before x is misclassified. It is remarkable that $m(x)$ may be less than zero. If so, this sample will be misclassified by the nearest neighbor rule. In this case we set $m(x) = 0$, and $\mathbb{N}(x_i) = \{x_j : \Delta(x_i, x_j) = 0\}$. If there are not two samples which belong to different classes and take the same feature values, then the neighborhood of each sample consistently belongs to one of the decision classes.

Certainly, the size of neighborhood can be specified in other ways, which lead to different neighborhood coverings.

The family of neighborhood $\mathbb{N} = \{\mathbb{N}(x_1), \mathbb{N}(x_2), \ldots, \mathbb{N}(x_n)\}$ generates a point-wise covering of the universe. Now we call $\langle U, \mathbb{N}\rangle$ a neighborhood covering approximation space and $\langle U, \mathbb{N}, D\rangle$ a neighborhood covering decision system. The neighborhood granules are used to approximate the decision regions.

Definition 14. Let $\langle U, \mathbb{N}, D\rangle$ be a neighborhood covering decision system. $X \subseteq U$ is an arbitrary subset of U. The lower and upper approximations of X in $\langle U, \mathbb{N}, D\rangle$ are defined as

$$\underline{\mathbb{N}_1}X = \{x \in U : \mathbb{N}(x) \subseteq X\}, \quad \underline{\mathbb{N}_2}X = \cup\{\mathbb{N}(x) : \mathbb{N}(x) \subseteq X\}$$
$$\overline{\mathbb{N}_1}X = \{x \in U : \mathbb{N}(x) \cap X \neq \emptyset\}, \quad \overline{\mathbb{N}_2}X = \cup\{\mathbb{N}(x) : \mathbb{N}(x) \cap X \neq \emptyset\},$$

where $\underline{\mathbb{N}_1}X$ and $\underline{\mathbb{N}_2}X$ are called Type 1 and Type 2 covering lower approximations, respectively, and $\overline{\mathbb{N}_1}X$ and $\overline{\mathbb{N}_2}X$ are Type 1 and Type 2 covering upper approximations, respectively.

Comparing Definitions 3 and 14, we can see that $\underline{\mathbb{N}_2}X = \underline{C}X$, $\overline{\mathbb{N}_2}X = \overline{C_2}X$. $\underline{\mathbb{N}_1}X$ and $\overline{\mathbb{N}_1}X$ are defined based on objects, instead of covering elements, while all the definitions in Definition 3 are computed with covering elements. However,$\underline{\mathbb{N}_1}X$ and $\overline{\mathbb{N}_1}X$ are the same as Definition 4. In order to compare and understand them, here we give the two types of definitions together.

If the neighborhood covering $\mathbb{N}$ is also a partition of the universe, then $\underline{\mathbb{N}_1}X = \underline{\mathbb{N}_2}X$ and $\overline{\mathbb{N}_1}X = \overline{\mathbb{N}_2}X$. However, usually $\mathbb{N}$ is not a partition. In this case the two definitions are different.

Example 1. Let $U = \{x_1, x_2, x_3, x_4, x_5\}$. $f(x_1, a) = 0.1$, $f(x_2, a) = 0.2$, $f(x_3, a) = 0.3$, $f(x_4, a) = 0.4$, $f(x_5, a) = 0.5$. Assume the samples are divided into two classes. $d_1 = \{x_1, x_2, x_3\}$ and $d_2 = \{x_4, x_5\}$. We compute the margins of samples. $m(x_1) = 0.2$, $m(x_2) = 0.1$, $m(x_3) = 0$, $m(x_4) = 0$, $m(x_5) = 0.1$. If the size of neighborhood is specified as its margin, we have $\mathbb{N}(x_1) = \{x_1, x_2, x_3\}$, $\mathbb{N}(x_2) = \{x_1, x_2, x_3\}$, $\mathbb{N}(x_3) = \{x_3\}$, $\mathbb{N}(x_4) = \{x_4\}$, $\mathbb{N}(x_5) = \{x_4, x_5\}$. $\underline{\mathbb{N}_1}d_1 = \{x_1, x_2, x_3\}$, $\underline{\mathbb{N}_2}d_1 = \{x_1, x_2, x_3\}$, $\overline{\mathbb{N}_1}d_1 = \{x_1, x_2, x_3\}$ and $\overline{\mathbb{N}_2}d_1 = \{x_1, x_2, x_3\}$. However, if we set $\delta = 0.15$, then $\mathbb{N}(x_1) = \{x_1, x_2\}$, $\mathbb{N}(x_2) = \{x_1, x_2, x_3\}$, $\mathbb{N}(x_3) = \{x_2, x_3, x_4\}$, $\mathbb{N}(x_4) = \{x_3, x_4, x_5\}$, $\mathbb{N}(x_5) = \{x_4, x_5\}$. $\underline{\mathbb{N}_1}d_1 = \{x_1, x_2\}$, $\underline{\mathbb{N}_2}d_1 = \{x_1, x_2, x_3\}$, $\overline{\mathbb{N}_1}d_1 = \{x_1, x_2, x_3, x_4\}$ and $\overline{\mathbb{N}_2}d_1 = \{x_1, x_2, x_3, x_4, x_5\}$.

Definition 15. Let $\langle U, \mathbb{N}, D\rangle$ be a neighborhood covering decision system, $U/D = \{X_1, X_2, \ldots, X_\ell\}$ be the partition of U induced with D. The lower and upper approximations of decision D are defined as

$$\underline{\mathbb{N}_1}D = \cup_{i=1}^{\ell}\underline{\mathbb{N}_1}X_i, \quad \underline{\mathbb{N}_2}D = \cup_{i=1}^{\ell}\underline{\mathbb{N}_2}X_i,$$
$$\overline{\mathbb{N}_1}D = \cup_{i=1}^{\ell}\overline{\mathbb{N}_1}X_i, \quad \overline{\mathbb{N}_2}D = \cup_{i=1}^{\ell}\overline{\mathbb{N}_2}X_i.$$

It is easy to see that $\overline{\mathbb{N}_1}D = U$ and $\overline{\mathbb{N}_2}D = U$. However, $\underline{\mathbb{N}_1}D \subseteq U$ and $\underline{\mathbb{N}_2}D \subseteq U$.

Definition 16. Let $\langle U, \mathbb{N}, D\rangle$ be a neighborhood covering decision system, $U/D = \{X_1, X_2, \ldots, X_\ell\}$. We say $x \in U$ is Type-1 consistent if there exists $X_i \in U/D$, such that $\mathbb{N}(x) \subseteq X_i$. We say x is Type-2 consistent if there exist $x' \in U$ and $X_i \in U/D$, such that $x \in \mathbb{N}(x')$ and $\mathbb{N}(x') \subseteq X_i$.

Definition 17. If all the samples in U are Type-1 consistent, we say that the neighborhood covering decision system is Type-1 consistent; otherwise we say the system is Type-1 inconsistent. If all the samples in U are Type-2 consistent, we say that the neighborhood covering decision system is Type-2 consistent; otherwise we say the system is Type-2 inconsistent.

Theorem 1. *Let $\langle U, \mathbb{N}, D\rangle$ be a neighborhood covering decision system.*

1. *$\underline{\mathbb{N}_1}D = U$ iff $\langle U, \mathbb{N}, D\rangle$ is Type-1 consistent.*
2. *$\underline{\mathbb{N}_2}D = U$ iff $\langle U, \mathbb{N}, D\rangle$ is Type-2 consistent.*

Proof. If $\langle U, \mathbb{N}, D\rangle$ is Type-1 consistent, $\forall x \in U$, x is Type-1 consistent. There exists $X_i \in U/D$, such that $\mathbb{N}(x) \subseteq X_i$. Then $x \in \underline{\mathbb{N}_1}D$. So we have $\underline{\mathbb{N}_1}D = U$. Analogically, we can also get the second term. □

Theorem 2. *If $\underline{\mathbb{N}_1}D = U$, then $\underline{\mathbb{N}_2}D = U$ holds. However, if $\underline{\mathbb{N}_2}D = U$, we cannot obtain $\underline{\mathbb{N}_1}D = U$.*

Proof. If $\underline{\mathbb{N}_1}D = U$, $\forall x \in U$, there exists $X_i \in U/D$, such that $\mathbb{N}(x) \subseteq X_i$. Thus $\forall\mathbb{N}(x)$,there exists $X_i \in U/D$, such that $\mathbb{N}(x) \subseteq X_i$. In this case, $\underline{\mathbb{N}_2}D = U$. However, if $\underline{\mathbb{N}_2}D = U$, there may be a sample x_i, such that $\mathbb{N}(x_i)$ is not consistent, but $\exists\mathbb{N}(x_j)$, $x_i \in \mathbb{N}(x_j)$ and $\mathbb{N}(x_j)$ is consistent. In this case, $\underline{\mathbb{N}_1}D \neq U$. □

$\underline{\mathbb{N}_1}D = U$ shows that the neighborhood of each sample is consistent. Thus, there exists $X_i \in U/D$, such that $\mathbb{N}(x) \subseteq X_i$. Therefore $\underline{\mathbb{N}_2}D = U$. Whereas, when $\underline{\mathbb{N}_2}D = U$, there may be some inconsistent covering element $\mathbb{N}(x)$. In this case $\underline{\mathbb{N}_1}D \neq U$.

Example 2. Continue Example 1. Assume that $U/D = \{d_1, d_2\}$, $d_1 = \{x_1, x_2, x_3\}$ and $d_2 = \{x_4, x_5\}$. The size of neighborhood is computed as the margin. Then we get $\underline{\mathbb{N}_1}d_1 = \{x_1, x_2, x_3\}$, $\underline{\mathbb{N}_1}d_2 = \{x_4, x_5\}$, $\underline{\mathbb{N}_2}d_1 = \{x_1, x_2, x_3\}$, $\underline{\mathbb{N}_2}d_2 = \{x_4, x_5\}$.

So $\underline{\mathbb{N}_1}D = \{x_1, x_2, x_3, x_4, x_5\}$, and $\underline{\mathbb{N}_2}D = \{x_1, x_2, x_3, x_4, x_5\}$. The decision system is both Type-2 consistent and Type-1 consistent.

However, if we set $\delta = 0.15$, then $\underline{\mathbb{N}_1}d_1 = \{x_1, x_2\}$, $\underline{\mathbb{N}_1}d_2 = \{x_5\}$, $\underline{\mathbb{N}_2}d_1 = \{x_1, x_2, x_3\}$, $\underline{\mathbb{N}_2}d_2 = \{x_4, x_5\}$. So $\underline{\mathbb{N}_1}D = \{x_1, x_2, x_5\}$, while $\underline{\mathbb{N}_2}D = \{x_1, x_2, x_3, x_4, x_5\}$. The decision system is Type-2 consistent and Type-1 inconsistent.

A covering element is said to be consistent if all the objects in it belong to the same decision class; otherwise it is inconsistent. There maybe exist two classes of covering elements in Type 2 consistent neighborhood covering decision systems: consistent and inconsistent covering elements. $\mathbb{N}(x_1)$, $\mathbb{N}(x_2)$ and $\mathbb{N}(x_5)$ are consistent, while $\mathbb{N}(x_3)$ and $\mathbb{N}(x_4)$ are inconsistent. However, all the covering elements are consistent if the decision system is type-1 consistent.

The fact is also reasonable that inconsistent covering elements exist in consistent covering decision systems. As to a covering decision system, there are some redundant covering elements. Although we can find a neighborhood granule for each sample such that this sample is consistent in this granule, this sample may also exist in other inconsistent neighborhood granules. We just require finding a granule which contains this sample and the granule is consistent.

Definition 18. Let $\langle U, \mathbb{N}, D\rangle$ be a Type-1 or Type-2 consistent neighborhood covering decision system. X_i is one of the decision classes. $\mathbb{N}(x') \in \mathbb{N}$. If $\exists \mathbb{N}(x) \in \mathbb{N}$, such that $\mathbb{N}(x') \subset \mathbb{N}(x) \subseteq X_i$, we say $\mathbb{N}(x')$ is relatively consistent reducible with respect to X_i; otherwise, we say $\mathbb{N}(x')$ is relatively consistent irreducible.

Definition 19. Let $\langle U, \mathbb{N}, D\rangle$ be a Type-2 consistent neighborhood covering decision system. If $\mathbb{N}(x) \in \mathbb{N}$ is an inconsistent covering element, we say $\mathbb{N}(x)$ is a relatively inconsistent reducible element.

There are two types of reducible elements: one is consistent and is contained by other consistent elements; the other is the inconsistent elements.

Definition 20. Let $\langle U, \mathbb{N}, D\rangle$ be a Type-1 consistent neighborhood covering decision system. If $\forall \mathbb{N}(x) \in \mathbb{N}$, there does not exist $\mathbb{N}(x') \in \mathbb{N}$, such that $\mathbb{N}(x') \subseteq \mathbb{N}(x) \subseteq X_i$, where X_i is an arbitrary decision class, then we say $\langle U, \mathbb{N}, D\rangle$ is relatively irreducible; otherwise, we say $\mathbb{N}(x')$ is relatively reducible.

Definition 21. Let $\langle U, \mathbb{N}, D\rangle$ be a Type-1 consistent neighborhood covering decision system. $\mathbb{N}' \subseteq \mathbb{N}$ is a derived covering from $\mathbb{N}$ by removing the relatively irreducible covering elements, and $\langle U, \mathbb{N}', D\rangle$ is relatively irreducible. Then we say that $\mathbb{N}'$ is a D-relative reduct of $\mathbb{N}$, denoted by $\text{reduct}_D(\mathbb{N})$.

Theorem 3. *Let $\langle U, \mathbb{N}, D\rangle$ be a Type-1 consistent neighborhood covering decision system and $\text{reduct}_D(\mathbb{N})$ be a D-relative reduct of $\mathbb{N}$. Then $\langle U, \text{reduct}_D(\mathbb{N}), D\rangle$ is also a Type-1 consistent covering decision system, and $\forall \mathbb{N}(x) \in \mathbb{N},\ \exists \mathbb{N}(x') \in \text{reduct}_D(\mathbb{N})$, such that $\mathbb{N}(x) \subseteq \mathbb{N}(x')$.*

The conclusions of Theorem 3 are straightforward because we just remove the redundant covering elements in the covering. Moreover, as to a Type-1 consistent covering decision system, all the elements are consistent; naturally the reduced covering decision system is also Type-1 consistent.

After covering element reduction, there is no redundant covering element in the covering decision system. All the selected covering elements are useful in approximating the decision classes. With a reduct of a covering decision system, we can generate covering rules in the form.

If $x' \in \mathbb{N}(x)$, then x' is assigned with the class of $\mathbb{N}(x)$.

The theoretic framework of neighborhood covering reduction forms a mechanism for classification rule learning from training samples. In next section, we will construct an algorithm for rule learning based on neighborhood covering reduction.

4. Covering reduction based rule learning algorithm

Rule learning is one of the most important tasks in machine learning and data mining, and it is also a main application domain of rough set theory. In this section, we introduce a novel rule learning algorithm based on neighborhood covering reduction.

Just like the problem of minimal attribute reduction, the search of minimal rule set is also NP-hard [2]. There are several strategies to search the suboptimal rule set, such as forward search, backward search and genetic algorithm [21]. Here we consider the forward search technique which start with an empty set of rules, and add new rules one by one. In each step, the consistent neighborhood which covers most samples is selected and generates a piece of rule.

In addition, we can see that some neighborhoods just cover several samples in the experiments. If we include these rules, the size of rule base is very large and the corresponding classification model would overfit the training samples. Thus a pruning strategy is required. The pruning techniques used in other rule learning systems are applicable [8]. In this work, we add rules one by one. In the meanwhile, we also test the current set of rules with training and test samples. We record the classification accuracies. The rule set yielding the best classification performance is outputted. Certainly, other pruning techniques can also adopted.

Algorithm NCR: Neighborhood covering reduction

```
Input:
Training set $U\_train = \{(x_1,d_1),\ldots,(x_i,d_i),\ldots,(x_n,d_n)\}$;
Test set $U\_test = \{(x'_1,d'_1),\ldots,(x'_i,d'_i),\ldots,(x'_m,d'_m)\}$.
Output: rule set $R$.
compute the margins of training samples $m(x_i), i = 1, 2, \ldots, n$. If $m(x_i) < 0$, set $m(x_i) = 0$.
compute $\mathbb{N}(x_i)$ of sample $x_i$, $i = 1, 2, \ldots, n$
$\mathbb{N} \leftarrow \{\mathbb{N}(x_i), i = 1, \ldots, n\}$, $R \leftarrow \varnothing$;
compute the number of the samples covered by each covering element in $\mathbb{N}$.
While ($\mathbb{N} \neq \emptyset$)
    select the covering element $\mathbb{N}(x)$ covering the largest number of samples.
    add a rule $(x, m(x), y)$ into $R$, where $y$ is the decision of $x$.
    remove $\mathbb{N}(x')$ if $\mathbb{N}(x') \subseteq \mathbb{N}(x)$.
end
sort the rules according to the size of the covering elements in the descending order.
select the first $h$ rules that producing the highest classification accuracy on training set or test set.
```

This algorithm greedily searches the largest neighborhood of samples in the forward search step and removes the rules covering the least samples. In this way, we generate a small set of rules which can cover most of the samples.

It is worth pointing out that the above algorithm is developed according to Type 2 neighborhood rough sets, instead of Type 1 neighborhood rough sets. However, if the size of neighborhood is set as the margin of samples, there is no inconsistent neighborhood in the derived neighborhood covering decision system. So the rules produced with the above algorithm are consistent.

According to the definition of margin of samples, we know the margins of samples far away from the classification border are large. So their neighborhoods are also larger than the neighborhoods of samples close to classification border, as shown in Fig. 1.

There are two classes of samples in this classification task, where "○" denotes d_1, and "+" denotes d_2. We consider samples x_1, x_2 and x_3.

$$m(x_1) = \Delta(x_1, nm(x_1)) - \Delta(x_1, nh(x_1)) > 0;$$
$$m(x_2) = \Delta(x_2, nm(x_2)) - \Delta(x_2, nh(x_2)) \approx 0;$$
$$m(x_3) = \Delta(x_3, nm(x_3)) - \Delta(x_3, nh(x_3)) < 0.$$

Correspondingly, $\mathbb{N}(x_2) = \{x_2\}$ and $\mathbb{N}(x_3) = \{x_3\}$. They are pruned in the pruning step, whereas $\mathbb{N}(x_1)$ covers many samples and it is selected and generates a classification rule that if $\Delta(x,x_1) \leqslant m(x_1)$, then $x \in d_1$.

It is notable that the pruned rules do not cover all the samples. In addition, the test sample may be beyond the region of the neighborhood of any sample. In this case, no rule matches the test sample. In this case, the test sample is classified to the class of the nearest neighborhood.

As we employ a greedy strategy in searching rules, the rule learning algorithm is very efficient. First, we should compute the margins of samples. The time complexity in this step are $O(n \times n)$. Then the time cost in computing the neighborhoods of

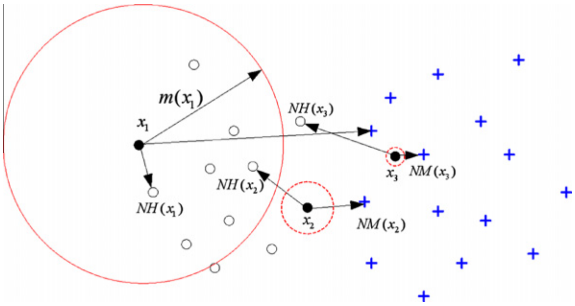

Fig. 1. Demonstration of neighborhood covering elements.

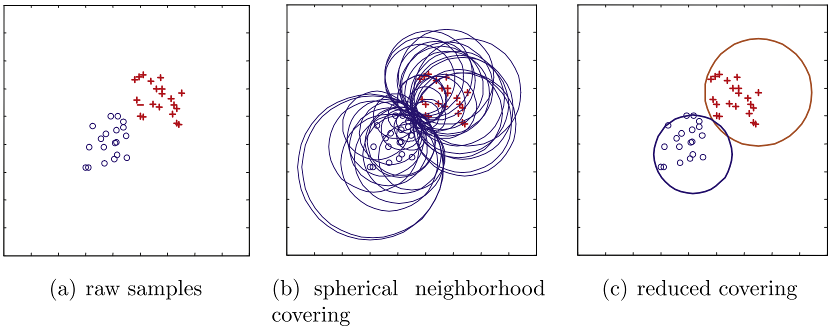
(a) raw samples (b) spherical neighborhood covering (c) reduced covering

Fig. 2. Rule learning with spherical neighborhoods.

samples is $O(n)$. The complexities of covering reduction and rule pruning are $O(n)$. Totally, the time complexity of algorithm NCR is $O(n \times n)$.

5. Experimental analysis

In order to test the proposed algorithm, we conduct some numerical experiments. First we present a toy example with artificial data, and then we give some real-world classification tasks.

We generate two binary classification tasks in 2-D feature spaces. The training samples are shown in Fig. 2(a) and Fig. 3(a), respectively. We compute the margin of every sample with Euclidean distance and infinite norm based distance, respectively. Then we can build a neighborhood of each sample. Euclidean distance produces spherical neighborhoods and infinite norm based distance yields quadrate neighborhoods, as shown in Fig. 2(b) and Fig. 3(b), respectively.

Obviously, there are a lot of redundant neighborhoods in the original neighborhood covering. Removing the superfluous covering elements leads to a compact and concise classification model. So we employ Algorithm NCR on the neighborhood covering. The reduced coverings are presented in Fig. 2(c) and Fig. 3(c), respectively. As to these simple tasks, two pieces of rules are produced for each task. The classification models are simple and easy to be understood.

Furthermore, we collect ten classification tasks from UCI machine learning repository. The description of these data sets is given in Table 1.

Now we employ NCR algorithm on these data sets. First, we observe the relationship between the classification accuracies and number of rules. We randomly divide the data into two subsets: 50% as training set and the others as test set. As we add the rule one by one, we also compute the training accuracies and test accuracies. The results on iris, wine heart and pima data sets are given in Fig. 4.

From Fig. 4, we see that both the training and test accuracies increase with the number of rules in the beginning. However, if the number of rules exceeds a certain value, the classification accuracies do not increase. On the contrary, the accuracies keep invariant, even decrease, which shows the classification models overfit the training samples if too many rules are used. Therefore, a pruning technique is useful for neighborhood covering reduction based rule learning.

The 10-fold cross validation accuracies based on nearest neighbor rule (NN), neighborhood classifier (NEC) [19], Learning Vector Quantization (LVQ) [28] and linear support vector machine (LSVM) are presented in Table 2. Moreover, we also try some decision rule learning algorithms, including CART, C4.5 and Jripper, the number of rules and classification performance are shown in Table 2.

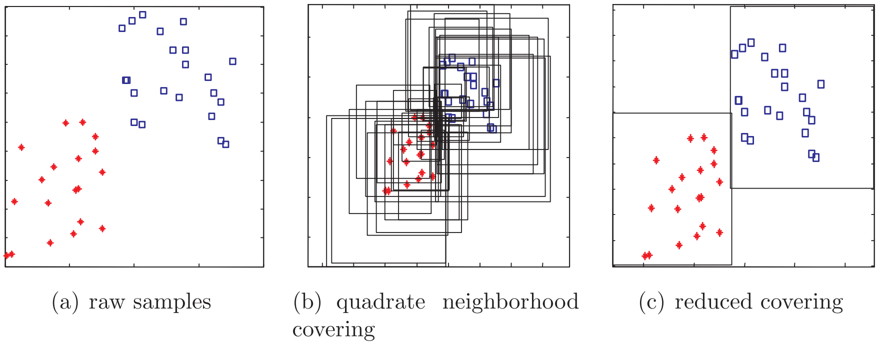
(a) raw samples (b) quadrate neighborhood covering (c) reduced covering

Fig. 3. Rule learning with quadrate neighborhoods.

Table 1
Description of these data sets.

	Features	Class	Instances
wine	14	3	178
iris	5	3	150
wdbc	31	2	569
pima	9	2	768
thyroid	6	3	215
german	20	2	1000
yeast	8	2	1484
heart	14	2	270
sick	30	2	2800
segment	18	7	2310

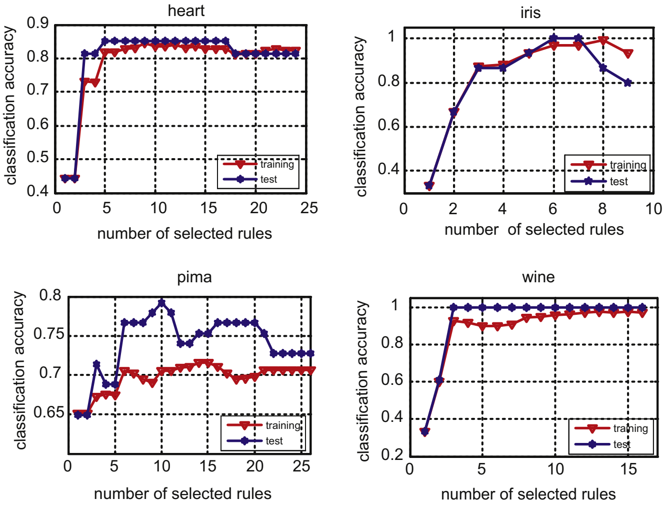

Fig. 4. Variation of classification performance with the number of decision rules.

Table 3 presents the classification results based on neighborhood covering reduction, where spherical and quadrate neighborhoods are considered. The classification accuracies and number of rules pruned on training sets and test sets are recorded.

Comparing Table 3 with Table 2, we see that neighborhood covering reduction based rule sets are better than or as good as the classical classification algorithms. If we consider the spherical neighborhood and the rule set is pruned based on test samples, the classification performance of reduced rule sets even better than linear support vector machine. This result shows that if the rules are elaborately tuned they are powerful in classification decision.

Unfortunately, we find the numbers of rules produced with neighborhood covering reduction are much more than those obtained with CART, C4.5 and Jripper. We observe some derived rules and find most of them can be merged by increase the size of neighborhood. We do not discuss this problem in this work.

Table 2
Classification performance of some classical classification algorithms.

	NEC	LVQ	1-NN	LSVM	CART		C4.5		Jripper	
	Acc.	Acc.	Acc.	Acc.	Acc.	N	Acc.	N	Acc.	N
wine	96.6 ± 2.9	96.0 ± 2.8	94.9 ± 5.0	98.9 ± 2.3	88.2 ± 6.2	10	91.6 ± 7.3	5	93.8 ± 8.1	3
iris	96.0 ± 4.7	96.6 ± 5.1	96.0 ± 5.6	97.3 ± 4.6	95.3 ± 5.9	5	96.0 ± 5.0	5	94.0 ± 7.1	4
wdbc	94.6 ± 2.5	95.4 ± 3.6	95.4 ± 3.3	97.7 ± 2.5	93.1 ± 2.4	9	93.3 ± 3.8	13	94.3 ± 2.0	5
pima	76.0 ± 3.0	73.3 ± 4.2	70.8 ± 3.7	76.7 ± 3.6	74.4 ± 5.4	13	75.2 ± 7.4	20	75.0 ± 5.3	3
thyroid	87.9 ± 7.0	89..7 ± 6.6	95.3 ± 3.4	89.8 ± 7.9	90.1 ± 4.7	4	91.6 ± 3.6	9	92.5 ± 7.0	5
german	73.5 ± 3.4	72.3 ± 3.5	68.8 ± 3.2	73.7 ± 4.7	75.0 ± 3.2	10	71.7 ± 3.1	90	74.2 ± 2.5	3
yeast	73.8 ± 3.0	75.5 ± 3.4	70.5 ± 5.7	73.9 ± 3.6	76.1 ± 4.2	5	76.0 ± 3.9	44	75.0 ± 4.3	5
heart	80.0 ± 5.6	82.6 ± 6.8	76.7 ± 9.4	83.3 ± 5.3	78.5 ± 5.2	16	76.6 ± 4.8	18	78.8 ± 5.1	4
sick	93.9 ± 1.1	93.8 ± 3.2	95.5 ± 0.1	93.9 ± 12.6	98.5 ± 1.2	24	98.6 ± 1.3	15	98.2 ± 1.7	7
segment	90.4 ± 5.5	82.5 ± 2.6	96.5 ± 2.8	90.7 ± 3.8	96.1 ± 1.8	46	97.1 ± 2.0	39	95.4 ± 1.9	21
Ave.	86.3	85.8	86.0	87.6	86.5	14.2	86.8	25.8	87.1	6

Table 3
Classification performance of neighborhood covering reduction based classification algorithms.

	Spherical				Quadrate			
	Test		Training		Test		Training	
	Accuracy	Rules	Accuracy	Rules	Accuracy	Rules	Accuracy	Rules
wine	97.6 ± 4.3	9.6	96.0 ± 4.8	11.7	95.4 ± 3.8	10.7	93.2 ± 3. 7	12.7
iris	99.3 ± 2.1	5.4	97.3 ± 4.7	7.3	100.0 ± 0.0	6.0	100.0 ± 0.0	8.2
wdbc	96.5 ± 3.7	23.7	94.6 ± 4.3	24.1	92.6 ± 2.1	15.0	92.3 ± 2.0	21.7
pima	75.3 ± 3.6	11.8	72.1 ± 3.7	15.2	75.8 ± 2.3	11.4	74.0 ± 2.1	13.9
thyroid	92.5 ± 6.3	9.6	91.2 ± 6.0	11	90.2 ± 5.1	7.2	89.8 ± 5.3	9
german	75.5 ± 9.4	4	67.9 ± 2.0	17.6	73.5 ± 3.0	61.1	72.3 ± 2.9	92.4
yeast	77.0 ± 3.4	52.3	75.6 ± 3.8	71.9	78.3 ± 1.2	55.5	76.7 ± 1.3	71.6
heart	86.3 ± 3.9	7.8	84.4 ± 4.6	9.4	75.6 ± 6.6	15	75.6 ± 6. 6	18.4
sick	95.2 ± 7.8	87.5	93.6 ± 5.5	97.6	95.0 ± 6.4	177.7	94. 7 ± 6.7	176.8
segment	91.7 ± 4.2	89.6	91.5 ± 4.3	118.5	95.2 ± 8.5	128.3	95. 0 ± 1.1	131
Ave.	88.7	30.1	86.4	38.4	87.2	48.5	86.4	55.6

Considering the classification performance, we think rule learning based on neighborhood covering reduction is promising and efficient.

6. Conclusions and future work

Although covering reduction was widely discussed in recent years, no successful application has been reported so far. We discuss the problem of relative covering reduction and study the difference between covering reduction and relative covering reduction. It is pointed out that covering reduction tries to select the small covering elements, whereas relative covering reduction selects the large consistent covering elements. Therefore, relative covering reduction can build a compact and simple rule set for classification, while covering reduction can keep the finest approximation of arbitrary concepts. So covering reduction is not applicable to rule learning for classification because the derived rule set would be very complex and overfit training samples.

Moreover, we introduce the concept of classification margin to compute the size of neighborhood of each sample, thus we generate a neighborhood covering of the universe. We define two classes of neighborhood lower and upper approximations. We find there are two types of reducible covering elements. With relative neighborhood covering reduction, we select the large neighborhoods and remove the small neighborhoods.

We test the classification performance of the rule learning algorithm. The experimental results show that the rule learning algorithm NCR is better than or as good as some existing classification algorithms, including NEC, NN, LVQ, LSVM, CART, C4.5 and Jripper.

As we mention in the experimental section, the rules derived with NCR sometimes are much more than C4.5 and other techniques. In fact, most of the derived rules can be merged. Thus the number of rules will be greatly reduced. Moreover, we do not consider reducing redundant attributes in this work. However, CART and C4.5 implicitly remove irrelevant features in rule learning. We will combine feature selection and covering element reduction for rule learning in the future.

Acknowledgments

We would like to express our acknowledgements to the referees for their constructive comments. This work is supported by National Natural Science Foundation of China under Grants 60703013 and 10978011.

References

[1] S. Greco, B. Matarazzo, R. Slowinski, Rough approximation of a preference relation by dominance relations, in: S. Tsumoto et al., (Eds.), Proceedings of the 4th International Workshop on Rough Sets, Fuzzy Sets and Machine Discovery - RSFD'96, Tokyo, Japan, 1996, pp. 125–130.

[2] T.L. Andersen, T.R. Martinez, Np-completeness of minimum rule sets, in: Proceedings of the 10th International Symposium on Computer and Information Sciences, pp. 411–418.

[3] R.B. Bhatt, M. Gopal, Frct: fuzzy-rough classification trees, Pattern Analysis & Applications 11 (2008) 73–88.

[4] J. Błaszczyński, S. Greco, R. Słowiński, Multi-criteria classification – A new scheme for application of dominance-based decision rules, European Journal of Operational Research 181 (2007) 1030–1044.

[5] Z. Bonikowski, E. Bryniarski, U. Wybraniec-Skardowska, Extensions and intentions in the rough set theory, Information Sciences 107 (1998) 149–167.

[6] 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 (2007) 3500–3518.

[7] D. Dubois, H. Prade, Rough fuzzy sets and fuzzy rough sets, International Journal of General Systems 17 (1990) 191–209.

[8] D. Fierens, J. Ramon, H. Blockeel, M. Bruynooghe, A comparison of pruning criteria for probability trees, Machine Learning 78 (2010) 251–285.

[9] R. Gilad-Bachrach, A. Navot, N. Tishby, Margin based feature selection-theory and algorithms, in: Proceedings of the Twenty-First International Conference on Machine Learning, ACM, p. 43.

[10] S. Greco, B. Matarazzo, R. Słowiński, Rough approximation of a preference relation by dominance relations, European Journal of Operational Research 117 (1999) 63–83.

[11] S. Greco, B. Matarazzo, R. Słowiński, Rough approximation by dominance relations, International Journal of Intelligent Systems 17 (2002) 153–171.
[12] T.P. Hong, T.T. Wang, S.L. Wang, B.C. Chien, Learning a coverage set of maximally general fuzzy rules by rough sets, Expert Systems with Applications 19 (2000) 97–103.
[13] J. Hu, G. Wang, Knowledge reduction of covering approximation space, Transactions on Computational Science, Special Issue on Cognitive Knowledge Representation (2009) 69–80.
[14] Q. Hu, S. An, D. Yu, Soft fuzzy rough sets for robust feature evaluation and selection, Information Sciences 180 (2010) 4384–4400.
[15] Q. Hu, W. Pedrycz, D. Yu, J. Lang, Selecting discrete and continuous features based on neighborhood decision error minimization, IEEE Transactions on Systems, Man, and Cybernetics – Part B: Cybernetics 40 (2010) 137–150.
[16] Q. Hu, Z. Xie, D. Yu, Hybrid attribute reduction based on a novel fuzzy-rough model and information granulation, Pattern Recognition 40 (2007) 3509–3521.
[17] Q. Hu, D. Yu, M. Guo, Fuzzy preference based rough sets, Information Sciences 180 (2010) 2003–2022.
[18] Q. Hu, D. Yu, J. Liu, C. Wu, Neighborhood rough set based heterogeneous feature subset selection, Information Sciences 178 (2008) 3577–3594.
[19] Q. Hu, D. Yu, Z. Xie, Neighborhood classifiers, Expert Systems with Applications 34 (2008) 866–876.
[20] N.N. Morsi, M.M. Yakout, Axiomatics for fuzzy rough sets, Fuzzy Sets and Systems 100 (1998) 327–342.
[21] M.J. Moshkov, A. Skowron, Z. Suraj, On minimal rule sets for almost all binary information systems, Fundamenta Informaticae 80 (2007) 247–258.
[22] Z. Pawlak, Rough Sets-Theoretical Aspects of Reasoning About Data, Kluwer Academic, 1991.
[23] Y. Qian, C. Dang, J. Liang, D. Tang, Set-valued ordered information systems, Information Sciences 179 (2009) 2809–2832.
[24] Y. Qian, J. Liang, W. Pedrycz, C. Dang, Positive approximation: an accelerator for attribute reduction in rough set theory, Artificial Intelligence 174 (2010) 597–618.
[25] A.M. Radzikowska, E.E. Kerre, A comparative study of fuzzy rough sets, Fuzzy Sets and Systems 126 (2002) 137–155.
[26] R. Słowiński, D. Vanderpooten, A generalized definition of rough approximations based on similarity, IEEE Transactions on Data and Knowledge Engineering 12 (2000) 331–336.
[27] D. Slęzak, W. Ziarko, The investigation of the Bayesian rough set model, International Journal of Approximate Reasoning 40 (2005) 81–91.
[28] M.F. Umer, M.S.H. Khiyal, Classification of textual documents using learning vector quantization, Information Technology 6 (2007) 154–159.
[29] C. Wang, C. Wu, D. Chen, A systematic study on attribute reduction with rough sets based on general binary relations, Information Sciences 178 (2008) 2237–2261.
[30] X. Wang, E.C. Tsang, S. Zhao, D. Chen, D.S. Yeung, Learning fuzzy rules from fuzzy samples based on rough set technique, Information Sciences 177 (2007) 4493–4514.
[31] D.R. Wilson, T.R. Martinez, Improved heterogeneous distance functions, Journal of Artificial Intelligence Research 6 (1997) 1–34.
[32] W. Wu, Attribute reduction based on evidence theory in incomplete decision systems, Information Sciences 178 (2008) 1355–1371.
[33] W. Wu, J. Mi, W. Zhang, Generalized fuzzy rough sets, Information Sciences 151 (2003) 263–282.
[34] W. Wu, W. Zhang, Constructive and axiomatic approaches of fuzzy approximation operators, Information Sciences 159 (2004) 233–254.
[35] T. Yang, Q. Li, Reduction about approximation spaces of covering generalized rough sets, International Journal of Approximate Reasoning 51 (2010) 335–345.
[36] J. Yao, Y. Yao, W. Ziarko, Probabilistic rough sets: approximations, decision-makings, and applications, International Journal of Approximate Reasoning 49 (2008) 253–254.
[37] Y. Yao, Probabilistic rough set approximations, International Journal of Approximate Reasoning 49 (2008) 255–271.
[38] Y. Yao, S.K.M. Wong, A decision theoretic framework for approximating concepts, International Journal of Man-Machine Studies 37 (1992) 793–809.
[39] Y. Yao, Y. Zhao, Attribute reduction in decision-theoretic rough set models, Information Sciences 178 (2008) 3356–3373.
[40] D.S. Yeung, D. Chen, E.C.C. Tsang, J.W.T. Lee, X. Wang, On the generalization of fuzzy rough sets, IEEE Transactions on Fuzzy Systems 13 (2005) 343–361.
[41] W. Zakowski, Approximations in the space (u, π), Demonstratio Mathematica 16 (1983) 761–769.
[42] S. Zhao, E.C.C. Tsang, D. Chen, The model of fuzzy variable precision rough rets, IEEE Transactions on Fuzzy Systems 17 (2009) 451–467.
[43] W. Zhu, Relationship between generalized rough sets based on binary relation and covering, Information Sciences 179 (2009) 210–225.
[44] W. Zhu, F. Wang, Reduction and axiomization of covering generalized rough sets, Information Science 152 (2003) 217–230.
[45] W. Zhu, F. Wang, On three types of covering-based rough sets, IEEE Transactions on Knowledge and Data Engineering 19 (2007) 1131–1144.
[46] W. Ziarko, Variable precision rough set model, Journal of Computer and System Sciences 46 (1993) 39–59.