SINGLE REDUCT GENERATION BASED ON RELATIVE INDISCERNIBILITY OF ROUGH SET THEORY

Shampa Sengupta and Asit Kr. Das

M.C.K.V. Institute Of Engineering - 243, G.T.Road (North), Liluah, Howrah 711204, West Bengal
Bengal Engineering And Science University, Shibpur, Howrah 711103, West Bengal
shampa2512@yahoo.co.in, akdas@cs.becs.ac.in

Abstract.

In real world everything is an object which represents particular classes. Every object can be fully
described by its attributes. Any real world dataset contains large number of attributes and objects. Classifiers give poor performance when these huge datasets are given as input to it for proper
classification. So from these huge dataset most useful attributes need to be extracted that contribute
the maximum to the decision. In the paper, attribute set is reduced by generating reducts using the
indiscernibility relation of Rough Set Theory (RST). The method measures similarity among the attributes
using relative indiscernibility relation and computes attribute similarity set. Then the set is minimized and
an attribute similarity table is constructed from which attribute similar to maximum number of attributes is
selected so that the resultant minimum set of selected attributes (called reduct) cover all attributes of the
attribute similarity table. The method has been applied on glass dataset collected from the UCI repository
and the classification accuracy is calculated by various classifiers. The result shows the efficiency of the
proposed method.

Keywords:

Rough Set Theory, Attribute Similarity, Relative Indiscernibility Relation, Reduct.

1. Introduction

In general, considering all attributes highest accuracy of a classifier should be achieved. But for
real-world problems, there is huge number of attributes, which degrades the efficiency of the
Classification algorithms. So, some attributes need to be neglected, which again decrease the
accuracy of the system. Therefore, a trade-off is required for which strong dimensionality
reduction or feature selection techniques are needed. The attributes contribute the most to the
decision must be retained. Rough Set Theory (RST) [1, 2], new mathematical approach to
imperfect knowledge, is popularly used to evaluate significance of attribute and helps to find
minimal set of attribute called reduct. Thus a reduct is a set of attributes that preserves partition. It
means that a reduct is the minimal subset of attributes that enables the same classification of
elements of the universe as the whole set of attributes. In other words, attributes that do not
belong to a reduct are superfluous with regard to classification of elements of the universe. Hu et

DOI : 10.5121/ijsc.2012.3109
al. [3] developed two new algorithms to calculate core attributes and reducts for feature selection. These algorithms can be extensively applied to a wide range of real-life applications with very large data sets. Jensen et al. [4] developed the Quickreduct algorithm to compute a minimal reduct without exhaustively generating all possible subsets and also they developed Fuzzy-Rough attribute reduction with application to web categorization. Zhong et al. [5] applies Rough Sets with Heuristics (RSH) and Rough Sets with Boolean Reasoning (RSBR) are used for attribute selection and discretization of real-valued attributes. Komorowsk et al. [6] studies an application of rough sets to modelling prognostic power of cardiac tests. Bazan [7] compares rough set-based methods, in particular dynamic reducts, with statistical methods, neural networks, decision trees and decision rules. Carlin et al. [8] presents an application of rough sets to diagnosing suspected acute appendicitis. The main advantage of rough set theory in data analysis is that it does not need any preliminary or additional information about data like probability in statistics [9], or basic probability assignment in Dempster-Shafer theory [10], grade of membership or the value of possibility in fuzzy set theory [11] and so on. But finding reduct for classification is an NP-Complete problem and so some heuristic approach should be applied. In the paper, a novel reduct generation method is proposed based on the indiscernibility relation of rough set theory.

![Flowchart of Single Reduct Generation Process](image)

In the method, a new kind of indiscernibility, called relative indiscernibility of an attribute with respect to other attribute is introduced. This relative indiscernibility relation induces the partitions of attributes, based on which similarity between conditional attributes is measured and an
attribute similarity set (ASS) is obtained. Then, the similarity set is minimized by removing the attribute similarities having similarity measure less than the average similarity. Lastly, an attribute similarity table is constructed for ASS each row of which describes the similarity of an attribute with some other attributes. Then traverse each row and select the attribute of that row which has maximum similar attributes. Next, all the rows associated with the selected attribute and its similar attributes are deleted from the table and similarly select another attribute from the modified table. The process continued until all the rows are deleted from the table and finally, selected attributes, covering all the attributes are considered as reduct, a minimum set of attributes.

The rest of the paper is organized as follows: Similarity measurement of attributes by relative indiscernibility and single reduct generation are described in section 2 and section 3 respectively. Section 4 explains the experimental analysis of the proposed method and finally conclusion of the paper is stated in section 5.

2. Relative Indiscernibility and Dependency of Attributes

Formally, a decision system DS can be seen as a system $DS = (U, A)$ where $U$ is the universe (a finite set of objects, $U = \{x_1, x_2, ..., x_m\}$) and $A$ is the set of attributes such that $A = C \cup D$ and $C \cap D = \emptyset$ where $C$ and $D$ are the set of condition attributes and the set of decision attributes, respectively.

2.1 Indiscernibility

A per the discussion in section II, each attribute $a \in A$ defines an information function: $f_a : U \rightarrow V_a$, where $V_a$ is the set of values of $a$, called the domain of attribute. Every subset of attributes $P$ determines an indiscernibility relation over $U$, and is denoted as $IND(P)$, which can be defined as, $IND(P) = \{(x, y) \in U \times U | \forall a \in P, \ f_a(x) = f_a(y)\}$. For each set of attributes $P$, an indiscernibility relation $IND(P)$ partitions the set of objects into a $m$-number of equivalence classes $\{\}$ defined as partition $U/IND(P)$ or $U/P$ is equal to $\{|x_i|\}$ where $|U/P| = m$. Elements belonging to the same equivalence class are indiscernible; otherwise elements are discernible with respect to $P$. If one considers a non-empty attributes subset, $R \subset P$ and $IND(R) = IND(P)$, then $P - R$ is dispensable. Any minimal $R$ such that $IND(R) = IND(P)$, is a minimal set of attributes that preserves the indiscernibility relation computed on the set of attributes $P$. $R$ is called reduct of $P$ and denoted as $R = RED(P)$. The core of $P$ is the intersection of these reductions, defined as $CORE(P) = \cap RED(P)$. Naturally, the core contains all the attributes from $P$ which are considered of greatest importance for classification, i.e., the most relevant for a correct classification of the objects of $U$. On the other hand, none of the attributes belonging to the core may be neglected without deteriorating the quality of the classification considered, that is, if any attribute in the core is eliminated from the given data, it will be impossible to obtain the highest quality of approximation with the remaining attributes.

2.2 Relative Indiscernibility

Here, the relation is defined based on the same information function: $f_a : U \rightarrow V_a$ where $V_a$ is the set of values of $a$, called the domain of attribute. Every conditional attribute $A_i$ of $C$ determines an relative (relative to decision attribute) indiscernibility relation (RIR) over $U$, and is denoted as $RIRD(A_i)$, which can be defined by equation (1).
For each conditional attribute \( A_i \), a relative indiscernibility relation \( RIRD(A_i) \) partitions the set of objects into a \( n \)-number of equivalence classes \([ \_ ]\) defined as partition \( U/IR_D(A_i) \) or \( U_{D/A_i} \) is equal to \( [\{x \in D]\} \), where \( |U_{D/A_i}| = n \). Obviously, each equivalence class \([\{x \in D]\}\) contains objects with same decision value which are indiscernible by attribute \( A_i \).

To illustrate the method, a sample dataset represented by Table 1 is considered with eight objects, four conditional and one decision attributes.

### Table 1. Sample Dataset

|    | Diploma | Experience | French | Reference | Decision |
|----|---------|------------|--------|-----------|----------|
| x_1 | MBA     | Medium     | Yes    | Excellent | Accept   |
| x_2 | MBA     | Low        | Yes    | Neutral   | Reject   |
| x_3 | MCE     | Low        | Yes    | Good      | Reject   |
| x_4 | MSc     | High       | Yes    | Neutral   | Accept   |
| x_5 | MSc     | Medium     | Yes    | Neutral   | Reject   |
| x_6 | MSc     | High       | Yes    | Excellent | Reject   |
| x_7 | MBA     | High       | No     | Good      | Accept   |
| x_8 | MCE     | Low        | No     | Excellent | Reject   |

### Table 2. Equivalence classes induces by indiscernibility and relative indiscernibility relations

| Equivalence classes for each attribute by relation \( IND(P) \) | Equivalence classes for each conditional attribute by relative indiscernibility relation \( RIRD(A_i) \) |
|-------------------------------------------------------------|--------------------------------------------------------------------------------------------------|
| \( U_D = (\{x_1, x_4, x_7\}, \{x_2, x_3, x_5, x_6, x_8\}) \) | \( UD_i = (\{x_1, x_7\}, \{x_2\}, \{x_3, x_8\}, \{x_4\}, \{x_5, x_6\}) \) |
| \( U_e = (\{x_1, x_2, x_7\}, \{x_3, x_8\}, \{x_4, x_5, x_6\}) \) | \( UD_e = (\{x_1\}, \{x_3\}, \{x_2, x_3, x_8\}, \{x_4, x_7\}, \{x_6\}) \) |
| \( U_i = (\{x_1, x_5\}, \{x_2, x_3, x_8\}, \{x_4, x_6, x_7\}) \) | \( UD_i = (\{x_1, x_4\}, \{x_2, x_3, x_5, x_6\}, \{x_7\}, \{x_8\}) \) |
| \( U_t = (\{x_1, x_2, x_3, x_4, x_5, x_6\}, \{x_7, x_8\}) \) | \( UD_t = (\{x_1\}, \{x_6, x_8\}, \{x_2, x_5\}, \{x_4\}, \{x_3, x_7\}) \) |

### 2.3 Attribute Similarity

An attribute \( A_i \) is similar to another attribute \( A_j \) in context of classification power if they induce the same equivalence classes of objects under their respective relative indiscernible relations. But in real situation, it rarely occurs and so similarity of attributes is measured by introducing the similarity measurement factor which indicates the degree of similarity of one attribute to another attribute. Here, an attribute \( A_i \) is said to be similar to an attribute \( A_j \) with degree of similarity (or
similarity factor) \( d_{ij} \) and is denoted by \( A_i \rightarrow A_j \) if the probability of inducing the same equivalence classes of objects under their respective relative indiscernible relations is \((d_{ij} \times 100)\%\), where \( d_{ij} \) is computed by equation (2). The details for computation of similarity measurement for the attribute similarity \( A_i \rightarrow A_j \) (\( A_i \neq A_j \)) is described in algorithm “SIM_FAC” below.

\[
d_{ij} = \frac{1}{|U_{D/A_i}|} \sum_{[x]_{A_i} \in U_{D/A_i}} \frac{1}{|U_{D/A_j}|} \max_{[y]_{A_j} \in U_{D/A_j}} |([x]_{A_i} \cap [y]_{A_j})|
\]  

(2)

Algorithm: SIM_FAC(A\( _i \), A\( _j \)) /* Similarity factor computation for attribute similarity \( A_i \rightarrow A_j \) */

Input: Partitions \( U_{D/A_i} = \{[x]_{A_i} \} \) and \( U_{D/A_j} = \{[y]_{A_j} \} \) obtained by applying relative indiscernibility relation RIR\( _D \) on \( A_i \) and \( A_j \) respectively.

Output: Similarity factor \( d_{ij} \)

Begin

For each conditional attribute \( A_i \) { 
/* compute relative indiscernibility RIRD (A\( _i \)) using (1) */

\[
\text{RIR}_{D}(A_i) = \{(x,y) \in \Pi_{R_i}(X_D) \times \Pi_{R_j}(X_D) | f_{R_i}(x) = f_{R_j}(y) \forall x \in U_{D/A_i} \}
\]

RIR\( _D \) (\( A_i \)) induces equivalence classes \( U_{D/A_i} = \{[x]_{A_i} \} \)

} /*end of for*/

/* similarity measurement of \( A_i \) to \( A_j \) */

\( d_{ij} = 0 \)

For each \([x]_{A_i} \in U_{D/A_i} \)

{ 
\( \text{max\_overlap} = 0 \)

For each \([x]_{A_j} \in U_{D/A_j} \)

{ 
\( \text{overlap} = |([x]_{A_i} \cap [x]_{A_j})| \)

if (\( \text{overlap} > \text{max\_overlap} \)) then

\( \text{max\_overlap} = \text{overlap} \)

}

\( d_{ij} = d_{ij} + \frac{\text{max\_overlap}}{|U_{D/A_i}|} \)

}

\( d_{ij} = \frac{d_{ij}}{|U_{D/A_i}|} \)

111
To illustrate the attribute similarity computation process, attribute similarity and its similarity factor are listed in Table 2 for all attributes of Table 1.

**Table 2.** Describe the degree of similarity of all pair of attributes

| Attribute Similarity | Equivalence Classes by $RIR_{D}(A_i)$ ($U_D/A_i$) | Equivalence Classes by $RIR_{D}(A_j)$ ($U_D/A_j$) | Similarity factor of $A_i$ to $A_j$ |
|----------------------|-----------------------------------------------|-----------------------------------------------|-----------------------------------|
| $i \rightarrow e$    | $\{x_1, x_7\}, \{x_2\}, \{x_3, x_8\}, \{x_4\}, \{x_5, x_6\}$ | $\{x_1\}, \{x_5\}, \{x_2, x_3, x_8\}, \{x_4, x_7\}, \{x_6\}$ | $\hat{f} = 0.8$ |
| $i \rightarrow f$    | $\{x_1, x_7\}, \{x_2\}, \{x_3, x_8\}, \{x_4\}, \{x_5, x_6\}$ | $\{x_1, x_4\}, \{x_2, x_3, x_5, x_6\}, \{x_7\}, \{x_8\}$ | $\hat{f} = 0.8$ |
| $i \rightarrow r$    | $\{x_1, x_7\}, \{x_2\}, \{x_3, x_8\}, \{x_4\}, \{x_5, x_6\}$ | $\{x_1\}, \{x_6, x_8\}, \{x_2, x_3, x_5\}, \{x_4\}, \{x_3, x_7\}$ | $\hat{f} = 0.7$ |
| $e \rightarrow i$    | $\{x_1\}, \{x_5\}, \{x_2, x_3, x_8\}, \{x_4, x_7\}, \{x_6\}$ | $\{x_1, x_7\}, \{x_2\}, \{x_3, x_8\}, \{x_4\}, \{x_5, x_6\}$ | $\hat{f} = 0.83$ |
| $e \rightarrow f$    | $\{x_1\}, \{x_5\}, \{x_2, x_3, x_8\}, \{x_4, x_7\}, \{x_6\}$ | $\{x_1, x_4\}, \{x_2, x_3, x_5, x_6\}, \{x_7\}, \{x_8\}$ | $\hat{f} = 0.83$ |
| $e \rightarrow r$    | $\{x_1\}, \{x_5\}, \{x_2, x_3, x_8\}, \{x_4, x_7\}, \{x_6\}$ | $\{x_1\}, \{x_6, x_8\}, \{x_2, x_3, x_5\}, \{x_4\}, \{x_3, x_7\}$ | $\hat{f} = 0.76$ |
| $f \rightarrow i$    | $\{x_1, x_4\}, \{x_2, x_3, x_5, x_6\}, \{x_7\}, \{x_8\}$ | $\{x_1, x_7\}, \{x_2\}, \{x_3, x_8\}, \{x_4\}, \{x_5, x_6\}$ | $\hat{f} = 0.75$ |
| $f \rightarrow e$    | $\{x_1, x_4\}, \{x_2, x_3, x_5, x_6\}, \{x_7\}, \{x_8\}$ | $\{x_1\}, \{x_5\}, \{x_2, x_3, x_8\}, \{x_4, x_7\}, \{x_6\}$ | $\hat{f} = 0.75$ |
| $f \rightarrow r$    | $\{x_1, x_4\}, \{x_2, x_3, x_5, x_6\}, \{x_7\}, \{x_8\}$ | $\{x_1\}, \{x_6, x_8\}, \{x_2, x_3, x_5\}, \{x_4\}, \{x_3, x_7\}$ | $\hat{f} = 0.75$ |
| $r \rightarrow i$    | $\{x_1\}, \{x_6, x_8\}, \{x_2, x_3\}, \{x_4\}, \{x_3, x_7\}$ | $\{x_1, x_7\}, \{x_2\}, \{x_3, x_8\}, \{x_4\}, \{x_5, x_6\}$ | $\hat{f} = 0.7$ |
| $r \rightarrow e$    | $\{x_1\}, \{x_6, x_8\}, \{x_2, x_3\}, \{x_4\}, \{x_3, x_7\}$ | $\{x_1\}, \{x_5\}, \{x_2, x_3, x_8\}, \{x_4, x_7\}, \{x_6\}$ | $\hat{f} = 0.7$ |
| $r \rightarrow f$    | $\{x_1\}, \{x_6, x_8\}, \{x_2, x_3\}, \{x_4\}, \{x_3, x_7\}$ | $\{x_1, x_4\}, \{x_2, x_3, x_5, x_6\}, \{x_7\}, \{x_8\}$ | $\hat{f} = 0.8$ |

The computation of $\hat{f}$ of each attribute similarity using equation (2) in Table 2 can be understood by Table 3, in which similarity $i \rightarrow e$ in first row of Table 2 is considered, where, $U_D/i = \{x_1, x_7\}, \{x_2\}, \{x_3, x_8\}, \{x_4\}, \{x_5, x_6\}$ and $U_D/e = \{x_1\}, \{x_5\}, \{x_2, x_3, x_8\}, \{x_4, x_7\}, \{x_6\}$. 

End.
Table 3. Illustrates the similarity factor computation for $i \rightarrow e$

| $U_B / t$ of $x_{i/o}$ | Overlapping of $U_B / t$ with $x_{i/o}$ of $U_B / t$ | $T = \frac{1}{\|x_{i/o}\|} \sum_{\delta f \in 2-D} \delta f / T$ |
|------------------------|-------------------------------------------------|--------------------------------------------------|
| $\{x_1, x_7\}$        | $\{x_1\}$                                      | $\frac{1}{8} \begin{pmatrix} 2 & 1 & 2 & 1 \\ 1 & 1 & 1 & 1 \end{pmatrix} = 0.8$ |
| $\{x_2\}$             | $\{x_2, x_3, x_8\}$                            | $\{x_2\} \cap \{x_2, x_3, x_8\}$               |
| $\{x_3, x_6\}$        | $\{x_2, x_3, x_6\}$                            | $\{x_3, x_6\} \cap \{x_2, x_3, x_8\}$          |
| $\{x_4\}$             | $\{x_4, x_7\}$                                 | $\{x_4\} \cap \{x_4, x_7\}$                     |
| $\{x_5, x_6\}$        | $\{x_3\}$                                      | $\{x_5, x_6\} \cap \{x_3\}$                     |

2.4 Attribute Similarity Set

For each pair of conditional attributes $(A_i, A_j)$, similarity factor is computed by “SIM_FAC” algorithm, described in section 2.3. The similarity factor of $A_i \rightarrow A_j$ is higher means that the relative indiscernibility relations $\text{RIRD}(A_i)$ and $\text{RIRD}(A_j)$ produce highly similar equivalence classes. This implies that both the attributes $A_i$ and $A_j$ have almost similar classification power and so $A_i \rightarrow A_j$ is considered as strong similarity of $A_i$ to $A_j$. Since, for any two attributes $A_i$ and $A_j$, two similarities $A_i \rightarrow A_j$ and $A_j \rightarrow A_i$ are computed, only one with higher similarity factor is selected in the list of attribute similarity set $\text{ASS}$. Thus, for $n$ conditional attributes, $n(n-1)/2$ similarities are selected, out of which some are strong and some are not. Out of these similarities, the similarity with $\delta f_i$ value less than the average $\bar{\delta} f_i$ value are discarded from $\text{ASS}$ and rest is considered as the set of attribute similarity. So, each element $x$ in $\text{ASS}$ is of the form $x: A_i \rightarrow A_j$ such that $\text{Left}(x) = A_i$ and $\text{Right}(x) = A_j$. The algorithm “ASS_GEN” described below, computes the attribute similarity set $\text{ASS}$.

Algorithm: ASS_GEN$(C, \bar{\delta} f)$

/* Computes attribute similarity set $\{A_i \rightarrow A_j\}$ */

Input: $C = \text{set of conditional attributes}$ and $\delta f = 2-D$ contains similarity factors between each pair of conditional attributes.

Output: Attribute Similarity Set $\text{ASS}$

Begin

$\text{ASS} = \{\}, \text{sum}_{\delta f} = 0$

113
/* compute only \( \frac{n(n-1)}{2} \) elements in \( \text{ASS} \) */

for \( i = 1 \) to \(|C| - 1\)

\{
\{ for \( j = i+1 \) to \(|C|\)

\{ if(\( \frac{\delta_i}{\delta_j} > \frac{\delta_j}{\delta_i} \))

\{ \begin{align*}
\text{sum}_\delta_i &= \text{sum}_\delta_i + \frac{\delta_i}{\delta_j} \\
\text{ASS} &= \text{ASS} \cup \{ Ai \rightarrow Aj \}
\end{align*}
\}
\}
else

\{ \begin{align*}
\text{sum}_\delta_i &= \text{sum}_\delta_i + \frac{\delta_j}{\delta_i} \\
\text{ASS} &= \text{ASS} \cup \{ Aj \rightarrow Ai \}
\end{align*}
\}
\}
\}

/* modify \( \text{ASS} \) by only elements \( Ai \rightarrow Aj \) for which \( \frac{\delta_i}{\delta_j} > \text{avg}_\delta \) */

\( \text{ASS}_{\text{mod}} = \{ \} \)

\( \text{avg}_\delta = \frac{2 \times \text{sum}_\delta}{|C|(|C|-1)} \)

for each \( \{ Ai \rightarrow Aj \} \in \text{ASS} \)

\{ if(\( \frac{\delta_i}{\delta_j} > \text{avg}_\delta \))

\{ \begin{align*}
\text{ASS}_{\text{mod}} &= \text{ASS}_{\text{mod}} \cup \{ Ai \rightarrow Aj \} \\
\text{ASS} &= \text{ASS} - \{ Ai \rightarrow Aj \}
\end{align*}
\}
\}

\text{ASS} = \text{ASS}_{\text{mod}}

End

Algorithm "ASS_GEN" is applied and Table 4 is constructed from Table 2, where only six out of twelve attribute similarities in Table 2 are considered. Thus, initially, \( \text{ASS} = \{ i \rightarrow f, i \rightarrow r, e \rightarrow i, e \rightarrow f, e \rightarrow r, r \rightarrow f \} \) and \( \text{avg}_\delta = 0.786 \). As the similarity factor for attribute similarities \( i \rightarrow f, e \rightarrow i, e \rightarrow f \) and \( r \rightarrow f \) are greater than \( \text{avg}_\delta \), they are considered in the final attribute similarity set \( \text{ASS} \). So, finally, \( \text{ASS} = \{ i \rightarrow f, e \rightarrow i, e \rightarrow f, r \rightarrow f \} \).
Table 4. Illustrates the selection of attribute similarities.

| Attribute Similarity | Similarity factor of $A_i$ to $A_j$ | $\delta_r > \delta_t$ |
|----------------------|-------------------------------------|-----------------------|
| i→f                  | $\delta_f = 0.8$                    | Yes                   |
| i→r                  | $\delta_r = 0.7$                    |                       |
| e→i                  | $\delta_i = 0.83$                   | Yes                   |
| e→f                  | $\delta_f = 0.83$                   | Yes                   |
| e→r                  | $\delta_r = 0.76$                   |                       |
| r→f                  | $\delta_f = 0.8$                    | Yes                   |
| Average $\delta_r$   | 0.786                               |                       |

3. Single Reduct Generation

The attribute similarity obtained so far is known as simple similarity of an attribute to other attribute. But, for simplifying the reduct generation process, the elements in ASS are minimized by combining some simple similarity. The new similarity obtained by the combination of some of the simple similarity is called compound similarity. Here, all $x$ from ASS with same Left($x$) are considered and obtained compound similarity is $\text{Left}(x) \rightarrow \cup \text{Right}(x)$ $\forall x$. Thus, introducing compound similarity, the set ASS is refined to a set with minimum elements so that for each attribute, there is at most one element in ASS representing either simple or compound similarity of the attribute. The detail algorithm for determining compound attribute similarity set is given below:

Algorithm: COMP_SIM(ASS)

/* Compute the compound attribute similarity of attributes*/

Input: Simple attribute similarity set ASS

Output: Compound attribute similarity set ASS

Begin

for each $x \in$ ASS

{  for each $y \neq x$ \in ASS

   { if(Left($x$) = = Left($y$)) then

     { Right($x$) = Right($x$) \cup Right($y$)

     ASS = ASS – \{y\}

     }

   }

}


Finally, from the compound attribute similarity set ASS, reduct is generated. First of all, select an element, say, x from ASS for which length of Right(x) i.e., |Right(x)| is maximum. This selection guaranteed that the attribute Left(x) is similar to maximum number of attributes and so Left(x) is an element of reduct RED. Then, all elements z of ASS for which Left(z) ⊆ Right(x) are deleted and also x is deleted from ASS. This process is repeated until the set ASS becomes empty which provides the reduct RED. The proposed single reduct generation algorithm is discussed below:

Algorithm: SIN_RED_GEN(ASS, RED)

Input: Compound attribute similarity set ASS

Output: Single reduct RED

Begin

RED = φ

While (ASS ≠ φ)

{  max = 0
  for each x ∈ ASS
  
  {   if(|Right(x)| > max) then
  
    {   max = |Right(x)|

        L = Left(x)

    }

  }

  for each x ∈ ASS

  {   if (Left(x) = = L) then

  
    {   RED = RED ∪ Left(x)

        R = Right(x)

        ASS = ASS – {x}

        for each z ∈ ASS

          if(Left(z) ⊆ R) then

            ASS = ASS – {z}

            break

        }

  }

  }

} /*end-while*/

Return (RED)
End

Applying “COMP_SIM” algorithm the set \( ASS = \{i \rightarrow f, e \rightarrow i, e \rightarrow f, r \rightarrow f\} \) is refined to compound similarity set \( ASS = \{i \rightarrow f, e \rightarrow \{i, f\}, r \rightarrow f\} \). So, the selected element from \( ASS \) is \( e \rightarrow \{i, f\} \), and thus \( e \in RED \) and \( ASS \) is modified as \( ASS = \{r \rightarrow f\} \). And, in the next iteration, \( r \in RED \) and \( ASS = \emptyset \). Thus, \( RED = \{e, r\} \).

4. Results and discussions

The proposed method computes a single reduct for datasets collected from UCI machine learning repository [12]. At first, all the numeric attributes are discretized by ChiMerge [13] discretization algorithm. To measure the efficiency of the method, k-fold cross-validations, where \( k \) ranges from 1 to 10 have been carried out on the dataset and classified using “Weka” tool [14]. The proposed method (PRP) and well known dimensionality reduction methods, such as,Correlated Feature Subset (CFS) method [15] and Consistency Subset Evaluator (CON) method [16] have been applied on the dataset for dimension reduction and the reduced datasets are classified on various classifiers. Original number of attributes, number of attributes after applying various reduction methods and the accuracies (in %) of the datasets are computed and listed in Table 5, which shows the efficiency of the proposed method.

**Table 5.** Accuracy Comparison of Proposed, CFS and CON methods
5. Conclusion

The relative indiscernibility relation introduces in the paper is an equivalence relation which induces a partition of equivalence classes for each attribute. Then, the degree of similarity is measured between two attributes based on their equivalence classes. Since, the target of the paper is to compute reduced attribute set for decision making, so application of equivalence classes for similarity measurement is the appropriate choice.

References

[1] Pawlak, Z.: “Rough sets.: International journal of information and computer sciences,” Vol. 11, pp. 341-356 (1982)

[2] Pawlak, Z.: “Rough set theory and its applications to data analysis,” Cybernetics and systems 29 (1998) 661-688, (1998)

[3] X. Hu, T.Y. Lin, J.Jianchao, “A New Rough Sets Model Based on Database Systems”, Fundamental Informaticae, pp.1-18, 2004

[4] R.Jensen, Qiang Shen, “Fuzzy-Rough Attribute Reduction with Application to Web Categorization”, Fuzzy Sets and Systems, Vol.141, No.3, pp.469-485, 2004

[5] N.Zhong and A. Skowron, “A Rough Set-Based Knowledge Discovery Process”, Int. Journal of Applied Mathematics and Computer Science., 11(3), 603-619, 2001. BIME Journal, Volume (05), Issue 2, 2005

[6] J. Komorowski and A.Ohrn, “Modelling Prognostic Power of Cardiac tests using rough sets”, Artificial Intelligence in Medicine, 15, 167-191, 1999.

[7] J.Bazan, “A Comparison of dynamic and nondynamic rough set methods for extracting laws from decision tables”, Rough Sets in Knowledge Discovery, Physica Verlag, 1998.

[8] U.Carlin and J.Komorowski and A.Ohrn, “Rough Set Analysis of Patients with Suspected Acute Appendicitis”, Proc., IPMU, 1998.
119

[9] Devroye, L., Gyorfi, L., and Lugosi, G.: A Probabilistic Theory of Pattern Recognition. Newyork: Springer-Verlag, (1996)

[10] Gupta, S., C., and Kapoor, V., K.: “Fundamental of Mathematical Statistics”, Published by: Sultan Chand & Sons, A.S. Printing Press, India, (1994)

[11] Pal, S., K., and Mitra, S.: Neuro-Fuzzy pattern Recognition: Methods in Soft Computing, New York: Willey (1999)

[12] Murphy, P. and Aha, W.: UCI repository of machine learning databases (1996), http://www.ics.uci.edu/ml/learn/MLRepository.html

[13] Kerber, R., ChiMerge: Discretization of Numeric Attributes, in Proceedings of AAAI-92, Ninth Int'l Conf. Artificial Intelligence, AAAI-Press, pp. 123-128, (1992)

[14] WEKA: Machine Learning Software, http://www.cs.waikato.ac.nz/ml/

[15] M.A. Hall, Correlation-Based Feature Selection for Machine Learning PhD thesis, Dept. of Computer Science, Univ. of Waikato, Hamilton, New Zealand, (1998)

[16] Liu and R. SetionoA Probabilistic Approach to Feature Selection: A Filter Solution Proc.13th Int'l Conf. Machine Learning, pp. 319-327, (1996)

Authors

Ms Shampa Sengupta is an Assistant Professor of Information Technology at MCKV Institute Of Engineering, Liluah, Howrah. She has received M.Tech degree in Information Technology from Bengal Engineering and Science University, Shibpur, Howrah. Since 2011, she has been working toward the PhD degree in Data Mining at Bengal Engineering and Science University, Shibpur, Howrah.

Dr. Asit Kr. Das is an Assistant Professor of Computer Science and Technology at Bengal Engineering and Science University, Shibpur, Howrah. He has received M.Tech degree in Computer Science and Engg from Calcutta University. He obtained PhD(Engg) degree from Bengal Engineering and Science University, Shibpur, Howrah. His research interests include Data Mining and Pattern Recognition and Rough Set Theory, Bio-informatics etc.