Single Reduct Generation by Attribute Similarity Measurement Based on Relative Indiscernibility

Shampa Sengupta[1] and Kr. Das[2]

[1] M.C.K.V. Institute of Engineering - 243, G.T. Road (North), Liluah, Howrah 711204, West Bengal
[2] 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 al. [3] developed two new algorithms

N. Meghanathan et al. (Eds.): CCSIT 2012, Part II, LNICST 85, pp. 476–487, 2012.

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. 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 2, 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, $IND(P) = \{(x, y) \in U \times U \mid \forall\ a \in P,\ f_a(x) = f_a(y)\}$. 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 $\{[x]_p\}$ 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 Ai 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).

$$RIR_D(A_i) = \{(x, y) \in \Pi_{A_i}[x]_D \times \Pi_{A_i}[x]_D \mid f_{A_i}(x) = f_{A_i}(y) \forall\ [x]_D \in U/D\} \tag{1}$$

For each conditional attribute A_i, a relative indiscernibility relation $RIR_D(A_i)$ partitions the set of objects into a n-number of equivalence classes [ ] defined as partition $U/\ RIR_D(A_i)$ or U_D/A_i is equal to $\{[x]_{A_{i/D}}\}$ where $|\ U_D/A_i\ | = n$. Obviously, each equivalence class $\{[x]_{A_{i/D}}\}$ 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(i)	*Experience*(e)	*French*(f)	*Reference*(r)	*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)
$U/_D = (\{x_1, x_4, x_7\}, \{x_2, x_3, x_5, x_6, x_8\})$	$UD/_i = (\{x_1, x_7\}, \{x_2\}, \{x_3, x_8\}, \{x_4\}, \{x_5, x_6\})$
$U/_i = (\{x_1, x_2, x_7\}, \{x_3, x_8\}, \{x_4, x_5, x_6\})$	$UD/_e = (\{x_1\}, \{x_5\}, \{x_2, x_3, x_8\}, \{x_4, x_7\}, \{x_6\})$
$U/_e = (\{x_1, x_5\}, \{x_2, x_3, x_8\}, \{x_4, x_6, x_7\})$	$UD/_f = (\{x_1, x_4\}, \{x_2, x_3, x_5, x_6\}, \{x_7\}, \{x_8\})$
$U/_f = (\{x_1, x_2, x_3, x_4, x_5, x_6\}, \{x_7, x_8\})$	$UD/_r = (\{x_1\}, \{x_6, x_8\}, \{x_2, x_5\}, \{x_4\}, \{x_3, x_7\})$
$U/_r = (\{x_1, x_6, x_8\}, \{x_2, x_4, x_5\}, \{x_3, x_7\})$	

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 Ai is said to be similar to an attribute A_j with degree of similarity (or similarity factor) $\delta_f^{i,j}$ 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 $(\delta_f^{i,j} \times 100)\%$, where $\delta_f^{i,j}$ is computed by equation (2). The details for computation of similarity measurement for the attribute similarity Ai → Aj (Ai ≠ Aj) is described in algorithm "SIM_FAC" below.

$$\delta_f^{i,j} = \frac{1}{|U_D/A_i|} \sum_{[x]_{A_{i/D}} \in U_D/A_i} \frac{1}{|[x]_{A_{i/D}}|} \max_{[x]_{A_{j/D}} \in U_D/A_j} ([x]_{A_{i/D}} \cap [x]_{A_{j/D}}) \tag{2}$$

Algorithm: SIM_FAC(A_i , A_j)/* Similarity factor computation for attribute similarity Ai → Aj */

```
Input:  Partitions U_D/A_i = {[x]_{A_{i/D}}} and U_D/A_j = {[x]_{A_{j/D}}}
      obtained by applying relative indiscernibility relation
      RIR_D on A_i and A_j respectively.
Output: Similarity factor δ_f^{i,j}

Begin
   For each conditional attribute A_i {
     /* compute relative indiscernibility RIRD (A_i) using (1)*/
```

$RIR_D(A_i) = \{(x,y) \in \Pi_{A_i}[x]_D \times \Pi_{A_i}[x]_D \mid f_{A_i}(x) = f_{A_i}(y) \forall [x]_D \in U/D\}$

```
        RIR_D (A_i) induces equivalence classes U_D/A_i = {[x]_{A_i/D}}
    } /*end of for*/
    /* similarity measurement of A_i to A_j */
```

$\delta_f^{i,j} = 0$

```
    For each [x]_{i/D} ∈ U_D/A_i
    {   max_overlap = 0
        For each [x]_{j/D} ∈ U_D/A_j
        {   overlap = | [x]_{i/D} ∩ [x]_{j/D} |
            if (overlap > max_overlap) then
                 max_overlap = overlap
        }
```

$$\delta_f^{i,j} = \delta_f^{i,j} + \frac{\max\_overlap}{|[x]_{i/D}|}$$

```
    }
```

$$\delta_f^{i,j} = \frac{\delta_f^{i,j}}{|U_D/A_i|}$$

```
End.
```

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 3. Describe the degree of similarity of all pair of attributes

Attribute Similarity ($A_i \rightarrow A_j$)	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 ($\delta_f^{i,j}$)
i → e	$\{x_1, x_7\}, \{x_2\}, \{x_3, x_8\}, \{x_4\}, \{x_5, x_6\}$	$\{x_1\}, \{x_5\}, \{x_2, x_3, x_8\}, \{x_4, x_7\}, \{x_6\}$	$\delta_f^{i,e} = 0.8$
i → f	$\{x_1, x_7\}, \{x_2\}, \{x_3, x_8\}, \{x_4\}, \{x_5, x_6\}$	$\{x_1, x_4\}, \{x_2, x_3, x_5, x_6\}, \{x_7\}, \{x_8\}$	$\delta_f^{i,f} = 0.8$
i → r	$\{x_1, x_7\}, \{x_2\}, \{x_3, x_8\}, \{x_4\}, \{x_5, x_6\}$	$\{x_1\}, \{x_6, x_8\}, \{x_2, x_5\}, \{x_4\}, \{x_3, x_7\}$	$\delta_f^{i,r} = 0.7$
e → i	$\{x_1\}, \{x_5\}, \{x_2, x_3, x_8\}, \{x_4, x_7\}, \{x_6\}$	$\{x_1, x_7\}, \{x_2\}, \{x_3, x_8\}, \{x_4\}, \{x_5, x_6\}$	$\delta_f^{e,i} = 0.83$
e → f	$\{x_1\}, \{x_5\}, \{x_2, x_3, x_8\}, \{x_4, x_7\}, \{x_6\}$	$\{x_1, x_4\}, \{x_2, x_3, x_5, x_6\}, \{x_7\}, \{x_8\}$	$\delta_f^{e,f} = 0.83$
e → r	$\{x_1\}, \{x_5\}, \{x_2, x_3, x_8\}, \{x_4, x_7\}, \{x_6\}$	$\{x_1\}, \{x_6, x_8\}, \{x_2, x_5\}, \{x_4\}, \{x_3, x_7\}$	$\delta_f^{e,r} = 0.76$

Table 3. (*Continued*)

f → i	$\{x_1, x_4\}, \{x_2, x_3, x_5, x_6\}, \{x_7\}, \{x_8\}$	$\{x_1, x_7\}, \{x_2\}, \{x_3, x_8\}, \{x_4\}, \{x_5, x_6\}$	$\delta_f^{f,i} = 0.75$
f → e	$\{x_1, x_4\}, \{x_2, x_3, x_5, x_6\}, \{x_7\}, \{x_8\}$	$\{x_1\}, \{x_5\}, \{x_2, x_3, x_8\}, \{x_4, x_7\}, \{x_6\}$	$\delta_f^{f,e} = 0.75$
f → r	$\{x_1, x_4\}, \{x_2, x_3, x_5, x_6\}, \{x_7\}, \{x_8\}$	$\{x_1\}, \{x_6, x_8\}, \{x_2, x_5\}, \{x_4\}, \{x_3, x_7\}$	$\delta_f^{f,r} = 0.75$
r → i	$\{x_1\}, \{x_6, x_8\}, \{x_2, x_5\}, \{x_4\}, \{x_3, x_7\}$	$\{x_1, x_7\}, \{x_2\}, \{x_3, x_8\}, \{x_4\}, \{x_5, x_6\}$	$\delta_f^{r,i} = 0.7$
r → e	$\{x_1\}, \{x_6, x_8\}, \{x_2, x_5\}, \{x_4\}, \{x_3, x_7\}$	$\{x_1\}, \{x_5\}, \{x_2, x_3, x_8\}, \{x_4, x_7\}, \{x_6\}$	$\delta_f^{r,i} = 0.7$
r → f	$\{x_1\}, \{x_6, x_8\}, \{x_2, x_5\}, \{x_4\}, \{x_3, x_7\}$	$\{x_1, x_4\}, \{x_2, x_3, x_5, x_6\}, \{x_7\}, \{x_8\}$	$\delta_f^{r,f} = 0.8$

The computation of $\delta_f^{i,j}$ of each attribute similarity using equation (2) in Table 2 can be understood by Table 3, in which similarity i → e in first row of Table 3 is considered, where, $U_D/i = \{x1, x7\}, \{x2\}, \{x3, x8\}, \{x4\}, \{x5, x6\})$ and $U_D/e = \{x1\}, \{x5\}, \{x2, x3, x8\}, \{x4, x7\}, \{x6\})$.

Table 4. Illustrates the similarity factor computation for i → e

$[x]_{i/D}$ **of** U_D/i	**Overlapping** $[x]_{e/D}$ **of** U_D/e **with** $[x]_{i/D}$ **of** U_D/i	$[x]_{i/D} \cap [x]_{e/D}$	$T = \frac{1}{\lvert[x]_{i/D}\rvert} \max_{[x]_{e/D} \in U_D/e}([x]_{i/D} \cap [x]_{e/D})$
$\{x_1, x_7\}$	$\{x_1\}$ $\{x_4, x_7\}$	$\{x_1, x_7\} \cap \{x_1\}$ $\{x_1, x_7\} \cap \{x_4, x_7\}$	$\frac{1}{2}$
$\{x_2\}$	$\{x_2, x_3, x_8\}$	$\{x_2\} \cap \{x_2, x_3, x_8\}$	$\frac{1}{1}$
$\{x_3, x_8\}$	$\{x_2, x_3, x_8\}$	$\{x_3, x_8\} \cap \{x_2, x_3, x_8\}$	$\frac{2}{2}$
$\{x_4\}$	$\{x_4, x_7\}$	$\{x_4\} \cap \{x_4, x_7\}$	$\frac{1}{1}$
$\{x_5, x_6\}$	$\{x_5\}$ $\{x_6\})$	$\{x_5, x_6\} \cap \{x_5\}$ $\{x_5, x_6\} \cap \{x_6\}$	$\frac{1}{2}$
$\delta_f^{ie} = \frac{1}{\lvert[x]_{i/D}\rvert} \sum_{[x]_{i/D} \in U_D/i} T = \frac{1}{5}\left(\frac{1}{2} + \frac{1}{1} + \frac{2}{2} + \frac{1}{1} + \frac{1}{2}\right) = \frac{4}{5} = 0.8$			

2.4 Attribute Similarity Set

For each pair of conditional attributes (Ai, Aj), 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 RIRD(Ai) and RIRD(Aj) produce highly similar equivalence classes. This implies that both the attributes Ai and Aj have almost similar classification power and so $A_i \rightarrow A_j$ is considered as strong similarity of Ai to Aj. Since, for any two attributes Ai and Aj, 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 ASS. Thus, for n conditional attributes, n(n-1)/2similarities are selected, out of which some are strong and some are not. Out of these similarities, the similarity with $\delta_f^{i,j}$ value less than the average δ_f value are discarded from ASS and rest is considered as the set of attribute similarity. So, each element x in ASS is of the form x: Ai→Aj such that Left(x) = Ai and Right(x) = Aj. The algorithm "ASS_GEN" described below, computes the attribute similarity set ASS.

Algorithm: ASS_GEN(C, δ_f)

```
/* Computes attribute similarity set {Ai→Aj} */
  Input: C = set of conditional attributes and δf =2-D
         contains    similarity factors between each pair
         of conditional attributes.
Output: Attribute Similarity Set ASS
Begin
    ASS = {}, sum_δf = 0
    /* compute only n(n - 1)/2 elements in ASS */
    for i = 1 to |C| - 1
    {  for j = i+1 to |C|
       {  if(δf^{i,j} > δf^{j,i})then
          {    sum_δf = sum_δf + δf^{i,j}
               ASS = ASS ∪{Ai → Aj}
          }
          else
          {    sum_δf = sum_δf + δf^{j,i}
               ASS = ASS ∪{Aj → Ai}
          }
       }
    }
/* modify ASS by only elements Ai → Aj for which δf^{i,j}>avg_δf */
```

```
ASSmod = {}

avg_δf = (2× sum_δf) / |C|(|C|-1)

    for each {Ai → Aj}∈ ASS
    {   if(δf^{i,j} >avg_δf) then
        {   ASSmod = ASSmod ∪ {Ai → Aj}
            ASS = ASS - { Ai → Aj}
        }
    }
    ASS = ASSmod
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, ASS = {i → f, i → r, e → i, e → f, e → r, r → f} and avg_δ_f = 0.786. As the similarity factor for attribute similarities i → f, e → i, e → f and r → f are greater than avg_δ_f, they are considered in the final attribute similarity set ASS. So, finally, ASS = {i → f, e → i, e → f, r → f }.

Table 5. Illustrates the selection of attribute similarities

Attribute Similarity ($A_i \rightarrow A_j$; i ≠ j and $\delta_f^{i,j} > \delta_f^{j,i}$)	Similarity factor of A_i to A_j ($\delta_f^{i,j}$)	$\delta_f^{i,j} > \delta_f$
i→f	$\delta_f^{i,f}$ = 0.8	Yes
i→r	$\delta_f^{i,r}$ = 0.7	
e→i	$\delta_f^{e,i}$ = 0.83	Yes
e→f	$\delta_f^{e,f}$ = 0.83	Yes
e→r	$\delta_f^{e,r}$ = 0.76	
r→f	$\delta_f^{r,f}$ = 0.8	Yes
Average δ_f	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 Left(x) $\rightarrow \cup$ 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 ∈ ASS
        {   for each y (•x) ∈ ASS
            {   if(Left(x) = = Left(y)) then
                {   Right(x) = Right(x) ∪ Right(y)
                    ASS = ASS - {y}
                }
            }
       }
End
```

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) $\subseteq$ 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 → f, e → i, e → f, r → f} is refined to compound similarity set ASS = {i → f, e → {i, f}, r → f}. So, the selected element from ASS is e → {i, f}, and thus e ∈ RED and ASS is modified as ASS = {r → f}. And, in the next iteration, r ∈ RED and ASS =ϕ. Thus, RED = {e, r}.

4 Results and Discussions

The proposed method computes a single reduct for glass dataset selected 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 are carried out on the reduced dataset and classified using “Weka” tool [14]. The proposed method and well known dimensionality reduction methods such as ‘Cfs Subset Eval’ (CFS) method [15], ‘Consistensy Subset Evaluator’ (CON) method [16] are applied on the dataset and observed that the proposed method, CFS and CON reduce the number of attributes into six, six and seven whereas the actual number of attributes is nine. Then the reduced dataset is applied on various classifiers and accuracies are measured, listed in Table 5. Average accuracy by proposed method is much higher than that by CFS and CON.

Table 6. Accuracy comparison by proposed, CFS and CON reduction proces

Classifier	Proposed Method	CFS	CON
Naïve Bayes	65.73	43.92	47.20
SMO	62.44	57.94	57.48
KSTAR	83.57	79.91	78.50
AdaBoost	44.60	44.86	44.86
Bagging	76.53	73.83	71.50
Multi Class Classifier	64.32	66.36	64.49
J48	72.30	68.69	64.02
PART	77.00	70.94	68.69
Average accuracy (%)	**68.31**	**63.31**	**62.09**

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 11, 341–356 (1982)
2. Pawlak, Z.: Rough set theory and its applications to data analysis. Cybernetics and Systems 29, 661–688 (1998)
3. Hu, X., Lin, T.Y., Jianchao, J.: A New Rough Sets Model Based on Database Systems. Fundamental Informaticae, 1–18 (2004)
4. Jensen, R., Shen, Q.: Fuzzy-Rough Attribute Reduction with Application to Web Categorization. Fuzzy Sets and Systems 141(3), 469–485 (2004)
5. Zhong, N., Skowron, A.: A Rough Set-Based Knowledge Discovery Process. Int. Journal of Applied Mathematics and Computer Science 11(3), 603–619 (2005); BIME Journal 05(1) (2005)
6. Komorowski, J., Ohrn, A.: Modelling Prognostic Power of Cardiac tests using rough sets. Artificial Intelligence in Medicine 15, 167–191 (1999)
7. Bazan, J.: A Comparison of dynamic and nondynamic rough set methods for extracting laws from decision tables. In: Rough Sets in Knowledge Discovery. Physica Verlag (1998)
8. Carlin, U., Komorowski, J., Ohrn, A.: Rough Set Analysis of Patients with Suspected Acute Appendicitis. In: Proc., IPMU (1998)

9. Devroye, L., Gyorfi, L., Lugosi, G.: A Probabilistic Theory of Pattern Recognition. Springer, New York (1996)
10. Gupta, S.C., Kapoor, V.K.: Fundamental of Mathematical Statistics. Sultan Chand & Sons, A.S. Printing Press, India (1994)
11. Pal, S.K., Mitra, S.: Neuro-Fuzzy pattern Recognition: Methods in Soft Computing. Willey, New York (1999)
12. Murphy, P., Aha, W.: UCI repository of machine learning databases (1996), `http://www.ics.uci.edu/mlearn/MLRepository.html`
13. Kerber, R.: ChiMerge: Discretization of Numeric Attributes. In: Proceedings of AAAI 1992, Ninth Int'l Conf. Artificial Intelligence, pp. 123–128. AAAI-Press (1992)
14. WEKA: Machine Learning Software, `http://www.cs.waikato.ac.nz/~ml/`
15. Hall, M.A.: Correlation-Based Feature Selection for Machine Learning PhD thesis, Dept. ofComputer Science, Univ. of Waikato, Hamilton, New Zealand (1998)
16. Liu, H., Setiono, R.: A Probabilistic Approach to Feature Selection: A Filter Solution. In: Proc.13th Int'l Conf. Machine Learning, pp. 319–327 (1996)