www.icgst.com

BIME

Effective Classification with Improved Quick Reduct For Medical Database Using Rough System

K.Thangavel[1], P. Jaganathan[2*], A. Pethalakshmi[3], M.Karnan[2]

[1]*Department of Computer Science, Periyar University, Salem*,
Tamil Nadu, India. E-mail: drktvelu@yahoo.com.

[2*] *Department of Computer Science and Applications, Gandhigram Rural Institute-Deemed University*, Gandhigram-624 302, Tamil Nadu, India.
E-mail: jaganathodc@gmail.com

[3]*Department of Computer Science, M.V.M Government Arts College (W),*
Dindigul-624 001, Tamil Nadu, India.
E-mail: pethalakshmi@yahoo.com

Abstract

The volume of data being generated nowadays is increasing at phenomenal rate. Extracting useful knowledge from such data collections is an important and challenging issue. A promising technique is the Rough set theory, a new mathematical approach to data analysis based on classification of objects of interest into similarity classes, which are indiscernible with respect to some features. This theory offers two fundamental concepts: reduct and core. In this paper, Quick Reduct and the proposed Improved Quick Reduct Algorithms are first presented, followed by the c4.5 approach for rule induction. Some experiment results are also given. The redundant attributes are eliminated in order to generate the effective reduct set (i.e., reduced set of necessary attributes) or to construct the core of the attribute set. This paper analyses the efficiency of the proposed Improved Quickreduct Algorithm against the standard Quick Reduct Algorithm. The experimental works are carried out on medical data sets of UCI machine learning repository and the Human Immuno deficiency Virus(HIV) data set.

Keywords: *Rough set theory, Data mining, Knowledge discovery, Feature selection, Quickreduct, Improved Quickreduct.*

Corresponding Author

P.Jaganathan, PSNA College of Engineering &Tech, Dindigul-624 622, +91-451-2554032-821

1. Introduction

1.1 Data mining

The 1990s has brought a growing data glut problem to many fields such as science, business and govement. Our capabilities for collecting and storing data of all kinds have far outpaced our abilities to analyze, summarize, and extract “knowledge” from this data.

Traditional data analysis methods are no longer efficient to handle voluminous data sets. How to understand and analyze large bodies of data is a difficult and unresolved problem. The way to extract the knowledge in a comprehensible form fiom the huge amount of data is the primary concern. Rough set theory is an efficient technique for knowledge discovery in databases. Rough set theory is a relatively new mathematical technique to describe quantitatively uncertainty, imprecision and vagueness. Classical set theory deals with crisp sets and rough set theory may be considered an extension of the classical set theory. In rough set theory, objects are classified into similarity classes, which are clusters of indiscernible ones with respect to some features. The indiscernible classifications are the basic building blocks of knowledge and they are used to find out hidden patterns in data.

Data mining refers to extracting or “mining” knowledge from large amounts of data. There are many other terms carrying a similar or slightly different meaning to data mining, such as knowledge mining from databases, knowledge extraction, data pattern analysis, data archaeology, and data dredging. Data mining treats as synonym for another popularly used term, Knowledge Discovery in Databases, or KDD[7]. KDD consists of the following steps to process it such as Data cleaning, Data integration, Data selection, Data transformation, data mining, Pattern evaluation and Knowledge presentation.

KDD is the nontrivial process of identifying valid, novel, potentially useful and ultimately understandable patterns in data. Data mining is not a single technique, some commonly used techniques are: Statistical Methods, Case-Based Reasoning (CBR), Neural Networks, Decision Trees, Rule Induction, Bayesian Belief Networks (BBN), Genetic Algorithms, Fuzzy Sets and Rough Sets.

1.2 Rough Set Based Feature Reduction

In 1982, Pawlak introduced the theory of Rough sets [19, 20]. This theory was initially developed for a finite universe of discourse in which the knowledge base is a partition, which is obtained by any equivalence relation defined on the universe of discourse. In rough sets theory, the data is organized in a table called decision table. Rows of the decision table correspond to objects, and columns correspond to attributes. In the data set, a class label to indicate the class to which each row belongs. The class label is called as decision attribute, the rest of the attributes are the condition attributes. Here, C is used to denote the condition attributes, D for decision attributes, where $C \cap D = \Phi$, and t_j denotes the j^{th} tuple of the data table. Rough sets theory defines three regions based on the equivalent classes induced by the attribute values: lower approximation, upper approximation, and boundary. Lower approximation contains all the objects, which are classified surely based on the data collected, and Upper approximation contains all the objects, which can be classified probably, while the boundary is the difference between the upper approximation and the lower approximation.

Let U be any finite universe of discourse. Let R be any equivalence relation defined on U. Clearly, the equivalence relation partitions U. Here, (U, R) which is the collection of all equivalence classes, is called the approximation space. Let W_1, W_2, W_3 , ..., W_n be the elements of the approximation space (U, R). This collection is known as knowledge base. Then for any subset A of U, the lower and upper approximations are defined as follows:

$$\underline{R}A = \cup\{W_i / W_i \subseteq A\}$$

$$\overline{R}A = \cup \{W_i / W_i \cap A \neq \varnothing\}$$

The ordered pair ($\underline{R}A$, $\overline{R}A$) is called a rough set. Once defined these approximations of A, the reference universe U is divided into three different regions: the positive region $POS_R(A)$, the negative region $NEG_R(A)$ and the boundary region $BND_R(A)$, defined as follows:

$$POS_R(A) = \underline{R}A$$
$$NEG_R(A) = U - \overline{R}A$$
$$BND_R(A) = \overline{R}A - \underline{R}A$$

Hence, it is trivial that if BND(A) = Φ, then A is exact. This approach provides a mathematical tool that can be used to find out all possible reduces.

Two kinds of attributes are generally perceived as being unnecessary: attributes that are irrelevant to the target concept (like the row ID, customer ID), and attributes that are redundant given other attributes. In actual applications, these two kinds of unnecessary attributes can exist at the same time but the latter redundant attributes are more difficult to eliminate because of the interactions between them. In order to reduce both kinds of unnecessary attributes to a minimum, feature selection is used. Feature selection process refers to choose a subset of attributes from the set of original attributes. Feature selection has been studied intensively in the past one decade [12, 13, 16, 17, 26, 27]. The purpose of the feature selection is to identify the significant features, eliminate the irrelevant of dispensable features to the learning task, and build a good learning model. The benefits of feature selection are twofold: it considerably decreased the computation time of the induction algorithm and increased the accuracy of the resulting mode.

A decision table may have more than one reduct. Anyone of them can be used to replace the original table. Finding all the reducts from a decision table is NP-Hard [15]. Fortunately, in many real applications it is usually not necessary to find all of them. One is sufficient. A natural question is which reduct is the best if there exist more than one reduct. The selection depends on the optimality criterion associated with the attributes. If it is possible to assign a cost function to attributes, then the selection can be naturally based on the combined minimum cost criteria. In the absence of an attribute cost function, the only source of information to select the reduct is the contents of the data table [17]. For simplicity, we adopt the criteria that the best reduct is the one with the minimal number of attributes and that if there are two or more reducts with the same number of attributes, then the reduct with the least number of combinations of values of its attributes is selected.

1.3 Literature review

Besides the introduction given here, the extensive literature of Rough sets theory can be referred to Orlowska[18], Peters et al.[21], Polkowski et al.[22] for recent comprehensive overviews of developments.

Hu et al.[8] 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.[9, 10, 11] 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.[28] 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.[14] studies an application of rough sets to modeling prognostic power of cardiac tests.

Bazan[2] compares rough set-based methods, in particular dynamic reducts, with statistical methods, neural networks, decision trees and decision rules. He analyzes data sets including medical data on lymphography, breast cancer and primary tumors, and finds that error rates for rough sets are fully comparable as well as often significantly lower than the ones obtained with other techniques. Carlin et al.[4] presents an application of rough sets to diagnosing suspected acute appendicitis.
The rest of the paper is organized as follows: section 2, briefs about the data sets used for this study. Section 3 describes the feature reduction algorithm using Quickreduct.The Improved Quickreduct algorithm and its implementation are described in section 4. Section 5 describes the c4.5 Algorithm. Section 6 explains the experimental analysis of Quickreduct and Improved Quickreduct, and Section 7 states the conclusion of this paper and the directions for further research are proposed herein.

2. Data Preparation

The medical data sets viz., Postoperative, Pima, New-Thyroid obtained from UCI machine learning repository [3] and the real HIV data set are considered for this study. The HIV database consists of information collected from the HIV Patients at Voluntary Counselling and Testing Centre (VCTC) of Government Hospital, Dindigul District, Tamilnadu, India, a well-known centre for diagnosis and treatment of HIV. The advantage of this data set is that it includes a sufficient number of records of different categories of people affected by HIV. The set of descriptors presents all the required information about patients. It contains the records of 500 patients. The record of every patient contains 49 attributes and this has been reduced to 22 attributes after consulting the Physician. The details of attributes are given as follows: The continuous attributes are Age, Sex, Marital-Status, Occupation, Area, Loss-of-Weight, Continuous-Fever, Continuous-Cough, Skin-Disease, Oral-Thrush, Tuberculosis, Diarrahoea, Anaemia, Sexual–Transmission-Disease, Swelling-on-Neck, Different-Count, Total-Count, Erythrocyte-Rate, Creatinine, Loss-of-Appetite, Lymphodenopathy and the decision attribute Result (Positive, Negative, Suspect).

3. Quickreduct Algorithm(QR)

The reduction of attributes is achieved by comparing equivalence relations generated by sets of attributes. Attributes are removed so that the reduced set provides the same predictive capability of the decision feature as the original. A reduct is defined as a subset of minimal cardinality R_{min} of the conditional attribute set C such that $\gamma_R(D) = \gamma_C(D)$.

$$R = \{X : X \subseteq C; \gamma_X(D) = \gamma_C(D)\}$$
$$R_{min} = \{X : X \in R; \forall Y \in R; |X| \leq |Y| \}$$

The intersection of all the sets in R_{min} is called the core, the elements of which are those attributes that cannot be eliminated without introducing more contradictions to the dataset. In this method a subset with minimum cardinality is searched for.
The problem of finding a reduct of an information system has been the subject of much research in [1, 25]. The most basic solution to locating such a subset is to simply generate all possible subsets and retrieve those with a maximum rough set dependency degree. Obviously, this is an expensive solution to the problem and is only practical for very simple datasets. Most of the time only one reduct is required as, typically, only one subset of features is used to reduce a dataset, so all the calculations involved in discovering the rest are pointless.

To improve the performance of the above method, an element of pruning can be introduced. By noting the cardinality of any pre-discovered reducts, the current possible subset can be ignored if it contains more elements. However, a better approach is needed - one that will avoid wasted computational effort. The pseudo code of the Quickreduct is given below:

```
QUICKREDUCT(C,D)
C, the set of all conditional features;
D, the set of decision features.
(a)  $R \leftarrow \{\}$
(b)  Do
(c)  $T \leftarrow R$
(d)  $\forall x \in (C-R)$
(e)  if $\gamma_{R\cup \{x\}}(D) > \gamma_T (D)$
                  where $\gamma_R(D)=card(POS_R(D)) / card(U)$
(f)  $T \leftarrow R\cup\{x\}$
(g)  $R \leftarrow T$
(h)  until $\gamma_R(D) = = \gamma_C(D)$
return R
```

4. Improved Quickreduct Algorithm(IQR)

The Quickreduct algorithm attempts to calculate a reduct without exhaustively generating all possible subsets. It starts off with an empty set and adds in turn, one at a time, those attributes that result in the greatest increase in the rough set dependency metric, until this produces its maximum possible value for the dataset. According to the Quickreduct algorithm, the dependency of each attribute is calculated, and the best candidate chosen. This, however, is not guaranteed to find a minimal subset as has been shown in [5]. Using the dependency function to discriminate between candidates may lead the search down a non-minimal path. It is impossible to predict which combinations of attributes will lead to an optimal reduct based on changes in dependency with the addition or deletion of single attributes. It does result in a close-to-minimal subset, though, which is still useful in greatly reducing dataset dimensionality. In [5], a potential solution to this problem has been proposed whereby the Quickreduct algorithm is altered, making it into an n-look ahead approach. However, even this cannot guarantee a reduct unless n is equal to the original number of attributes, but this reverts back to generate-and-test. It still suffers from the same problem as the original Quickreduct, i.e. it is impossible to tell at any stage whether the current path will be the shortest to a reduct.

As per literature, the fuzzy p-lower and p-upper approximations are defined as [6]:

$$\mu \underline{p}X(F_i) = \inf_x \max\{1 - \mu F_i(x), \mu X(x)\} \qquad \forall i \qquad (1)$$

$$\mu \overline{p}X(F_i) = \sup_x \min\{ \mu F_i(x), \mu X(x)\} \qquad \forall i \qquad (2)$$

where F_i denotes a fuzzy equivalence class belonging to U / P. As the universe of discourse in feature selection is finite, the use of sup and inf are to be altered.

As a result of this, the fuzzy lower and upper approximations are herein redefined as:

$$\underline{p}X(x) = \max(F(x), (\inf_{y \in U} \max\{1 - F(y), X(y)\}) \qquad (3)$$

$$\overline{p}X(x) = \max(F(x), (\sup_{y \in U} \min\{ F(y), X(y)\}) \qquad (4)$$

Here F(i) denotes the normalized object values. Using the lower approximation in equation (3), the positive region can be defined as

$$Pos_{p(Q)}(x) = \sup_{X \in U/Q} \underline{p}X(x) \qquad (5)$$

Using the definition of the positive region, the new dependency function can be defined as follows:

$$\gamma_p(Q) = |Pos_{p(Q)}(x)| / |U| = \sum_{x \in U} Pos_{p(Q)}(x) / |U| \qquad (6)$$

The Quickreduct algorithm is improved herein under using the lower and upper approximations in equation (3) and (4) in order to obtain the best degree of dependency value by normalizing the information system, that is the maximum value is selected and all other values are divided by the maximum value [17]. Then apply the Improved Quickreduct Algorithm to obtain the minimal reduct.

The Improved Quickreduct Algorithm is as follows:

```
IMPROVED QUICKREDUCT(C, D)
C, the set of all conditional features
Q, the set of all decision features
        C ← the set of all normalized values
        RED ← {}
        Do
        TEMP ← RED
        $\gamma_{best} = 0$
        For $x \in C$
        if $\gamma_{RED \cup (x)}(Q) > \gamma_{best}(Q)$
              where $\underline{p}X(x) = \max(F(x), (\inf_{y \in U} \max\{1 - F(y), X(y)\})$
              $Pos_{p(Q)}(x) = \sup_{X \in U/Q} \underline{p}X(x)$
              $\gamma_p(Q) = |Pos_{p(Q)}(x)| / |U| = \sum_{x \in U} Pos_{p(Q)}(x) / |U|$
  TEMP ← RED ∪ (x)
  $\gamma_{best}$ ← TEMP
  RED ← TEMP
  until $\gamma_{best} == \gamma_C(Q)$
return RED
```

A system of 8 data points consisting four condition attributes and a decision attribute, borrowed from[8] is taken into consideration (Table 4.1). To train any Neural Network, the input values are to be normalized, that is the maximum value is selected and all other values are divided by the maximum value[24]. The same idea is used to get the Normalized Table 4.2.

Table 4.1: Data Set

OBJ	WEIGHT	DOOR	SIZE	CYLINDER	MILEAGE
1	LOW	2	COM	4	HIGH
2	LOW	4	SUB	6	LOW
3	MEDIUM	4	COM	4	HIGH
4	HIGH	2	COM	6	LOW
5	HIGH	4	COM	4	LOW

6	LOW	4	COM	4	HIGH
7	HIGH	4	SUB	6	LOW
8	LOW	2	SUB	6	LOW

Table 4.2: Normalized Values

OBJ	WEIGHT	DOOR	SIZE	CYLI NDER	MILEAGE	
					(1 ,3,6)	(2,4,5,7,8)
1	0.1667	0.3333	0.1667	0.6667	1	0
2	0.1667	0.6667	0.3333	1.0000	0	1
3	0.3333	0.6667	0.1667	0.6667	1	0
4	0.5000	0.3333	0.1667	1.0000	0	1
5	0.5000	0.6667	0.1667	0.6667	0	1
6	0.1667	0.6667	0.1667	0.6667	1	0
7	0.5000	0.6667	0.3333	1.0000	0	1
8	0.1667	0.3333	0.3333	1.0000	0	1

In Table 4.2, {Weight} is assigned to A, {Door} is assigned to B, {Size} is assigned to C, {Cylinder} is assigned to D and {Mileage} is assigned to Q. The lower approximations of A, B, C and D are calculated using the equation (3). The method of computing the lower approximation of the attribute A is elaborated here. For a class X = {1, 3, 6} in decision attribute, $\underline{A}_{\{1, 3, 6\}}(x)$ needs to be calculated in order to compute the decision equivalence class.

$$\underline{A}_{\{1, 3, 6\}}(x) = \max(F(x), (\inf_{y\in U} \max\{1 - F(y), X(y)\})$$

For object 1, this can be calculated as follows:

$\max (1 - a(1), X(1)) = \max(0.8333, 1.0) = 1.0$
$\max (1 - a(2), X(2)) = \max(0.8333, 0.0) = 0.8333$
$\max (1 - a(3), X(3)) = \max(0.6667, 1.0) = 1.0$
$\max (1 - a(4), X(4)) = \max(0.6667, 0.0) = 0.6667$
$\max (1 - a(5), X(5)) = \max(0.5000, 0.0) = 0.5000$
$\max (1 - a(6), X(6)) = \max(0.8333, 1.0) = 1.0$
$\max (1 - a(7), X(7)) = \max(0.5000, 0.0) = 0.5000$
$\max (1 - a(8), X(8)) = \max(0.8333, 0.0) = 0.8333$

Therefore, $\max(A(x), (\inf_{y\in U} \max\{1 - A(y), X(y)\})$

$$= \max(0.1667, \inf\{1.0, 0.8333, 1.0, 0.6667, 0.5000, 1.0, 0.5000, 0.8333\})$$
$$= \max(0.1667, 0.5000) = 0.5000$$

Thus, $\underline{A}_{\{1, 3, 6\}}(1) = 0.5000$. Similarly, $\underline{A}_{\{1, 3, 6\}}$ can be computed for the other objects. Then the corresponding class values for X = {2, 4, 5, 7, 8} can also be determined. For object1, $\underline{A}_{\{2, 4, 5, 7,8\}}(1) =0.6667$. Similarly, $\underline{A}_{\{2, 4, 5, 7, 8\}}$ can be computed for the other objects. Using these values, the positive region for each object can be calculated using

$$\mathrm{Pos}_{p(Q)}(x) = \sup_{X\in U/Q} \underline{A}X(x)$$

For object1,
$\mathrm{Pos}_{p(Q)}(1) = \sup(0.5000, 0.6667) = 0.6667$. Similarly, $\mathrm{Pos}_{p(Q)}$ can be computed for the other objects. The next step is to determine the degree of dependency of Q on A:

$$\gamma_A(Q) = \sum_{x\in U} \mathrm{Pos}_{A(Q)}(x) / |U| = 5.3336/ 8 = 0.6667$$

Similarly B, C and D are calculated to get the following degree of dependency

$\gamma_B(Q) = 0.5417$
$\gamma_C(Q) = 0.8333$
$\gamma_D(Q) = 0.8333$

The attribute 'A' is chosen and added to the potential reduct. The max(A ,B, C, D) should be calculated from the Table 3.2 to compute $\gamma_{\{A, B, C, D\}}(Q)$ which yields

$\gamma_{\{A, B, C, D\}}(Q) = 0.8333$

and then it has to be compared with $\gamma_A(Q)$.
Only if the values are equal, the reduct attributes can be obtained. Otherwise this attribute value should be combined with the next attribute value. That is, take the max(A, B) from the Table 4.2. This process iterates and the two dependency degrees are calculated as,
$\gamma_{\{A, B\}}(Q) = 0.6667$.

This value is equal to $\gamma_A(Q)$, and so "if" condition is not satisfied, hence the next combination of max(A, C) should be taken and the dependency degrees are calculated as

$\gamma_{(A, C)}(Q) = 0.8333$.

This value is greater than $\gamma_A(Q)$, and it is added to the potential reduct and the "until" condition is now satisfied , and the final output of reduct attributes for car data set {Weight, Size) is displayed.

5. C4.5 Rule Algorithm

C4.5 is a powerful induction algorithm developed by Quinlan[23]. C4.5 uses a divide and conquer strategy to produce decision trees, from which crisp rules can be derived. C4.5 is an extension of ID3 that accounts for unavailable values, continuous attributes value range , pruning decision trees and rule derivation. In building a decision tree, we can deal with training sets that have records with unknown attribute values by evaluating the gain, or the gain ratio, for an attribute by considering only those records where those attributes values are available. We can classify records that have unknown attributes values by eliminating the probability of the various possible results. Unlike CART, which generates a binary decision tree, c4.5 produces trees with variable branches per node. When a discrete variable is chosen as the splitting attributes in c4.5, there will be one branch for each value of the attributes.

6. Experimental Analysis

The Quickreduct algorithm has been implemented using MATLAB for medical databases available in the UCI data repository and the HIV data directly collected from the 500 HIV patients. However, the Quickreduct algorithm is not guaranteed to find a minimal subset. The fuzzy lower and upper approximations are used to improve the Quickreduct algorithm. The Improved Quickreduct algorithm is implemented using MATLAB and the Comparative Analysis of Quickreduct and Improved Quickreduct Algorithm is tabulated in Table 5.1 as follows:

Table 6.1:Comparative Analysis

DATA SET	INSTANCES	No. of Attr.	QR	IQR
POSTOPERATIVE	90	8	4	3
PIMA	768	8	5	3
NEW-THYROID	215	5	4	4
HIV	500	21	17	9

The reduced attribute set obtained for HIV data set after applying the Improved Quickreduct Algorithm is : Age, Sex, Occupation, Area, Loss-of-Weight, Continuous-Fever, Continuous-Cough, Sexual-Transmission-Disease and Swelling-on-neck.

The reduced data set is then used to construct the decision tree with the help of well known c4.5 Algorithm. By browsing the decision tree from the root node to leaf node, decision rules are generated, which in turn used to classify the new instances.The experiment is conducted with the following data sets available in UCI Data Repository. The results are tabulated in Table 6.2 and table 6.3

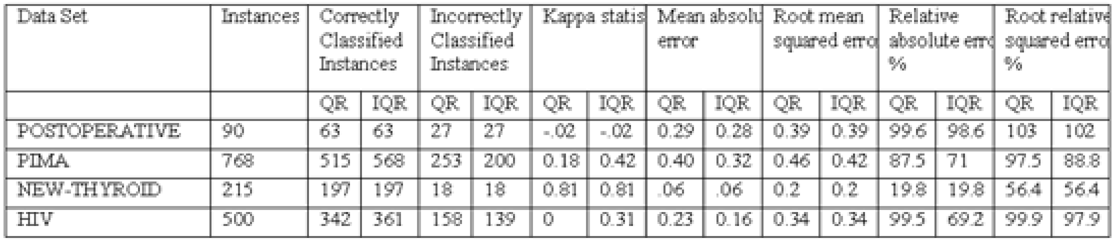
It is observed from the above tables, that the performance of the Improved Quick Reduct algorithm is comparable with the Quick reduct Algorithm and it out performs the later in some data sets. The various useful statistics with respect to rule induction are also provided in tables 6.2 and 6.3.

7. Conclusion

Quickreduct algorithm was implemented for medical database, which consists of both numeric and non-numeric attributes. In [9, 10, 11], Fuzzy-Rough Quickreduct can be used only for the numeric attribute values. In this approach, each attribute value is mapped into two classes with the help of fuzzy member function values. However, this could not be extended for non-numeric attributes. The given information system was normalized as it was done in the training process of Neural Network in the proposed algorithm and the computational procedure for computing degree of dependency of each attribute has been modified suitably. The Comparative Analysis Table 6.1 proves that the Improved Quickreduct produces minimal reduct from the data set containing large number of attributes. The rules induced through c4.5 Algorithm reveals that the improved Quick reduct is performing well. The effect of various reducts can also be studied with c4.5 algorithm, which is really suitable for real life data with continuous valued attributes.

The proposed work can be improved by introducing the Neural Network in order to train the system.

Table 6.2



Data Set	Instances	Correctly Classified Instances		Incorrectly Classified Instances		Kappa statis		Mean absolu error		Root mean squared erro		Relative absolute erro %		Root relative squared erro %	
		QR	IQR	QR	IQR	QR	IQR	QR	IQR	QR	IQR	QR	IQR	QR	IQR
POSTOPERATIVE	90	63	63	27	27	-.02	-.02	0.29	0.28	0.39	0.39	99.6	98.6	103	102
PIMA	768	515	568	253	200	0.18	0.42	0.40	0.32	0.46	0.42	87.5	71	97.5	88.8
NEW-THYROID	215	197	197	18	18	0.81	0.81	.06	.06	0.2	0.2	19.8	19.8	56.4	56.4
HIV	500	342	361	158	139	0	0.31	0.23	0.16	0.34	0.34	99.5	69.2	99.9	97.9

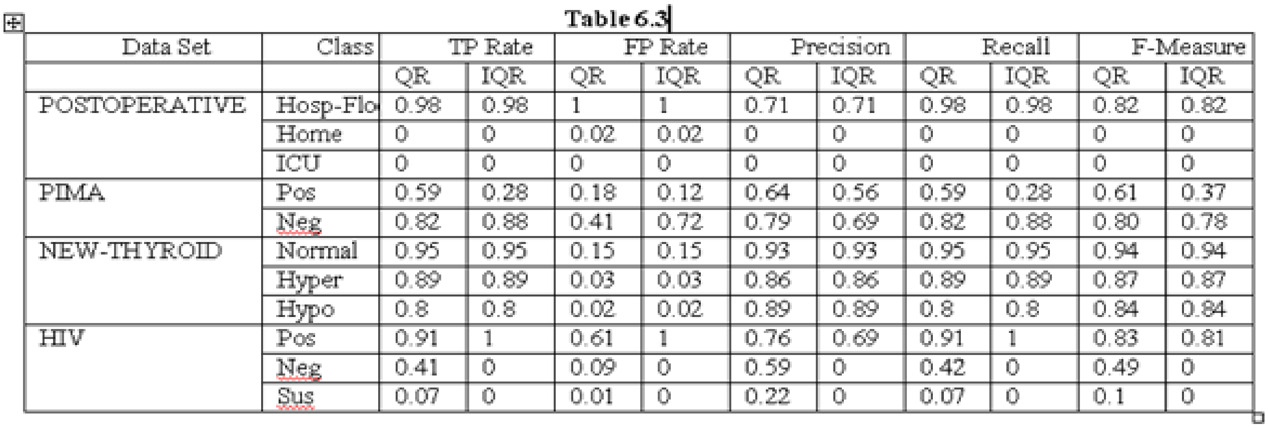


Table 6.3

Data Set	Class	TP Rate		FP Rate		Precision		Recall		F-Measure	
		QR	IQR	QR	IQR	QR	IQR	QR	IQR	QR	IQR
POSTOPERATIVE	Hosp-Flo	0.98	0.98	1	1	0.71	0.71	0.98	0.98	0.82	0.82
	Home	0	0	0.02	0.02	0	0	0	0	0	0
	ICU	0	0	0	0	0	0	0	0	0	0
PIMA	Pos	0.59	0.28	0.18	0.12	0.64	0.56	0.59	0.28	0.61	0.37
	Neg	0.82	0.88	0.41	0.72	0.79	0.69	0.82	0.88	0.80	0.78
NEW-THYROID	Normal	0.95	0.95	0.15	0.15	0.93	0.93	0.95	0.95	0.94	0.94
	Hyper	0.89	0.89	0.03	0.03	0.86	0.86	0.89	0.89	0.87	0.87
	Hypo	0.8	0.8	0.02	0.02	0.89	0.89	0.8	0.8	0.84	0.84
HIV	Pos	0.91	1	0.61	1	0.76	0.69	0.91	1	0.83	0.81
	Neg	0.41	0	0.09	0	0.59	0	0.42	0	0.49	0
	Sus	0.07	0	0.01	0	0.22	0	0.07	0	0.1	0

Author Information

Kuttiannan Thangavel received the master degree of science from the Department of Mathematics, Bharthidasan University in 1986 and Master of Computer Applications degree from Madurai Kamaraj University, Madurai in 2001. He obtained his Ph.D degree from Gandhigram Rural Institute-Deemed University, Gandhigram, Tamilnadu, India in 1996. Currently he is working as a Professor of Computer Science, Periyar University, Salem, Tamilnadu, India. He is in teaching since 1989. His areas of interest include Data mining, Genetic Algorithms, Neural Networks, Rough set theory, Machine Learning and Artificial Intelligence.

Palanichamy Jaganathan received the master degree of science in Physics and another master degree in Computer Applications from the Madurai Kamaraj University, Madurai, Tamilnadu, India, in 1993, and 2001, respectively and currently pursuing the Ph.D., degree in Computer Science and Applications from the Department of Computer Science and Applications, Gandhigram Rural Institute-Deemed University, Gandhigram, India. He joined the PSNA College of Engineering and Technology, Tamilnadu, India in 1995 and has been an Assistant Professor in the Department of Computer Applications. He has been in teaching since 1993. His areas of interest include Data mining, Genetic Algorithms, Neural networks, Machine Learning and Artificial intelligence.

PETHALAKSHMI ANNAMALAI, received the Master of Computer Science degree from Alagappa University, Tamil Nadu, India, in 1988 and the Master of Philosophy in Computer Science from Mother Teresa Women's University, Tamil Nadu, India, in 2000. She is working as Selection Grade Lecturer, Department of Computer Science, M.V.M. Govt. Arts College (w), Dindigul, Tamil Nadu, India, and currently she is the full-time research scholar in the Department of Computer Science, Mother Teresa Women's University, Tamil Nadu, India. Her areas of interests include fuzzy, rough set and neural network.

KARNAN MARCUS, received the Master of Computer Science and Engineering Degree from Computer Science and Engineering Department, from Government College of Engineering, Manonmaniam Sundaranar University, Tamil Nadu, India, in 2000. Currently he is working as Professor, Department of Computer Science & Engineering, HINDUSTHAN College of Engineering & Technology, Tamil Nadu, India. And doing part-time research in the Department of computer Science and Applications, Gandhigram Rural Institute-Deemed University, Tamil Nadu, India. His area of interests includes medical image processing, artificial intelligence, neural network, genetic algorithm, pattern recognition and fuzzy logic.

References

[1] J.J. Alpigini, J.F. Peters, J. Skowronek and N. Zhong (Eds.), "Rough Sets and Current Trends in Computing", Third International Conference, RSCTC 2002, Malvern, PA, USA, October 14-16, 2002, Proceedings. Lecture Notes in Computer Science 2475, Springer, ISBN 3-540-44274-X. 2002.

[2] J.Bazan, "A Comparison of dynamic and non-dynamic rough set methods for extracting laws from decision tables", Rough Sets in Knowledge Discovery, Physica Verlag, 1998.

[3] C.L.Blake, C.J. Merz, UCI Repository of machine learning databases, Irvine, University of California, 1998, http://www.ics.uci.edu/~mlearn/.

[4] U.Carlin and J.Komorowski and A.Ohrn, "Rough Set Analysis of Patients with Suspected Acute Appendicitis", Proc., IPMU, 1998.

[5] A.Chouchoulas, J.Halliwell, Q. Shen. "On the Implementation of Rough Set Attribute Reduction", Proceedings of the 2002 UK Workshop on Computational Intelligence, pp.18-23, 2002.

[6] D. Dubois, H. Prade, "Putting Rough Sets and Fuzzy Sets Together", Intelligent Decision Support, pp.203-232, 1992.

[7] J.Han and M.Kamber, "Datamining: Concepts and Techniques", Morgan Kaufmann Publishers, 1992.

[8] X. Hu, T.Y. Lin, J.Jianchao, "A New Rough Sets Model Based on Database Systems", Fundamenta Informaticae , pp.1-18, 2004.

[9] 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.

[10] R.Jensen, Qiang Shen, "Semantics Preserving Dimensionality Reduction: Rough and Fuzzy-Rough Based Approaches", IEEE Transactions on Knowledge and Data Engineering, Vol.16, No.12, pp. 1457-1471, 2004.

[11] R.Jensen, "Combining Rough and Fuzzy Sets for Feature Selection", Ph.D Thesis, School of Informatics, University of Edinburgh, 2005.

[12] G.H. John, R. Kohavi, K. Pfleger, "Irrelevant Features and the Subset Selection Problem", Proceedings of 11th International Conference on Machine Learning ICML94, pp.121-129, 1994.

[13] K.Kira, L.A. Rendell, "The Feature selection Problem: Traditional Methods and a New Algorithm", Proceedings of AAAI, MIT Press, 129-134, 1992.

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

[15] T.Y. Lin, N. Cercone (Eds.), "Rough sets and DataMining: Analysis of Imprecise Data", Kluwer Academic Publishers, 1997.

[16] H. Liu, H. Motoda (Eds.), "Feature Extraction Construction and Selection: A Datamining Perspective", (Kluwer International Series in Engineering and Computer Science), Kluwer Academic Publishers, 1998.

[17] M. Modrzejewski, "Feature Selection Using Rough Sets Theory", Proceedings of the 11th International Conference on Machine Learning, pp. 213-226,1993.

[18]E.Orlowska, "Incomplete Information:Rough Set Analysis", Physica-Verlag, Heidelberg, 1998.

[19] Z. Pawlak, "Rough Sets", International Journal of Computer and Information Sciences, Vol.11, No.5, pp. 341-356, 1982.

[20] Z.Pawlak, "Rough Sets: Theoritical Aspects and Reasoning about Data", Kluwer Academic Publishers, Dordrecht, 1991.

[21]J.F.Peters and A.Skowron(eds.), "Transactions on Rough Sets 1", Springer-Verlag, Berlin, 2004.

[22]L.Polkowski, S.Tsumoto and T.Y.Lin(eds.), "Rough Set Methods and Applications:New Developments in Knowledge Discovery in Information Systems", Physica-Verlag, Heidelberg, 2000.

[23] J.Quinlan, C4.5:Programs for Machine Learning, Morgan Kaufman, San Mateo, 1993.

[24] T. J. Ross, "Fuzzy Logic with Engineering Applications", McGraw-Hill, Inc, 1995.

[25] R.W. Swiniarski, A. Skowron, " Rough Set Methods in Feature Selection and Recognition", Pattern Recognition Letters, Vol. 24, No. 6, pp. 833-849, 2003.

[26] K.Thangavel and A.Pethalakshmi, "Feature Selection for Medical Database Using Rough System", International Journal on Artificial Intelligence and Machine Learning, Vol.5, No.4, 2005.

[27] K.Thangavel, M.Karnan and A.Pethalakshmi , "Performance Analysis of Rough Reduct Algorithms in Mammograms", International Journal on Graphics Vision and Image Processing, Vol.8, 13-21, 2005.

[28] 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.