Unsupervised Quick Reduct Algorithm Using Rough Set Theory

C. Velayutham and K. Thangavel

***Abstract*—Feature selection (FS) is a process to select features which are more informative. It is one of the important steps in knowledge discovery. The problem is that not all features are important. Some of the features may be redundant, and others may be irrelevant and noisy. The conventional supervised FS methods evaluate various feature subsets using an evaluation function or metric to select only those features which are related to the decision classes of the data under consideration. However, for many data mining applications, decision class labels are often unknown or incomplete, thus indicating the significance of unsupervised feature selection. However, in unsupervised learning, decision class labels are not provided. In this paper, we propose a new unsupervised quick reduct (QR) algorithm using rough set theory. The quality of the reduced data is measured by the classification performance and it is evaluated using WEKA classifier tool. The method is compared with existing supervised methods and the result demonstrates the efficiency of the proposed algorithm.**

***Index Terms*—Data mining, rough set, supervised and unsupervised feature selection, unsupervised quick reduct algorithm.**

1. Introduction

Feature selection (FS)[1] is an important problem in the research of knowledge discovery. In some cases, too many redundant or irrelevant features may overpower main features for classification. FS can bring about a remedy to this problem and therefore improve the prediction accuracy and reduce the computational overhead of classification algorithms. The high dimensionality of databases can be reduced using suitable techniques. The main aim of FS is to determine a minimal feature subset from a problem domain while retaining a suitably high accuracy in representing the original features.

Conventional supervised FS methods evaluate various feature subsets using an evaluation function or metric to select only those features which are related to, or lead to, the decision classes of the data under consideration. However, for many data mining applications, decision class labels are often unknown or incomplete, thus indicating the significance of unsupervised feature selection. In a broad sense, two different types of approach to unsupervised FS have been adopted: those which maximize clustering performance using an index function[2], [3], and those which consider features for selection on the basis of dependency or relevance. The central idea, behind the latter, is that any single feature which carries little or no further information than that subsumed by the remaining features is redundant and can therefore be eliminated[4]–[6]. The method described in this work is related to these techniques since it involves the removal of features which are considered to be redundant.

The rest of the paper is organized as follows. Section 2 gives an introduction to the rough set theory. Section 3 describes the existing supervised algorithms. Section 4 presents the proposed unsupervised quick reduct (USQR) algorithm. Section 5 describes the classifier tool. The experimental results are discussed in Section 6 and conclusion is presented in Section 7.

2. Rough Set Theory

Rough set theory (RST) has been used as a tool to discover data dependencies and to reduce the number of attributes contained in a dataset using the data alone, requiring no additional information[1], [6]. Over the past ten years, RST has become a topic of great interest to researchers and has been applied to many domains. Given a dataset with discretized attribute values, it is possible to find a subset (termed a reduct) of the original attributes using RST that are the most informative; all other attributes can be removed from the dataset with minimal information loss.

RST possesses many features in common (to a certain extent) with the Dempster-Shafer theory of evidence[4], and fuzzy set theory[7]. The rough set itself is the approximation

Manuscript received May 5, 2011; revised June 20, 2011. This work was supported by the UGC, SERO, Hyderabad under FDP during XI plan period, and the UGC, New Delhi for financial assistance under major research project Grant No. F-34-105/2008.

C. Velayutham is with the Department of Computer Science, Aditanar College of Arts and Science, Tiruchendur 62821, India (e-mail: cvelayutham22@yahoo.com).

K. Thangavel is with the Department of Computer Science, Periyar University, Salem 636011, India (e-mail: drktvelu@ yahoo.com).

Digital Object Identifier: 10.3969/j.issn.1674-862X.2011.03.001

of a vague concept (set) by a pair of precise concepts, called lower and upper approximations. These are the classification of the domain of interest into disjoint categories. The lower approximation is a description of the domain objects which are known with certainty to belong to the subset of interest; whereas the upper approximation is a description of the objects which possibly belong to the subset. The approximations are constructed with regard to a particular subset of features.

RST works by making use of the granularity structure of the data only. This is a major difference when compared with Dempster-Shafer theory and fuzzy set theory which require probability assignments and membership values respectively. However, this does not mean that no model assumptions are made. In fact by using only the given information, the theory assumes that the data is true and accurate reflection of the real world (which may not be the case). The numerical and other contextual aspects of the data are ignored which may seem to be a significant omission, but keeps model assumptions to a minimum.

Here the basic rough set concepts are introduced for the discussion in the rest of the paper.

Let $I=(U, A\bigcup\{d\})$ be an information system, where U is the universe with a non-empty set of finite objects, A is a non-empty finite set of condition attributes, and d is the decision attribute (such a table is also called decision table), $\forall a\in A$ there is a corresponding function $f_a: U\to V_a$, where V_a is the set of values of a. If $P\subseteq A$, there is an associated equivalence relation:

$$\mathrm{IND}(P)=\left\{(x,y)\in U\times U\,\middle|\,\forall a\in P, f_a(x)=f_a(y)\right\}. \quad (1)$$

The partition of U generated by IND(P) is denoted U/P. If $(x,y)\in\mathrm{IND}(P)$, then x and y are indiscernible by attributes from P. The equivalence classes of the P-indiscernability relation are denoted by $[x]_P$. Let $X\subseteq U$, the P-lower approximation $\underline{P}X$ and P-upper approximation $\overline{P}X$ of set X can be defined as:

$$\underline{P}X=\{x\in U\,|\,[x]_P\subseteq X\} \quad (2)$$

$$\overline{P}X=\{x\in U\,|\,[x]_P\bigcap X\neq\phi\}. \quad (3)$$

Let P, $Q\subseteq A$ be equivalence relations over U, then the positive, negative and boundary regions can be defined as:

$$\mathrm{POS}_P(Q)=\bigcup_{X\in U/Q}\underline{P}X \quad (4)$$

$$\mathrm{NEG}_P(Q)=U-\bigcup_{X\in U/Q}\overline{P}X \quad (5)$$

$$\mathrm{BND}_P(Q)=\bigcup_{X\in U/Q}\overline{P}X-\bigcup_{X\in U/Q}\underline{P}X. \quad (6)$$

The positive region of the partition U/Q with respect to P, $\mathrm{POS}_P(Q)$, is the set of all objects of U that can be certainly classified to blocks of the partition U/Q by means of P. Q depends on P in a degree k ($0\le k\le 1$) denoted by $P\Rightarrow_k Q$

$$k=\gamma_P(Q)=\frac{\left|\mathrm{POS}_P(Q)\right|}{|U|} \quad (7)$$

where P is a set of condition attributes, Q is the decision, and $\gamma_P(Q)$ is the quality of classification. If k=1, Q depends totally on P; if 0<k<1, Q depends partially on P; and if k=0 then Q does not depend on P. The goal of attribute reduction is to remove redundant attributes so that the reduced set provides the same quality of classification as the original. The set of all reducts is defined as:

$$\mathrm{Red}(C)=\left\{R\subseteq C\,\middle|\,\gamma_R(D)=\gamma_C(D),\forall B\subset R,\ \gamma_B(D)\neq\gamma_C(D)\right\}. \quad (8)$$

A dataset may have many attribute reducts. The set of all optimal reducts is:

$$\mathrm{Red}(C)_{\min}=\left\{R\in\mathrm{Red}\,\middle|\,\forall R'\in\mathrm{Red},|R|\le|R'|\right\}. \quad (9)$$

3. Supervised Feature Selection

The supervised FS methods evaluate various feature subsets using an evaluation function or metric to select only those features which are related to, or lead to, the decision classes of the data under consideration.

In many real-world problems, FS is a necessity due to the abundance of noisy, irrelevant or misleading features. For instance, by removing these factors, learning from data techniques can take place very effectively. A detailed review of feature selection techniques devised for classification tasks can be found in [8] and [9].

The usefulness of a feature or feature subset is determined by both its relevancy and redundancy. A feature is said to be relevant if it is predictive of the decision feature(s); otherwise, it is irrelevant. A feature is considered to be redundant if it is highly correlated with other features. Hence, the search for a good feature subset involves finding those features that are highly correlated with the decision feature(s), but are not correlated with each other.

3.1 Supervised QR Algorithm

The quick reduct (QR) algorithm given in Algorithm 1 attempts to calculate a reduct without exhaustively generating all possible subsets[10]–[12]. 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 algorithm,

the dependency of each attribute is calculated and the best candidate is chosen.

Algorithm 1. The supervised QR algorithm.

QR (C, D)
C, the set of all conditional features;
D, the set of decision features.
(1) $R \leftarrow \{\}$
(2) do
(3) $T \leftarrow R$
(4) $\forall x \in (C - R)$
(5) if $\gamma_{R\cup\{x\}}(D) > \gamma_T(D)$
(6) $T \leftarrow R \bigcup \{x\}$
(7) $R \leftarrow T$
(8) until $\gamma_R(D) = \gamma_C(D)$
(9) return R

3.2 Entropy-Based Reduct (EBR)

The EBR, developed from work carried out in [10], is based on the entropy heuristic employed by machine learning techniques[13]. A similar approach has been adopted in [8] where an entropy measure was used for ranking features. Using this entropy measure, the algorithm used in rough set-based attribute reduction[14] can be modified. This algorithm requires no thresholds and its search for the best feature subset is stopped when the resulting subset entropy is equal to that of the entire feature set. The EBR algorithm is given in Algorithm 2.

Algorithm 2. Entropy-based reduction algorithm

EBR (C, D)
C, the set of all conditional features;
D, the set of decision features.
(1) $R \leftarrow \{\}$
(2) do
(3) $T \leftarrow R$
(4) $\forall x \in (C - R)$
(5) if $H(R \bigcup \{x\}) < H(T)$
(6) $T \leftarrow R \bigcup \{x\}$
(7) $R \leftarrow T$
(8) until $H(D \mid R) = H(D \mid C)$
(9) return R

3.3 Relative Reduct (RR)

In [15], a FS method based on a relative dependency measure was presented. The technique was originally proposed to avoid the calculation of discernability functions or positive regions, which can be computationally expensive without optimizations. The authors replaced the traditional rough set degree of dependency with an alternative measure, the relative dependency. The RR algorithm is given in Algorithm 3.

Algorithm 3. The relative reduct algorithm.

Relative reduct (C, D)
C, the set of all conditional features;
D, the set of decision features;
(1) $R \leftarrow C$
(2) $\forall a \in C$
(3) if $(\kappa_{R-\{a\}}(D) = 1)$
(4) $R \leftarrow R - \{a\}$
(5) return R

4. Unsupervised Feature Selection

In this section, a novel USQR Algorithm is proposed. In data mining applications, decision class labels are often unknown or incomplete. In this situation the unsupervised feature selection plays a vital role to select features.

This section introduces a new positive region based unsupervised subset evaluation measure using RST, and the corresponding reduct algorithm is shown in Algorithm 4. The existing supervised QR algorithm has two input parameters: conditional attribute and decision attribute, and its evaluation of degree of dependency value leads to the decision attribute. But the proposed USQR has only one input parameter which is a conditional attribute. Here the evaluation of degree of dependency value for a feature's subset leads to each conditional attribute and evaluate mean of dependency values for all conditional attributes.

4.1 USQR Algorithm

The USQR 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 algorithm, the mean dependency of each attribute subset is calculated and the best candidate is chosen:

$$\gamma_P(a) = \frac{\left|\mathrm{POS}_P(a)\right|}{\left|U\right|}, \quad \forall a \in A. \tag{10}$$

Table 1: Example dataset

$x \in U$	a	b	c	d
1	1	0	2	1
2	1	0	2	0
3	1	2	0	0
4	1	2	2	1
5	2	1	0	0
6	2	1	1	0
7	2	1	2	1

Algorithm 4. The USQR algorithm.

USQR (C)
C, the set of all conditional features;
(1) $R \leftarrow \{\}$
(2) do
(3) $T \leftarrow R$
(4) $\forall x \in (C-R)$
(5) $\forall y \in C$
(6) $\gamma_{R\cup\{x\}}(y) = \dfrac{|\mathrm{POS}_{R\cup\{x\}}(y)|}{|U|}$
(7) if $\overline{\gamma_{R\cup\{x\}}(y)}, \forall y \in C > \overline{\gamma_T(y)}, \forall y \in C$
(8) $T \leftarrow R\cup\{x\}$
(9) $R \leftarrow T$
(10) until $\overline{\gamma_R(y)}, \forall y \in C = \overline{\gamma_C(y)}, \forall y \in C$
(11) return R

4.2 Worked Example

Now consider the example dataset given in Table 1, {*a*, *b*, *c*, *d*} are conditional attributes. The dependency values of each attribute are calculated in Step 1.

Step 1.

$$\gamma_{\{a\}}(\{a\}) = \frac{\left|\mathrm{POS}_{\{a\}}(\{a\})\right|}{|U|} = \frac{\left|\{1, 2, 3, 4, 5, 6, 7\}\right|}{\left|\{1, 2, 3, 4, 5, 6, 7\}\right|} = \frac{7}{7}$$

$$\gamma_{\{a\}}(\{b\}) = \frac{\left|\mathrm{POS}_{\{a\}}(\{b\})\right|}{|U|} = \frac{\{|5, 6, 7|\}}{\{|1, 2, 3, 4, 5, 6, 7|\}} = \frac{3}{7}$$

$$\gamma_{\{a\}}(\{c\}) = \frac{\left|\mathrm{POS}_{\{a\}}(\{c\})\right|}{|U|} = \frac{|\{\}|}{\{|1, 2, 3, 4, 5, 6, 7|\}} = \frac{0}{7}$$

$$\gamma_{\{a\}}(\{d\}) = \frac{\left|\mathrm{POS}_{\{a\}}(\{d\})\right|}{|U|} = \frac{|\{\}|}{\left|\{1, 2, 3, 4, 5, 6, 7\}\right|} = \frac{0}{7}$$

$$\sum_{\forall y\in C}\gamma_{\{a\}}(\{y\}) = \frac{7}{7}+\frac{3}{7}+\frac{0}{7}+\frac{0}{7} = \frac{10}{7}$$

$$\overline{\gamma_{\{a\}}(\{y\})}, \forall y \in C = \frac{\left(\frac{10}{7}\right)}{4} = 0.35714.$$

Similarly, the other degree of dependency values are calculated and tabulated in Table 2.

Attribute *b* generates the highest dependency degree, hence attribute *b* is chosen to evaluate the indiscernability of sets {*a*, *b*}, {*b*, *c*} and {*b*, *d*} and calculate the degree of dependency as given in Step 2.

Table 2: Dependency values

$y\|x$	{*a*}	{*b*}	{*c*}	{*d*}
a	1.0000	1.0000	0.1429	0.0000
b	0.4286	1.0000	0.1429	0.0000
c	0.0000	0.2857	1.0000	0.4286
d	0.0000	0.0000	0.4286	1.0000
$\overline{\gamma_{\{y\}}(\{y\}), \forall y \in C}$	0.3571	0.5714	0.4285	0.3571

Step 2:

$$\gamma_{\{a,b\}}(\{a\}) = \frac{\left|\mathrm{POS}_{\{a,b\}}(\{a\})\right|}{|U|} = \frac{\left|\{1, 2, 3, 4, 5, 6, 7\}\right|}{\left|\{1, 2, 3, 4, 5, 6, 7\}\right|} = \frac{7}{7}$$

$$\gamma_{\{a,b\}}(\{b\}) = \frac{\left|\mathrm{POS}_{\{a,b\}}(\{b\})\right|}{|U|} = \frac{\left|\{1, 2, 3, 4, 5, 6, 7\}\right|}{\left|\{1, 2, 3, 4, 5, 6, 7\}\right|} = \frac{7}{7}$$

$$\gamma_{\{a,b\}}(\{c\}) = \frac{\left|\mathrm{POS}_{\{a,b\}}(\{c\})\right|}{|U|} = \frac{\left|\{1, 2\}\right|}{\{|1, 2, 3, 4, 5, 6, 7|\}} = \frac{2}{7}$$

$$\gamma_{\{a,b\}}(\{d\}) = \frac{\left|\mathrm{POS}_{\{a,b\}}(\{d\})\right|}{|U|} = \frac{|\{\}|}{\left|\{1, 2, 3, 4, 5, 6, 7\}\right|} = \frac{0}{7}$$

$$\sum_{\forall y\in C}\gamma_{\{a,b\}}(\{y\}) = \frac{7}{7}+\frac{7}{7}+\frac{2}{7}+\frac{0}{7} = \frac{16}{7}$$

$$\overline{\gamma_{\{a,b\}}(\{y\})}, \forall y \in C = \frac{\left(\frac{16}{7}\right)}{4} = 0.57143.$$

Similarly, the other degree of dependency values are calculated and tabulated in Table 3.

Attribute's sets {*b*, *c*} and {*b*, *d*} generate the highest dependency degrees. The algorithm selects the attribute's set {*b*, *c*} (since {*b*, *c*} is evaluated first), and then evaluate the indiscernability of sets {*a*, *b*, *c*} and {*b*, *c*, *d*} and calculate the degree of dependency as given in Step 3.

Step 3:

$$\gamma_{\{a,b,c\}}(\{a\}) = \frac{\left|\mathrm{POS}_{\{a,b,c\}}(\{a\})\right|}{|U|} = \frac{\left|\{1, 2, 3, 4, 5, 6, 7\}\right|}{\left|\{1, 2, 3, 4, 5, 6, 7\}\right|} = \frac{7}{7}$$

$$\gamma_{\{a,b,c\}}(\{b\}) = \frac{\left|\mathrm{POS}_{\{a,b,c\}}(\{b\})\right|}{|U|} = \frac{\left|\{1, 2, 3, 4, 5, 6, 7\}\right|}{\left|\{1, 2, 3, 4, 5, 6, 7\}\right|} = \frac{7}{7}$$

$$\gamma_{\{a,b,c\}}(\{c\}) = \frac{\left|\mathrm{POS}_{\{a,b,c\}}(\{c\})\right|}{|U|} = \frac{\left|\{1, 2\}\right|}{\left|\{1, 2, 3, 4, 5, 6, 7\}\right|} = \frac{2}{7}$$

$$\gamma_{\{a,b,c\}}(\{d\}) = \frac{\left|\mathrm{POS}_{\{a,b,c\}}(\{d\})\right|}{|U|} = \frac{\left|\{3, 4, 5, 6, 7\}\right|}{\left|\{1, 2, 3, 4, 5, 6, 7\}\right|} = \frac{5}{7}$$

$$\sum_{\forall y\in C}\gamma_{\{a,b,c\}}(\{y\}) = \frac{7}{7}+\frac{7}{7}+\frac{7}{7}+\frac{5}{7} = \frac{26}{7}$$

$$\overline{\gamma_{\{a,b,c\}}(\{y\})},\ \forall y \in C = \frac{\left(\frac{26}{7}\right)}{4} = 0.92857.$$

Similarly,

$$\overline{\gamma_{\{b,c,d\}}(\{y\})}, \forall y \in C = 1.$$

The dependency values calculated in Step 3 are tabulated in Table 4.

This process continues until the dependency value is 1. If the dependency value is 1 then the given dataset is consistent; otherwise, the dataset is inconsistent. Since dependency value of $\{b, c, d\}$ is 1, the algorithm terminates after evaluating the subset $\{b, c, d\}$, and return this feature subset. The dataset can now be reduced to these features only. The returned subset is a rough set reduct.

5. Classifications

The classifier tool WEKA[16] is open source java based machine-learning workbench that can be run on any computer in which a java run time environment is installed. It brings together many machine learning algorithm and tools under a common frame work. The WEKA is a well known package of data mining tools which provides a variety of known, well maintained classifying algorithms. This allows us to do experiments with several kinds of classifiers quickly and easily. The tool is used to perform benchmark experiment. Four classifier learners were employed for the classification of the data, DTNB, JRip, J48, and LMT.

Table 3: Dependency values

$y\|x$	$\{a, b\}$	$\{b, c\}$	$\{b, d\}$
a	1.0000	1.0000	1.0000
b	1.0000	1.0000	1.0000
c	0.2857	1.0000	0.7143
d	0.0000	0.7143	1.0000
$\overline{\gamma_{\{P\}}(\{y\})}, \forall y \in C$	0.57143	0.92857	0.92857

Table. 4: Dependency values

$y\|x$	$\{a, b, c\}$	$\{b, c, d\}$
a	1.0000	1.0000
b	1.0000	1.0000
c	1.0000	1.0000
d	0.7143	1.0000
$\overline{\gamma_{\{P\}}(\{y\})}, \forall y \in C$	0.9285	1.0000

6. Experimental Results

This section presents the results of experimental studies using both crisp-valued and real-valued data sets. The USQR method is compared with the QR, EBR, RR methods. All data sets have been obtained from the UCI Repository Machine Learning Database[16]. A comparison of the QR, EBR, RR and USQR methods are made based on the subset, time taken to discover subsets, and classification accuracy. A short experimental evaluation for 9 benchmark datasets is presented. The information of the data sets contains names of dataset, number of objects, number of classes and number of attributes, which are given in Table 5.

6.1 Feature Selection by QR, EBR, RR and USQR

The features are reduced by the supervised QR algorithm, EBR algorithm, RR algorithm and the USQR algorithm. The selected features are tabulated in Table 6.

Table 5: Dataset information

Index	Data set	Instances	Class	Attr_size
1	Iris	150	3	4
2	WBCD	699	2	9
3	Car	1728	4	6
4	AnginaRisk	121	2	11
5	ECOLI	336	8	7
6	Heart_s	270	2	14
7	BUPALiver	345	2	6
8	PimaIn Diabetes	768	2	8
9	Wine	178	3	13

Table 6: Selected features

Data set index	QR	EBR	RR	USQR
1	(1,2,3)	(1,2,3)	(2,3,4)	(1, 2, 3, 4)
2	(1, 2, 6, 7)	(1, 2, 6, 7)	(5, 6, 7, 8, 9)	(1, 2, 3, 4, 5, 6, 7, 8, 9)
3	(1, 2, 3, 4, 5, 6)	(1, 2, 3, 4, 5, 6)	(1, 2, 3, 4, 5, 6)	(1, 2, 3, 4, 5, 6)
4	(2, 3, 6)	(2, 3, 6)	(6, 8, 11)	(2, 3, 6)
5	(1, 2, 6)	(1, 2, 6)	(2, 5, 7)	(1, 2, 6)
6	(1, 4, 5)	(1, 4, 5)	(6, 11, 12, 13)	(1, 4, 5)
7	(1, 2, 5)	(1, 2, 5)	(3, 4, 5)	(2, 3, 5)
8	(1, 2, 7)	(1, 2, 7)	(6, 7, 8)	(1, 2, 7)
9	(1, 7)	(1, 7)	(12, 13)	(1, 2)

Table 7: Time (in seconds) taken to reduct features

Data set index	QR	EBR	RR	USQR
1	0.8086	0.8473	0.8602	8.4466
2	2.1448	2.1383	4.4469	69.5898
3	1.7609	1.7161	1.9830	7.9247
4	2.5544	2.5978	5.6646	28.2733
5	7.6714	7.5769	16.3145	58.7206
6	6.2835	6.2121	23.3529	102.7714
7	5.0202	5.0118	9.3458	58.2995
8	50.7979	50.8006	150.9474	642.7480
9	9.1299	9.2622	59.0704	104.5222

In an attempt to compare the complexity of the SQR, EBR, RR and USQR methods from an application viewpoint, both FS methods were applied to the data sets and the time taken to find a reduct recorded in each case. The time (in seconds) taken to find reduct is tabulated in Table 7. The results show that there is an increase in runtime for the USQR method. Fig. 1 demonstrates the overall differences in runtime among these methods.

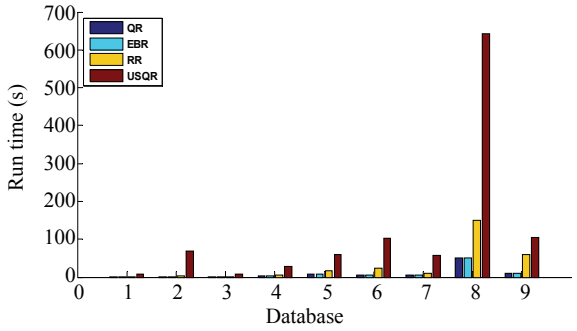

Fig. 1. Time (in seconds) taken to reduct features

6.2 Comparison of Proposed Method with Supervised Methods

In this section, the USQR method is compared with the QR, EBR, and RR methods. The classification was initially performed on the unreduced data set, followed by the reduced data sets which were obtained by using the QR, EBR, RR and USQR methods. Results are presented in terms of classification accuracy and classification mean absolute error. The data presented in Table 8 and Table 9 shows the classification accuracy values and classification mean absolute error values, respectively.

Table 8: Classification accuracy values

Algorithm	Classifiers	Data set								
		1	2	3	4	5	6	7	8	9
Unreduced data	DTNB	92.00	96.85	95.25	92.56	81.25	81.48	57.68	73.82	98.88
	JRip	94.00	95.42	86.45	90.91	81.25	78.89	64.63	76.04	92.13
	J48	96.00	94.56	92.36	95.87	84.22	76.67	68.69	73.82	93.82
	LMT	94.00	95.99	98.78	94.21	87.20	83.33	66.37	77.47	97.19
QR	DTNB	94.00	97.28	95.25	95.04	61.38	61.75	57.68	73.31	92.13
	JRip	92.67	95.70	86.45	92.56	78.27	58.52	62.32	71.88	90.44
	J48	93.33	95.56	92.36	95.04	77.38	58.52	62.90	72.14	94.94
	LMT	93.33	95.28	98.78	92.56	79.76	58.89	60.87	75.00	90.45
EBR	DTNB	94.00	97.28	95.25	95.04	61.38	61.75	57.68	73.31	92.13
	JRip	92.67	95.70	86.45	92.56	78.27	58.52	62.32	71.88	90.44
	J48	93.33	95.56	92.36	95.04	77.38	58.52	62.90	72.14	94.94
	LMT	93.33	95.28	98.78	92.56	79.76	58.89	60.87	75.00	90.45
RR	DTNB	92.67	96.13	95.25	95.04	75.30	81.85	57.68	67.18	95.04
	JRip	95.33	94.13	86.45	95.87	73.80	81.11	64.92	68.61	92.56
	J48	96.00	94.27	92.36	97.52	78.27	82.96	63.76	67.70	91.74
	LMT	95.33	95.42	98.78	95.87	76.48	80.00	64.63	69.92	94.21
USQR	DTNB	92.00	96.85	95.25	95.04	77.38	61.75	57.68	73.31	78.65
	JRip	94.00	95.42	86.45	92.56	78.27	58.52	63.77	71.88	78.65
	J48	96.00	94.56	92.36	95.04	77.38	58.52	66.96	72.14	78.09
	LMT	94.00	95.99	98.78	92.56	79.76	58.89	65.50	75.00	80.34

Table 9: Classification mean absolute error values

Algorithm	Classifiers	Data set								
		1	2	3	4	5	6	7	8	9
Un-reduced data	DTNB	0.07	0.03	0.14	0.12	0.06	0.23	0.47	0.31	0.02
	JRip	0.05	0.06	0.08	0.10	0.06	0.28	0.41	0.34	0.06
	J48	0.03	0.06	0.04	0.05	0.05	0.24	0.36	0.31	0.04
	LMT	0.04	0.05	0.01	0.08	0.05	0.23	0.40	0.31	0.03
QR	DTNB	0.09	0.04	0.14	0.08	0.09	0.46	0.47	0.34	0.09
	JRip	0.04	0.06	0.08	0.10	0.08	0.47	0.45	0.37	0.09
	J48	0.06	0.06	0.04	0.08	0.07	0.47	0.45	0.36	0.06
	LMT	0.06	0.06	0.01	0.13	0.08	0.47	0.45	0.34	0.13
EBR	DTNB	0.09	0.04	0.14	0.08	0.09	0.46	0.47	0.34	0.09
	JRip	0.04	0.06	0.08	0.10	0.08	0.47	0.45	0.37	0.09
	J48	0.06	0.06	0.04	0.08	0.07	0.47	0.45	0.36	0.06
	LMT	0.06	0.06	0.01	0.13	0.08	0.47	0.45	0.34	0.13
RR	DTNB	0.06	0.04	0.14	0.08	0.09	0.27	0.47	0.38	0.11
	JRip	0.04	0.08	0.08	0.06	0.08	0.29	0.43	0.40	0.09
	J48	0.03	0.08	0.04	0.03	0.08	0.26	0.23	0.38	0.08
	LMT	0.04	0.07	0.01	0.07	0.09	0.27	0.44	0.37	0.08
USQR	DTNB	0.07	0.03	0.14	0.08	0.09	0.46	0.47	0.34	0.19
	JRip	0.05	0.06	0.08	0.10	0.08	0.47	0.43	0.37	0.21
	J48	0.03	0.06	0.04	0.08	0.07	0.47	0.43	0.36	0.20
	LMT	0.04	0.05	0.01	0.13	0.08	0.47	0.43	0.34	0.18

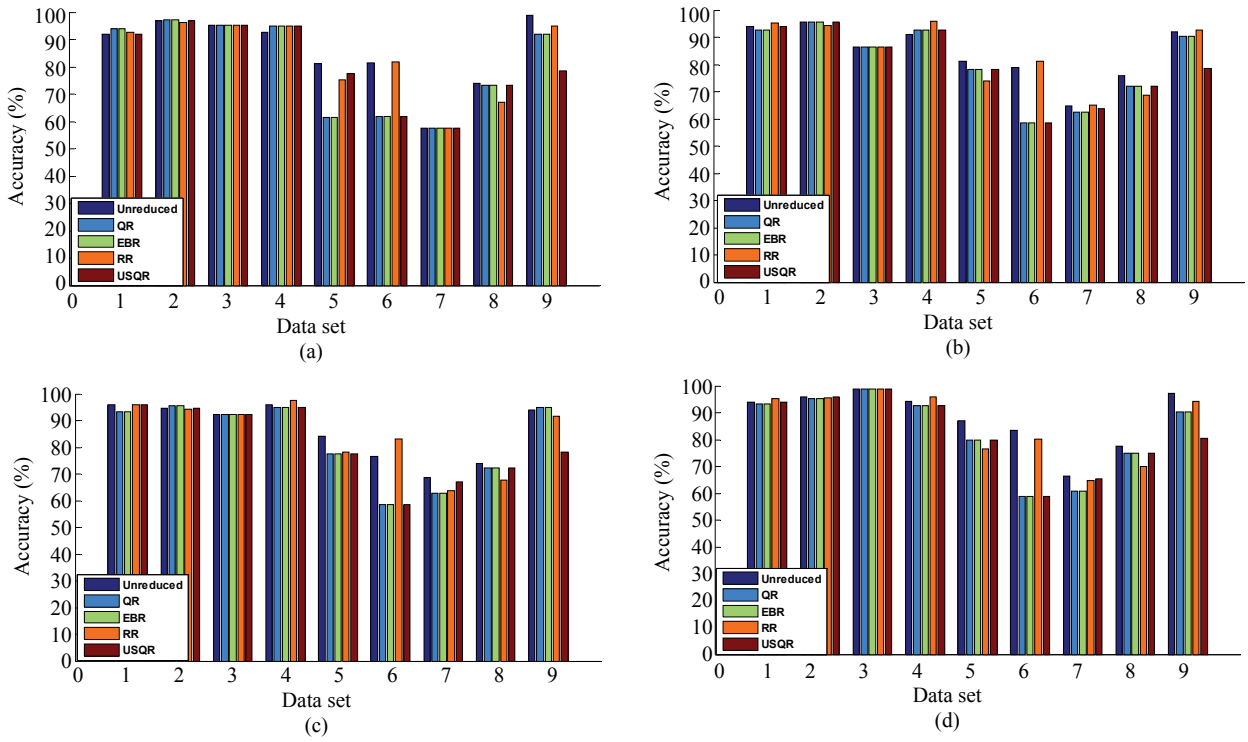

Fig. 2. Classification accuracy values of (a) DTNB classifier, (b) JRip classifier, (c) J48 classifier, and (d) LMT classifier.

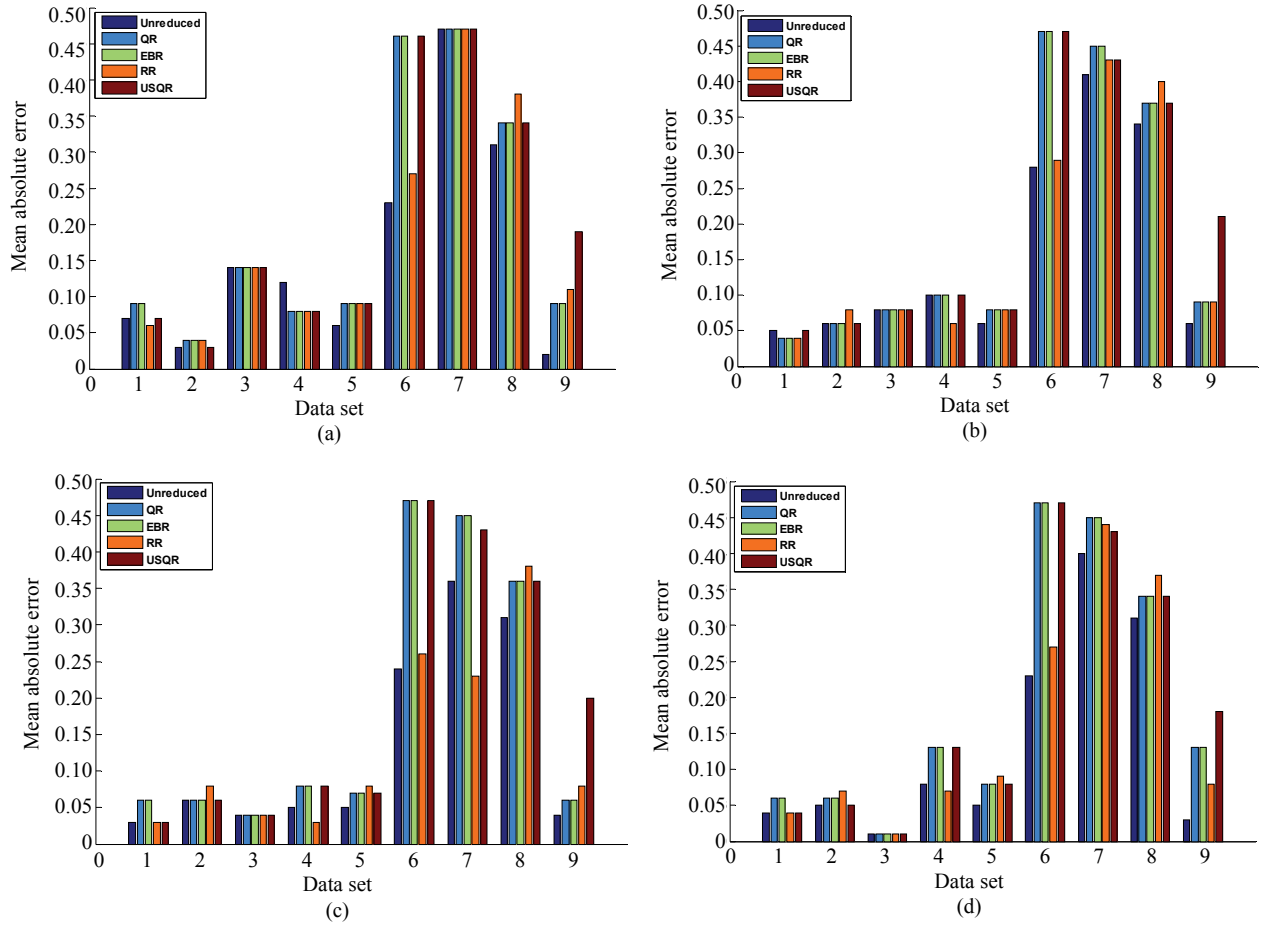

Fig. 3. Classification mean absolute error values of (a) DTNB classifier, (b) JRip classifier (c) J48 classifier, and (d) LMT classifier.

It is interesting to note that an increase in classification accuracy is recorded for the proposed and the QR, EBR, RR methods, with respect to the unreduced data in some cases (dataset index 1, 2, 4 in DTNB and JRip classier, 2 and 9 in J48 classifier). This increase in classification accuracy is a little bit high when comparing the QR, EBR, RR and the USQR methods to the unreduced data. Also, when comparing classification results, the USQR and QR, EBR, RR methods have the same classification accuracy values which are recorded in most of the cases. They are demonstrated in Fig. 2.

It should also be noted that a decrease in classification mean absolute error is recorded for the USQR and the QR, EBR, RR methods in some cases, with respect to the unreduced data. Also, when comparing classification results, the USQR and QR, EBR, RR methods have the same classification mean absolute error values which are recorded in most of the cases, as demonstrated in Fig. 3.

7. Conclusions

In this work, a new USQR algorithm using rough set theory is proposed. The method 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. The WEKA tool is used to classify the data and the classification performance is evaluated by using classification accuracy and mean absolute error. This method is compared with an existing supervised method and it demonstrates that it can effectively remove redundant features. The subsets returned by this unsupervised method are of similar size to that of the supervised method and classification of the reduced data shows that the method selects useful features which are of comparable quality. In future, the same approach can be extended to mammogram image datasets for breast cancer diagnosis.

Acknowledgement

The first author wholeheartedly thanks the service rendered by the UGC, SERO, Hyderabad to carry out this research under FDP during XI plan period.

The second author immensely thanks the UGC, New Delhi for its financial assistance under major research project grant No. F-34-105/2008.

References

[1] J. W. Grzymala-Busse, "LERS-a system for learning from examples based on rough sets," in *Intelligent Decision Support*, R. Slowinski, Ed. 1992, Dordrecht: Kluwer Academic Publishers, pp. 3–18.

[2] M. Dash and H. Liu, "Unsupervised feature selection," in *Proc. of the Pacific and Asia Conf. on Knowledge Discovery and Data Mining*, Kyoto, 2000, pp. 110–121.

[3] S. K. Pal, R. K. De, and J. Basak. "Unsupervised feature evaluation: a neuro-fuzzy approach," *IEEE Trans. on Neural Networks*, vol. 11, no. 2, pp. 366–376, 2000.

[4] S. K. Das, "Feature selection with a linear dependence measure," *IEEE Trans. on Computers*, vol. 20, no. 9, pp. 1106–1109, 1971.

[5] M. A. Hall, "Correlation-based feature selection machine learning," Ph.D. Dissertation, Department of Computer Science, University of Waikato, Hamilton, New Zealand, 1998.

[6] P. Mitra, C. A. Murthy, and S. K. Pal, "Unsupervised feature selection using feature similarity," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 24, no. 4, pp. 1–13, 2002.

[7] Z. Pawlak, *Rough Sets: Theoretical Aspects of Reasoning about Data*, Dordrecht: Kluwer Academic Publishers, 1991.

[8] M. Dashand H. Liu, "Feature selection for classification," *Intelligent Data Analysis*, vol. 1, no. 3, pp. 131–156, 1997.

[9] K. Thankavel and A. Pethalakshmi, "Dimensionality reduction based on rough set theory: a review," *Applied Soft Computing*, vol. 9, no. 1, pp. 1–12, 2009

[10] R. Jensen and Q. Shen, "Semantics-preserving dimensionality reduction: rough and fuzzy-rough based approaches," *IEEE Trans. on Knowledge and Data Engineering*, vol. 16, no. 12, pp. 1457–1471, 2004.

[11] R. Jensen and Q. Shen, "Fuzzy-rough attribute reduction with application to web categorization," *Fuzzy Sets and Systems* vol. 141, no. 3, pp. 469–485, 2004.

[12] R. Jensen, "Combining rough and fuzzy sets for feature selection," Ph.D. Dissertation, School of Informatics, University of Edinburgh, Edinburgh, 2004.

[13] J. R. Quinlan, *C4.5: Programs for Machine Learning. The Morgan Kaufmann Series in Machine Learning*, San Mateo: Morgan Kaufmann Publishers, 1993.

[14] Q. Shen and A. Chouchoulas, "A rough-fuzzy approach for generating classification rules," *Pattern Recognition*, vol. 35, no. 11, pp. 2425–2438, 2002.

[15] J. Han, X. Hu, and T.-Y. Lin, "Feature subset selection based on relative dependency between attributes," in *Proc. of the 4th International Conf. on Rough Sets and Current Trends in Computing*, Uppsala, 2004, pp. 176–185.

[16] C. L. Blake and C. J. Merz, UCI Repository of machine learning databases. [Online]. Available: http://www.ics.uci.edu/~mlearn/.

C. Velayutham was born in 1965 at Thanjavur, Tamilnadu, India. He received the Master of Science in applied mathematics in 1989, and Post Graduate Diploma in computer application in 1990, from Bharathidasan University, Trichy, India. He obtained his M.Phil (computer science) degree from Manonmaniam Sundaranar University, Trunelveli, India in 2002. Currently, he is an associate professor with the Department of Computer Science, Aditanar College of Arts and Science, Tiruchendur, Tamil Nadu, India. His research interests include medical image processing, data mining, neural network, fuzzy logic, and rough set.

K. Thangavel was born in 1964 at Namakkal, Tamilnadu, India. He received his M.S. degree from the Department of Mathematics, Bharathidasan University in 1986, and Master of Computer Applications Degree from Madurai Kamaraj University, India in 2001. He obtained his Ph.D. degree from the Department of Mathematics, Gandhigram Rural Institute-Deemed University, Gandhigram, India in 1999. Currently, he is a professor and the Head of the Department of Computer Science, Periyar University, Salem. He was a recipient of Tamilnadu Scientist award for the year 2009. His research interests include medical image processing, artificial intelligence, neural network, fuzzy logic, data mining, and rough set.