
PERGAMON

Available at
www.ElsevierComputerScience.com
POWERED BY SCIENCE DIRECT®

Pattern Recognition 37 (2004) 1351–1363

PATTERN RECOGNITION
THE JOURNAL OF THE PATTERN RECOGNITION SOCIETY
www.elsevier.com/locate/patcog

Selecting informative features with fuzzy-rough sets and its application for complex systems monitoring

Qiang Shen*, Richard Jensen

School of Informatics, The University of Edinburgh, Centre for Intelligent Systems and their Applications, Appleton Tower, Crichton Street, Edinburgh EH8 9LE, UK

Received 30 January 2003; accepted 2 October 2003

Abstract

One of the main obstacles facing current intelligent pattern recognition applications is that of dataset dimensionality. To enable these systems to be effective, a redundancy-removing step is usually carried out beforehand. Rough set theory (RST) has been used as such a dataset pre-processor with much success, however it is reliant upon a *crisp* dataset; important information may be lost as a result of quantisation of the underlying numerical features. This paper proposes a feature selection technique that employs a hybrid variant of rough sets, *fuzzy-rough* sets, to avoid this information loss. The current work retains dataset semantics, allowing for the creation of clear, readable fuzzy models. Experimental results, of applying the present work to complex systems monitoring, show that fuzzy-rough selection is more powerful than conventional entropy-, PCA- and random-based methods.

Keywords: Feature selection; Feature dependency; Fuzzy-rough sets; Reduct search; Rule induction; Systems monitoring

1. Introduction

The ever-increasing demand for dependable, trustworthy intelligent diagnostic and monitoring systems, as well as knowledge-based systems in general, has focused much of the attention of researchers on the knowledge-acquisition bottleneck. The task of gathering information and extracting general knowledge from it is known to be the most difficult part of creating a knowledge-based system. Complex application problems, such as reliable monitoring and diagnosis of industrial plants, are likely to present large numbers of features, many of which will be redundant for the task at hand [1,2]. Additionally, inaccurate and/or uncertain values cannot be ruled out. Such applications typically require convincing explanations about the inference performed, therefore a method to allow automated generation of knowledge models of clear semantics is highly desirable.

The most common approach to developing expressive and human readable representations of knowledge is the use of if–then production rules [3]. Yet, real-life problem domains usually lack generic and systematic expert rules for mapping feature patterns onto their underlying classes. The present work aims to induce low-dimensionality rule sets from historical descriptions of domain features which are often of high dimensionality. In particular, a recent fuzzy rule induction algorithm (RIA), as first reported in Ref. [4], is taken to act as the starting point for this. It should be noted, however, that the flexibility of the system discussed here allows the incorporation of almost any rule induction algorithm that uses descriptive set representation of features. The choice of the current RIA is largely due to its recency and the simplicity in implementation. Provided with sets of continuous feature values, the RIA is able to induce classification rules to partition the feature patterns into underlying categories.

* Corresponding author. Tel.: +44-1316502705; fax: +44-131-6506513.

E-mail addresses: qiangs@inF.ed.ac.uk (Q. Shen), richjens@dai.ed.ac.uk (R. Jensen).

0031-3203/$30.00
doi:10.1016/j.patcog.2003.10.016

In order to speed up the RIA and reduce rule complexity, a preprocessing step is required. This is particularly important for tasks where learned rulesets need regular updating to reflect the changes in the description of domain features. This step reduces the dimensionality of potentially very large feature sets while minimising the loss of information needed for rule induction. It has an advantageous side-effect in that it removes redundancy from the historical data. This also helps simplify the design and implementation of the actual pattern classifier itself, by determining what features should be made available to the system. In addition, the reduced input dimensionality increases the processing speed of the classifier, leading to better response times. Most significant, however, is the fact that fuzzy-rough feature selection (FRFS) preserves the semantics of the surviving features after removing any redundant ones. This is essential in satisfying the requirement of user readability of the generated knowledge model, as well as ensuring the understandability of the pattern classification process.

There exists a number of approaches relevant to the rule induction task at hand, both from the point of view of applications and that of computational methods. For example, the *fuzzy automatic pattern analysis and classification system* (FAPACS) algorithm documented in Refs. [5,6] is able to discover fuzzy association rules in relational databases. It works by locating pairs of features that satisfy an 'interestingness' measure that is defined in terms of an adjusted difference between the observed and expected values of relations. This algorithm is capable of expressing linguistically both the regularities and the exceptions discovered within the data. Modifications to the fuzzy ID3 (itself an augmentation of Quinlan's original ID3 [7]) rule induction algorithm have been documented [8] to better support fuzzy learning. In a similar attempt, Janikow [9] has proposed modifications to decision trees to combine traditional symbolic decision trees with approximate reasoning, offered by fuzzy representation. This approach redefines the methodology for knowledge inference, resulting in a method best suited to relatively stationary problems.

A common disadvantage of these techniques is their sensitivity to high dimensionality. This may be remedied using conventional work such as principal components analysis (PCA) [10,11]. Unfortunately, although efficient, PCA irreversibly destroys the underlying semantics of the feature set. Further reasoning about the derivation from transformed principal features is almost always humanly impossible. Most semantics-preserving dimensionality reduction (or feature selection) approaches tend to be domain specific, however, relying on the use of well-known features of the particular application domains.

Over the past 10 years, rough set theory (RST [12]) has become a topic of great interest to researchers and has been applied to many domains. Given a dataset with discretised feature values, it is possible to find a subset (termed a *reduct*) of the original features using RST that are the most informative; all other features can be removed from the dataset with minimal information loss. RST offers an alternative approach that preserves the underlying semantics of the data while allowing reasonable generality. It is, therefore, desirable to develop this technique to provide the means of data reduction for crisp and real-valued datasets which utilises the extent to which values are similar. Indeed, this can be achieved through the use of *fuzzy-rough* sets.

Fuzzy-rough sets encapsulate the related but distinct concepts of vagueness (for fuzzy sets [13]) and indiscernibility (for rough sets), both of which occur as a result of uncertainty in knowledge [14]. This paper, based on the most recent work as reported in Refs. [15,16], presents such a method which employs fuzzy-rough sets to improve the handling of this uncertainty. The theoretical domain independence of the approach allows it to be used with different rule induction algorithms, in addition to the specific RIA adopted herein. In light of this, the present work is developed in a highly modular manner. Note that the approach given in Ref. [17] forms a kin to this work. However, unlike the present research, it only reports on the result of a direct combination of crisp RST (not fuzzy-rough set theory) and the fuzzy learning algorithm proposed in Ref. [18] that is rather sensitive to the training data in ensuring the coverage of learned rules.

The rest of this paper is structured as follows. Section 2 first summarises the theoretical background of the basic ideas of RST that are relevant to this work. Then, it describes the proposed fuzzy-rough set feature selection method. To put the development in the context of rule induction, the RIA algorithm adopted is outlined. Important design and implementation issues involved are addressed throughout the discussion. To illustrate the operation of both FRFS and the RIA, worked examples are included. A real problem case of complex system monitoring is detailed in Section 3, along with the modular design of the software system built for testing the approach. Section 4 shows the results of applying the present work to the problem case, supported by comparisons to the applications of entropy-based [7], PCA-based and random selection to the same domain. Section 5 concludes the paper, and proposes further work in this area.

2. Fuzzy-rough feature selection

This section details the theoretical work involved in this paper, including the relevant ideas of RST and a crisp feature selection method directly using these ideas, the description of the present work on fuzzy-rough set-based feature selection, and the introduction of the RIA algorithm for fuzzy rule induction from data.

2.1. Relevant ideas of RST

The theory of rough sets provides rigorous mathematical techniques for creating approximate descriptions of objects

for data analysis, optimisation and recognition. A rough set itself is an approximation of a vague concept by a pair of precise concepts, called lower and upper approximations [12]. 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.

2.1.1. Basic concepts

Rough sets have been employed to remove redundant conditional features from discrete-valued datasets, while retaining their information content. A successful example of this is the rough set feature selection (RSFS) method [17]. Central to RSFS is the concept of indiscernibility. Let $I=(\mathbb{U},A)$ be an information system, where $\mathbb{U}$ is a non-empty set of finite objects (the universe of discourse); A is a non-empty finite set of features such that $a:\mathbb{U}\rightarrow V_a\ \forall a\in A$, V_a being the value set of feature a. In a decision system, $A=\{C\cup D\}$ where C is the set of conditional features and D is the set of decision features. With any $P\subseteq A$ there is an associated equivalence relation $IND(P)$:

$$IND(P)=\{(x,y)\in\mathbb{U}^2\mid\forall a\in P,\ a(x)=a(y)\}. \tag{1}$$

The partition of $\mathbb{U}$, generated by $IND(P)$ is denoted $\mathbb{U}/P$ and can be calculated as follows:

$$\mathbb{U}/P=\otimes\{a\in P:\ \mathbb{U}/IND(\{a\})\}, \tag{2}$$

where

$$A\otimes B=\{X\cap Y:\ \forall X\in A,\forall Y\in B,X\cap Y\neq\emptyset\}. \tag{3}$$

If $(x,y)\in IND(P)$, then x and y are indiscernible by features from P. The equivalence classes of the P-indiscernibility relation are denoted $[x]_P$. Let $X\subseteq\mathbb{U}$, the P-*lower* approximation of a set can now be defined as

$$\underline{P}X=\{x\mid[x]_P\subseteq X\}. \tag{4}$$

Let P and Q be equivalence relations over $\mathbb{U}$, then the positive region is defined as

$$POS_P(Q)=\bigcup_{X\in\mathbb{U}/Q}\underline{P}X. \tag{5}$$

In terms of feature pattern-based classification, the positive region contains all objects of $\mathbb{U}$ that can be classified to classes of $\mathbb{U}/Q$ using the knowledge in features P.

2.1.2. Feature dependency and significance

An important issue concerned here, as with many data analysis tasks, is discovering dependencies between features. Intuitively, a set of features Q depends totally on a set of features P, denoted $P\Rightarrow Q$, if all feature values from Q are uniquely determined by values of features from P. Dependency can be defined in different ways (e.g. via conditional probabilities and information gains). In RST, it is typically defined in the following way [12,17]:

For $P,Q\subseteq A$, Q depends on P in a degree k $(0\leqslant k\leqslant 1)$, denoted $P\Rightarrow_k Q$, if

$$k=\gamma_P(Q)=\frac{|POS_P(Q)|}{|\mathbb{U}|}, \tag{6}$$

where $|S|$ stands for the cardinality of set S.

If $k=1$ Q depends totally on P, if $0<k<1$ Q depends partially (in a degree k) on P, and if $k=0$ Q does not depend on P.

By calculating the change in dependency when a feature is removed from the set of considered conditional features, a measure of the significance of the feature can be obtained. The higher the change in dependency, the more significant the feature is. If the significance is 0, then the feature is dispensible. More formally, given P,Q and a feature $x\in P$, the significance of feature x upon Q is defined by

$$\sigma_P(Q,x)=\gamma_P(Q)-\gamma_{P-\{x\}}(Q). \tag{7}$$

2.1.3. Feature reducts and reduct search

The reduction of features is achieved by comparing equivalence relations generated by sets of features. Features are removed from a given set so that the reduced set provides the same quality of classification as the original. In the context of decision systems, a *reduct* is formally defined as a subset R of the conditional feature set C such that $\gamma_R(D)=\gamma_C(D)$. A given dataset may have many feature reduct sets, and the collection of all reducts is denoted by

$$R=\{X:X\subseteq C,\ \gamma_X(D)=\gamma_C(D)\}. \tag{8}$$

The intersection of all the sets in R is called the *core*, the elements of which are those features that cannot be eliminated without introducing contradictions to the dataset.

In RSFS, a reduct with minimum cardinality is searched for; in other words an attempt is made to locate a single element of the minimal reduct set $R_{\min}\subseteq R$:

$$R_{\min}=\{X:X\in R,\ \forall Y\in R,|X|\leqslant|Y|\}. \tag{9}$$

A basic way of achieving this is to calculate the dependencies of all possible subsets of C. Any subset X with $\gamma_X(D)=1$ is a reduct; the smallest subset with this property is a minimal reduct. However, for large datasets this method is impractical and an alternative strategy is required.

The QUICKREDUCT algorithm given in Fig. 1, borrowed from Refs. [15,17], attempts to calculate a minimal reduct without exhaustively generating all possible subsets. It starts off with an empty set and adds in turn, one at a time, those features that will result in the greatest increase in $\gamma_P(Q)$, until this produces its maximum possible value for the dataset (usually 1). However, it has been proved that this method does not always generate a *minimal* reduct, as $\gamma_P(Q)$ is not a perfect heuristic [19]. It does result in a close-to-minimal reduct, though, which is still useful in greatly reducing dataset dimensionality. Note that an intuitive understanding of QUICKREDUCT implies that, for a dimensionality of n, $(n^2+n)/2$ evaluations of the dependency function may be

QUICKREDUCT(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) $T \leftarrow R \cup \{x\}$
(7) $R \leftarrow T$
(8) **until** $\gamma_R(D) = \gamma_C(D)$
(9) **return** R

Fig. 1. The QUICKREDUCT algorithm.

performed for the worst-case dataset. In fact, as feature selection can only take place when $n \geqslant 2$, the base case is $n=2$. Suppose that the set of conditional features in this case is $\{a_1, a_2\}$, the QUICKREDUCT algorithm makes two initial dependency evaluations (for a_1 and a_2) and a final evaluation for $\{a_1, a_2\}$ (in the worst case). Hence, the order of complexity of the algorithm is 3 (or $(n^2 + n)/2$) for $n = 2$.

Suppose that for $n = k$ the order of complexity of the algorithm is

$$\frac{(k^2 + k)}{2} \tag{10}$$

For $k + 1$ features, $\{a_1, \ldots, a_k, a_{k+1}\}$, QUICKREDUCT makes $k + 1$ initial evaluations of the dependency function to determine the best feature (call this a_i). Once a_i is chosen, for the remaining features there are $(k^2 + k)/2$ more evaluations in the worst case according to Eq. (10). Hence, the total number of evaluations for $n = k + 1$ is

$$\frac{k^2 + k}{2} + (k + 1) = \frac{k^2 + 3k + 2}{2} = \frac{(k + 1)^2 + (k + 1)}{2}.$$

As has been shown in Ref. [15], important information is lost due to the discretisation process required for RSFS. Additionally, there is no way of handling noisy data. As an initial approach to addressing these issues, an attempt has been made to combine rough and fuzzy methods for fuzzy rule induction [17]. Although the method claims to be fuzzy-rough, there is no real hybridisation of the two theories. Instead, crisp rough sets are used for dimensionality reduction (after data discretisation has been performed) followed by fuzzy rule induction. The new approach proposed here uses the fuzzy sets employed later in the rule induction phase to guide the reduct search; it uses hybrid fuzzy-rough sets rather than crisp rough sets to compute the dependency degree.

2.2. The proposed method

The RSFS process described previously can only operate effectively with datasets containing discrete values. As most datasets contain real-valued features, it is necessary to perform a discretisation step beforehand. This is typically implemented by standard fuzzification techniques [17]. However, membership degrees of feature values to fuzzy sets are not exploited in the process of dimensionality reduction. By using *fuzzy-rough* sets [14,20,26], it is possible to use this information to better guide feature selection.

2.2.1. Fuzzy equivalence classes

In the same way that crisp equivalence classes are central to rough sets, *fuzzy* equivalence classes are central to the fuzzy-rough set approach [14]. For typical RSFS applications, this means that the decision values and the conditional values may all be fuzzy. The concept of crisp equivalence classes can be extended by the inclusion of a fuzzy similarity relation S on the universe, which determines the extent to which two elements are similar in S. The usual properties of reflexivity ($\mu_S(x,x) = 1$), symmetry ($\mu_S(x, y) = \mu_S(y,x)$) and transitivity ($\mu_S(x,z) \geqslant \mu_S(x, y) \wedge \mu_S(y,z)$) hold.

Using the fuzzy similarity relation, the fuzzy equivalence class $[x]_S$ for objects close to x can be defined:

$$\mu_{[x]_S}(y) = \mu_S(x, y). \tag{11}$$

The following axioms should hold for a fuzzy equivalence class F [21]:

- $\exists x,\ \mu_F(x) = 1$,
- $\mu_F(x) \wedge \mu_S(x, y) \leqslant \mu_F(y)$,
- $\mu_F(x) \wedge \mu_F(y) \leqslant \mu_S(x, y)$.

The first axiom corresponds to the requirement that an equivalence class is non-empty. The second axiom states that elements in y's neighbourhood are in the equivalence class of y. The final axiom states that any two elements in F are related via S. Obviously, this definition degenerates to the normal definition of equivalence classes when S is non-fuzzy.

The family of normal fuzzy sets produced by a fuzzy partitioning of the universe of discourse can play the role of fuzzy equivalence classes [14]. Consider the crisp partitioning $\mathbb{U}/Q = \{\{1,3,6\},\{2,4,5\}\}$. This contains two equivalence classes ({1,3,6} and {2,4,5}) that can be thought of as degenerated fuzzy sets, with those elements belonging to the class possessing a membership of one, zero otherwise. For the first class, for instance, objects 2, 4 and 5 have a membership of zero. Extending this to the case of fuzzy equivalence classes is straightforward: objects can be allowed to assume membership values, with respect to any given class, in the interval [0,1]. $\mathbb{U}/Q$ is not restricted to crisp partitions only; fuzzy partitions are equally acceptable.

2.2.2. Fuzzy lower and upper approximations

From the literature, the fuzzy P-lower and P-upper approximations are defined as [14]:

$$\mu_{\underline{P}X}(F_i) = \inf_x \max\{1 - \mu_{F_i}(x), \mu_X(x)\} \quad \forall i, \tag{12}$$

$$\mu_{\bar{P}X}(F_i) = \sup_x \min\{\mu_{F_i}(x), \mu_X(x)\} \quad \forall i, \tag{13}$$

where F_i denotes a fuzzy equivalence class belonging to $\mathbb{U}/P$. Note that although the universe of discourse in feature reduction is finite, this is not the case in general, hence the use of sup and inf. These definitions diverge a little from the crisp upper and lower approximations, as the memberships of individual objects to the approximations are not explicitly available. As a result of this, the fuzzy lower and upper approximations are herein redefined as

$$\mu_{\underline{P}X}(x) = \sup_{F\in\mathbb{U}/P} \min\big(\mu_F(x), \times \inf_{y\in\mathbb{U}} \max\{1-\mu_F(y), \mu_X(y)\}\big), \tag{14}$$

$$\mu_{\bar{P}X}(x) = \sup_{F\in\mathbb{U}/P} \min\left(\mu_F(x), \sup_{y\in\mathbb{U}} \min\{\mu_F(y), \mu_X(y)\}\right). \tag{15}$$

In implementation, not all $y\in\mathbb{U}$ are needed to be considered—only those where $\mu_F(y)$ is non-zero, i.e. where object y is a fuzzy member of (fuzzy) equivalence class F. The tuple $\langle \underline{P}X, \bar{P}X\rangle$ is called a fuzzy-rough set. It can be seen that these definitions degenerate to traditional rough sets when all equivalence classes are crisp. It is useful to think of the crisp lower approximation as characterised by the following membership function:

$$\mu_{\underline{P}X}(x) = \begin{cases} 1, & x\in F,\ F\subseteq X, \\ 0, & \text{otherwise.} \end{cases} \tag{16}$$

This states that an object x belongs to the P-lower approximation of X if it belongs to an equivalence class that is a subset of X. Obviously, the behaviour of the fuzzy lower approximation must be exactly that of the crisp definition for crisp situations. This is indeed the case as the fuzzy lower approximation may be rewritten as

$$\mu_{\underline{P}X}(x) = \sup_{F\in\mathbb{U}/P} \min\left(\mu_F(x), \inf_{y\in\mathbb{U}} \{\mu_F(y) \to \mu_X(y)\}\right), \tag{17}$$

where "$\to$" stands for fuzzy implication (using the conventional min–max interpretation). In the crisp case, $\mu_F(x)$ and $\mu_X(x)$ will take values from $\{0,1\}$. Hence, it is clear that the only time $\mu_{\underline{P}X}(x)$ will be zero is when at least one object in its equivalence class F fully belongs to F but not to X. This is exactly the same as the definition for the crisp lower approximation. Similarly, the definition for the P-upper approximation can be established.

2.2.3. Fuzzy-rough reduction process

FRFS builds on the notion of the fuzzy lower approximation to enable reduction of datasets containing real-valued features. As will be shown, the process becomes identical to the crisp approach when dealing with nominal well-defined features.

The crisp positive region in traditional rough set theory is defined as the union of the lower approximations. By the extension principle, the membership of an object $x\in\mathbb{U}$, belonging to the fuzzy positive region can be defined by

$$\mu_{POS_P(Q)}(x) = \sup_{X\in\mathbb{U}/Q} \mu_{\underline{P}X}(x). \tag{18}$$

Object x will not belong to the positive region only if the equivalence class it belongs to is not a constituent of the positive region. This is equivalent to the crisp version where objects belong to the positive region only if their underlying equivalence class does so.

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

$$\gamma'_P(Q) = \frac{|\mu_{POS_P(Q)}(x)|}{|\mathbb{U}|} = \frac{\sum_{x\in\mathbb{U}} \mu_{POS_P(Q)}(x)}{|\mathbb{U}|}. \tag{19}$$

As with crisp rough sets, the dependency of Q on P is the proportion of objects that are discernible out of the entire dataset. In the present approach, this corresponds to determining the fuzzy cardinality of $\mu_{POS_P(Q)}(x)$ divided by the total number of objects in the universe.

The definition of dependency degree covers the crisp case as its specific instance. This can be easily shown by recalling the definition of the crisp dependency degree given in Eq. (6). If a function $\mu_{POS_P(Q)}(x)$ is defined which returns 1 if the object x belongs to the positive region, 0 otherwise, then the above definition may be rewritten as

$$\gamma_P(Q) = \frac{\sum_{x\in\mathbb{U}} \mu_{POS_P(Q)}(x)}{|\mathbb{U}|} \tag{20}$$

which is identical to Eq. (19).

If the fuzzy-rough reduction process is to be useful, it must be able to deal with multiple features, finding the dependency between various subsets of the original feature set. For example, it may be necessary to be able to determine the degree of dependency of the decision feature(s) with respect to $P=\{a,b\}$. In the crisp case, $\mathbb{U}/P$ contains sets of objects grouped together that are indiscernible according to both features a and b. In the fuzzy case, objects may belong to many equivalence classes, so the cartesian product of $\mathbb{U}/IND(\{a\})$ and $\mathbb{U}/IND(\{b\})$ must be considered in determining $\mathbb{U}/P$. In general,

$$\mathbb{U}/P = \otimes\{a\in P\colon \mathbb{U}/IND(\{a\})\}. \tag{21}$$

Each set in $\mathbb{U}/P$ denotes an equivalence class. For example, if $P=\{a,b\}$, $\mathbb{U}/IND(\{a\}) = \{N_a, Z_a\}$ and $\mathbb{U}/IND(\{b\}) = \{N_b, Z_b\}$, then

$$\mathbb{U}/P = \{N_a\cap N_b, N_a\cap Z_b, Z_a\cap N_b, Z_a\cap Z_b\}.$$

The extent to which an object belongs to such an equivalence class is therefore calculated by using the conjunction of constituent fuzzy equivalence classes, say $F_i,\ i=1,2,\ldots,n$: .2

$$\mu_{F_1\cap\cdots\cap F_n}(x) = \min(\mu_{F_1}(x), \mu_{F_2}(x), \ldots, \mu_{F_n}(x)). \tag{22}$$

A problem may arise when this approach is compared to the crisp approach. In conventional RSFS, a reduct is defined as

FRQUICKREDUCT(C,D).
C, the set of all conditional features;
D, the set of decision features.

(1) $R \leftarrow \{\}$, $\gamma'_{best} \leftarrow 0$, $\gamma'_{prev} \leftarrow 0$
(2) **do**
(3) $T \leftarrow R$
(4) $\gamma'_{prev} \leftarrow \gamma'_{best}$
(5) $\forall x \in (C - R)$
(6) **if** $\gamma'_{R\cup\{x\}}(D) > \gamma'_T(D)$
(7) $T \leftarrow R \cup \{x\}$
(8) $\gamma'_{best} \leftarrow \gamma'_T(D)$
(9) $R \leftarrow T$
(10) **until** $\gamma'_{best} = \gamma'_{prev}$
(11) **return** R

Fig. 2. The fuzzy-rough QUICKREDUCT algorithm.

subset R of the features which have the same information content as the full feature set A. In terms of the dependency function this means that the values $\gamma(R)$ and $\gamma(A)$ are identical and equal to 1 if the dataset is consistent. However, in the fuzzy-rough approach this is not necessarily the case as the uncertainty encountered when objects belong to many fuzzy equivalence classes results in a reduced total dependency.

A possible way of combatting this would be to determine the degree of dependency of a set of decision features D upon the full feature set and use this as the denominator rather than $|\mathbb{U}|$ (for normalisation), allowing γ' to reach 1. With these issues in mind, a new QUICKREDUCT algorithm has been developed as given in Fig. 2. It employs the new dependency function γ' to choose which features to add to the current reduct candidate in the same way as the original QUICKREDUCT process. The algorithm terminates when the addition of any remaining feature does not increase the dependency (such a criterion could be used with the original QUICKREDUCT algorithm). As with the original algorithm, for a dimensionality of n, the worst case dataset will result in $(n^2+n)/2$ evaluations of the dependency function. However, as FRFS is used for dimensionality reduction prior to any involvement of the system which will employ those features belonging to the resultant reduct, this operation has no negative impact upon the run-time efficiency of the system.

Note that it is also possible to reverse the search process; that is, start with the full set of features and incrementally remove the least informative features. This process continues until no more features can be removed without reducing the total number of discernible objects in the dataset.

2.2.4. A worked example

Using the fuzzy-rough QUICKREDUCT algorithm, Table 1 can be reduced in size. First of all the lower approximations need to be determined. Consider the first feature in the dataset; setting $P = \{A\}$ produces the fuzzy partitioning $\mathbb{U}/P = \{A1, A2, A3\}$. Additionally, setting $Q = \{Plan\}$ produces the fuzzy partitioning $\mathbb{U}/Q = \{X, Y, Z\}$. To determine the fuzzy P-lower approximation of Plan X ($\mu_{\underline{P}X}(x)$), each $F \in \mathbb{U}/P$ must be considered. For $F = A1$:

$$\min\left(\mu_{A1}(x), \inf_{y\in\mathbb{U}} \max\{1-\mu_{A1}(y),\ \mu_X(y)\}\right) = \min(\mu_{A1}(x), 0.6).$$

Similarly, for $F = A2$, $\min(\mu_{A2}(x), 0.3)$ and $F = A3$, $\min(\mu_{A3}(x), 0.0)$. To calculate the extent to which an object x in the dataset belongs to the fuzzy P-lower approximation of X, the union of these values is calculated. For example, object 0 belongs to $\underline{P}X$ with a membership of

$$\sup\{\min(\mu_{A1}(0), 0.6),\ \min(\mu_{A2}(0), 0.3),\ \min(\mu_{A3}(0), 0.0)\} = 0.3.$$

Likewise, for Y and Z:

$$\mu_{\underline{P}Y}(0) = 0.2 \quad \mu_{\underline{P}Z}(0) = 0.3.$$

The extent to which object 0 belongs to the fuzzy positive region can be determined by considering the union of fuzzy P-lower approximations:

$$\mu_{POS_P(Q)}(0) = \sup_{S\in\mathbb{U}/Q} \mu_{\underline{P}S}(0) = 0.3.$$

Similarly, for the remaining objects,

$$\mu_{POS_P(Q)}(1) = 0.6, \quad \mu_{POS_P(Q)}(2) = 0.3,$$

$$\mu_{POS_P(Q)}(3) = 0.6, \quad \mu_{POS_P(Q)}(4) = 0.5,$$

$$\mu_{POS_P(Q)}(5) = 0.3, \quad \mu_{POS_P(Q)}(6) = 0.6,$$

$$\mu_{POS_P(Q)}(7) = 0.3, \quad \mu_{POS_P(Q)}(8) = 0.3.$$

Using these values, the new degree of dependency of Q on $P = \{A\}$ can be calculated:

$$\gamma'_P(Q) = \frac{\sum_{x\in\mathbb{U}} \mu_{POS_P(Q)}(x)}{|0,1,2,3,4,5,6,7,8|} = 3.8/9.$$

The fuzzy-rough QUICKREDUCT algorithm uses this process to evaluate subsets of features in an incremental fashion. The algorithm starts with an empty set and considers the addition of each individual feature:

$$\gamma'_{\{A\}}(Q) = 3.8/9,$$

$$\gamma'_{\{B\}}(Q) = 2.1/9,$$

$$\gamma'_{\{C\}}(Q) = 2.7/9.$$

As feature A causes the greatest increase in dependency degree, it is added to the reduct candidate and the search progresses:

$$\gamma'_{\{A,B\}}(Q) = 4.0/9,$$

$$\gamma'_{\{A,C\}}(Q) = 5.7/9.$$

Here, C is added to the reduct candidate as the dependency is increased. There is only one feature addition to be checked at the next stage, namely

$$\gamma'_{\{A,B,C\}}(Q) = 5.7/9.$$

Table 1
Example dataset

Case	A			B			C		Plan		
	$A1$	$A2$	$A3$	$B1$	$B2$	$B3$	$C1$	$C2$	X	Y	Z
1	0.3	0.7	0.0	0.2	0.7	0.1	0.3	0.7	0.1	0.9	0.0
2	1.0	0.0	0.0	1.0	0.0	0.0	0.7	0.3	0.8	0.2	0.0
3	0.0	0.3	0.7	0.0	0.7	0.3	0.6	0.4	0.0	0.2	0.8
4	0.8	0.2	0.0	0.0	0.7	0.3	0.2	0.8	0.6	0.3	0.1
5	0.5	0.5	0.0	1.0	0.0	0.0	0.0	1.0	0.6	0.8	0.0
6	0.0	0.2	0.8	0.0	1.0	0.0	0.0	1.0	0.0	0.7	0.3
7	1.0	0.0	0.0	0.7	0.3	0.0	0.2	0.8	0.7	0.4	0.0
8	0.1	0.8	0.1	0.0	0.9	0.1	0.7	0.3	0.0	0.0	1.0
9	0.3	0.7	0.0	0.9	0.1	0.0	1.0	0.0	0.0	0.0	1.0

This causes no dependency increase, resulting in the algorithm terminating and outputting the reduct $\{A, C\}$. Hence, the original dataset can be reduced to these features with minimal information loss (according to the algorithm). Fuzzy rule induction can now be performed on the resulting reduced dataset.

2.3. Fuzzy rule induction

To show the potential utility of fuzzy-rough feature selection, the FRFS method is applied as a pre-processor to an existing fuzzy rule induction algorithm (RIA). The algorithm used is a recent one as described in Ref. [4]. For self-containedness, a brief overview of the RIA is provided here. For simplicity in outlining this induction procedure the original dataset given in Table 1 (see Section 2.2.4) is reused. There are three features each with corresponding linguistic terms, e.g. A has terms $A1$, $A2$ and $A3$. The decision feature $Plan$ is also fuzzy, separated into three linguistic decisions X, Y and Z.

The algorithm begins by organising the dataset objects into subgroups according to their highest decision value. Within each subgroup, the fuzzy subsethood [22,23] is calculated between the decisions of the subgroup and each feature term. Fuzzy subsethood is defined as follows:

$$S(A,B) = \frac{M(A \cap B)}{M(A)} = \frac{\sum_{u \in \mathbb{U}} \min(\mu_A(u), \mu_B(u))}{\sum_{u \in \mathbb{U}} \mu_A(u)}. \quad (23)$$

From this the subsethood values listed in Table 2 can be obtained. Where, for instance, $S(X, A1) = 1$ is obtained by taking the subgroup of objects that belong to the decision X, while

$$M(X) = 0.8 + 0.6 + 0.7 = 2.1,$$

$$M(X \cap A1) = \min(0.8, 1) + \min(0.6, 0.8) + \min(0.7, 1)$$
$$= 0.8 + 0.6 + 0.7 = 2.1,$$

Thus $S(X, A1) = 2.1/2.1 = 1$.

These subsethood values are an indication of the relatedness of the individual terms of the conditional features (or values of the features) to the decisions. A suitable level threshold, $\alpha \in [0, 1]$, must be chosen beforehand in order to determine whether terms are close enough or not. At most, one term is selected per feature. For example, setting $\alpha = 0.9$ means that the term with the highest fuzzy subsethood value (or its negation) above this threshold will be chosen. Applying this process to the first two decision values X and Y generates the rules:

Rule 1. IF A is $A1$ THEN $Plan$ is X.

Rule 2. IF B is NOT $B3$ AND C is $C2$ THEN $Plan$ is Y.

A problem is encountered here when there are no suitably representative terms for a decision (as is the case for decision Z). In this situation, a rule is produced that classifies cases to the decision value if the other rules do not produce reasonable classifications, in order to entail full coverage of the learned rules over the entire problem domain. This requires another threshold value, $\beta \in [0, 1]$, which determines whether a classification is reasonable or not. For decision Z, the following rule is produced:

Rule 3. IF $MF(Rule1) < \beta$ AND $MF(Rule2) < \beta$ THEN $Plan$ is Z

where MF(Rule i) = MF(condition part of Rule i) and MF means the membership function value.

The classification results when using these rules on the example dataset can be found in Table 3. It shows the membership degrees of the cases to each classification for the classified plan and the underlying plan present in the training dataset. Clearly, the resulting classifications are the same when the min t-norm is used.

This technique has been shown to produce highly competitive results [4] in terms of both classification accuracy and number of rules generated. However, as is the case for most rule induction algorithms, the resultant rules may be unnecessarily complex due to the presence of redundant or misleading features. Fuzzy-rough feature selection may be used to significantly reduce dataset dimensionality, removing redundant features that would otherwise increase rule

Table 2
Subsethood values between conditional feature terms and the decision terms

Plan	Linguistic term							
	$A1$	$A2$	$A3$	$B1$	$B2$	$B3$	$C1$	$C2$
X	1	0.1	0	0.71	0.43	0.14	0.52	0.76
Y	0.33	0.58	0.29	0.42	0.58	0.04	0.13	0.92
Z	0.14	0.64	0.29	0.32	0.61	0.14	0.82	0.25

Table 3
Classified plan with all features and the actual plan

Case	Classified			Actual		
	X	Y	Z	X	Y	Z
1	0.3	0.7	0.0	0.1	0.9	0.0
2	1.0	0.3	0.0	0.8	0.2	0.0
3	0.0	0.4	1.0	0.0	0.2	0.8
4	0.8	0.7	0.0	0.6	0.3	0.1
5	0.5	1.0	0.0	0.6	0.8	0.0
6	0.0	1.0	0.0	0.0	0.7	0.3
7	1.0	0.8	0.0	0.7	0.4	0.0
8	0.1	0.3	1.0	0.0	0.0	1.0
9	0.3	0.0	1.0	0.0	0.0	1.0

Rule 1: IF A is $A1$ THEN $Plan$ is X

Rule 2: IF C is $C2$ THEN $Plan$ is Y

Rule 3: IF $MF(Rule1) < \beta$ AND $MF(Rule2) < \beta$ THEN $Plan$ is Z

Fig. 3. Generated rules using the reduced dataset.

complexity and reducing the time for the induction process itself.

As has been demonstrated previously, the example dataset may be reduced by the removal of feature B with little reduction in classification accuracy (according to FRFS). Using this reduced dataset, the RIA generates the rules given in Fig. 3. From this, it can be seen that rule 2 has been simplified due to the redundancy of feature B. Although the extent of simplification is small in this case, with larger datasets the effect can be expected to be greater.

The results using the FRFS-reduced dataset are provided in Table 4. The differences between the classifications of the reduced and unreduced approaches have been highlighted (cases 4 and 7). In case 4, only the membership degree for Y has changed. This value has increased from 0.7 to 0.8, resulting in an ambiguous classification. Again, for case 7, the membership degree for Y is the only value to have changed; this time it more closely resembles the classification present in the training dataset.

3. A realistic application

In order to evaluate the utility of the FRFS approach and to illustrate its domain independence, a challenging test dataset was chosen, namely the Water Treatment Plant Database [24]. The dataset itself is a set of historical data charted over 521 days, with 38 different input features measured daily. Each day is classified into one of thirteen categories depending on the operational status of the plant. However, these can be collapsed into just two or three categories (i.e. *Normal* and *Faulty*, or *OK*, *Good* and *Faulty*) for plant monitoring purposes as many classifications reflect similar performance. Because of the efficiency of the actual plant the measurements were taken from, all faults appear for short periods (usually single days) and are dealt with immediately. This does not allow for a lot of training examples of faults, which is a clear drawback if a monitoring system is to be produced. Note that this dataset has been utilised in many previous studies, including that reported in Ref. [17] (to illustrate the effectiveness of applying crisp RSFS as a pre-processing step to rule induction).

The 38 conditional features account for the following five aspects of the water treatment plant's operation (see Fig. 4):

(1) input to plant (9 features),
(2) input to primary settler (6 features),
(3) input to secondary settler (7 features),
(4) output from plant (7 features),
(5) overall plant performance (9 features).

It is likely that not all of the 38 input features are required to determine the status of the plant, hence the dimensionality reduction step. However, choosing the most informative features is a difficult task as there will be many dependencies

Table 4
Classified plan with reduced features and the actual plan

Case	Classified			Actual		
	X	*Y*	*Z*	*X*	*Y*	*Z*
1	0.3	0.7	0.0	0.1	0.9	0.0
2	1.0	0.3	0.0	0.8	0.2	0.0
3	0.0	0.4	1.0	0.0	0.2	0.8
4	0.8	0.8	0.0	0.6	0.3	0.1
5	0.5	1.0	0.0	0.6	0.8	0.0
6	0.0	1.0	0.0	0.0	0.7	0.3
7	1.0	0.3	0.0	0.7	0.4	0.0
8	0.1	0.3	1.0	0.0	0.0	1.0
9	0.3	0.0	1.0	0.0	0.0	1.0

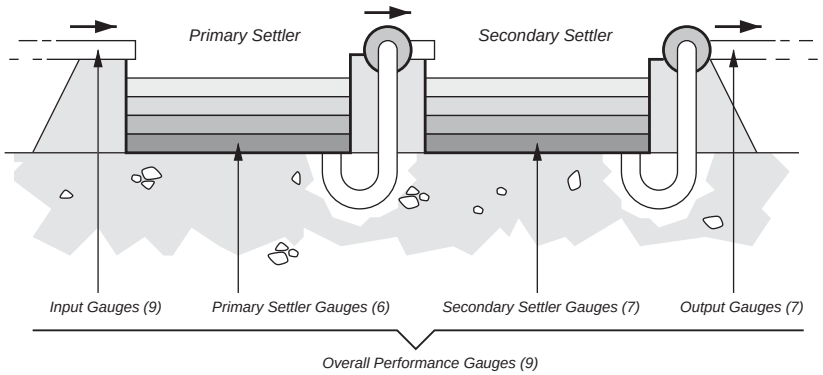

Fig. 4. Water treatment plant, with number of measurements shown at different points in the system.

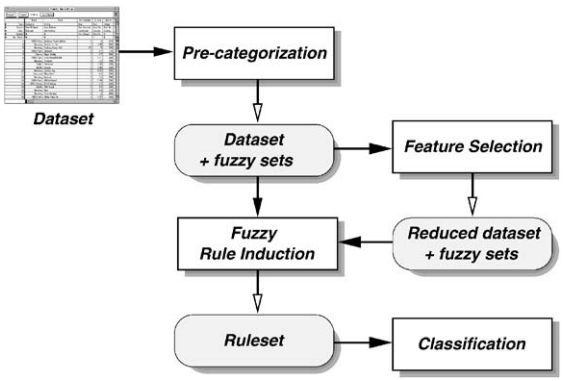

Fig. 5. Modular decomposition of the implemented system.

between subsets of features. There is also a monetary cost involved in monitoring these inputs, so it is desirable to reduce this number.

Note that the original monitoring system (Fig. 5) developed in Ref. [17] consisted of several modules; it is this modular structure that allows the new FRFS technique to replace the existing crisp method. Originally, a precategorisation step preceded feature selection where feature values were quantised. To reduce potential loss of information, the original use of just the dominant symbolic labels of the discretised fuzzy terms is now replaced by a fuzzification procedure. This leaves the underlying feature values unchanged but generates a series of fuzzy sets for each feature. These sets are generated entirely from the data while exploiting the statistical data attached to the dataset (in keeping with the rough set ideology in that the dependence of learning upon information provided outside of the training dataset is minimised). This module may be replaced by alternative fuzzifiers, or expert-defined fuzzification if available.

Based on these fuzzy sets and the original real-valued dataset, FRFS calculates a reduct and reduces the dataset accordingly. Finally, fuzzy rule induction is performed on the reduced dataset using the modelling algorithm given in [4]. Note that this algorithm is not optimal, nor is the fuzzification. Yet the comparisons given below are fair due to their common background. Alternative fuzzy modelling techniques can be employed for this if available.

4. Experimental results

This section first provides the results for the FRFS-based approach compared with the unreduced approach. Next, a comparative experimental study is carried out between various dimensionality reduction methods; namely FRFS, entropy-based feature selection, PCA and a random reduction technique.

The experiments were carried out over a tolerance range (with regard to the employment of the RIA). As mentioned earlier, a suitable value for the threshold α must be chosen before rule induction can take place. However, the selection of α tends to be an application-specific task. A good choice for this threshold that provides a balance between a resultant ruleset's complexity and accuracy can be found by experiment. It should be noted here that due to the fuzzy rule induction method chosen, all approaches generate exactly the same number of rules (as the number of classes of interest), but the arities in different rulesets differ.

4.1. Comparison with the use of unreduced features

First of all, it is important to show that, at least, the use of features selected does not significantly reduce the classification accuracy as compared to the use of the full set of original features. For the 2-class problem, the fuzzy-rough set-based feature selector returns 10 features out of the original 38.

Fig. 6 compares the classification accuracies of the reduced and unreduced datasets on both the training and testing data. As can be seen, the FRFS results are almost always better than the unreduced accuracies over the tolerance range. The best results for FRFS were obtained when α is in the range 0.86–0.90, producing a classification accuracy of 83.3% on the training set and 83.9% for the test data. Compare this with the optimum for the unreduced approach, which gave an accuracy of 78.5% for the training data and 83.9% for the test data.

By using the FRFS-based approach, rule complexity is greatly reduced. Fig. 7 charts the average rule complexity over the tolerance range for the two approaches. Over the range of α values, FRFS produces significantly less complex rules while having a higher resultant classification accuracy. The average rule arity of the FRFS optimum is 1.5 ($\alpha \in (0.86, 0.9)$) which is less than that of the unreduced optimum, 6.0.

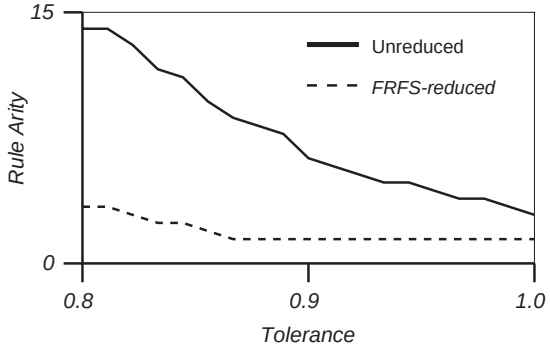

Fig. 7. Average rule arities for the 2-class dataset.

The 3-class dataset is a more challenging problem, reflected in the overall lower classification accuracies produced. The fuzzy-rough method chooses 11 out of the original 38 features. The results of both approaches can be seen in Fig. 8. Again, it can be seen that FRFS outperforms the unreduced approach on the whole. The best classification accuracy obtained for FRFS was 70.0% using the training data, 71.8% for the test data ($\alpha = 0.81$). For the unreduced approach, the best accuracy obtained was 64.4% using the training data, 64.1% for the test data ($\alpha = 0.88$).

Fig. 9 compares the resulting rule complexity of the two approaches. It is evident that rules induced using FRFS as a preprocessor are simpler, with little loss in classification accuracy. In fact, the simple rules produced regularly outperform the more complex ones generated by the unreduced approach. The average rule arity of the FRFS optimum is 4.0 which is less than that of the unreduced optimum, 8.33.

These results show that FRFS is useful not only in removing redundant feature measures but also in dealing with the noise associated with such measurements. To demonstrate that the resulting rules are comprehensible, two sets of rules produced by the induction mechanism are given in Fig. 10. The rules produced are reasonably short and understandable. However, when semantics-destroying dimensionality reduction techniques are applied, such readability is lost.

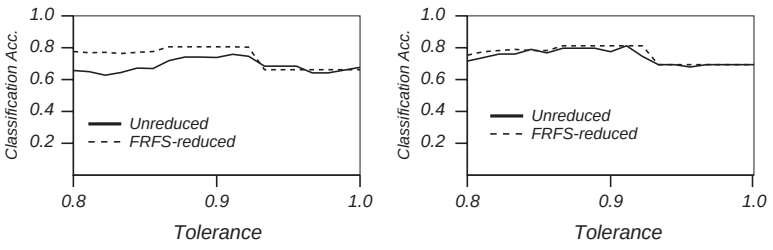

Fig. 6. Training and testing accuracies for the 2-class dataset over the tolerance range.

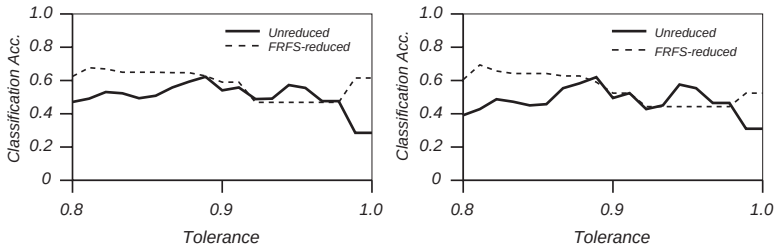

Fig. 8. Training and testing accuracies for the 3-class dataset over the tolerance range.

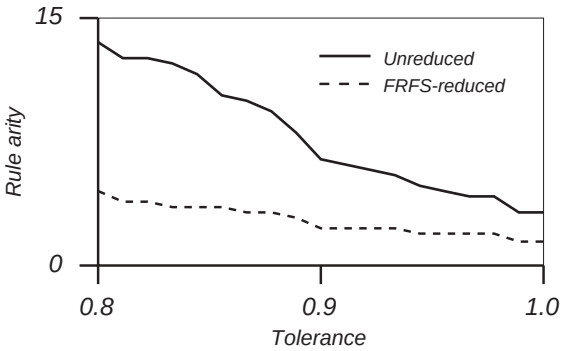

Fig. 9. Average rule arities for the 3-class dataset.

Rules from FRFS-reduced data

IF SED-S IS Medium THEN Situation IS Normal

IF PH-E IS NOT High AND SSV-E IS Low AND SSV-P IS NOT Medium AND PH-D IS NOT High AND DQO-D IS NOT Medium AND PH-S IS NOT High THEN Situation IS Good

IF PH-E IS NOT High AND SSV-E IS Low AND SSV-P IS Low AND DQO-D IS NOT High AND SED-S IS Medium THEN Situation IS Faulty

Rules from unreduced data

IF ZN-E IS NOT High AND SS-E IS NOT High AND SED-E IS NOT High AND SSV-D IS NOT High AND DBO-S IS Low AND SS-S IS NOT High AND SED-S IS Low THEN Situation IS Normal

IF ZN-E IS Low AND PH-E IS NOT High AND SSV-E IS NOT High AND PH-P IS NOT High AND SSV-P IS NOT High AND PH-D IS NOT High AND DBO-D IS NOT Medium AND SSV-D IS NOT High AND SS-S IS NOT High THEN Situation IS Good

IF SSV-E IS NOT High AND SSV-P IS Low AND DQO-D IS NOT High AND SSV-D IS NOT High AND SED-D IS NOT High AND DBO-S IS Low AND SS-S IS NOT High AND SSV-S IS NOT High AND SED-S IS Low THEN Situation IS Faulty

Fig. 10. A selection of generated rulesets.

4.2. Comparison with entropy-based feature selection

To support the study of the performance of FRFS for use as a pre-processor to rule induction, a conventional entropy-based technique is used for comparison. This approach utilises the entropy heuristic employed by machine learning techniques such as C4.5 [7]. Those features that provide the most gain in information are selected. A summary of the results of this comparison can be seen in Table 5.

For both the 2- and 3-class datasets, FRFS selects three fewer features than the entropy-based method. FRFS has a higher training accuracy and the same testing accuracy for the 2-class data using less features. However, for the 3-class data, the entropy-based method produces a very slightly higher testing accuracy. Again, it should be noted that this is obtained with three additional features over the FRFS approach.

4.3. Comparison with PCA and random reduction

The above comparison ensured that little information loss is incurred due to FRFS. The question now is whether any other feature sets of a dimensionality 10 (for the 2-class dataset) and 11 (for the 3-class dataset) would perform similarly. To avoid a biased answer to this, without resorting to exhaustive computation, 70 sets of random reducts were chosen of size 10 for the 2-class dataset, and a further 70 of size 11 for the 3-class dataset to see what classification results might be achieved. The classification accuracies for each tolerance value are averaged.

The effect of using a different dimensionality reduction technique, namely PCA, is also investigated. To ensure that the comparisons are fair, only the first 10 principal components are chosen for the 2-class dataset (likewise, the first 11 for the 3-class dataset). As PCA irreversibly destroys the underlying dataset semantics, the resulting rules are not human comprehensible but may still provide useful automatic classifications of new data.

The results of FRFS, PCA and random approaches can be seen in Fig. 11 for the 2-class dataset. On the whole, FRFS produces a higher classification accuracy than both PCA-based and random-based methods over the tolerance range. FRFS results in the highest individual classification accuracy for training and testing data (see Table 6).

For the 3-class dataset, the results of FRFS, PCA and random selection can be seen in Fig. 12. The individual best accuracies can be seen in Table 7. Again, FRFS produces the highest classification accuracy (71.8%), and is almost

Table 5
Comparison of FRFS and entropy-based feature selection

Approach	No. of classes	Selected features	No. of features	Training accuracy (%)	Testing accuracy (%)
FRFS	2	{0, 2, 6, 10, 12, 15, 22, 24, 26, 37}	10	83.3	83.9
Entropy	2	{1, 5, 6, 7, 9, 12, 15, 16, 20, 22, 29, 30, 33}	13	80.7	83.9
FRFS	3	{2, 3, 6, 10, 12, 15, 17, 22, 27, 29, 37}	11	70.0	71.8
Entropy	3	{6, 8, 10, 12, 17, 21, 23, 25, 26, 27, 29, 30, 34, 36}	14	70.0	72.5

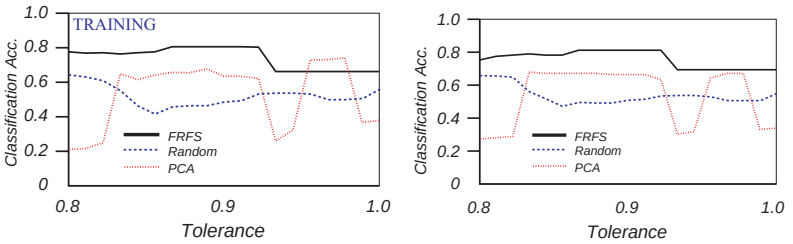

Fig. 11. Training and testing accuracies for the 2-class dataset: comparison with PCA and random-reduction methods.

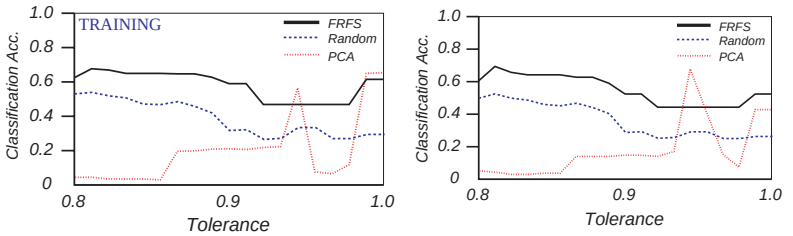

Fig. 12. Training and testing accuracies for the 3-class dataset: comparison with PCA and random-reduction methods.

Table 6
Best individual classification accuracies (2-class dataset) for FRFS, PCA and random approaches

Approach	Training accuracy (%)	Testing accuracy (%)
FRFS	83.3	83.9
Random	66.4	68.1
PCA	76.7	70.3

Table 7
Best resultant classification accuracies (3-class dataset) for FRFS, PCA and random approaches

Approach	Training accuracy (%)	Testing accuracy (%)
FRFS	70.0	71.8
Random	55.7	54.3
PCA	67.7	70.2

always the best over the tolerance range. Although PCA produces a comparatively high accuracy of 70.2%, this is at the expense of incomprehensible rules.

5. Conclusion

Automated generation of feature pattern-based if–then rules is essential to the success of many intelligent pattern classifiers, especially when their inference results are expected to be directly human-comprehensible. This paper has presented such an approach which integrates a recent fuzzy rule induction algorithm with a fuzzy-rough method for feature selection. Unlike semantics-destroying approaches such as PCA, this approach maintains the underlying semantics of the feature set, thereby ensuring that the resulting models are interpretable and the inference explainable. Not only are the rules simplified by the use of FRFS, but the resulting classification accuracies are in fact *improved*. The method alleviates important problems encountered by traditional RSFS such as dealing with noise and real-valued features.

In all experimental studies there has been no attempt to optimise the fuzzifications or the classifiers employed. It can be expected that the results obtained with optimisation would be even better than those already observed. The generality of this approach should enable it to be applied to other domains. The ruleset generated by the RIA was not processed by any post-processing tools so as to allow its behaviour

and capabilities to be revealed fully. By enhancing the induced ruleset through post-processing, performance should improve. Additionally, other fuzzy rule induction algorithms may be used. The current RIA may be easily replaced due to the modularity of the system. Similar work has been carried out using Lozowski's algorithm [15,18] which, being exhaustive in nature, benefits greatly from a feature selection pre-processing stage.

Work is being carried out on a fuzzified dependency function [25]. Ordinarily, the dependency function returns values for sets of features in the range [0,1]; the fuzzy dependency function will return qualitative fuzzy labels for use in the new QUICKREDUCT algorithm. With this mechanism in place, several features may be chosen at one time according to their labels, speeding up the feature selection process. Additionally, research is being carried out into the potential utility of *fuzzy reducts*, which would allow features to have a varying possibility of becoming a member of the resultant reduct. Further work also includes broadening the comparative studies to include comparisons with other feature selection and dimensionality reduction techniques. In particular, studies using the Isomap algorithm [27], a recent successful dimensionality reduction technique, should be beneficial.

Acknowledgements

This work is partly funded by the UK EPSRC grant 00317404. The authors are very grateful to Alexios Chouchoulas and David Robertson for their support.

References

[1] W. Pedrycz, G. Vukovich, Feature analysis through information granulation, Pattern Recognition 35 (4) (2002) 825–834.
[2] M. Sebban, R. Nock, A hybrid filter/wrapper approach of feature selection using information theory, Pattern Recognition 35 (4) (2002) 835–846.
[3] N. Kwak, C.H. Choi, Input feature selection for classification problems, IEEE Trans. Neural Networks 13 (1) (2002) 143–159.
[4] S. Chen, S.L. Lee, C. Lee, A new method for generating fuzzy rules from numerical data for handling classification problems, Appl. Artif. Intell. 15 (7) (2001) 645–664.
[5] W.H. Au, K.C.C. Chan, An effective algorithm for discovering fuzzy rules in relational databases, Proceedings of the Seventh IEEE International Conference on Fuzzy Systems, Anchorage, Alaska, 1998, pp. 1314–1319.
[6] K. Chan, A. Wong, APACS: a system for automatic analysis and classification of conceptual patterns, Comput. Intell. 6 (1990) 119–131.
[7] J.R. Quinlan, C4.5: Programs for Machine Learning, The Morgan Kaufmann Series in Machine Learning. Morgan Kaufmann Publishers, San Mateo, CA, 1993.
[8] I. Hayashi, T. Maeda, A. Bastian, L.C. Jain, Generation of fuzzy decision trees by fuzzy ID3 with adjusting mechanism of AND/OR operators, Proceedings of the Seventh IEEE International Conference on Fuzzy Systems, Anchorage, Alaska, 1998, pp. 681–685.
[9] C.Z. Janikow, Fuzzy decision trees: issues and methods, IEEE Trans. Systems Man Cybernet.—Part B: Cybernet. 28 (1998) 1–14.
[10] P. Devijver, J. Kittler, Pattern Recognition: A Statistical Approach, Prentice-Hall, Englewood Cliffs, NJ, 1982.
[11] B. Flury, H. Riedwyl, Multivariate Statistics: A Practical Approach, Prentice-Hall, Englewood Cliffs, NJ, 1988.
[12] Z. Pawlak, Rough Sets: Theoretical Aspects of Reasoning About Data, Kluwer Academic Publishing, Dordrecht, 1991.
[13] L.A. Zadeh, Fuzzy sets, Inform. and Control 8 (1965) 338–353.
[14] D. Dubois, H. Prade, Putting rough sets and fuzzy sets together, in: R. Slowinski (Ed.), Intelligent Decision Support, Kluwer Academic Publishers, Dordrecht, 1992, pp. 203–232.
[15] R. Jensen, Q. Shen, Fuzzy-rough sets for descriptive dimensionality reduction, Proceedings of the 11th International Conference on Fuzzy Systems, 2002, pp. 29–34.
[16] R. Jensen, Q. Shen, Fuzzy-rough attribute reduction with application to web categorization, Fuzzy Sets Systems 141 (3) (2004) 469–485.
[17] Q. Shen, A. Chouchoulas, A fuzzy-rough approach for generating classification rules, Pattern Recognition 35 (11) (2002) 341–354.
[18] A. Lozowski, T.J. Cholewo, J.M. Zurada, Crisp rule extraction from perceptron network classifiers, Proceedings of International Conference on Neural Networks, Volume of Plenary, Panel and Special Sessions, 1996, pp. 94–99.
[19] A. Chouchoulas, J. Halliwell, Q. Shen, On the implementation of rough set attribute reduction, Proceedings of the 2002 UK Workshop on Computational Intelligence, 2002, pp. 18–23.
[20] S.K. Pal, A. Skowron (Eds.), Rough-Fuzzy Hybridization: A New Trend in Decision Making, Springer, Singapore, 1999.
[21] U. Höhle, Quotients with respect to similarity relations, Fuzzy Sets and Systems 27 (1988) 31–44.
[22] B. Kosko, Fuzzy entropy and conditioning, Inform. Sci. 40 (2) (1986) 165–174.
[23] Y. Yuan, M.J. Shaw, Induction of fuzzy decision trees, Fuzzy Sets Systems 69 (2) (1995) 125–139.
[24] C.L. Blake, C.J. Merz, UCI Repository of machine learning databases, Department of Information and Computer Science, University of California, Irvine, CA, 1998, http://www.ics.uci.edu/~mlearn/MLRepository.html.
[25] R. Jensen, Q. Shen, Using fuzzy dependency-guided attribute grouping in feature selection, in: G. Wang, et al., (Eds.), Rough Sets, Fuzzy Sets, Data Mining and Granular Computing, Springer, Berlin, 2003, pp. 250–254.
[26] R. Slowinski (Ed.), Intelligent Decision Support, Kluwer Academic Publishers, Dordrecht, 1992.
[27] J.B. Tenenbaum, V. de Silva, J.C. Langford, A global geometric framework for nonlinear dimensionality reduction, Science 290 (2000) 2319–2323.