Simultaneous feature and instance selection using fuzzy-rough bireducts

MacParthaláin, Neil Seosamh; Jensen, Richard

Published in:
Proceedings of the 22nd IEEE International Conference on Fuzzy Systems

Publication date:
2013

Citation for published version (APA):
MacParthaláin, N. S., & Jensen, R. (2013). Simultaneous feature and instance selection using fuzzy-rough bireducts. In *Proceedings of the 22nd IEEE International Conference on Fuzzy Systems* IEEE Press.

Simultaneous Feature And Instance Selection Using Fuzzy-Rough Bireducts

Neil Mac Parthaláin
Dept. of Computer Science
Aberystwyth University
Aberystwyth, Ceredigion, Wales. UK
Email: ncm@aber.ac.uk

Richard Jensen
Dept. of Computer Science
Aberystwyth University
Aberystwyth, Ceredigion, Wales. UK
Email: rkj@aber.ac.uk

***Abstract*—Rough set theory has proven to be a useful mathematical basis for developing automated computational approaches which are able to deal with and utilise imperfect knowledge. Ever since its inception, this theory has been successfully employed for developing computationally efficient techniques for addressing problems such as the discovery of hidden patterns in data, decision rule induction, and feature selection. As an extension to this theory, fuzzy-rough sets enhance the ability to model uncertainty and vagueness more effectively. The efficacy of fuzzy-rough set based approaches for the tasks of feature selection and rule induction is now well established in the literature. Although some work has been carried out using fuzzy-rough set theory for the tasks of feature selection and instance selection in isolation, the potential of this theory for its application to tasks for the simultaneous selection of both features and instances has not been investigated thus far. This paper proposes a novel method for simultaneous instance and feature selection based on fuzzy-rough sets. The initial experimentation demonstrates that the method can significantly reduce both the number of instances and features whilst maintaining high classification accuracies.**

***Index Terms*—fuzzy-rough sets, feature selection, instance selection, discernibility.**

I. Introduction

The increasing trend towards the archiving of enormous amounts of data has led to a situation where such data is now stored and maintained in the hope that it will be processed at a later date. The continuing growth of this data means that the corresponding data collections are also expansive; both in terms of dimensionality (number of features), and number of data objects or instances. This means that there is an ever-increasing demand in terms of resources for the storage and maintenance of data. There is therefore, a growing need for methods such as feature selection (FS) which can contribute greatly to reducing the size of such data. FS is focused on the reduction of dimensionality by removing features which may be noisy, redundant, irrelevant, or even misleading [1] The central aim of FS is to select a minimal subset of features from a given domain whilst retaining a suitably high accuracy in representing the original set of features. In addition to the high dimensionality of data, there is also the problem where prohibitively high numbers of training instances may be present. In this case, approaches such as instance selection (IS) [2] are desirable in order to reduce the volume of the data to a more manageable size, whilst also removing misleading training instances. This also has the effect of improving any models which are learned from the data. By employing techniques such as FS and IS, data size can be reduced considerably, thereby alleviating management, storage and maintenance resource requirements.

Rough set theory (RST) [3] has attracted great interest amongst researchers in recent years and has been applied to various real-world application domains. One of the main reasons for the popularity of RST stems from a number of appealing aspects of the theory. Indeed, the focus of RST on grouping information entities into 'granules' in terms of some form of relatedness, offers a certain universal intuitive appeal. In addition to this, it has other desirable attributes; for example, no tunable parameters are required, thus eliminating the need for (possibly erroneous) subjective human input, and it also finds a minimal knowledge representation. One of the problems for RST however, is that it is constrained to problem domains where the data is crisp-valued. It is this inability to deal with real-valued data which has resulted in the development of fuzzy-rough sets. Much work has been carried out in the area of FS using both rough and fuzzy-rough sets [4], and indeed some work has also been developed recently for the task of IS [2]. However, there has been little or no work in the area of fuzzy-rough sets for the task of *simultaneous* feature and instance selection. This paper proposes a novel approach to simultaneous instance and feature selection based upon fuzzy-rough sets. The central idea behind this approach is to remove instances and features from a dataset in an iterative but simultaneous fashion by extending the work on (crisp) bireducts described in [5], to the fuzzy case. The reason for the fuzzification of such an approach is that fuzzy-rough sets have the ability to consider real-valued data, while rough sets can only deal with crisp-valued discrete data. The new fuzzy-rough based bireduct definitions are then used to frame the problem as a satisfiability problem using a Boolean representation in CNF form. By removing both instances, and features, the dataset can be quickly reduced both in terms of dimensionality and number of training instances.

The remainder of this paper is structured as follows: Section 2 summarises the necessary theoretical basis and concepts of fuzzy-rough sets. Section 3 details the proposed approach to simultaneous fuzzy-rough instance and feature selection. Initial experimental results are demonstrated in Section 4

that show the potential of the approach. Finally, the paper is concluded in Section 5 with some discussion and suggestions for future work.

II. Theoretical Background

A. Rough and fuzzy-rough sets

At the very heart of the RST is the concept of indiscernibility [3]. Let $I = (\mathbb{U}, \mathbb{S})$ be an information system, where $\mathbb{U}$ is a non-empty set of finite instances (the universe of discourse) and $\mathbb{S}$ is a non-empty finite set of features so that $a : \mathbb{U} \rightarrow V_a$ for every $a \in \mathbb{S}$. V_a is the set of values that a can take. For any $P \subseteq \mathbb{S}$, there exists an associated equivalence relation $IND(P)$:

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

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

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

where,

$$S \otimes T = \{X \cap Y : \forall X \in S, \forall Y \in T, X \cap Y \neq \emptyset\} \quad (3)$$

If $(x, y) \in IND(P)$, then x and y are indiscernible by attributes from P. The equivalence classes of the P-indiscernibility relation are denoted $[x]_P$. Let $X \subseteq \mathbb{U}$. X can be approximated using only the information contained in P by constructing the P-*lower* and P-*upper* approximations of X:

$$\underline{P}X = \{x : [x]_P \subseteq X\} \quad (4)$$

$$\overline{P}X = \{x : [x]_P \cap X \neq \emptyset\} \quad (5)$$

Let P and Q be equivalence relations over $\mathbb{U}$, then the concepts of the positive, negative and boundary regions can be defined:

$$POS_P(Q) = \bigcup_{X \in \mathbb{U}/Q} \underline{P}X \quad (6)$$

$$NEG_P(Q) = \mathbb{U} - \bigcup_{X \in \mathbb{U}/Q} \overline{P}X \quad (7)$$

$$BND_P(Q) = \bigcup_{X \in \mathbb{U}/Q} \overline{P}X - \bigcup_{X \in \mathbb{U}/Q} \underline{P}X \quad (8)$$

By employing this definition of the positive region it is possible to calculate the rough set degree of dependency of a set of attributes Q on a set of attributes P. This can be achieved as follows: For P,$Q \subseteq S$, it can be said that Q depends on P in a degree k $(0 \leq k \leq 1)$, thus the higher the value of k the more dependent Q is upon P. This is denoted $(P \Rightarrow_k Q)$ if:

$$k = \gamma_P(Q) = \frac{|POS_P(Q)|}{|\mathbb{U}|} \quad (9)$$

Fuzzy-rough sets have been proposed in order to improve the ability to deal with uncertainty and vagueness present in data. A fuzzy-rough set [4] is defined by two fuzzy sets, fuzzy lower and upper approximations, obtained by extending the corresponding crisp rough set notions defined in (4) and (5) previously. In the crisp case, elements that belong to the lower approximation (i.e. have a membership of 1) are said to belong to the approximated set with absolute certainty. In the fuzzy-rough case, elements may have a membership in the range [0,1], thus allowing greater flexibility in handling uncertainty. Fuzzy-rough sets encapsulate the related but distinct concepts of vagueness (for fuzzy sets) and indiscernibility (for rough sets), both of which occur as a result of uncertainty in knowledge.

Definitions for the fuzzy lower and upper approximations can be found in [6], where a $\mathcal{T}$-transitive fuzzy similarity relation is used to approximate a fuzzy concept X:

$$\mu_{\underline{R_P}X}(x) = \inf_{y \in \mathbb{U}} \mathcal{I}(\mu_{R_P}(x, y), \mu_X(y)) \quad (10)$$

$$\mu_{\overline{R_P}X}(x) = \sup_{y \in \mathbb{U}} \mathcal{T}(\mu_{R_P}(x, y), \mu_X(y)) \quad (11)$$

Here, $\mathcal{I}$ is a fuzzy implicator and $\mathcal{T}$ a t-norm. A fuzzy implicator is any $[0, 1]^2 \rightarrow [0, 1]$-mapping $\mathcal{I}$ satisfying $\mathcal{I}(0, 0) = 1, \mathcal{I}(1, x) = x$ for all x in $[0, 1]$. R_P is the fuzzy similarity relation induced by the subset of features P:

$$\mu_{R_P}(x, y) = \mathcal{T}_{a \in P}\{\mu_{R_a}(x, y)\} \quad (12)$$

$\mu_{R_a}(x, y)$ is the degree to which instances x and y are similar for feature a, and may be defined in many ways, for example:

$$\mu_{R_a}(x, y) = 1 - \frac{|a(x) - a(y)|}{|a_{max} - a_{min}|} \quad (13)$$

$$\mu_{R_a}(x, y) = \exp(-\frac{(a(x) - a(y))^2}{2{\sigma_a}^2}) \quad (14)$$

$$\mu_{R_a}(x, y) = \max(\min(\frac{(a(y) - (a(x) - \sigma_a))}{\sigma_a}, \frac{((a(x) + \sigma_a) - a(y))}{\sigma_a}, 0) \quad (15)$$

where ${\sigma_a}^2$ is the variance of feature a. The choice of relation is largely determined by the intended application. For feature selection, a relation such as (15) may be appropriate as this permits only small differences between attribute values of differing instances. For classification tasks, a more gradual and inclusive relation such as (13) should be used. Other fuzzy relation definitions can also be found in [7].

In a similar way to the original crisp rough set approach, the fuzzy positive region [1] can be defined as :

$$\mu_{POS_P(\mathbb{D})}(x) = \sup_{X \in \mathbb{U}/\mathbb{D}} \mu_{\underline{R_P}X}(x) \quad (16)$$

An important issue in data analysis is discovering dependencies between attributes. The fuzzy-rough degree of dependency

of $\mathbb{D}$ on the attribute subset P can be defined in the following way:

$$\gamma'_P(\mathbb{D}) = \frac{\sum_{x \in \mathbb{U}} \mu_{POS_P(\mathbb{D})}(x)}{|\mathbb{U}|} \tag{17}$$

A fuzzy-rough reduct R can be defined as a minimal subset of features that preserves the dependency degree of the entire dataset, i.e. $\gamma'_R(\mathbb{D}) = \gamma'_{\mathbb{C}}(\mathbb{D})$. Based on this, a fuzzy-rough greedy hill-climbing algorithm can be constructed that uses equation (17) to gauge subset quality. In [1], it has been shown that the dependency function is monotonic and that fuzzy discernibility matrices may also be used to discover reducts.

B. Fuzzy Discernibility Matrices

There are two main branches of research in crisp rough set-based approaches: those based on the dependency degree and those based on discernibility matrices and functions. Therefore, it is a natural progression to extend concepts in the latter branch to the fuzzy-rough domain [8], [1].

1) Fuzzy Discernibility: The crisp discernibility matrix is herein extended by employing fuzzy clauses. Entries in the fuzzy discernibility matrix is a fuzzy set, to which every feature belongs to a certain degree. The extent to which a feature a belongs to the fuzzy clause C_{ij} is determined by the fuzzy discernibility measure:

$$\mu_{C_{ij}}(a) = N(\mu_{R_a}(i,j)) \tag{18}$$

where N denotes fuzzy negation and $\mu_{R_a}(i,j)$ is the fuzzy similarity of instances i and j, and hence $\mu_{C_{ij}}(a)$ is a measure of the fuzzy discernibility. For the crisp case, if $\mu_{C_{ij}}(a)$ = 1 then the two instances are distinct for this feature; if $\mu_{C_{ij}}(a)$ = 0, the two instances are identical. For fuzzy cases where $\mu_{C_{ij}}(a) \in (0,1)$, the instances are partly discernible. Note that the choice of fuzzy similarity relation must be identical to that of the fuzzy-rough dependency degree approach to find corresponding reducts. Each entry (or clause) in the fuzzy indiscernibility matrix is a set of attributes and their memberships:

$$C_{ij} = \{a_x | a \in \mathbb{C}, x = N(\mu_{R_a}(i,j))\}\ i,j = 1,...,|\mathbb{U}| \tag{19}$$

For example, an entry C_{ij} in the fuzzy discernibility matrix may be: $\{a_{0.4}, b_{0.8}, c_{0.2}, d_{0.0}\}$. This denotes that $\mu_{C_{ij}}(a) = 0.4$, $\mu_{C_{ij}}(b) = 0.8$, etc. In crisp discernibility matrices, these values are either 0 or 1 as the underlying relation is an equivalence relation. The example clause can be viewed as indicating the significance value of each feature - the extent to which the feature discriminates between the two instances i and j. The core of the dataset is defined as:

$$\begin{aligned} Core(\mathbb{C}) = \{a \in \mathbb{C} | \exists C_{ij}, \mu_{C_{ij}}(a) > 0, \\ \forall f \in \{\mathbb{C} - a\}\ \mu_{C_{ij}}(f) = 0\} \end{aligned} \tag{20}$$

2) Fuzzy Discernibility Function: As with the crisp approach, the entries in the matrix can be used to construct the fuzzy discernibility function:

$$f_D(a^*_1, ..., a^*_m) = \wedge\{\vee C^*_{ij} | 1 \leq j < i \leq |\mathbb{U}|\} \tag{21}$$

where $C^*_{ij} = \{a^*_x | a_x \in C_{ij}\}$. The function returns values in $[0,1]$, which can be seen to be a measure of the extent to which the function is satisfied for a given assignment of truth values to variables. To discover reducts from the fuzzy discernibility function, the task is to find the minimal assignment of the value `true` to the variables such that the formula is maximally satisfied. By setting all variables to `true`, the maximal value for the function can be obtained as this provides the greatest discernibility between instances.

3) Decision-relative Fuzzy Discernibility Matrix: As with the crisp discernibility matrix, for a decision system the decision feature must be taken into account for achieving reductions; only those clauses with different decision values are included in the crisp discernibility matrix. For the fuzzy version, this is encoded as:

$$\begin{aligned} f_D(a^*_1, ..., a^*_m) = \{\wedge\{\{\vee C^*_{ij}\} \leftarrow q_{N(\mu_{R_q}(i,j))}\} | \\ 1 \leq j < i \leq |\mathbb{U}|\} \end{aligned} \tag{22}$$

for decision feature q, where $\leftarrow$ denotes fuzzy implication. This allows the extent to which decision values differ to affect the overall satisfiability of the clause. If $\mu_{C_{ij}}(q)$ = 1 then this clause provides maximum discernibility (i.e., the two instances are maximally different according to the fuzzy similarity measure). When the decision is crisp and crisp equivalence is used, $\mu_{C_{ij}}(q)$ becomes either 0 or 1.

C. Formulation

The degree of satisfaction for a clause C_{ij} for a given subset of features P is defined as:

$$SAT_P(C_{ij}) = \mathcal{S}_{a \in P}\{\mu_{C_{ij}}(a)\} \tag{23}$$

for a t-conorm $\mathcal{S}$. Returning to the example clause $\{a_{0.4}, b_{0.8}, c_{0.2}, d_{0.0}\}$, if the subset $P = \{a, c\}$ is chosen, the resulting degree of satisfaction of the clause is

$$SAT_P(C_{ij}) = \mathcal{S}\{0.4, 0.2\} = 0.6$$

using the Łukasiewicz t-conorm, $\min(1, x + y)$.

In traditional (crisp) propositional satisfiability, a clause is fully satisfied if at least one variable in the clause has been set to `true`. For the fuzzy case, clauses may be satisfied to a certain degree depending on which variables have been assigned the value `true`. By setting $P = \mathbb{C}$, the maximum satisfiability degree of a clause can be obtained:

$$maxSAT_{ij} = SAT_{\mathbb{C}}(C_{ij}) = \mathcal{S}_{a \in \mathbb{C}}\{\mu_{C_{ij}}(a)\} \tag{24}$$

This is the maximal amount that clause C_{ij} may be satisfied. The maximum satisfiability degree of the example clause is $\mathcal{S}(0.4, 0.8, 0.2, 0.0)$ which evaluates to 1 if the Łukasiewicz t-conorm is used. Here it can be seen that, depending on the t-conorm used, clauses may in fact be maximally satisfied by

the selection of several sub-maximal features. Using the $\max$ t-conorm, the maximum satisfiability degree is 0.8, obtained only by the inclusion of feature b in P.

In this setting, a fuzzy-rough reduct corresponds to a (minimal) truth assignment to variables such that each clause has been satisfied to its maximum extent.

D. Crisp Bireducts

In the work of [5], the authors introduce the idea of a rough set bireduct. This concept is based on a similar, but non-equivalent approach termed approximate reducts [9]. The definition of a bireduct focuses on a subset of features that describes the decision class, and a subset of instances for which such a description is valid. The motivation for this definition draws on the work in the area of unsupervised learning known as biclustering [10]. In [5], the authors define bireducts in the following way:

Def. 1 *Let $\mathbb{I} = (\mathbb{U}, \mathbb{S})$, be an information system. A tuple (B, X), where $B \subseteq \mathbb{S}$ and $X \subseteq \mathbb{U}$ is an information bireduct, iff B discerns all pairs of instances in X, $\forall i, j \in X$, $\exists b \in B \mid b(i) \neq b(j)$, and:*

1) *There is no proper subset $C \subset B$ such that C discerns all pairs in X*
2) *There is no proper superset $Y \supset X$ such that B discerns all pairs in Y*

This idea is further extended to the case where $\mathbb{I}$ is a decision system:

Def. 2 *Let $\mathbb{I} = (\mathbb{U}, \mathbb{S} \cup \{d\})$, be an information system. A tuple (B, X), where $B \subseteq \mathbb{S}$ and $X \subseteq \mathbb{U}$ is a decision bireduct, iff B discerns all pairs of instances $i, j \in X$, where $d(i) \neq d(j)$, and:*

1) *There is no proper subset $C \subset B$ such that C discerns all pairs $i, j \in X$, where $d(i) \neq d(j)$*
2) *There is no proper superset $Y \supset X$ such that B discerns all pairs $i, j \in Y$, where $d(i) \neq d(j)$*

Referring to the earlier RST definitions, it can be seen that a decision bireduct can be regarded as a type of inexact functional dependence which links the subset of features in B with the decision feature d to a degree X. Those instances contained in $\mathbb{U} \setminus X$ are treated as outliers. Conversely those instances in X can be used to learn from the data using the features in B.

The definition of a bireduct in [5], relies on two properties which aim to ensure that the feature subset is minimal and that the coverage is maximal. This can also be formulated in a Boolean propositional setting by considering bireducts as prime implicants of a CNF formula generated from data akin to that used in finding reducts via crisp discernibility matrices [5]. This is exploited and used as the foundation for the fuzzy-rough approach described in the following section.

III. Simultaneous fuzzy-rough instance and feature selection

This section details the proposed approach to simultaneous fuzzy-rough instance and feature selection (SFRIFS). In the traditional approach to finding reducts, each clause is generated by the comparison of pairs of instances, with features appearing in the clauses if their values differ for these instances. Hence, to discern between these pairs of instances, at least one of these features must be selected. However, in the bireduct approach, we can also satisfy a clause by removing either (or both) of the instances that generated it. The reason behind this is that one of the instances may be noisy or even an outlier, and therefore not useful for any subsequent learning process(es) which may be employed. In this case, rather than selecting a number of features to discriminate between this instance and others, it may be better to remove the misleading instance altogether. This provides the motivation for the proposed method for using fuzzy-rough sets to perform simultaneous instance and feature selection.

The previous definition of the fuzzy discernibility function can be extended to the bireduct case as follows:

$$f_D(a_1^*, ..., a_m^*) = \wedge\{i^* \vee j^* \vee C_{ij}^* | 1 \leq j < i \leq |\mathbb{U}|\} \quad (25)$$

Here, it can be seen that a clause is satisfied if C_{ij}^* is maximally satisfied, or if either i^* or j^* are selected (which means they are ultimately removed from the dataset). The task is then to select a subset of features B and remove a subset of instances Z in order to maximally satisfy all of the generated clauses for the dataset. The pair (B, X), $X = \mathbb{U} - Z$, will then be a bireduct as no proper subset of B will discern all instances in X, and no proper superset of X will be discernible by B thus fulfilling the same conditions defined earlier for crisp bireducts.

Having framed the problem in this way, it is clear that there needs to be some mechanism by which to select appropriate features and instances in a systematic and sensible way. Many different search mechanisms are appropriate for this task. However, for the purpose of the initial investigation detailed in this paper, a simple heuristic frequency-of-occurrence type approach based on the *Johnson reducer* is adopted.

A. Algorithm

The algorithm for SFRIFS is shown in Fig. 1. Its first step is to generate the CNF clause list from the data. This is achieved by examining the data in the manner described above before generating a clause for each case where the decision feature differs for any pair of instances. In each case, the clause contains the degree of (fuzzy) discernibility between the features of that pairwise comparison as well as those instances which are compared in order to construct the clause. Once the set of CNF clauses have been formed, the algorithm then considers which features and instances to select. As mentioned previously, a simple but nevertheless effective heuristic based on the Johnson reducer is used here. The basis for this is that the maximum number of clauses is satisfied at each iteration, and therefore the greatest level of reduction is achieved. Obviously, in terms of the features for real-valued data, the discernibility is fuzzy. One approach to selection in this particular situation is to search for a

SFRIS($\mathcal{S}$)
S, the dataset.

(1) $f_D \leftarrow$ GenerateCNF(S)
(2) $R \leftarrow \{\}$; bestA $\leftarrow$ 0; bestFeat $\leftarrow \emptyset$
(3) $O \leftarrow \mathbb{U}$; bestO $\leftarrow$ 0; bestInst $\leftarrow \emptyset$
(4) **while** $f_D! = \emptyset$
(5) **foreach** $a \in (\mathbb{C} - R)$ //choose a feature
(6) c = heuristic(a)
(7) **if** c $>$ bestA
(8) bestA = c; bestFeat $\leftarrow a$
(9) $R \leftarrow R \cup$ bestFeat
(10) f_D = removeClauses(f_D,bestFeat)
(11) **if** $f_D == \emptyset$ **return** (R, O)
(12) **foreach** $o \in O$ //choose an instance
(13) c = heuristic(o)
(14) **if** c $>$ bestO
(15) bestO = c; bestInst $\leftarrow o$
(16) $O \leftarrow O-$ bestInst
(17) f_D = removeClauses(f_D,bestInst)
(18) **if** $f_D == \emptyset$ **return** (R, O)

Fig. 1. The SFRIFS Algorithm

feature that has a non-zero value for discernibility in the greatest number of clauses. Alternatively, the sum of the fuzzy discernibility values for a particular feature across all of the clauses gives a good indication of feature importance. Indeed, this is the heuristic adopted for this work. The selection of instances is much simpler given that the presence or absence of any instance is indicated in the crisp sense. Therefore, an instance is either present or absent from any given clause. This simplifies the selection of instances since their frequency of occurrence may be noted in a conventional crisp manner.

The main loop of the SFRIFS algorithm begins by selecting features initially. There is no particular reason for performing the selection of features prior to instances and indeed the algorithm can also be formulated by selecting instances initially. It is worth noting that this can result in the discovery of a different fuzzy-rough bireduct, however. The algorithm continues to select features and instances alternately until the clause list is fully satisfied ($f_D = \emptyset$). Therefore, for each phase of either feature or instance selection, one or the other is selected. When a feature is selected, all clauses that have been maximally satisfied are removed (f_D = removeClauses(f_D,bestFeat)). In the case of instances, a simple count of frequency-of-occurrence identifies those clauses to be removed (f_D = removeClauses(f_D,bestInst)). Following the selection of either features or instances (and removal of the relevant clauses), the feature (or instance) that is selected is recorded. The process continues until all clauses have been satisfied and the fuzzy-rough bireduct is then returned (lines (11) and (18)).

B. Worked example

An example dataset is shown in Table I which is used in order to illustrate the operation of SFRIFS. The fuzzy connectives used are the Łukasiewicz t-norm ($\max(x+y-1, 0)$) and the Łukasiewicz fuzzy implicator ($\min(1-x+y, 1)$). The use of this implicator is recommended as it is both a residual and S-implicator.

TABLE I
EXAMPLE DATASET

Instance	a	b	c	q
1	-0.4	-0.3	-0.5	no
2	-0.4	0.2	-0.1	yes
3	-0.3	-0.4	-0.3	no
4	0.3	-0.3	0	yes
5	0.2	-0.3	0	yes
6	0.2	0	0	no

Using the fuzzy similarity measure in (15), the resulting relations are as follows for each feature in the dataset:

$$R_a(x,y) = \begin{pmatrix} 1.0 & 1.0 & 0.699 & 0.0 & 0.0 & 0.0 \\ 1.0 & 1.0 & 0.699 & 0.0 & 0.0 & 0.0 \\ 0.699 & 0.699 & 1.0 & 0.0 & 0.0 & 0.0 \\ 0.0 & 0.0 & 0.0 & 1.0 & 0.699 & 0.699 \\ 0.0 & 0.0 & 0.0 & 0.699 & 1.0 & 1.0 \\ 0.0 & 0.0 & 0.0 & 0.699 & 1.0 & 1.0 \end{pmatrix}$$

$$R_b(x,y) = \begin{pmatrix} 1.0 & 0.0 & 0.568 & 1.0 & 1.0 & 0.0 \\ 0.0 & 1.0 & 0.0 & 0.0 & 0.0 & 0.137 \\ 0.568 & 0.0 & 1.0 & 0.568 & 0.568 & 0.0 \\ 1.0 & 0.0 & 0.568 & 1.0 & 1.0 & 0.0 \\ 1.0 & 0.0 & 0.568 & 1.0 & 1.0 & 0.0 \\ 0.0 & 0.137 & 0.0 & 0.0 & 0.0 & 1.0 \end{pmatrix}$$

$$R_c(x,y) = \begin{pmatrix} 1.0 & 0.0 & 0.036 & 0.0 & 0.0 & 0.0 \\ 0.0 & 1.0 & 0.036 & 0.518 & 0.518 & 0.518 \\ 0.036 & 0.036 & 1.0 & 0.0 & 0.0 & 0.0 \\ 0.0 & 0.518 & 0.0 & 1.0 & 1.0 & 1.0 \\ 0.0 & 0.518 & 0.0 & 1.0 & 1.0 & 1.0 \\ 0.0 & 0.518 & 0.0 & 1.0 & 1.0 & 1.0 \end{pmatrix}$$

Next, the fuzzy discernibility matrix needs to be constructed on the basis of the fuzzy discernibility given in equation (18). For instances 2 and 3, the resulting fuzzy clause is $\{2^* \vee 3^* \vee a_{0.301} \vee b_{1.0} \vee c_{0.964}\} \leftarrow q_{1.0}$.

The fuzzy discernibility of instances 2 and 3 for attribute a is 0.301, indicating that the instances are partly discernible for this feature. The instances are fully discernible with respect to the decision feature, indicated by $q_{1.0}$. The set of clauses is:

$$
\begin{array}{lrcl}
C_{12}: & \{1^* \vee 2^* \vee a_{0.0} \vee b_{1.0} \vee c_{1.0}\} & \leftarrow & q_{1.0} \\
C_{13}: & \{1^* \vee 3^* \vee a_{0.301} \vee b_{0.432} \vee c_{0.964}\} & \leftarrow & q_{0.0} \\
C_{14}: & \{1^* \vee 4^* \vee a_{1.0} \vee b_{0.0} \vee c_{1.0}\} & \leftarrow & q_{1.0} \\
C_{15}: & \{1^* \vee 5^* \vee a_{1.0} \vee b_{0.0} \vee c_{1.0}\} & \leftarrow & q_{1.0} \\
C_{16}: & \{1^* \vee 6^* \vee a_{1.0} \vee b_{1.0} \vee c_{1.0}\} & \leftarrow & q_{0.0} \\
C_{23}: & \{2^* \vee 3^* \vee a_{0.301} \vee b_{1.0} \vee c_{0.964}\} & \leftarrow & q_{1.0} \\
C_{24}: & \{2^* \vee 4^* \vee a_{1.0} \vee b_{1.0} \vee c_{0.482}\} & \leftarrow & q_{0.0} \\
C_{25}: & \{2^* \vee 5^* \vee a_{1.0} \vee b_{1.0} \vee c_{0.482}\} & \leftarrow & q_{0.0} \\
C_{26}: & \{2^* \vee 6^* \vee a_{1.0} \vee b_{0.863} \vee c_{0.482}\} & \leftarrow & q_{1.0} \\
C_{34}: & \{3^* \vee 4^* \vee a_{1.0} \vee b_{0.431} \vee c_{1.0}\} & \leftarrow & q_{1.0} \\
C_{35}: & \{3^* \vee 5^* \vee a_{1.0} \vee b_{0.431} \vee c_{1.0}\} & \leftarrow & q_{1.0} \\
C_{36}: & \{3^* \vee 6^* \vee a_{1.0} \vee b_{1.0} \vee c_{1.0}\} & \leftarrow & q_{0.0} \\
C_{45}: & \{4^* \vee 5^* \vee a_{0.301} \vee b_{0.0} \vee c_{0.0}\} & \leftarrow & q_{0.0} \\
C_{46}: & \{4^* \vee 6^* \vee a_{0.301} \vee b_{1.0} \vee c_{0.0}\} & \leftarrow & q_{1.0} \\
C_{56}: & \{5^* \vee 6^* \vee a_{0.0} \vee b_{1.0} \vee c_{0.0}\} & \leftarrow & q_{1.0}
\end{array}
$$

Due to the properties of implicators, all clauses with $q_{0.0}$ may be removed without influencing the returned bireduct, hence the clause list can be reduced to:

$$
\begin{array}{lrcl}
C_{12}: & \{1^* \vee 2^* \vee a_{0.0} \vee b_{1.0} \vee c_{1.0}\} & \leftarrow & q_{1.0} \\
C_{14}: & \{1^* \vee 4^* \vee a_{1.0} \vee b_{0.0} \vee c_{1.0}\} & \leftarrow & q_{1.0} \\
C_{15}: & \{1^* \vee 5^* \vee a_{1.0} \vee b_{0.0} \vee c_{1.0}\} & \leftarrow & q_{1.0} \\
C_{16}: & \{1^* \vee 6^* \vee a_{1.0} \vee b_{1.0} \vee c_{1.0}\} & \leftarrow & q_{0.0} \\
C_{23}: & \{2^* \vee 3^* \vee a_{0.301} \vee b_{1.0} \vee c_{0.964}\} & \leftarrow & q_{1.0} \\
C_{26}: & \{2^* \vee 6^* \vee a_{1.0} \vee b_{0.863} \vee c_{0.482}\} & \leftarrow & q_{1.0} \\
C_{34}: & \{3^* \vee 4^* \vee a_{1.0} \vee b_{0.431} \vee c_{1.0}\} & \leftarrow & q_{1.0} \\
C_{35}: & \{3^* \vee 5^* \vee a_{1.0} \vee b_{0.431} \vee c_{1.0}\} & \leftarrow & q_{1.0} \\
C_{46}: & \{4^* \vee 6^* \vee a_{0.301} \vee b_{1.0} \vee c_{0.0}\} & \leftarrow & q_{1.0} \\
C_{56}: & \{5^* \vee 6^* \vee a_{0.0} \vee b_{1.0} \vee c_{0.0}\} & \leftarrow & q_{1.0}
\end{array}
$$

Having generated the set of clauses, the SFRIFS algorithm then proceeds to select the feature that occurs with the highest sum of its fuzzy discernibilities. Here, the values are calculated as $a = 6.602$, $b = 6.725$, $c = 7.446$, and so feature c is chosen. The clauses that have been maximally satisfied are then removed, leaving:

$$
\begin{array}{lrcl}
C_{23}: & \{2^* \vee 3^* \vee a_{0.301} \vee b_{1.0} \vee c_{0.964}\} & \leftarrow & q_{1.0} \\
C_{26}: & \{2^* \vee 6^* \vee a_{1.0} \vee b_{0.863} \vee c_{0.482}\} & \leftarrow & q_{1.0} \\
C_{46}: & \{4^* \vee 6^* \vee a_{0.301} \vee b_{1.0} \vee c_{0.0}\} & \leftarrow & q_{1.0} \\
C_{56}: & \{5^* \vee 6^* \vee a_{0.0} \vee b_{1.0} \vee c_{0.0}\} & \leftarrow & q_{1.0}
\end{array}
$$

As there are clauses remaining yet to be satisfied, the algorithm continues. The next stage of SFRIFS is to consider instances. It can be seen from the clause list above that instance 6 occurs most frequently, and hence this instance is selected (for removal) and the satisfied clauses are then removed:

$$
\begin{array}{lrcl}
C_{23}: & \{2^* \vee 3^* \vee a_{0.301} \vee b_{1.0} \vee c_{0.964}\} & \leftarrow & q_{1.0}
\end{array}
$$

One clause remains, and so the algorithm continues. The next step is to select a feature that has the highest sum of fuzzy discernibilities, which in this case is feature b. Having selected this feature, the final clause is satisfied and the algorithm terminates, returning the bireduct which has the constituent: feature subset $\{b, c\}$ and set of instances $\{1, 2, 3, 4, 5\}$.

IV. Experimental Evaluation

In this section the results of applying the novel simultaneous instance and feature selection method are presented. The datasets used for this evaluation are drawn from [11]. A series of experiments are carried out on this data and the results are presented which demonstrate the advantages of the proposed method. The SFRIFS method is also compared with a fuzzy-rough feature selection method [1]. The comparison with such methods helps to demonstrate that useful and valuable features are indeed retained by the feature and instance selection method described in this paper. An important note is that, it is not possible to compare the work proposed here directly with that of [5], because the fuzzy-rough extension allows the consideration of real-valued data. The work in [5], considers only crisp data which is usually obtained by discretising the real-valued data, which can result in information loss. Indeed this is one of the main motivations for this work. In addition, the framing of the feature and instance selection task as a propositional satisfiability problem means that the heuristic for the generation of bireducts is also carried out in a very different way to that of [5].

1) Experimental Setup: A total of 11 different datasets are employed for the experimental evaluation detailed in Table II. For the SFRIFS method, three different similarity relations were employed in order to investigate the effect that this has on the algorithm. These relations are termed $sim1$, $sim2$ and $sim3$ hereafter and refer to those defined earlier as eqns. (13), (14) and (15) in that respective order in section II-A. In this paper, the Łukasiewicz t-norm ($\max(x + y - 1, 0)$) and the Łukasiewicz fuzzy implicator ($\min(1 - x + y, 1)$) are adopted to implement the fuzzy connectives for both SFRIFS and the fuzzy-rough FS approaches. For FRFS, only $sim3$ was used for the comparison with SFRIFS as this is the similarity relation used in standard fuzzy-rough feature selection.

For the generation of classification results, three different classifier learners have been employed: J48 which is based on ID3 [12]; JRip, a rule-based classifier [13]; and PART, another rule-based classifier [14]. Stratified 10 $\times$ 10-fold crossvalidation was employed in generating the classification results. Finally for the comparison of SFRIFS with FRFS, a statistical significance test was performed using a paired t-test (significance value: 0.05) in order to ensure that the results obtained were statistically significant.

2) Results: The results of applying the SFIS approach is detailed in Tables III - VI. The bireduct size is represented here in terms of the number of features selected, followed by the number of instances *remaining* in the data set following reduction. The classification accuracy recorded is reported with respect to the reduced dataset in each case.

Perhaps what is most apparent from the results in Tables III- V is the extent to which the choice of fuzzy similarity relation tends to have on the sizes of the fuzzy bireducts that are discovered. Overall, and certainly in terms of the number of features selected, $sim3$ tends to be the most aggressive and results in the greatest overall reduction for all datasets.

TABLE II
BENCHMARK DATA AND UNREDUCED CLASSIFICATION ACCURACIES (%)

Dataset	Features	Instances	J48	JRip	PART
water2	39	390	83.18	82.08	83.85
water3	39	390	81.59	82.26	82.72
Cleveland	13	297	53.39	54.16	52.44
Glass	9	214	68.08	67.05	69.12
Heart	13	270	78.15	79.19	77.33
ionosphere	34	230	86.13	87.09	87.39
iris	4	150	94.80	94.40	94.27
olitos	25	120	65.75	68.83	67.00
web	2557	149	57.63	55.09	51.50
wine	13	178	93.37	92.75	92.24
wisconsin	9	699	95.01	95.69	94.69

The use of $sim1$ seems to favour a reduction in the number of instances when compared with either $sim2$ or $sim3$, and generally indicates that this is the most conservative with regard to bireduct generation. It is of note that none of the similarity relations employed results in a reduction of dimensionality for the *iris* dataset, but all return the same number of instances. However, if the classification results are examined, it can be seen that the results for each similarity relation are indeed different, demonstrating that although the number of instances selected are the same, a *different* set of instances is removed from the dataset depending on the similarity relation employed. Given the nature of the *iris* dataset, this also seems to suggest that if it is not possible to select features from a particular dataset, then the approach may just return a reduced set of instances. The results for the *glass* dataset show a similar tendency with all three similarity relations demonstrating a similar pattern. It is important to note that all of these tendencies are related to the way in which instances are compared before the construction of the clauses as detailed in section II-C.

TABLE III
RESULTS FOR SFRIFS USING $sim1$

Dataset	Bireduct		Classification Accy (%)		
	reduct size	dataset size	J48	JRip	PART
water2	19	372	85.48	88.70	83.87
water3	18	373	84.71	83.91	82.03
cleve	13	284	54.22	57.39	58.09
glass	8	205	67.31	65.36	67.31
heart	12	257	82.1	80.54	82.10
ionosphere	13	217	94.48	89.86	94.93
iris	4	146	95.20	89.72	95.89
olitos	12	108	71.29	62.04	68.51
web	27	122	59.01	47.54	60.65
wine	10	168	97.02	95.83	98.21
wisconsin	9	690	95.50	96.52	95.65

In terms of classification accuracy, there does not seem to be any clear advantage of one relation over another. However, $sim3$ does seem to have marginally lower overall accuracies particularly in the case of the *heart* and *wine* datasets. It is important to note, however, that despite this slightly reduced accuracy, the bireduct sizes are significantly smaller than those returned for the same datasets for $sim1$ and $sim2$. Indeed in some cases $sim3$ seems to do better in terms of the classification accuracies returned as well as achieving a reduction in data size, e.g. *water2* and *wisconsin*.

TABLE IV
RESULTS FOR SFRIFS USING $sim2$

Dataset	Bireduct		Classification Accy (%)		
	reduct size	dataset size	J48	JRip	PART
water2	12	379	82.84	79.41	81.53
water3	12	379	83.11	83.37	83.90
cleve	12	284	54.22	57.39	58.09
glass	8	205	69.26	66.82	69.75
heart	12	258	79.84	81.39	83.72
ionosphere	13	217	91.70	91.24	89.86
iris	4	146	94.52	95.20	94.52
olitos	9	111	68.46	69.36	61.26
web	21	128	50.78	53.12	60.65
wine	8	170	94.70	92.35	95.88
wisconsin	9	690	94.34	96.52	95.21

TABLE V
RESULTS FOR SFRIFS USING $sim3$

Dataset	Bireduct		Classification Accy (%)		
	reduct size	dataset size	J48	JRip	PART
water2	7	384	83.33	83.59	81.51
water3	6	384	83.07	82.29	81.77
cleve	7	290	55.51	55.17	55.86
glass	8	206	67.00	64.07	67.47
heart	6	264	76.89	76.13	77.65
ionosphere	6	224	88.39	87.94	89.73
iris	4	146	96.57	97.26	96.57
olitos	5	115	64.34	63.47	67.82
web	13	136	55.88	54.41	55.88
wine	5	173	91.90	88.43	92.48
wisconsin	6	693	96.53	95.95	96.39

In addition to the analysis of SFRIFS in terms of the bireducts generated and the classification accuracies of the generated models, a comparison is also made with a fuzzy-rough feature selection approach (FRFS) [1]. The reason for this comparison is to demonstrate that SFRIFS can offer a reduction in both dimensionality and the number of instances without any loss of generality. Table VI shows the results of running FRFS on the same 11 datasets used previously for evaluating SFRIFS. Also included in this table are the summary of a statistical comparison using a paired t-test with SFRIFS as the base for comparison; * indicates a result that was statistically worse than SFRIFS, - indicates no statistical difference, and v indicates a result where FRFS was statistically better. The results are compared with those of SFRIFS employing $sim3$.

In Table VI, it is clear that SFRIFS outperforms FRFS for six out of the total 11 datasets in terms of the subset size: *cleveland*, *glass*, *heart*, *ionosphere*, *web* and *wisconsin*. Indeed, as mentioned previously, whilst SFRIFS does not achieve a reduction in the dimensionality for *iris*, the number of instances is reduced meaning that some reduction is achieved for all of the datasets. In terms of the classification accuracy, there is little difference in the performance of the

TABLE VI
RESULTS FOR FRFS

Dataset	Reduct size	Classification Accy (%)			T-test
		J48	JRIP	PART	(*/-/v)
water2	7	86.41	84.05	84.61	0/3/0
water3	6	80.51	81.28	79.23	1/2/0
cleveland	8	50.84	54.54	53.19	1/2/0
glass	N/R	68.08	67.05	69.12	–
heart	7	79.25	78.57	76.29	0/3/0
ionosphere	8	85.36	88.26	86.07	1/2/0
iris	N/R	94.80	94.40	94.27	–
olitos	5	63.30	64.17	58.33	1/2/0
web	20	57.08	55.03	58.33	0/3/0
wine	5	95.50	94.38	95.50	0/2/1
wisconsin	8	96.80	95.70	94.70	0/3/0

N/R: denotes that no reduction was achieved for that particular dataset

models generated. The paired t-test only identified a single classifier result (for the *wine* dataset) out of a total of 27 which was statistically better for FRFS. The remainder were comparable or better than FRFS.

V. CONCLUSION

This paper has presented a novel method for simultaneous instance and feature selection using fuzzy-rough sets. The approach is based on the removal of instances and selection of features that appear most often in the fuzzy clauses generated from the data. Instances and features are removed alternately until all of the terms are satisfied. From the initial experimentation it can be seen that SFRIFS consistently results in some reduction of the data. The removal of features or instances does not seem to have any significant impact on classification accuracy and in some cases can simultaneously remove instances and features whilst maintaining classifier performance.

There is much potential for further development in this area. One particular aspect that would be interesting to investigate is the comparison of the results of the application of stand-alone FS and IS approaches with SFRIFS. This might provide some insight into those features and instances which are selected for each iteration of SFRIFS. The application of the approach to very large data in order to assess scalability is another aspect which may also provide an additional perspective. Throughout the paper, a frequency-based approach has been adopted as a heuristic in order to select features or instances alternately. This is rather a greedy type of solution, and the result is therefore unlikely to be optimal. Other alternatives could be investigated by adopting a more intelligent approach to selection, or even by employing a nature-inspired meta-heuristic such as particle swarm optimisation or harmony search [15]. Also, this paper has focused solely on the task of simultaneous feature and instance selection, but the underlying ideas may also be useful for areas such as dynamic feature selection, or semi-supervised learning where the data is non-static.

ACKNOWLEDGMENT

Neil Mac Parthaláin would like to acknowledge the financial support for this research through NISCHR (*National Institute for Social Care and Health Research)* Wales, Grant reference: RFS-12-37.

REFERENCES

[1] R. Jensen and Q. Shen. New Approaches to Fuzzy-Rough Feature Selection, IEEE Transactions on Fuzzy Systems, vol. 17, no. 4, pp. 824–838, 2009.

[2] R. Jensen and C. Cornelis. Fuzzy-rough instance selection. Proceedings of the 19th International Conference on Fuzzy Systems (FUZZ-IEEE '10), pp. 1776–1782, 2010.

[3] Z. Pawlak. Rough Sets: Theoretical Aspects of Reasoning About Data, Kluwer Academic Publishing, 1991.

[4] D. Dubois and H. Prade, Putting Rough Sets and Fuzzy Sets Together, Intelligent Decision Support, vol.11, pp. 203–232, 1992.

[5] D. Ślęzak and A. Janusz. Ensembles of bireducts: towards robust classification and simple representation, In Proceedings of the Third international conference on Future Generation Information Technology (FGIT'11), pp. 64–77, 2011.

[6] A.M. Radzikowska and E.E. Kerre, A comparative study of fuzzy rough sets, Fuzzy Sets and Systems, vol. 126, no. 2, pp. 137–155, 2002.

[7] D. Li, and C. Cheng. New similarity measures of intuitionistic fuzzy sets and application to pattern recognitions. Pattern Recognition Letters, vol. 23, no. 1–3, pp. 221–225, 2002.

[8] C. Cornelis, G. Hurtado Martín, R. Jensen, D. Ślęzak, Feature Selection with Fuzzy Decision Reducts, 3rd Int. Conf. on Rough Sets and Knowledge Technology (RSKT'08), pp. 284–291, 2008.

[9] S. Stawicki and S. Widz. Decision bireducts and approximate decision reducts: Comparison of two approaches to attribute subset ensemble construction, Federated Conference on Computer Science and Information Systems (FedCSIS), pp. 331–338, 2012.

[10] B. Mirkin. Mathematical Classification and Clustering, Kluwer Academic Publishers, 1996.

[11] A. Frank and A. Asuncion. UCI Machine Learning Repository [http://archive.ics.uci.edu/ml]. Irvine, CA: University of California, School of Information and Computer Science, 2010.

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

[13] W.W. Cohen. Fast effective rule induction, Proceedings of the 12th International Conference on Machine Learning, pp. 115–123, 1995.

[14] I.H. Witten and E. Frank. Generating Accurate Rule Sets Without Global Optimization, Proceedings of the 15th International Conference on Machine Learning, Morgan Kaufmann Publishers, San Francisco, 1998.

[15] R. Diao and Q. Shen. Feature selection with harmony search, IEEE Trans. Syst., Man, Cybernetics Part B, vol. 42, no.6, pp. 1509–1523, 2012.