Fuzzy-Rough Set Bireducts for Data Reduction

MacParthaláin, Neil; Jensen, Richard; Diao, Ren

Published in:
IEEE Transactions on Fuzzy Systems

DOI:
10.1109/TFUZZ.2019.2921935

Publication date:
2020

Citation for published version (APA):
MacParthaláin, N., Jensen, R., & Diao, R. (2020). Fuzzy-Rough Set Bireducts for Data Reduction. *IEEE Transactions on Fuzzy Systems*, *28*(8), 1840-1850. Article 8740984.
https://doi.org/10.1109/TFUZZ.2019.2921935

Fuzzy-rough set bireducts for data reduction

Neil Mac Parthaláin, Richard Jensen, and Ren Diao

***Abstract*—Data reduction is an important step which helps to ease the computational intractability for learning techniques when data is large. This is particularly true for the huge datasets which have become commonplace in recent times. The main problem facing both data preprocessors and learning techniques is that data is expanding both in terms of dimensionality and also in terms of the number of data instances. Approaches based on fuzzy-rough sets offer many advantages for both feature selection and classification, particularly for real-valued and noisy data; however, the majority of recent approaches tend to address the task of data reduction in terms of either dimensionality or training data size in isolation. This paper demonstrates how the notion of fuzzy-rough bireducts can be used for the simultaneous reduction of data size and dimensionality. It also shows how bireducts and therefore reduced subtables of data can be used not only as a preprocessing tool but also for the learning of compact and robust classifiers. Furthermore, the ideas can also be extended to the unsupervised domain when dealing with unlabelled data. The experimental evaluation of the various techniques demonstrate that high levels of simultaneous reduction of both dimensionality and data size can be achieved whilst maintaining robust performance.**

***Keywords*—Fuzzy-rough sets, bireducts, instance selection, feature selection**

I. Introduction

The continual archiving of information, facilitated by ever-increasing ease in doing so, has meant that vast amounts of data are drawn from ever-expanding networks of sensors, streamed content, and interconnected devices. This has led to a situation where the ratio of growth in the amounts of available data to the growth of suitable tools for data analysis is huge. Collections of data are also becoming ever-expansive; both with respect to the dimensionality (number of features), and the number of training data instances. Methods that serve to reduce data to a size that is more tractable computationally are therefore becoming increasingly necessary. Traditional tools for dealing with data dimensionality such as feature selection (FS) are only part of the solution to this problem, as the numbers of data instances also continue to grow apace. Techniques which therefore focus upon reducing the number of data instances, such as instance or object selection [10] and prototype selection, are also becoming increasingly important. It is this dual focus on both aspects of data size which has resulted in approaches which attempt to combine both dimensionality reduction and instance selection techniques [4], [14].

Great interest has developed in data mining in recent years on rough set theory (RST) [16] and its related extensions. The increase in the popularity of rough sets is largely the result of a series of desirable theoretical aspects. Indeed, the grouping of information into equivalence classes is intuitive and offers a certain universal appeal. Additionally, RST possesses other properties that are advantageous. Parameters are not needed, thus obviating any requirement for user input, which is subjective and potentially erroneous. RST also determines a representation of the data that is minimal. However, the primary obstacle for traditional rough set theory is that it can only be applied to crisp or discrete-valued data. This inability to handle real-valued and noisy data has led to the exploration of approaches which hybridise RST with other techniques. One of these hybridisations is fuzzy-rough sets [3] which offer the ability to model fuzzy uncertainty in both the conditional and decision attributes.

In the areas of both rough and fuzzy-rough sets, a considerable amount of work has been published relating to FS [8]. Indeed, much of this work focuses on the search for and discovery of *decision reducts*. Reducts are subsets of attributes that can fully characterise the knowledge present in datasets. More recently, the concept of rough set *bireducts* [20], [21] have emerged that extend further the concept of decision reducts. Bireducts draw upon the ideas that underpin bi-clustering and focus on two aspects: attribute subsets that are consistent with decision concepts *and* the instance subset where this summarisation is consistent. Bireducts thus represent a subtable of the data, which is described by a reduced set of features and a corresponding reduced set of instances. These definitions have been further extended to the fuzzy-rough set framework in order to apply bireducts to data which possess real-valued information [14]. Work such as [21] and [14] offer a solid basis for further extension of the fundamental concepts. Some initial attempts have also been made to quantify the optimality of any given bireduct using the idea of ε-bireducts [22], and heuristic search strategies [7].

This paper describes the further extension of the notion of fuzzy-rough bireducts as a general approach that can perform dimensionality reduction and data size reduction *simultaneously*. Note that most existing techniques can only perform reductions as discrete individual data preprocessing steps. The approach therefore can be framed in several ways; as a pre-processor, a means to learn robust, accurate and compact ensemble classifiers or for dealing with unsupervised data, where no decision labels are present. The structure of the rest of this paper is as follows: Section II presents the theoretical underpinnings that are at the heart of rough and fuzzy-rough sets and bireducts. The implementation of a fuzzy-rough framework for data reduction is given in Section III. Some worked examples of the proposed techniques for data reduction are also included. Section IV presents the experimental evaluation that demonstrates the effectiveness of the different techniques; and finally, Section

N. Mac Parthaláin and R. Jensen are with the Dept. of Computer Science, Aberystwyth University, Aberystwyth, Ceredigion, Wales. UK, e-mail: {ncm,rkj}@aber.ac.uk

R. Diao is with Candela (Shenzhen) Technology Innovate Co. Ltd, Shenzhen, Guangdong, China.

V concludes the paper and proposes some ideas for further development of the work.

II. Theoretical Background

A. Concepts

The notion of indiscernibility is core to rough set theory [16]. Consider: $I = (\mathbb{U}, \mathbb{S})$ - an information system, where $\mathbb{U}$ (the universe of discourse) is a non-empty set of finite instances 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 may take. For any subset $P \subseteq \mathbb{S}$, there exists an equivalence relation $IND(P) = \{(x, y) \in \mathbb{U}^2 | \forall a \in P, a(x) = a(y)\}$. The partition produced by $IND(P)$ can be denoted $\mathbb{U}/IND(P)$. If $(x, y) \in IND(P)$, then instances x and y cannot be discerned by the features contained in P. For the P-indiscernibility relation, the equivalence classes are denoted $[x]_P$. Let X be a subset of the instances in the universe. Using only the information contained in P, X can be approximated via the P-lower and P-upper approximations:

$$\underline{P}X = \{x : [x]_P \subseteq X\} \tag{1}$$

$$\overline{P}X = \{x : [x]_P \cap X \neq \emptyset\} \tag{2}$$

Based on the lower approximation, the concept of the positive region can be defined for two feature subsets P and Q:

$$POS_P(Q) = \bigcup_{X \in \mathbb{U}/Q} \underline{P}X \tag{3}$$

The positive region contains those instances in the universe that belong to the P-lower approximation of granules produced by the partition generated by Q. For decision systems, the decision feature(s) correspond to Q. A measure of dependency between sets of features is useful to determine their utility. In rough set theory, the degree of dependency is calculated using the positive region above. For P,$Q \subseteq S$, Q depends on P to degree k (with $0 \leq k \leq 1$):

$$k = \gamma_P(Q) = \frac{|POS_P(Q)|}{|\mathbb{U}|} \tag{4}$$

In order to better handle vagueness and uncertainty in data, crisp rough set theory can be extended to fuzzy-rough set theory. The crisp definitions in (1) and (2) are replaced by the fuzzy lower and upper approximations which define a fuzzy-rough set [3]. For the traditional crisp definition, instances that belong fully to the lower approximation can be said to definitely belong to a concept. However, by extending the definitions, instances can belong to the approximations with varying degrees (between 0 and 1 inclusive). This range of values facilitates better robustness and modeling in the presence of noise and uncertainty.

The basis of the work presented in this paper are the definitions of the approximations found in [18], where a fuzzy similarity relation is employed in order to approximate the fuzzy concept X:

$$\mu_{\underline{R_P}X}(x) = \inf_{y \in \mathbb{U}} \mathcal{I}(\mu_{R_P}(x, y), \mu_X(y)) \tag{5}$$

$$\mu_{\overline{R_P}X}(x) = \sup_{y \in \mathbb{U}} \mathcal{T}(\mu_{R_P}(x, y), \mu_X(y)) \tag{6}$$

In this case, $\mathcal{I}$ is a fuzzy implicator and $\mathcal{T}$ a t-norm. The feature subset P induces the fuzzy similarity relation R_P:

$$\mu_{R_P}(x, y) = \mathcal{T}_{a \in P}\{\mu_{R_a}(x, y)\} \tag{7}$$

$\mu_{R_a}(x, y)$ is the degree of similarity based on one feature a between objects x and y. This can be defined in a number of different ways, e.g.:

$$\mu_{R_a}(x, y) = 1 - \frac{|a(x) - a(y)|}{|a_{max} - a_{min}|} \tag{8}$$

$$\mu_{R_a}(x, y) = \exp(-\frac{(a(x) - a(y))^2}{2{\sigma_a}^2}) \tag{9}$$

$$\mu_{R_a}(x, y) = \max(\min(\frac{(a(y) - (a(x) - \sigma_a))}{\sigma_a}, \frac{((a(x) + \sigma_a) - a(y))}{\sigma_a}), 0) \tag{10}$$

where ${\sigma_a}^2$ denotes the variance of the feature a. It has been found that (10) is best for FS and (8) is better for classification. There are a number of other alternative fuzzy relation definitions which are applicable and these are presented in detail in [12].

The *fuzzy* positive region [8] can be defined in a similar way to the crisp approach, as:

$$\mu_{POS_P(\mathbb{D})}(x) = \sup_{X \in \mathbb{U}/\mathbb{D}} \mu_{\underline{R_P}X}(x) \tag{11}$$

One key issue in data reduction is the discovery of feature dependencies, particularly between the conditional and decision feature(s) for decision systems. For fuzzy-rough set theory, this is achieved through the extension of the crisp dependency degree and is defined:

$$\gamma'_P(\mathbb{D}) = \frac{\sum_{x \in \mathbb{U}} \mu_{POS_P(\mathbb{D})}(x)}{|\mathbb{U}|} \tag{12}$$

which is the degree of dependency of $\mathbb{D}$ upon the feature subset P. A reduct R in this framework can be defined as a subset of features that maintains the degree of dependency of the unreduced data, i.e. $\gamma'_R(\mathbb{D}) = \gamma'_{\mathbb{C}}(\mathbb{D})$, whilst not possessing any extraneous features. Using this measure, algorithms may be defined that search through the space of feature subsets to find reducts or superreducts.

B. Fuzzy Discernibility Matrices

In crisp RST-based approaches, there are two primary branches of research; dependency degree-based approaches and discernibility matrix-based approaches. The bireduct approach presented later relies on the fuzzy extension of discernibility matrices [2], [8].

1) Fuzzy Discernibility: Crisp discernibility matrices can be extended to the fuzzy case and this is implemented by using fuzzy clauses. A fuzzy discernibility matrix entry can be considered to be a fuzzy set, with features belonging to certain degrees. For a feature a, the membership degree to a clause C_{ij} can be computed as follows:

$$\mu_{C_{ij}}(a) = N(\mu_{R_a}(i, j)) \tag{13}$$

Here N stands for fuzzy negation and the fuzzy similarity of i and j is denoted $\mu_{R_a}(i,j)$. If $\mu_{C_{ij}}(a) = 1$ then i and j are fully distinct for feature a; if $\mu_{C_{ij}}(a) = 0$, then the instances are considered to be identical. When $\mu_{C_{ij}}(a) \in (0,1)$, the instances have partial discernibility. Each clause in the matrix is a set of features along with their associated memberships:

$$C_{ij} = \{a_x | a \in \mathbb{C}, x = N(\mu_{R_a}(i,j))\}\ i,j = 1,...,|\mathbb{U}| \quad (14)$$

For example, a clause C_{ij} in the matrix could be: $\{a_{0.35}, b_{0.6}, c_{0.17}, d_{0.0}\}$, representing the memberships $\mu_{C_{ij}}(a) = 0.35$, $\mu_{C_{ij}}(b) = 0.6$, etc. These memberships can be considered to indicate the significance of the features.

2) Fuzzy Discernibility Function: The fuzzy discernibility function can be constructed using the matrix entries in a manner similar to the RST-based discernibility matrix approach:

$$f_D(a_1^*, ..., a_m^*) = \wedge\{\vee C_{ij}^* | 1 \leq j < i \leq |\mathbb{U}|\} \quad (15)$$

where $C_{ij}^* = \{a_x^* | a_x \in C_{ij}\}$. This function takes values in $[0,1]$ and can be thought of as a measure of the *satisfaction* of the function for an assignment of truth values to the variables representing the features. The process for finding reducts in this context is to search through the space of truth assignments to variables to find the smallest assignment that satisfies the formula maximally. Note that the largest satisfaction degree can be calculated by setting all the variables to *true*.

3) Decision-relative Fuzzy Discernibility Matrix: When considering data with one or more decision features, the decision-relative matrix must be constructed. For the fuzzified version, this can be described:

$$f_D(a_1^*, ..., a_m^*) = \{\wedge\{\{\vee C_{ij}^*\} \leftarrow q_{N(\mu_{R_q}(i,j))}\} | 1 \leq j < i \leq |\mathbb{U}|\} \quad (16)$$

for the decision attribute q. Here, $\leftarrow$ denotes a fuzzy implicator.

C. Definition and Formulation

Given a feature subset P, the satisfaction degree for C_{ij} is defined as follows:

$$SAT_P(C_{ij}) = \mathcal{S}_{a \in P}\{\mu_{C_{ij}}(a)\} \quad (17)$$

where $\mathcal{S}$ is an s-norm. To illustrate this, consider the earlier mentioned clause $\{a_{0.35}, b_{0.6}, c_{0.17}, d_{0.0}\}$. Given $P = \{c, a\}$, the satisfaction degree is $SAT_P(C_{ij}) = \mathcal{S}\{0.17, 0.35\} = 0.52$ for the Łukasiewicz s-norm. Note that a clause can be satisfied to a certain extent depending on the truth assignments. In the crisp case, clauses are either fully satisfied or not. The maximum satisfiability of a clause can be calculated thus:

$$maxSAT_{ij} = SAT_{\mathbb{C}}(C_{ij}) = \mathcal{S}_{a \in \mathbb{C}}\{\mu_{C_{ij}}(a)\} \quad (18)$$

With this configuration, reducts are variable truth assignments that maximally satisfy each clause.

D. Bireduct definitions

The authors in [20] introduce the initial concept of a rough set bireduct, an idea similar to *approximate reducts* [21]. A bireduct and its definition focus upon feature subsets that describe the decision feature(s), with a corresponding subset of data objects for which such descriptions are valid.

In [20], the authors define decision bireducts in the following way. For a decision system $\mathbb{I} = (\mathbb{U}, \mathbb{S} \cup \{d\})$, 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 $d(i) \neq d(j)$

This definition depends on two properties: that the subset of features is minimal and the coverage of instances is maximal. This idea is central to the work proposed in this paper, as this can be framed as a satisfiability problem.

E. Fuzzy-Rough Bireducts

The work described previously laid the foundations and formalised the concepts of crisp bireducts. These concepts have since been extended and fuzzified, which led to the definition of fuzzy-rough bireducts [14]. The RST approach to discovering reducts, using rough set discernibility, generates each clause by comparing pairs of instances. Attributes will appear in clauses if they differ between instances. Therefore, at least one feature that appears in a clause must be selected in order to discern between a particular instance pair.

In the case of bireducts however, a clause can also be satisfied by removing either (or indeed both) of the data instances that resulted in the generation of that clause. The rationale for the removal of a data instance is that one of either of the pair of instances under consideration can be viewed as 'noisy' or even an outlier. The inclusion of such data instances may therefore not be useful. Hence, it could be more advantageous to remove a problematic instance rather than choosing a number of features to discriminate between this instance and the rest of the data. The previous definition of the fuzzy discernibility function can thus be extended as follows:

$$f_D(a_1^*, ..., a_m^*) = \wedge\{i^* \vee j^* \vee C_{ij}^* | 1 \leq j < i \leq |\mathbb{U}|\} \quad (19)$$

The discernibility function f_D is a boolean function of boolean variables corresponding to the membership of attributes $a_1, ..., a_m$ to a given entry of the discernibility matrix generated by instances i and j. The operator $\vee$ represents logical OR, and $\wedge$ represents logical AND. Further detail regarding fuzzy discernibility matrices can be found in [11].

A clause is satisfied if i^* or j^* are chosen (i.e. deleted from the data) or if C_{ij}^* is maximally satisfied. The goal is then to remove a set of instances Z and choose a set of features B to satisfy maximally all of the clauses, ultimately resulting in a fuzzy bireduct. This is the case as no proper subset of B will discern all instances in X, and no proper superset of X will be discernible by B.

Fuzzy-rough bireducts offer a way of generalising the concept of bireducts such that it is applicable to real-valued domains.

This offers many new possibilities for application to different tasks. Indeed in the work of [14] fuzzy-rough bireducts can be framed in the context of simultaneous feature and instance selection. Furthermore by combining ensembles of bireducts, the size and complexity of data models can be reduced considerably.

III. Data Reduction using Fuzzy-Rough Bireducts

Fuzzy-rough bireducts essentially represent subtables of the data for which the conditions described in section II-D hold. Ostensibly, the generation of fuzzy-rough bireducts appears to be of little use since the result of the process as described in [20] depends wholly on which particular feature or data instance is chosen as the starting point (and whether a feature or data object is chosen first). Also, it is noted in [21] that even small-sized datasets can contain large numbers of bireducts. The work detailed here attempts to formalise some approaches which can be used in a targeted way to generate fuzzy-rough bireducts, which are then assessed using some heuristic methods in order to perform data reduction.

A. Simultaneous Fuzzy-rough Instance and Feature Selection

In conventional discernibility matrix-based approaches, clauses are produced by comparing pairs of objects, where attributes appear only if their values for these two objects differ. Hence, in order to distinguish object pairs, one or more features must be chosen. For the fuzzy-rough bireduct formulation (as noted previously), a clause can also be satisfied by removing either (or both) of the instances that are responsible for generating it. The underlying rationale for this, is that either of the data instances may be noisy or perhaps even outliers. Therefore, it may prove more valuable to remove these instances entirely. This is the basis for the method proposed in this section that can undertake simultaneous fuzzy-rough instance and feature selection (SFRIFS).

When fuzzy discernibilty is extended to the bireduct case, as shown in equation (19) and discussed in the previous section, then the removal of either data instances or features results in the generation of fuzzy-rough bireducts. In order for this to be used for feature and instance selection, it is clear that some systematic heuristic search method is required. Here, a simple heuristic frequency-of-occurrence approach is adopted [14] although there are many alternative appropriate search mechanisms. In order to provide an understanding of how this process works, a toy example (Table I) is described below.

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

The Łukasiewicz s-norm ($\min(x+y,1)$) and the Łukasiewicz fuzzy implicator ($\min(1-x+y,1)$) are used for this example. Using the fuzzy similarity measure in (10), the relations can be computed for each feature in the dataset. For example, for feature a:

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

The matrix is then calculated using equation (13). For instances 2 and 3, the resulting fuzzy clause is: $\{2^* \vee 3^* \vee a_{0.301} \vee b_{1.0} \vee c_{0.964}\} \leftarrow q_{1.0}$. This means that the discernibility of these two instances is 0.301 for a, therefore they are considered to be somewhat discernible for this feature, and are fully discernible for the decision (hence $q_{1.0}$). Due to the properties of implicators, all entries with $q_{0.0}$ can be eliminated as this will not alter the final bireduct:

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

The next step of the algorithm is to evaluate the features via the sum of fuzzy discernibilities, choosing the feature with the highest value. These are determined to be: $a = 6.602$, $b = 6.725$, $c = 7.446$. Therefore c is chosen as it has the highest value, and the satisfied clauses are eliminated:

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

As can be seen, the list of clauses is not empty and so the algorithm will continue its execution. Instances are considered next: as instance 6 is most frequent it is removed and the corresponding clauses are eliminated:

$$C_{23}: \quad \{2^* \vee 3^* \vee a_{0.301} \vee b_{1.0} \vee c_{0.964}\} \quad \leftarrow \quad q_{1.0}$$

Now, C_{23} is the only clause remaining, so the algorithm will proceed. This time, the feature with the highest fuzzy discernibility sum will be chosen: feature b. With the satisfaction of this clause the algorithm can stop and return the bireduct consisting of features $\{b, c\}$ and instances $\{1, 2, 3, 4, 5\}$.

1) Problems with SFRIFS: The initial implementation of SFRIFS used the approach where the frequency of occurrence was used as a heuristic for the removal of both features and instances. The use of such a strategy soon highlighted a particular problem where the underlying class representation (of data instances) was unbalanced in terms of the ratio. Take, for instance, the extreme situation where the data may contain only two classes: class A may be represented by five data instances, and class B by a single data instance. Recall from section II-C

that a clause may be generated for a pair of objects when they are considered to be similar but belong to different decision classes. For this example, if the single instance of class B is similar to those instances of class A, this will result in the generation of a comparatively large number of clauses with a corresponding high frequency of the object that represents class B. This will make the selection of the minority instance certain, and the minority class B will not be present in the final reduced dataset. Thus, when it comes to applying this as a strategy for the reduction of data instances, the situation can arise where minority class instances are removed from the dataset as their pairwise comparison results in the generation of a large number of clauses where the single object of class B is present.

In order to address this problem, a modified fitness proportional selection strategy for the instance selection/removal phase is employed. The basic probability relating to the removal of an object of a particular class is defined in equation (20) and is based on the overall class distribution in the original dataset:

$$p_i = \frac{f_i}{\sum_{j=1}^{N} f_j} \tag{20}$$

However, some datasets contain relatively large numbers of classes, and varying numbers of representative instances. If a particular class contains a very small number of instances, then it is effectively ignored for the approach used here. Essentially if a single class represents more than 60% of the total data and the number of classes ≥ 3, then all other smaller classes will be ignored such that the probability of selection of instances of the larger classes increases accordingly. For the problem at hand, rather than simply searching for any clause generated as a result of any instance from a majority class, it is better to find a clause which is the result of a comparison of the selected majority class and one of the minority class instances. A clause is only satisfied if this criterion is fulfilled and the corresponding majority class instance is then removed from the dataset. This extension is denoted $\text{SFRIFS}_{\text{MCP}}$ hereafter.

2) Bireducts for Unsupervised Data: The concepts of fuzzy-rough bireducts have, up until now, been limited to supervised domains where the decision labels/concepts are known [14]. They can also, by definition, be extended to the unsupervised domain - see Equation (19). This can be achieved in a few different ways, but an approach based upon that of [15] is described below where each data instance is allowed to belong to its own decision class and thus discernibility only needs to be preserved with respect to each data instance.

B. Heuristic Search for Fuzzy-Rough Bireducts

In the previous section it was shown how fuzzy-rough bireducts form the foundation for approaches that can perform simultaneous feature and instance selection. This section describes how ensembles of fuzzy-rough bireducts (and thus subtables of data) can be used for the task of classification and data reduction. This is advantageous since it means that the full dataset is not required in order to build accurate and stable models. The difficulty of the reduction task requires effective and efficient search strategies, without exhaustively searching through all possibilities [13]. Several such approaches are based on phenomena or patterns observed in nature. For these, fitness functions play an important role as they help to guide the search to areas where better candidate solutions may be discovered.

How to evaluate what constitutes an optimal bireduct (B, Y) is not straightforward as this can be considered from alternative viewpoints, such as the number of features chosen in B, the number of instances covered, and the ratios of these. Additionally, B may lead to several bireducts as different object combinations could equally satisfy the formula in equation (19). To evaluate bireduct quality, ε-bireducts have been proposed [22]:

$$|Y| \geq (1-\varepsilon)|\mathbb{U}|, 0 \leq \varepsilon < 1 \tag{21}$$

This parameter, ε, tries to constrain bireducts so that at least a certain proportion of the original instances will be chosen. This will also have a bearing on the size of the subsets of features that are selected. Intuitively, by setting a high value for ε, the number of covered instances will be small, and the feature subset sizes will also be small. Setting ε to 0 will produce a traditional reduct as all instances will be covered.

To focus the search for ε-bireducts on smaller feature subset sizes, in the approach described below the fitness of a subset B is defined as follows:

$$\text{fitness}(B) = \begin{cases} \text{cov}(B) & , \ 1-\text{cov}(B) \leq \varepsilon \\ 2-2\varepsilon-\text{cov}(B) & , \ 1-\text{cov}(B) > \varepsilon \end{cases} \tag{22}$$

with the maximum fitness attainable being $1-\varepsilon$ and $\text{cov}(B)$ representing the coverage of instances by B:

$$\text{cov}(B) = \max_{Y \in \mathbb{Y}_B} \left(\frac{|Y|}{|\mathbb{A}|}\right) \tag{23}$$

The sizes of the subsets of features are compared, as the objective of the approach is to find feature subsets of small cardinality that cover a sufficiently large number of instances. Hence any search mechanism and fitness function must take into account this dual-objective nature of the problem.

Harmony search (HS) [6], is a technique that is inspired by music and its performance. Central to HS are the notions of *notes, musicians, harmonies*, and a *harmony memory*. These can be mapped to the feature selection problem but will not be explored here due to space constraints. However, specific detail on how this is achieved is shown in [6]. Instead, the modifications required to re-frame the original algorithm (HSFS) [6] as a method to search for optimal bireducts in a fuzzy-rough context are described below.

The harmony search-based algorithm for discovering bireducts (referred to hereafter as HSFS_{BR}) employs four parameters: 1) the harmony memory size ($|\mathbb{H}|$), 2) the number of musicians $|P|$ (each one selects a single feature), 3) the harmony memory consideration rate (denoted δ), and 4) the number of iterations ($g_{\max}$). A musician p^i has its own domain of notes $\aleph^i$ from which it selects, but can also randomly choose from the set of possible features. This is controlled by $\delta, 0 \leq \delta \leq 1$.

The proposed algorithm is given in Algorithm 1. The initial values of the parameters $|\mathbb{H}|$, $|P|$, δ, and $g_{\max}$ are set to their defaults for feature selection. The harmony memory $|\mathbb{H}|$ is

initialized randomly. For these subsets, corresponding fuzzy-rough bireducts are produced and evaluated via Equations (22) and (23). Each selector has a note domain $\aleph$ of $|\mathbb{H}|$ features, which may include identical or empty choices.

If a randomly generated number r_δ is less than δ, each p^j in P nominates a random feature from the full feature set. Otherwise, a note is chosen from its own note domain $\aleph^j$. This generates a new harmony $H^{'}$ and a corresponding new subset of features, $B_{H^{'}}$. If the new subset obtains a higher score than the weakest subset in the memory, or if it has an equal evaluation but is of a smaller size, then the new subset replaces the worst one. This process of improvization and updating repeats until the maximum number of iterations is reached. Finally, the best harmony in the harmony memory $\dot{H} = \arg\max_{H\in\mathbb{H}} \text{fitness}(B_H)$ is determined, and its associated ε-bireduct $(B_{\dot{H}}, Y)$ is returned as the final search output.

Algorithm 1: The HSFS_{BR} Algorithm

```
p^i ∈ P, with |P| musicians;
H^j ∈ ℍ, j = 1 to |ℍ|;
ℵ_i = ⋃_{j=1}^{|ℍ|} H_i^j;
δ;
ℂ, fuzzy clauses;
for g = 1 to g_max do
    H' = ∅;
    for i = 1 to |P| do
        if r_δ < δ then
            a_r = randomFeature(𝔸);
            H' = H' ∪ {a_r};
        else
            r = randomInteger(|ℍ|);
            H' = H' ∪ {ℵ_ir};
    for ∀C_ij ∈ ℂ do
        if SAT_{B_H'}(C_ij) = SAT_max(C_ij) then
            ℂ = ℂ − C_ij
    Determine O (outliers) satisfying the remaining clauses;
    Construct bireduct (B_H', 𝕌 − O);
    if fitness(H') ≥ min_{H∈ℍ} fitness(H) then
        ℍ = ℍ ∪ {H'};
        ℍ = ℍ − {arg min_{H∈ℍ} fitness(H)}
return best ε-bireduct;
```

Note that both `randomFeature` and `randomInteger` functions are used in the decision to choose the next feature; a musician p^j may choose to add a random feature from the full set of features, or only from its own note domain.

1) Walkthrough: The previous dataset (Table I) is once again used for the illustration of the main aspects of HSFS_{BR}. In the interests of brevity, the construction of the fuzzy similarity matrices $R_a(x, y), x, y \in \mathbb{U}, a \in \mathbb{A}$, and lists of clauses are not repeated since they are the same as those shown in the previous section. For HSFS_{BR}, the number of musicians (single-feature selectors) is 3. The harmony memory is generated randomly at first, populating the note domains $\aleph_i$ of each musician p^i with a random selection of features. For this example, a new harmony is improvised $H^{'}$ which could be $\{c, c, -\}$. This represents the feature subset $B_{H^{'}} = \{c\}$. Having generated this, the satisfied clauses are removed (i.e. those clauses that are maximally satisfied by setting only $c^* = \texttt{true}$). This leaves the following set of clauses:

$$\begin{array}{lrcl} C_{23}: & \{2^* \vee 3^* \vee a_{0.301} \vee b_{1.0} \vee c_{0.964}\} & \leftarrow & q_{1.0} \\ C_{26}: & \{2^* \vee 6^* \vee a_{1.0} \vee b_{0.863} \vee c_{0.483}\} & \leftarrow & q_{1.0} \\ C_{46}: & \{4^* \vee 6^* \vee a_{0.301} \vee b_{0.301} \vee c_{0.0}\} & \leftarrow & q_{1.0} \\ C_{56}: & \{5^* \vee 6^* \vee a_{0.0} \vee b_{1.0} \vee c_{0.0}\} & \leftarrow & q_{1.0} \end{array}$$

The maximum coverage for this subset $\{c\}$ is therefore $\{x_1, x_3, x_4, x_5\}$ or $\{x_1, x_2, x_4, x_5\}$, if the instances $\{2, 6\}$ or $\{3, 6\}$ are outliers and $cov(B_{H^{'}}) = \frac{4}{6} = 0.66$. If ε is set to 0.4, then $H^{'}$ will evaluate to $2 - 2 \times 0.4 - 0.66 = 0.54$. This will replace the current weakest harmony if its fitness is better or if the fitness is equivalent but the corresponding subset size is smaller. This continues until $g_{\max}$ iterations have been reached.

It can be seen that the worst case complexity of the algorithm is $O(g_{\max}(|P| + |\mathbb{C}| + |\mathbb{H}|))$. In each iteration of the algorithm, the $|P|$ musicians generate notes, forming subsets of features. Then, each clause in $\mathbb{C}$ is considered for potential removal if satisfied. Determining outliers (i.e., instance removal) and constructing the bireduct are linear processes. Finally, the fitnesses of the harmonies in $\mathbb{H}$ are calculated.

2) Ensembles of Classifiers using ε-Bireducts: Stochastic search algorithms may discover many good quality subsets $\mathbb{B}$ for large datasets. Any one of these subsets $B \in \mathbb{B}$ can be used in training classifiers. Therefore, an ensemble of classifiers can be constructed [26] which may well attain a higher classification performance due to the diversity of the internal models [24]. The approach presented here investigates the utility of ensembles constructed from ε-bireducts via fuzzy-rough sets. Bireducts are particularly useful for this as the instances in Y are known to be consistent with the set of features in B. These instances should be the most relevant ones for constructing models with the feature subset B and can be adjusted via ε, controlling the respective 'footprints' of the resultant models.

The overall process of learning classifier ensembles from ε-bireducts is illustrated in Fig. 1. To develop a classifier ensemble $\mathbb{E} = \{E^l \mid l = 1, \cdots, |\mathbb{E}|\}$, a number of bireducts $\{(B^l, Y^l) | l = 1, \cdots, |\mathbb{E}|\}$ must first be determined. Note that in the diagram, each subsystem is a single ε-bireduct classifier E^l, $l \in \{1, \cdots, |\mathbb{E}|\}$. The execution process for this approach is similar to traditional ensemble methods for FS classifier learning, e.g. [26] and [21]. Hence, for the sake of brevity, further explanation and detail is not included. Aggregation of the results is achieved via majority vote [23], but other aggregators can be used.

IV. Experimentation

An experimental evaluation is detailed here in order to show how fuzzy-rough bireducts can be employed in different ways for data reduction, using the methods described in the previous sections. Here, the results are presented along with a

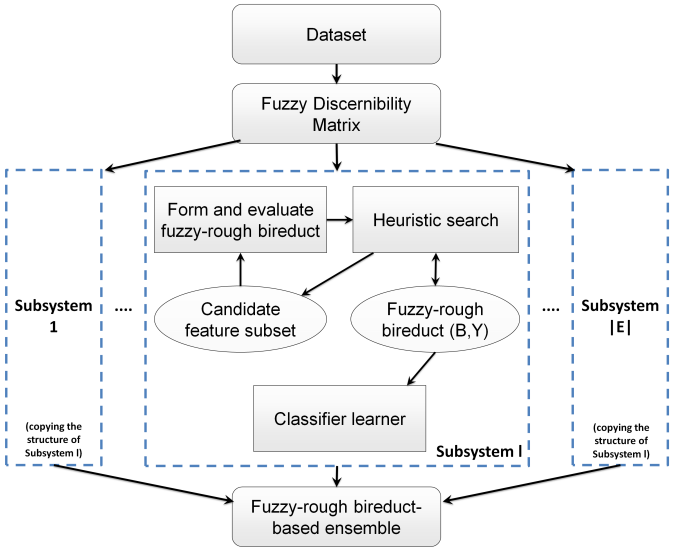

Fig. 1. Overview of learning classifier ensembles via ε-bireducts.

discussion of how such methods can offer useful and compact representations.

A. Experimental Setup

Four sets of experiments are included here representing the respective approaches and variants described earlier. In total, 14 benchmark datasets drawn from [5] are employed and summarised in Table II. The respective unreduced classification accuracies generated using 10 $\times$ 10-fold cross-validation are also shown. For SFRIFS, three different similarity relations (as defined in equations (8), (9), and (10)) are employed in order to assess their effects on the simultaneous selection of features and instances. Whilst for brevity in the later comparison with FRFS, only $sim3$ is used for $\text{SFRIFS}_{\text{MCP}}$ and unsupervised SFRIFS.

The parameter settings for HSFS_{BR} are: $|\mathbb{H}| = 10$, $|P| = |\mathbb{A}|$, and $g_{\max} = 2000$. In addition, three different values for ε are used: 0.1, 0.2, 0.3 to generate different levels of covering for the ε-bireduct. The ensemble size is set to 10 (i.e. $|\mathbb{E}| = 10$) since this provided a good trade-off between relatively fast execution and good quality results. The classifier learners adopted include: J48 [17], JRip [1], PART [25], and VQNN [9]. The use of these classifiers enables a diversity in model construction, which should enable a more complete knowledge of the quality of the discovered bireducts and therefore the resulting performance.

The results are validated using stratified 10-fold cross-validation. This is to ensure that the decision classes have an appropriate number of instances to learn from. For HSFS_{BR}, the ε-bireducts are constructed in each training fold and the resulting reduced data used for training the classifiers for each fold.

B. Results: SFRIFS

The SFRIFS experimental results are illustrated in Tables III–V. The average sizes of the feature subsets and instances

TABLE II. DATASETS

Dataset	No. of feats.	No. of insts.	classes	Accuracy (unreduced) (%) J48	JRIP	PART	VQNN
cleveland	14	297	5	53.39	54.22	52.44	59.21
ecoli	8	336	8	82.83	81.65	81.79	85.92
glass	9	214	6	68.08	67.05	69.12	72.73
heart	13	270	2	78.15	79.19	77.33	81.93
ionosphere	35	230	2	86.13	87.09	87.39	87.96
libras	91	360	15	69.31	54.53	68.14	78.89
olitos	26	120	4	65.75	68.83	67.00	80.83
sonar	61	208	2	73.61	73.40	77.40	83.36
water2	39	390	2	83.18	82.08	83.85	85.31
water3	39	390	3	81.59	82.26	82.72	82.82
web	2557	149	5	57.63	55.09	51.50	38.71
wine	14	178	3	93.97	92.75	92.24	96.51
wisconsin	10	683	2	95.44	96.08	95.68	96.66
vehicle	19	846	4	72.28	68.31	72.21	72.15

selected for the bireducts are given first. The accuracies for the classifiers are presented next. Clearly the choice of similarity measure has a large impact on bireduct size, particularly with respect to the number of features chosen. It can be seen that $sim3$ tends to produce smaller subset sizes. There is less of a difference for the number of instances selected, though $sim1$ generally removes slightly more instances.

TABLE III. SFRIFS RESULTS FOR $sim1$

Dataset	Bireduct size subset	insts.	Accuracy (%) J48	JRip	PART	VQNN
cleveland	12.98	255.30	53.57	53.97	53.44	58.25
ecoli	6.00	297.02	82.57	80.88	82.42	85.84
glass	9.00	184.60	67.85	65.20	67.31	67.55
heart	12.72	231.24	77.00	76.07	79.48	83.44
ionosphere	11.99	195.54	87.35	87.78	87.65	87.48
libras	17.69	305.87	64.24	49.71	66.07	65.70
olitos	11.01	97.53	63.25	63.33	64.08	75.00
sonar	11.61	176.26	72.72	71.69	71.54	76.98
water2	16.95	334.47	83.72	83.56	83.26	83.72
water3	16.22	335.26	81.92	82.69	82.23	83.36
web	25.67	109.05	46.16	46.30	45.27	39.20
wine	10.08	150.73	92.86	90.96	92.24	97.06
wisconsin	9.00	621.10	94.49	95.61	95.11	95.72
vehicle	18.00	744.40	72.53	67.76	71.90	71.05

When considering the results for the classification accuracy, no single similarity relation seems to offer any clear advantage over the others. It would seem that $sim3$ does have lower overall accuracies (albeit marginally). This seems to be particularly true for both the *heart* and the *wine* dataset. Despite this however, it should be emphasised that the size of the bireducts are much smaller than the bireducts obtained for these data for similarity relations $sim1$ and $sim2$.

Overall, $sim1$ and $sim2$ favor instance reduction over feature selection for the discovered bireducts. The average reduction for the number of instances for example, is in the range: 9-26.8% for $sim1$, 9-22.6% for $sim2$ but only 8.7-13% for $sim3$. Those figures are contrasted sharply however by the level of dimensionality reduction that is achieved when using $sim3$ with a mean reduction of 66% whereas $sim1$ and $sim2$ only offer average reductions of 42% and 47%, respectively.

Overall, $sim3$ offers the best average dataset reduction (39%) with $sim1$ and $sim2$ demonstrating similar performance: 28-31%. As mentioned previously, $sim1$ and $sim2$ offer marginally better classifier performance. However, for some of the datasets,

$sim3$ produces bireducts that offer higher accuracy along with a greater reduction both for the number of features and instances; for example in the case of the *water2* and *wisconsin* datasets.

TABLE IV. SFRIFS RESULTS FOR $sim2$

Dataset	Bireduct size		Accuracy (%)			
	subset	insts.	J48	JRip	PART	VQNN
cleveland	11.35	256.72	55.33	54.36	53.11	57.37
ecoli	6.00	296.95	82.20	81.93	83.78	86.46
glass	9.00	184.60	68.19	66.90	68.91	67.44
heart	9.88	233.58	80.89	76.48	79.96	83.30
ionosphere	12.05	195.36	87.17	86.30	88.52	87.30
libras	18.44	305.07	63.17	49.83	64.26	63.67
olitos	8.74	99.89	61.17	63.50	61.25	72.42
sonar	10.02	177.95	73.35	72.02	72.25	75.26
water2	10.62	340.95	81.87	82.56	82.77	83.90
water3	10.70	340.91	80.95	82.79	80.33	83.59
web	19.47	115.30	46.85	48.82	47.93	42.48
wine	7.92	152.96	90.91	89.81	91.20	96.94
wisconsin	9.00	621.10	94.64	95.34	95.34	95.69
vehicle	18.00	744.40	73.05	68.17	72.31	71.31

TABLE V. SFRIFS RESULTS FOR $sim3$

Dataset	Bireduct size		Accuracy (%)			
	subset	insts.	J48	JRip	PART	VQNN
cleveland	6.98	261.22	52.91	54.19	52.62	58.34
ecoli	5.91	297.40	82.38	81.49	83.52	86.52
glass	8.15	185.16	67.51	64.96	68.20	67.70
heart	6.44	237.09	72.37	72.19	73.52	76.67
ionosphere	6.02	201.59	85.43	84.74	85.26	87.00
libras	6.91	316.76	56.18	43.94	57.81	62.27
olitos	4.97	103.98	60.83	60.00	60.58	67.42
sonar	5.12	182.85	72.51	73.01	71.29	75.48
water2	5.91	345.99	81.54	81.72	81.51	83.79
water3	5.88	346.02	80.85	81.10	81.05	82.69
web	12.60	122.11	45.01	46.62	44.25	38.35
wine	4.26	156.48	92.42	90.34	92.20	94.72
wisconsin	6.13	623.45	94.67	95.35	94.89	95.47
vehicle	7.99	754.08	66.67	60.86	65.92	65.41

There are no current existing approaches for simultaneous feature and instance selection, and therefore it becomes difficult to provide a comparative analysis of SFRIFS. However, the series of experiments presented here attempts to address this by offering a comparison with the state-of-the-art stand alone feature selection and instance selection approaches. The evaluation has three different parts: 1) In the first an FS step is performed initially, using FRFS [8] with a hillclimbing search. The reduced dimensionality dataset is then passed to an instance selector, FRIS [10]. Finally, the reduced datasets (dimensionality and data size) are then passed to the same classifier learners that have been used previously to assess the performance of SFRIFS and compared statistically using a paired t-test. This is termed FS-IS hereafter. 2) A further set of experiments is also conducted where FRIS is employed initially before being passed to FRFS (again using a hillclimbing search), and then finally to the classifier learners. This is termed IS-FS hereafter. The results of these experiments can be seen in Tables VI and VII. 3) Finally, the execution time per-fold for each of these methods is also examined and compared with SFRIFS in Table VIII.

Tables VI-VII show a number of different trends. The first is that all of the models built by SFRIFS are either statistically comparable or better than those of either of the FS-IS or IS-FS results. This is independent of the similarity relation employed by SFRIFS, however *sim3* does offer the best reductions whilst also outperforming both FS-IS and IF-FS.

Interestingly, both the FS-IS and IS-FS methods perform poorly on the *ecoli*, *glass*, and *vehicle* datasets - the reductions are better than SFRIFS (hence the lower execution times for IS-FS) but the models generated as a result are not sufficient. Also, for IS-FS, there is no reduction in data size for six of the 14 datasets, i.e. no instance selection takes place (highlighted in bold typeface in Table VII). Whilst SFRIFS may not always return the smallest number of instances, the models generated and the reductions obtained are consistent and outperform those of FS-IS and IS-FS.

Of particular note is the execution time of SFRIFS. It is obvious that the execution times of FS-IS and IF-FS will simply be the aggregation of their respective complexities (which is considerably more complex than SFRIFS). Indeed, this is borne out in the results shown in Table VIII where it is clear that SFRIFS outperforms both methods. There are two particularly interesting results in Table VIII for *libras* and *web*. Both of these datasets have relatively high feature:instance ratios when compared to the other datasets. The results returned for these datasets suggest that SFRIFS is particularly effective in dealing with such situations and can do so effectively and much more quickly.

In summary, SFRIFS will return more consistent reductions that generalize better and have lower execution times when compared to the trivial combination of either standalone FS or IS methods.

TABLE VI. FS-IS: DISCRETE FEATURE SELECTION FOLLOWED BY INSTANCE SELECTION

Dataset	Average no. of		Accuracy (reduced) (%)			
	feats.	insts.	J48	JRIP	PART	VQNN
cleveland	7.62	131.88	51.60	55.77	52.01	57.54
ecoli	6.00	61.07	59.43*	64.31*	59.01*	73.00*
glass	9.00	80.72	46.94*	37.29*	44.36*	47.47*
heart	7.07	134.42	73.41	74.41	74.15	78.33
ionosphere	6.99	105.75	74.26	68.74	75.13*	69.57*
libras	7.73	121.94	29.33*	20.08*	29.06*	30.17*
olitos	5.00	35.71	55.50	55.33	53.67	57.00
sonar	5.24	54.35	61.48*	60.73*	61.87	62.59*
water2	5.99	204.65	82.10	81.72	82.36	81.85
water3	6.00	169.99	79.87	80.67	79.10	82.00
web	19.08	92.15	50.96	48.20	50.52	45.63
wine	5.00	114.11	85.73*	87.07	85.68*	97.35
wisconsin	6.75	361.36	92.97	92.98*	92.01	96.79
vehicle	8.45	155.17	52.35*	41.58*	52.11*	46.65*

* indicates statistically inferior result to that of SFRIFS for the same learner

To avoid situations described previously in section III-A1, where SFRIFS may remove instances of a minority class (due to the fact that they can result in the generation of a larger number of clauses), a series of experiments was also conducted. Two additional datasets which have poorly represented minority classes are also included for this evaluation, and are listed in Table IX. These particular additional datasets were chosen because they had a relatively large number of decision classes (*anneal*:6, *arrhythmia*:16) when compared with the other data. Some of the datasets have distributions of classes such that some decision classes are only represented by two data instances.

TABLE VII. IF-FS: DISCRETE INSTANCE SELECTION FOLLOWED BY FEATURE SELECTION

Dataset	Average no. of feats.	Average no. of insts.	Accuracy (reduced) (%) J48	JRIP	PART	VQNN
cleveland	7.00	268.01	53.24	53.83	50.70	57.23
ecoli	3.85	64.43	51.84*	59.47*	51.65*	62.65*
glass	5.30	80.72	38.24*	30.01*	35.89*	36.56*
heart	6.49	253.29	72.22	72.89	71.22	77.67
ionosphere	6.40	221.34	86.70	86.09	86.35	87.96
libras	6.35	357.50	57.67	42.67	56.47	62.47
olitos	5.00	**120.00**	62.92	62.75	61.75	67.17
sonar	5.24	**208.00**	70.85	71.63	70.03	75.84
water2	5.99	**390.00**	83.79	84.23	84.44	85.64
water3	6.00	**390.00**	81.05	81.90	80.36	83.54
web	19.08	**149.00**	51.20	49.87	52.45	45.57
wine	5.00	**178.00**	94.71	93.25	93.53	96.67
wisconsin	5.32	448.96	94.41	95.49	95.43	96.76
vehicle	6.98	478.98	62.33*	58.80*	62.16*	61.70*

* indicates statistically inferior result to that of SFRIFS for the same learner
bold indicates that instance selection did not perform any reduction for this dataset

TABLE VIII. TIME TAKEN PER TRAINING FOLD

Dataset	Average execution time (s) SFRIFS	FS-IS	IS-FS
cleveland	0.225	0.385	0.295
ecoli	0.215	0.185	0.034
glass	0.125	0.125	0.020
heart	0.1675	0.295	0.254
ionosphere	0.420	0.723	0.635
libras	2.625	5.455	4.5375
olitos	0.095	0.124	0.117
sonar	0.600	0.865	0.861
water2	1.42	2.15	2.032
water3	1.45	2.11	2.067
web	6.982	84.91	86.28
wine	0.100	0.112	0.118
wisconsin	0.780	1.176	0.7935
vehicle	3.177	5.338	5.385

The preservation of such classes therefore becomes even more important. In order to assess the impact of SFRIFS on minority class instances, a metric based on the mean of the class distributions and their contribution to the overall score is proposed. This is termed *Class Distribution Distortion* (CDD) and is defined as:

$$CDD = \frac{1}{|C|}\sum_{i=1}^{|C|}\left(\frac{n_i^*}{n_i}\right) \tag{24}$$

where $|C|$, is the number of decision classes in the dataset, n_i represents the number of data instances of class i, and n_i^* represents the number of instances in class i for the SFRIFS reduced dataset. Higher values of CDD indicate lower levels of overall distortion of the class distribution. Clearly, decision classes which have fewer representative data instances have a greater impact on this score.

It can be seen that when the CDD scores of the reductions of both approaches are compared, $\text{SFRIFS}_{\text{MCP}}$ is able to achieve higher scores than SFRIFS. Note that scores are lower, or indeed zero, when the class representations in the data are closer to being equal (e.g. *ionosphere*, *wine* and *vehicle*). Also, when the number of instances in the data for the $\text{SFRIFS}_{\text{MCP}}$ reduced data are examined and compared with those of SFRIFS in Table X, it can be seen that there is no significant difference indicating

TABLE IX. CLASS DISTRIBUTION DISTORTION

Dataset	SFRIFS	$\text{SFRIFS}_{\text{MCP}}$	+/-
arrhythmia	0.8967	0.9935	**0.0968**
anneal	0.9538	0.9968	**0.0431**
cleveland	0.9483	0.9880	**0.0397**
ecoli	0.8685	0.9925	**0.1240**
glass	0.9454	0.9803	**0.0349**
heart	0.9741	0.9758	**0.0017**
ionosphere	0.9738	0.9740	**0.0002**
libras	0.9821	0.9843	**0.0022**
olitos	0.9243	0.9248	**0.0005**
sonar	0.9800	0.9800	0.00
water2	0.9727	0.9871	**0.0144**
water3	0.9487	0.9894	**0.0407**
web	0.8630	0.8822	**0.0192**
wine	0.9793	0.9793	0.00
wisconsin	0.9884	0.9903	**0.0019**
vehicle	0.9916	0.9919	**0.0003**

that $\text{SFRIFS}_{\text{MCP}}$ does not negatively impact on the ability to reduce the data size. In fact, in some cases, $\text{SFRIFS}_{\text{MCP}}$ results in a slightly better reduction.

TABLE X. EXAMPLE OF TOTAL OVERALL REDUCTION IN DATA SIZE: % (USING SIM3)

Dataset	SFRIFS	$\text{SFRIFS}_{\text{MCP}}$
arrhythmia	3.95	4.06
anneal	37.98	37.98
cleveland	3.86	3.86
ecoli	1.74	1.74
glass	3.57	4.02
heart	4.58	4.58
ionosphere	12.83	12.45
libras	19.96	19.51
olitos	15.86	15.86
sonar	21.93	21.93
water2	8.62	8.62
water3	8.62	8.62
web	94.42	94.31
wine	5.76	5.76
wisconsin	1.15	1.15
vehicle	1.97	1.97

C. Results: $HSFS_{BR}$

Tables XI to XIII provide the results for the classifier ensembles, built using both J48 and VQNN, for $\varepsilon = 0.1, 0.2, 0.3$. A value of 0.2 for this parameter means that there must be 80% coverage of the training data for any generated bireduct. To confirm that the algorithm has indeed found such coverage, the final column in these tables gives the average number of instances covered, AVG($\frac{|Y|}{|\mathbb{U}|}$). Due to the extra runtime incurred for the ensemble approach, not all datasets are considered in this section. For J48, it can be seen that the performance is improved for most of the datasets compared to the original data. For VQNN, an improvement can be seen for the *cleveland*, *heart*, and *wine* datasets. With higher values of parameter ε, the average subset size decreases with little improvement of classifier accuracy compared to the original data. This can be expected as the data have been significantly reduced, meaning that the subtables of data are much smaller.

It should be noted that the base classifiers have only been exposed to instances chosen by the discovered ε-bireducts with the dimensionality significantly reduced also. Hence,

TABLE XI. CLASSIFIER ENSEMBLE RESULTS WITH $\varepsilon = 0.1$ (90% INTENDED OBJECT COVERAGE)

Dataset	J48 accuracy (%)			VQNN accuracy (%)			Bireduct coverage (%)	
	Ensemble	AVG Base	Unreduced	Ensemble	AVG Base	Unreduced	AVG($\frac{\|B\|}{\|C\|}$)	AVG($\frac{\|Y\|}{\|U\|}$)
cleveland	52.25	52.43	53.85	**56.98**	53.85	52.22	45.57	89.90
ecoli	79.73	79.91	80.61	84.52	84.75	84.48	63.75	90.58
glass	**68.25**	68.16	67.79	64.90	63.19	64.87	94.33	90.29
heart	**81.48**	76.33	74.81	**78.89**	75.41	75.93	47.08	90.12
ionosphere	**89.13**	83.17	81.30	80.87	76.43	82.61	14.29	89.86
libras	**70.56**	48.42	64.72	63.61	49.44	65.28	6.91	90.12
sonar	**76.45**	65.43	74.57	74.95	67.07	76.00	9.33	89.85
water3	**82.05**	78.18	81.28	78.97	78.33	81.79	14.49	90.03
wine	**94.31**	84.62	93.73	**93.86**	87.53	93.20	30.36	90.01

TABLE XII. CLASSIFIER ENSEMBLE RESULTS WITH $\varepsilon = 0.2$ (80% INTENDED OBJECT COVERAGE)

Dataset	J48 accuracy (%)			VQNN accuracy (%)			Bireduct coverage (%)	
	Ensemble	AVG Base	Unreduced	Ensemble	AVG Base	Unreduced	AVG($\frac{\|B\|}{\|C\|}$)	AVG($\frac{\|Y\|}{\|U\|}$)
cleveland	**55.57**	52.54	53.85	**56.23**	52.06	52.21	44.36	80.17
ecoli	77.65	77.97	80.62	81.53	81.41	84.48	64.38	80.09
glass	**67.45**	65.32	65.43	63.18	62.76	68.23	69.44	80.27
heart	74.82	73.37	74.82	75.93	70.37	75.93	32.54	80.25
ionosphere	**89.13**	79.65	81.30	79.13	72.83	83.04	11.57	80.19
libras	**71.67**	43.69	64.72	61.94	43.36	64.72	6.62	79.94
sonar	73.14	62.80	74.57	73.07	64.13	75.60	6.56	80.23
water3	77.69	76.15	81.28	74.87	76.10	81.54	11.31	80.06
wine	88.17	78.53	93.73	85.29	78.68	93.20	27.36	80.02

TABLE XIII. CLASSIFIER ENSEMBLE RESULTS WITH $\varepsilon = 0.3$ (70% INTENDED OBJECT COVERAGE)

Dataset	J48 accuracy (%)			VQNN accuracy (%)			Bireduct coverage (%)	
	Ensemble	AVG Base	Unreduced	Ensemble	AVG Base	Unreduced	AVG($\frac{\|B\|}{\|C\|}$)	AVG($\frac{\|Y\|}{\|U\|}$)
cleveland	**57.25**	54.17	50.59	**57.61**	54.15	54.94	37.50	70.07
ecoli	72.26	71.54	80.62	74.06	73.47	84.48	50.75	70.45
glass	57.49	56.95	65.82	55.63	56.42	65.48	57.78	70.09
heart	67.41	68.15	77.78	66.67	65.07	75.56	26.92	70.03
ionosphere	76.52	72.48	85.65	73.04	67.04	83.04	9.37	70.05
libras	62.50	41.50	70.83	58.61	40.58	67.78	5.80	70.06
sonar	**77.81**	66.33	73.12	75.45	66.14	76.93	6.80	70.09
water3	76.38	73.33	83.09	78.17	76.00	81.54	11.84	70.12
wine	80.35	75.42	93.73	74.15	73.70	93.20	25.50	70.06

the performance of the individual classifiers seems to be poorer than those for the original datasets. In most of the cases, the accuracies can be significantly improved through the combination of weaker classifiers. The results for the *libras* dataset are worth further consideration. This dataset has a larger number of features (91) and 15 decision classes. The combined performance of the base classifiers, which have only seen a small proportion of the data (less than 7%), is better than that obtained with J48 for the original full dataset. This is a strong indicator that the algorithm is effective and that the discovered bireducts provide enough diversity.

As mentioned previously in section III-A1, there is a potential problem with maintaining representative instances from minority classes and this appears to be overcome by an ensemble approach. Also, not only does the parameter ε affect object coverage but it also affects the resulting feature subset sizes. This is worth noting as reductions that have been made by a bireduct result in a subtable of the unreduced dataset. The results would indicate that ensembles of these subtables can improve the performance compared to the use of the full data and support the theoretical assumptions made in section III-B.

D. Results: SFRIFS for unsupervised data

Table XIV shows the results obtained by applying SFRIFS to the datasets outlined previously. Essentially, the decision feature is ignored and discernibility is preserved by removing instances and features which have no effect on the overall discernibility. The results show that for unsupervised data SFRIFS is able to achieve good reductions. For the majority of the results, the number of selected features is greater than the equivalent number of features for the supervised approach. A slight increase in the number of selected data instances is also evident for some datasets. This can be expected as the decision class encodes much discriminatory information.

V. CONCLUSION

This paper has demonstrated a number of different ways in which fuzzy-rough bireducts can be used for the reduction of

TABLE XIV. UNSUPERVISED SFRIFS RESULTS USING $sim3$

Dataset	Bireduct size		Accuracy (%)			
	subset	insts.	J48	JRip	PART	VQNN
cleveland	8.88	258.94	53.02	53.88	52.72	57.49
ecoli	6.00	297.40	82.86	81.08	82.74	86.97
glass	9.00	184.60	68.68	66.63	69.77	68.77
heart	8.51	235.00	75.33	73.93	75.67	80.41
ionosphere	13.50	193.96	86.52	82.96	86.65	88.26
libras	10.96	312.65	61.06	45.59	59.35	64.93
olitos	5.40	103.23	62.25	61.67	62.83	68.75
sonar	6.40	181.46	70.88	71.55	68.66	72.72
water2	7.04	344.73	82.54	81.95	81.26	84.49
water3	7.00	344.86	76.18	77.41	76.28	80.10
web	14.58	120.11	43.57	45.79	43.52	41.70
wine	6.63	154.24	89.62	87.66	89.61	93.37
wisconsin	9.00	621.10	94.88	95.48	95.08	96.20
vehicle	10.33	751.43	67.09	65.79	65.50	68.49

data, both in terms of features selected and numbers of objects. These reductions show that the data can be reduced whilst still retaining the useful predictive aspects, as reflected in the performance. Furthermore by combining ensembles of fuzzy-rough bireducts, classifiers which are compact and robust can be built. The introduction of ε-bireducts is important for the discovery of the best solutions because this allows the partial quantification of the balance between instance and feature reduction. Harmony search was employed for the purpose of generating several bireducts of similar quality but different enough to make their use in ensemble classification constructive. An ensemble approach also helps to reduce the impact of bireducts that are too restrictive, producing a system with higher accuracy and greater robustness.

There are several ways in which the present work could be improved. The time and space complexity when generating the list of clauses is prohibitive when applying the proposed approach to very large datasets. Some optimization can be achieved by applying grouping or absorption [11], but alternative and more efficient approaches to the representation and generation of clauses may be desirable. Also, a better approach to handling class imbalance (such as [19]) would be beneficial.

A further investigation would be to assess the stability and diversity of ensembles, as the performance may be improved somewhat by the removal of unhelpful members. The identification of such redundant members could potentially be achieved through an automated approach based on feature selection that could find the best subset of classifiers from the ensemble.

REFERENCES

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

[2] C. Cornelis, R. Jensen, G. Hurtado Martín, D. Ślęzak, "Attribute Selection with Fuzzy Decision Reducts", *Information Sciences*, vol. 180, no. 2, pp. 209–224, 2010.

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

[4] D. Fragoudis, D. Meretakis, and S. Likothanassis, "Integrating feature and instance selection for text classification", *Proceedings of the eighth ACM SIGKDD international conference on Knowledge discovery and data mining*, pp. 501–506, ACM, 2002.

[5] 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.

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

[7] R. Diao, N. Mac Parthaláin, R. Jensen, and Q. Shen, "Heuristic search for fuzzy-rough bireducts and its use in classifier ensembles", In *Proceedings of 23rd IEEE International Conference on Fuzzy Systems (FUZZ-IEEE '14)*, pp. 1504–1511, 2014.

[8] 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.

[9] R. Jensen and C. Cornelis, "Fuzzy-rough nearest neighbour classification and prediction", *Theoretical Computer Science*, vol. 412, no. 42, pp. 5871–5884, 2011.

[10] 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.

[11] R. Jensen, A. Tuson, and Q. Shen, "Finding Rough and Fuzzy-Rough Set Reducts with SAT", *Information Sciences*, vol. 255, pp. 100–120, 2014.

[12] 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.

[13] H. Liu and H. Motoda, "Feature selection for knowledge discovery and data mining", *Springer Science and Business Media*, vol. 454, 2012.

[14] N. Mac Parthaláin and R. Jensen, "Simultaneous feature and instance selection using fuzzy-rough bireducts", *Proceedings of the 22nd IEEE International Conference on Fuzzy Systems (FUZZ-IEEE '13)*, pp. 1–8, 2013.

[15] N. Mac Parthaláin, and R. Jensen, "Unsupervised fuzzy-rough set-based dimensionality reduction", *Information Sciences*, vol. 229, pp. 106–121, 2013.

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

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

[18] 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.

[19] E. Ramentol, S. Vluymans, N. Verbiest, Y. Caballero, R. Bello Perez, C. Cornelis, F. Herrera, "IFROWANN: Imbalanced Fuzzy-Rough Ordered Weighted Average Nearest Neighbor Classification", *IEEE Transactions on Fuzzy Systems*, vol. 23, no. 5, pp. 1622–1637, Oct. 2015.

[20] 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.

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

[22] S. Stawicki and D. Ślęzak, "Recent advances in decision bireducts: Complexity, heuristics and streams", in *Rough Sets and Knowledge Technology*, Lecture Notes in Computer Science, P. Lingras, M. Wolski, C. Cornelis, S. Mitra, and P. Wasilewski, Eds. Springer, Berlin Heidelberg, vol. 8171, pp. 200-212, 2013.

[23] V. Torra and Y. Narukawa, *Modeling Decisions: Information Fusion and Aggregation Operators*, Springer, 2007.

[24] A. Tsymbal, M. Pechenizkiy, and P. Cunningham, "Diversity in search strategies for ensemble feature selection", *Information Fusion*, vol. 6, no. 1, pp. 83–98, 2005.

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

[26] Z. Zhou, *Ensemble Methods: Foundations and Algorithms*, Chapman and Hall/CRC Data Mining and Knowledge Discovery Series. Taylor & Francis, 2012.