Fuzzy-Rough Instance Selection

Richard Jensen and Chris Cornelis

Abstract— **Rough set theory provides a useful mathematical foundation for developing automated computational systems that can help understand and make use of imperfect knowledge. Since its introduction, this theory has been successfully utilised to devise mathematically sound and often, computationally efficient techniques for addressing problems such as hidden pattern discovery from data, feature selection and decision rule generation. Fuzzy-rough set theory improves upon this by enabling uncertainty and vagueness to be modeled more effectively. Recently, the value of fuzzy-rough sets for feature selection and rule induction has been established. However, the potential of this theory for instance selection has not been investigated at all. This paper proposes three novel methods for instance selection based on fuzzy-rough sets. The initial experimentation demonstrates that the methods can significantly reduce the number of instances whilst maintaining high classification accuracies.**

I. Introduction

Dealing with incomplete or imperfect knowledge is the core of much research in computational intelligence and cognitive sciences. Being able to understand and manipulate such knowledge is of fundamental significance to many theoretical developments and practical applications of automation and computing, especially in the areas of decision analysis, machine learning and data mining, intelligent control and pattern recognition.

An additional hurdle faced by many of these techniques is the sheer volume of data that must be processed and analysed. This increases the chances that learning algorithms find spurious patterns that are not valid in general. One effective way of dealing with this is through the use of feature selection, where redundant or irrelevant features are detected and removed before further processing takes place. However, sometimes the problem encountered is the prohibitively high number of training instances present or conflicting information between them. In this case, instance selection is desired to make the volume of data manageable and to remove misleading training instances in an effort to improve learned models from this data.

Over the past ten years, rough set theory (RST [10]) has become a topic of great interest to researchers and has been applied to many domains. RST offers an alternative approach that preserves the underlying semantics of the data while allowing reasonable generality. It possesses many attributes that are highly desirable; for example, it requires no parameters (eliminating the need for, possibly erroneous, human input) and it finds a minimal knowledge representation. The two main areas of highly successful application for RST are feature selection and rule induction. However, almost no research has been carried out into the use of rough set theory for instance selection. This paper proposes three approaches to fuzzy-rough instance selection (FRIS). The main idea behind these approaches is to remove instances that cause conflicts with other instances as determined by the fuzzy-rough positive region. By removing these instances, the quality of training data can be improved and classifier training time reduced.

The remainder of this paper is structured as follows: in Section II, the necessary theoretical background is provided concerning the required rough and fuzzy-rough set concepts. Section III details the proposed approaches to fuzzy-rough instance selection. Initial experimental results are provided in Section IV that demonstrate the potential of the approaches, and the paper is concluded in Section V.

II. Theoretical Background

A. Rough Set Analysis

In rough set analysis [10], data is represented as an *information system* $(X, \mathcal{A})$, where $X = \{x_1, \ldots, x_n\}$ and $\mathcal{A} = \{a_1, \ldots, a_m\}$ are finite, non-empty sets of objects and attributes, respectively. Each a in $\mathcal{A}$ corresponds to an $X \to V_a$ mapping, in which V_a is the value set of a over X. For every subset B of $\mathcal{A}$, the B-indiscernibility relation[1] R_B is defined as

$$R_B = \{(x, y) \in X^2 \text{ and } (\forall a \in B)(a(x) = a(y))\} \quad (1)$$

Clearly, R_B is an equivalence relation. Its equivalence classes $[x]_{R_B}$ can be used to approximate concepts, i.e., subsets of the universe X. Given $A \subseteq X$, its lower and upper approximation w.r.t. R_B are defined by

$$R_B{\downarrow}A = \{x \in X | [x]_{R_B} \subseteq A\} \quad (2)$$

$$R_B{\uparrow}A = \{x \in X | [x]_{R_B} \cap A \neq \emptyset\} \quad (3)$$

A *decision system* $(X, \mathcal{A} \cup \{d\})$ is a special kind of information system, used in the context of classification, in which d ($d \notin \mathcal{A}$) is a designated attribute called the decision attribute. Its equivalence classes $[x]_{R_d}$ are called decision classes. Given $B \subseteq \mathcal{A}$, the B-positive region POS_B contains those objects from X for which the values of B allow to predict the decision class unequivocally:

$$POS_B = \bigcup_{x \in X} R_B{\downarrow}[x]_{R_d} \quad (4)$$

[1]When $B = \{a\}$, i.e., B is a singleton, we will write R_a instead of $R_{\{a\}}$

R. Jensen is with the Department of Computer Science, Aberystwyth University, UK (email: rkj@aber.ac.uk)

C. Cornelis is with the Department of Applied Mathematics and Computer Science, Ghent University, Gent, Belgium (email: Chris.Cornelis@UGent.be)

978-1-4244-8126-2/10/$26.00 ©2010 Crown

Indeed, if $x \in POS_B$, it means that whenever an object has the same values as x for the attributes in B, it will also belong to the same decision class as x. The predictive ability w.r.t. d of the attributes in B is then measured by the following value (degree of dependency of d on B):

$$\gamma_B = \frac{|POS_B|}{|X|} \tag{5}$$

$(X, \mathcal{A} \cup \{d\})$ is called *consistent* if $\gamma_{\mathcal{A}} = 1$. A subset B of $\mathcal{A}$ is called a *decision reduct* if it satisfies $POS_B = POS_{\mathcal{A}}$, i.e., B preserves the decision making power of $\mathcal{A}$, and moreover it cannot be further reduced, i.e., there exists no proper subset B' of B such that $POS_{B'} = POS_{\mathcal{A}}$. If the latter constraint is lifted, i.e., B is not necessarily minimal, we call B a decision superreduct.

B. Rough Instance Selection

Preliminary work on crisp rough set-based instance selection was reported in [2]. Here, the authors developed two methods for this purpose. The starting point of both methods is the calculation of a decision reduct from the data set.

For the first method, once a reduct has been calculated, the positive region is calculated. Any objects not appearing in the positive region are then removed. One limitation with this approach is that only inconsistent instances are removed by this process - all other instances remain. The second method takes a similar approach, but also considers those objects appearing in the boundary region (those objects in the upper approximation but not in the lower approximation) of decision classes. Objects that appear in the boundary region are uncertain in that there is not enough information to determine their class membership with certainty. These objects are then altered using the Generalized Editing Algorithm [11], where class labels are changed and suspicious instances are removed.

C. Fuzzy-rough Sets

Research on the hybridization of fuzzy sets and rough sets emerged in the early 1990s [6] and has flourished recently [7]. It has focused predominantly on fuzzifying the formulas (2) and (3) for lower and upper approximation. In doing so, the following two guiding principles have been widely adopted:

- The set A may be generalized to a fuzzy set in X, allowing that objects can belong to a given concept (i.e., meet its characteristics) to varying degrees.
- Rather than assessing objects' indiscernibility, we may measure their *approximate equality*, represented by a fuzzy relation R. As a result, objects are categorized into classes, or granules, with "soft" boundaries based on their similarity to one another. As such, abrupt transitions between classes are replaced by gradual ones, allowing that an element can belong (to varying degrees) to more than one class.

Typically, we assume that R is at least a fuzzy tolerance relation. It should be mentioned that many authors impose an additional requirement of $\mathcal{T}$-transitivity, i.e., given a t-norm $\mathcal{T}$,

$$\mathcal{T}(R(x, y), R(y, z)) \leq R(x, z)$$

should hold for any x, y and z in X; R is then called a fuzzy $\mathcal{T}$-equivalence relation, or similarity relation. While $\mathcal{T}$-equivalence relations naturally extend the transitivity of their classical counterparts, they may exhibit some undesirable effects, which were pointed out e.g. in [5].

Assuming that for a qualitative (i.e., nominal) attribute a, the classical way of discerning objects is used, i.e., $R_a(x, y) = 1$ if $a(x) = a(y)$ and $R_a(x, y) = 0$ otherwise, we can define, for any subset B of $\mathcal{A}$, the fuzzy B-indiscernibility relation by

$$R_B(x, y) = \mathcal{T}(\underbrace{R_a(x, y)}_{a \in B}) \tag{6}$$

in which $\mathcal{T}$ represents a t-norm. It can easily be seen that if only qualitative attributes (possibly originating from discretization) are used, then the traditional concept of B-indiscernibility relation is recovered.

For the lower and upper approximation of a fuzzy set A in X by means of a fuzzy tolerance relation R, we adopt the definitions proposed by Radzikowska and Kerre in [12]: given an implicator $\mathcal{I}$ and a t-norm $\mathcal{T}$, they paraphrased formulas (2) and (3) to define $R{\downarrow}A$ and $R{\uparrow}A$ by

$$(R{\downarrow}A)(y) = \inf_{x \in X} \mathcal{I}(R(x, y), A(x)) \tag{7}$$

$$(R{\uparrow}A)(y) = \sup_{x \in X} \mathcal{T}(R(x, y), A(x)) \tag{8}$$

for all y in X.

Using fuzzy B-indiscernibility relations, we can define the fuzzy B-positive region by, for y in U,

$$POS_B(y) = \left(\bigcup_{x \in X} R_B{\downarrow}R_d x \right)(y) \tag{9}$$

This means that the fuzzy positive region is a fuzzy set in X, to which an object y belongs to the extent that its R_B-foreset is included into *at least one* of the decision classes.

While formula (9) provides the most faithful way to define the fuzzy positive region, it is not the most practically useful one in this case, since the computational complexity is high (cubic in the number of objects for computing the entire positive region). Therefore we may opt to replace it by

$$POS'_B(y) = (R_B{\downarrow}R_d y)(y) \tag{10}$$

which results in smaller positive regions (as shown above), that are easier to compute (quadratic complexity in the number of objects for computing the entire positive region). Equation (9) becomes equation (10) when the decision feature is crisp [4].

Once we have fixed the fuzzy positive region, we can define an increasing $[0, 1]$-valued measure to gauge the degree of dependency of a subset of features on another subset of features. For feature selection it is useful to phrase this in terms of the dependency of the decision feature on a subset of the conditional features:

$$\gamma_B = \frac{|POS_B|}{|POS_{\mathcal{A}}|} \text{ and } \gamma'_B = \frac{|POS'_B|}{|POS'_{\mathcal{A}}|} \tag{11}$$

III. Fuzzy-rough Instance Selection

This section details three approaches to achieving instance selection via fuzzy-rough sets. The central notion for all algorithms is the use of information in the positive region to determine how useful instances are and whether they can be removed.

Assume $S \subset X$ is a set of training examples. For our purposes, given a decision system $(X, \mathcal{A} \cup \{d\})$, let a be a quantitative attribute in $\mathcal{A} \cup \{d\}$ with range $l(a)$. To express the approximate equality between two objects w.r.t. a, in this paper we use the fuzzy relation R_a for x and y in S:

$$R_a^\alpha(x,y) = \max\left(0, 1 - \alpha\frac{|a(x) - a(y)|}{l(a)}\right) \tag{12}$$

The parameter α ($\alpha \geq 0$) determines the granularity of R_a^α. It should also be noted that equation (12) is not the only possibility to define the similarity of x and y based on attribute a. For any subset B of $\mathcal{A}$, the fuzzy B-indiscernibility relation is:

$$R_B^\alpha(x,y) = \mathcal{T}(\underbrace{R_a^\alpha(x,y)}_{a \in B}) \tag{13}$$

In this paper, the lower approximation $R_B^\alpha{\downarrow}^S A$ of a fuzzy set A in S by means of a fuzzy relation R_B^α is defined by, for $y \in S$:

$$(R_B^\alpha{\downarrow}^S A)(y) = \inf_{x \in S} \mathcal{I}(R_B^\alpha(x,y), A(x)) \tag{14}$$

We can then define the fuzzy B-positive region $POS_B^{\alpha,S}$ by, for y in S,

$$POS_B^{\alpha,S}(y) = (R_B^\alpha{\downarrow}^S R_d^\alpha y)(y) \tag{15}$$

The corresponding degree of dependency is then defined by

$$\gamma_B^{\alpha,S} = \frac{\sum_{y \in S} POS_B^{\alpha,S}(y)}{|S|} \tag{16}$$

A. FRIS-I

A simple approach to fuzzy-rough instance selection can be seen in Fig. 1. The algorithm requires as input the set of objects that are to be reduced, the parameter α that is to be used in the fuzzy similarity measure, and an optional parameter τ that can be used to remove a greater number of objects if required (see line (3)). Typically, τ should be set to 1.

The algorithm evaluates the degree of membership of each object x to the positive region; if this is less than the threshold, then the object can be removed. When an object membership is less than 1, this means that there is some uncertainty as to which class this object truly belongs. If all such objects are removed, then there is no inconsistency exhibited by the remaining objects.

FRIS-I(S,α,τ).
S, the set of objects to be reduced;
α, the granularity parameter;
τ, a selection threshold.

(1) $Y \leftarrow S$
(2) **foreach** $x \in S$
(3)   **if** ($POS_{\mathcal{A}}^{\alpha,S}(x) < \tau$)
(4) $Y \leftarrow Y - \{x\}$
(5) **return** Y

Fig. 1. The fuzzy-rough instance selection algorithm I

B. FRIS-II

Although the above approach is quite efficient, it will remove more objects than is strictly necessary as the removal of one object might affect the positive region membership of the remaining objects. For instance, the positive region membership of an object y might be reduced due to its proximity to a noisy object x. In this case, the removal of x will result in y belonging fully to the positive region and should therefore not be removed in the absence of x. The removal of an object cannot reduce the positive region memberships of the remaining objects. Given z, an object to be removed,

$$\begin{aligned} POS_B^{\alpha,S\cup z}(y) &= (R_B^\alpha{\downarrow}^{S\cup z} R_d^\alpha y)(y) \\ &= \min(\min_{x \in S} \mathcal{I}(R_B^\alpha(x,y), R_d^\alpha y(x)), \\ &\qquad \mathcal{I}(R_B^\alpha(z,y), R_d^\alpha y(z))) \\ &\leq \min_{x \in S} \mathcal{I}(R_B^\alpha(x,y), R_d^\alpha y(x)) \\ &\leq POS_B^{\alpha,S}(y) \end{aligned}$$

A better method is to use the positive region information to select the object with lowest membership for removal and then recalculate each object's membership to the positive region with this object removed. This process can then be repeated until all objects belong fully. The algorithm can be seen in Fig. 2.

C. FRIS-III

An alternative method to both FRIS-I and FRIS-II is to perform a backward elimination of objects. The algorithm can be found in Fig. 3. Here, we start with the full training set X and then look for the object x such that $\gamma_{\mathcal{A}}^{\alpha,X-\{x\}}$ is maximal. That is, the object whose removal expands the positive region the most. This object is then removed from the set of instances and the process repeats until the degree of dependency is 1 (i.e. all objects belong maximally to the positive region).

This algorithm is the most computationally complex as the degree of dependency is calculated for each possible removal of an object.

FRIS-II(S,α).
S, the set of objects to be reduced;
α, the granularity parameter.

while (`true`)
  $z \leftarrow \emptyset$, $\rho_z \leftarrow 1$
  foreach $x \in S$
    if $(POS_{\mathcal{A}}^{\alpha,S}(x) < \rho_z)$
      $z \leftarrow x$
      $\rho_z \leftarrow POS_{\mathcal{A}}^{\alpha,S}(x)$
  if $(z \neq \emptyset)$
    $S \leftarrow S - \{z\}$
  else return S

Fig. 2. The fuzzy-rough instance selection algorithm II

FRIS-III(S,α).
S, the set of objects to be reduced;
α, the granularity parameter.

$\rho \leftarrow \gamma_{\mathcal{A}}^{\alpha,S}$
while $(\rho \neq 1)$
  $z \leftarrow \emptyset$, $\rho_z \leftarrow 0$
  foreach $x \in S$
    if $(\gamma_{\mathcal{A}}^{\alpha,S-\{x\}} > \rho_z)$
      $z \leftarrow x$, $\rho_z \leftarrow \gamma_{\mathcal{A}}^{\alpha,S-\{x\}}$
  $S \leftarrow S - z$
  $\rho \leftarrow \rho_z$
return S

Fig. 3. The fuzzy-rough instance selection algorithm III

For all algorithms, the fuzzy positive region or degree of dependency have been used to evaluate the worth of objects. Similarly, extensions of these concepts can be used for the same purpose. For example, by replacing these concepts by their vaguely quantified equivalents [3], the approaches could be more robust to noise.

IV. Experimentation

This section presents the initial experimental evaluation of the proposed method for the task of instance selection, over three benchmark datasets from [1] with several classifiers. The details of the datasets used can be found in Table I.

TABLE I
Dataset characteristics

Dataset	Objects	Attributes	Classes
Cleveland	297	13	5
Heart	214	9	2
Wine	178	13	3

The classifiers themselves are obtained from the Weka toolkit [14], and are evaluated using their default parameter settings[2]. In order to determine the general benefit of fuzzy-rough instance selection, a variety of classifiers have been used; namely, a support vector machine-based method (SMO), a decision tree learner (J48) and an instance-based learner (IBk). Additionally, two approaches for nearest neighbor classification are used based on fuzzy-rough sets (FRNN and VQNN) [9].

The overview of the experimental setup can be seen in Fig. 4. The procedure first splits the given dataset into ten cross-validation folds. Each training fold is passed to the instance selection process where instance selection is carried out, and the classifier is trained on this reduced fold. Classification is then carried out using the unreduced test fold each time. When instance selection is not required, this process is omitted. Identical initial folds are produced for runs both with and without instance selection and therefore the classification results are comparable.

The fuzzy tolerance relation used in this paper requires a parameter, α, to be selected. As this choice is dependent on the dataset (and also, to a lesser extent, on the classifiers themselves) a range of parameter values is evaluated. In the results presented in the following sections, the tables show the (average) classifier performance for no instance selection, followed by the classifier performance with instance selection for a particular α value, and finally the best classifier performance achieved as a a result of selection. N is the number of instances removed by the method for that particular algorithm and α value. In this experimentation, we used the Łukasiewicz implicator and t-norm as used in fuzzy-rough feature selection. For FRIS-I, the parameter τ was set to 1.

A. Cleveland *dataset*

The results for the three methods for the Cleveland dataset can be found in Tables II, III, and IV.

TABLE II
Cleveland dataset: FRIS-I

Alpha	FRNN	VQNN	J48	SMO	IBk	N
No selection	56.23	57.91	54.55	58.92	54.88	-
0.1	58.25	57.24	54.21	57.24	56.90	77
0.3	58.25	57.24	54.21	57.24	56.90	77
0.5	58.25	57.24	54.21	57.24	56.90	77
0.7	57.91	57.24	53.87	57.58	56.57	73
0.9	57.24	57.58	52.19	57.91	56.57	71
1.1	55.89	56.23	54.88	59.26	55.22	56
1.3	56.90	57.24	54.88	58.92	54.88	46
1.5	57.58	57.24	54.21	59.60	54.55	37
1.7	56.23	57.24	55.22	59.93	54.21	31
1.9	56.57	57.58	54.88	58.59	53.87	24
2.1	56.57	57.24	54.21	58.59	54.55	21
2.3	57.24	57.24	53.87	58.92	55.56	14
2.5	56.90	58.59	55.56	59.60	55.22	9
2.7	55.89	58.25	55.56	58.92	54.55	7
2.9	56.23	57.24	53.20	58.59	54.88	6
Best	58.25	58.59	55.56	59.93	56.90	

[2] All techniques described in this paper have been implemented in Weka. The program can be downloaded from http://users.aber.ac.uk/rkj/book/programs.php

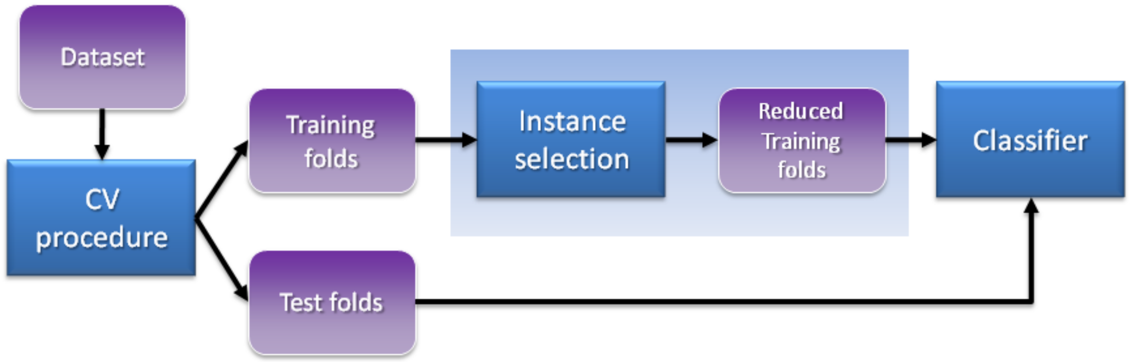

Fig. 4. Experimentation overview

TABLE III
CLEVELAND DATASET: FRIS-II

Alpha	FRNN	VQNN	J48	SMO	IBk	N
No selection	56.23	57.91	54.55	58.92	54.88	-
0.1	56.23	57.24	54.88	57.58	55.89	42
0.3	56.23	57.24	54.88	57.58	55.89	42
0.5	56.23	57.24	54.88	57.58	55.89	42
0.7	56.23	57.58	54.88	56.90	55.56	40
0.9	55.56	57.91	54.21	56.90	53.87	38
1.1	55.22	58.59	57.24	59.26	53.87	30
1.3	55.56	57.58	53.87	58.59	53.54	25
1.5	55.56	57.58	52.86	58.25	53.54	18
1.7	55.22	57.58	54.21	57.58	53.54	15
1.9	55.22	57.58	53.20	57.91	53.54	12
2.1	55.56	57.58	54.21	59.26	54.55	10
2.3	55.89	57.91	55.22	59.26	55.22	7
2.5	55.56	58.25	54.55	59.60	55.22	4
2.7	55.89	58.25	54.88	58.25	54.88	3
2.9	56.23	57.91	54.21	58.92	54.88	3
Best	56.23	58.59	57.24	59.60	55.89	

TABLE IV
CLEVELAND DATASET: FRIS-III

Alpha	FRNN	VQNN	J48	SMO	IBk	N
No selection	56.23	57.91	54.55	58.92	54.88	-
0.1	56.23	57.91	53.87	58.92	55.22	38
0.3	56.23	57.91	53.87	58.59	55.22	37
0.5	56.23	57.24	52.53	58.59	55.89	35
0.7	54.55	56.90	54.88	57.24	55.22	34
0.9	54.55	57.58	54.88	58.59	56.23	32
1.1	53.54	58.59	51.52	58.59	55.22	23
1.3	55.56	57.91	54.21	58.59	54.55	19
1.5	54.88	57.24	56.23	58.59	55.22	16
1.7	55.22	57.91	53.87	58.92	55.89	12
1.9	54.88	58.25	55.22	59.93	55.22	7
2.1	54.55	58.59	55.22	57.24	55.22	6
2.3	54.55	57.91	55.22	58.59	55.22	5
2.5	54.55	59.26	55.56	58.59	54.88	3
2.7	53.87	58.92	54.21	58.25	54.55	3
2.9	54.55	58.59	54.55	58.59	54.88	1
Best	56.23	59.26	56.23	59.93	56.23	

It can be seen that all instance selection methods manage to retain or improve the performance of all classifiers. For FRIS-I, $\alpha \in [0.1, 0.5]$ results in 77 instances being removed on average (28% of the original training fold) whilst preserving classification accuracy across all classifiers. FRIS-II and FRIS-III perform similarly, but these methods do not remove as many objects. This demonstrates that, for this dataset, a significant proportion of objects are unnecessary for constructing robust models. As the number of removed instances increases, there is not much change in the classification accuracies.

TABLE V
HEART DATASET: FRIS-I

Alpha	FRNN	VQNN	J48	SMO	IBk	N
No selection	75.56	82.59	76.30	82.96	75.56	-
0.1	79.63	83.70	80.74	83.70	77.04	39
0.3	79.63	83.70	80.74	83.70	77.04	39
0.5	79.63	83.70	80.74	83.70	77.04	39
0.7	78.89	84.07	79.63	83.33	77.04	35
0.9	78.52	84.07	79.26	82.96	76.67	33
1.1	77.04	83.70	78.89	83.33	76.30	24
1.3	76.67	83.33	77.78	84.07	75.93	15
1.5	76.67	82.59	77.41	83.33	75.93	8
1.7	76.30	82.59	77.78	82.96	75.19	7
1.9	77.04	82.59	77.04	82.59	75.19	4
Best	79.63	84.07	80.74	84.07	77.04	

B. HEART *dataset*

The results for the three methods for the HEART dataset can be found in Tables V, VI, and VII.

All instance selection algorithms improve or maintain the accuracies resulting from the classifiers trained on the unreduced data. FRIS-I benefits the nearest neighbor classifiers the most, particularly when α is small, resulting in a large number of instances being removed and higher accuracies. Again, the extent of instance reduction seen for FRIS-II and FRIS-III is not as large as for FRIS-I. Overall, the number of instances removed is less than that for the CLEVELAND dataset which suggests that the data contains fewer noisy instances.

It is interesting to note here that FRIS-I removes around twice the number of instances the other two algorithms remove, and yet attains higher accuracies for the classifiers. It appears to be the case that, for this dataset, it is better to remove all potentially noisy objects, rather than attempting

TABLE VI

HEART DATASET: FRIS-II

Alpha	FRNN	VQNN	J48	SMO	IBk	*N*
No selection	75.56	82.59	76.30	82.96	75.56	-
0.1	78.15	82.96	75.93	82.22	75.93	18
0.3	78.15	82.96	75.93	82.22	75.93	18
0.5	78.15	82.96	75.93	82.22	75.93	18
0.7	78.15	82.59	77.04	82.22	76.67	16
0.9	78.15	82.59	77.04	82.22	76.67	15
1.1	77.78	82.96	76.30	82.59	76.67	12
1.3	77.78	82.96	75.19	82.96	76.67	8
1.5	77.78	82.59	75.56	82.96	75.93	4
1.7	77.41	82.59	76.30	83.33	75.93	3
1.9	77.04	82.59	76.67	83.33	75.56	2
Best	78.15	82.96	77.04	83.33	76.67	

TABLE VII

HEART DATASET: FRIS-III

Alpha	FRNN	VQNN	J48	SMO	IBk	*N*
No selection	75.56	82.59	76.30	82.96	75.56	-
0.1	77.78	82.59	76.67	81.85	75.93	15
0.3	77.78	82.59	77.41	81.11	75.93	14
0.5	78.15	82.59	77.41	81.48	76.30	14
0.7	77.41	82.22	77.04	81.85	76.30	14
0.9	77.04	82.96	77.04	81.85	76.67	13
1.1	76.67	82.22	76.30	83.33	75.93	10
1.3	77.41	82.22	75.93	84.44	75.56	6
1.5	78.15	82.22	75.56	83.70	75.56	5
1.7	77.04	82.22	75.93	84.81	75.93	2
1.9	76.67	82.59	75.93	84.07	75.93	1
Best	78.15	82.96	77.41	84.81	76.67	

to retain some of them (the approach taken by FRIS-II and FRIS-III).

C. WINE *dataset*

The results for the three methods for the WINE dataset can be found in Tables VIII, IX, and X.

TABLE VIII

WINE DATASET: FRIS-I

Alpha	FRNN	VQNN	J48	SMO	IBk	*N*
No selection	97.19	97.75	94.38	98.31	94.94	-
0.4	95.51	65.17	69.10	91.57	93.26	148
0.5	96.63	97.75	80.34	97.19	96.63	105
0.6	94.94	96.63	91.01	97.19	94.94	51
0.7	97.19	98.31	92.13	97.75	95.51	17
0.8	97.19	97.75	93.82	98.31	94.94	8
0.9	97.19	97.75	94.38	98.31	94.94	2
Best	97.19	98.31	94.38	98.31	95.51	

For this dataset, FRIS-III performs better in terms of improving classification accuracy and is more consistent across the α range. As the accuracies for the unreduced methods is already high, instance selection can only improve the performance by a small amount. FRIS-I removes too many instances for small values of α. For example, when $\alpha = 0.4$ on average 91.9% of each training fold is removed. Surprisingly, this does not affect FRNN and IBk adversely. For $\alpha = 0.5$, all classifiers except J48 perform well even though 65.2% of the training objects have been removed for each cross-validation iteration.

TABLE IX

WINE DATASET: FRIS-II

Alpha	FRNN	VQNN	J48	SMO	IBk	*N*
No selection	97.19	97.75	94.38	98.31	94.94	-
0.4	90.45	93.26	86.52	93.26	88.76	77
0.5	96.07	95.51	83.71	94.94	93.82	39
0.6	96.00	98.31	89.89	96.07	94.94	21
0.7	97.19	97.75	93.26	97.75	94.94	11
0.8	97.19	97.75	93.26	97.75	94.94	4
0.9	97.19	97.75	93.82	98.31	94.94	1
Best	97.19	98.31	93.82	98.31	94.94	

TABLE X

WINE DATASET: FRIS-III

Alpha	FRNN	VQNN	J48	SMO	IBk	*N*
No selection	97.19	97.75	94.38	98.31	94.94	-
0.4	95.51	95.51	87.64	94.38	94.38	47
0.5	96.63	97.19	89.89	97.19	93.82	23
0.6	98.31	98.31	92.13	97.75	95.51	9
0.7	97.75	97.75	92.13	97.75	94.94	3
0.8	97.19	97.75	94.94	97.75	94.94	2
0.9	97.19	98.31	94.38	98.31	94.94	1
Best	98.31	98.31	94.94	98.31	95.51	

V. CONCLUSIONS

This paper has presented three methods for instance selection via fuzzy-rough sets. All methods are based on the removal of instances that negatively affect the fuzzy positive region. Instances are removed until there is no uncertainty amongst them, i.e. all remaining objects belong fully to the positive region. From the initial experimentation it can be seen that all selection methods can improve classifier performance. Overall, FRIS-I appears to be best for removing the most amount of instances whilst keeping the classification accuracy unaffected. This is also the simplest method, computationally.

There is much potential for further developments in this area. Throughout the paper, the fuzzy-rough positive region has been used to gauge the quality of objects. This can be replaced by existing extensions such as the vaguely quantified or variable precision fuzzy-rough positive region that may give a better indication as to object quality through superior noise tolerance. The computational complexity of FRIS-III will need to be improved in order for it to be applied effectively to datasets containing thousands of objects. Also, this paper has focused solely on the removal of objects, whereas instance editing (i.e. altering instances rather than removing them) may provide a better alternative for improving classifier performance. Finally, there is the potential for a combined fuzzy-rough instance selection and feature selection method that would perform both types of data reduction simultaneously.

ACKNOWLEDGMENT

Chris Cornelis would like to thank the Research Foundation—Flanders for funding his research.

REFERENCES

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

[2] Y. Caballero, R. Bello, Y. Salgado, M.M. Garca, "A Method to Edit Training Set Based on Rough Sets," *International Journal of Computational Intelligence Research*, vol. 3, no. 3, pp. 219–229, 2007.

[3] C. Cornelis and R. Jensen, "A Noise-tolerant Approach to Fuzzy-Rough Feature Selection," *Proc. 17th International Conference on Fuzzy Systems (FUZZ-IEEE'08)*, pp. 1598–1605, 2008.

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

[5] M. De Cock, C. Cornelis, and E.E. Kerre, "Fuzzy Rough Sets: The Forgotten Step," *IEEE Transactions on Fuzzy Systems*, vol. 15, no. 1, pp. 121–130, 2007.

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

[7] R. Jensen and P. Lingras, "Survey of Rough and Fuzzy Hybridization," *Proc. 16th International Conference on Fuzzy Systems (FUZZ-IEEE'07)*, pp. 125–130, 2007.

[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, "A New Approach to Fuzzy-Rough Nearest Neighbour Classification," *Proc. 6th International Conference on Rough Sets and Current Trends in Computing*, pp. 310–319, 2008.

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

[11] J. Koplowitz and T.A. Brown, "On the relation of performance to editing in nearest neighbor rules," *Pattern Recognition*, vol. 13, no. 3, pp. 251–255, 1981.

[12] 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.

[13] G.C.Y. Tsang, D. Chen, E.C.C. Tsang, J.W.T. Lee, and D.S. Yeung, "On attributes reduction with fuzzy rough sets," *Proc. 2005 IEEE International Conference on Systems, Man and Cybernetics*, vol. 3, pp. 2775–2780, 2005.

[14] I. H. Witten and E. Frank, *Data Mining: Practical machine learning tools with Java implementations*, Morgan Kaufmann Publishers, San Francisco, 2000.