Exploring the Boundary Region of Tolerance Rough Sets for Feature Selection

Neil Mac Parthaláin and Qiang Shen

Department of Computer Science, Aberystwyth University, Wales, UK.

Abstract

Of all of the challenges which face the effective application of computational intelligence technologies for pattern recognition, dataset dimensionality is undoubtedly one of the primary impediments. In order for pattern classifiers to be efficient, a dimensionality reduction stage is usually performed prior to classification. Much use has been made of Rough Set Theory for this purpose as it is completely data-driven and no other information is required; most other methods require some additional knowledge. However, traditional rough set-based methods in the literature are restricted to the requirement that all data must be discrete. It is therefore not possible to consider real-valued or noisy data. This is usually addressed by employing a discretisation method, which can result in information loss. This paper proposes a new approach based on the tolerance rough set model, which has the ability to deal with real-valued data whilst simultaneously retaining dataset semantics. More significantly, this paper describes the underlying mechanism for this new approach to utilise the information contained within the boundary region or region of uncertainty. The use of this information can result in the discovery of more compact feature subsets and improved classification accuracy. These results are supported by an experimental evaluation which compares the proposed approach with a number of existing feature selection techniques.

Key words: feature selection, attribute reduction, rough sets, classification

1 Introduction

Quite often, at the point of data collection every single aspect of a domain may be recorded such that complete representation can be achieved. The problems associated with such large dimensionality however mean that any attempt to use machine learning tools to extract knowledge, results in very poor performance. Feature Selection (FS) [4], [9], [10], [11], [23], [24], [46] is a process which attempts to select features which are information-rich and also retain the original meaning of the features following reduction. It is not surprising therefore, that feature selection has been applied to problems which have very large dimensionality (>10 000) [2].

Problems of such scale are outside the scope of most learning algorithms, and in cases where they are not, the learning algorithm will often find patterns that are spurious and invalid. As mentioned previously, it may be expected that the inclusion of an increasing number of features would also improve the likelihood of the ability to distinguish between classes. This may not be the case however if the training data size does not also increase significantly with the addition of each feature. Most learning approaches utilise a reduction step to overcome such problems when dealing with high dimensionality.

Rough set theory (RST) [30] is an approach that can be used for dimensionality reduction, whilst simultaneously preserving the semantics of the features [38]. Also, as RST operates only on the data and does not require any thresholding information, it is completely data-driven. Other useful approaches may also be employed for dimensionality reduction and FS such as; [9], [23], [32], [46], unlike RST however these approaches require additional information or transform the data. The main disadvantage of RST is its inability to deal with real-valued data. In order to tackle this problem, methods of discretising the data were employed prior to the application of RST. The use of such methods can result in information loss however, and a number of extensions to RST have emerged [7], [39], [48] which have attempted to address this inability to operate on real-valued domains. Perhaps the most significant of these is the tolerance rough set model (TRSM) [39]. TRSM has the ability to operate effectively on real-valued (and crisp) data, thus minimising any information loss.

This paper presents a new method for feature selection which is based on the TRSM. It employs a distance metric to examine the uncertain information contained in the boundary region of tolerance rough sets, and uses this information to guide the feature selection process. This uncertain information is normally ignored in the traditional RST and TRSM approaches to FS which can result in information loss. The remainder of this paper is structured as follows. Section 2 introduces the theoretical background to RST and TRSM and their application to feature selection. Section 3 presents the new distance metric-assisted tolerance rough set selection method with a worked example to demonstrate the approach fully. All experimental evaluation and results for both approaches is presented in section 4, as well as a comparison with the Principal Component Analysis (PCA) dimensionality reduction technique [12], and also four additional FS techniques CFS [13], consistency-based FS [5], ReliefF [19], and a wrapper FS approach which employs J48 [33] as an evaluation metric. The paper is then concluded with a brief discussion of future work in section 5.

2 Background

Although the principal focus of this paper lies in the examination of the information contained in the boundary region of tolerance rough sets, an in-depth view of

both the RST and TRSM methodologies is necessary in order to demonstrate the motivation for the investigation of the information in the boundary region.

Rough Set Theory [30] is an extension of conventional set theory which supports approximations in decision making. A rough set is the approximation of a vague concept by a pair of precise concepts which are known as upper and lower approximations. These concepts are illustrated in Fig.1. The lower approximation is a definition of the domain objects which are known with absolute certainty to belong the concept of interest (set X), whilst the upper approximation is the set of those objects which possibly belong to the concept of interest. The boundary region or region of uncertainty is the difference between the upper and lower approximations. Equivalence classes are groups of objects which are indiscernible from each other, such as a group of objects in which all of the condition features are the same for each object.

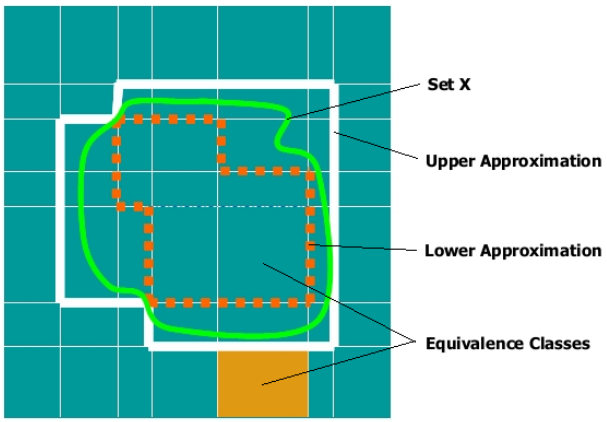

Fig. 1. Rough Set representation

2.1 Rough Set Attribute Reduction

At the heart of the RSAR approach is the concept of indiscernibility. Let $I = (\mathbb{U}, \mathbb{A})$ be an information system, where $\mathbb{U}$ is a non-empty set of finite objects (the universe) and $\mathbb{A}$ is a non-empty finite set of attributes so that $a : \mathbb{U} \rightarrow V_a$ for every $a \in \mathbb{A}$. V_a is the set of values that a can take. For any $P \subseteq \mathbb{A}$, there exists an associated equivalence relation $IND(P)$:

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

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

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

where,

$$\mathbb{U}/IND(\{a\}) = \{\{x \mid a(x) = b,\ x \in \mathbb{U}\} \mid b \in V_a\} \tag{3}$$

and,

$$A \otimes B = \{X \cap Y \mid \forall X \in A, \forall Y \in B, X \cap Y \neq \emptyset\} \tag{4}$$

where A and B are families of sets.

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

$$\underline{P}X = \{x \mid [x]_p \subseteq X\} \tag{5}$$

$$\overline{P}X = \{x \mid [x]_p \cap X \neq \emptyset\} \tag{6}$$

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

$$POS_P(Q) = \bigcup_{X \in \mathbb{U}/Q} \underline{P}X \tag{7}$$

$$NEG_P(Q) = \mathbb{U} - \bigcup_{X \in \mathbb{U}/Q} \overline{P}X \tag{8}$$

$$BND_P(Q) = \bigcup_{X \in \mathbb{U}/Q} \overline{P}X - \bigcup_{X \in \mathbb{U}/Q} \underline{P}X \tag{9}$$

By employing this definition of the positive region it is possible to calculate the rough set degree of dependency of a set of attributes Q on a set of attributes P. This can be achieved as follows: For P,$Q \subseteq A$, it can be said that Q depends on P

in a degree k $(0 \leq k \leq 1)$, this is denoted $(P \Rightarrow_k Q)$ if:

$$k = \gamma_P(Q) = \frac{\mid POS_P(Q) \mid}{\mid \mathbb{U} \mid} \tag{10}$$

The reduction of attributes or selection of survival features can be achieved through the comparison of equivalence relations generated by sets of attributes. Attributes are removed such that the reduced set provides identical predictive capability of the decision feature or features as that of the original or unreduced set of features. A $reduct$ (R) can be defined as a subset of minimal cardinality of the conditional attribute set (C) where $\gamma_R(\mathbb{D}) = \gamma_{\mathbb{C}}(\mathbb{D})$, where D is the decision attribute set.

The *QuickReduct* algorithm in [38] also shown below searches for a minimal subset without exhaustively generating all possible subsets. The search begins with an empty subset, attributes which result in the greatest increase in the rough set dependency value are added iteratively. This process continues until the search produces its maximum possible dependency value for that dataset $(\gamma_c(\mathbb{D}))$. Note that this type of hill-climbing search does not guarantee a minimal subset and may only discover a local minimum.

Algorithm *QuickReduct*
Input: C, the set of all conditional features
Input: D, the set of all decisional features
Output: R, a feature subset
1. $R \leftarrow \{\}$
2. **repeat**
3. $\quad T \leftarrow R$
4. $\quad \forall x \in (C - R)$
5. $\quad$ **if** $\gamma_{R \cup \{x\}}(D) > \gamma_T(D)$
6. $\quad T \leftarrow R \cup \{x\}$
7. $\quad R \leftarrow T$
8. **until** $\gamma_R(D) = \gamma_C(D)$
9. **return** R

2.2 Distance Metric Assisted Rough Set Attribute Reduction

Almost all techniques for rough set attribute reduction [14], [21], [22], [28], [38], [41], [47] adopt an approach to minimisation that employs the information contained within the lower approximation of a set. Very little work [6], [16], [26] has been carried out where the information in the boundary region is considered for the purpose of minimisation. This information is uncertain but may be useful in determining subset quality and hence the discovery of more compact and useful feature subsets.

The approach described in this section uses both the information contained in the lower approximation and the information contained in the boundary region to search for reducts. This work is based on an approach applied to traditional rough sets in [26]. The positive region (as defined above) is the union of lower approximations, and this is used (as described previously) for the minimisation of data. The lower approximation is the set of those objects which can be said with certainty to belong to a set X. The upper approximation is the set of objects which either definitely or possibly belong to the set X. The difference between the upper and lower approximation is the area known as the boundary region. The boundary region is an area of uncertainty. When the boundary region is empty, there is no uncertainty regarding the concept which is being approximated and all objects belong with certainty to the subset of interest.

Any useful information that may be contained in the boundary region when it is non-empty is therefore lost when only the lower approximation is employed for minimisation. In order to address this, the DMRSAR method [26] uses a distance metric to determine the proximity of objects in the boundary region to those in the lower approximation and assign a significance value to these distances.

2.2.1 Distance Metric and Mean Lower Approximation Definitions

The distance metric employed in this work attempts to qualify the objects in the boundary region with regard to their proximity to the lower approximation. Similar research although not specifically involving the lower approximation can be found in [40]. Intuitively, the closer the proximity of an object in the boundary region to the upper margin of the lower approximation, the higher the likelihood that it belongs to the set of interest. For the method outlined here, all of the distances of objects in the boundary region are calculated. From this the significance value for a set can be obtained.

Since calculating the margin of the lower approximation for an n-dimensional space would involve considerable computational effort, a more pragmatic solution is adopted, - the mean of all object attribute values in the P-lower approximation is calculated. This can be defined as follows:

$$\underline{P}X_{MEAN} = \left\{ \frac{\sum_{o \in \underline{P}X} a(o)}{\mid \underline{P}X \mid} \mid \forall a \in P \right\} \tag{11}$$

Using this definition of the mean of the P-lower approximation, the distance function for the proximity of objects in the boundary region from the P-lower approximation mean can be defined, $\delta_P(\underline{P}X_{MEAN}, y), \quad y \in BND_P(Q)$.

The exact function (δ_P)is not defined here as a number of different strategies can be employed for the calculation of the distance of objects in the boundary such as

Hausdorff distance. In the worked example section a Euclidean type distance metric is employed.

In order to measure the quality of the boundary region, a significance value ω for subset P is calculated by obtaining the sum of all object distances and inverting it such that:

$$\omega_P(Q) = (\sum_{y \in BND_P(Q))} \delta_P(\underline{P}X_{MEAN}, y))^{-1} \tag{12}$$

This significance measure takes values from the interval [0,1] and is used in conjunction with the rough set dependency value to gauge the utility of attribute subsets in a similar way to that of the rough set dependency measure. As one measure only operates on the objects in the lower approximation and the other only on the objects in the boundary, both entities can therefore be considered separately and then combined to create a new evaluation measure $\mathbb{M}$:

$$\mathbb{M}(X) = \frac{\omega_X(Q) + \gamma_X(Q)}{2} \tag{13}$$

A mean of both values is obtained as both operate in the range [0,1]. With this in mind, a new feature selection mechanism can be constructed that uses both the significance value and the rough dependency value to guide the search for the best feature subset.

2.2.2 Distance Metric-based QUICKREDUCT

DMQuickReduct shows a distance metric-based QUICKREDUCT algorithm based on the previously described rough algorithm.

DMQUICKREDUCT is similar to the RSAR algorithm but uses a combined distance and rough-set dependency value of a subset to guide the feature selection process. If the combined value ($\mathbb{M}$) of the current reduct candidate is greater than that of the previous, then this subset is retained and used in the next iteration of the loop. It is worth pointing out that the subset is evaluated by examining the value of $\mathbb{M}$, termination only occurs when the addition of any remaining features results in the dependency function value (γ_T) reaching that of the unreduced dataset. The value of $\mathbb{M}$ is therefore not used as a termination criterion.

Algorithm *DMQuickReduct*
Input: $\mathbb{C}$, the set of all conditional features
Input: $\mathbb{D}$, the set of all decisional features
Output: R, a feature subset

1. $T \leftarrow \{\}$, $R \leftarrow \{\}$
2. **repeat**
3. $\quad \forall x \in (\mathbb{C} - R)$
4. $\quad$ **if** $\mathbb{M}_{R \cup \{x\}} > \mathbb{M}(T)$
5. $\quad T \leftarrow R \cup \{x\}$
6. $\quad R \leftarrow T$
7. **until** $\gamma_R(\mathbb{D}) = \gamma_{\mathbb{C}}(\mathbb{D})$
8. **return** R

The algorithm begins with an empty subset R. The repeat-until loop works by examining the combined dependency/significance value of a subset and incrementally adding a single conditional feature at a time. For each iteration, a conditional feature that has not already been evaluated will be temporarily added to the subset R. The combined measure of the subset currently being examined (line 6) is then evaluated and compared with that of T (the previous subset). If the combined measure of the current subset is greater, then the attribute added in (line 5) is retained as part of the new subset T (line 6).

The loop continues to evaluate in the above manner by adding conditional features, until the dependency value of the current reduct candidate ($\gamma_R(\mathbb{D})$) equals the consistency of the dataset (1 if the dataset is consistent).

2.3 Tolerance-based Feature Selection

The tolerance rough set model (TRSM) [39] can be useful for application to real-valued data. TRSM employs a similarity relation to minimise data as opposed to the indiscernibility relation used in classical rough-sets. This allows a relaxation in the way equivalence classes are considered. Fig.2 shows the effect of employing this relaxation, where the granularity of the rough equivalence classes has been reduced. This flexibility allows a blurring of the boundaries of the former rough or crisp equivalence classes and objects may now belong to more than one tolerance class.

In this approach [35], suitable similarity relations must be defined for each feature, although the same definition can be used for all features if applicable. A standard measure for this purpose, given in [39], is:

$$SIM_a(x, y) = 1 - \frac{\mid a(x) - a(y) \mid}{\mid a_{max} - a_{min} \mid} \tag{14}$$

where a is a considered feature, and a_{max} and a_{min} denote the maximum and minimum values of a respectively. When considering the case where there is more than one feature, the defined similarities must be combined to provide an overall mea-

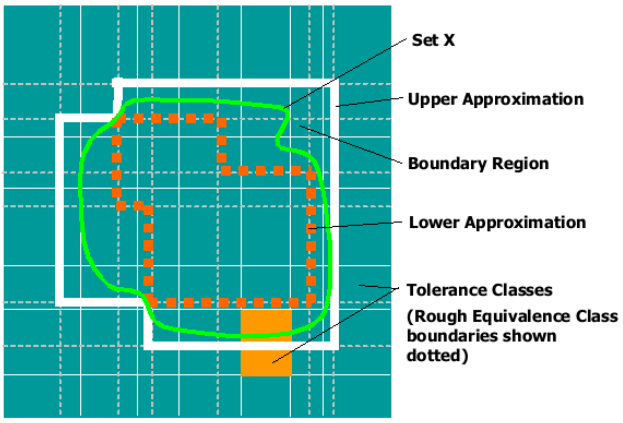

Fig. 2. Tolerance Rough Set Model

sure of similarity of objects. For a subset of features, P, this can be achieved in many ways including the following approaches:

$$(x, y) \in SIM_{P,\tau} \iff \prod_{a \in P} SIM_a(x, y) \geq \tau \tag{15}$$

$$(x, y) \in SIM_{P,\tau} \iff \frac{\sum_{a \in P} SIM_a(x, y)}{\mid P \mid} \geq \tau \tag{16}$$

where τ is a global similarity threshold and determines the required level of similarity for inclusion within a tolerance class. This framework allows for the specific case of traditional rough sets by defining a suitable similarity measure (e.g. complete equality of features and the use of equation (15)) and threshold ($\tau = 1$). Further similarity relations are summarised in [29], but are not included here. From this, the so-called tolerance classes that are generated by a given similarity relation for an object x are defined as:

$$SIM_{P,\tau}(x) = \{y \in U \mid (x, y) \in SIM_{P,\tau}\} \tag{17}$$

Lower and upper approximations can now be defined in a similar way to that of traditional rough set theory:

$$\underline{P_\tau} X = \{x \mid SIM_{P,\tau}(x) \subseteq X\} \tag{18}$$

$$\overline{P_\tau} X = \{x \mid SIM_{P,\tau}(x) \cap X \neq \emptyset\} \tag{19}$$

The tuple $\langle \underline{P_\tau} X,\ \overline{P_\tau} X \rangle$ is known as a tolerance rough set [39]. Using this, the positive region and dependency functions can be defined as follows:

$$POS_{P,\tau}(Q) = \bigcup_{X \in U/Q} \underline{P_\tau} X \tag{20}$$

$$\gamma_{P,\tau}(Q) = \frac{\mid POS_{P,\tau}(Q) \mid}{\mid U \mid} \tag{21}$$

From these definitions, an attribute reduction method can be formulated that uses the tolerance-based degree of dependency, $\gamma_{P,\tau}(Q)$, to measure the significance of feature subsets (in a similar way to the rough set QUICKREDUCT algorithm described previously). Although this allows the consideration of real-valued data, the inclusion of the tolerance threshold (τ) also now means that TRSM departs from the traditional rough set approach which requires no additional thresholding information.

3 Distance Metric-Assisted Tolerance Rough Set Feature Selection

The Distance Metric-Assisted Tolerance Rough Set Feature Selection (DM-TRS) is an extension of the TRSM approach described previously which has the ability to operate on real-valued data. It marries the TRSM with the distance metric assisted rough set approaches. This allows the information of the TRSM boundary region that is otherwise ignored to be examined and used for FS. This ability to deal with real-valued data along with the consideration of the uncertain boundary region information allows a more flexible approach for FS.

*3.1 Distance Metric-based Tolerance*QUICKREDUCT

Following the outline of TRSM in Section II, a similarity relation is defined on all features using (16). Employing the already defined tolerance lower and upper approximations (see (18) & (19)) definition the boundary region can be computed:

$$BND_{P,\tau}(Q) = \bigcup_{X \in U/Q} \underline{P_\tau} X - \bigcup_{X \in U/Q} \overline{P_\tau} X \tag{22}$$

This and the similarity relation form the principal concepts required for the application of the distance metric. However, in an attempt to quantify the value of the boundary region objects, a metric is required. As argued previously in the intuitive

sense, by introducing the *P*-lower approximation mean, the distance function for the calculation of the proximity of objects in the boundary region can be formulated:

$$\delta_P(\underline{P_\tau} X_{MEAN}, y), \quad y \in BND_{P_\tau}(Q) \tag{23}$$

Once again, various distance metrics can be employed for this distance function. To measure the quality of the boundary region, a significance value ω is obtained by measuring all of the distances of the objects and combining them such that:

$$\omega_P(Q) = \left(\sum_{y \in BND_P(Q))} \delta_P(\underline{P} X_{MEAN}, y) \right)^{-1} \tag{24}$$

An alternative to the mean lower approximation and distance metric is another approach which uses the ***Hausdorff*** metric to calculate the distance between non-empty sets. It measures the extent to which each point in a set is located relative to those of another set. The ***Hausdorff*** metric has been applied to facial recognition [36], image processing [37] and FS [31] with much success. It can be defined as:

$$h(A, B) = max_{a \in A}\{min_{b \in B}\{d(a, b)\}\} \tag{25}$$

where a and b are points (objects) of sets A and B respectively, and $d(a, b)$ is any metric between these points. A basic implementation of this has been incorporated into the above framework using Euclidean distance as a metric. Experimentation using this approach can be found in section 4. The primary disadvantage of this approach however is the computational overhead involved in calculating the distance of all objects in the boundary region from each other. For n boundary region objects, this means that $O(n^2)$ distance calculations must be made, unlike the mean lower approximation which results in $O(n)$ distance calculations.

As with the previously described rough set-based method the significance measure takes values in the interval $[0, 1]$. This measure can now be combined with the tolerance rough set dependency value and used to gauge the utility of attribute subsets, using exactly the same mechanism as defined in (13). It should perhaps be emphasised at this point that no transformation of the data takes place (in contrast to approaches such as [12,32]) and that the distance measure is only used in conjunction with the dependency value to form the evaluation metric. This ensures that this method is stable and will always return the same subset of features for a given training dataset.

Incidentally, it is worth indicating that although conceptually similar to the work presented in [44,45], this research focuses on real-valued data entries rather than

image information retrieval. The ability of the proposed approach in handling problems captured in data tables will be demonstrated both with a worked example and nine real-valued datasets later.

DMTQuickReduct shows a distance metric tolerance rough set (DM-TRS) algorithm, that implements the ideas presented above, based on the previously described *DMQuickReduct* algorithm.

Algorithm *DMTQuickReduct*
Input: $\mathbb{C}$, the set of all conditional features
Input: $\mathbb{D}$, the set of all decisional features
Output: R, a feature subset
1. $R \leftarrow \{\}$, $\mathbb{M}_{best} \leftarrow \{\}$, $\mathbb{M}_{prev} \leftarrow \{\}$
2. $\mathbb{M} \leftarrow 0$, $\gamma'_{best} \leftarrow 0$, $\gamma'_{prev} \leftarrow 0$
3. **repeat**
4. $T \leftarrow R$
5. $\mathbb{M}_{prev} \leftarrow \mathbb{M}_{best}$
6. $\gamma'_{prev} \leftarrow \gamma'_{best}$
7. $\forall x \in (\mathbb{C} - R)$
8. **if** $\mathbb{M}_{R \cup \{x\}}(D) > \mathbb{M}_T(D)$
9. $T \leftarrow R \cup \{x\}$
10. $\mathbb{M}_{best} \leftarrow \mathbb{M}_T(D)$
11. $\gamma'_{best} \leftarrow \gamma'_T(D)$
12. $R \leftarrow T$
13. **until** $\gamma_R(\mathbb{D}) = \gamma_{\mathbb{C}}(\mathbb{D})$
14. **return** R

The algorithm employs the combined significance and dependency value $\mathbb{M}$ to choose which features to add to the current reduct candidate. The metric $\mathbb{M}$ is only used to select subsets. The termination criterion is the tolerance rough set dependency value; the algorithm terminates when the addition of any single remaining feature does not result in an increase in the dependency.

Whereas the combined evaluation metric determines the utility of each subset, the stopping criteria is automatically defined through the use of the dependency measure and the subset search is complete either; when the addition of further features does not result in an increase in dependency or when it is equal to 1.

3.2 A Worked Example

To illustrate the operation of the new distance metric-based approach which combines the tolerance rough set and distance metric methods a small example dataset is considered, containing real-valued conditional attributes and crisp decision attributes.

Table 1 contains six objects. It has three real-valued conditional attributes and a single crisp-valued decision attribute. For this example, the similarity measure is the same as that given in (15) for all conditional attributes, with $\tau = 0.8$. The choice of this threshold allows attribute values to differ to a limited degree, with close values considered as though they are identical.

Table 1
Example dataset

Object	a	b	c	f
0	-0.4	-0.3	-0.5	no
1	-0.4	0.2	-0.1	yes
2	-0.3	-0.4	-0.3	no
3	0.3	-0.3	0	yes
4	0.2	-0.3	0	yes
5	0.2	0	0	no

Thus by making $A = \{a\}, B = \{b\}, C = \{c\}$ and $F = \{f\}$, the following tolerance classes are generated:

$$
\begin{aligned}
\mathbb{U}/SIM_{A,\tau} &= \{\{0,1,2\},\{3,4,5\}\} \\
\mathbb{U}/SIM_{B,\tau} &= \{\{0,2,3,4\},\{1\},\{5\}\} \\
\mathbb{U}/SIM_{C,\tau} &= \{\{0\},\{1\},\{3,4,5\},\{2\}\} \\
\mathbb{U}/SIM_{F,\tau} &= \{\{0,2,5\},\{1,3,4\}\} \\
\mathbb{U}/SIM_{\{a,b\},\tau} &= \{\{0,2\},\{1\},\{3,4\},\{3,4,5\},\{4,5\}\} \\
\mathbb{U}/SIM_{\{a,c\},\tau} &= \{\{0\},\{1\},\{2\},\{3,4,5\},\{3,4,5\}\} \\
\mathbb{U}/SIM_{\{b,c\},\tau} &= \{\{0,2\},\{1\},\{3,4\},\{5\}\} \\
\mathbb{U}/SIM_{\{a,b,c\},\tau} &= \{\{0\},\{1\},\{2\},\{3,4\},\{4,5\}\}
\end{aligned}
$$

It is apparent that some objects belong to more than one tolerance class. This is a result of employing a similarity measure rather than the strict equivalence of the conventional rough set model. Using these partitions, a degree of dependency can be calculated for attribute subsets, providing an evaluation of their significance in the same way as previously outlined for the crisp rough case.

The *DMTQuickReduct* algorithm described previously can now be employed. It considers the addition of attributes to the stored best current subset (initially the empty set) and selects the feature that results in the greatest increase of the dependency degree. Considering attribute b, the lower approximations of the decision classes are calculated as follows:

$\underline{B_\tau}$ {0,2,5} = $\{x \mid SIM_{B,\tau}(x) \subseteq \{0,2,5\}\} = \{5\}$

$\underline{B_\tau}$ {1,3,4} = $\{x \mid SIM_{B,\tau}(x) \subseteq \{0,2,5\}\} = \{1\}$

Also the upper approximations:

$\overline{B_\tau}$ {0,2,5} = $\{x \mid SIM_{B,\tau}(x) \cap \{0,2,5\}\} = \{0,2,5\}$

$\overline{B_\tau}$ {1,3,4} = $\{x \mid SIM_{B,\tau}(x) \cap \{0,2,5\}\} = \{1,3,4\}$

From the lower approximation, the positive and boundary regions can then be generated:

$$POS_{B,\tau}(F) = \bigcup_{X \in U/F} \underline{B_\tau} X = \{1,5\}$$

$$BND_{B,\tau}(F) = \bigcup_{X \in U/F} \overline{B_\tau} X - POS_{B,\tau}(F) = \{0,2,3,4\}$$

To calculate the distances of the boundary objects from the lower approximation, it is necessary to generate a lower approximation mean object as described previously:

$$\underline{P}X_{MEAN} = \left\{ \frac{\sum_{o \in \underline{P}X} a(o)}{\mid \underline{P}X \mid} : \forall a \in P \right\}$$

$$= \left\{ \frac{\sum a(1), a(5)}{\mid 2 \mid} \right\} = 0.1$$

There are many distance metrics which can be applied to measure the distance of the objects in the boundary from the lower approximation mean. For simplicity, a variation of Euclidean distance is used in the approach documented here, and this is defined as:

$$\delta_P(\underline{P}X_{MEAN}, y) = \sqrt{\sum_{a \in P} f_a(\underline{P}X_{MEAN}, y)^2} \tag{26}$$

where:

$$f_a(x, y) = a(x) - a(y) \tag{27}$$

From this, the distances of all of the objects in the boundary region in relation to the lower approximation mean can now be calculated:

$$\begin{array}{llll} obj & 0 & \sqrt{f_b(\underline{P}X_{MEAN}, 0)^2} = 0.4 \\ obj & 2 & \sqrt{f_b(\underline{P}X_{MEAN}, 2)^2} = 0.5 \\ obj & 3 & \sqrt{f_b(\underline{P}X_{MEAN}, 3)^2} = 0.4 \\ obj & 4 & \sqrt{f_b(\underline{P}X_{MEAN}, 4)^2} = 0.4 \end{array}$$

The significance value is therefore:

$$\begin{aligned} \omega_B(F) &= \left(\textstyle\sum_{y \in BND_P(Q))} \delta_P(\underline{P}X_{MEAN}, y)\right)^{-1} \\ &= \left(\textstyle\sum (0.4, 0.5, 0.4, 0.4)\right)^{-1} = 0.588 \end{aligned}$$

The significance value is combined with the rough set dependency to form a subset measure $(\mathbb{M})$ such that the value for $\{b\}$:

$$\mathbb{M}(B) = \frac{\omega_B(F) + \gamma_B(F)}{2} = \frac{0.588 + 0.333}{2} = 0.461$$

By calculating the change in combined significance and dependency value $(\mathbb{M})$ when an attribute is removed from the set of considered conditional attributes, a measure of the goodness of that attribute can be obtained. The greater the change in $\mathbb{M}$ the greater the measure of goodness that attribute has attached to it.

The values for the combined metric can be calculated for all considered subsets of conditional attributes using DMRSAR:

$$\begin{array}{ll} \mathbb{M}_{\{a\}}(\{f\}) = 0.0 & \mathbb{M}_{\{a,c\}}(\{f\}) = 0.498 \\ \mathbb{M}_{\{b\}}(\{f\}) = 0.461 & \mathbb{M}_{\{b,c\}}(\{f\}) = 1.0 \\ \mathbb{M}_{\{c\}}(\{f\}) = 0.805 & \mathbb{M}_{\{a,b,c\}}(\{f\}) = 0.492 \end{array}$$

It is obvious from the above example that the search finds a subset in the manner

$\{c\} \rightarrow \{b, c\}$. As $\{a\}$ and $\{a, c\}$ and also $\{a, b, c\}$ do not result in the same increase in combined metric these subsets are ignored.

3.3 Computational Complexity

As the DMTQUICKREDUCT algorithm is based on a greedy hill-climbing type of search. The computational complexity will be similar to that of other approaches which use this method.

However, in addition to the factors which govern the computational complexity of the rough set QUICKREDUCT algorithm, other factors must also be taken into account. In the DM-TRS approach objects in the boundary region are also considered and this inevitably adds to the computational overhead. Furthermore, all of those objects in the lower approximation are also considered when calculating a collapsed lower approximation object for each concept. At this lower level the additional factors that must be considered (also those that are not employed in the rough set case) include; the calculation of the collapsed lower approximation mean, the calculation of the upper approximation, and the calculation of the distances of objects in the boundary from the collapsed lower approximation mean.

From a high level point-of-view the DMTQUICKREDUCT has an intuitive complexity of $(n^2 + n)/2$ for a dimensionality of n. This is the number of evaluations of the dependency function and distance measure performed in the 'worst case'. For instance if the feature set consists of $\{a_1, a_2\}$, then the DMTQUICKREDUCT algorithm will make 3 evaluations, one each for $\{a_1\}$ and $\{a_2\}$, and finally one for $\{a_1, a_2\}$ in the worst case.

4 Experimentation

This section presents the results of experimental studies using 8 real-valued datasets. These datasets are of the same format as that used for the worked example in the previous section. They are small-to-medium in size, with between 120 and 390 objects per dataset and feature sets ranging from 5 to 39 - a detailed description can be found in the appendix. All datasets have been obtained from [1] and [27] A comparison of both the tolerance rough set algorithm and the distance-metric based tolerance rough set dimensionality reduction techniques is made based on subset size, and classification accuracy. Furthermore, the DM-TRS approach is also compared with five other FS techniques. The comparison is made in terms of both subset size and classification accuracy and also in terms of classification accuracy for each given subset size discovered by the DM-TRS method where applicable.

4.1 Experimental Setup

A range of 4 tolerance values, (0.80–0.95 in intervals of 0.05) were employed when considering the datasets. It should be borne in mind that the ideal tolerance value for any given dataset can only be optimised for that dataset by repeated experimentation. This is true of the TRSM as well as to any extensions applied to it, such as described in this paper. Therefore, the range of values chosen is an attempt to demonstrate the ideal tolerance threshold for each dataset without exhaustive experimentation.

Table 2
Classification Accuracy using QSBA

	QSBA	$\tau = 0.8$		$\tau = 0.85$		$\tau = 0.90$		$\tau = 0.95$	
Dataset	Unred.	TRS	DM-TRS	TRS	DM-TRS	TRS	DM-TRS	TRS	DM-TRS
water 2	57.94	77.76	77.76	73.16	73.16	74.53	74.53	67.79	76.38
water 3	48.97	63.12	63.12	74.34	74.34	73.56	73.56	68.25	63.83
cleveland	37.46	36.51	39.47	35.78	35.78	43.58	46.61	43.28	43.28
glass	43.65	37.60	37.60	38.51	38.51	25.88	25.88	42.12	39.43
heart	64.07	77.41	77.42	73.33	74.07	70.00	70.00	74.81	74.81
ionosphere	80.67	74.34	74.34	68.26	68.26	68.26	69.14	64.10	65.65
olitos	64.16	61.66	64.16	57.50	86.08	61.66	62.36	54.16	60.01
wine	94.86	85.39	85.39	81.40	81.40	84.11	84.11	83.72	84.10

Table 3
Classification Accuracy using JRIP, PART, and J48 Classifiers ($\tau = 0.80$)

	TRS			DM-TRS		
Dataset	JRIP	PART	J48	JRIP	PART	J48
water 2	83.58	84.61	83.58	83.58	84.61	83.58
water 3	84.61	81.80	83.84	84.61	81.80	83.84
cleveland	52.86	52.18	53.19	55.55	53.53	54.20
glass	50.00	49.53	48.13	50.00	49.53	48.13
heart	73.70	78.89	75.56	73.70	78.89	75.56
ionosphere	89.13	88.26	88.26	89.13	88.26	88.26
olitos	67.50	70.00	64.16	65.83	62.50	59.16
wine	95.50	94.38	81.40	81.40	84.11	84.11

4.2 Classifier Learning Methods

In the generation and discussion of results for classification accuracies, a fuzzy classifier learning method QSBA [34], and three other classifier learners - J48, JRip,

Table 4
Classification Accuracy using JRIP, PART, and J48 Classifiers ($\tau = 0.85$)

	TRS			DM-TRS		
Dataset	JRIP	PART	J48	JRIP	PART	J48
water 2	84.61	82.30	84.87	84.61	82.30	84.87
water 3	83.58	82.30	81.02	83.58	82.30	81.02
cleveland	53.87	50.84	54.54	53.87	50.54	54.54
glass	64.95	60.74	68.22	61.93	66.82	68.70
heart	75.55	77.40	82.59	81.85	80.74	82.63
ionosphere	90.42	88.69	86.52	90.42	88.69	86.52
olitos	62.50	60.83	60.00	62.50	65.83	67.50
wine	95.25	95.50	96.06	95.25	95.50	96.06

Table 5
Classification Accuracy using JRIP, PART, and J48 Classifiers ($\tau = 0.90$)

	TRS			DM-TRS		
Dataset	JRIP	PART	J48	JRIP	PART	J48
water 2	85.38	82.30	87.43	85.38	82.30	87.43
water 3	80.00	81.53	76.67	80.00	81.53	76.67
cleveland	54.20	53.87	52.52	54.03	55.55	54.88
glass	65.88	69.15	68.69	65.88	69.15	68.69
heart	79.25	75.19	78.88	79.25	75.19	78.88
ionosphere	85.65	86.52	85.21	86.01	89.56	89.13
olitos	70.00	65.83	61.66	59.17	60.84	67.50
wine	96.06	94.94	96.62	96.06	94.94	96.62

and PART [42] - were employed. These are briefly outlined below.

QSBA [34] works by generating fuzzy rules using the fuzzy subsethood measure for each decision class and a threshold to determine what appears in the rule for that decision class. The fuzzy subsethood measure is then used to act as weights, and the algorithm then modifies the weights to act as fuzzy quantifiers.

J48 [33] creates decision trees by choosing the most informative features and recursively partitioning the data into subtables based on their values. Each node in the tree represents a feature with branches from a node representing the alternative values this feature can take according to the current subtable. Partitioning stops when all data items in the subtable have the same classification. A leaf node is then

Table 6
Classification Accuracy using JRIP, PART, and J48 Classifiers ($\tau = 0.95$)

	TRS			DM-TRS		
Dataset	JRIP	PART	J48	JRIP	PART	J48
water 2	82.82	83.07	82.05	84.10	84.10	80.77
water 3	81.02	80.77	81.02	83.59	78.98	81.80
cleveland	50.54	50.84	54.54	50.54	50.84	54.54
glass	69.62	68.22	69.62	65.42	64.95	62.00
heart	80.38	78.57	81.48	80.38	78.57	81.48
ionosphere	86.08	87.39	87.39	85.93	87.82	87.82
olitos	64.16	66.67	64.16	64.16	65.88	64.16
wine	93.25	95.50	96.02	91.57	94.98	97.19

Table 7
Comparison of Subset size for each tolerance threshold value

	Original	Subset size		Subset size		Subset size		Subset size	
	number	($\tau = 0.8$)		($\tau = 0.85$)		($\tau = 0.90$)		($\tau = 0.95$)	
Dataset	of features	TRS	DM-TRS	TRS	DM-TRS	TRS	DM-TRS	TRS	DM-TRS
water 2	39	6	6	5	5	8	8	12	12*
water 3	39	5	5	9	9	9	9	12	11
cleveland	14	3	2	2	2	11	10	8	8
glass	10	3	3	5	5*	3	3	8	3
heart	14	4	4	6	8	12	12	8	8
ionosphere	34	3	3	6	6	6	6*	8	8*
olitos	25	8	5	5	5*	9	8	6	6*
wine	13	5	5	4	4	5	5	5	5*

** - Denotes subset whose size was the same as TRS but for which different attributes had been selected*

created, and this classification assigned.

JRip [3] learns propositional rules by repeatedly growing rules and pruning them. During the growth phase, antecedents are added greedily until a termination condition is satisfied. Antecedents are then pruned in the next phase subject to a pruning metric. Once the ruleset is generated, a further optimization is performed where rules are evaluated and deleted based on their performance on randomized data.

PART [43] generates rules by means of repeatedly creating partial decision trees from data. The algorithm adopts a divide-and-conquer strategy such that it removes instances covered by the current ruleset during processing. Essentially, a rule is created by building a pruned tree for the current set of instances; the leaf with the highest coverage is promoted to a rule.

4.3 Comparison of Classification Accuracy

The data presented in Table 2 shows the average classification accuracy using the classifiers learned by the four learner methods described previously. The recorded values are expressed as a percentage and obtained using 10-fold cross validation. Classification was initially performed on the unreduced dataset, followed by the reduced datasets, which were obtained by using both the TRS and DM-TRS dimensionality reduction techniques respectively for each of the tolerance values.

Examining the classification values obtained using QSBA, even when the subset size in Table 7 is of a similar value to that of the TRS approach, the corresponding classification figures for DM-TRS demonstrate the selection of better quality subsets. In some cases the DM-TRS approach even manages to select a subset of smaller cardinality for a given dataset, whilst also maintaining a similar level of classification as TRS.

Obviously, where DM-TRS discovers identical subsets to those found by TRS, the classification accuracies will also be identical. Where this is not the case however, the results can differ substantially depending on whether fuzzy or crisp classifiers have been employed in obtaining the results e.g. for the *water 3* dataset with $(\tau = 0.95)$, the crisp classifiers show an average result for DM-TRS that is better than TRS, whilst the fuzzy classifier shows a result that is poorer than TRS. For the same tolerance value (0.95), the *glass* dataset, also demonstrates a small decrease in the order of up to 7% (for all classifiers), however when the corresponding decrease in dimensionality of 37.5% is considered over the TRS method, this decrease is not significant. In all other cases where the crisp classifiers show a decrease in classification accuracy, this is reflected as an increase when QSBA is employed for classification. This is due mainly to the fact that although J48, JRip, and PART are intended to handle real-valued data, they are unable to examine data in the same way that a fuzzy classifier learner such as QSBA can.

4.4 Subset size

Table 7 presents the results of a comparison of subset size, for both the TRS and DM-TRS approaches, with DM-TRS showing a small but clear advantage in terms of more compact subsets.

Examining the results in Table 7, the DM-TRS method shows that there is much information contained in the boundary region of a tolerance rough set. This is reflected in the subset sizes obtained. DM-TRS succeeds in finding subsets of cardinality that are at least equal and sometimes smaller than those obtained using the TRS method, with the exception of the *heart* dataset for $\tau = 0.85$. However if the classification results are examined closely, it is clear that although the subset size

is of greater cardinality for this particular case, the subset is of greater quality than that obtained using TRS. The results also demonstrate that the nature of the data along with a particular value of τ can mean that there is little or no information in the boundary region and therefore DM-TRS relies purely on the information contained in the lower approximation dependency value. This can in-turn result in subsets that are identical to those discovered by the TRS method.

Whilst it may be expected that a change in τ would reflect a change in performance in terms of subset size for the TRS method such that an optimal value is arrived at after a period of experimentation. The results for subset size demonstrate an interesting trend where the DM-TRS method may discover smaller subset sizes than TRS. As the DM-TRS method examines the boundary region information, it would be expected that a decrease in τ (thereby increasing the number of objects in the lower approximation and decreasing the number of objects in the boundary region) would result in the DM-TRS performing poorly for the next decrement of threshold value documented above – as there is less information contained in the boundary region for the DM-TRS method to examine. However, if the results in Table 7 and 2 are examined for e.g. the dataset *olitos*, it can be seen that DM-TRS selects subsets which are of smaller size and in some cases of better quality. This suggests that, as long as there is some information in the boundary region, regardless of whether the optimal value of τ has been obtained, DM-TRS can select subsets of better quality than TRS.

4.5 Comparison with Randomly Selected Subsets

The FS process helps to remove measurement noise as a positive by-product. A valid question therefore is whether other subsets of dimensionality 5 (e.g. for the "water 2" dataset) would perform similarly as those identified by DM-TRS selection. To avoid a biased answer to this, and without resorting to exhaustive computation 30 sets of five features have been randomly chosen in order to see what classification results might be achieved.

Figure 3 shows the error rate of the corresponding 30 classifiers, along with the error rate of the classifier that uses the DM-TRS selected subset. The average error of the classifiers that each employ five randomly selected features is 22.32%, far higher than that attained by the classifier which utilises the DM-TRS selected subset of the same dimensionality. This implies that those randomly selected entail important information loss in the course of feature selection; this is not the case for the DM-TRS selection-based approach.

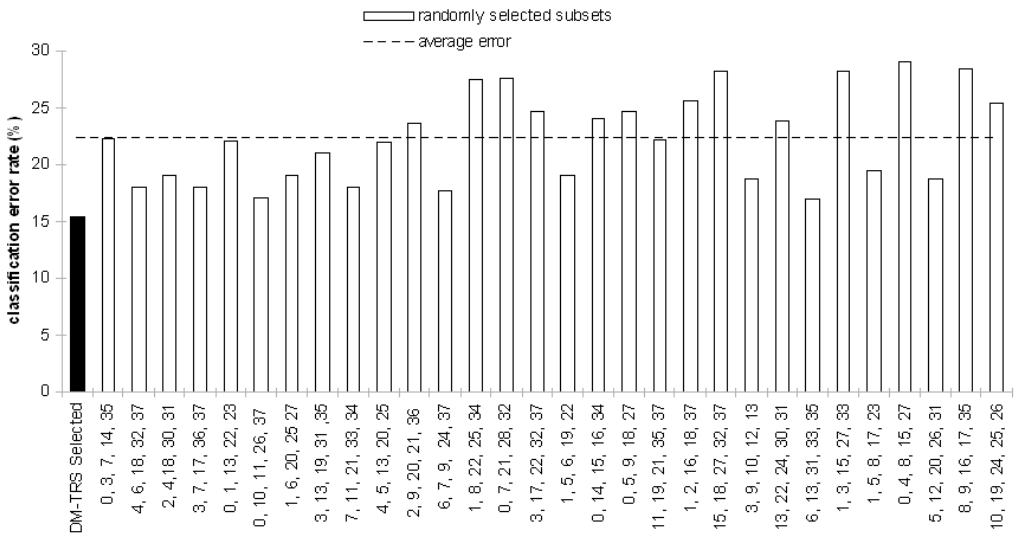

Fig. 3. DM-TRS vs. randomly selected subsets

4.6 Hausdorff Metric Implementation

The Hausdorff metric approach to distance measurement has been described previously as an alternative to the mean lower approximation and Euclidean distance based method which was used to generate the empirical results described previously.

The DM-TRS approach was augmented with the Hausdorff metric to measure the distance between the lower approximation and the boundary region was implemented in order to investigate the performance of this method in terms of subset size. The results of this investigation are included here in Table 8. For brevity only the results for a single tolerance value are included here.

It is apparent that this particular implementation of the *Hausdorff* metric fails to capture the useful information of the boundary region in the same way that the mean lower approximation method does. Examining the results for subset size, it can be seen that the existing DMRSAR approach returns superior results in all cases. This approach took a considerable length of time to run, however this was to be expected as there are a large number of distance calculations performed even for small datasets (exponential $O(n^2)$ for n upper approximation objects).

4.7 Comparison of DM-TRS with existing FS methods

It is appreciated that there are many other FS methods with which DM-TRS could be compared e.g. [8], such examples however are focussed primarily on text clas-

Table 8
DMRSAR – Hausdorff Metric Implementation (τ =0.90)

Dataset	DM-TRS Subset Size	Hausdorff Metric Subset Size
water2	8	10
water3	9	32
cleveland	10	12
glass	3	9
heart	12	13
ionosphere	6	16
olitos	8	14
wine	5	13

sification. The motivation for the development of the DM-TRS method lies in its ability operate on real-valued domains, although it can also handle discrete data.

In this section further comparison of DM-TRS with some of the more traditional dimensionality reduction and FS techniques demonstrates the approach in a more comprehensive manner. DM-TRS is compared with principal component analysis (PCA) [12], ReliefF [19], CFS [13], consistency-based FS [42], and a wrapper method employing J48 [33] as an evaluation metric.

4.7.1 PCA

PCA is a versatile transformation-based DR technique which projects the data onto a new coordinate system of reduced dimensions. This process of linear transformation however also transforms the underlying semantics or meaning of the data. This results in data that difficult for humans to interpret, but which may still provide useful automatic classification of new data. In order to ensure that the comparison of DM-TRS and PCA is balanced, the same subset sizes discovered for each dataset and tolerance level are also employed in the analysis of PCA, e.g. *olitos* in Table 7 has subsets of size 5, 6, and 8. Each of the best number of transformed features are utilised for PCA, (in this case the best 5, 6, and 8).

The results in Table 9 show that of the eight datasets only *olitos* demonstrates a consistent decrease in classification accuracy performance for DM-TRS (see future work for further discussion). There are other instances where PCA slightly outperforms the DM-TRS method but this is not consistent and in a majority of cases DM-TRS usually shows equal performance or an increase in classification accuracy.

Table 9
PCA & DM-TRS – Comparison of Classification Accuracy

subset size		PCA			DM-TRS		
		J48	JRIP	PART	J48	JRIP	PART
water 2	5	83.33	83.84	83.07	84.61	82.30	84.87
	6	86.41	85.38	87.69	84.87	84.61	82.30
	8	81.02	83.58	83.33	85.38	82.30	87.43
	12	85.89	84.36	81.28	84.10	84.10	80.77
water 3	5	87.94	85.64	83.58	84.61	81.80	83.84
	9	82.30	84.36	81.35	83.58	82.30	81.02
	11	84.35	85.38	83.07	83.59	78.98	81.80
cleveland	2	58.92	53.87	57.23	55.55	53.53	54.20
	8	56.90	57.91	54.20	50.54	50.84	54.54
	10	51.85	52.18	50.16	54.03	55.55	54.88
glass	3	64.48	61.68	65.42	65.88	69.15	68.69
	5	68.61	61.21	66.35	61.93	66.82	68.70
heart	4	82.96	82.59	82.96	73.70	78.89	75.56
	8	79.25	83.33	79.62	81.85	80.74	82.63
	12	82.59	84.07	78.14	79.25	75.19	78.88
ionosphere	3	77.39	77.39	79.56	89.13	88.26	88.26
	6	83.04	86.08	79.56	90.42	88.69	86.52
	8	82.60	85.21	82.17	85.93	87.82	87.82
olitos	5	85.00	80.00	82.50	62.50	65.83	75.56
	6	85.00	81.66	81.66	64.16	65.88	64.16
	8	80.33	75.00	80.33	59.17	60.84	67.50
wine	4	93.25	92.69	93.82	95.25	95.50	96.06
	5	93.25	89.88	94.38	96.06	94.94	96.62

It should be emphasised however that while PCA might outperform DM-TRS in some instances in terms of classification accuracy, the semantics of the data is irreversibly transformed following dimensionality reduction. This can have consequences where human interpretability of the data is important, which is one of the key reasons for performing feature selection tasks to begin with. As DM-TRS is a *feature selection* approach as opposed to a *feature ranking* method, a predefined threshold is not required; selection is complete as soon as the termination criterion

(rough set dependency) is fulfilled. The rough set dependency value is integral to the selection process and as such, in contrast to PCA does not need to be predefined.

Finally, it is worth noting that PCA is selected for comparison here due to recognition of the fact that it is an established approach for dimensionality reduction. However, such comparison uses PCA as a global step prior to classification. This may not maximise the potential of PCA serving as a powerful feature reduction tool. It may be a better approach to include PCA as an intrinsic substep of LDA [32], [15]. However, the FS method employed here is a global preprocessor for semantics-preserving dimensionality reduction and hence PCA is examined in a similar manner.

4.7.2 CFS - Correlation-based Feature Selection

CFS [13] is a filter-based approach to FS and uses a search algorithm along with an evaluation metric to decide on the 'goodness' or merit of potential feature subsets. Rather than scoring (and ranking) individual features, the method scores (and ranks) the worth of subsets of features. As the feature subset space is usually large, CFS employs a best-first-search heuristic. This heuristic algorithm takes into account the usefulness of individual features for predicting the class along with the level of intercorrelation amongst features using the premise that good feature subsets contain features that are highly correlated to the class, yet not correlated to each other. CFS calculates a matrix of feature-to-class and feature-to-feature correlations from the training data.

The subset generation technique employed in this case was a greedy-hillclimbing type similar to DM-TRS, where features are added greedily until the termination criteria is fulfilled. The results for subset size and classification values for the three classifier learners are illustrated in Table 10.

Table 10
CFS Subset size and Classification Accuracy

Dataset	subset size	JRIP	PART	J48
water 2	9	83.33	83.07	84.61
water 3	11	82.30	82.05	81.79
cleveland	7	55.54	57.91	58.92
glass	7	65.42	68.69	69.15
heart	7	77.40	77.03	81.11
ionosphere	11	90.00	90.00	90.00
olitos	16	69.16	71.67	69.16
wine	11	94.38	93.82	94.38

Unlike DM-TRS, CFS has no tunable parameters which means that it can be quite difficult to compare the results of Tables 3–7 with those obtained here. It would be easy just to pick the optimal result for DM-TRS and state that the approach is better based on those performance figures. Two different approaches have therefore been adopted. The first approach is to obtain a mean for all of the subset sizes and classification values for DM-TRS for all values of τ and compare these with CFS. The second is to compare CFS and DM-TRS by finding a subset size in the results for DM-TRS that is comparable to that obtained by CFS and use the associated classification figures. So, if CFS has a subset size of 10 for a particular dataset, find a subset of identical or similar size in the DM-TRS results in Table 7 and use this to compare classification accuracy.

Table 11
Average Subset size and Classification Accuracy for DM-TRS

Dataset	subset size	JRIP	PART	J48
water 2	7.75	84.29	83.32	84.16
water 3	8.50	82.94	81.15	80.83
cleveland	5.50	53.49	52.61	54.54
glass	3.5	60.80	62.61	61.88
heart	8	78.79	78.34	79.63
ionosphere	5.75	87.87	68.83	87.93
olitos	6	62.91	63.76	64.58
wine	4.75	91.07	92.38	93.49

The results for CFS when compared with the mean values for DM-TRS demonstrate that the DM-TRS method has a clear advantage in terms of subset size. The only exception perhaps is the result for the *heart* dataset, however if Table 7 is examined, it can be seen that DM-TRS is capable of reducing this value to *4*. The mean classification values for DM-TRS although not as clear as those for subset size show that the difference in classification accuracy between both approaches is less than 8% even in the most extreme cases e.g. *olitos* and *glass*. It must be remembered however that the figures are *mean* values, and that DM-TRS outperforms CFS in many of the examples for individual values of τ.

The second approach to comparing CFS with DM-TRS uses information which is derived from Tables 3–7, perhaps most apparent is the fact that DM-TRS on the whole selects subsets which are more compact than those selected by CFS. The classification values tell a similar story, however some values are lower than those obtained by CFS. The reason for this is related to the fact that suboptimal results must be chosen in order to find a way to compare this approach with CFS, e.g. the *glass* dataset shows comparable classification results to the values recorded in Table 12 as it does in Table 4 and Table 7 but with a subset size of only *5*. Thus

Table 12
Closest Comparable Subset size and Classification Accuracy for DM-TRS

Dataset	subset size	JRIP	PART	J48
water 2	8*	85.38	82.30	87.43
water 3	11	83.59	78.98	81.80
cleveland	8*	50.54	50.84	54.54
glass	5	61.93	66.82	68.70
heart	8*	81.85	80.74	82.63
ionosphere	8*	85.93	87.82	87.82
olitos	8*	59.17	60.84	67.50
wine	5*	96.06	94.94	96.62

** - Denotes subset whose size was not identical to that obtained by CFS but represents the closest available value*

it achieves greater reduction in dimensionality yet retains the classification ability, and easily outperforms CFS.

4.7.3 Consistency-based Feature Selection

Consistency-based feature selection [5] employs a consistency measure for objects in a dataset. Consistency is measured by comparing the values of a given feature set over a set of objects. There are three steps necessary to calculate the consistency rate for a set of objects: a) Consider two objects where the feature values of both are identical but their respective decision feature classes are not, e.g *object1* =$\{1$ 0 1 a$\}$, and *object2* = $\{1$ 0 1 b$\}$, (where $a \neq b$) in this case objects 1 and 2 are said to be inconsistent; b) The inconsistency count for an object is the number of times objects with the same feature values appear in the dataset minus the largest number amongst different decision feature classes, e.g. for n objects with identical decision feature values for which $o1$ objects belong to the $d1$ decision feature class, $o2$ to the $d2$ decision feature class, and $o3$ to the $d3$ decision feature class, and $d1 + d2 + d3 = n$ Assume that $d2$ is the greatest of all three, the consistency count can be calculated as: $n - d2$; c) The consistency rate can then be calculated by summing the consistency counts for the number of groups of objects of given feature values of a subset, divided by the total number of objects.

The FS approach used in this consistency-based method employs a greedy stepwise subset generation technique similar to that of DM-TRS. Again, as with CFS, this method has no tunable parameters, and must be compared with DM-TRS in the same manner as that employed in the previous subsection.

Examining the results in Table 14 and comparing them with those of Table 11 it is

Table 13
Subset size and Classification Accuracy results for consistency based FS

Dataset	subset size	JRIP	PART	J48
water 2	14	84.35	85.60	83.58
water 3	11	83.84	82.56	81.02
cleveland	9	54.54	55.21	56.22
glass	7	65.42	71.96	64.48
heart	10	78.88	74.04	78.88
ionosphere	7	89.56	88.69	89.56
olitos	11	67.50	65.00	68.33
wine	5	90.43	97.19	97.12

clear that like CFS, the subset sizes obtained for consistency-based FS are greater than the average result obtained using DM-TRS. The classification results show similar performance to CFS with some insignificant increases or decreases with respect to certain datasets, but overall comparable to DM-TRS.

4.7.4 ReliefF

ReliefF [19] is an extension of Relief [18] but which has the ability to deal with multiple decision classes. In ReliefF each feature is given a relevance weighting that reflects its ability to discern between the decision class labels. The first threshold, specifies the number of sampled objects used for constructing the weights. For each sampling, an object x is randomly chosen, and its 'nearHit' and 'nearMiss' are calculated. These are x's nearest objects with the same class label and different class label respectively. The user has to supply a threshold which determines the level of relevance that features must surpass in order to be finally chosen.

ReliefF is typically used in conjunction with a feature ranking method employed for the selection of features. In this experimental comparison, the number of nearest neighbours for feature estimation was set to 10, and the other parameter ReliefF requires namely sigma or the influence of nearest neighbours was set to 2. The number of features to select was applied according to the optimal subset sizes obtained for DM-TRS.

The classification results obtained show that despite the improved search method employed by ReliefF, the DM-TRS classification accuracies are comparable with little difference or even a small increase in most cases for DM-TRS.

Table 14
Subset size and Classification Accuracy results for ReliefF

Dataset	(predefined) subset size	JRIP	PART	J48
water 2	5	83.33	84.61	84.10
water 3	5	83.84	81.02	81.53
cleveland	2	58.24	58.21	53.87
glass	3	68.22	68.69	65.42
heart	4	78.50	77.77	78.51
ionosphere	7	86.02	87.82	86.52
olitos	5	65.00	70.03	65.00
wine	4	91.00	93.82	89.87

4.7.5 Wrapper FS employing J48

Although DM-TRS is a filter type FS method, it is interesting to compare it with wrapper-based FS techniques also. Having recognised this, a comparison of the performance of DM-TRS with that of C4.5 [33] which is one of the well known wrapper methods is presented here.

To compare these two approaches meaningfully, the 8 datasets were divided into training and test data respectively. This was accomplished by removing half of the objects from the original data at random and using this data as 'test' data whilst the remainder is used as 'training' data. The results illustrated in Table 15 show the classification accuracies recorded having performed FS on the 'test' data.

Table 15
Subset size and Classification Accuracy results for consistency based FS

	C4.5 Wrapper			DM-TRS		
Dataset	JRIP	PART	J48	JRIP	PART	J48
water 2	90.76	91.28	89.74	90.88	91.65	90.10
water 3	83.84	81.02	81.53	88.71	84.61	86.67
cleveland	51.67	47.65	53.60	52.03	54.05	56.67
glass	78.50	74.76	82.24	79.86	74.76	83.85
heart	75.37	76.86	77.61	77.03	77.77	80.27
ionosphere	86.08	85.21	84.34	90.63	92.45	94.44
olitos	61.66	71.66	63.33	65.33	71.78	65.00
wine	88.76	88.76	87.64	96.62	96.62	92.13

One would expect that the wrapper should outperform any filter method in terms

of classification accuracy as the validation step is carried out using a classifier. The results demonstrate however that this is not strictly the case, and DM-TRS shows a clear increase in classification accuracy over the wrapper method. The increase is small and in some cases in the order of a few percent, but the wrapper method has an extremely high computational overhead. This means that execution times are considerably affected as a result.

5 Conclusions

Comparison of both TRS and DM-TRS has shown that there is often much information to be extracted from the boundary region of tolerance rough sets. Careful selection of the tolerance threshold value is important, as decreasing this value obviously allows further relaxation of the membership of objects to tolerance classes. If this value is relaxed excessively, there may be no information or insufficient boundary region information available, for the DM-TRS method to be effective. It should be stressed however that if this stage has been reached, much information has already been lost and the TRSM alone will perform poorly in any case.

Classification accuracy results have been shown to be similar to those of TRS, and in some cases the DM-TRS method has even shown an increase whilst also simultaneously demonstrating a reduction in dimensionality. Where a decrease has been observed in relation to TRS, in the majority of cases it has been small and, as discussed previously, the actual decrease is not significant as it is usually as a result of fewer features having been selected.

When comparing DM-TRS to the more traditional FS methods, it is clear that DM-TRS offers many performance improvements particularly in terms of dimensionality reduction. Comparison with the CFS [13] and consistency-based FS [5] in particular demonstrate this. When compared with the wrapper approach it is also apparent that DM-TRS also has a significant contribution in terms of classification accuracy.

There are some aspects of the DM-TRS method which require further investigation such that the classification accuracy performance could be improved. In particular the investigation as to why the DM-TRS shows such a significant fall in classification accuracy for the *olitos* dataset when compared with PCA. The reason for this may be related to the way in which PCA manages to capture information that is not approximated well by the DM-TRS method. However the TRS method also shows similar results, and this may have more to do with the TRSM as an approach to dealing with data than the DM-TRS approach itself.

The experimental work detailed in this paper did not take advantage of any optimisations that would improve the performance of DMRSAR further. One such

proposed optimisation would be the level of participation of the distance metric. In the current approach both the dependency and distance metrics are allowed equal participation for the selection of subsets. Preliminary work indicates however that smaller subsets can be obtained by weighting the participation of each of the metrics.

Future work would include a re-evaluation of how the mean lower approximation, is formulated. Implementation of a more accurate representation of the lower approximation would also mean that distances of objects in the boundary region could be more accurately measured.

The significance measure which is employed for DM-TRS is basic, and considers the complete boundary region as a single significance value which is expressed as membership of a unary fuzzy set. Through the re-definition of this as a number of fuzzy sets, the boundary region could be quantified more accurately by expressing membership in terms of weights of objects in the boundary in relation to distance from the lower approximation. As well as this fuzzification, the relationship of objects in the boundary region is another area which requires investigation. Examining the correlation of objects and their individual distances, using for instance fuzzy-entropy [20], [25] it may be possible to qualify the individual objects and their information value in relation to the lower approximation.

Other areas worthy of investigation include the distance metric itself and also the application area of the approach. For the worked example described in this paper a Euclidean distance metric is employed. Metrics such as Hausdorff distance, ellipsoid distance, and others could also be considered. Additionally, the distance metric-assisted tolerance rough set approach is equally applicable to other areas such as clustering, and rule induction.

Acknowledgements

The authors wish to thank both the referees and the editor for their invaluable advice in revising this paper.

References

[1] C. Armanino, R. Leardi, S. Lanteri, and, G. Modi, Chemometrics Intelligent Lab Systems, vol. 5, pp. 343–354, 1989.

[2] A. Chouchoulas and Q. Shen, Rough set-aided keyword reduction for text categorisation, Applied Artificial Intelligence, vol. 15, no. 9, pp. 843–873, 2001.

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

[4] M. Dash and H. Liu, Feature Selection for Classification, Intelligent Data Analysis, vol. 1, no. 3, pp. 131–156, 1997.

[5] M. Dash, Huan Liu, and Hiroshi Motoda, Consistency Based Feature Selection, Pacific-Asia Conference on Knowledge Discovery and Data Mining, pp. 98-109, 2000.

[6] J.S. Deogun, V.V. Raghavan, and H. Sever, Exploiting upper approximations in the rough set methodology, Proceedings of the First International Conference on Knowledge Discovery and Data Mining, Quebec, Canada, pp. 1–10, 1995.

[7] D. Dubois, and H. Prade, Putting Rough Sets and Fuzzy Sets Together, Intelligent Decision Support, pp. 203-232, 1992.

[8] G. Forman, An extensive empirical study of feature selection metrics for text classification, Journal of Machine Learning Research, vol. 3, pp. 1289-1305, 2003.

[9] B. Guo, R. I. Damper, S.R. Gunn, and J.D.B. Nelson, A fast separability-based feature-selection method for high-dimensional remotely sensed image classification, Pattern Recognition, vol. 41, no. 5, pp. 1670–1679, 2008.

[10] I. Guyon and A. Elisseeff, An Introduction to Variable and Feature Selection, Journal of Machine Learning Research, vol. 3, pp. 1157–1182, 2003.

[11] I. Guyon, S. Gunn, M. Nikravesh, and L. Zadeh,(eds.), Feature Extraction: Foundations and Applications, Series Studies in Fuzziness and Soft Computing, Physica-Verlag, Springer, 2006.

[12] P. Devijver and J. Kittler, Pattern Recognition: A Statistical Approach, Prentice Hall, 1982.

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

[14] A. Hassanien, Rough set approach for attribute reduction and rule generation: a case of patients with suspected breast cancer, Journal of the American Society for Information Science and Technology, vol. 55, no. 11, pp. 954–962, 2004.

[15] T. Hastie, R. Tibshirani, and J. Friedman, The Elements of Statistcal Learning: Data Mining, Inference, and Prediction, Springer, 2001.

[16] M. Inuiguchi, and M. Tsurumi, Measures Based on Upper Approximations of Rough Sets for Analysis of Attribute Importance and Interaction, International Journal of Innovative Computing Information and Control, vol. 2, no. 1, pp. 1–12, 2006.

[17] R. Jensen, and Q. Shen, Fuzzy-Rough Attribute Reduction with Application to Web Categorization, Fuzzy Sets and Systems, vol. 141, no. 3, pp. 469–485, 2004.

[18] K. Kira and L.A. Rendell, The feature selection problem: Traditional methods and a new algorithm, Proceedings of Ninth National Conference on Artificial Intelligence, pp. 129-134. 1992.

[19] I. Kononenko, Estimating attributes: Analysis and extensions of RELIEF, Proceedings of the European Conference on Machine Learning, L. De Raedt and F. Bergadano (eds.), Springer-Verlag: Catania, pp. 171-182, 1994.

[20] B. Kosko, Fuzzy entropy and conditioning, Information Sciences, vol. 40, no. 2, pp. 165–174, 1986.

[21] H.R. Li, and W.X. Zhang, Applying Indiscernibility Attribute Sets to Knowledge Reduction, Lecture Notes in Artifical Intelligence, pp. 816–821, 2005.

[22] K. Li, and Y. Liu, Rough set based attribute reduction approach in data mining, Proceedings of the 2002 International Conference on Machine Learning and Cybernetics, vol. 1, pp. 60–63, 2002.

[23] J. Liang, S. Yang, and A. Winstanley, Invariant optimal feature selection: A distance discriminant and feature ranking based solution, Pattern Recognition, vol. 41, no. 5, pp. 1429–1439, 2008.

[24] H. Liu, and H. Motada (eds.), Computational Methods of Feature Selection, Chapman and Hall/CRC Data Mining and Knowledge Discovery Series, 2008.

[25] N. Mac Parthaláin, R. Jensen and Q. Shen. Finding fuzzy-rough reducts with fuzzy entropy. Proceedings of the 17th International Conference on Fuzzy Systems (FUZZ-IEEE'08), pp. 1282–1288, 2008.

[26] N. Mac Parthaláin, Q. Shen, and R. Jensen, Distance Measure Assisted Rough Set Feature Selection, Proceedings of the 16th International Conference on Fuzzy Systems (FUZZ-IEEE'07), pp. 1084–1089, 2007.

[27] D.J. Newman, S. Hettich, C.L. Blake, and C.J. Merz, UCI Repository of machine learning databases [http://www.ics.uci.edu/ mlearn/MLRepository.html], Irvine, CA: University of California, Department of Information and Computer Science, 1998.

[28] M. Modrzejewski, Feature selection using rough sets theory, in: Brazdil, P.B. (Ed.), Proceedings of the European Conference on Machine Learning, Vienna, Austria, pp. 213–226, 1993.

[29] S.H. Nguyen, and A. Skowron, Searching for Relational Patterns in Data, Proceedings of the first European Symposium on Principles of Data Mining and Knowledge Discovery, pp. 265-276, 1997.

[30] Z. Pawlak, Rough sets, International Journal of Computing and Information Sciences, vol. 11, pp. 341–356. 1982.

[31] S. Piramuthu, The Hausdorff Distance Measure for Feature Selection in Learning Applications. in Procs of the 32nd Annual International Conference on System Sciences Hawaii, vol. 6, 1999

[32] A.K. Qin, P.N. Suganthan, and M. Loog, Uncorrelated heteroscedastic LDA based on the weighted pairwise Chernoff criterion, Pattern Recognition, vol. 41, no. 5, pp. 613–616, 2005.

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

[34] K.A. Rasmani and Q. Shen, Data-driven fuzzy rule generation and its application for student academic performance evaluation, Applied Intelligence, vol. 25, no. 3, pp. 305–319, 2006.

[35] R. Jensen and Q. Shen, Tolerance-based and Fuzzy-Rough Feature Selection, Proceedings of the 16th International Conference on Fuzzy Systems (FUZZ-IEEE'07), pp. 877–882, 2007.

[36] W. Rucklidge. Efficient Visual Recognition Using the Hausdorff Distance. vol. 1173, Lecture notes in computer science. Springer, 1996.

[37] B Sendov, Hausdorff distance and image processing, Russian Math Surveys, 59 (2), pp. 319-328, 2004.

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

[39] A. Skowron, and J. Stepaniuk, Tolerance Approximation Spaces, Fundamenta Informaticae, vol. 27, pp. 245–253, 1996.

[40] D. Slezak, Various Approaches to Reasoning with Frequency Based Decision Reducts: A Survey, in: L. Polkowski, S. Tsumoto, T.Y. Lin, (eds.), Rough Set Methods and Applications, Heidelberg: Physica-Verlag, pp. 235–285, 2000.

[41] R.W. Swiniarski, and A. Skowron, Rough Set Methods in Feature Selection and Recognition, Pattern Recognition Letters, vol. 24, no. 6, pp. 833–849, 2003.

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

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

[44] K. Wu and K.-H. Yap, Fuzzy SVM for content-based image retrieval: a pseudo-label support vector machine framework, IEEE Computational Intelligence Magazine, vol. 1, no. 2, pp. 10–16, May 2006.

[45] K. Yap and K.-H. Wu, A soft relevance framework in content-based image retrieval systems IEEE Transactions on Circuits and Systems for Video Technology, vol. 15, no. 12, pp. 1557–1568, Dec. 2005.

[46] D. Zhang, S. Chen, and Z. Zhou, Constraint Score: A new filter method for feature selection with pairwise constraints, Pattern Recognition, vol. 41, no. 5, pp. 1440–1451, 2008.

[47] N. Zhong, J. Dong, and S. Ohsuga, Using Rough Sets with Heuristics for Feature Selection, Journal of Intelligent Information Systems, vol. 16, no. 3, pp. 199–214, 2001.

[48] W. Ziarko, Variable Precision Rough Set Model, Journal of Computer and System Sciences, vol. 46, no. 1, pp. 39–59, 1993.

Appendix A - Datasets

This appendix has a short summary of each of the datasets that were used in experimental evaluation.

water 2 and water 3

These datasets have been generated from the data collected during daily sensor measurements in a urban waste water treatment plant. The original dataset has been divided into two datasets; one dataset with a decision feature which has a binary class, and one dataset which has a decision feature with 3 classes. Both datasets have 39 conditional features which are both real-valued and noisy. The objective is to classify the operational state of the plant in order to predict faults. This should be achieved by examining the state variables of the plant sensor data at each stage of the treatment processes.

cleveland

This data has been compiled from a list of patients who were suspected of being at risk from heart disease. The decision feature refers to the risk of heart disease in the patient and is integer valued from 0 (no-risk) to 4(high-risk). The 14 conditional features are both integer and real-valued and consist of categories such as age, sex, chest pains, location of pain, etc. The integer-valued features are coded categories (e.g. male/female = 0/1 etc.)

glass

This study of classification of types of glass was motivated by criminological investigation. Glass which is recovered at the scene of a crime can be used as evidence if it can be correctly identified. This dataset has six types of glass which can be defined in terms of their oxide content (e.g. Fe, Na, etc.). The dataset has 10 conditional features some of which are continuously valued and others which are integer valued. The single decision feature has seven classes numbered 1-7 which relate to the application of the glass (vehicle, domestic, etc.), and the manufacturing process.

heart

This data is from the same family of datasets as ‘cleveland’ (see above). It is also a heart disease dataset and contains similar data to that previously mentioned, however the data source is European rather than north American.

ionosphere

This dataset comprises of data that was collected by a radar system in north America. In terms of structure the dataset consists of 34 conditional features and a single

decisional feature. All conditional features are continuous and real-valued, whilst the decisional feature is binary (‘good’/‘bad’). The radar targets are free electrons in the ionosphere - ‘Good’ radar returns are those which show evidence of some structure in the ionosphere. Whilst ‘Bad’ returns are those which do not; their signals pass through the ionosphere.

olitos

This data has been collected from olive oils which have been produced in four different areas of Italy which have a Protected Denomination of Origin (PDO). The idea is to try to classify the oils relative to chemical and physical characteristics. The dataset consists 25 conditional features which are real-valued and a decision feature which has four classes, each relating to one of the areas of production.

wine

The data contained in this dataset are the result of a chemical analysis of wines which are grown in a region of Italy but derived from three different vineyards. The analysis determines the quantities of 13 constituents (e.g. alcohol level, malic acidity, hue, etc.) found in each of the three types of wines. All 13 conditional features are continuous/real-valued. The single decisional feature is divided into three crisp values, which relate to the vineyards mentioned previously.