An Empirical Comparison of Rough Sets Reducts and Other Filters Approaches for Feature Subset Selection

Adriano Donizete Pila and Maria Carolina Monard

University of São Paulo
Institute of Mathematics and Computer Sciences
Department of Computer Science and Statistics
Laboratory of Computational Intelligence
P.O. Box 668, 13560-970 - São Carlos, SP, Brazil
{pila, mcmonard}@icmc.sc.usp.br

Abstract The Feature Subset Selection — FSS — is an important problem within the Machine Learning area where the learning algorithm is faced with the problem of selecting relevant features while ignoring the rest. Rough Sets Theory is a mathematical tool to deal with vagueness and uncertainty information. This theory has been applied to rule induction in Machine Learning problems where the information is given in the form of a *decision table*. One of the main features of this approach is the *reduct*, which is a minimal feature set that preserves the ability to discern each object from the others. In this work we propose the use of Rough Sets for FSS and present in detail several experiments, results and comparisons using Rough Sets as well as other inducers and tools as a filter approach for the Feature Subset Selection problem. All the experiments where run on real word datasets, most of them obtained from the UCI Repository.

1 Introduction

With the technological evolution, the amount of information that can be gathered and stored increases very rapidly every day. As Artificial Intelligence systems depend strongly on knowledge, which can be obtained from previous information sources, one problem that has to be faced is how to focus on the most relevant information in order of extracting useful knowledge.

In supervised Machine Learning — ML — an induction algorithm is typically presented with a set of training instances, where each instance is described by a vector of feature values and a class label. The task of the induction algorithm (inducer) is to induce a classifier that will be useful in classifying new cases.

One of the main problems in ML is the Feature Subset Selection — FSS — problem, *i.e.* the learning algorithm is faced with the problem of selecting some subset of features upon which to focus its attention, while ignoring the rest [2]. There are several reasons for doing FSS, such as improving the accuracy of the classifiers, improving the comprehensibility of rules generated by symbolic ML algorithms as well as reducing

This work was also published in the Proceedings of Simpósio Ibero-Americano de Reconhecimento de Padrões [1].

the cost of processing huge quantity of data. Basically, there are three approaches in Machine Learning for FSS [3]: *Embedded*, where the FSS process is embedded within the basic induction algorithm; *Filter*, where the FSS is used to filter the features before the induction process occurs; and *Wrapper*, where the induction algorithm is used as a black box, *i.e.* the FSS algorithm exists as a wrapper around the induction algorithm.

In this work we propose Rough Sets reducts as a filter approach for FSS. Rough Sets is a theory introduced by Zdzislaw Pawlak [4] in the early 1980's where the main feature is the *reduct*. A reduct is a minimal subset of features that preserves the ability to discern the examples from each other. In order to compare the results obtained using Rough Sets reduct as a filter approach for FSS, we selected the inducers $\mathcal{C}4.5$, $\mathcal{C}4.5$-rules, $\mathcal{CN}2$, ID3 implemented in $\mathcal{MLC}^{++}$ library [5] as well as the Column Importance facility provided by MineSet$^{\mathrm{TM}}$. To find the Rough Sets reducts we selected Rosetta [6] — Rough Set Toolkit for Analysis of Data — which presents all functionalities needed to perform several tasks using the Rough Sets approach.

This work is organized as follows: Section 2 describes some important concepts about the Rough Sets Theory. Section 3 briefly describes the algorithms and tools used as filters. Section 4 gives a short description of the datasets used in the experiments. Section 5 shows the experimental setup used to run the experiments and Section 6 describes the results obtained from these experiments. Section 7 reports analysis and comparison of results. Finally, Section 8 gives some conclusions.

2 Rough Sets

This section deals with fundamental issues of the Rough Sets theory, which is a theory strongly connected with the field of Machine Learning. The theory was introduced by Zdzislaw Pawlak in the early 1980's [4], and based on this theory one can propose a formal framework for the automated transformation of data into knowledge. Pawlak has shown that the principles for learning by examples can be formulated in the basis of his theory [7,8,9,10]. An important result from this theory is that it simplifies the search for dominating attributes leading to specific properties.

The Rough Set theory is mathematically relatively simple. Despite of this, it has shown its fruitfulness in a variety of knowledge discovery areas. Among these are information retrieval, decision support, machine learning, and knowledge based systems. A wide range of applications use the ideas of the theory. Medical data analysis, aircraft pilot performance evaluation, image processing, and voice recognition are a few examples. In this work we present Rough Sets as a support for selecting relevant features in supervised machine learning problems.

Almost inevitably the database used for ML will contain imperfection, such as noise, unknown values or errors due to inaccurate measuring equipment. The Rough Set theory comes handy for dealing with these types of problems, as it is a tool for handling vagueness and uncertainty inherent to decision situations as shown in [11,12]. In this section, a set of definitions from the world of Rough Sets is given.

2.1 Information System

An *information system* consists of a set of *objects* where each object has a number of *attributes* with *attribute values* related to it. The attributes are the same for all objects, but the attribute values may differ. An information system is thus more or less the same as a relational database (dataset).

Definition 1 (Information System, Decision System).
An Information System *— IS — is an ordered pair $\mathcal{A}$ = (U,A) where U is a nonempty finite set of objects — the* Universe*, and A is a nonempty, finite set of elements called* Attributes*. The elements of the Universe will in the following be referred to as Objects. Every attribute $a \in A$ is a total function $a : U \rightarrow V_a$, where V_a is the set of allowed values for the attribute (its range). A* Decision System *— DS — is an IS $\mathcal{A}$ = (U,A) for which the attributes in A are further classified into disjoint sets of* condition *attributes C and* decision *attributes D.* $(A = C \cup D, C \cap D = \emptyset)$.

2.2 Discerning Objects

The next definition introduces the concept of an *indiscernibility relation*. If such a relation exists between two objects, it means that all their attribute values are identical with respect to the attributes under consideration, and thus cannot be discerned (distinguished) between when considering those attributes.

Definition 2 (Indiscernibility Relation).
With every subset of attributes $B \subseteq A$ in the IS $\mathcal{A}$ = (U, A), an equivalence relation IND(B) is associated, called an Indiscernibility Relation*, which is defined as follows:*

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

By $U/IND(B)$ is meant the set of all equivalence classes in the relation $IND(B)$.

2.3 Discernibility Matrix

A *Discernibility Matrix* is a matrix in which the classes are indexes. In the matrix, the (condition) attributes which can be used to discern between the classes in the corresponding row and column are inserted.

Definition 3 (Discernibility Matrix).
For a set of attributes $B \subseteq A \in \mathcal{A} = (U, A)$, the Discernibility Matrix *is defined as follows:*

$$M_D(B) = \{m_D(i, j)\}_{n \times n}, 1 \leq i, j \leq n = |U/IND(B)| \tag{2}$$

where $m_D(i, j) = \{a \in B | a(E_i) \neq a(E_j)\} for\ i, j = 1, 2, ..., n$

The entry $m_D(i, j)$ in the discernibility matrix is the set of attributes from B that discern object classes $E_i, E_j \in U/IND(B)$.

If some of the classes have the same decision value, one might decide not to discern between these classes. By doing so, attributes are not added to the matrix for classes with the same decision value. This can result in more simplistic rules if any classes have the same decision value.

2.4 Discernibility Functions

Definition 4 (Discernibility Function). *The* Discernibility Function *f(B) of a set of attributes B ⊆ A is*

$$f(B) = \bigwedge_{i,j \in \{1,\ldots,n\}} \bigvee \overline{m}_D(E_i, E_j) \tag{3}$$

where $n = |U/IND(B)|$, and $\bigvee \overline{m}_D(E_i, E_j)$ is the disjunction taken over the set of boolean variables $\overline{m}_D(i,j)$ corresponding to the discernibility matrix element $m_D(i,j)$. The Relative *Discernibility Function f(E,B) of an object class E, attributes B ⊆ A is*

$$f(E, B) = \bigwedge_{j \in \{1,\ldots,n\}} \bigvee \overline{m}(E, E_j) \tag{4}$$

where $n = |U/IND(B)|$.

This implies that the discernibility function $f(B)$ computes the minimal sets of attributes required to discern any equivalence class from all the others. Similarly, the relative discernibility function $f(E, B)$ computes the minimal sets of attributes required to discern a given class E from the others.

Definition 5 (Dispensability). *An attribute a is said to be dispensable or superfluous in B ⊆ A if IND(B) = IND(B-{a}), otherwise the attribute is* indispensable *in B. If all attributes $a \in B$ are indispensable in B, B is called orthogonal.*

2.5 Reducing Representation

The data in the information system can be used to discern classes only to a certain degree. However, not all attributes may be required in order to be able to do so. This is why the next definition is helpful.

Definition 6 (Reduct, Relative Reduct).
A Reduct *of B is a set of attributes $B' \subseteq B$ such that all attributes $a \in B - B'$ are dispensable, and IND(B')=IND(B). The term RED(B) is used to denote de family of reducts of B. The set of prime implicants of the discernibility function f(B) determines the reducts of B. The set of prime implicants of the relative discernibility function f(E,B) determines the relative reducts of B. The term RED(E,B) denotes the family of relative reducts of B for an object class E.*

What this implies is that a relative reduct contains enough information to discern objects in one class from all the other classes in the information system. To find the relative reducts, the discernibility functions are employed and each function is minimized to a sum of products form. The relative reducts are minimal, because each discernibility function was minimized. A minimal (relative) reduct is thus a reduct in which none of the attributes may be removed without removing the reduct property.

As a reduct is a minimal subset of features that preserves the ability to discern objects from each other, in this work we propose its use as a filter method for Feature Subset Selection.

3 Inducers and Tools

The following inducers, also found in the $\mathcal{MLC}^{++}$ library [5], have been used in this work: ID3, a very basic decision tree algorithm; $\mathcal{C}4.5$, one of the ID3 successors with many extensions to the basic one; $\mathcal{C}4.5$-rules, which examines the original decision tree produced by $\mathcal{C}4.5$ and derives from it a set of rules. It is important to note that $\mathcal{C}4.5$-rules does not simply rewrite the tree as a collection of rules but it generalizes the rules by deleting superfluous conditions; and, $\mathcal{CN}2$, a Machine Learning algorithm that induces rules.

The following tools were also used: CI, a "Column Importance facility" provided by MineSet$^{\mathrm{TM}}$ from Silicon Graphics. It is useful for determining how important various features are in making a particular classification; and, Rosetta [6] — Rough Set Toolkit for Analysis of Data — , a tool that presents all functionalities needed to perform some tasks using the Rough Sets approach. Methods for discretization, finding reducts, rules induction and cross-validation are also provided.

In supervised Machine Learning the inducers use a set of training instances where each instance consists of a vector of feature values and a class label . Generally this vector, denoted by $(\mathbf{X}, Y)$, is in the attribute-value format.

Table 3.1 illustrates this organization where a row i refers to the i-th example or instance $\mathbf{X}_i$ and column entries x_{ij} refer to the individual value of the j-th feature f_j of instance i. The column labelled as $class$ refers to the label or classification of that instance.

Table 3.1. Feature-Value or Spreadsheet Format

$\mathbf{X}$	f_1	f_2	...	f_m	$class$
X_1	x_{11}	x_{12}	...	x_{1m}	y_1
X_2	x_{21}	x_{22}	...	x_{2m}	y_2
...	...	...	...	...	...
X_n	x_{n1}	x_{n2}	...	x_{nm}	y_n

By default each dataset recognized by $\mathcal{MLC}^{++}$ needs three separated files with extensions *data*, *test* and *names*, where the *data* and *test* files contain labelled instances of the training and test set respectively. The *names* file defines the scheme that allows parsing these two previous files besides the name and domain for each attribute and for the label. The accuracy of the classifier produced by the inducer is measured on unseen data *i.e.* the test set. More details can be found in [13,14]. The tools CI and Rosetta recognize different datasets format.

4 Datasets

Experiments were conducted on several real world domains. Most datasets are from the UCI Irvine Repository [15]. To assist comparisons, the datasets chosen also have

different type of attributes. They involve continuous attributes, either alone or in combination with nominal attributes, as well as unknown values. Section 4.2 summarizes datasets characteristics.

4.1 General Description

It follows a brief description of all datasets used in this work: the TA dataset consists of evaluation of teaching performance at the Statistics Department of the University of Wisconsin – Madison; the Bupa dataset consists of predicting whether or not a male patient has liver disorders based on various blood tests and the amount of alcohol consumption; In the Pima dataset the problem is to predict whether a patient would test positive for diabetes; the Breast-cancer2 dataset is one of the breast cancer datasets at UCI where the problem is to predict the recurrence or not of breast cancer; In the CMC dataset the problem is to predict the current contraceptive method choice of a woman based on her demographic and socio-economic characteristics; In the Breast-cancer dataset the problem is to predict whether a tissue sample taken from a patient's breast is malignant or benign; In the Smoke dataset the problem is to predict attitude toward restrictions on smoking in the workplace; the Hepatitis dataset is for predicting life expectation of patients with hepatitis; and, the Hungaria dataset is for diagnosing heart diseases.

4.2 Datasets Summary

Table 4.2.1 summarizes the datasets employed in this study. It shows, for each dataset, the number of instances (#Instances), number and percentage of duplicate (appearing more than once) or conflicting (same attribute-value but different class label) instances, number of features (#Features) continuous and nominal, class distribution, the majority error and if the dataset have at least one missing value.

Datasets are presented in ascending order of the number of features, as will be in the remaining tables and graphs.

5 Experimental Setup

A series of experiments were performed, using the algorithms and datasets described respectively in Sections 3 and 4. It is also important to note that the original data has not been pre-processed in any way such as trying to remove or replace missing values or transform continuous attributes in categorical attributes. Furthermore, each individual inducer was run with default setting for all parameters, *i.e.* no attempt was made to tune any inducer.

For each dataset, the performed experiments can be divided into two main steps — Figure 5.1:

Step 1 $\mathcal{C}4.5$, ID3, CI and Rosetta are used as filters.

Step 2 Features selected by each filter in step 1 are used to compute the accuracy of $\mathcal{C}4.5$, $\mathcal{C}4.5$-rules and $\mathcal{CN}2$ inducers.

Table 4.2.1. Datasets Summary Descriptions

Dataset	# Instances	#Duplicate or conflicting (%)	# Features (cont.,nom.)	Class	Class %	Majority Error	Missing Values
ta	151	45 (39.13%)	5 (1,4)	1	32.45%	65.56%	N
				2	33.11%	on value 3	
				3	34.44%		
bupa	345	4 (1.16%)	6 (6,0)	1	42.03%	42.03%	N
				2	57.97%	on value 2	
pima	769	1 (0.13%)	8 (8,0)	0	65.02%	34.98%	N
				1	34.98%	on value 0	
breast-cancer2	285	2 (0.7%)	9 (4,5)	recurrence	29.47%	29.47%	Y
				no-recurrence	70.53%	on value no-recurrence	
cmc	1473	115 (7.81%)	9 (2,7)	1	42.70%	57.30%	N
				2	22.61%	on value 1	
				3	34.69%		
breast-cancer	699	8 (1.15%)	9 (9,0)	2	65.52%	34.48%	Y
				4	34.48%	on value 2	
smoke	2855	29 (1.02%)	13 (2,11)	0	5.29%	30.47%	N
				1	25.18%	on value 2	
				2	69.53%		
hungaria	294	1 (0.34%)	13 (13,0)	presence	36.05%	36.05%	Y
				absence	63.95%	on value absence	
hepatitis	155	0 (0%)	19 (6,13)	die	20.65%	20.65%	Y
				live	79.35%	on value live	

The filter process was conducted as follows: ID3, $\mathcal{C}4.5$, CI and Rosetta were applied as filters for all the datasets described earlier.

It is important to note that when using Rosetta as a filter the result is a set of subsets where each subset is a set of selected features (reducts) and there can be several reducts. Rosetta has a default setting to compute a set of reducts where all resulting reducts have the same ability to discern the examples (objects) from each other. So each reduct is a subset of selected features where the number of selected features may be different.

In this work we decided to select as filter the reduct with the smallest number of features, *i.e.* if Rosetta brought up five different reducts that preserve the same indiscerbility relation of the entire set of features in the dataset, we selected the reduct with less number of features — thus introducing some bias in the experiments. Our choice is based on **Occam's Razor** [16] that says *"prefer the simplest hypothesis that fits the data"*. We expect that selecting the smallest reduct as filter, *i.e.* smallest number of relevant features for the Rough Set approach, will allow the inducers to find more simple rules.

After selecting the smallest reduct, the subset of features of the reduct were used to compute the accuracy for $\mathcal{C}4.5$, $\mathcal{C}4.5$-rules and $\mathcal{CN}2$ inducers.

6 Experimental Results

This section presents the results obtained through these experiments. For each dataset two tables are presented:

1. The first table describes filter selected features. To specify the experiment, it is used the notation
   FSS(*method,inducer*) where:

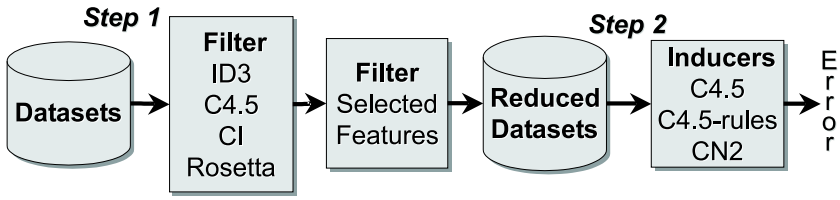

Figure 5.1. Experiment Steps

- *method* $\in \{\text{f}\}$ indicating that filter (f) approach has been used[1];
- *inducer* $\in \{\text{CI}, \mathcal{C}4.5, \text{ID3}, \text{RS}\}$ indicating the algorithm or tool used as filter.

This table shows, for each FSS(*method*,*inducer*), the features subset selected, the number of features in the selected subset (#F) as well as the proportion of selected features (%F.) The features are related to a number that indicates the order of each feature in the original dataset.

2. The second table shows the error of each inducer (mean and standard deviation) using 10-fold cross-validation[2] (10-cv) using all features as well as the features subset selected by each FSS(*method*,*inducer*) considered. Each column represents the inducer used for accuracy estimation and each row represents the feature subset used.

In the corresponding two tables, errors marked with a bullet ($\bullet$) indicate that these errors are grater than the majority class error, considering only the mean error; and errors marked with an up triangle ($\triangle$) indicate that these errors are significantly higher at 95% confidence level.

Table 6.1. TA – Selected Features

Inducer	Selected Features	#F	%F
FSS(f,CI)	0 1 2 3	4	80.00%
FSS(f,$\mathcal{C}4.5$)	0 1 2 3 4	5	100.00%
FSS(f,ID3)	0 1 2 3 4	5	100.00%
FSS(f,RS)	1 2 4	3	60.00%

Table 6.2. TA – Errors

ta 10-cv	$\mathcal{C}4.5$	$\mathcal{CN}2$	$\mathcal{C}4.5$-rules
all features	52.92±6.36	51.67±3.42	53.58±6.00
FSS(f,CI)	51.58±5.41	50.28±3.92	50.25±5.25
FSS(f,$\mathcal{C}4.5$)	52.92±6.36	51.67±3.42	53.58±6.00
FSS(f,ID3)	52.92±6.36	51.67±3.42	53.58±6.00
FSS(f,RS)	54.25±6.19	51.06±4.03	48.33±5.86

[1] Although in this work only method f is treated, we decided to keep the same notation used in [17,18]

[2] A 10-fold cross-validation (cv) is performed by dividing the dataset into 10 mutually exclusive subsets (folds) of cases of approximately equal size. The inducer is trained and tested 10 times, each time tested on a fold and trained on the dataset minus that fold. The cv estimate of accuracy is the average of the estimated accuracies from the 10 folds.

Table 6.3. Bupa – Selected Features

Inducer	Selected Features	#F	%F
FSS(f,CI)	4	1	16.67%
FSS(f,$\mathcal{C}4.5$)	0 1 2 3 4 5	6	100.00%
FSS(f,ID3)	0 1 2 3 4 5	6	100.00%
FSS(f,RS)	0 1 2	3	50.00%

Table 6.4. Bupa – Errors

bupa 10-cv	$\mathcal{C}4.5$	$\mathcal{CN}2$	$\mathcal{C}4.5$-rules
all features	32.70±2.79	35.35±2.01	34.13±2.85
FSS(f,CI)	41.42±2.85△	45.21±1.98●△	41.42±2.85△
FSS(f,$\mathcal{C}4.5$)	32.70±2.79	35.35±2.01	34.13±2.85
FSS(f,ID3)	32.70±2.79	35.35±2.01	34.13±2.85
FSS(f,RS)	43.19±2.18●△	38.53±2.94	42.62±2.49●△

Table 6.5. Pima – Selected Features

Inducer	Selected Features	#F	%F
FSS(f,CI)	0 1 4 5 6 7	6	75.00%
FSS(f,C4.5)	0 1 2 4 5 6 7	7	87.50%
FSS(f,ID3)	0 1 2 3 4 5 6 7	8	100.00%
FSS(f,RS)	1 2 6	3	37.50%

Table 6.6. Pima – Errors

pima 10-cv	$\mathcal{C}4.5$	$\mathcal{CN}2$	$\mathcal{C}4.5$-rules
all features	25.87±1.28	25.12±1.97	25.87±1.07
FSS(f,CI)	26.53±0.73	25.13±1.49	26.53±0.78
FSS(f,$\mathcal{C}4.5$)	25.88±0.99	23.69±1.22	26.39±1.13
FSS(f,ID3)	25.87±1.28	25.12±1.97	25.87±1.07
FSS(f,RS)	27.45±1.57	29.15±1.31△	27.71±1.49

Table 6.7. Breast Cancer2 – Selected Features

Inducer	Selected Features	# F	%F
FSS(f,CI)	1 2 3 4 5 6 7 8	8	88.89%
FSS(f,C4.5)	0 1 3 4 5 6 7 8	8	88.89%
FSS(f,ID3)	0 1 2 3 4 5 6 7 8	9	100.00%
FSS(f,RS)	0 2 3 5 7	5	55.56%

Table 6.8. Breast Cancer2 – Errors

breast-cancer2 10-cv	$\mathcal{C}4.5$	$\mathcal{CN}2$	$\mathcal{C}4.5$-rules
all features	26.66±2.89	27.03±2.29	27.71±1.73
FSS(f,CI)	25.63±2.59	27.71±1.68	29.46±2.48
FSS(f,$\mathcal{C}4.5$)	22.81±2.92	29.16±2.75	24.19±2.37
FSS(f,ID3)	26.66±2.89	27.03±2.29	27.71±1.73
FSS(f,RS)	24.95±1.89	27.75±2.79	25.70±2.37

Table 6.9. Cmc – Selected Features

Inducer	Selected Features	#F	%F
FSS(f,CI)	0 1 2 3 4 5 6 7 8	9	100.00%
FSS(f,C4.5)	0 1 2 3 4 5 6 7 8	9	100.00%
FSS(f,ID3)	0 1 2 3 4 5 6 7 8	9	100.00%
FSS(f,RS)	0 1 2 3 4 5 6 7 8	9	100.00%

Table 6.10. Cmc – Errors

cmc 10-cv	$\mathcal{C}4.5$	$\mathcal{CN}2$	$\mathcal{C}4.5$-rules
all features	47.94±1.49	49.64±1.01	45.90±1.38
FSS(f,CI)	47.94±1.49	49.64±1.01	45.90±1.38
FSS(f,$\mathcal{C}4.5$)	47.94±1.49	49.64±1.01	45.90±1.38
FSS(f,ID3)	47.94±1.49	49.64±1.01	45.90±1.38
FSS(f,RS)	47.94±1.49	49.22±1.05	45.90±1.38

Table 6.11. Breast Cancer – Selected Features

Inducer	Selected Features	#F	%F
FSS(f,CI)	0 1 2 3 4 5 6 7 8	9	100.00%
FSS(f,C4.5)	0 1 2 3 4 5 6 8	8	88.89%
FSS(f,ID3)	0 1 2 3 4 5 6 7	8	88.89%
FSS(f,RS)	0 3 5 6	4	44.44%

Table 6.12. Breast Cancer – Errors

breast-cancer 10-cv	C4.5	CN2	C4.5-rules
all features	5.86±0.84	4.87±0.77	4.29±0.60
FSS(f,CI)	5.86±0.84	4.87±0.77	4.29±0.60
FSS(f,C4.5)	6.01±0.76	4.44±0.61	4.29±0.60
FSS(f,ID3)	5.72±0.74	5.16±0.86	4.86±0.80
FSS(f,RS)	4.86±0.71	6.72±0.79△	4.29±0.67

Table 6.13. Smoke – Selected Features

Inducer	Selected Features	#F	%F
FSS(f,CI)	1 2 3 4 5 6 7 8 9 10 12	11	84.62%
FSS(f,C4.5)	0 1 2 3 4 5 6 7 8 9 10 11 12	13	100.00%
FSS(f,ID3)	0 1 2 3 4 5 6 7 8 9 10 11 12	13	100.00%
FSS(f,RS)	0 2 3 4 5 6 7 8 9 11 12	11	84.62%

Table 6.14. Smoke – Errors

smoke 10-cv	$\mathcal{C}4.5$	$\mathcal{CN}2$	$\mathcal{C}4.5$-rules
all features	31.45±0.93●	32.18±0.64●	32.54±0.68●
FSS(f,CI)	30.47±0.95△	35.02±0.71●△	33.21±0.82●
FSS(f,$\mathcal{C}4.5$)	31.45±0.93●	32.18±0.64●	32.54±0.68●
FSS(f,ID3)	31.45±0.93●	32.18±0.64●	32.54±0.68●
FSS(f,RS)	31.42±0.84●	32.01±0.82●	33.10±1.01●

Table 6.15. Hungaria – Selected Features

Inducer	Selected Features	#F	%F
FSS(f,CI)	1 2 4 5 6 7 8 9 11 12	10	76.92%
FSS(f,$\mathcal{C}4.5$)	0 1 2 3 4 5 6 7 8 9 10	11	84.62%
FSS(f,ID3)	0 1 2 3 4 5 7 8 9 10 12	11	84.62%
FSS(f,RS-b)	4 7 9	3	23.07%

Table 6.16. Hungaria – Errors

hungaria 10-cv	$\mathcal{C}4.5$	$\mathcal{CN}2$	$\mathcal{C}4.5$-rules
all features	20.08±2.69	21.44±2.19	20.05±2.90
FSS(f,CI)	19.74±2.50	21.79±2.22	20.41±2.18
FSS(f,$\mathcal{C}4.5$)	20.09±2.59	20.02±2.62	19.40±2.66
FSS(f,ID3)	20.75±2.68	21.09±2.23	18.03±2.21
FSS(f,RS)	21.41±3.45	26.17±3.11	20.75±3.61

Table 6.17. Hepatitis – Selected Features

Inducer	Selected Features	#F	%F
FSS(f,CI)	2 3 5 8 10 11 13 16 17 18	10	52.63%
FSS(f,C4.5)	0 1 3 4 5 7 8 10 11 15 16 17	12	63.16%
FSS(f,ID3)	0 3 7 10 11 13 14 16 17	9	47.37%
FSS(f,RS)	0 10 16	3	15.79%

Table 6.18. Hepatitis – Errors

hepatitis 10-cv	$\mathcal{C}4.5$	$\mathcal{CN}2$	$\mathcal{C}4.5$-rules
all features	21.92±3.20•	16.18±1.80	20.54±3.02
FSS(f,CI)	20.75±3.54•	20.09±3.42	18.71±3.36
FSS(f,$\mathcal{C}4.5$)	17.42±1.64	14.86±2.53	18.75±2.03
FSS(f,ID3)	19.46±2.93	18.17±2.21	19.46±2.44
FSS(f,RS)	19.33±3.42	20.66±3.01•	18.71±3.86

7 Results Comparison

The following subsections show a summary of these results.

7.1 Number of Features Selected

Table 7.1.1 shows, for each dataset, the number of selected features as well as the proportion and average of selected features using the various filters. It also shows the percentage of the total number of features selected by each filter approach considering all datasets.

One important result obtained using the Rough Sets approach as filter is that the number of features selected by RS is always smaller than or equal to the number of features selected by $\mathcal{C}4.5$ and ID3, *i.e.*

$$\text{\#FSS(f,RS)} \leq \text{\#FSS(f,}\mathcal{C}4.5\text{)} \text{ and } \text{\#FSS(f,RS)} \leq \text{\#FSS(f,ID3)}$$

Furthermore, the number of features selected by RS is smaller than or equal to the number of features selected by CI, except for bupa dataset. It can be observed that the overall percentage of selected features using RS is less than 50% while the overall percentage of selected features using CI is not less than 70%.

As expected, since $\mathcal{C}4.5$ and ID3 induce decision trees, the number of features selected by both algorithms is more or less the same, not considering hepatitis dataset. Furthermore, the overall percentage of selected features is around 85%.

From these results and only considering the number of features selected by each one of the four filters, *i.e.* CI, $\mathcal{C}4.5$, ID3 and RS, it is possible to conclude that RS is the overall winner.

The second step of the experiments — Figure 5.1 — is described in next section.

Table 7.1.1. Number and Proportion of Selected Features

Dataset	#F	FSS							
		(f,CI)		(f,$\mathcal{C}4.5$)		(f,ID3)		(f,RS)	
ta	5	4	80.00%	5	100.00%	5	100.00%	3	60.00%
bupa	6	1	16.67%	6	100.00%	6	100.00%	3	50.00%
pima	8	6	75.00%	7	87.50%	8	100.00%	3	37.50%
breast cancer2	9	8	88.89%	8	88.89%	9	100.00%	5	55.56%
cmc	9	9	100.00%	9	100.00%	9	100.00%	9	100.00%
breast cancer	9	9	100.00%	8	88.89%	8	88.89%	4	44.44%
smoke	13	11	84.62%	13	100.00%	13	100.00%	11	84.62%
hungaria	13	10	76.92%	11	84.62%	11	84.62%	3	23.07%
hepatitis	19	10	52.63%	12	63.16%	9	47.37%	3	15.79%
Total / Average	10.11	74.73%	74.97%	86.81%	90.34%	85.71%	91.21%	48.35%	52.33%

7.2 Comparing No FSS and Filter FSS

To determine whether the difference between two algorithms — say A_1 and A_2 — is significant or not a table is presented in this section — Table 7.2.1. Each cell in the table corresponds to the mean error divided by the standard deviation where ten-fold cross-validation has been used. When the number in the cell is greater than two, the results are significant at 95% confidence level.

The comparisons are made such that A_2 represents the inducer using the filter selected features and A_1 is the inducer itself using all features. When the number is bellow zero it means that A_2 outperforms A_1 — meaning that using only the filter selected features did improve the accuracy of the standard algorithm.

For each dataset, the combined mean $\mathrm{m}(A_2 - A_1)$ and standard deviation $\mathrm{sd}(A_2 - A_1)$ are calculated, respectively, according to Equations 5 and 6. The difference in standard deviations is given by Equation 7.

$$\mathrm{m}(A_2 - A_1) = \mathrm{m}(A_2) - \mathrm{m}(A_1) \tag{5}$$

$$\mathrm{sd}(A_2 - A_1) = \sqrt{\frac{\mathrm{sd}(A_2)^2 + \mathrm{sd}(A_1)^2}{2}} \tag{6}$$

$$\mathrm{ad}(A_2 - A_1) = \frac{\mathrm{m}(A_2 - A_1)}{\mathrm{sd}(A_2 - A_1)} \tag{7}$$

Table 7.2.1 shows the results obtained by Equation 7, for each inducer error using no feature selection ($inducer$) and the results for ID3, $\mathcal{C}4.5$, CI and RS used as filters — FSS(f,$inducer$).

Considering only the cases where the filter approach outperforms the standard inducer at the 95% confidence level, or the other way round where the standard inducer outperforms the filter approach at the 95% level, we have for $\mathcal{C}4.5$ — see Table 7.2.1:

– For the bupa dataset, there are two cases where the standard inducer outperforms CI and RS used as filter, respectively.

Table 7.2.1. Difference in Standard Deviations of Errors

Dataset	FSS(f,CI)			FSS(f,$\mathcal{C}4.5$)			FSS(f,ID3)			FSS(f,RS)		
	$-\mathcal{C}4.5$	$-\mathcal{CN}2$	$-\mathcal{C}4.5$-rules	$-\mathcal{C}4.5$	$-\mathcal{CN}2$	$-\mathcal{C}4.5$-rules	$-\mathcal{C}4.5$	$-\mathcal{CN}2$	$-\mathcal{C}4.5$-rules	$-\mathcal{C}4.5$	$-\mathcal{CN}2$	$-\mathcal{C}4.5$-rules
ta	-0.52	1.97	-0.70	0.00	0.00	0.00	0.00	0.00	0.00	0.21	-0.16	-1.04
bupa	4.05	5.39	3.35	0.00	0.00	0.00	0.00	0.00	0.00	4.19	1.26	3.17
pima	1.51	0.44	1.62	0.26	-0.11	0.00	0.00	0.00	0.00	1.10	2.41	1.42
breast-cancer2	-0.19	-0.44	0.65	-1.18	-1.18	0.83	0.00	0.00	0.00	-0.70	0.29	-0.97
cmc	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	-0.40	0.00
breast-cancer	-0.46	0.00	0.00	0.00	-1.01	0.00	-1.40	-0.38	0.00	-1.29	2.37	0.64
smoke	-2.08	6.33	-0.47	0.00	0.00	0.00	0.00	0.00	0.00	-0.03	-0.23	0.65
hungarian	-0.69	-0.45	-0.17	0.08	0.19	-0.17	0.09	-0.70	0.11	0.43	1.76	0.21
hepatitis	-0.51	1.59	-0.01	-1.79	-1.11	-1.51	-0.19	-0.33	0.07	-0.78	1.81	-0.53

– For the smoke dataset, CI used as filter outperforms once the standard inducer.

Similarly for $\mathcal{CN}2$, we have:

– For datasets bupa and smoke, there are two cases where the standard inducer outperforms the filter approach.
– For datasets pima and breast cancer, there are two cases where the standard inducer outperforms RS used as filter.

However, for $\mathcal{C}4.5$-rules, it can be noted that the standard inducer outperforms the filter approach in two cases and there is no case where the filter approach is better, at the 95% confidence level, than the standard inducer.

Table 7.2.2 shows improved accuracies at the significance level (95% confidence) for filter selection, compared with the standard inducers $\mathcal{C}4.5$, $\mathcal{C}4.5$-rules and $\mathcal{CN}2$.

Observe that Table 7.2.2 only shows CI and RS filter selection compared with the standard inducers, since no improved accuracy at the 95% confidence level was obtained by using $\mathcal{C}4.5$ and ID3 as filters.

Table 7.2.2. Improved Accuracies at the Significance Level

Dataset	FSS						#	#
	(f,CI) $\mathcal{C}4.5$	(f,CI) CN2	(f,CI) $\mathcal{C}4.5$-rules	(f,RS) $\mathcal{C}4.5$	(f,RS) CN2	(f,RS) $\mathcal{C}4.5$-rules	$\triangle$	$\triangledown$
ta							0	0
bupa	$\triangledown$	$\triangledown$	$\triangledown$	$\triangledown$		$\triangledown$	0	5
pima					$\triangledown$		0	1
breast cancer2							0	0
cmc							0	0
breast cancer					$\triangledown$		0	1
smoke	$\triangle$	$\triangledown$					1	1
hungaria							0	0
hepatiti							0	0
# $\triangle$	1	0	0	0	0	0	1	
# $\triangledown$	1	2	1	1	2	1		8

Improvements bellow 2 standard deviations are reported with $\triangle$, *i.e.* the filter approach outperforms the standard inducer at the 95% confidence level, and those bellow, where the standard inducer outperforms the filter approach, with $\triangledown$.

Through Table 7.2.2, it can be seen that the filter approach outperforms the standard inducer in only one case at the 95% confidence level. On the other hand, the standard inducer outperforms eight times the filter approach at the 95% confidence level. Specifically, when using RS as filter, there is not a single case where the accuracy was improved at the 95% confidence level, and in four cases the standard inducer outperforms RS filter approach at the 95% confidence level.

Although there is only one case where the filter approach outperforms at the 95% confidence level the error rate of the standard inducer — dataset smoke using FSS(f,CI) and $\mathcal{C}4.5$ as inducer as shown in Table 7.2.2 — we decided to investigate these results further.

One of the reasons is that the filter approach is a very fast method, in contrast with other FSS methods [19]. Furthermore, in some cases, for example high cost in measuring features, it may be worth to consider the possibility of allowing a slight increase in classification error if some costly features can be discarded.

7.3 Other Results for Filter FSS

In this section some tables are presented showing, for each dataset and inducer used as filter, a coefficient that represents the proportion of discarded features after filter FSS. This coefficient is calculated as shown in Equation 8.

$$Dec(f, D) = 1 - \frac{|Features_f|}{|Features_D|} \tag{8}$$

where $|Features_D|$ is the total number of features present in dataset D and $|Features_f|$ is the number of features selected using the filter method f. Thus, $Dec(f, D)$ gives the percentage of discarded features after FSS.

Taking only into account the percentage of discarded features after FSS, it can be observed that Rough Sets is similar or outperforms the other filters, except for dataset bupa, where CI discarded more features — Table 7.3.1.

However, the classification error should be taken into account for choosing a convenient pair *(Filter,Inducer)* such that the increase in classification error is reasonable considering the decrease in the number of features. Thus, this choice is subjective since it depends on which value is more important: the error in classification or the decrease in the number of features.

Considering Table 7.2.1 that presents the difference in standard deviation of errors and Table 7.3.1 that presents the percentage of discarded features, it is possible to point out some results:

- For dataset TA, FSS(f,RS) is appropriated for the three inducers.
- For dataset Bupa, FSS(f,RS) is the best option but only for $\mathcal{CN}2$. In fact, this dataset shows the worst results for the filter approach.
- For dataset Pima, FSS(f,RS) is appropriated but only for $\mathcal{C}4.5$ and $\mathcal{C}4.5$-rules, and FSS(f,CI) for $\mathcal{CN}2$. However, if the classification error is the main concern, then FSS(f,$\mathcal{C}4.5$) should be selected for the three inducers.
- For dataset Breast Cancer2, FSS(f,RS) is more appropriated for $\mathcal{C}4.5$ and $\mathcal{C}4.5$-rules, while FSS(f,CI) should be used with $\mathcal{CN}2$.

Table 7.3.1. Proportion of Discarded Features

Dataset	#F	FSS			
		(f,CI)	(f,$\mathcal{C}4.5$)	(f,ID3)	(f,RS)
ta	5	20.00%	0.00%	0.00%	40.00%
bupa	6	83.33%	0.00%	0.00%	50.00%
pima	8	25.00%	12.50%	0.00%	62.50%
breast cancer2	9	11.11%	11.11%	0.00%	44.44%
cmc	9	0.00%	0.00%	0.00%	0.00%
breast cancer	9	0.00%	11.11%	11.11%	55.56%
smoke	13	15.38%	0.00%	0.00%	15.38%
hungaria	13	23.08%	15.38%	15.38%	76.93%
hepatitis	19	47.37%	36.84%	52.63%	84.21%

- For dataset Cmc, all features seem to be relevant since none of the filters was able to discard any feature.
- For dataset Breast Cancer, FSS(f,RS) is appropriated for $\mathcal{C}4.5$ and $\mathcal{C}4.5$-rules but not for $\mathcal{CN}2$, since the standard inducer outperforms it at the 95% confidence level. For $\mathcal{CN}2$, FSS(f,$\mathcal{C}4.5$) is more appropriated.
- For dataset Smoke, FSS(f,RS) is appropriated for $\mathcal{CN}2$, and FSS(f,CI) for $\mathcal{C}4.5$ and $\mathcal{C}4.5$-rules.
- For dataset Hungarian, FSS(f,RS) is appropriated for $\mathcal{C}4.5$, and $\mathcal{C}4.5$-rules and FSS(f,CI) is appropriated for $\mathcal{CN}2$. Again, if the classification error is the main concern, then FSS(f,CI) should be selected for the three inducers.
- For dataset Hepatitis, FSS(f,RS) is appropriated for $\mathcal{C}4.5$ and $\mathcal{C}4.5$-rules, and FSS(f, ID3) for $\mathcal{CN}2$. However, if classification error is the main concern, the FSS(f,$\mathcal{C}4.5$) is a good option for the three inducers.

Through Table 7.3.2 it is possible to note that, although RS did not reach a result at the 95% confidence level, this approach can be used with most of the inducers used in this work. From 27 possible combinations of the pair *(Filter,Inducer)*, it is possible to use the RS as a filter for FSS in 15 cases.

Table 7.3.2. Possible Choose for the Pair *(Filter,Inducer)*

Dataset	Inducers		
	$\mathcal{C}4.5$	$\mathcal{CN}2$	$\mathcal{C}4.5$-rules
ta	(f,RS)	(f,RS)	(f,RS)
bupa	—	(f,RS)	—
pima	(f,RS)	(f,CI)	(f,RS)
breast cancer2	(f,RS)	(f,CI)	(f,RS)
cmc	—	—	—
breast cancer	(f,RS)	(f,$\mathcal{C}4.5$)	(f,RS)
smoke	(f,CI)	(f,RS)	(f,CI)
hungarian	(f,RS)	(f,CI)	(f,RS)
hepatitis	(f,RS)	(f,ID3)	(f,RS)

8 Conclusions

At a conceptual level, the problem of Feature Subset Selection is that of finding a subset of the original features of a dataset, such that given this subset to an induction algorithm, it induces a classifier with the lowest possible error. It is important to notice that FSS chooses a set of features from the existing features and does not construct new ones, *i.e.* the description space is not increased.

In practice, it is desirable that the FSS process removes features which are not essential for learning since ML algorithms do not work well in the presence of many features. Furthermore, FSS can improve comprehensibility and can reduce the cost of processing huge quantities of data.

In this work we propose Rough Sets reducts as a filter method for FSS comparing its performance with other filter approaches for FSS on nine real world datasets. The reducts, *i.e.* the filtered features using Rough Sets, were found using the software Rosetta [6]. Afterwards the inducers $\mathcal{C}4.5$, $\mathcal{CN}2$ and $\mathcal{C}4.5$-rules were run using the $\mathcal{MLC}^{++}$ library with its default option setting.

Results show that the filter approach is a method that, except for one case, it does not outperforms the standard inducer at the 95% confidence level. Furthermore, in a few cases the standard inducer outperforms the filter approach at the 95% level.

Still, not considering bupa dataset, in most cases the increase in classification error is reasonable considering the decrease in the number of selected features. Results using Rough Sets reducts as filter show that for almost all datasets used in this work, this method selects the smallest subset of features, although not necessary with the smallest increase in classification error.

We consider that a general procedure to follow in the filter approach is to test several methods. Afterwards, based on the allowed classification error *versus* the decrease in the number of features as proposed in Section 7.3, it is possible to choose the more appropriated method for the specific problem.

It should be observed that we have considered all the errors as having equal importance, not paying attention to unbalanced number of examples [20]. However, for many applications, distinction among different type of errors turn out to be important. A natural alternative is to assign different misclassification costs to each type of error, *i.e.* a penalty for making a mistake [21].

In symbolic Machine Learning it is also important to consider the number and the kind of rules induced. We are currently investigating the impact of filters on the induced rules. Work in this direction is important for Datamining as shown in [22].

References

1. A. D. Pila and M. C. Monard, "An empirical comparison of rough sets reducts and other filters approaches for feature subset selection," in *Proceedings do Simpósio Ibero-Americano de Recolhecimento de Padrões*, (Florianópolis, SC, Brazil), November 2001.
2. R. Kohavi and G. H. John, "Wrappers for feature subset selection," *Artificial Intelligence*, pp. 273–324, 1997.
3. A. L. Blum and P. Langley, "Selection of relevant features and examples in machine learning," *Artificial Intelligence*, pp. 245–271, 1997.

4. Z. Pawlak, "Rough sets," *International Jornal of Computer and Information Sciences*, pp. 341–356, 1982.
5. R. Kohavi, D. Sommerfield, and J. Dougherty, "Data mining using $\mathcal{MLC}^{++}$: A machine learning library in $\mathcal{C}^{++}$," *Tools with IA*, pp. 234–245, 1996.
6. A. Øhrn, "Rosetta: Technical reference manual," tech. rep., Knowledge System Group, November 1999. http://www.idi.ntnu.no/∼aleks/rosetta.
7. Z. Pawlak, "Rough set approach to knowledge-based decision support," *14th European Conference on Operational Research*, p. 12, 1995.
8. Z. Pawlak, J. Grzymala-Busse, R. Slowinski, and W. Ziarko, "Rough sets," *Comunications of the ACM*, pp. 89–95, 1995.
9. Z. Pawlak, "Rough sets, rough relations and rough functions," *Fundamenta Informaticae, 27*, pp. 103–108, 1996.
10. J. Komorowski, Z. Pawlak, L. Polkowski, and A. Skowron, "Rough sets: A tutorial," tech. rep., Warsaw University, December 1999.
11. J. Komorowski and A. Øhrn, "Modelling prognostic power of cardiac tests using rough sets," *Artificial Intelligence in Medicine*, pp. 167–191, 1999.
12. A. Øhrn, *Discernibility and Rough Sets in Medicine: Tools and Applications*. PhD thesis, Norwegian University on Science and Technology, 1999.
13. R. Kohavi, D. Sommerfield, and J. Dougherty, *$\mathcal{MLC}^{++}$: A Machine Learning Library in C++*. IEEE Computer Society Press, 1994.
14. L. C. M. Felix, S. O. Rezende, C. Y. Doi, M. F. de Paula, and M. J. Romanato, "MLC^{++} biblioteca de aprendizado de máquina em C^{++}," Tech. Rep. 72, ICMC-USP, mar 1998. ftp://ftp.icmc.sc.usp.br/pub/BIBLIOTECA/rel_tec/Rt_72.ps.zip.
15. C. Blake, E. Keogh, and C. Merz, "UCI Irvine repository of machine learning databases," 1998. http://www.ics.uci.edu/∼mlearn/MLRepository.html.
16. T. M. Mitchell, *Machine Learning*. WCB/McGraw-Hill, 1997.
17. A. D. Pila and M. C. Monard, "Rough sets reducts as a filter approach for feature subset selection: An empirical comparison with wrapper and other filters," Tech. Rep. 134, ICMC-USP, january 2001. ftp://ftp.icmc.sc.usp.br/pub/BIBLIOTECA/rel_tec/Rt_134.ps.zip.
18. A. D. Pila and M. C. Monard, "Rules induction using rough sets reducts as feature subset selection: An empirical comparison with other filter approaches," Tech. Rep. 139, ICMC-USP, march 2001. ftp://ftp.icmc.sc.usp.br/pub/BIBLIOTECA/rel_tec/Rt_139.ps.zip.
19. H. D. Lee, M. C. Monard, and J. A. Baranauskas, "Empirical comparison of wrapper and filter approaches for feature subset selection," Tech. Rep. 94, ICMC-USP, oct 1999. ftp://ftp.icmc.sc.usp.br/pub/BIBLIOTECA/rel_tec/Rt_94.ps.zip.
20. G. E. A. P. A. Batista, A. C. P. L. Carvalho, and M. C. Monard, "Applying one-sided selection to unbalanced datasets," in *Proceedings of the Mexican Congress on Artificial Intelligence (MICAI-2000), Lecture Notes in Artificial Intelligence*, pp. 315–325, 2000.
21. S. M. Weiss and C. A. Kulikowski, *Computer Sysytems that Learn*. Morgan Kaufmann Publishers, Inc, 1990.
22. T. Y. Lin and N. Cercone, *Rough Sets and Data Mining: Analysis of Imprecise Data*. Kluwer Academic Publishers, 1997.