Analysing Rough Sets weighting methods for Case-Based Reasoning Systems*

Maria Salamó and Elisabet Golobardes
Enginyeria i Arquitectura La Salle,
Universitat Ramon Llull
Psg. Bonanova 8, 08022 Barcelona
{mariasal,elisabet}@salleurl.edu

Abstract: *Case-Based Reasoning systems retrieve cases using a similarity function based on the* K*-NN or some derivatives. These functions are sensitive to irrelevant, interacting or noisy features. Many similarity functions weigh the relevance of features to avoid this problem. This paper proposes two weighting methods based on Rough Sets theory: Proportional Rough Sets and Dependence Rough Sets. Both weighting methods use the representative knowledge extracted from the original data to compute the feature relevance using two different policies. The first one computes the proportional participation of the features in the representative knowledge. The second one computes the dependence of each feature in the representative knowledge. This dependence denotes if a feature is superfluous within the knowledge. Experiments using different domains show that weighting methods based on Rough Sets maintain or even improve the classification accuracy of Case-Based Reasoning Systems, compared to non-weighting approaches or well-known weighting methods.*

Keywords: Case-Based Reasoning, Feature Selection, Knowledge Discovery

1 Introduction

Case-Based Reasoning (CBR) systems [RS89] retrieve cases using a similarity function. However, the similarity degrades when there are irrelevant or redundant features, or the data is noisy and unreliable. Feature selection, also known as weighting method, is the process of identifying as much of the irrelevant information as possible.

Many algorithms that perform feature selection have been proposed in the Artificial Intelligence literature in recent years. These algorithms can be placed in two main categories: *wrappers* and *filters*. *Wrapper* methods use the performance algorithm itself as

*This work was partially supported by the *Ministerio de Sanidad y Consumo, Instituto de Salud Carlos III. Fondo de Investigación Sanitaria* Grant No. 00/0033-02

an evaluation function to estimate the accuracy of feature subsets [KJ97]. This approach tend to be expensive computationally because the learning algorithm is called repeatedly. For this reason, wrappers do not scale well on large data sets containing many features. On the other hand, *filter* methods do not use feedback of the learning algorithm. Undesirable features are filtered out of the data set before learning takes place. Filters typically make use of all the available training data when selecting a subset of features. For example, some induce a decision tree [Qui93], keeping the features selected that remain in the tree after pruning [Car93].

This paper presents two different *filter* approaches based on Rough Sets theory. Both filter methods have been introduced into our Case-Based Classifier System called BASTIAN. Case-Based Reasoning and Rough Sets theory has usually been used separately in the literature. The weighting methods are: Proportional Rough Sets (PRS) and Dependence Rough Sets (DRS). First weighting method, PRS, proposes a measure that computes the proportional participation of the features in the representative knowledge. The second one, DRS, obtains the dependence of each feature in the knowledge. This dependence denotes if a feature is superfluous within the representative knowledge.

The paper is structured as described: section 2 introduces the related work on filter methods; next, section 3 explains the Rough Sets theory; section 4 details the Rough Sets weighting methods; section 5 exposes the experiments and the results obtained using the weighting techniques; and finally, section 6 presents the conclusions and further work.

2 Related work

Many filter methods for feature selection have been proposed recently, a review of them can be found in [BL97]. Filters use general characteristics of the data to evaluate features and operate independently of any learning algorithm. Filters have been proven to be much faster than wrappers and hence they can be applied efficiently to large data sets containing many features. However, some weighting methods can handle regression problems, that is, when the class is a numeric rather than discrete valued variable.

The simplest filtering scheme is to evaluate each feature individually measuring its correlation to the target function (e.g. using a mutual information measure) and then select K features with the highest value. *Relief* algorithm, proposed by Kira and Rendell's [KR92], follows this general paradigm. *Relief* samples randomly an instance, locating its nearest neighbour from the same and opposite class. It was originally defined for two-class problems. *Relief* selects features constructing a decision tree, other induction methods can also be used. *Relief* was extended by Kononenko. The extension called *ReliefF* [Kon94] can handle noisy and multiclass problems. *ReliefF* smoothes the influence of noise in the data by averaging from the same and opposite class of each sampled instance instead of a single nearest neighbour. Domingos [Dom97] introduced *RC*, an algorithm reminiscent of *Relief*. *RC* hill-climbs features, guided by leave-one-out cross validation error (LOOCE) on the training set, only if feature selection increases predictive accuracy. Unlike *Relief*, *CFS* [Hal00] evaluates and hence ranks feature subsets rather than indi-

vidual features. *CFS* algorithm is a subset evaluation heuristic that takes into account the usefulness of individual features for predicting the class along with the level of inter-correlation among them. Some filters induce a decision tree, where the features selected for similarity computations are those that remain in the tree after pruning [Car93].

3 Rough Sets theory

Zdzislaw Pawlak introduced Rough Sets theory in 1982 [Paw91]. The idea of Rough Sets consists of the approximation of a set by a pair of sets, called the lower and the upper approximation of this set. In fact, these approximations are inner and closure operations in a certain topology. These approximations are generated by the available data about the elements of the set. The nature of Rough Sets theory makes them useful for reducing knowledge, extracting dependencies in knowledge, pattern recognition, etc.

We use Rough Sets theory for reducing and extracting the representative knowledge. This representative knowledge is the basis for computing the relevance of each feature into the Case-Based Reasoning system. We use that representative knowledge in two different ways. The first one is **Proportional Rough Sets** (PRS) and the second one is **Dependence Rough Sets** (DRS). First of all, we incorporate some **basic concepts and definitions**. Then, we explain how to obtain the representative knowledge, in order to select the best weighting.

We have a **Universe** (U) (finite not null set of objects that describes our problem, i.e. the case memory). We compute from our universe the **concepts** (cases) that form partitions. The union of all the *concepts* make the entire Universe. Using *all the concepts* we can describe all the **equivalence relations** (R) over the universe U. Let an equivalence relation be a *set of features* that describes a specific concept. U/R is the family of all **equivalence classes** of R. The universe and the relations form the **knowledge base** (K), defined as $K =< U, \hat{R} >$. Where $\hat{R}$ is the family of equivalence relations over U. Every relation over U is an elementary concept in the knowledge base. All the concepts are formed by a set of equivalence relations that describe them. Thus, we search for the minimal set of equivalence relations that defines the same concept as the initial set.

Definition 1 (Indiscernibility Relations)
$IND(\hat{P}) = \bigcap \hat{R}$ where $\hat{P} \subseteq \hat{R}$. The indiscernibility relation is an equivalence relation over U. Hence, it partitions the concepts (cases) into equivalence classes. These sets of classes are sets of instances indiscernible with respect to the features in P. Such a partition (classification) is denoted as $U/IND(P)$. In supervised machine learning, the sets of cases indiscernible with respect to the class attribute contain the cases of each class.

4 Rough Sets as a weighting method

In this section we explain how to extract the representative knowledge and how to weigh features using the Rough Sets theory. We obtain the representative knowledge unifying two concepts: (1) approximation sets of knowledge and (2) reduction of search space.

This representative knowledge is the basis for the PRS and DRS weighting methods. Both methods are filters based on Rough Sets theory. Next, it describes the unification of both concepts to extract the feature relevance using two policies: PRS and DRS.

Representative knowledge

Approximation Sets This is main idea of Rough Sets to approximate a set by other sets. The *condition set* contains all cases present in the case memory. The *decision set* presents all the classes that the condition set has to classify. We are searching for a subset of the condition set able to classify the same as the initial set, so it approximates the same decision set. The following definitions explain this idea.

For any subset of cases $X \subseteq U$ and an equivalence relation $R \in IND(K)$ we associate two subsets called: (1) Lower approximation $\underline{R}X$ and (2) Positive Region $POS_P(R)$.

Definition 2 (Lower approximation)
The lower approximation defined as: $\underline{R}X = \bigcup\{Y \in U/R : Y \subseteq X\}$ is the set of all elements of U which can be certainly classified as elements of X in knowledge R.

Definition 3 (Positive Region)
Let P and R be equivalence relations over U. The P-positive region of R defined as $POS_P(R) = \bigcup_{X \in U/P} \underline{P}X$ is the set of all objects of the universe U which can be properly classified to classes of U/R, employing knowledge expressed by the classification U/P.

Reduction search space: Reducts and Core of knowledge Intuitively, a **reduct** of knowledge is its essential part, which suffices to define all concepts occurring in the considered knowledge, whereas the **core** is the most important part of the knowledge.

Let $\hat{R}$ be a family of equivalence relations and let $R \in \hat{R}$. We will say that:

- R is *indispensable* if $IND(\hat{R}) \neq IND(\hat{R} - R)$; otherwise it is *dispensable*. $IND(\hat{R} - R)$ is the family of equivalence$\hat{R}$ extracting R.
- The family $\hat{R}$ is *independent* if each $R \in \hat{R}$ is *indispensable* in R; otherwise it is *dependent*.

Definition 4 (Reduct)
$\hat{Q} \in \hat{R}$ is a reduct of $\hat{R}$ if : $\hat{Q}$ is *independent* and $IND(\hat{Q}) = IND(\hat{R})$. Obviously $\hat{R}$ may have many reducts. Using $\hat{Q}$ it is possible to approximate the same as using $\hat{R}$. Each reduct has the property that a feature can not be removed from it without changing the indiscernibility relation.

Definition 5 (Core)
The set of all indispensable relations in $\hat{R}$ will be called the *core* of $\hat{R}$, and will be denoted as $CORE(\hat{R}) = \bigcap RED(\hat{R})$. Where $RED(\hat{R})$ is the family of all reducts of $\hat{R}$. The core can be interpreted as the set of the most characteristic part of knowledge, which can not be eliminated when reducing the knowledge.

EXAMPLE 1
If we consider a set of 8 objects in our Universe, $U = (x_1, x_2, x_3, x_4, x_5, x_6, x_7, x_8)$, using $\hat{R} = (P, Q, S)$ as a family of equivalence relations over U. Where P can be colours (green, blue, red, yellow); Q can be sizes (small, large, medium); and S can be shapes (square, round, triangular, rectangular). For example, we can suppose that the equivalence classes are:
$U/P = \{ (x_1, x_4, x_5), (x_2, x_8), (x_3), (x_6, x_7) \}$
$U/Q = \{ (x_1, x_3, x_5), (x_6), (x_2, x_4, x_7, x_8) \}$
$U/S = \{ (x_1, x_5), (x_6), (x_2, x_7, x_8), (x_3, x_4) \}$

As it can be seen, every equivalence class divides the Universe in a different way. Thus the relation $IND(R)$ has the *equivalence classes*:
$U/IND(\hat{R}) = \{(x_1, x_5), (x_2, x_8), (x_3), (x_4), (x_6), (x_7)\}$
The relation P is indispensable in $\hat{R}$, since:
$U/IND(\hat{R} - P) = \{ (x_1, x_5), (x_2, x_7, x_8), (x_3), (x_4), (x_6) \} \neq$ U/IND($\hat{R}$).
The information obtained removing Q is equal, so Q is dispensable in $\hat{R}$.
$U/IND(\hat{R} - Q) = \{ (x_1, x_5), (x_2, x_8), (x_3), (x_4), (x_6), (x_7) \} =$ U/IND($\hat{R}$).
Hence the relation S is also dispensable in $\hat{R}$.
$U/IND(\hat{R} - S) = \{ (x_1, x_5), (x_2, x_8), (x_3), (x_4), (x_6), (x_7) \} =$ U/IND($\hat{R}$).

That means that the classification defined by the set of three equivalence relations P, Q and S is the same as the classification defined by relation P and Q or P and S. Thus, the reducts and core are: $RED(\hat{R}) = \{(P, Q), (P, S)\}$ and $CORE(\hat{R}) = \{P\}$

Computing the Feature Relevance

Our weighting methods deal with continuous and nominal features. Rough Sets weighting methods perform search approximating sets by other sets and both proposals are global. Global means that we select the feature relevance for all cases, without take into account which class each case classify. PRS assumes a proportional dependence in our reduced information set, where irrelevant features are those that do not appear. However, DRS irrelevant features are those that do not contain significance dependence in the reduced set. These policies induce two different behaviours. We want to remark that PRS and DRS can be used in multiclass tasks. Finally, PRS and DRS can learn good features weights in different domains, with continuous or nominal features and missing values.

The definition of PRS and DRS weighting methods use the information of reducts and core to weigh the feature relevance.

Proportional Rough Sets (PRS). The relevance of each feature in the system is computed using the proportional appearance at the reducts and core of information.

$$For\ each\ feature\ f\ computes:$$
$$\mu(f) = \frac{card(appearance\ f\ in\ RED(R))}{card(\ all\ RED(R))} \tag{1}$$

An attribute f that does not appear in the reducts has a feature weight value $\mu(f) = 0$, whereas a feature that appears in the core has a feature value $\mu(f) = 1$. The remaining

attributes have a feature weight value depending on the proportional appearance in the reducts.

Dependence Rough Sets (DRS). In this weighting method we use the significant attribute Dependence coefficient, computed using the core and reducts of information. The *significant dependence* coefficient is computed as:

$$For\ each\ feature\ f\ computes: \\ \mu(f) = \frac{card(\ POS_P(RED(R))\ -\ POS_{(P-f)}(RED(R))\ )}{card(\ all\ cases)} \tag{2}$$

where f is the feature from which we are computing the weight; P is the set of feature reducts, RED(R), obtained from the original data; R is the set of all relations; *card* is the cardinality; $POS_P(R)$ is the positive region of all relations (features) present in the reducts; and finally, $POS_{(P-f)}(R)$ is the positive region of all relations present in the reducts extracting feature f.

The value $\mu(f) = 1$ means that R totally depends on P. Whereas if the value is $0 < \mu(f) < 1$, we say that R partially depends on P. And if $\mu(f) = 0$ we say that R is totally independent from P. The measure $\mu(f)$ does not capture how this partial dependency is actually distributed among the classes of U/R.

The study described in this paper was carried out in the context of BASTIAN, a case-**BA**sed **S**ys**T**em **I**n cl**A**ssificatio**N**[SGVN00]. BASTIAN configuration in this study is a simple 1-NN algorithm using weighted Minkowski's metric. For details according to BASTIAN platform see [SGVN00]. Although the introduction of Rough Sets weighting methods is described in terms of BASTIAN platform, these feature relevance methods can be applied in other machine learning algorithms.

Three steps divide the Rough Sets process: The **first** one discretises the cases, it is necessary to use Rough Sets theory. In that case, we discretise continuous features using Fayyad and Irani's algorithm [FI93]. The discretisation is only performed to extract the feature relevance, whereas CBR system works using normalised data. The missing values are treated by Rough Sets as values that matches everything. CBR system treats missing values as a value that can not be used to compute the similarity between two cases. **Second** step searches for the reducts and the core of knowledge using the Rough Sets theory, as it has been described. Finally, the **third** step uses the core and the reducts of knowledge to decide the feature relevance values using PRS and DRS methods.

Rough Sets theory has been introduced as weighting methods in two phases of the CBR cycle. The first phase is the *start-up* phase and the second one is the *retain* phase. The start-up phase computes the weights from the initial case memory, which will be used by the retrieval phase later. The retain phase computes the weights from the case memory if a new case is stored. The code of Rough Sets theory into the Case-Based Reasoning has been implemented using a public Rough Sets Library [GS93].

5 Empirical study

This section is structured as follows: first, we describe the testbed used in the empirical study; next, we show the results using PRS and DRS and we also compare them in front of the Sample Correlation [GGBL97], ReliefF, CFS and with unweighted CBR.

5.1 Testbed

In order to evaluate the performance rate, we use twelve datasets grouped in two categories: *public* and *private*. Table I shows the datasets and their characteristics.

Table I. Datasets and their characteristics used in the empirical study.

	Dataset	Ref.	Samples	Numeric Feat.	Simbolic Feat.	Classes	Inconsistent
1	*Biopsy*	*BI*	1027	24	-	2	Yes
2	*Breast cancer (Wisconsin)*	*BC*	699	9	-	2	Yes
3	*Glass*	*GL*	214	9	-	6	No
4	*Ionosphere*	*IO*	351	34	-	2	No
5	*Iris*	*IR*	150	4	-	3	No
6	*LED*	*LE*	2000	-	7	10	Yes
7	*Mammogram problem*	*MA*	216	23	-	2	Yes
8	*MX11*	*MX*	2048	-	11	2	No
9	*Sonar*	*SO*	208	60	-	2	No
10	*TAO-Grid*	*TG*	1888	2	-	2	No
11	*Vehicle*	*VE*	846	18	-	4	No
12	*Vowel*	*VO*	990	10	3	11	No

The *Public datasets* are obtained from the UCI repository [MM98]. They are: *breast cancer, glass, ionosphere, iris, led, sonar, vehicle and vowel. Private datasets* are from our own repository. They deal with *diagnosis* of breast cancer and *synthetic* datasets. Datasets related to diagnosis are *biopsy* and *mammogram. Biopsy* is the result of digitally processed biopsy images, whereas *mammogram* consists of detecting breast cancer using the N microcalcifications present in a mammogram [GLSM01]. On the other hand, we use two *synthetic* datasets: *MX11* is the eleven input multiplexer and TAO-*grid* is obtained from sampling the TAO figure using a grid.

These datasets were chosen in order to provide a wide variety of application areas, sizes, combinations of feature types, and difficulty as measured by the accuracy achieved on them by current algorithms. The choice is also made with the goal of having enough data points to extract conclusions.

All systems were run using the same parameters for all datasets. The percentage of correct classifications has been averaged over stratified ten-fold cross-validation runs, with their corresponding standard deviations. To study the performance we use a paired one-sided t-test on these runs, except for the LED dataset, which was run using *hold-out* with a training set of 2000 instances and a test set of 4000 instances.

5.2 Experimental analysis of weighting methods

Table II shows the experimental results for each dataset using unweighted CBR system (CBR), CFS (Correlation-Based Feature Selection)[Hal00]. ReliefF [Kon94], SampleCorrelation (Corr), PRS and DRS. We compute the Sample Correlation between features and the class that classify. CFS and Sample Correlation have the same original nature, but they compute the feature relevance in a different way. The CFS and ReliefF weighting methods are coded into the *Waikako Environment Knowledge Analysis (WEKA)* [WF00]. The classifier scheme used with these two weighting methods is IB1 [AK91]. The ReliefF was codified to use K=10 neighbours and equal influence of nearest neighbours. CFS was used with default configuration provided in WEKA. We have select these filtering weighting methods because they can deal with numeric and nominal features and with multiclass problems, like both weighting methods proposed.

The results are compared in terms of percentage of correct classifications. Time performance is out of the scope of this paper, being part of the further work.

Table II. Results for all datasets showing the percentage of correct classifications and standard deviation. Bold font indicates the best result for each dataset. A $\surd$ and $\times$ show an increase or decrease in prediction accuracy with regard to unweighted CBR.

Ref.	CBR	CFS	ReliefF	Corr	PRS	DRS
BI	83.15(3.55)	79.87(2.81)$\times$	83.17(3.15)$\surd$	83.73(3.53)$\surd$	**84.42**(2.39)$\surd$	83.54(4.37)$\surd$
BC	96.28(1.71)	96.00(1.45)$\times$	96.00(1.45)$\times$	95.99(1.69)$\times$	**96.85**(1.69)$\surd$	95.70(1.59)$\times$
GL	72.42(7.46)	**73.29**(8.82)$\surd$	66.30(10.93)$\times$	71.96(6.23)$\times$	72.89(5.60)$\surd$	72.89(5.65)$\surd$
IO	90.59(3.65)	89.46(4.26)$\times$	86.92(4.86)$\times$	90.88(4.38)$\surd$	**93.44**(3.41)$\surd$	90.59(3.39)$\surd$
IR	**96.0**(3.26)	96.0(3.44)	96.00(3.26)	**96.0**(3.26)	**96.0**(3.26)	**96.0**(3.26)
LE	62.40(-)	62.40(-)	62.40(-)	**62.72**(-)$\times$	62.40(-)	62.40(-)
MA	64.81(9.12)	59.58(12.40)$\times$	63.47(12.15)$\times$	65.27(8.06)$\surd$	**66.20**(11.12)$\surd$	65.27(10.57)$\surd$
MX	78.61(3.96)	53.85(3.33)$\times$	78.61(3.96)	50.97(3.62)$\times$	81.44(2.91)$\surd$	**89.11**(1.41)$\surd$
SO	84.61(6.75)	85.30(7.01$\times$)	**87.27**(9.70)$\surd$	87.01(4.22)$\surd$	85.09(6.54)$\surd$	80.76(7.84)$\times$
TG	95.76(1.27)	67.21(1.71)$\times$	**96.13**(1.19)$\surd$	95.97(1.18)$\surd$	95.86(1.45)$\surd$	**95.97**(1.82)$\surd$
VE	67.37(5.05)	64.31(4.36)$\times$	69.43(5.30)$\surd$	64.77(3.65)$\times$	68.67(4.70)$\surd$	**69.97**(5.12)$\surd$
VO	99.29(0.78)	62.32(4.85)$\times$	99.09(1.00)	99.09(0.83)$\times$	**99.49**(0.50)$\surd$	98.78(1.67)$\times$

Comparing PRS and DRS approaches, we can observe that PRS has a behaviour more conservative than the results obtained by DRS. As it can be seen, PRS improves or maintains the results in all data sets with respect to unweighted CBR. On the other hand, DRS feature weighting method improves or decreases the results in some data sets, as it happens in the Sample Correlation. This behaviour is due to the weighting nature. DRS looks for the significance into the reduced set of feature space. Meanwhile, PRS selects a feature relevance depending only if it is needed or not in the representative space and not on the degree of significance in this space. This effect can be seen on the results presented in table II. PRS does not decrease the classification accuracy rate, it maintains the results in two data sets and improves the results in ten data sets. The results that are maintained belong to *iris* and *Led* data sets. The *iris* problem contains few instances and features to classify three classes, so it is difficult to denote an accurate weight settings. This effect is shown in all weighting methods tested. Meanwhile, the *Led* problem contains few instances to classify a great number of classes. However, PRS weighting method has been working successfully on ten data sets, the most important

point is that can deal with problems that contains a great number of features and also with multi class problems. On the other hand, DRS decreases in three data sets from the twelve data sets tested, improves in seven data sets and maintains on the rest. The most successful results have been achieved in *multiplexer* and *vehicle*, which are better than those obtained by the PRS. DRS is able to deal better with non linear separable problems. Although the results sometimes decrease in DRS approach, it is important to remark that the maximum values obtained are higher than these obtained using unweighted CBR. The Sample Correlation obtains a similar classification accuracy to that obtained by DRS, but the results on average are worse than the results obtained using PRS approach.

Table III. Results of paired one-sided t-test (p= 0.01). Number indicates how often methods in a row significantly outperforms methods in the column.

	CBR	**CFS**	**Corr**	**ReliefF**	**PRS**	**DRS**
CBR	-	5	1	2	0	0
CFS	0	-	1	1	0	0
Corr	0	4	-	1	0	0
ReliefF	1	4	1	-	1	1
PRS	1	5	1	1	-	0
DRS	1	6	2	1	1	-

Table III shows the comparative using paired one-sided t-test on all weighting methods. We have notice that the results obtained by PRS and DRS are similar to ReliefF, but the results on average are a bit higher. On the other hand, the results using CFS are worse for some datasets. The low percentage of CFS is due to the original nature of some datasets (i.e. multiplexer) or to the configuration selected in these experiments.

In conclusion, PRS and DRS obtain different results because they follow a different policy to compute the relevance of attributes. PRS searches for the proportional appearance of a feature in the reducts and core, in this sense it maintains near all the features obtaining accurate weighting values. The number of features that PRS reduces is not as great as the DRS approach. On the other hand, DRS searches for the dependence in the representative knowledge. This policy produces a slow number of features than PRS. These two policies produce different behaviours. The first one, PRS, maintains better the prediction accuracy but reduces less the number of features. However, PRS treats insignificant features with small weight values. On the other hand, DRS reduces as much as possible the number of features present in the data. This DRS behaviour produces that the prediction accuracy decreases in some data sets and obtains higher results in those that are non linear separable.

6 Conclusions and further work

This paper introduces two weighting methods based on the Rough Sets theory. Empirical studies show that these weighting methods often produce a higher or equal accuracy on classification tasks. Comparing these results with other weighting techniques, we show that on average the results are good. We also show that both weighting methods have different behaviours due to policy they follow. Further research consists of improving some of the weakness points as: searching new discretisations methods in order to improve

the pre-processing of the data; analysing the influence of the case memory size in these weighting methods; and developing our weighting methods in order to compute the feature relevance depending on the class each case classify.

References

[AK91] D. Aha and D. Kibler. Instance-based learning algorithms. *Machine Learning, Vol. 6*, pages 37–66, 1991.

[BL97] A.L. Blum and P. Langley. Selection of Relevant features and Examples in Machine Learning. In *Artificial Intelligence*, volume 97, pages 245–271, 1997.

[Car93] C. Cardie. Using decision trees to improve case-based learning. In *Tenth ICML*, pages 25–32. Amherst, MA. Morgan Kaufmann, 1993.

[Dom97] P. Domingos. Context-sensitive feature selection for lazy learners. In *AI Review*, volume 11, pages 227–253, 1997.

[FI93] U.M. Fayyad and K.B. Irani. Multi-interval discretization of continuous-valued attributes for classification learning. In *19th International Joint Conference on Artificial Intelligence*, pages 1022–1027, 1993.

[GGBL97] J.M. Garrell, E. Golobardes, E. Bernadó, and X. Llorà. Automatic Classification of Mamary Biopsy Images with Machine Learning techniques. In *Proceedings of EIS'98*, 1997.

[GLSM01] E. Golobardes, X. Llorà, M. Salamó, and J. Martí. Computer Aided Diagnosis with Case-Based Reasoning and Genetic Algorithms. *Elsevier Science Ltd.*, page In Press, 2001.

[GS93] M. Gawry's and J. Sienkiewicz. Rough Set Library user's Manual. Technical Report 00-665, Institute of Computer Science, Warsaw University of Technology, 1993.

[Hal00] M.A. Hall. Correlation-based feature selection of discrete and numeric class machine learning. In *Proc. International Conference on Machine Learning*, pages 359–366. Morgan Kaufmann, 2000.

[KJ97] R. Kohavi and G.H. John. Wrappers for feature subset selection. In *Artificial Intelligence*, volume 97, pages 273–324, 1997.

[Kon94] I. Kononenko. Estimating attributes: Analysis and extensions of RELIEF. In *Proceedings of the Seventh European Conference on Machine Learning*, pages 171–182, 1994.

[KR92] K. Kira and L. Rendell. A practical approach to feature selection. In *Proceedings of the 9th International Conference on Machine Learning*, pages 249–256, 1992.

[MM98] C. J. Merz and P. M. Murphy. UCI Repository for Machine Learning Data-Bases [http://www.ics.uci.edu/~mlearn/MLRepository.html]. *Irvine, CA: University of California, Department of Information and Computer Science*, 1998.

[Paw91] Z. Pawlak. *Rough Sets: Theoretical Aspects of Reasoning about Data.* Kluwer Academic Publishers, 1991.

[Qui93] R. Quinlan. *C4.5: Programs for Machine Learning.* Morgan Kaufmann Publishers, 1993.

[RS89] C.K. Riesbeck and R.C. Schank. *Inside Case-Based Reasoning.* Lawrence Erlbaum Associates, Hillsdale, NJ, US, 1989.

[SGVN00] M. Salamó, E. Golobardes, D. Vernet, and M. Nieto. Weighting methods for a Case-Based Classifier System. In *LEARNING'00*, Madrid, Spain, October 2000. IEEE.

[WF00] I. H. Witten and E. Frank. *DataMining: practical machine learning tools and techniques with Java implementations.* Morgan Kaufmann Publishers, 2000.