A Feature Selection Approach in the Study of Azorean Proverbs

Luís Cavique,
Universidade Aberta, Portugal, luis.cavique@uab.pt
Armando B. Mendes,
Universidade dos Açores, Portugal, amendes@uac.pt
Matthias Funk,
Universidade dos Açores, Portugal, mfunk@uac.pt
Jorge M.A. Santos,
Universidade Évora, Portugal, jmas@uevora.pt

ABSTRACT
A paremiologic (study of proverbs) case is presented as part of a wider project based on data collected among the Azorean population. Given the considerable distance between the Azores islands, we present the hypothesis that there are significant differences in the proverbs from each island, thus permitting the identification of the native island of the interviewee, based on his or her knowledge of proverbs. In this chapter, a feature selection algorithm that combines Rough Sets and the Logical Analysis of Data (LAD) is presented. The algorithm named LAID (Logical Analysis of Inconsistent Data) deals with noisy data, and we believe that an important link was established between the two different schools with similar approaches. The algorithm was applied to a real world dataset based on data collected using thousands of interviews of Azoreans, involving an initial set of twenty-two thousand Portuguese proverbs.

Keywords: data mining, feature selection, logical analysis of data, rough sets, paremiology

1. INTRODUCTION
Proverb, "proverbium" in Latin, can be defined as a condensed saying with popular roots, recorded by an anonymous author and expressed by a minimal text, which is generally known and is based on oral tradition of a particular region, such as: "an apple a day keeps the doctor away".

This study is based on several paremiologic works. Paremiology is the science that deals with the description, classification, etymology and pragmatics of proverbs. One of these works is the relevant collection of three books about the "Pearls of the Portuguese Popular Wisdom" (Funk, Funk 2001a), (Funk, Funk 2001b), (Funk, Funk 2003).

In a series of interviews, several million records were collected from thousands of people denoting whether or not they recognized Portuguese proverbs, based on an initial set of twenty-two thousand proverbs. This constitutes a unique source for a socio-cultural analysis of the transmission mechanisms involved in oral culture in geographically separated places.

Two forms of knowledge validation were used: passive and active. In passive recognition, the interviewer read the proverb and the interviewee stated whether he recognized it. In active recognition, the interviewer read only the initial part of the proverb and the interviewee completed it. For example, the interviewer began by reading "An apple a day…" and the interviewee completed it with "…keeps the doctor away".

This case study is based on data collected in eleven geographically separated areas inside the Azorean community cultural space. This community lives in the Portuguese Azorean archipelago located in the mid-Atlantic rift. In this particular case, it is interesting to analyze the relationship between local and overall knowledge within a common linguistic and cultural space. On the one hand, there is the geographical distance and isolation brought about by the natural sea barrier of this archipelago composed of nine inhabited islands. On the other hand, this archipelago not only extends over 2,330 km2, but it also spreads over a 630 km rectangle in a west-east direction and a 130 km in a north-south direction. The original groups can be seen in three main geographical clusters composed of the occidental, central and oriental groups as can be seen in Figure 1.

We can find some geographical continuity in the Central group, which is composed of 5 islands that are relatively close to each other: Faial (15,063 inhabitants), Pico (14,806), São Jorge (9,674), Terceira (55,823) and Graciosa (4,780). The same is true of the two Occidental islands, Corvo (425) and Flores (3,995). However, the Oriental group composed of Santa Maria (5,578) and São Miguel (131,609) is separated by 80 km.

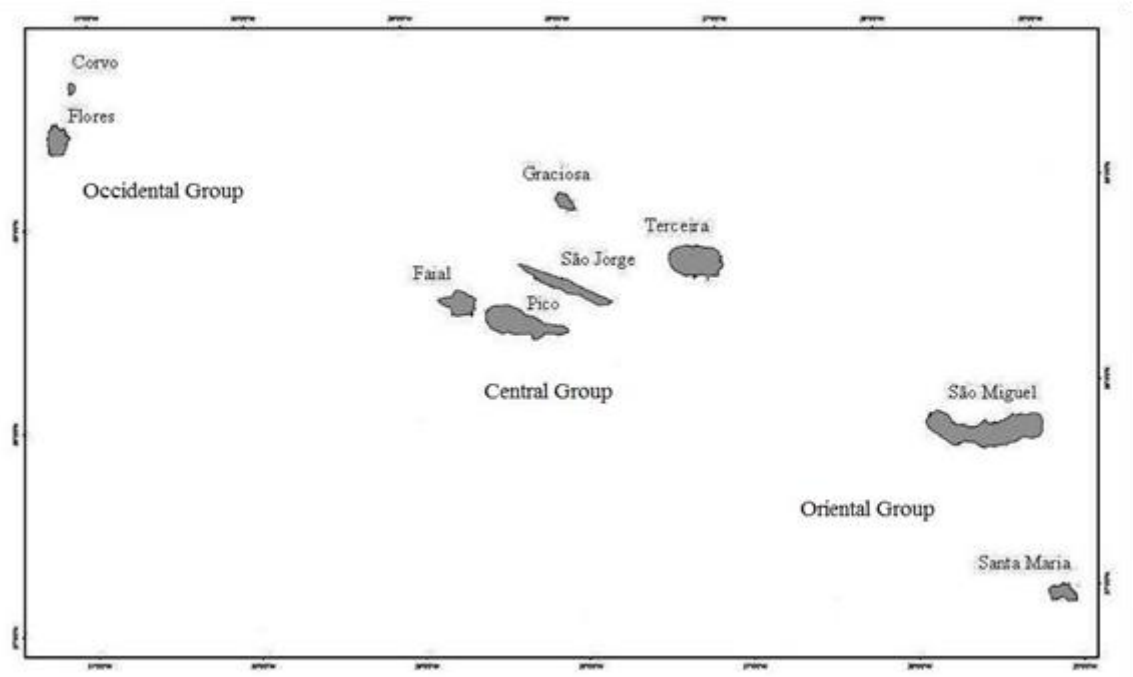

Fig.1. The Azores Archipelago showing the geographical distribution of the nine islands

Due to the significant waves of emigrants from the Azorean archipelago entering the United States, from the end of the 19th century until the end of the 20th century, twice as many emigrants left the archipelago as those who remained there, about 250,000 inhabitants. The population flux also includes the Azorean migration, mainly characterized by the attraction to urban centers, namely the former administrative capitals: Ponta Delgada in São Miguel, Angra in Terceira and Horta in Faial.

On small islands like Corvo, Santa Maria, Graciosa, Flores and São Jorge, the population shows low mobility as the large majority lived their entire life on only one island. On the other hand,

the islands with local capitals are characterized by higher mobility rates, probably because they are deemed more attractive by the population of the surrounding islands. By "mobile" we are referring to the persons that lived on at least two different islands or other locations outside the Azorean archipelago for an uninterrupted period of five years.

In this chapter, the proverb recognition surveys were collected on nine Azores islands and in two regions of emigration to the United States: California and New England.

The data was collected between 1997 and 2000 and published in (Funk and Funk, 2001a) and (Funk and Funk, 2001b) the studies of the Azorean proverbs in the USA and in São Miguel. The same authors (Funk and Funk, 2003) also published proverbs of the Azores central group. All these data were collected and recorded in a relational database, named "Knowledge of the Azorean Proverbs 2000". Finally, the database was restructured, cleaned and statistically analyzed, using simple description statistics, hypothesis testing and cluster analysis, as described in a previous publication (Mendes, Funk, Funk 2009). This work is part of a larger project based on the oral tradition of proverbs published in (Mendes, Funk, Cavique 2010).

Given the widely scattered Azores islands, in this chapter we aim to highlight the following hypothesis.

H1: there are significant differences in the sayings from each island, thus it is possible to identify the native island of an interviewee based on his/her knowledge of proverbs.

The purpose of this chapter is to find the minimum number of proverbs that allows the identification of the native island of an interviewee, based on the assumption of H1 which states that there are significant differences in the proverbs from each island. This is clearly a Feature Selection problem, where the classes are islands, the attributes are proverbs and the observations are interviewees.

The concept of native island was extended to locations where the interviewees lived for more than five years, thus unintentionally introducing inconsistencies in feature selection and in the data classification task. The mobile persons introduce an obstacle to the feature selection process, *i.e.* the same person with the same proverb knowledge is classified in different classes as they lived in different places. To overcome this handicap, we use an approach based on Rough Sets (Pawlak 1991), which are tolerant of these types of inconsistencies.

Rough Sets differ from classical sets in their ability to deal with inconsistent data. Rough Sets have a particular application in feature selection since they filter the attributes while keeping the underlying semantics of the data. A parallel approach of feature selection is the Logical Analysis of Data, LAD, (Crama et al. 1988), (Boros et al. 2000). The LAD procedure can be divided into two steps: first, transform the data reduction problem into an optimization problem, and then solve the transformed problem as a minimum set covering problem.

In this chapter, we combine both techniques, by using the flexibility of Rough Sets and the straightforwardness of LAD. The known LAD handicaps, like the inability to cope with the contradictions and the limited number of classification classes, will be overcome using this new approach we have called Logical Analysis of Inconsistent Data (LAID).

The main contribution of this work is the development of a Feature Selection algorithm which combines Rough Sets and LAD characteristics, named LAID because it deals with noisy data. We believe that an important link was established between the two different schools with similar approaches. The algorithm was applied to a real world dataset with data collected from the Azorean population and involving a large set of proverbs.

The chapter is organized in six sections. In section 2, two parallel ways of feature selection are compared, the Rough Sets and LAD. In section 3, we present LAID algorithm, which combines the flexibility of Rough Sets and the efficiency of LAD. Section 4 is dedicated to LAID validation techniques and the method results are presented in section 5. Finally, in section 6 we draw some conclusions.

2. FEATURE SELECTION

Given that every 20 months, the amount of information in the world doubles, the motivation for feature selection is to reduce the dimension of the feature space. The reason is associated with Occam's razor principle and aims to obtain the simplest model. There are two basic models in feature selection, Filter model and Wrapper model.

Filter Model is divided into two sequential steps. The feature selection step is executed before the prediction model and there is no interaction between the selection and the prediction model. The best-known algorithms are FOCUS (Almuallim, Dietterich 1991) and RELIEF (Kira, Rendell 1992). FOCUS algorithm uses the concept of conflict, that is, two examples with the same feature values but different class values. The algorithm searches subsets with a minimum number of conflicts. RELIEF algorithm evaluates a feature subset based on the difference between the distance from the nearest example of the same class and the nearest example of a different class.

Wrapper model (John, Kohavi, Pfleger 1994) is also divided into two steps, but there is a strong interaction between the feature selection and the prediction model, where the results of the prediction are used as a criterion of choice of the features.

Filter models present a faster performance since they only build one solution and are more intuitive. On the other hand, they present the disadvantages of ignoring the model, i.e. most of the relevant features might not be optimal in the prediction model and the selection criterion is hard to estimate.

In the Wrapper models, the selection criterion is easy to estimate since the features are chosen by the prediction model and for the same reason they are classified as model-aware, i.e. they incorporate the knowledge of the predictor. They also present the opposite disadvantages of the Filter models, which are computationally too expensive and are not intuitive. In other words, the Wrapper models do not identify statistical dependency, so the features might not be the most explanatory variables and therefore the model lacks some theoretical basis.

To sum up, in the Filter models, the selection criterion is hard to estimate, whereas the Wrapper models tend to destroy the underlying semantics of the features after reduction.

It would be highly desirable to find a theory that could not only reduce the number of features, but also preserve the data semantics. In this context, Rough Set theory emerges as the desired tool by discovering the data dependencies and reducing the dimension (Pawlak 1991). In parallel, Peter Hammer's group (Crama et al. 1988), (Boros et al. 2000), with works in discrete optimization, developed the LAD approach. The key features of LAD are the discovery of the minimum number of attributes that are necessary for explaining all observations and the detection of hidden patterns in a dataset with two classes.

Rough Sets and LAD approaches are a subset of Filter models that aim to reduce the number of attributes of datasets using the same two phases. Their specificity is to keep the semantics of the data by removing only the redundant data based on a combinatorial optimization problem.

Although the methods have many similarities, the works that compare the two approaches are scarce. A very recent book (Chikalov et al. 2013) presents three similar approaches to data analysis: Test Theory, Rough Sets and Logical Analysis of Data.

2.1. Rough Sets

Rough Sets theory was initially proposed as a tool to reason about vagueness and uncertainty in information systems by Pawlak (1982) and later it was also proposed for attribute selection by Pawlak (1991). The applications of the Rough Sets method are wide; it leads to significant results in many fields, such as conflict analysis, finance, industry, multimedia, medicine, and most recently bioinformatics (Polkowski 2002) (Peters, Skowron 2010). Below we review the basics of Rough Sets.

A dataset D={O, X∪C} where the observations O={o_1,o_2,…,o_n} is a non-empty set of objects (observations, cases or lines), X={x_1,x_2,…,x_m} is a non-empty set of attributes and C is the class attribute. The following table will serve as a running example in this section.

D=

O	**x_1**	**x_2**	**x_3**	**x_4**	**C**
o_1	1	1	0	1	1
o_2	1	0	1	0	0
o_3	1	0	1	1	1
o_4	1	0	1	0	1
o_5	0	1	1	1	2
o_6	0	1	1	1	2

In a Rough Set table values other than the binary values are allowed. Note also that the table has redundant values (o_5 and o_6) and inconsistent values (o_2 and o_4). By inconsistencies we mean, two cases having the same values for all attributes, but belonging to different decision classes D. A practical example is two sick people that have the same symptoms but different diseases. With real data this is possible, because the table might have a missing attribute that could discriminate between them.

Rough Sets do not correct or exclude the inconsistencies, but rather for each class they determine a lower and an upper approximation. Given D={O,X∪C}, the subset of objects Y⊆O and the subset of attributes B⊆X, Pawlak's Rough Sets theory defines two approximation spaces: the lower and upper rough approximation. The lower approximation $B_L(Y)$ is the least composed set that is contained in Y, and the upper approximation $B^U(Y)$ is the greatest composed set that contains Y. Example, for B=X={x_1,x_2,x_3,x_4} and Y={o_1,o_3,o_4} with C=1, the lower and upper approximations are, $B_L(Y)$={o_1,o_3} and $B^U(Y)$={o_1,o_2,o_4,o_3}.

As a consequence of the approximation space $B_L(Y)⊆Y⊆B^U(Y)$. Also, the lower and upper approximations of a subset Y⊆O can be seen as operators in the universe of objects *O* that divides it into three disjoint regions, the positive region POS(Y), the negative region NEG(Y) and the boundary region BR(Y):

- $POS(Y) = B_L(Y)$
- $NEG(Y) = O - B^U(Y)$
- $BR(Y) = B^U(Y) - B_L(Y)$

In the example, the decision class is rough since the boundary region is not empty, BR(Y)= {(o_2,o_4)}.

In Figure 2 the lower and upper bound of subset Y with C=1 are shown, where the grey area represents the Rough Set. The observations of lower bound $B_L(Y)$ = {o_1,o_3} are entirely covered by the grey area and they are associated with only one class. The observations of boundary region BR(Y) = {o_2,o_4} has two colors, white and grey and they belong to different classes. And the upper bound $B^U(Y)$={o_1,o_2,o_4,o_3} is the union of the described subsets. The negative region NEG(Y)= {o_5,o_6} is also shown in the figure.

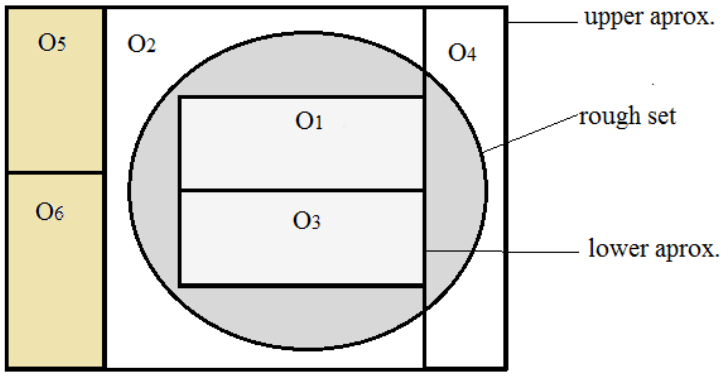

Fig.2. Lower and upper bound

When the lower and upper approximations are equal, $B_L(Y)=B^U(Y)$, there are no inconsistencies and the rough set is called crispy rough set.

Another way to identify the roughness of the set is using measures. The accuracy approximation measure is given by:

$$\alpha(Y) = \frac{|B_L(Y)|}{|B^U(Y)|}$$

where |Y| denotes the cardinality of $Y\neq 0$ and $0\leq\alpha(Y)\leq 1$. If $\alpha(Y)=1$, X is crisp; otherwise, it is a rough set.

The goal of Rough Sets is to discover decision rules from dataset D. The minimum number of attributes needed to explain all the classes need to be found. In other words, the number of attributes must be reduced in order to find the core ones. Discovering the minimum number of attributes is an NP-hard problem. One of the following techniques is normally used: Reduction by Heuristics or Discernibility Matrix.

In Reduction by Heuristics, the search for a core is given by the following procedure: for each iteration, one attribute is removed and the augmentation of inconsistency is verified. As already referred, inconsistency occurs when two or more observations have the same values in all attributes, but belong to different decision classes. If the inconsistency does not increase, the attribute can be removed. When no further attributes can be removed, the remaining ones are considered indispensable and thus the core is found.

Using a discernibility matrix of T, denoted by M; an (n×n) matrix is defined as follows, where $M(i,j)=\varnothing$ denotes that this case does not need to be considered.

$$M(i,j)=\begin{cases}\{x\in X: x(o_i)\neq x(o_j)\} & \text{if } c(o_i)\neq c(o_j)\\ \varnothing & \text{otherwise}\end{cases}$$

The discernibility matrix keeps the distinct attributes for each pair of observations belonging to different classes. In our example, the discernibility matrix M is as follows:

	o_1	o_2	o_3	o_4	o_5	o_6
o_1	-					
o_2	x_2,x_3,x_4	-				
o_3	∅	x_4	-			
o_4	∅	∅ (inconsistency)	∅	-		
o_5	x_1,x_3	x_1,x_2,x_4	x_1,x_2	x_1,x_2,x_4	-	
o_6	x_1,x_3	x_1,x_2,x_4	x_1,x_2	x_1,x_2,x_4	∅ (redundancy)	-

Note that for the pair (x_2, x_4) the result of the matrix is empty due to the inconsistency of the data. On the other hand, the pair (x_5, x_6) is an empty-set due to the redundancy of the data.

Discernibility function F(B) is a Boolean function, written in the disjunctive normal form (DNF), that is a normalization of a logical formula which is a conjunction of disjunction clauses. F(B) determines the minimum subset of attributes that allows the differentiation of classes: $F(B)= \wedge\{\vee M(i, j): i, j= 1, 2, \ldots, n; M(i,j) \neq \varnothing\}$.

The F(B) decision problem is equivalent to the Satisfiability problem (SAT), which was the first known example of an NP-complete problem.

In our running example: $F= (x_2 \vee x_3 \vee x_4) \wedge (x_1 \vee x_3) \wedge (x_4) \wedge (x_1 \vee x_2 \vee x_4) \wedge (x_1 \vee x_4)$. The solution for the reduction of the attributes is, $x_1=1$, $x_2=0$, $x_3=0$ and $x_4=1$, where the core= $\{x_1, x_4\}$ and the attributes x_2 and x_3 are redundant. Consequently, the decision rules are:
if (x_1=1) and (x_4=1) then C=1;
if (x_1=1) and (x_4=0) then C=0;
if (x_1=1) and (x_4=0) then C=1;
if (x_1=0) and (x_4=1) then C=2;

Rough Sets do not exclude or correct the inconsistencies of the data, permitting discordant output decision rules. For instance (if (x_1=1) and (x_4=0) then C=0) and (if (x_1=1) and (x_4=0) then C=1), making it difficult for the final user to interpret the results.

2.2. Logical Analysis of Data (LAD)

The LAD method developed by P. Hammer´s group refers to the discovery of the minimum number of attributes that are necessary for explaining all observations and the detection of hidden patterns in a dataset with two classes.

The method works on binary data. Let D be the dataset of all observations, then each observation is described using several attributes, and each observation belongs to a class.

An extension of the Boolean approach is needed when nominal non-binary attributes are used. The binarization (or discretization) of these attributes is performed by associating to attribute x, the value v_s, a Boolean variable $b(x, v_s)$ such that:

$$b(x, v_s) = \begin{cases} 1 & \text{if } x = v_s \\ 0 & \text{otherwise} \end{cases}$$

Dataset D is given as a D^+ set for "positive" observations and as a set D^- set for "negative" observations, where $D = D^+ \cup D^-$ and the sets are disjoint $D^+ \cap D^- = \varnothing$. Observations are classified as positive or negative based on a hidden function, and the goal of the LAD method is to approximate this hidden function with a union of intervals.

The following dataset will serve as a running example in this section, where $D^{+}=\{o_1,o_2\}$ and $D^{-}=\{o_3, o_4, o_5\}$.

$$D=\begin{vmatrix} \mathbf{O} & \mathbf{x_1} & \mathbf{x_2} & \mathbf{x_3} & \mathbf{x_4} & \mathbf{C} \\ o_1 & 1 & 1 & 0 & 1 & 1 \\ o_2 & 1 & 0 & 1 & 1 & 1 \\ o_3 & 0 & 1 & 1 & 1 & 0 \\ o_4 & 1 & 0 & 1 & 0 & 0 \\ o_5 & 0 & 0 & 1 & 0 & 0 \end{vmatrix}$$

To guarantee disjointness of D^{+} and D^{-}, let us compare $o_2=(1,0,1,1)\in D^{+}$ and $o_3=(0,1,1,1)\in D^{-}$.

In order to identify the redundant attributes we are going to transform the variables *x*. In the transformation problem, the variable *x* will be transformed into a new variable *y*. To keep the differences between o_2 and o_3, $y_1=1$ or $y_2=1$, because $x_1(o_2)\neq x_1(o_3)$ and $x_2(o_2)\neq x_2(o_3)$, so the constraint is expressed by $y_1+y_2\geq 1$. Similarly, $y_1+y_3\geq 1$, $y_2+y_3+y_4\geq 1$, $y_1+y_2+y_3+y_4\geq 1$, $y_4\geq 1$ and $y_1+y_6\geq 1$.

Keeping the reduction of attributes in mind and with this set of constraints, a combinatorial optimization problem can be applied. The minimal support set corresponds to the following linear programming formulation:

minimize $y_1+y_2+y_3+y_4$

subject to $y_1+y_3\geq 1$ (comparing o_1 with o_3)

$y_2+y_3+y_4\geq 1$ (comparing o_1 with o_4)

$y_1+y_2+y_3+y_4\geq 1$ (comparing o_1 with o_5)

$y_1+y_2\geq 1$ (comparing o_2 with o_3)

$y_4\geq 1$ (comparing o_2 with o_4)

$y_1+y_6\geq 1$ (comparing o_2 with o_5)

and $y_i\in\{0,1\}$, $i=1,\ldots,4$

In order to systematize the process, a disjoint matrix [a(i,j)] will be defined and applied in a well-established optimization problem.

By a disjoint matrix [a(i,j)], we mean a matrix with at most $\underline{n.(n-1)}/2$ constraints and with *m* attributes, defined as:

$$a(i,j)=\begin{cases} 1 & \forall i:\ x_j(o_a)\neq x_j(o_b),\ C(o_a)\neq C(o_b),\ (o_a,o_b)\in O\times O \\ 0 & \text{otherwise} \end{cases}$$

The dimension of index *i* in matrix [a(i,j)] depends on the constraint structure of data set D. However, the upper bound of variable *i* is (n.(n-1))/2 constraints, due to the comparison of pairs of observations (o_a,o_b). Each constraint results from the comparison of two different arbitrary observations o_a and o_b that belong to distinct classes. If one attribute *j* is different in the observations o_a and o_b the value of a(i,j) is assigned with 1, denoting that at least one column (attribute) *j* must be maintained in order to differentiate the rows (constraint) *i*. The optimization problem that finds the minimum number of columns that covers all the rows is the Set Covering problem.

In experimental research where each experiment is represented by an attribute, the expense of each experiment can be included in the optimization model. So, for each attribute y, an expense can be associated by using a vector e_j, allowing a cost differentiation among attributes.

The disjoint matrix and the expense vector are then used in the set covering problem, defined as:

$$\text{minimize } z = \sum e_j y_j$$
$$\text{subject to } \sum a_{i,j} y_j \geq 1$$
$$\text{and } y_j \in \{0,1\} \qquad j=1,\ldots,m$$

The Set Covering problem is a very well-studied problem in Combinatorial Optimization, with many computational resources which implement quasi-exact algorithms and heuristic approaches.

For the given example, the minimal support set is $\{y_1, y_4\}$, so the columns 1 and 4 will be chosen, and the new dataset D^* is as follows:

$$D^* = \begin{vmatrix} \mathbf{O} & \mathbf{x_1} & \mathbf{x_4} & \mathbf{C} \\ o_{1,2} & 1 & 1 & 1 \\ o_3 & 0 & 1 & 0 \\ o_4 & 1 & 0 & 0 \\ o_5 & 0 & 0 & 0 \end{vmatrix}$$

3. LOGICAL ANALYSIS OF INCONSISTENT DATA (LAID)

3.1 Motivation

The goal of Rough Sets and LAD is to reduce the number of attributes and subsequent generation of rules in order to classify the given dataset. Both procedures can be divided into two steps: first, the transformation step and second, the reduction of the number of attributes.

The classic LAD approach uses two non-intersected classes and binary values for the attributes. This method has the drawback of only working with dichotomous attributes, which can be overcome with the discretization of the attribute values. In contrast, the Rough Sets support inconsistency, many classes and different nominal attribute values.

An advantage of LAD over Rough Sets is the possibility of using expenses associated to the attributes by minimizing not only the number of attributes but also the global cost.

Since Rough Sets do not exclude inconsistencies from real data, a large number of rules are generated, thus the interpretation of the results may become difficult. On the other hand, LAD presents a systematic, accurate, robust and flexible approach that avoids ambiguities and is easy to interpret by its users.

To summarize, Rough Sets are more flexible and diffuse and LAD more basic and straightforward. By combining the two approaches, we propose LAID, developed in the next section, which blends the best features of the presented methods. The LAID method should deal with integer attributes associated with costs such as LAD, and is tolerant to inconsistency and deals with more than two classes like Rough Sets.

In the following sub-sections, the inconsistency tolerance and the capacity to deal with many classes will be reported for the LAID algorithm.

3.2. “Deroughfication”

The driving idea of this work is to solve an inconsistency, created by the way the sample was developed, allowing an interviewee to belong to more than one class.

In a medical diagnosis, it is possible for two sick people to have the same symptoms but different diseases. To overcome this situation, medical doctors run one more test to identify the disease. In our approach, the solution will be similar, where the new test corresponds to a new attribute in the dataset. For each inconsistency, a dummy binary variable will be added to explain “je ne sais quoi” that should be tested, in such a way that the LAD procedure could be used without the need for any change. This approach values the simplicity principle over complexity in understanding the contradictory rules of Rough Sets.

If two observations are repeated, but belong to different classes, then a new dummy binary variable is needed. If three or four observations belonging to distinct classes are repeated, then two new dummy binary variables must be added. So, this number of unexplained variables is equal to the logarithm, base 2, of the number of repeated observations within a diverse class. This basic approach, besides the removal of the inconsistencies, avoids the upper and lower approximation approach, which is a very relevant research field in Rough Sets theory (Yao, 2007). In the following lines the link between lower and upper rough approximations and the dummy binary variable will be established.

Figure 3, on the left, shows the data present in subsection 2.1, using orthogonal areas o_1 to o_6, where the grey area represents the boundary of the Rough Set. For class C=1, Y=$\{o_1,o_3,o_4\}$, the lower approximation $B_L(Y)=\{o_1,o_3\}$ is completely shaded in grey, the upper approximation $B^U(Y)=\{o_1,o_2,o_3,o_4\}$ combines grey and white colors and the boundary region $BR(Y)= B^U(Y)–B_L(Y) =\{o_2,o_4\}$.

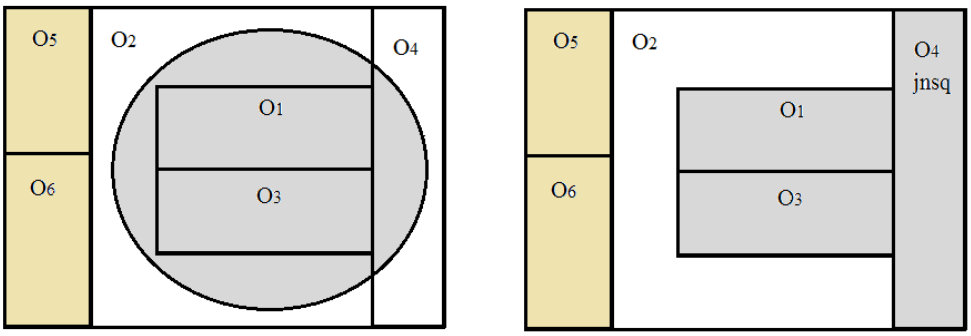

Fig.3. Rough Set (left) and Deroughficated Set (right)

As already defined, the boundary region BR, is given by the upper approximation minus the lower approximation. The process to remove the roughness, or “deroughfication”, is implemented by adding the dummy binary variable “je ne sais quoi” which is given by the intersection of BR and class D=1. For the given example, the dummy binary variable can be established as follows:

$$jnsq_i = \begin{cases} 1 & \text{if } (o_i \in BR) \text{ and } (class=1) \\ 0 & \text{otherwise} \end{cases}$$

Figure 2, on the right, shows the Deroughficated Set, where for observation 4, the $jnsp_4$ was set equal to 1. In the Deroughficated Set, the grey area belongs to class D=1 and the white area to class D≠1.

As with Rough Sets, the LAID method is tolerant with inconsistencies. We believe that a hidden property causes the inconsistency of the classification. This issue is overcome by introducing the dummy binary variable. At this stage the problem is no longer inexact (or rough) and a straightforward method, like LAD, can be applied.

3.3. Two Phase Algorithm

In order to implement the reduction of the dataset, a two-phase algorithm is presented. First, the problem is transformed by generating a matrix with a disjoint constraint. Second, a minimal subset of attributes is chosen using an algorithm for the Set Covering Problem.

Procedure 1: A Two-Phase Algorithm
Input: dataset D with dummy binary variables
Output: minimal subset of attributes S
1. Disjoint Constraints Matrix Generation
2. Algorithm for the Set-Covering Problem

The reduced dataset is obtained by the projection of the minimal subset of attributes. The number of lines in the dataset is also reduced by removing the repeated observations in the new reduced set of attributes.

3.4. Disjoint Matrix Generation

Dataset $D=\{O, X \cup C\}$ has a relational database table format, with a key-column observation O and remaining columns attributes X_i and class C. Each class has a set of observations and each observation is measured by a set of attributes.

LAD only deals with two classes. The proposed disjoint [a(i,j)] matrix generation works with an unlimited number of classes. The procedure is described as follows:

Procedure 2: Disjoint Matrix Generation
Input: Dataset $D=\{O, X \cup C\}$
Output: Matrix [a(constraint, attribute)]
1. For all observations o_i: i=1,...,n
2. class=c_i;
3. For all observations o_j: j=i+1,...,n, with $c_j \neq$class
4. constraint = succ(constraint);
5. For all attributes x_k: k=1,...,m
6. if ($x_k(o_i) \neq x_k(o_j)$) a(constraint, k)=1
7. else a(constraint, k)=0;
8. End for
9. End for
10. End for

Given that *n* is the number of observations, the upper bound for the number of constraints is (n.(n-1))/2, resulting from the comparison of all pairs of observations.

A disjoint matrix [a(i,j)] will be used as input in the Minimum Set Covering problem, where all constraints must be covered, at least once by some attributes.

3.5. Minimum Set Covering Problem

In this section, a heuristic approach is presented for the Minimum Set Covering Problem. The set covering heuristic, proposed by Chvatal (1979), is described in the following pseudo-code.

In the Linear Integer Programming formulation we can identify the matrix [a(i,j)] and the vector [c(j)]. We consider the following notation: [a(constraint, attribute)] for the input constraint matrix, [e(attribute)] for the expense of each attribute, and S for the set covering solution.

Procedure 3: Heuristic for the Set-Covering Problem
Input: [a(constraint, attribute)], [e(attribute)]
Output: the minimum set cover S
1. Initialize R=[a(i,j)], S=∅
2. While $R \neq \emptyset$ do
3. Choose the best line $i^* \in R$ such that $|a(i^*,j)|=\min |a(i,j)|, \forall j$
4. Choose the best column j* that covers line i*, considering f(e,j)
5. Update R and S, $R=R \setminus a(i,j^*), \forall i, S=S \cup \{j^*\}$
6. End while
7. Sort cover S by descending order of costs
8. For each Si do if (S\Si is still a cover) then S=S\Si
9. Return S

In this constructive heuristic, for each iteration, a line is chosen to be covered. Next, the best column that covers the line and finally, solution S is built and the remaining matrix R updated. The chosen line is usually the line that is more difficult to cover, i.e. the line that corresponds to fewer columns. After reaching the cover set, the second step is to remove redundancy, by sorting the cover in descending order of cost and checking if each attribute is really essential.

This constructive heuristic is improved by using a Tabu Search heuristic that removes the most redundant columns and re-builds a new solution as presented in (Cavique, Rego, Themido 1999) and (Gomes, Cavique, Themido 2006).

The Tabu Search procedure used for improving the Set Covering Problem solutions can be described as follows:

- J is the set of all columns
- J* is the set of columns in the current solution S
- F(S) is the objective function value of solution S
- S* is the best solution found so far
- TL is the tabu list
- Two neighbourhood structures are used:
- the constructive $N^+(S) = \{S': S' = S \cup \{j\} \text{ with } j \in J \setminus J^* \text{ and } j \notin TL\}$
- and the obliterate $N^-(S) = \{S': S' = S \setminus \{j\} \text{ with } j \in J^*\}$

Procedure 4: Tabu Search for the Set-Covering Problem
Input: J, J*
Output: the minimum set cover S*
1. Set TL = ∅;
2. Construct initial solution S; (Proc.3)
3. Set S* = S;
4. Repeat for k iterations:
5. Set S"=S;
6. Repeat $S' \in N^-(S")$, S"=S', $TL=TL \cup \{j\}$ until end condition;
7. Repeat $S' \in N^+(S")$, S"=S' until S' is a cover;
8. If F(S') > F(S*) then set S* = S';
9. Set S=S';
10. Update J and J*;
11. End repeat

3.6. Numeric Example

In this LAID numeric example, dataset D will be used as a running example in this section.

$$D=\begin{vmatrix} \mathbf{O} & \mathbf{x_1} & \mathbf{x_2} & \mathbf{x_3} & \mathbf{x_4} & \mathbf{C} \\ o_1 & 1 & 0 & 1 & 0 & 0 \\ o_2 & 1 & 1 & 0 & 1 & 1 \\ o_3 & 1 & 0 & 1 & 1 & 1 \\ o_4 & 1 & 0 & 1 & 0 & 1 \\ o_5 & 0 & 1 & 1 & 1 & 2 \\ o_6 & 0 & 1 & 1 & 1 & 2 \end{vmatrix}$$

As already noted, dataset D has redundancy in observations o_5 and o_6, and inconsistencies in observations o_1 and o_4. As proposed in the previous section, the "deroughfication" is obtained by adding a dummy binary attribute to allow the differentiation of observations o_1 and o_4. With these adaptations, dataset D' is as follows:

$$D'=\begin{vmatrix} \mathbf{O} & \mathbf{x_1} & \mathbf{x_2} & \mathbf{x_3} & \mathbf{x_4} & \mathbf{jnsq} & \mathbf{C} \\ o_1 & 1 & 0 & 1 & 0 & 0 & 0 \\ o_2 & 1 & 1 & 0 & 1 & 0 & 1 \\ o_3 & 1 & 0 & 1 & 1 & 0 & 1 \\ o_4 & 1 & 0 & 1 & 0 & 1 & 1 \\ o_{5,6} & 0 & 1 & 1 & 1 & 0 & 2 \end{vmatrix}$$

As stated in 2.2., with the LAD Disjoint Matrix procedure, matrix [a(i,j)] is obtained, by comparing each pair of observations from different classes, such as:

$$[a_{i,j}]=\begin{bmatrix} 0 & 1 & 1 & 1 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \\ 1 & 1 & 0 & 1 & 0 \\ 1 & 0 & 1 & 0 & 0 \\ 1 & 1 & 0 & 0 & 0 \\ 1 & 1 & 0 & 1 & 1 \end{bmatrix} \begin{matrix} \text{comparing } o_1 \text{ with } o_2 \\ o_1 \text{ with } o_3 \\ o_1 \text{with } o_4 \\ o_1 \text{with } o_{5,6} \\ o_2 \text{with } o_{5,6} \\ o_3 \text{with } o_{5,6} \\ o_4 \text{with } o_{5,6} \end{matrix}$$

In the original problem *x* variables are used while in the transformed problem *y* variables are used. When a(i,j)=1 it means that column y(j) is relevant to discriminate between the two observations reported in line *i*. In order to reduce the problem while maintaining the discrimination of the pairs of observations, for each line *i* at least one column with a(i,j)=1 must be chosen. Accordingly, by applying the set covering problem, variables y_1, y_4 and jnsq ensure the cover of all lines. The solution obtained is S={y_1,y_4, jnsq}.

The reduced dataset D* is given by the projection of variables x_1, x_4 and jnsq, as follows:

$$D^*=\begin{vmatrix} \mathbf{O} & \mathbf{x_1} & \mathbf{x_4} & \mathbf{jnsq} & \mathbf{C} \\ o_1 & 1 & 0 & 0 & 0 \\ o_{2,3} & 1 & 1 & 0 & 1 \\ o_4 & 1 & 0 & 1 & 1 \\ o_{5,6} & 0 & 1 & 0 & 2 \\ \_ & 0 & 0 & 0 & ? \end{vmatrix}$$

One last line was added to the dataset to show that entries x1=0 and x4=0 are not presented in the set of observations and therefore this entry is not explained.

Given the limited size of the example consisting of only two variables, all possible variable entries are presented except the unexplained area represented by symbol "?", as in table 1.

Table 1. Classes of the attribute x_1 and x_4

	$x_1=0$	$x_1=1$
$x_4=0$	?	0,1
$x_4=1$	2	1

We must refer that unexplained areas are different from inconsistent areas. Unexplained areas are a result of no observations, while inconsistent areas result from contradictory observations.

The validation of the feature selection method presented in the following sections, is a means used to measure the quality of the data reduction, and is based on the prediction of the unexplained areas.

4. LAID VALIDATION

The performance of a feature selection algorithm is usually measured by the computational results of a classification problem after the data reduction.

Suppose dataset D={O, X∪C} where O={o_1,o_2,…,o_n} is a non-empty set of observations (or instances), X={x_1,x_2,…,x_m} is a non-empty set of attributes and C is the decision attribute (or class), exemplified as follows:

$$D=\begin{vmatrix} \mathbf{O} & \mathbf{x_1} & \mathbf{x_2} & \mathbf{x_3} & \mathbf{x_4} & \mathbf{C} \\ o_1 & 1 & 0 & 0 & 0 & 0 \\ o_2 & 1 & 1 & 0 & 0 & 0 \\ o_3 & 1 & 0 & 1 & 1 & 0 \\ o_4 & 0 & 1 & 0 & 0 & 1 \\ o_5 & 0 & 1 & 1 & 0 & 1 \\ o_6 & 1 & 1 & 0 & 1 & 1 \end{vmatrix}$$

The general approach for solving classification problems split D into two datasets: the Training Set and the Test Set. The classification task is to identify which class a new Test Set observation belongs to, based on a Training Set which contains observations whose class is known.

In the predicting process, the original LAD algorithm used to distinguish between observations belonging to two disjoint classes uses the so-called *discriminant function*, which is similar to the k-nearest neighbor classification algorithm. To classify an unknown observation x, let P(x) and N(x) be associated with the positive and negative pattern, where Pos(x): $x \subseteq D^+$ and Neg(x): $x \subseteq D^-$, as follows: $\Delta(x)=|Pos(x)|-|Neg(x)|$. If $\Delta(x)>0$ then x is classified as 1, and 0 otherwise.

In the prediction class process, the discriminant function for *p* classes is expressed as:

$$\text{predicted\_class}(x)=\text{arg\_min}\left(H_0(x), H_1(x), H_2(x),\ldots,H_p(x)\right)$$

Observation *x* contains binary values of the *m* proverbs, showing knowledge of the proverb or lack of it. The function $H_i(x)$ returns a value that is an average of the Hamming Distance

between *x* and the known *i* observations. The function predicted_class(x) returns the class of the k-nearest neighbors of *x*. The function used to find and measure the nearest neighbors takes into consideration the Hamming Distance.

Using dataset D, defined in this section, the process to predict the class of x=(1,0,1,0), for k=3 is the following:

- The Hamming distance between the known observation o_1=(1,0,0,0) and x=(1,0,1,0) is 1, since there is a difference in the third element. This process is repeated for all observations.
- The value of function $H_0(x)$=(1+2+1)/3 and the value of $H_1(x)$=(3+2+3)/3.
- The predicted class is class 0, since it has the lowest average Hamming distance.

The method presented initially, that splits the dataset into two sets, has the drawback of not using all available samples. The major method for supervised learning models validation is the cross-validation, which uses all samples available. This is a very useful technique, involving the partition of the sample dataset into *n* subsamples, using *(n-1)* for repeated training and the remaining for testing. In this work, we adopt the special case of Leave-One-Out cross-validation, which consists in removing one observation from the original sample, training the algorithm and then, testing the observation using the resulting model.

The performance of a classification model is based on the total counts of correct and incorrect predictions. These counts are tabulated in a table known as Confusion Matrix. The Confusion Matrix is a specific square table, actual class versus predicted class, which allows for the visualization of the performance of an algorithm, see table 2.

Table 2. Confusion Matrix for a 2-class problem

		Predicted Class	
		Class=1	Class=0
Actual Class	Class=1	f_{11}	f_{10}
	Class=0	f_{01}	f_{00}

Given dataset D with *n* observations, and vector of classes C, the procedure to generate the Confusion Matrix is described as follows:

Procedure 5: Confusion Matrix
Input: Confusion-Matrix=∅, dataset D;
Output: Confusion-Matrix;
1. for i=1 to n-observations
2. for j=1 to n-observations
3. if i≠j // leave this out
4. update every $H_p(i)$;
5. end if
6. end for
7. predicted-class = arg_min($H_1(i)$, $H_2(i)$, …,$H_n(i)$);
8. update Confusion-Matrix with predicted-class;
9. update Confusion-Matrix with real-class;
10. end for

Based on the Confusion Matrix many performance measures can be extracted. In this work the accuracy or hit-rate measure will be used, expressed by:

$$\text{hit\_rate}=\frac{\text{number of correct predictions}}{\text{total number of predictions}}=\frac{f_{11}+f_{00}}{f_{11}+f_{10}+f_{01}+f_{00}}$$

Another way to evaluate a classifier consists in the use of Receiver Operating Characteristic, ROC curve (Fawcett 2006), which shows the performance of a two-class classifier. The overall performance of a classifier can be measured by Area Under the Curve, AUC. A scale for classifying the accuracy of AUC of the ROC curve is: excellent (0.91-1.00), good (0.81-0.90), fair (0.71-0.80), poor (0.61-0.70) and fail (0.51-0.60).

For a number of classes greater than two, a multi-class ROC analysis will be needed, using a multi-objective optimization function in order to find the Pareto front in the convex hull. In our approach we will use the Hit-rate comparison and Cohen's Kappa-statistic (Cohen 1960). When hit-rate is greater than the modal class assure that the classifier is better than a random classifier; by modal class we mean the most frequent class. Kappa-statistic measures the agreement using the input data of the Confusion Matrices for a number of classes larger than two, with the following scale: 0–0.20 as slight, 0.21–0.40 as fair, 0.41–0.60 as moderate, 0.61–0.80 as substantial, and 0.81–1 as perfect agreement.

5. LAID RESULTS

In this section the computational results of the data reduction and the chosen proverbs are presented. After setting the study datasets extracted from the "Knowledge of the Azorean Proverbs 2000" database, the computational performance of the algorithms and the test of the hypotheses are discussed. Finally, the chosen proverbs are shown.

5.1 The Study Datasets

The survey data was organized in a relational database called "Knowledge of the Azorean Proverbs 2000", which centralizes all information about the recognition of proverbs. This exhaustive survey was taken to clarify the relevance of over 22,000 Portuguese proverbs in the Azorean society and to calculate the percentage of recognition of these proverbs within the cultural community as a whole or within a single location.

The initial set includes different linguistic forms for the same proverb. For practical reasons, it was impossible to submit the 22,000 proverbs to a single person. Therefore, the corpus was reduced to 1,500 proverbs and divided into 14 packages. The database included 900 respondents and a group of 1,500 proverbs recognized in different locations.

The following relational database exemplified in figure 4, supports two large volume tables respondents-proverbs and respondents-locations. For the respondents a sampling procedure was used that controls the following factors: gender, three classes for age and two classes for education.

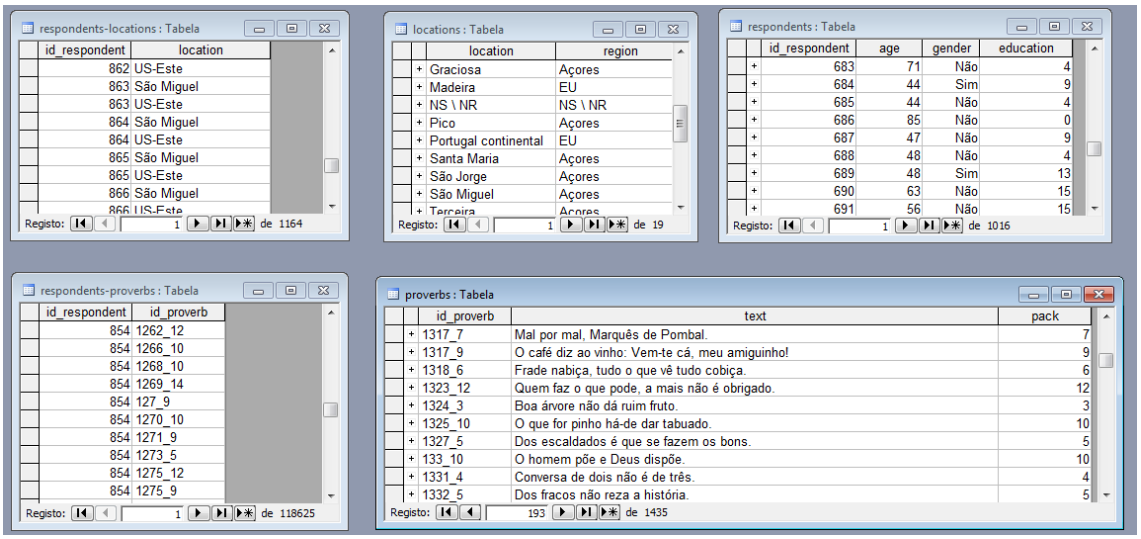

Fig.4. Example of "Knowledge of the Azorean Proverbs 2000"

The relational joint operation of the respondents table with the respondents-location table produces a duplication of the responses for the respondents with more than one location. In the classification table, the same respondent, with the same knowledge of proverbs, will be classified in different classes, which is a clear inconsistency. For each inconsistency a dummy binary variable was added.

This study uses the number 9 package from the "Knowledge of the Azorean Proverbs 2000" database, as the test dataset, with 240 respondents, 180 proverbs and 15,300 records in the table of person-proverb. In this dataset, the percentage of proverb recognition is 35%, i.e. 15300/(240x180).

Due to the poor results obtained in previous studies, we decided to enhance the persons with greater knowledge of proverbs by removing the respondent from the database who recognized less than 50% of the proverbs.

By applying these successive filters, only seven of the locations were selected from the Occidental and Central Group: Corvo, Flores, Faial, Graciosa, Pico, São Jorge and Terceira.

Two datasets were prepared to run the algorithms, the first with two classes, which corresponds to the two chosen groups and a second with 7 classes, which relates to the seven referred islands. See table 3.

Table 3. Datasets used in the study

Number of classes	**Description**
2	1- Ocidental group, 2- Central group
7	1- Corvo, 2- Flores, 3- Faial, 4- Graciosa, 5- Pico, 6 - São Jorge and 7 - Terceira

5.2. LAID Computational Results

To implement the computational results of the Two-Phase Algorithm, some choices had to be made, such as the computational environment and the performance measures. The computer programs were written in C language and the Dev-C++ compiler was used. The computational results were obtained using a 2.53GHz Intel Core-2Duo processor with 4.00 GB of main memory running under the Windows Vista operating system.

Table 4. Number of constraints and computing time

	Disjoint Constraint Matrix (DCM)			**Set Covering Heuristic (SCH)**	
Num Class	Num constraints	Time in seconds		Num attributes	Time in seconds
2	703	<1		6	<1
3	1,711	<1		9	<1
4	3,043	<1		11	<1
5	5,803	<1		12	<1
6	8,731	<1		14	1
7	12,965	<1		15	1

The performance measures for the first phase of the algorithm, the Disjoint Constraint Matrix (DCM), are the number of constraints and the time in seconds. For the second phase, the Set Covering Heuristic (SCH), the number of attributes and also the time in seconds were taken into account. In table 4, the computational results are presented, from class 2 to 7.

The number of constraints tends to grow exponentially with the number of classes (or birthplaces), while the growth of the number of attributes (or proverbs) and computational times remain linear, showing the good scalability of the algorithm, see figure 5.

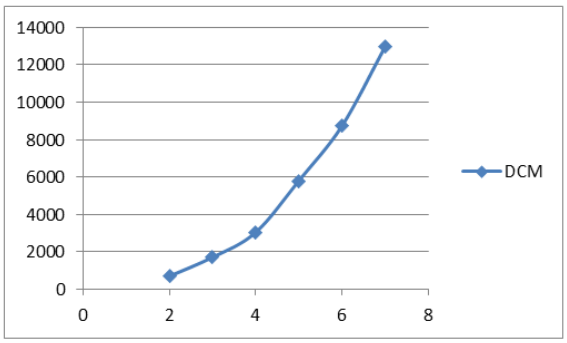

Fig.5. Growth of the number of constraints (DCM)

5.3 Testing the Hypotheses

Hypothesis H1, stated in the introduction, where «there are significant differences in the sayings of each island, so that it is possible to identify the native island of an interviewee based on his/her knowledge of proverbs», will be checked in this section for each dataset. So, H1 will be split into H2 and H3, as follows:

H2: there are significant differences in the sayings from the Occidental and Central group, so that it is possible to identify the group of islands of an interviewee's birthplace based on his/her knowledge of proverbs;

H3: there are significant differences in the sayings from each of the seven islands, so that it is possible to identify the native island of an interviewee based on his/her knowledge of proverbs;

To test the hypotheses, the ROC curve is very helpful for 2 classes. For 3 or more classes we are going to use the Hit-rate and the Kappa-statistic.

Given the Confusion Matrix 2x2, the ROC curve is the usual way to measure the performance of the classifiers. The AUC is equal to the probability that a classifier will rank a positive instance higher than a randomly negative one, so a random classifier has an AUC=0.5, and the ideal value is AUC=1.0. In the dataset for 2 classes, the Hit-rate of the Confusion matrix, the area under the ROC curve are the following: Hit-rate=89% and AUC=80%.

The values with a Hit-rate close to 90% and an AUC greater than 80% take us to an accepted H2, in that there are significant differences in the sayings from the Occidental and Central group.

Hit-rate and modal class classification are calculated in table 5, where we can see that the algorithm performs better with 2 classes than with 7 classes and where the hit-rate values overcome modal class classification in the first case. For the second case, with 7 classes the condition (hit rate > modal class) is false. It is clear that, as the number of classes increases, the classification procedure tends to degenerate its performance.

Table 5. Performance measures

Number of classes	AUC ROC	Hit rate	Modal class	Hit rate > Modal class	Kappa-statistic	Agreement
2	80%	89%	64%	True	67%	substantial
7	---	33%	48%	False	19%	slight

Cohen's Kappa measures the agreement between the classification performances of two algorithms when both are rating the same object. A value of 1 indicates perfect agreement while a value of 0 indicates that the agreement is no better than a random value. Also in table 5, the Kappa-statistic shows a substantial agreement for the dataset with 2 classes and a weak agreement for the dataset with 7 classes.

Finally, the four measures, Hit-rate, AUC, Hit-rate comparison and Kappa-statistic show that there are significant differences in the sayings from the Occidental and Central group, allowing us to conclude that H2 should be accepted. On the other hand, the three measures, Hit-rate, Hit-rate comparison with modal class and Kappa-statistic, show that the identification of the native island of an interviewee cannot be proved, as stated in H3.

5.4. The Chosen Proverbs

Given that hypothesis H3 is plausible, the results for the selected 2 locations (or classes), returned 6 proverbs as the minimum information needed to identify the birthplace of the interviewees, which are presented in table 6.

Analyzing the meaning of the 6 proverbs sounds bittersweet, because the chosen proverbs are not the best known, nor the most beautiful, but they are surely those that best differentiated the two groups.

In conclusion, the LAID algorithm reduces the number of attributes from 180 to a mere 6 proverbs in a few seconds.

Table 6. The six proverbs needed to differentiate the groups

Proverb id	**Proverb text**
010_9	In foreign land the ox is cow. (O boi em terra alheia é vaca.)
011_9	Do not let the rightful be the doubtful. (Não se deixa o certo pelo duvidoso.)
115_9	The expensive is cheap, cheap is expensive. (O caro é barato, o barato sai caro.)
121_9	The devil sets the trap. (O diabo tece-as.)
122_9	The devil has a blanket to cover him and another to uncover him. (O diabo tem uma manta com que cobre e outra com que se descobre.)
148_9	Nobody knows his dawn nor his dusk. (Ninguém sabe para que amanhece, nem para que anoitece.)
999_9	dummy binary variable « je ne sais quoi »

6. CONCLUSIONS

To sum up, we would like to clarify the major motivations and contributions in this chapter. The motivation was the paremiologic study, which aims to discover the native island of the interviewees. The major contribution of LAID was a new hybrid feature selection algorithm for inconsistent data that dealt successfully with the paremiologic challenge.

This work was part of a wider paremiologic project, based on data collected from thousands of interviews with people from the Azores, asking for the recognition of thousands of Portuguese proverbs, with the purpose of discovering the minimum information needed to guess the native island of an interviewee.

In this chapter we tried to highlight the hypothesis that, given the fact that the islands are widely scattered, there are significant differences in the sayings from each island, so that it is possible to identify the native island of an interviewee based on his/her knowledge of proverbs.

In the sample that included mobile persons, i.e. persons that lived in several locations for at least five years in each, there was some inconsistency (or class noise) introduced by the mobile persons, that made the problem more difficult to solve.

In this study, many unreported attempts were carried out using several feature selection and classification techniques but without success. The approach of combining Rough Sets and LAD was the most successful.

Although Rough Sets and LAD have many similarities, their comparison in the literature is scarce or nonexistent. A brief review of Rough Sets and LAD is presented and combined in the proposed LAID. This new technique includes the inconsistency tolerance and multiplicity of classes of Rough Sets, and the efficiency and attributes cost optimization of the LAD. Although Rough Sets do not exclude or correct the inconsistencies of the data, LAID does not exclude but correct the inconsistencies by adding dummy binary variables. The integration of both approaches is so tight that LAID can be seen as a Rough Set extension.

The paremiologic case study used a dataset with 240 interviewees (observations) and 180 proverbs (attributes), classified into 2 and 7 locations (or classes). The LAID algorithm reduces the number of attributes from 180 to a mere 6 proverbs in a few seconds.

The computational results showed that there are significant differences in the sayings from the Occidental and Central group, which allow us to conclude that H2 should be accepted, therefore it is possible to identify the group of islands of an interviewee's birthplace based on his/her knowledge of proverbs. On the other hand, a more detailed approach to identify the native island of an interviewee cannot be proved, as stated in H3.

Finally, we believe that an important link was established between Rough Sets and Logical Analysis of Data with the LAID method. In future work, we also intend to associate these techniques with the instance selection method, and explore the potential of these three strategies, of including, excluding or adding discerning variables to inconsistent data.

REFERENCES

Almuallim, H. & Dietterich, T.G. (1991). Learning with many irrelevant features, *Proceedings of the 9th National Conference on Artificial Intelligence* (pp. 547-552) MIT Press.

Beasley, J.E. & Jörnsten, K. (1992). Enhancing an algorithm for set covering problems. *European Journal of Operational Research*, 58, 293-300.

Boros, E., P.L. Hammer, T. Ibaraki, A. Kogan, E. Mayoraz, & I. Muchnik (2000). An Implementation of Logical Analysis of Data. *IEEE Transactions on Knowledge and Data Engineering*, 12(2), 292-306.

Cavique, L., C. Rego & I. Themido (1999). Subgraph Ejection Chains and Tabu Search for the Crew Scheduling Problem. *Journal of Operational Research Society*, 50(6), 608-616.

Chikalov, I., Lozin, V., Lozina, I., Moshkov, M., Nguyen, H.S., Skowron, A. & Zielosko, B. (2013). *Three Approaches to Data Analysis: Test Theory, Rough Sets and Logical Analysis of Data.* Series: Intelligent Systems Reference Library, vol. 41, Springer.

Chvatal, V. (1979). A greedy heuristic for the set-covering problem. *Mathematics of Operations Research*, 4, 233-235.

Cohen, J. (1960). A coefficient of agreement for nominal scales. *Educational and Psychological Measurement*, 20, 37-46.

Crama, Y., P. L. Hammer, & T. Ibaraki (1988). Cause-effect relationships and partially defined boolean functions. *Annals of Operations Research*, 16, 299-326.

Fawcett, T. (2006). An introduction to ROC analysis. *Pattern Recognition Letters*, 27, 861–874.

Funk, G. & M. Funk (2001a). *Pérolas da sabedoria popular: Os provérbios Açoreanos nos EUA.* Salamandra, Lisboa.

Funk, G. & M. Funk (2001b). *Pérolas da sabedoria popular: Os provérbios de S. Miguel*. Salamandra, Lisboa.

Funk, G. & M. Funk (2003). *Pérolas da sabedoria popular: Provérbios das Ilhas do Grupo Central dos Açores (Faial, Graciosa, Pico, São Jorge e Terceira).* Salamandra, Lisboa.

Gomes M., L. Cavique & I. Themido (2006). The Crew Time Tabling Problem: an extension of the Crew Scheduling Problem. *Annals of Operations Research, Optimization in transportation*, 144(1), 111-132.

John G.H., R. Kohavi & K. Pfleger (1994). Irrelevant Features and the Subset Selection Problem. *Proceedings of the 11th International Conference on Machine Learning,* ICML 94, (pp. 121-129).

Kira K.& L.A. Rendell (1992). The feature selection problem: traditional methods and a new algorithm. *Proceedings of Ninth National Conference on Artificial Intelligence*, (pp. 129-134).

Mendes, A., G. Funk & M. Funk (2009). Extrair Conhecimento de Provérbios. In M. F. Salgueiro (Eds.), *Temas em Métodos Quantitativos*, (pp. 89-107), Sílabo, Lisboa.

Mendes, A.B., M. Funk & L. Cavique (2010). Knowledge Discovery in the Virtual Social Network Due to Common Knowledge of Proverbs. *Proceedings of DMIN'10, the 6th International Conference on Data Mining,* (pp. 213-219), ISBN 1-60132-138-4.

Pawlak, Z. (1982). Rough Sets. *International Journal of Computer and Information Science*, 11, 341-356.

Pawlak, Z. (1991). *Rough Sets: Theoretical Aspects of Reasoning about Data*. Kluwer Academic Publishers, Boston.

Peters, J.F. & Skowron A. (2010). *Transactions on Rough Sets XI*. Lecture Notes in Computer Science / Transactions on Rough Sets, Springer.

Polkowski, L. (2002). *Rough Sets, Mathematical Foundations.* Advances in Soft Computing, Physica-Verlag Heidelberg, Germany.

Yao, Y. Y. (2007). Neighborhood Systems and Approximate Retrieval. *Information Sciences*, 176, 3431-3452.