Logical Analysis of Data as a Tool for the Analysis of Probabilistic Discrete Choice Behavior

Renato Bruni[a,*], Gianpiero Bianchi[b], Cosimo Dolente[c], Claudio Leporelli[a]

[a] *Dep. of Computer Control and Management Engineering, Sapienza Univer., Rome, Italy*
[b] *Dir. for Methodology and Statistical Process Design, Istat, Rome, Italy*
[c] *Fondazione Ugo Bordoni, Rome, Italy*

Abstract

Probabilistic Discrete Choice Models (PDCM) have been extensively used to interpret the behavior of heterogeneous decision makers that face discrete alternatives. The classification approach of Logical Analysis of Data (LAD) uses discrete optimization to generate patterns, which are logic formulas characterizing the different classes. Patterns can be seen as rules explaining the phenomenon under analysis. In this work we discuss how LAD can be used as the first phase of the specification of PDCM. Since in this task the number of patterns generated may be extremely large, and many of them may be nearly equivalent, additional processing is necessary to obtain practically meaningful information. Hence, we propose computationally viable techniques to obtain small sets of patterns that constitute meaningful representations of the phenomenon and allow to discover significant associations between subsets of explanatory variables and the output. We consider the complex socio-economic problem of the analysis of the utilization of the Internet in Italy, using real data gathered by the Italian National Institute of Statistics.

Keywords: Classification algorithms, Rule learning, Socio-economic analyses, Data analytics, Digital divide.

1. Introduction

Probabilistic Discrete Choice Models (PDCM) have been extensively used for decades as a powerful method to interpret the behavior of heterogeneous decision makers that face differentiated, discrete alternatives [32, 40]. Modern methods allow a rich and flexible specification of both the deterministic and stochastic component of the model, and the estimation, possibly recurring to simulation. However, given the high computational burden of these procedures and the large number of available explanatory variables, an initial extensive exploratory data analyses is necessary. In this work we discuss how a data classification technique can be used in this first phase of the specification of PDCM.

Classification is a fundamental task in the field of data mining, and many approaches to solve this problem have been proposed, based on different paradigms and data models.

*Corresponding author
Email address: bruni@dis.uniroma1.it (Renato Bruni)

Established ones include: Neural Networks, Support Vector Machines, k-Nearest Neighbors, Bayesian approaches, Decision Trees, Logistic regression, Boolean approaches (see for references [29, 31]). One effective Boolean approach is the *Logical Analysis of Data* (LAD) (see, e.g., [20, 6, 7, 8]), which is based on Boolean Logic and on Discrete Optimization. LAD methodology is closely related to decision trees [37] and nearest neighbor [19] methods, and constitutes an extension of those two approaches, as shown in [8]. There are also affinities with DNF learning in Computational Learning Theory, see, e.g., [10] which captures certain aspects of LAD. Other connections exist with the empirical machine learning approaches based on production or implication rules, for instance those based on Rough Set theory [35]. The joint use of many patterns has similarities with the usage of an ensemble of classifiers, as it is done in boosting [27] and bagging [11] techniques.

We consider data organized into records. Each record is a different observation of the phenomenon, and it is composed of *fields* containing the observed *values*. Each field has its *domain*, that is the set of its possible values. A record may also have a *class label*. In this case, the class is also called the *output*, while the other fields are also called *explanatory variables*. To apply LAD approach, all values must be converted into binary form by means of a discretization process called *binarization*. The domain of each field is partitioned in a finite number of subdomains that are encoded using *binary attributes*. Since the number of obtained binary attributes is often very large, a selection step is performed. After this, the selected binary attributes are used to build the patterns. A pattern is a conjunction of binary attributes, also called *conditions*, characterizing one class. Finally, each unlabeled record is classified on the basis of the patterns covering that record. Patterns can be seen as an interpretation of the phenomenon under analysis (see, e.g., [21]). Therefore, this procedure can perform rules extraction tasks, which, in the study of PDCM, may be even more important than the classification itself. Indeed, in socio-economic studies, the main goal is often the comprehension of people's behavior and its determinants. To this aim, different theories and hypotheses suggested by the human analysts are tested against data. On the contrary, we propose here to start the interpretation process extracting rules from the data by means of pattern generation techniques based on LAD.

When dealing with the probabilistic behavior of economic agents, a large number of explanatory variables is available. Consequently, the number of patterns generated can be extremely large, and unfortunately most of them may have scarce practical meaning. For example, they may cover only a few records, or they may differ only in the selection within sets of highly associated explanatory variables, hence, the subsets of records covered by different patterns largely overlap.

We present here criteria to identify a reduced set of practically meaningful rules within the large set of all the patterns, along with ordering and filtering techniques for their practical implementation. These techniques are computationally viable and can also produce a set of rules which are internally *orthogonal*, i.e., the coverages of every pair of rules have empty intersection. Patterns are generated by using a version of the LAD methodology developed to deal with very large datasets. It is adapted from that proposed in [12] and designed to keep the computational burden under control.

Therefore, the main contribution of this work is a computationally viable methodology to obtain an internally consistent, non-redundant and statistically accurate set of practically meaningful explanatory rules from a set of available data in a probabilistic

discrete choice setting. Several works in the literature have similarities either in the methods or in the goals, though none of them considers PDCM. For example, the generation of propositional logic formulas to provide a classification is applied to biological problems in [3]. An ELECTRE-based method to identify the best decision rules generated in the training process of a generic classification algorithm is proposed in [34]. Genetic algorithms are used to construct logic trees that best represent empirical data in [33]. Techniques to obtain a certain degree of orthogonality in the sets of Boolean rules are described in [25, 38]. The automatic individuation of the most important variables and of their values or intervals that are critical for a classification using Support Vector Machines is in [14, 15]. The problem of the selection of features has been addressed also in [4, 13, 18, 30, 39, 41].

The paper is organized as follows. Section 2 describes the binarization and the generation of the patterns. Section 3 presents the criteria to identify the small set of practically meaningful rules. In particular, we describe ordering techniques devised to bring out patterns which are the best compromises between accuracy and coverage; one technique aims at providing a sufficient disjunction of the coverages, the other at the complete disjunction of the coverages. Section 4 reports the results of the described techniques in the analysis of the individual use of the Internet, by considering data provided by the Italian National Institute of Statistics (Istat) and describing socio-economic status and daily habits of more than 46,000 individuals chosen to represent the whole Italian population. This analysis is important for the design of effective policies fostering Internet usage in Italy, in order to meet the goals of the European Digital Agenda [23], a plan established by the European Union which sets goals for 2020 regarding many aspects of the digitalization in all Member States.

2. Binarization and Pattern Generation

The structure of the data records consists of a set of fields f_i, with $i = 1, \ldots, m$. A *record instance* r consists of a set of values v_i, one for each field. A record r is *labeled*, or *classified*, if it is assigned to an element of a set of possible classes C. In many cases, C has only two elements, denoted by $+$ and $-$, and we speak of *binary classification*. We will hereinafter consider this case. A positive record instance is denoted by r^+, a negative one by r^-. A *training set* S of labeled records is available, with S^+ the set of its positive records and S^- the set of its negative ones. These sets constitute our source of information in learning the classifier.

LAD methodology begins with binarization, which converts each (non-binary) field f_i into a set of binary *attributes* a_i^j, with $j = 1 \ldots n_i$. The total number of binary attributes is $n = \sum_{i=1}^{m} n_i$. Note that the term "attribute" is not used here as a synonym for "field". The values of a qualitative field f_i can simply be encoded by means of a suitable number of binary attributes a_i^j. For each numerical field f_i, on the contrary, we introduce n_i thresholds called *cut-points* $\alpha_i^1, \ldots, \alpha_i^{n_i} \in \mathbb{R}$, and the binarization of a value v_i is obtained by considering whether v_i lies above or below each α_i^j.

$$b_i^j = \begin{cases} 1 & \text{if } \; v_i \geq \alpha_i^j \\ 0 & \text{if } \; v_i < \alpha_i^j \end{cases}$$

The α_i^j are computed as the semi-sums of each couple of values v_i' and v_i'' belonging

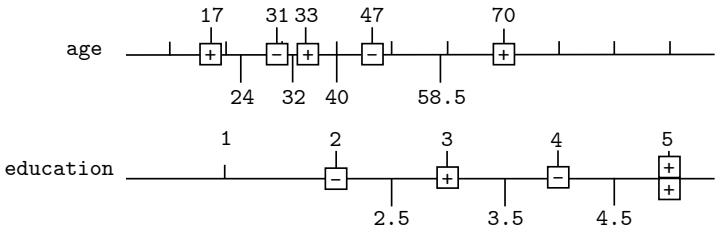
to training records from opposite classes and adjacent on f_i: $\alpha_i^j = (v_i' + v_i'')/2$. This identifies the borders between regions corresponding to opposite classes. When positive and negative records excessively overlap, and purely positive or negative regions become rare, this technique can still provide the borders between regions having opposite class predominance (see also [12]).

Example 1. *Consider a small training set of 5 records representing persons, with fields* `age`*, in years, and* `education`*, containing the highest degree obtained. The latter is an ordered categorical field, which can be seen as numerical.*

We use the following 5 levels: 1=elementary school or no title; 2=middle school; 3= high school; 4= bachelor's degree; 5=master's degree or Ph.D. The classification is "has mobile Internet connection" or not.

	record ID	age	education	mobile Internet ?
S^+	s_1^+	17	3	yes
	s_2^+	33	5	yes
	s_3^+	70	5	yes
S^-	s_1^-	31	2	no
	s_2^-	47	4	no

To visualize the cut-points, we plot the records' values by using a framed + for the positive ones and a framed − for the negative ones.



The cut-points obtainable from this set are: $\alpha^1_{\texttt{age}}$=24; $\alpha^2_{\texttt{age}}$=32; $\alpha^3_{\texttt{age}}$=40; $\alpha^4_{\texttt{age}}$=58.5; $\alpha^1_{\texttt{education}}$=2.5; $\alpha^2_{\texttt{education}}$=3.5; $\alpha^3_{\texttt{education}}$=4.5.
The corresponding binary attributes are:
$a^1_{\texttt{age}}$ *meaning* age $\geq$ 24; $a^2_{\texttt{age}}$ *meaning* age $\geq$ 32; $a^3_{\texttt{age}}$ *meaning* age $\geq$ 40; $a^4_{\texttt{age}}$ *meaning* age $\geq$ 58.5; $a^1_{\texttt{education}}$ *meaning* has high school; $a^2_{\texttt{education}}$ *meaning* has bachelor's; $a^3_{\texttt{education}}$ *meaning* has master's or Ph.D.

A set of binary attributes $\{a_i^j\}$ used to binarize a dataset S is called *support set* U. We are interested in selecting a small (and meaningful) support set. This selection is necessary for reducing the computational complexity of the remaining part of any LAD-based procedure, which may otherwise become impracticable. This combinatorial optimization problem is modeled by using a binary decision variable x_i^j for each a_i^j, such that

$$x_i^j = \begin{cases} 1 & \text{if } a_i^j \text{ is retained in the support set;} \\ 0 & \text{if } a_i^j \text{ is excluded from the support set.} \end{cases}$$

In classical LAD methodology, the problem is formulated as an unweighted set covering problem (see, e.g., [6]). On the other hand, [12] proposes a technique to evaluate the *quality* q_i^j of each a_i^j, based on its power of separation. The q_i^j are computed so that the total quality of a set of binary attributes should correspond to the sum of their individual quality values. One can evaluate the computational burden added to the pattern generation by retaining each single attribute a_i^j, and call it its *size* σ_i^j. When no specific evaluations can be done, all sizes could be set at 1. Thus, by setting a maximum affordable computational burden d (for instance on the basis of the available hardware, time, etc.) the support set selection problem can be modeled as *binary knapsack*:

$$\begin{cases} \max \sum_{i=1}^{m} \sum_{j=1}^{n_i} \; q_i^j \; x_i^j \\ \text{s.t.} \; \sum_{i=1}^{m} \sum_{j=1}^{n_i} \; \sigma_i^j \; x_i^j \leq d \\ x_i^j \in \{0,1\}. \end{cases} \tag{1}$$

In our case, all $\sigma_i^j = 1$, and this model can be solved by simply sorting the q_i^j values and taking the best d of them. The selection is actually performed twice, for positive and negative attributes, to find the set U^+ of the selected positive attributes and the set U^- of the selected negative ones.

The selected support set $U = U^+ \cup U^-$ is then used to create patterns. A *pattern* P is a conjunction ($\wedge$) of literals, also called *conditions*, that characterizes one class. We denote a positive pattern by P^+ and a negative one by P^-; when the class is not specified we simply use P. Literals are binary attributes $a_i^j \in U$ or negated binary attributes $\neg a_i^j$. Given a binarized record r_b, i.e., the set of binary values $\{b_i^j\}$ for each $a_i^j \in U$, each literal of P receives the value $b_i^j \in \{0,1\}$ for literal a_i^j and $(1 - b_i^j) \in \{0,1\}$ for literal $\neg a_i^j$. We have that $P = 1$ if all literals of P are 1, $P = 0$ otherwise. We say that a pattern P *covers* a record r, and that r *activates* P, if the set of values $r_b = \{b_i^j\}$ makes $P = 1$. We write $P(r)$ to denote the value of pattern P applied to record r:

$$P(r) = \begin{cases} 1 & \text{if } \; P \; \text{ covers } \; r; \\ 0 & \text{if } \; P \; \text{ does not cover } r. \end{cases}$$

A *positive* pattern P^+ is defined as a pattern covering at least c^+ positive records but no more than e^+ negative ones. A *negative* pattern P^- is defined as a pattern covering at least c^- negative records but no more than e^- positive ones. We call the pair of values (c^+, e^+) the *requirements* for being a positive pattern; conversely, (c^-, e^-) are the *requirements* for being a negative pattern. Values are such that the minimum correct coverage (c^+ or c^-) is always larger than the corresponding maximum erroneous coverage (e^+ or e^-). Patterns with $e^+ = 0$ or $e^- = 0$, namely, patterns not covering any record of the opposite class, are called *pure*, while patterns with $e^+ > 0$ or $e^- > 0$ are called *fuzzy*. Several works in the stream of research devoted to LAD use only pure patterns.

Finally, to perform the classification, weights w_h are assigned to all patterns, with $w_h \geq 0$ for positive patterns and $w_h \leq 0$ for negative ones. Such weights represent a measure of the positive or negative valence of each pattern. Several criteria to determine the w_h exist (e.g., [7, 12]). Now, an unlabeled record r is classified on the basis of the activated patterns, by computing the following weighted sum, called *discriminant* $\Delta(r) = \sum_h w_h P_h(r)$, and by selecting a threshold δ: r is predicted to be positive if $\Delta(r) > \delta$, and negative if $\Delta(r) \leq \delta$.

Example 2. *By continuing Example 1, if we set* $c^+ = c^- = 1$ *and* $e^+ = e^- = 0$*, a positive pattern is for instance:* $P_1^+ = a^3_{\texttt{education}}$ *, which means* "one has mobile Internet if education is master's or Ph.D. *Pattern* P_1^+ *has 1 condition; it covers 2 positive records* $(s_2^+,\ s_3^+)$ *and no negative ones. Another positive pattern is:* $P_2^+ = \neg a^3_{\texttt{age}} \wedge a^1_{\texttt{education}}$ *, which means* "one has mobile Internet if age is ≤ 40 and education is at least high school. *Pattern* P_2^+ *has 2 conditions; it covers 2 positive records* $(s_1^+,\ s_2^+)$ *and no negative ones. A negative pattern is for instance:* $P_1^- = a^1_{\texttt{age}} \wedge \neg a^3_{\texttt{education}}$ *, which means* "one has NO mobile Internet if age is ≥ 24 and education is no more than bachelor's. *This pattern has 2 conditions; it covers all the 2 negative records* $(s_1^-,\ s_2^-)$ *and no positive ones. All these patterns are pure. Of course, even for this simple example there exist many other patterns not listed here.*

Patterns can be generated by using combinatorial enumeration techniques based on two types of procedures: bottom-up or top-down. The bottom-up generation of a positive pattern proceeds by conjoining one by one single conditions until obtaining a formula that respects the requirements for being a positive pattern. We generate bottom-up patterns by using literals in greedy order, i.e., by decreasing values of q_i^j, and avoiding specializations of conjunctions that already are patterns. Though in principle all such combinations of literals could be generated, the enumeration can be guided by setting requirements on the coverages of the patterns, their length, etc. The computational burden is controlled by setting a not excessive value of d in (1).

3. Identifying the most interesting Patterns to Characterize Probabilistic Discrete Choice Behavior

All the patterns generated by the procedure give their contribution for the classification, and, roughly speaking, a large set of patterns allows better accuracy. However, if we aim at finding explanations of socio-economic phenomena, the interpretation of a large set of patterns may be problematic. In this case, the set of all the available labeled records is used as training set S, and there is no need of a classification step after the generation of the patterns. Instead, we want to set up an algorithmic procedure which should identify the most practically meaningful patterns within the large set $\mathcal{P}$ of all the patterns.

Given a generic pattern P_h, we define the following sets and values to describe its features. The set of its conditions is $Lit(P_h)$. The number of such conditions is $l(P_h) = |Lit(P_h)|$. In the space defined by the d binary attributes of U, the Boolean hypercube $\mathbb{B}^d$ is the set of the 2^d points having as coordinates all the possible binary strings of length d, that is, all the possible binarized records. A $(d-l)$-dimensional *subcube* consists of the 2^{d-l} points of $\mathbb{B}^d$ for which $l < d$ coordinates are fixed to 0 or 1. A positive pattern P_h^+ is a particular $(d - l(P_h^+))$-dimensional subcube such that the cardinalities of its intersections with $S^+ \subset \mathbb{B}^d$ and $S^- \subset \mathbb{B}^d$ satisfy the requirement (c^+, e^+) for being a positive pattern. A specular situation holds for a negative pattern. The size $s(P_h^+)$ of P_h^+ is the number of points $2^{d-l(P_h^+)}$ of the corresponding subcube. The correct coverage $Cov(P_h^+)$ of P_h^+ is the set of the records of S^+ covered by P_h^+. The number of hits is $c(P_h^+) = |Cov(P_h^+)|$. Similarly, its erroneous coverage $Err(P_h^+)$ is the set of the records of S^- covered by P_h^+ . The number of errors is $e(P_h^+) = |Err(P_h^+)|$. Moreover, we are considering data obtained from a sample survey. Hence, each record r does not

correspond to only one individual, but to a number μ_r of individuals: the multiplicity of r in the target population of the survey, also called universe of the survey. Such multiplicities are not the same for the different records. If we bring the values *back to the universe*, by counting each record for its multiplicity, the above defined values become: the number of hits reported to the universe $c^u(P_h^+) = \sum_{r \in Cov(P_h^+)} \mu_r$, and the number of errors reported to the universe $e^u(P_h^+) = \sum_{r \in Err(P_h^+)} \mu_r$. Specular definitions hold for a negative pattern.

Pattern features have been studied in several works belonging to the stream of research devoted to LAD (e.g., [28, 2, 5]). Some preference criteria have been described in [28]. For instance, simplicity preference is defined as follows: a pattern P_h is simplicity-wise preferred to a pattern P_k if and only if $Lit(P_h) \subseteq Lit(P_k)$. Selectivity preference is defined as follows: a pattern P_h is selectivity-wise preferred to a pattern P_k if and only if $s(P_h) \subseteq s(P_k)$. Evidential preference is defined as follows: a pattern P_h is evidentially preferred to a pattern P_k if and only if $Cov(P_k) \subseteq Cov(P_h)$. However, the above criteria are not suitable for evaluating the ability of a pattern in identifying an homogeneous set of individuals and to estimate the associated probabilities of the output. In these cases, we suppose that each labeled record in the sample S reports the outcome of a discrete choice operated by an observed decision maker, and the set of factors that have affected that choice. The first is called the output, while the seconds are called explanatory variables. Discrete choice methods [32, 40] build models of the decision making process that result in the estimation of the probability of each choice (the output) given the values of m explanatory variables. The probabilistic nature of these models reflects the heterogeneity of the decision makers and the limits of the explanatory variables in characterizing the choice. Theoretically, with a sufficiently large sample, we could estimate the probabilities $Pr(+|r)$ and $Pr(-|r)$ of the output by computing the frequencies of the output classes for each combination r of all the explanatory variables. In practice, the size of the available sample usually does not allow this granularity, and, even in more sophisticated estimation procedures such as logistic regressions, the number of explanatory variables has to be reduced in order to avoid over-fitting and multicollinearity.

To define a subset (i.e., a *category* of individuals), we fix a tuple of values for $k < m$ of the m explanatory variables, and we denote it by $(v_{i_1}, \ldots, v_{i_k})$. In general, each explanatory variable has a different importance in determining the behavior. If the chosen k variables gather enough importance, the frequencies of the output classes in the subset defined by $(v_{i_1}, \ldots, v_{i_k})$ would approximate the respective probabilities. Since each pattern corresponds to the binarization of a tuple $(v_{i_1}, \ldots, v_{i_k})$, it defines a specific category of individuals. Given, w.l.o.g., a positive pattern P_h^+, the individuals in $Cov(P_h^+)$ are those which, in that category of individuals, behave so as to have positive output, while those in $Err(P_h^+)$ are those which, in the same category of individuals, behave oppositely. Practically meaningful patterns should posses the following properties:

- generality, i.e., they have large correct coverage (large value of c);
- accuracy, i.e., they have small erroneous coverage (small value of e);
- simplicity, i.e., they require few conditions (small value of l).

Thus, in choosing the patterns, we have to deal with multiple criteria. Given the set of the patterns of one class, we define the *efficient* patterns. We discuss the case of positive

patterns; the discussion for negative ones is straightforward.

Definition 3. *Given a dataset S, a generic record $r \in S$, and the set $\mathcal{P}^+(r)$ of the positive patterns having r in their correct coverages, a pattern $P_h \in \mathcal{P}^+(r)$ is efficient if it does not exist another pattern $P_k \in \mathcal{P}^+(r)$ such that $c(P_k) \geq c(P_h)$, $e(P_k) \leq e(P_h)$, $l(P_k) \leq l(P_h)$, and at least one inequality holds strictly.*

In other words, an efficient pattern is non-dominated among those in $\mathcal{P}^+(r)$. The set of all efficient patterns is the *efficient frontier* $\mathcal{P}^{*+}(r) \subseteq \mathcal{P}^+(r)$ of the above set $\mathcal{P}^+(r)$.

Remark 4. *Given a dataset S, a generic record $r \in S$, and the corresponding set of efficient patterns $\mathcal{P}^{*+}(r)$, then there is a trade off between the accuracy (small value of e) of a generic pattern on one side, and its generality (large value of c) and simplicity (small value of l) on the other side.*

A simple motivation is obtained by using Boolean arguments. Consider a generic positive pattern $P^+ \in \mathcal{P}^{*+}(r)$ composed of the conjunction of l conditions written using a support set U with cardinality d, thus defined on the Boolean hypercube $\mathbb{B}^d$. Since P^+ represents a $(d - l(P^+))$-dimensional subcube $F \subseteq \mathbb{B}^d$ with size $s(P^+) = 2^{d-l}$, its size $s(P^+)$ increases when decreasing l, and vice versa. The records of S are distributed along the vertices of $\mathbb{B}^d$. In particular, S^+ and S^- tend to be scattered along the vertices of $\mathbb{B}^d$ and to constitute sets that are hardly coincident with subcubes, even though the d binary attributes are selected in (1) by pursuing the best separation of S^+ and S^-. Denote by F^+ the set of the positive records lying on the vertices of F, and by F^- the set of the negative records lying on the vertices of F. The cardinalities of both F^+ and F^- cannot decrease when increasing $s(P^+)$, and actually they always increase, except in the case when the removed condition was irrelevant among those in P^+. Now, an increase in the cardinality of F^+ increases the generality, while an increase in the cardinality of F^- decreases the accuracy. With respect to the requirements (c^+, e^+), we observe that, when the requirement c^+ is increased (we pursue generality), patterns need larger s, hence they tend to have smaller l and larger e, so the requirement e^+ should also be increased. On the contrary, if the requirement e^+ is decreased (we pursue accuracy), the value of l tends to increase and the requirement c^+ should also be decreased, since it must be small enough with respect to s, that also decreases.

As a consequence, without an explicit identification the above efficient frontiers, we should consider both a measure of generality and a measure of accuracy to evaluate a generic pattern P_h, and search for patterns providing good compromises between the two values. We now introduce a slightly different measure of accuracy: the *error percentage*, defined as follows:

$$\varepsilon(P_h^+) = \frac{100\ e(P_h^+)}{c(P_h^+) + e(P_h^+)}\%.$$

Note that $\varepsilon(P_h^+)$ is actually the probability that an individual belonging to the category defined by P_h^+ has negative output. Now, we can combine $c(P_h)$ and $\varepsilon(P_h)$, in several ways. In what follows, we will simply present our procedures by using $c(P_h)/\varepsilon(P_h)$. However, it should be understood that different combinations of coverage and accuracy can be considered, depending on the preferred area in the mentioned trade-off between generality and accuracy, and that we can also use $c^u(P)$ and $e^u(P)$ to take into account the underlying universe.

We define a first preference criterion, called *evidential-probability*: a pattern P_h is preferred to a pattern P_k with evidential-probability if and only if $c(P_h)/\varepsilon(P_h) > c(P_k)/\varepsilon(P_k)$. This practically means that P_h defines a large category of individuals that behave quite uniformly w.r.t. the phenomenon under analysis; therefore P_h is meaningful. To apply this criterion, we can sort, separately for each class, all patterns by decreasing values of $c(P)/\varepsilon(P)$, and take those in the initial positions of the ordering of each class.

However, to obtain meaningful patterns, we also have to deal with another aspect. In complex phenomena, there are variables that are strongly associated (for example, level of education and professional position). Hence, the same aspect of the phenomenon can be explained by patterns alternatively using these strongly associated variables, since such patterns would cover highly overlapping sets of records. In general, we would like to avoid this kind of redundancy, even if the presence of such alternative explanations may be interesting when they derive from the joint effect of different groups of variables in two different patterns, rather than from the direct association of two single variables.

3.1. Ordering by using Incremental Coverage

A crucial weakness of the simple criterion described above is that the coverages of different patterns may overlap. A pattern P_h may have a high value for $c(P_h)/\varepsilon(P_h)$; however, if P_h covers almost the same records of another pattern P_k preceding P_h in the above defined ordering, it becomes far less interesting. To overcome this weakness, we define the *incremental* coverage $Cov^I(P_h)$. Given an ordering $\mathcal{O}$ of the patterns of one class, positive w.l.o.g., and expressing that P_k precedes P_h in $\mathcal{O}$ by $P_k \prec P_h$, the incremental coverage $Cov^I(P_h)$ is the set of the records of S^+ covered by P_h and not in $Cov(P_k)$ for all k such that $P_k \prec P_h$. For the first pattern in the ordering, the incremental coverage coincides with the coverage. The incremental number of hits is $c^I(P_h) = |Cov^I(P_h)|$. Given $\mathcal{O}$, we define similarly the incremental erroneous coverage $Err^I(P_h)$, the incremental number of errors $e^I(P_h)$ and the incremental error percentage $\varepsilon^I(P_h) = 100\ e^I(P_h)/(c^I(P_h) + e^I(P_h))\%$. One way to compute these values is to keep a matrix M of the incidences between patterns and records: each element m_{hk} is: 1 if $r_k \in Cov(P_h)$; -1 if $r_k \in Err(P_h)$; 0 otherwise. We can now define a second preference criterion, called *disjoint evidential-probability*: a pattern P_h is preferred to a pattern P_k with disjoint evidential-probability if and only if $c^I(P_h)/\varepsilon^I(P_h) > c^I(P_k)/\varepsilon^I(P_k)$. This means in practice that P_h defines a large category of individuals that behave uniformly, and that are disjoint enough (even if not completely) from the categories defined by the patterns preceding P_h in $\mathcal{O}$. To identify meaningful pattern according to this criterion, we use the following Procedure 1 for each class separately.

Procedure 1: find sufficiently disjoint meaningful patterns

Input The set of patterns $\mathcal{P}$ of one class;
the pattern-records incidence matrix M.

Output An ordering $\mathcal{O}^I$ of $\mathcal{P}$ by disjoint evidential-probability.

1. **Initialization:** Order the patterns by decreasing values of $c(P)/\varepsilon(P)$, obtaining the initial ordering $\mathcal{O}_0$. Let $\pi := 1$.
2. **Iteration t:**

(a) For the patterns in the positions from $\pi+1$ to the last one of $\mathcal{O}_t$, compute c_t^I and ε_t^I corresponding to the current ordering $\mathcal{O}_t$ using matrix M.
(b) From position $\pi+1$ to the last one of $\mathcal{O}_t$, sort the patterns by decreasing values of c_t^I/ε_t^I, obtaining a new ordering $\mathcal{O}_{t+1}$.
(c) Compare $\mathcal{O}_{t+1}$ to $\mathcal{O}_t$ and let π be the last position until which $\mathcal{O}_t$ and $\mathcal{O}_{t+1}$ coincide.
(d) Check if the pattern in position $\pi+1$ already assumed that position in a previous iterations and subsequently left it. If YES, then fix it in position $\pi+1$ and let $\pi := \pi+1$.
(e) If $\mathcal{O}_{t+1} \neq \mathcal{O}_t$, then let $t := t+1$ and repeat the **Iteration**.
Else, let $\mathcal{O}^I := \mathcal{O}_{t+1}$ and **exit**.

The above Procedure 1 terminates, because the ordering will always converge. Indeed, consider the sequences $\mathcal{O}_t$ and $\mathcal{O}_{t+1}$ obtained at two generic consecutive iterations. Sequences $\mathcal{O}_t$ and $\mathcal{O}_{t+1}$ coincide at least in the first position, because the first pattern in the ordering $\mathcal{O}_0$ will always maintain its position when switching to incremental coverages. Now, let π be the last position of the ordering in which $\mathcal{O}_t$ and $\mathcal{O}_{t+1}$ coincide. Each time that the incremental coverages are recomputed, they cannot change for all the positions that go from the first till the π-th. Hence, π cannot decrease. There is a slight chance that it could remain the same, when two (or more) patterns cyclically swap their positions from one ordering to the next. To avoid this and similar situations, we perform step (d): whenever the pattern in position $(\pi+1)$ cyclically assumes different positions and then returns to position $(\pi+1)$, we fix it to that position and we proceed. Therefore, π is forced to increase, at least after a certain number of iterations in which it remains constant. Consequently, the ordering will converge to a final one called $\mathcal{O}^I$. The procedure is computationally viable, since it essentially recomputes incremental coverages and sorts values.

Patterns in the first positions of $\mathcal{O}^I$ represent good compromises between coverage, accuracy and disjointness of the coverages. The number of patterns to bring out can be chosen, for instance by taking them until they cover at least a certain portion of the dataset, or until their value for c_t^I/ε_t^I is above a certain threshold (a pattern with $c^I < e^I$ would add more errors than correct cases, so there would be reasons to reject it). Experimentally, we pass from a set of several thousands of patterns to a few hundreds that are able to cover almost the whole dataset.

3.2. Generation of Orthogonal Patterns

The above described Procedure 1 heuristically aims at obtaining patterns that correspond to disjoint categories of individuals. However, it cannot provide a bound on the amount of disjunction between such sets. We say that two patterns are *orthogonal* if their coverages are disjoint, i.e., they have empty intersection. The following Procedure 2 is a new ordering procedure able to provide the pairwise orthogonality of the generated patterns.

Procedure 2: find orthogonal meaningful patterns

Input The set of patterns $\mathcal{P}$ of one class;
the pattern-records incidence matrix M.

Output A set of internally orthogonal patterns $\mathcal{P}^O$ build over $\mathcal{P}$.

1. **Initialization:** Order $\mathcal{P}$ by decreasing values of $c(P)/\varepsilon(P)$, select the first pattern and call it P_1. That is also the first orthogonal pattern P_1^O.
2. **Iteration t:**
   (a) Drop from M all columns (records) in $Cov(P_t^O)$. If the remaining columns are less than a threshold ν, **exit**.
   (b) For each row of M, update the number of hits and of errors to determine a new set of patterns $\mathcal{P}_t$.
   (c) In $\mathcal{P}_t$ select the pattern which maximizes $c(P)/\varepsilon(P)$ and call it P_{t+1}.
   (d) Compute the logic negation of the previous orthogonal pattern P_t^O and generate the next orthogonal pattern $P_{t+1}^O = P_{t+1} \wedge \neg P_t^O$; let $t := t+1$ and repeat the **Iteration**.

The above Procedure 2 will generate patterns until they cover at least a certain portion of the dataset (for example, almost all). Clearly, the length of the orthogonal patterns $l(P_t^O)$ rapidly increases. However, experimentally, we only need a very small number of patterns to cover almost the entire dataset. Moreover, even the sequence of patterns P_t (those without the negation of the previous ones) will have an interesting practical significance, with the advantage of being more easily readable. Furthermore, the conditions defined by each P_t, for $t = 1, \ldots, \tau$, when τ is not greater than 4 or 5, can be used to produce a partition of the whole dataset by considering all the 2^τ combinations of their assertions/negations. This constitutes a partition of the individuals in categories that are relevant for the phenomenon under analysis. The procedure is computationally viable, since it essentially updates number of hits and errors, finds the maximum of a vector and writes logical negations. The above Procedures 1 and 2 were selected as the most representative of the many others developed and tested.

4. The Analysis of the Diffusion of the Internet

We apply the methodology presented above to analyze the diffusion of the Internet among Italian population. The Italian lag in the household Internet demand, and in particular in broadband services demand, has been widely analyzed by international sources [24], proposing several explanations. On the demand side, a major barrier is the Italian population structure, characterized by high elderly-to-youth ratios (similar only to those of Germany, in the whole European Union). Moreover, education levels (and consequent skills and interests) are sometimes lower than in other European comparable countries (i.e., Germany, France, Spain, United Kingdom), especially for the elderly people. Finally, Italy has also a low level of labor market participation, due to the number of retired people, housewives and NEET (Not in Education, Employment, or Training), high level of unemployment, and a high percentage of unskilled or blue collar workers. In [22], contingency matrices show that the four variables Education, Age, Working status, and Professional level strongly influence the individual use of the Internet. However, since *digital divide* is a complex and multifaceted issue, we expect that, in different homogeneous segments of the population, different factors are relevant, or that their effects have different intensity. The building of causal statistical models of Internet usage is a prerequisite for the design of effective policies aimed at fostering Internet connection

demand in Italy, in order to meet the goals of the European Digital Agenda [23], which sets objectives for the growth of the European Union digitalization by 2020.

In the literature, causal analyses of these phenomena usually adopt Logit and Probit models [1] of individual discrete choice. Examples can be found in [17, 16, 26, 42]. A first step in developing a casual model for Italy is performed in [22] using a logistic regression. While the model appears satisfactory in terms of fitting and interpretation of the phenomenon, further exploration of data is needed to understand whether:

(*i*) Different and simpler explanations are possible for different segments of the population. A clusterization of the cases may improve the explanation of the endogenous variables in each cluster; for example, in some segments of the population, the Internet is not used because of lack of interest or skills, while in others budget constraints prevent the adoption by interested people.

(*ii*) Alternative classification rules produce equivalent aggregated characterization of the data; for example the joint effect of working status and professional level may be used instead of a qualitative variable on the economic satisfaction of the individual.

(*iii*) Any additional variable not included in the original logistic model is discovered to play a role in the causal explanation of the output.

(*iv*) Any additional variable without strict causal effect can nevertheless improve classification and explanation. This may occur when an observed behavioral variable is associated to the endogenous variable through the effect of latent factors affecting both, as in seemingly unrelated regression models. Examples are the association between Internet use and the use of credit cards or the propensity to tourism or cultural consumption.

4.1. The Dataset

A survey provided by the Italian National Institute of Statistics (Istat) collects every year a large variety of data about socio-economic status and daily habits of more than 46,000 individuals belonging to about 20,000 households, chosen to represent the whole Italian population. The survey can be used to estimate a number of statistical tables, significant at regional level. However, the use of microdata (i.e., the set of all the answers provided by each respondent) allows a far greater flexibility in exploring individual behavior. The aspects investigated in the survey include: Socio-demographic and professional characterization of the individual; Education; Household structure and composition; Dwelling features, issues and surrounding area; Nutrition and lifestyle; Drugs consumption and medical conditions; ICT related behavior of individuals and households (as required by the harmonized Eurostat surveys that support the European Digital Agenda goal assessment and policy development); Daily commuting; Cultural consumption, spare time and social participation; Household goods ownership; Environment and recycling; Security; Satisfaction for different aspects of life.

We extracted a dataset composed of 39 explanatory variables and 34,455 records from the 2012 edition of survey, by considering all the variables that may represent socio-economic and cultural determinants related to Internet use, and the output class, which is 1 if the individual is an Internet user (at least once a week) and 0 otherwise. Note that the selected subset of possible explanatory variables is much larger than the set of variables typically used in a discrete choice model. In particular, in [22] the model of the phenomenon takes into account 9 variables. Indeed, we are also interested in verifying if our approach can help in identifying the best subset of those variables based on accuracy, generality and parsimony criteria.

4.2. Empirical Results

We perform experiments using the described version of LAD, called SLAD (Statistical and Logical Analysis of Data) [12], with and without Procedures 1 and 2, to evaluate their advantages. The binary attributes obtained from the original variables are 70. We also apply to the same dataset a C4.5 decision tree algorithm [37]. Results are summarized in Table 1. For each test, we report input parameters (see Section 3) and the following output performance indicator for the patterns in each class:

- the total number of patterns selected, the most relevant indicator of problems of redundancy and fragmentation;
- the number of patterns that provide a positive incremental coverage;
- the maximum cardinality of the coverage of a single pattern, that indicates whether we obtain at least one pattern with large generality;
- the average cardinality of the incremental coverage, computed on the whole set of selected patterns, that indicates the capacity of the algorithm in limiting the redundancy of the generated patterns;
- the maximum number of conditions appearing in a single pattern, useful to assess the generality and the readability of the selected patterns;
- the percentage of records in the sample covered by the whole set of selected patterns, in order to provide evidence of the suitability of the algorithm;
- the average number of patterns that correctly cover a single record, a useful indicator of the redundancy of the selected patterns.

In Test 1 we generate patterns as described in Section 2, and we allow only pure patterns. This means that we are actually ignoring the probabilistic nature of the phenomenon, looking for subsets of the sample characterized by the same value of the output class. We obtain a huge number of patterns, with a small maximum and average incremental coverage, as a consequence of Remark 4. Most importantly, only 7,641 out of the 18,077 (about 42%) positive records in the data set are identified by at least one pattern, while 14,762 out of 16,378 (about 90%) negative records are identified. This suggests that it is much easier to find categories of people that do not use the Internet at all. The selected patterns, especially the negative ones, have large overlaps: each positive [negative] identified record is covered by 21 [respect. 341] patterns, on average. This may mostly be due to correlation between different explanatory variables. The results of this first test are clearly not satisfactory: the large number of patterns, even restricting to those that provide an incremental coverage, makes it difficult to gain insights on the phenomenon. Note that, when using the techniques described in Section 2 (tests 1 and 2), the incremental coverage of each pattern is computed ex-post, after ordering the patterns by decreasing total coverage. When using Procedure 1 and 2 of Section 3 (tests 3 and 4), patterns with no incremental coverage are directly excluded by the algorithm.

To improve the above results, in Test 2 we change some parameters, still using only the pattern generation of Section 2. First, we allow patterns selecting a given maximum percentage of records of the opposite class. Secondly, we use asymmetric parameters,

		Test 1 pure SLAD	Test 2 fuzzy SLAD	Test 3 SLAD + Procedure 1	Test 4 SLAD + Procedure 2	Comparison with C 4.5 Decision Tree
Parameters	c^+	10	5	10	10	n.a.
	c^-	10	10	10	10	n.a.
	ε^+	0 %	20 %	20 %	40 %	n.a.
	ε^-	0 %	10 %	20 %	40 %	n.a.
Positive patt.	Total patterns selected	6,650	6,761	198	7	32
	With incremental coverage	1,314	420	198	7	32
	Avg. incremental coverage	1.1	2.6	87	2,057	618
	Max coverage	164	6,544	8,096	8,096	8,142
	Max number of conditions	10	10	6	2	11
	% records covered (+)	42 %	96 %	96 %	80 %	100%
	Avg. patterns covering each record	21	148	85	1	1
Negative patt.	Total patterns selected	72,682	10,856	243	7	27
	With incremental coverage	2,621	1,034	243	7	27
	Avg. incremental coverage	0.2	1.3	65	1,872	543
	Max coverage	1,673	4,175	8,144	7,236	3,945
	Max number of conditions	6	13	8	3	11
	% records covered (-)	90 %	86 %	96 %	80 %	100%
	Avg. patterns covering each record	341	114	98	1	1
Running time of the whole test (in secs)		5,330	2,680	970	590	60

Table 1: Summary of the results in the different tests.

since negative records are more homogeneous. For example, by setting $c^+ = 5$, $\varepsilon^+ \leq 20\%$, $c^- = 10$, $\varepsilon^- \leq 10\%$, we obtain 6,761 positive and 10,856 negative patterns. Both the maximum and the average incremental coverages are higher than in Test 1, for positive and negative patterns. Moreover, there is a significant improvement in the percentage of positive records covered. However, the resulting set of patterns is still too large to provide useful insights of the phenomenon. These results highlight the need for procedures tailored on the specificities of the probabilistic discrete choice settings, and allow us to explore the trade-offs among our different goals:

- discrimination power between the two output classes (Internet users and non-users), however considering that probabilistic behavior implies that we deal with non-homogeneous subsets, and that the distributions of the output classes in each region of the space are not known in advance;
- relevance, i.e. the patterns should cover a large number of observations;
- simplicity, i.e. the patterns should consist in a small number of conditions;
- non-redundancy, i.e., the pattern should have limited (Procedure 1) or no overlap (Procedure 2) with each other.

Procedure 1. We perform Test 3 using the incremental coverage as ordering criteria (see Section 3.2). The algorithm selects 198 positive patterns and 243 negative patterns. This is an important improvement with respect to the thousands of patterns obtained in the first two tests. Moreover, in order to increase the readability of the results and therefore the understanding of the phenomenon, we can set some ex-post criteria and drastically reduce the number of patterns. If we select patterns with an incremental coverage of at least 200 records, and with a maximum incremental error of less than 45%, we obtain only 8 positive patterns, which alone correctly identify 84% of the Internet users in the dataset, and 5 negative patterns, which alone correctly identify 75% of the non-users. Their analysis is in Tables 2 and 3. However, these patterns still partially overlap. Every selected record is covered by about 3 different patterns.

Procedure 2. In Test 4, a pattern added in the step i of the procedure selects records that were not selected in the previous $i-1$ steps. Hence, each pattern is generated only if it is strictly needed to cover a predefined minimum number of additional records. Moreover, we can logically define the region of the space added at the step i as the conjunction of the conditions in pattern i with the logical negation of all the previous $i-1$ patterns. In principle, this allows us to identify the logical expressions of a partition of the set of all the selected records. In practice, the resulting expressions may become cumbersome. However, the logical disjunction of the selected patterns provides a readable expression for the set of all the selected records in the output class.

The results for positive pattens are in Tables 4. The total number of users covered by this set of positive patterns is 14,399 out of 18,077, that is 80% of the sample. In particular, the first pattern shows how 8.096 users are described by the conjunction of just two conditions: education level at least high school and high level of cultural consumption. The latter represents an index that summarizes the cultural habits of an individual in the last 12 months, such as attending concerts, sports events, visiting museums, going to cinema, reading books etc. Moreover, the percentage of non-users

Pattern		P_1^+	P_2^+	P_3^+	P_4^+	P_5^+	P_6^+	P_7^+	P_8^+	Total
Conditions	Credit card holder						Yes		Yes	
	Labour status			Working			Working			
	Marital status		Single		Single					
	Labour status of female head of HH					Working		Working		
	Economic Index [0;1]						≥ 0.47			
	Education	High School or university		High School or university				High School or university	High School or university	
	Age				≤ 34					
	Week working h			≥ 22						
	Cultural consumption	High	High			High				
Incremental coverage (records)		8,096	2,244	2,232	1,042	721	493	227	224	15,279
Incremental error percentage		14.45 %	23.78 %	27.56 %	36.35 %	39.72 %	37.99 %	42.24 %	44.42 %	23.27 %

Table 2: Positive patterns produced by Test 3.

Pattern		P_1^-	P_2^-	P_3^-	P_4^-	P_5^-	Total
Conditions	Credit card holder	No	No			No	
	Age		≥ 55	≥ 55	≥ 55		
	Education	Middle school or lower			Middle school or lower	Middle school or lower	
	Cultural consumption	Low		Low			
	Holidays			No		No	
	Marital status					Married	
Incremental coverage (records)		8,144	2,273	730	484	402	12,033
Incremental error percentage		15.64 %	27.61 %	30.67 %	35.89%	39.64 %	21.20 %

Table 3: Negative patterns produced by Test 3.

selected by this pattern is just 14.45 %. Interestingly, the second pattern covers under-35 users that are not covered by the previous pattern, i.e. that do not have a high level of education or cultural consumption. This pattern is nevertheless quite large, covering 3,759 users with a non-users incidence of 26.25 %.

By applying De Morgans Laws, it is possible to simplify the disjunction of the selected positive patterns, obtaining the following expression:

$$(\text{Age} \leq 34) \vee (\text{Income source} = \text{Salaried job}) \vee (\text{AC ownership} = \text{yes}) \vee (\text{Credit card} = \text{yes}) \vee (\text{Education} = \text{High School or University})$$

This is an important finding, showing that Internet use is influenced by the young age, the income source and the level of education. The algorithm has selected the last two variables as the most parsimonious proxies of the economic habits of the Italian households. Air conditioning is still considered a luxury good, and holding a credit card may be a proxy of both income and a positive attitude towards innovation.

The results for negative pattens are in Tables 5. In this case, 13,102 out of 16,378 non-users are covered by the selected patterns, again 80% of the sample. The largest (7,236 records) and most homogeneous (18% of users) pattern is made up of non-users that do not possess a credit card, have a lower degree of education and are currently married. Another 3,520 non-users have a low level of education and of cultural consumption, but do have a credit card or are currently married. Negative patterns involve a larger number of conditions overall (11 vs 7) and do not allow the level of simplification of the previous class. Anyway, considering just the first four patterns, which cover most of the cases in the partition (13,032 out of 13,102), we obtain this important description:

$$(\text{Age} \geq 55) \vee (\text{Edu.} = \text{Middle or lower} \wedge \text{Cultural cons.} = \text{low}) \vee (\text{Credit card} = \text{No} \wedge \text{Holidays} = \text{No}) \vee (\text{Edu.} = \text{Middle or lower} \wedge \text{Credit card} = \text{No} \wedge \text{Marital Status} = \text{Married})$$

We finally compare the above results to the output of a C4.5 decision tree. We note that the comparison is not straightforward, because the tree algorithm gives directly by construction a partition of the dataset, so 100% of the records are covered, and each record is covered by only one set of conditions: those obtained visiting the tree from the root to the leaf containing that record. On the other hand, the sets of conditions characterizing each leaf, that we will here call leaf patterns, do not have predetermined length and may be considerably more complicated than the patterns generated by Procedures 1 and 2. The decision tree finds 27 positive and 32 negative leaf patterns, using at most 11 binary splitting conditions based on 18 explanatory variables. By comparison, Procedure 2 uses 12 explanatory variables and the maximum number of conditions in the patterns is 3.

The following 8 explanatory variables are in common between the decision tree and the patterns: Age; Credit card holder; Cultural consumption; Education; Geographic area; Income source; Kinship with head of the HH; Labour status. One more variable is almost in common, since it appears in two specular versions: Number of income receivers and Number of persons without income. The 9 variables used in the decision tree and not in the patterns are: At least a 4-day holiday in the past year; Education of head of the HH; Professional condition; Sector of activity; Takes courses in informatics; Sex; Title for the house (ownership, rent, etc.); Type of house; Presence of fixed telephone. On the other hand, the 3 variables used in the patterns and not in the decision tree are:

Pattern		P_1^+	P_2^+	P_3^+	P_4^+	P_5^+	P_6^+	P_7^+	Total
Conditions	Credit card holder			Yes			Yes		
	Income source				Salaried job				
	Labour status			Working					
	A.C. ownership					Yes			
	Education	High School or University						High School or University	
	Age		<=34						
	Cultural consumption	High							
Incremental coverage (records)		8,096	3,759	1,678	533	149	87	97	14,399
Incremental error percentage		14.45 %	26.25 %	23.76 %	36.85 %	36.86 %	37.41 %	37.42 %	20.60 %

Table 4: Positive patterns produced by Test 4.

Pattern		P_1^-	P_2^-	P_3^-	P_4^-	P_5^-	P_6^-	P_7^-	Total
Conditions	Labour Status					Housewife	Housewife		20
	Number of income receivers							0-1	
	Geographic area					South or major islands			
	Credit card holder	No		No					
	Age				≥ 55				
	Education	Middle school or lower	Middle school or lower					Middle school or lower	
	Kinship with head of the HH					Spouse	Spouse		
	Marital status	Married							
	Cultural consumption		Low						
	Average cars per person							<= 0.585	
	Holidays			No					
Incremental coverage (records)		7,236	3,520	1,519	757	34	19	17	13,102
Incremental error percentage		18.45 %	26.00 %	36.26 %	35.74 %	32.00 %	32.14 %	39.29 %	24.25 %

Table 5: Negative patterns produced by Test 4.

Average number of cars per person; Marital status; Presence of A.C. We observe that some of the variables selected only in the decision tree are actually highly correlated (Professional condition; Sector of activity, Takes courses in informatics), while some of the variables selected only in the patterns constitute a very good and compact description of the lifestyle of the household (Average number of cars per person; Presence of A.C.).

The coverage of the largest leaves and patterns (*i.e.*, those covering more records) have comparable values, and in general the homogeneous categories of individuals obtained by the two approaches have similar sizes. The patterns generated by Procedure 2 differ from the leaves of the C4.5 tree, because of course there exist differences both in the algorithms and in the type of settings that the user can chose. However, patterns and leaves often represent similar concepts expressed using different variables, due to the high degree of correlation existing among the variables. For example, by comparing the composition of the largest leaves and patterns, we find that 73.2% of the records that are in the largest positive leaf are also contained in the largest positive pattern, while, on the other hand, 65% of the records in the largest negative leaf are also contained in the largest negative pattern. Thus, Procedure 2 was able to produce a description that can be considered functionally equivalent to that of the decision tree but whose format can be somehow more controlled.

Though it was not the main purpose of our study, because the behavior we analyze is inherently probabilistic, we also evaluate the accuracy obtainable by the sets of selected patterns (Tests 3 and 4) in classifying unseen data of the same nature using the LAD classification techniques described in Section 2, with pattern weights w_h based on coverage. The overall classification accuracy is defined as the percentage of cases in which the predicted class coincides with the observed class. We perform a cross validation using the dataset of the same Istat survey about socio-economic status and daily habits in the year 2013, which is composed of entirely different individuals, and we obtain an accuracy of 77.0% for the patterns of Test 3 and 75.5% for those of Test 4. Since the same sets of patterns, if used to classify the 2012 dataset, give an accuracy of 78.5% and 75.8% respectively, we conclude that our procedures are able to identify rather stable probabilistic phenomena whose incidence is similar in the training and test sets. When using the smaller set of patterns (Test 4), the number of records which are not covered by any of the selected patterns slightly increases. However, the considerable improvement in the intelligibility of the patterns compensates such a small decrease in the coverage.

Furthermore, to evaluate the intrinsic difficulty of the classification task over the same datasets, we repeat the same classification using different classifiers. This was done by means of scikit learn [36], that is a very good machine learning package currently included into scientific Python distributions. In our case, the best results have been obtained with Random Forest classifier, producing an accuracy of 77.3%. This means that, for these datasets, the two classes are inherently overlapping, and that the patterns selected by our procedures also possess an appreciable ability of generalization.

5. Conclusions

We have presented here a new and computationally viable approach to obtain patterns that can be practically meaningful for the analyses of Probabilistic Discrete Choice Behavior. In particular, we have developed procedures carefully designed to satisfy the

requirements specific to this class of problems. These procedures are aimed at the identification of patterns representing the best compromises between accuracy and coverage and at providing disjoint coverages. We report results for the important case of the analysis of the individual use of the Internet in Italy. Our procedures could identify surprisingly small sets of patterns that are able to describe this complex phenomenon. Although these automatic procedures do not provide a proper interpretative model, the selected patterns greatly support the identification of different categories of people that may need different actions to be encouraged to use the Internet. The described approach works only at the formal level and automatically; thus, it can be applied to problems arising also in very different contexts.

References

[1] Agresti A. An introduction to categorical data analysis. Wiley-Interscience, Hoboken, New Jersey, 2007.

[2] Alexe G, Alexen S, Bonates TO, Kogan A. Logical analysis of data: the vision of Peter L. Hammer. Annals of Mathematics and Artificial Intelligence 49(1), 265-312, 2007.

[3] Bertolazzi P, Felici G, Festa P. Logic based methods for SNPs tagging and reconstruction, Computers & Operations Research 37, 1419-1426, 2010.

[4] Bertolazzi P, Felici G, Festa P, Fiscon G, Weitschek E. Integer programming models for feature selection: New extensions and a randomized solution algorithm, European Journal of Operational Research 250(2), 389-399, 2016.

[5] Bonates T, Hammer PL, Kogan A. Maximum patterns in datasets. Discrete Applied Mathematics 156(6), 846-861, 2008.

[6] Boros E, Hammer PL, Ibaraki T, Kogan A. Logical Analysis of Numerical Data, Mathematical Programming 79, 163-190, 1997.

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

[8] Boros E, Crama Y, Hammer PL, Ibaraki T, Kogan A, Makino K. Logical Analysis of Data: Classification with Justification, Annals of Operations Research 188, 33-61, 2011.

[9] Boros E, Horiyama T, Ibaraki T, Makino K, Yagiura M. Finding Essential Attributes from Binary Data, Annals of Mathematics and Artificial Intelligence, 39(3), 223-257, 2003.

[10] Bshouty NH, Eiron N. Learning monotone DNF from a teacher that almost does not answer membership queries, J. Mach. Learning Res. 3(1), 49-57, 2003.

[11] Breiman L. Bagging predictors, Machine Learning 24(2), 123-140, 1996.

[12] Bruni R, Bianchi G. Effective Classification Using a Small Training Set Based on Discretization and Statistical Analysis, IEEE Transactions on Knowledge and Data Engineering 27(9), 2349-2361, 2015.

[13] Bruni R. Reformulation of the Support Set Selection Problem in the Logical Analysis of Data, Annals of Operations Research 150(1), 79-92, 2007.

[14] Carrizosa E, Martín-Barragán B, Romero Morales D. Binarized support vector machines, INFORMS Journal on Computing 22(1), 154-167, 2010.

[15] Carrizosa E, Romero Morales D. Supervised classification and mathematical optimization. Computers & Operations Research 40 (2013) 150165

[16] Cerno L, Pérez Amaral T. Demand for Internet Access and Use in Spain. In: Preissl B, Muller J (eds.). Governance of Communication Networks: Connecting Societies and Markets with IT. Physica-Verlag, Heidelberg, Germany 2006. ISBN 3790817457.

[17] Chaudhuri A, Flamm K, Horrigan J. An analysis of the determinants of Internet access. Telecommunications Policy, 29, 731-755, 2005.

[18] Chou CA, Bonates TO, Lee C, Chaovalitwongse WA. Multi-pattern generation framework for logical analysis of data, Annals of Operations Research 249, 329-349, 2017.

[19] Cover TM, Hart PE. Nearest neighbor pattern classification. IEEE Transactions on Information Theory 13 (1), 21-27, 1967.

[20] Crama Y, Hammer PL, Ibaraki T. Cause-effect Relationships and Partially Defined Boolean Functions, Annals of Operations Research 16, 299-326, 1988.

[21] Crama Y, Hammer PL. *Boolean Functions: Theory, Algorithms, and Applications*, Cambridge University Press, New York, 2011. ISBN 9780521847513.
[22] Dolente C, Galea J, Leporelli C. Next Generation Access and Digital Divide: Opposite Sides of the Same Coin? ITS Europe Conference 2010, Copenhagen, Denmark, 2010.
[23] European Commission. Digital Agenda for Europe, Publications Office of the European Union, Luxembourg, 2014. ISBN 978-92-79-41904-1
[24] Eurostat, Statistics on Internet access and use, http://ec.europa.eu/eurostat/statistics-explained/index.php/Internet_access_and_use_statistics_-_households_and_individuals. Last accessed: April 4, 2017.
[25] Felici G, Truemper K. A minsat approach for learning in logic domains. INFORMS Journal on computing 13(3), 1-17, 2001.
[26] Flamm K, Chaudhuri A. An analysis of the determinants of broadband access. Telecommunications Policy 31, 312-326, 2007.
[27] Freund Y. Boosting a weak learning algorithm by majority, Inform. Comput. 121(2), 256-285, 1995.
[28] Hammer PL, Kogan A, Simeone B, Szedmak S. Pareto-optimal patterns in logical analysis of data. Discrete Applied Mathematics 144(1), 79-102, 2004.
[29] Hastie T, Tibshirani R, Friedman J. *The Elements of Statistical Learning*, Springer-Verlag, New York, Berlin, Heidelberg, 2002.
[30] Janssens D, Brijs T, Vanhoof K, Wets G. Evaluating the performance of cost-based discretization versus entropy- and error-based discretization, Computers & Operations Research 33, 3107-3123, 2006.
[31] Klosgen W, Zytkow JM (eds). *Handbook of Data Mining and Knowledge Discovery*, Oxford University Press, 2002.
[32] Manski CF, McFadden D (eds). Structural analysis of descrete data with econometric applications, MIT Press, Cambridge, MA, 1981.
[33] Mak B, Blanning R, Ho S. Genetic algorithms in logic tree decision modeling, European Journal of Operational Research 170, 597-612, 2006.
[34] Mastrogiannis N, Boutsinas B, Giannikos I. A method for improving the accuracy of data mining classification algorithms, Computers & Operations Research 36, 2829-2839, 2009.
[35] Pawlak Z. Rough Sets: Theoretical Aspects of Reasoning about Data, Kluwer Academic Publishers, Norwell, MA, USA, 1992.
[36] F. Pedregosa et al., Scikit-learn: Machine Learning in Python, Journal of Machine Learning Research, 12, 2825-2830 (2011).
[37] Quinlan JR. C4.5: Programs for Machine Learning, Morgan Kaufmann, Los Altos, CA, 1993.
[38] Sanchez SN, Triantaphyllou E, Chen J, Liao TW. An incremental learning algorithm for constructing Boolean functions from positive and negative examples. Computers & Operations Research 29, 1677-1700, 2002.
[39] Sikora R, Piramuthu S. Framework for efficient feature selection in genetic algorithm based data mining, European Journal of Operational Research 180, 723-737, 2007.
[40] Train K. Discrete Choice Methods with Simulation, Cambridge University Press, New York, NY, 2009.
[41] Unler A, Murat A. A discrete particle swarm optimization method for feature selection in binary classification problems, European Journal of Operational Research 206, 528-539, 2010.
[42] Whitacre B, Rhinesmith C. Broadband un-adopters. Telecommunications Policy 40(1), 1-13, 2016.