DETECTION OF DISCRIMINATING RULES

Fabrizio Angiulli, Fabio Fassetti, Luigi Palopoli and Domenico Trimboli
DEIS, University of Calabria, Italy

Keywords: Data mining, Rule induction, Exceptional properties.

Abstract: Assume a population partitioned in two subpopulations, e.g. a set of normal individuals and a set of abnormal individuals, is given. Assume, moreover, that we look for a characterization of the reasons discriminating one subpopulation from the other. In this paper, we provide a technique by which such an evidence can be mined, by introducing the notion of discriminating rule, that is a kind of logical implication which is much more valid in one of the two subpopulations than in the other one. In order to avoid mining a potentially huge number of (not necessarily interesting) rule, we define a preference relationship among rules and exploit a suitable graph encoding in order to single out the most interesting ones, which we call *outstanding rules*. We provide an algorithm for detecting the outstanding discriminating rules and present experimental results obtained by applying the technique in several scenarios.

1 INTRODUCTION

In domains where there is no well assessed knowledge, and given a population partitioned in two subpopulations, it is of interest to single out the explanations distinguishing the members of one subpopulation from the members of the other subpopulation. Such a knowledge can be suitably expressed in the form of *rules*. Here, we introduce the concept of *discriminating rule*. Intuitively, a rule is a discriminating one if it is "much more valid" in one of the two given subpopulations than in the other one. The *discriminating power* of a rule is related to the difference between the confidences it attains over the two subpopulations under analysis, and can indeed be used to characterize its quality. In particular, a rule is said to be *discriminating* if its discriminating power is above a user-provided threshold. In this respect, outstanding discriminating rules are pieces of mined knowledge which appear to be promising as building blocks for the induced domain knowledge to be eventually reconstructed by the domain expert analyst.

An interesting application scenario thereof concerns the analysis of anomalous subpopulations, where it is needed to detect the motivations making some given individuals anomalous. As an example, assume a population containing genetic information about both longevous and non-longevous human individuals is given; here, it would be very useful to single out justifications for the individuals to be longevous or not. In this respect, this technique can be regarded as an extension to groups of anomalies of the technique presented in (Angiulli et al., 2009), where outlying properties of a single anomalous individual are searched for, as accounted for next in this section.

A common problem of any knowledge extractor system is that the size of mined knowledge might be so huge to be useless for the analysis purposes. And, in fact, also the number of discriminating rules can be very large, whereas only a subset thereof are usually interesting enough to be prompted to the analyst, inasmuch as most of them will encode redundant knowledge. However, selecting the rules which maximize the discriminating power value is too a weak criterion to isolate only interesting ones. Indeed, in most cases, by augmenting the body of a rule with an arbitrary simple condition, the discriminating power value associated with that rule slightly increases due to statistical fluctuations of the confidence value. To overcome this problem, we define a novel *preference relation* notion relating discriminating rules in order to single out the most interesting ones, also called *outstanding rules*. The novelty of this preference relation is that it is based on a statistical significance test rather than on generality/specificity criteria.

We point out that, even if a general analogy holds between the kind of knowledge we consider and several pattern discovery tasks, such as those of emerg-

Angiulli F., Fassetti F., Palopoli L. and Trimboli D. (2010).
DETECTION OF DISCRIMINATING RULES.
In *Proceedings of the 2nd International Conference on Agents and Artificial Intelligence - Artificial Intelligence*, pages 169-177
DOI: 10.5220/0002699701690177

ing patterns, contrasts sets and frequent pattern-based classification ((Dong and Li, 1999; Zhang et al., 2000; Bay and Pazzani, 2001; De Raedt and Kramer, 2001; Cheng et al., 2008), to cite a few), our task considerably differs from the mentioned ones. First, we notice that, to a closer look, the knowledge mined by the techniques we are presenting below is actually different. Indeed, emerging patterns, contrast sets and discriminative patterns can be well represented in the form of rules, but the only attribute allowed to occur in their heads is the class attribute, wheras we search for generic rules with any attribute in their head, while the class attribute is not considered at all. Moreover, the interestingness measure characterizing patterns searched for in the cited literature is based on measuring the frequency gap for the pattern in the two classes, while we use the confidence gap. While the former measures are (anti-)monotonic with respect to pattern generality, the latter one is non-monotonic and, hence, much more challenging to deal with. Also, these patterns tend to capture knowledge characterizing the data in a *global* sense, since they are based on the notion of absolute frequency. Conversely, the knowledge mined by means of discriminating rules characterizes the data in a *local* sense. Indeed, the confidence is related to the frequency of the condition in the head of a rule in the subpopulation of the data selected by its body. Finally, we define an innovative preference relation based on a statistical significance test, while most pattern discovery methods prefer patterns on the basis of generality and/or measure maximization.

As already noted, the technique presented here can be regarded as an extension to groups of anomalies of the technique presented in (Angiulli et al., 2009). Indeed, being the confidence insensitive to absolute frequency, it is more suitable for characterizing unbalanced subpopulations, as usually occurs when a group of anomalous individuals is compared to a whole normal population, than the support. The major differences between this work and (Angiulli et al., 2009) are as follows. In this work two subpopulations are compared, while in (Angiulli et al., 2009) only a single (outlier) object can be compared with the overall (normal) population; the discriminating measure adopted there is very different from the one developed here, since it is designed for a single object, and it is not at all clear ho to generalize it, if even possible, to deal with more than a very limited number of anomalous individuals.

The rest of the work is organized as follows. Section 2 presents preliminary definitions. Section 3 defines discriminating rule. Section 4 introduces the notion of outstanding discriminating rule. Section 5 describes the DRUID algorithm for mining outstanding rules. Section 6 presents experimental results. Finally, Section 7 concludes the work.

2 PRELIMINARIES

In this section some preliminary notions are presented.

Let $\mathbf{A} = \{a_1, \ldots, a_m\}$ be a set of attributes and T a database on $\mathbf{A}$ (multi-set of tuples on $\mathbf{A}$). A *simple condition* c on $\mathbf{A}$ is an expression of the form $a = v$, where $a \in \mathbf{A}$ and v belongs to the domain of a. A *condition* C on $\mathbf{A}$ is a conjunction $c_1 \wedge \ldots \wedge c_k$ of k ($k \geq 0$) simple conditions on $\mathbf{A}$. A condition with $k = 0$ is called an *empty condition*. In the following, for a condition C of the form $c_1 \wedge \ldots \wedge c_k$, $cond(C)$ denotes the set of simple conditions $\{c_1, \ldots, c_k\}$, while $attr(C)$ denotes the set $\{a_i \mid (a_i = v_i) \in C\}$, that is the subset of attributes of $\mathbf{A}$ appearing in simple conditions c_i of C.

Let T be a database on a set of attributes $\mathbf{A}$, let t be a tuple of T. Let $c \equiv a = v$ be a simple condition on $\mathbf{A}$. The tuple t *satisfies* c iff $t[a] = v$, where $t[a]$ denotes the value the tuple t assumes on a. Let C be a condition on $\mathbf{A}$. The tuple t *satisfies* C iff t satisfies each simple condition c_i of C. If C is an empty condition then each tuple t satisfies C. T_C denotes the database including the tuples of T which satisfy C.

Let $\mathbf{A} = \{a_1, \ldots, a_m\}$ be a set of attributes, a *rule* on $\mathbf{A}$ is an expression of the form $B \Rightarrow h$, where B is a condition on $\mathbf{A}$ and h is a simple condition on $\mathbf{A}$. B and h are called the *body* and the *head* of the rule, respectively. The *size* of the rule $R \equiv B \Rightarrow h$, denoted by $|R|$, is the cardinality of the set $cond(B)$. Let T be a database on a set of attributes $\mathbf{A}$, let t be a tuple of T, and let $R \equiv B \Rightarrow h$ be a rule on $\mathbf{A}$. t satisfies R iff t satisfies $B \wedge h$. Let $R \equiv B \Rightarrow h$ and $R' \equiv B' \Rightarrow h'$ be two rules such that $h = h'$ and $cond(B) \supset cond(B')$. Then R is said to be a *superrule* of R' and R' is said to be a *subrule* of R.

Let T be a database on a set of attributes $\mathbf{A}$, and let C be a condition on $\mathbf{A}$. The *support* of C in T, denoted by $sup_T(C)$, is the ratio $\frac{|T_C|}{|T|}$ of the number of tuples of T satisfying C over the size of T. Given a database T on $\mathbf{A}$ and a threshold σ, $0 \leq \sigma \leq 1$, a condition C is said to be σ-*supported* by T iff $sup_T(C) \geq \sigma$.

Let T be a database on a set of attributes $\mathbf{A}$, and let R be a rule $B \Rightarrow h$ on $\mathbf{A}$. The *confidence* of R in T, denoted by $cnf_T(R)$, is the ratio $\frac{|T_{B \wedge h}|}{|T_B|}$ of the number of tuples of T satisfying R over the number of tuples satisfying B.

MotherHair	ChildHair
brown	brown
brown	brown
brown	brown
brown	brown
brown	brown
brown	blonde
brown	blonde
blonde	brown
blonde	brown
blonde	brown
blonde	brown
blonde	brown
blonde	brown
blonde	blonde
blonde	blonde

(a) T_{br} : Brown father

MotherHair	ChildHair
brown	blonde
brown	blonde
brown	blonde
brown	brown
brown	brown
brown	brown
brown	brown
brown	brown
brown	brown
brown	brown
brown	brown
blonde	blonde
blonde	blonde
blonde	blonde
blonde	blonde
blonde	blonde

(b) T_{bl} : Blonde father

Figure 1: Hair color databases.

3 DISCRIMINATING RULES

In this section the notion of discriminating rule is introduced . We will make use of a running example in order to help illustrating the discussed matter.

Example 1. Figure 1 shows two databases reporting hair colors of wives and children of some male individuals. Specifically, the first database, T_{br}, is associated with males with brown hair whereas the second one, T_{bl}, is associated with males with blonde hair. We aim at discovering rules characterizing only one of the two databases.

We start by providing the definition of discriminating power. Let T' and T'' be two databases on a set of attributes $\mathbf{A}$, and let R be a rule on $\mathbf{A}$. The *discriminating power* of R (with respect to T' and T'') is:

$$pow(R) = \frac{|\, cnf_{T'}(R) - cnf_{T''}(R)\,|}{\max\{\, cnf_{T'}(R),\, cnf_{T''}(R)\,\}}.$$

The discriminating power measures the relative gap between the confidence value associated with a rule when we move from a database to the other. Note that, the larger the absolute difference between $cnf_{T'}(R)$ and $cnf_{T''}(R)$, the larger the discriminating power of R.

Example 1 (continued). Consider Figure 1 again, and the rule R_{ex}:

$$\text{MotherHair} = \text{“blonde”} \Rightarrow \text{ChildHair} = \text{“blonde”}.$$

The confidence of r on T_{br} is $\frac{2}{8} = 0.25$ whereas on T_{bl} is $\frac{5}{5} = 1$, and then the discriminating power of R_{ex} is $pow(R_{ex}) = \frac{|0.25-1|}{\max\{0.25,1\}} = 0.75$. The rule R_{ex} asserts that for a child having a blonde mother, the probability of being blonde is much higher if its father is blonde rather than brown. And, in particular, such a probability is 1 in the former case and 0.25 in the latter case. This knowledge hidden in the data at hand is clearly expected by the well-known *Mendelian inheritance law*. Since brown hair is dominating over blonde hair, if both parents are blonde haired the child is blonde. This justifies the value 1 for the confidence of r on T_{bl}. Conversely, if the father is brown and the mother is blonde, than two cases can arise: the genotype of the father (*i*) includes two genes associated with brown hair, or (*ii*) includes one gene associated with brown hair and one associated with blonde hair. In the case (*i*) the child is brown for sure, while in case (*ii*) the probability of being brown (or, equivalently, blonde) is about fifty percent. Summarizing, if (for the sake of simplicity) we assume that cases (*i*) and (*ii*) occur with the same frequency in the considered population, than the probability of having a blonde haired child with a brown father and a blonde mother is about twenty-five percent, which agrees with the value 0.25 for the confidence of r on T_{br}. We also note that R_{ex} is more interesting than the empty-body rule $\emptyset \Rightarrow$ ChildHair = “blonde”, corresponding to the frequency of the value “blonde” on the attribute “ChildHair” which is approximatively 0.27 on T_{br} and 0.42 on T_{bl}, resulting in a discriminating power of about 0.37.

The definition of discriminating rule builds on that of discriminating power.

Let T' and T'' be two databases on a set of attributes $\mathbf{A}$, let θ_{pow} be a *threshold* (real number in the range $[0,1]$), and let $R \equiv B \Rightarrow h$ be a rule on $\mathbf{A}$. Then, R is a *discriminating rule* iff $pow(R) \geq \theta_{pow}$.

Intuitively, a discriminating rule characterizes sufficiently well the tuples of one database with respect to those of the other. Optionally, we may require that the rule satisfies some additional constraints concerning support and confidence, that are (c_1) $sup_{T'}(B) \geq \theta'_{sup}$, ($c_2$) $sup_{T''}(B) \geq \theta''_{sup}$, and ($c_3$) $\max\{cnf_{T'}(R), cnf_{T''}(R)\} \geq \theta_{cnf}$, where θ'_{sup}, θ''_{sup}, and θ_{cnf} are suitable thresholds.

Example 1 (continued). For instance, the rule R_{ex} is discriminating for $\theta'_{sup} = \theta''_{sup} = 0.25$, $\theta_{cnf} = 0.5$, and $\theta_{pow} = 0.7$, since $sup_{T_{br}}(r) = \frac{8}{15} = 0.533$, $sup_{T_{bl}}(r) = \frac{5}{19} = 0.263$, $cnf_{T_{br}}(r) = \frac{8}{15} = 0.533$, and $pow(r) = 0.75$.

4 OUTSTANDING RULES

As already remarked, while the number of discriminating rules can be very large, only a subset thereof can be considered interesting enough to be prompted to the analyst. Hence, in order to single out the most interesting rules out of a set of discriminating ones,

we are next defining a preference relation between discriminating rules.

4.1 Preference Relation

The preference relation is defined only between pairs of rules which are one the superrule of the other.

Let T' and T'' be two databases defined on the same set of attributes $\mathbf{A}$, let R be a rule on $\mathbf{A}$ and let R' be a subrule of R. Then, R is *preferred* to R', denoted $R \prec R'$, iff

1. $pow(R) > pow(R')$, and
2. either the difference $cnf_{T'}(R) - cnf_{T'}(R')$ or the difference $cnf_{T''}(R) - cnf_{T''}(R')$ is statistically significative.

Otherwise, R' is *preferred* to R, and denoted $R' \prec R$. According to the above definition, a subrule is always to be preferred to a superrule having a smaller or equal discriminating power value. To be preferred, a superrule needs not only to have a greater discriminating power than the subrule, but also a significative gap in confidence.

The significance of the gap between two confidences can be measured by exploiting a suitable *statistical test*. We will describe next in this section the statistical test employed in the current implementation of the algorithm.

The rationale underlying this definition is that shorter rules are generally preferable over longer ones since longer rules tend to overfit and, also, to be less intelligible. Moreover, a notion of preference solely based on the discriminating power is seemingly far too weak to be practically effective. As already pointed out, indeed, augmenting the body of a rule with a randomly selected simple conditions may often increase the discriminating power associated with the rule due simply to statistical fluctuations of the confidence values. Hence, the definition states that a longer rule is to be preferred only if there is evidence for at least one of the confidence values associated with it to be undoubtedly higher.

Note that the relation is not transitive since, for some three rules r, r' andr'', even if both the differences $|cnf(r) - cnf(r')|$ and $|cnf(r') - cnf(r'')|$ do not pass the test, it can be the case that the difference $|cnf(r) - cnf(r'')|$ is indeed large enough to pass the test.

Significance Test. The statistical significance of the difference between two confidence values can be computed by means of the *binomial test* as described in the rest of this section.

Let T be a database on $\mathbf{A}$. Let $R \equiv B \Rightarrow h$ and $R' \equiv B' \Rightarrow h$ be two rules on $\mathbf{A}$ such that R is a superrule of R'. Let n_B be the value $|T_B|$ and n_R be the value $|T_{B \wedge h}|$. Then, $cnf_T(R) = \frac{n_R}{n_B}$. Moreover, let $n_{B'}$ be the value $|T_{B'}|$ and $n_{R'}$ be the value $|T_{B' \wedge h}|$. Then, $cnf_T(R') = \frac{n_{R'}}{n_{B'}}$.

Since R is a superrule of R', then the tuples in T_B are a subset of $T_{B'}$ and, hence, n_B is smaller than or equal to $n_{B'}$. Analogously, the tuples in $T_{B \wedge h}$ are a subset of $T_{B' \wedge h}$ and, hence, n_R is smaller than or equal to $n_{R'}$.

If the attributes belonging to the set $attr(B) \backslash attr(B')$ were not correlated to the attributes in $attr(B')$, then the tuples in T_B could be assumed as generated by a sequence of n_B random extractions from $T_{B'}$. Hence, the random variable X, representing the number of tuples in T_B satisfying h, is distributed according to a binomial distribution, where a success represents the extraction of a tuple satisfying h. The number of extractions is n_B and the probability of success is the probability of extracting a tuple satisfying h, which corresponds to $\frac{n_{R'}}{n_{B'}}$. The expected value $E[X]$ is the product of the number of extractions and the probability of success, namely $\overline{n_R} = n_B \cdot \frac{n_{R'}}{n_{B'}}$. Hence, the expected confidence of the rule R is

$$cnf_T(R) = \frac{n_B \cdot \frac{n_{R'}}{n_{B'}}}{n_B} = \frac{n_{R'}}{n_{B'}}$$

which is equal to the confidence of R'.

Clear enough, due to statistical fluctuations, the number n_R of tuples satisfying $B \wedge h$ will not be exactly equal to $\overline{n_R}$, and then the value of $cnf_T(R)$ can be slightly different from the value of $cnf_T(R')$.

In order to test if such a difference is due to statistical fluctuation, it must be checked if it is statistically significative. To this end the binomial test can be employed. Let X be a random variable following the binomial distribution with parameters $n = n_B$ and $p = \frac{n_{R'}}{n_{B'}}$. This test computes the probability to get a value for the binomial random variable X farther from $\overline{n_R}$ than n_R, and then checks if this probability is lower than the significance level 0.05. In other words, it must be verified if the following inequality holds:

$$Pr(|X - \overline{n_R}| \geq |n_R - \overline{n_R}|) < 0.05.$$

Let $\mathcal{F}(x,y)$ denote the cumulative binomial distribution function with parameters x and y. The relation above can be rewritten as:

$$\mathcal{F}(\overline{n_R} + |n_R - \overline{n_R}|) - \mathcal{F}(\overline{n_R} - |n_R - \overline{n_R}|) \geq 0.95. \quad (1)$$

Clear enough, within the proposed approach, any other sensible statistical significance test could replace the adopted one.

Example 1 (continued). Consider rules R_{ex} and R'_{ex} again. Let us check the significance of the difference between the

confidence values associated to R_{ex} and R'_{ex} on the database T_{bl}. Thus, $n_{R_{ex}} = 5$, $n_B = 5$, $n_{R'_{ex}} = 8$ and $n_{B'} = 19$. $\overline{n_R}$ can be computed as $5 \cdot \frac{8}{19}$ and then $\overline{n_R} = 2$. In order to evaluate the test the following value has to be determined: $\mathcal{F}(2+|5-2|) - \mathcal{F}(2-|5-2|)$. Since the value of the above expression is 1, hence greater than 0.95, then it can be concluded that R_{ex} is actually preferred to R'_{ex}.

4.2 Outstanding Rules

Here, we define the notion of preferability graph, which encodes discriminating rules (by means of nodes) and preferability relations (by means of arcs). The preferability graph will be exploited to single out the *outstanding* discriminating rules.

We have already noted that the number of discriminating rules can be very large, but in general only a subset thereof can be considered interesting enough to be prompted to the analyst. In that respect, loosely speaking, the outstanding discriminating rules will represent rules whose interestingness for the analyst is maximal.

Given databases T' and T'', and a condition h, a *preferability graph* $\mathcal{G}^h = (V,U,E)$ *w.r.t. the condition* h (whenever the head condition h is clear by the context, we will omit the superscript of $\mathcal{G}$ in referring to a graph), is a directed graph, with V a set of preferability nodes (or, simply, nodes – see, below, the definition of preferability node), $U \subseteq V$ a set of (*blocked*) preferability nodes, and E a set of arcs on V.

A *preferability node* n of a graph $\mathcal{G}^h$ is a node having associated a discriminating rule $R(n) \equiv B \Rightarrow h$. Hence, all the rules associated with nodes of a preferability graph $\mathcal{G}^h$ have the same condition h in their head. For each discriminating rule of the form $B \Rightarrow h$ there exists at most one node in $\mathcal{G}^h$ associated with it. There exists an arc (n,m) in $\mathcal{G}^h$ from node n to node m iff $R(n)$ is preferred to $R(m)$.

By $\widehat{\mathcal{G}}^h$ we denote the preferability graph $(V,\emptyset,E)$ where *all* discriminating rules $R \equiv B \Rightarrow h$ are represented.

Given two nodes n and m, m is *reachable* from n in $\mathcal{G}^h$, denoted $n \rightarrow m$, iff there exists a directed path from n to m in $\mathcal{G}^h$ It is assumed that, for each node n, it holds that $n \rightarrow n$. Otherwise, m is *not reachable* from n, denoted as $n \not\rightarrow m$. A node n is said to be a *supernode* (*subnode*, resp.) of a node m if $R(n)$ is a superrule (subrule, resp.) of $R(m)$. Note that by definition of preferability graph $\mathcal{G}^h$, for each pairs of nodes n and m of $\mathcal{G}^h$ such that m is a supernode of n there exists in $\mathcal{G}^h$ either the arc (n,m) or the arc (m,n), but not both. A *connected component* $\mathcal{C}$ of $\mathcal{G}$ is a maximal subset of the nodes of $\mathcal{G}$ such that, for each $n,m \in \mathcal{C}$, $n \rightarrow m$ hold. Given a node n, the connected component in $\mathcal{G}^h$ which n belongs to is denoted $conn(n,\mathcal{G}^h)$ (or, simply, $conn(n)$ in the following).

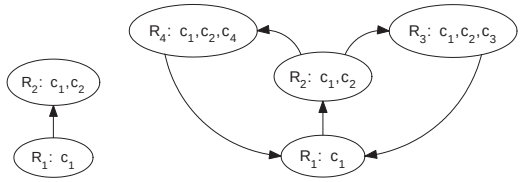


Figure 2: Preferability Graph - Example.

Given a subset N of V, the *restriction* $\mathcal{G}_N$ of the graph $\mathcal{G} = (V,U,E)$ on the set of nodes N, is the subgraph of $\mathcal{G}$ induced by the nodes in N, that is $\mathcal{G}_N = (N, U \cap N, \{(n,m) \mid n,m \in N \wedge (n,m) \in E\})$.

Example 2. Consider two databases T' and T''. For the sake of simplicity, assume that all the rules considered in the following score confidence 1 on T'', so that whenever we need to evaluate the statistical significance of the difference between two confidences, we restrict our attention on T' only. Suppose that the set $\mathcal{R}$ of rules complying with the support constraints consists in the following two rules:

- $R_1 \equiv c_1 \Rightarrow h$, $|T'_{c_1}| = 250$, $|T'_{c_1 \wedge h}| = 100$;
- $R_2 \equiv c_1 \wedge c_2 \Rightarrow h$, $|T'_{c_1 \wedge c_2}| = 250$, $|T'_{c_1 \wedge c_2 \wedge h}| = 100$,

where c_1, c_2 and h are simple conditions.

In order to establish the preference relation between R_1 and R_2, first their discriminating power has to be computed. The confidence of R_1 on T' is $\frac{100}{250} = 0.4$, whereas it is 1 on T''. Then, $pow(R_1) = 0.6$. Conversely, the confidence of R_2 on T' is $\frac{50}{150} = 0.3$, and it is 1 on T''. Then, $pow(R_2) = 0.7$. Since $pow(R_1) < pow(R_2)$ and since R_1 is a subrule of R_2, we need to evaluate if the gap between the confidences of R_1 and R_2 is statistically significative in at least one of the two databases. Because of the gap between the confidences of R_1 and R_2 on T'' is 0, we compute the binomial test only on T': $\mathcal{F}(60+|50-60|) - \mathcal{F}(60-|50-60|) = 0.9036 < 0.95$. Since this gap is not statistically significative, R_1 is preferred to R_2. The associated preferability graph is reported in Figure 2(a).

Suppose, now, that $\mathcal{R}$ contains two further rules:

- $R_3 \equiv c_1 \wedge c_2 \wedge c_3 \Rightarrow h$, $|T'_{c_1 \wedge c_2 \wedge c_3}| = 45$, $|T'_{c_1 \wedge c_2 \wedge c_3 \wedge h}| = 9$,
- $R_4 \equiv c_1 \wedge c_2 \wedge c_4 \Rightarrow h$, $|T'_{c_1 \wedge c_2 \wedge c_4}| = 45$, $|T''_{c_1 \wedge c_2 \wedge c_4 \wedge h}| = 9$,

and let us compute the discriminating powers of R_3 and R_4. We obtain that $pow(R_3) = 0.8$ and $pow(R_4) = 0.8$.

First, note that no preferability relation holds for R_3 and R_4 and, then, no arc connects them in the preferability graph. Note that all the rules have confidence 1 on T''. Consider, now, the pair R_2 and R_3. Since $pow(R_2) < pow(R_3)$ and R_2 is a subrule of R_3, we compute the binomial test obtaining: $\mathcal{F}(15+|9-15|) - \mathcal{F}(15-|9-15|) = 0.9410 < 0.95$, asserting that R_2 is preferred to R_3, and then an arc

from R_2 to R_3 is there in the preferability graph. Consider the pair R_1 and R_3. Since $pow(R_1) < pow(R_3)$ but R_1 is a subrule of R_3, we compute the binomial test obtaining: $\mathcal{F}\ (18+|9-18|)-\mathcal{F}\ (18-|9-18|) = 0.9942 > 0.95$, asserting that R_3 is preferred to R_1, and then an arc from R_3 to R_1 is there in the preferability graph. This example confirms that, in general, the preferability relation is not transitive.

As far as R_4 is concerned, its relations with R_1 and R_2 are exactly the same as R_3. The resulting preferability graph is reported in Figure 2(b). Observe that R_1, R_2, R_3 and R_4 form a connected component.

In order to characterize outstanding discriminating rules, we next introduce the concept of *candidate* rule.

First of all, it is considered the basic situation in which the graph is a single connected component, and the notion of candidate node in such a graph is defined. Intuitively, a candidate node is associated with a potentially outstanding rule.

Let $\mathcal{G} = (V,U,E)$ be a preferability graph such that V is a connected component of $\mathcal{G}$; a node n in V is said to be *candidate* in $\mathcal{G}$ iff both the two following conditions hold:

1. for each supernode u of n, it holds that $pow(R(n)) \geq pow(R(u))$, and
2. for each subnode u of n, it holds that $pow(R(n)) > pow(R(u))$.

The rationale underlying this definition is that, for each node n in a connected component, there exists an other node n' in the same component such that $R(n')$ is preferred to $R(n)$, thus from the point of view of the preference relation, within the same connected component, there is no node which is preferable to all the others. Hence, it is seemingly sensible to single out as candidates those nodes whose associated rules score the maximal discriminative power value among their associated supernodes and subnodes. Moreover, the equal sign in condition 1 makes it shortest rules preferable when ties are there in the inclusion hierarchy.

Example 2 (continued). Consider the graph of Figure 2(b). This graph forms a connected component. According to the definition provided above, the candidate nodes are R_3 and R_4, since their discriminating power is maximum among those associated with the nodes of the graph and each of their subrules has strictly smaller discriminating power. Note that, if the discriminating power of R_1 (or, equivalently, R_2, resp.) were larger than that of all the other rules, then the candidate node would only be n_1 (or n_2, resp.).

Clear enough, in general, a graph does not include a single connected component. Thus, we provide next the definition of source node, which is conducive to the definition of candidate node in a general preferability graph.

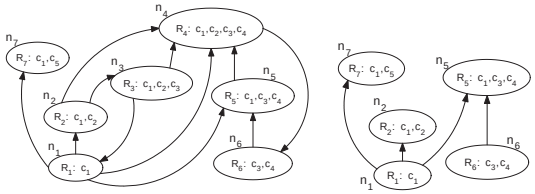


Figure 3: Example Graph.

Let $\mathcal{G}$ be a preferability graph; a node n of $\mathcal{G}$ is a *source* if the following condition holds: for each node m such that $m \rightarrow n$, it holds that $n \rightarrow m$.

Hence, a source node is a node that reaches all the nodes that reach it in turn. Note that there might be nodes that are reached from a source but not reach the source.

Example 3. Consider Figure 3a. The node n_2 is a source since nodes reaching n_2 (namely n_1 and n_3) are also reached from it. Conversely, n_6 is not a source since it is reached, for example, by n_2 but n_6 does not reach n_2.

Now we are in the position of providing the definition of candidate node in a general graph.

Let $\mathcal{G}$ be a preferability graph. A node n of $\mathcal{G}$ is said to be *candidate* in $\mathcal{G}$ iff n is a source node of $\mathcal{G}$ and n is candidate in $\mathcal{G}_{conn(n)}$ (according to Def. 4.2 above).

Clear enough, if a node in a connected component $\mathcal{C}$ is source, then all the nodes in $\mathcal{C}$ are sources as well. Hence, in the graph, there are no nodes outside $\mathcal{C}$ which are preferable to the nodes in $\mathcal{C}$ and, therefore, the candidate nodes have to be singled out among those in $\mathcal{C}$.

Example 3 (continued). Consider Figure 3a again. In the graph the source nodes are n_1, n_2 and n_3, all belonging to the same connected component. Then, the candidate node is that node amongst n_1, n_2 and n_3 scoring the highest discriminating power.

Next the definition of transformed graph associated with a preferability graph $\mathcal{G}$, leading to the definition of outstanding rule, is given.

Let $\mathcal{G} = (V,U,E)$ be a preferability graph. The *transformed graph* $t(\mathcal{G}) = (V',U',E')$ associated with $\mathcal{G}$ is the graph obtained as follows:

- V' is obtained from V by removing both the candidate nodes in $\mathcal{G}$ and all their supernodes,
- U' is $(U \cup S) \cap V'$, where S is the set containing all the subnodes of the candidate nodes in $\mathcal{G}$, and
- E' is the subset of the arcs in E linking the nodes in V'.

Since for each $\mathcal{G} = (V,U,E)$, with $V \neq \emptyset$, there exists at least one candidate node in $\mathcal{G}$, the set of nodes of the graph $t(\mathcal{G})$ is always a strict subset of V (unless $V = \emptyset$).

Note that the transformed graph $t(\mathcal{G})$ is again a preferability graph, hence the operator $t(\cdot)$ can be applied also to it. Then, given a non-negative integer number $k \geq 0$, it can be defined the concept of *transformed graph of order k associated with* $\mathcal{G}$, $t^k(\mathcal{G})$, which is defined recursively as follows: $t^0(\mathcal{G})$ is $\mathcal{G}$, and, for $k > 0$, $t^k(\mathcal{G})$ is $t(t^{k-1}(\mathcal{G}))$.

Let $\mathcal{G}^\emptyset$ be the preferability graph $(\emptyset,\emptyset,\emptyset)$. We note that $t(\mathcal{G}^\emptyset) = \mathcal{G}^\emptyset$. Moreover, since $t(\mathcal{G})$ is a strict subgraph of $\mathcal{G}$ (unless $\mathcal{G} = \mathcal{G}^\emptyset$), it follows that for each preferability graph $\mathcal{G}$, there exists a finite integer number $K \leq |V|$ such that $t^K(\mathcal{G}) = \mathcal{G}^\emptyset$. Hence, the operator $t(\cdot)$ always finitely converges to the graph $\mathcal{G}^\emptyset$.

Now we are in the position of providing the notion of outstanding node and outstanding rule. A node n is said to be *outstanding* in $\mathcal{G}$ iff there exists an integer $k \geq 0$ such that the node n is candidate in $t^k(\mathcal{G}) = (V,U,E)$ and does not belong to U. A rule $R \equiv B \Rightarrow h$ is *outstanding* iff there exists an outstanding node n in $\widehat{\mathcal{G}}^h$ such that $R = R(n)$.

Example 3 (continued). Consider the graph $\widehat{\mathcal{G}}^h$ shown in Figure 3(a), then $\widehat{\mathcal{G}}^h = (\{n_1,n_2,n_3,n_4,n_5,n_6,n_7\},\emptyset,$ $\{(n_1,n_2),\ (n_1,n_4),\ (n_1,n_5),\ (n_2,n_3),\ (n_2,n_4),\ (n_3,n_1),$ $(n_3,n_4),(n_4,n_6),(n_5,n_4),(n_6,n_5),(n_1,n_7)\})$. Assume that the discriminating power of R_3 is greater than that of both R_1 and R_2. Thus, the only candidate node in $\widehat{\mathcal{G}}^h$ is n_3, and, hence, $t^1(\widehat{\mathcal{G}}^h) = (V',U',E')$ where:

$V' = \{n_1,n_2,n_5,n_6,n_7\}$,

$U' = (\emptyset \cup \{n_1,n_2\}) \cap \{n_1,n_2,n_5,n_6,n_7\} = \{n_1,n_2\}$, and

$E' = \{(n_1,n_2),\ (n_1,n_5),\ (n_6,n_5),\ (n_1,n_7)\}$.

The resulting graph is that reported in Figure 3(b). Moreover, n_3 is an outstanding node, since it is a candidate in $t^0(\widehat{\mathcal{G}}^h) = \widehat{\mathcal{G}}^h$ and does not belong to U and, as such, R_3 is an outstanding rule. In $t^1(\widehat{\mathcal{G}}^h)$ there are two source nodes: n_1 and n_6 which are also candidate nodes. Nevertheless, n_1 is not an outstanding node in $t^1(\widehat{\mathcal{G}}^h)$ since it belongs to U', while n_6 is. By applying the $t(\cdot)$ operator again, we obtain $t^2(\widehat{\mathcal{G}}^h) = (V'',U'',E'')$ where: $V'' = \emptyset$, $U'' = \{n_1,n_2\} \cap V'' = \emptyset$, and $E'' = \emptyset$. Hence, $t^2(\widehat{\mathcal{G}}^h) = \mathcal{G}^\emptyset$.

Summarizing, in $\widehat{\mathcal{G}}^h$ there are two outstanding nodes, that are, n_3 and n_6 and, hence, R_3 and R_6 are the outstanding rules.

Before leaving the section, we provide the rationale underlying the asymmetry of the operator $t(\cdot)$ in treating supernodes and subnodes of candidate nodes.

Assume that the supernodes $\{n'\}$ of a candidate node n are maintained in the transformed graph $t(\mathcal{G})$ and marked as blocked, as it is the case for the subnodes of n. Thus, if one such a node n' becomes candidate in $t(\mathcal{G})$, then all its subnodes n'' are marked as blocked and prevented to be selected as outstanding. Clearly, while the rule $R(n')$ is not interesting enough to be prompted to the analyst since its (better) subrule $R(n)$ has been already selected, this is not the case for the rule $R(n'')$ which, conversely, is neither a subnode nor a supernode of $R(n)$.

Assume, conversely, that the subnodes $\{n'\}$ of a candidate node n are deleted from the transformed graph $t(\mathcal{G})$, as it is the case for the supernodes of n. Moreover, assume that n' has a supernode n'' in $\mathcal{G}$ such that $R(n')$ is preferred to $R(n'')$. Since the node n' is not in $t(\mathcal{G})$, n'' could become an outstanding node. Recall that the rule $R(n')$ is a subrule of both rules $R(n)$ and $R(n'')$. Since $R(n)$ is preferred to $R(n')$, it is the case that the rule $R(n)$ significantly increases the discriminating power of $R(n')$ by augmenting its body with some interesting, that is to say correlated, simple conditions. Furthermore, since $R(n')$ is preferred to $R(n'')$, it is also the case that the rule $R(n'')$ augments the body of $R(n')$ with some simple conditions, but this time they cannot be considered interesting, as the discriminating power of $R(n'')$ is worse than that of $R(n')$.

Phase 1:
Determine the set $\mathcal{B}$ of conditions co-supported by the databases T' and T''

Phase 2:
For each simple condition h that can be built on the set of attributes **A**:

a. build the graph $\widehat{\mathcal{G}}^h$

b. Determine the outstanding nodes $\mathcal{N}$ in $\widehat{\mathcal{G}}^h$

c. Augment the solution set $\mathcal{R}$ with the set of rules $\{R(n) \mid n \in \mathcal{N}\}$

Return the rules in $\mathcal{R}$ ranked by decreasing discriminating power

Figure 4: The Discriminating RUle InDuctor (DRUID) algorithm.

5 ALGORITHM

Given two databases T' and T'' on the same set of attributes **A**, we are interested in finding the outstanding rules discriminating T' from T''. In this section we present the algorithm DRUID (for Discriminating RUle InDuctor) solving this task. The algorithm consists in two main phases (see Figure 4).

We say that a condition is co-supported by databases T' and T'' if its support on database T' is

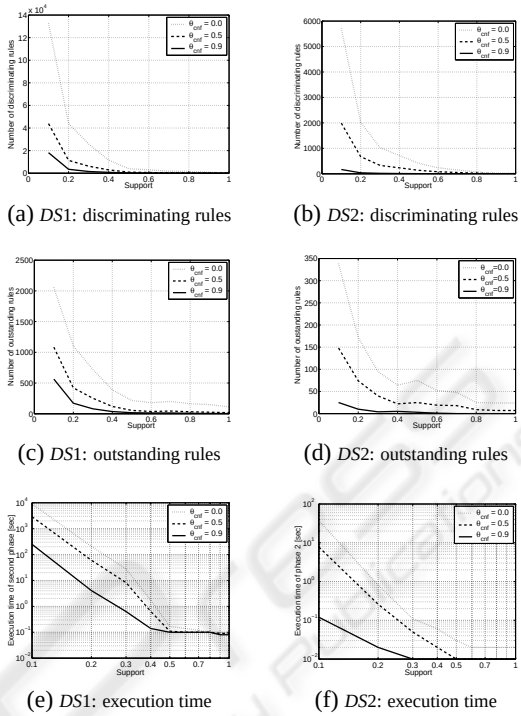
above threshold θ'_{sup} and its support on database T'' is above threshold θ''_{sup}. First of all the set $\mathcal{B}$ of co-supported conditions in the two databases has to be determined (phase 1). This can be done by adapting any efficient frequent itemset mining algorithm to work simultaneously on two databases in order to take into account only co-supported conditions. In our current implementation an A-priori like algorithm (Rakesh et al., 1993) is employed to compute the set $\mathcal{B}$ of co-supported conditions. The set $\mathcal{B}$ is mined only once, since it can be "reused" for each potential head.

During Phase 2 the outstanding discriminating rules are mined. For each simple condition h employable as head of a discriminating rule, phase 2a of the algorithm builds the graph $\widehat{\mathcal{G}}^h$ associated with h. Subsequent phase 2b determines the outstanding nodes in $\widehat{\mathcal{G}}^h$ by applying the operator $t(\cdot)$, until the graph becomes empty. The outstanding nodes in the graphs $\widehat{\mathcal{G}}^h$ are collected into the set $\mathcal{R}$, and the associated outstanding rules are eventually presented to the user.

As for the temporal cost of the method, the cost of Phase 1, corresponding to the execution of the A-priori algorithm, is in general exponential with respect to the number of database attributes. As for the cost of Phase 2, it is polynomial in the size of the graph, whose number of nodes is upper bounded by the size $|\mathcal{B}|$ of the output of the A-priori algorithm, and linear in the number of tuples of the database, due to the need of computing the confidence of the rules.

6 EXPERIMENTAL RESULTS

In this section, we present experimental results obtained by applying the proposed technique on some real databases. We considered two extensively used test datasets, that are *Mushroom*[1] and *Census*[2] (also referred to in the following as *DS*1 and *DS*2, respectively). The Mushroom dataset includes descriptions of 8,124 hypothetical samples corresponding to 23 species of gilled mushrooms in the Agaricus and Lepiota Family. There are 22 categorical attributes. Each species is identified as *edible* (4,208 instances) or *poisonous* (3,916 instances). On the basis of this classification, the data was partitioned in two databases T_e and T_p. The Census dataset contains information about old people. It consists of 333,011 tuples each of which is composed of 10 categorical attributes plus one class attribute *Income*, which represents the annual income, assuming two distinct values, that are *"below50K"* and *"over50K"*. Hence, we split it in two databases, $T_{<50}$ (consisting of 327,216 tuples) and $T_{>50}$ (consisting of 5,795 tuples), on the basis of the value of the class attribute. We considered this dataset in order to verify the technique on two significantly unbalanced subpopulations. Indeed, the $T_{>50}$ subpopulation can be considered here as including "anomalous" individuals to be compared against the "normal" subpopulation $T_{<50}$.



(a) *DS*1: discriminating rules (b) *DS*2: discriminating rules

(c) *DS*1: outstanding rules (d) *DS*2: outstanding rules

(e) *DS*1: execution time (f) *DS*2: execution time

Figure 5: Experimental results.

Experiments are organized as follows. First of all, we present a sensitivity analysis of the method by measuring execution time, number of discriminating rules, and number of outstanding rules, for various combinations of the threshold parameters θ_{sup} and θ_{cnf}. Following that, we shall comment upon some outstanding rules.

Figure 5 reports the results of the sensitivity analysis. The parameter θ_{sup} was varied between 0.1 and 1.0, while three distinct values for the parameter θ_{cnf} were considered: 0.0, 0.5, and 0.9. Figures 5(a) and 5(b) report the number of discriminating rules. Figures 5(c) and 5(d) report the number of outstanding rules. Finally, Figures 5(e) and 5(f) report the execution time (in seconds). The time required by Phase 2 clearly depends on the number of discriminating rules in the databases. This number increases sensibly only for low support values, but in all cases the DRUID al-

[1] http://archive.ics.uci.edu/ml/.

[2] http://www.cs.waikato.ac.nz/ml/weka/index_datasets.html.

gorithm terminated its work in a reasonable amount of time. It employed about three hours on the hardest instance considered on *Census*. We point out that this execution time was reached for very low values of the thresholds and, in particular, for $\theta_{cnf} = 0$. Indeed, for more sensible values of the parameters it rapidly decreases to few seconds. Finally, the following table shows the execution times (in seconds) of the Phase 1 of the algorithm, that is the variant of the A-priori algorithm for mining co-supported conditions.

θ_{sup} =	0.1	0.2	0.3	0.5	0.7	0.9	1.0
Mushroom	0.41	0.24	0.16	0.05	0.03	0.01	0.01
Census	4.30	2.55	2.19	1.28	0.54	0.01	0.01

Next we comment upon some oustanding rules returned by running DRUID. Consider the *Mushroom* dataset. The rule $cap{-}surface = f \wedge cap{-}shape = x \Rightarrow odor = n$, has $pow = 0.99$, $cnf_e = 0.97$, $cnf_p = 0.01$, $sup_e = 0.17$, $sup_p = 0.11$. It concerns mushrooms with fibrous cap surface and convex cap shape. The rule asserts that edible mushrooms thereof are very likely to be odorless, while poisonous are very likely to be odorous.

The rule $cap{-}color = g \wedge gill{-}spacing = c \Rightarrow ring{-}type = p$, has $pow = 0.84$, $cnf_e = 1.00$, $cnf_p = 0.16$, $sup_e = 0.15$, $sup_p = 0.20$. It concerns mushrooms with gray cap color and closed gills. The rule asserts that edible mushrooms thereof are more likely to have a pendant ring than poisonous ones. The rule $stalk{-}surface{-}b{-}r = s \wedge ring{-}number = o \Rightarrow gill{-}size = n$, has $pow = 0.92$, $cnf_e = 0.07$, $cnf_p = 0.90$, $sup_e = 0.72$, $sup_p = 0.37$. It concerns mushrooms with smooth surface of the stalk under the ring and one ring. The rule asserts that poisonous mushrooms thereof are more likely to have narrow gills than edible ones.

Consider now the *Census* dataset. The rule $immigr = before75 \Rightarrow english = poor$, has $pow = 0.83$, $cnf_{T_{<50}} = 0.42$, $cnf_{T_{>50}} = 0.07$, $sup_{T_{<50}} = 0.10$, $sup_{T_{>50}} = 0.12$. It concerns people immigrated before year 1975. The rule asserts that the individuals thereof whose income is below 50K are more likely to speak a poor English than those having income above 50K. The rule $urban = false \Rightarrow race = black$, has $pow(R_2) = 0.80$, $cnf_{T_{<50}} = 0.42$, $cnf_{T_{>50}} = 0.09$, $sup_{T_{<50}} = 0.23$, $sup_{T_{>50}} = 0.15$. It concerns people living in rural areas. The rule asserts that the individuals thereof whose income is below 50K are more likely to be black than those having income above 50K. The rule $region = midw \wedge age = below75 \Rightarrow sex = male$, has $pow = 0.80$, $cnf_{T_{<50}} = 0.27$, $cnf_{T_{>50}} = 0.55$, $sup_{T_{<50}} = 0.11$, $sup_{T_{>50}} = 0.12$. It concerns people whose age is below 75 years and living in the Midwest. The rule asserts that the individuals thereof whose income is above 50K are more likely to be male than those having income below above 50K.

7 CONCLUSIONS

In this paper, the problem of characterizing the features distinguishing two given populations has been analyzed. We introduced the notion of discriminating rule, a kind of logical implication which is much more valid in a population than in the other one. We suggested their use for characterizing anomalous subpopulations. In order to avoid for the analyst to be overwhelmed by the potentially huge number of rules discriminating the two populations, we defined an original notion of preference relation among discriminating rules, which is interesting from a semantical viewpoint, but it is challenging to deal with since it is not transitive and, hence, no monotonicity property can be exploited to efficiently guide the search. We proposed the DRUID algorithm for detecting the outstanding discriminating rules, and discussed preliminary experimental results.

REFERENCES

Angiulli, F., Fassetti, F., and Palopoli, L. (2009). Detecting outlying properties of exceptional objects. *ACM Trans. on Database Systems (TODS)*, 34(1).

Bay, S. D. and Pazzani, M. J. (2001). Detecting group differences: Mining contrast sets. *Data Mining and Knowledge Discovery*, 5(3):213–246.

Cheng, H., Yan, X., Han, J., and Yu, P. S. (2008). Direct discriminative pattern mining for effective classification. In *ICDE*, pages 169–178.

De Raedt, L. and Kramer, S. (2001). The levelwise version space algorithm and its application to molecular fragment finding. In *IJCAI*, pages 853–862.

Dong, G. and Li, J. (1999). Efficient mining of emerging patterns: Discovering trends and differences. In *KDD*, pages 43–52.

Rakesh, A., Tomasz, I., and Arun, S. (1993). Mining association rules between sets of items in large databases. In *SIGMOD*, pages 207–216.

Zhang, X., Dong, G., and Ramamohanarao, K. (2000). Exploring constraints to efficiently mine emerging patterns from large high-dimensional datasets. In *KDD*, pages 310–314.