Proc. 3rd European Conference on Principles and Practice of Knowledge Discovery in Databases -- PKDD'99 (LNCS 1704), September 15-18, 1999, Prague, 536-541.

# Mining Possibilistic Set-Valued Rules by Generating Prime Disjunctions

Alexandr A. Savinov

GMD — German National Research Center for Information Technology
Schloss Birlinghoven, Sankt-Augustin, D-53754 Germany
E-mail: savinov@gmd.de, http://borneo.gmd.de/~savinov/

**Abstract.** We describe the problem of mining possibilistic set-valued rules in large relational tables containing categorical attributes taking a finite number of values. An example of such a rule might be "IF HOUSEHOLDSIZE={Two OR Tree} AND OCCUPATION={Professional OR Clerical} THEN PAYMENT_METHOD={CashCheck (Max=249) OR DebitCard (Max=175)}. The table semantics is supposed to be represented by a frequency distribution, which is interpreted with the help of minimum and maximum operations as a possibility distribution over the corresponding finite multidimensional space. This distribution is approximated by a number of possibilistic prime disjunctions, which represent the strongest patterns. We present an original formal framework generalising the conventional boolean approach on the case of (i) finite-valued variables and (ii) continuos-valued semantics, and propose a new algorithm, called Optimist, for the computationally difficult dual transformation which generates all the strongest prime disjunctions (possibilistic patterns) given a table of data. The algorithm consists of generation, absorption and filtration parts. The generated prime disjunctions can then be used to build rules or for prediction purposes.

## 1. Introduction

One specific data analysis task consists in discovering hidden patterns that characterise the problem domain behaviour and then representing them in the form of rules, which can be used either for description or for prediction purposes. The analysed database consists of a number of records, each of which is a sequence of attribute values. In the case where variables in condition and conclusion may take only one value we obtain well known association rules which describe the dependencies (associations) among individual values rather than among sets of the values. If the variables in rules may be constrained by any subset of possible values then we obtain so called possibilistic set-valued rules, e.g.:

$$\text{IF } x_1 = \{a_{13}, a_{14}\} \text{ AND } x_2 = \{a_{21}, a_{27}\} \text{ THEN } x_3 = \{a_{33} : p_{33}, a_{36} : p_{36}\}$$

where $a_{ij}$ are values of the $i$-th variable and $p_{ij}$ are degrees of possibility expressed as maximal frequencies of the corresponding values within the interval. This rule means that if $x_1$ is either $a_{13}$ or $a_{14}$, and $x_2$ is either $a_{21}$ or $a_{27}$, then $x_3$ is either

$a_{33}$ or $a_{36}$ with the possibilities $p_{33}$ and $p_{36}$, respectively, while the rest of the values such as $a_{31}$ are impossible within this condition interval (the frequency is 0).

In the paper we consider the problem of mining set-valued rules for the case where all variables may take only a finite number of values, and the possibilistic semantics is represented by a frequency distribution (the number of observations belonging to each point). The problem is that the number of all possible conjunctive intervals of the multidimensional space is extremely large. However, most of them are not interesting since the projection of the restricted distribution onto all variables does not have much information (it is highly homogeneous). Thus, informally, the more general the rule condition (the wider the selected interval), and the narrower the conclusion are (the closer the conclusion distribution to the singular form), the more interesting and informative the rule is. To find such maximally general in condition and specific in conclusion rules we use an approach [1] according to which any multidimensional possibility distribution can be formally represented (Fig. 1) by a set of possibilistic disjunctions combined with the connective AND (possibilistic CNF). The disjunction (possibilistic pattern) is made up of several one-dimensional possibility distributions (propositions) over the values of individual variables combined with the connective OR (interpreted as maximum). Note that we use an original definition of possibilistic disjunction, which generalises the conventional boolean analogue in two directions: (i) the variables are finite-valued [2] (instead of only 2-valued), and (ii) the semantics is continuous-valued [1] (instead of only 0 and 1). Particularly, this feature distinguishes our approach from other methods including Boolean reasonging and rough sets. The strongest disjunctions, called primes, are used to form the optimal and the most interesting rules, i.e., possibilistic prime disjunctions allow us to reach both goals when generating rules — maximal generality of condition and maximal specificity of conclusion. In contrast to the previous version [3] requiring all records to be in memory, the Optimist is based on the explicit formula [4] of transformation from possibilistic DNF representing data into CNF consisting of prime disjunctions (knowledge). The advantage is that all prime disjunctions are built for one pass through the record set by updating the current set of prime disjunctions each time new record is processed. The algorithm efficiently solves the problem of computational complexity by filtering out too specific disjunctions interpreted as noise or exceptions and generating only the most informative of them. Once the patterns have been found they can be easily written as rules.

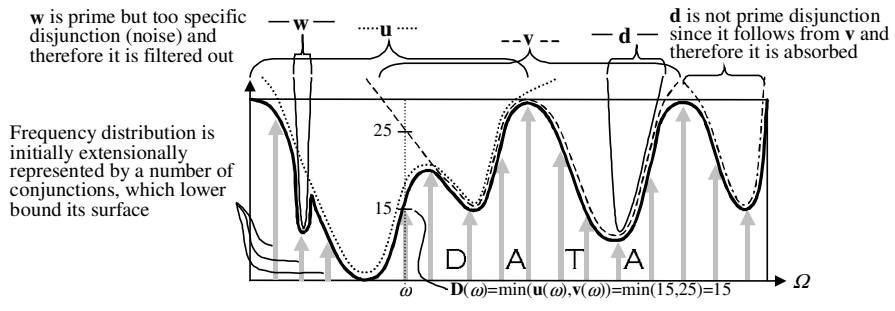


**Fig. 1.** The data semantics is approximated (upper bound) by prime disjunctions (patterns).

## 2. Data and Knowledge Representation

Let some problem domain on the syntactic level be described by a finite number of *variables* or *attributes* $x_1, x_2, \ldots, x_n$ each of which takes a finite number of *values* and corresponds to one column of data table: $x_i \in A_i = \{a_{i1}, a_{i2}, \ldots, a_{in_i}\}$, $i = 1,2,\ldots,n$, where $n_i$ is the number of values of *i*-th variable and $A_i$ is its set of values. The *state space* or the *universe of discourse* is defined as the Cartesian product of all sets of the values: $\Omega = A_1 \times A_2 \times \ldots \times A_n$. The universe of discourse is a finite set with the multidimensional structure. Each syntactic object (state) from the universe of discourse is represented by a combination of values of all variables: $\omega = \langle x_1, x_2, \ldots, x_n \rangle \in \Omega$. The number of such objects is equal to the power of the universe of discourse: $|\Omega| = n_1 \times n_2 \times \ldots \times n_n$.

Formally the problem domain *semantics* is represented by a frequency distribution over the state space which assigns the number of occurrences to each combination of values. Then 0 is interpreted as the absolute impossibility of the state while all positive numbers are interpreted as various degrees of possibility. We do not map this distribution into the interval [0,1] since for rule induction it is simpler to work directly with frequencies. The semantics will be represented by a special technique called the method of sectioned vectors and matrixes [1–3]. Each construction of this mechanism along with interpretation rules imposes constraints of certain form on possible combinations of attribute values. The sectioned constructions are written in bold font with the two lower indexes corresponding to the number of variable and to the number of value, respectively.

The *component* $\mathbf{u}_{ij}$ of the sectioned vector $\mathbf{u}$ is a natural number assigned to *j*-th value of *i*-th variable. The *section* $\mathbf{u}_i$ of the sectioned vector $\mathbf{u}$ is an ordered sequence of $n_i$ components assigned to *i*-th variable and representing some distribution over all values of one variable. The sectioned *vector* $\mathbf{u}$ is an ordered sequence of *n* sections for all variables. The total number of components in sectioned vector is equal to $n_1 + n_2 + \ldots + n_n$. The sectioned *matrix* consists of a number of sectioned vectors written as its lines. For example, the construction $\mathbf{u} = 01.567.0090$ or $\mathbf{u} = \{0,1\}.\{5,6,7\}.\{0,0,9,0\}$ is a sectioned vector written in different ways (with sections separated by dots) where $\mathbf{u}_1 = \{0,1\}$, $\mathbf{u}_2 = \{5,6,7\}$, $\mathbf{u}_{11} = 0$ and so on.

There are two interpretations of sectioned vectors: as conjunction and as disjunction. If the sectioned vector $\mathbf{d}$ is interpreted as *disjunction* then it defines the distribution, which is equal to the maximum of the vector components corresponding to the point coordinates:

$$\mathbf{d}(\omega) = \mathbf{d}(\langle x_1, x_2, \ldots, x_n \rangle) = \mathbf{d}_1(x_1) \vee \mathbf{d}_2(x_2) \vee \ldots \vee \mathbf{d}_n(x_n) = \max_{i=1,\ldots,n} \mathbf{d}_i(x_i)$$

(The minimum is taken among $n$ components — one from each section.) The *conjunction* is interpreted in the dual way by means of the minimum operation.

Sectioned matrixes can be interpreted as DNF or CNF. If the matrix $\mathbf{K}$ is interpreted as DNF then its sectioned vector-lines are combined with the connective $\vee$ and interpreted as conjunctions (disjunction of conjunctions). In the dual way, if

the matrix $\mathbf{D}$ is interpreted as CNF then its sectioned vector-lines are combined with the connective $\vee$ and interpreted as disjunctions (conjunction of disjunctions).

The data can be easily represented in the form of DNF so that each conjunction represents one record along with the number of its occurrence in the data set. The conjunction corresponding to one record consists of all 0's except for one component in each section, which is equal to the number of record occurrences.

One distribution is said to be a *consequence* of another if its values in all points of the universe of discourse are greater or equal to the values of the second distribution. We will say also that the first distribution *covers* the second one. The operation of *elementary induction* consists in increasing one component of a disjunction so that it becomes weaker. The disjunction is referred to as *prime* one if it is a consequence of the source distribution but is not a consequence of any other distribution except of itself. The prime disjunctions are considered as possibilistic *patterns* expressing dependencies among attributes by imposing the strongest constraints on the possible combinations of values. Thus formally the problem of finding dependencies is reduced to the problem of generating possibilistic prime disjunctions.

## 3. Generation, Absorption and Filtration of Disjunctions

To add the conjunction $\mathbf{k}$ (record) to the matrix of CNF $\mathbf{D}$ (current knowledge) it is necessary to add it to all $m$ disjunctions of the matrix:

$$\mathbf{k} \vee \mathbf{D} = \mathbf{k} \vee (\mathbf{d}^1 \wedge \mathbf{d}^2 \wedge \ldots \wedge \mathbf{d}^m) = (\mathbf{k} \vee \mathbf{d}^1) \wedge (\mathbf{k} \vee \mathbf{d}^2) \wedge \ldots \wedge (\mathbf{k} \vee \mathbf{d}^m)$$

Addition of conjunction to disjunction is carried out by the formula:

$$\mathbf{k} \vee \mathbf{d} = (\mathbf{k}_1 \vee \mathbf{d}) \wedge (\mathbf{k}_2 \vee \mathbf{d}) \wedge \ldots \wedge (\mathbf{k}_n \vee \mathbf{d}) =$$

$$\begin{vmatrix} \mathbf{k}_1 \vee (\mathbf{d}_1 \vee \mathbf{d}_2 \vee \ldots \vee \mathbf{d}_n) \\ \mathbf{k}_2 \vee (\mathbf{d}_1 \vee \mathbf{d}_2 \vee \ldots \vee \mathbf{d}_n) \\ \mathbf{k}_n \vee (\mathbf{d}_1 \vee \mathbf{d}_2 \vee \ldots \vee \mathbf{d}_n) \end{vmatrix} = \begin{vmatrix} \mathbf{k}_1 \vee \mathbf{d}_1 & \vee & \mathbf{d}_2 & \vee \ldots \vee & \mathbf{d}_n \\ \mathbf{d}_1 & \vee & \mathbf{k}_2 \vee \mathbf{d}_2 & \vee \ldots \vee & \mathbf{d}_n \\ \mathbf{d}_1 & \vee & \mathbf{d}_2 & \vee \ldots \vee & \mathbf{k}_n \vee \mathbf{d}_n \end{vmatrix}$$

In general case $n$ new disjunctions are generated from one source disjunction by applying the elementary induction, i.e., by increasing one component. For example, according to this formula addition of the conjunction $\mathbf{k}$ = 05.005.0005 to the disjunction $\mathbf{d}$ = 01.070.0102 results in three new disjunctions (increased components are underlined): 0<u>5</u>.070.0102, 01.07<u>5</u>.0102, and 01.070.010<u>5</u>.

If the elementary induction does not change one of the disjunctions then it means that the source disjunction already covers the conjunction. In this case the disjunction can be simply copied to the new matrix with no modifications. Thus the whole set of new disjunctions can be divided into two subsets: modified and non-modified.

As new disjunctions are generated and added to the new matrix the *absorption* procedure should be carried out to remove the lines which are not prime and follow from others, e.g., $\mathbf{d}$ in Fig. 1. In general, each new disjunction can either be absorbed itself or absorb other lines. Thus the comparison of lines has to be fulfilled in both directions. To check for the consequence relation between two disjunctions we have to reduce them [1] and then compare all their components. Let us formulate properties, which significantly simplify the absorption process.

Property 1. The disjunctions, which cover the current conjunction and hence are not modified, cannot be absorbed by any other disjunction.

This property follows from the fact that the matrix of disjunctions is always maintained in the state where it contains only prime disjunctions, which do not absorb each other.

Let us suppose that $\mathbf{u}$ is non-modified disjunction while $\mathbf{v}$ is modified on the component $\mathbf{v}_{rs}$, and $\mathbf{v}'_{rs}$ is old value of modified component ($\mathbf{u}_{ij} = \mathbf{u}'_{ij}$ since $\mathbf{u}$ is not modified). Then the following property takes place.

Property 2. If $\mathbf{u}_{rs} \le \mathbf{v}'_{rs}$ then $\mathbf{v}$ does not follow from $\mathbf{u}$. (This property is valid only if the constant [1] of $\mathbf{v}$ has not been changed.

To use this property each line has to store information on the old value $\mathbf{v}'_{rs}$ of modified component and its number ($r$ and $s$). These properties are valuable since frequently they allow us to say that one line is not a consequence of another by comparing only one pair of components.

Property 3. If the sum of components in $\mathbf{v}$ or in any of its sections $\mathbf{v}_i$ is less then the corresponding sum in the disjunction $\mathbf{u}$ then $\mathbf{v}$ does not follow from $\mathbf{u}$.

To use this property we have to maintain the sums of the vector and section components in the corresponding headers. If all these necessary conditions are satisfied then we have to carry out a component-wise comparison of two vectors in the loop consisting of $n_1 + n_2 + \ldots + n_n$ steps.

To cope with complexity problem and to generate only interesting rules the algorithm has been modified so that the number of lines in the matrix of prime disjunctions is limited by a special user-defined parameter while the lines are ordered by a criterion of interestingness. Before a new disjunction is to be generated we calculate its criterion value (the degree of interestingness) which is compared with that of the last line of the matrix. If the new disjunction does not go into the matrix (e.g., **w** in Fig. 1), it is simply not generated. Otherwise, if it is interesting enough, it is first generated, then checked for absorption, and finally inserted into the corresponding position in the matrix (the last line is removed).

The Optimist algorithm uses the criterion of interestingness in the form of the impossibility interval size. Informally, the more points of the distribution have smaller values, the more general and stronger the corresponding disjunction is. Formally the following formula is used to calculate the degree of interestingness:

$$H = \frac{1}{n_1}\sum_{j=1}^{n_1}\mathbf{d}_{1j} + \frac{1}{n_2}\sum_{j=1}^{n_2}\mathbf{d}_{2j} + \ldots + \frac{1}{n_n}\sum_{j=1}^{n_n}\mathbf{d}_{nj}$$

according to which $H$ is equal to the weighted sum of components, and the less this value, the stronger the disjunction. For example, changing one component from 0 to 1 in two-valued section is equivalent to changing three components from 0 to 1 in six-valued section. Generally, each attribute or even each attribute value may have their own user-defined weights, which influence the direction of induction and reflect their informative importance or subjective interestingness for the user. This mechanism provides the capability of more flexible control over the rule induction process.

The set of generated prime disjunctions is an approximate semantic equivalent of the data. Once they have been generated they can be used for prediction purposes or

to build rules. The patterns are rewritten in the form of rules in the conventional way by negating the propositions (sections) which should be in the condition and thus obtaining an implication. The only problem here is that we obtain conditions with possibilistic weights while it is more preferable to have crisp conditions (without uncertainty). The most straightforward way to do it consists in negating the condition section $\mathbf{d}_i$ as follows: $\mathbf{d}_{ij} = \mathbf{d}_{\max}$, if $\mathbf{d}_{ij} \le \mathbf{d}_{\min}$, and $\mathbf{d}_{ij} = \mathbf{d}_{\min}$, otherwise, where $\mathbf{d}_{\min}$ and $\mathbf{d}_{\max}$ are minimal and maximal components of the disjunction, respectively ($\mathbf{d}_{\max}$ is usually mapped into 1 within [0,1] interval). For example, the pattern $\mathbf{d} = \{0,8\} \vee \{0,6,0\} \vee \{0,2,9,5\}$ with $\mathbf{d}_{\min} = 0$ and $\mathbf{d}_{\max} = 9$ can be transformed into the implication $\{9,0\} \wedge \{9,0,9\} \rightarrow \{0,2,9,5\}$ which is interpreted as the possibilistic rule IF $x_1 = \{a_{11}\}$ AND $x_2 = \{a_{21}, a_{23}\}$ THEN $x_3 = \{a_{31} : 0, a_{32} : 2, a_{33} : 9, a_{34} : 5\}$ This method can be generalised by applying any user-defined value instead of $\mathbf{d}_{\min}$. In addition, the values in conclusion can be easily weighted by their frequencies (the sum of occurrences within the condition interval) or necessity degrees (the minimal number of occurrences), e.g., $x_3 = \{a_{32} : (Min = 0)\,(Max = 2)\,(Sum = 4), a_{33} : \ldots\}$.

## 4. Conclusion

The described approach to mining possibilistic set-valued rules has the following characteristic features: (i) it is based on the original formal framework generalising boolean approach on the case of finite-valued attributes and continuous-valued semantics, (ii) the notion of prime disjunction as a pattern allows us to reach optimality of rules (maximal generality of condition and specificity of conclusion), (iii) it guarantees finding only the strongest patterns (too weak ones are filtered out), (iv) all attributes as well as all rules have equal rights, particularly, we do not need the target attribute and all rules are interpreted independently, (v) the rules are generated for one pass, (vi) the patterns can be easily used for prediction as well as for other tasks since they approximately represent the data semantics in an intensional form, (vii) a minus is a large number of generated rules especially for dense distributions what can be overcome by a more sophisticated filtration and search.

## References


1. A.A. Savinov. Fuzzy Multi-dimensional Analysis and Resolution Operation. Computer Sci. J. of Moldova **6**(3), 252–285, 1998.
2. A.D. Zakrevsky, Yu.N. Pechersky and F.V. Frolov. DIES — Expert System for Diagnosis of Technical Objects. Preprint of the Institute of Mathematics and CC, Academy of Sciences of Moldova, Kishinev, 1988 (in Russian).
3. A. Savinov. Application of multi-dimensional fuzzy analysis to decision making. In: Advances in Soft Computing — Engineering Design and Manufacturing. R. Roy, T. Furuhashi and P.K. Chawdhry (eds.), Springer-Verlag London, 1999.
4. A. Savinov. Forming Knowledge by Examples in Fuzzy Finite Predicates. Proc. conf. “Hybrid Intellectual Systems”, Rostov-na-Donu—Terskol, 177–179, 1991 (in Russian).