International Journal of Information, Control and Computer Sciences
ISSN: 2517-9942
Vol:1, No:4, 2007

Generator of Hypotheses – an Approach of Data Mining Based on Monotone Systems Theory

Rein Kuusik, and Grete Lind

***Abstract*—** Generator of hypotheses is a new method for data mining. It makes possible to classify the source data automatically and produces a particular enumeration of patterns. *Pattern* is an expression (in a certain language) describing facts in a subset of facts. The goal is to describe the source data via patterns and/or IF...THEN rules. Used evaluation criteria are deterministic (not probabilistic). The search results are trees – form that is easy to comprehend and interpret.

Generator of hypotheses uses very effective algorithm based on the theory of monotone systems (MS) named MONSA (MONotone System Algorithm).

***Keywords*—** data mining, monotone systems, pattern, rule.

I. Introduction

ACCORDING to [3] data mining (DM) is a part of the process called knowledge discovery in databases (KDD), which consists of particular data mining algorithms and produces a particular enumeration of patterns.

Pattern is an expression (in a certain language) describing facts in a subset of facts.

Data mining has two high-level primary goals:

- *Prediction* involves using some variables or fields in the database to predict unknown or future values of other variables of interest.
- *Description* focuses on finding human-interpretable patterns describing the data.

In the context of KDD, description tends to be more important than prediction. These goals are achieved by using primary DM tasks:

- Classification
- Regression
- Clustering
- Summarization
- Dependency modeling
- Change and deviation detection

On the basis of these DM tasks several DM solutions have been developed [2], [4]. In this paper we describe a new hierarchical clustering algorithm called MONSA and on the basis of it a sense of a new DM method called Hypotheses Generator.

Manuscript received September 21, 2004. This work was supported in part by the Estonian Information Technology Foundation under Grant 03-03-00-03.

R. Kuusik is with the Department of Informatics, Tallinn University of Technology, Raja 15, 12618 Tallinn, Estonia (corresponding author to provide phone: +372 620 2301; Fax: +372 620 2305; e-mail: kuusik@cc.ttu.ee).

G. Lind is with the Department of Informatics, Tallinn University of Technology, Raja 15, 12618 Tallinn, Estonia (e-mail: grete@cc.ttu.ee).

II. The Clustering Task

Clustering is a common descriptive task identifying a finite set of categories or clusters to describe the data. The categories may be mutually exclusive and exhaustive, or consist of a richer representation such as hierarchical or overlapping categories [3].

The set of objects X (of object-attribute type) is given. One of possible techniques for extracting clusters is finding intersections.

TABLE I
EXAMPLE. X(2,3)

Object \ Attribute	*A1*	*A2*	*A3*
O1	1	2	2
O2	1	1	2
Intersection	1	*	2

$IntSec_X$ = 1.1 AND 3.2

Intersection of two sets is set of elements, which belong to both sets, simultaneously.

Clustering is realised via using intersections. An intersection describes a pattern. All objects meeting the description form a cluster.

The purpose is in case of need to find all existing intersections of attributes in set X or intersections with certain properties (for example: by frequency, by length of intersection, etc).

One way to do it is to generate all theoretical value combinations and check their existence (at once or later on), but this approach involves a lot of work done for nothing. For hierarchical clustering several algorithms can be used.

A. Drawbacks of classical hierarchical clustering algorithms

The classical algorithms based on intersections have the following drawbacks:

- They find repeating intersections
- They find empty intersections
- The order of finding intersections is spontaneous, it depends on the initial order of objects. It makes optimizing difficult.

III. Algorithm MONSA

Proposed algorithm used for finding intersections is based on the theory of monotone systems [10] and it is called MONSA (MONotone System Algorithm).

MONSA does not have the drawbacks the classical hierarchical clustering algorithms based on intersections have.

It can be used for discovering of patterns and associations. It enables easily to construct the rules in the form IF $IntSec_{X_t}$ THEN $IntSec_{X_{t+1}}$.

The depth-first search is used, which makes possible to hold in memory only one branch of the tree at the same time. That is an important advantage compared with algorithms, which hold the whole classification tree in the memory together.

It finds only really existing intersections without additional checking. The order they are found doesn't depend on the order of objects. Also, it works with larger set of discrete values.

For finding intersections the frequency table is used. Element's frequency is the number of times the element (i.e. attribute with its certain value) occurs in given data. The frequency table consists of counts of occurrences of elements in the set or subset of data. Example data are given in Table II and corresponding frequencies in Table III. The results of using MONSA based on those data are shown in III.D "Example".

TABLE II
EXAMPLE. X(8, 4)

Object \ Attribute	*A1*	*A2*	*A3*	*A4*
O1	2	1	1	1
O2	1	1	1	1
O3	2	3	1	2
O4	2	2	1	2
O5	2	3	2	1
O6	1	3	1	2
O7	1	3	2	1
O8	2	1	2	1

TABLE III
THE CORRESPONDING FREQUENCY TABLE

Value \ Attribute	*A1*	*A2*	*A3*	*A4*
1	3	3	5	5
2	5	1	3	3
3	0	4	0	0

MONSA uses frequency tables and special techniques to prevent repetitions (activities Eliminate1 and Eliminate2 in III.C „Description of MONSA").

A. What is a monotone system?

Definition 1:

Let a finite discrete **set** X and function π_x on it which maps to each **element** $\alpha \in X$ a certain nonnegative number $\pi_x(\alpha)$, be given.

The function π_x is called a **weight function** if it is defined on any subset $X' \subseteq X$; the number $\pi_x(\alpha)$ is called a **weight** of element α on X'.

Definition 2:

A set X with a weight function π_x is called a **system** and is denoted by $S=(X, \pi_x)$.

Definition 3:

The system $S'=(X', \pi_{x'})$ where $X' \subseteq X$ is called a **subsystem** of the system $S=(X, \pi_x)$.

Definition 4:

The system $S=(X, \pi_x)$ is called **monotone** if in the case of any $\alpha \in X' \setminus \{b\}$, $b \in X$:

$\pi_{x' \setminus \{b\}}(\alpha) \le \pi_{x'}(\alpha)$ where X' is any subset of X.

B. How to create and use a monotone system

Let be given data set X(N,M), where N is the number of objects (i=1,…,N) and M is the number of attributes (j=1,…,M). Element α can be X_{ij}, row i, column j or any subtable of X.

To use the method of monotone systems we have to fulfil two conditions:

1) There has to be a weight function $\pi x(\alpha)$ which will give a measure of influence for every element α of the monotone system on X;
2) Certain activities (adding or removing) can be applied to the elements. There have to be rules f to recompute the weight of the system elements after used activities. Weights can be changed only to one direction (increasing or decreasing).

These conditions give a lot of freedom (to user) to choose the weight functions and rules of weight change in the system. The only constraint we have to keep in mind is that after eliminating all elements α from the system X the final weights of $\alpha \in X$ must be equal to zero. In our case:

1) A suitable weight function is object's frequency in a (concerned) system. In case of tables of object-attribute type we weigh the attribute's value and the weight is a number of objects having that certain value.
2) Rules for recomputing the weights:

- Choose the element(s) of interest.
- Extract the objects having the(se) element(s) from the concerned set. So the set of objects under consideration can only decrease.
- For the rest of objects calculate new weights using the same weight function. If there are no objects with given elements then the weight is zero.

C. Description of MONSA

MONSA finds all existing intersections in given set of objects X(N,M), where N is the number of objects, M is the number of attributes and each attribute j has an integer value $h_j=0,1,2,\ldots,K-1$.

By essence MONSA is a recursive algorithm. Here its backtracking version is presented.

In this algorithm the following denotations are used:

t	the number of the step (or level) of recursion
FT_t	frequency table for a set X_t
$IntSec_t$	vector of elements over set X_t

Init	activity for initial evaluation
Eliminate1(t+1)	activity, that prohibits arbitrary output repetition of already separated intersection on level (t+1) – bringing zeroes down (from FT_t to FT_{t+1})
Eliminate2(t+1)	activity, that does not allow the output of the separated intersection on the same (current) level t+1 and on steps t, t-1, ..., 0 – bringing zeroes up (from FT_{t+1} to FT_t)

Algorithm MONSA

```
Init
t=0, IntSec_0={}
To find a table of frequencies FT_0 for all attributes in X_0
DO WHILE there exists FT_s#Ø in {FT_s}, s≤t
  FOR an element h_f∈FT_t with frequency V=max FT_t(h_f)#0 DO
    To separate submatrix X_{t+1}⊂X_t such that X_{t+1}={Xij∈X_t | X.f=h_f}
    To find a table of frequencies on X_{t+1}
    IF there exist on X_{t+1} h_u, 1≤u≤M such, that [h_u∈IntSec_{t+1} AND FT_{t+1}(h_u)=0 AND frequency of h_u in X_{t+1}=V]
      THEN goto BACK
    ELSE
      Eliminate1(t+1)
      To add elements j with FT_{t+1}(j)=V into vector IntSec_{t+1}
      Eliminate2(t+1)
      IF there exist attributes to analyse THEN t=t+1
      Output of IntSec_t
    ENDIF
  ENDFOR
  BACK: t=t-1
  IntSec_{t+1}←IntSec_t
ENDDO
All rules are found
END: end of algorithm
```

The main idea of the work of MONSA is simple:

- subset Xt+1⊂Xt of objects with certain properties is being separated;
- then intersection over this subset Xt+1 (IntSecXt+1) is being found.

To find an intersection over the set X_{t+1}, the frequency table FT_t of the set X_t can be used effectively. Maximal frequency $MAX=|X_{t+1}|$ of a certain element $Xij \in X_{t+1}$ in frequency table FT_{t+1} is defined by the frequency in FT_t of intersection which was the base to separate X_{t+1}. Consequently, all elements $Xij(=h_j) \in X_{t+1}$ the frequencies of which equal MAX, appear simultaneously in all objects of X_{t+1} and define an intersection over the set X_{t+1}.

TABLE IV
EXAMPLE. X(2,3)

Object \ Attribute	*1*	*2*	*3*
1	1	2	2
2	1	1	2

TABLE V
FREQUENCY TABLE

Value \ Attribute	*1*	*2*	*3*
1	**2**	1	0
2	0	1	**2**

Number of objects N = 2 => $IntSec_X$ = 1.1 AND 3.2

Thereupon set X_{t+1} is being eliminated at the level t+1 from further analysis and the whole procedure is being repeated once again until all subsets X_{t+1}, t=0,1,...,M-1 are found.

Elimination of set X_{t+1} guarantees non repetition of set X_{t+1} separation.

D. Example

For objects (from Table II) having element A4.1 MONSA finds (with minimal frequency allowed = 2) five intersections and patterns as descriptions of intersections:

***1 A4.1=5
*1*1 A4.1&A2.1=3
**21 A4.1&A3.2=3
*321 A4.1&A3.2&A2.3=2
1**1 A4.1&A1.1=3

(After "=" the frequency of intersection is shown.)

- 5 different
- no empty
- no repeating

There is no special effort to check the intersection's uniqueness during the extraction process. Repetitions are prevented just by using nullifying techniques.

E. Complexity of MONSA

It is proved that if a finite discrete data matrix X(N,M) is given, where $N=K^M$, then the complexity of algorithm MONSA to find all $(K+1)^M$ patterns as existing value combinations is $O(N^2)$ operations [5].

By our estimation in practice the upper bound of the number of intersections (with minimal frequency allowed = 1) is

$$L_{UP} \approx N(1+1/K)^M , \quad (1)$$

but usually it is lesser.

The precise formula for number of intersections is as follows:

$$L = \sum_{f=1}^{F} \sum_{p=1}^{(M*K-t)} N_p , \quad (2)$$

where F is the number of formatted frequency tables on set X, t is the number of empty cells in the frequency table FT_f, N_p is the absolute number in a cell of the frequency table (frequency of attribute value).

IV. Generator of Hypotheses

A. What is Generator of Hypotheses?

Generator of hypotheses (GH) is a method for data mining which main aim is mining for patterns and association rules

[5]. It solves the task of hierarchical clustering; also it makes possible to classify the source data automatically. The goal is to describe the source data. Used evaluation criteria are deterministic (not probabilistic). The association rules it produces are represented as trees, which are easy to comprehend and interpret.

The method is derived from analysis of determinations (AD) (which can be considered as semi-manual data mining) [1] while automating it. In addition, the original method has been expanded. Description of AD is left out from this paper. Only AD drawbacks are listed here:

- Large amount of manual work
- Searching for significant associations by testing – method doesn't determine the direction for subsequent search
- The realization algorithm uses lot of memory

GH does not have these drawbacks.

B. Properties of Generator of Hypotheses

- The amount of results can be controlled via pruning
- Several pruning criteria can be used
- Large datasets can be treated
- The subset of objects and attributes to analyze can be extracted (from the whole data set)
- The results are found as association rules and patterns
- The output has the form of trees

C. Output of Generator of Hypotheses

By depth-first search (from root to leaves) GH forms a hierarchical grouping tree. A fragment of such tree is given below.

```
(5)      0.600(3)  0.333(1)
Class .1=>Hair  .1->Height.1
                   0.333(1)
                   ->Eyes  .2
         0.600(3)  0.667(2)  0.500(1)
         =>Eyes  .2->Hair  .3->Height.1
         0.400(2)
         =>Height.1
```

Used data (given in Table II, see 3 "Algorithm MONSA") are taken from [11], meanings of used values are given in Table VI.

TABLE VI
MEANINGS OF USED ATTRIBUTES' VALUES

Attribute	Attribute's value	Value's meaning
Class (A4)	1	positive
Hair (A2)	1	dark
Hair (A2)	3	blond
Height (A1)	1	short
Eyes (A3)	2	brown

The numbers above node show node's absolute frequency (in parentheses) and node's relative (to previous level) frequency (before parentheses).

Absolute frequency of node t shows how many objects have certain attribute with certain value (among objects having properties (i.e. certain attributes with certain values) of all previous levels t-1,…,1). Relative frequency is a ratio A/B, where A is the absolute frequency of node t and B is the absolute frequency of node t-1. For the first level the relative frequency is not calculated.

V. Conclusion

In the article we have presented the algorithm MONSA for clustering based on monotone systems theory. The algorithm is very efficient compared to classical algorithms based on intersections because every intersection is found only once and no empty intersections are found.

A new method for data mining called Generator of Hypothesis based on MONSA has been developed. GH has the following properties:

- For every pattern its frequency (i.e. how many and which objects contain it) is known at the moment it is found, also other parameters based on frequency can be calculated
- GH guarantees immediate and simple output of rules in the form IF=>THEN
- GH enables larger set of discrete values (not only binary)
- GH enables to use several pruning techniques
- The result is presented in form of trees
- GH enables to treat large datasets
- Enables sampling

This approach has been used first in the field of data mining [8], [13]. Algorithm MONSA with several modifications has been used also in other fields like Graphs Theory [6], [7], Machine Learning [12], Data Analysis [9], [14] etc.

References

[1] S. Chesnokov, *Analysis of determinations for socio-economic data.* Nauka, Moscow (in Russian), 1982.
[2] M. H. Dunham, *Data Mining: Introductory and Advanced Topics.* Prentice Hall, 2002.
[3] U. M. Fayyad, G. Piatetsky-Shapiro, P. Smyth, "From Data Mining to Knowledge Discovery: An Overview," *U. M. Fayyad, G. Piatetsky-Shapiro, P. Smyth, R. Uthurusamy, Advances in Knowledge Discovery and Data Mining*, AAAI Press/ The MIT Press, 1996, pp. 1-36.
[4] T. Hastie, R. Tibshirani, J. H. Friedman, *The Elements of Statistical Learning: Data Mining, Inference, and Prediction,* Springer Series in Statistics, Springer Verlag, 2001.
[5] R. Kuusik, "The Super-Fast Algorithm of Hierarchical Clustering and the Theory of Monotone Systems," *Transactions of Tallinn Technical University*, 734, 1993, pp. 37-62.
[6] R. Kuusik, "Extracting of all maximal cliques: monotone system approach," *Proceedings of the Estonian Academy of Sciences. Engineering*, 1, 1995, pp. 113-138.
[7] R. Kuusik and L. Võhandu, "Cliques and algorithms with a hidden parallelity," *Transactions of Tallinn Technical University*, 734, 1993, pp. 63-74.
[8] G. Lind, "Method for Data Mining – Generator of Hypotheses," in *Databases and Information Systems. Proceedings of the 4th International Baltic Workshop*, Vol. 2, Vilnius, 2000, pp. 304-305.
[9] G. Lind, "Monotone Systems in Data Mining," in *Databases and Information Systems. Proceedings of the Fifth International Baltic Conference*, Vol. 2, Tallinn, 2002, pp. 249-254.

[10] I. Mullat, "Extremal monotone systems," *Automation and Remote Control*, 1976, 5, pp. 130-139; 8, pp. 169-178 (in Russian).
[11] J. R. Quinlan, "Learning efficient classification procedures and their application to chess and games," *J. G. Carbonell, R. S. Michalski, T. M. Mitchell (Eds.), Machine Learning. An Artificial Intelligence Approach.* Springer-Verlag, 1984.
[12] P. Roosmann, "A new method for learning from examples," *Computers and Data Processing*, 8, 1991, pp. 31-51 (in Estonian).
[13] A. Udras, and R. Kuusik, "Evaluation of probability of coronary artery stenoses in patients with ischemic heart disease by "Hypothesis generator" technique," *Kardiologiya*, Vol. 34, 7, 1994, pp. 93-97 (in Russian).
[14] L. Võhandu, R. Kuusik, and P. Roosmann, "Database mining and GIS," in *Proceedings Conf. GIS Baltic Sea States '95*, Tallinn, 1997, pp. 159-163.