Nonlinear Analysis: Modelling and Control, 2001, v. 6, No. 2, 29-41

Clustering through Decision Tree Construction in Geology

A. Juozapavičius, V. Rapševičius
Faculty of Mathematics and Computer Science
Vilnius University, Naugarduko 24, 2006, Vilnius, Lithuania
algimantas.juozapavicius@maf.vu.lt, valdo@lgt.lt

Received: 22.10.2001
Accepted: 31.10.2001

Abstract

The article presents a tool to analyze the application of efficient algorithms of data mining, namely hierarchical clustering algorithms to be used in the analysis of geological data. It introduces a description of hierarchical clustering principles and methods for learning dependencies from geological data. The authors are using statistical formulation of algorithms to represent the most natural framework for learning from data. The geological data come from mining holes, and describe the structure of sedimental layers of vertical section of geological body. The analysis of such data is intended to give a basis for uniform description of lithological characteristics, and for the identification of them via formal methods.

Keywords: data mining, hierarchical clustering algorithms, lithological characteristics, analysis of geological data.

1 Introduction

In recent years there has been an explosive growth of methods for learning (or estimating dependencies) from data. This growth was in width and depth one, and was caused by proliferation of computers (especially of low-cost ones, implementing learning methods in software), low-cost sensors for data detection, database technology (for collecting and storing data), and highly computer-literate application experts [1]. All such methods define content of data mining.

A learning method is an algorithm that estimates an unknown mapping (dependency) between system's inputs and outputs from the data available, namely from known samples. Once such a dependency has been accurately estimated it can be formulated in conceptual terms and used for the prediction of future system outputs. Such a dependency describes also how system's outputs depend upon the known input values. Among the most popular data mining methods are hierarchical clustering ones. Nevertheless such data mining methods depend very much on the area of application.

This article is intended to analyze most efficient algorithms used in data clustering and provide a description of hierarchical clustering principles and methods for learning dependencies from geological data. The authors are using statistical formulation of algorithms to represent the most natural framework for learning from data.

2 Description of Geological Data

Most of the geological data describe a rock – a sole lithological characteristic of the geological body, to be considered as layer [2]. Lithological layer is a geological body used to be described by a set of common lithological characteristics – names of rock solid. Lithology as a science focuses on sedimentary rocks; and the science of petrology – on magmatic and methamorphic rocks. To avoid complexities, these two concepts will be unified and referenced as lithology.

The lithological bodies do not match in common another important geological body – a stratigraphic body. Stratigraphic body is a geological layer that has a simultaneous formation (in time). These concepts are presented in fig. 1: four lithological bodies – layers of clay, limestone, second layer of clay and the layer of marl have formed in Silurian period. The clay layer and the start of Silurian period match each other, but the end of Silurian period cuts the layer of marl.

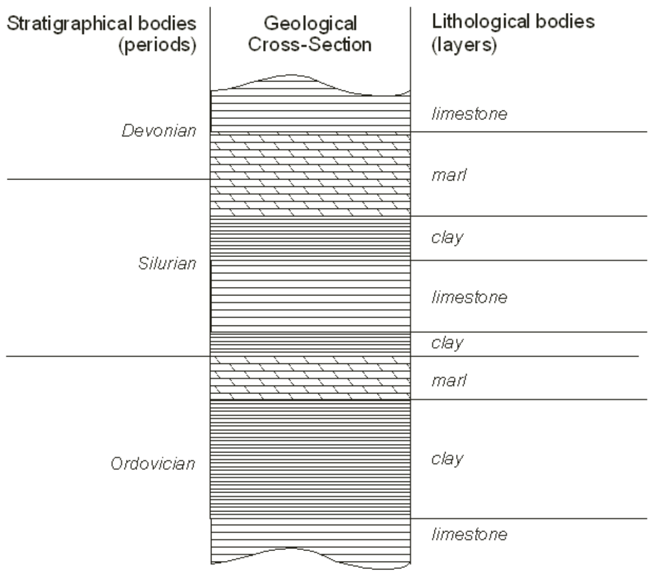

Fig. 1. Geological bodies, and their relations

The determination of stratigraphical boundaries in some geographic area is a complicated palaentological task, involving methods from various branches of earth science. This task cannot be solved by data mining techniques only, it requires various procedures of structural description and analysis. Lithological data are however much more numerical and precise, so data mining techniques are expected to be successful in this area.

Rock can be explicated by descriptive text attributes, like name, texture (internal structure), structure (external structure), some other attributes, as well as by scalar attributes like geochemical or geophysical ones, followed by additional data.

Name is the primary definition of a rock, and it is pointed out by the chemical composition of a rock mainly. Geochemical data following this name are among the most important and precise characteristics in lithological analysis, and describes the chemical composition of a rock. Geochemical data being more precise describe the composition of a sample, which is considered to be taken from a certain layer, so they are suitable for primary identification of a layer.

Nevertheless geochemical data involve large sets of data (numbers and/or categorical values), even for one layer. To find meaningful properties of a layer, or to classify sets of data into categories, to simplify the data and reduce the dimensionality of a description of a layer, data mining techniques are to be used. Clustering techniques are expected, and considered in this article as to solve data simplification and reduction analysis.

3 Data Mining in Geology

Data mining, in it's formal way is a process of posing queries of various kind to large quantities of data, possible stored in databases [3], and extracting relevant information, mainly in the form of patterns, trends, etc. As for geologists essentially, the goals of data mining include classification of samples of shelves, determination of hidden lithological structures, detecting abnormal patterns, predicting structure and composition, some other goals. The decision is also based on experiences from other sites, trend determination, etc. The influence of data mining procedures is to be increased in geology. It offers much cheaper and less time consuming methods as compared to classical ones. For instance, oil companies can do less extremely expensive drillings by applying data mining methods to existing data.

Geologists are applying commonly either a set of statistical methods or just some specific technique alone to produce results wanted. Such a methodology is not enough for data mining, and not good one. The recent understanding of data mining by geologist includes firstly the hunting for methods to be applied, as well as the order of application of such methods. The goal is to find a sequence of methods, that “opens” dataset the best to the needed aspect. There is no method to be taking out of the box and to be applied to geological data. For each dataset the method has to be found and used just for data of given kind exclusively. This gives another motivation in this article to explore data mining procedures for geological data from algorithmic point of view.

4 Clustering through Decision Tree Construction

Traditional clustering techniques are broadly categorized into two classes: partition clustering techniques and hierarchical clustering ones. Although clustering has been studied extensively for a long time, algorithms able to find "natural" or "true" clusters are scarce. Many algorithms have a significant shortcoming – they need to be provided with input parameters, and are very sensitive to them. Such a shortcoming means that only an area expert can define input and confirm results.

Clustering technique based on decision tree construction presented in [4], and applied to geological data, is a novel one. The main algorithmic task – construction of a decision tree is based on popular classification technique, employing partitioning the data space into different classes by using a purity function.

4.1 Decision Tree Construction

On each dimension in *d*-dimensional space (input data: rectangle matrix of data, containing a set of *d* attributes vectors) the algorithm computes *gini* index – a function for splitting, used to evaluate the "goodness" of the alternative splits for an attribute. The current tree node splits on that attribute which produces the best (the smallest) *gini* index. Index at the node is split for its left and right branches (regions limited by previous cuts). The simplified algorithm can be presented as following:

```
function split(node)
   for each attribute $A_i \in (A_1, A_2, .., A_d)$ of the
   dataset D do
         for each value x of $A_i$ in D do
         /* each value considered as possible split */
         compute the gini index on current x
```

```
        end; end
    save the best split for the current node
    call split function on the node's left child
    call split function on the node's right child
end function
```

A resulting decision tree example is shown in Figure 2.

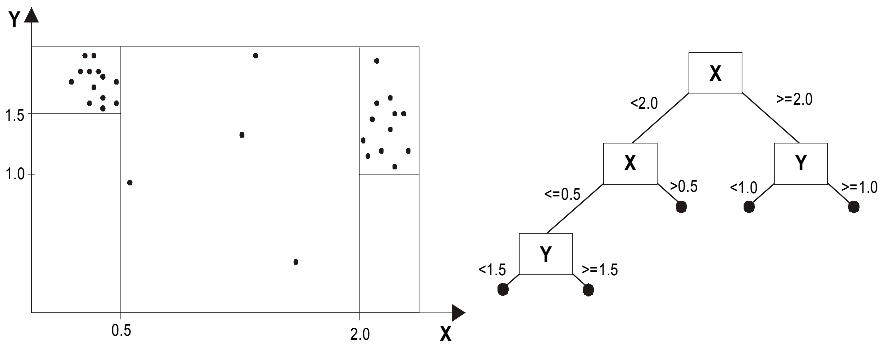

Fig. 2. Decision tree example

4.2 *Gini* Index Calculation

The *gini* index of split is computed for attribute points and some uniformly distributed "non-existing" N points. Each value in dimension A_i is considered as possible split and its *gini* index is calculated. The best (smallest) value has to be taken for the next split.

Let's assume we have an set M with |M| values (|M| denotes the number of values). We assume in addition that there is a set N of uniformly distributed points with |N|=|M| (these numbers are defined on the start; later N points are inherited from the parent node). Each value $x \in M$ divides the set into two regions evaluated by x- (left) and x+(right). On the left side of the current point $x \in M$ there are sets m_{x-} and n_{x-} of points, their values are less than the given value; on the right side – sets of points are $m_{x+}=|M|-m_{x-}$ and $n_{x+}=|N|-n_{x-}$ respectively. The computation formulas for n_{x-} and n_{x+} are defined as

$$n_{x-} = |M| - n_{x+} = \frac{|N|(x - \min(M))}{\max(M) - \min(M)},\ n_{x+} = |M| - n_{x-} = \frac{|N|(\max(M) - x)}{\max(M) - \min(M)}$$

where min(M) is a minimum value in M, max(M) – maximum value in M. This formula can be redefined as "if in the range between min(M) and max(M) there are |N| uniformly distributed fitting points, then in the range between min(M) and x (current value) fit n_{x-} points". This statement gives a powerful tool for uniform set computation on the fly.

The general formula of *gini* index computation on x-split can be defined as:

$$g_x = \frac{n_{x-} + m_{x-}}{|M| + |N|} g_{x-} + \frac{n_{x+} + m_{x+}}{|M| + |N|} g_{x+}$$

where *gini* indexes for the subsets x- and x+ are calculated according to the formula:

$$g_{x*} = 1 - \frac{n_{x*}^2 + m_{x*}^2}{(n_{x*} + m_{x*})^2}$$

where * is a subset symbol (+ or -). In general the *gini* index of the subset can be defined as one minus the sum of set classes' relative frequency to the second power.

When the best split is evaluated and saved for the current node, its left and right child inherits the set N, consisting of n_{x-} and n_{x+} points respectively.

4.4 Calculation Process Management

The calculation of splits must continue while:

- current node contains more points |M|, than the minimum defined by user, and this is the only one input parameter; others are optional and

- current dataset has groups with at least two points; points with the same value are grouped initially providing group value and number of points.

The node split process must continue until these conditions are met. Otherwise calculations on such a branch must stop.

4.4. clusTree Application Overview

While implementing the procedure described above for data mining of geological data, the clusTree application was developed. It provides the complete environment for decision tree construction. The high level of abstraction schema – component diagram is presented in figure 3. Data are stored in MySQL RDBMS table. The MySQL generic C API is used to connect to the database. The IBM XML4C API is used to export/import data and the results of computation to/from external XML files needed.

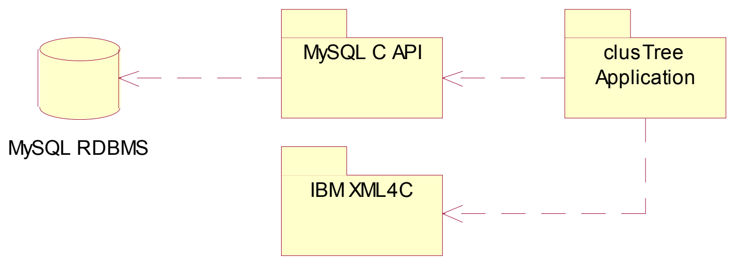

Figure 3. clusTree application: component view

The customizable command system for this application was implemented. These commands can be customized by an end-user with the help of editing the commands.xml file. The default command notations are used in the presentation to follow in this article.

4.5 Database Specific Commands

The database specific commands are designed for database connection management. The preferred input sequence of commands could be as a following one: the definition of input parameters (**dbset**), the connection to database (**dbcon**), and the disconnection from database (**dbdiscon**), after the work has been done.

Command	Syntax	Description
dbset	dbset [-h ..] [-p ..] [-d ..] [-u ..] [-pw ..]	Sets database connection parameters: h - host address (default: localhost) p - port number (default: 3306) d - database name (default: mysql) u - username (default: root) pw - password (default: NULL)
dbcon	dbcon	Connects to the specified database
dbshow	dbshow	Displays database information and status
dbdiscon	dbdiscon	Disconnects from the connected database

4.6 Matrix Specific Commands

Matrix specific commands present an interface for matrix (table) manipulation techniques. The input sequence could be the following one: the **mxset** command sets the desired matrix (this produces an object responsible for operation on each table attribute), the commands **mxshow** and **mxsplit** display information on selected attribute (attributes), and the command **mxrem** removes any constructions from the main memory, after the work has been done.

Command	Syntax	Description
mxset	mxset ..	sets and constructs matrix, .. – desired table name.
mxshow	mxshow [..]	displays attribute specific information or, if the attribute parameter is not typed in – lists all attributes, .. – desired attribute number (starting from 0).
mxsplit	mxsplit [..]	splits the specified attribute, and displays split information. If the attribute number is not typed in – displays information of the best split, .. – desired attribute number (starting from 0).
mxrem	mxrem	removes the active matrix.

4.7 Decision Tree Specific Commands

With the set of commands designed user is able to construct the decision tree, and manipulate with it. To construct the decision tree the connection must be established, and the matrix of data must be set. If these conditions are met the following tasks could be performed: the decision tree construction (**dtconst**), the export of a tree (**dtexport**), and the destruction of a tree (**dtdestr**) when the tree has been done.

Command	**Syntax**	**Description**
dtconst	dtconst [-m ..]	constructs the decision tree on existing matrix, parameters: m – minimum points in the region to perform a split.
dtexport	dtexport ..	exports the decision tree to the specified XML file.
dtdestr	dtdestr	removes decision tree from the main memory.

4.8 Other Commands

These commands implement help, command file execution and other tasks.

Command	**Syntax**	**Description**
exit	exit	destroy everything and leave.
?	? [..]	displays help screen, if command is not specified as an argument – lists all available commands.
runfile	runfile .. [-c]	reads the specified file and runs commands, each command must be written in a separate line. Comments must start with **;** (semicolumn). Key –c commands to print comments to console (otherwise comments will be skipped).
sql	sql	starts an SQL session on existing database connection. To exit it – type **exit**.

5 The Initial Analysis of Clustering Results

The tool developed enables us to produce an analysis of geological data. Because of high dimensionality of such data and complicated distribution of the values of each attribute, as well as complex interrelation of attributes to each other, a very preliminary analysis of data is given, just to present how the tool works.

Figure 4 presents a drawing of decision tree, generated automatically by the tool, for the case of clustering data from a sample. This sample presents the geochemical part of data received from boring holes, enabling us to research the structure of sedimental layers of geological body. Such figures, like figure 4 gives an impression about the structural features of a decision tree constructed (how much it is unbalanced, what is a height of such tree, how paths of different length are distributed, etc.).

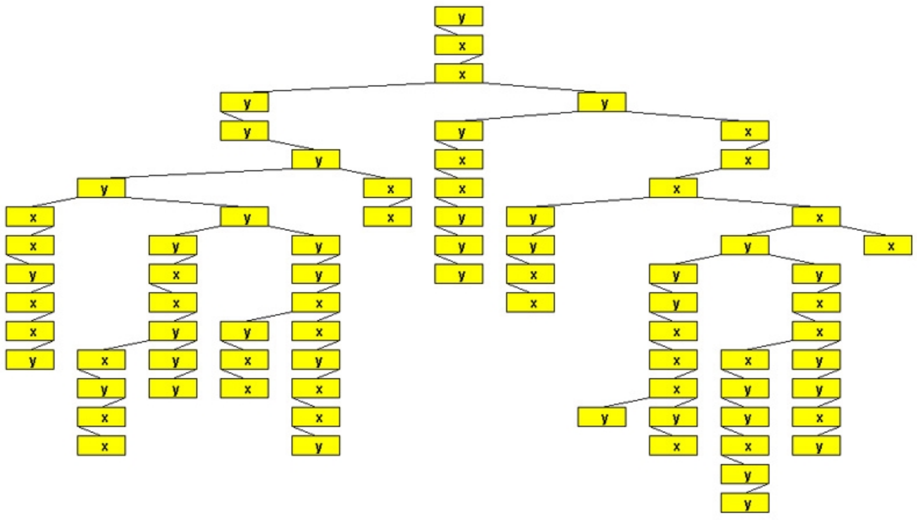

Figure 4. The sample of decision tree produced by the system.

The structural properties of a tree produced don't give information on attributes to be used for splitting of nodes. The height, as well as the length of paths of a tree gives us a possibility to evaluate the volume of clusters received, and the complexity of a method, by which they are produced. To identify attributes participating in splits of nodes (in other words to name specific geological characteristics partitioning data), some further modifications have to be done. Figure 5 presents a tree, with names of attributes written at nodes. These attributes show which geological characteristic was used at which node, while splitting operations were produced.

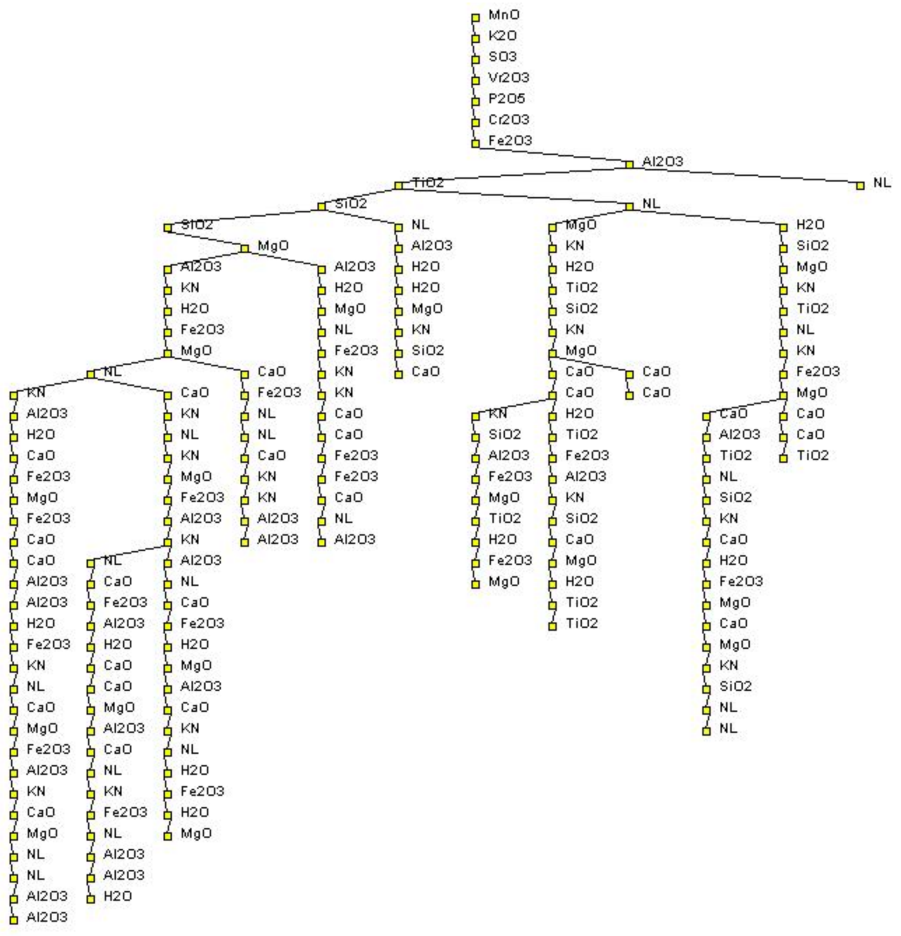

Figure 5. The tree with names of attributes splitting nodes.

The tree in figure 5 gives also an impression about the complexity of analysis of geological data to be done. This constitutes that data mining in geology is a complicated task indeed, and it requires many statistical and clustering methods to be applied (like reduction of dimensionality of data, character of distribution, interrelations of different attributes, etc.).

Such an analysis will be produced in the future, and presented in research articles to be published later.

6 References

1. Cherkassky V., Mulier F. *Learning from Data. Concepts, Theory, and Methods.* John Wiley & Sons, Inc. 1998
2. Taylor, S.R. The origin of the Earth. *AGSO Journal of Australian Geology and Geophysics*, 17 (1), 27–31, 1997
3. Thuraisingham B. *Data Mining. Technologies, Techniques, Tools, and Trends*. John Wiley & Sons, Inc. 2000
4. Liu B., Xia X., Yu P. S. Clustering Through Decision Tree Construction. in *SIGMOD-00*, 2000
5. Mehta M., Agrawal R., Rissanen J. SLIQ: A Fast Scalable Classifier for Data Mining. *Lecture Notes in Computer Science*, v. 1057