

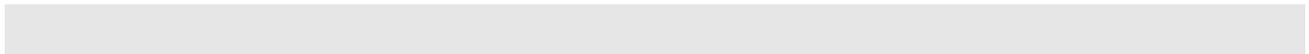
# Market Basket Analysis User Manual

**Author:** Petra Vaníčková  
**Library:** MBA.dll  
**Runnable class:** MBAMain

---

Content:

|          |  |           |
|----------|--|-----------|
| <b>1</b> | <b>INTRODUCTION TO MARKET BASKET ANALYSIS</b> .....    | <b>3</b>  |
| <b>2</b> | <b>MARKET BASKET ANALYSIS – USER INTERFACE</b> .....   | <b>4</b>  |
| 2.1      | MAIN WINDOW .....                                      | 4         |
| 2.1.1    | Menu .....   | 4         |
| 2.1.2    | Buttons on tool bar.....                               | 8         |
| 2.1.3    | Parameters .....                                       | 8         |
| 2.1.4    | Data table .....                                       | 9         |
| 2.1.5    | Result table.....                                      | 10        |
| 2.2      | TREE OF LARGE SETS WINDOW.....                         | 11        |
| 2.3      | LARGE SETS CHART WINDOW .....                          | 13        |
| <b>3</b> | <b>MARKET BASKET ANALYSIS – TUTORIALS</b> .....        | <b>14</b> |
| 3.1      | PREPARE YOUR DATA.....                                 | 14        |
| 3.2      | ADD YOUR DATA AS VERSION IN MAIN APPLICATION .....     | 14        |
| 3.3      | LOAD MBA MODULE (IF NOT LOADED) INTO APPLICATION ..... | 14        |
| 3.4      | RUN MBA MODULE.....                                    | 15        |
| 3.5      | LOAD DATA TO MBA MODULE .....                          | 15        |
| 3.6      | RUN ALGORITHM.....                                     | 15        |
| 3.7      | SAVE ALGORITHM RESULTS .....                           | 15        |
| <b>4</b> | <b>REQUIREMENTS</b> .....                              | <b>16</b> |
| <b>5</b> | <b>SAMPLES</b> .....                                   | <b>17</b> |



# 1 Introduction to Market Basket Analysis

Market Basket Analysis (MBA) is a method for recognition of dependencies in data.

Let's imagine that our data represents transactions in supermarket. Each column of the table represents one item (e.g. milk, butter ...), each row represents one transaction (one customer). MBA is an algorithm which takes the data and creates appropriate rules.

**Rule** can look like "Item1, Item5->Item7" which means: if the first item and fifth item are bought together, then the seventh item is also bought.

Each such rule has a measure called **confidence**. The bigger this confidence is, the more you can trust the rule.

Another parameter which can be set to the algorithm is called **support**. Support is a percentage parameter which says how often an item or a set of items must occur in all rows to be considered large (how many customers have bought the item).

**Large item** is an item with occurrence more or equal than support, e.g. milk and butter have been bought by 50% of customers and thus set consisting of milk and butter is large for support 0.5 and less.

Rules are created only from large sets.

Algorithm can be split in two steps:

- a) Searching for large item sets
- b) Making rules from large items

This module focuses on graphical illustration of process of searching for large items and also shows a result of the second part of the MBA algorithm.

The second part of module takes each found large set (L), then recursively creates two subsets A,B (where  $A \cup B = L$  and  $A \cap B = \text{empty set}$ ) and finally checks if the rule  $A \rightarrow B$  has sufficient confidence.

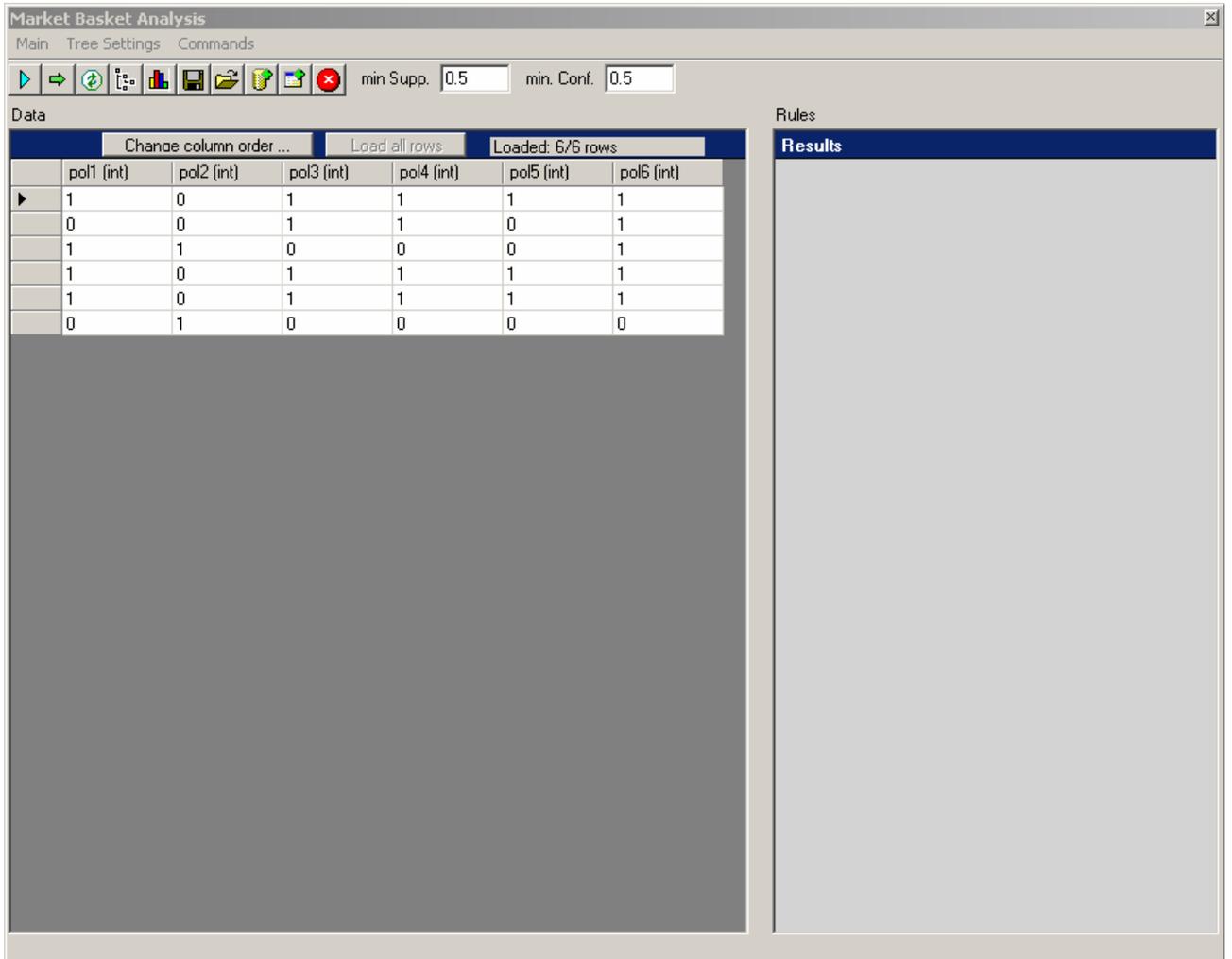
Detailed information about MBA can be found at the following links:

[www-db.stanford.edu/~sergey/dic.ps](http://www-db.stanford.edu/~sergey/dic.ps)

[www2.cs.uregina.ca/~hamilton/courses/831/notes/itemsets/DIC.html](http://www2.cs.uregina.ca/~hamilton/courses/831/notes/itemsets/DIC.html)

## 2 Market Basket Analysis - User Interface

### 2.1 Main Window



Market Basket Analysis

Main Tree Settings Commands

min. Supp. 0.5 min. Conf. 0.5

Data

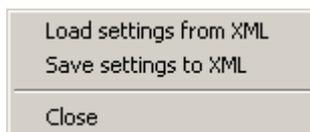
|   | pol1 (int) | pol2 (int) | pol3 (int) | pol4 (int) | pol5 (int) | pol6 (int) |
|---|------------|------------|------------|------------|------------|------------|
| ▶ | 1          | 0          | 1          | 1          | 1          | 1          |
|   | 0          | 0          | 1          | 1          | 0          | 1          |
|   | 1          | 1          | 0          | 0          | 0          | 1          |
|   | 1          | 0          | 1          | 1          | 1          | 1          |
|   | 1          | 0          | 1          | 1          | 1          | 1          |
|   | 0          | 1          | 0          | 0          | 0          | 0          |

Rules

Results

#### 2.1.1 Menu

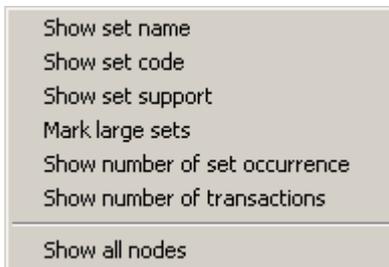
##### Main



- Load settings   
Loads parameters (min. support, min. confidence) from MBA.xml file.
- Save settings   
Saves parameters to xml file.

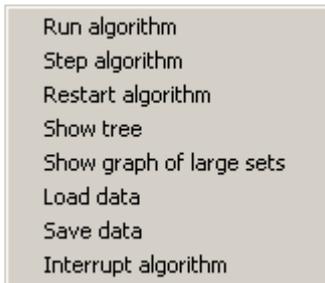
- Close   
Closes MBA module.

### Tree settings



- Show set name  
This flag determines if the set description (e.g. {1, 3, 5}), which means „set of item1, item3 and item5“, should be shown directly near node in „[Tree of large sets](#)“ window.
- Show set code  
This flag determines if the set code (e.g. 0000101) should be shown directly near node in „[Tree of large sets](#)“ window. Meaning of the code is following: if you read the code from right to left, the rightmost number (0 or 1) has index 1 and represents item1 in the set, the second position from right represents item2 etc. If there is 1 at nth position from the right, the item n is included in the set.  
E.g. 0101 is code for set {1, 3}.
- Show set support  
This flag determines if the set support should be shown directly near node in „[Tree of large sets](#)“ window.
- Mark large sets  
This flag determines if the set state (large or not) should be shown directly near node in „[Tree of large sets](#)“ window.
- Show number of set occurrences  
This flag determines if the number of set occurrences in data (e.g. how many times we can find the set of milk and butter together in customers' basket) should be shown directly near node in „[Tree of large sets](#)“ window.
- Show number of transactions  
This flag determines if the number of rows (number of customers) should be shown directly near node in „[Tree of large sets](#)“ window.  
  
If the flags above are not checked, you can find detailed information about a node in the „Info“ window for the appropriate node by selecting the node in „[Tree of large sets](#)“ window.
- Show all nodes  
This flag determines if all sets (including the small ones) should be drawn in „[Tree of large sets](#)“ window.

### Commands



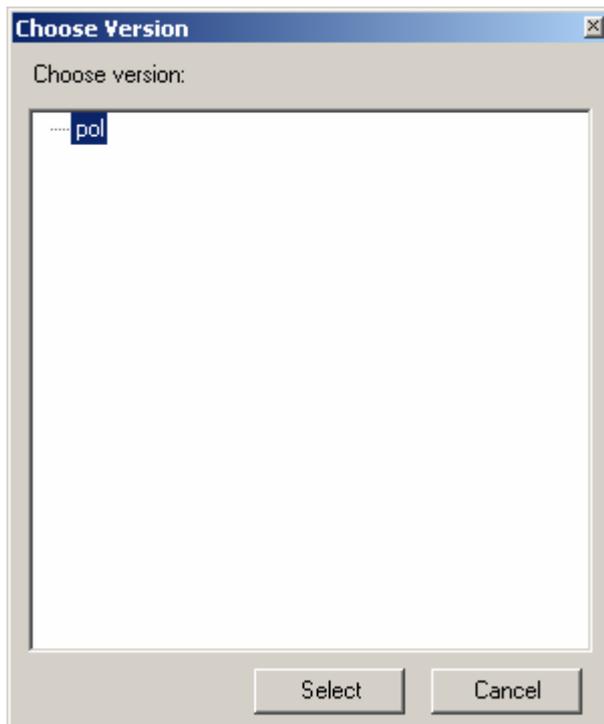
- Run algorithm   
This command starts the whole algorithm (both the large set creation and the process of making rules from large sets) at once, you can check the result in “Result table”.
- Step algorithm   
This command makes just one step of algorithm, which means, that one large set (parent set) is already joined with each of current single-member large sets to create candidates for new large sets (each candidate is a union of the parent set and one single-member large set). These candidates are tested and the large ones added to the “large set repository”. Parent set is shown in “[Tree of large sets](#)” and in “[Large sets chart](#)” window.
- Restart algorithm   
This command is useful especially if you want to restart algorithm with following “stepping” of algorithm. Restart erases “[Tree of large sets](#)”, “[Large sets chart](#)” and finds one-member large sets.
- Show tree   
This command shows “Tree of large sets” window.

If there is no data loaded for MBA following window appears:

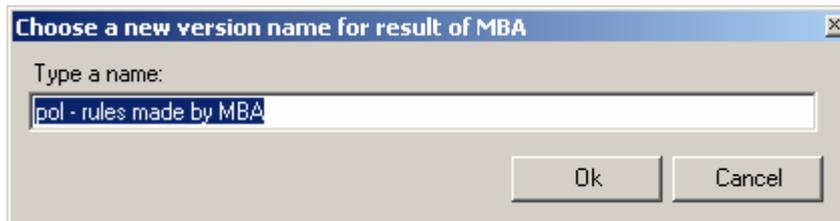


You can solve such situation by loading data with command “Load data” (see bellow).

- Show graph of large sets   
This command shows “Large sets chart” window.
- Load data   
This command shows “Choose Version” dialog which allows user to choose different data for MBA method. Chosen data must consist of {0, 1} integer values.



- Save data   
This command shows dialog



You can choose a name for version which will contain result rules made by MBA module. New version is created after pushing the "Ok" button.  
If the algorithm has not created such result yet or there are no rules to save, following message appears:



- Interrupt algorithm  
This command cancels algorithm run.

## 2.1.2 Buttons on tool bar



-  The same functionality as “Run algorithm” command.
-  The same functionality as “Step algorithm” command.
-  The same functionality as “Restart algorithm” command.
-  The same functionality as “Show tree” command.
-  The same functionality as “Show graph of large sets” command.
-  The same functionality as “Save configuration” command.
-  The same functionality as “Load configuration” command.
-  The same functionality as “Load data” command.
-  The same functionality as “Save data” command.
-  The same functionality as “Interrupt algorithm” command.

## 2.1.3 Parameters



You can set minimal support and minimal confidence manually in these text boxes. These new changes take effect directly after new values setting.

---

## 2.1.4 Data table

| Change column order ... |            | Load all rows |            | Loaded: 6/6 rows |            |            |
|-------------------------|------------|---------------|------------|------------------|------------|------------|
|                         | pol1 (int) | pol2 (int)    | pol3 (int) | pol4 (int)       | pol5 (int) | pol6 (int) |
| ▶                       | 1          | 0             | 1          | 1                | 1          | 1          |
|                         | 0          | 0             | 1          | 1                | 0          | 1          |
|                         | 1          | 1             | 0          | 0                | 0          | 1          |
|                         | 1          | 0             | 1          | 1                | 1          | 1          |
|                         | 1          | 0             | 1          | 1                | 1          | 1          |
|                         | 0          | 1             | 0          | 0                | 0          | 0          |

This part of main window shows currently used data.

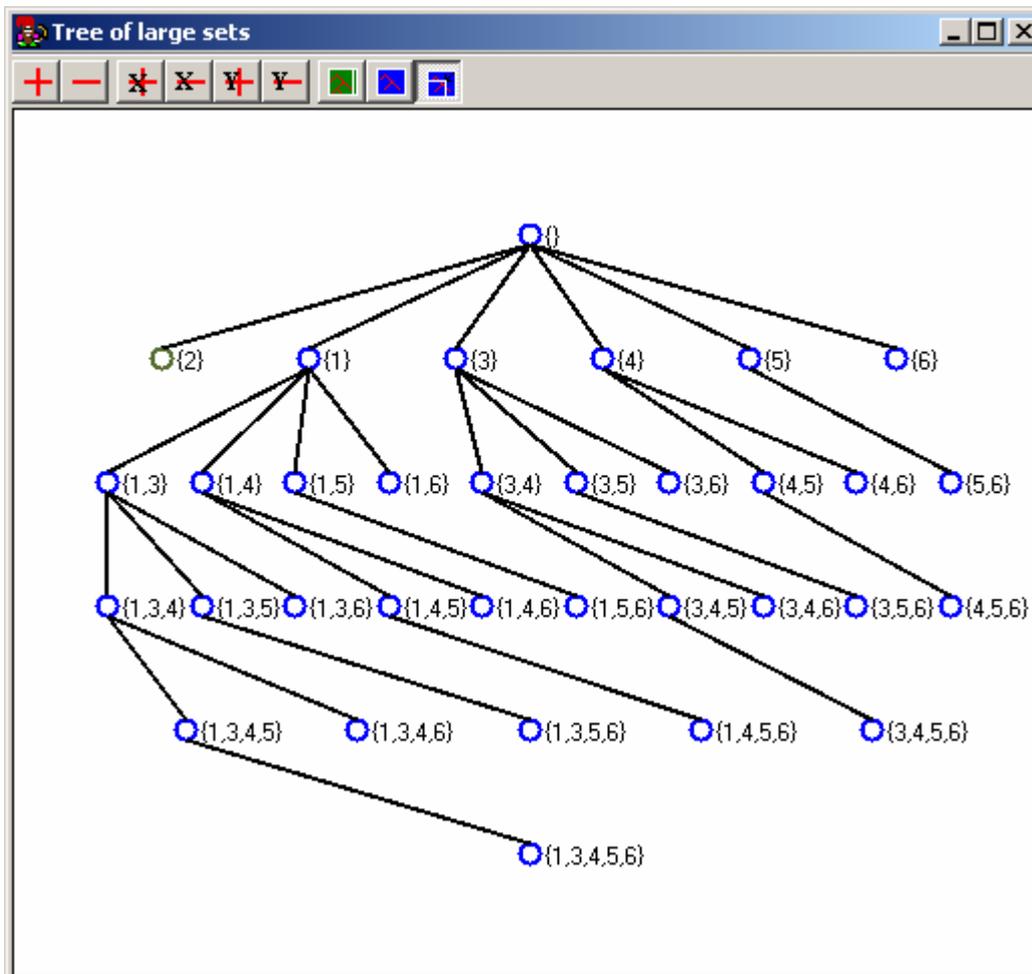
## 2.1.5 Result table

Rules

| Results     |            |
|-------------|------------|
| Rule        | Confidence |
| {1} → {3}   | 0,75       |
| {3} → {1}   | 0,75       |
| {1} → {4}   | 0,75       |
| {4} → {1}   | 0,75       |
| {1} → {5}   | 0,75       |
| {5} → {1}   | 1          |
| {1} → {6}   | 1          |
| {3} → {4}   | 1          |
| {4} → {3}   | 1          |
| {3} → {5}   | 0,75       |
| {5} → {3}   | 1          |
| {3} → {6}   | 1          |
| {4} → {5}   | 0,75       |
| {5} → {4}   | 1          |
| {4} → {6}   | 1          |
| {5} → {6}   | 1          |
| {1} → {3,4} | 0,75       |
| {1,3} → {4} | 1          |
| {1,4} → {3} | 1          |
| {3} → {1,4} | 0,75       |
| {3,4} → {1} | 0,75       |
| {4} → {1,3} | 0,75       |
| {1} → {3,5} | 0,75       |
| {1,3} → {5} | 1          |
| {1,5} → {3} | 1          |
| {3} → {1,5} | 0,75       |
| {3,5} → {1} | 1          |
| {5} → {1,3} | 1          |
| {1} → {3,6} | 0,75       |
| {1,3} → {6} | 1          |

This part of main window shows results of MBA algorithm.

## 2.2 Tree of large sets window



This window illustrates creating of large sets. This is tree that is shown in “Show all nodes” mode, which means that you can see both small sets (green color) and large sets (blue color).

New sets are generated from parent set by joining with large single-member sets which are bigger than biggest member of parent set.

### Buttons on the tree view's tool bar



Zoom in and zoom out.



Zoom in and zoom out axis independently.



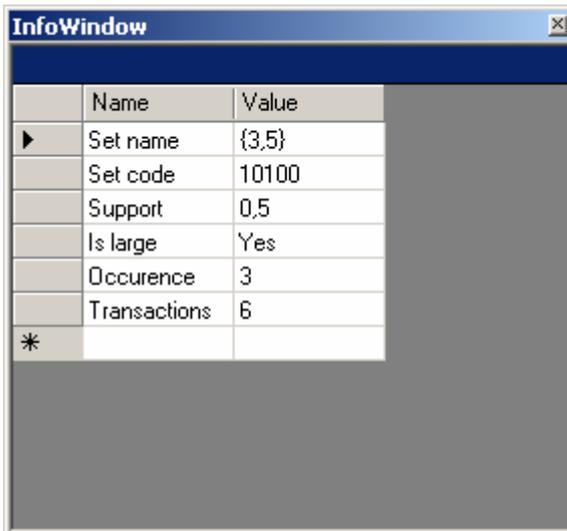
Set exact space of each node in the pixels.



Fit the tree to the window.



Fit the tree to the window always when resizing the window. Toggle button, default is pushed.

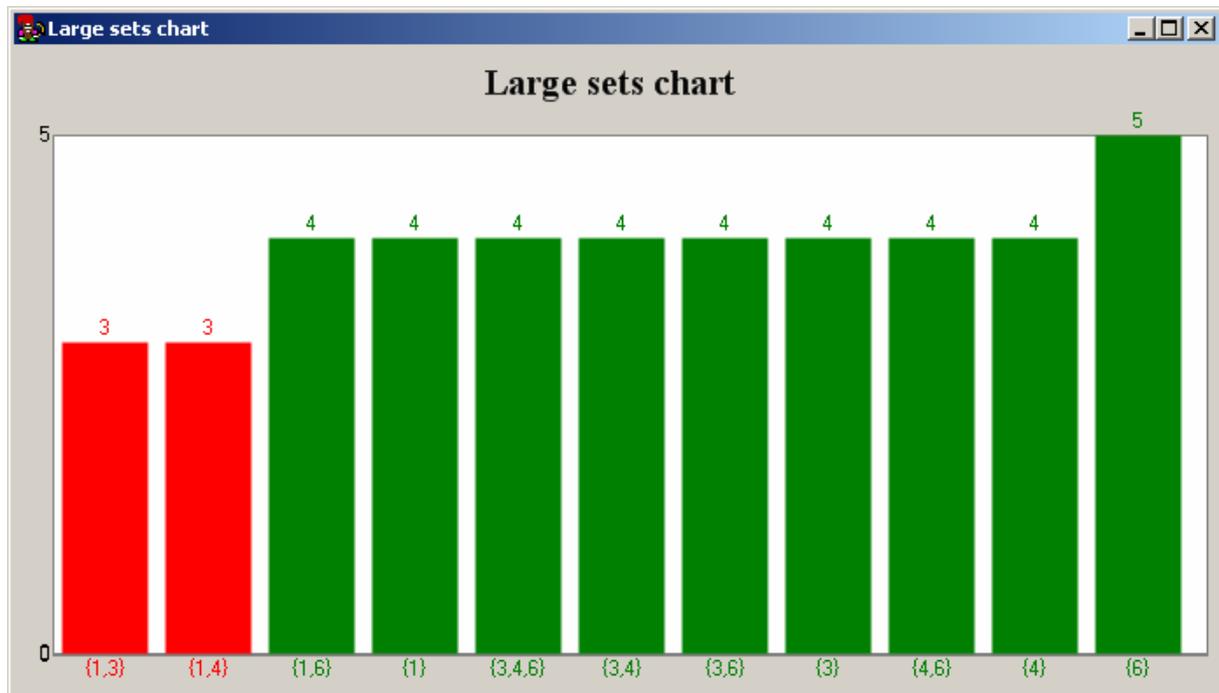


The image shows a software window titled "InfoWindow" with a close button in the top right corner. The window contains a table with two columns: "Name" and "Value". The table has seven rows of data, followed by a row with an asterisk in the "Name" column. The background of the window is a solid grey color.

|   | Name         | Value |
|---|--------------|-------|
| ▶ | Set name     | {3,5} |
|   | Set code     | 10100 |
|   | Support      | 0,5   |
|   | Is large     | Yes   |
|   | Occurence    | 3     |
|   | Transactions | 6     |
| * |              |       |

This window appears after selecting a node from tree in "Tree of large sets" window. It contains basic information about set represented by selected node.

## 2.3 Large sets chart window



This window graphically illustrates large sets (green) and small sets (red) and shows number of occurrences of each set.

## 3 Market Basket Analysis - Tutorials

Goal of this tutorial is to show you how to use MBA module.

### 3.1 Prepare your data

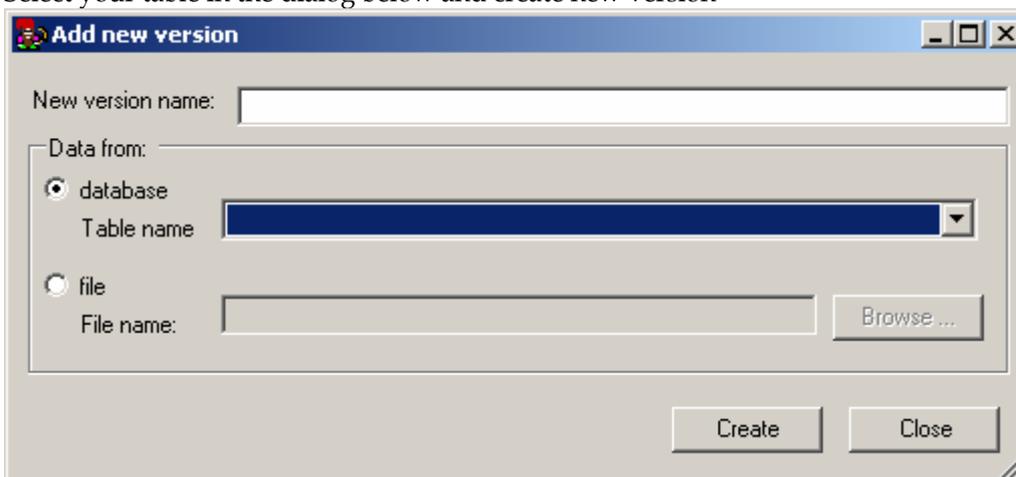
You will need database table, which is filled with zeros and ones (each column of type number) in which each row represents set of items (one in  $n^{\text{th}}$  position means membership of  $n^{\text{th}}$  item in the set). For example, columns can represent people, and rows photographs. Occurrence of number one in the  $x^{\text{th}}$  row and  $y^{\text{th}}$  column means that  $x^{\text{th}}$  person is present in  $y^{\text{th}}$  photograph.

### 3.2 Add your data as version in main application



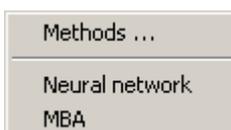
Click this button in main knocker window

Select your table in the dialog below and create new version



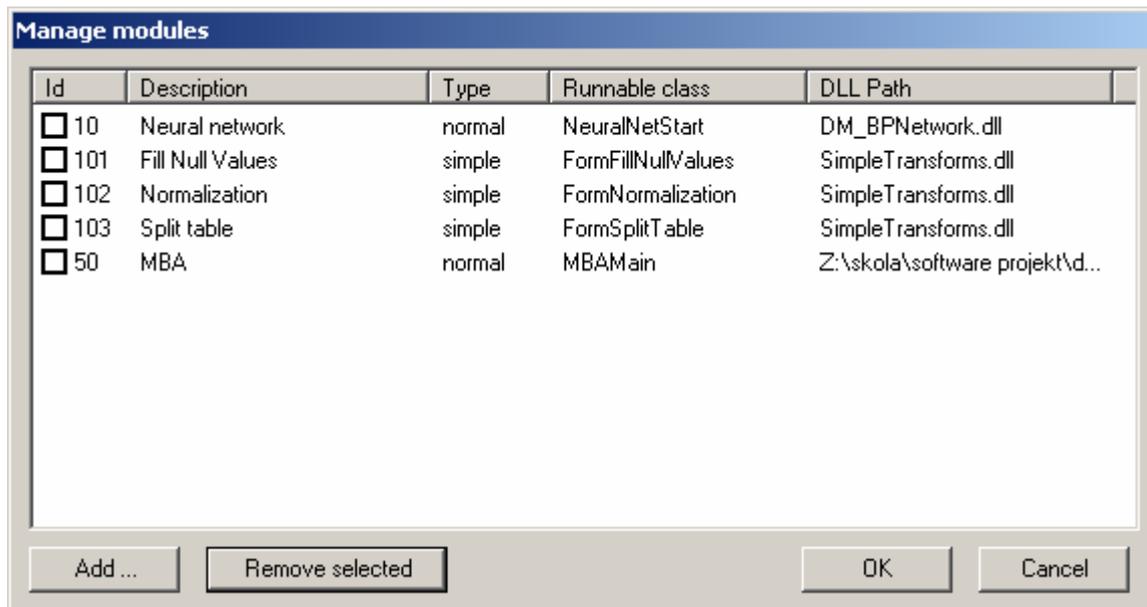
### 3.3 Load MBA module (if not loaded) into application

You can find MBA in menu "Methods" of main application window.



If there is no MBA method loaded, add it to the list of methods by choosing a command Methods -> Methods as shown bellow.

Click "Add" button, find path of MBA.dll and choose MBAMain from class list.



### 3.4 Run MBA module

Choose MBA module from list in “Methods” menu.

### 3.5 Load data to MBA module

Load data for method by pressing  and select your version.

### 3.6 Run algorithm

Open tree form and chart form by pressing  and .

Press  or  to start the algorithm.

### 3.7 Save algorithm results

Save list of rules to database by pressing .

---

## 4 Requirements

Files needed to run MBA module:

- • all common components of main application Knocker
- • MBA.dll
- • DMTransformStruct.dll
- • PtreeViewer.dll
- • GuiExt.dll
- • Gui.dll
- • DasNetBarChart.dll

If you wish to load parameters, use xml file created by MBA module called MBA.xml. This file is created after saving configuration from Main menu -> Main.

---

## 5 Samples

You can find sample data for MBA in file Photos.csv. Column represents a person; row represents a picture. Zero in column means, that the person is not present at the photo, one means the person is at the picture.

---