

1 Topic

Description of the SIPINA method through a case study.

SIPINA is a data mining tool. But it is also a machine learning method. It corresponds to an algorithm for the induction of decision graphs (see References, section 9). A decision graph is a generalization of a decision tree¹ where we can merge any two terminal nodes of the graph, and not only the leaves issued from the same node.

The idea of the merging already exists for some decision tree algorithms e.g. CHAID or CART. But in this case, the joining operation is restricted for the leaves coming from the same node, during the split operation. It is known also as a grouping of the values of the discrete predictors. For CART, the merging is continued until we obtain two groups; CHAID method searches the most relevant grouping, sometimes it performs no grouping, sometimes all the values are merged into one group.

SIPINA generalizes this idea by allowing the combination of any two leaves of the graph.

Roughly speaking, at each step, the induction algorithm searches the best configuration of the decision graph, the one which improves the overall evaluation measure of the partition. This next configuration can come (1) from a splitting of a node or (2) from a merging of two nodes. This is possible because the criterion assesses the whole decision graph². The ability to merge nodes allows to avoid the data fragmentation which is one of the main drawback of the decision tree algorithms, especially when we deal with a small dataset.

The SIPINA method is only available into the version 2.5 of SIPINA data mining tool. This version has some drawbacks. Among others, it cannot handle large datasets (higher than 16.383 instances). But it is the only tool which implements the decision graphs algorithm. This is the main reason for which this version is available online to date. **If we want to implement a decision tree algorithm such as C4.5 or CHAID, or if we want to create interactively a decision tree³, it is more advantageous to use the research version (named also version 3.0).** The research version is more powerful and it supplies much functionality for the data exploration.

In this tutorial, we show how to implement the Sipina decision graph algorithm with the Sipina software version 2.5. We want to predict the low birth weight of newborns from the characteristics of their mothers. We want foremost to show how to use this 2.5 version which is not well documented. We want also to point out the interest of the decision

¹ http://en.wikipedia.org/wiki/Decision_tree_learning

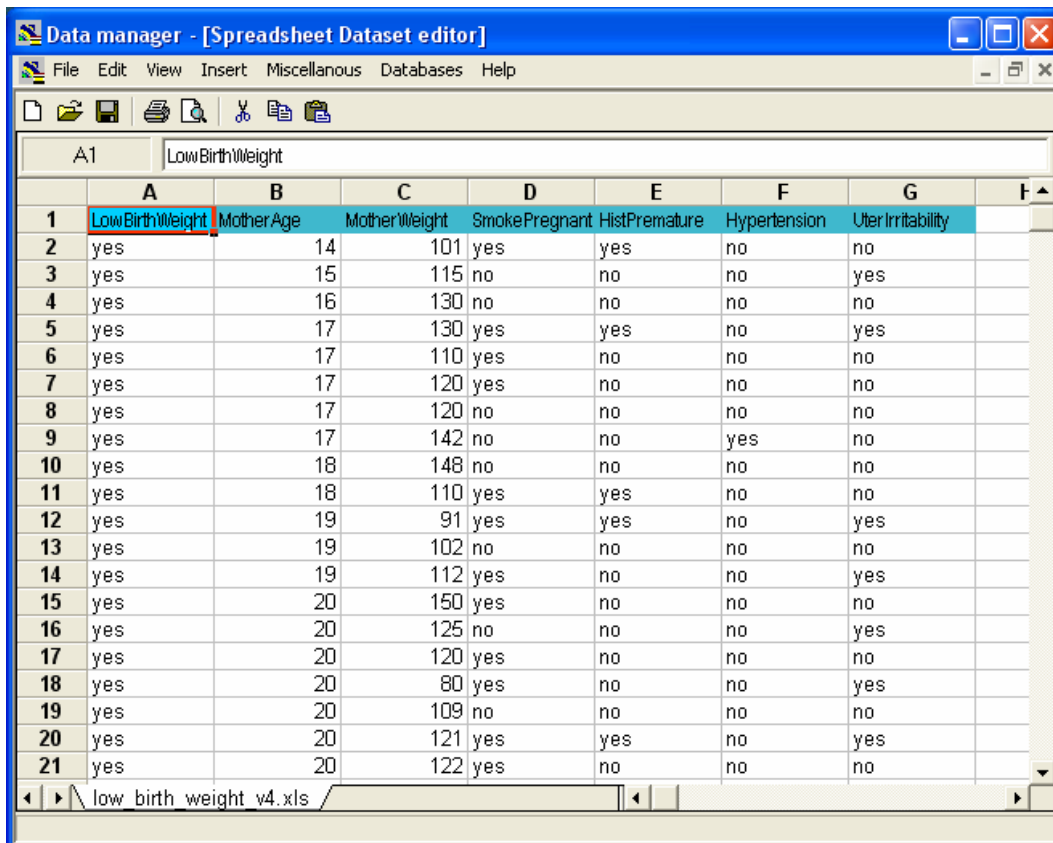
² A variant of SIPINA incorporates a variant. It assesses also the profitability of a merging-splitting process i.e. it evaluates the relevance of a splitting after the merging of two nodes. But the computation time is much longer.

³ <http://data-mining-tutorials.blogspot.com/2008/11/interactive-induction-of-decision-tree.html> or <http://data-mining-tutorials.blogspot.com/2008/10/interactive-tree-builder.html>

graphs when we treat a small dataset i.e. when the data fragmentation becomes a crucial problem.

2 Dataset

The data file LOW_BIRTH_WEIGHT_V4.XLS contains 113 instances and 7 attributes. The class attribute is LOWBIRTHWEIGHT (YES or NO). There are 6 descriptors (Figure 1).



	A	B	C	D	E	F	G	H
1	LowBirthWeight	MotherAge	MotherWeight	SmokePregnant	HistPremature	Hypertension	UterIrritability	
2	yes	14	101	yes	yes	no	no	
3	yes	15	115	no	no	no	yes	
4	yes	16	130	no	no	no	no	
5	yes	17	130	yes	yes	no	yes	
6	yes	17	110	yes	no	no	no	
7	yes	17	120	yes	no	no	no	
8	yes	17	120	no	no	no	no	
9	yes	17	142	no	no	yes	no	
10	yes	18	148	no	no	no	no	
11	yes	18	110	yes	yes	no	no	
12	yes	19	91	yes	yes	no	yes	
13	yes	19	102	no	no	no	no	
14	yes	19	112	yes	no	no	yes	
15	yes	20	150	yes	no	no	no	
16	yes	20	125	no	no	no	yes	
17	yes	20	120	yes	no	no	no	
18	yes	20	80	yes	no	no	yes	
19	yes	20	109	no	no	no	no	
20	yes	20	121	yes	yes	no	yes	
21	yes	20	122	yes	no	no	no	

Figure 1 - The 20 first instances of the LOW_BIRTH_WEIGHT_V4.XLS data file

3 Installation of the 2.5 version

This old 2.5 version of SIPINA is available on the download page of the SIPINA website (Figure 2). The setup file is http://eric.univ-lyon2.fr/~ricco/softs/Setup_Sipina_V25.exe

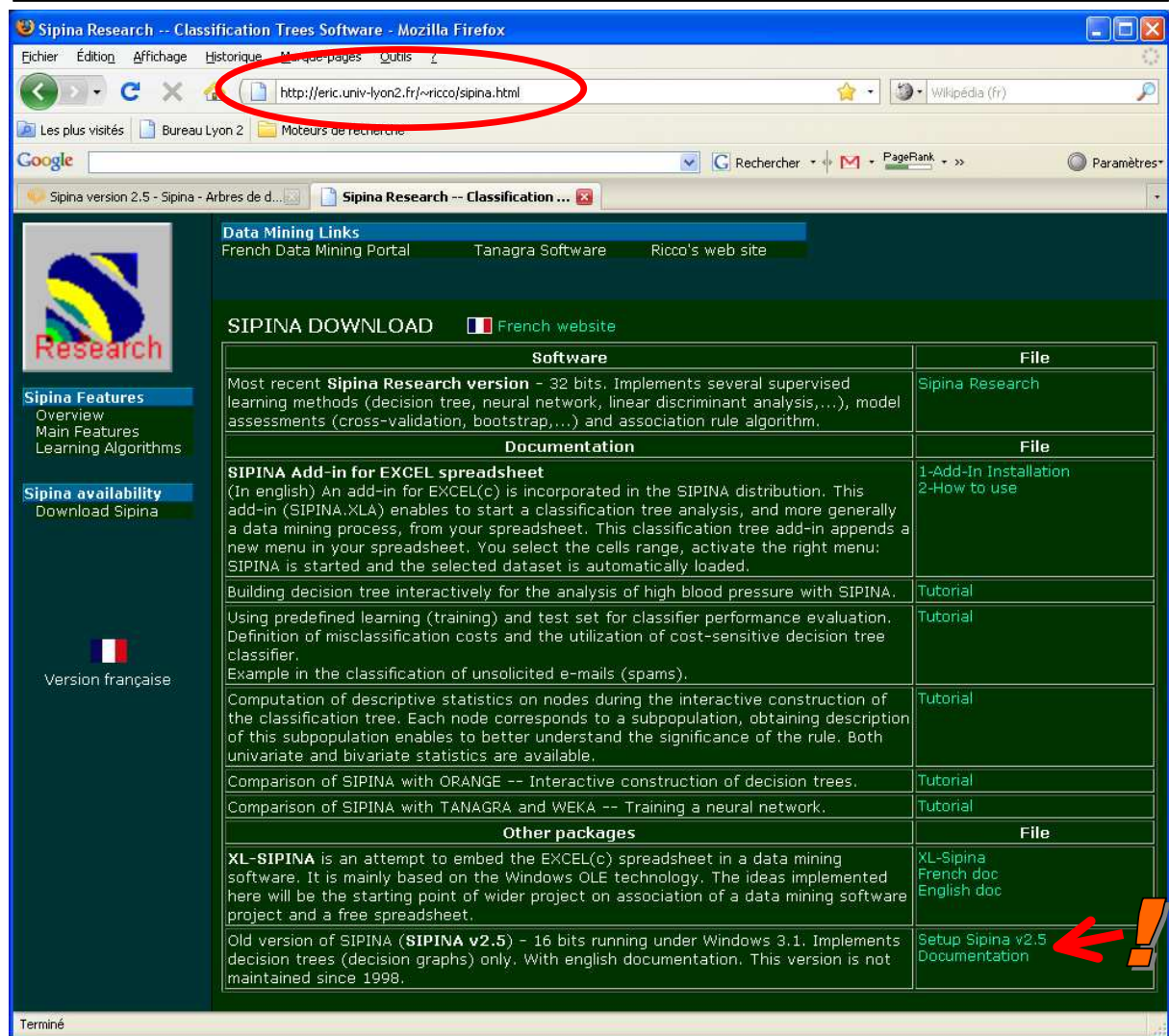
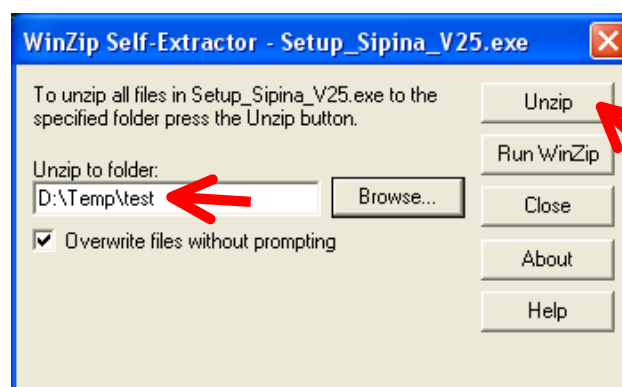


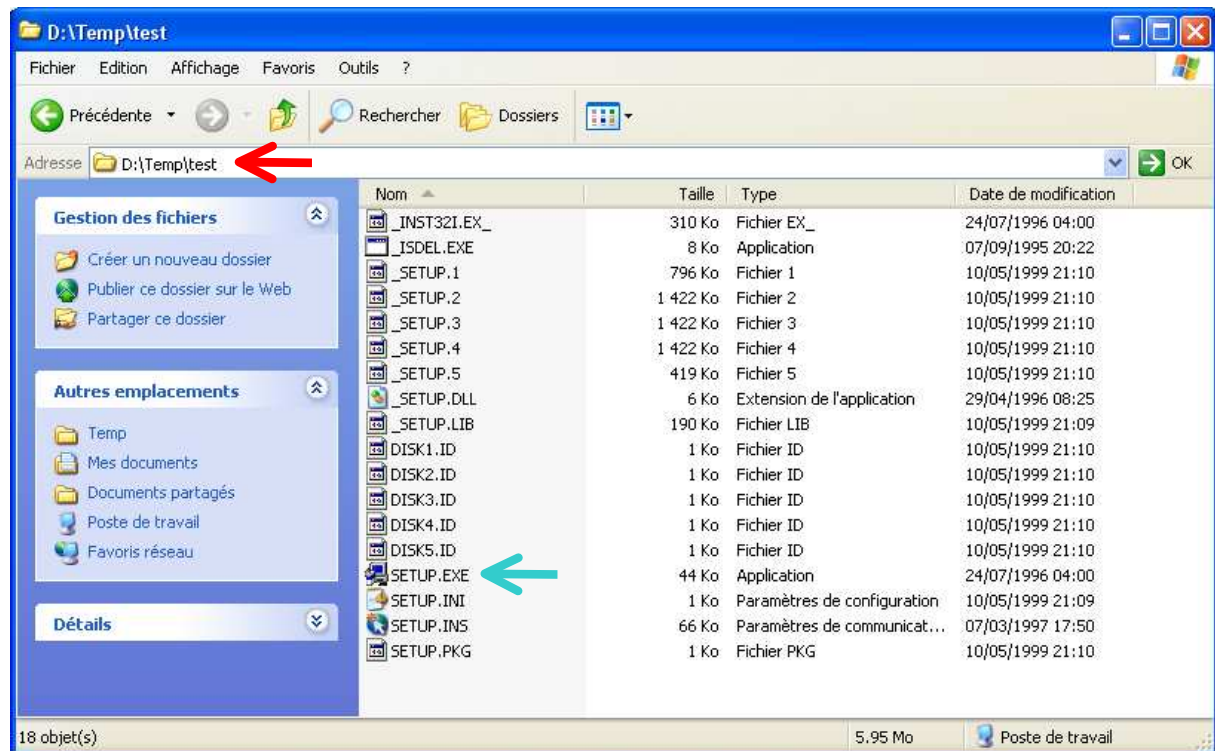
Figure 2 – Sipina website – Download page

A user's guide is available (<http://data-mining-tutorials.blogspot.com/2010/05/users-guide-for-old-sipina-25-version.html>). It is a bit deprecated because it is related to the 1.1 version. But it described the main functionalities of the tool.

After we download the setup file, we launch it. The installation files are copied in a temporary directory by clicking on the UNZIP button.



The installation files are the following.



We double click on the SETUP.EXE executable file to launch the installation. The process complies with the standard installation process under Windows OS ([Sipina 2.5 is operational from Windows 3.1 up to Windows Vista](#)). Some libraries (DLL and OCX) are copied into the executable directory.

Once the installation is processed, the program is available in the START / PROGRAMS menu of Windows. The folder name is **Sipina v2.5**.

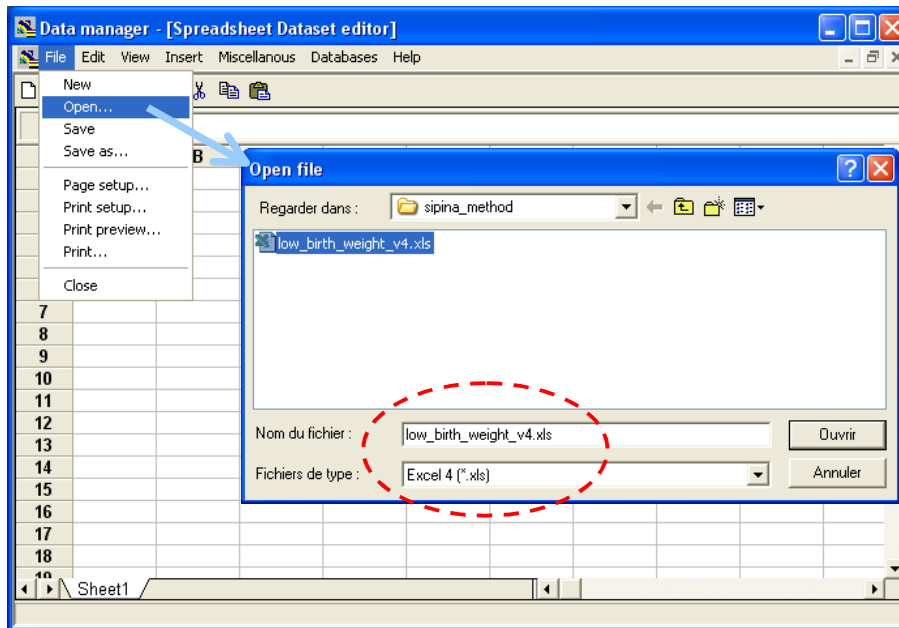
4 Preparing the dataset with DATAMANAGER

This 2.5 version uses its own file format. A dataset is subdivided in two files: the PAR file which corresponds to the description of the attributes; the DAT file which describes the values for each instance. A specific program, DATA MANAGER, is intended to prepare correctly these files (PAR and DAT) from a standard file format such as text file format or spreadsheet file format (Excel).

4.1 Handling the EXCEL 4.0 file format

DATA MANAGER can handle the Excel 4.0 or 5.0 file format (*there is only one sheet into the Excel 4.0 workbook*). The easiest way is to export the dataset in the Excel 4.0 format from a spreadsheet such as Open Office or a more recent version of Excel.

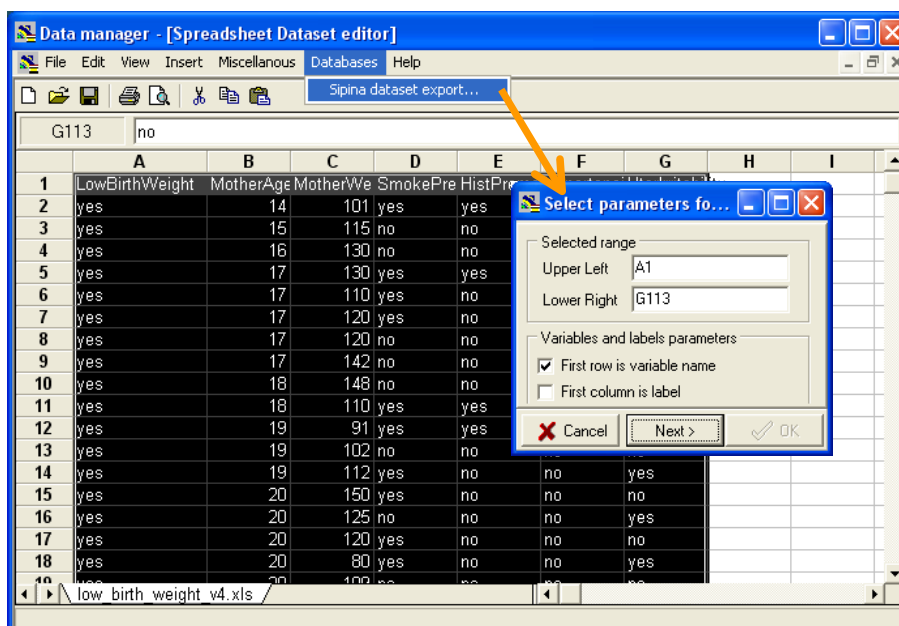
Then we can launch DATA MANAGER and we load the data file by clicking on the FILE / OPEN menu. In this tutorial, LOW_BIRTH_WEIGHT_V4.XLS is already in the Excel 4.0 file format.



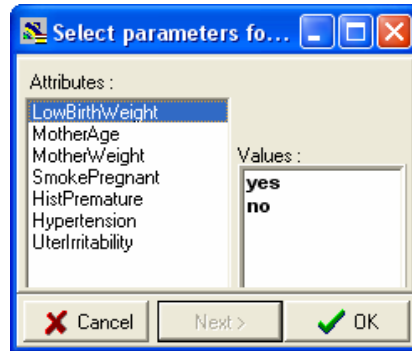
DATA MANAGER incorporates the functions of a standard spreadsheet application. We can add a column, insert values, define formulas, etc. The only restriction is that the number of rows is limited to 16384 (i.e. 16383 instances because the first row is dedicated to the variable names). But it is rather adequate for academic case studies.

4.2 Creating the DAT and PAR data files for SIPINA

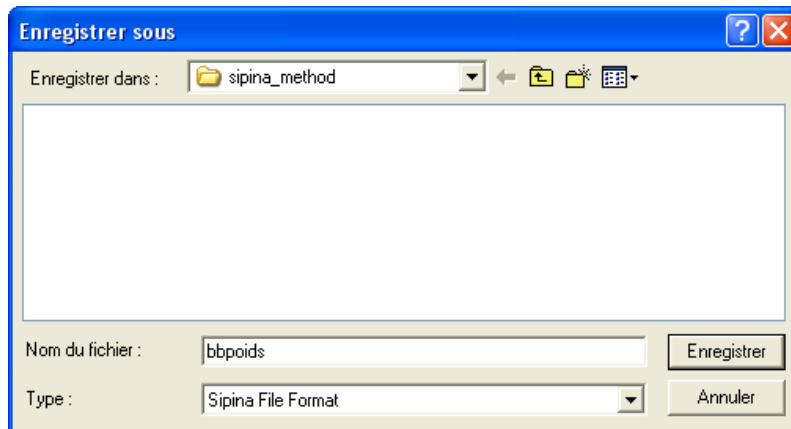
We must create the DAT and PAR files before starting the analysis. We select the dataset, including the first row. Then, we click on the DATABASES / SIPINA DATASET EXPORT menu. A dialog box appears. We check the selected cells coordinates.



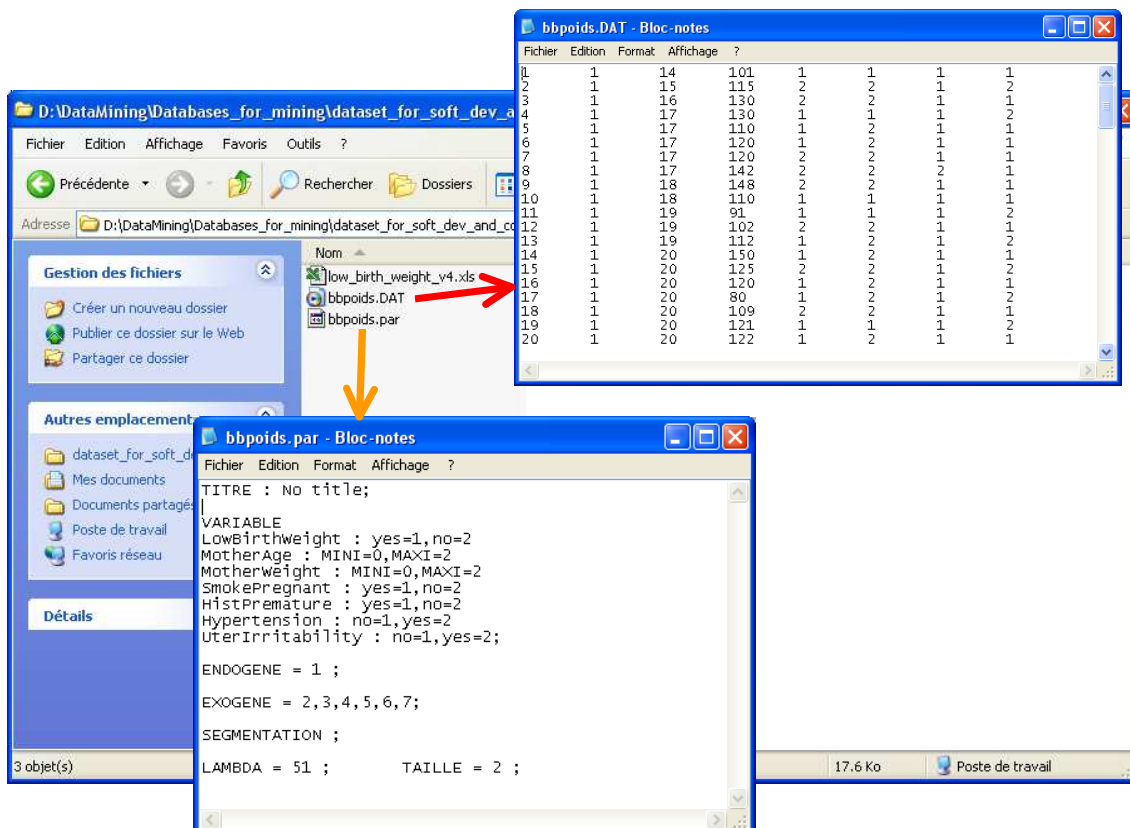
We click on the NEXT button. In the following tab, we must specify the target attribute. This is LOWBIRTHWEIGHT. The values of the attribute are automatically displayed.



Another dialog box allows to define the name of the data files. We set BBPOIDS.



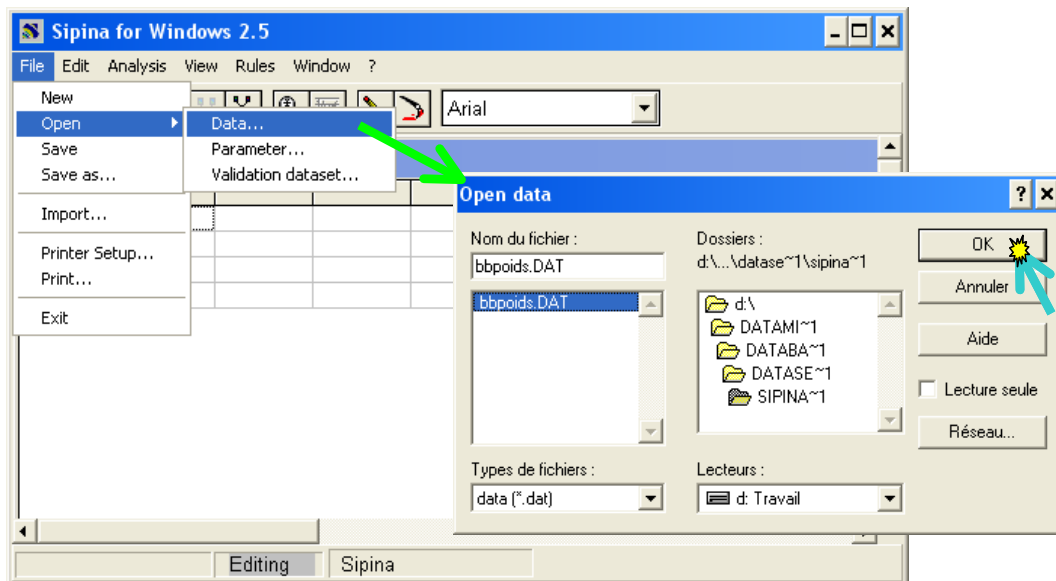
We visualize the two files generated by DATA MANAGER with Windows explorer.



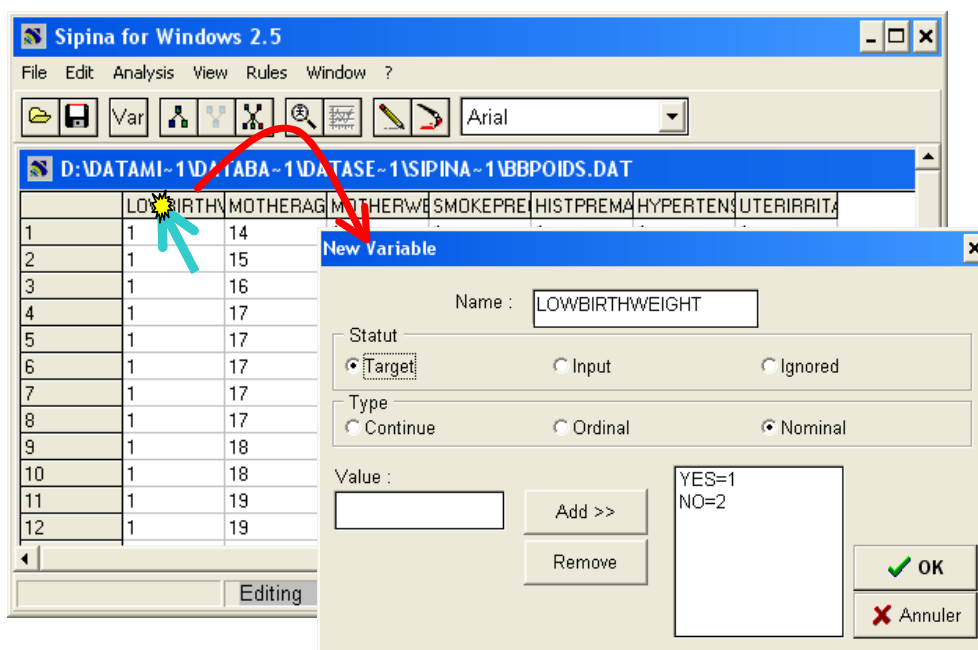
5 Utilization of SIPINA v2.5

5.1 Loading the dataset

We can now launch SIPINA version 2.5. We click on the Windows menu START / PROGRAMS / SIPINA 2.5 / SIPINA FOR WINDOWS. To importing the dataset, we click on the FILE / OPEN / DATA menu. A dialog box allows to select the file. *Caution: Sipina 2.5 uses the old 8.3 file name specification.*

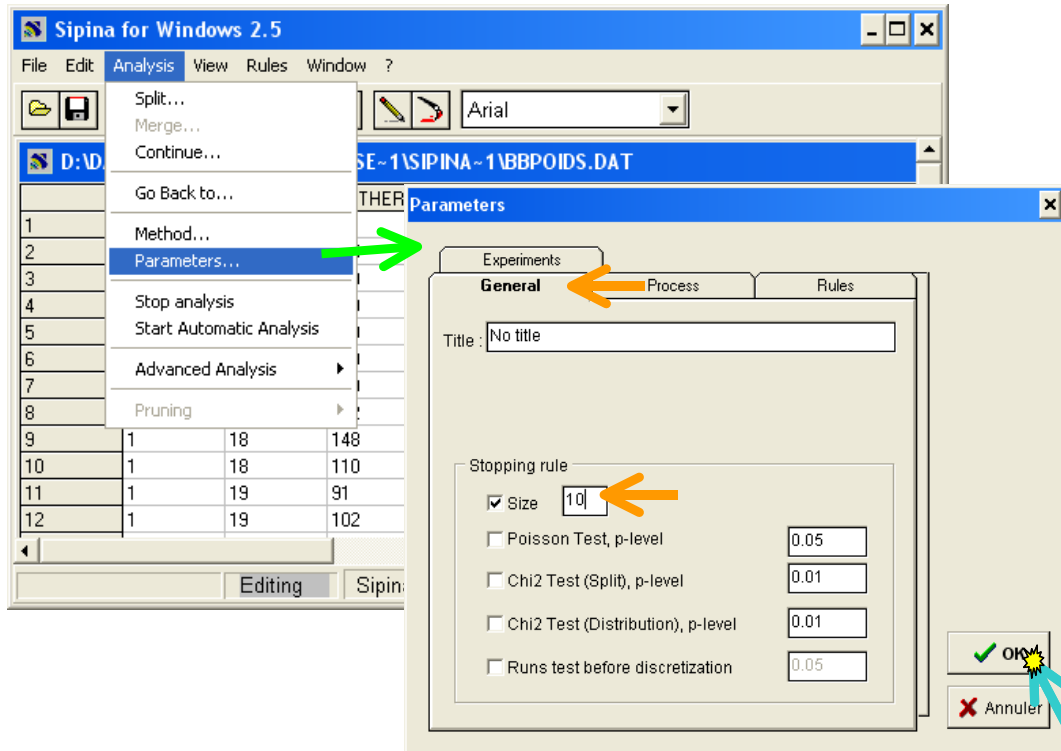


The two data files are handled together. We specify the DAT file into the dialog box. The tool imports automatically the PAR file. We can obtain the description of the attributes by double clicking the header of the columns. We observe the type of the attribute and its status for the analysis.



5.2 Parameters of the method

The SIPINA algorithm is automatically selected. To modify the settings of the approach, we click on the ANALYSIS / PARAMETERS menu. We activate the GENERAL tab.



There are many parameters for adjusting the behavior of the learning algorithm. We handle the simplest ones in this tutorial⁴. For instance, we set SIZE = 10 i.e. we do not want to generate a node with lower than 10 examples during the learning process. We confirm our choice by clicking on the OK button.

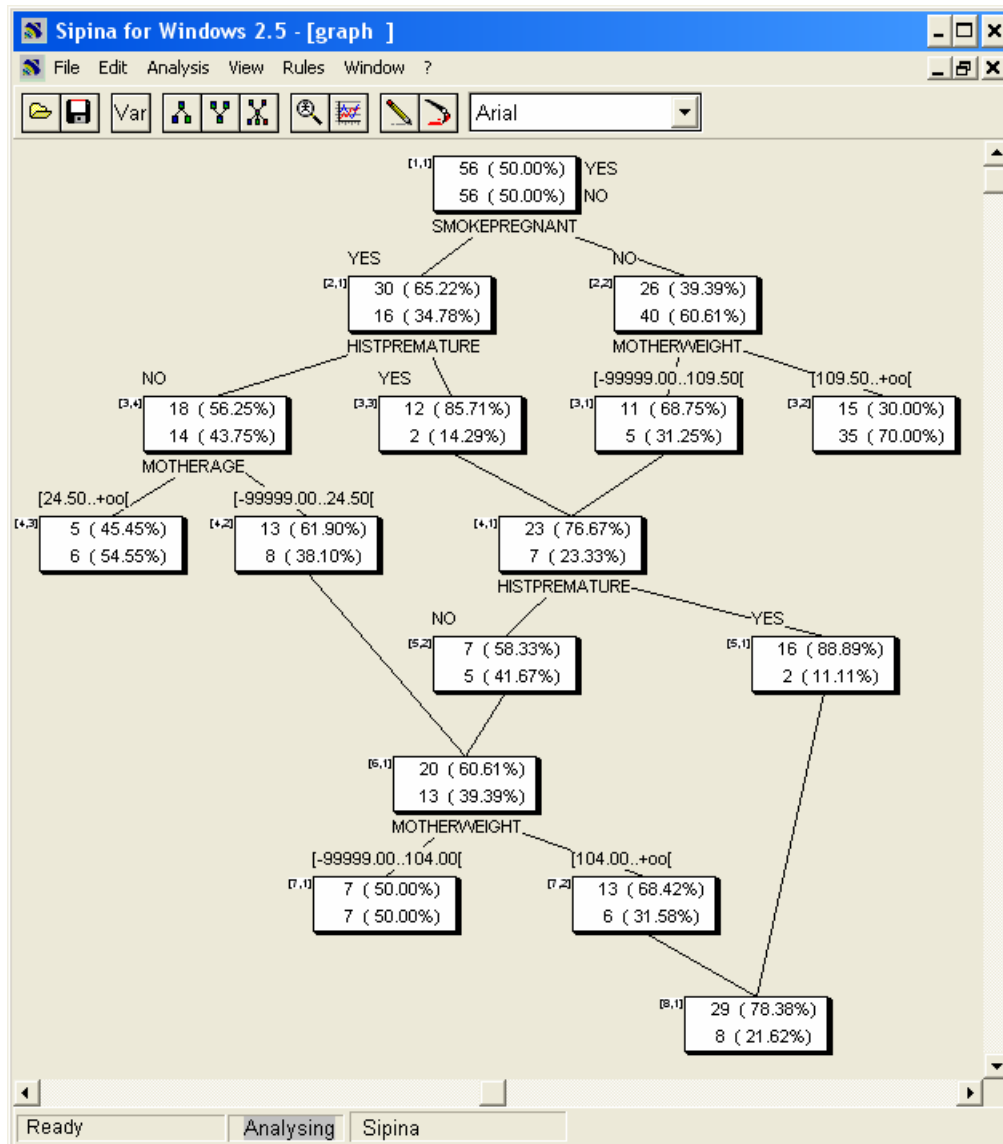
5.3 Decision graph construction

We can launch the analysis by clicking on the ANALYSIS / START AUTOMATIC ANALYSIS menu. A decision graph is created. The graph structure is rather confused. We can improve the display by arranging manually the nodes⁵.

We obtain the following decision graph.

⁴ Other settings are available. I will describe them in a more technical tutorial later.

⁵ We have tried various algorithms in order to reorganize automatically the graph. We have never found a satisfactory one. I do not know if this is possible. This is the reason for which we set an interactive functionality which allows the user to reorganize manually the nodes.



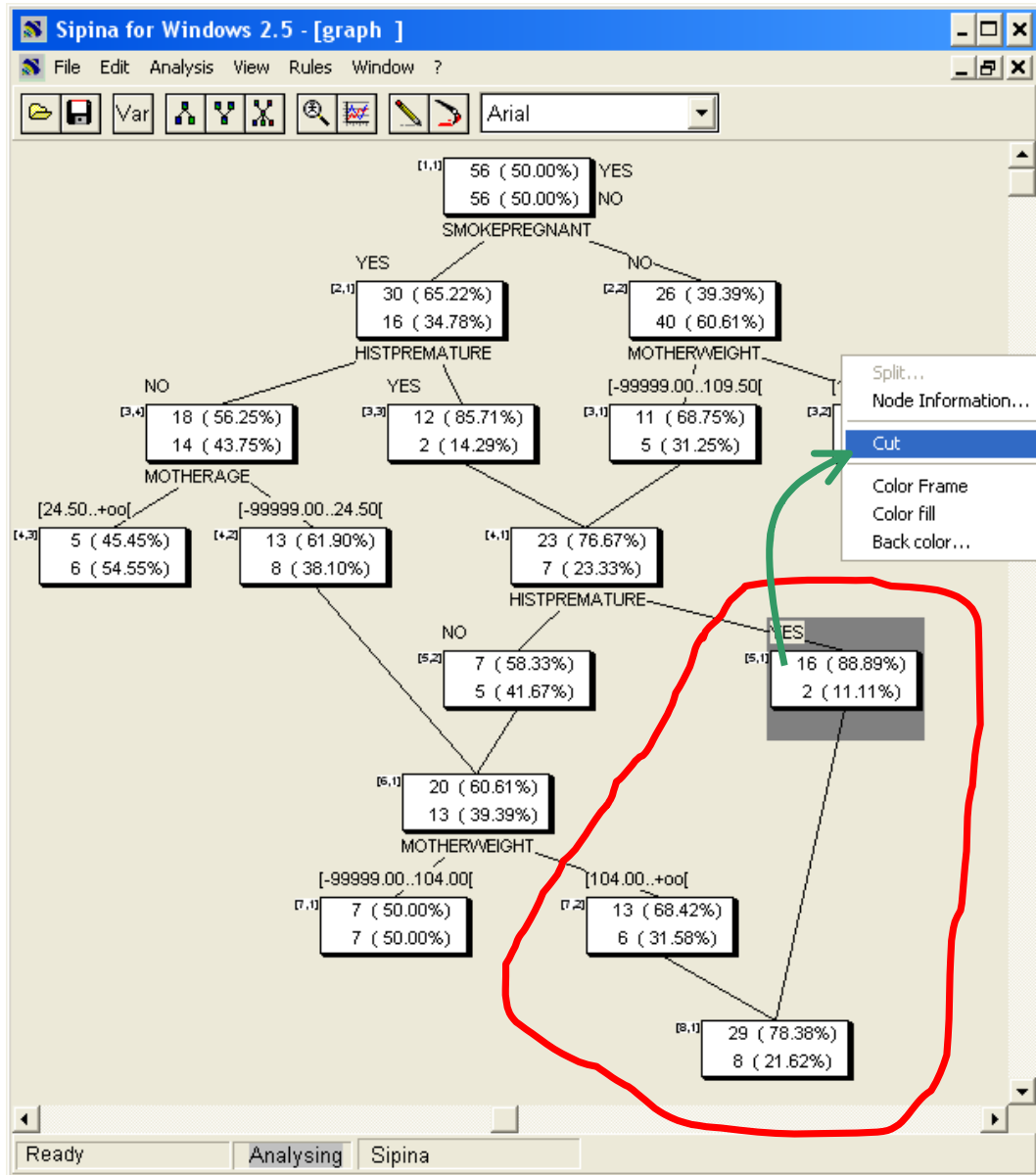
We note that it is really a decision graph. Both splitting and merging operations are succeeded during the construction of the structure.

5.4 Pruning the decision graph

Some merging operations are unnecessary into the graph structure. These are those which are not followed by a splitting operation in the bottom part of the graph. The terminal node is only a joining of two prediction rules with the same conclusion. We can remove the merged terminal node. It does not deteriorate the behavior of the classifier which is logically equivalent.

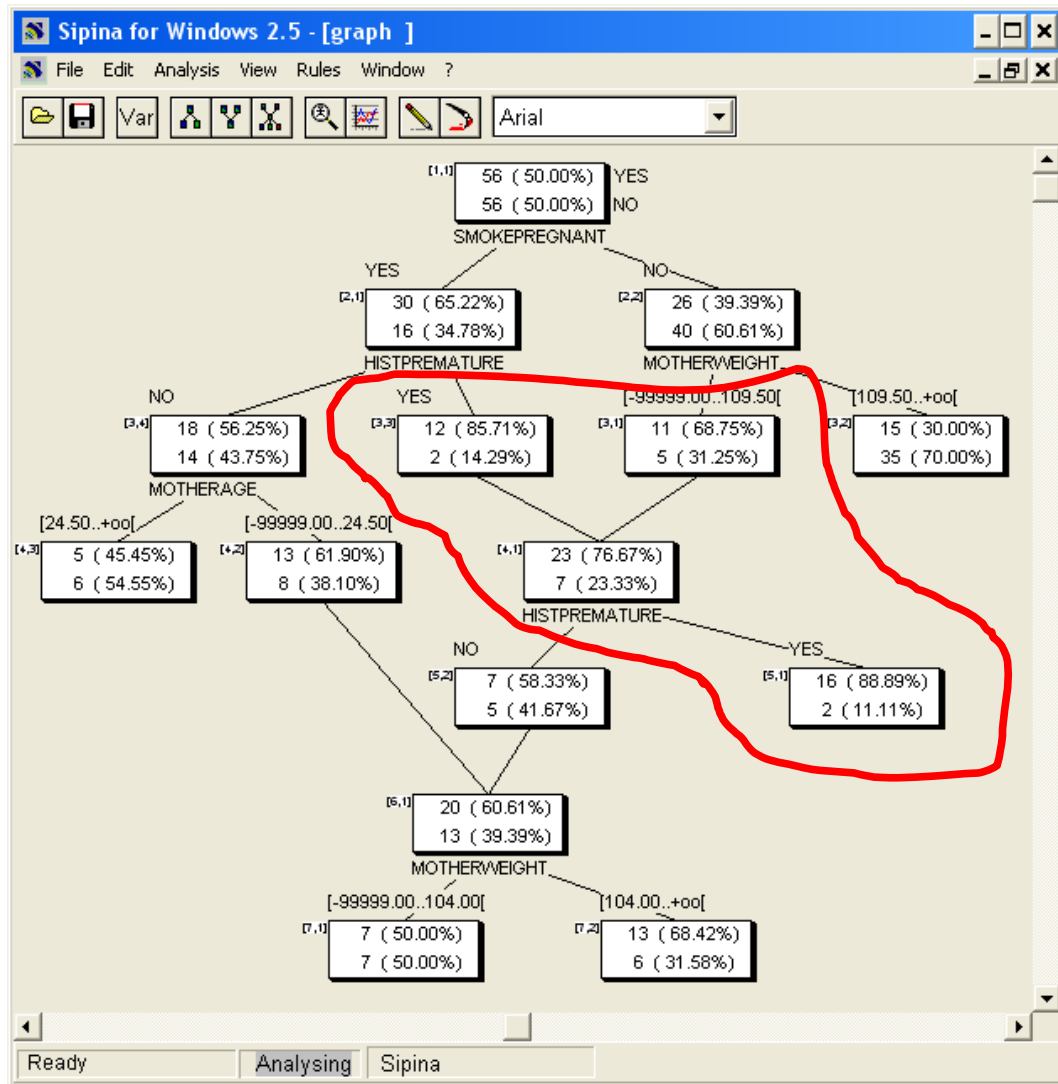
To pruning a bottom part of the graph, we select the preceding node and we activate the contextual menu with a right clicking. We select the CUT item.

Note: On the other hand, if a merging operation is followed of a splitting one. We cannot prune the decision graph without carefully analyzing its consequence on the performance of the classifier.



6 Advantages (and drawbacks) of the SIPINA approach

Let us consider a specific part of the learned decision graph in order to understand the main advantage of the decision graph in comparison with the standard decision tree.



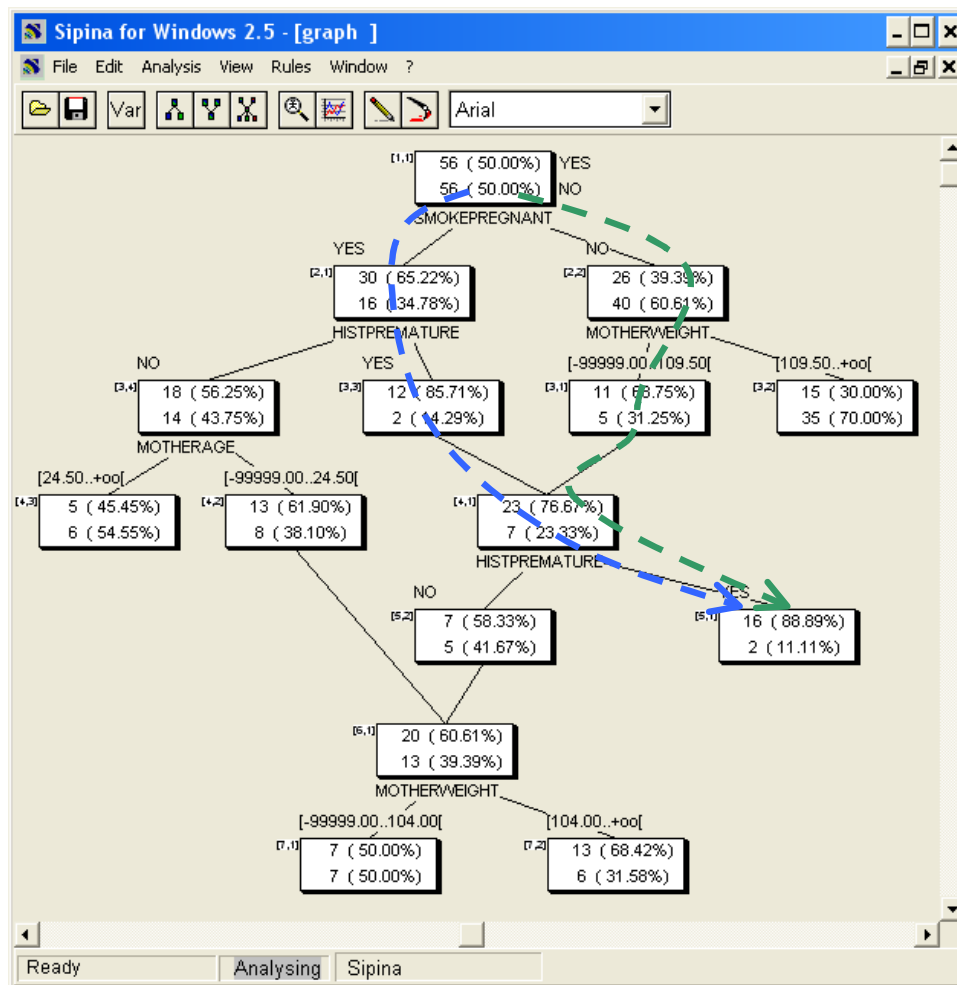
A method of induction of trees would have stopped at third level, with the leaves [3,3] and [3,1]. We would have obtained two prediction rules leading to the same conclusion. The support of the rule [3, 3] is $14/112 = 12.50\%$, its confidence $12/14 = 85.71\%$. For the rule [3, 1], we have respectively support = $16/112 = 14.29\%$ and confidence $11/16 = 68.75\%$.

But the SIPINA algorithm does not stop at this step. It performs a merging operation which groups the instances covered by these nodes. Then, it applies a splitting operation in order to obtain better prediction rules. Thus, if we consider the terminal node [5, 1], the support ($18/112 = 16.07\%$) and the confidence ($16/18 = 88.89\%$) are simultaneously better than those associated to the nodes [3, 3] and [3, 1].

This ability to more deeply explore the solutions is the main advantage of the decision graph compared with the decision tree. It is related to a more powerful representation bias. But this is also its main drawback. Indeed, a rule is now constituted from conjunctions and disjunctions of propositions. The reading of a rule is more complex.

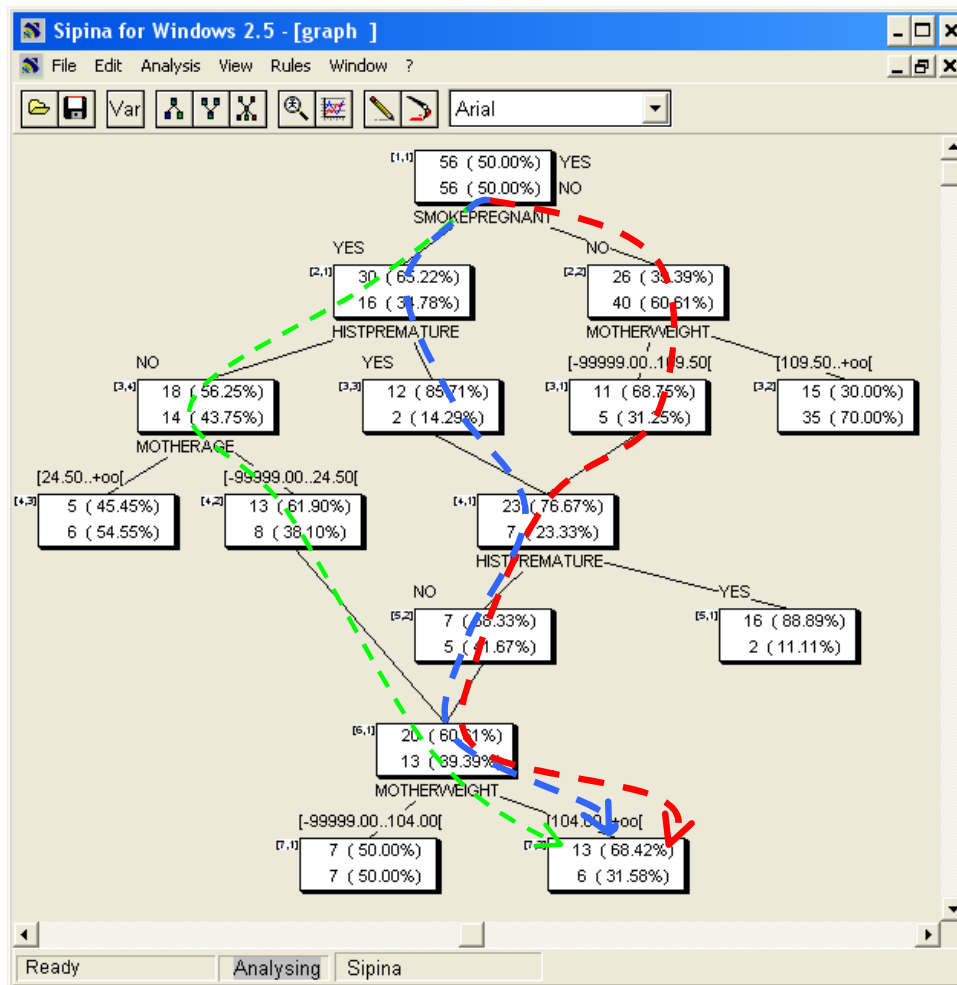
About the rule associated to the terminal node [5, 1], there are two paths from the root to this leaf. The rule is thus: « **IF** (SMOKEPREGNANT = YES **and** HISTPREMATURE = YES

and HISTPREMATURE = YES) OR (SMOKEPREGNANT = NO and MOTHERWEIGHT < 109.5 and HISTPREMATURE = YES) THEN LOWERBIRTHWEIGHT = YES ».



Of course, it is sometimes possible to make simplifications. In our example, the rule can be simplified by removing the redundant proposals. However we lose (a little bit) what makes the attraction of decision trees: proposing a classifier which is easy to interpret and manipulate.

If we consider the rule related to the terminal node [7, 2], the support is $19/112 = 16.96\%$ and the confidence $13/19 = 68.42\%$. We observe that there are three paths from the root to the leaf. The understanding of the classification rule becomes tedious.



7 Some functionalities of the SIPINA v2.5 tool

This 2.5 version incorporates other rather original functionalities, let's do a quick outline.

7.1 Extracting the rules from the graph

As seen above, the extraction of the rules from the graph is one of the major drawbacks of the decision graph. SIPINA supplies some tools to handle the rule easily. Especially, it can extract the rules defined by each path from the root to terminal nodes.

To do that, we click on the RULES / COMPUTE menu. A visualization window appears. We observe that 8 rules are extracted to the decision graph, whereas there are 6 terminal nodes into the structure. For each rule, SIPINA computes some statistical indicators.

The screenshot shows the Sipina for Windows 2.5 interface. The main window displays a decision tree with nodes for 'MOTHERAGE' and 'MOTHERWEIGHT'. A 'View rules' dialog box is open, showing a list of rules. A red arrow points from the 'Simplify' menu option to the 'View rules' dialog. The status bar at the bottom indicates 'Analysing Sipina'.

7.2 Simplifying the ruleset

More the classifier is complex with many rules, more its interpretation will be difficult. SPIRINA incorporates several procedures to simplify rule bases. The aim is to improve the readability of the model by simplifying the rules and decreasing their number, while maintaining their prediction performance.

For instance, we want to process the previous ruleset with the “C4.5 rules” algorithm (Quinlan, 1993). We note that this approach does not preserve the logical characteristics of the rule set. It uses the available dataset to simplify the rule base. In some circumstances, it allows to improve the generalization performance of the resulting classifier.

We click on the RULES / SIMPLIFY / C4.5 RULES menu. We set the file name of the generated rule set in the dialog box (BBPRUNE.KBA). A new visualization window appears. We note that there are only 4 rules now, they are also more concise.

The screenshot displays the Sipina for Windows 2.5 software interface. The main window shows a decision tree with nodes for 'NO', 'MOTHERAGE', and 'HISTPREMATURE'. A 'Simplify rules File' dialog box is open, showing the file name 'bbprune.kba' and the directory 'd:\...dabase~1\sipina~1'. A 'View rules' window is also open, displaying the following rules:

```

HISTPREMATURE=1 and SMOKEPREGNANT=1 then LOWBIRTHWEIGHT=YES with <0.86#14#0.115589#0
SMOKEPREGNANT=1 and MOTHERAGE=[-99999.00..24.50[ then LOWBIRTHWEIGHT=YES with <0.70#30#
SMOKEPREGNANT=2 and MOTHERWEIGHT=[109.50..+oo[ then LOWBIRTHWEIGHT=NO with <0.70#50#0.
HISTPREMATURE=2 and MOTHERAGE=[24.50..+oo[ then LOWBIRTHWEIGHT=NO with <0.62#26#0.076590
  
```

The 'View rules' window also includes a legend: 'if Condition then Conclusion with <1-error rate# size# j-Measure# 1-p-value test>' and an 'OK' button.

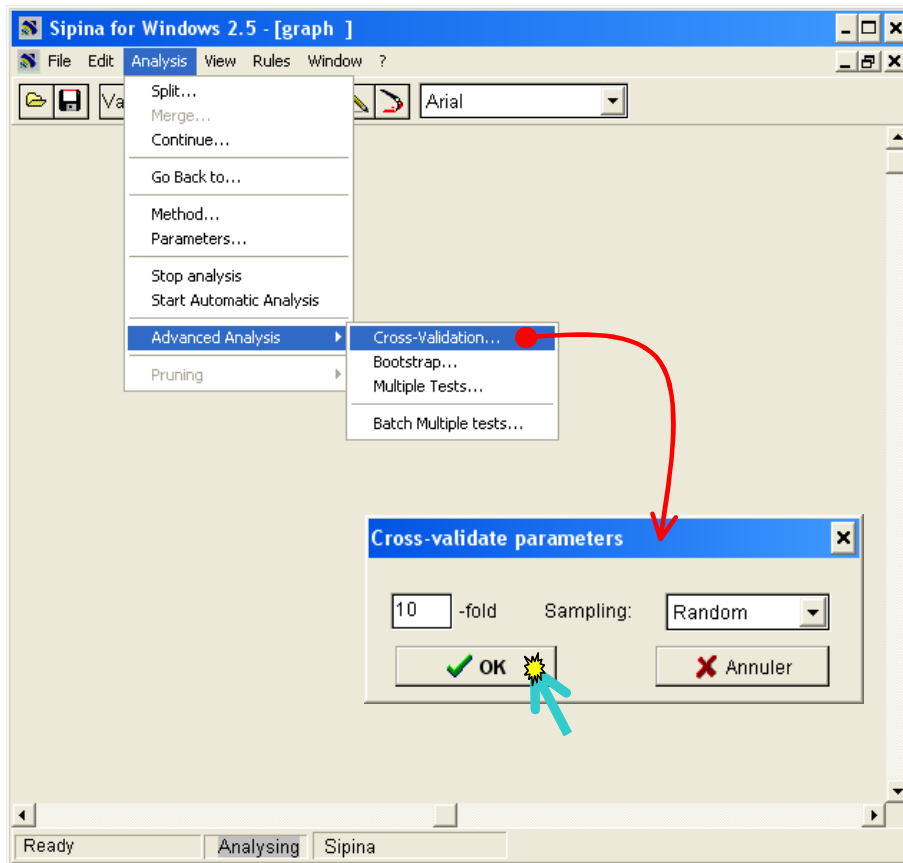
7.3 Cross validation

We can implement a K-fold cross validation for measuring the accuracy rate⁶.

We stop the current session by clicking on the ANALYSIS / STOP ANALYSIS menu. Then, we activate ANALYSIS / ADVANCED ANALYSIS / CROSS-VALIDATION.

A dialog box appears. We set the number of folds: FOLD = 10.

⁶ http://en.wikipedia.org/wiki/Cross-validation_%28statistics%29



The generated decision graphs during the process are successively displayed. Lastly, the results are summarized in a new visualization window. The estimated accuracy rate is 62%.

Samples	1	2	3	4	5	6	7	8	9	10
Accuracy	0.75	0.50	0.73	0.60	0.60	0.50	0.75	0.36	0.82	0.56
Size Learning	101	101	101	101	101	101	101	101	101	99
Size Test	11	11	11	11	11	11	11	11	11	13
Unclassified	3	1	0	1	1	5	3	0	0	4
Rule base size	9	5	5	8	6	7	7	5	6	5
Size Pruning	0	0	0	0	0	0	0	0	0	0

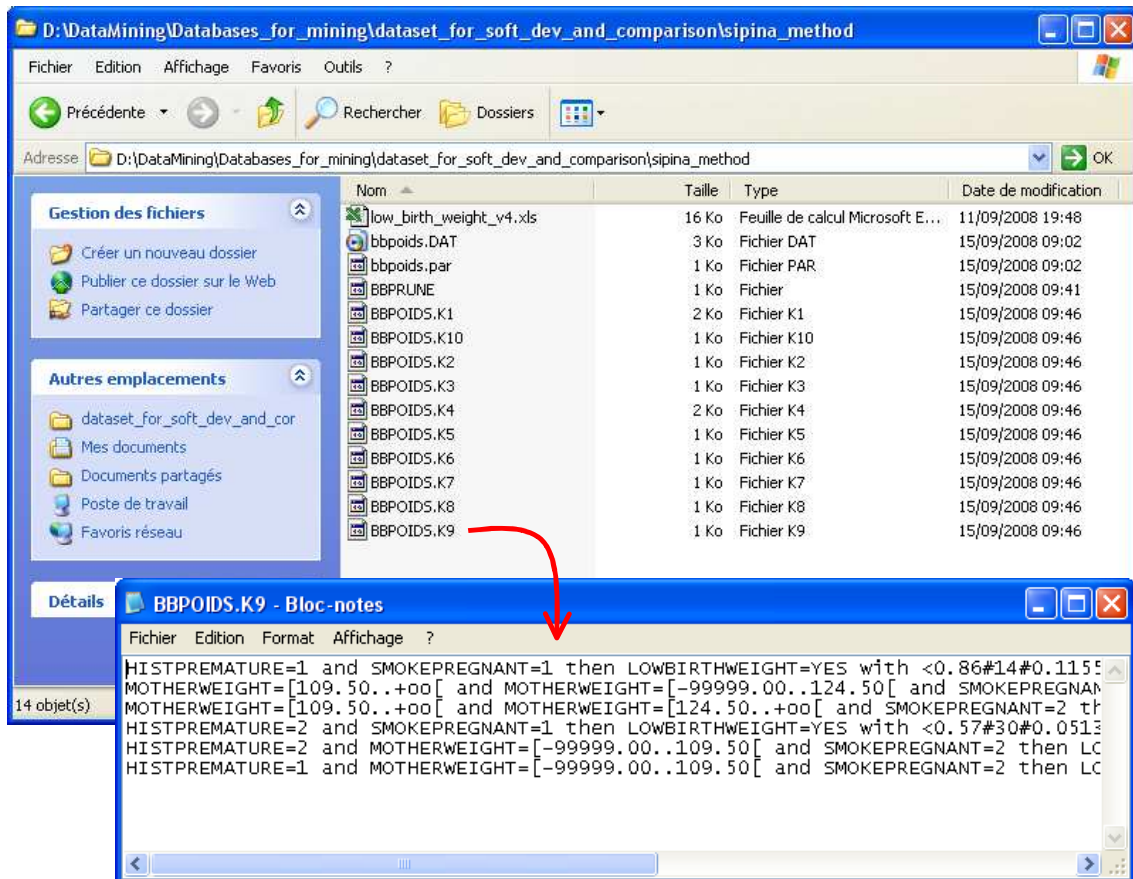
Accuracy
 Mean : 0.62
 Std Deviation : 0.14

Number of rules
 Mean : 6.30
 Std Deviation : 1.35

Computing time : 1750 ms

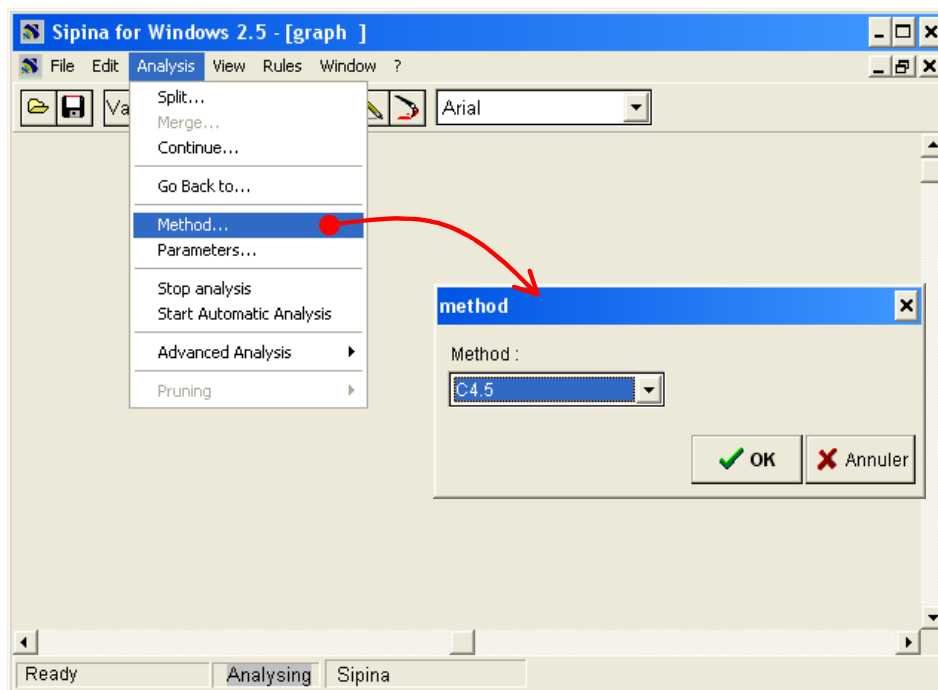
OK

Each rule base is numbered in accordance with the associated cross-validation session. For instance, ruleset file name of the treatment of the 9-th fold during the cross validation is "BBPOIDS.K9".

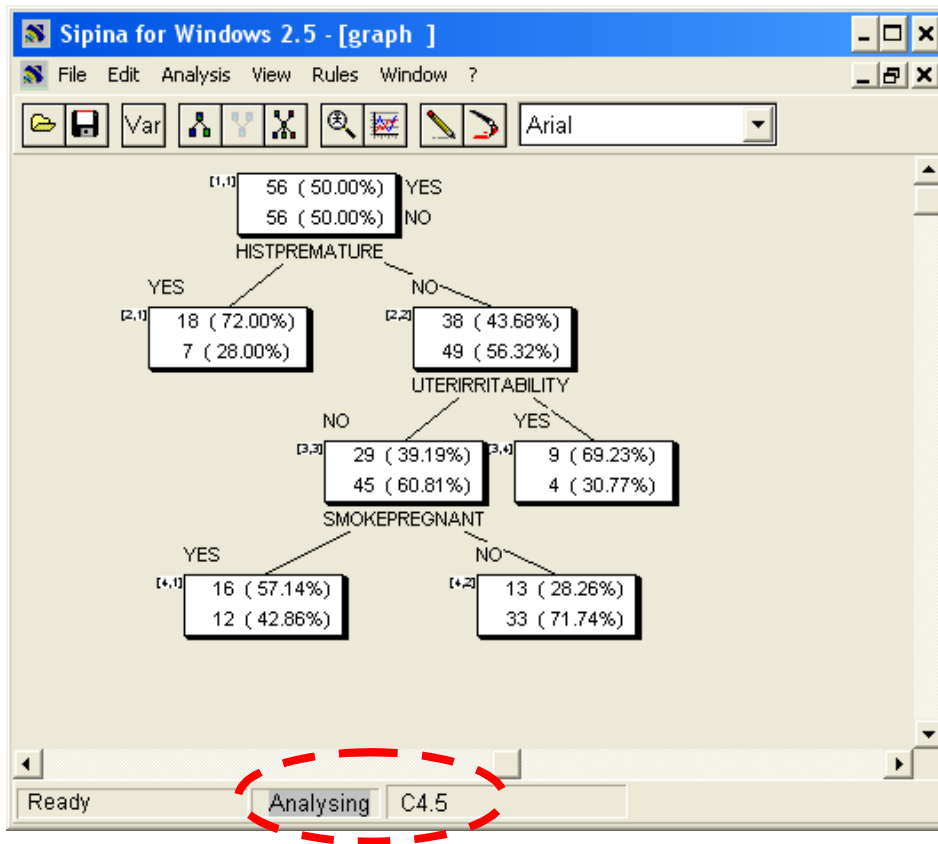


7.4 Implementing a decision tree algorithm

SIPINA version 2.5 provides other learning algorithms such as C4.5 (Quinlan, 1993). To modify the current method, we click on the ANALYSIS / METHOD menu. We select **C4.5**.



The used algorithm is displayed into the status bar. We click on the ANALYSIS / START AUTOMATIC ANALYSIS menu to launch the decision tree construction.



The decision tree contains 4 leaves i.e. 4 rules. It is clearly simpler than the classifier induced by Sipina (8 rules).

We perform again the cross validation to assess the performance of C4.5. We stop the current analysis (ANALYSIS / STOP ANALYSIS). Then we perform the 10-fold cross validation (ANALYSIS / ADVANCED ANALYSIS / CROSS-VALIDATION). The cross-validation accuracy rate is 66%. On this dataset, C4.5 is slightly better than SIPINA with fewer rules.

Results

Samples	1	2	3	4	5	6	7	8	9	10
Accuracy	0.73	0.64	0.73	0.64	0.55	0.73	0.73	0.55	0.82	0.54
Size Learning	101	101	101	101	101	101	101	101	101	99
Size Test	11	11	11	11	11	11	11	11	11	13
Unclassified	0	0	0	0	0	0	0	0	0	0
Rule base size	4	4	3	2	3	3	3	3	3	4
Size Pruning	0	0	0	0	0	0	0	0	0	0

Accuracy: Mean: 0.66, Std Deviation: 0.09
 Number of rules: Mean: 3.20, Std Deviation: 0.60
 Computing time: 1109 ms

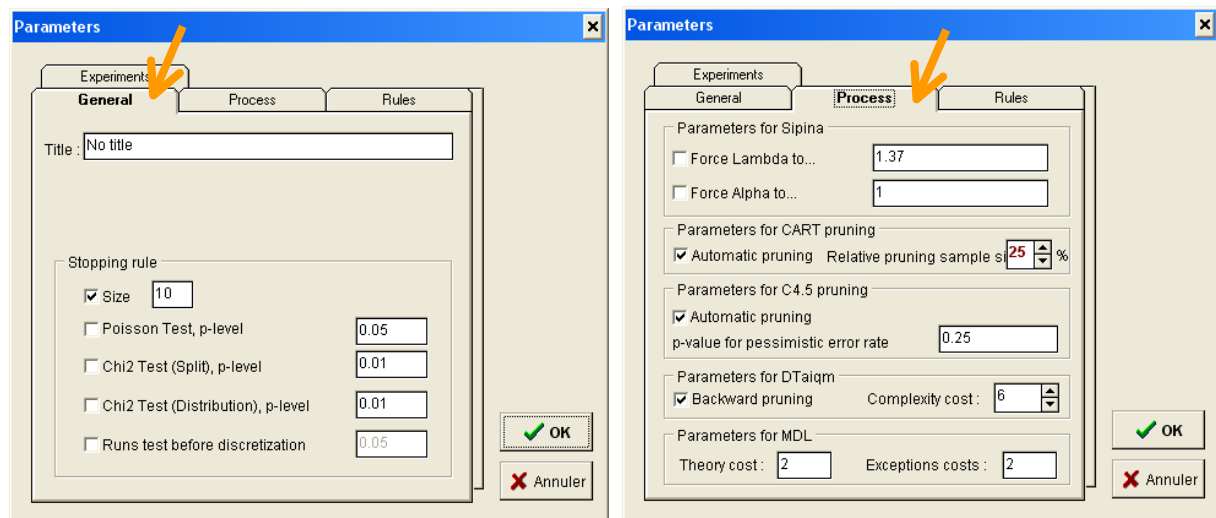
OK

Of course, this type of experiment is certainly not evidence about the general behavior of methods. The idea is mainly to show how to compare SIPINA to another method on a dataset, by using exclusively a performance criterion.

8 Conclusion

SIPINA version 2.5 is the only software, to my knowledge, which incorporates the SIPINA method, and more generally a decision graph algorithm. For this reason it is still available on our website even if its development was stopped long ago now. This version incorporates also some decision tree methods unobtainable elsewhere (Wehenkel, 1993; Elisee; etc..).

Like all learning methods, some parameters allows to refined the behavior of SIPINA. They are accessible via the menu ANALYSIS / PARAMETERS. They are used to adjust the search bias according to the goal of analysis and the data characteristics. **It is even possible to set parameters so that the method builds only decision trees.**



9 References

- D. Zighed, J.P. Auray, G. Duru, *SIPINA : Méthode et logiciel*, Lacassagne, 1992 (in French).
- J. Oliver, *Decision Graphs: An extension of Decision Trees*, in Proc. of Int. Conf. on Artificial Intelligence and Statistics, 1993.
- R. Quinlan, *C4.5 : Programs for Machine Learning*, Morgan Kaufmann, 1993.
- R. Rakotomalala, *Graphes d'induction*, Thèse de Doctorat, Université Lyon 1, 1997 (URL : <http://eric.univ-lyon2.fr/~ricco/publications.html> ; in french).
- D. Zighed, R. Rakotomalala, *Graphes d'induction : Apprentissage et Data Mining*, Hermès, 2000 (in French).
- M. Tenenhaus, *Statistique – Méthodes pour décrire, expliquer et prévoir*, Dunod, 2007; pages 540 à 545 (in French).