

# 1 Topic

## Creating reports with Tanagra

The ability to create automatically reports from the results of an analysis is a valuable functionality for Data Mining. But this is rather an asset to the professional tools. The programming of this kind of functionality is not really promoted in the academic domain. I do not think that I can publish a paper in a journal where I describe the ability of Tanagra to create attractive reports. This is the reason for which the output of the academic tools, such as R or Weka, is mainly in a formatted text shape.

Tanagra, which is an academic tool, provides also text outputs. The programming remains simple if we see at a glance the source code. But, in order to make the presentation more attractive, it uses the HTML to format the results. I take advantage of this special feature to generate reports **without making a particular programming effort**. Tanagra is one of the few academic tools to be able to produce reports that can easily be displayed in office automation software. For instances, the tables can be copied into Excel spreadsheets for further calculations. More generally, the results can be viewed in a browser, regardless of data mining software.

These are the reporting features of Tanagra that we present in this tutorial.

## 2 Dataset

We use the « [heart\\_disease\\_male\\_for\\_reporting.xls](#) » dataset. We want to predict the presence or absence of DISEASE from the characteristics of patients. We perform the following tasks: descriptive statistics to compare the two groups (presence vs. absence of disease); subdividing the dataset into a learning sample and a test sample; creating and assessing a decision tree; creating and assessing a logistic regression, for which we recode some descriptors; constructing the ROC curve from the results of the logistic regression. All the results are described in a report that we can visualize in a browser, regardless of Tanagra.

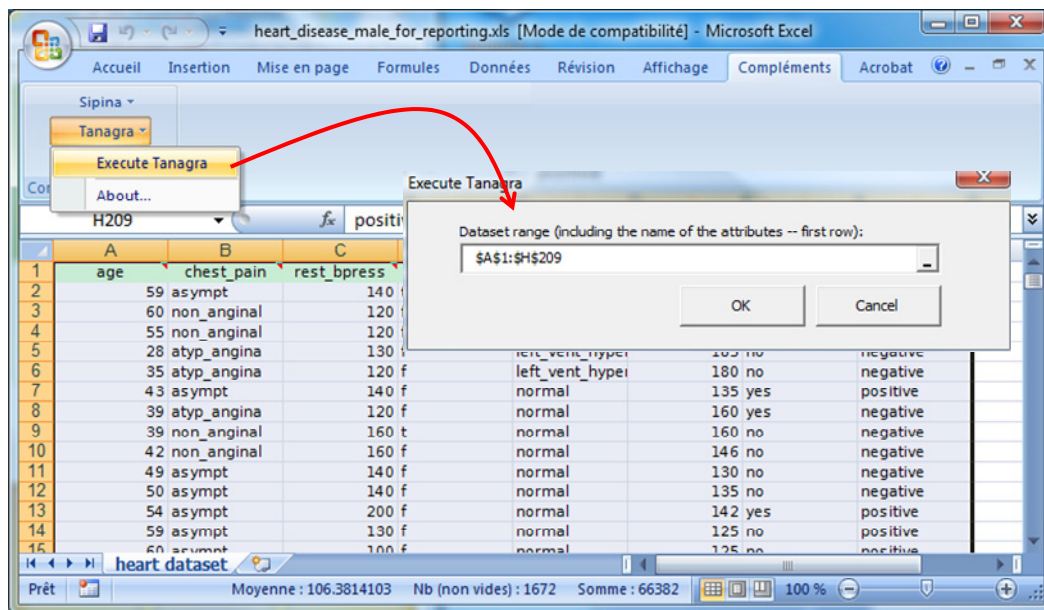
## 3 Creating report with Tanagra

### 3.1 Loading the data file

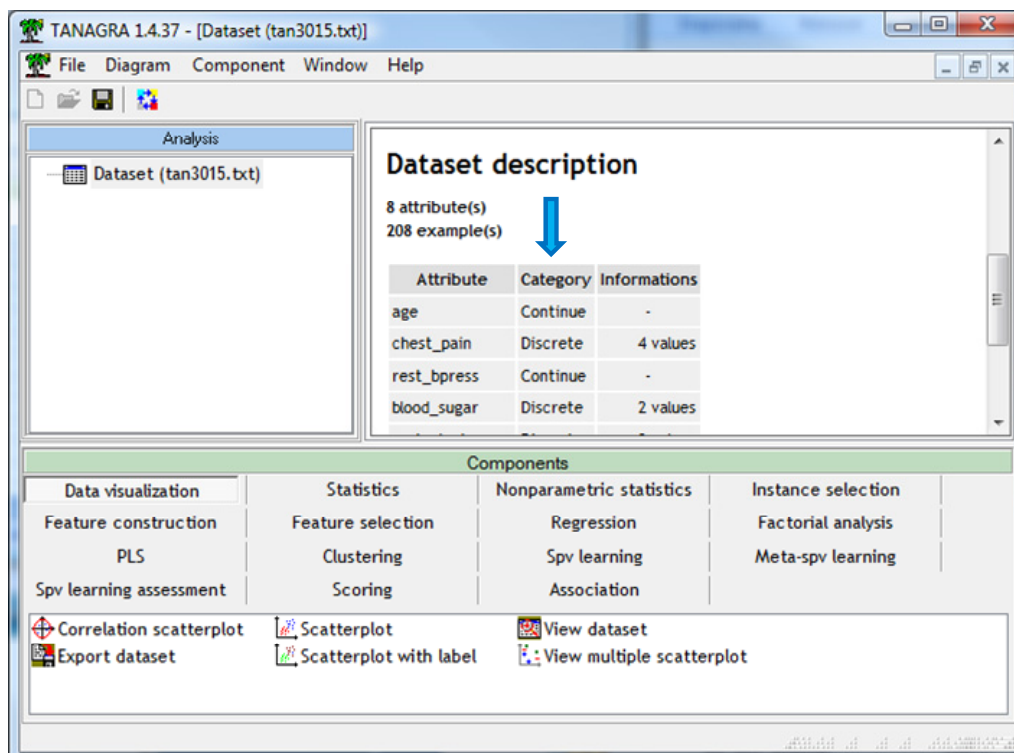
First, we want to import the data file into Tanagra. The easiest way to do that is to open the file into Excel. Then, using the Tanagra add-in, we send the dataset to Tanagra (COMPLEMENTS / TANAGRA / EXECUTE TANAGRA in the French version of Excel)<sup>1</sup>.

---

<sup>1</sup> See <http://data-mining-tutorials.blogspot.com/2010/08/tanagra-add-in-for-office-2007-and.html> for further description of the installation and utilization of the add-in (Excel 2007 and 2010). We describe the process for Excel 2003 and previous version: <http://data-mining-tutorials.blogspot.com/2008/10/excel-file-handling-using-add-in.html>. A similar tool is available for Open Office: <http://data-mining-tutorials.blogspot.com/2008/10/ooocalc-file-handling-using-add-in.html>



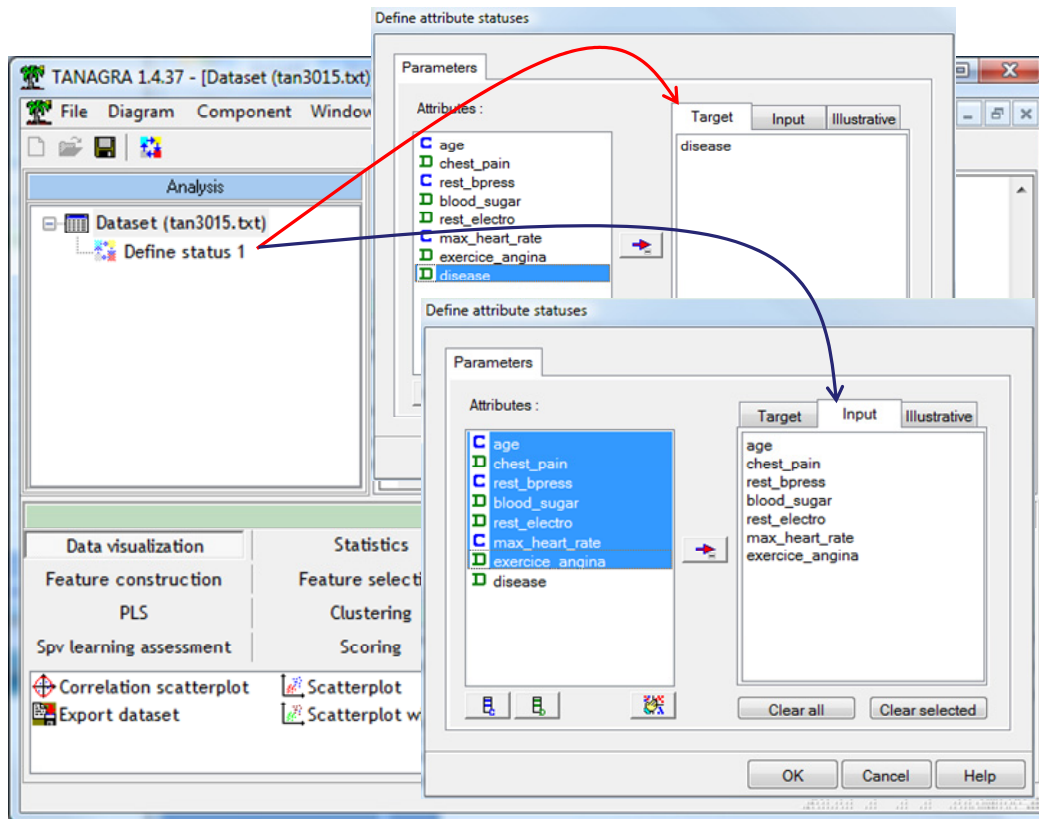
Tanagra is automatically launched. We check that we have 208 instances and 8 variables. We note that Tanagra detects automatically the type of the variables.



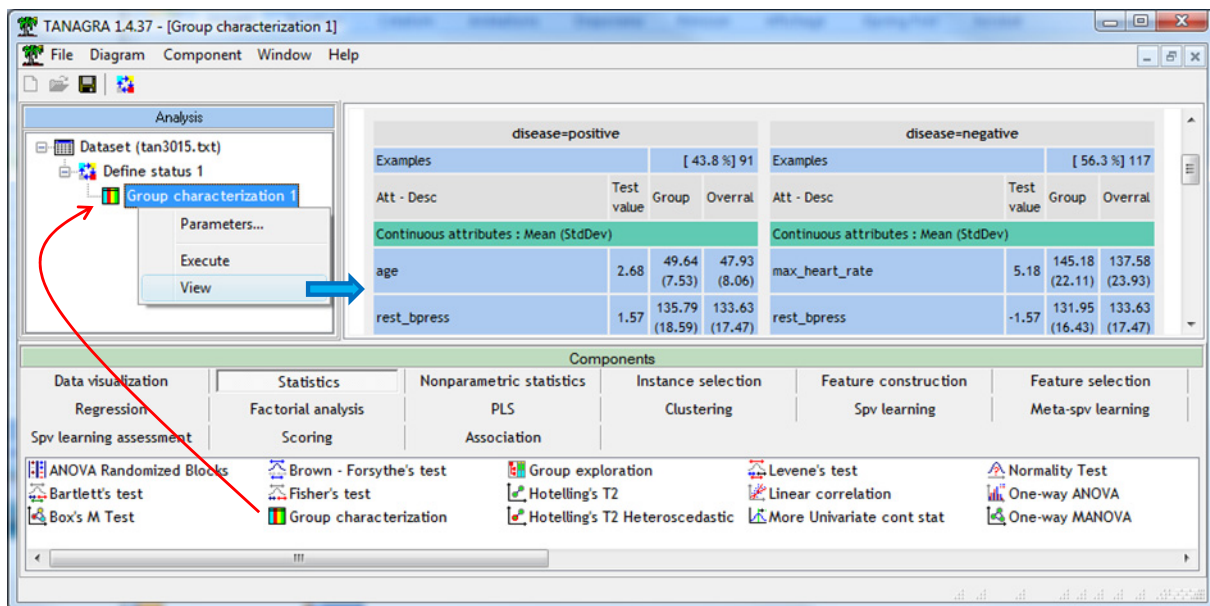
### 3.2 Analyzing the groups using descriptive statistics

In the supervised learning framework, computing the descriptive statistics of the descriptors according to the group membership is often informative. These are univariate descriptions of course. But I think it gives some ideas about the results obtained later.

We insert the DEFINE STATUS component into the diagram. We set DISEASE as TARGET, the other variables as INPUT.



Then, we add the GROUP CHARACTERIZATION component (STATISTICS tab) into the diagram.



We obtain the following results.

Results							
Description of "disease"							
disease=positive				disease=negative			
Examples [ 43.8 %] 91				Examples [ 56.3 %] 117			
Att - Desc	Test value	Group	Overall	Att - Desc	Test value	Group	Overall
Continuous attributes : Mean (StdDev)				Continuous attributes : Mean (StdDev)			
age	2.68	49.64 (7.53)	47.93 (8.06)	max_heart_rate	5.18	145.18 (22.11)	137.58 (23.93)
rest_bpress	1.57	135.79 (18.59)	133.63 (17.47)	rest_bpress	-1.57	131.95 (16.43)	133.63 (17.47)
max_heart_rate	-5.18	127.81 (22.72)	137.58 (23.93)	age	-2.68	46.61 (8.24)	47.93 (8.06)
Discrete attributes : [Recall] Accuracy				Discrete attributes : [Recall] Accuracy			
chest_pain=asympt	8.47	[ 73.5 %] 82.4 %	49.0 %	exercice_angina=no	8.35	[ 77.2 %] 89.7 %	65.4 %
exercice_angina=yes	8.35	[ 83.3 %] 65.9 %	34.6 %	chest_pain=atyp_angina	6.75	[ 90.8 %] 50.4 %	31.3 %
blood_sugar=t	2.09	[ 68.8 %] 12.1 %	7.7 %	chest_pain=non_anginal	3.22	[ 80.6 %] 24.8 %	17.3 %
rest_electro=st_t_wave_abnormality	1.54	[ 56.7 %] 18.7 %	14.4 %	blood_sugar=f	2.09	[ 58.3 %] 95.7 %	92.3 %
chest_pain=typ_angina	0.74	[ 60.0 %] 3.3 %	2.4 %	rest_electro=left_vent_hyper	1.08	[ 80.0 %] 3.4 %	2.4 %
rest_electro=normal	-1.00	[ 42.2 %] 80.2 %	83.2 %	rest_electro=normal	1.00	[ 57.8 %] 85.5 %	83.2 %
rest_electro=left_vent_hyper	-1.08	[ 20.0 %] 1.1 %	2.4 %	chest_pain=typ_angina	-0.74	[ 40.0 %] 1.7 %	2.4 %
blood_sugar=f	-2.09	[ 41.7 %] 87.9 %	92.3 %	rest_electro=st_t_wave_abnormality	-1.54	[ 43.3 %] 11.1 %	14.4 %
chest_pain=non_anginal	-3.22	[ 19.4 %] 7.7 %	17.3 %	blood_sugar=t	-2.09	[ 31.3 %] 4.3 %	7.7 %
chest_pain=atyp_angina	-6.75	[ 9.2 %] 6.6 %	31.3 %	exercice_angina=yes	-8.35	[ 16.7 %] 10.3 %	34.6 %
exercice_angina=no	-8.35	[ 22.8 %] 34.1 %	65.4 %	chest_pain=asympt	-8.47	[ 26.5 %] 23.1 %	49.0 %

Among the diseased individuals (DISEASE = POSITIVE), the mean of age is higher (49.64 vs. 47.93 in the whole dataset)<sup>2</sup>. The MAX\_HEART\_RATE is lower on the other hand (127.81 vs. 137.58).

About the categorical predictors, we observe an overrepresentation of CHEST\_PAIN = ASYMP (82.4% of the instances have this feature in this group vs. 49% in the whole dataset). This is the case also of EXERCICE\_ANGINA = YES (65.9% vs. 34.6%). Conversely, we observe an underrepresentation of CHEST\_PAIN = ATYP\_ANGINA and EXERCICE\_ANGINA = NO.

We observe the opposite characteristics for the healthy individuals (DISEASE = ABSENCE).

It is therefore possible to obtain a real differentiation between the groups. It is foreseeable that variables highlighted here will play an important role in the subsequent modeling process (decision tree, logistic regression).

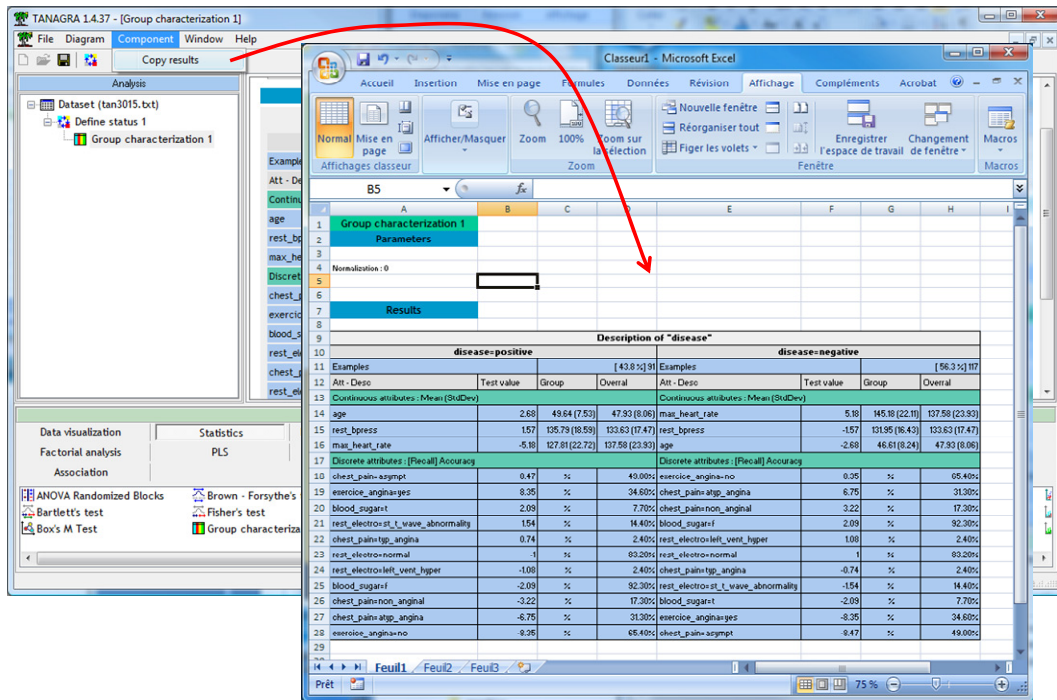
### 3.3 Copying the results into Excel

The previous table, if we know how to read it, is very interesting. We must be able to recover it easily in order to incorporate these values into a report. Using the standardized HTML format is of great help here. Indeed it is recognized by the vast majority of editing tools.

Into Tanagra, we click on the COMPONENT / COPY RESULTS menu. We paste the table into a new worksheet (CTRL + V). The values are properly entered into cells. The table structure is kept.

Of course, from Excel, we can incorporate any kind of information into another document.

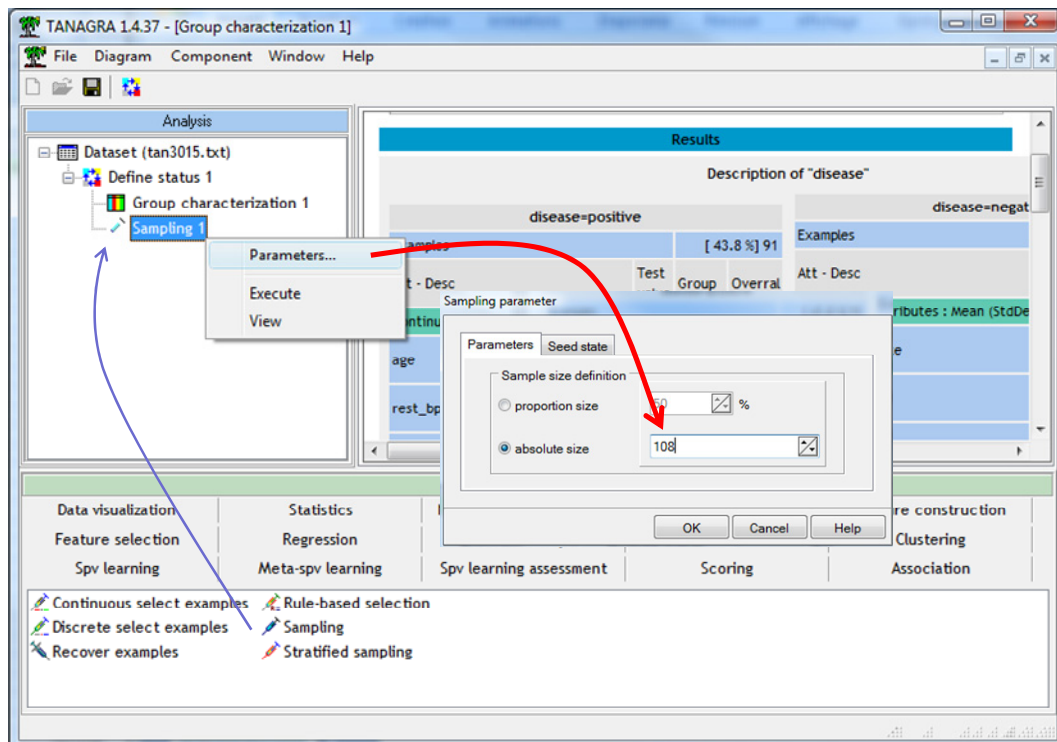
<sup>2</sup> See <http://data-mining-tutorials.blogspot.com/2009/05/understanding-test-value-criterion.html> for the reading of this table and the interpretation of the test value criterion.



### 3.4 Creating the learning and the testing samples

We want to learn a predictive model from the dataset. To obtain an unbiased evaluation of its performance, we subdivide the available dataset into learning sample and test sample. The model is created from the first sample, and it is assessed on the second one.

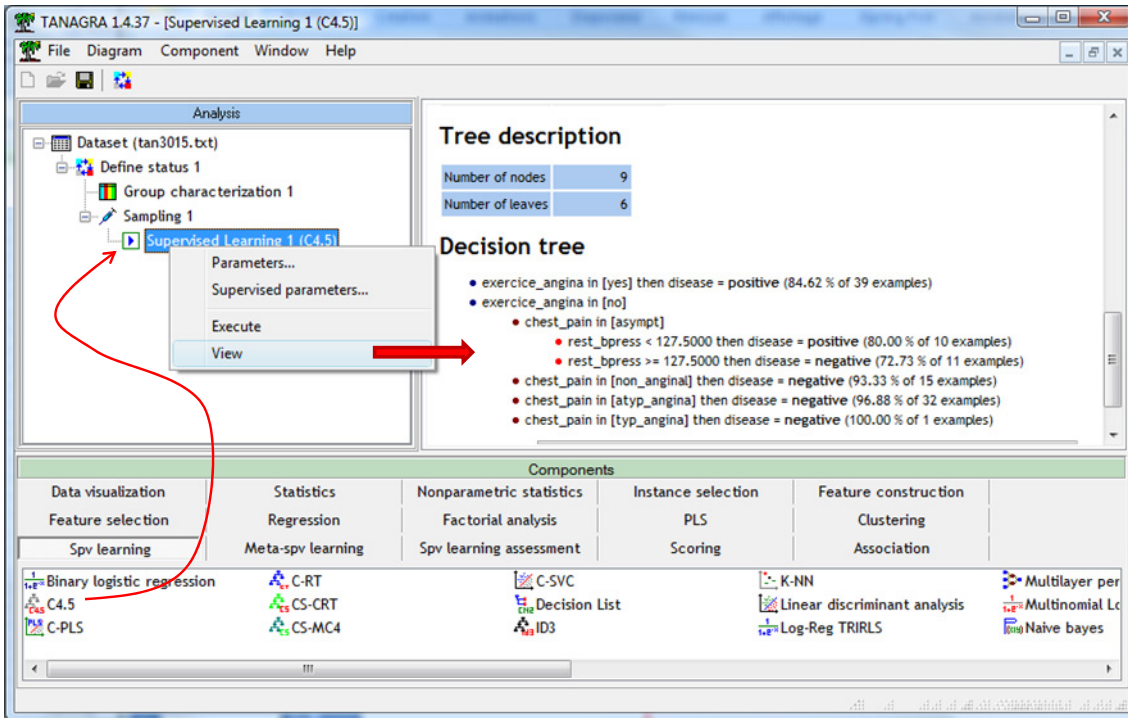
We insert the SAMPLING component (INSTANCE SELECTION tab) into the diagram. We click on the contextual PARAMETERS menu. We select 108 instances for the training sample.



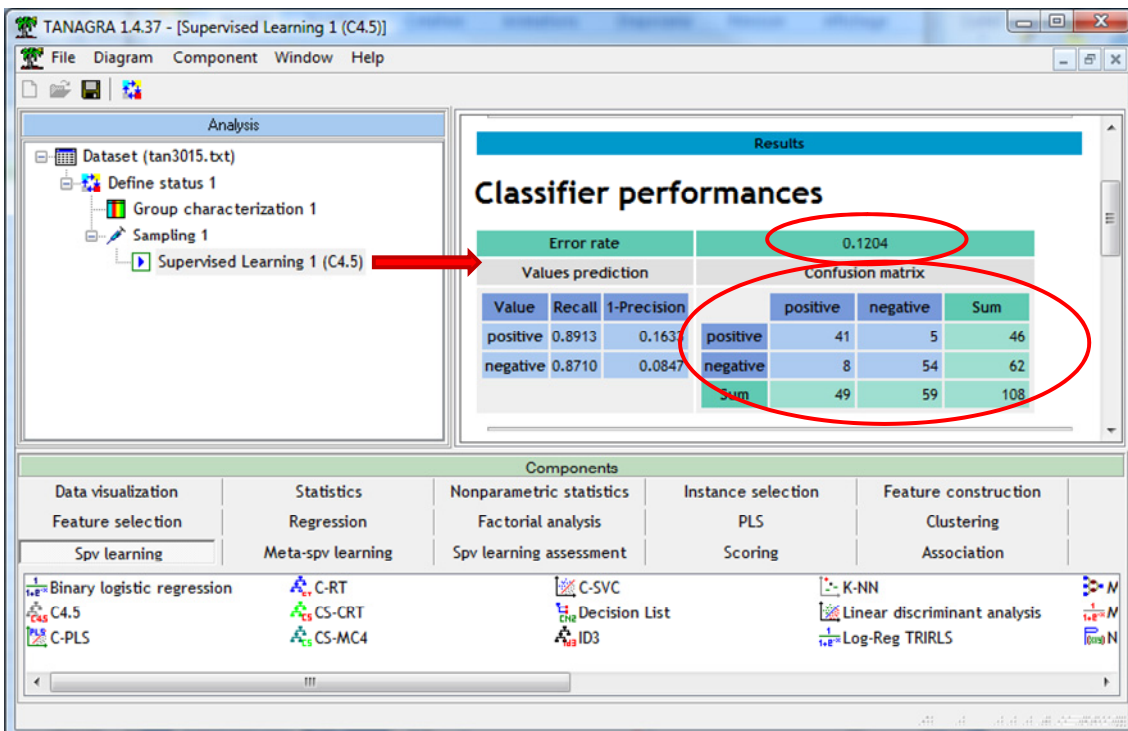
We confirm the settings by clicking on the OK button. Then we click on the VIEW menu to perform the sampling.

### 3.5 Decision tree induction with C4.5

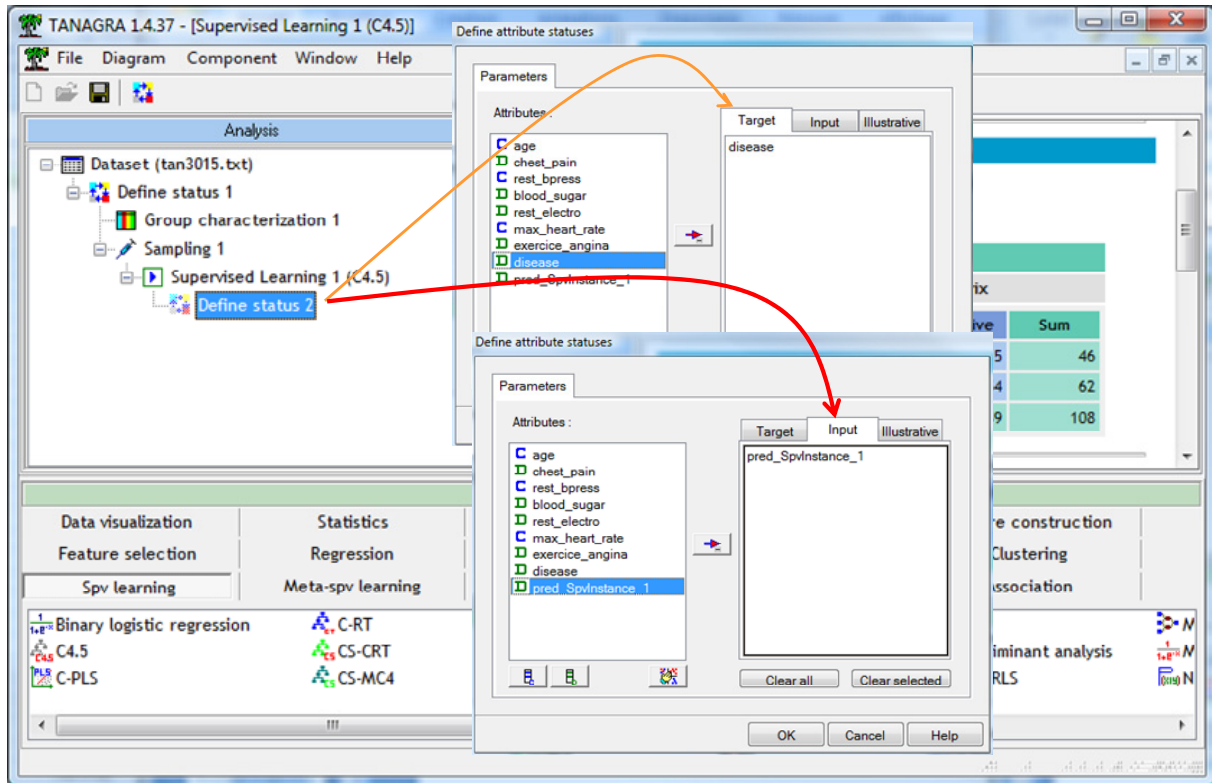
**Training step.** We want to create a decision tree using the C4.5 approach (Quinlan, 1993) (SPV LEARNING tab). We add the component into the diagram and we click on the VIEW menu. We obtain the following tree. Actually, CHEST\_PAIN and EXERCICE\_ANGINA are incorporated into the model. REST\_BPRESS, which is not highlighted in a univariate way, is used into the model. Detecting the interactions between the predictors is the main advantage of multivariate techniques.



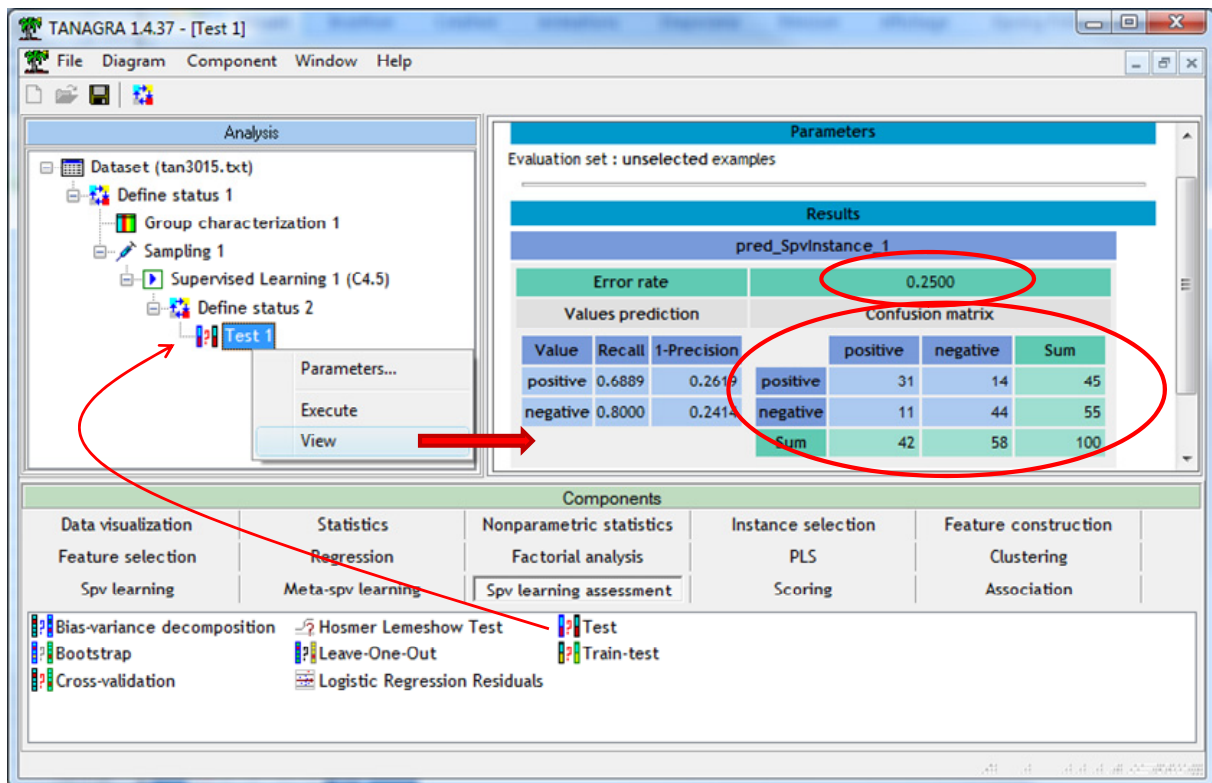
**Testing step.** The resubstitution error rate is 12.04%. We know that this value is often optimistic, especially for decision tree model.



We add the DEFINE STATUS component. We set DISEASE as TARGET, the predicted values (PRED\_SPV\_INSTANCES\_1) as INPUT.



Then, we insert the TEST component (SPV LEARNING ASSESSMENT tab). We click on the VIEW menu. The confusion matrix is computed automatically on the test set (unselected instances).

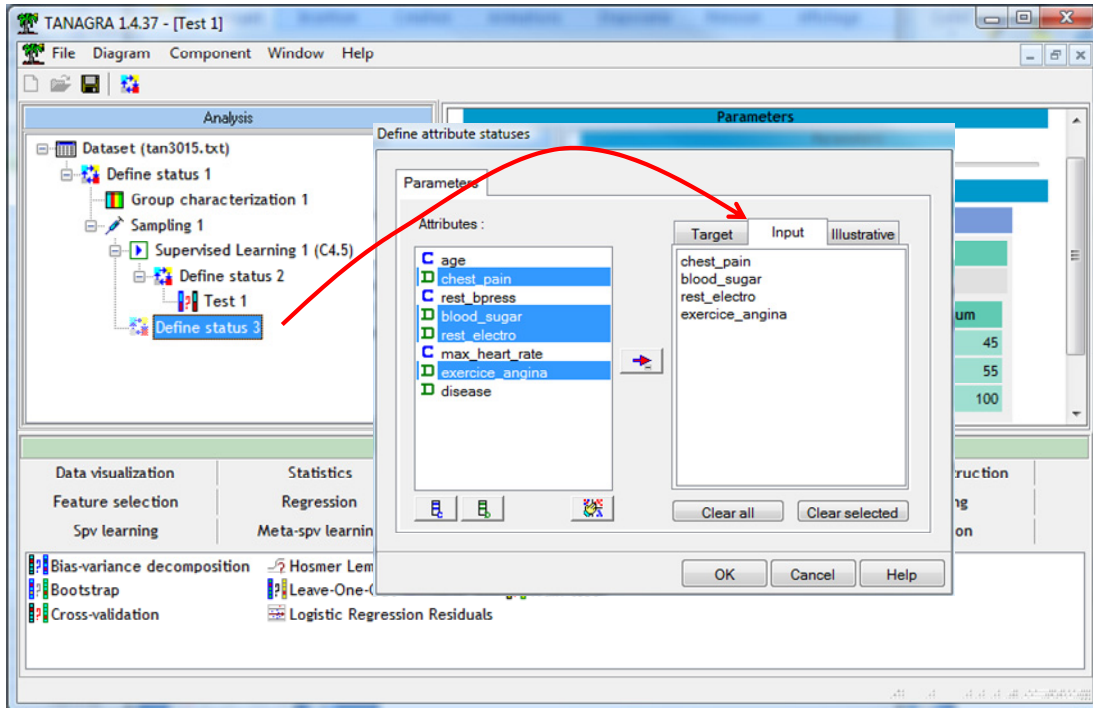


The "true" (unbiased estimation of the) generalization error rate is in effect 25%.

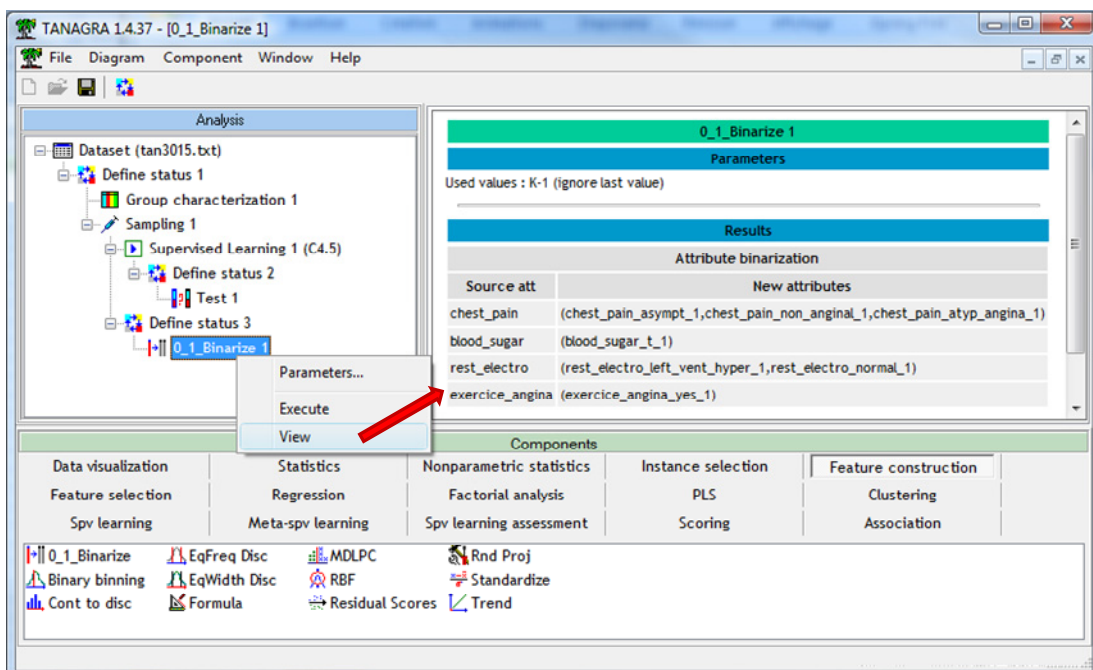
### 3.6 Logistic regression

**Learning step.** We want to analyze the behavior of the logistic regression on the same dataset. We cannot directly implement the approach because some predictive attributes are categorical. Thus, we must recode them as binary (0/1).

We insert the DEFINE STATUS into the diagram. We set as INPUT the categorical predictive attributes: CHEST\_PAIN, BLOOD\_SUGAR, REST\_ELECTRO, EXERCICE\_ANGINA.



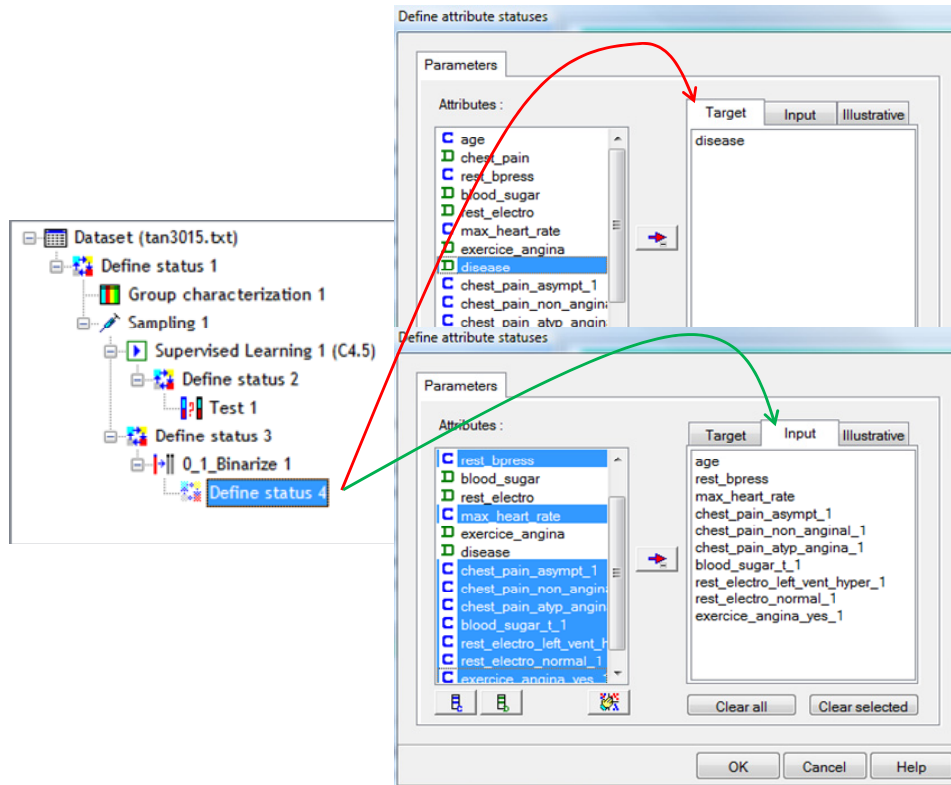
Then, we insert o\_1\_BINARIZE (FEATURE CONSTRUCTION tab). We activate the VIEW menu.



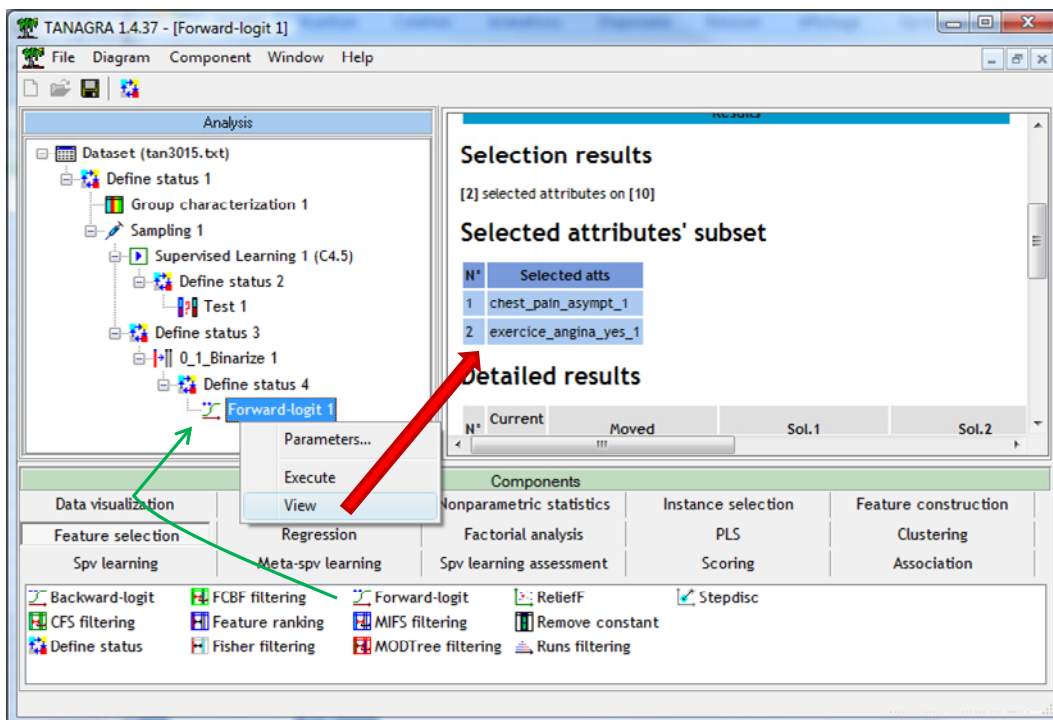
For a categorical attribute with K values, Tanagra generates (K-1) binary variables. The last value is the reference.



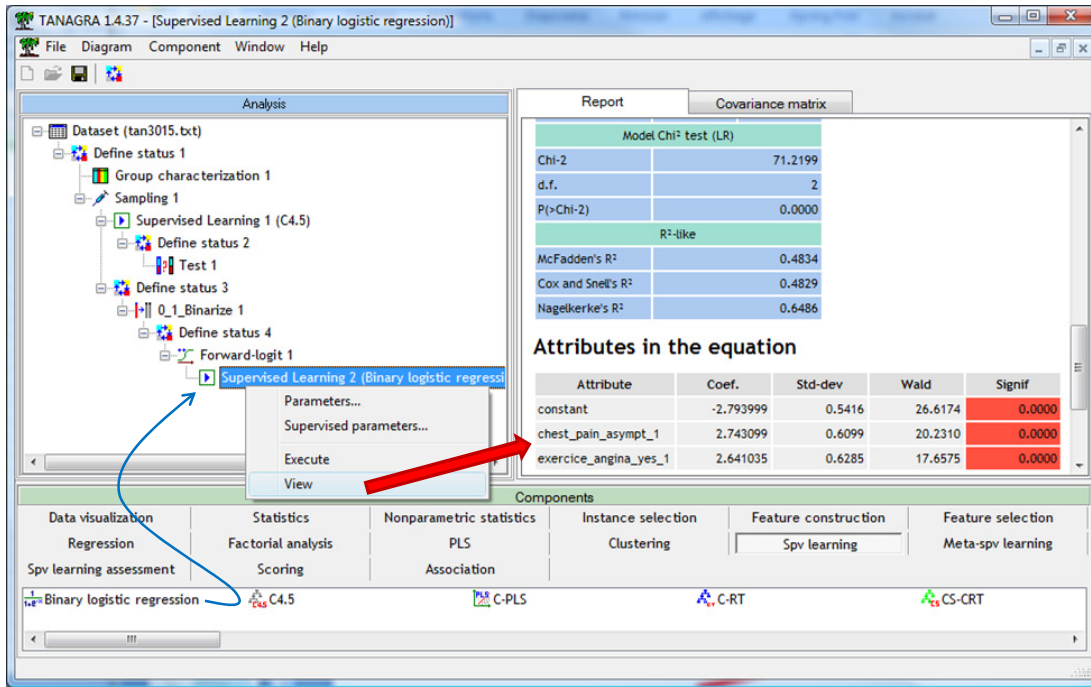
We can launch the analysis. We add the DEFINE STATUS component to set the target attribute (DISEASE) and the input ones (all the numerical/continuous attributes).



We add the FORWARD LOGIT component (FEATURE SELECTION tab) in order to select the relevant attributes.

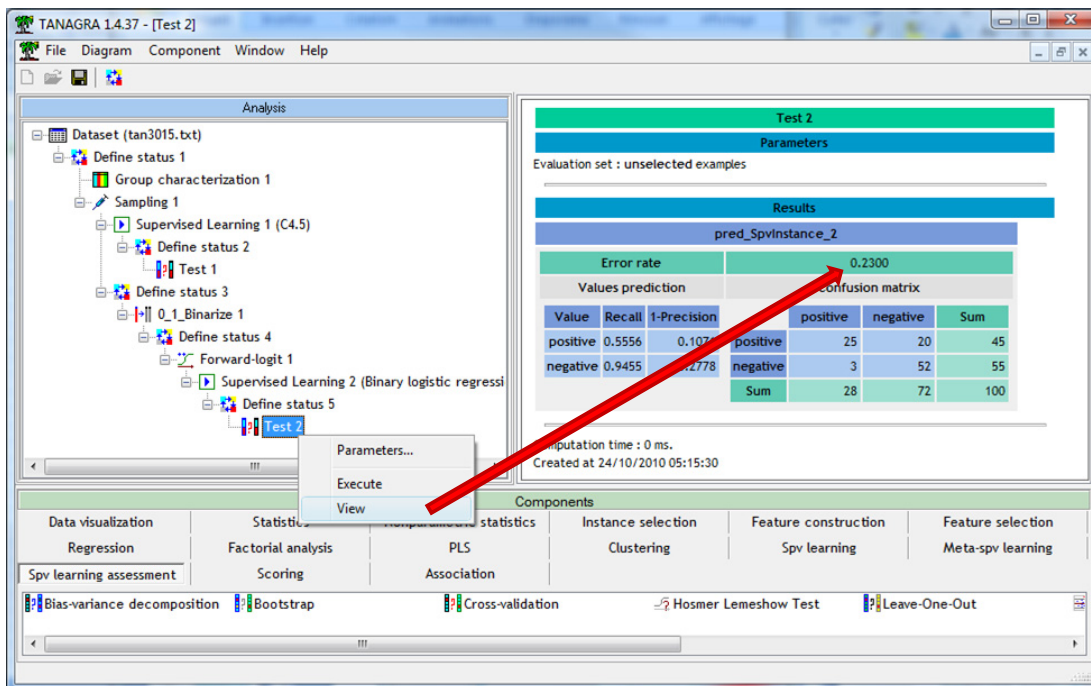


Two binary attributes are selected (CHEST\_PAIN = ASYMPT) and (EXERCICE\_ANGINA = YES) (*this is very interesting if one recalls the results of descriptive statistics, section 3.2*). We can launch the logistic regression now (BINARY LOGISTIC REGRESSION, SPV LEARNING tab).



The resubstitution error rate is 18.52%. The model seems worse than the decision tree.

**Testing step.** Again we apply the classifier on the test sample [DEFINE STATUS (target = disease, input = pred\_spvinstance\_2) and TEST].

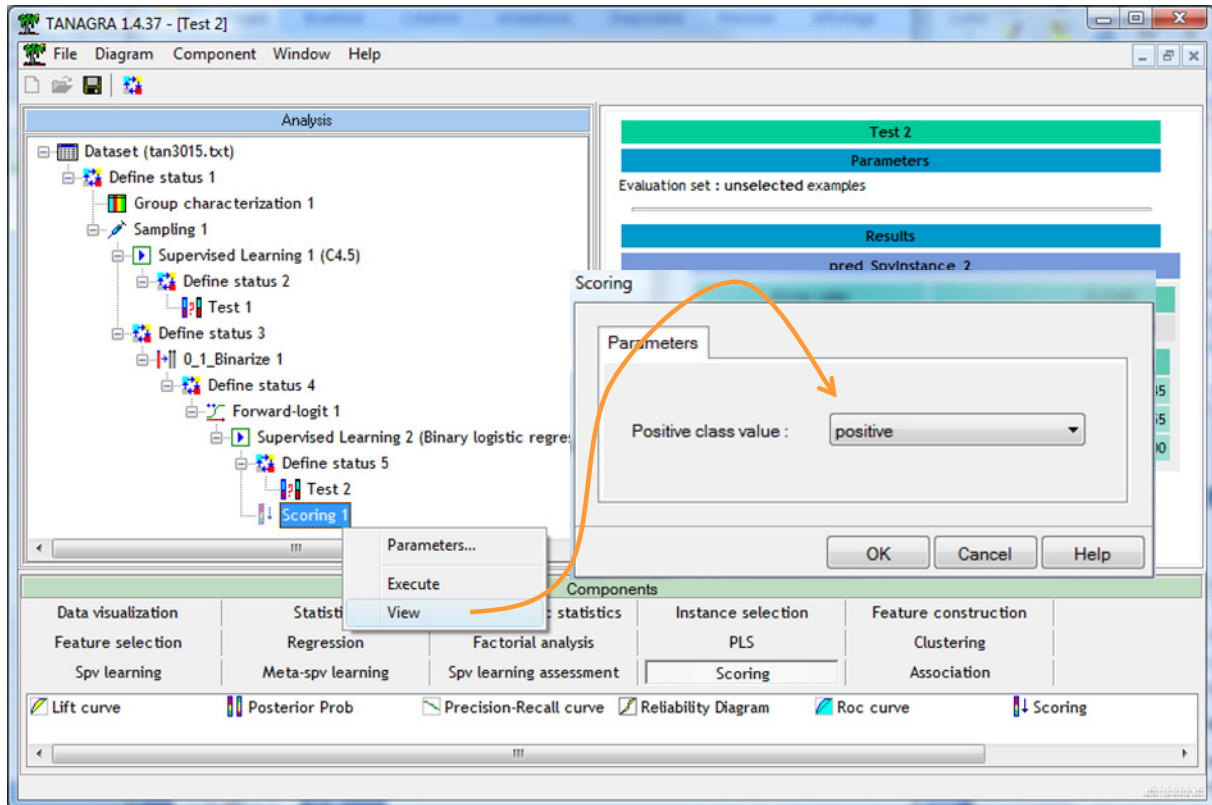


The test error rate is 23%. Slightly better than the decision tree, but above all, with a different behavior: the recall is lesser ( $25/45 = 56\%$  vs.  $69\%$ ), the precision is better ( $25/28 = 89\%$  vs.  $74\%$ ).

### 3.7 ROC curve for assessing the logistic regression

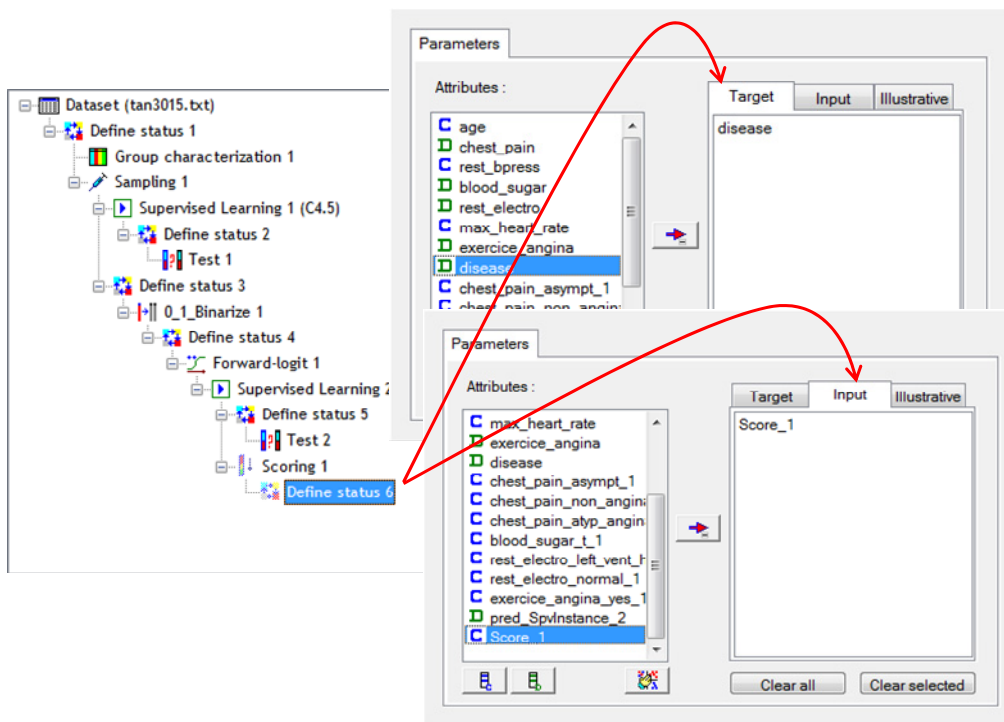
Contrarily to the decision tree, the logistic regression provides a good estimation of the posterior probability of the class value for the individuals. We use this singularity to evaluate the classifier with another tool: the ROC curve. It gives a broader point of view about the behavior of the classifier.

First, we compute the probability to be positive (the score) for each instance of the dataset. We use the SCORING component (SCORING tab). DISEASE = POSITIVE are the positive instances.

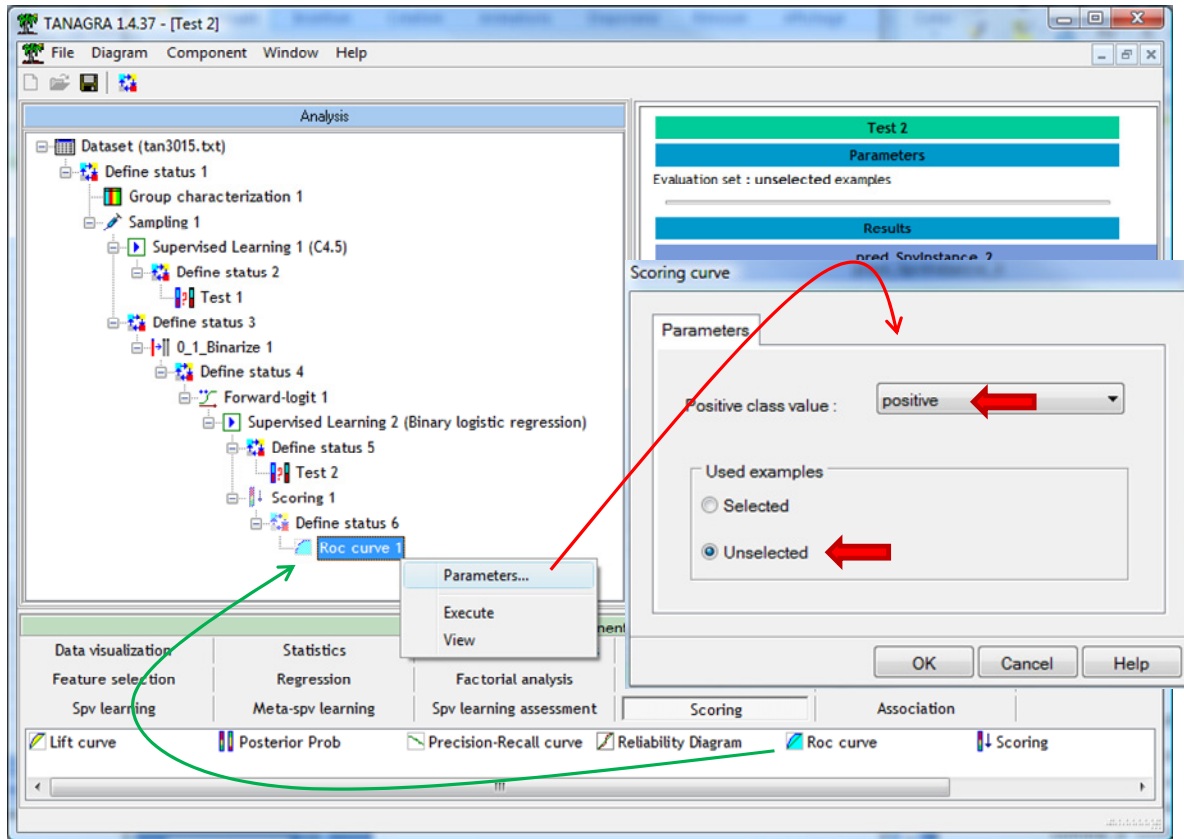


We validate and we click on VIEW. A new column is added to the current dataset.

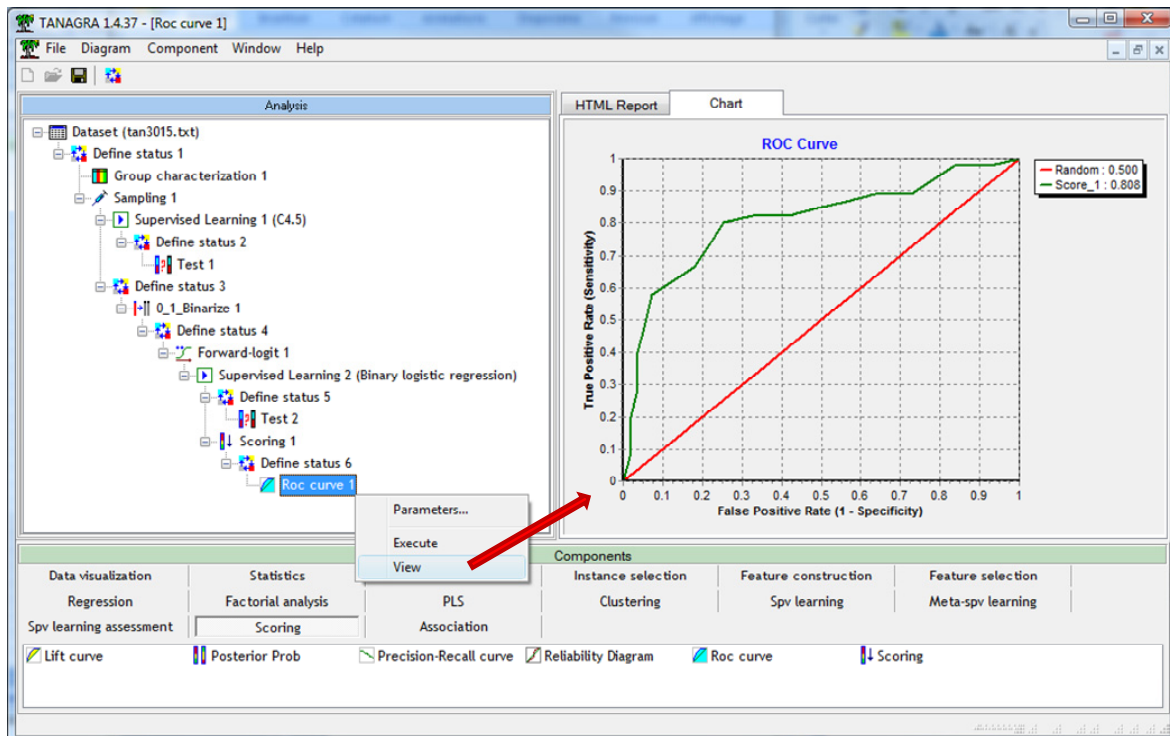
We add the DEFINE STATUS component into the diagram. We set DISEASE as TARGET, and the score (SCORE\_1) as INPUT (we can set many score columns to compare classifiers).



We add the ROC CURVE component (SCORING tab). We set the following parameters (DISEASE = POSITIVE are the positive instances, we compute the curve on the test set – UNSELECTED).

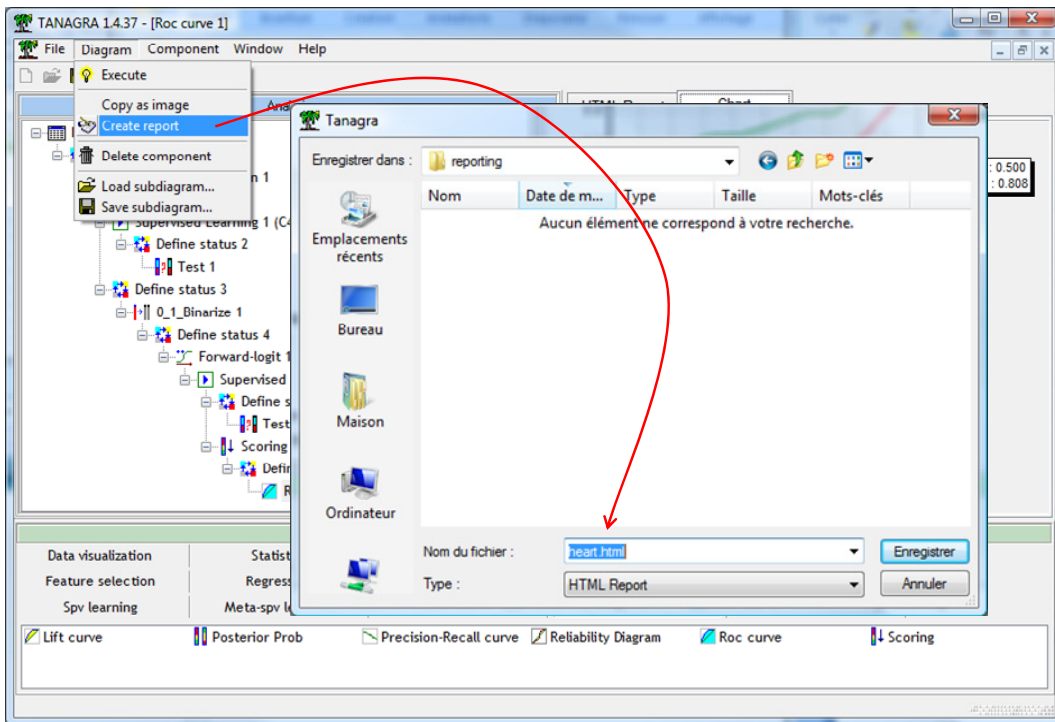


We click on the VIEW menu. We obtain the ROC curve. The AUC (area under curve) is 80.8%. The classifier is rather good (if the model is not better than the random scoring, the AUC is 50%).

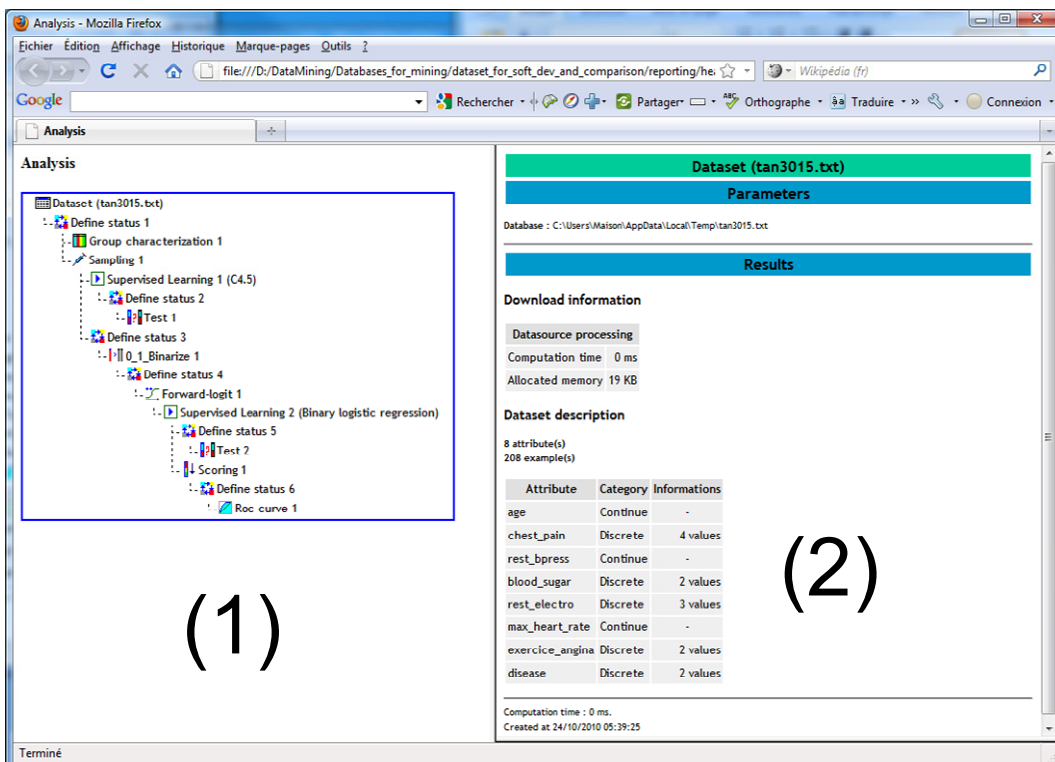


### 3.8 Creating and visualizing the report

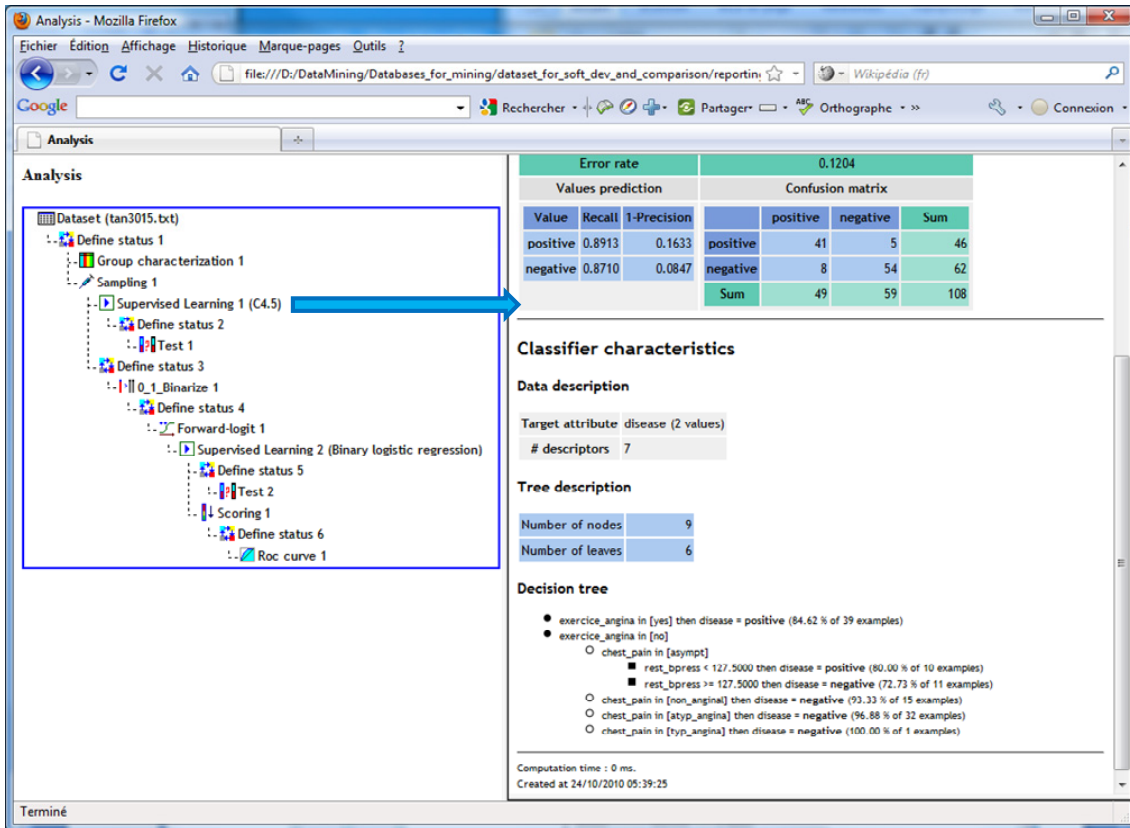
We want to share the whole results to other persons. Tanagra can export them in a HTML file format that we can visualize in a browser.



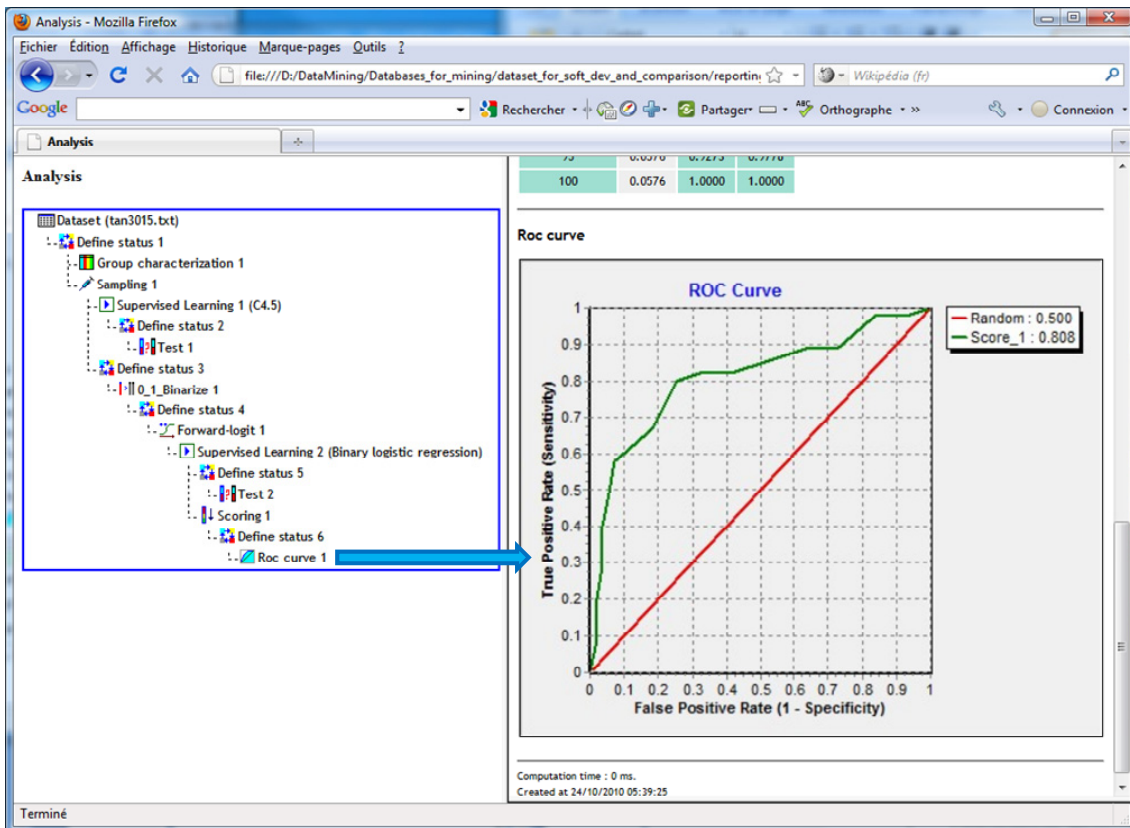
To create a report, we click on the DIAGRAM / CREATE REPORT menu. We set the master file name of the report e.g. « heart.html ». The report is automatically loaded in a browser. At the left side, we have the picture of the diagram (1). We can select any component. The associated results are displayed in the right side of the window (2).



If we want to see the tree for instance, we select « Supervised Learning 1 (C4.5) » at the left side.

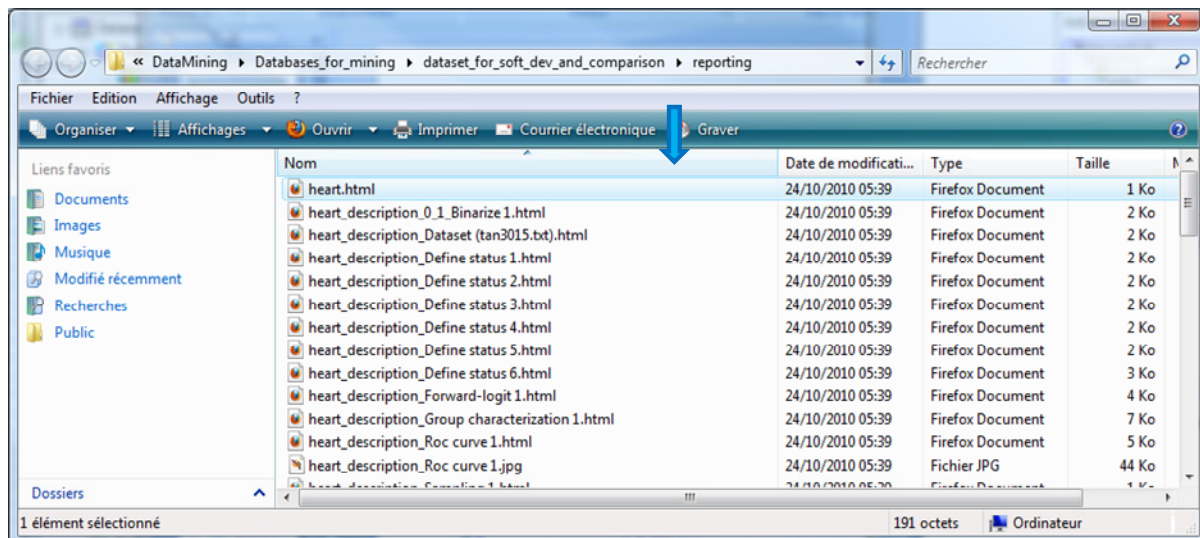


In the same manner, for the ROC curve, we obtain:



The ROC CURVE figure is inserted into the HTML page.

Of course, the HTML file can be edited in any word processing. We can add comments to the results. To spread the results, we must simply copy the whole directory containing the files associated to the report.



## 4 Conclusion

In this tutorial, we show that it is easy to collect the results provided by Tanagra by creating a report. This feature is very important in the professional world. A priori, it is less essential in the academic domain. But I think that the possibility to copy the results tables in Excel to perform other calculations thereafter is an undeniable educational asset.