1 Subject

New "Partial Least Squares Discriminant Analysis" methods with TANAGRA 1.4.23.

PLS Regression can be viewed as a multivariate regression framework where we want to predict the values of several target variables (Y1, Y2, …) from the values of several input variables (X1, X2, …) (Tenenhaus[1], 1998).

Roughly speaking, the algorithm is the following: "The components of X are used to predict the scores on the Y components, and the predicted Y component scores are used to predict the actual values of the Y variables. In constructing the principal components of X, the PLS algorithm iteratively maximizes the strength of the relation of successive pairs of X and Y component scores by maximizing the covariance of each X-score with the Y variables. This strategy means that while the original X variables may be multicollinear, the X components used to predict Y will be orthogonal" (Garson, http://www2.chass.ncsu.edu/garson/PA765/pls.htm).

The choice of the number of factors is very important in this process. Results in various domains show that a few numbers of X-components are enough for an efficient prediction.

The PLS Regression is initially defined for the prediction of continuous target variable. But it seems it can be useful in the supervised learning problem where we want to predict the values of discrete attributes. In this tutorial we propose a few variants of PLS Regression adapted to the prediction of discrete variable.

The generic name "PLS-DA (Partial Least Square Discriminant Analysis" is often used in the literature. The various algorithms always perform in two main steps: (1) transform the categorical variable Y in an indicator matrix, the PLS regression operates on this dataset; (2) use wisely the PLS results in order to predict the class membership of individuals[2].

Note: This tutorial intends only to describe the utilization and the comprehension of the results of PLS-DA. In a future work, we try to show the benefits of this approach when we have a very number of descriptors compared with the available examples. In this context, PLS-DA approaches are as accurate as well regularized methods such as Support Vector Machine (SVM).

2 Dataset

We use the BREAST-CANCER-PLS-DA.XLS (http://eric.univ-lyon2.fr/~ricco/tanagra/fichiers/breast-cancer-pls-da.xls) dataset. It is a part of the "breast-cancer-wisconsin" dataset which is available on the UCI Server[3]. The target attribute is CLASS which can be "positive" (the cells come from malignant tumor) or "negative".

[1] This book is probably the main French reference: M. Tenenhaus, « La régression PLS – Théorie et Pratique », Technip, 1998.

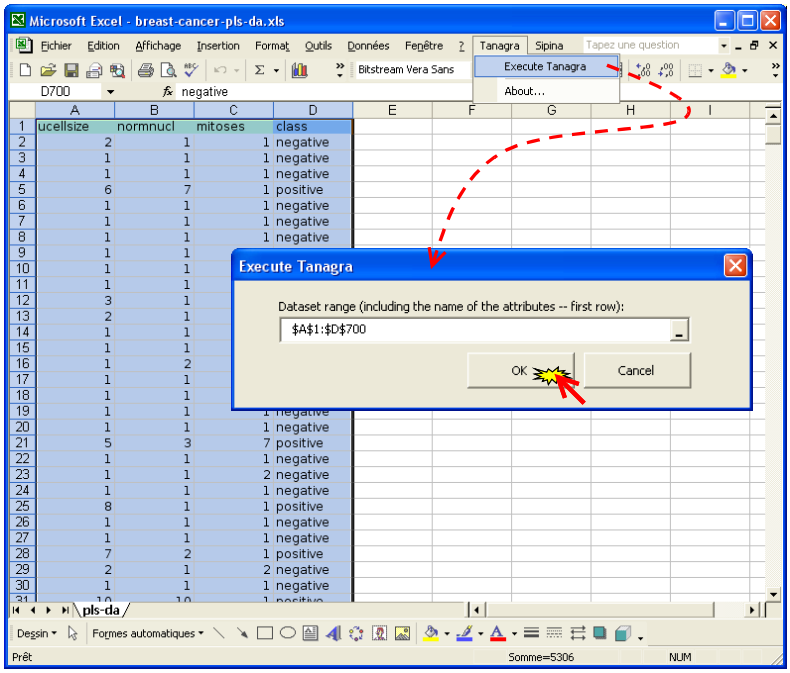
[2] Our main reference for the implementation is **S. Chevallier, D. Bertrand, A. Kohler, P. Courcoux, « Application of PLS-DA in multivariate image analysis », in J. Chemometrics, 20: 221-229, 2006**.

[3] http://archive.ics.uci.edu/ml/datasets/Breast+Cancer+Wisconsin+(Original)

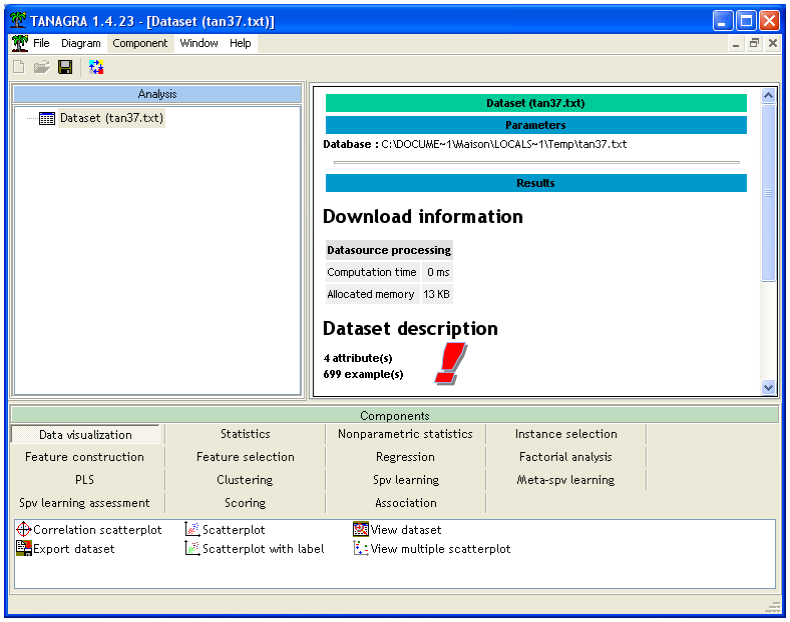
3 Data preparation

3.1 Creating a new diagram

We load the XLS data file in the EXCEL spreadsheet. Then, we select the cells and we click on the menu TANAGRA/EXECUTE TANAGRA[4]. We check the selection and we click on OK.



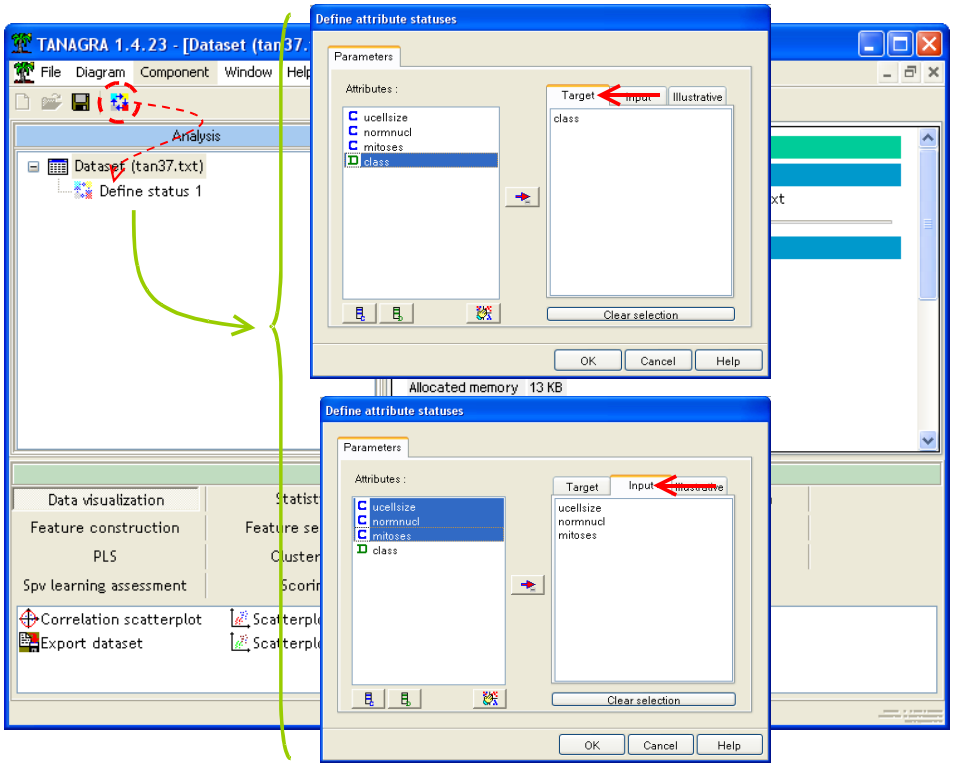
TANAGRA is launched and the dataset loaded: there are 699 instances and 4 variables.



[4] The EXCEL add-in TANAGRA.XLA is available since the version 1.4.11. See the tutorial on the web site for the utilization of this add-in: http://eric.univ-lyon2.fr/~ricco/tanagra/fichiers/en_Tanagra_Excel_AddIn.pdf

3.2 TARGET and INPUT variables

We insert the DEFINE STATUS component using the shortcut in the tool bar. We set CLASS as TARGET and the other ones as INPUT.



4 Partial Least Squares Discriminant Analysis

4.1 The C-PLS component

4.1.1 Description of the method

The component C-PLS is dedicated to binary problem i.e. **the TARGET attribute must have 2 values only.**

The discrete TARGET attribute is replaced by a continuous attribute using a specific code (Tomassone et al., 1988[5]).

Y is the target attribute, $Y = \{+,-\}$. If n_+ (n_-) is the number of examples with POSITIVE value (NEGATIVE), and $n = n_+ + n_-$, The indicator variable Z is defined as follows:

$$Z = \begin{cases} \frac{n_-}{n}, \text{si } Y = + \\ -\frac{n+}{n}, \text{si } Y = - \end{cases}$$

The approach produces one discriminant function D(X). It enables to classify an unseen instance using the following rule (p is the number of descriptors):

[5] R. Tomassone, M. Danzart, J.J. Daudin, J.P. Masson, « Discrimination et classement », Masson, 1988 ; page 38.

$$D(X) = a_0 + a_1 X_1 + \ldots + a_p X_p \begin{cases} \geq 0 \Rightarrow Y = + \\ < 0 \Rightarrow Y = - \end{cases}$$

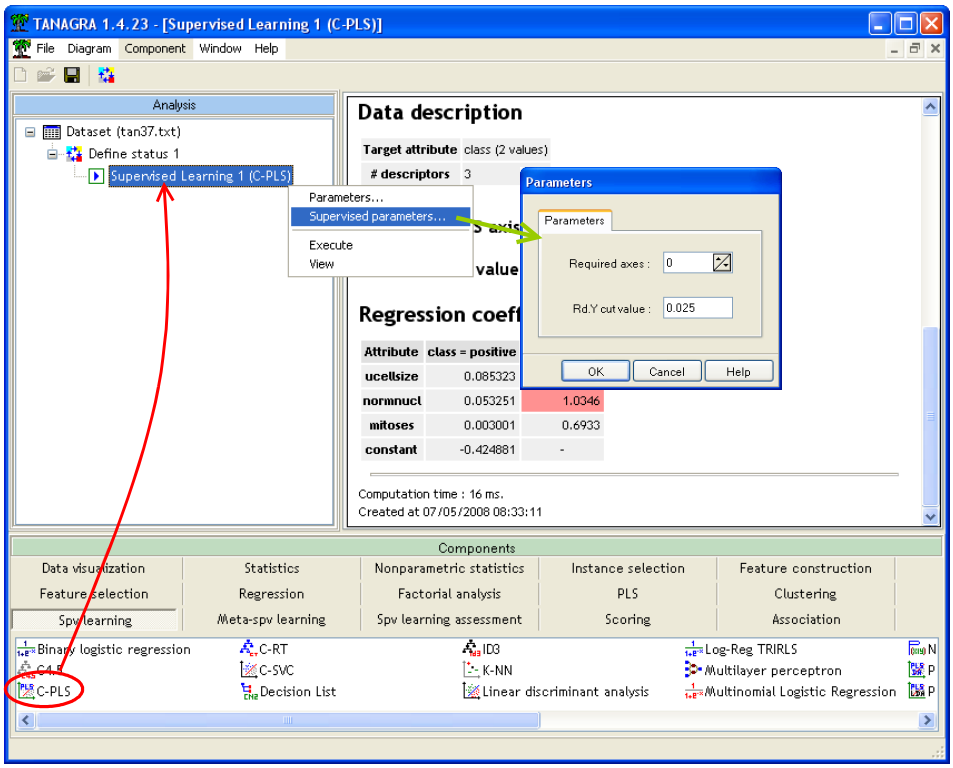
We can explicitly specify the number of factors (X component) for C-PLS.

But, C-PLS can also to detect the right number of factors using a very simple heuristic. It adds step by step a new factor. If the additional variability of Y explained by a new factor is lower than 0.025, we stop the process. This is a very simplistic approach, but it is enough in the first time in order to evaluate the efficient of the method on a dataset.

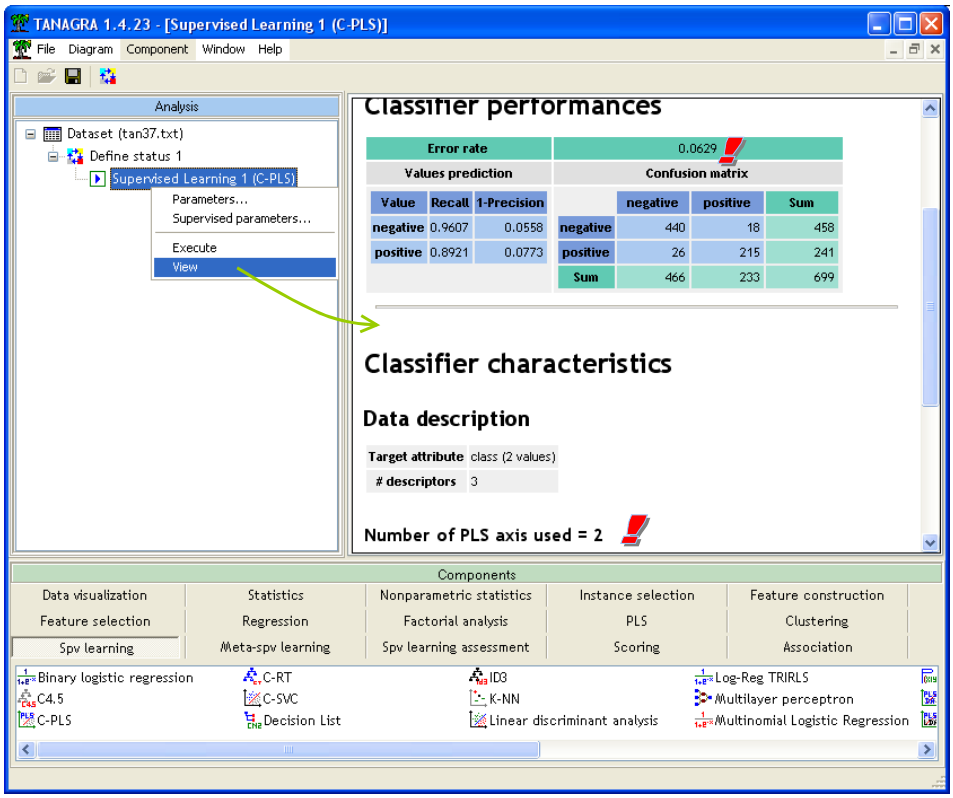
4.1.2 Viewing and interpreting the results

We insert the C-PLS component (SPV LEARNING tab) into the diagram.

We click on the SUPERVISED PARAMETERS contextual menu. The dialog box settings appears. The required number of factors (axes) is 0. It means that the method tries to detect automatically the right number of factors. The stopping rule is defined with the "RdY Cut Value".



We leave these default parameters. We close the dialog box by clicking on the OK button. Then, we activate the VIEW contextual menu. The computation is automatically launched and the results appear in a new window.



We obtain the usual confusion matrix and the resubstitution[6] error rate (6.29%). The process has selected the 2 first factors (X components) for the prediction of the TARGET variable.

Classification function. In the low part of the window, the classification function is displayed. The positive value of the class attribute is "CLASS = POSITIVE".

Regression coefficients

Attribute	class = positive	VIP
ucellsize	0.085323	1.2038
normnucl	0.053251	1.0346
mitoses	0.003001	0.6933
constant	-0.424881	-

Figure 1 – Classification function C-PLS

We try to classify a new instance with this function. Let UCELLSIZE = 2 ; NORMNUCL = 1 ; MITOSES = 1. We obtain:

$$D(X) = -0.425 + 0.085 \times 2 + 0.053 \times 1 + 0.003 \times 1 = -0.198 < 0 \Rightarrow Y = -$$

We conclude that this instance belongs to CLASS = NEGATIVE group.

Variable importance in Projection. VIP coefficients reflects the relative importance of each input variables for the selected factors (Tenenhaus, 1998 ; page 139 ; Garson,

[6] The resubstitution error rate is an optimistic estimator of the error rate because we use the same dataset for the learning phase and the testing phase. But in this tutorial, it does not matter, we want to show mainly how to use the "PLS Discriminant Analysis" components into TANAGRA.

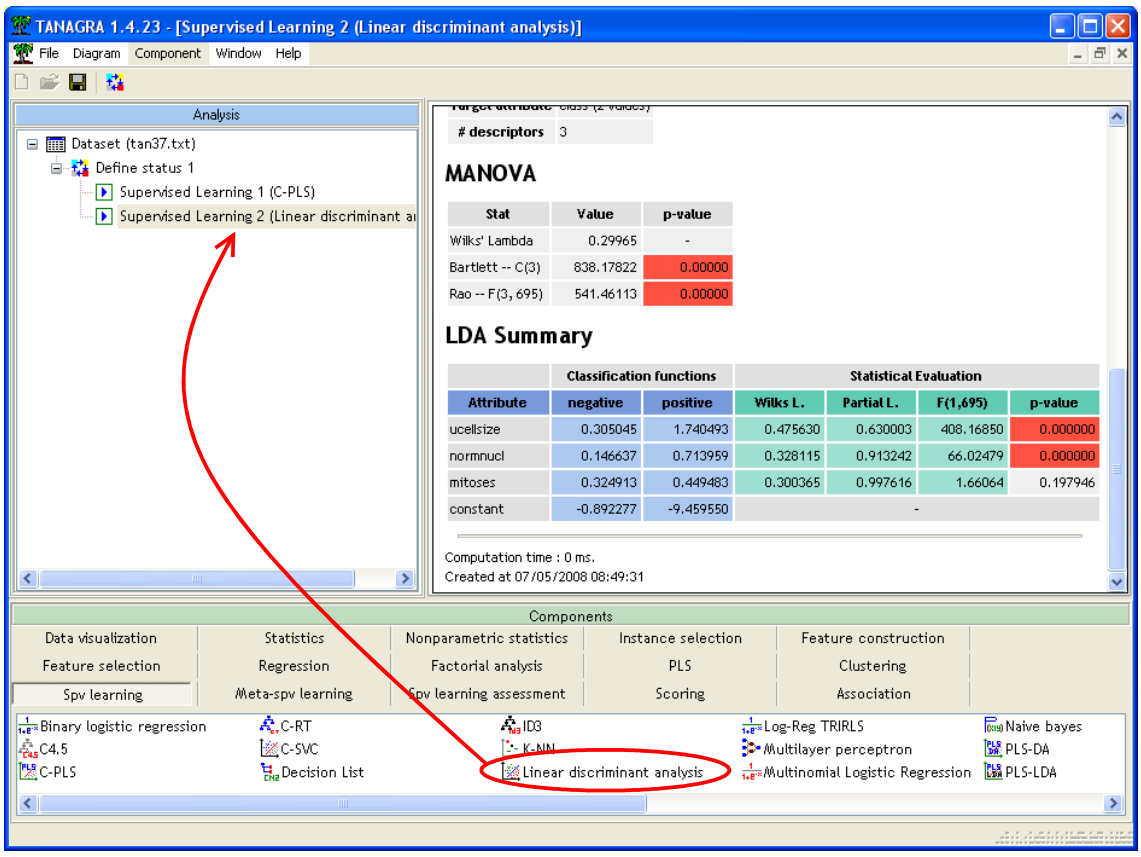
http://www2.chass.ncsu.edu/garson/PA765/pls.htm). We can consider that a variable is relevant if (VIP > 1). Another selection rule is to drop a variable if its VIP < 0.8 and the regression coefficient is very small in absolute size.

Note: These selections rules must be use with caution because the VIP reflects only the relative importance (each others) of the input variables. It does not mean that a variable with a low VIP is not relevant for the classification.

In our problem (Figure 1), UCELLSIZE and NORMNUCL seem the most relevant descriptors.

4.1.3 Comparison with linear discriminant analysis

We compare the results of C-PLS with the state of the art linear discriminant analysis (LDA). We insert the LINEAR DISCRIMANT ANALYSIS component (SPV LEARNING tab) into the diagram. We click on the VIEW contextual menu in order to obtain the results.



The overall accuracy is similar. But LDA produces one classification function for each value of the TARGET attribute. The coefficients are not comparable. LDA shows also that MITOSES attribute seems not relevant in the classification task.

4.2 The PLS-DA component

4.2.1 Description of the method

PLS-DA can handle multiclass problem i.e. the **TARGET attribute can have K (K ≥ 2) values.** It relies on the same principle as C-PLS about the detection of the number of factors.

In this approach, we create K indicator variable (as much as the number of TARGET attribute values) using the following coding scheme:

$$Z_k = \begin{cases} 1, \text{si } Y = y_k \\ 0, \text{otherwise} \end{cases}$$

The PLS algorithm handles the Z target variables and the X descriptors. We obtain K classification function.

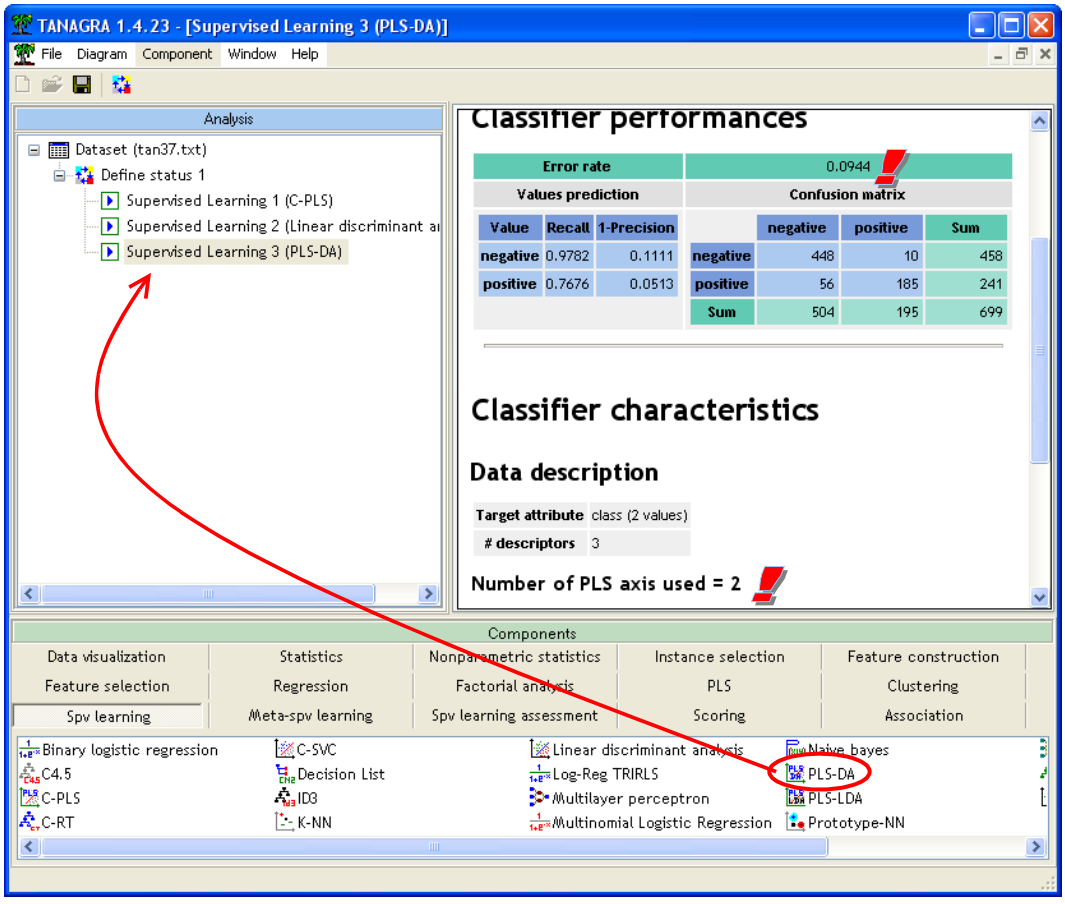
$$\hat{Z}_k = b_{0,k} + b_{1,k} X_1 + \cdots + b_{p,k} X_p$$

In order to classify an unseen case, we compute the predicted value for each function; the prediction corresponds to the highest value i.e.

$$Y = y_{k^*} \Leftrightarrow k^* = \arg\max_k \left(\hat{Z}_k \right)$$

4.2.2 Viewing and interpreting the results

We insert the PLS-DA component (SPV LEARNING tab) into the diagram. We click on the VIEW contextual menu. The results are displayed in a new window. The method detects automatically two factors. The resubstitution error rate is 9.44%.



We obtain two classification functions (one for each target value, Figure 2). The relevance of each attribute can be evaluated with the VIP criterion.

Classification functions

Attribute	negative	positive	VIP
ucellsize	-0.085323	0.085323	1.2038
normnucl	-0.053251	0.053251	1.0346
mitoses	-0.003001	0.003001	0.6933
constant	1.080103	-0.080103	-

Figure 2 – PLS-DA classification functions

For an unseen instance with (UCELLSIZE = 2; NORMNUCL = 1; MITOSES = 1), we obtain the following prediction:

$$\begin{cases} D(-,X) = 1.080 - 0.085 \times 2 - 0.053 \times 1 - 0.003 \times 1 = 0.853 \\ D(+,X) = -0.080 + 0.085 \times 2 + 0.053 \times 1 + 0.003 \times 1 = 0.147 \end{cases}$$
$$\text{Then, } D(-,X) > D(+,X) \Rightarrow Y = -$$

4.3 The PLS-LDA component

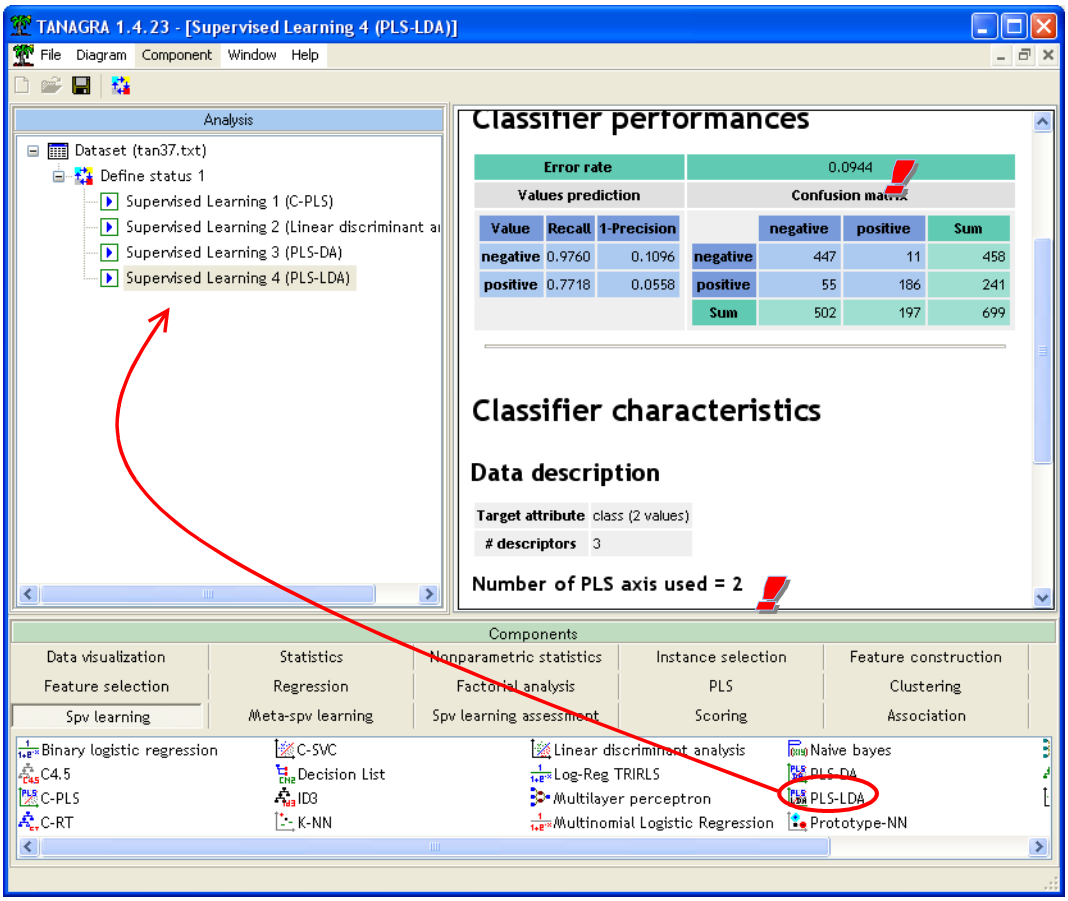
4.3.1 Description of the method

PLS-LDA combines PLS Regression and Linear Discriminant Analysis. It can handle multiclass problem i.e. the **TARGET attribute can have K (K ≥ 2) values.** It relies on the same principle (as C-PLS and PLS-DA) about the number of factors detection.

There are two main steps in the learning process. Firstly, using the same coding scheme as PLS-DA, we launch the PLS algorithm. Secondly, we launch the linear discriminant analysis on the X component scores (factors). Because these factors are orthogonal, the LDA is more reliable. This kind of data transformation is very useful when the original input variables are highly correlated.

4.3.2 Viewing and interpreting the results

We insert the PLS-LDA component into the diagram. We activate the VIEW contextual menu. TANAGRA computes immediately the unstandardized coefficients of the classification function, one for each value of the TARGET attribute. The classification rule is the same as the PLS-DA component.



The method selects the two first factors. The resubstitution error rate is identical to PLS-DA method, but not the classification matrix. The classification functions are in the low part of the window (Figure 3).

Classification functions

Attribute	negative	positive
ucellsize	-0.426839	0.811171
normnucl	-0.266394	0.506258
mitoses	-0.015013	0.028531
constant	1.102686	-7.271344

Figure 3- Fonctions de classement PLS-LDA

Let us to classify the same unseen case (UCELLSIZE = 2 ; NORMNUCL = 1 ; MITOSES = 1):

$$\begin{cases} D(-,X) = 1.103 - 0.427 \times 2 - 0.266 \times 1 - 0.015 \times 1 = -0.032 \\ D(+,X) = -7.271 + 0.811 \times 2 + 0.506 \times 1 + 0.029 \times 1 = -5.114 \end{cases}$$

$$\text{Then, } D(-,X) > D(+,X) \Rightarrow Y = -$$

Note: Because we combine two approaches, it is very difficult to evaluate the influence of the original input attributes. For this reason no information is provided about the relevance of input attributes.

5 Conclusion

With the version 1.4.23 of TANAGRA, we wanted to highlight the supervised learning methods based on the PLS regression, commonly called PLS Discriminant Analysis. PLS Regression is very popular in many research areas, but it is less diffused in the machine learning community. Yet its characteristics are very interesting, even decisive in some contexts, especially when the descriptors are very numerous and highly redundant. This kind of situation occurs frequently in real DATA MINING problems.

In this tutorial, we show how to implement these methods with TANAGRA, how to read and interpret the results.