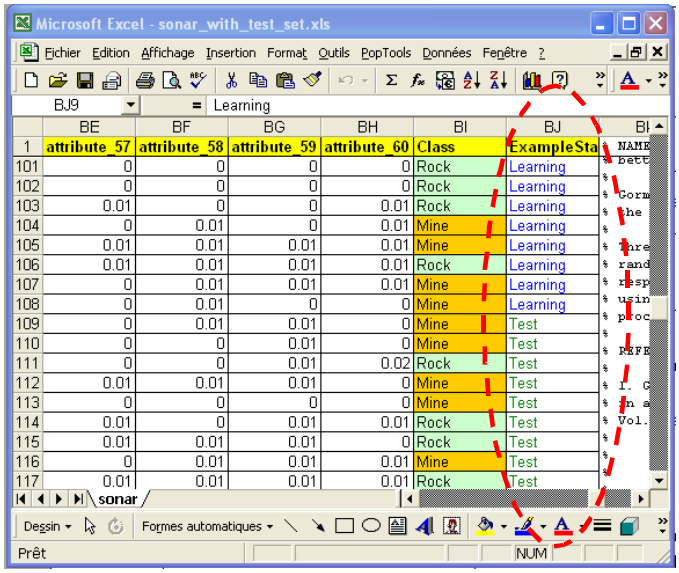
Objectif

Pour évaluer un algorithme d'apprentissage supervisé, on conseille souvent de subdiviser les données en deux sous-ensembles disjoints : l'ensemble d'apprentissage (*learning set*) qui sert à élaborer le modèle de prédiction ; l'ensemble test (*test set*) qui sert à en mesurer les performances. TANAGRA dispose d'outils permettant de construire automatiquement ces sous-ensembles à partir d'un échantillonnage mais, dans certains cas, l'utilisateur peut vouloir procéder lui-même à cette subdivision afin d'utiliser les mêmes ensembles d'apprentissage et de test pour comparer les algorithmes d'apprentissage.

Dans ce didacticiel, nous utiliserons un fichier de données dans lequel nous avons introduit une colonne supplémentaire permettant de désigner les individus à utiliser pour l'apprentissage et ceux à utiliser lors de l'évaluation. Nous montrerons alors quels composants utiliser pour désigner les observations qui vont servir à construire les modèles de prédiction, nous utiliserons un autre composant pour comparer leurs performances sur l'ensemble test.

Fichier

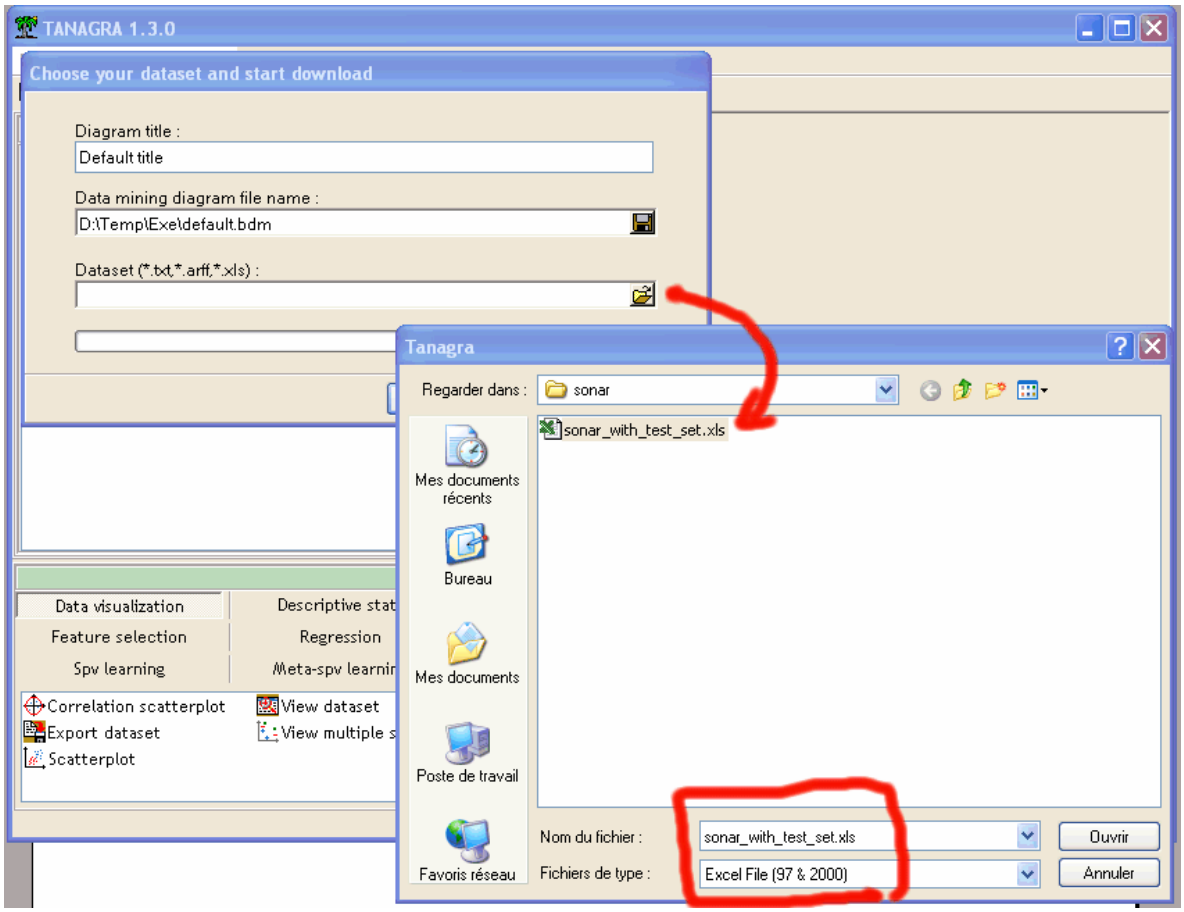
Les données recensent différentes mesures recueillies à l'aide d'un sonar, l'objectif est de discerner les roches des mines à partir de ces relevés. Le fichier SONAR provient du site UCI (http://www.ics.uci.edu/~mlearn/MLRepository.html -- **sonar_with_test_set.xls**). Il contient 208 observations avec 60 descripteurs (ATTRIBUTE_1 à ATTRIBUTE_60), tous continus, et une variable à prédire binaire (CLASS). Plutôt que d'utiliser deux fichiers distincts pour chaque ensemble de données, nous préférons la solution simple qui consiste à les réunir dans un seul et même fichier, puis ajouter une colonne supplémentaire indiquant le rôle que doit jouer chaque observation (EXAMPLESTATUS).



Comparer les algorithmes

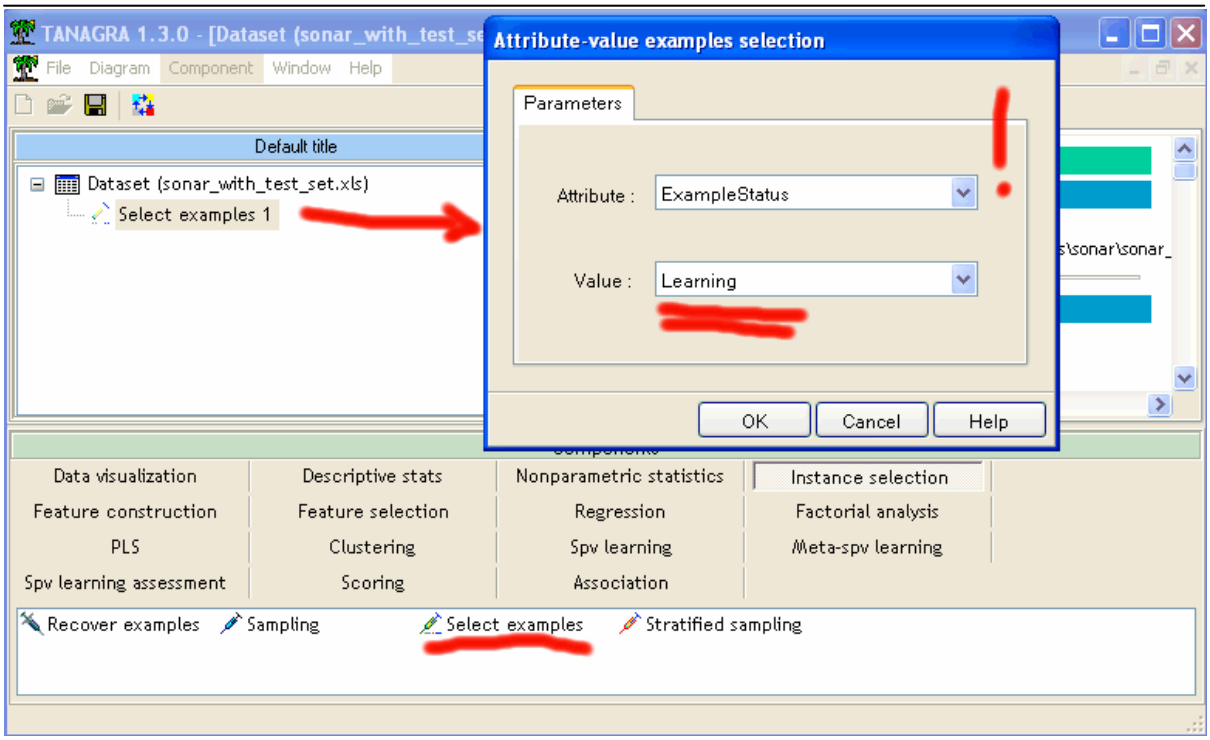
Charger le fichier de données

Créez un nouveau diagramme de traitements et sélectionnez le fichier SONAR_WITH_TEST_SET.XLS (menu FILE / NEW).

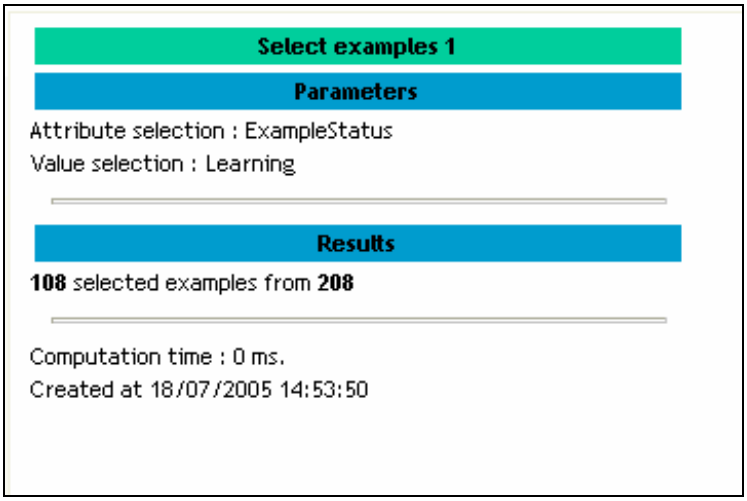


Subdiviser les données en « apprentissage » et « test »

L'étape suivante consiste à définir les observations qui seront utilisés dans la construction des modèles de prédiction, et celles qui serviront à leur évaluation. Placez le composant SELECT EXAMPLES de la palette INSTANCE SELECTION. Lors du paramétrage, les individus sélectionnés doivent correspondre aux individus « apprentissage », nous sélectionnons donc la modalité *Learning* de l'attribut *ExampleStatus* dans la boîte de dialogue.

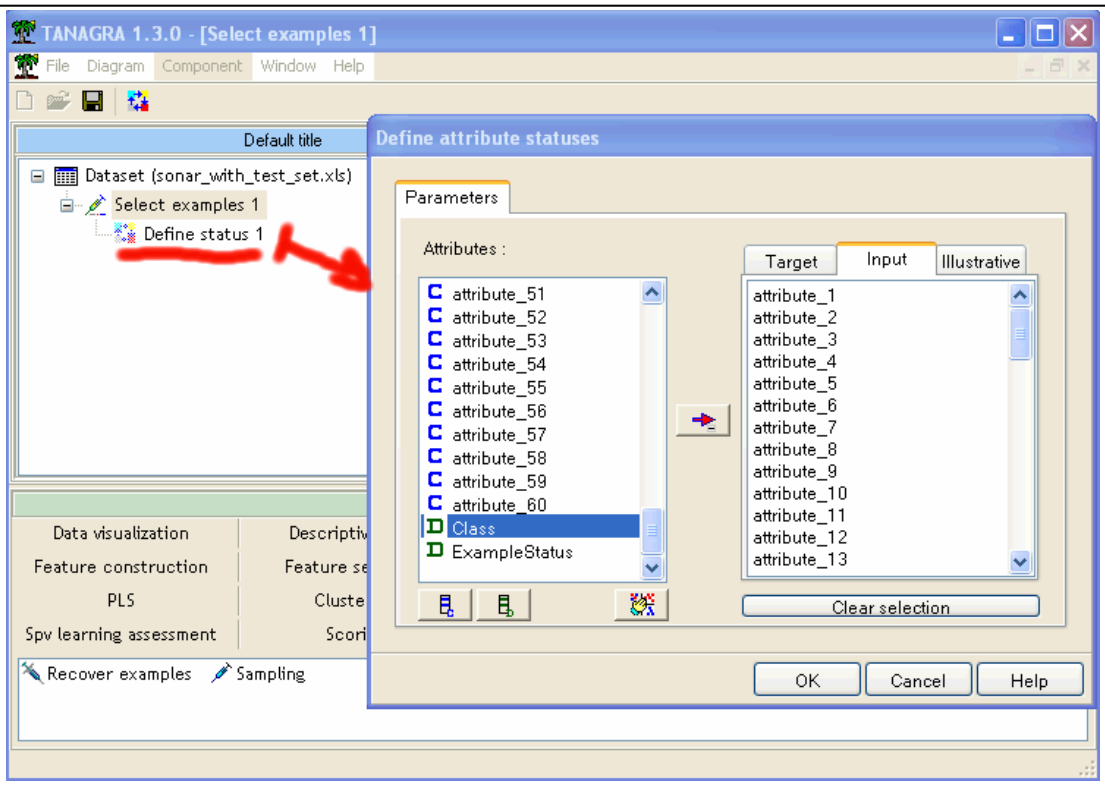


Nous observons que 108 observations parmi 208 seront consacrés à l'apprentissage, les 100 autres constitueront l'ensemble test.



Définir le problème

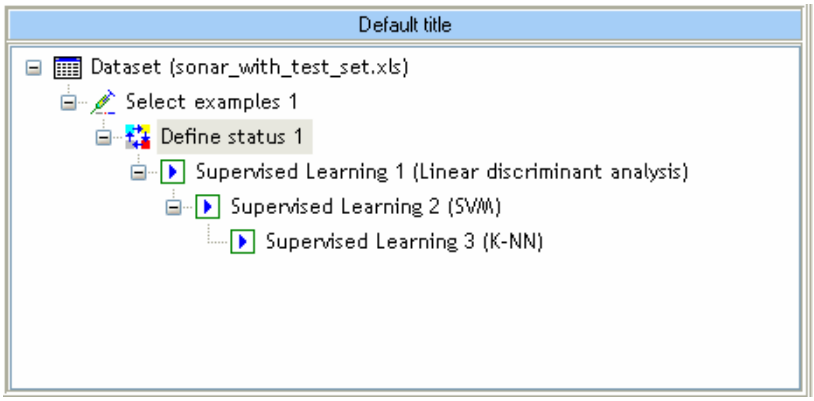
Placez un composant DEFINE STATUS pour définir la variable à prédire (CLASS) et les descripteurs (ATTRIBUTE_1 à ATTRIBUTE_60). Bien entendu, il n'est pas question d'utiliser la variable EXAMPLESTATUS à ce stade.



Apprentissage

Dans ce didacticiel, nous comparerons 3 méthodes d'apprentissage reposant sur des biais de représentation et de préférence très différents.

Placez successivement une analyse discriminante (LINEAR DISCRIMINANT ANALYSIS), une machine à vecteurs de supports (SVM) et un algorithme des plus proches voisins (K-NN). Rappelons qu'un algorithme d'apprentissage supervisé doit être encapsulé dans un composant méta-apprentissage (META SPV LEARNING), nous utiliserons un apprentissage unique simple (SUPERVISED LEARNING)[1].



[1] Concrètement, il s'agit de placer les composants en deux temps : d'abord nous plaçons le composant SUPERVISED LEARNING (onglet META SPV LEARNING) dans le diagramme, puis nous y intégrons le composant d'apprentissage (ex. LINEAR DISCRIMANT ANALYSIS) en provenance de l'onglet SPV LEARNING.

Nous observons que le taux d'erreur en resubstitution (mesuré sur l'ensemble d'apprentissage) est de :

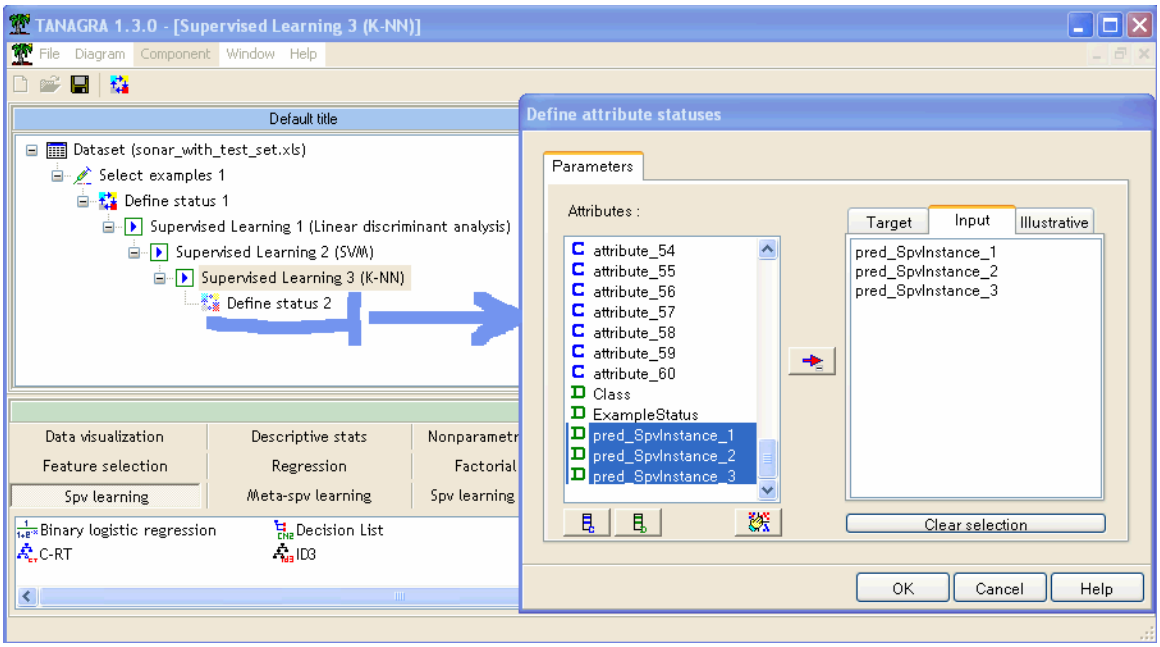
- 2.78% pour la LDA ;
- 12.04% pour le SVM linéaire ;
- 11.11% pour le K-NN (5-NN).

Il semblerait que l'analyse discriminante soit la plus performante sur ce problème ?

Evaluation sur l'ensemble test

Pour lever le doute, nous allons mesurer le taux d'erreur de ces algorithmes, sur l'ensemble test cette fois-ci.

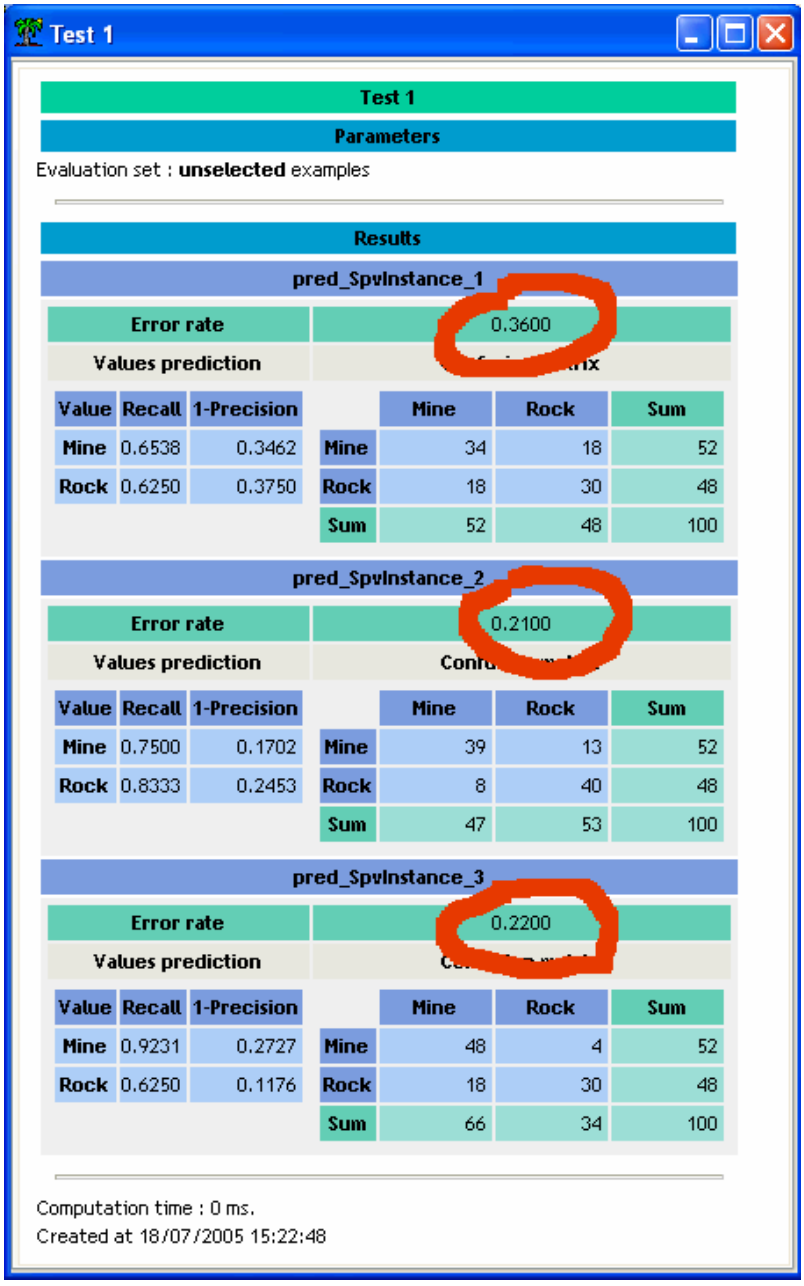
Placez de nouveau un composant DEFINE STATUS, définissez la variable à prédire observée en TARGET, et les variables projetées par les 3 méthodes, générées par l'apprentissage, en INPUT.



Il reste à placer le composant TEST de la palette SPV LEARNING ASSESMENT, par défaut il est paramétré pour calculer les matrices de confusions sur les individus non-sélectionnées, c'est-à-dire l'ensemble test[2].

[2] Notons qu'il est également possible de paramétrer ce composant de manière à ce qu'il effectue le calcul sur les individus sélectionnés, dans ce cas, nous devons retrouver les taux d'erreur en resubstitution, calculés sur l'ensemble d'apprentissage.

Les résultats sont assez amusants, ils illustrent bien le phénomène de sur-apprentissage, ce qui était prévisible au vu des caractéristiques de nos données : beaucoup de variables (60 descripteurs) comparativement à l'effectif (108 observations pour l'apprentissage).



Le taux d'erreur en test est de :

- 36% pour l'analyse discriminante (SUPERVISED LEARNING 1) ;
- 21% pour les SVM (SUPERVISED LEARNING 2) ;
- 22% pour les K-NN (SUPERVISED LEARNING 3).

Il apparaît que l'analyse discriminante linéaire a particulièrement souffert sur cet exemple, les SVM (linéaire) et les K-NN présentent en revanche des performances comparables.