Objectif

Comparer TANAGRA, ORANGE et WEKA lors de la construction d'un arbre de décision. Evaluation de la précision à l'aide de la validation croisée.

S'agissant de la construction d'un arbre de décision, quel que soit le logiciel utilisé, nous devons impérativement passer par les étapes suivantes :

- Importer les données dans le logiciel ;
- Définir le problème à résoudre, c.-à-d. choisir la variable à prédire (l'attribut « classe ») et les descripteurs ;
- Sélectionner la méthode d'induction d'arbres de décision, selon les logiciels et selon les implémentations, les résultats peuvent être différents ;
- Lancer l'apprentissage et visualiser l'arbre ;
- Utiliser la validation croisée pour évaluer la qualité du modèle induit.

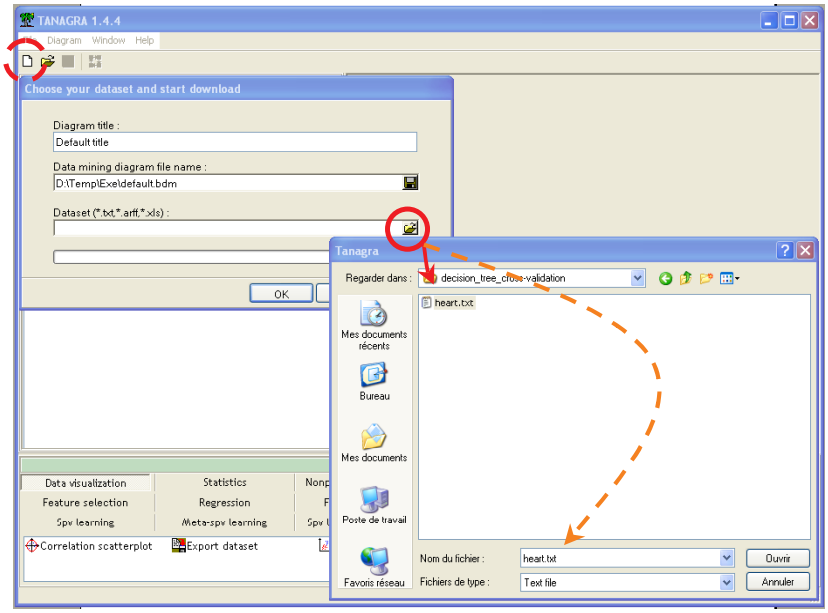
Fichier

Nous utilisons le fichier HEART.TXT (source UCI IRVINE), certaines variables ont été supprimées, le fichier contient 270 observations.

Construire un arbre avec TANAGRA

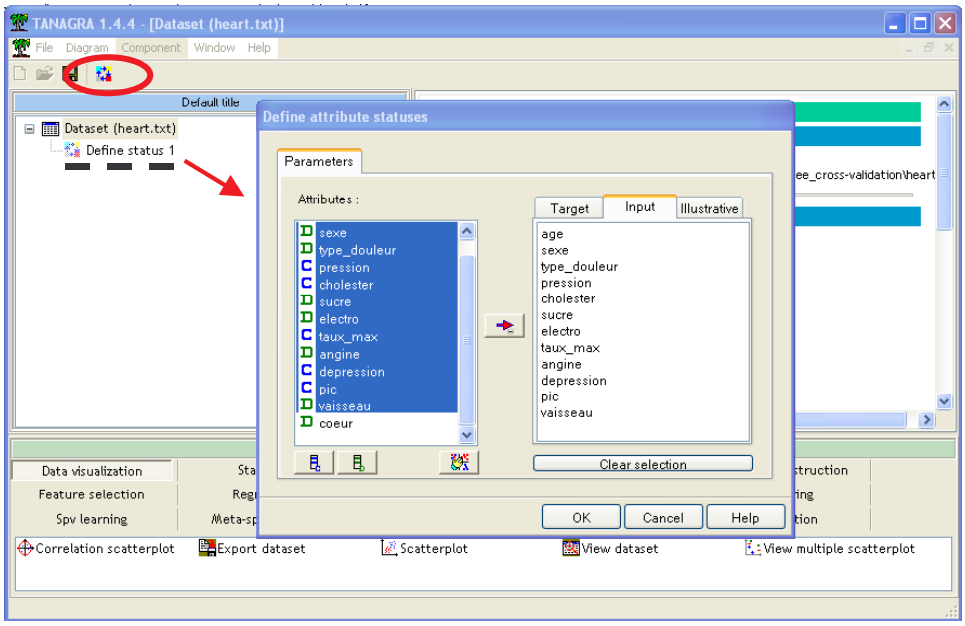
Charger les données

Première étape toujours, après le lancement du logiciel, nous devons charger les données. Nous activons le menu FILE/NEW pour créer un nouveau diagramme et nous sélectionnons le fichier HEART.TXT.



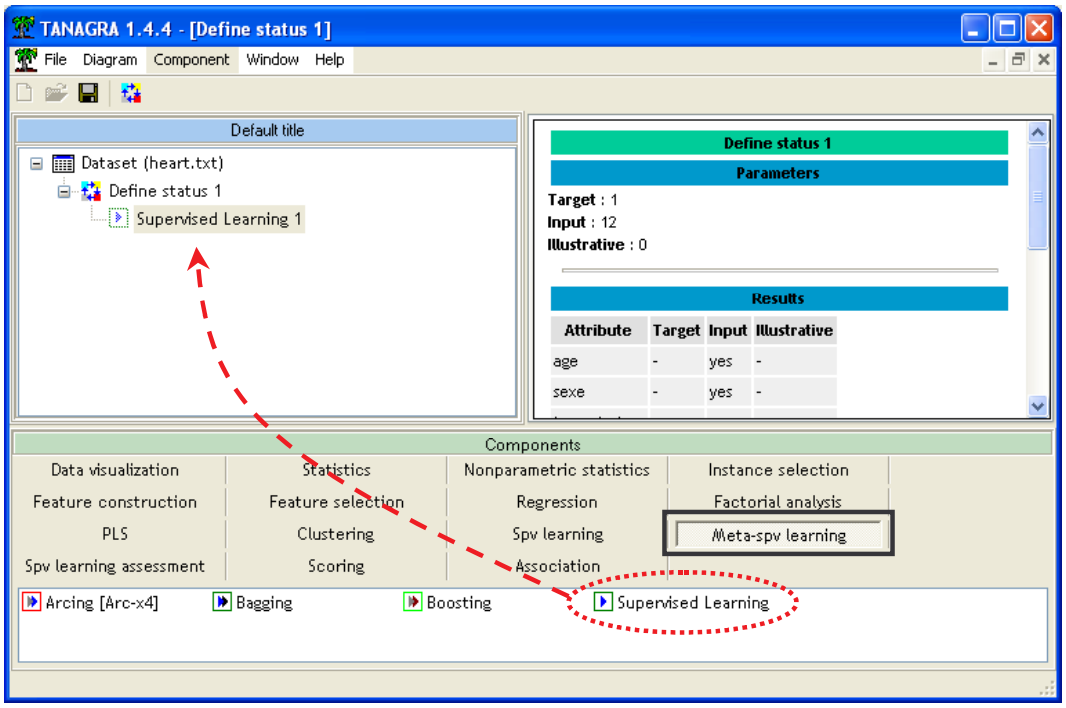
Sélectionner les variables

Nous ajoutons le composant DEFINE STATUS dans le diagramme en cliquant sur le raccourci dans la barre d'outils. Nous plaçons alors en TARGET la variable CŒUR, en INPUT les autres variables.

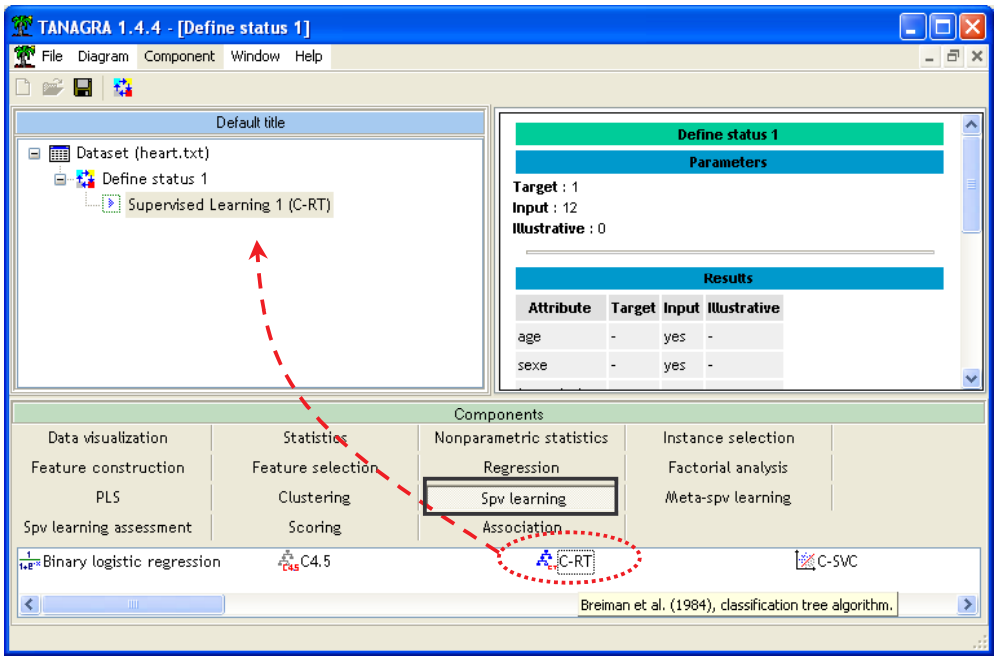


Choisir la méthode d'apprentissage

Nous voulons maintenant insérer la méthode *Classification and Regression Tree* (Breiman et al.) dans le diagramme. Cette opération se fait en 2 étapes : (a) d'abord insérer la méthode générique d'apprentissage, nous parlons de méta-apprentissage, nous prenons l'apprentissage simple dans l'onglet META SPV LEARNING …

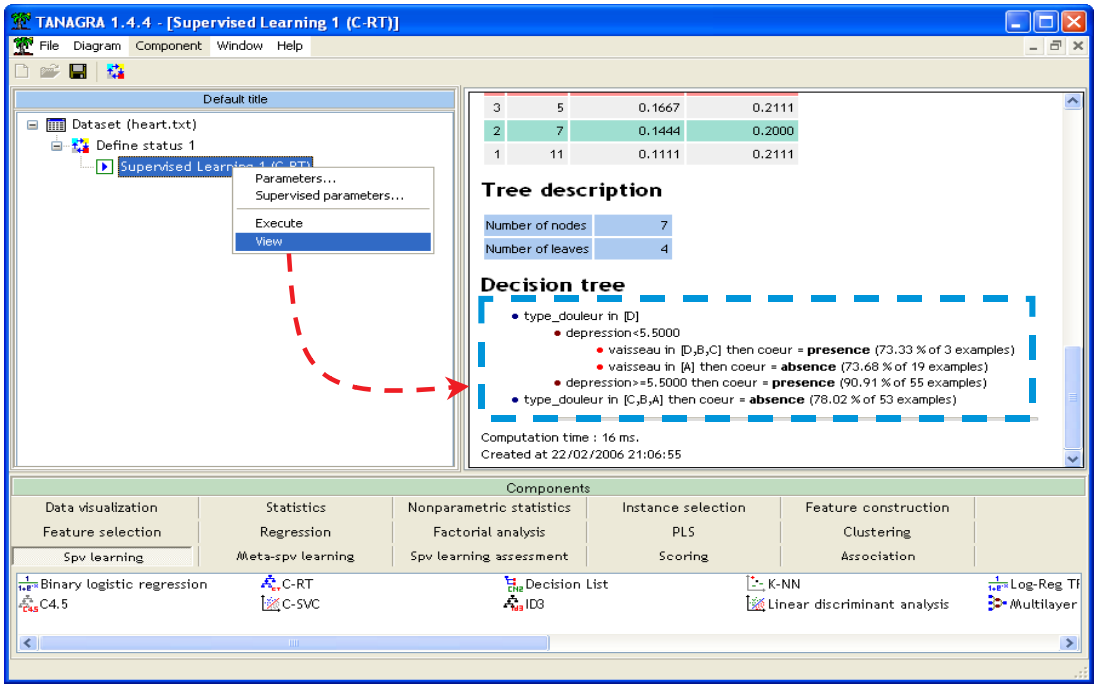


(b) puis insérer la méthode d'apprentissage proprement dite, C-RT dans l'onglet SPV LEARNING.



Exécution

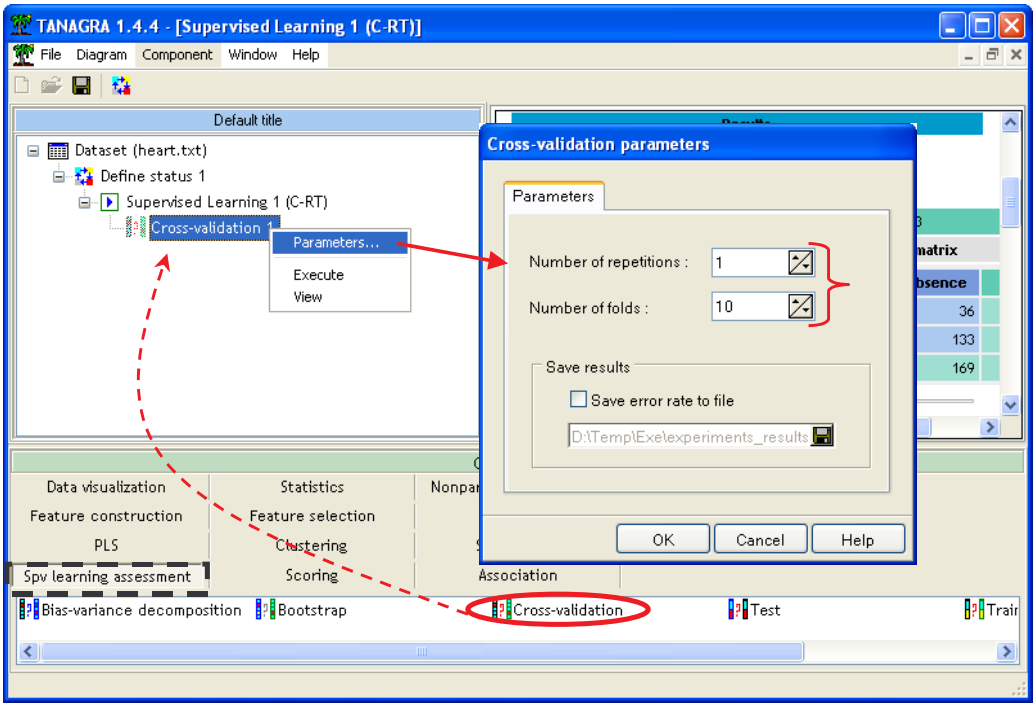
Pour voir les résultats, nous devons activer le menu VIEW du dernier composant du diagramme, l'arbre apparaît dans la fenêtre de droite[1] : le taux d'erreur en resubstitution (calculée sur les données en apprentissage) est de 19.63% ; l'arbre comporte 4 feuilles (4 règles).



[1] L'arbre apparaît sous une forme textuelle dans TANAGRA. Pour obtenir une représentation graphique, il faudrait plutôt le logiciel SIPINA du même auteur (http://eric.univ-lyon2.fr/~ricco/sipina.html).

Evaluation en validation croisée

Nous savons que le taux d'erreur calculé sur l'échantillon d'apprentissage est trop optimiste, nous utilisons une méthode de ré-échantillonnage pour obtenir une estimation plus fidèle de la « vraie » valeur de l'erreur. Pour ce faire, nous ajoutons le composant CROSS-VALIDATION à la suite de la chaîne de traitement (onglet SPV LEARNING ASSESMENT), puis nous la paramétrons en activant son menu PARAMETERS. Nous fixons à 10 le nombre de portions (FOLDS), cette évaluation est lancée une seule fois (REPETITION).



Après exécution du composant, nous obtenons le taux d'erreur estimé en validation croisée, il est de 24.81%

Cross-validation 1

Parameters

Cross-validation parameters	
Folds	10
Trials	1

Results

CV error rate

Range	
MIN	0.2481
MAX	0.2481
Trial	**Err rate**
1	0.2481

Overall cross-validation error rate

Error rate	0.2481

Values prediction

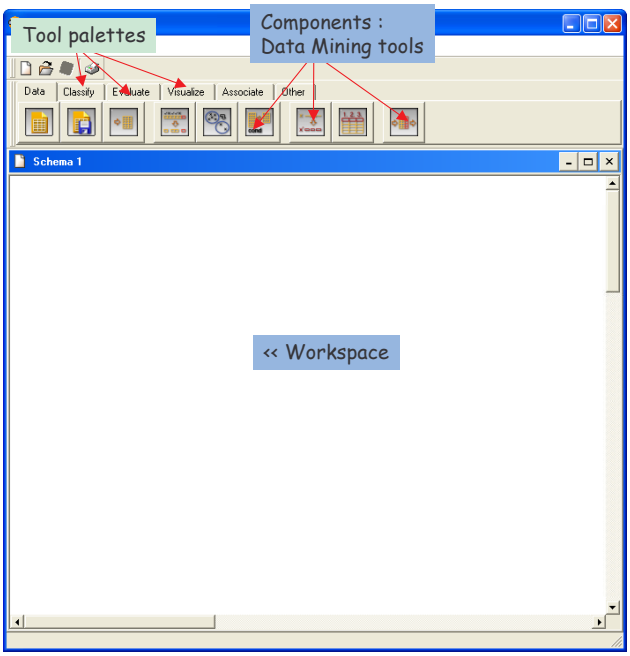
Value	Recall	1-Precision
presence	0.6250	0.2268
absence	0.8533	0.2601

Confusion matrix

	presence	absence	Sum
presence	75	45	120
absence	22	128	150
Sum	97	173	270

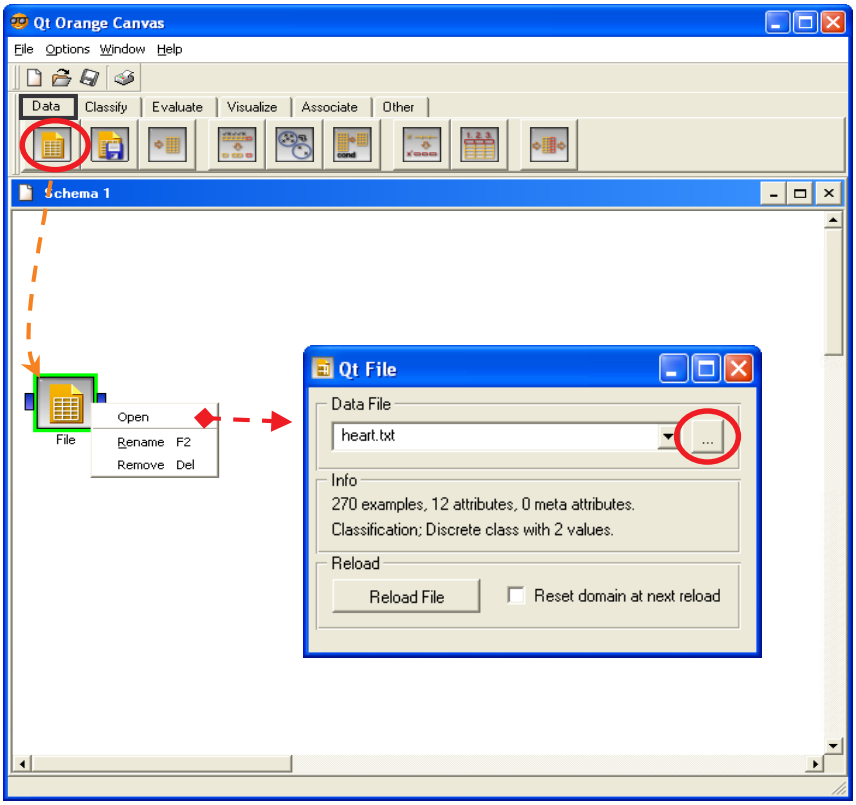
Construire un arbre avec ORANGE

ORANGE propose une interface composée de deux parties distinctes : un espace pour définir les traitements ; une palette d'outils située dans la partie haute de la fenêtre principale.



Importer les données

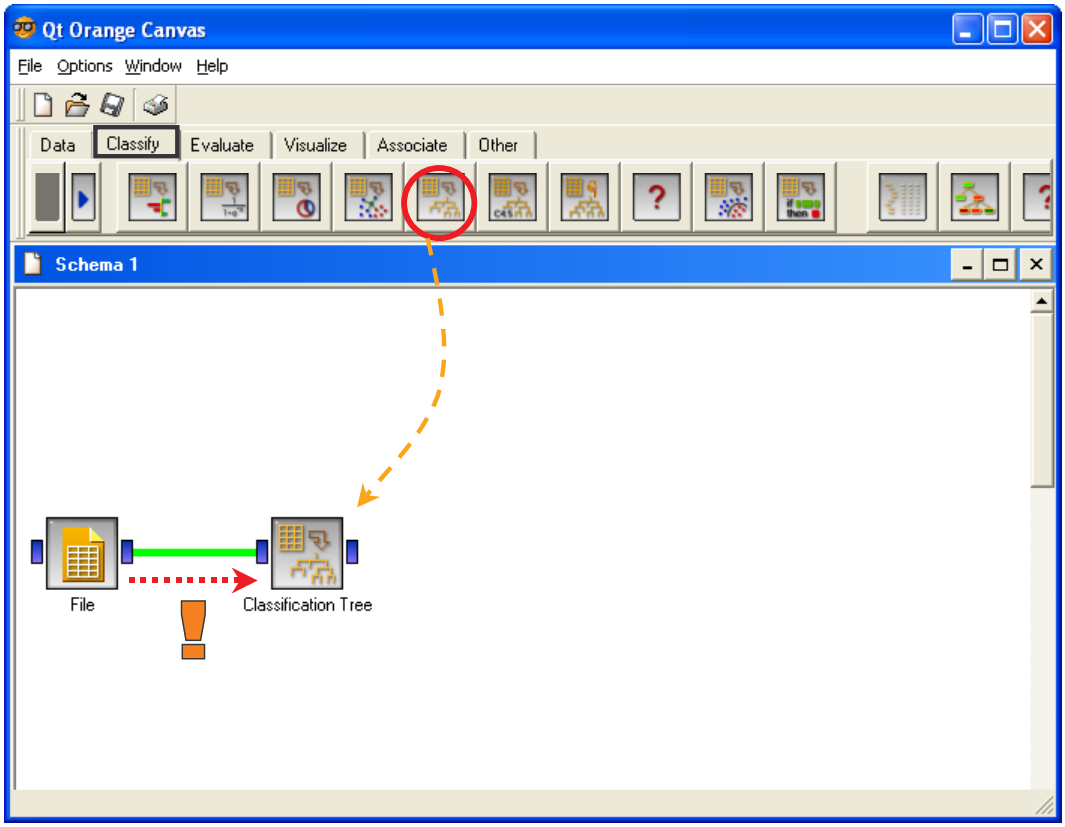
ORANGE peut importer des données au format texte (séparateur tabulation). Il suffit de cliquer sur l'outil pour qu'il s'insère automatiquement dans l'espace de travail. Nous sélectionnons donc notre fichier HEART.TXT en cliquant sur le menu contextuel OPEN.



Placer la méthode d'apprentissage

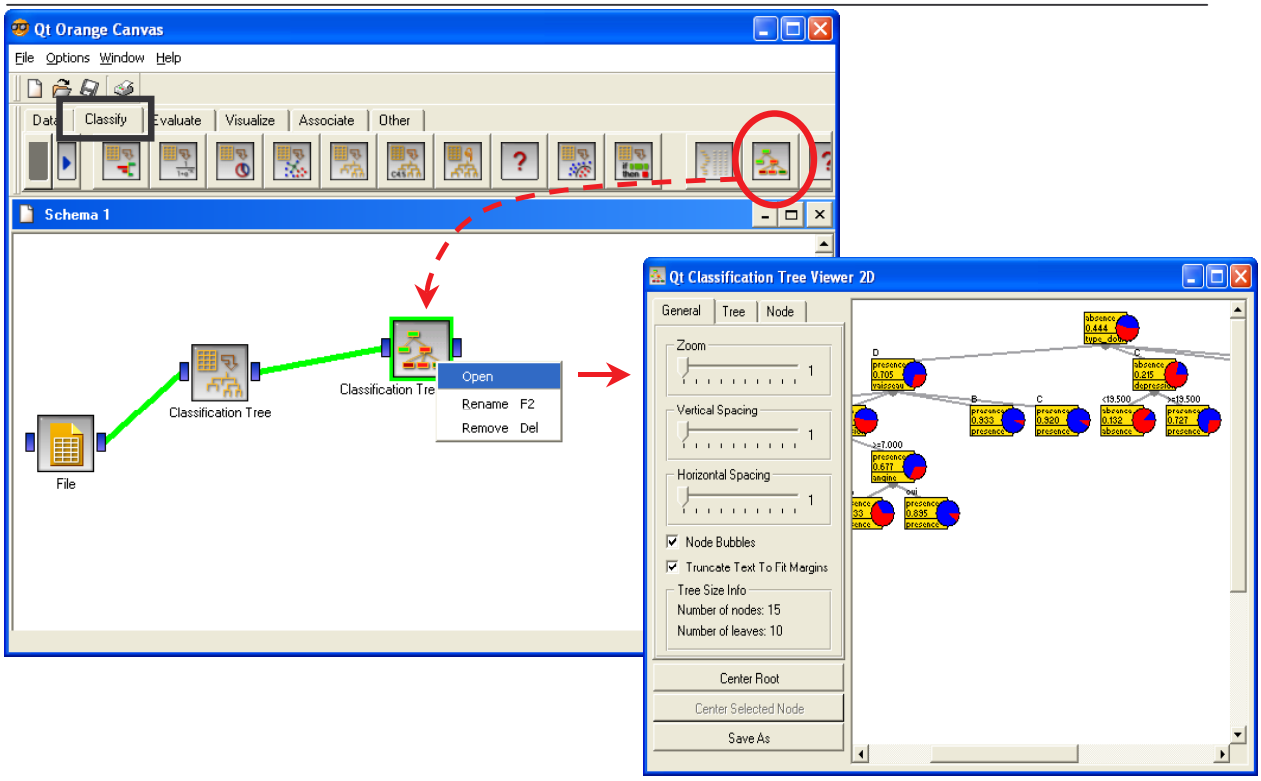
Par défaut, la dernière colonne représente la variable à prédire – TARGET -- dans ORANGE, toutes les autres variables sont les prédictives (INPUT). Il est possible de modifier cette sélection par défaut, mais ce ne sera pas nécessaire ici car notre fichier est configuré correctement.

Nous pouvons enchaîner directement avec le composant d'apprentissage que nous allons chercher dans l'onglet CLASSIFY. Nous ne modifions pas les paramètres du composant, nous relions directement le composant DATA avec ce dernier, ce qui lance automatiquement l'exécution des traitements.



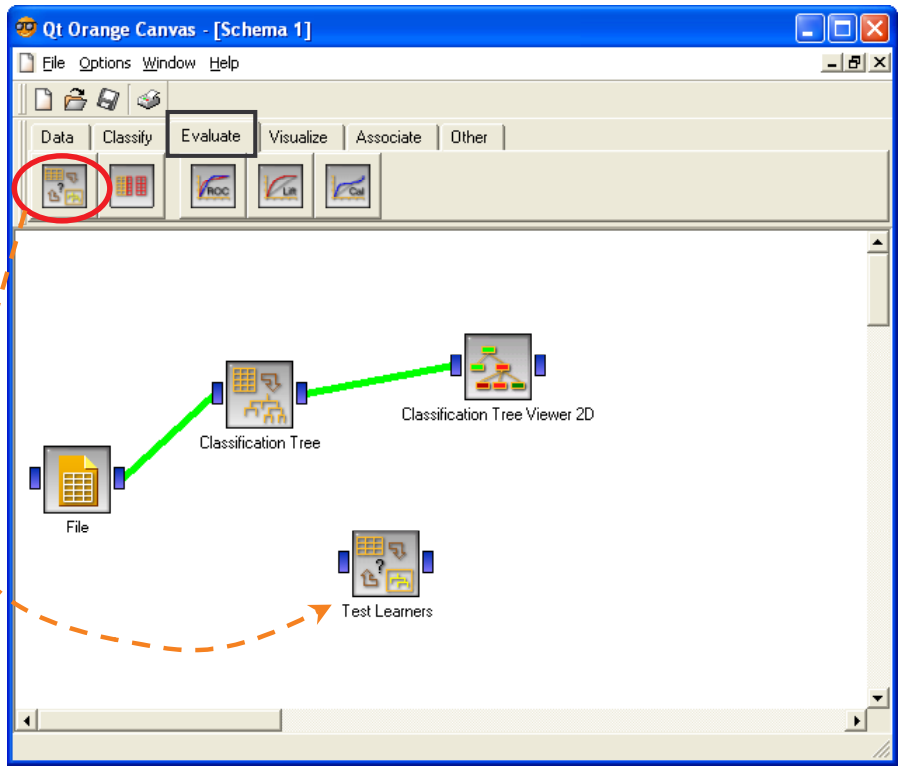
Voir les résultats

ORANGE propose deux outils pour visualiser l'arbre de décision : un outil textuel, particulièrement adapté si l'arbre de grande taille ; un outil graphique, autrement plus attractif mais inopérant dès que l'arbre comporte de nombreux sommets. Dans notre cas, nous choisissons donc l'outil graphique CLASSIFICATION TREE VIEWER 2D (onglet CLASSIFY). Après lui avoir branché le composant précédent, nous pouvons voir l'arbre en cliquant sur le menu OPEN. L'arbre comporte 10 feuilles (10 règles).

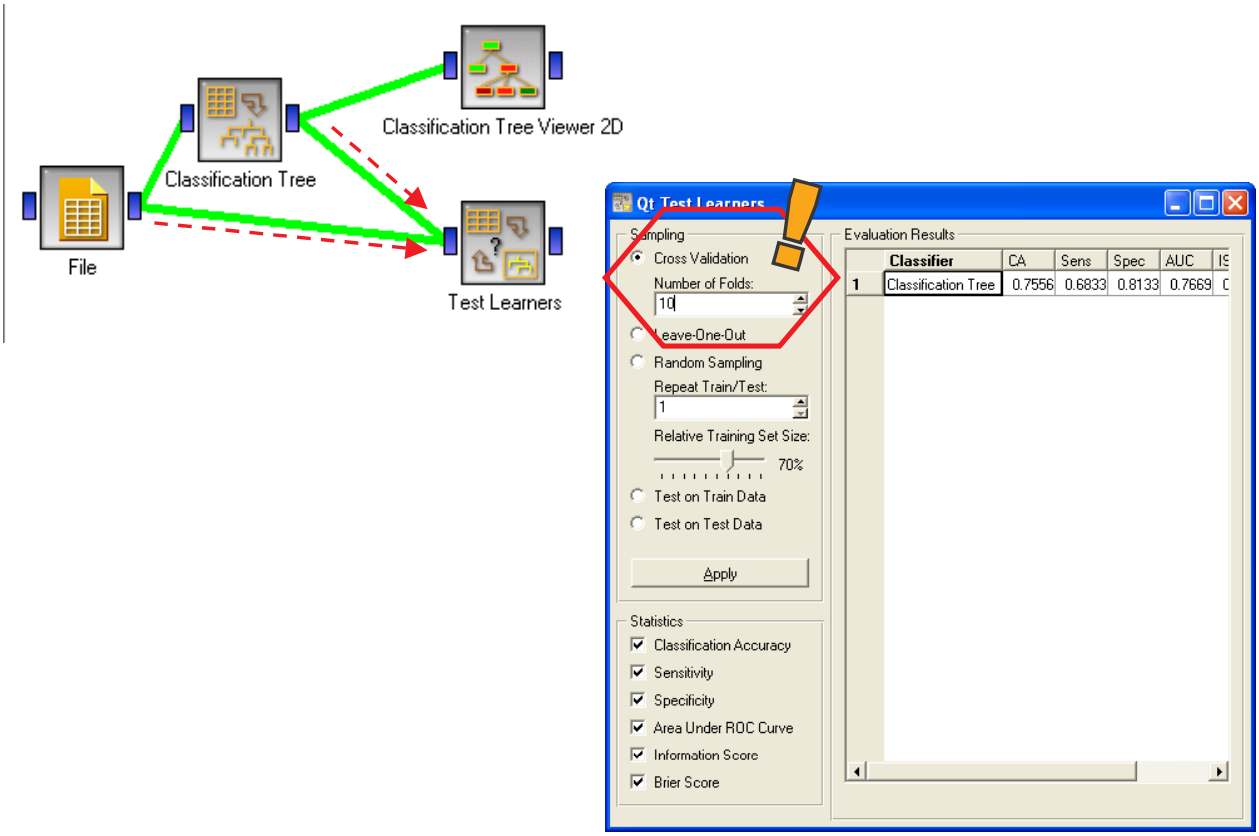


Validation croisée

Le composant TEST LEARNERS (onglet EVALUATE) est chargé de calculer les performances de l'arbre. Il regroupe les différentes méthodes d'évaluation, y compris la validation croisée qui nous intéresse dans ce didacticiel. Nous plaçons le composant dans le diagramme.



Pour que ce composant soit opérationnel, nous devons lui spécifier la source de données et la méthode d'apprentissage utilisée pour les calculs, notons çà ce sujet qu'il est possible de brancher simultanément plusieurs méthodes d'apprentissage, ce qui permet de réaliser, très facilement, des comparaisons de performances. Nous effectuons donc les branchements idoines, puis nous affichons les résultats toujours à l'aide du menu OPEN.

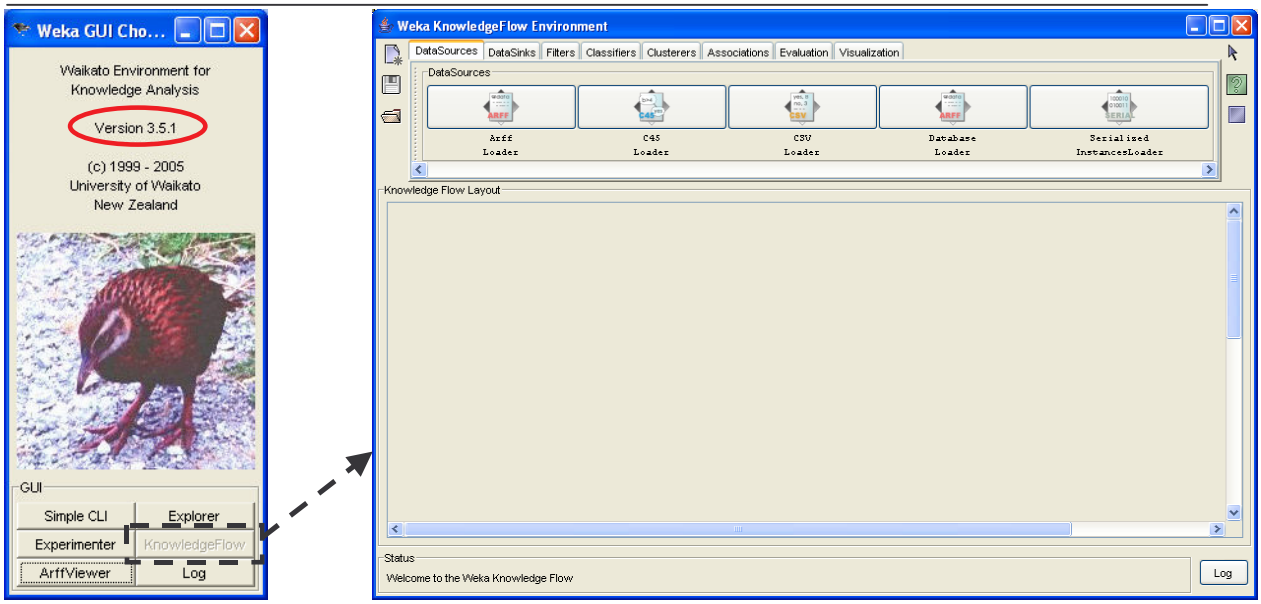


Le taux de biens classés est de 75.56%, ce qui correspond à un taux d'erreur de 24.44%, tout à fait comparable avec ce qui a été obtenu avec TANAGRA. Le contraire eût été inquiétant même si les méthodes ne sont pas exactement les mêmes.

D'autres indicateurs sont disponibles (sensibilité, etc.). Il est possible de modifier directement la méthode d'évaluation dans la fenêtre de résultats (leave-one-out, apprentissage-test, etc.) et de relancer le calcul (APPLY) sans passer par un autre composant.

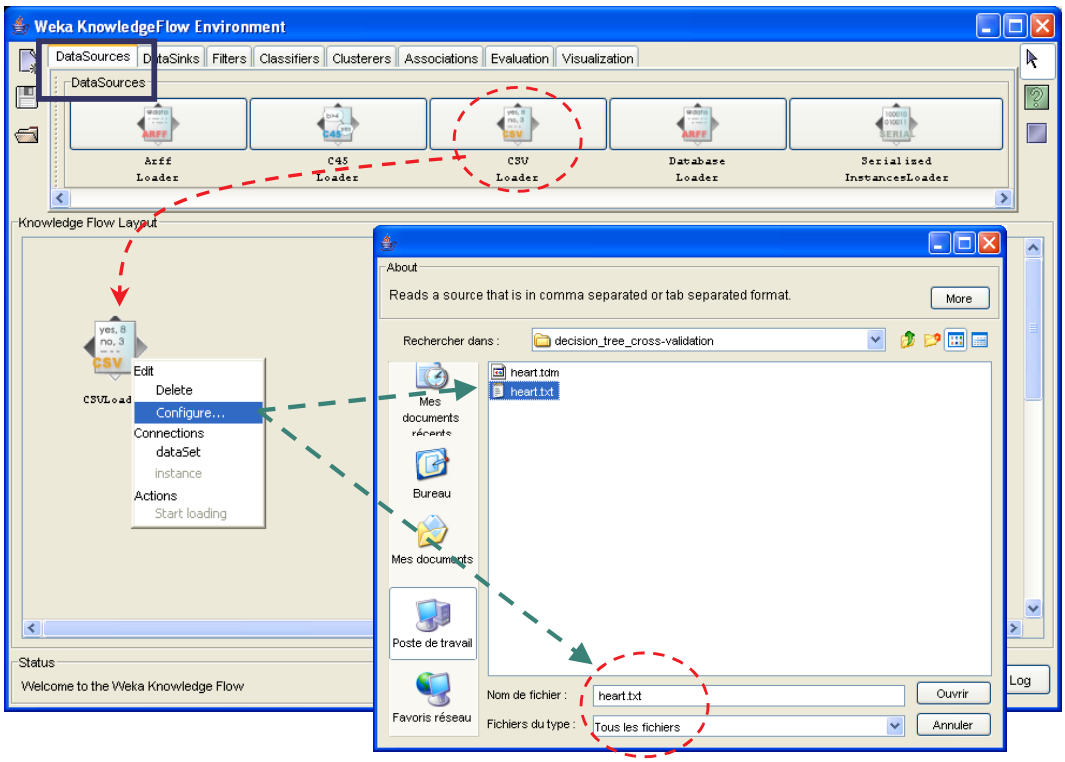
Construire un arbre avec WEKA

Au lancement de WEKA, un panneau permet de choisir le mode d'exécution du logiciel. Nous choisissons le mode **KNOWLEDGE FLOW**. Nous avons utilisé la version **3.5.1** dans ce didacticiel. Nous parvenons alors dans l'espace de travail dans lequel nous allons définir nos traitements. Dans la partie haute, nous trouvons les icônes organisées dans des palettes, ils représentent les opérateurs de traitement.



Charger les données

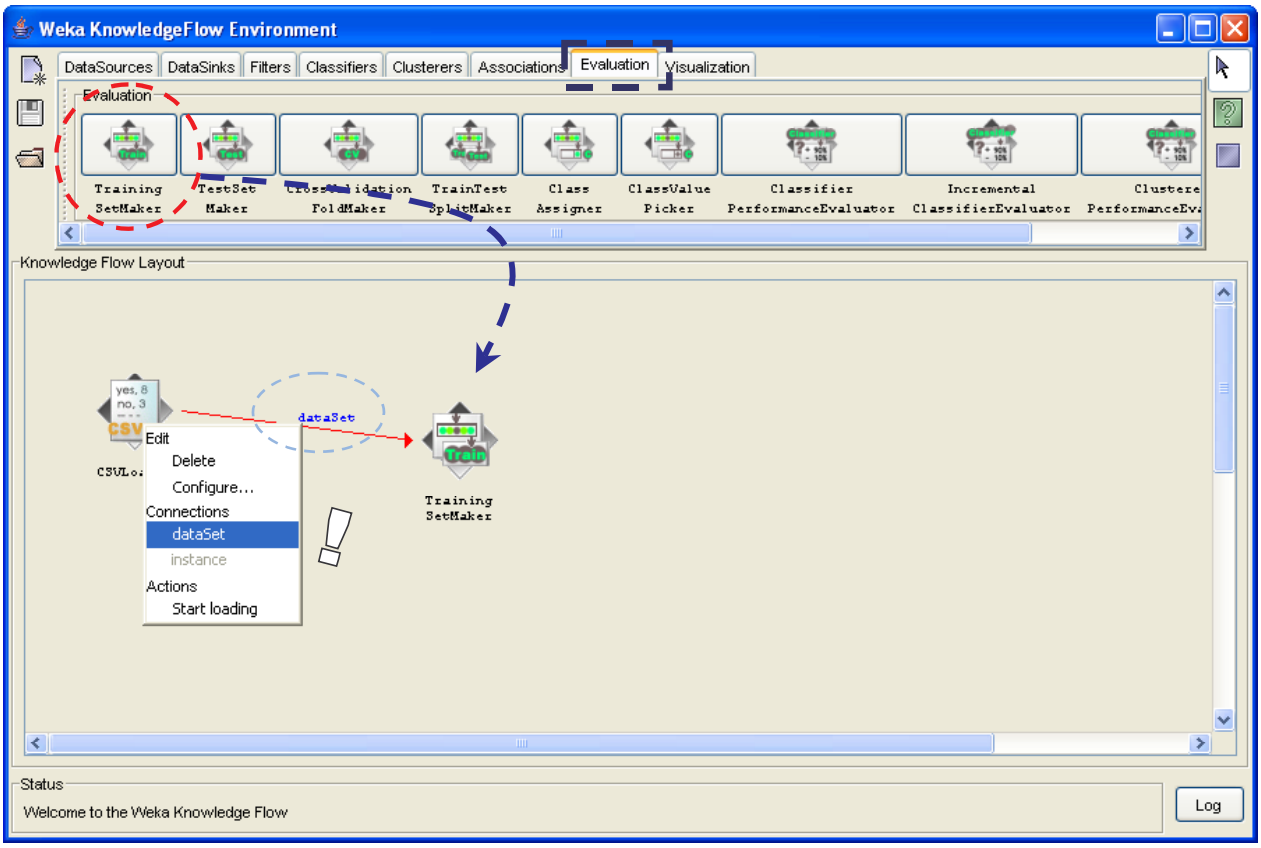
Le composant CSV LOADER (onglet DATASOURCES) permet de charger les données. Nous ajoutons donc ce composant dans notre espace de travail, nous pouvons le paramétrer à l'aide de l'option CONFIGURE de son menu contextuel. Nous chargeons le fichier HEART.TXT.



Définir l'apprentissage

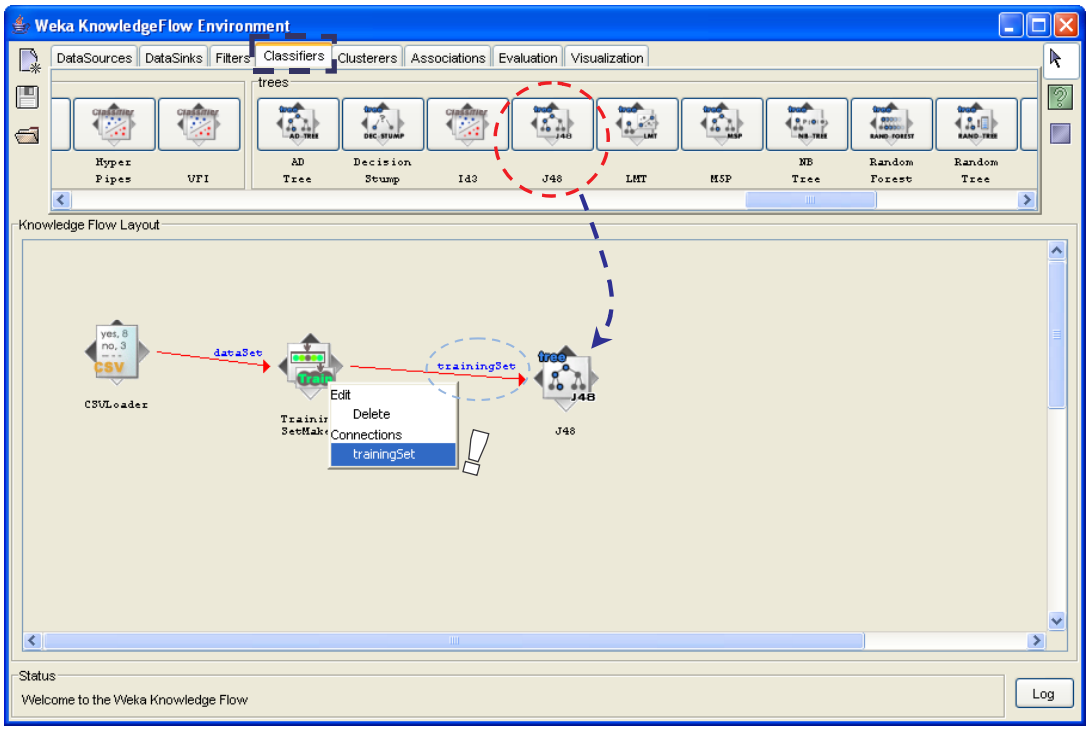
Par défaut, tout comme dans ORANGE, la variable à prédire correspond à la dernière colonne du fichier ; toutes les autres variables représentent les variables prédictives. Ce qui est notre cas.

En revanche, il est obligatoire de spécifier explicitement les individus à utiliser pour la construction du modèle. Il s'agit de la totalité du fichier pour nous. Nous utilisons donc le composant TRAINING SET MAKER (onglet EVALUATION). Nous le plaçons dans le diagramme ou nous lui relions le composant précédent en précisant que c'est la connexion DATASET que nous voulons utiliser. Cette particularité de WEKA est très intéressante, cela permet d'exclure toute mauvaise interprétation lorsque nous associons les sommets dans le diagramme.

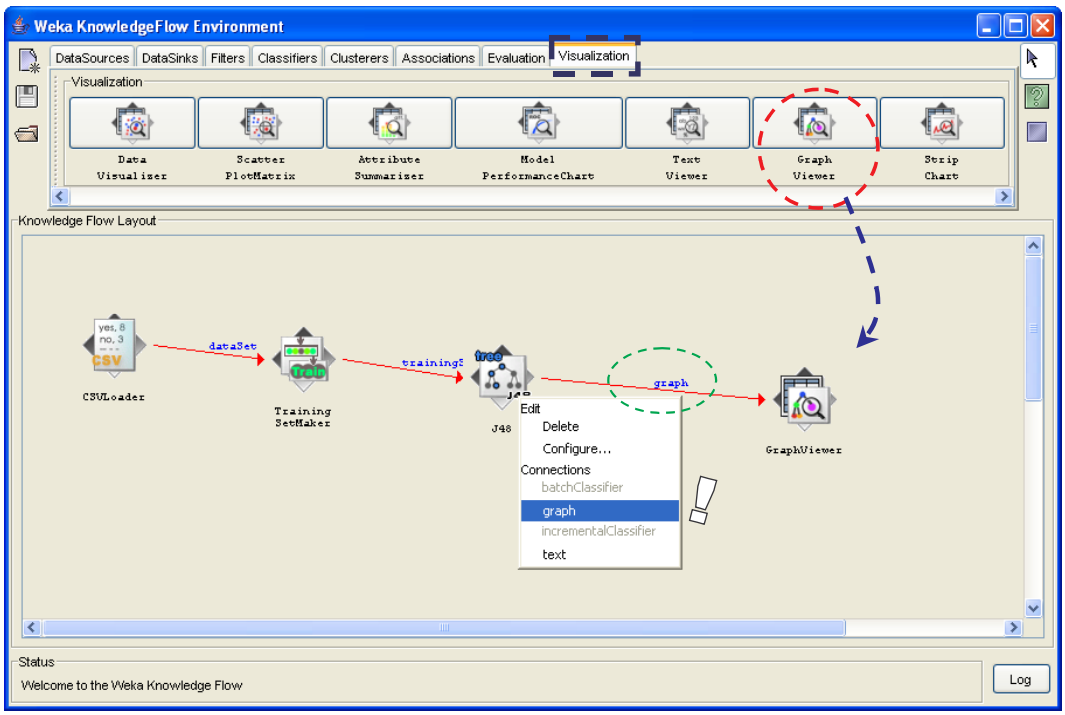


L'étape suivante consiste à placer la méthode d'induction d'arbre de décision. La panoplie de WEKA est considérable. Nous choisissons la méthode J48 (onglet CLASSIFIERS) qui est un peu un C4.5 maison avec quelques améliorations. Nous lui relions alors le composant TRAINING SET MAKER en précisant qu'il s'agit bien des données d'apprentissage.

Voir l'arbre de décision

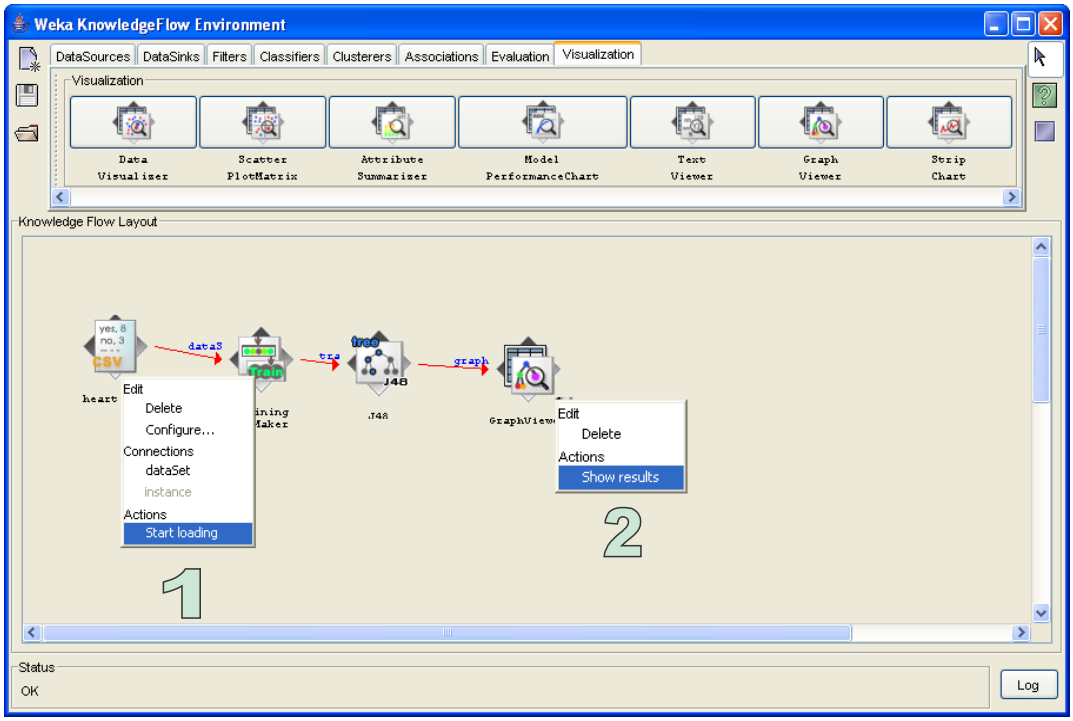


A ce stade, les calculs ne sont pas encore exécutés, il nous faut au préalable placer le composant de visualisation. Il en existe deux dans WEKA : le premier propose une représentation textuelle, intéressante lorsque l'arbre est de grande taille ; le second est graphique, plus avenant mais très lent dès que l'arbre atteint une taille conséquente. Nous choisissons cet outil, notre arbre comporte peu de feuilles. Nous plaçons l'outil GRAPH VIEWER (onglet VISUALISATION) dans le diagramme, nous lui raccordons l'arbre en veillant à utiliser la connexion de type GRAPH.

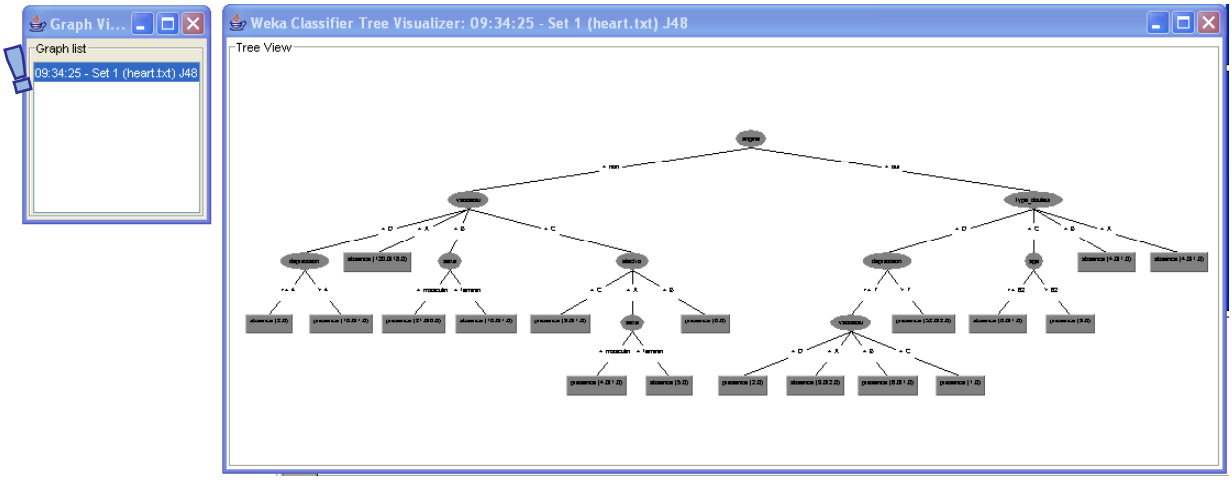


Pour lancer les calculs, il faut revenir sur le premier sommet, les données, puis activer le menu START LOADING : tous les composants de la chaîne s'activent à tour de rôle, il est possible de suivre le déroulement des opérations dans la barre de message STATUS.

Enfin, les calculs réalisés, pour afficher les résultats, nous activons le menu SHOW GRAPH du composant GRAPH VIEWER.

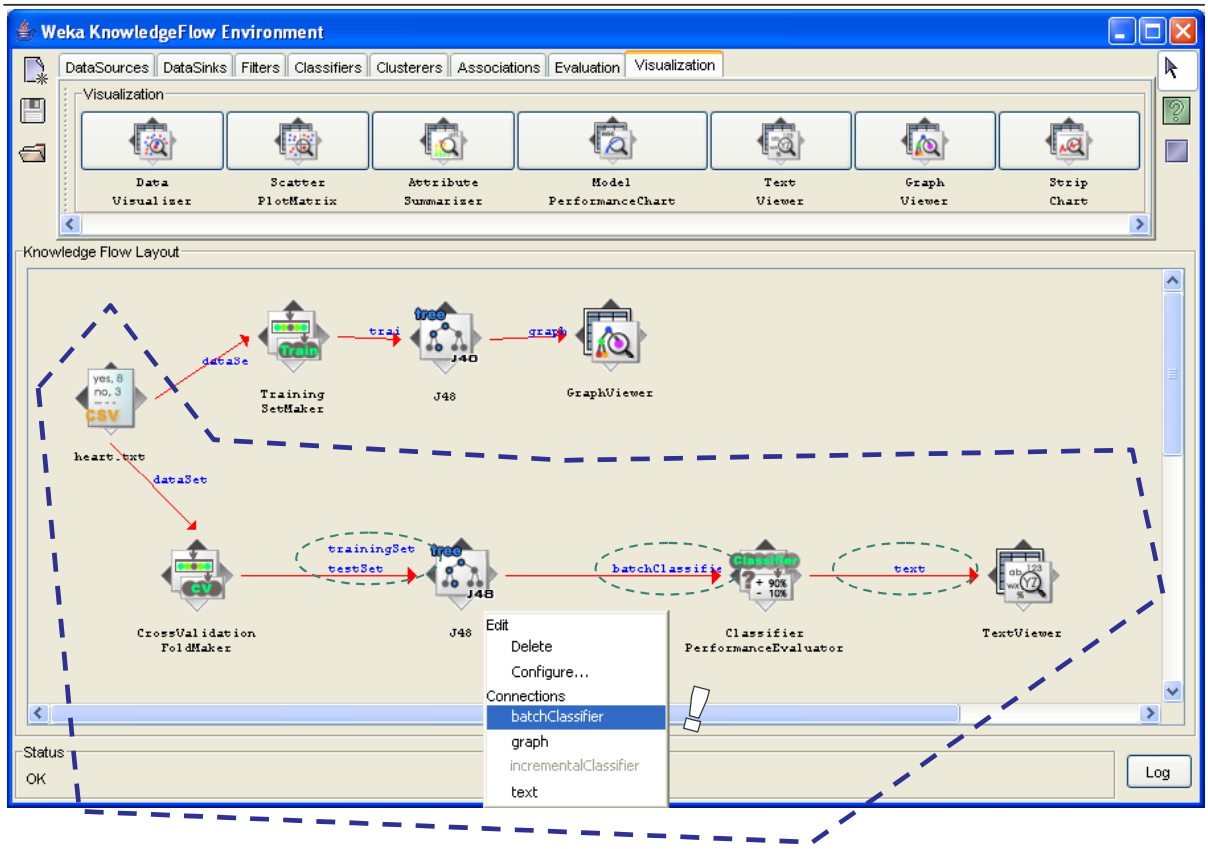


Nous obtenons l'arbre de décision, il comporte 18 feuilles.



Validation croisée

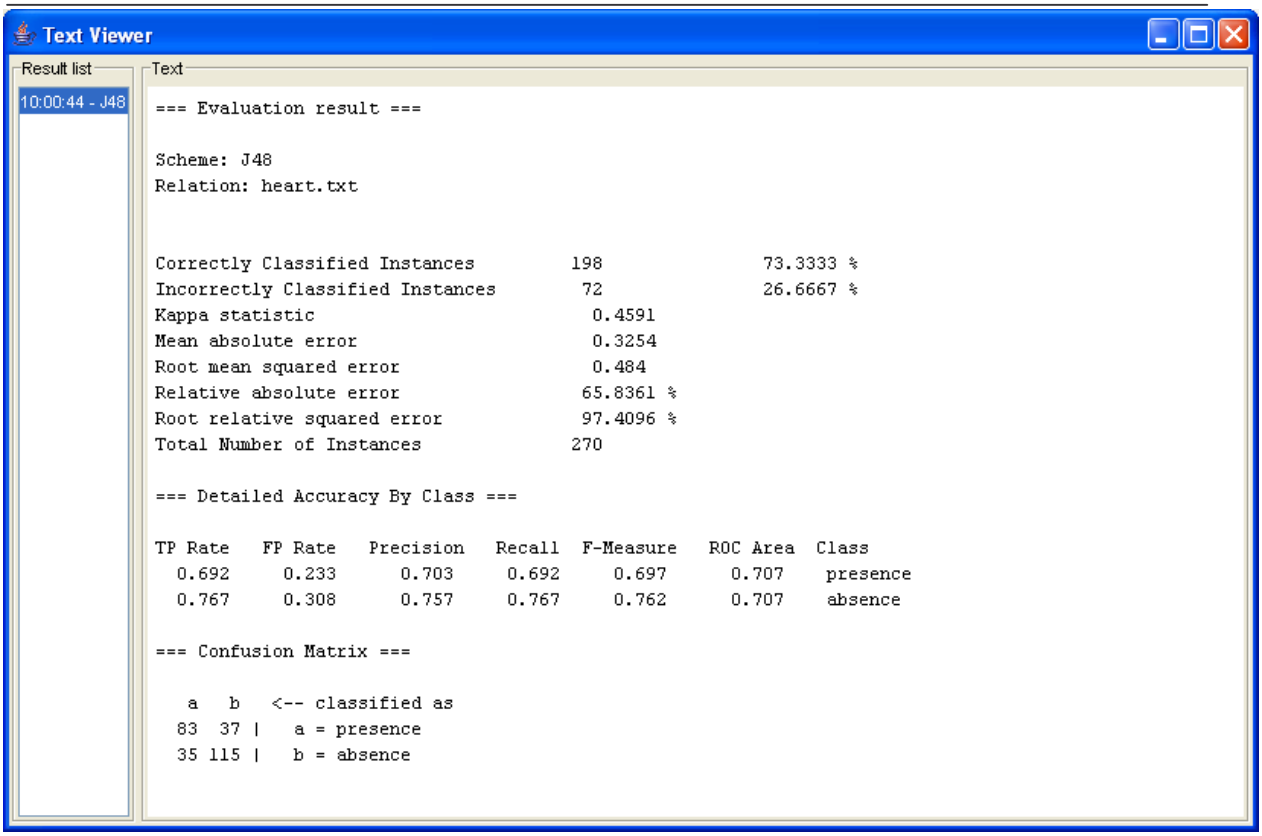
Il nous reste à estimer la précision de cet arbre en nous appuyant sur la validation croisée. La procédure est un peu plus complexe dans WEKA, il faut redéfinir un diagramme en réalité, en modifiant la définition des échantillons et en la complétant avec des composants d'évaluation de performances.



Nous devons donc insérer successivement :

- CROSS VALIDATION FOLD MAKER (onglet EVALUATION) qui se charge du découpage des données, nous lui connectons le composant de données (connexion DATASET). Par défaut, le nombre de portions est égal à 10, il est possible de le modifier à l'aide du menu CONFIGURE.
- Le composant J48 (CLASSIFY), attention, ***il est impératif d'adopter les mêmes paramètres d'apprentissage, sinon l'évaluation est faussée***. Ce composant prend en entrée 2 connexions, une première pour les données d'apprentissage (TRAINING SET), la seconde pour les données de test (TEST SET).
- Le composant CLASSIFIER PERFORMANCE EVALUATOR (EVALUATION) chargé de collecter les séries des calculs de chaque FOLD. Nous utilisons la sortie BATCH CLASSIFIER du composant arbre de décision.
- Et enfin, le composant TEXT VIEWER affiche les résultats.

Nous lançons l'exécution à partir du menu START LOADING du composant de données, nous pouvons suivre la progression des calculs dans la barre de STATUS. Pour consulter les résultats, nous cliquons sur le menu SHOW RESULTS du dernier composant.



Le taux d'erreur estimé est de 26.67%, d'autres indicateurs sont disponibles.

Notons une particularité de WEKA, il est possible de visualiser les 10 arbres de décision produits lors du processus de validation croisée. Il faudrait pour cela brancher un composant TEXT VIEWER à la sortie du composant J48, nous pouvons ainsi nous rendre compte des éventuelles différences entre les arbres et juger de la stabilité des calculs.

Conclusion

La validation croisée est une procédure couramment utilisée pour évaluer les performances des méthodes d'apprentissage supervisé, surtout lorsque les bases sont de taille réduite. Nous constatons qu'avec des stratégies différentes, ces trois logiciels permettent de la mettre en oeuvre assez facilement.

Lorsque nous voulons construire un arbre de décision sur des bases de données de grande taille -- plusieurs centaines de milliers d'individus et une centaine de variables – les logiciels réagissent très différemment. Les capacités de calcul dépendent fortement de la gestion de la mémoire de la technologie utilisée, WEKA (JAVA) et ORANGE (PYTHON) semblent avoir des problèmes par rapport à WINDOWS. Ces deux logiciels ont en revanche l'avantage de pouvoir s'exécuter sur d'autres plates-formes, ce qui n'est pas le cas de TANAGRA.