1 Objectif

Comprendre le modèle d'indépendance conditionnelle (Classifieur Bayesien Naïf).

Le classifieur bayesien naïf est une méthode d'apprentissage supervisé qui repose sur une hypothèse simplificatrice forte : les descripteurs (Xj) sont deux à deux indépendants conditionnellement aux valeurs de la variable à prédire (Y)[1]. Pourtant, malgré cela, il se révèle robuste et efficace. Ses performances sont comparables aux autres techniques d'apprentissage. Diverses raisons sont avancées dans la littérature. Dans ce document, nous mettrons en avant une explication basée sur le biais de représentation. Le modèle d'indépendance conditionnel est ni plus ni moins qu'un classifieur linéaire, au même titre que l'analyse discriminante linéaire ou la régression logistique. Seul diffère le mode d'estimation des coefficients de la fonction de classement.

De fait, nous introduisons dans Tanagra (version 1.4.36 et ultérieure) une nouvelle présentation des résultats qui rend le modèle plus explicite et facilite ainsi son déploiement. En effet, si le classifieur bayesien naïf est très utilisé dans le monde de la recherche, il est en revanche très peu répandu parmi les praticiens du Data Mining dans le monde de l'entreprise. Dans le premier cas, les chercheurs constatent surtout qu'il est très aisé de le programmer et de le mettre en œuvre, ses paramètres sont faciles à estimer, son apprentissage est très rapide y compris sur de très grandes bases en nombre d'observations et en nombre de variables, ses performances en classement sont raisonnablement bonnes. Dans le second cas, le néophyte ne disposant pas de modèle explicite facile à interpréter et à déployer, ne comprend pas l'intérêt d'une telle technique.

Dans la première partie de ce tutoriel, nous présentons tout d'abord brièvement les aspects théoriques relatifs à la méthode. Puis, nous l'implémentons à l'aide du logiciel **Tanagra 1.4.36**. Nous comparerons les résultats obtenus (les coefficients de l'hyperplan séparateur) avec ceux de la régression logistique, de l'analyse discriminante et d'un SVM (support vector machine) linéaire. Nous constaterons qu'ils sont étonnamment cohérents, expliquant ainsi la bonne tenue du classifieur bayesien naïf dans la grande majorité des situations.

Dans la seconde partie, nous montrons la mise en œuvre de la technique dans les plusieurs logiciels libres tels que **Weka 3.6.0**, **R 2.9.2**, **Knime 2.1.1**, **Orange 2.0b** et **RapidMiner 4.6.0**. Nous nous attacherons avant tout à lire correctement les résultats. Un des aspects qui dessert souvent la méthode auprès des praticiens du Data Mining.

2 Le classifieur bayesien naïf

Soient $\aleph = (X_1, \ldots, X_J)$ l'ensemble des descripteurs, Y la variable à prédire (l'attribut classe comportant K modalités). Nous considérons qu'ils sont tous catégoriels dans ce document[2]. En

[1] http://en.wikipedia.org/wiki/Naive_Bayes_classifier

[2] La variable à prédire est forcément catégorielle en apprentissage supervisé. Concernant les variables prédictives, nous pouvons les discrétiser (les découper en intervalles) si elles sont initialement quantitatives. Voir http://tutoriels-data-mining.blogspot.com/2010/02/discretisation-comparaison-de-logiciels.html. Même

apprentissage supervisé, pour un individu ω à classer, la règle bayésienne d'affectation optimale revient à maximiser la probabilité a posteriori d'appartenance aux classes c.-à-d.

$$\hat{y}(\omega) = y_{k^*} \Leftrightarrow y_{k^*} = \arg\max_k P[Y = y_k / \aleph(\omega)]$$

La décision repose donc sur une estimation viable de la probabilité conditionnelle P(Y/X). Cette dernière peut s'écrire d'une manière différente

$$P[Y = y_k / \aleph(\omega)] = \frac{P(Y = y_k) \times P[\aleph(\omega)/Y = y_k]}{P[\aleph(\omega)]}$$

Comme l'objectif est de détecter le maximum de cette quantité selon y_k, et que le dénominateur n'en dépend pas, nous pouvons ré écrire la règle d'affectation ci-dessus

$$\hat{y}(\omega) = y_{k^*} \Leftrightarrow y_{k^*} = \arg\max_k P(Y = y_k) \times P[\aleph(\omega)/Y = y_k]$$

2.1 Hypothèse : l'indépendance conditionnelle des descripteurs

La quantité $P(Y = y_k)$ est facile à estimer à partir d'un échantillon d'observations. Il suffit de calculer les proportions de chaque modalité de la variable cible. En pratique, on utilise souvent la « m probabilité estimate » pour « lisser » les estimations sur les petits effectifs. Si n_k est le nombre d'individu de la modalité y_k dans un échantillon de n observations, nous utilisons

$$\hat{P}(Y = y_k) = p_k = \frac{n_k + m}{n + m \times K}$$

Lorsque nous fixons $m = 1$, nous obtenons l'estimateur laplacien des probabilités[3].

La véritable difficulté réside finalement dans la production d'une estimation viable de la quantité $P[\aleph(\omega)/Y = y_k]$. Nous sommes souvent obligés d'introduire des hypothèses pour rendre le calcul réalisable. L'analyse discriminante paramétrique stipule que la distribution est gaussienne[4] ; la régression logistique binaire ($Y \in \{+,-\}$) part sur l'idée que le rapport $\frac{P[\aleph(\omega)/Y = +]}{P[\aleph(\omega)/Y = -]}$ appartient à une famille de lois particulières[5].

---

si le classifieur bayesien naïf sait appréhender les descripteurs quantitatifs, il semble qu'il soit préférable de les discrétiser systématiquement en pratique, les performances du modèle prédictif n'en seront que meilleures. Voir http://www.springerlink.com/content/8703dcg1u8gek4r5/ pour les aspects théoriques, cette stratégie est systématisée dans certains outils, http://msdn.microsoft.com/en-us/library/ms174806.aspx

[3] Voir http://eric.univ-lyon2.fr/~ricco/doc/Graphes_Induction_These_Rakotomalala_1997.pdf pour la justification de ces formulations (page 57, section 3.4.3).

[4] http://fr.wikipedia.org/wiki/Analyse_discriminante_linéaire

[5] http://fr.wikipedia.org/wiki/Régression_logistique

Dans le cadre du classifieur bayesien naïf, on considère que les descripteurs sont deux à deux indépendants conditionnellement aux valeurs de la variable cible. Par conséquent,

$$P[\aleph(\omega)/Y = y_k] = \prod_{j=1}^{J} P[X_j(\omega)/Y = y_k]$$

Le nombre de paramètres à estimer est réduit de manière drastique. Pour une variable quelconque X comportant L valeurs, nous utiliserons l'estimation suivante

$$\hat{P}[X = l/Y = y_k] = p_{l/k} = \frac{n_{kl} + m}{n_k + m \times L}$$

Usuellement, nous fixons m = 1. On peut toujours vouloir produire une valeur « optimale » de la constante m, mais c'est assez illusoire. Il faut surtout qu'elle soit supérieure à o pour éviter les probabilités estimées nulles de P (X_j / Y) qui auraient pour conséquence de rendre caduc le calcul de la probabilité conditionnelle $P[\aleph(\omega)/Y = y_k]$.

Traditionnellement, nous passons par les logarithmes lors de la mise en œuvre de la méthode, surtout lorsque le nombre de descripteurs est élevé. En effet, le produit de nombreuses valeurs inférieures à 1 (les probabilités conditionnelles estimées) peut rapidement provoquer des débordements de capacités, même lorsque nous utilisons des flottants à double précision dans notre implémentation. La règle d'affectation devient

$$\hat{y}(\omega) = y_{k^*} \Leftrightarrow y_{k^*} = \arg\max_k \left\{ \ln P(Y = y_k) + \sum_{j=1}^{J} \ln P[X_j(\omega)/Y = y_k] \right\}$$

Dans ce qui suit, nous appellerons fonction de classement $d(y_k, \aleph)$ la quantité entre accolades dans l'équation ci-dessus.

2.2 Exemple numérique

2.2.1 Calcul des probabilités conditionnelles

Nous traitons un problème de prédiction de maladie cardiaque. La variable à prédire est DISEASE (positive + vs. négative -). Les descripteurs sont EXANG (yes, no) et CHEST_PAIN (asympt, atyp_angina, non_anginal, typ_angina).

Nous calculons dans un premier temps les fréquences absolues des classes.

Nombre de disease	
disease	Total
positive	104
negative	182
Total	286

Nous en déduisons l'estimation des probabilités (avec l'estimateur laplacien, m = 1)

$$p_+ = \frac{104+1}{286+2} = 0.3646$$
$$p_- = \frac{182+1}{286+2} = 0.6354$$

De même, en croisant chaque descripteur avec la variable cible, nous obtenons les fréquences absolues, nous en déduisons les estimations des probabilités conditionnelles.

Nombre de disease	exang		
disease	yes	no	Total
positive	68	36	104
negative	19	163	182
Total	87	199	286

P(exang/disease)	
0.6509	0.3491
0.1087	0.8913

Nombre de disease	chest_pain				
disease	asympt	atyp_angina	non_anginal	typ_angina	Total
positive	81	8	11	4	104
negative	39	95	41	7	182
Total	120	103	52	11	286

P(chest_pain/disease)			
0.7593	0.0833	0.1111	0.0463
0.2151	0.5161	0.2258	0.0430

Il s'agit de « profils lignes ». Les sommes des fréquences en ligne doivent être égales à 1 bien évidemment.

2.2.2 Classement d'un nouvel individu

Lors du classement, pour un individu ω présentant les caractéristiques (EXANG : yes, CHEST_PAIN : asympt), nous réalisons les calculs suivants :

$$\begin{aligned} d[+, \aleph(\omega)] &= \ln 0.3646 + (\ln 0.6509 + \ln 0.7593) \\ &= -1.7137 \\ d[-, \aleph(\omega)] &= \ln 0.6354 + (\ln 0.1087 + \ln 0.2151) \\ &= -4.2095 \end{aligned}$$

Comme $d(+, \aleph) > d(-, \aleph)$, nous l'affectons au groupe des DISEASE = positive.

Pour un informaticien, c'est le bonheur. Les données peuvent résider sur disque sans que cela ne présente de difficultés particulières. Une lecture séquentielle est tout à fait adaptée. Un seul passage suffit pour construire les tableaux de contingence. Reste qu'il faudrait apparemment (?) conserver ces derniers en mémoire pour le classement de nouveaux individus.

2.3 Pourquoi le classifieur bayesien naïf est-il performant ?

2.3.1 Le modèle bayesien naïf est un classifieur linéaire

Beaucoup s'étonnent qu'une méthode reposant sur une hypothèse (apparemment) aussi farfelue présente d'excellentes performances en prédiction, comparables aux autres techniques dont l'efficacité est reconnue. Mieux même, elle est très utilisée dans la communauté des chercheurs. Parce que d'une part elle est très facile à programmer, sa mise en œuvre est aisée ; d'autre part, parce que l'estimation de ses paramètres, la construction du modèle, est très rapide sur de très grandes bases de données, que ce soit en nombre de variables ou en nombre d'observations.

Il en va autrement auprès des praticiens. Il y a beaucoup de méfiance vis-à-vis du classifieur bayesien naïf parce qu'il est mal compris. De plus, la règle d'affectation n'est pas (semble-t-il) explicite, le modèle n'est pas interprétable. On ne voit pas très bien de quelle manière chaque variable pèse sur la décision. L'interprétation des résultats n'est pas aisée.

Cette opinion, largement répandue, est pourtant erronée. En se penchant attentivement sur les formules, on se rend compte que le modèle bayesien naïf est un classifieur linéaire. Il propose un biais de représentation similaire à celui de l'analyse discriminante, de la régression logistique ou des SVM (support vector machine) linéaires !!! Ce qui explique en grande partie sa bonne tenue en prédiction, comparable souvent à ces techniques[6,7].

2.3.2 Modèle à une variable explicative

Pour clarifier les idées, nous considérons dans un premier temps le cas d'un modèle à une seule variable prédictive X comportant L modalités $\{1,2,\ldots,L\}$. Nous créons autant d'indicatrices I_l qu'il y a de modalités, avec

$$I_l(\omega) = \begin{cases} 1 \; si \; X(\omega) = l \\ 0 \; \text{sinon} \end{cases}$$

La fonction de classement $d(y_k, \aleph)$ pour la modalité y_k s'écrit alors

$$\begin{aligned} d(y_k, \aleph) &= a_0 + \sum_{l=1}^{L} a_l \times I_l \\ &= \ln p_k + \sum_{l=1}^{L} \ln p_{l/k} \times I_l \end{aligned}$$

Or, par construction

$$I_1 + \cdots + I_L = 1 \Rightarrow I_L = 1 - (I_1 + \cdots + I_{L-1})$$

Nous pouvons ré écrire la fonction de classement de la manière suivante

[6] Bien sûr, d'autres explications sont avancées dans la littérature (rien n'est monolithique en recherche). La plus largement répandue est que le classifieur bayesien est un mauvais estimateur de P(Y / X), mais un bon estimateur du mode de P(Y / X). Et c'est ce qui importe en classement puisque nous affectons à la modalité de Y qui présente la probabilité a posteriori maximale. Voir http://www.cs.washington.edu/ai/naive.html

[7] La performance ne repose pas uniquement sur le biais de représentation, le biais d'apprentissage joue aussi un rôle important. Deux méthodes reposant sur le même système de représentation peuvent présenter des performances très différentes dans certaines configurations. Par exemple, lorsque le ratio entre le nombre de descripteurs et le nombre d'observations est dangereusement élevé, les SVM, avec le principe de maximisation de la marge, s'avèrent nettement plus avantageux.

$$
\begin{aligned}
d(y_k, \aleph) &= \ln p_k + \sum_{l=1}^{L} \ln p_{l/k} \times I_l \\
&= \left(\ln p_k + \ln p_{L/k}\right) + \sum_{l=1}^{L-1} \ln \frac{p_{l/k}}{p_{L/k}} \times I_l \\
&= a_{0,k} + a_{1,k} \times I_1 + a_{2,k} \times I_2 + \cdots
\end{aligned}
$$

Le résultat ressemble quand même beaucoup à celui produit par l'analyse discriminante linéaire ou la régression logistique lorsque nous transformons à l'aide d'un codage disjonctif complet la variable explicative catégorielle. Le mode d'estimation des coefficients de la combinaison linéaire des indicatrices est en revanche différent.

2.3.3 Expliciter la fonction de classement pour un modèle à J descripteurs

Le modèle étant additif, le passage à J variables explicatives ne pose aucun problème. Le j^{ème} descripteur Xj prend Lj modalités, nous lui associons Lj indicatrices. La fonction de classement s'écrit maintenant

$$
d(y_k, \aleph) = \left(\ln p_k + \sum_{j=1}^{J} \ln p^j_{L_j/k} \right) + \sum_{j=1}^{J} \sum_{l=1}^{L_j - 1} \ln \frac{p^j_{l/k}}{p^j_{L_j/k}} \times I^j_l
$$

où $p^j_{l/k} = \hat{P}\left(X_j = l / Y = y_k\right)$, I^j_l est l'indicatrice n° l pour la variable Xj.

2.3.4 Exemple numérique

Reprenons notre modèle à deux variables explicatives ci-dessus. Nous notons A l'indicatrice (exsang = yes), et respectivement B1, B2 et B3, les indicatrices (chest_pain = asympt), (chest_pain = atyp_angina) et (chest_pain = non_angina).

$$
\begin{aligned}
d[+, \aleph] &= \ln 0.3646 + \ln 0.3491 + \ln 0.0463 + \ln \frac{0.6509}{0.3491} \times A + \ln \frac{0.7593}{0.0463} \times B1 + \ln \frac{0.0833}{0.0463} \times B2 + \ln \frac{0.1111}{0.0463} \times B3 \\
&= -5.1342 + 0.6232 \times A + 2.7973 \times B1 + 0.5878 \times B2 + 0.8755 \times B3
\end{aligned}
$$

$$
\begin{aligned}
d[-, \aleph] &= \ln 0.6534 + \ln 0.8913 + \ln 0.0430 + \ln \frac{0.1087}{0.8913} \times A + \ln \frac{0.2151}{0.0430} \times B1 + \ln \frac{0.5161}{0.0430} \times B2 + \ln \frac{0.2258}{0.0430} \times B3 \\
&= -3.7148 - 2.1041 \times A + 1.6094 \times B1 + 2.4849 \times B2 + 1.6582 \times B3
\end{aligned}
$$

Pour le même individu à classer (EXANG : yes, CHEST_PAIN : asympt), les valeurs des indicatrices deviennent (A : 1 ; B1 : 1 ; B2 : 0 ; B3 : 0). Nous obtenons

$$
\begin{aligned}
d[+, \aleph] &= -5.1342 + 0.6232 \times 1 + 2.7973 \times 1 + 0.5878 \times 0 + 0.8755 \times 0 = -1.7337 \\
d[-, \aleph] &= -3.7148 - 2.1041 \times 1 + 1.6094 \times 1 + 2.4849 \times 0 + 1.6582 \times 0 = -4.2095
\end{aligned}
$$

Les chiffres obtenus sont en adéquation avec ceux obtenus précédemment (section 2.2.2). Le contraire eût été inquiétant. La différence est que **nous disposons maintenant d'un modèle explicite facile à déployer**. Pourtant, comme nous le verrons par la suite, aucun logiciel, mis à part Tanagra (**version 1.4.36**) ne présente les résultats sous cette forme.

2.3.5 Cas particulier du modèle binaire

Lorsque la variable à prédire est binaire Y ∈ {+, - }, nous pouvons effectuer une différence termes à termes entre les deux fonctions de classement. Nous obtenons une fonction score $d(\aleph)$ unique comme en régression logistique. Pour notre exemple, nous faisons

$$\begin{aligned} d(\aleph) &= d(+,\aleph) - d(-,\aleph) \\ &= -1.4194 + 2.7273 \times A + 1.1878 \times B1 - 1.8971 \times B2 - 0.7828 \times B3 \end{aligned}$$

La règle d'affectation devient

$$\text{Si } d[\aleph(\omega)] > 0 \text{ Alors } \hat{y}(\omega) = + \text{ Sinon } \hat{y}(\omega) = -$$

Toujours pour l'individu (EXANG : yes, CHEST_PAIN : asympt), nous avons

$$\begin{aligned} d(\aleph) &= -1.4194 + 2.7273 \times 1 + 1.1878 \times 1 - 1.8971 \times 0 - 0.7828 \times 0 \\ &= 2.4958 \end{aligned}$$

Tout comme en régression logistique binaire, les coefficients se lisent comme des logarithmes d'odds ratio. A la différence qu'ils se lisent en ligne dans le tableau de contingence !

Prenons, le coefficient de l'indicatrice A (exsang = yes), nous avons $e^{2.7273} = 15.2919$. Nous dirons : par rapport aux négatifs, les positifs ont 15.2919 fois plus de chances d'avoir le caractère (exsang = yes) que d'avoir (exsang = no). On pourrait d'ailleurs imaginer un mécanisme basé sur un test de l'odds-ratio (l'odds-ratio est il significativement différent de 1 ?) pour évaluer le rôle d'une indicatrice dans le modèle. Il faudrait surtout pouvoir l'étendre au cadre multi-classes c.-à-d. la variable cible peut prendre plus de 2 modalités.

3 Données

Nous avions présenté partiellement nos données[8] précédemment. Elles proviennent du site UCI (Heart Disease Dataset - http://archive.ics.uci.edu/ml/datasets/Heart+Disease).

Après nettoyage, nous disposons de 286 observations. Nous souhaitons prédire la variable DISEASE à l'aide de 2 descripteurs catégoriels (EXANG : 2 modalités, CHEST_PAIN : 4 modalités). La base originelle est beaucoup plus fournie. Nous nous sommes restreints à ces variables afin de pouvoir décrire de manière détaillée les calculs intermédiaires.

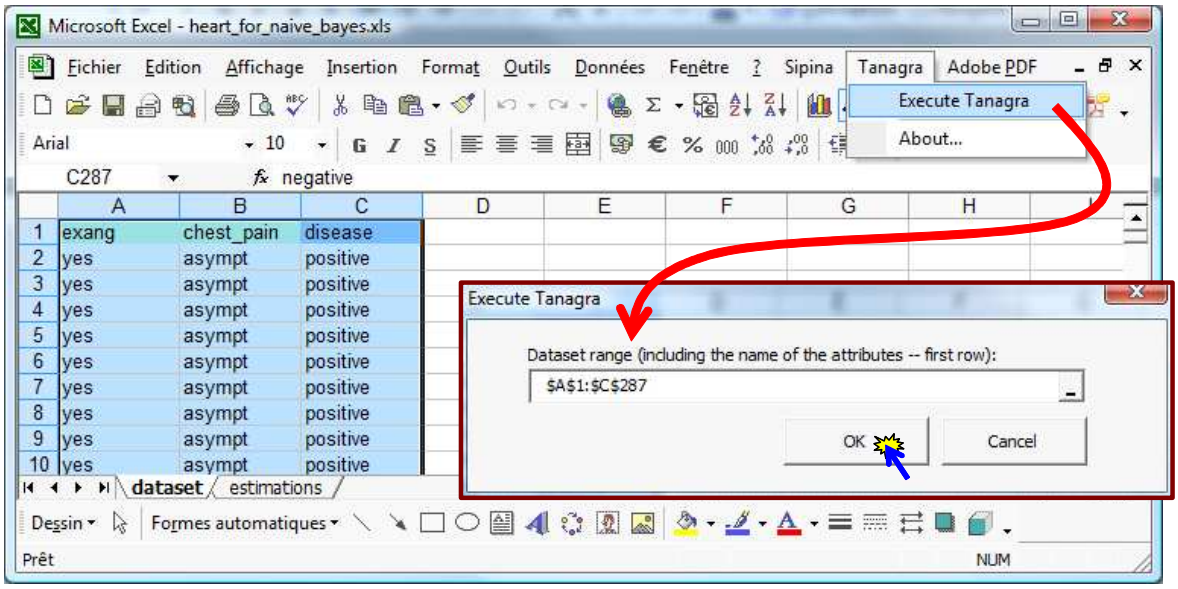
4 Traitements avec Tanagra

4.1 Le modèle bayesien naïf sous Tanagra

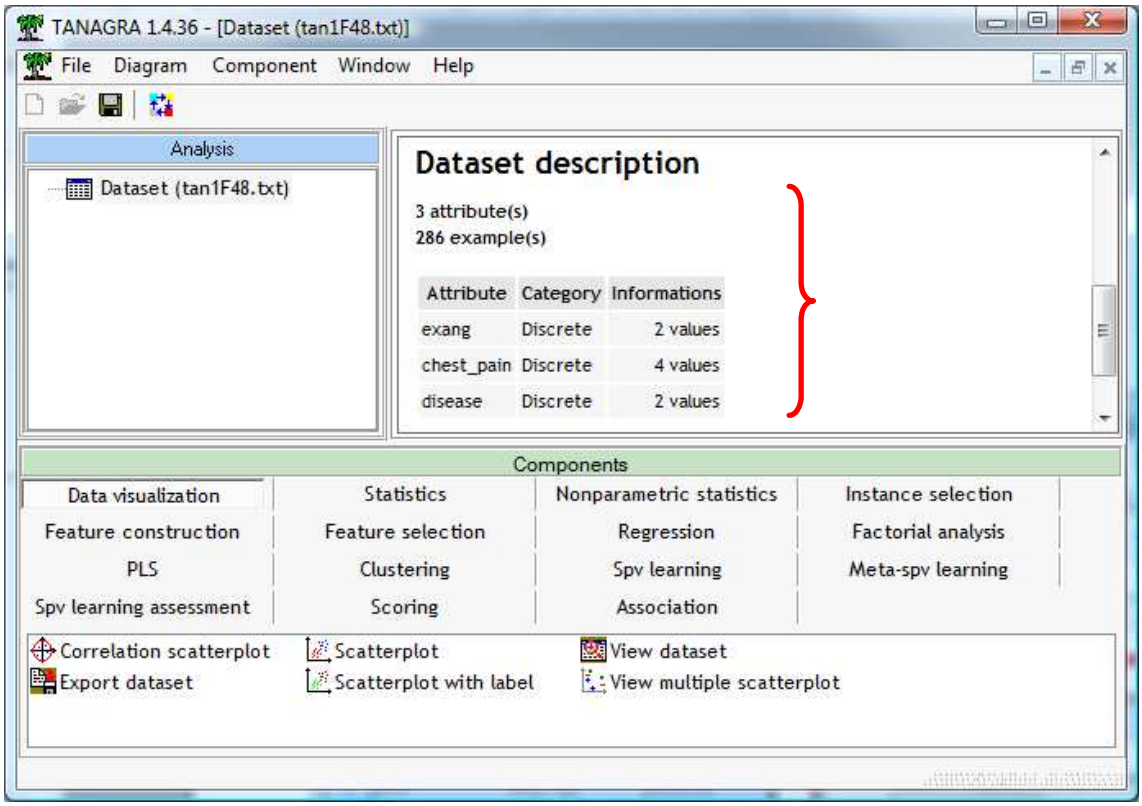
Importation des données. Pour charger les données dans Tanagra, le plus simple est d'ouvrir tout d'abord le fichier HAERT_FOR_NAIVE_BAYES.XLS dans le tableur Excel (ou Open Office Calc), puis d'envoyer les données dans Tanagra via l'add-on (macro complémentaire) préalablement installée[9].

[8] http://eric.univ-lyon2.fr/~ricco/tanagra/fichiers/heart_for_naive_bayes.zip

Après avoir sélectionné la plage de données, nous actionnons le menu TANAGRA / EXECUTE TANAGRA. Nous vérifions les coordonnées (\$A\$1 :\$C\$287). Enfin, nous validons en cliquant sur le bouton OK.

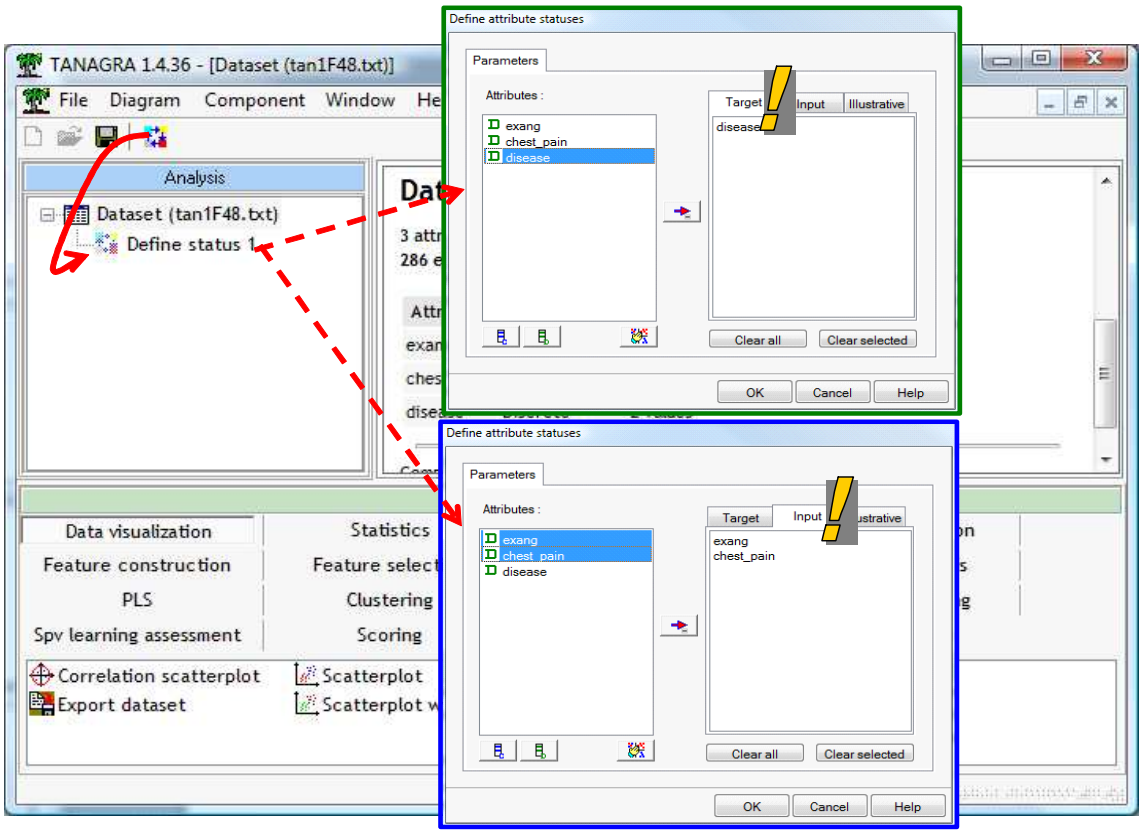


Tanagra est automatiquement démarré : 286 observations et 3 variables, toutes discrètes, sont disponibles pour les traitements.

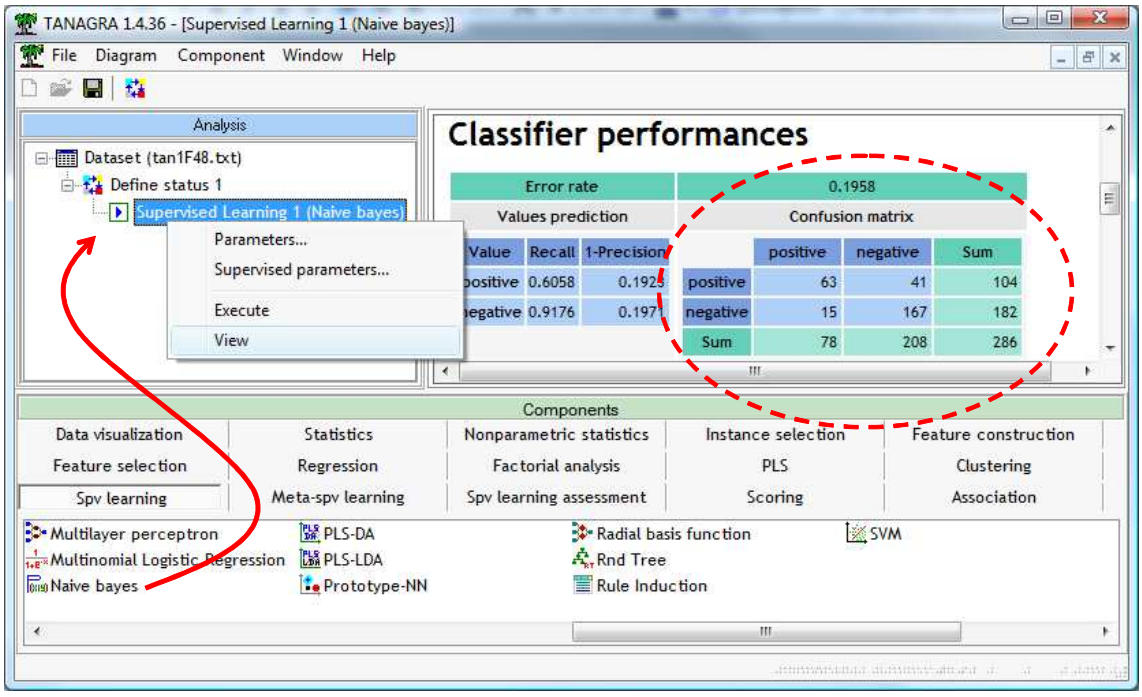


[9] Voir http://tutoriels-data-mining.blogspot.com/2008/03/importation-fichier-xls-excel-macro.html pour l'installation et l'utilisation de la macro complémentaire. Voir http://tutoriels-data-mining.blogspot.com/2008/03/connexion-open-office-calc.html pour Open Office Calc.

Construction du classifieur. Avant de passer à la phase d'apprentissage, il faut désigner la variable à prédire et les variables prédictives. Nous utilisons le composant DEFINE STATUS pour cela. Nous l'introduisons dans le diagramme de traitements via le raccourci dans la barre d'outils. Nous plaçons DISEASE en TARGET, EXSANG et CHEST_PAIN en INPUT.

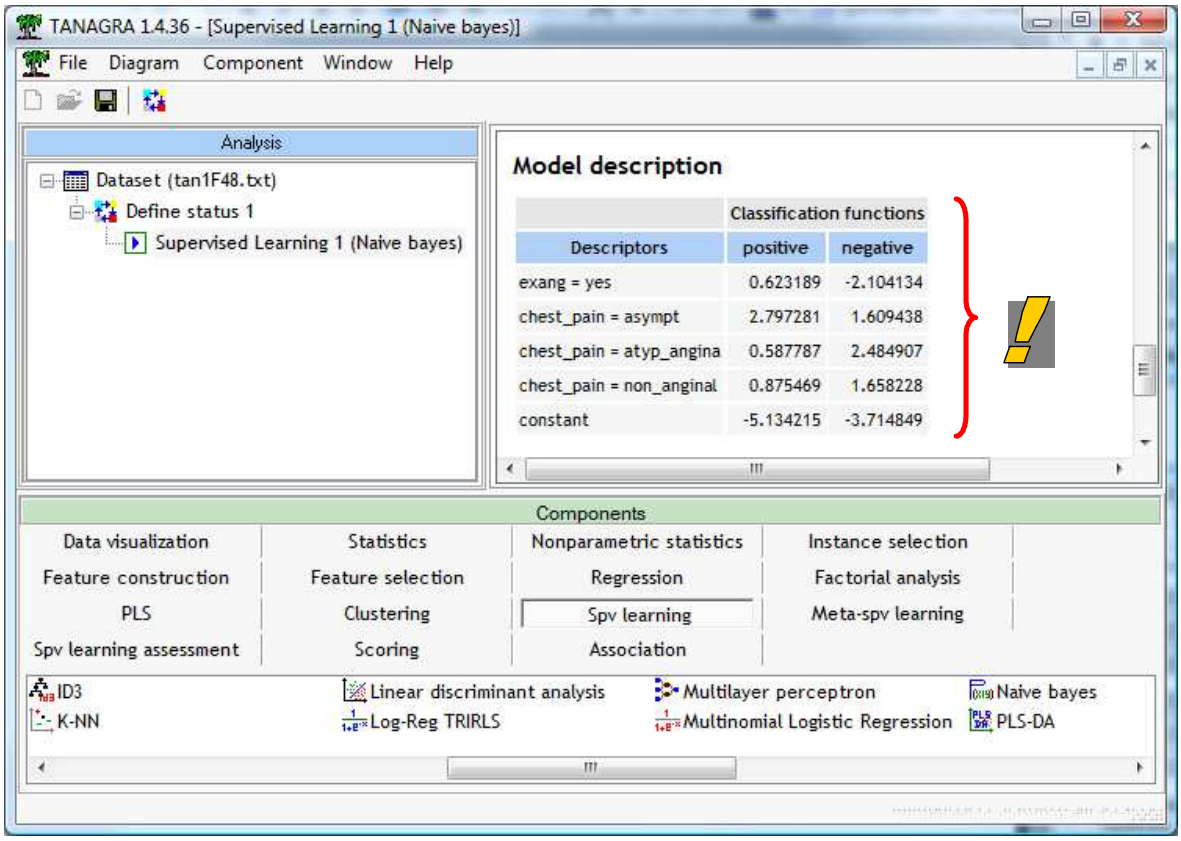


Nous pouvons insérer alors le composant NAIVE BAYES CLASSIFIER (onglet SPV LEARNING) dans le diagramme. Nous cliquons sur VIEW pour obtenir les résultats.



Dans la partie haute de la fenêtre, nous avons la matrice de confusion en resubstitution c.-à-d. calculée sur les données qui ont servi à l'élaboration du modèle. Elle est affichée à titre indicatif.

La principale nouveauté de la version 1.4.36 de Tanagra est dans la partie basse de la fenêtre. « **Model description** » introduit la description explicite des fonctions de classement, une pour chaque modalité de la variable à prédire. Nous pouvons l'appliquer directement lors du déploiement du modèle c.-à-d. pour classer de nouveaux individus issus de la population.



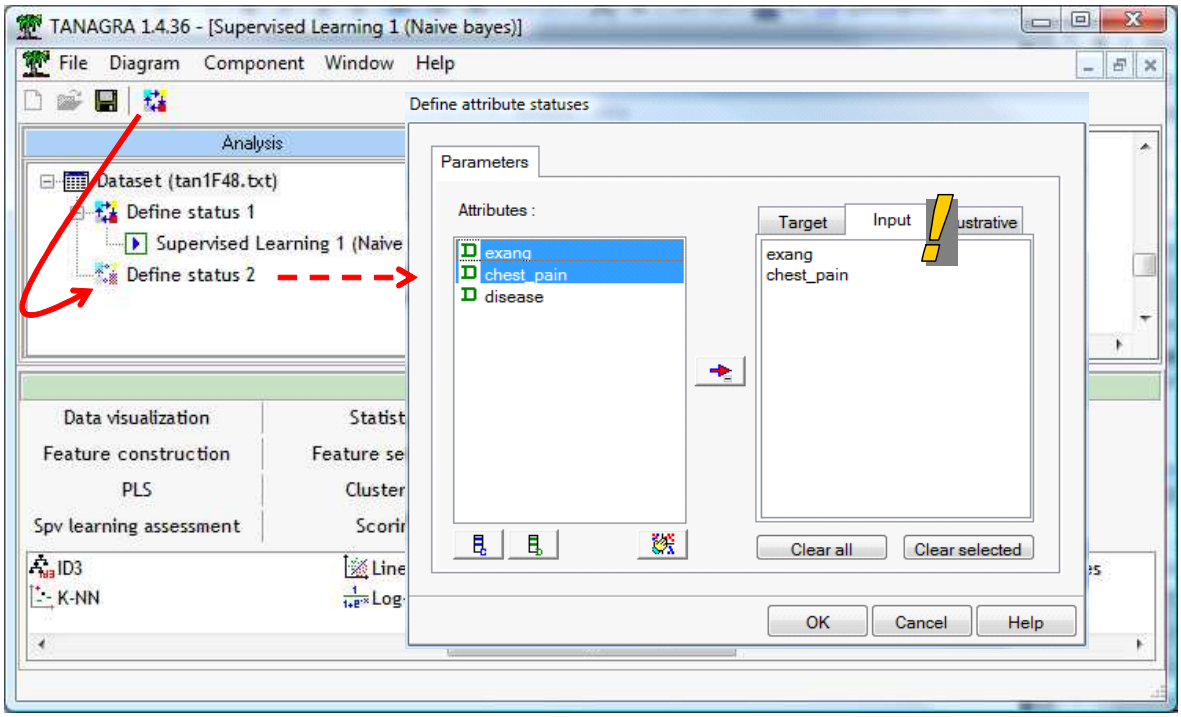
Par ailleurs, nous retrouvons bien les fonctions de classement calculées manuellement plus haut (section 2.3.4). Le contraire eût été inquiétant. En effectuant une différence termes à termes entre les deux fonctions de classement, nous obtenons la « **fonction score** ».

Descriptors	Classification functions		Score function
	positive	negative	
exang = yes	0.623189	-2.104134	2.7273
chest_pain = asympt	2.797281	1.609438	1.1878
chest_pain = atyp_angina	0.587787	2.484907	-1.8971
chest_pain = non_anginal	0.875469	1.658228	-0.7828
constant	-5.134215	-3.714849	-1.4194

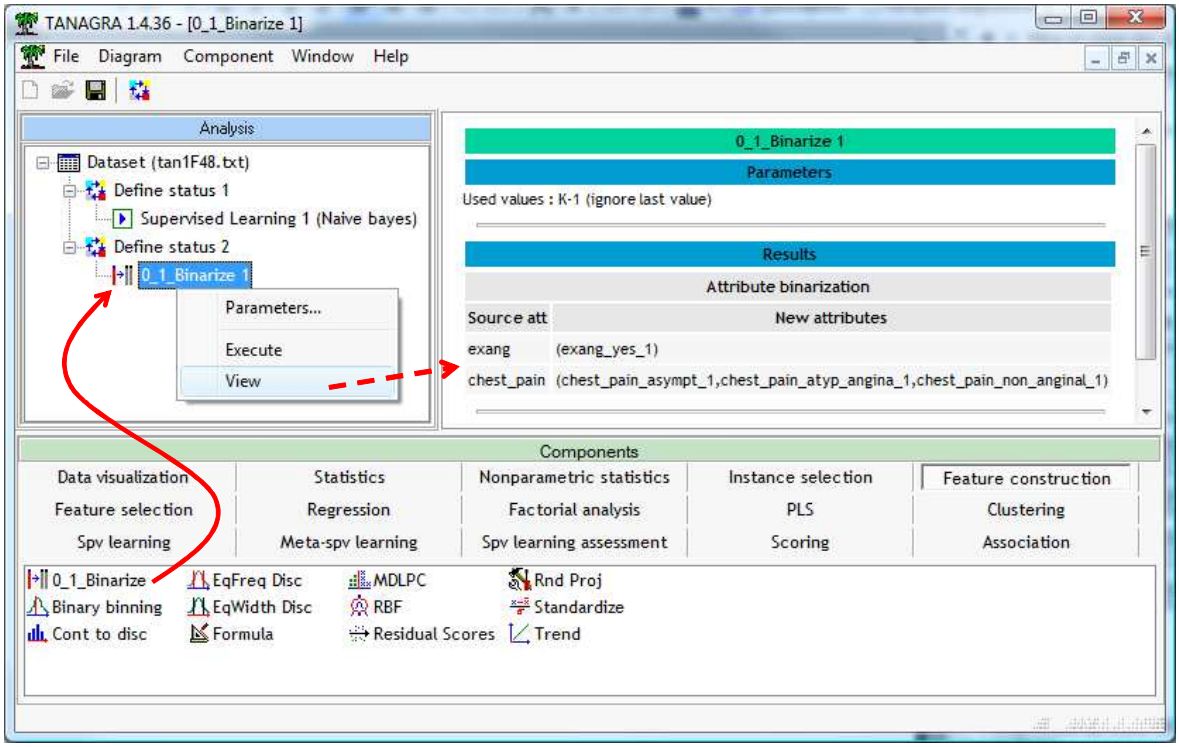
4.2 Les autres séparateurs linéaires

Les méthodes étudiées dans cette section savent, d'une manière ou d'une autre, produire une fonction score sous forme d'une combinaison linéaire de variables numériques. Nous devons préalablement recoder nos descripteurs.

Nous insérons le composant DEFINE STATUS à la racine du diagramme. Nous plaçons EXANG et CHEST_PAIN en INPUT.

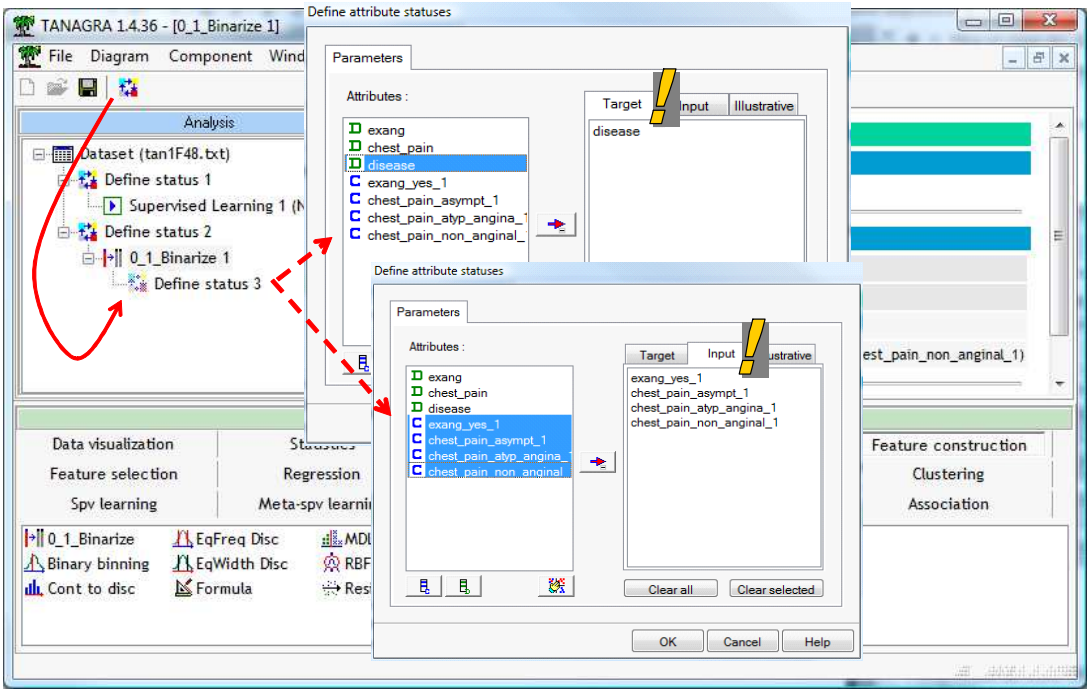


Nous ajoutons le composant 0_1_BINARIZE (onglet FEATURE CONSTRUCTION), nous actionnons le menu contextuel VIEW.



EXANG a été codée en une seule variable 1 (yes) / 0 (no), CHEST_PAIN en 3 indicatrices.

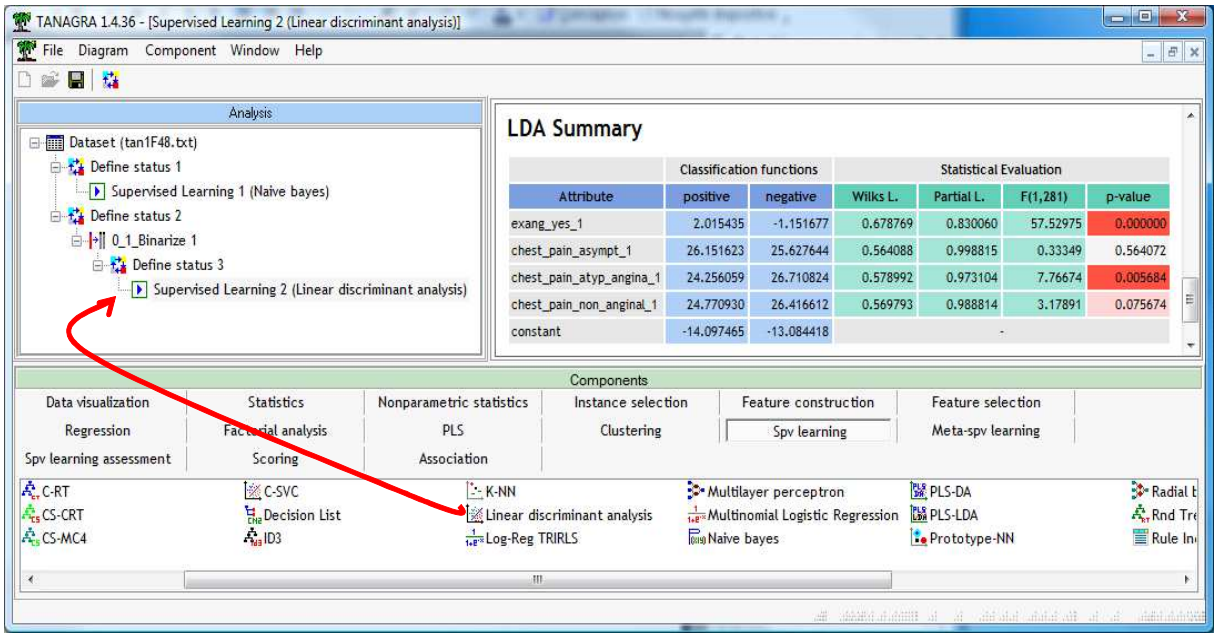
Nous introduisons de nouveau un DEFINE STATUS pour définir le rôle des variables. Nous plaçons DISEASE en TARGE T, les indicatrices en INPUT.



Nous sommes prêts à lancer la construction des fonctions scores à l'aide des différentes méthodes supervisées.

4.2.1 L'analyse discriminante (LDA)

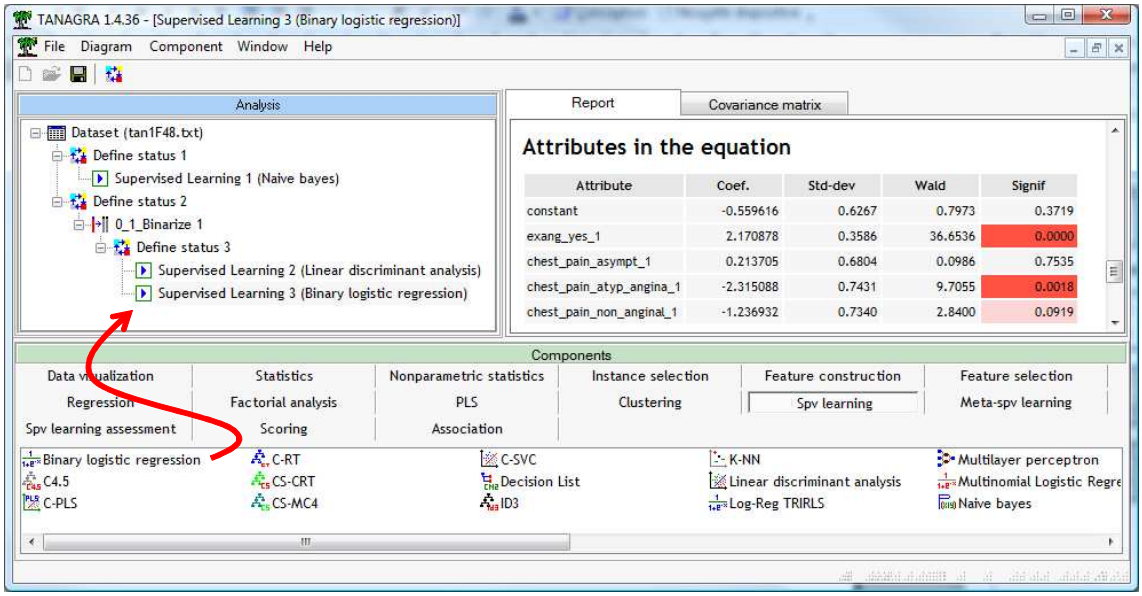
L'analyse discriminante (LINEAR DISCRIMINANT ANALYSIS – onglet SPV LEARNING) produit des fonctions de classement, une pour chaque modalité de la variable à prédire. Plusieurs outils permettent de juger de la qualité globale du modèle et de la pertinence individuelle des variables.



Dans le cas d'une variable cible binaire, nous pouvons déduire la fonction score.

4.2.2 La régression logistique

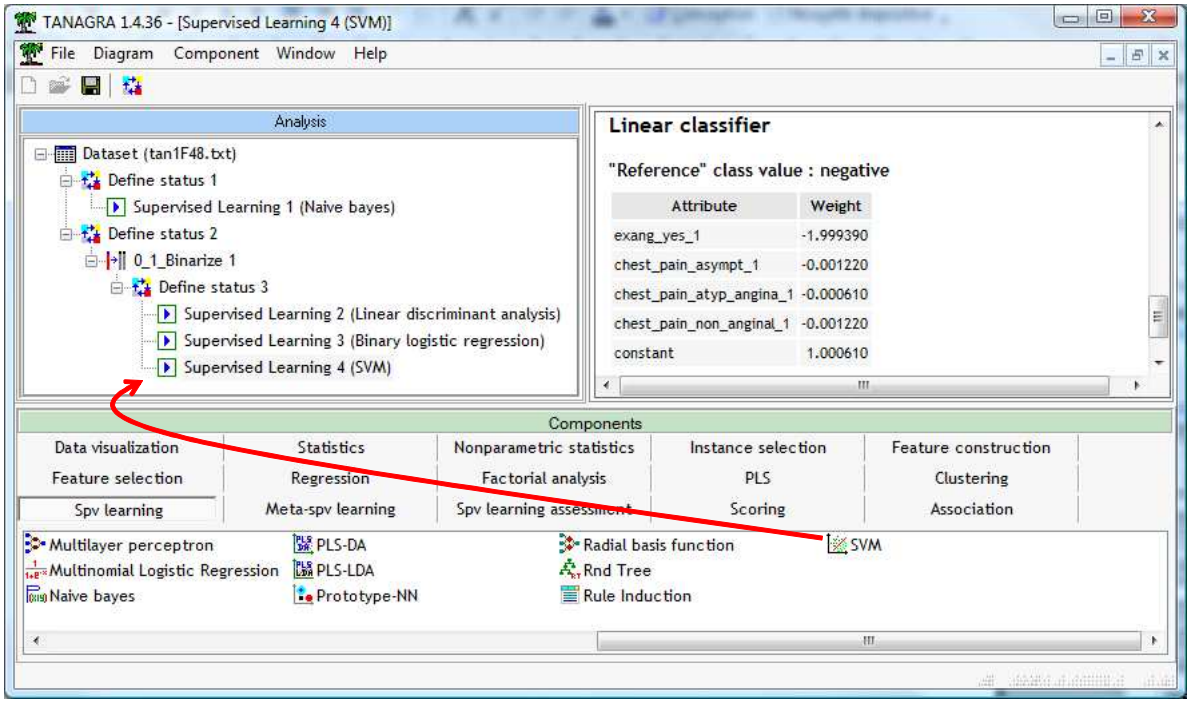
La régression logistique (BINARY LOGISTIC REGRESSION – onglet SPV LEARNING) ne traite que les variables cibles binaires. En conséquence, elle fournit directement la fonction score. Attention, il faut bien vérifier la valeur de référence de la variable cible.



La valeur référence est la modalité « positive ». Ca tombe bien, la fonction score est directement comparable à celle produite par les autres méthodes.

4.2.3 SVM (Support Vector Machine) linéaire

Le composant SVM (onglet SPV LEARNING) produit directement la fonction score. Il n'est opérant que pour les problèmes à deux classes. Ici également, il faudra vérifier la modalité de référence avant de passer à la lecture des résultats.



La modalité de référence est « négative » dans SVM. Il faut donc multiplier les coefficients de la fonction score par -1 avant de procéder à la comparaison.

4.3 Comparaison des fonctions score

Même biais de représentation, mais biais d'apprentissage distinct, voyons justement jusqu'à quel point les coefficients fournis par les différentes méthodes diffèrent sur notre fichier.

Descriptors	Naive Bayes	LDA	Logistic Reg.	Linear SVM
exang = yes	2.7273	3.1671	2.1709	1.9994
chest_pain = asympt	1.1878	0.5240	0.2137	0.0012
chest_pain = atyp_angina	-1.8971	-2.4548	-2.3151	0.0006
chest_pain = non_anginal	-0.7828	-1.6457	-1.2369	0.0012
constant	-1.4194	-1.0130	-0.5596	-1.0006

3 méthodes présentent des coefficients avec des signes cohérents (Naive Bayes, LDA et Régression logistique), même si par ailleurs les valeurs sont sensiblement différentes.

Seule SVM se démarque réellement. En assignant des coefficients très proches de 0 aux indicatrices de la variable CHEST_PAIN, il semble vouloir réduire l'impact de cette dernière lors du déploiement du modèle[10]. Les signes des autres coefficients (EXANG = YES et la constante) sont cohérents avec les autres modèles.

Concernant les performances en prédiction, sur un fichier aussi simple, les méthodes proposent des taux d'erreur quasi identiques, qu'elles soient évaluées en resubstitution (≈ 19.5%) ou par ré échantillonnage (boostrap ≈ 20.5% -- non détaillée dans ce didacticiel, voir http://tutoriels-data-mining.blogspot.com/2008/03/validation-croise-bootstrap-leave-one.html pour un exemple d'utilisation des techniques de ré échantillonnage pour l'évaluation des classifieurs).

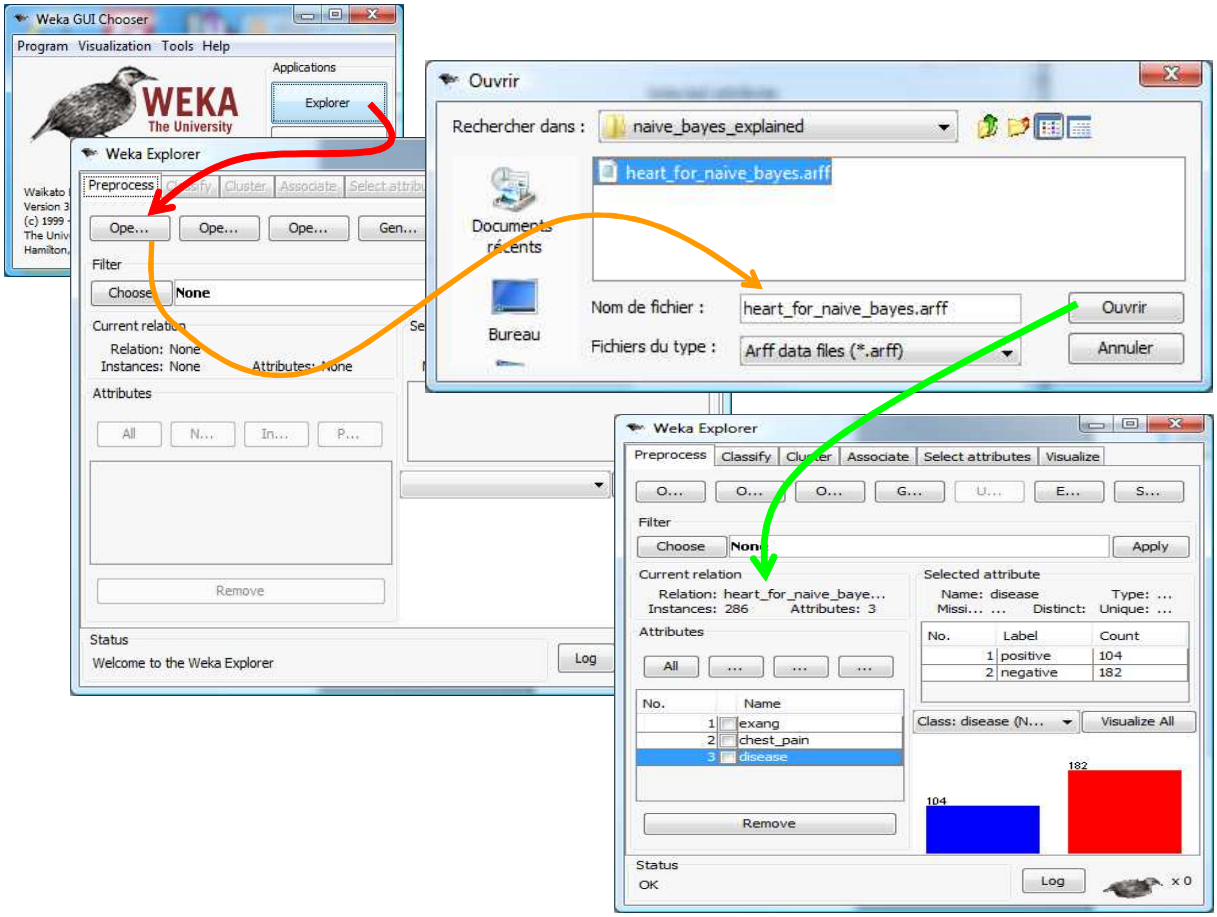
Sauf configurations très particulières, ces méthodes présentent souvent des performances similaires.

5 Traitements avec les autres logiciels

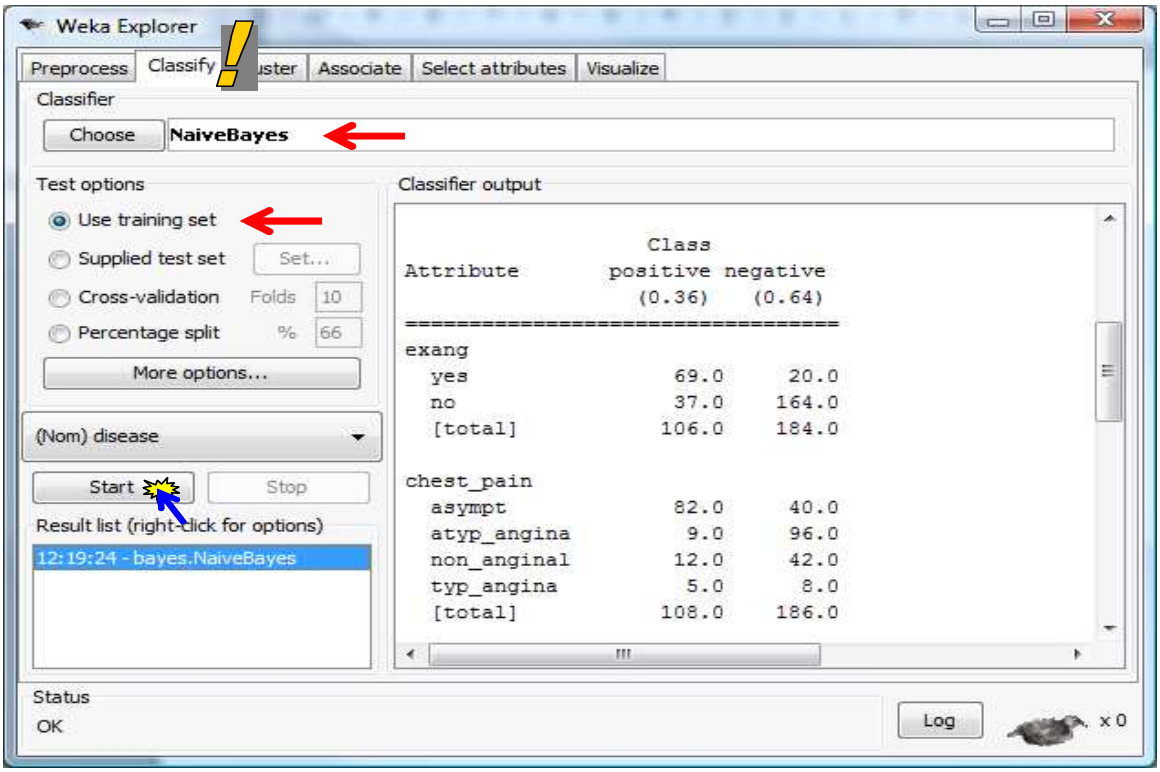
5.1 Weka 3.6.0

Nous utilisons Weka (http://www.cs.waikato.ac.nz/ml/weka/) en mode EXPLORER. Après avoir démarré le logiciel, nous chargeons le fichier en cliquant sur le bouton OPEN FILE. Nous sélectionnons le fichier HEART_FOR_NAIVE_BAYES.ARFF au format Weka (fichier texte avec des balises spécifiques).

[10] Il faudrait un critère en harmonie avec le principe de la maximisation de la marge pour évaluer la significativité des coefficients. Des avis éclairés sont le bienvenu sur ce point. Il est hors de question en tous les cas de se baser sur des indicateurs issus d'autres démarches (maximum de vraisemblance, etc.) pour évaluer l'impact d'une variable dans un SVM. A titre de curiosité, nous avons retiré la variable CHEST_PAIN de l'analyse. Le classifieur SVM fondé sur la seule explicative EXSANG s'avère aussi bon (sinon meilleur) avec un taux d'erreur en bootstrap proche de 19%.



Nous passons à l'onglet CLASSIFY et nous optons pour la méthode NAIVE BAYES. Nous évaluons les performances sur les données d'apprentissage afin de pouvoir comparer les résultats avec ceux de Tanagra. Nous actionnons le bouton START.



Nous obtenons les mêmes caractéristiques (matrice de confusion + taux d'erreur) qu'avec Tanagra. Dans la partie intermédiaire de la fenêtre, Weka propose une description des tableaux de contingence qui servent à estimer les probabilités d'affectation, à rapprocher avec les tableaux que nous avions décrit dans la section 2.2.1.

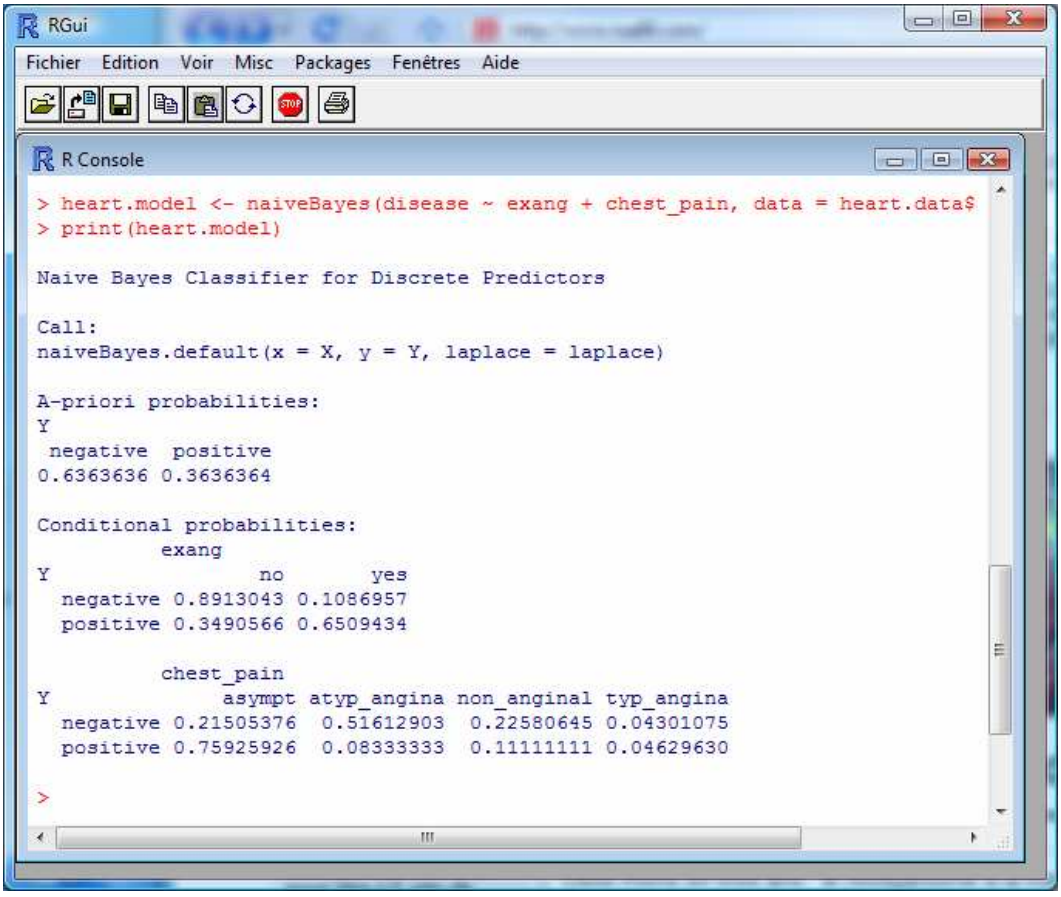
Il n'est fait mention nulle part en revanche d'une quelconque fonction de classement.

5.2 R 2.9.2

Pour R (http://www.r-project.org/), nous utilisons le package e1071 (http://cran.r-project.org/web/packages/e1071/index.html). Le code source de notre programme est le suivant.

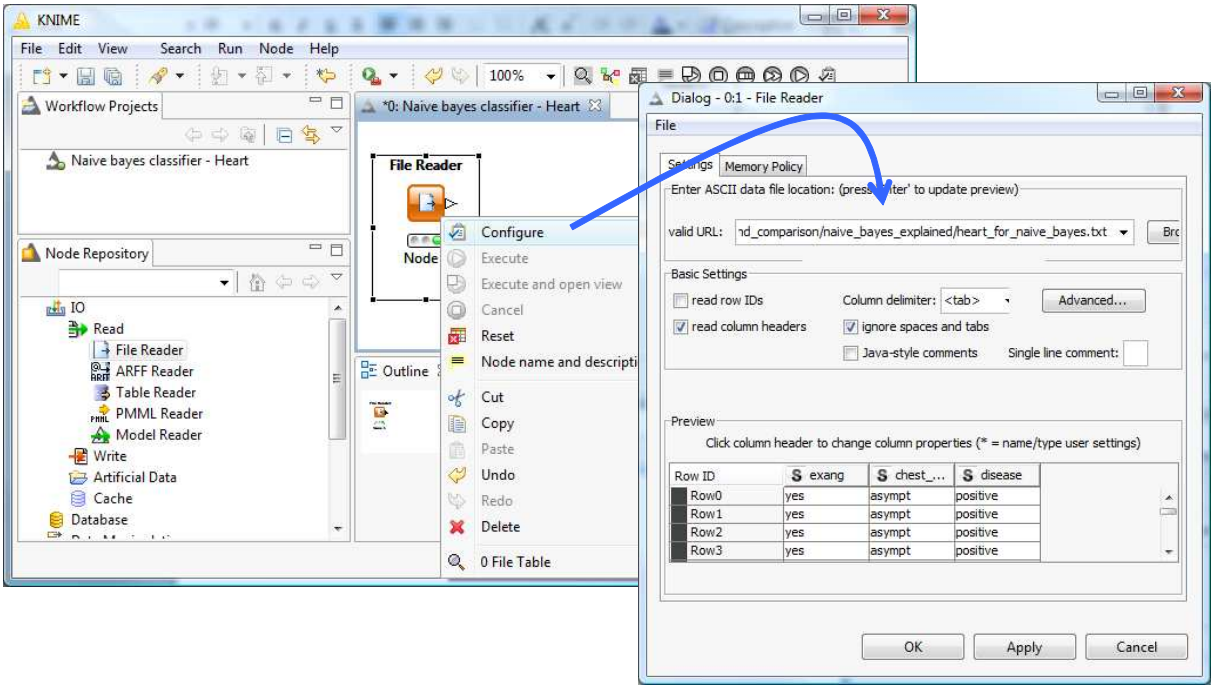
```
#clear the memory
rm (list=ls())
#load the dataset
heart.data <- read.table(file="heart_for_naive_bayes.txt",sep="\t",header=T)
#build the model
library(e1071)
heart.model <- naiveBayes(disease ~ exang + chest_pain, data = heart.data, laplace = 1.0)
print(heart.model)
```

Lors du paramétrage de la procédure **naiveBayes**, nous noterons le rôle de « laplace = 1.0 » qui permet de produire l'estimation laplacienne des probabilités. R affiche les tables des probabilités, identiques à celles que nous avions calculées manuellement sous Excel (section 2.2.1).

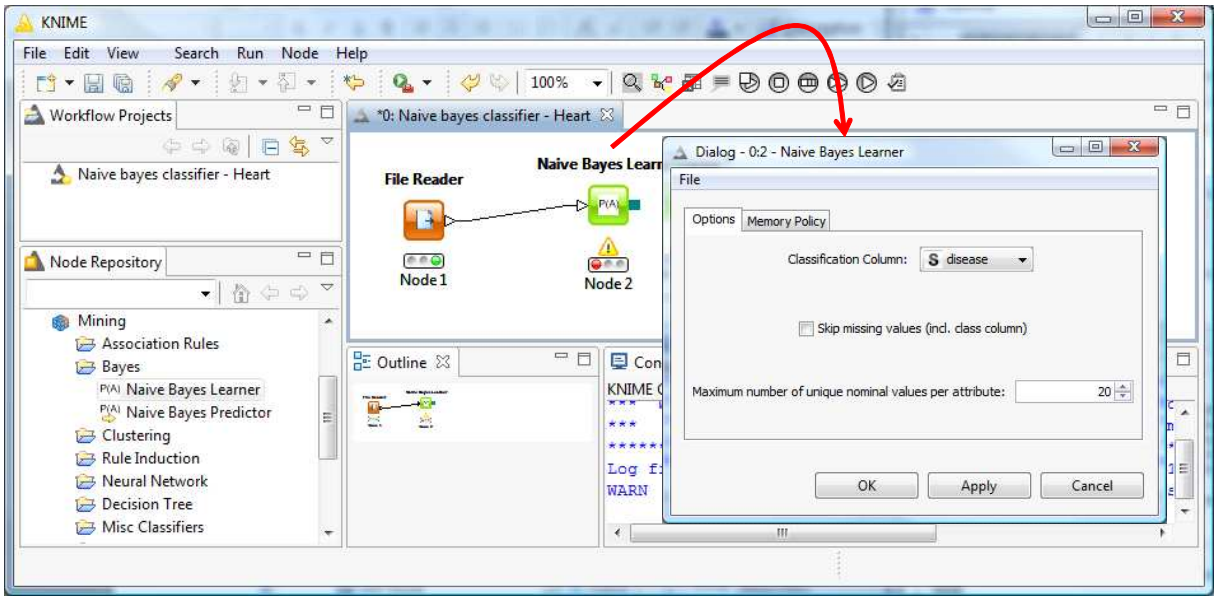


5.3 Knime 2.1.1

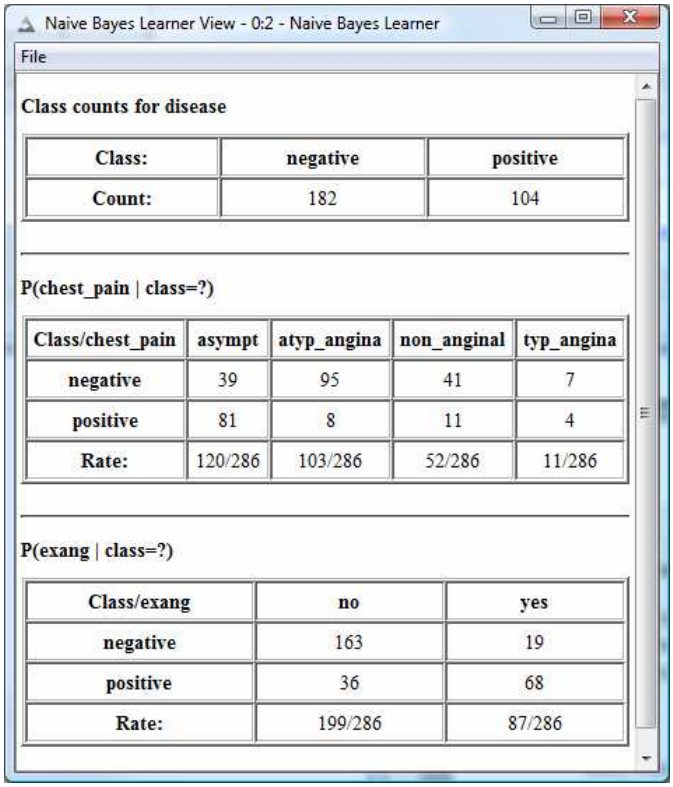
Après avoir démarré Knime (http://www.knime.org/), nous créons un nouveau « workflow » (FILE / NEW). Nous insérons le composant FILE READER (branche IO / READ). Nous le configurons (menu CONFIGURE) de manière à lire le fichier HEART_FOR_NAIVE_BAYES.TXT.



Nous insérons ensuite le composant NAIVE BAYES LEARNER (branche MINING / BAYES), nous le configurons pour spécifier la variable cible DISEASE.



Il ne reste plus qu'à cliquer sur le menu contextuel EXECUTE AND OPEN VIEW. Nous obtenons les tableaux de contingence qui servent à calculer les probabilités conditionnelles (section 2.2.1).



Class counts for disease

Class:	negative	positive
Count:	182	104

P(chest_pain | class=?)

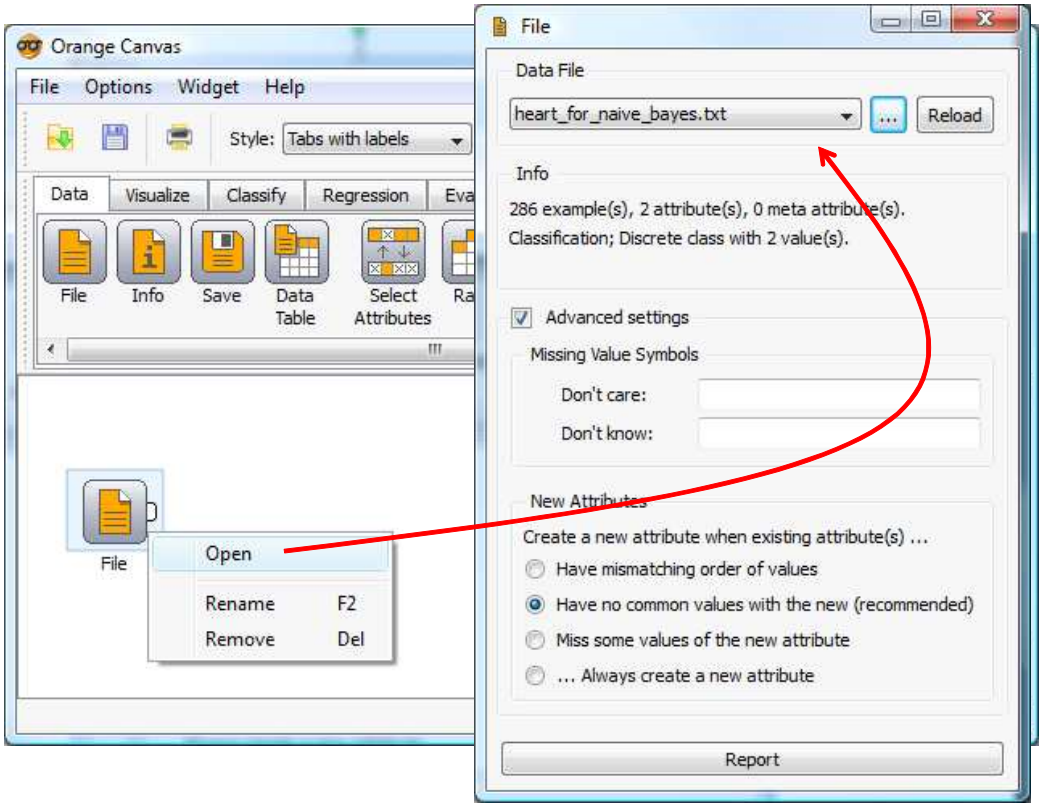
Class/chest_pain	asympt	atyp_angina	non_anginal	typ_angina
negative	39	95	41	7
positive	81	8	11	4
Rate:	120/286	103/286	52/286	11/286

P(exang | class=?)

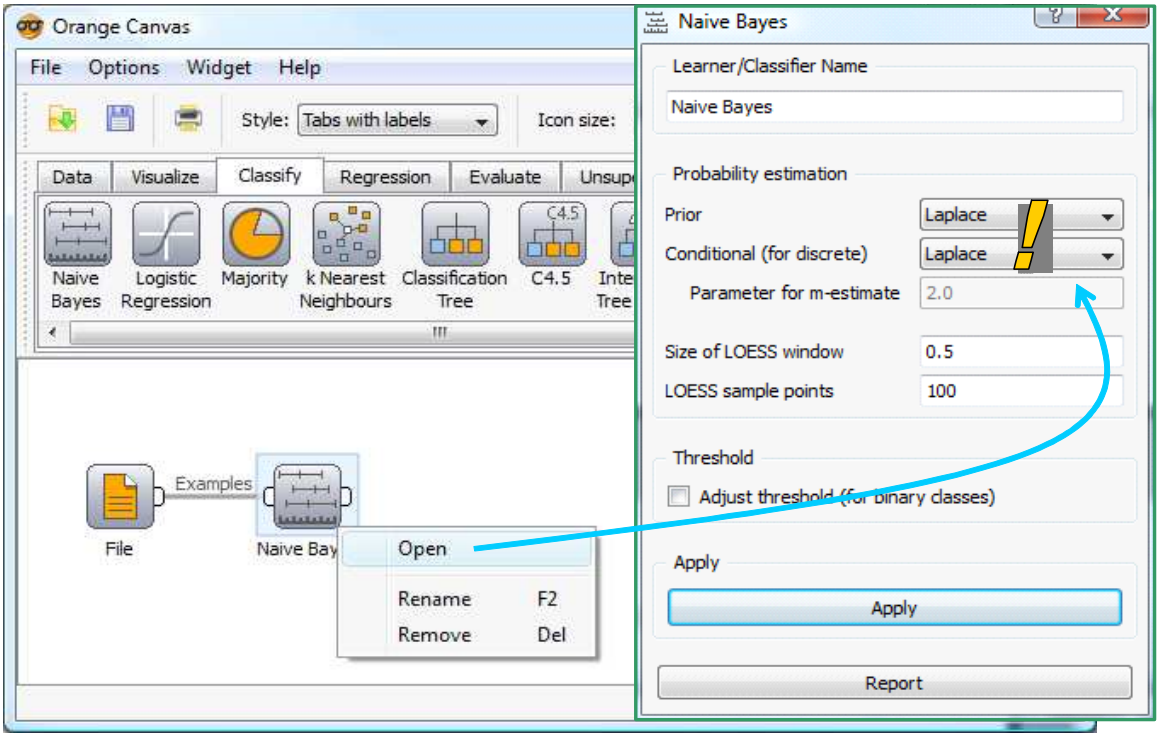
Class/exang	no	yes
negative	163	19
positive	36	68
Rate:	199/286	87/286

5.4 Orange 2.0b

Nous lançons Orange (http://www.ailab.si/orange/). Nous ajoutons le composant FILE (onglet DATA) dans l'espace de travail. Nous le paramétrons avec le menu OPEN. Nous sélectionnons le fichier HEART_FOR_NAIVE_BAYES.TXT au format texte avec séparateur tabulation.

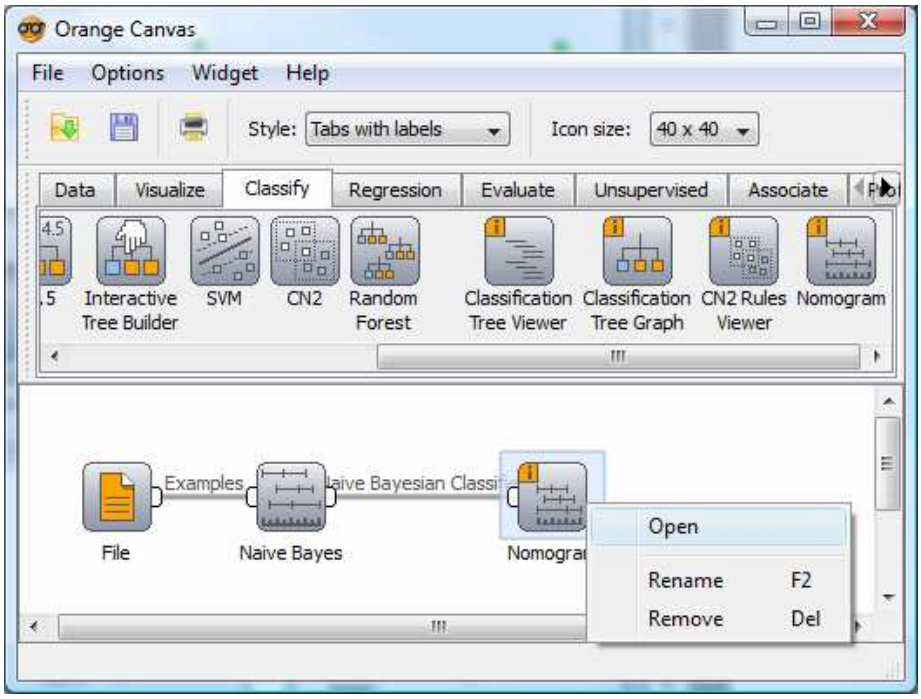


Nous insérons le composant NAIVE BAYES (onglet CLASSIFY). Nous lui connectons FILE puis nous le paramétrons en actionnant le menu OPEN.

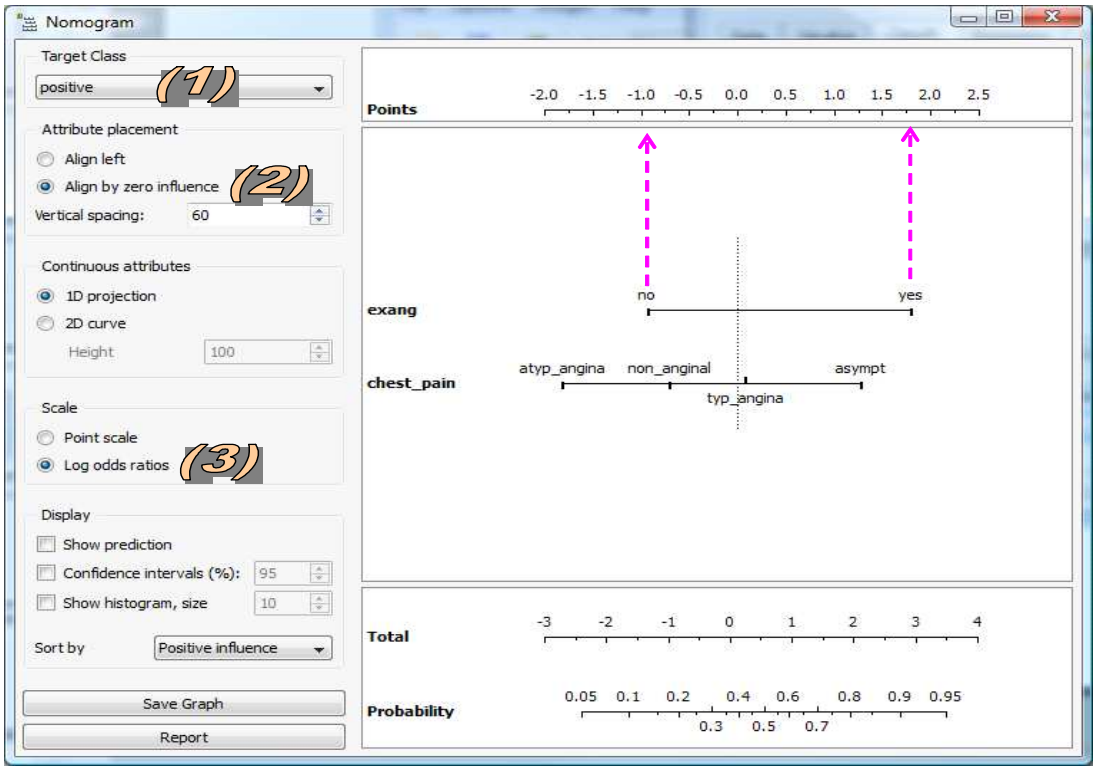


Nous demandons l'estimateur laplacien des probabilités, tant pour le calcul des probabilités a priori des classes, que pour le calcul des probabilités conditionnelles.

Pour visualiser le modèle, nous ajoutons le composant NOMOGRAM, connecté à NAIVE BAYES. Nous cliquons sur le menu contextuel OPEN.



Le nomogramme est un outil de visualisation qui n'est pas très connu. Pourtant il est remarquablement intuitif. Il vise à montrer rapidement l'impact de chaque variable explicative sur les classes[11]. Nous adoptons le paramétrage suivant : nous désignons la modalité POSITIVE comme « TARGET CLASS » (1) ; nous alignons les attributs selon une influence nulle (2) ; l'échelle adoptée est l'odds-ratio. Nous obtenons la représentation suivante.



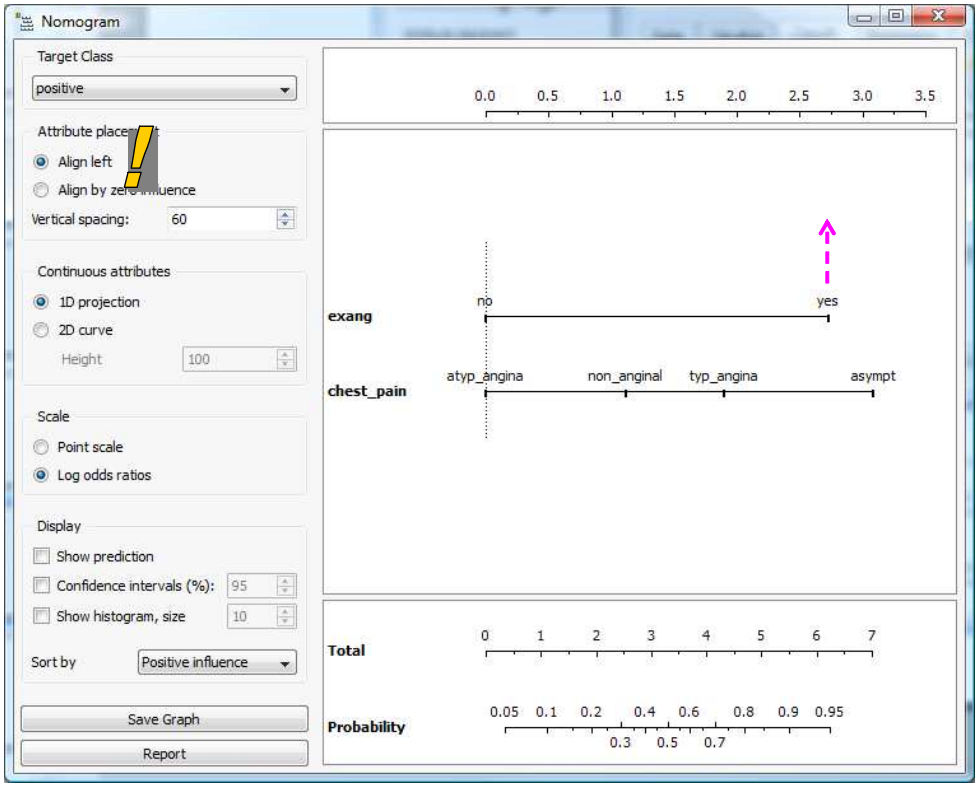
Détaillons les calculs pour la variable EXANG. Pour rappel, voici le tableau de contingence la croisant avec la variable cible DISEASE.

Nombre de disease	exang		
disease	yes	no	Total
positive	68	36	104
negative	19	163	182
Total	87	199	286

Le rapport entre les positifs et les négatifs dans l'ensemble de l'échantillon, on parle d'odds, est de $\frac{104}{182} = 0.57$. Chez les personnes avec « EXANG = yes », il est $\frac{68}{19} = 3.58$, Orange forme alors le logarithme du rapport entre ces odds, soit $\ln\frac{3.58}{0.57} = 1.83$. Ce qui explique le positionnement de la modalité à cette coordonnée dans le nomogramme. De même, pour « EXANG = no », nous obtenons $\ln\frac{0.22}{0.57} = -0.95$.

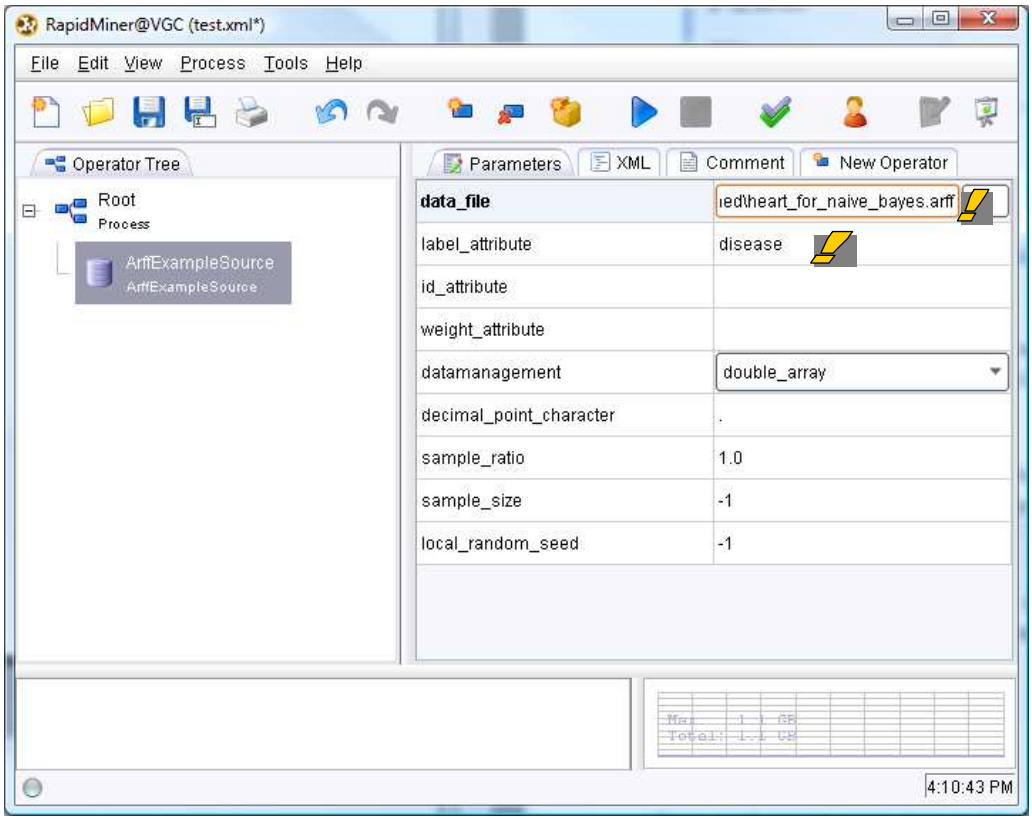
Lorsque nous passons sur un alignement à gauche dans le graphique, Orange calcule cette fois ci le logarithme de l'odds-ratio entre les « yes » et les « no ». Nous avons ainsi $\ln\frac{3.58}{0.22} = 2.79$

[11] http://www.ailab.si/blaz/papers/2004-PKDD.pdf

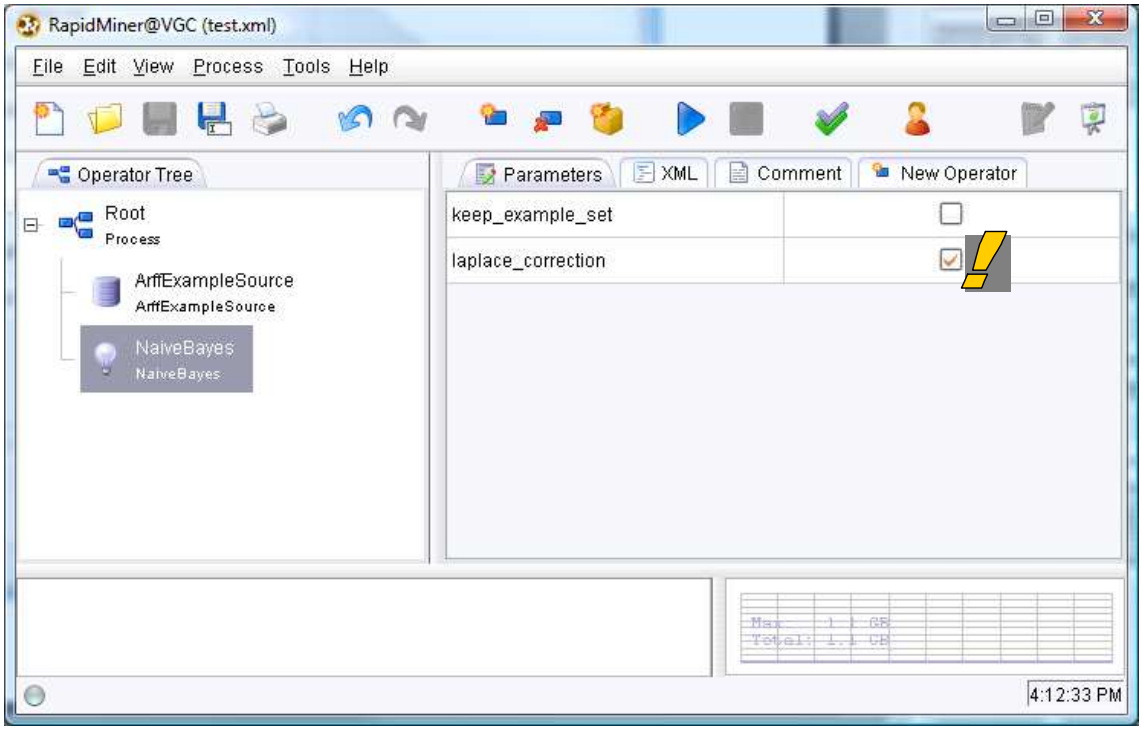


5.5 Rapidminer 4.6.0

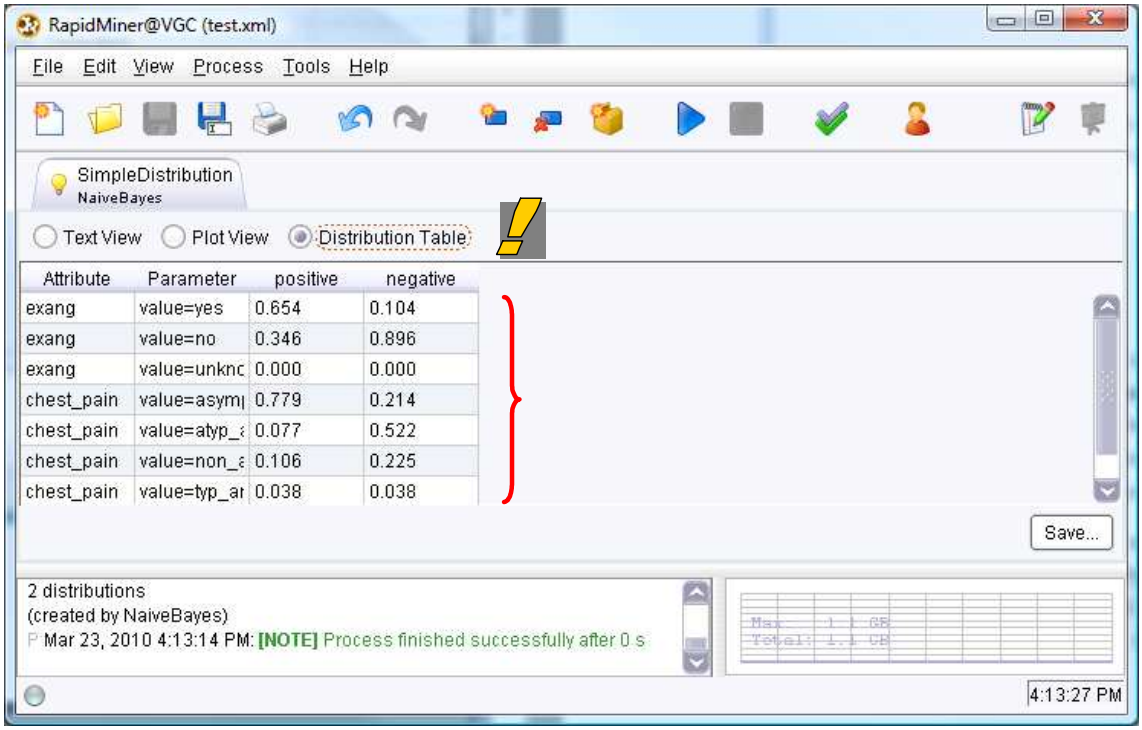
Nous démarrons RapidMiner (http://rapid-i.com/content/view/181/190/). Nous créons un nouveau projet avec le menu FILE / NEW. Nous y intégrons le composant ARFF EXAMPLE SOURCE que nous paramétrons de la manière suivante.



Nous ajoutons à la suite le composant NAIVE BAYES. Notons qu'il peut introduire la correction laplacienne lors des estimations des probabilités.



Il ne nous reste plus qu'à lancer les calculs (PROCESS RUN) après avoir sauvegardé le projet. Différentes représentations sont proposées, parmi lesquels l'affichage des probabilités conditionnelles utilisées pour le classement d'un nouvel individu.



6 Conclusion

Nous avant mis en avant dans ce tutoriel une représentation directement exploitable pour le déploiement des résultats fournis par le classifieur bayesien naïf, elle est sous la forme d'une fonction de classement linéaire portant sur les indicatrices des modalités des descripteurs.

Nous pouvons le faire dans le logiciel Tanagra (version 1.4.36 et ultérieure) parce qu'il restreint les calculs aux seul descripteurs discrets (variables catégorielles).

Ce n'est pas le cas des autres logiciels qui, eux, peuvent traiter indifféremment les descripteurs discrets ou continus (on fait souvent l'hypothèse d'une distribution gaussienne). Cet avantage se révèle être aussi un inconvénient. En effet il devient impossible de proposer une représentation explicite, simple, facilement interprétable et exploitable, du modèle prédictif. Cela constitue un frein certain quant à son utilisation dans les études réelles.

Notons enfin que le logiciel Orange, avec le principe des nomogrammes, propose une représentation très intuitive qui permet de juger rapidement, en un coup d'œil, l'impact comparé des descripteurs sur l'identification des modalités de la variable cible.