1 Objectif

Démarche de modélisation prédictive sous Knime - Régression Logistique.

Knime est un logiciel de data mining librement téléchargeable en ligne (Knime Analytics Platform - https://www.knime.org/knime). Je l'étudie depuis longtemps. Mon premier tutoriel à son propos date de 2008. Je me suis rendu compte récemment que je n'avais jamais écrit un guide « simple » montrant une démarche d'analyse prédictive basique sous cet outil, à savoir : (1) importer les données ; (2) les partitionner en échantillons d'apprentissage et test ; (3) construire le modèle à partir de l'ensemble d'apprentissage ; (4) l'appliquer sur l'ensemble de test pour obtenir la prédiction du modèle ; (5) confronter les valeurs observées et prédites de la variable à prédire à travers la matrice de confusion ; (6) en déduire les indicateurs (mesures) de performances des modèles (taux d'erreur, etc.).

Schématiquement, nous avons :

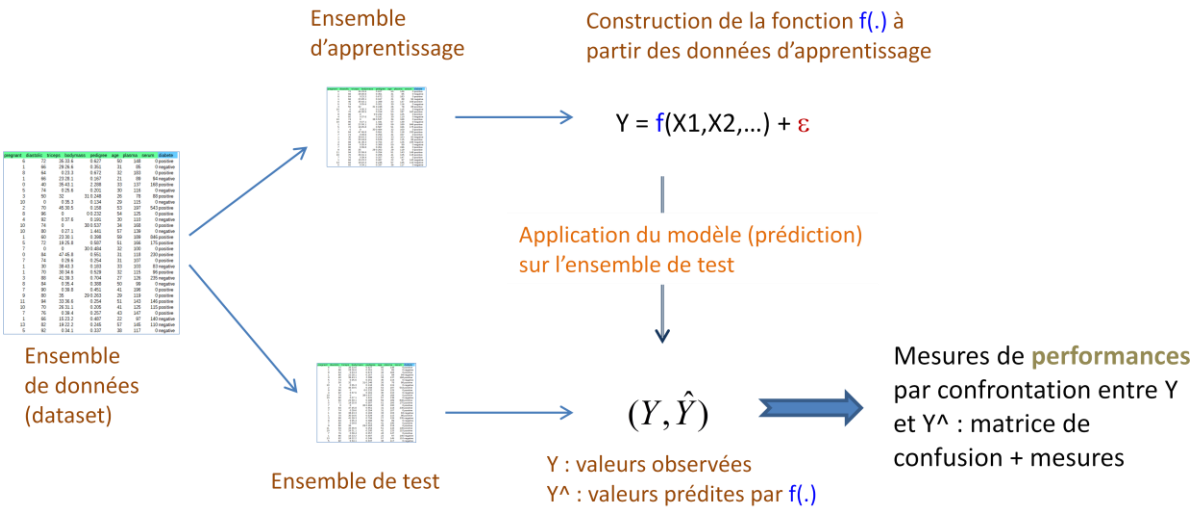


Figure 1 - Schéma apprentissage - test en analyse prédictive (classement)

Dans un processus de « scoring », une variante est apportée à partir du point n°4 : (4') appliquer le modèle sur l'échantillon test pour obtenir le score des individus ; (5') construire la courbe lift cumulée ou courbe de gain à partir des valeurs observées de la variable cible et les scores.

Ce tutoriel retrace toutes ces étapes avec force copies d'écrans comme toujours. La régression logistique est mise à contribution mais le processus est transposable à toute méthode de machine learning. Nous introduirons très brièvement la sélection de variables - à la sauce Knime - dans la dernière partie.

2 Données

Le fichier « Pima Indian Diabetes » décrit un ensemble de personnes amérindiennes de sexe féminin souffrant ou non du diabète. Il a le mérite d'être parfaitement identifié par les data miners et est relativement propre même si, par ailleurs, certains éléments interrogent. Des individus par exemple présentent un BMI (*body mass index* - indice de masse corporelle) nul. « Légère comme une plume » est une vue de l'esprit. On peut penser que les valeurs n'ont pas été référencées dans ce cas. Nous passons outre néanmoins. Notre objectif est de décrire les manipulations sous Knime, et non pas procéder à une analyse approfondie des résultats.

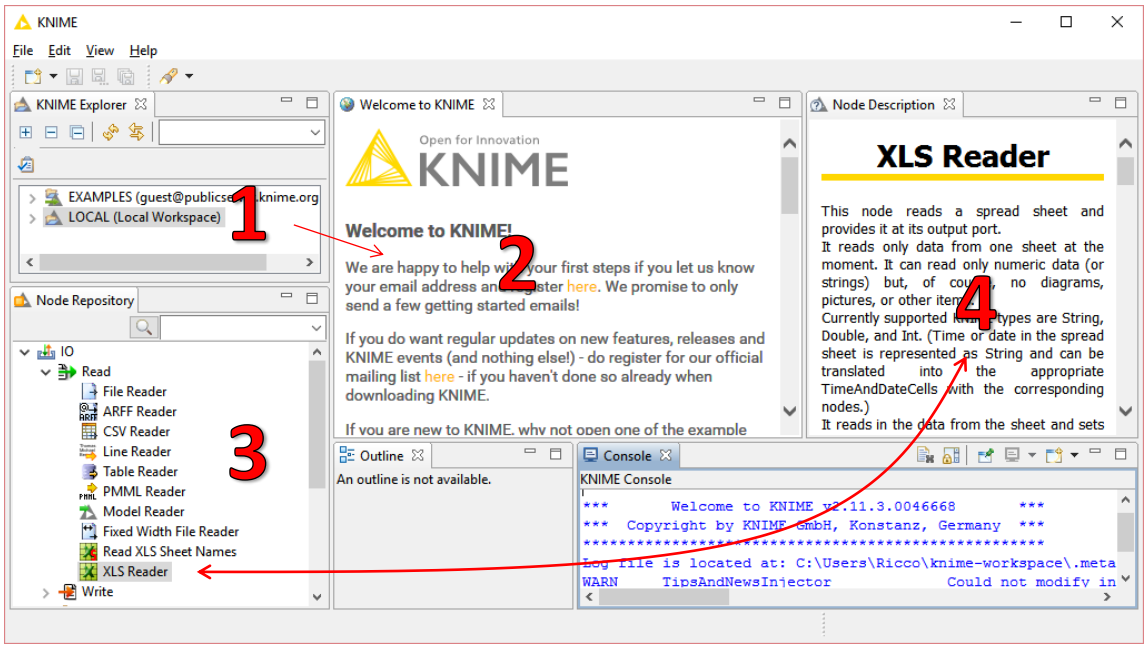
Voici les premières lignes de notre fichier. DIABETE est la cible. Les autres colonnes constituent les variables prédictives candidates. Nous disposons de 768 observations.

pregnant	diastolic	triceps	bodymass	pedigree	age	plasma	serum	diabete
6	72	35	33.6	0.627	50	148	0	positive
1	66	29	26.6	0.351	31	85	0	negative
8	64	0	23.3	0.672	32	183	0	positive
1	66	23	28.1	0.167	21	89	94	negative
0	40	35	43.1	2.288	33	137	168	positive

3 Apprentissage - Test sous Knime

La version **Knime Analytics Platform** peut être chargée librement sur le site de l'éditeur. Son installation sur notre machine n'appelle pas de compétences particulières.

3.1 Démarrage du logiciel



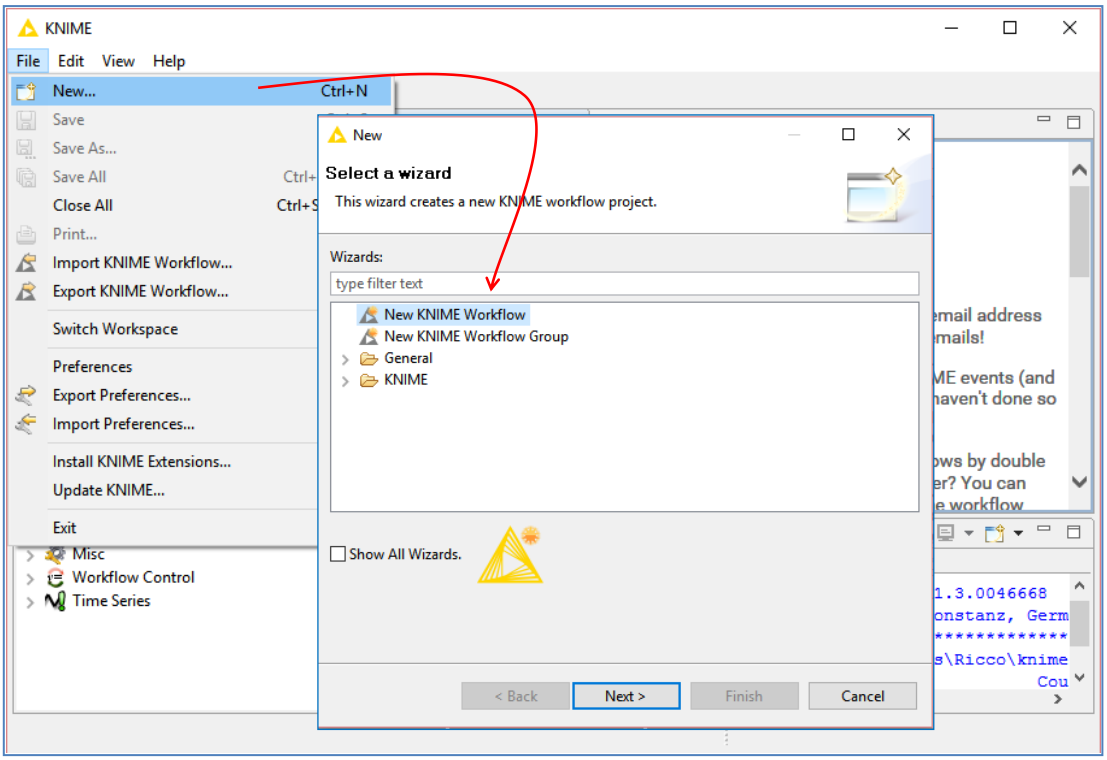
Au démarrage de Knime, nous disposons d'une fenêtre principale subdivisée en plusieurs parties. Voyons-en les plus importantes dans notre contexte :

1. « L'explorer » fait office de gestionnaire de projets. Nous y observons les différentes analyses que nous menons lors de l'utilisation de Knime.
2. Correspond à l'espace de travail (un message d'accueil y est affiché puisque nous n'avons créé aucun projet pour l'instant). Nous y formerons notre « workflow » c.-à-d. notre flux de traitements consistant à appliquer différentes opérations sur nos données, éventuellement transformées en cours de route. Le workflow visible correspond au projet sélectionné dans la fenêtre (1).
3. Le « node repository » recense les outils disponibles dans Knime. Les plus intéressantes pour nous, dans une phase de prise en main, sont les opérateurs de manipulation de données (importation, transformation, etc.), les algorithmes de machine learning (régression logistique, réseaux de neurones, arbres de décision, etc.), et les composants d'évaluation des classifieurs (matrice de confusion, courbes, etc.).
4. Attention appréciable, une aide apparaît lorsque nous sélectionnons un outil. Nous savons tous combien lire la documentation n'est pas une activité naturelle lorsqu'on prend en main un logiciel (on commence à la lire quand on n'arrive pas à le faire fonctionner correctement généralement, rarement avant). Là, nous disposons immédiatement d'une description succincte de la finalité du composant et de ses paramètres. Dans la copie d'écran, l'outil XLS Reader permettant de charger des feuilles Excel est activé dans (3), nous voyons dans (4) : « The node reads a spreadsheet… ». Remarque : Nous retirerons cette partie dans les copies d'écran de notre tutoriel afin de disposer de plus d'espace pour décrire les successions d'opérations dans le *workflow*.

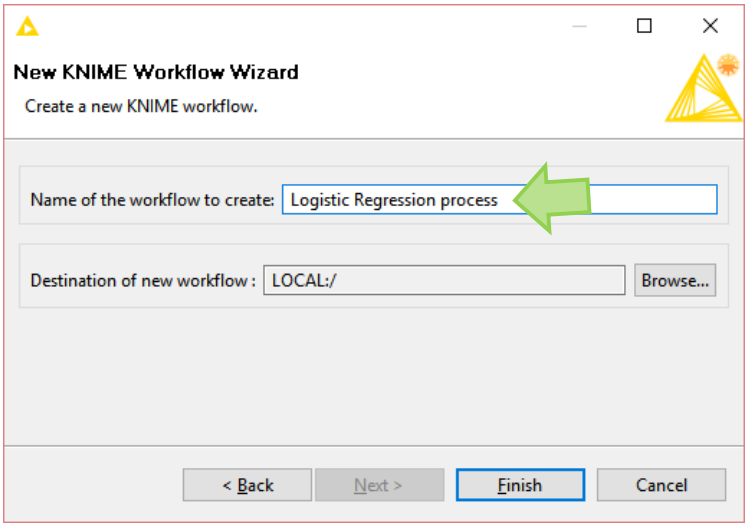
Dans ce qui suit, nous créons un nouveau « Workflow » pour définir notre étude.

3.2 Création d'un projet

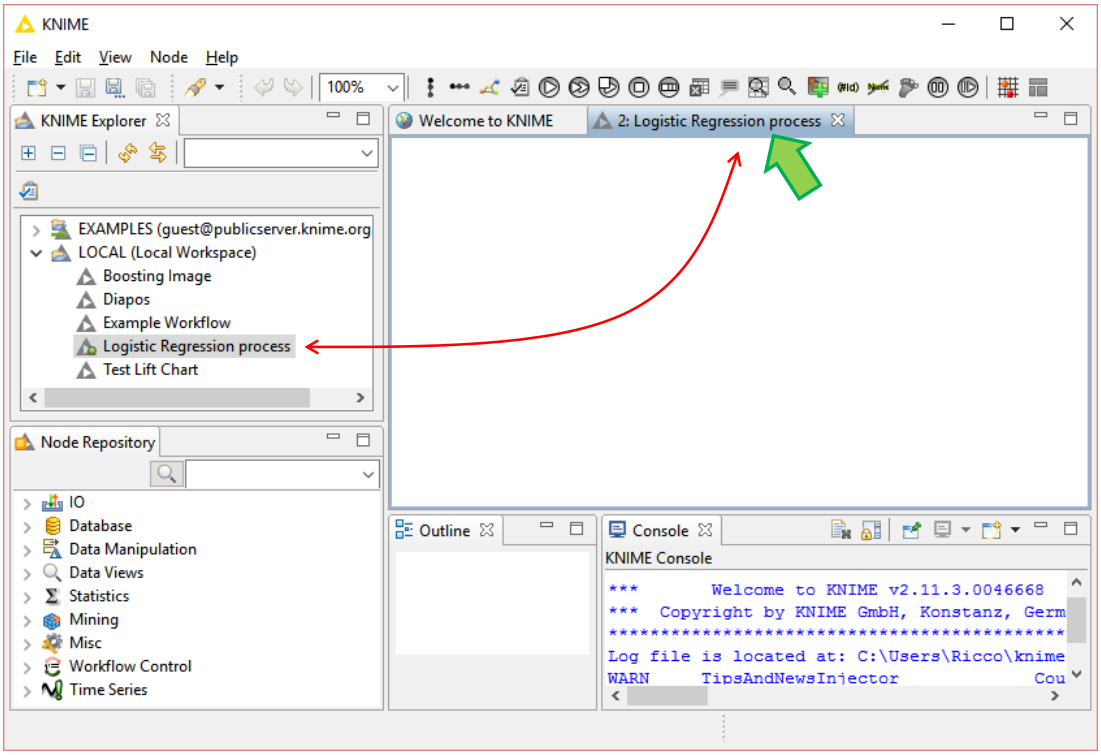
Nous actionnons le menu FILE / NEW pour définir un projet. Une boîte de dialogue apparaît, nous invitant à spécifier le type de projet que nous souhaitons créer. Nous optons pour « New KNIME Workflow ». Puis, nous cliquons sur SUIVANT (NEXT).



La fenêtre suivante nous invite à saisir le nom de notre projet. Nous l'appelons « Logistic Regression process » :

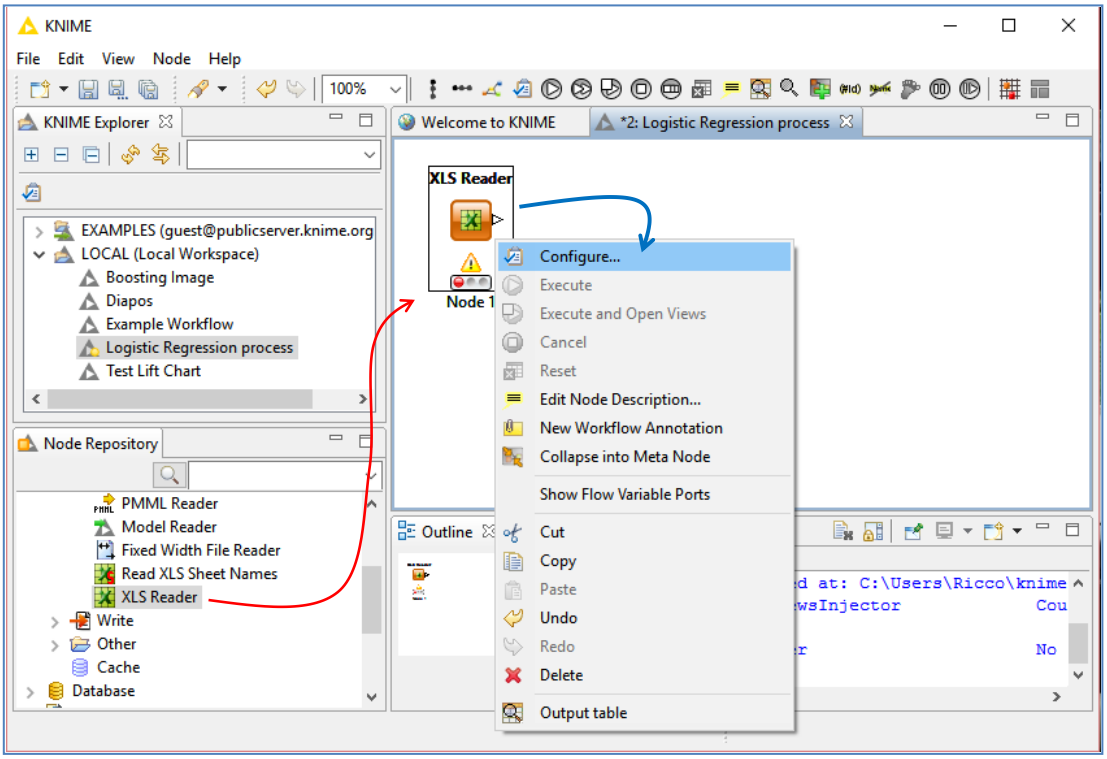


Le nouveau projet est visible dans l'explorer (en compagnie des projets définis pour des études antérieures sur ma machine). L'espace de travail permet de définir notre workflow (diagramme de traitements) consistant à appliquer différentes opérations sur nos données.

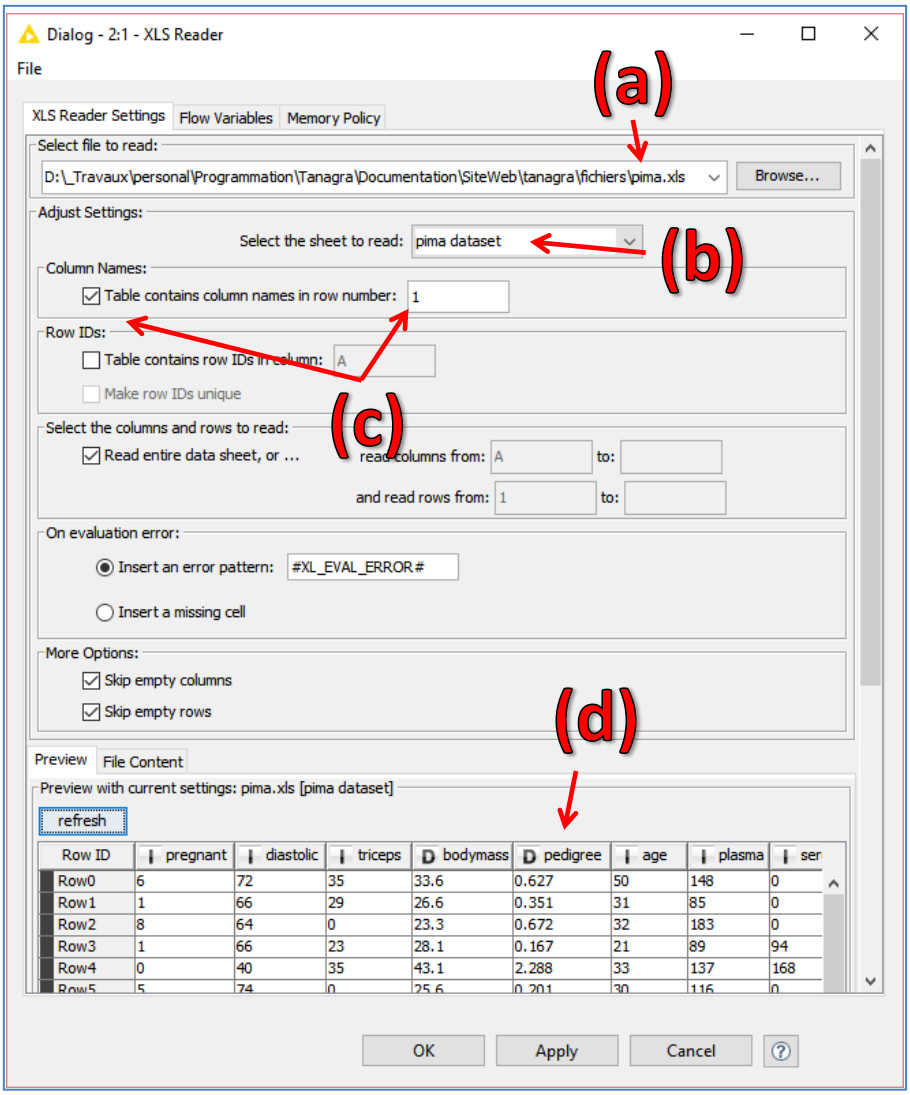


3.3 Chargement des données

Nous insérons le composant **XLSReader** dans l'espace de travail pour importer le fichier « **pima.xls** ». Un led (petit bouton) rouge lui est associé. Le composant est en attente de paramétrage. Il n'est pas possible d'exécuter l'opération à ce stade.

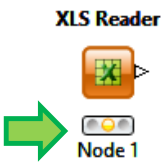


Nous paramétrons le composant en actionnant le menu contextuel CONFIGURE.

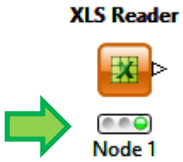


Nous indiquons : (**a**) le nom du fichier ; (**b**) la feuille à traiter dans le classeur Excel (par défaut, la première feuille est utilisée) ; (**c**) le rôle joué par la première ligne qui correspond en fait aux noms des variables ; (**d**) il est possible pré-visualiser les données dans PREVIEW, que l'on peut rafraîchir (REFRESH) au gré des modifications d'options que nous effectuons.

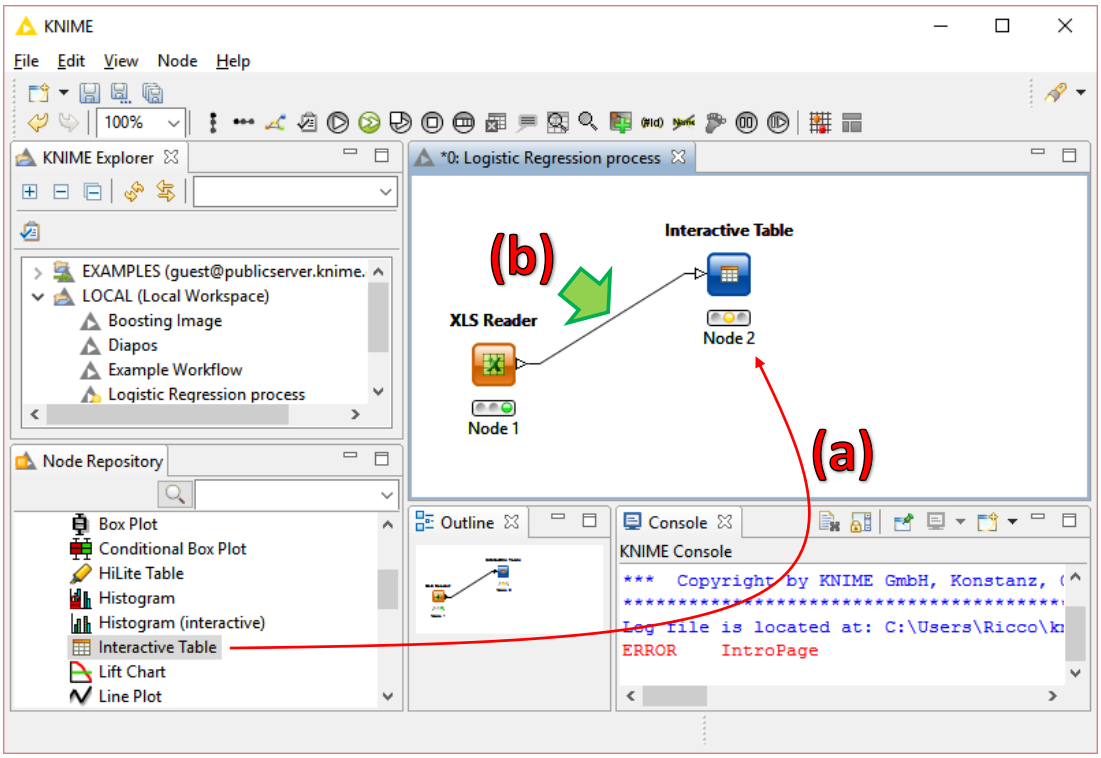
Lorsque nous cliquons sur OK, le led passe au jaune. Le paramétrage a été effectué. Il est possible de lancer les calculs, ce qui n'a pas été fait encore.



Nous actionnons le menu contextuel EXECUTE pour concrétiser l'importation des données. Le led témoin passe au vert, indiquant le succès de l'opération.



Pour visualiser le contenu du jeu de données, nous utilisons le composant **INTERACTIVE TABLE**. Nous l'insérons dans l'espace de travail (**a**) et nous lui relions (et non pas l'inverse) l'outil **XLS Reader** (**b**).



Il n'y a pas de paramètres à spécifier. Nous cliquons sur le menu EXECUTE and OPEN VIEWS pour visualiser les informations.

Table View - 0:2 - Interactive Table

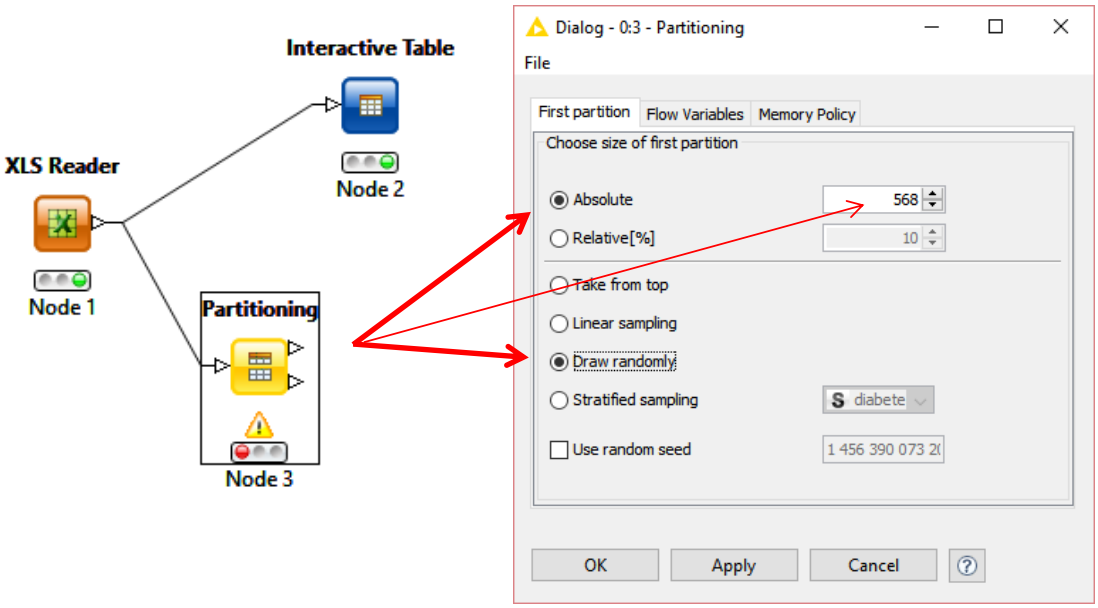
File Hilite Navigation View Output

Row ID	pregnant	diastolic	triceps	bodymass	pedigree	age	plasma	serum	diabete
Row0	6	72	35	33.6	0.627	50	148	0	positive
Row1	1	66	29	26.6	0.351	31	85	0	negative
Row2	8	64	0	23.3	0.672	32	183	0	positive
Row3	1	66	23	28.1	0.167	21	89	94	negative
Row4	0	40	35	43.1	2.288	33	137	168	positive
Row5	5	74	0	25.6	0.201	30	116	0	negative
Row6	3	50	32	31	0.248	26	78	88	positive
Row7	10	0	0	35.3	0.134	29	115	0	negative

Nous disposons exactement des mêmes valeurs que lors de la prévisualisation du fichier Excel (Section 2). Heureusement !

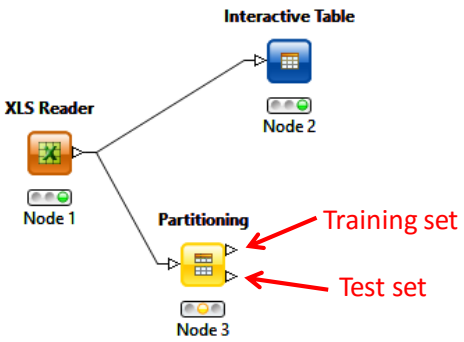
3.4 Partitionnement en apprentissage et test

Nous exploitons l'outil **PARTITIONING** (branche Data Manipulation / Row / Transform dans le Node Repository) pour scinder aléatoirement la base en échantillons d'apprentissage (568 observations) et de test (200). Nous lui connectons la source **XLSReader**.



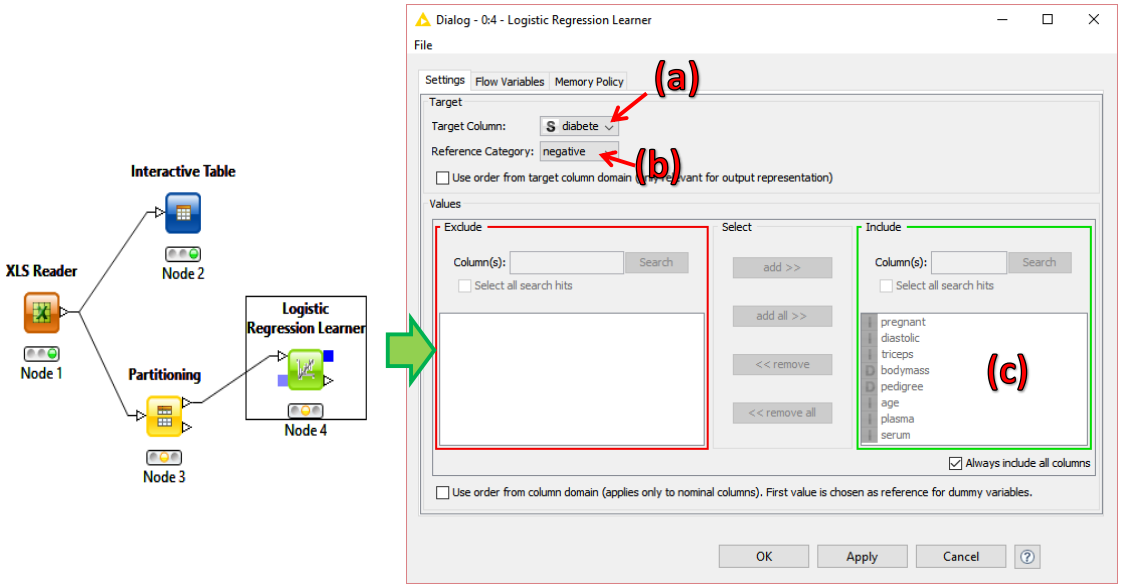
Nous indiquons les informations adéquates dans la fenêtre de paramétrage que nous faisons afficher avec le menu contextuel CONFIGURE.

Après la réalisation du partitionnement avec le menu EXECUTE, il est possible de visualiser les sous-ensembles de données avec INTERACTIVE TABLE. Pour notre part, nous observerons surtout que PARTITIONING propose deux sorties : la première pour les individus sélectionnés (first partition), lesquels correspondent à l'échantillon d'apprentissage pour nous ; la seconde pour l'échantillon de test.



3.5 Construction du modèle

Nous ajoutons le composant LOGISTIC REGRESSION LEARNER (Statistics / Regression) dans le workflow. Nous lui connectons la sortie « training set » de PARTITIONING. Nous le paramétrons comme suit.



(a) DIABETE est la variable à prédire, seuls les attributs catégoriels sont autorisés ici ; (b) « negative » est la modalité de référence c.-à-d. nous ciblons la modalité « positive » (vs. « negative ») ; (c) toutes les autres colonnes sont considérées comme variables explicatives (les explicatives catégorielles sont automatiquement codées à l'aide d'indicatrices dixit l'aide, la première modalité - dans l'ordre lexicographique - servant de référence).

Logistic Regression Result View - 0:4 - Logistic Regressi...

File

Statistics on Logistic Regression

Logit	Variable	Coeff.	Std. Err.	z-score	P>|z|
positive	pregnant	0,1345	0,0368	3,6547	0,0003
	diastolic	-0,0096	0,0066	-1,4509	0,1468
	triceps	-0,0006	0,0079	-0,0815	0,935
	bodymass	0,0806	0,0174	4,6387	3,51E-6
	pedigree	1,3438	0,3637	3,6947	0,0002
	age	0,014	0,0106	1,3183	0,1874
	plasma	0,0322	0,0041	7,8723	3,44E-15
	serum	-0,0005	0,001	-0,5398	0,5893
	Constant	-8,2354	0,8271	-9,9567	0.0

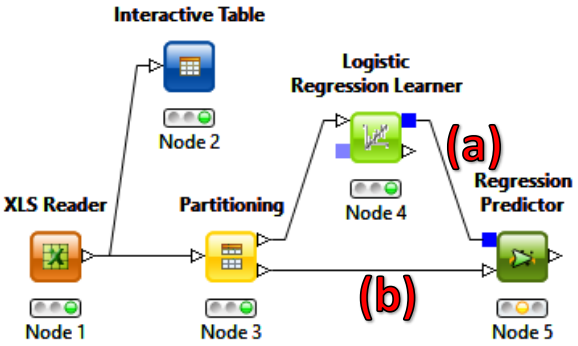
Log-likelihood = -267,8158
Number of iterations = 10

Figure 2 - Paramètres estimés de la régression logistique

Les coefficients estimés sont affichés dans une nouvelle fenêtre après que l'on ait cliqué sur le menu EXECUTE and OPEN VIEWS. Nous disposons des coefficients estimés, de leurs écarts-type, des z-score pour les tests individuels de significativité, et de la p-value dudit test. La log-vraisemblance du modèle est égale à **LL = -267.8158**.

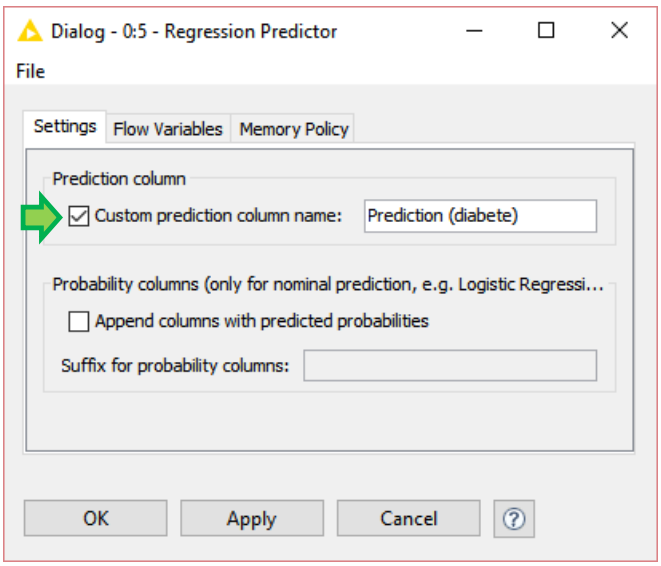
3.6 Prédiction sur l'échantillon test

Nous devons appliquer le modèle sur l'échantillon test pour obtenir les prédictions. Le composant REGRESSION PREDICTOR (Statistics / Regression) prend en entrée (**a**) le classifieur élaboré par le LEARNER et (**b**) l'échantillon test issu de PARTITIONING. Nous avons la configuration suivante.



Notons que le design du workflow est parfaitement raccord avec la présentation adoptée dans notre introduction (Section 1, Figure 1 ; c'est plutôt l'inverse, je me suis calé sur l'organisation de Knime). Pédagogiquement, cette similitude visuelle est très intéressante.

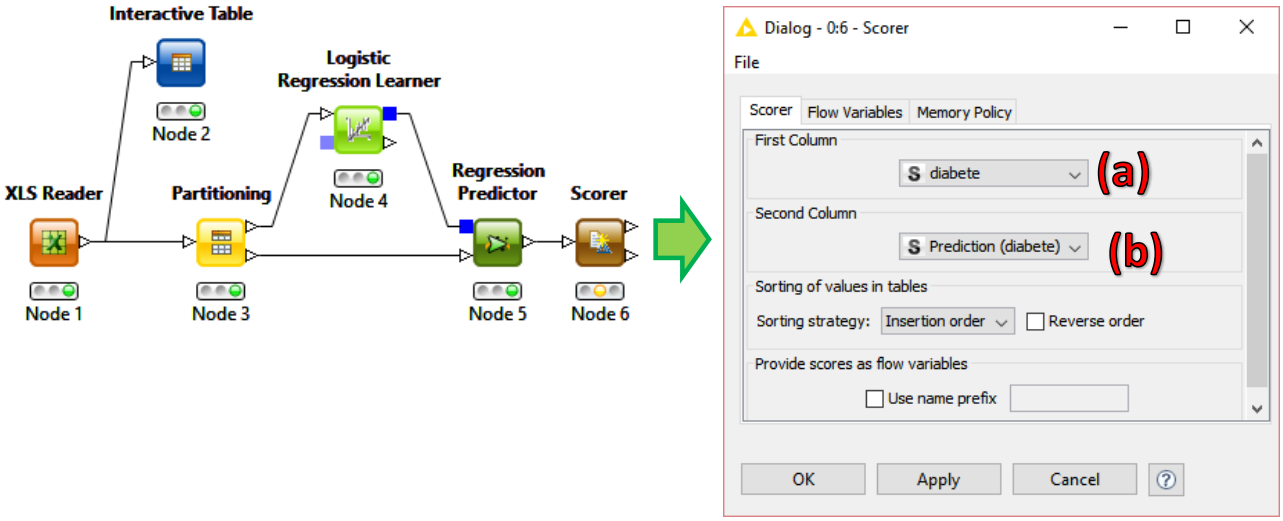
Spécifions les options de REGRESSION PREDICTOR…



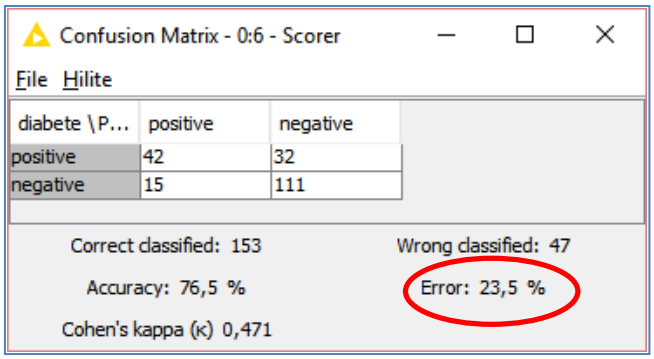
Pour l'instant, nous reviendrons dessus lors du scoring, nous nous contentons de produire la prédiction de la variable DIABETE.

3.7 Matrice de confusion et indicateurs

Le composant SCORER (Mining / Scoring) complète notre dispositif. Il sert à produire la matrice de confusion à partir des valeurs observées et prédites de DIAEBETE c.-à-d. nous mettons en opposition DIABETE (**a**) et PREDICTION(DIABETE) (**b**).



Un clic sur EXECUTE and OPEN VIEWS produit une nouvelle fenêtre de visualisation.

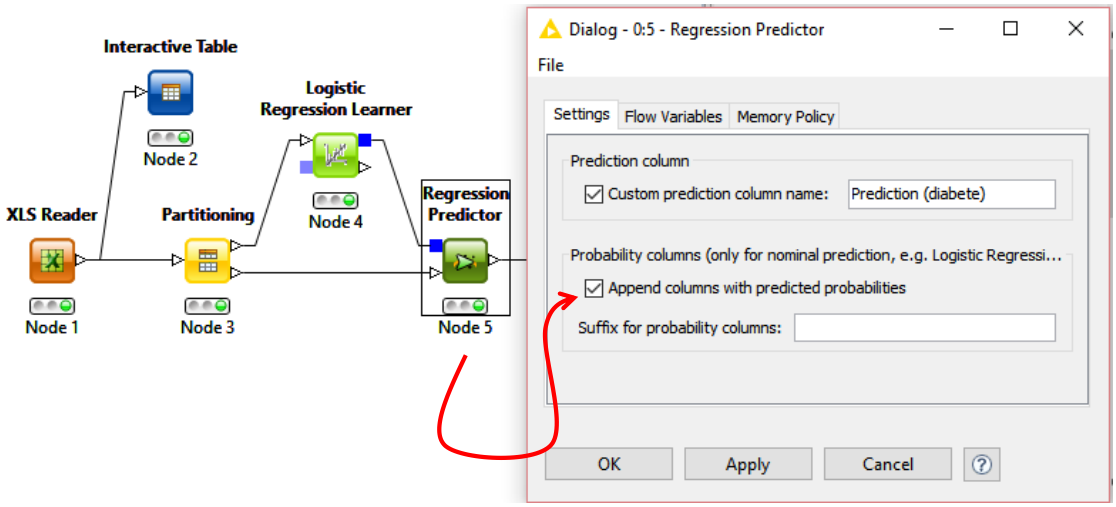


Nous distinguons la matrice de confusion. Le taux d'erreur en test est de **23.5%**.

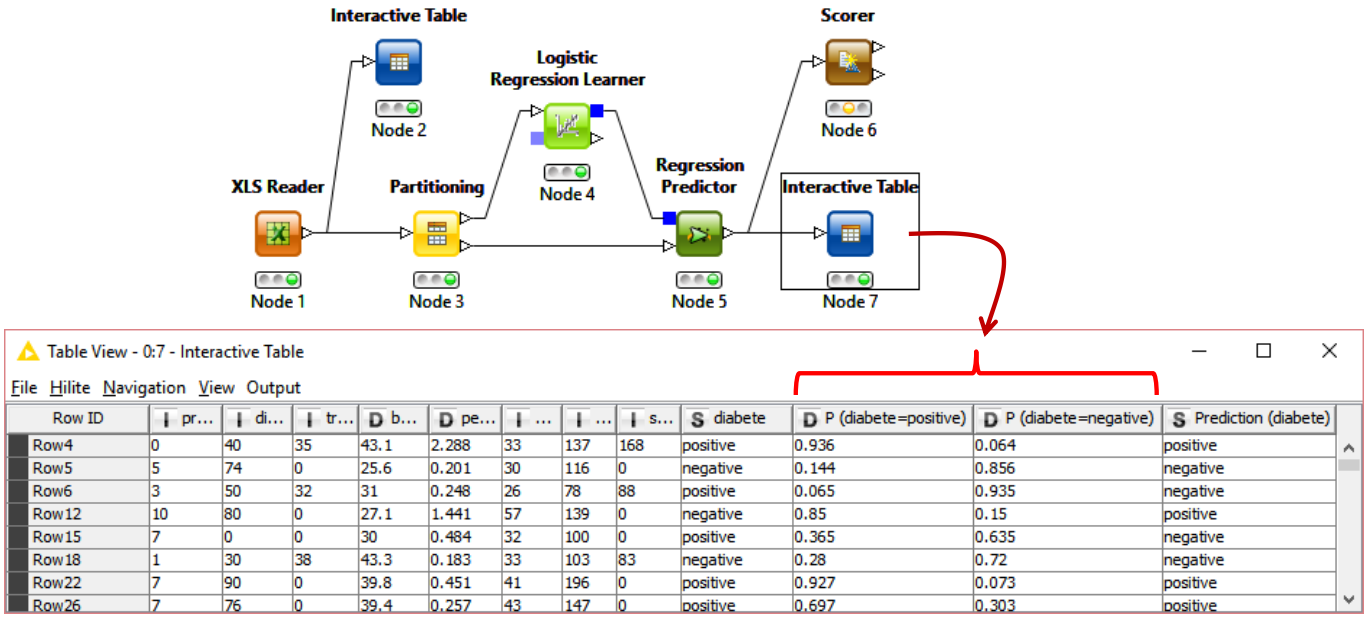
4 Scoring et courbe de gain

Dans le ciblage (ciblage marketing notamment), l'objectif n'est pas tant de prédire la classe d'appartenance que de quantifier le degré de positivité. A la sortie, nous devons pouvoir annoncer, pour un nombre de personnes sollicitées dans une campagne commerciale par exemple, combien achèteraient réellement le produit si l'on sélectionnait les individus qui présentent les scores les plus élevés.

Nous devons modifier le paramétrage de REGRESSION PREDICTOR afin qu'il produise les probabilités d'appartenance aux classes.



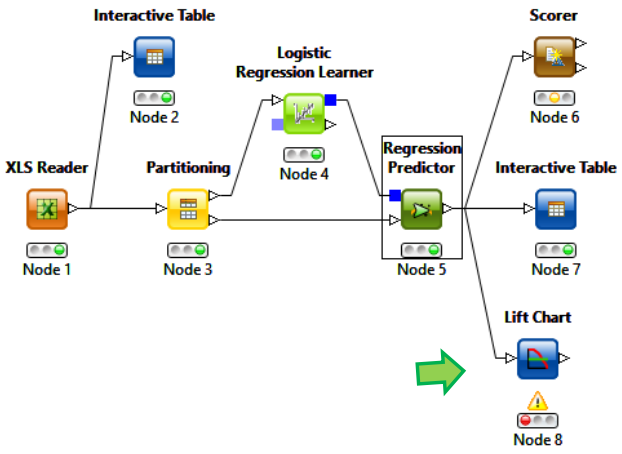
Lançons les calculs et ajoutons un INTERACTIVE TABLE pour voir ce que donne tout ça.



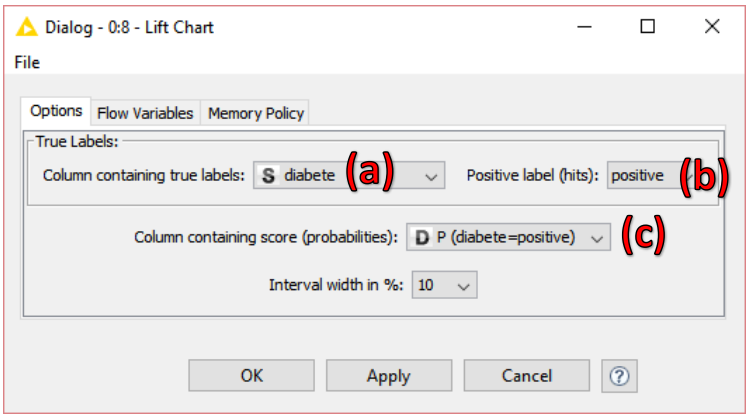
Row ID	pr...	di...	tr...	b...	pe...	...	...	s...	diabete	P (diabete=positive)	P (diabete=negative)	Prediction (diabete)
Row4	0	40	35	43.1	2.288	33	137	168	positive	0.936	0.064	positive
Row5	5	74	0	25.6	0.201	30	116	0	negative	0.144	0.856	negative
Row6	3	50	32	31	0.248	26	78	88	positive	0.065	0.935	negative
Row12	10	80	0	27.1	1.441	57	139	0	negative	0.85	0.15	positive
Row15	7	0	0	30	0.484	32	100	0	positive	0.365	0.635	negative
Row18	1	30	38	43.3	0.183	33	103	83	negative	0.28	0.72	negative
Row22	7	90	0	39.8	0.451	41	196	0	positive	0.927	0.073	positive
Row26	7	76	0	39.4	0.257	43	147	0	positive	0.697	0.303	positive

Deux nouvelles colonnes apparaissent [P(diabete = positive) et P(diabete = negative)]. Le premier nous intéresse en particulier puisqu'il correspond au « score » de positivité des individus que nous exploiterons par la suite (la régression logistique a de particulier qu'il produit une bonne estimation de la probabilité d'être positif, ce qui explique en partie son large succès dans les différents domaines d'applications).

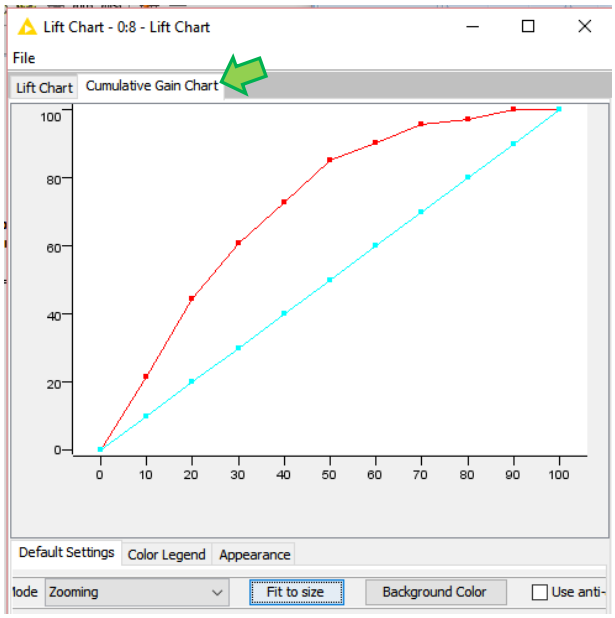
Il ne nous reste plus qu'à introduire le composant LIFT CHART (Data Views) et lui connecter le REGRESSION PREDICTOR.



La boîte de dialogue de paramétrage permet d'indiquer la variable à prédire (**a**), la modalité cible (**b**) et la colonne de score (**c**).



Avec EXECUTE and OPEN VIEWS, nous accédons à la fenêtre de visualisation. Nous activons l'onglet « Cumulative Gain Chart ».

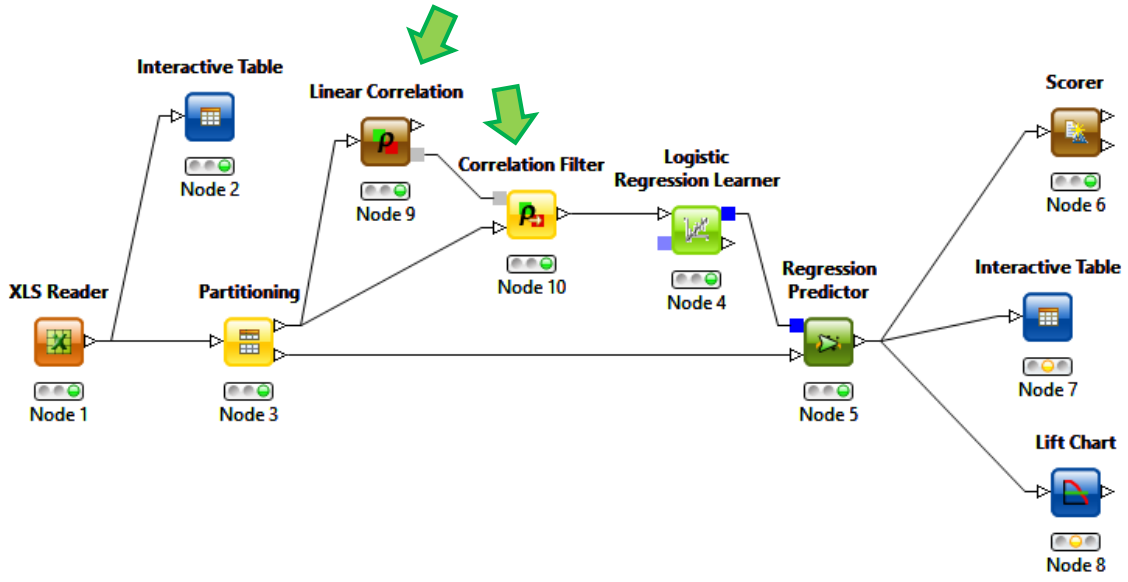


5 Sélection de variables

Plusieurs coefficients ne sont pas significatifs (au sens du test de Wald à 5%) dans notre régression logistique (Figure 2). Knime ne semble pas proposer un dispositif intégré de sélection de variables pour la régression logistique (de type optimisation d'un critère Akaike à l'instar de ce que fait StepAIC [MASS] de R, ou de type statistique comme le propose Tanagra)[1]. Une procédure « wrapper » est en revanche disponible[2].

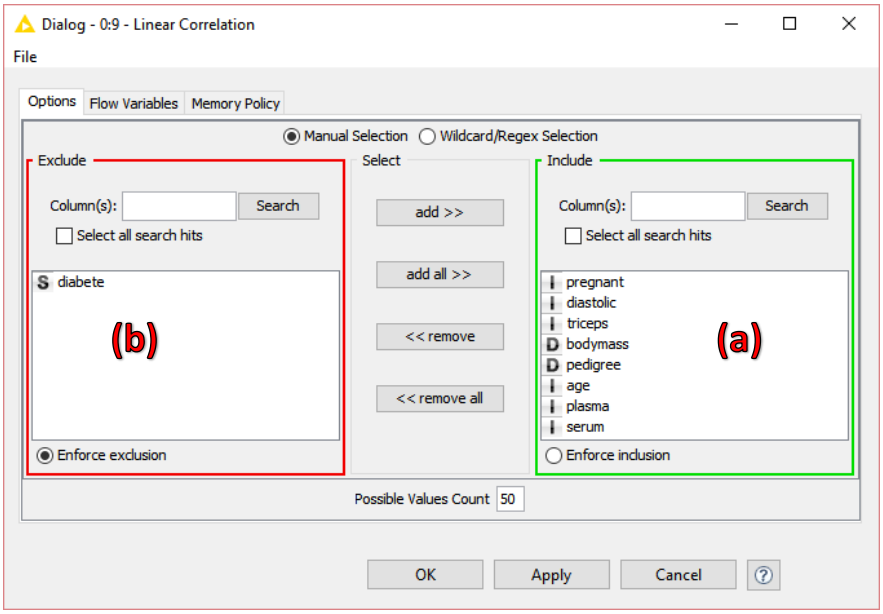
Dans ce tutoriel, nous nous intéressons à une procédure de filtrage basée sur les corrélations. Les variables redondantes sont éliminées en amont, avant le processus d'apprentissage du modèle. L'approche peut être très rapide pour le traitement des grandes bases, c'est sont principal intérêt. En revanche, rien ne dit que les variables ainsi sélectionnées seront pertinentes dans le modèle qui viendra par la suite. En effet, les caractéristiques de la méthode d'apprentissage utilisée en aval n'entre pas en ligne de compte lors du filtrage des attributs.

Nous complétons le diagramme avec deux nouveaux outils. **Notons attentivement la réorganisation des connexions entre les composants**.

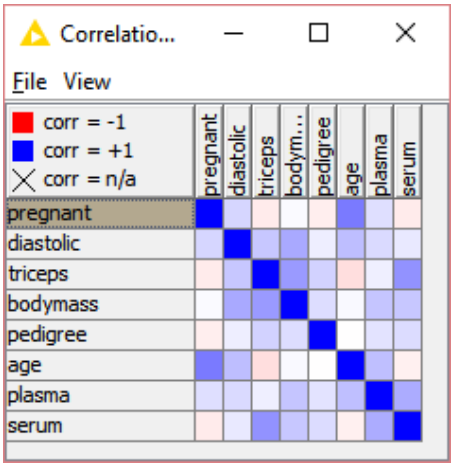


LINEAR CORRELATION se charge de calculer les corrélations entre les variables explicatives (**a**) prises deux à deux sur l'échantillon d'apprentissage. La variable cible n'est pas intégrée dans ce processus (**b**). Nous spécifions les paramètres suivants avec CONFIGURE.

[1] http://tutoriels-data-mining.blogspot.fr/2012/01/regression-logistique-sur-les-grandes.html
[2] http://tutoriels-data-mining.blogspot.fr/2010/01/wrapper-pour-la-selection-de-variables.html



La matrice des corrélations est présentée de manière visuelle (EXECUTE and OPEN VIEWS).



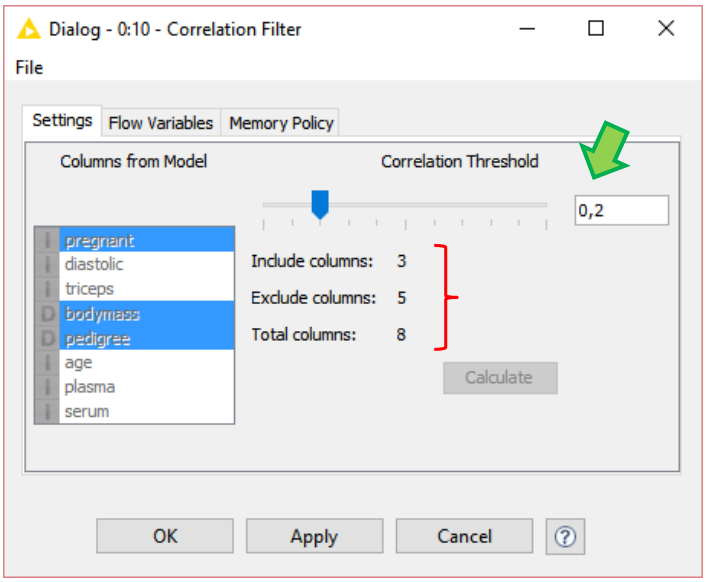
On note par exemple que ''pregnant'' est corrélée positivement avec ''age'', mais l'est négativement avec ''triceps'', etc.

CORRELATION FILTER se charge d'éliminer les variables redondantes. L'outil s'appuie sur une stratégie assez simple (voir la documentation). Pour chaque variable, nous comptons le nombre de congénères qui lui sont significativement liés en fixant une valeur seuil de corrélation (paramètre à fixer). La variable qui comporte le plus de victoires (au sens de corrélations significatives) est conservée. Celles qui lui sont significativement liées sont éliminées. Et l'approche procède ainsi par élimination successives jusqu'à ce qu'il ne soit plus possible de procéder à un retrait.

Deux remarques viennent tout de suite à l'esprit :

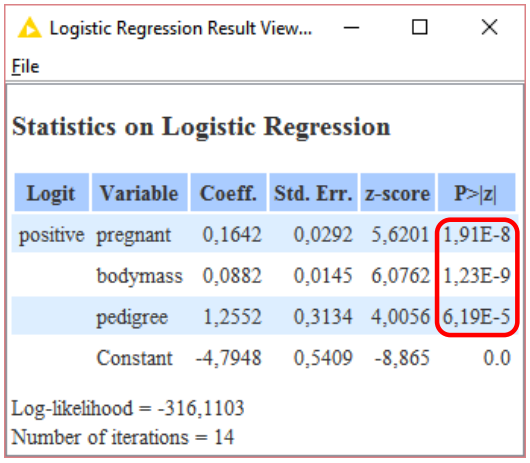
1. Le nombre de variables sélectionnées dépend du seuil. Plus nous réduisons ce dernier, moins nous obtiendrons de variables à la sortie puisque nous avons tendance à considérer comme significatifs tous les liens. L'approche est très fortement dépendante du paramétrage et le choix du seuil n'est pas évident de prime abord.
2. Plus préoccupant encore, on ne tient pas compte du rôle de la variable à prédire dans ce processus. Deux variables corrélées entres elles n'ont pas un statut équivalent selon le degré de liaison (on espère explication) qu'elles ont avec la cible.

Une fois ces réserves émise, voyons comment paramétrer l'outil (CONFIGURE).



Si l'on fixe le seuil à 0.2, nous conserverons : ''pregnant'', ''bodymass'' et ''pedigree''. J'ai clairement procédé par tâtonnement dans notre cas, espérant que conserver 3 variables sur 8 serait une bonne solution. Ne sachant pas bien sûr si elles sont pertinentes dans la prédiction de DIABETE.

Pour le savoir justement, il suffit de relancer LOGISTIC REGRESSION LEARNER.



Logistic Regression Result View...

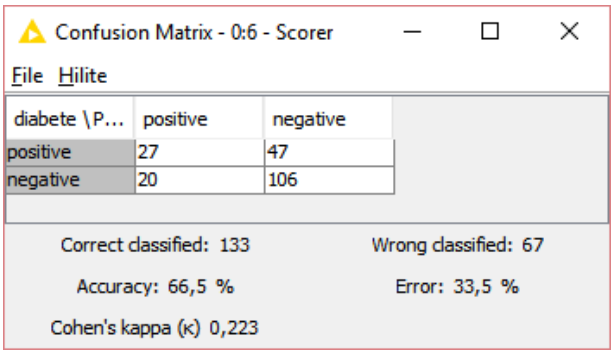
File

Statistics on Logistic Regression

Logit	Variable	Coeff.	Std. Err.	z-score	P>|z|
positive	pregnant	0,1642	0,0292	5,6201	1,91E-8
	bodymass	0,0882	0,0145	6,0762	1,23E-9
	pedigree	1,2552	0,3134	4,0056	6,19E-5
	Constant	-4,7948	0,5409	-8,865	0.0

Log-likelihood = -316,1103
Number of iterations = 14

Tous les coefficients sont significatifs à 5%. Mais est-ce que ces variables sont efficaces pour la prédiction ? Voyons ce que donne la matrice de confusion.



Le taux d'erreur passe à **33.5%** (contre 23.5% avec l'ensemble des prédicteurs).

Les variables conservées semblent pertinentes puisque les coefficients associés sont fortement significatifs dans la régression. En revanche, il semble que nous en ayons trop retirées, il faudrait remonter un peu le seuil, mais de combien ? (**Note :** des tests sous d'autres logiciels montre qu'adjoindre ''plasma'' à ce sous-ensemble de prédicteurs permet de retrouver la qualité prédictive initiale).

6 Conclusion

Qu'importe le logiciel pourvu qu'on ait l'ivresse ai-je coutume de dire aux étudiants. Ce tutoriel montre bien que si nous avons bien en tête les schémas de l'analyse prédictive, pouvoir les mettre en œuvre sous tout type d'outils - Knime ici en l'occurrence - n'est jamais vraiment compliqué. Il faut simplement être en mesure de se poser les bonnes questions à chaque étape du processus.