1 Objectif

Vérification et traitement des données manquantes et/ou incohérentes durant l'importation des données dans Tanagra.

Jusqu'à la version 1.4.41, Tanagra ne gérait pas les données manquantes parce qu'il me semblait pédagogiquement intéressant que les étudiants, qui constituent quand même le principal public de Tanagra, réfléchissent et traitent explicitement en amont ce problème difficile. Le pire serait de s'en remettre aveuglément au logiciel c.-à-d. de le laisser choisir à notre place un traitement automatique inadapté au cadre de notre étude, aux caractéristiques de nos données, etc.

Ainsi, Tanagra se contentait de tronquer le fichier à l'importation dès le premier obstacle rencontré. Ce traitement sans concessions déroutait souvent l'utilisateur, d'autant plus qu'aucun message d'erreur n'était envoyé. Il se demandait alors pourquoi, alors que toutes les conditions semblent réunies, les données n'étaient pas correctement chargées.

Avec la nouvelle **version 1.4.42**, l'importation des fichiers TXT (fichiers textes avec séparateur tabulation), des fichiers XLS (Excel 97-2003), et le transfert des données via les add-in pour Excel (jusqu'à Excel 2010) et LibreOffice 3.4/OpenOffice 3.3, ont été modifiés. Tanagra parcourt bien toutes les lignes de la base. Il se contente simplement de sauter les observations incomplètes et/ou comportant des incohérences (ex. une valeur non numérique pour un attribut initialement détecté quantitatif). Et, surtout, **un message d'erreur explicite comptabilise le nombre de lignes ignorées**. Ainsi, l'utilisateur est mieux informé. Cette approche très simpliste correspond à la stratégie « listwise deletion »[1]. Ses faiblesses sont largement identifiées[2]. Pour nous, il s'agit surtout d'alerter l'utilisateur sur les problèmes rencontrés lors de la lecture du fichier de données. Libre à lui de poursuivre directement si ce traitement par défaut lui convient. Ce qui n'est pas très conseillé quand même dans la plupart des cas. Les études que nous avons menées pour la régression logistique le montrent bien[3].

Dans ce tutoriel, nous montrons la gestion des données manquantes lors de l'envoi des données d'Excel vers Tanagra via la macro complémentaire Tanagra.xla. Certaines cellules du fichier Excel sont vides. Cet exemple illustre bien le nouveau comportement de Tanagra. Nous obtiendrions des résultats identiques si nous importions directement le fichier XLS ou si nous importions le fichier au format TXT correspondant.

2 Données

Nous utilisons le fichier « ronflement_with_missing_empty.xls »[4]. Il contient 30 observations et 7 variables. Certaines lignes [14 pour être précis] comportent des cellules vides. Elles seront éliminées durant l'importation des données dans Tanagra.

[1] http://www.uvm.edu/~dhowell/StatPages/More_Stuff/Missing_Data/Missing.html

[2] http://en.wikipedia.org/wiki/Listwise_deletion

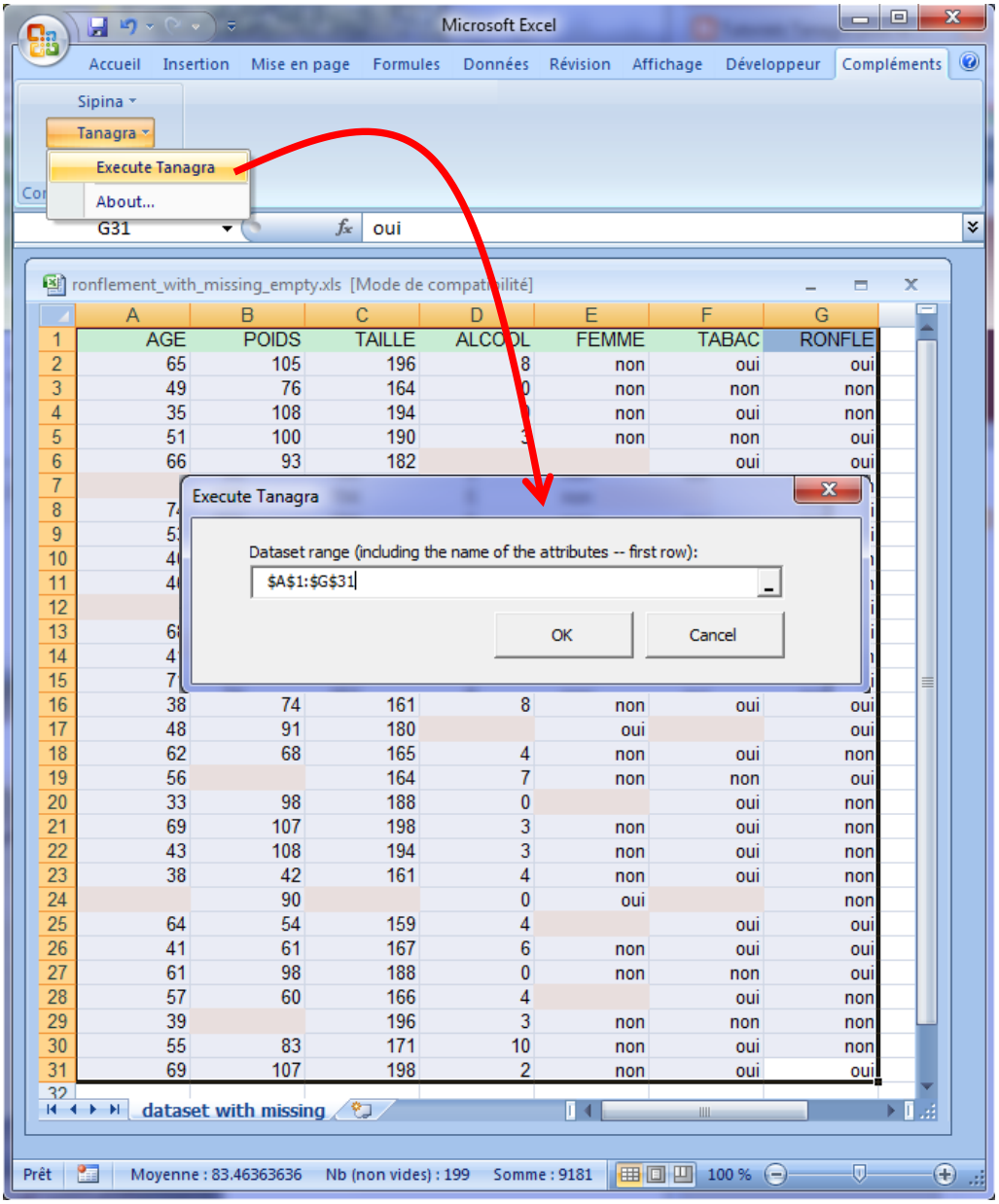
[3] http://tutoriels-data-mining.blogspot.com/2011/12/donnees-manquantes-regression.html

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

3 Transfert des données d'Excel vers Tanagra

Nous utilisons la macro complémentaire tanagra.xla pour importer les données dans Tanagra[5]. Nous aurions tout aussi bien pu lire directement le fichier[6] ou encore l'importer après l'avoir transformé au format texte avec séparateur tabulation[7].

Après avoir sélectionné la plage de données, nous actionnons le menu COMPLEMENTS / TANAGRA / EXECUTE TANAGRA. Nous vérifions que les références des cellules sont correctes. Nous validons en cliquant sur le bouton OK.

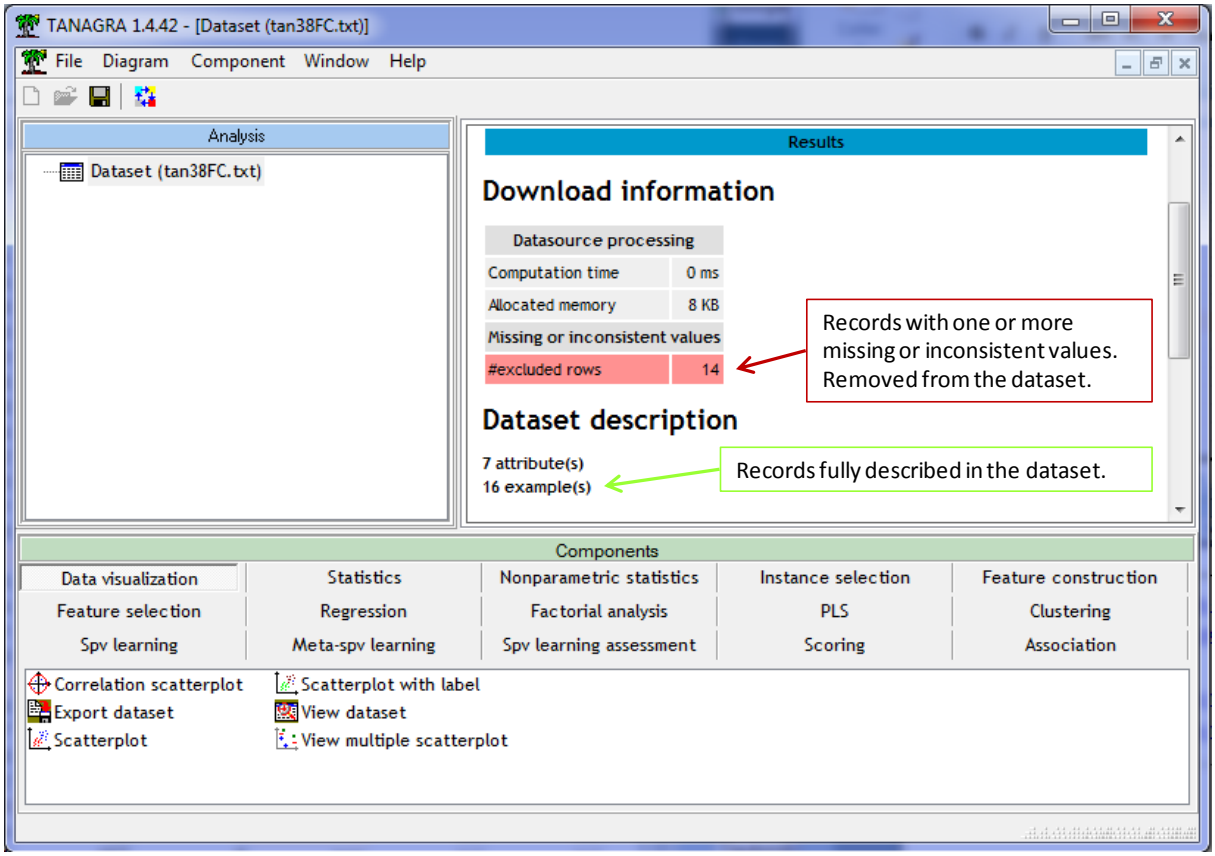


[5] http://tutoriels-data-mining.blogspot.com/2010/08/ladd-in-tanagra-pour-excel-2007-et-2010.html pour Excel 2007 et 2010 ; http://tutoriels-data-mining.blogspot.com/2008/03/importation-fichier-xls-excel-macro.html pour les versions antérieures.

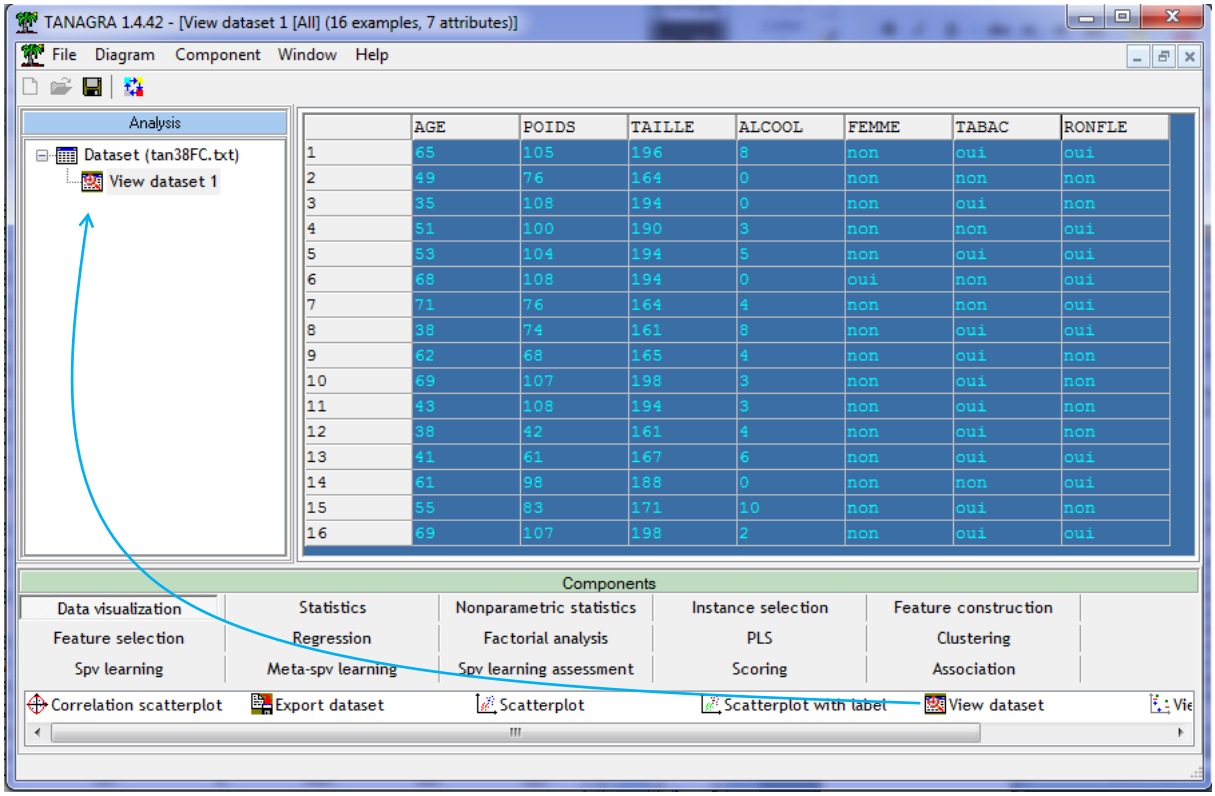
[6] http://tutoriels-data-mining.blogspot.com/2008/03/importation-fichier-xls-excel-mode.html

[7] http://tutoriels-data-mining.blogspot.com/2008/03/importation-fichier-texte.html

Tanagra est automatiquement démarré, les données sont importées. Il nous signale que 16 observations sont disponibles. Il indique également que 14 lignes ont été ignorées.



Nous pouvons visualiser les observations conservées à l'aide du composant VIEW DATASET.



4 Conclusion

Par rapport aux précédentes versions de Tanagra, on pouvait craindre que l'introduction de ces vérifications supplémentaires ne grève les performances. Nous avons multiplié les tests sur de nombreux fichiers de données. Il en ressort que l'augmentation du temps de traitement est minime. Elle n'est réellement perceptible que sur les bases de très grande taille. Pour donner un ordre d'idées, sur la base « waveform » utilisée pour la comparaison de performances de plusieurs implémentations de la régression logistique[8], avec 300.000 observations et 122 colonnes, à configuration machine égale, la durée de l'importation est passée de 9 secondes (version 1.4.41) à 11 secondes (version 1.4.42). L'écart n'est certes pas négligeable. Mais le temps de traitement de Tanagra reste quand même très en deçà des autres logiciels (pour rappel : 34 secondes pour Knime 2.4.2 ; 51 pour R 2.13.2 ; 63 pour Weka 3.5.7, etc.).

[8] http://tutoriels-data-mining.blogspot.com/2012/01/regression-logistique-sur-les-grandes.html