Objectif

Construire un arbre de régression avec TANAGRA.

La régression consiste à produire un modèle qui permet de prédire ou d'expliquer les valeurs d'une variable à prédire continue (endogène) à partir des valeurs d'une série de variables prédictives (exogènes), continues ou discrètes. La régression linéaire multiple est certainement l'approche la plus connue, mais d'autres méthodes, moins connues en économétrie mais plus populaire dans la communauté de l'apprentissage automatique, permettent de remplir cette tâche. Dans ce didacticiel nous présentons la méthode de régression par arbres de TANAGRA.

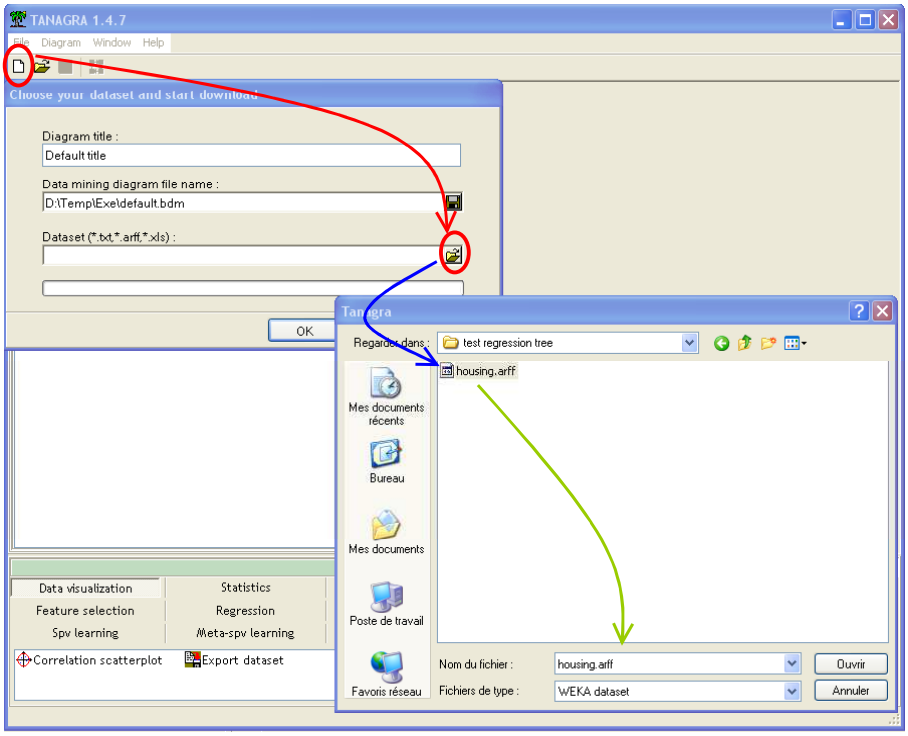
Fichier

Nous traitons le fichier HOUSING. Il s'agit de prédire la valeur médiane des habitations dans différentes zones urbaines de Boston, à partir d'une série d'indicateurs relatifs à la zone (criminalité, éloignement par rapport aux zones d'activité, etc.).

Arbre de régression avec TANAGRA

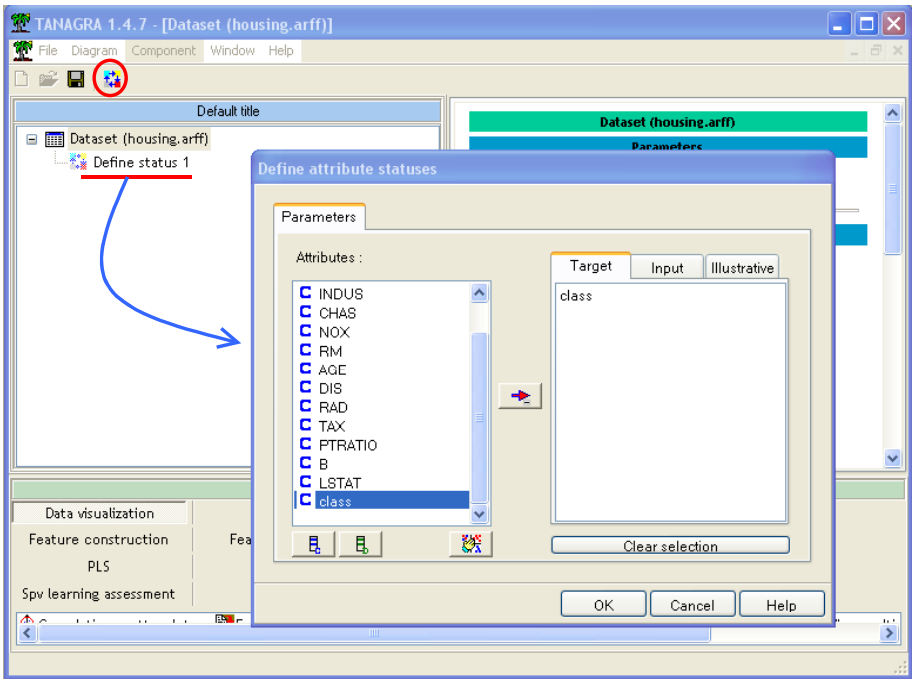
Charger les données

Nous devons dans un premier temps créer un diagramme et charger les données. Pour ce faire, nous cliquons sur le menu FILE/NEW. Nous sélectionnons le fichier HOUSING.ARFF, au format WEKA.



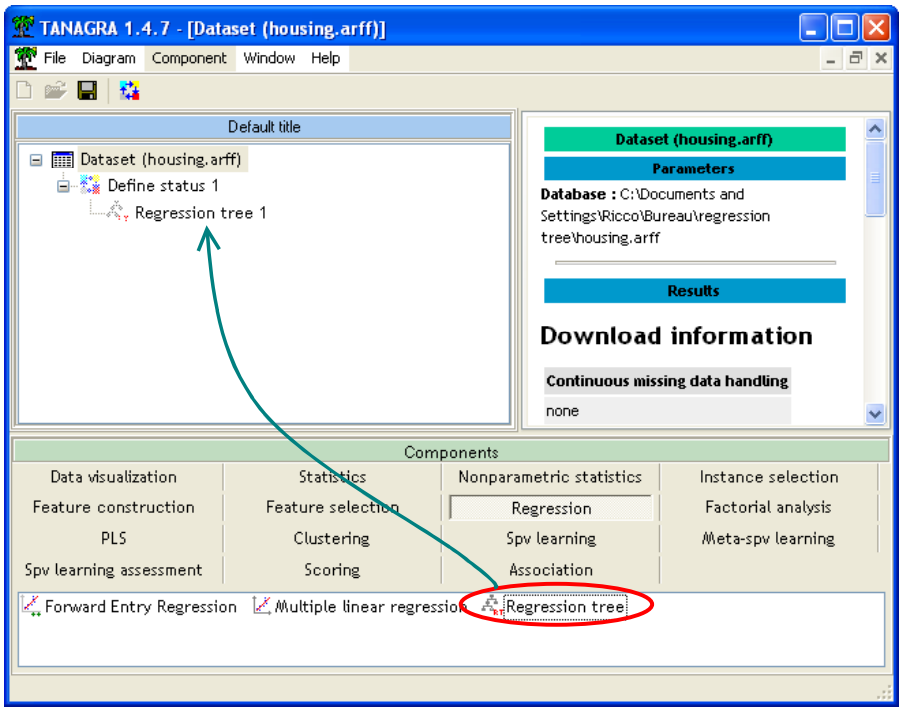
Définir le problème

L'étape suivante consiste à désigner la variable à prédire et les variables prédictives. Nous insérons le composant DEFINE STATUS dans le diagramme et nous plaçons en TARGET la variable « class », en INPUT toutes les autres variables. Contrairement à la régression linéaire, les variables peuvent être continues ou discrètes dans la régression par arbres.



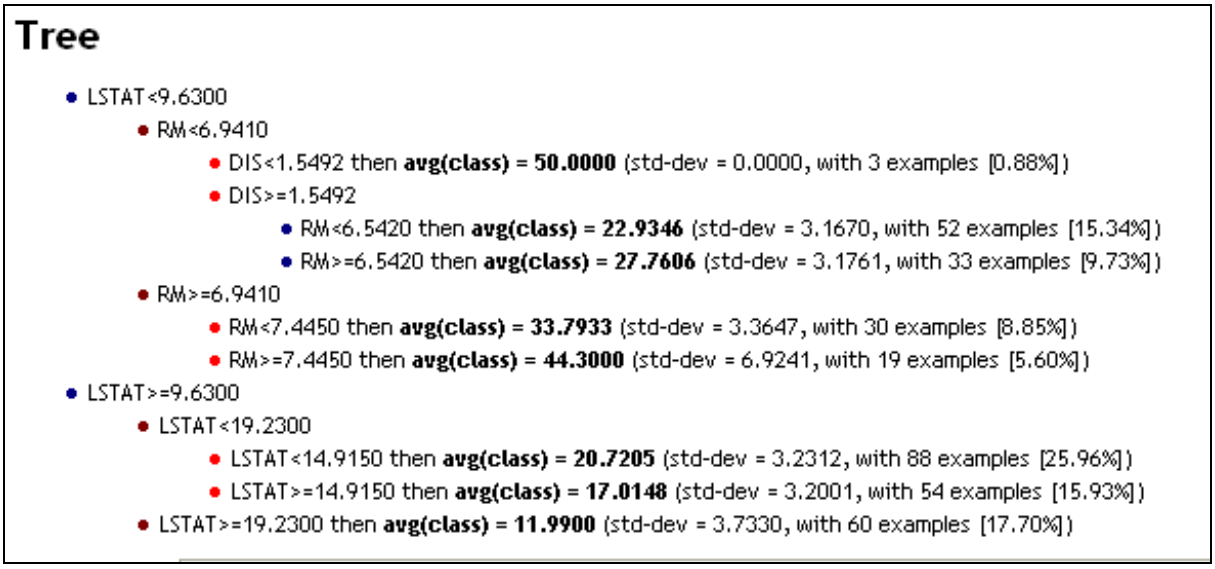
Arbres de régression

Le composant REGRESSION TREE est situé dans l'onglet REGRESSION. Nous l'insérons à la suite du composant précédent.



Nous lançons l'exécution à l'aide du menu VIEW, la fenêtre de droite recense les résultats, elle est subdivisée en plusieurs zones.

La zone **TREE** affiche l'arbre de régression. Nous constatons que la première variable de segmentation est LSTAT avec un seuil de coupure de 9.63. Les autres variables qui entrent en jeu dans la construction de l'arbre sont DIS et RM.

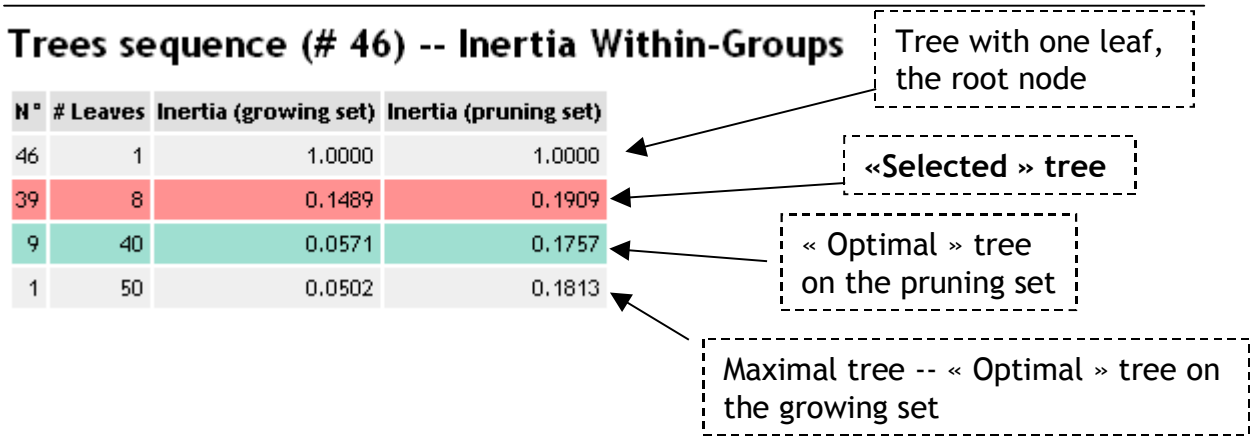


La zone **GLOBAL RESULTS** résume les caractéristiques du problème à résoudre, à noter particulièrement le R^2 qui indique la qualité de la régression : plus il est proche de 1, meilleur est l'arbre ; lorsque que le R^2 est égal à zéro, cela veut dire que l'arbre ne fait pas mieux qu'un arbre composé uniquement de sa racine, la prédiction est alors la moyenne de la variable à prédire dans la totalité de l'échantillon.

Global results

Endogenous attribute	**class**
Examples	506
R^2	0.8376

Enfin, la zone TREE SEQUENCES indique l'évolution de la réduction de l'erreur RE (RE = 1 – R^2) en fonction du nombre de feuilles de l'arbre, sur le fichier d'expansion (growing set) et le fichier d'élagage (pruning set = 33% de l'échantillon, nous pouvons le paramétrer). Généralement, le tableau est composé de 4 lignes, l'arbre réduit à la racine (1 feuille), l'arbre maximal (50 feuilles dans notre exemple), l'arbre minimisant RE (souligné en bleu, 40 feuilles dans notre exemple), et l'arbre qui a finalement été produit (souligné en rouge, 8 feuilles dans notre exemple).



Résultats détaillés – Détermination de la taille de l'arbre

La question qui vient immédiatement est « pourquoi la méthode n'a-t-elle pas tout simplement sélectionné l'arbre qui minimise RE ? »

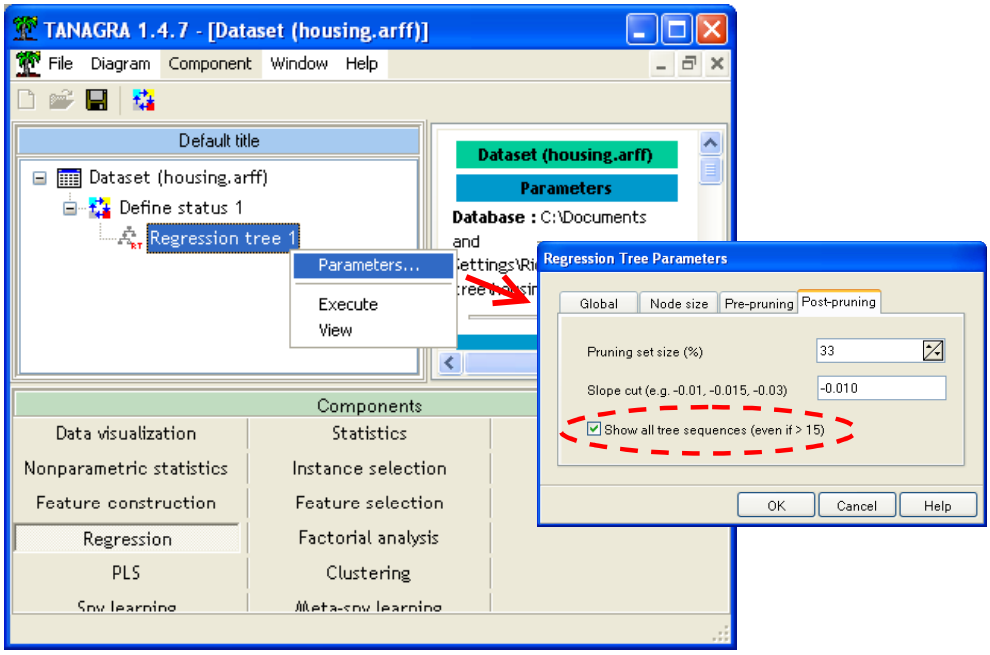
Pour répondre à la question, il nous faut tout d'abord détailler l'algorithme d'induction de l'arbre de régression. La méthode implémentée ici est directement inspirée de la méthode CART (Breiman et al., 1984). Les données sont subdivisées en deux parties.

Le premier échantillon, dit d'expansion (growing set en anglais) permet de construire l'arbre A_{max}, l'objectif est de produire des feuilles aussi pures que possible, cet arbre minimise RE sur l'échantillon d'expansion. Bien entendu, il ne faut certainement pas utiliser cet arbre pour la prédiction, il est trop spécialisé, il colle exagérément aux données d'expansion, ingérant des informations spécifiques à ce fichier.

Le deuxième échantillon, l'échantillon d'élagage (pruning set) va alors servir à réduire l'arbre. L'algorithme réduit petit à petit l'arbre initial et, à chaque étape, évalue les performances des sous-arbres candidats sur le fichier d'élagage. Nous pouvons ainsi déterminer l'arbre optimal sur cet échantillon A_{opt}. Ici également, l'arbre optimal n'est pas le modèle définitif, en effet, nous transposerions dans ce cas la dépendance à l'échantillon d'expansion à l'échantillon d'élagage, ingérant les informations spécifiques à ce dernier.

Il reste à définir l'arbre sélectionné A_{sel}, le modèle que nous utiliserons pour la prédiction par la suite. Le principe est la préférence à la simplicité. Breiman et al. (1984) proposent de calculer l'écart type SE de RE_{opt} correspondant à l'arbre optimal, puis de choisir l'arbre le plus simple dont le la réduction de l'erreur est inférieure à (RE_{opt} + 1 x SE). C'est une heuristique comme une autre, le calcul de l'écart type est assez acrobatique (Breiman et al., 1984 ; Chapitre 11), et ce seuil est tout à fait arbitraire, nous pourrions prendre comme référence 2 fois l'écart type, ou une autre valeur. C'est pour cela que nous proposons dans TANAGRA une autre approche.

Pour mieux l'appréhender, nous allons demander à TANAGRA de détailler les résultats. Pour ce faire, nous activons le menu PARAMETERS de REGRESSION TREE, et nous activons l'option SHOW ALL TREE SEQUENCES.

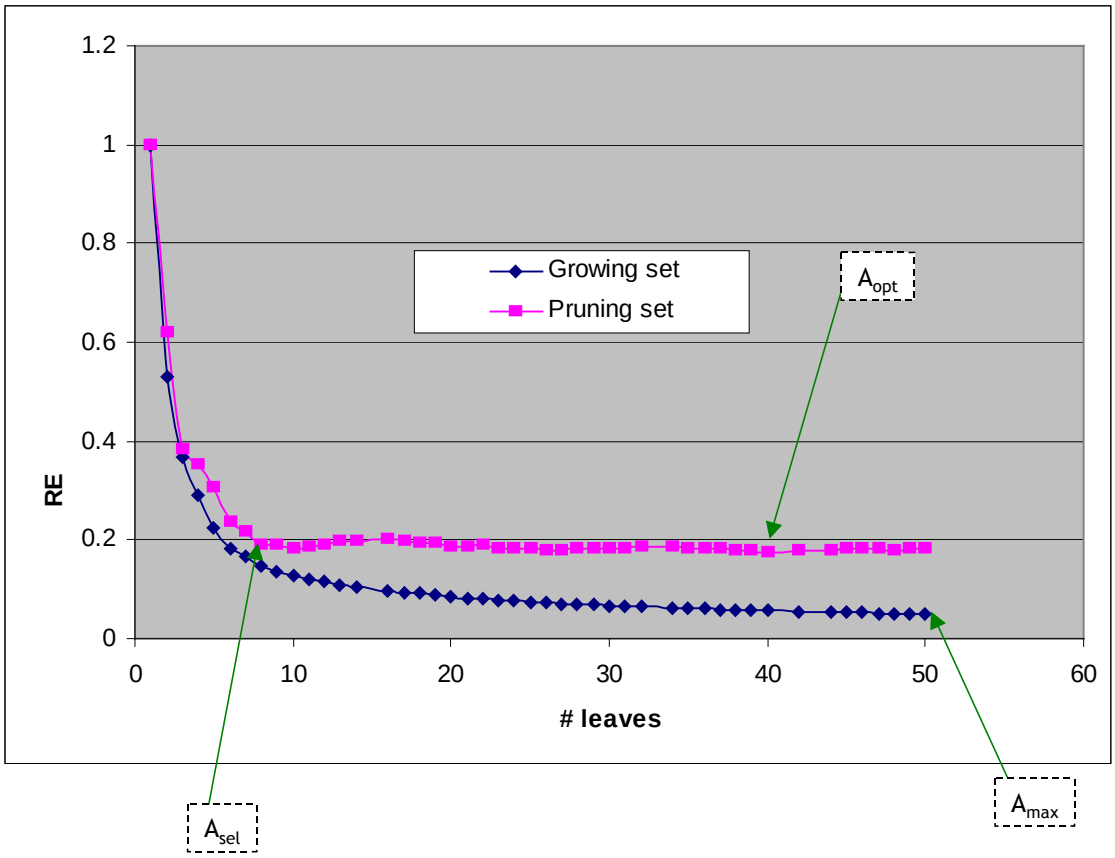


A l'exécution du composant (menu VIEW), nous observons maintenant toutes les séquences d'arbres qui ont été testés pour rechercher l'arbre de prédiction.

Trees sequence (# 46) -- Inertia Within-Groups

N°	# Leaves	Inertia (growing set)	Inertia (pruning set)
46	1	1.0000	1.0000
45	2	0.5299	0.6196
44	3	0.3676	0.3819
43	4	0.2893	0.3512
42	5	0.2256	0.3047
41	6	0.1811	0.2377
40	7	0.1648	0.2179
39	8	0.1489	0.1909
38	9	0.1369	0.1899
37	10	0.1283	0.1826
36	11	0.1213	0.1873

En copiant ces données dans un tableur, et en construisant la courbe de la réduction de l'erreur selon le nombre de feuilles des arbres, nous observons l'évolution caractéristique de l'erreur selon le nombre de feuille, sur l'échantillon d'expansion et l'échantillon d'élagage.



La stratégie que nous avons mise en place consiste donc à repérer automatiquement le « coude » de la courbe de la réduction de l'erreur RE sur l'échantillon d'élagage. Nous constatons dans cet exemple que nous pouvons ainsi réduire considérablement la taille de l'arbre en faisant un minimum de concession sur la performance, la réduction de l'erreur sur l'échantillon d'élagage.

L'heuristique pour repérer le « coude » est très simple : nous traçons une suite de ligne brisée en réalisant une régression sur 3 points. Dès que la pente de la courbe est fortement modifiée, nous suspectons la présence d'un coude. Le paramètre SLOPE CUT (pente de la droite) permet de traduire la sensibilité du dispositif, si nous la fixons à zéro, cela veut dire que nous arrêtons la recherche dès que la réduction de l'erreur n'a pas évolué (ou s'est dégradé) sur 3 points successifs.

C'est (ce n'est que) une heuristique comme une autre, plus important à mon sens sont les principes qui la sous-tendent : préférence à la simplicité et lissage dans l'exploration de l'espace des solutions. Tous ces éléments concourent à prévenir, autant que possible, la sur-dépendance aux données d'apprentissage.