MTH8302 - Analyse de régression et d'analyse de variance

chapitre 9 : modèles de partition

- Arbres de Classification et de Régression = ACR (CRT)
- Forêst Aléatoires = FA (bootstrap forest)

- **Analyse supervisée (input-output)**

  Y = variable réponse X = variables explicatives (covariables)

  but : prédiction Y continue ou classification Y catégorique

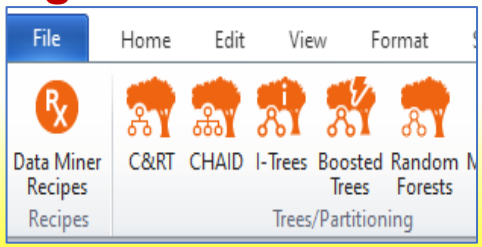
- **Méthode Arbres de Classification - données artificielles**
- **Théorie : quelques eléments**
- **Exemples** n = nb obs. p = nb variables

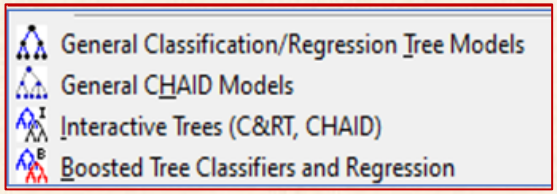
  cont = continue cat = catégorique

nom	n	p	Y cont	Y cat	X cont	X cat
Iris	150	5	0	1	4	0
Boston	1006	13	1	0	12	0
Press	540	36	0	1	26	1
Diabètes	442	11	1	2	9	1

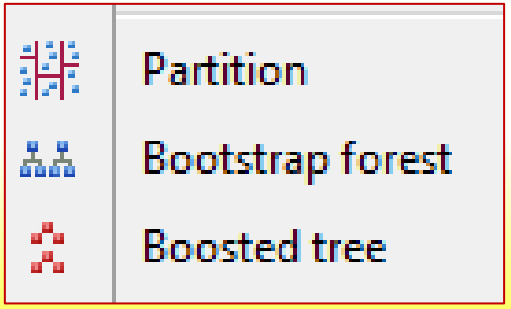
- **Références**

Logiciel STATISTICA





Logiciel JMP Pro



Classification des analyse statistiques

Problème central en analyse des données

prédire une variable Y continue ou catégorique (variable de réponse)
à l'aide de plusieurs variables X continues (et/ou catégoriques) (prédicteurs)

- **Y continue : problème de régression**
- **Y catégorique : problème de classification**

Grande variété de méthodes et algorithmes existent pour traiter ce problème

Y	MÉTHODES classiques	MODULES STATISTICA	DATA MINING (méthodes nouvelles)
continue	régression ANOVA	GRM , GLM , GLZ GDA , PLS	RNA , C&RT , CHAID Boosted Trees Random Forests MARS Machine learning
catégorique	Discriminante	DA , GDA	comme plus haut

Classification et Arbres de Régression : propriétés

- **Classification&Regression Tree (C&RT) : méthode TRES employée**
- **Avantages**
  - simplicité des résultats
  - méthode non paramétrique - non linéaire - robuste
  - permet la présence de données manquantes
- **Plusieurs algorithmes / méthodes itératives**

  STATISTICA propose 6 modules : choix à faire
- **Points importants à régler**
  - quand arrêter le processus de division?
  - éviter le sur ajustement (« overfitting »)
  - évaluation de la qualité de l'ajustement
  - émondage (« pruning ») de l'arbre
  - arbre peut devenir gros et difficile à consulter pour l'analyste
- **Validation méthodes**
  - croisée : échantillon d'entrainement / échantillon test
  - V-fold : ré échantillonnage répété des données

Classification et Arbres de Régression : propriétés

sur ajustement : overfitting

- condition où le modèle prédictif est trop spécifique
- reproduit la variation aléatoire (bruit) dans les données
- prédictions erronnées avec de nouvelles données

pour éviter le sur ajustement : validation !

- validation croisée (« cross-validation »)
- validation v-fold souvent v = 10

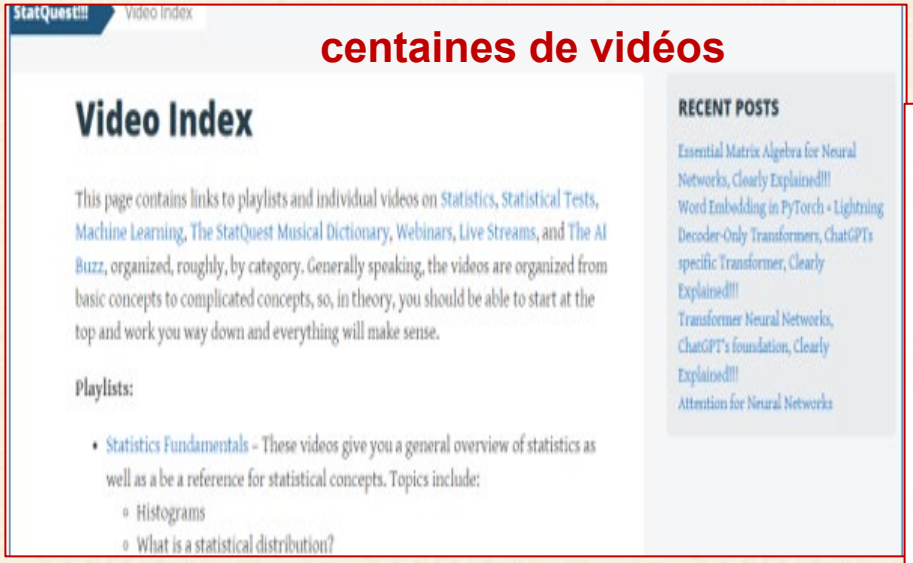
méthodes de validation

- croisée: diviser données en 2 groupes
  train (70%-80%) test (30%-20%)
  si n ≥ 100 observations
- si données insuffisantes : validation v-fold

STATQUEST : site de Joshua Starmer

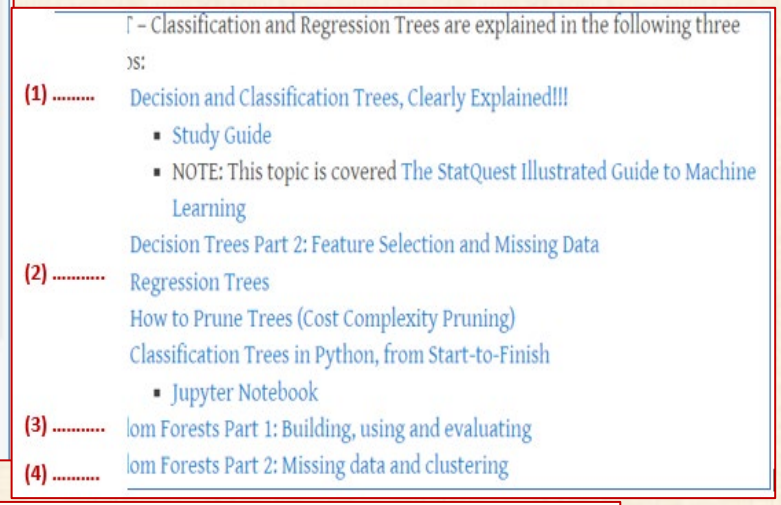
STATQUEST!!!

An epic journey through statistics and machine learning



centaines de vidéos

https://statquest.org/video-index/



(1)	**https://www.youtube.com/watch?v=_L39rN6gz7Y**	**Arbre classification : Y catégorique**
(2)	**https://www.youtube.com/watch?v=g9c66TUylZ4**	**Arbre de régression : Y continue**
(3)	**https://www.youtube.com/watch?v=J4Wdy0Wc_xQ**	**Forêts aléatoires 1**
(4)	**https://www.youtube.com/watch?v=sQ870aTKqiM**	**Forêts aléatoires 2**

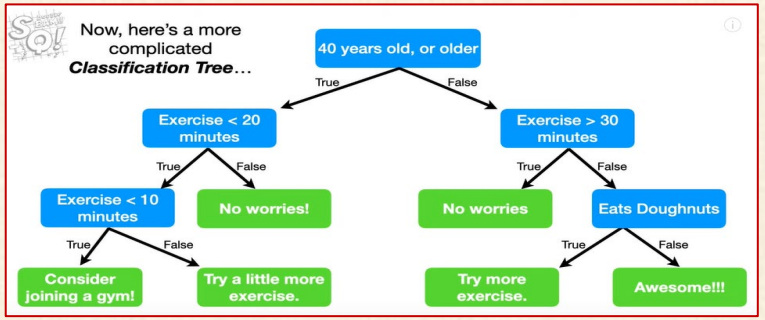
https://www.youtube.com/watch?v=_L39rN6gz7Y

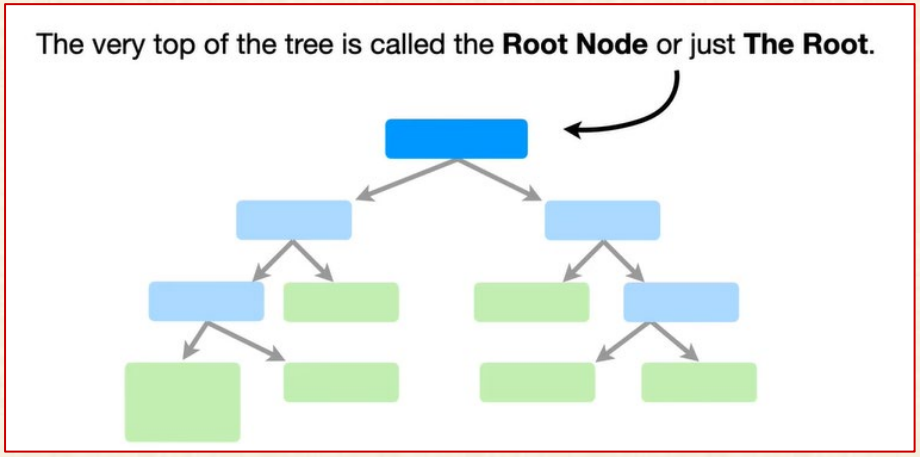
Exemple du vidéo

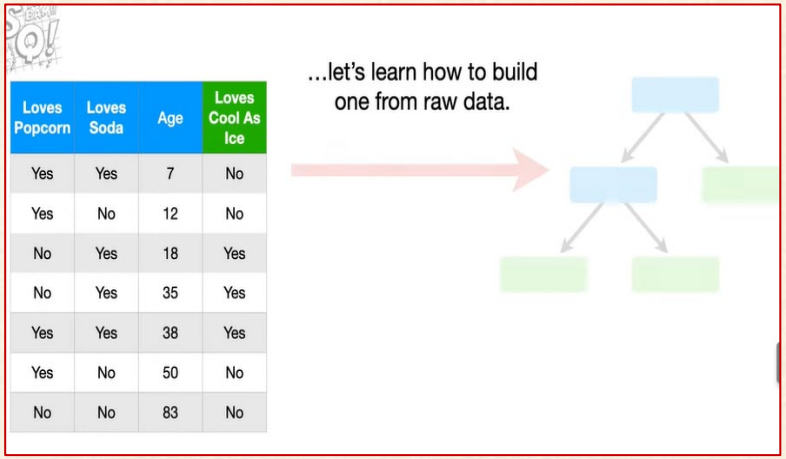
	1 ID	2 X1_loves Popcorn	3 X2_loves Soda	4 X3_Age	5 Y_Loves Cool as Ice
1	1	yes	yes	7	no
2	2	yes	no	12	no
3	3	no	yes	18	yes
4	4	no	yes	35	yes
5	5	yes	yes	38	yes
6	6	yes	no	50	no
7	7	no	no	83	no

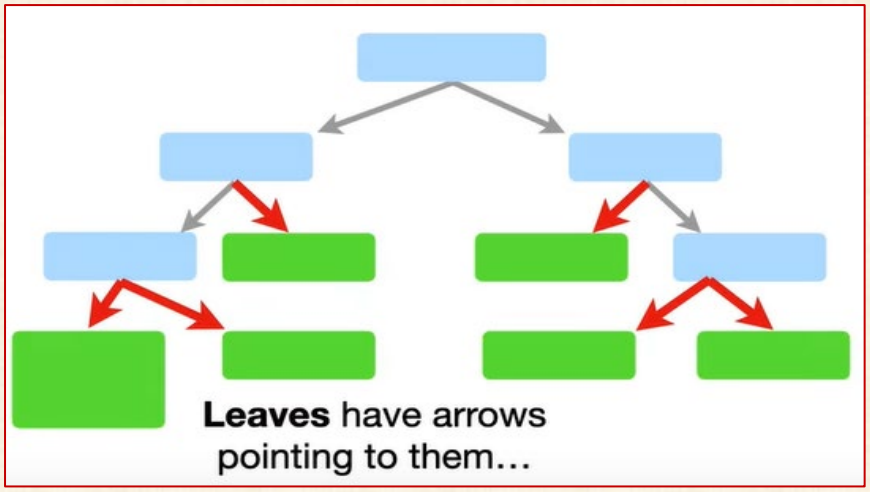
étapes principales du vidéo

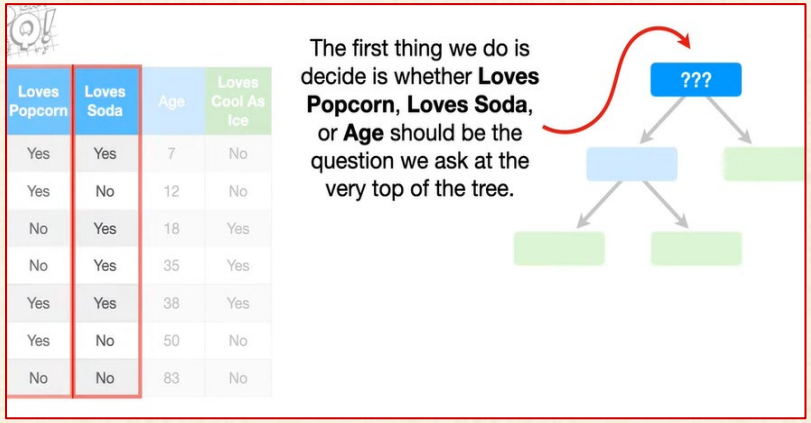
	1 ID	2 X1_loves Popcorn	3 X2_loves Soda	4 X3_Age	5 Y_Loves Cool as Ice
1	1	yes	yes	7	no
2	2	yes	no	12	no
3	3	no	yes	18	yes
4	4	no	yes	35	yes
5	5	yes	yes	38	yes
6	6	yes	no	50	no
7	7	no	no	83	no

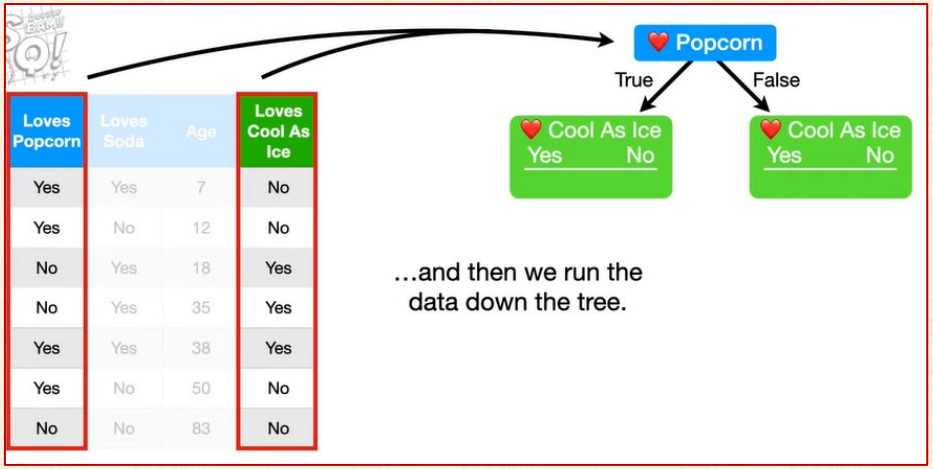


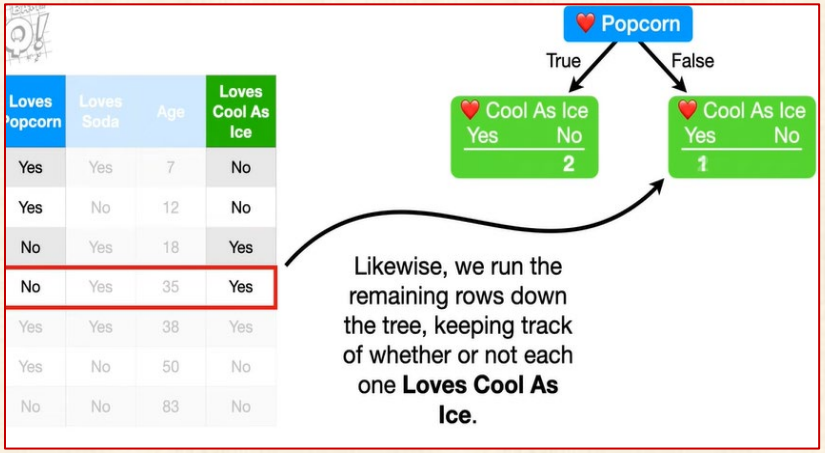


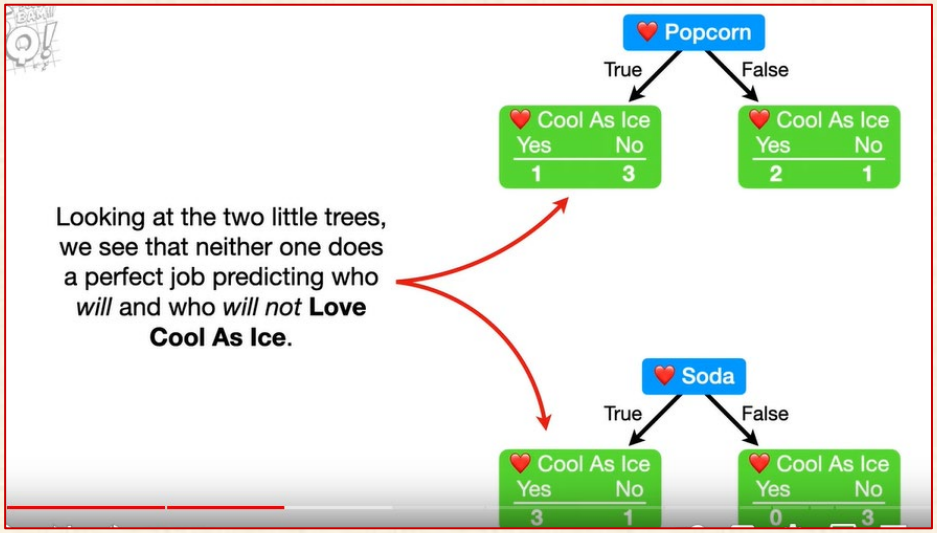


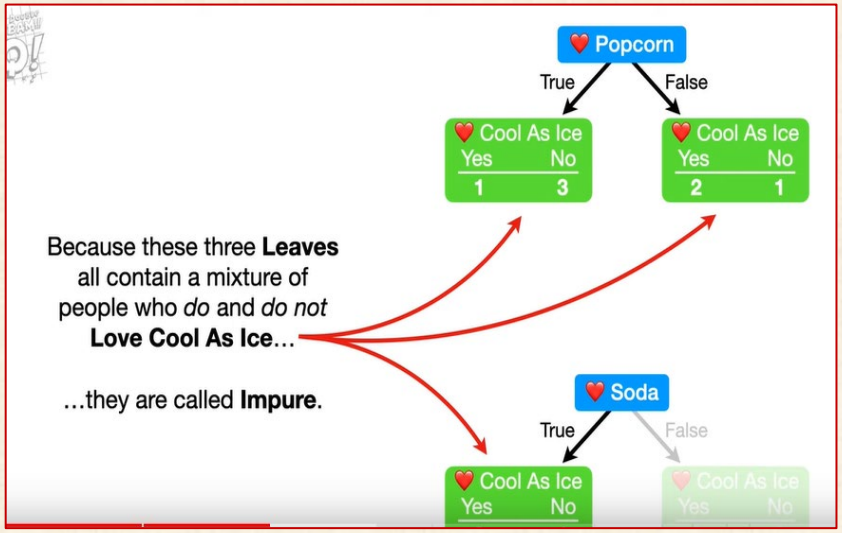












suite : [video arbres classification.pdf]

CART : Classification and Regression Tree

voir références +
Statistica Electronic manual

Basic ideas of TREES

partition the covariate space X into m regions R1, …., Rm

partition : each possible value of the vector X = (X1, …., Xp)
falls in one and only one region R1, …., Rm

model for predicting Y

$$\hat{f}(\mathbf{X}) = \sum_{j=1}^{m} d_j I(\mathbf{X} \in R_j)$$

I (X ε Rj) = 1 if the condition is true and = 0 otherwise

If X is in region Rk, then the prediction, for that X is dk

3 elements need to be specified for the model:

1. the number of regions in the partition m
2. the partition $R_1, \ldots, R_m$
3. the values $d_1, \ldots, d_m$

- Impossible to search through all possible partitions to find the best one
- Tree-based methods : solution to this problem by specifying the 3 elements
- Estimated from the data, following a recursive partitioning algorithm.

CART : Classification and Regression Tree

voir références +
Statistica Electronic manual

Original CART (Classification And Regression Trees) recursive algorithm

1. **Starting at the root node** with all the observations, the best split of the data using a single covariate is found.
   Most algorithm restrict themselves to **binary splits**

2. **The process is repeated** with the two resulting nodes, until a stopping criterion is reached. When splitting a node, only the observations in the node are used.

3. **two types of splits** depending on the covariate.
   - If a covariate X is **continuous** (or at least ordered), the possible splits take the form I (X > c) where c is a constant.
   - If the covariate is **categorical** and unordered (nominal variable), then the possible splits take the form I (X ε / {c1, …., cr }) where c1, …., cr are a subset of the possible values of X.

   If the condition is true, then the observation goes in the **left node** and if it is false, it goes in the **right node.**

   CART proceeds by **evaluating all possible splits on all covariates** and selecting the split that has the best value of a criterion.

4. **Certain constraint can be imposed** on a split to be allowed.
   For example, we might impose that a node has a minimum number of observations. Hence the search for the best split is really a search for the best split among the allowable splits.

5. Splitting stops when a **stopping criterion is reached**.
   For example, when the parent nodes has fewer than a certain number of observations, or when the parent node has reached the maximum depth allowed.

6. **Large trees** will usually **overfit the data**.
   The CART method builds a large tree, and then prunes it to get a honest tree that will capture the signal present in the data but not the noise.

7. CART uses **cost-complexity pruning** using cross-validation.

Regression Trees

The best split is the one that minimizes

$$\sum_{i \in t_L} (y_i - \bar{y}_L)^2 + \sum_{i \in t_R} (y_i - \bar{y}_R)^2$$

$i \in t_L$ $(i \in t_R)$ set on indices of the observation that are in the left (right) node

$\bar{y}_L$ $(\bar{y}_R)$ is the average of the observations in the left (right) node.

to predict an observation, we find in which terminal node it ends up and the prediction is the average of the Y in that terminal node.

CART : Classification and Regression Tree

voir références +
Statistica Electronic manual

TREES properties

- important property of CART and similar methods is that they are invariant to monotone transformations of the predictor variables.
- transformed variable or the original one will produce the same tree.
- we do not need to find a transformation (e.g. the log) of the predictor that fits the data well.
- automatically detect certain types of interactions between the predictors.
- conditions leading to a terminal node are interactions
  (products) of binary variables of the type I (X > c) or I (X ε {c1, …, cr})
- If the root node is the depth 0 node,
  - at depth 1 : main effects only
  - at depth 2 :order 2 interactions
  - at depth 3 : order 3 interactions etc.
- can handle any type of covariates
- can model any types of target Y
- can detect certain types of interactions automatically
- scale well to large sample sizes
- small trees are easy to interpret
- a single tree can be beaten by other methods for prediction performance
- Interpretability of trees is quickly lost when the tree is large
- combining many trees can often improve drastically the prediction performance of a single tree
- ensemble methods are often among the best performer in terms of prediction accuracy but difficult to interpret
- combining trees : random forests and boosting

CART : Classification and Regression Tree

voir références +
Statistica Electronic manual

Basic Random Forest Algorithm

Assume p covariates. Select the number of trees B, and the number of covariates $p_0 < p$, to select at random at a node to find the best split.

For b = 1 to B

1. Create a bootstrap sample from the original data.
2. Build a tree with the bootstrap sample (large trees are usually built and no pruning is performed).

At each node, select at random p_0 out of the p covariates and find the best split with these covariates only.

The subset of covariates can vary from node to node. Let

$\hat{T}_b(x)$ be this tree.

The final prediction model is obtained by aggregating (averaging in some way) the predictions of all trees in the forest.

Averaging depends on the type of the target.

Y continuous target, each tree returns a predicted value (average of the observations in the terminal node).

Final prediction model = average of these predictions

$$\hat{T}(x) = \frac{1}{B}\sum_{b=1}^{B} \hat{T}_b(x)$$

where $\hat{T}_b(x)$ is the prediction form the b^{th} tree.

Y categorical target, there are a few ways to combine the trees.

The first one is to get the prediction from each tree and define the forest prediction as the majority vote (the class with the most votes among all trees). The forest prediction is then the class with the highest averaged proportion.

voir références +
Statistica Electronic manual

RF : Random Forest

There are two sources of randomness in the RF algorithm.

first one : use a new bootstrap sample for each tree.

second one : select a subset of covariates at each node.

second source of randomness has another benefit.

= speeds up the computations since less splits need to be evaluated.

= reduce the correlation between the trees and smooth the final learner

by diminishing the impact of the natural instability of a single tree.

default values

$p0 = p / 3$ regression forest

$p0 = (p)^{0.5}$ classification forest

Bagging (Breiman, 1996): ancestor of RF particular case of RF

use all covariates p to find the best split at each node

Classification et arbres de régression: quelques formules (ref : documentation Statistica)

Prior Probabilities, the Gini Measure of Node Impurity, and Misclassification Cost

In *Classification and Regression Trees* (GC&RT, Interactive Trees), the default option for the goodness-of-fit measure in classification problems is Gini; further, options are provided for specifying the prior classification probabilities (or Priors). The selection of prior probabilities can affect the splits that are chosen for the final tree, and greatly affect the accuracy of the final *C&RT* model for predicting particular classes.

Prior Probabilities and the Gini Measure of Node Impurity

According to Breiman, Friedman, Olshen, & Stone (1984), the Gini measure of node impurity at node (which *STATISTICA* uses by default in *GC&RT* and, therefore, *Boosted Trees*) is defined to be

$$\text{impurity}(t) = \sum_{i \neq j} p(i \mid t)\, p(j \mid t)$$

$$p(j \mid t) = \frac{p(j,t)}{p(t)}$$

$$p(j,t) = \frac{\pi(j) N_j(t)}{N_j}$$

$p(j \mid t)$ is the estimated probability that an observation belongs to group j given that it is in node t,
$p(j, t)$ is the estimated probability that an observation is in group j and at node t ,
$p(t)$ is the estimated probability that an observation is at node t, xx / aa
$\pi(j)$ is the prior probability for group j,
$N_j(t)$ is the number of group j members at node t, and N_j is the size of group j.

Therefore, the prior probabilities play a role in every Gini Measure computation at every node. However, when the prior probabilities are estimated from the data,

$$\pi(j) = \frac{N_j}{N} \quad \Rightarrow \quad p(j \mid t) = \frac{N_j(t)}{N(t)}$$

This fact can cause higher misclassification rates in under-represented groups (see *Prior Probabilities and Misclassification Costs* below).

Prior Probabilities and Misclassification Costs

When non-uniform misclassification costs are specified for the *GC&RT* analysis, the Gini measure is modified to account for these costs .

$$\text{impurity}(t) = \sum_{i \neq j} C(i \mid j)\, p(i \mid t)\, p(j \mid t)$$

where $C(i \mid j)$ is the cost of misclassifying an observation in class j as belonging to class i. This feature enables the user to effectively penalize certain types of misclassifications in the analysis.

However, as noted above in *Prior Probabilities and the Gini Measure of Node Impurity*, $p(j \mid t)$ is a function of $p(j)$, the prior probability for class j. Therefore, for a given $C(i \mid j)$ and $p(j)$, one can find $C'(i \mid j)$ and $p'(j)$, such that

$$C(i \mid j)\pi(j) = C'(i \mid j)\pi'(j)$$

Consequently, if $C'(i \mid j)$ is taken to be unity for all i π'(j) j and $p'(j)$ can be found, such that the above relationship is satisfied, then this adjustment of the prior probabilities can have the same net effect as the specification of non-uniform misclassification costs. This property can be readily observed in classification problems where one of the classes is underrepresented in the data. In this case, for uniform misclassification costs, prior probabilities that are estimated from the sample proportions will produce a model that tends to under-perform with respect to the underrepresented class. However, if one increases the prior probability for the underrepresented class, then the model will tend to do a better job of classifying cases in this group.

Classification et arbres de régression: quelques formules (ref : documentation Statistica)

classification

$$R(d) = \frac{1}{N}\sum_{i=1}^{N} X(d(x_n) \neq j_n)$$

$X = 1$, if the statement $X(d(x_n) \neq j_n)$ is true

$X = 0$, if the statement $X(d(x_n) \neq j_n)$ is false

and $d(x)$ is the classifier.

Let the learning sample Z of size N be partitione

$$R^{ts}(d) = \frac{1}{N_2}\sum_{(x_n, j_n) \in Z_2} X(d(x_n) \neq j_n)$$

$$R^{ts}(d^{(v)}) = \frac{1}{N_v}\sum_{(x_n, j_n) \in Z_v} X(d^{(v)}(x_n) \neq j_n)$$

where $d^{(v)}(x)$ is computed from the sub sample $Z - Z_v$.

régression

the predictor of the continuous depende

$$R(d) = \frac{1}{N}\sum_{i=1}^{N} (y_i - d(x_i))^2$$

$$R^{ts}(d) = \frac{1}{N_2}\sum_{(x_i, y_i) \in Z_2} (y_i - d(x_i))^2$$

$$R^{CV}(d) = \frac{1}{N_v}\sum_{v}\sum_{(x_n, y_n) \in Z_v} (y_i - d^{(v)}(x_n))^2$$

$$g(t) = \sum_{j \neq i} p(j/t)\, p(i/t)$$

$$= \sum_{j \neq i} C(i/j)\, p(j/t)\, p(i/t)$$

$$R(t) = \frac{1}{N_w(t)}\sum_{i \in t} w_i f_i (y_i - \bar{y}(t))^2$$

CART : Classification and Regression Tree

Decision Tree and Bootstrap Forest

- If the response is categorical, then it is fitting the probabilities estimated for the response levels, minimizing the residual log-likelihood chi-square [2*entropy].
- If the response is continuous, then the platform fits means, minimizing the sum of squared errors.
- If the factor is continuous, then the partition is done according to a splitting "cut" value for the factor.
- If the factor is categorical, then it divides the *X* categories into two groups of levels and considers all possible groupings into two levels.

Splitting Criterion

Node splitting is based on the LogWorth statistic, which is reported in Candidate reports for nodes.
LogWorth is calculated as follows: LogWorth = - log10(p_value)

where the adjusted *p*-value is calculated in a complex manner that takes into account the number of different ways splits can occur. This calculation is very fair compared to the unadjusted *p*-value, which favors *X*s with many levels, and the Bonferroni *p*-value, which favors *X*s with small numbers of levels.

For continuous responses, the Sum of Squares (SS) is reported in node reports. This is the change in the error sum-of-squares due to the split. A candidate SS that has been chosen is:

SStest = SSparent - (SSright + SSleft) where SS in a node is just $s^2(n - 1)$.

Also reported for continuous responses is the Difference statistic. This is the difference between the predicted values for the two child nodes of a parent node.
For categorical responses, the G2 (likelihood-ratio chi-square) appears in the report. This is actually twice the [natural log] entropy or twice the change in the entropy. Entropy is Σ -log(*p*) for each observation, where *p* is the probability attributed to the response that occurred. A candidate *G*2 that has been chosen is:

*G*2 test = *G*2 parent - (*G*2 left + *G*2 right)

Partition actually has two rates; one used for training that is the usual ratio of count to total, and another that is slightly biased away from zero. By never having attributed probabilities of zero, this allows logs of probabilities to be calculated on validation or excluded sets of data, used in Entropy R-Square.

CART : Classification and Regression Tree

Predicted Probabilities in Decision Tree and Bootstrap Forest

The predicted probabilities for the Decision Tree and Bootstrap Forest methods are calculated as described below by the Prob statistic.

For categorical responses in Decision Tree, the Show Split Prob command shows the following statistics:

Rate The proportion of observations at the node for each response level.

Prob The predicted probability for that node of the tree.
The method for calculating Prob for the ith response level at a given node is as follows:

$$\text{Prob}_i = \frac{n_i + \text{prior}_i}{\sum(n_i + \text{prior}_i)}$$

where the summation is across all response levels; ni is the number of observations at the node for the ith response level; and priori is the prior probability for the ith response level, calculated as follows:

$$\text{prior}_i = \lambda p_i + (1-\lambda)P_i$$

where pi is the priori from the parent node, P_i is the Prob_i from the parent node, and λ is a weighting factor currently set at 0.9.

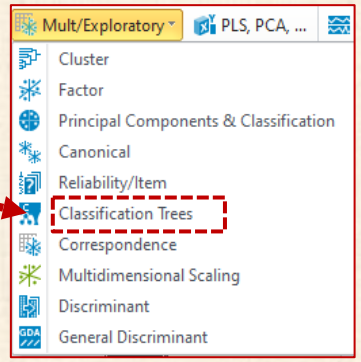
The method for calculating Prob assures that the predicted probabilities are always nonzero.

MÉTHODES & OUTILS

- Data Miner Recipes
- General Classification/Regression Tree Models
- General CHAID Models
- Interactive Trees (C&RT, CHAID)
- Boosted Tree Classifiers and Regression
- Random Forests for Regression and Classification
- Generalized Additive Models
- MARSplines (Multivariate Adaptive Regression Splines)
- Generalized EM & k-Means Cluster Analysis
- Automated Neural Networks
- Machine Learning (Bayesian, Support Vectors, K-Nearest)
- Independent Components Analysis
- Text & Document Mining
- Web Crawling, Document Retrieval
- Association Rules
- Sequence, Association, and Link Analysis
- Rapid Deployment of Predictive Models (PMML)
- Goodness of Fit, Classification, Prediction
- Feature Selection and Variable Screening
- Optimal Binning for Predictive Data Mining
- Data Mining - Workspaces
- Process Optimization

CRT Classification Regression Tree

procédure suite *Statistics*



CRT Classification Regression Tree

4 procédures suite *Data Mining*

FA Forêsts Aléatoires

GAM Gen. Additive Models

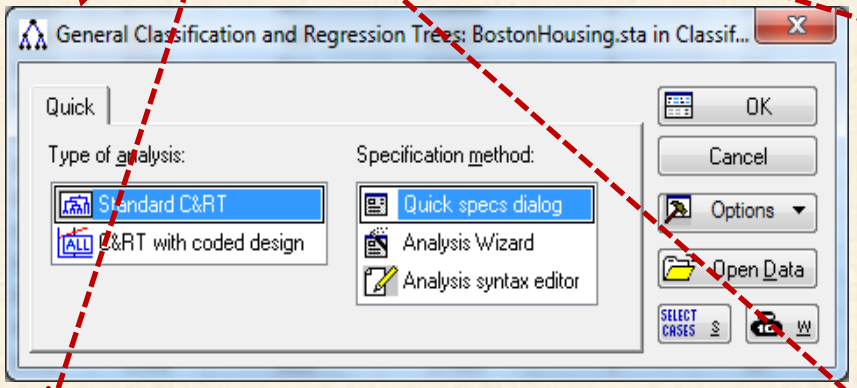
MARS Multivariate Adaptive Regression Splines

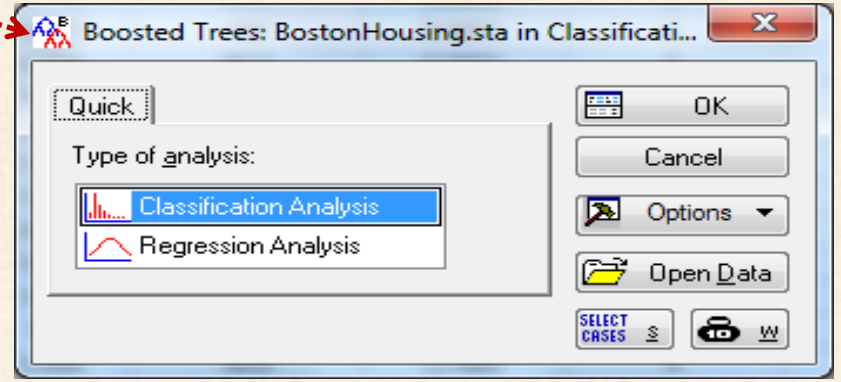
ANN Artificial Neural Networok

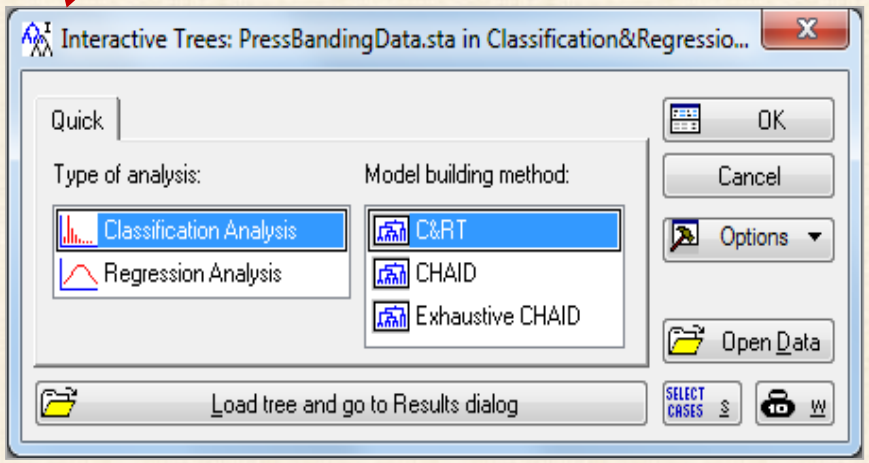
autres procédures = analyse non supervisée

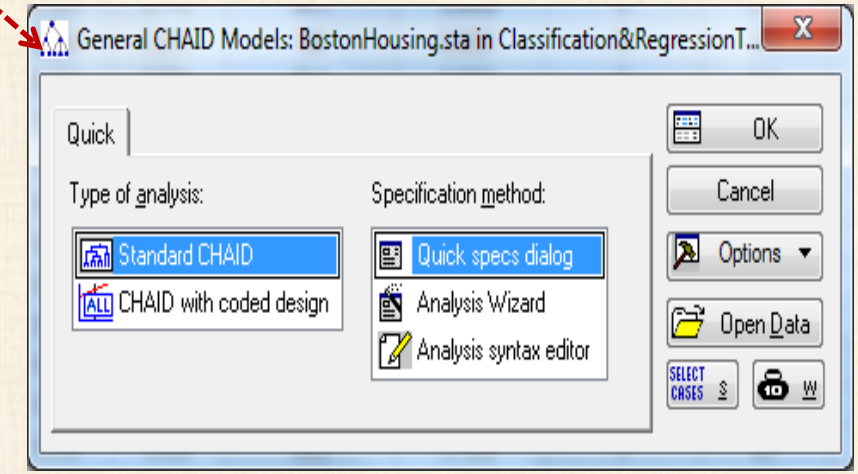
Procédures de STATISTICA : arbres de classification

Menu *Data Mining* : 4 procédures pour produire des arbres





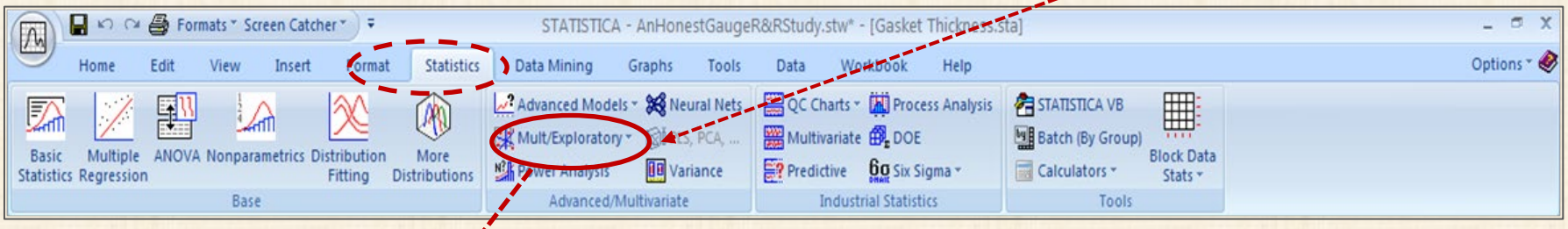


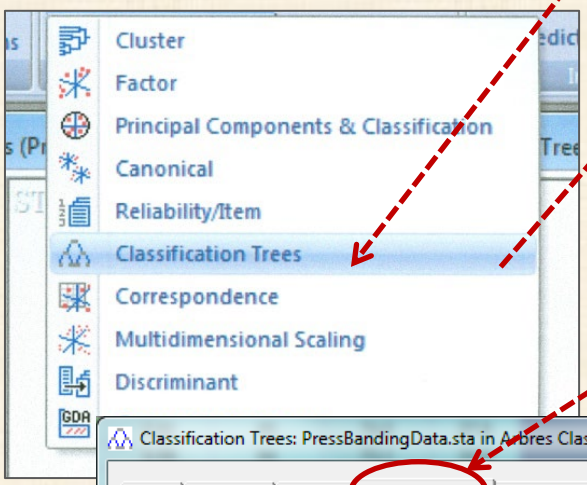


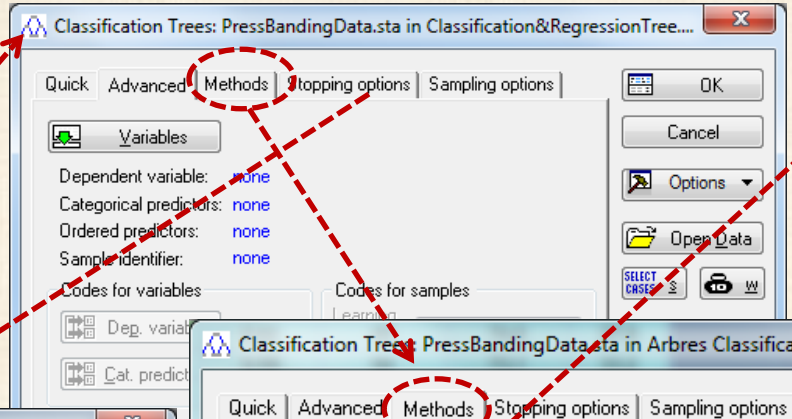
Procédures de STATISTICA : arbres de classification

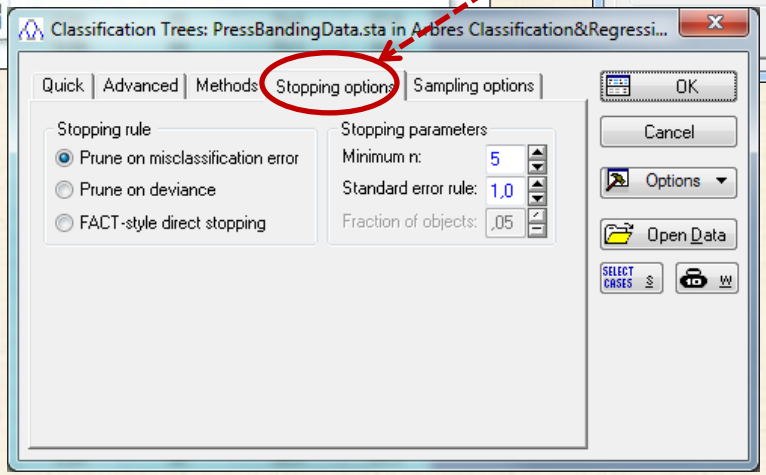
Menu *Statistics* : procédure pour produire des arbres
Mult / Exploratory … Classification Trees

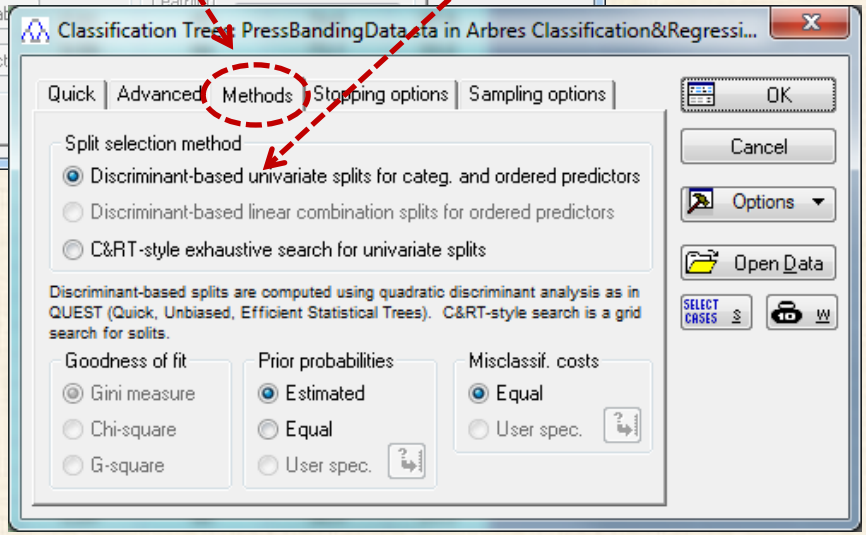
procédure recommandée











base = analyse discriminante

STATISTICA : 35 vidéos – mise en œuvre – pas de théorie

5 vidéos sur arbres classification - partitionnement - CART

Session 12 : Méthodes de partitionnement récursif
https://www.youtube.com/watch?v=7EHiWvAsc6s
Session 13 : Classification par l'arbre C&RT
https://www.youtube.com/watch?v=t-f0qKq4Ecs
Session 14 : Classification par l'arbre CHAID
https://www.youtube.com/watch?v=z-I5_dWroZo
Session 15 : Classification par la Technique du Boosting
https://www.youtube.com/watch?v=-CGg6QPcci0
Session 16 : Classification par les Forêts Aléatoires
https://www.youtube.com/watch?v=f97tbm0dAkI

Classification et Arbres de régression : exemple avec Statistica – data IRIS

IRIS data set — **IIRIS data Set**

Source : Fisher, R.A. (1936) "*The use of multiple measurements in taxonomic problems*" Annual Eugenics, 7, Part II, 179-188

Data Information

This is perhaps the best known database to be found in the pattern recognition literature.
Fisher's paper is a classic in the field and is one of the most referenced frequently to this day.
The data set contains 3 classes of 50 instances each, where each class refers to a type of iris plant.
One class (Setosa) is linearly separable from Versicolor and Virginica.
Versicolor and Virginica are NOT linearly separable from each other.
Predicted attribute = class of iris plant = X_class.

Correction of some data values

This data on the web site <https://archive.ics.uci.edu/ml/index.php> differs from the data presented in Fisher's article. This correction was identified by Steve Chadwick.
The 35th sample should be: 4.9, 3.1, 1.5, 0.2,"Iris_setosa" where the error is in the fourth feature.
The 38th sample: 4.9, 3.6, 1.4, 0.1,"Iris_setosa" where the errors are in the second and third features.
Remarque : le fichier Statistica que j'ai créé Iris.sta contient les valeurs corrigées.

Variables description

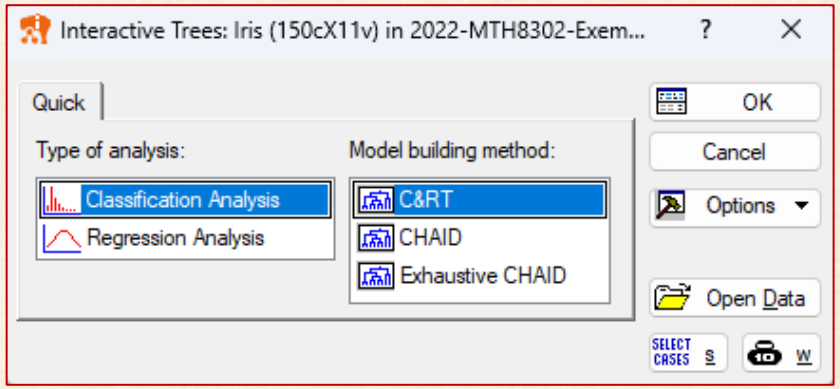
Y1_Sepal_Length (cm)
Y2_Sepal_Width (cm)
Y3_Petal_Length (cm)
Y4_Petal_Width (cm)
Y_class : Setosa / Versicolor / Virginica
Indicator variables (0/1) : X1_setosa X2_versicolor X3_virginica

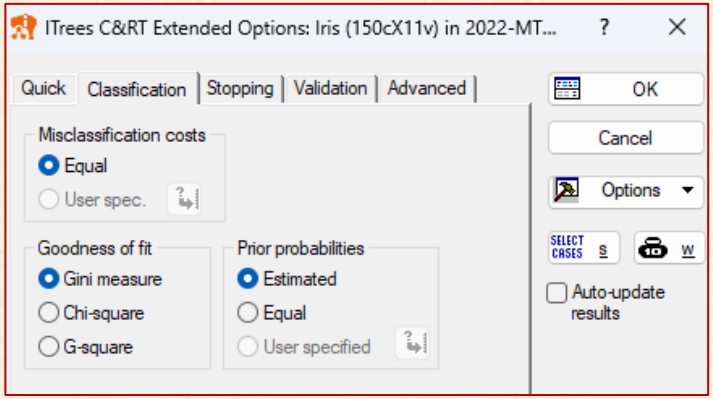
1 ID	2 Y1_Sepal_Length	3 Y2_Sepal_Width	4 Y3_Petal_Length	5 Y4_Petal_Width	6 Y_Class	7 info	8 X1_setosa	9 X2_versicolor	10 X3_virginica	11 NNSET
1	5,1	3,5	1,4	0,2	Setosa	les variables	1	0	0	Select
2	4,9	3,0	1,4	0,2	Setosa	X1 X2 X3	1	0	0	Select
3	4,7	3,2	1,3	0,2	Setosa	constituent un codage	1	0	0	Train
4	4,6	3,1	1,5	0,2	Setosa	disjonctif complet	1	0	0	Select
5	5,0	3,6	1,4	0,2	Setosa	de la variable Y_Class	1	0	0	Train
6	5,4	3,9	1,7	0,4	Setosa		1	0	0	Train
7	4,6	3,4	1,4	0,3	Setosa	Ce sont les variables	1	0	0	Train
8	5,0	3,4	1,5	0,2	Setosa	de sortie du processus	1	0	0	Select
9	4,4	2,9	1,4	0,2	Setosa	de classification	1	0	0	Select
10	4,9	3,1	1,5	0,1	Setosa	des fleurs IRIS	1	0	0	Select
11	5,4	3,7	1,5	0,2	Setosa		1	0	0	Train

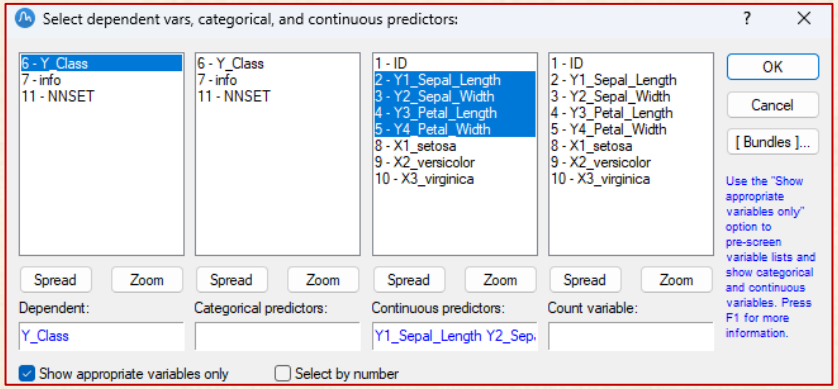
51	51	7,0	3,2	4,7	1,4	Versicolor		0	1	0	Train
52	52	6,4	3,2	4,5	1,5	Versicolor		0	1	0	Select
53	53	6,9	3,1	4,9	1,5	Versicolor		0	1	0	Train
54	54	5,5	2,3	4,0	1,3	Versicolor		0	1	0	Select
55	55	6,5	2,8	4,6	1,5	Versicolor		0	1	0	Train
56	56	5,7	2,8	4,5	1,3	Versicolor		0	1	0	Train

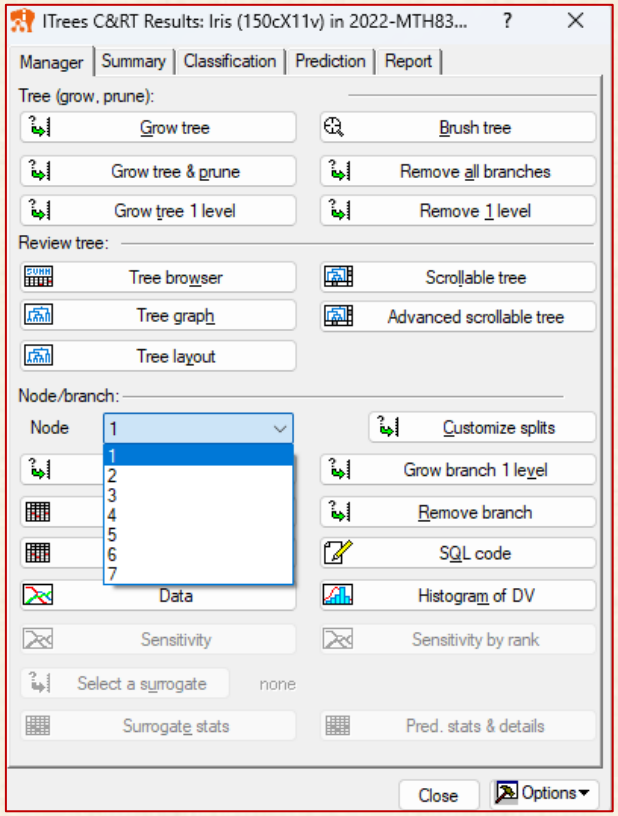
101	101	6,3	3,3	6,0	2,5	Virginica		0	0	1	Select
102	102	5,8	2,7	5,1	1,9	Virginica		0	0	1	Select
103	103	7,1	3,0	5,9	2,1	Virginica		0	0	1	Select
104	104	6,3	2,9	5,6	1,8	Virginica		0	0	1	Train
105	105	6,5	3,0	5,8	2,2	Virginica		0	0	1	Select
106	106	7,6	3,0	6,6	2,1	Virginica		0	0	1	Train
107	107	4,9	2,5	4,5	1,7	Virginica		0	0	1	Select

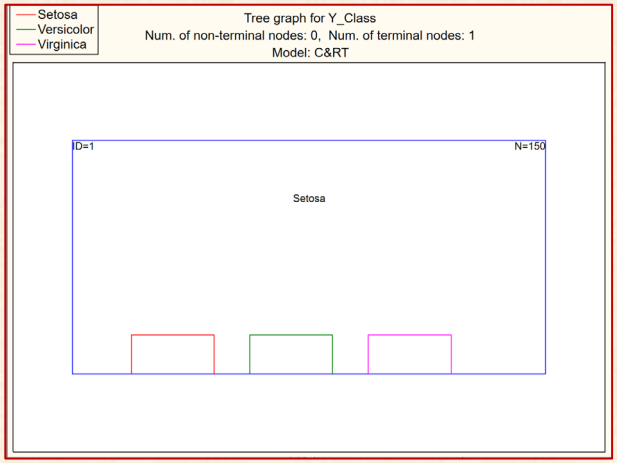
Classification et Arbres de régression : exemple avec Statistica – data IRIS



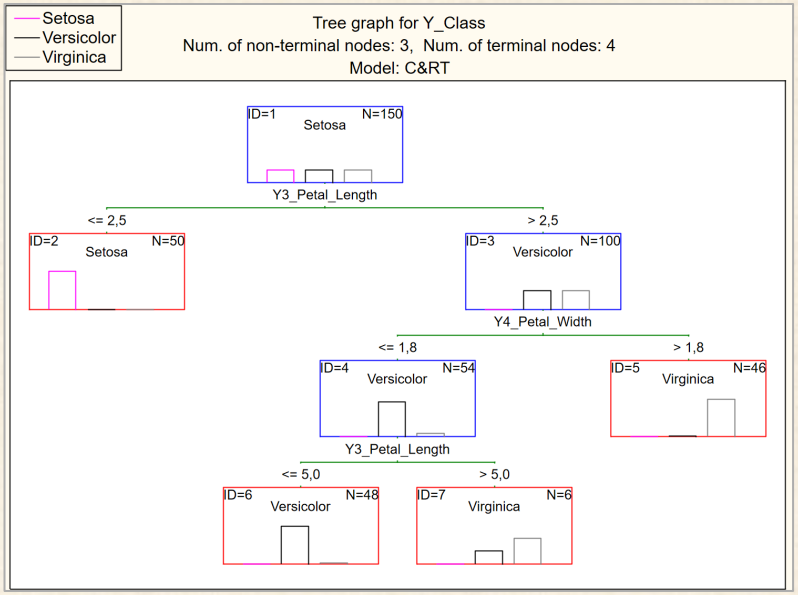




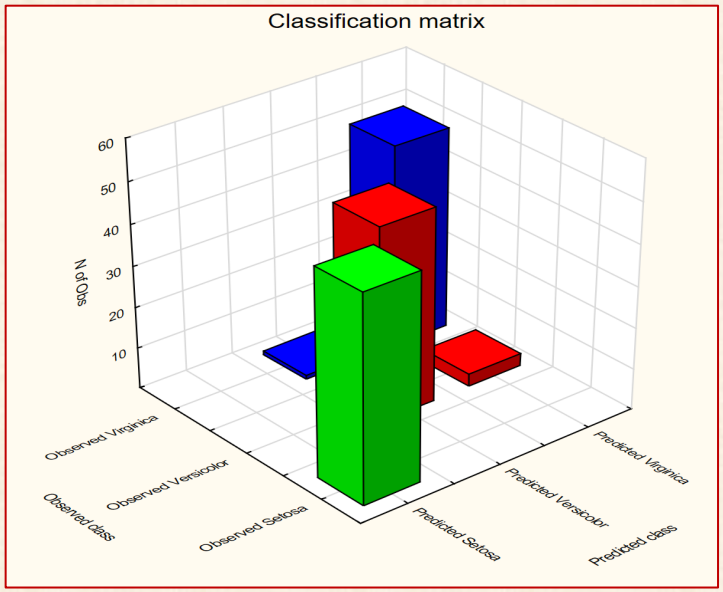


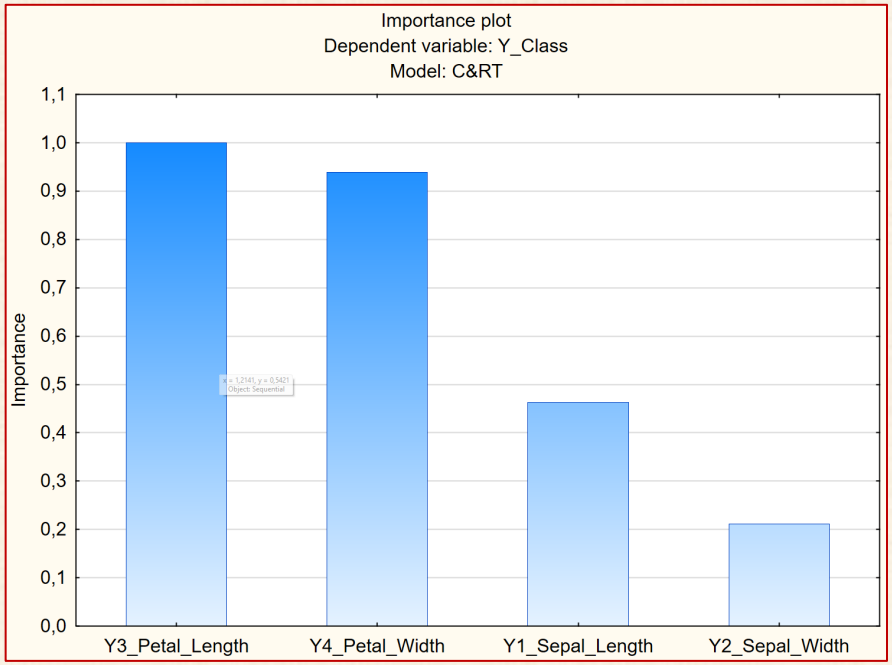


Classification et Arbres de régression : exemple avec Statistica – data IRIS



	Observed	Predicted Setosa	Predicted Versicolor	Predicted Virginica	Row Total
Number	Setosa	50			50
Column Percentage		100.00%	0.00%	0.00%	
Row Percentage		100.00%	0.00%	0.00%	
Total Percentage		33.33%	0.00%	0.00%	33.33%
Number	Versicolor		49	1	50
Column Percentage		0.00%	90.74%	2.17%	
Row Percentage		0.00%	98.00%	2.00%	
Total Percentage		0.00%	32.67%	0.67%	33.33%
Number	Virginica		5	45	50
Column Percentage		0.00%	9.26%	97.83%	
Row Percentage		0.00%	10.00%	90.00%	
Total Percentage		0.00%	3.33%	30.00%	33.33%
Count	All Groups	50	54	46	150
Total Percent		33.33%	36.00%	30.67%	





Classification et Arbres de régression : exemple avec JMP – data IRIS

Iris (150cX11v) - JMP Pro

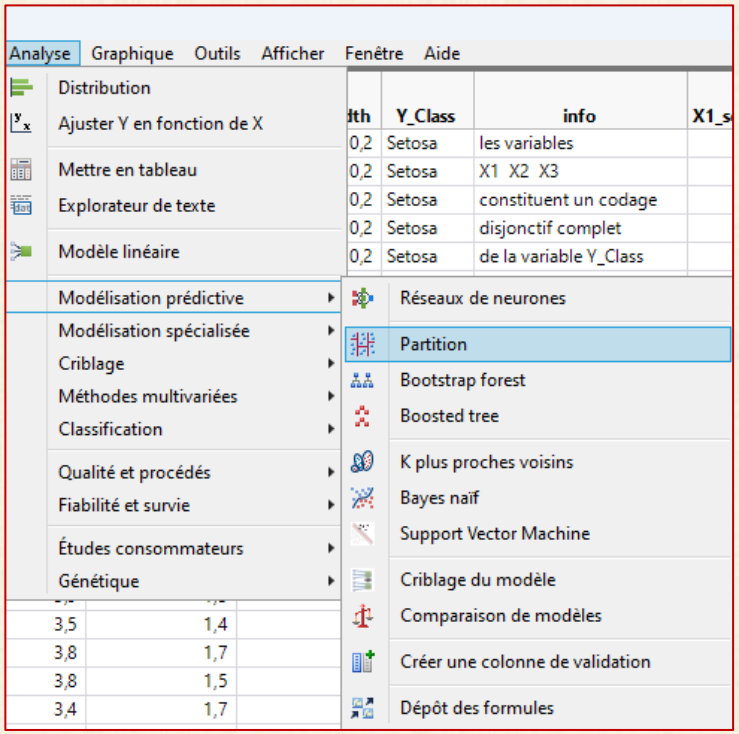
Fichier Édition Tables de données Lignes Colonnes Plan d'expérience Analyse Graphique Outils Afficher Fenêtre Aide

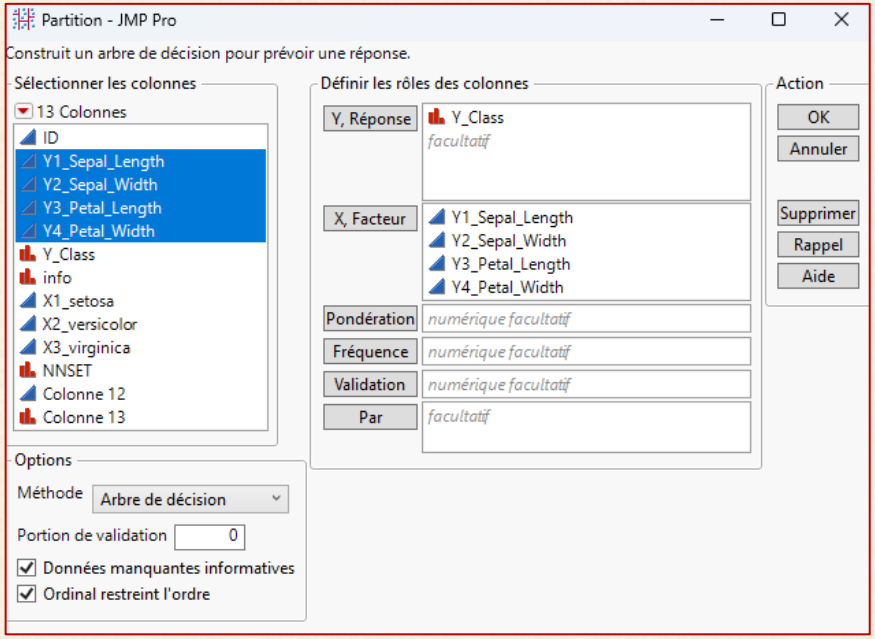
Iris (150cX11v)
Source

Colonnes (13/1)

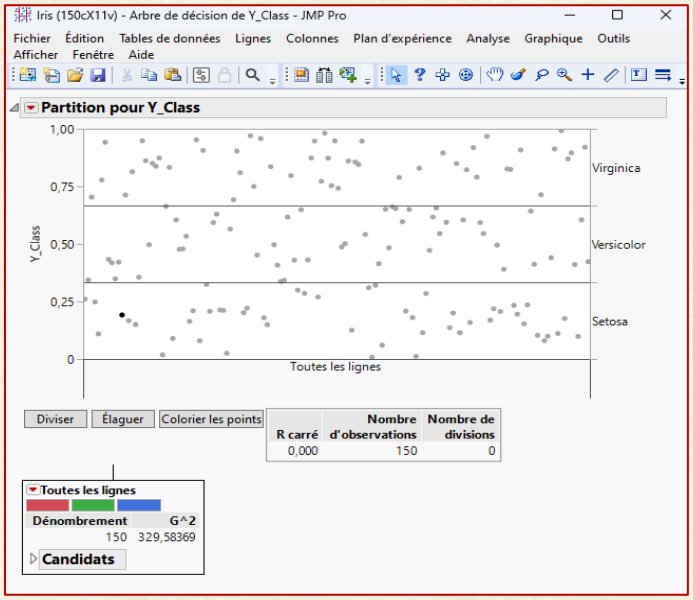
ID
Y1_Sepal_Length
Y2_Sepal_Width
Y3_Petal_Length
Y4_Petal_Width
Y_Class
info
X1_setosa
X2_versicolor
X3_virginica
NNSET
Colonne 12
Colonne 13

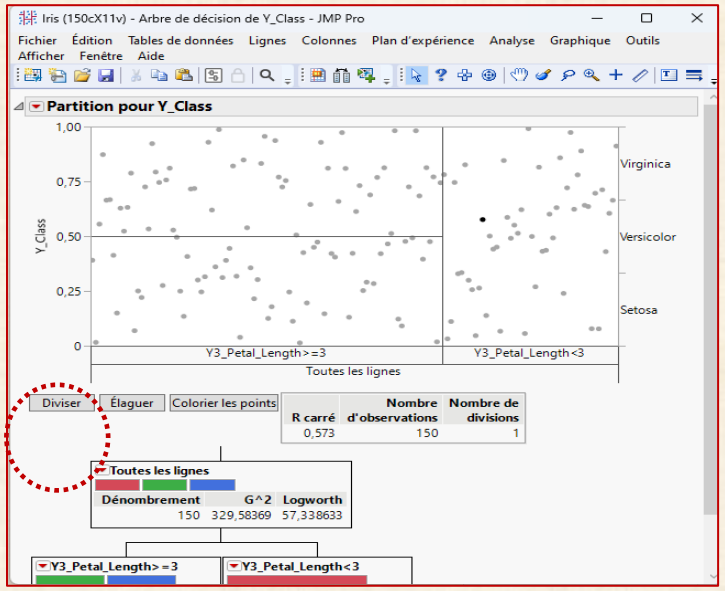
	ID	Y1_Sepal_Length	Y2_Sepal_Width	Y3_Petal_Length	Y4_Petal_Width	Y_Class	info	X1_setosa	X2_versicolor	X3_virginica	NNSET	Colonne 12	Colonne 13
1	1	5,1	3,5	1,4	0,2	Setosa	les variables	1	0	0	Select	•	IRIS data set IIRIS data Set
2	2	4,9	3	1,4	0,2	Setosa	X1 X2 X3	1	0	0	Select	•	Source : Fisher, R.A. (1936) "The use of multiple measurements in taxonomic problems"
3	3	4,7	3,2	1,3	0,2	Setosa	constituent un codage	1	0	0	Train	•	Annual Eugenics, 7, Part II, 179-188
4	4	4,6	3,1	1,5	0,2	Setosa	disjonctif complet	1	0	0	Select	•	Data Information
5	5	5	3,6	1,4	0,2	Setosa	de la variable Y_Class	1	0	0	Train	•	This is perhaps the best known database to be found in the pattern recognition literature.
6	6	5,4	3,9	1,7	0,4	Setosa		1	0	0	Train	•	Fisher's paper is a classic in the field and is one of the most referenced frequently to this day.
7	7	4,6	3,4	1,4	0,3	Setosa	Ce sont les variables	1	0	0	Train	•	The data set contains 3 classes of 50 instances each, where each class refers to a type of iris plant.
8	8	5	3,4	1,5	0,2	Setosa	de sortie du processus	1	0	0	Select	•	One class (Setosa) is linearly separable from Versicolor and Virginica.
9	9	4,4	2,9	1,4	0,2	Setosa	de classification	1	0	0	Select	•	Versicolor and Virginica are NOT linearly separable from each other.
10	10	4,9	3,1	1,5	0,1	Setosa	des fleurs IRIS	1	0	0	Select	•	Predicted attribute = class of iris plant = X_class.
11	11	5,4	3,7	1,5	0,2	Setosa		1	0	0	Train	•	Correction of some data values
12	12	4,8	3,4	1,6	0,2	Setosa		1	0	0	Train	•	This data on the web site <https://archive.ics.uci.edu/ml/index.php> differs
13	13	4,8	3	1,4	0,1	Setosa		1	0	0	Train	•	in Fisher's article. This correction was identified by Steve Chadwick.
14	14	4,3	3	1,1	0,1	Setosa		1	0	0	Train	•	The 35th sample should be: 4.9, 3.1, 1.5, 0.2,"Iris_setosa" where the error is in the fourth feature.
15	15	5,8	4	1,2	0,2	Setosa		1	0	0	Train	•	The 38th sample: 4.9, 3.6, 1.4, 0.1,"Iris_setosa" where the errors are in the second and third features.
16	16	5,7	4,4	1,5	0,4	Setosa		1	0	0	Select	•	Remarque : le fichier Statistica que j'ai créé Iris.sta contient les valeurs corrigées.
17	17	5,4	3,9	1,3	0,4	Setosa		1	0	0	Select	•	Variables description
18	18	5,1	3,5	1,4	0,3	Setosa		1	0	0	Train	•	Y1_Sepal_Length (cm)
19	19	5,7	3,8	1,7	0,3	Setosa		1	0	0	Select	•	Y2_Sepal_Width (cm)

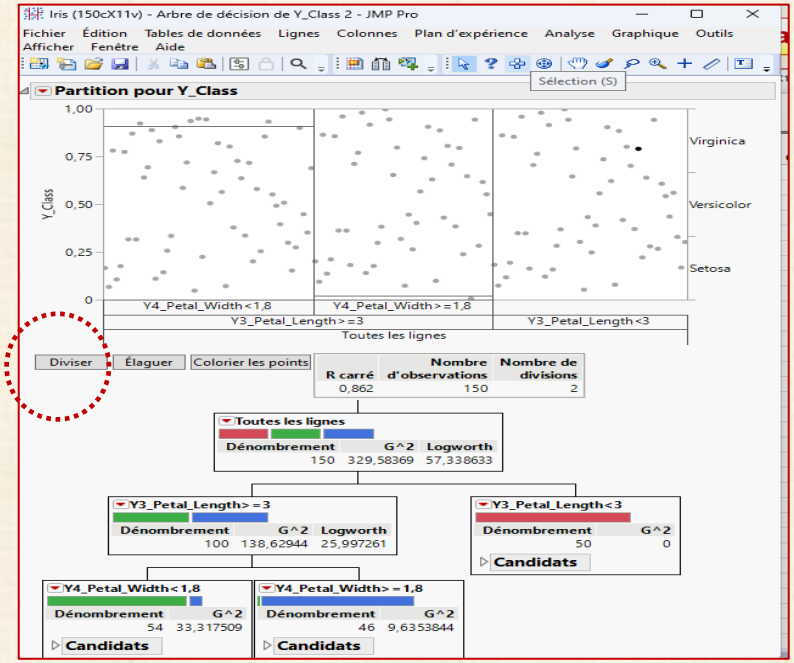


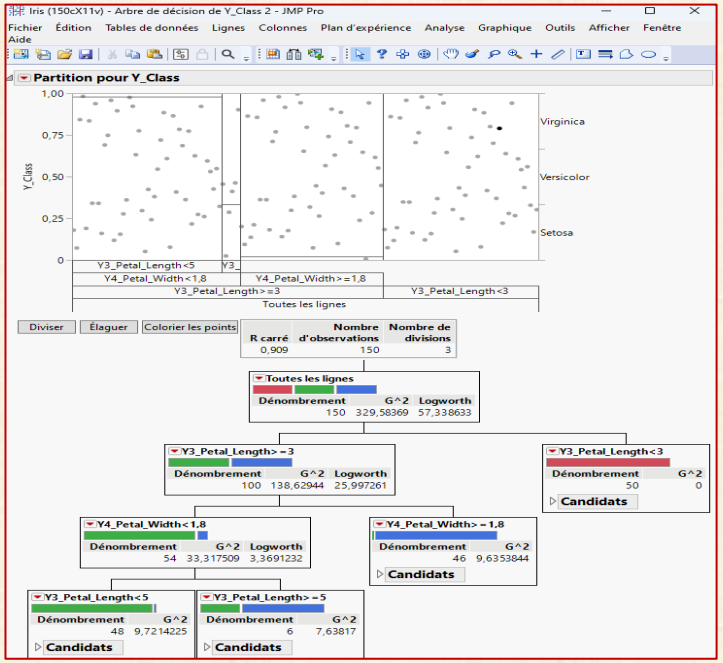


Classification et Arbres de régression : exemple avec JMP – data IRIS

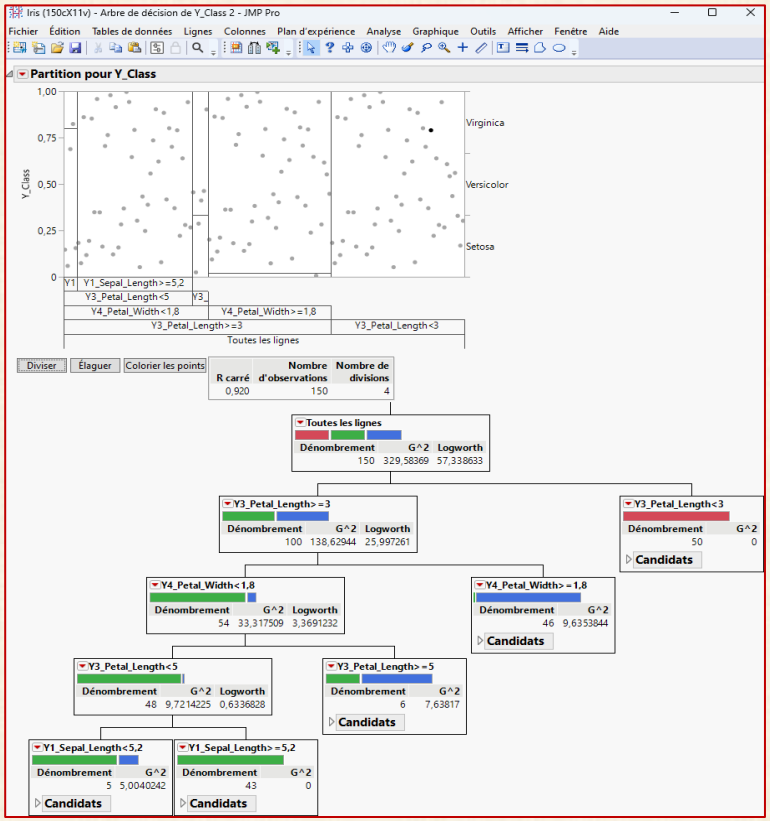


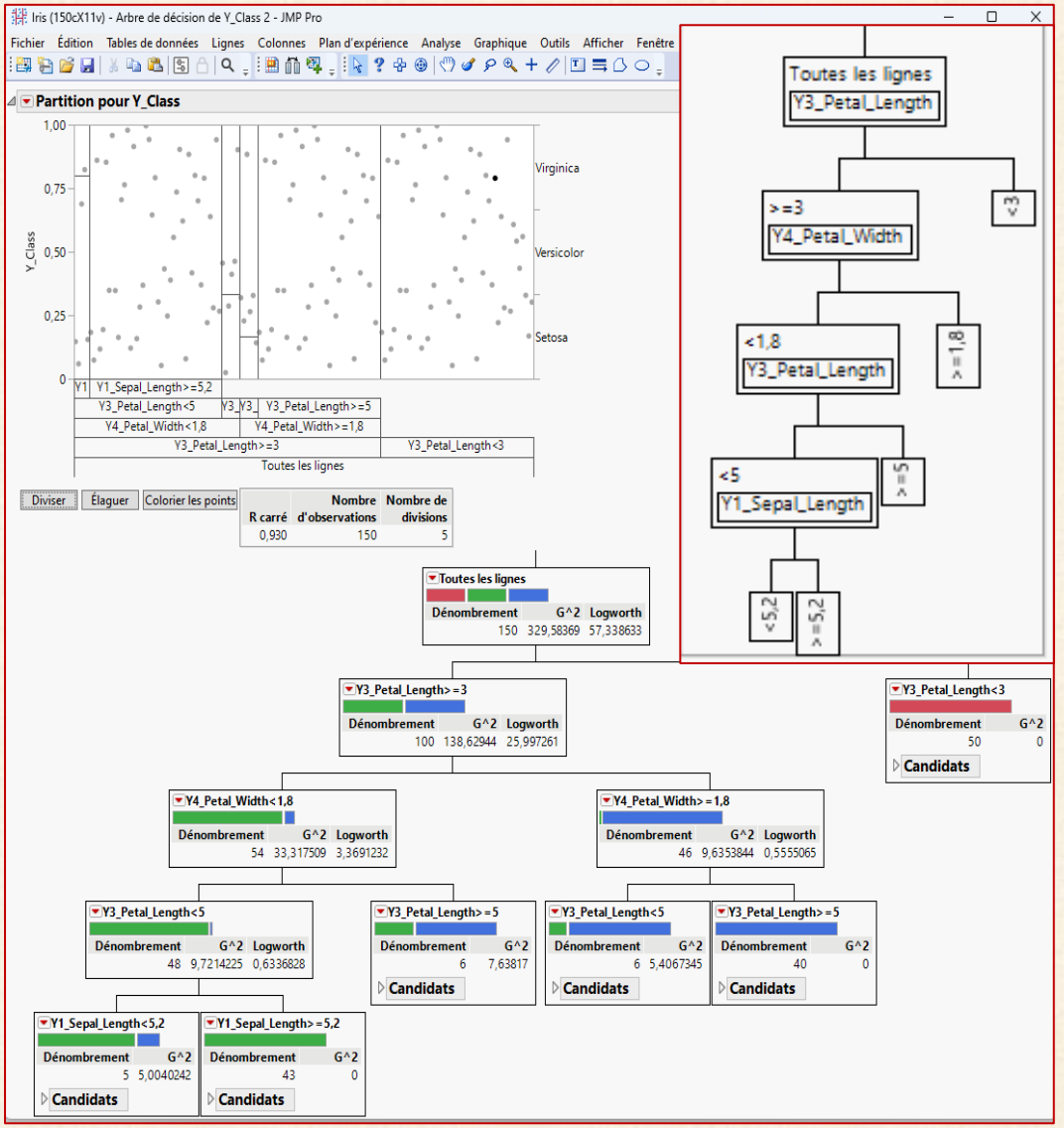






Classification et Arbres de régression : exemple avec JMP – data IRIS



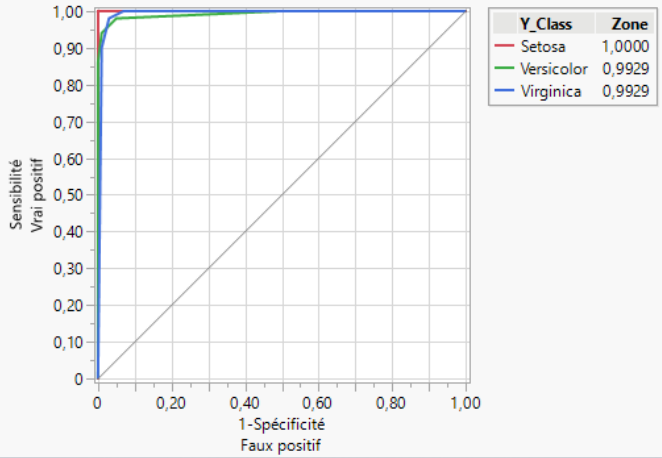


Classification et Arbres de régression : exemple avec JMP – data IRIS

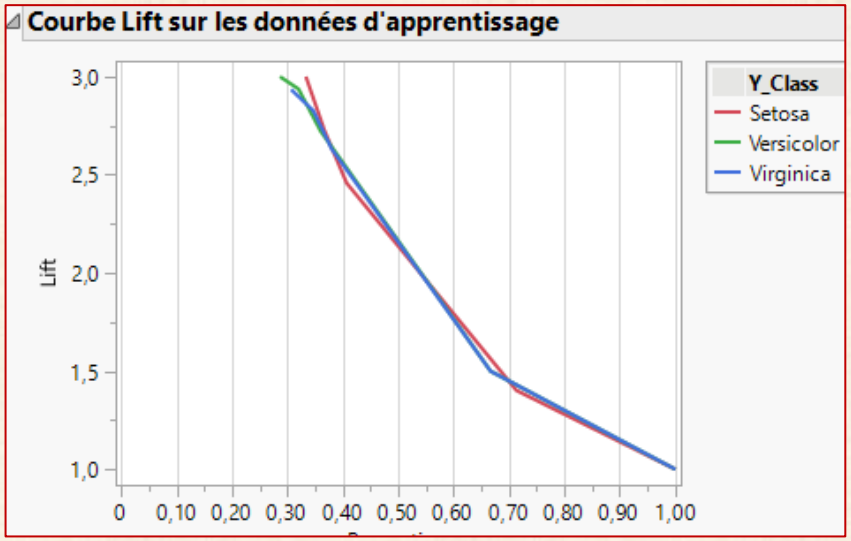
Contributions des colonnes

Terme	Nombre de divisions	G^2	Proportion
Y3_Petal_Length	2	206,912167	0,6733
Y4_Petal_Width	1	95,676543	0,3113
Y1_Sepal_Length	1	4,71739825	0,0154
Y2_Sepal_Width	0	0	0,0000

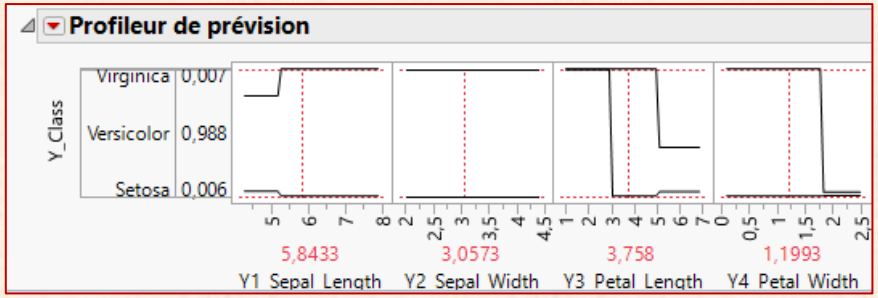
Courbe ROC sur les données d'apprentissage



Courbe Lift sur les données d'apprentissage



Profileur de prévision



Classification et Arbres de régression : exemples

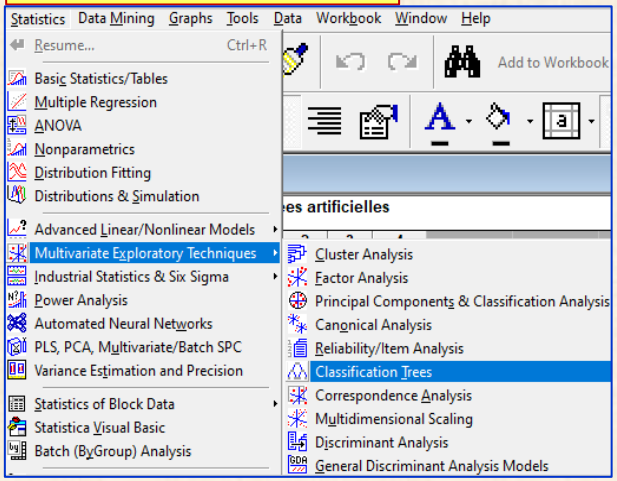
Exemple 2 : Boston Houses - n = 1006 - divisé en LEARNING (503 cas) et TEST (503 cas)

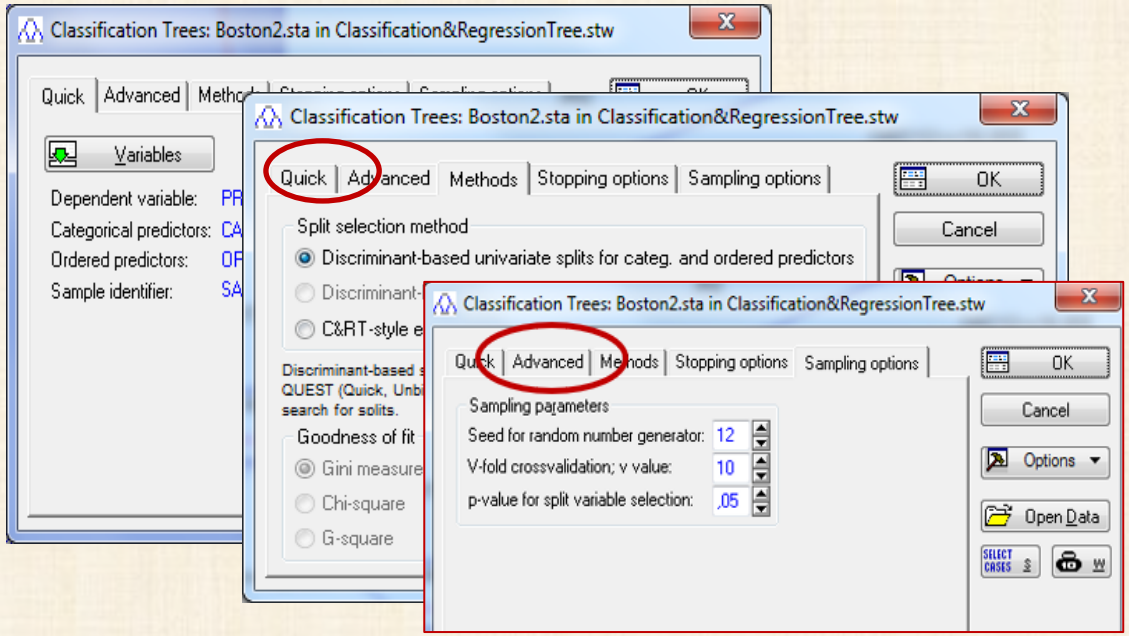
14 variables Y = PRICE classée selon (low, medium, high)

13 X : X1 = CAT1(catég. 0-1) variables continues X2 = ORD1, …, X13 = ORD12

	PRICE	CAT1	ORD1	ORD2	ORD3	ORD4	ORD5	ORD6	ORD7	ORD8	ORD9	ORD10	ORD11	ORD12	SAMPLE
1	HIGH	0	0,00632	18,0	2,31	0,538	6,575	65,2	4,090	1,0	296,0	15,30	396,90	4,980	LEARNIN
2	MEDIUM	0	0,02731	0,0	7,07	0,469	6,421	78,9	4,967	2,0	242,0	17,80	396,90	9,140	LEARNIN
3	HIGH	0	0,02729	0,0	7,07	0,469	7,185	61,1	4,967	2,0	242,0	17,80	392,83	4,030	LEARNIN
.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.
1005	MEDIUM	0	0,10959	0,0	11,93	0,573	6,794	89,3	2,389	1,0	273,0	21,00	393,45	6,480	TEST
1006	LOW	0	0,04741	0,0	11,93	0,573	6,030	80,8	2,505	1,0	273,0	21,00	396,90	7,880	TEST

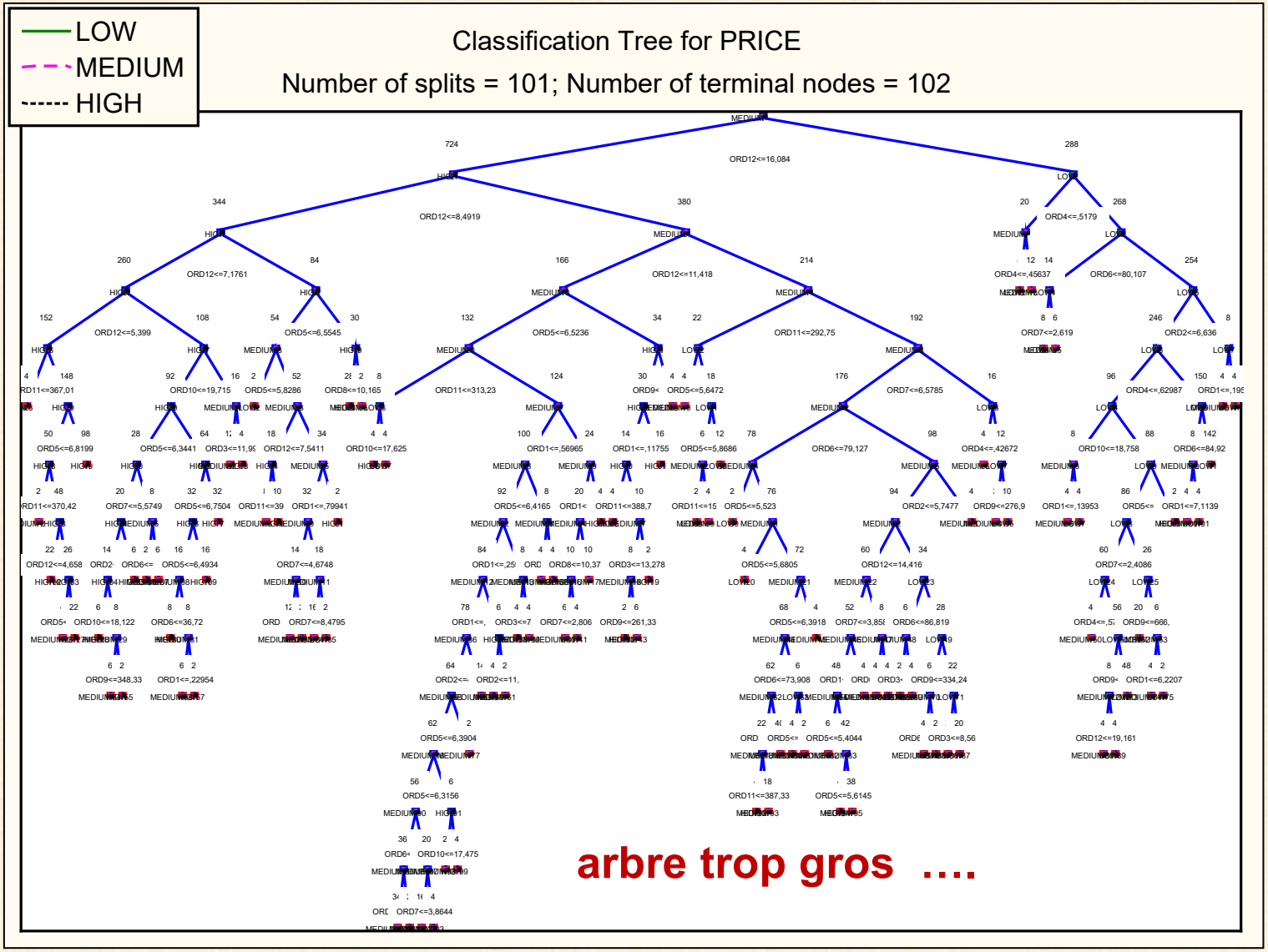
Menu Statistics





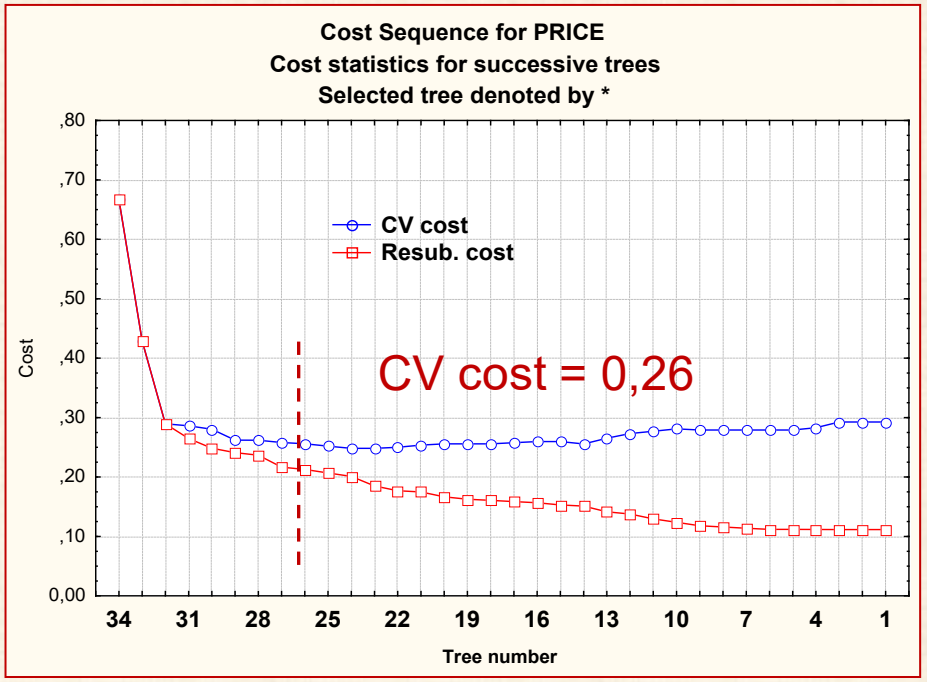
Classification et Arbres de régression : exemples

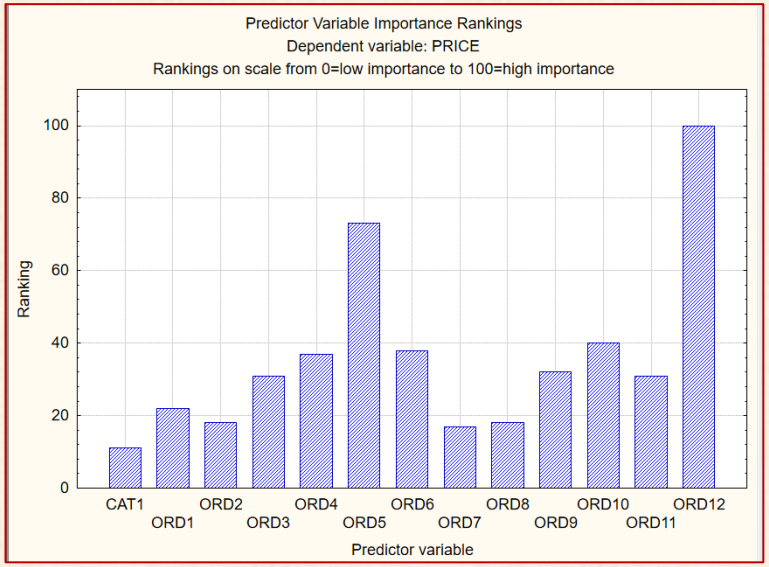
Exemple 2 : Boston Houses – n =1006



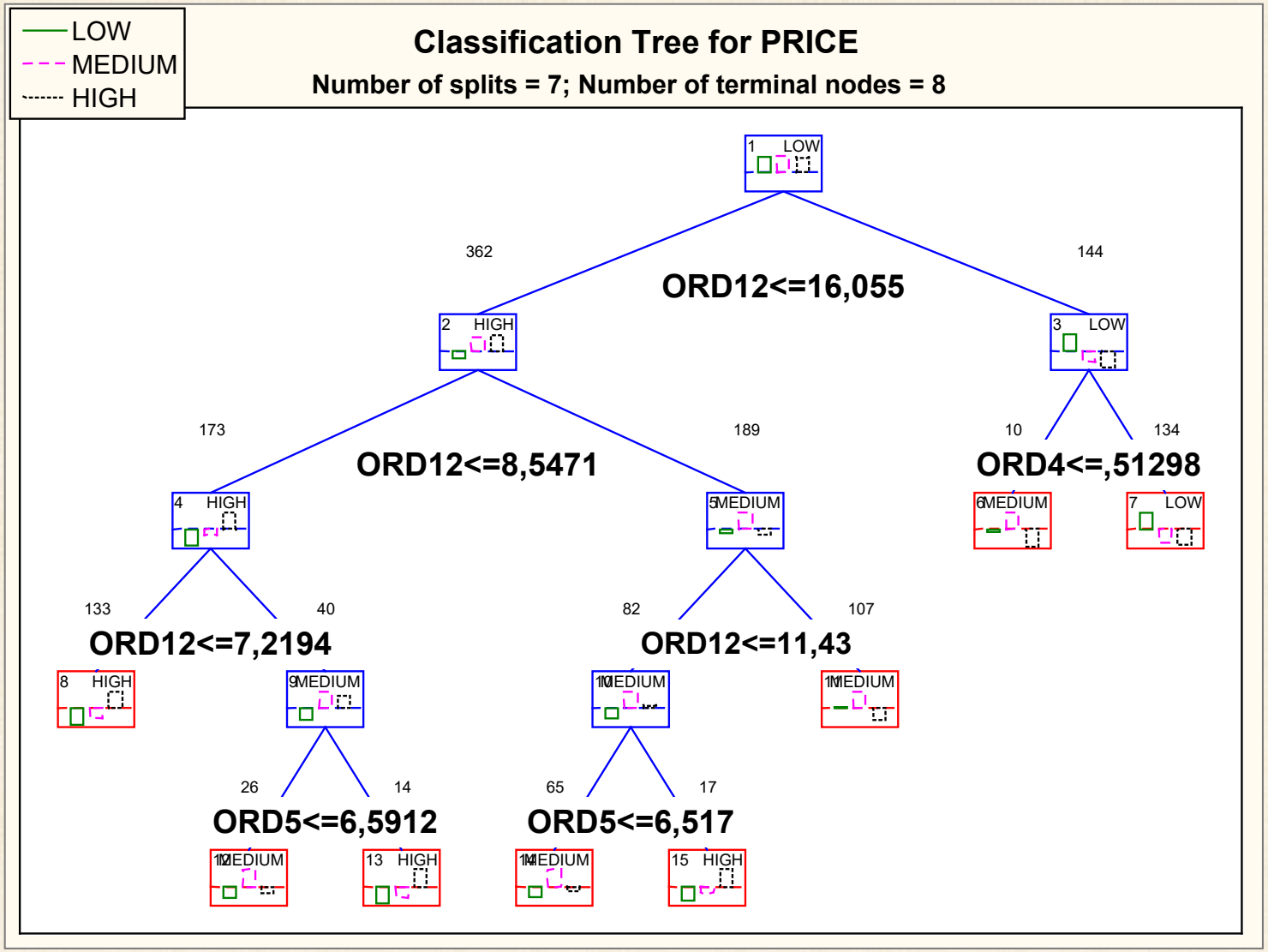
Exemple 2 : Boston Houses

Tree Sequence (Boston2.sta) Statistics for successive trees Selected tree denoted by *					
	Terminal nodes	CV cost	Std. error	Resub. cost	Node complxty
1	87	0,292544	0,019607	0,111515	0,000000
2	86	0,292544	0,019607	0,111527	0,000012
3	84	0,292613	0,019657	0,111596	0,000035
4	83	0,282979	0,019586	0,111665	0,000069
5	81	0,279126	0,019549	0,111828	0,000081
6	80	0,279126	0,019549	0,111966	0,000138
7	77	0,279126	0,019549	0,113893	0,000642
8	74	0,279126	0,019549	0,115901	0,000669
9	71	0,279126	0,019549	0,117966	0,000688
10	63	0,281052	0,019568	0,123747	0,000723
11	57	0,277060	0,019445	0,129527	0,000963
12	50	0,273125	0,019370	0,137686	0,001166
13	47	0,265256	0,019213	0,141458	0,001257
14	41	0,255703	0,019121	0,151092	0,001606
15	40	0,259638	0,019207	0,152949	0,001858
16	38	0,259638	0,019207	0,156722	0,001886
17	37	0,257711	0,019179	0,158649	0,001927
18	36	0,255854	0,019183	0,160645	0,001996
19	35	0,255854	0,019183	0,162653	0,002008
20	33	0,255772	0,019155	0,166714	0,002031
21	29	0,253764	0,019096	0,174906	0,002048
22	28	0,249980	0,019064	0,176971	0,002065
23	25	0,248204	0,019080	0,184967	0,002665
24	19	0,248204	0,019080	0,200971	0,002667
25	17	0,251988	0,019123	0,206751	0,002890
26	15	0,255980	0,019244	0,212775	0,003012
27	14	0,258090	0,019336	0,216629	0,003854
28	9	0,261944	0,019404	0,236709	0,004016
*29	8	0,262128	0,019424	0,240978	0,004269
30	7	0,280326	0,019936	0,248478	0,007499
31	5	0,286131	0,020057	0,264867	0,008195
32	3	0,289509	0,019991	0,289509	0,012321
33	2	0,429166	0,011864	0,429166	0,139657
34	1	0,666667	0,000000	0,666667	0,237501





Exemple 2 : Boston Houses



validation croisée v-fold

v (souvent v = 10) échantillons aléatoires de même taille sont obtenus pour analyse

On met de côté un groupe pour tester.

On développe un modèle prédictif / classifications avec l'ensemble des autres groupes.

On utilise le groupe mis de côté évaluer la performance / precision du modèle développé.

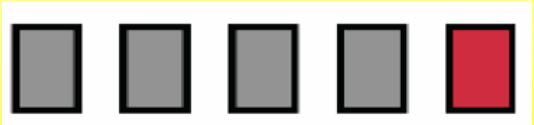
On répète le processus pour chacun des sous groupes

première analyse et évaluation

répétition

……

dernière analyse et évaluation



Mesures de qualité du modèle

- erreurs dans les prédictions numériques
- courbes: Lift - Gains - ROC
- proportion d'erreurs

ROC Curve

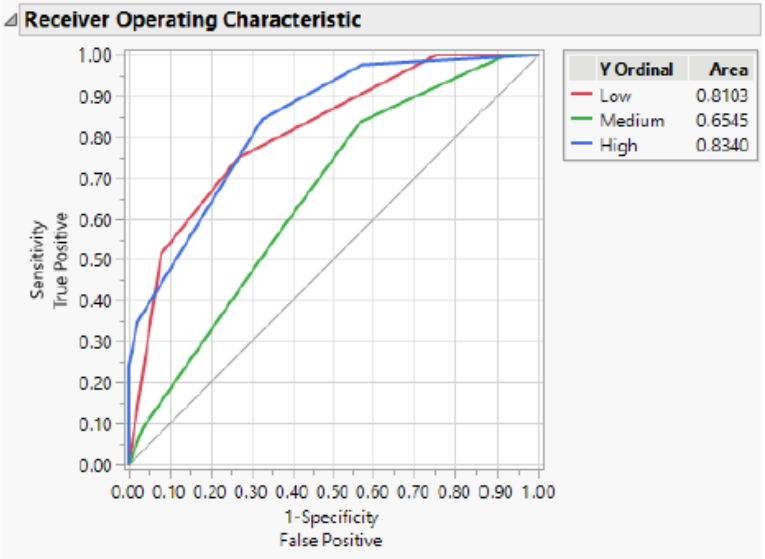
source : JMP

The ROC Curve option available only for categorical responses.
Receiver Operating Characteristic (ROC) curves display the efficiency of a model's fitted probabilities in sorting the response levels.
An introduction to ROC curves is found in the Logistic Analysis chapter in *Basic Analysis*.
The predicted response for each observation in a partition model is a value between 0 and 1.
To use the predicted response to classify observations as positive or negative, a *cut point* is used.
For example, if the cut point is 0.5, an observation with a predicted response at or above 0.5 would be classified as positive, and an observation below 0.5 as negative. There are trade offs in classification as the cut point is varied.
The following values are computed for each possible cut point:

- The *sensitivity* is the proportion of true positives or the percent of positive observations with a predicted response greater than the cut point.
- The *specificity* is the proportion of true negatives or the proportion of negative observations with a predicted response less than the cut point.

The ROC curve plots sensitivity against 1 - specificity.
If your response has more than two levels, the Partition report contains a separate ROC curve for each response level versus the other levels. Each curve is the representation of a level as the positive response level.
If there are 2 levels, one curve is the reflection of the other.

Figure 4.17 ROC Curves for a Three Level Response



Mesures de qualité du modèle

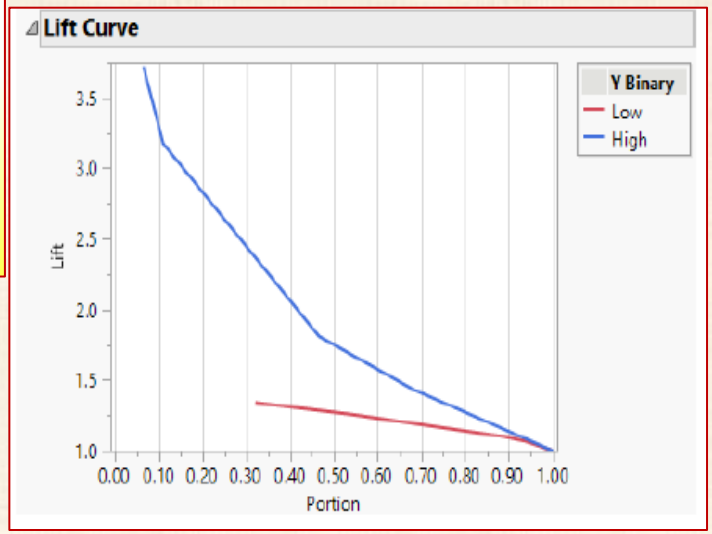
- erreurs dans les prédictions numériques
- courbes: Lift - Gains - ROC
- proportion d'erreurs

Lift Curve source : JMP Pro

The Lift Curve option provides another plot to display the predictive ability of a partition model.

The lift curve plots the lift versus the portion of the observations. There is a point for each unique predicted probability value. Each predicted probability of a response level defines a portion of The observations that have a predicted probability greater than or equal to the unique predicted probability value.

For a particular level of the response, the *lift* value is the ratio of the proportion of observed Responses in that portion to the overall proportion of observed responses.

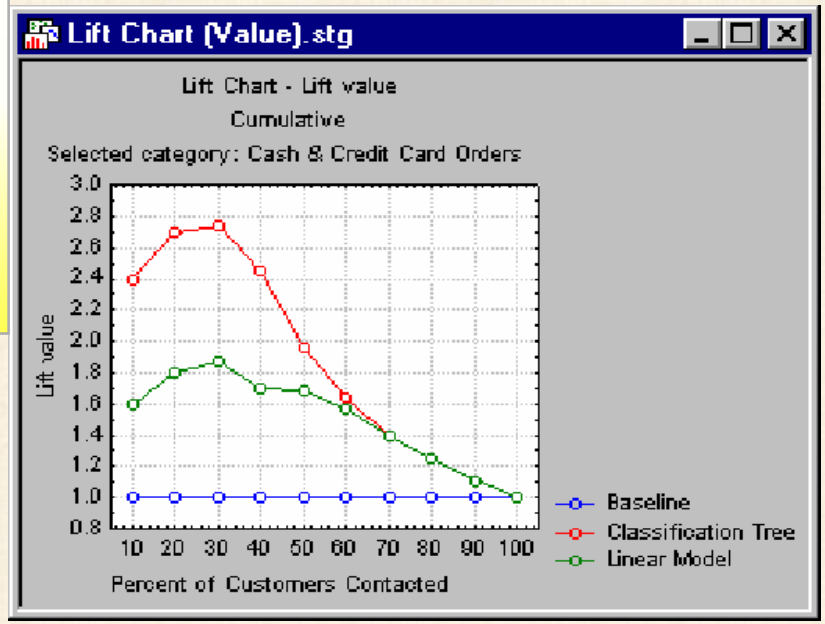


Lift chart source : STATISTICA

Sommaire visuel de l'utilité de l'information fournie par un ou plusieurs modèles pour prédire une variable de réponse binaire.

Pour une variable de réponse multinomiale, on calcule un lift chart pour chaque catégorie.

La valeur du lift (axe vertical) résume l'utilité que l'on peut s'attendre comparativement au modèle neutre ('baseline' : choix au hasard)



Exemple 2 : Boston Houses

Predicted Class x Observed Class n's
Predicted (row) x observed (column) matrix
Learning sample N = 506

predicted	observ LOW	observ MEDIUM	observ HIGH
LOW	118	14	2
MEDIUM	48	131	29
HIGH	1	28	135

Global CV Sample Misclassification Matrix
Predicted (row) x observed (column) matrix
Global CV cost = ,2814;
s.d. CV cost = ,01973

predicted	observ LOW	observ MEDIUM	observ HIGH
LOW		22	3
MEDIUM	43		30
HIGH	3	42	

Test Sample Misclassification Matrix
Predicted (row) x observed (column) matrix
CV cost = ,24098; s.d. CV cost = ,01891

predicted	obser LOW	observ MEDIUM	observ HIGH
LOW		14	2
MEDIUM	48		29
HIGH	1	28	

Exemple 3 data = PressBanding

Source: Gaudard, M. Ramsey, P., Stephens, M. (2006). Interactive Data Mining and Design of Experiments

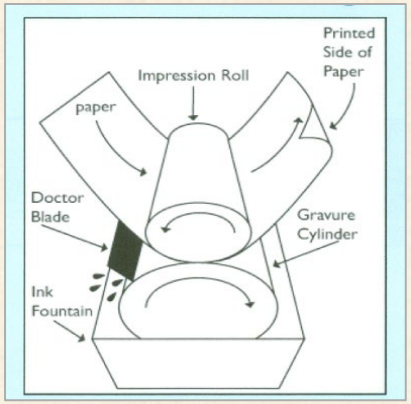
Example printing industry - process description

Problem defect on cylinder = « banding » ➡ image of poor quality occurring 40% of time

Action stop production – repair cylinder many hours - cost = 10 000$ per incident

Project what conditions lead to banding?

Data 540 observations x 39 variables Y = band occured (red)
X : 11 categorical var (blue) + 17 continuous var (green)



data

1 ID	2 Year	3 Month	4 Day	5 Job Number	6 Cylinder No.	7 Customer	8 Band Occurred	9 grain screened	10 proof on ctd ink	11 paper type	12 ink type	13 solvent type	14 type on cylinder	15 cylinder size	16 location	17 plating tank	18 press type
1	1990	3	30	23040	X750	GUIDEPOSTS	yes	YES	YES	UNCOATED	UNCOATED	LINE	NO	TABLOID			Motter94
2	1990	4	9	34683	G467	ECKERD	no	NO	YES	COATED	COATED	LINE	YES	TABLOID	NorthUS	1910	WoodHoe7
3	1990	4	9	25416	X203	TVGUIDE	yes	YES	YES	UNCOATED	UNCOATED	LINE	NO	TABLOID	NorthUS	1911	Motter94

19 press	20 proof cut	21 viscosity	22 caliper	23 ink temperature	24 humidity	25 roughness	26 blade pressure	27 varnish pct	28 press speed	29 ink pct	30 solvent pct	31 wax	32 hardener	33 roller durometer	34 current density	35 anode space ratio	36 chrome content	37 Time Stamp	38 ESA Voltage
821	50,0	59	0,33	14,5	71	0,63	20	11,1	1650	55,5	33,3	2,5		40			100	19900330	
815	40,0	38	0,30	16,0	92	0,63	25	1,1	1900	53,8	45,2	2,5		30	33	100,0	100	19900409	
821	47,5	41	0,33	18,0	74	0,50	30	17,7	1900	44,2	38,1	2,5		40	33	100,0	100	19900409	

Analyses pour identifier les facteurs importants

ANALYSES et GRAPHIQUES

identification des facteurs X plus responsables du défaut :

Y = band occured (yes, no)

- ANOVA Y vs X continues (v19 v20 …v36)
- COMPTAGES Y vs X continues (v19 v20 …v36)
- TABLEAU CROISÉ Y vs X catégoriques (v9 …v18)
- Arbre (CRT) : Y vs X continues et X catégoriques

ANOVA

groupe = Y

facteurs = X continues

X considéré réponse

Y groupe

1 variables continues	2 SS Effect	3 df Effect	4 MS Effect	5 SS Error	6 df Error	7 MS Error	8 SS total =v1+v4;	9 R2 =v1 / v7;	10 F	11 p
press speed	2236084	1	2236084	51482144	526	97874,80	53718228,44	0,0416	22,84638	0,000002
roller durometer	189	1	189	9637	483	19,95	9826,00	0,0192	9,45253	0,002228
hardener	1	1	1	51	319	0,16	51,74	0,0226	7,37875	0,006960
ink pct	215	1	215	14724	482	30,55	14938,84	0,0144	7,04061	0,008231
current density	38	1	38	2902	530	5,48	2940,31	0,0130	6,99319	0,008425
humidity	398	1	398	31727	536	59,19	32125,68	0,0124	6,73023	0,009739
roughness	0	1	0	12	431	0,03	12,20	0,0113	4,91894	0,027084
chrome content	16	1	16	1823	534	3,41	1838,76	0,0085	4,57918	0,032815
wax	1	1	1	57	392	0,15	57,61	0,0090	3,54708	0,060390
viscosity	195	1	195	34420	532	64,70	34615,63	0,0056	3,02102	0,082770
varnish pct	58	1	58	9679	257	37,66	9737,14	0,0060	1,54646	0,214792
solvent pct	15	1	15	5906	482	12,25	5921,59	0,0026	1,25149	0,263826
ink temperature	2	1	2	874	535	1,63	875,43	0,0022	1,16328	0,281273
proof cut	80	1	80	39509	483	81,80	39589,12	0,0020	0,97479	0,323983
caliper	0	1	0	2	510	0,00	2,47	0,0004	0,19738	0,657030
blade pressure	4	1	4	39534	474	83,40	39537,93	0,0001	0,05118	0,821119
anode space ratio	0	1	0	13322	530	25,14	13322,42	0,0000	0,00987	0,920904

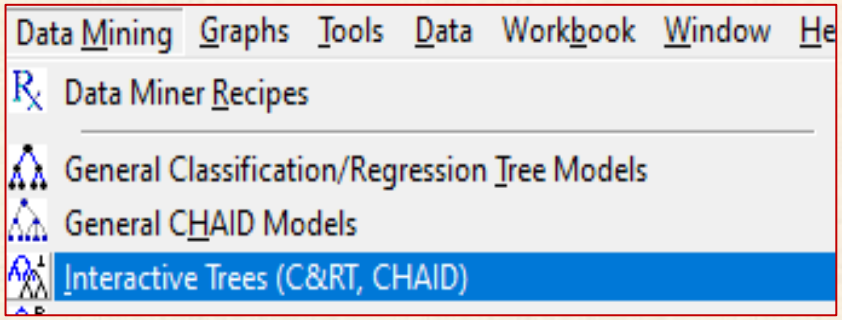
COMPTAGES

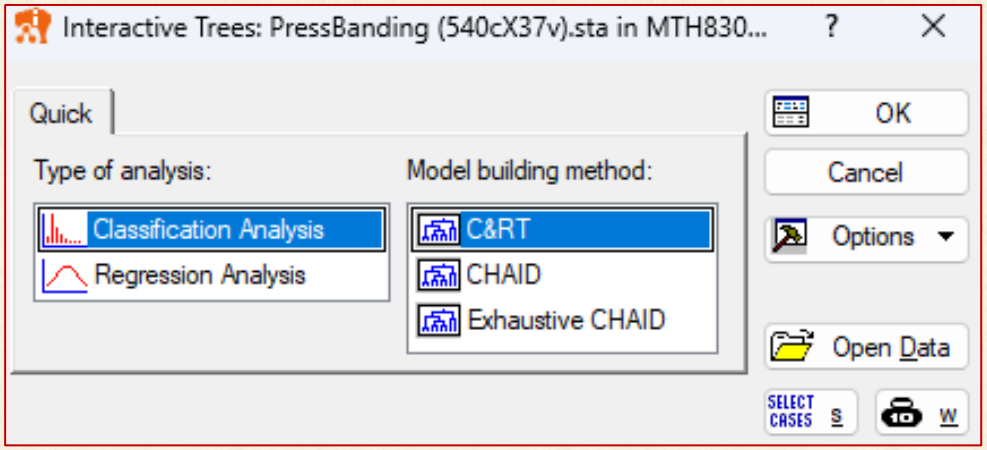
ratio

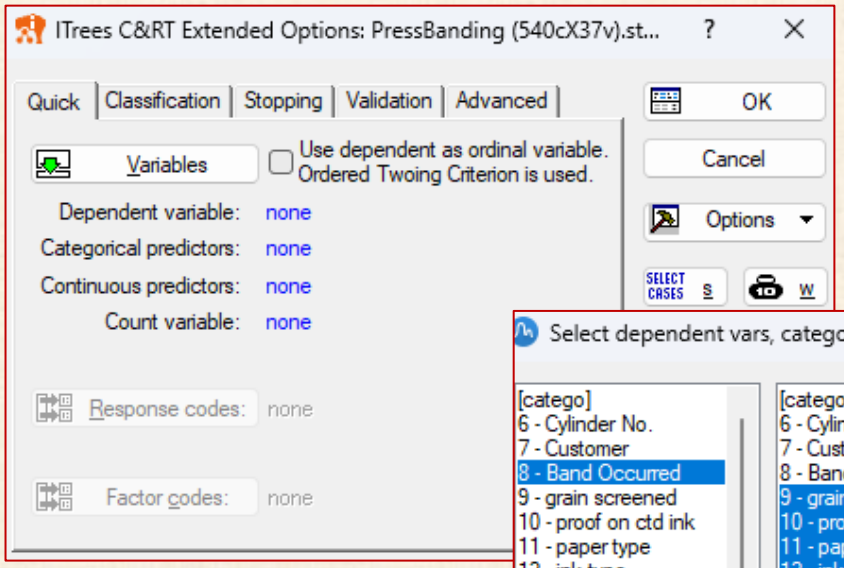
Y = Yes / n

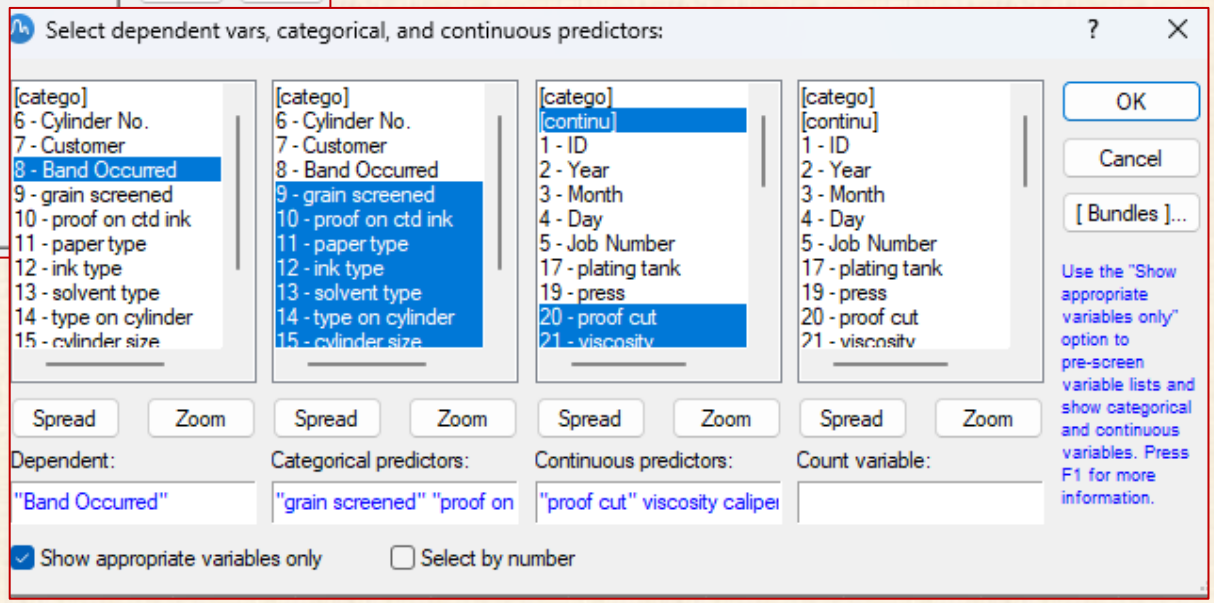
1 variables continues	2 Band Occur YES n	3 Band occur NO n	4 Band Occur total n	5 ratio yes / total
blade pressure	212	264	476	44,5
wax	172	222	394	43,7
roughness	187	246	433	43,2
press speed	225	303	528	42,6
ink temperature	227	310	537	42,3
humididy	227	311	538	42,2
current density	223	309	532	41,9
anode space ratio	223	309	532	41,9
chrome content	224	312	536	41,8
viscosity	222	312	534	41,6
caliper	209	303	512	40,8
proof cut	180	305	485	37,1
roller durometer	173	312	485	35,7
ink pct	172	312	484	35,5
solvent	172	312	484	35,5
hardener	110	211	321	34,3
varnish	73	186	259	28,2

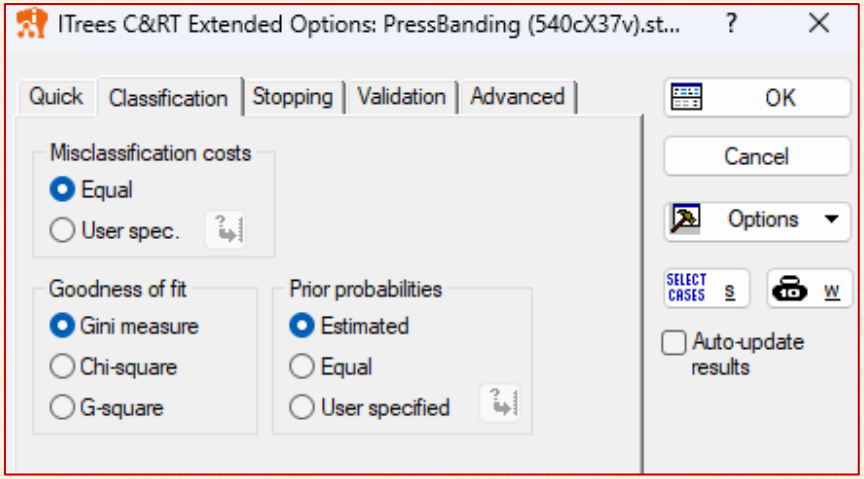
Arbre de classification : méthode interactive

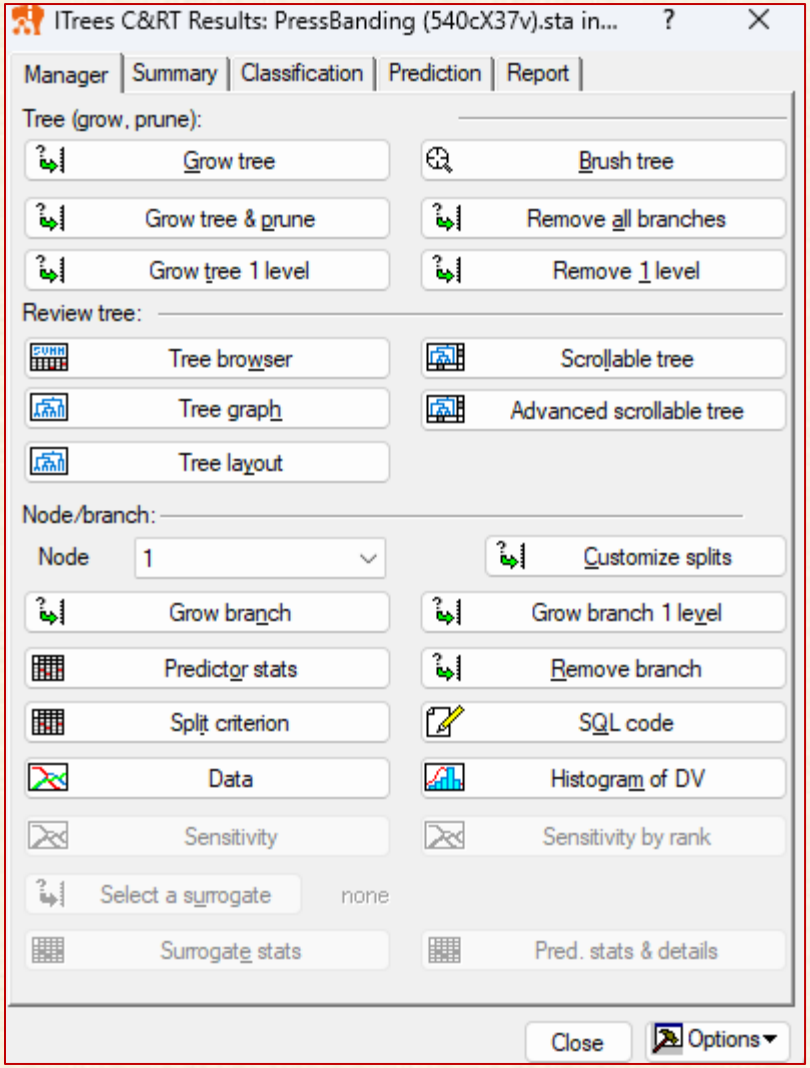




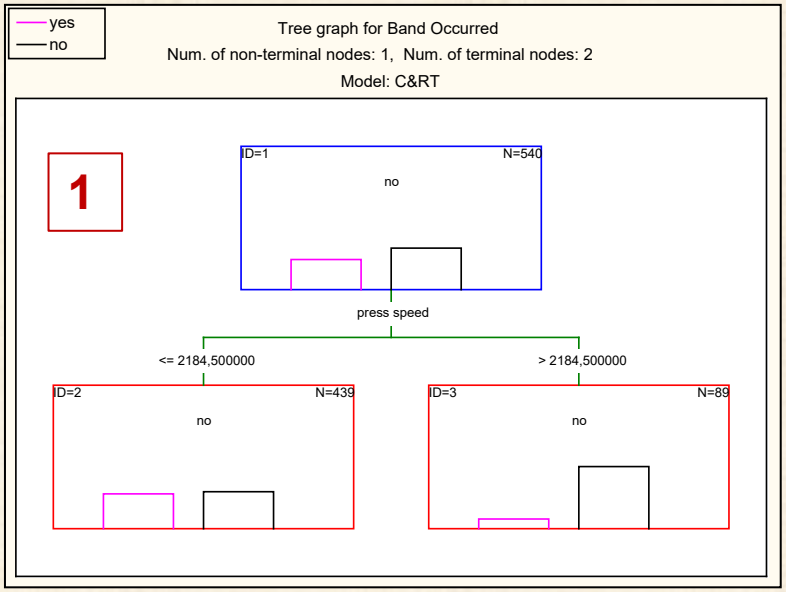


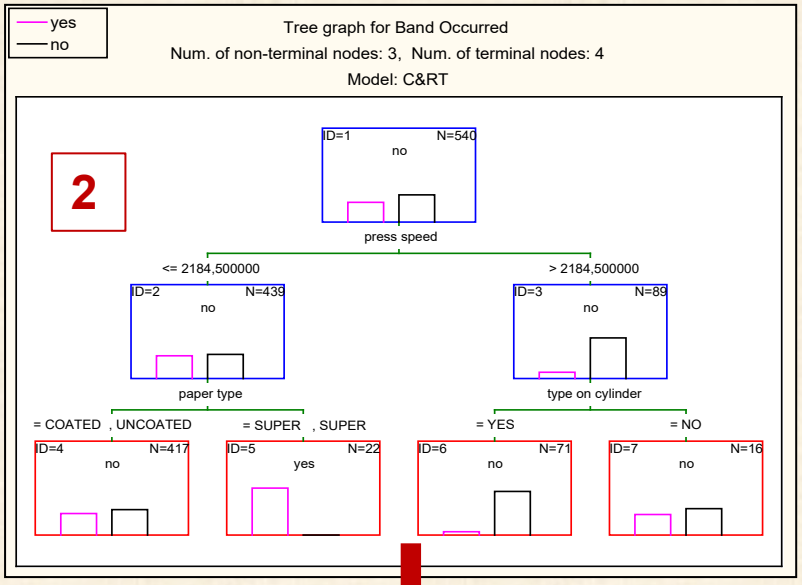


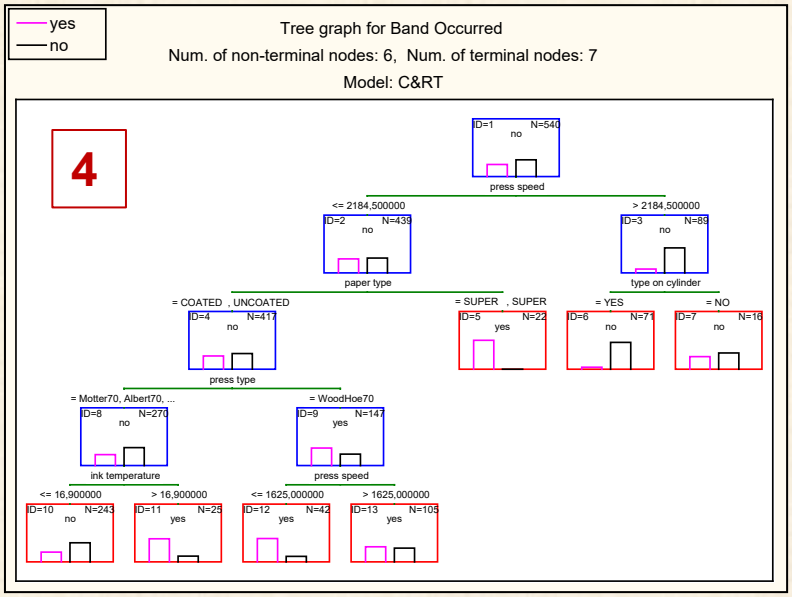

ITrees C&RT Extended Options: PressBanding (540cX37v).st...
Quick
Classification
Stopping
Validation
Advanced
Misclassification costs
Equal
User spec.
Goodness of fit
Gini measure
Chi-square
G-square
Prior probabilities
Estimated
Equal
User specified
OK
Cancel
Options
SELECT CASES
Auto-update results

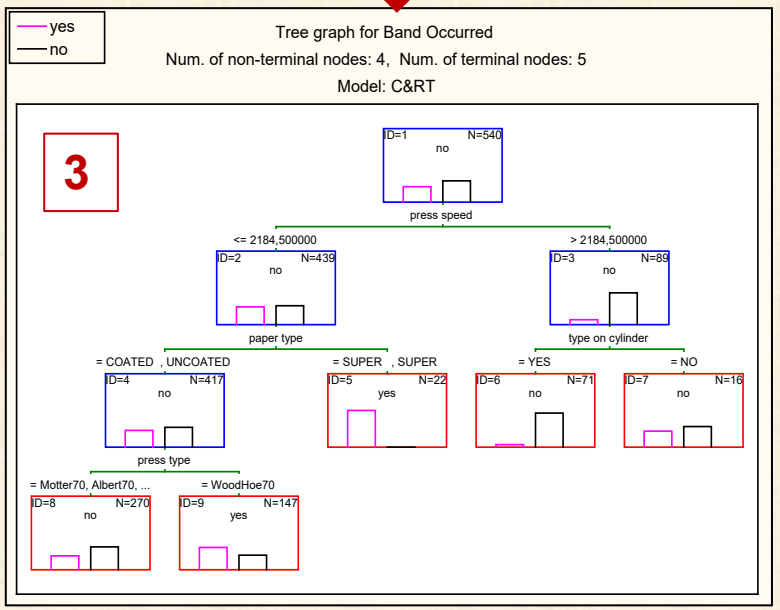

ITrees C&RT Results: PressBanding (540cX37v).sta in...
Manager
Summary
Classification
Prediction
Report
Tree (grow, prune):
Grow tree
Brush tree
Grow tree & prune
Remove all branches
Grow tree 1 level
Remove 1 level
Review tree:
Tree browser
Scrollable tree
Tree graph
Advanced scrollable tree
Tree layout
Node/branch:
Node
1
Customize splits
Grow branch
Grow branch 1 level
Predictor stats
Remove branch
Split criterion
SQL code
Data
Histogram of DV
Sensitivity
Sensitivity by rank
Select a surrogate
none
Surrogate stats
Pred. stats & details
Close
Options

Exemple 3 data = PressBanding

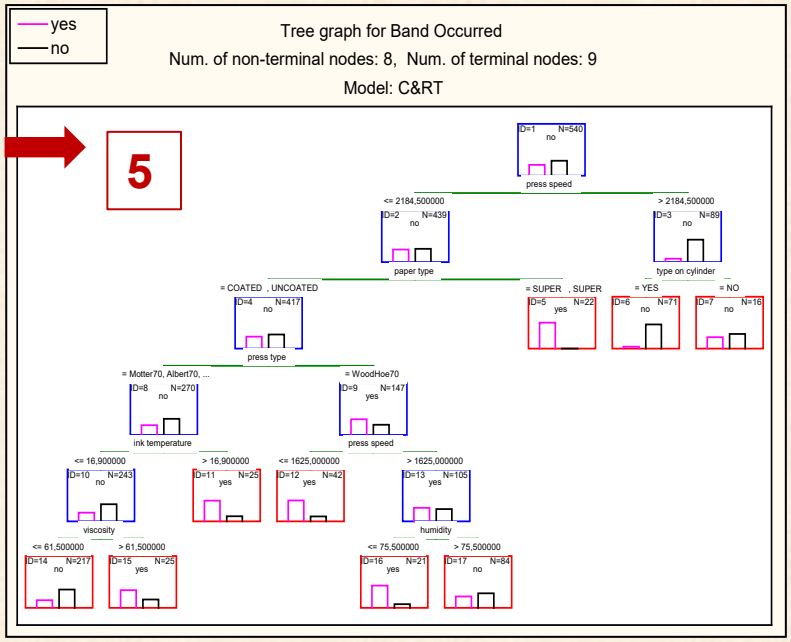


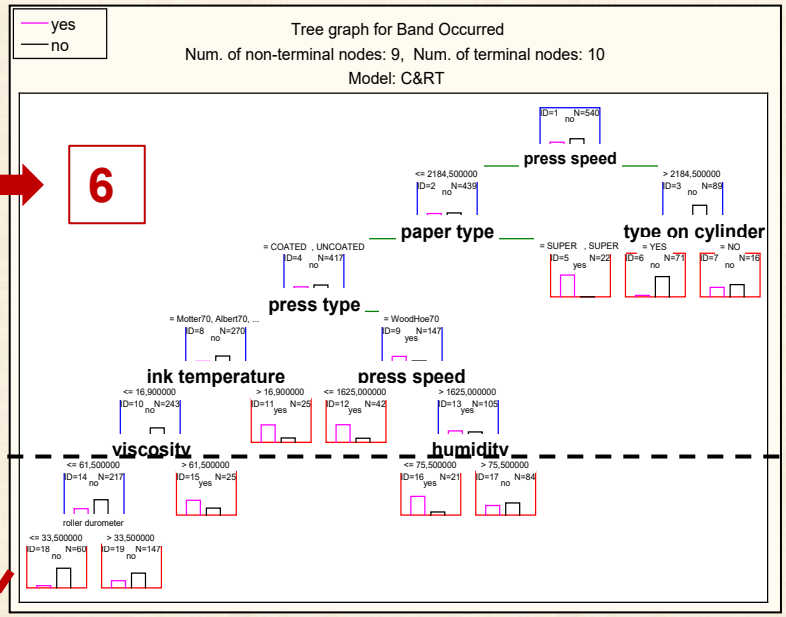






Exemple 3 data = PressBanding





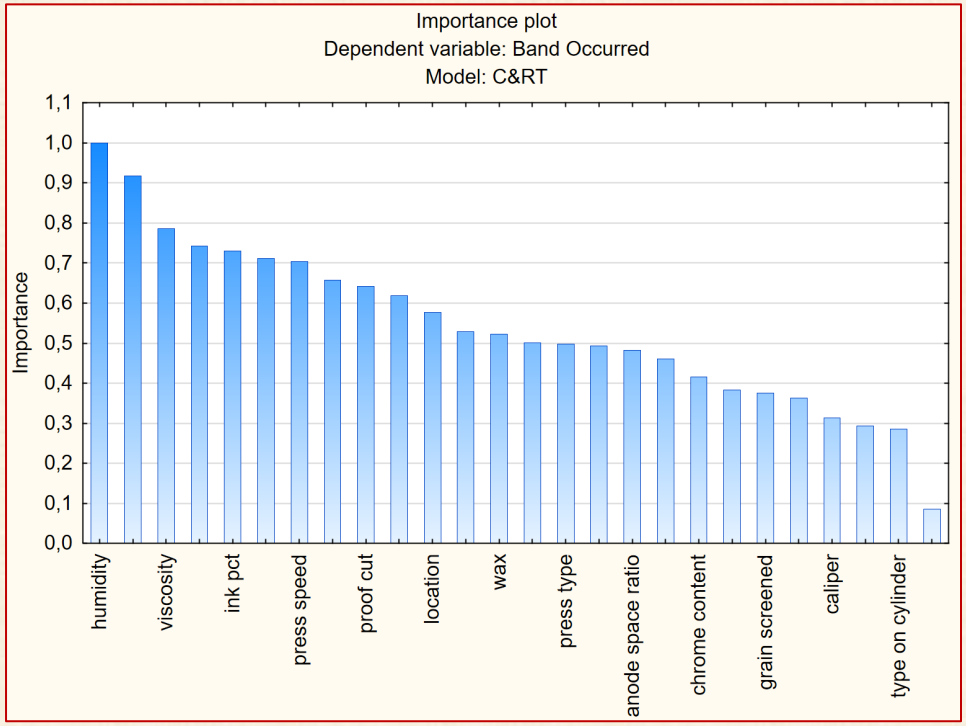
8 variables critiques et contrôlables
input variables (X) identifiées

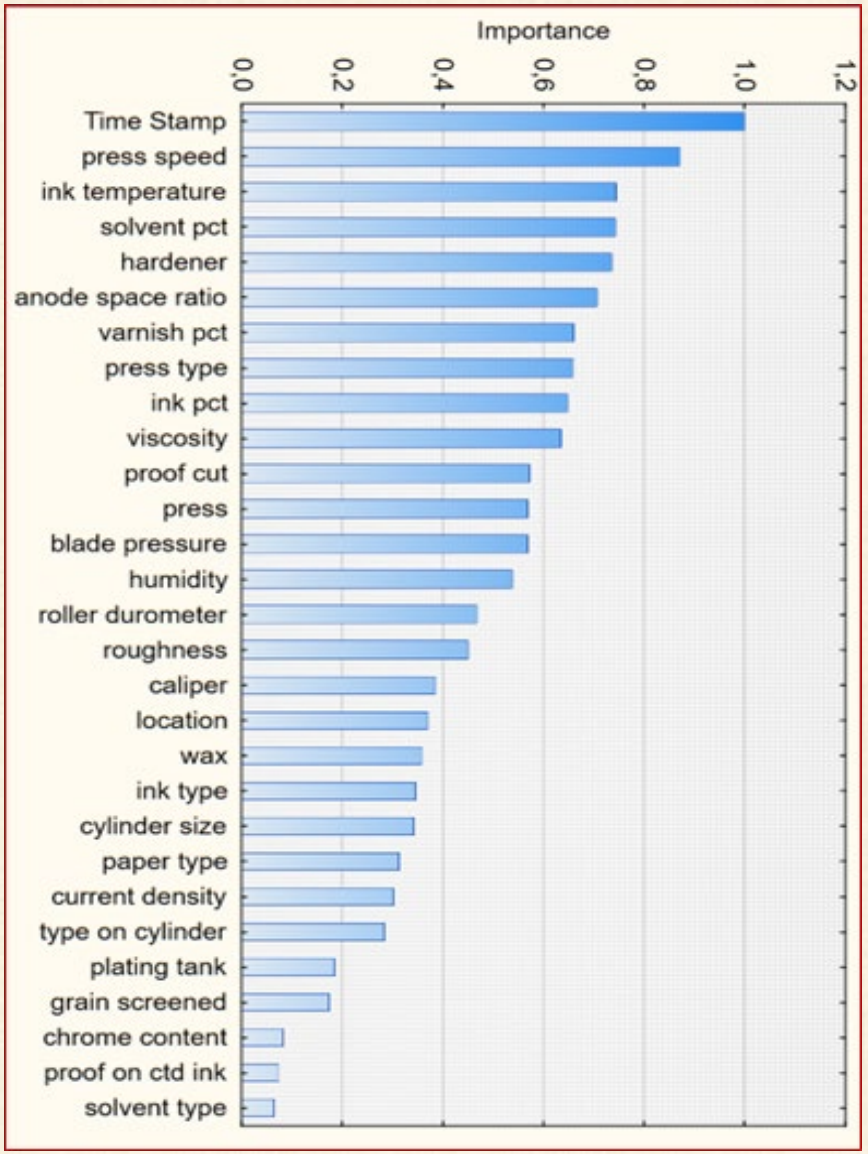
1.	press speed	continue
2.	paper type	catégorique
3.	type on cylinder	catégorique
4.	press type	catégorique
5.	ink temperature	continue
6.	grain screened	catégorique
7.	Viscosity	continue
8.	Humidity	continue

Suite : Planification expérience

Design Of Experiment (DOE)
8 input X variables
24 essais - nouvelles obs. Y
Modele Y avec X
Valeurs optimales de X déterminées
tells que Y = yes (banding) MIN
Résultat : Y= 4,5% maintenant
vs. Y = 40% avant

Arbre de classification : Importance des facteurs

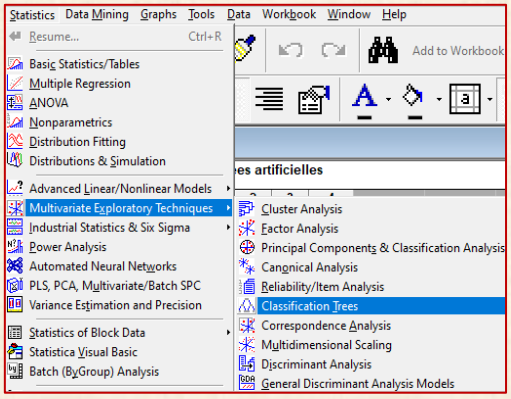


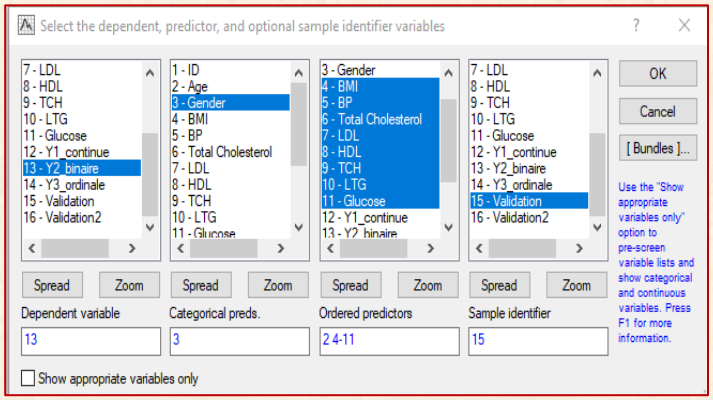


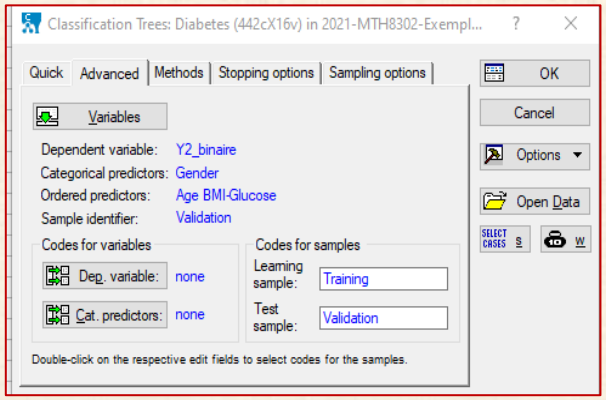
Exemple 4 data = diabètes – analyse avec STATISTICA

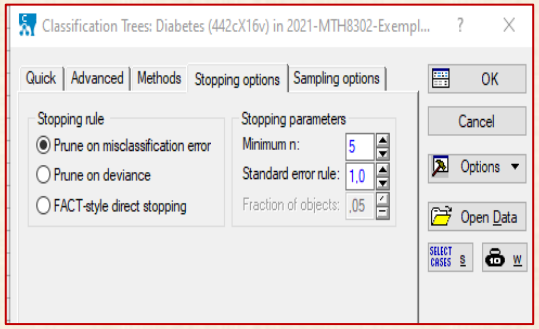
Efron, B., Hastie, T., Johnstone, J., and Tibshirani, R. (2004). Least Angle Regression.
Annals of Statistics (with discussion), vol. 32, pp. 407-499.
DATA Diabetes / n = 442 obs. X 13 variables : 10 X explicatives (v2 v3 ... v11) et 3 Y (continue, binaire, ordinale)
Y1_continue is a quantitative measure of disease progression one year after baseline. (25 à 346)
Y2_binaire et Y3_ordinale sont des recodages de Y1_continue
Y2_binaire = Low si Y_continue = 200 ou moins / = Hig si Y_continue 201 ou plus
Y3_ordinale = low si Y_continue = 150 ou moins / = Medium si Y_continue comprise entre 151 et 200 / = High si Y_continue = 201 ou plus
données séparées en 2 groupes : Training (309 obs.) Validation (133 obs.)

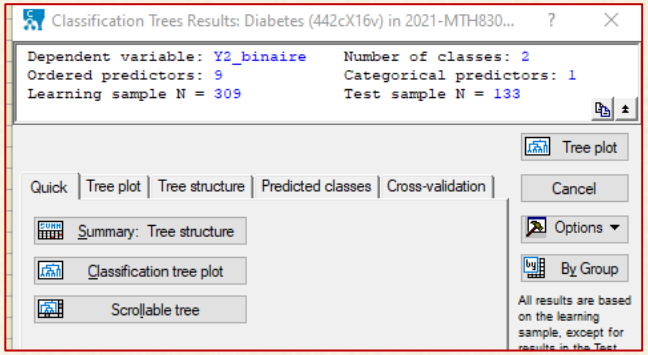
	1 ID	2 Age	3 Gender	4 BMI	5 BP	6 Total Cholesterol	7 LDL	8 HDL	9 TCH	10 LTG	11 Glucose	12 Y1_continue	13 Y2_binaire	14 Y3_ordinale	15 Validation	16 Validation2
1	1	59	2	32,1	101,00	157	93,2	38	4,00	4,8598	87	151	Low	Medium	Training	1
2	2	48	1	21,6	87,00	183	103,2	70	3,00	3,8918	69	75	Low	Low	Validation	2
3	3	72	2	30,5	93,00	156	93,6	41	4,00	4,6728	85	141	Low	Low	Training	1
4	4	24	1	25,3	84,00	198	131,4	40	5,00	4,8903	89	206	High	High	Training	1

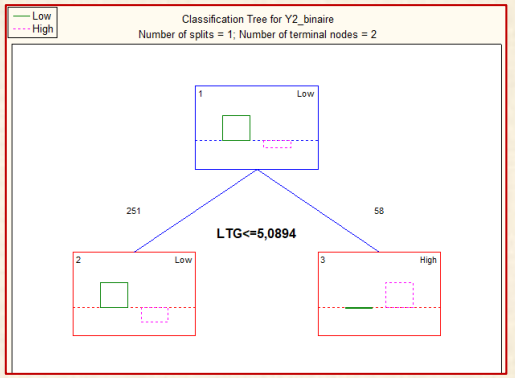




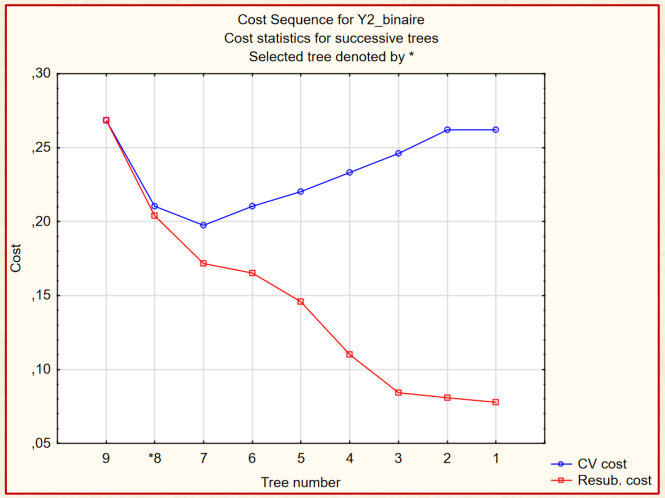


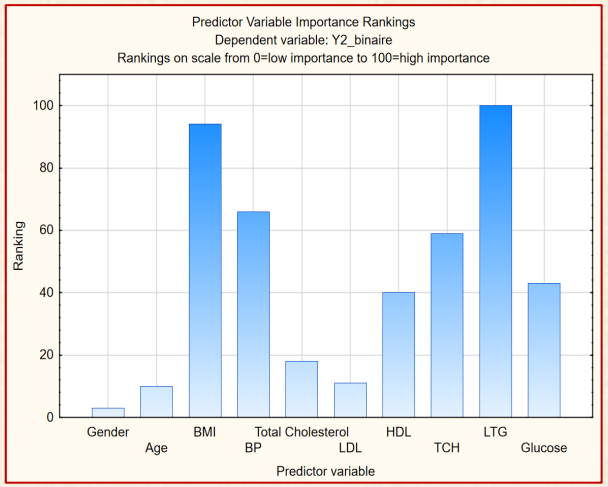






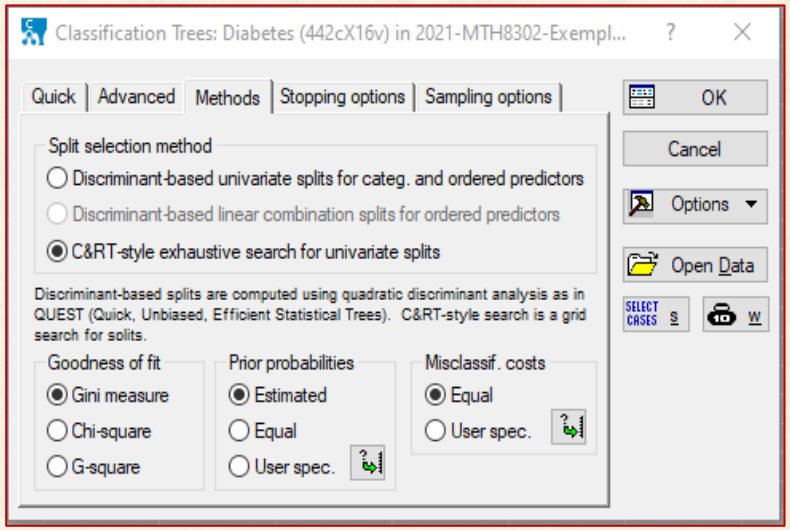
Exemple 4 data = diabètes – analyse avec STATISTICA

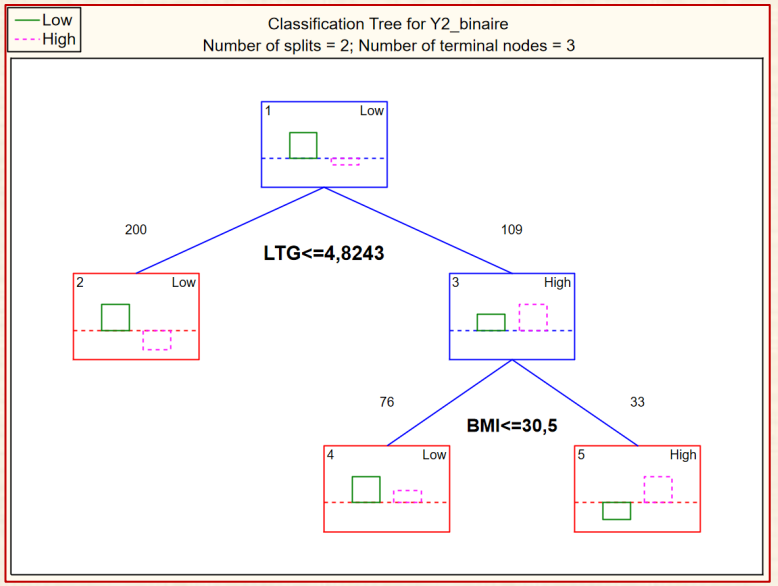




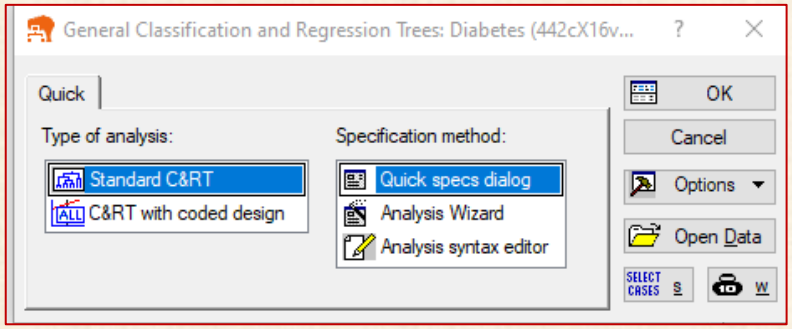
Predicted Class x Observed Class n's
Predicted (row) x observed (column) matrix
Learning sample N = 309

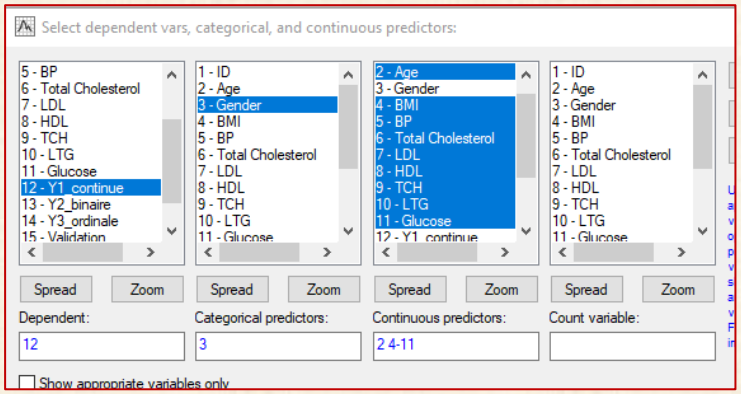
Class	Class Low	Class High
Low	207	44
High	19	39

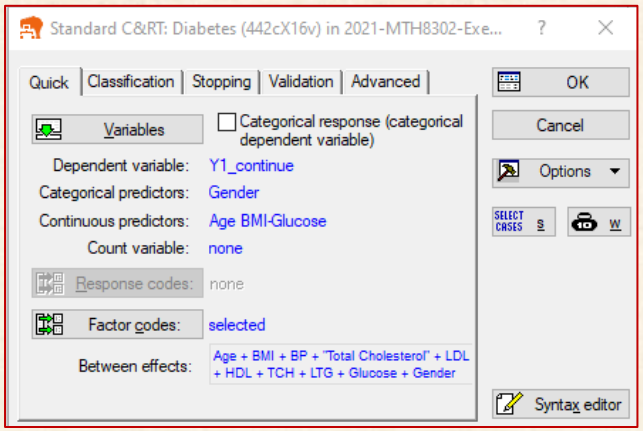


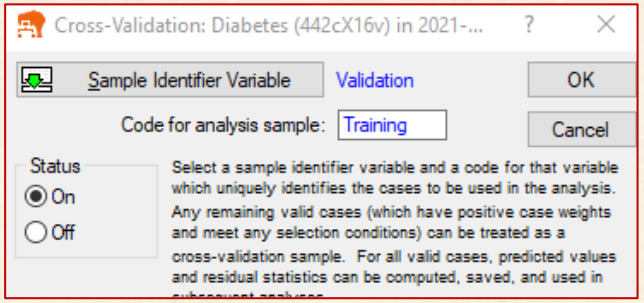


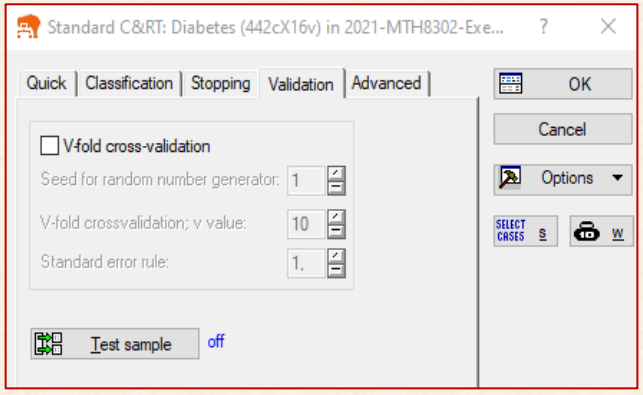
Exemple 4 data = diabètes – analyse avec STATISTICA

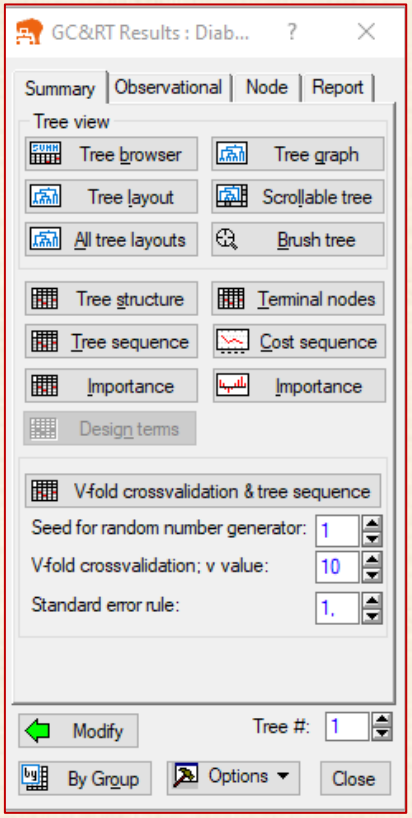




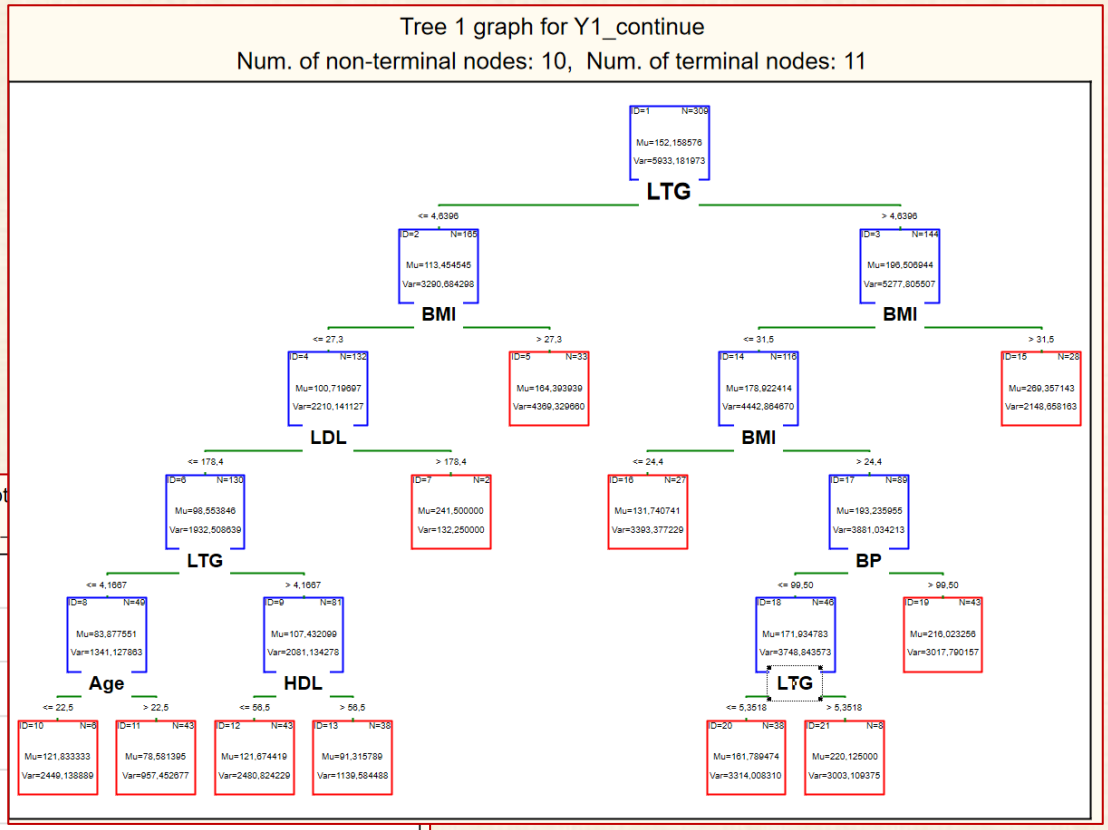


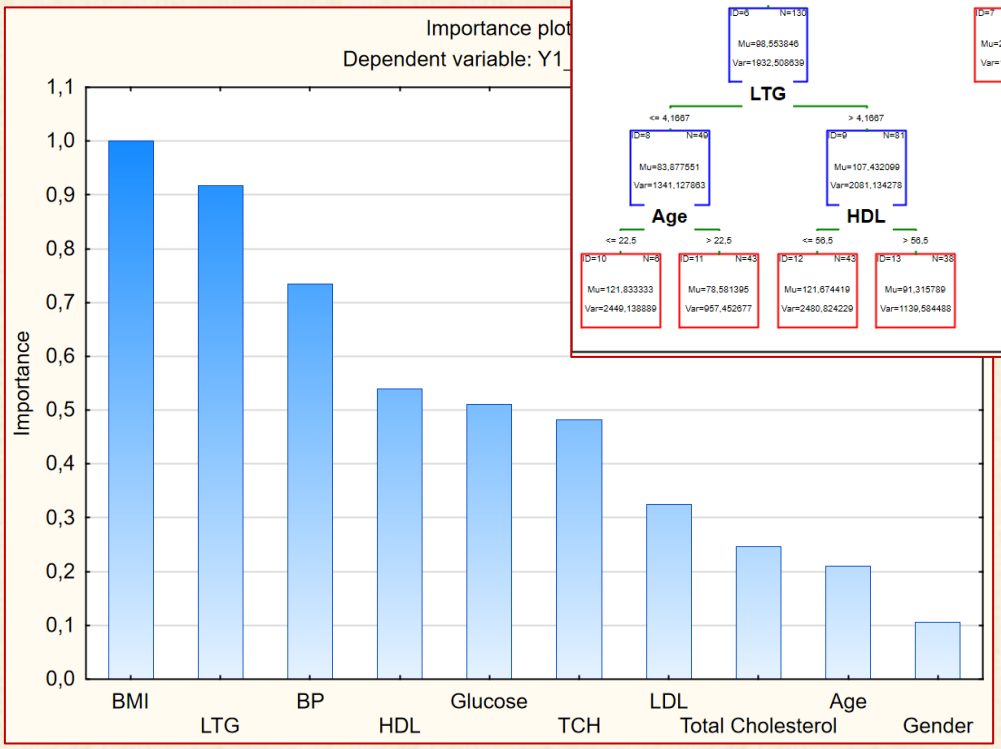




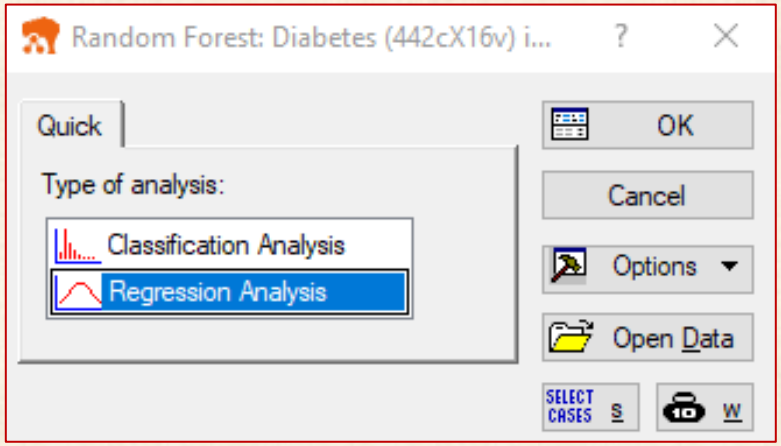


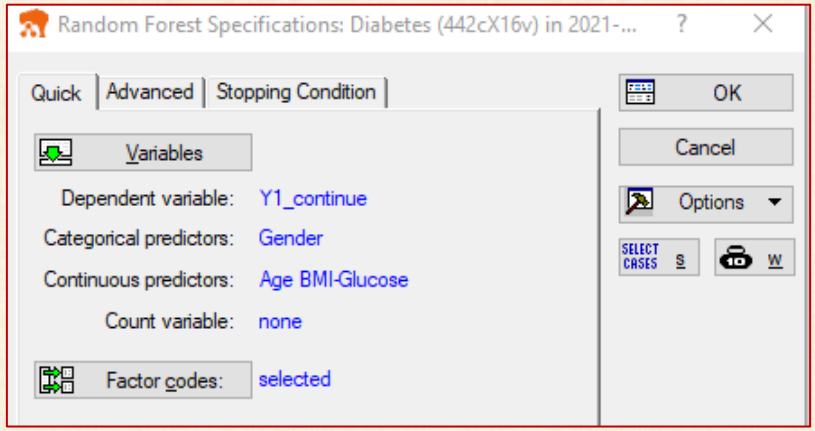
Exemple 4 data = diabètes – analyse avec STATISTICA

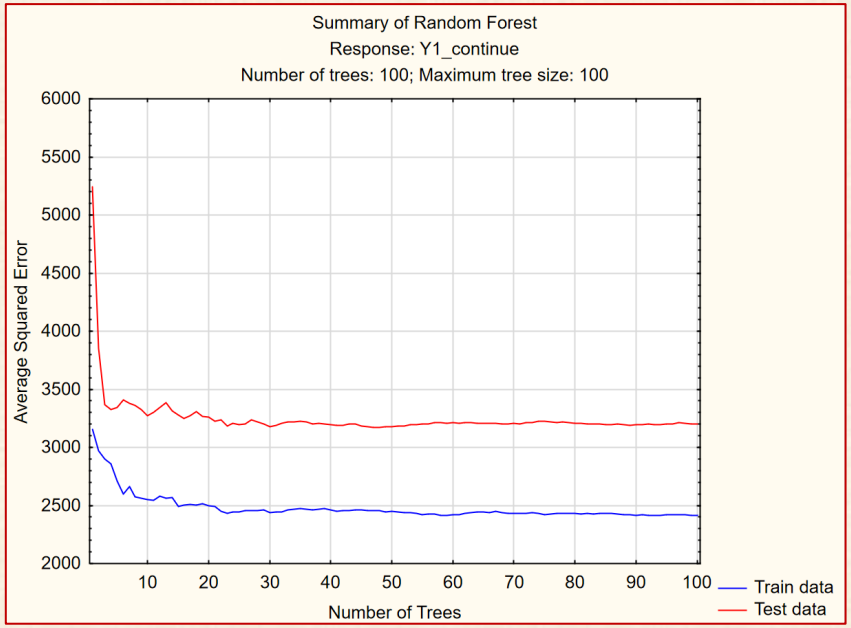


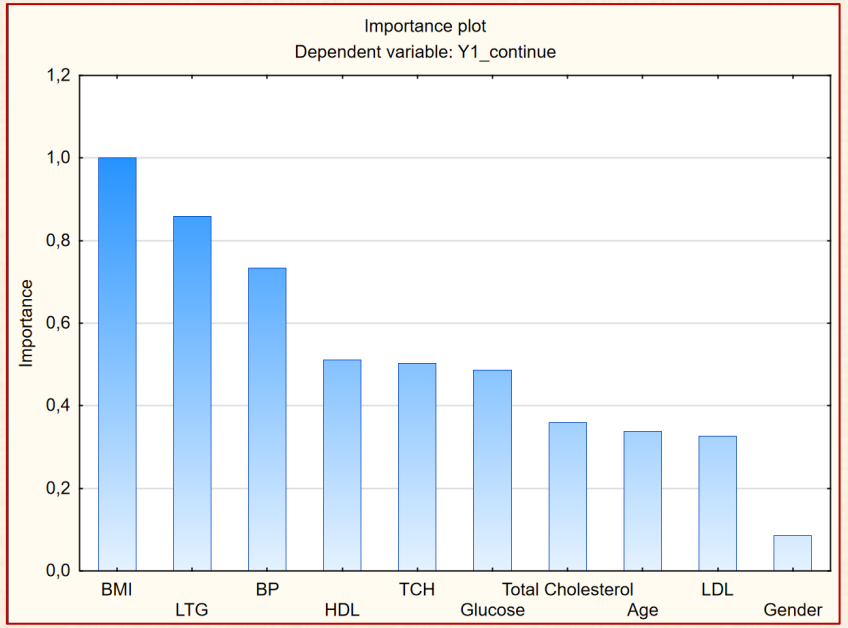


Exemple 4 data = diabètes – analyse avec STATISTICA



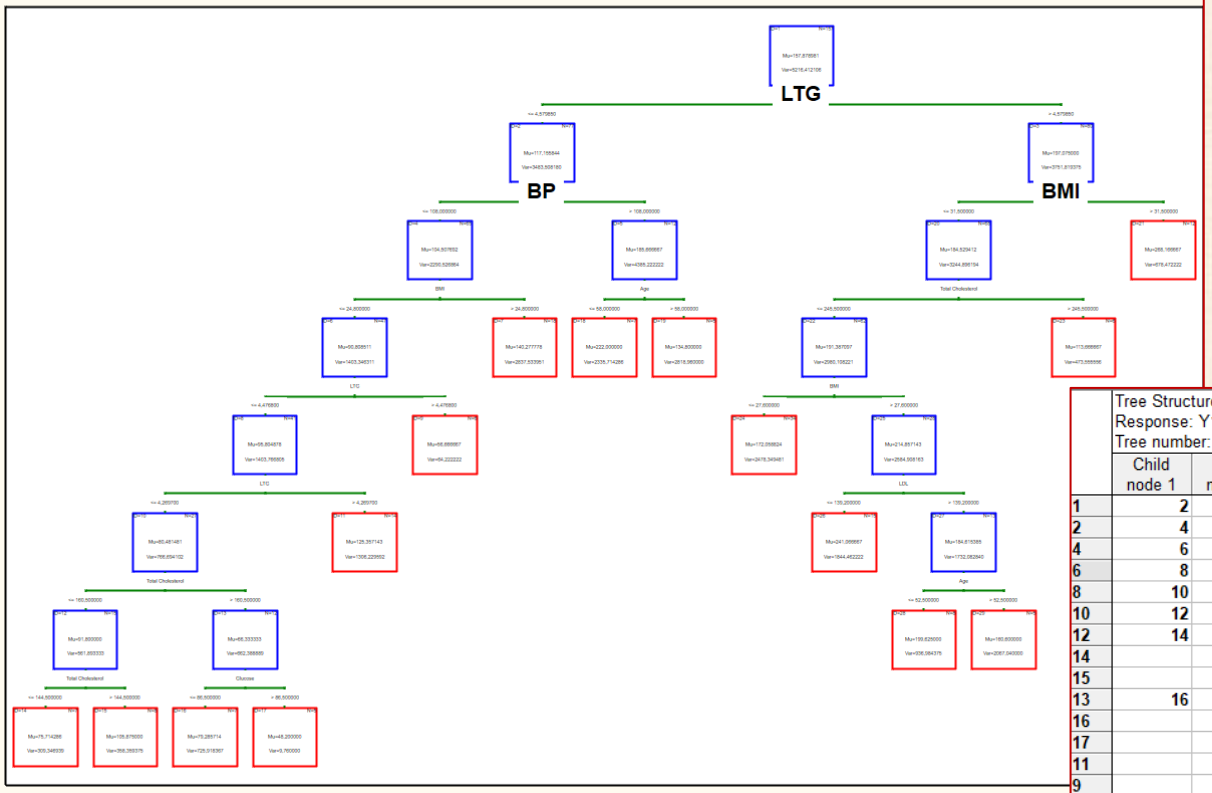






Exemple 4 data = diabètes – analyse avec STATISTICA

Tree graph for Y1_continue
Num. of non-terminal nodes: 14, Num. of terminal nodes: 15
Tree number: 1

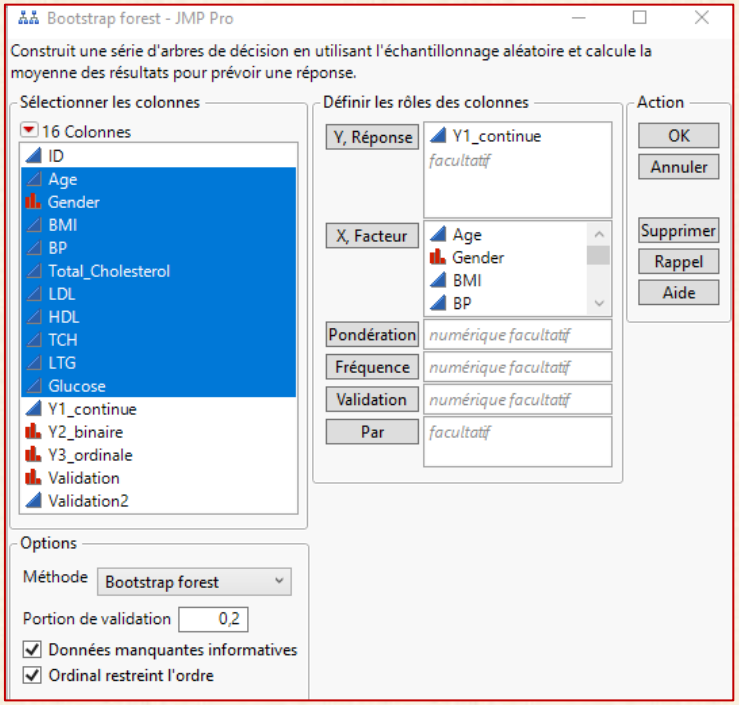


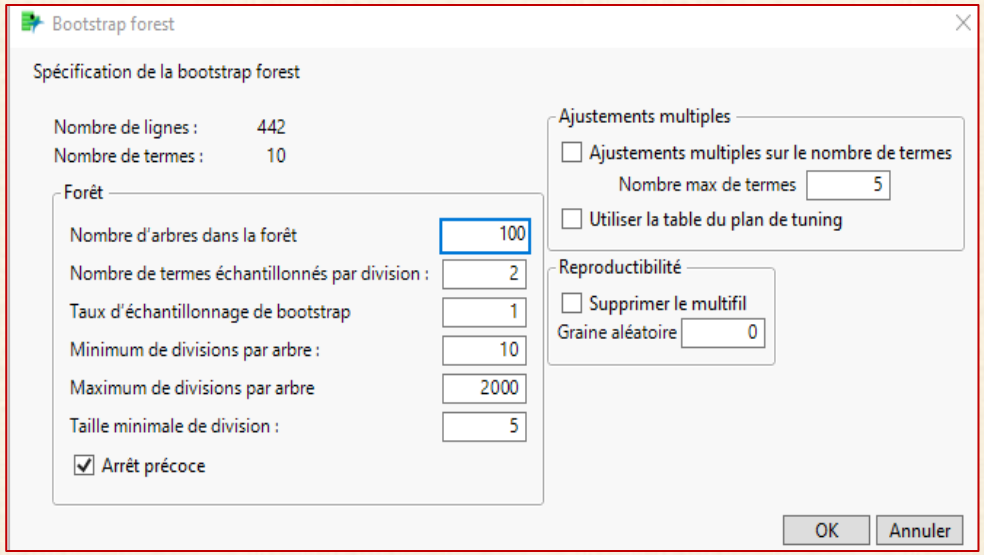
Tree Structure (Diabetes (442cX16v) in 2021-MTH8302-Exemples-MINING-CART)
Response: Y1_continue
Tree number: 1

	Child node 1	Child node 2	Size of node	Node mean	Node variance	Split variable	Split constant
1	2	3	157	157,8790	5216,412	LTG	4,5799
2	4	5	77	117,1558	3483,508	BP	108,0000
4	6	7	65	104,5077	2290,527	BMI	24,8000
6	8	9	47	90,8085	1403,346	LTG	4,4768
8	10	11	41	95,8049	1403,767	LTG	4,2697
10	12	13	27	80,4815	766,694	Total Cholesterol	160,5000
12	14	15	15	91,8000	561,893	Total Cholesterol	144,5000
14			7	75,7143	309,347		
15			8	105,8750	358,359		
13	16	17	12	66,3333	662,389	Glucose	86,5000
16			7	79,2857	725,918		
17			5	48,2000	9,760		
11			14	125,3571	1306,230		
9			6	56,6667	64,222		
7			18	140,2778	2837,534		
5	18	19	12	185,6667	4385,222	Age	58,0000
18			7	222,0000	2335,714		
19			5	134,8000	2818,960		
3	20	21	80	197,0750	3751,819	BMI	31,5000
20	22	23	68	184,5294	3244,896	Total Cholesterol	245,5000
22	24	25	62	191,3871	2980,108	BMI	27,6000
24			34	172,0588	2478,349		
25	26	27	28	214,8571	2584,908	LDL	139,2000
26			15	241,0667	1844,462		
27	28	29	13	184,6154	1732,083	Age	52,5000
28			8	199,6250	936,984		
29			5	160,6000	2067,040		
23			6	113,6667	473,556		
21			12	268,1667	678,472		

Exemple 4 data = diabètes – analyse avec JMP Pro

	ID	Age	Gender	BMI	BP	Total_Cholesterol	LDL	HDL	TCH	LTG	Glucose	Y1_continue	Y2_binaire	Y3_ordinale	Validation	Validation2
1	1	59	2	32,1	101	157	93,2	38	4	4,86	87	151	Low	Medium	Training	1
2	2	48	1	21,6	87	183	103,2	70	3	3,89	69	75	Low	Low	Validation	2
3	3	72	2	30,5	93	156	93,6	41	4	4,67	85	141	Low	Low	Training	1
4	4	24	1	25,3	84	198	131,4	40	5	4,89	89	206	High	High	Training	1
5	5	50	1	23,0	101	192	125,4	52	4	4,29	80	135	Low	Low	Training	1





Exemple 4 data = diabètes – analyse avec JMP Pro

Bootstrap forest pour Y1_continue

Spécifications

Cible	Y1_continue	Lignes d'apprentissage :	371
		Lignes de validation :	71
Nombre d'arbres dans la forêt:	100	Lignes de test :	0
Nombre de termes échantillonnés par division :	2	Nombre de termes :	10
		Échantillons de bootstrap :	371
		Minimum de divisions par arbre :	10
		Taille minimale de division :	5

Statistiques globales

Arbres individuels	Racine de l'erreur quadratique moyenne (RASE)
Dans le sachet	40,69209
Hors du sachet	71,42503

	R carré	Racine de l'erreur quadratique moyenne (RASE)	Nombre d'observations
Apprentissage	0,641	46,450194	371
Validation	0,556	48,876965	71

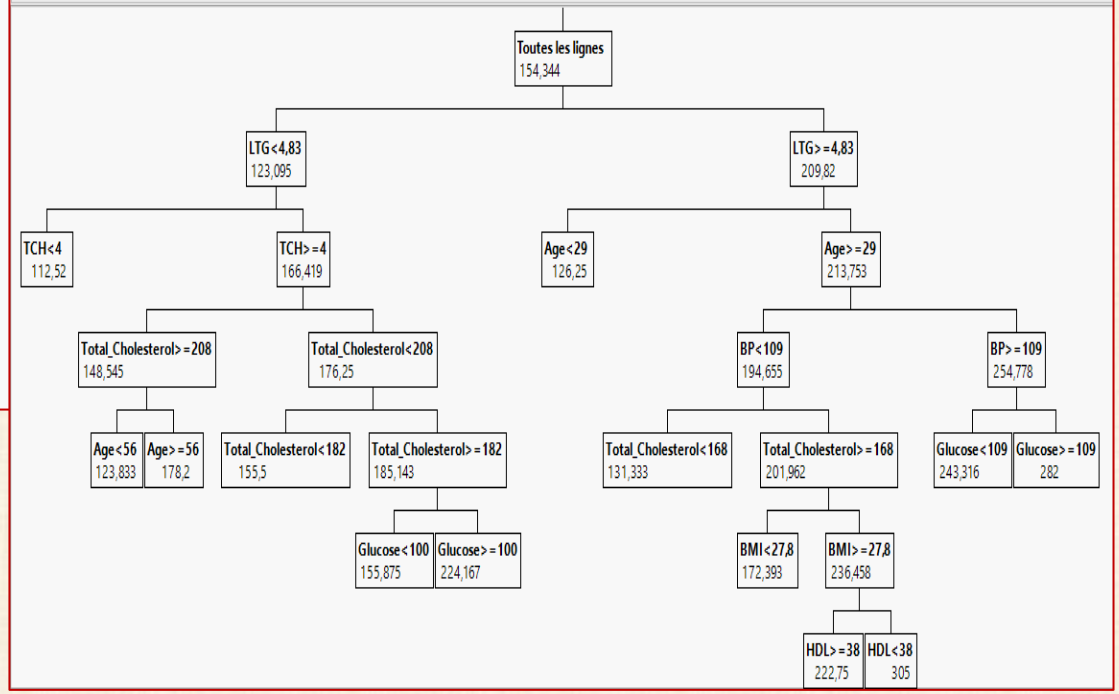
Validation cumulée

Résumés par arbre

Contributions des colonnes

Terme	Nombre de divisions	Somme des carrés	Proportion
BMI	381	203960,704	0,2513
LTG	388	182519,131	0,2249
BP	354	103268,274	0,1273
HDL	359	77538,1216	0,0955
Glucose	333	64596,1461	0,0796
TCH	258	61037,7763	0,0752
Total_Cholesterol	330	39299,7678	0,0484
LDL	289	37621,5799	0,0464
Age	292	28503,7535	0,0351
Gender	187	13183,2479	0,0162

Arbre1



Monographies & Articles

Berry, M., J., A., & Linoff, G., S., (2000). *Mastering Data Mining*. New York: Wiley

D.Hand (1999): *Why data mining is more than statistics write large*, ISI, Helsinki http://www.stat.fi/isi99/index.html

D.Hand (2000): *Methodological Issues in Data Mining*, in Compstat 2000, Physica-Verlag, 77-85, 2000

Edelstein, H., A. (1999). *Introduction to Data Mining and Knowledge Discovery (3rd ed)*. Potomac, MD: Two Crows Corp.

Fayyad, U. M., Piatetsky-Shapiro, G., Smyth, P., & Uthurusamy, R. (1996). *Advances In Knowledge Discovery & Data Mining*. Cambridge, MA: MIT Press.

Friedman J. (1997): *Data Mining and Statistics, What's the Connection?* http://www-stat.stanford.edu/~jhf/ftp/dm-stat.ps

Friedman J. (1999): *The role of Statistics in Data Revolution*, ISI, Helsinki, http://www.stat.fi/isi99/index.html

Gaudard, M. Ramsey, P., Stephens, M. ((2006). *Interactive Data Mining and Design of Experiments: The JMP Partition and Custom Design Platforms*. North Haven Group, LLC

Giudici, P. (2003*). Applied Data Mining: Statistical Methods for Industry*, John Wiley & Sons.

Han, J., Kamber, M. (2000). *Data Mining: Concepts and Techniques*. New York: Morgan-Kaufman.

Hastie, T., Tibshirani, R., & Friedman, J. H. (2001). *The elements of Statistical Learning : Data Mining, Inference, and Prediction*. New York: Springer.

Kantardzic, M.M., Zurada, J. (editors) (2005). *Next Generation of Data-Mining Applications.* John Wiley & Sons, Copyright the Institute of electrical and Electronic Enginineers (IEEE).

Larocque, Denis (2021) Random Forest as a Weight-Generating Machine for Local Estimation and Prediction, conférence IVADO

Nisbet R., Elder, J., Miner, G. (2009*) Handbook of Statistical Analysis & Data Mining Apllications*, Academic Press. ISBN 978-0-12-374765-5

Pregibon, D. (1997). *Data Mining*. Statistical Computing and Graphics, 7, 8.

StatSoft : 35 vidéos de 8-10 minutes sur YouTube http://www.statsoft.com/support/download/video-tutorials/

Tufféry, S. (2007). *Data Mining et statistique décisionnelle*, Éditions TECHNIP, Paris.

Weiss, S. M., & Indurkhya, N. (1997). *Predictive Data Mining: A practical Guide*. New York: Morgan-Kaufman.

Westphal, C., Blaxton, T. (1998). *Data Mining Solutions*. New York: Wiley.

Witten, I. H., & Frank, E. (2000). *Data Mining*. New York: Morgan-Kaufmann.