1 Objectif

Comparer les fonctionnalités de plusieurs logiciels libres de Data Mining lors d'une typologie à l'aide des K-Means.

La méthode des K-Means (méthode des centres mobiles) est une technique de classification automatique (clustering en anglais). Elle vise à produire un regroupement de manière à ce que les individus du même groupe soient semblables, les individus dans des groupes différents soient dissemblables.

Nous l'avons déjà décrite (http://tutoriels-data-mining.blogspot.com/search?q=k-means) par ailleurs. Notre idée dans ce didacticiel est de montrer sa mise en œuvre dans différents logiciels libres de Data Mining. Nous souhaitons utiliser la démarche suivante :

- Importer les données ;
- Réaliser quelques statistiques descriptives sur les variables actives ;
- Centrer et réduire les variables ;
- Réaliser la classification automatique via les K-Means sur les variables transformées, en décidant nous même du nombre de classes ;
- Visualiser les données avec la nouvelle colonne représentant la classe d'appartenance des individus ;
- Illustrer les classes à l'aide des variables actives, via des statistiques descriptives comparatives et/ou des graphiques judicieusement choisis ;
- Croiser la partition obtenue avec une variable catégorielle illustrative ;
- Exporter les données, avec la colonne additionnelle, dans un fichier.

Ces étapes sont usuelles lors de la construction d'une typologie. L'intérêt de ce didacticiel est de montrer qu'elles **sont pour la plupart**, sous des formes parfois diverses certes, **réalisables dans les logiciels libres de Data Mining**. Il faut simplement trouver les bons composants et le bon enchaînement. Nous étudierons les logiciels suivants : **Tanagra 1.4.28** ; **R 2.7.2** (sans package additionnel spécifique) ; **Knime 1.3.5** ; **Orange 1.0b2** et **RapidMiner Community Edition**.

Nous utilisons la méthode des centres mobiles dans ce tutoriel. Il est possible de suivre la même démarche globale en lui substituant n'importer quelle autre technique de classification automatique (la classification ascendante hiérarchique, les cartes de Kohonen, etc.).

Bien évidemment, je ne peux prétendre maîtriser complètement les différents logiciels. Il se peut que des fonctionnalités m'échappent pour certains d'entre eux. Il faut surtout voir les grandes lignes et le parallèle entre les outils, les experts pourront compléter les opérations à leur guise.

2 Données

Nous utilisons le fichier « cars_dataset.txt »[1], un fichier texte avec séparateur tabulation. Il décrit les caractéristiques de 392 véhicules. Les variables actives qui participeront au calcul sont : la consommation (MPG, miles per galon, plus le chiffre est élevé, moins la voiture consomme) ; la taille du moteur (DISPLACEMENT) ; la puissance (HORSEPOWER) ; le poids (WEIGHT) et l'accélération (ACCELERATION, le temps mis pour atteindre une certaine vitesse, plus le chiffre est faible plus la voiture est performante).

[1] http://eric.univ-lyon2.fr/~ricco/tanagra/fichiers/cars_dataset.zip ; le fichier originel provient du serveur STATLIB, http://lib.stat.cmu.edu/datasets/cars.desc

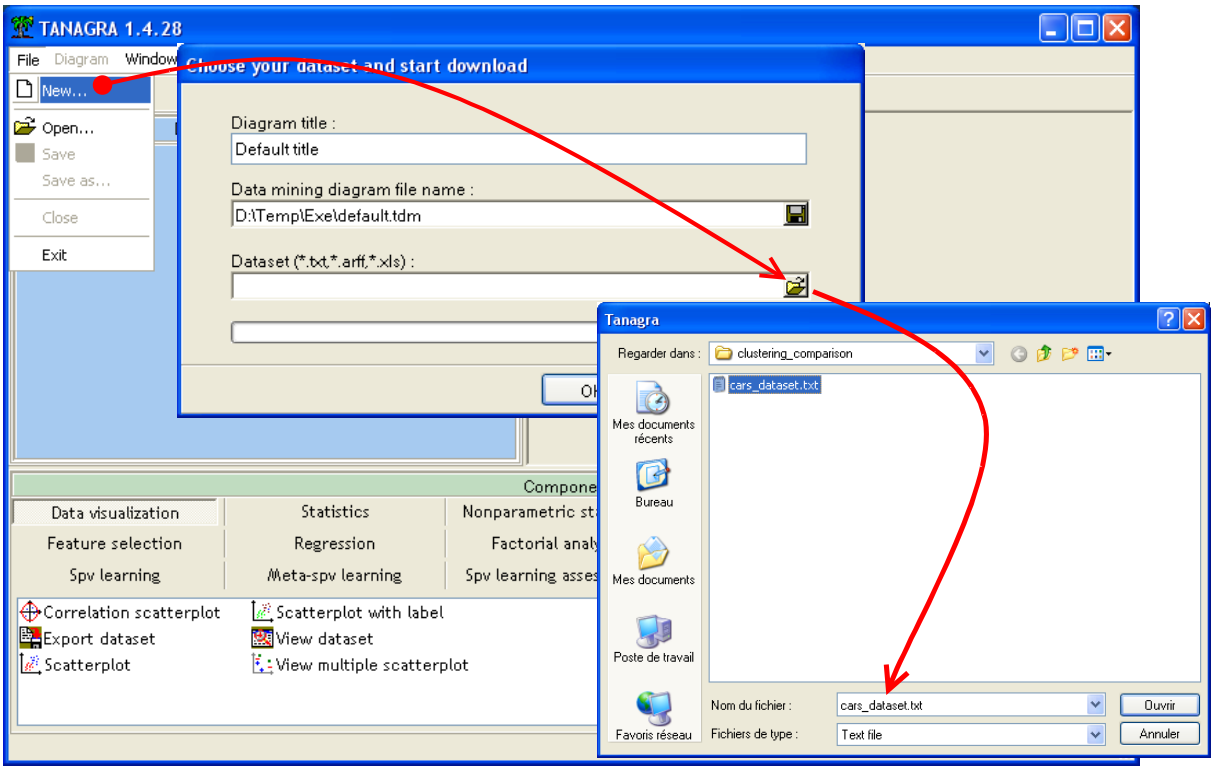
La variable illustrative « origine des véhicules » (ORIGIN : Japon, Europe, Etats Unis) servira à renforcer l'interprétation des groupes.

3 K-Means avec TANAGRA

Les traitements réalisés dans TANAGRA nous serviront de trame. Pour cette raison, nous détaillons chacune des étapes dans cette première section. Nous irons à l'essentiel en ce qui concerne les autres logiciels.

3.1 Création d'un diagramme et importation des données

Après avoir démarré TANAGRA, nous créons un diagramme à l'aide du menu FILE / NEW. Nous spécifions le fichier de données CARS_DATASET.TXT.



TANAGRA nous indique que 392 observations et 6 variables ont été importées.

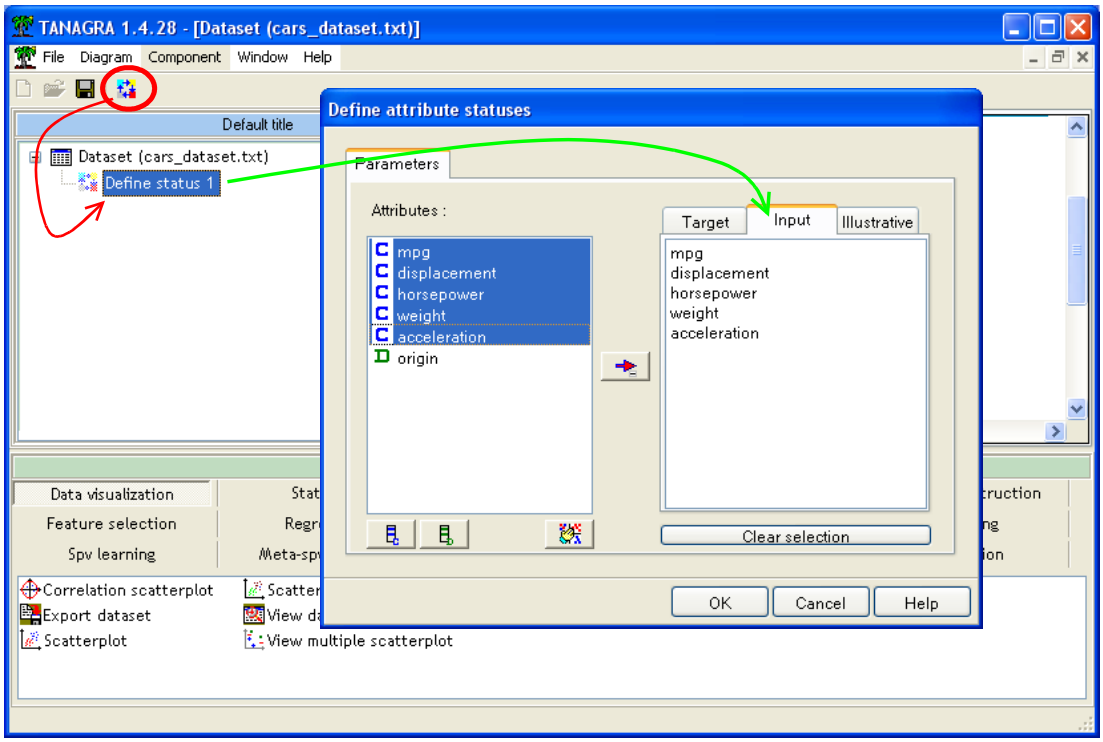
Dataset description

6 attribute(s)
392 example(s)

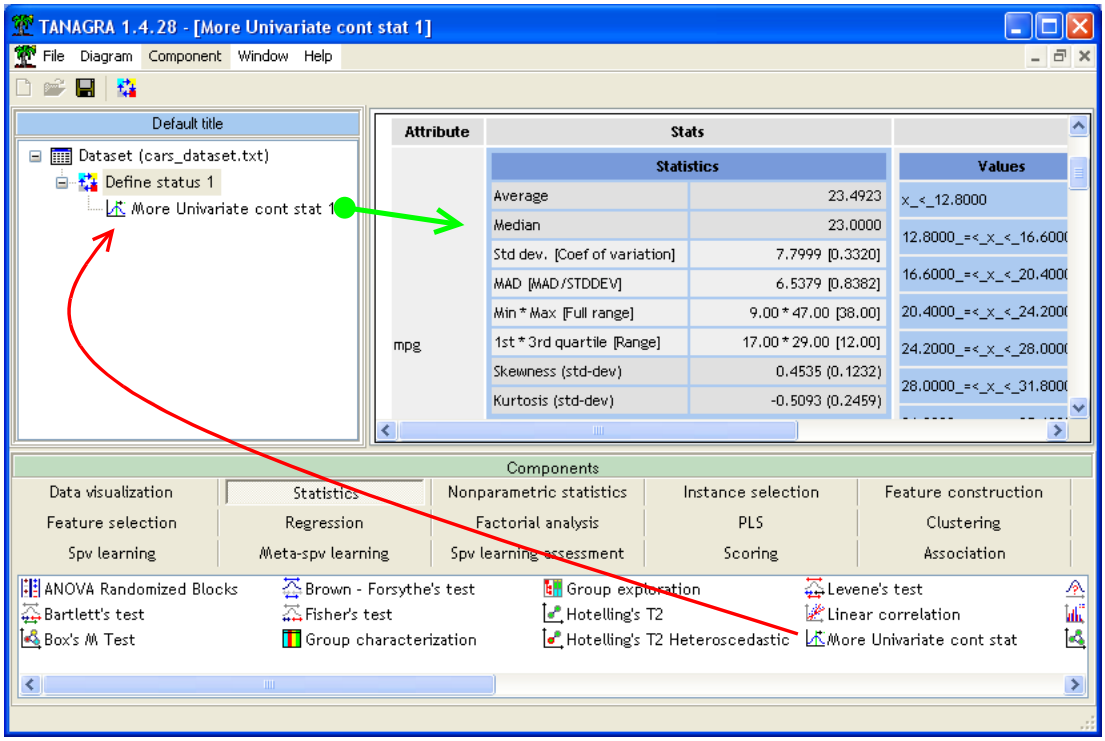
Attribute	Category	Informations
mpg	Continue	-
displacement	Continue	-
horsepower	Continue	-
weight	Continue	-
acceleration	Continue	-
origin	Discrete	3 values

3.2 Statistiques descriptives

Première étape, nous calculons quelques indicateurs de statistique descriptive sur les données. L'idée est de détecter les éventuelles anomalies. Nous insérons le composant DEFINE STATUS via le raccourci dans la barre d'outils. Nous plaçons les variables continues en INPUT. Il s'agit des variables actives de l'analyse.



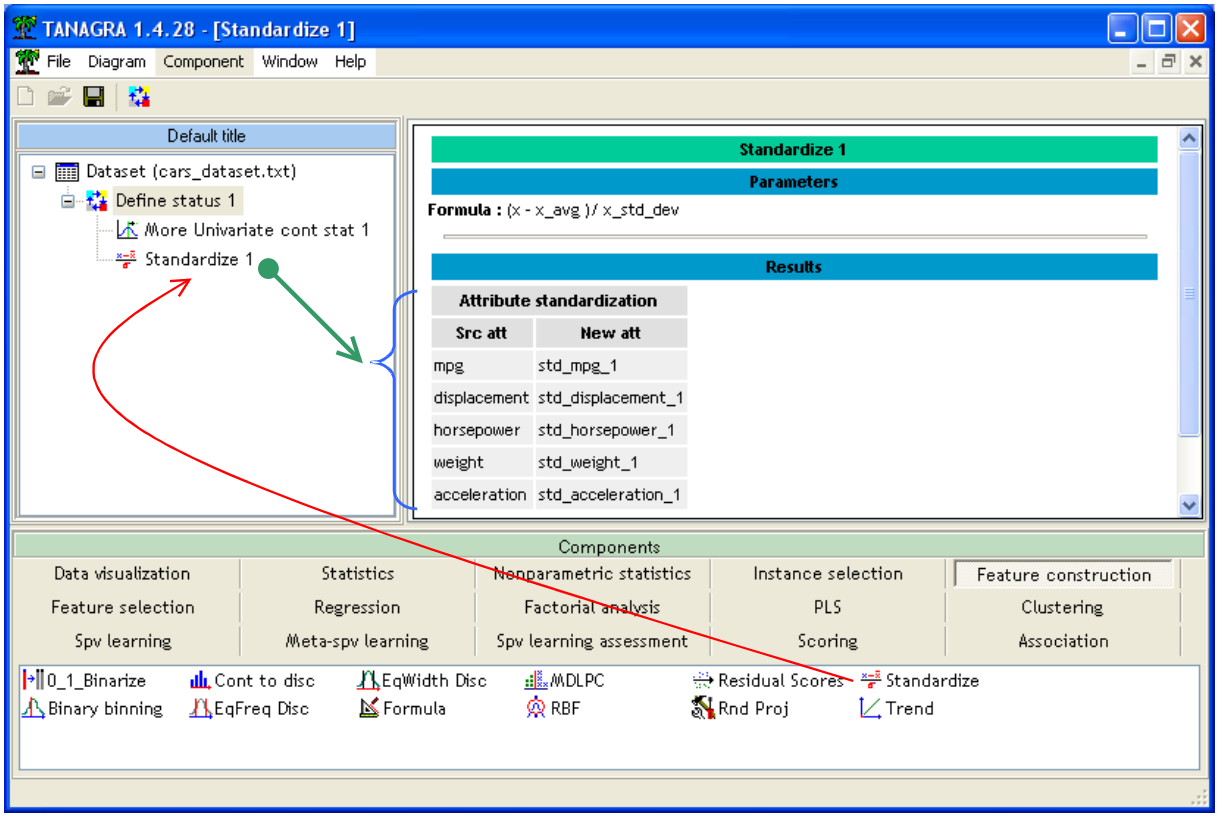
Nous insérons à la suite le composant MORE UNIVARIATE CONT STAT (onglet STATISTICS). Nous actionnons le menu contextuel VIEW.



Il n'y a pas de données franchement atypiques ni d'asymétrie marquée, rien en tous les cas qui justifie un pré traitement spécifique.

3.3 Centrer et réduire les variables actives

Nous souhaitons centrer et réduire les variables avant de le présenter à la méthode des K-MEANS, ceci afin d'éliminer les disparités d'échelle[2]. Nous introduisons le composant STANDARDIZE (onglet FEATURE CONSTRUCTION) dans le diagramme, à la suite de DEFINE STATUS 1. Nous actionnons le menu contextuel VIEW.

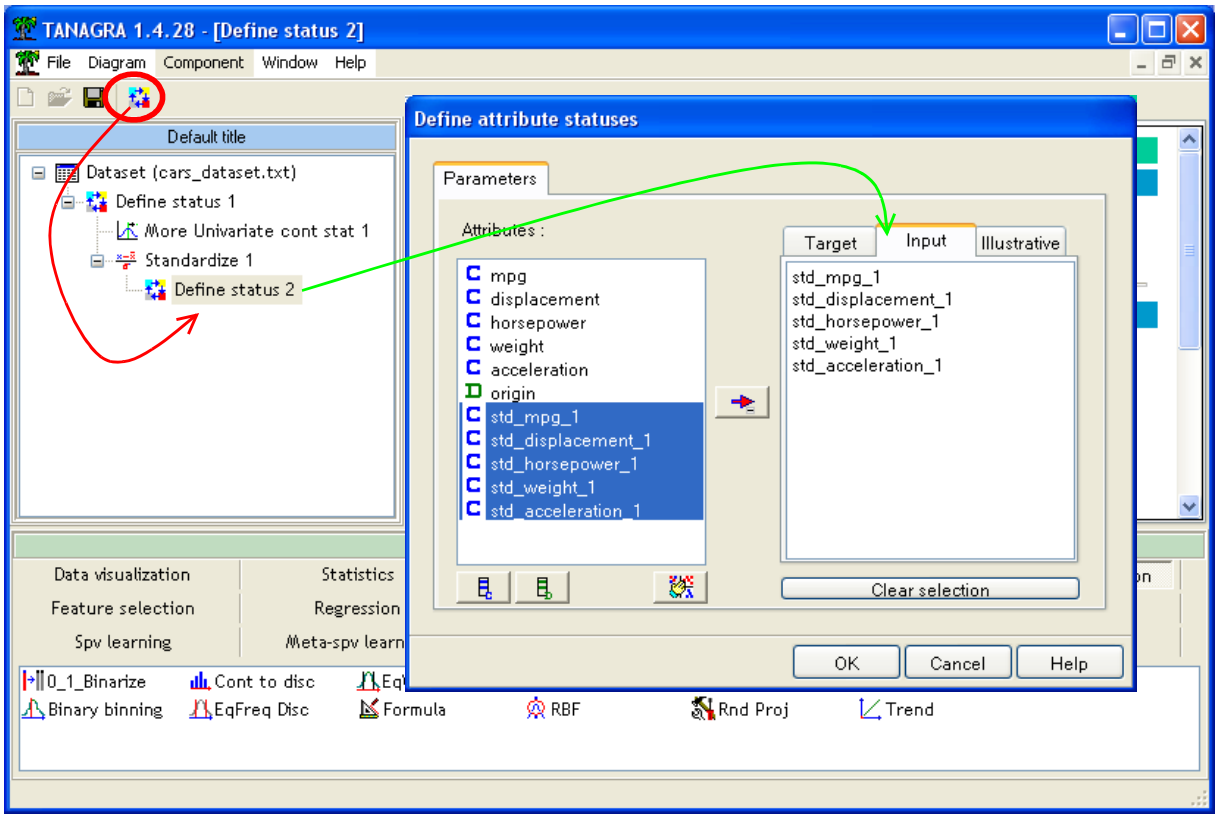


5 nouvelles variables intermédiaires sont maintenant disponibles pour les traitements ultérieurs.

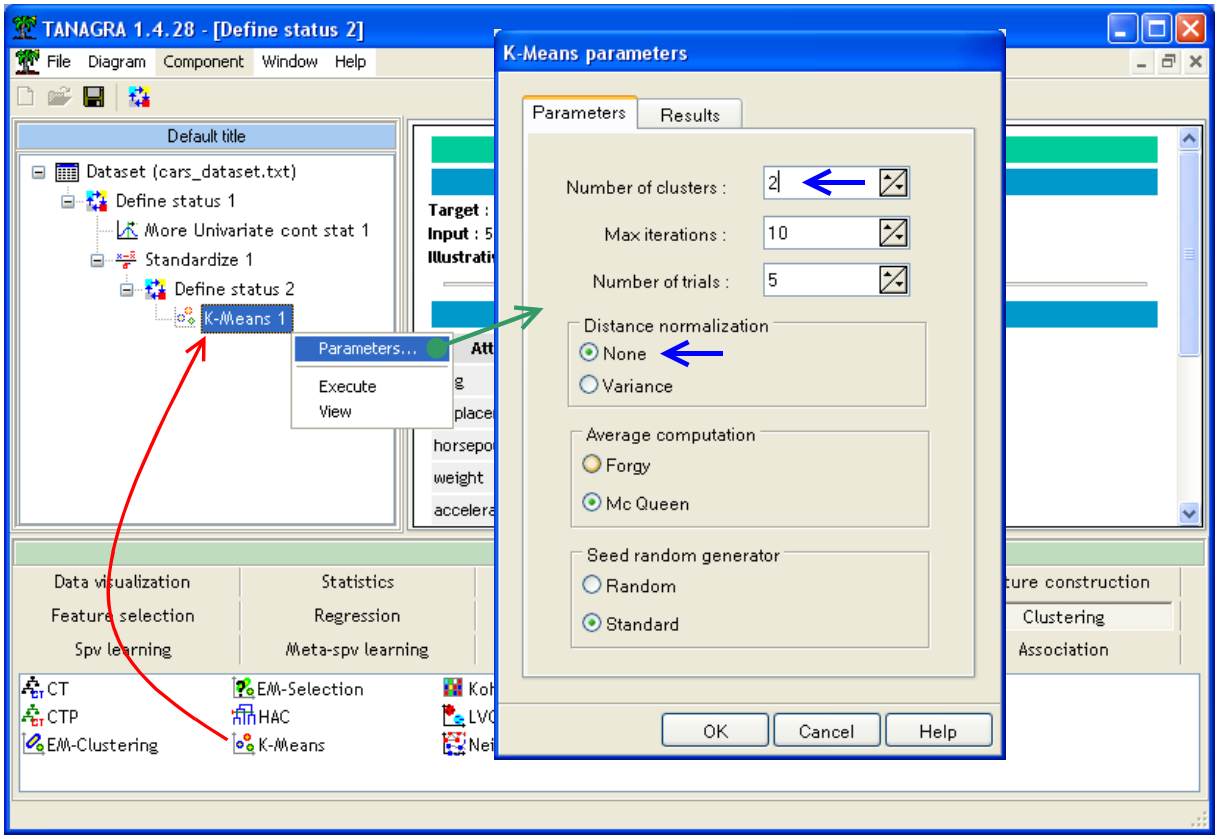
3.4 K-Means

Nous devons préciser à TANAGRA que ce sont ces variables transformées qui seront utilisées pour les calculs. Nous insérons un nouveau DEFINE STATUS, nous plaçons en INPUT les variables STD_MPG_1 à STD_ACCELERATION_1.

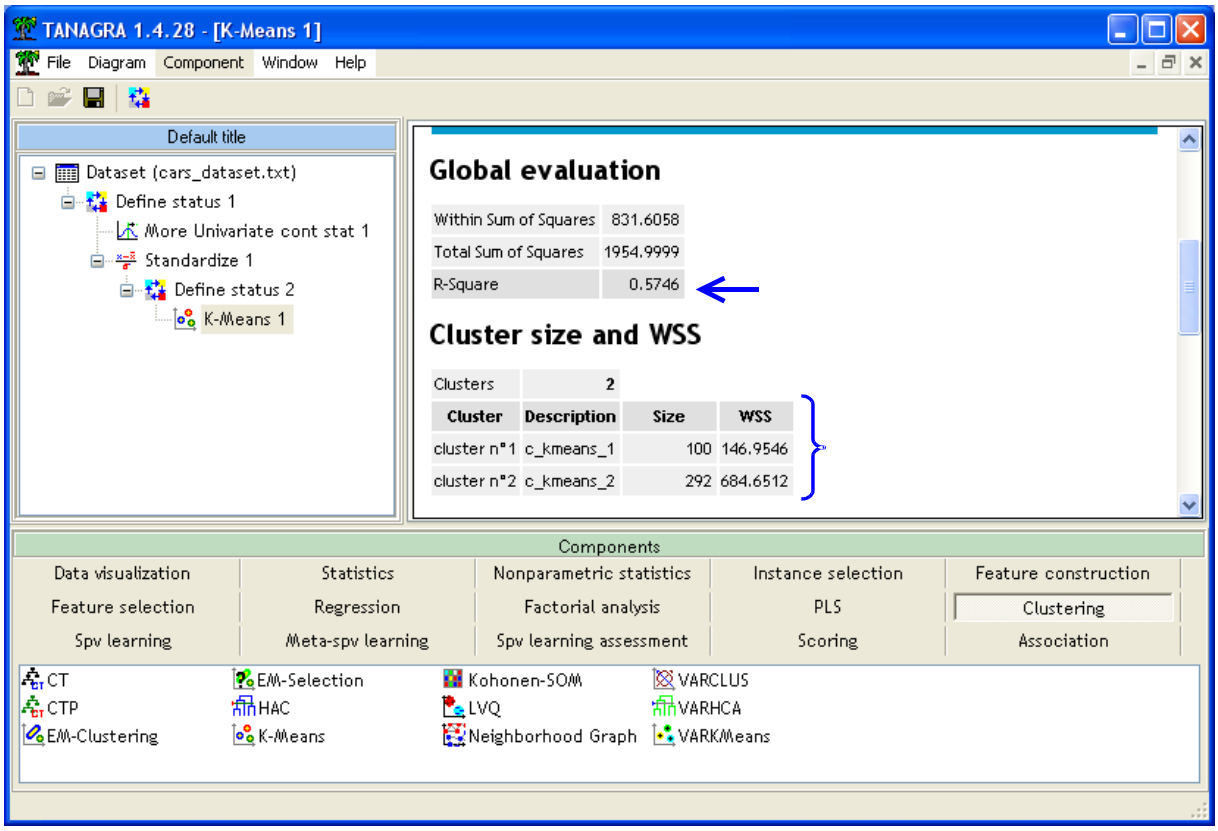
[2] En réalité, cette opération n'est pas indispensable dans TANAGRA. Le composant K-MEANS sait normaliser automatiquement les données si l'utilisateur le demande. Nous introduisons néanmoins cette étape car nous avons l'obligation de le faire explicitement dans les autres logiciels.



Nous pouvons introduire le composant K-MEANS (onglet CLUSTERING). Nous actionnons dans un premier temps le menu PARAMETERS pour spécifier les paramètres de traitement.



Nous demandons une partition en 2 groupes. Il n'est pas nécessaire de normaliser la distance puisque les variables sont déjà réduites. Nous validons, puis nous cliquons sur le menu VIEW.



TANAGRA nous annonce qu'il y a respectivement 292 et 100 observations dans chaque groupe. La partition explique 57.46% de l'inertie totale.

Dans la partie basse de la fenêtre de visualisation, Tanagra affiche les moyennes conditionnelles sur les variables ayant participé à la construction de la partition. Elles sont donc calculées sur les données centrées et réduites. Elles permettent de comprendre les différenciations entre les groupes mais, n'étant pas exprimées dans les unités des variables initiales, elles ne sont pas vraiment utilisables pour l'interprétation.

Cluster centroids

Attribute	Cluster n°1	Cluster n°2
std_mpg_1	-1.120825	0.383844
std_displacement_1	1.451760	-0.497178
std_horsepower_1	1.459572	-0.499853
std_weight_1	1.355324	-0.464152
std_acceleration_1	-1.018068	0.348653

Use GROUP CHARACTERIZATION for detailed comparisons

Tout ce que l'on peut dire à ce stade est que, pour toutes les variables, les écarts vont au-delà de 1 écart type, voire 2 pour certaines d'entre elles (DISPLACEMENT, HORSEPOWER, WEIGHT).

3.5 Interprétation des groupes

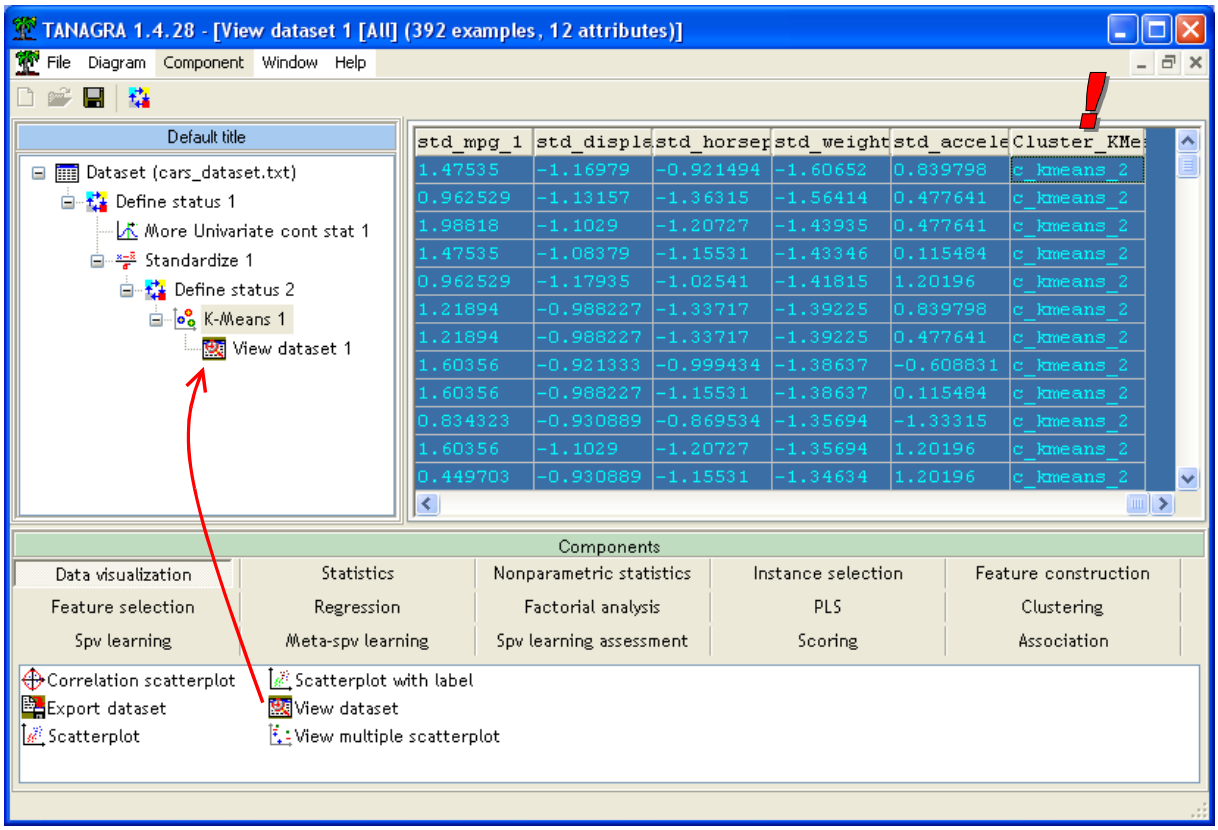
Arrive la partie délicate de la typologie : l'interprétation des résultats. Nous abordons ici les différentes pistes qui permettent de comprendre la formation des groupes.

3.5.1 Affectation des classes aux individus

Première approche, revenir aux données en inspectant l'affectation des classes aux individus. Elle est réellement viable si les effectifs sont faibles et les individus identifiés (cela pourrait être la marque et le modèle du véhicule dans notre exemple).

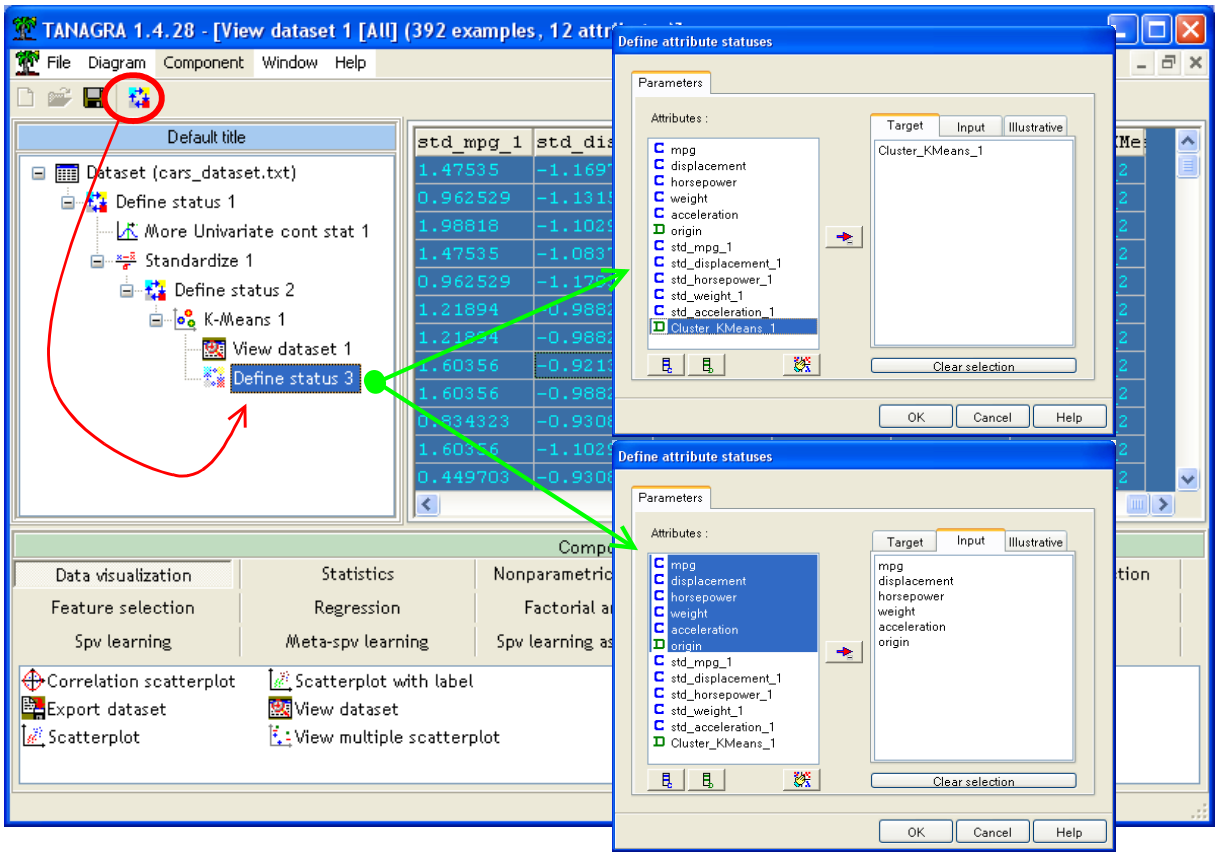
TANAGRA produit automatiquement une variable supplémentaire dans la base courante. Elle décrit la classe affectée à chaque individu. Nous pouvons la visualiser avec le composant VIEW DATASET (onglet DATA VISUALIZATION). Elle est positionnée en dernière colonne.

L'intérêt de cet outil est très limité dès que la base va au-delà de quelques dizaines d'observations.

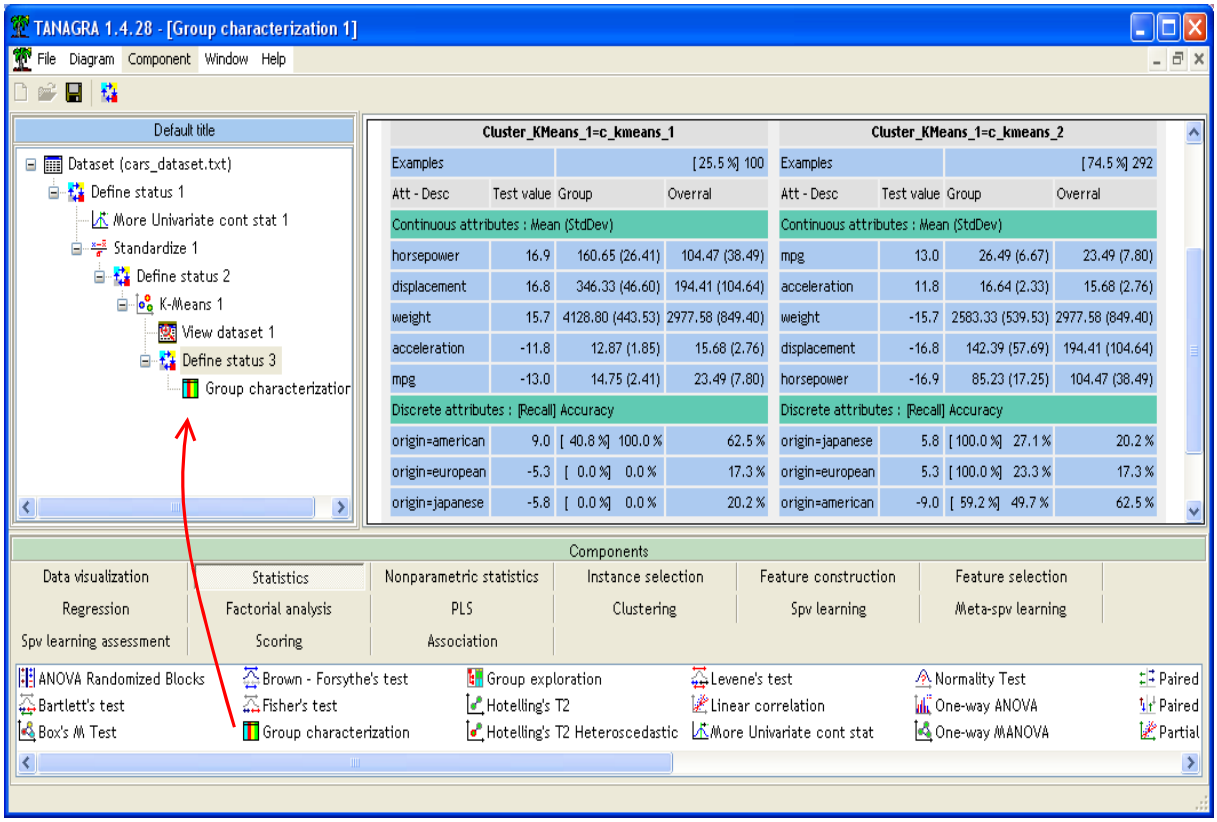


3.5.2 Statistiques descriptives comparatives

Seconde approche, très simple et pourtant très instructive, nous pouvons calculer les statistiques descriptives conditionnelles sur les variables actives et illustratives. Les oppositions permettent souvent de situer les caractéristiques marquantes des groupes.



Nous introduisons le composant DEFINE STATUS dans le diagramme. Nous plaçons en TARGET la variable désignant les classes CLUSTER_KMEANS_1, en INPUT les variables actives originelles MPG à ACCELERATION, et la variable illustrative catégorielle ORIGIN. Puis, nous insérons le composant GROUP CHARACTERIZATION (onglet STATISTICS).

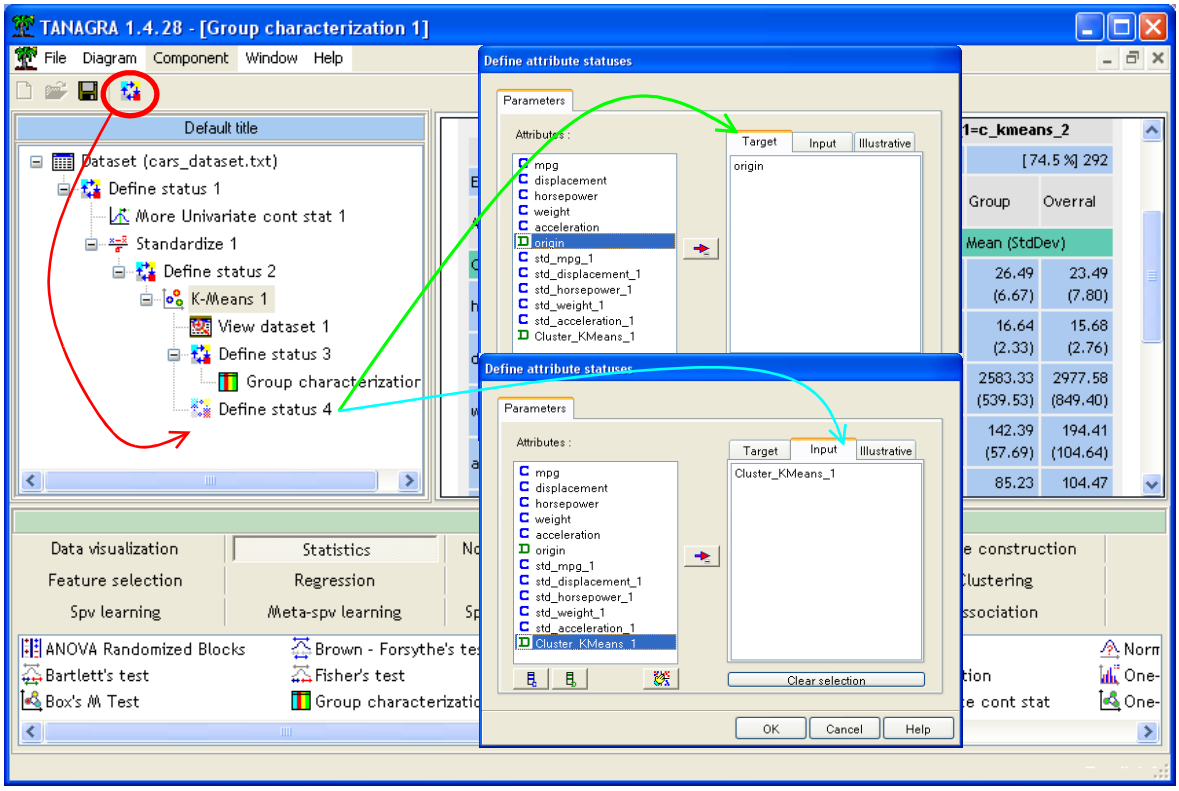


Nous constatons que le premier groupe C_K_MEANS_1 correspond plutôt aux véhicules moyens ou petits : consommant peu (le MPG moyen est de 26.66 dans le groupe, il est de 23.49 dans la totalité du fichier), pas très vives (ACCELERATION prend une valeur plus élevée, c.-à-d. met plus de temps à atteindre une certaine vitesse), avec un petit moteur (DISPLACEMENT) et peu puissantes (HORSEPOWER). Pour qualifier l'importance des écarts, TANAGRA utilise la valeur test, explicitée en détail dans un de nos didacticiels (http://tutoriels-data-mining.blogspot.com/2008/04/interprter-la-valeur-test.html).

Le principal intérêt de GROUP CHARACTERIZATION est qu'il permet d'introduire à la fois les variables actives et illustratives, qu'elles soient quantitatives ou qualitatives. Dans notre exemple, la variable ORIGIN permet de mieux comprendre les classes. Le 1er groupe est composé pour moitié (48.6%) de voitures américaines et pour le reste (27.6% + 23.8% = 51.4%) des voitures d'autres origines. En revanche, 100% des véhicules japonais se trouvent dans ce groupe, il en est de même pour les voitures européennes. La 2nde classe est formée exclusivement de véhicules américains, lourds, puissants, avec une consommation élevée. Il s'agit de véhicules des années 70-80 vendus aux Etats-Unis. Ce contexte éclaire mieux cette opposition.

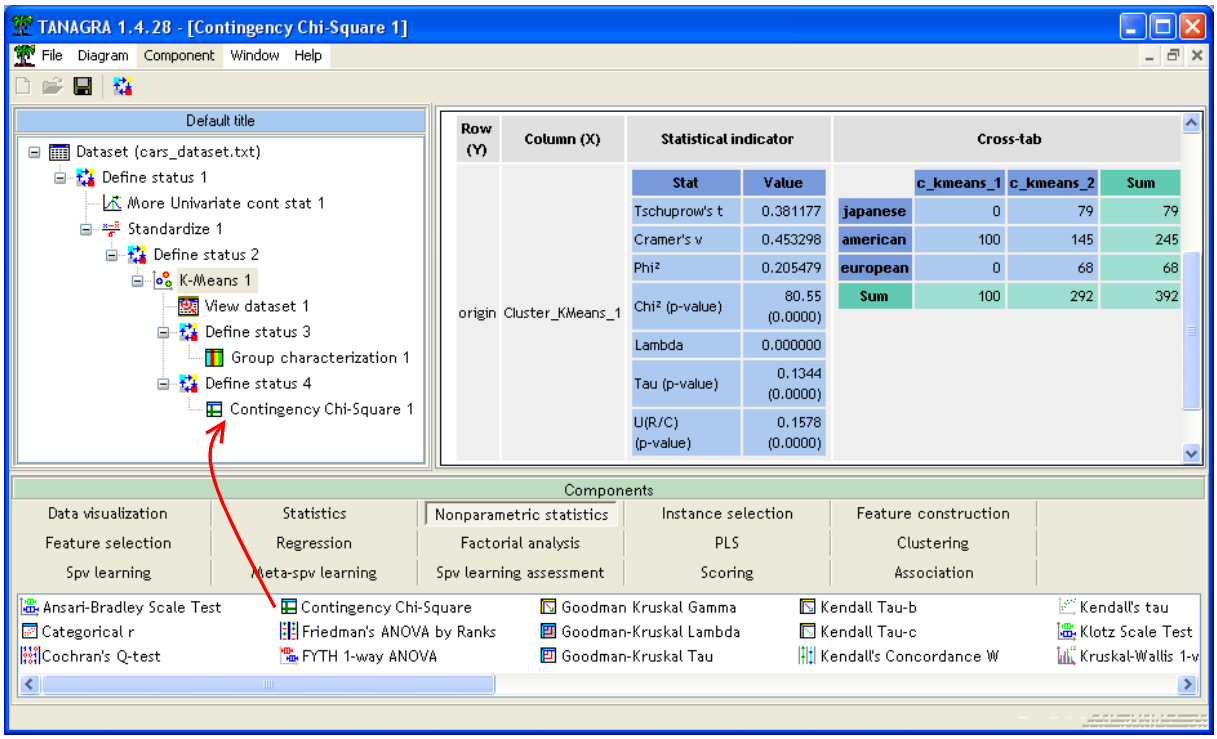
3.5.3 Croisement avec une variable catégorielle illustrative

Il existe une autre manière de mettre en exergue le lien entre les groupes et les variables illustratives catégorielles dans TANAGRA. Nous introduisons un nouveau DEFINE STATUS, nous plaçons en TARGET la variable ORIGIN, en INPUT l'indicatrice des classes C_KMEANS_1.



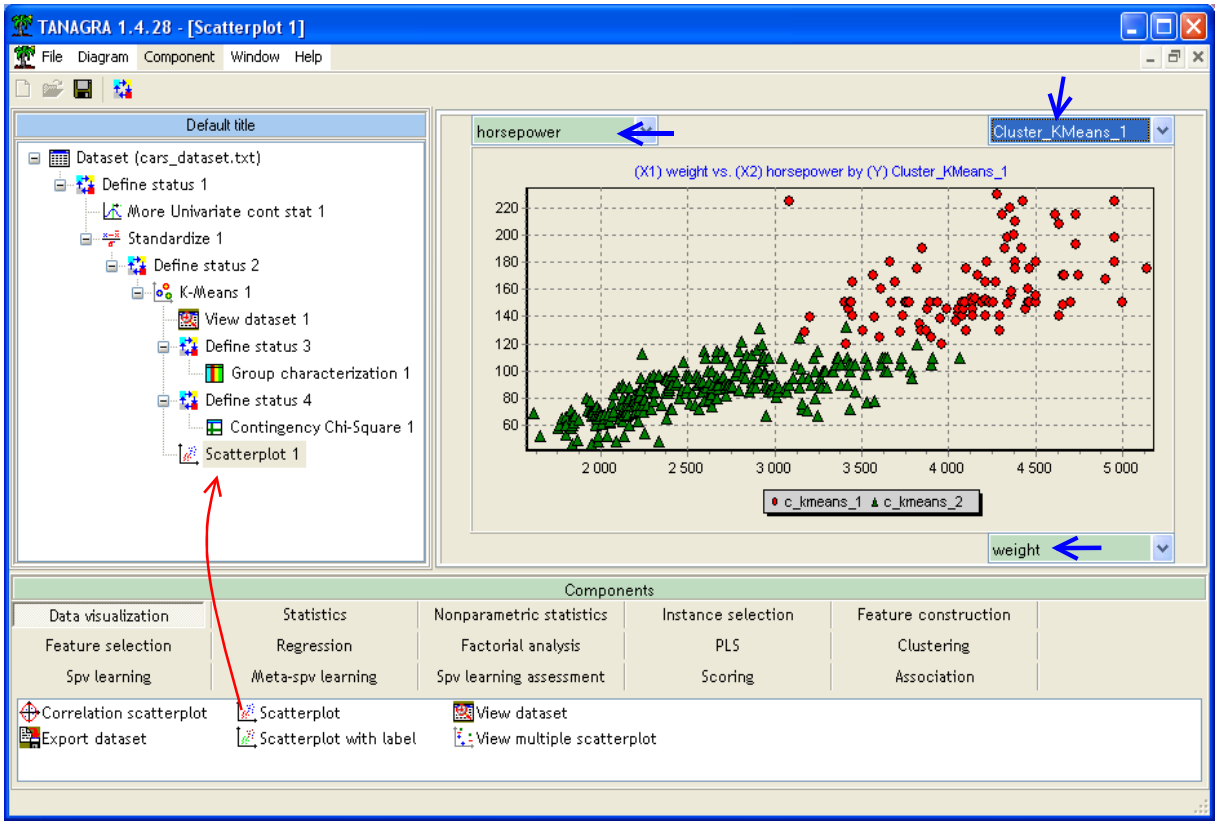
Nous plaçons le composant CONTINGENCY CHI-SQUARE (onglet NONPARAMETRIC STATISTICS) dans le diagramme. Nous actionnons le menu VIEW.

Le résultat est bien évidemment cohérent avec celui de GROUP CHARACTERIZATION. L'intérêt est que nous disposons de surcroît des indicateurs de liaison entre les variables (v de Cramer, etc.). Nous pouvons aussi multiplier les points de vue en demandant les profils lignes, colonnes, ou la contribution aux KHI-2 des cellules.



3.5.4 Graphique « Nuage de points »

Une autre manière d'interpréter les résultats est de positionner les groupes dans l'espace des couples de variables. On peut ainsi analyser l'action conjointe de deux variables. Le graphique « nuage de points » est un outil privilégié pour cela.



Dans TANAGRA, il suffit d'introduire l'outil SCATTERPLOT (onglet DATA VISUALIZATION). Nous cliquons sur VIEW. Dans l'outil de visualisation, nous mettons en abscisse le poids (WEIGHT) et en ordonnée la puissance (HORSEPOWER). Il nous reste à coloriser les données selon leur classe d'appartenance.

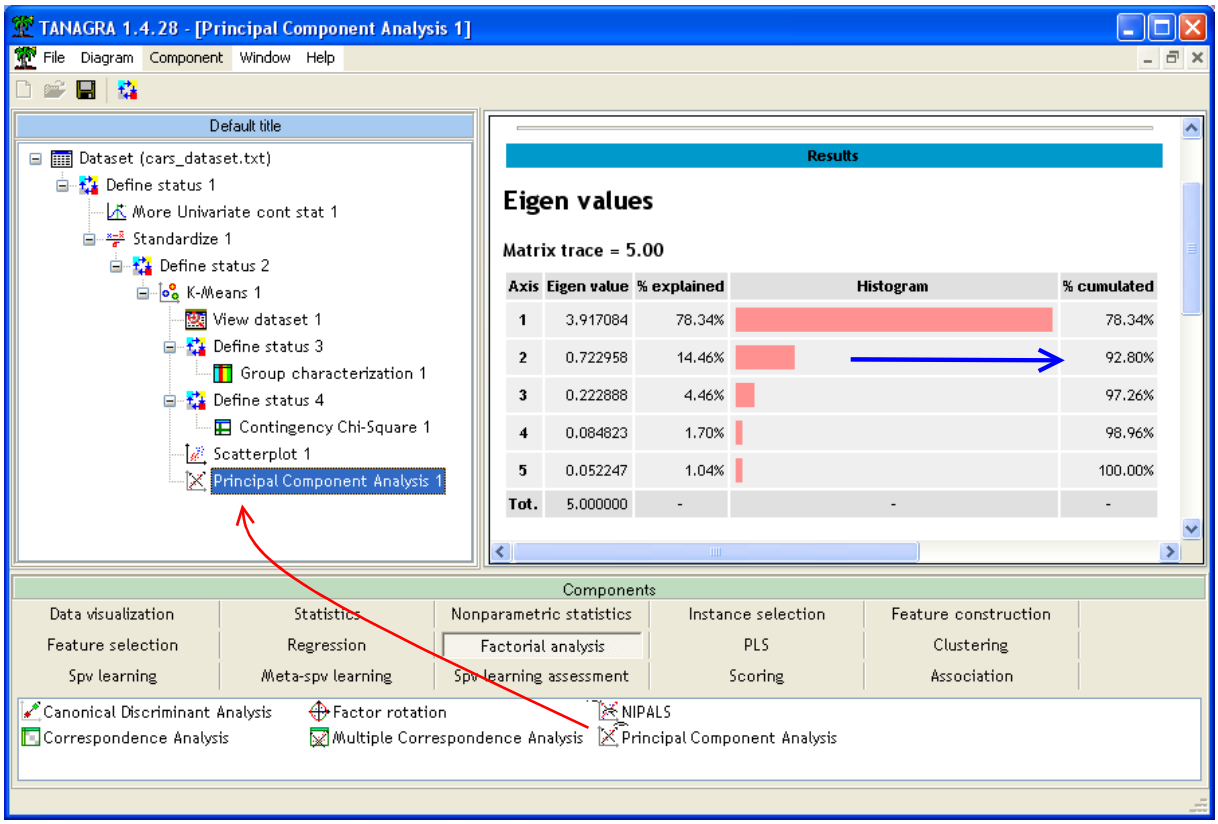
Le graphique confirme ce que nous pressentions dans l'analyse univariée (variable par variable). Il y a une opposition « voitures lourdes et puissantes » et « voitures légères et peu puissantes » dans ce fichier.

3.5.5 Projection dans le premier plan factoriel

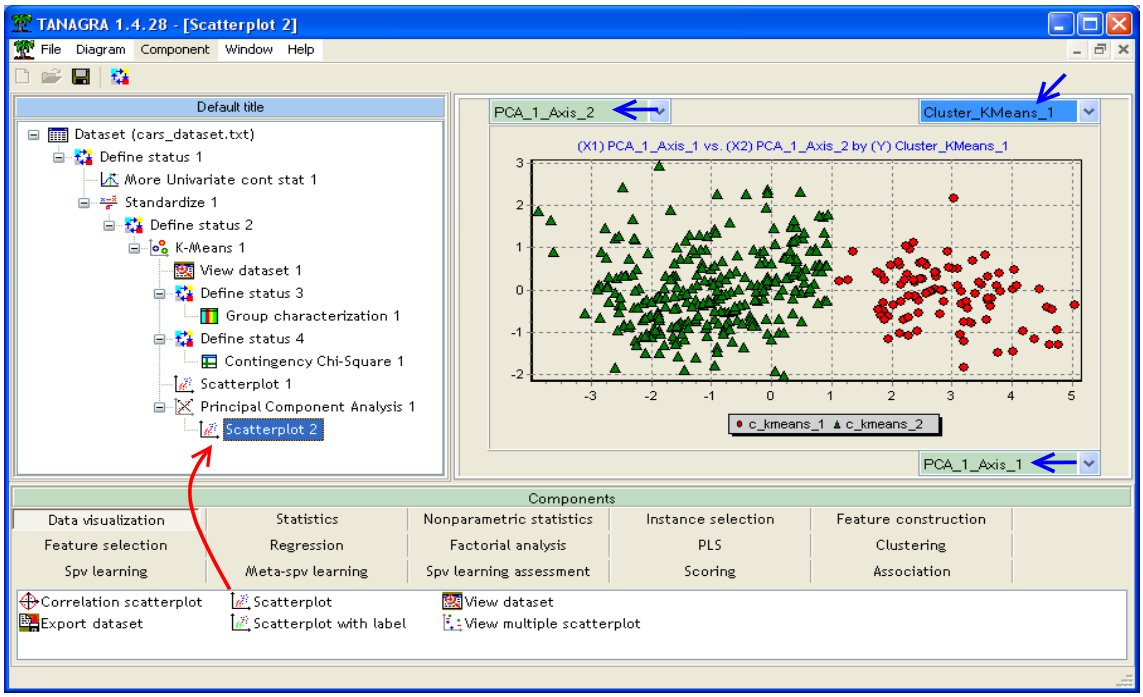
Pour tenir compte du rôle simultané des variables actives, nous pouvons projeter les données dans le premier plan factoriel de l'ACP. S'il traduit une part suffisamment importante de l'information contenue dans les données, le positionnement des groupes dans cet espace devrait être assez fidèle du mécanisme de formation des groupes.

Nous insérons le composant PRINCIPAL COMPONENT ANALYSIS (onglet FACTORIAL ANALYSIS) à la suite du K-MEANS 1. Il utilisera de fait les mêmes variables actives. Nous cliquons sur VIEW.

Le premier plan factoriel traduit 92.8% de l'information disponible. Le premier axe repose essentiellement sur l'opposition entre la sobriété (MPG) et la placidité (ACCELERATION) d'une part, la puissance (HORSEPOWER) et l'embonpoint (WEIGHT, DISPLACEMENT) d'autre part.



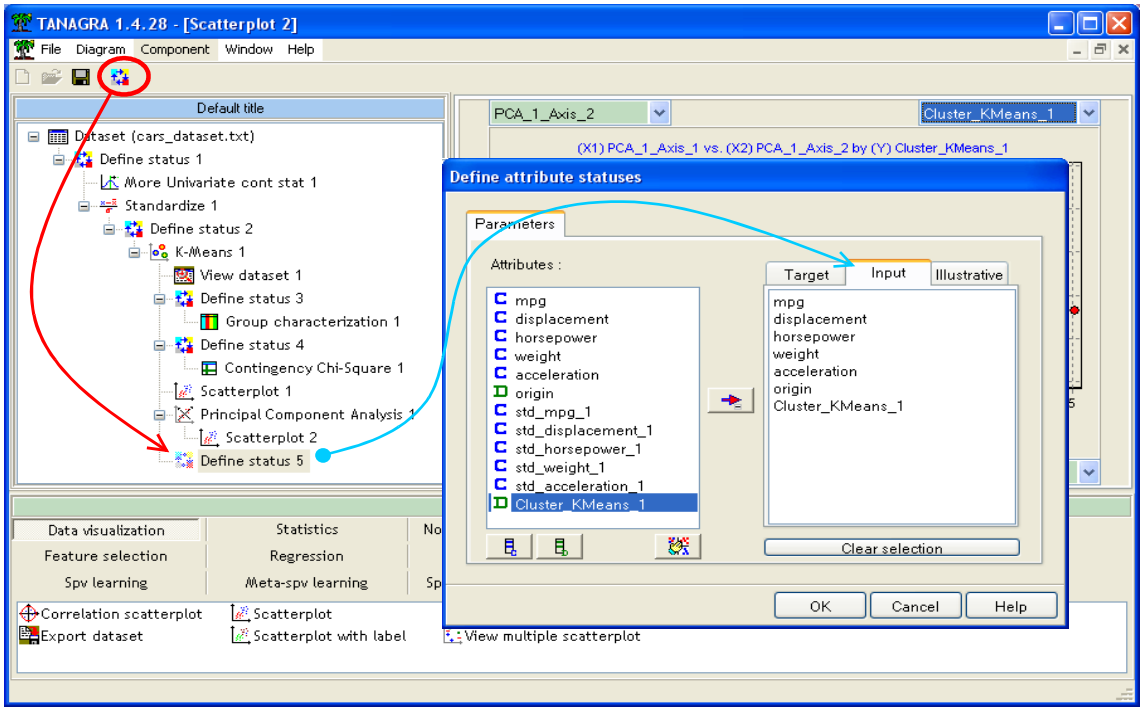
Lorsque nous construisons le graphique à l'aide du composant SCATTERPLOT, en abscisse le premier axe PCA_1_AXIS_1, en ordonnée le second PCA_1_AXIS_2. Les groupes se démarquent très nettement.



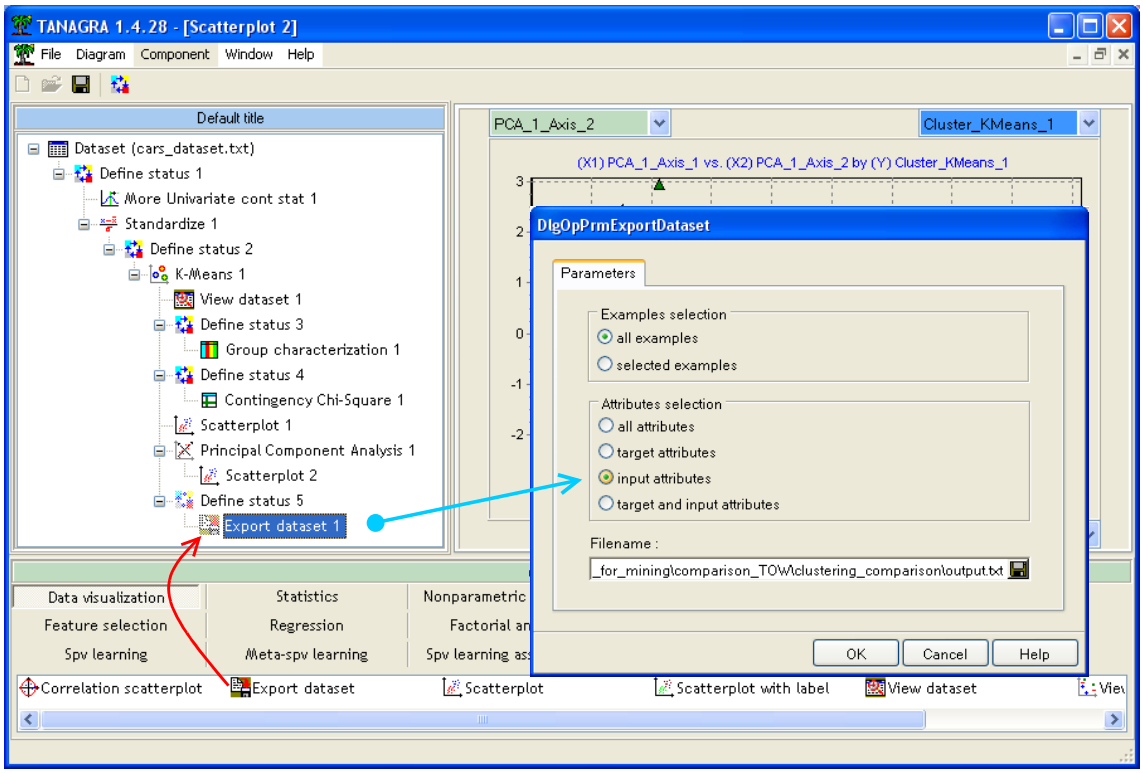
3.6 Exportation des données

Dernière étape de notre analyse, nous souhaitons exporter les données avec la colonne additionnelle représentant la classe d'affectation des individus. TANAGRA sait produire des fichiers textes avec séparateur tabulation, format accepté par la grande majorité des outils de Data Mining et des tableurs.

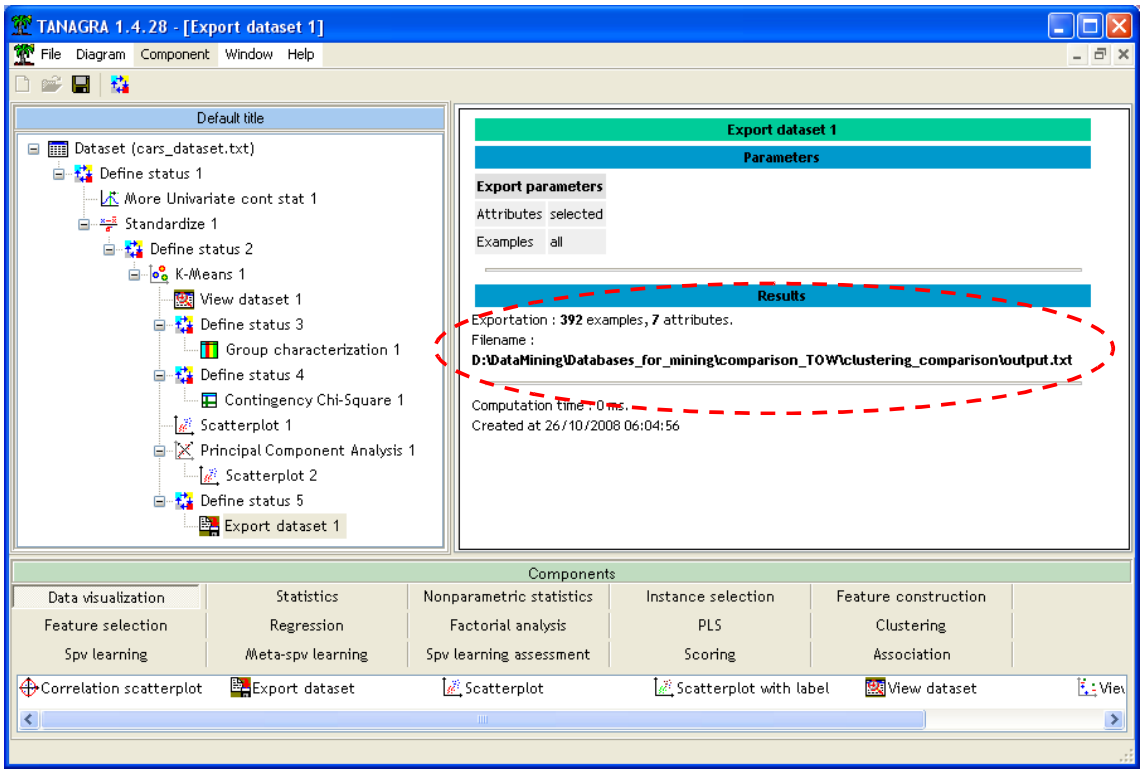
Nous devons dans un premier temps spécifier les variables à exporter. Nous utilisons pour cela le composant DEFINE STATUS. Nous plaçons en INPUT les variables originelles (MPG à ORIGIN) et la variable produite par les K-MEANS (CLUSTER_KMEANS_1).



Nous insérons le composant EXPORT DATASET (onglet DATA VISUALIZATION) dans le diagramme. Nous actionnons le menu PARAMETERS, nous précisons que ce sont les attributs INPUT qui doivent être exportés. Nous pouvons éventuellement modifier le nom et le répertoire du fichier. Pour notre part, nous laissons le nom de fichier par défaut OUTPUT.TXT. Nous validons et nous actionnons le menu VIEW.



TANAGRA indique qu'il a produit un fichier comportant 392 observations et 7 variables.



4 K-Means avec R

4.1 Importation des données et statistiques descriptives

Nous introduisons les commandes suivantes pour importer les données et calculer les statistiques descriptives avec R.

```
#importation des données
setwd("D:/DataMining/Databases_for_mining/comparison_TOW/clustering_comparison")
voitures <- read.table(file="cars_dataset.txt",header=T,dec=".")
#description et statistiques descriptives
summary(voitures)
```

Ce qui donne...

```
> summary(voitures)
      mpg          displacement     horsepower        weight      acceleration        origin
 Min.   : 9.00   Min.   : 68.0   Min.   : 46.0   Min.   :1613   Min.   : 8.00   american:245
 1st Qu.:17.00   1st Qu.:105.0   1st Qu.: 75.0   1st Qu.:2225   1st Qu.:14.00   european: 68
 Median :23.00   Median :151.0   Median : 93.5   Median :2804   Median :16.00   japanese: 79
 Mean   :23.49   Mean   :194.4   Mean   :104.5   Mean   :2978   Mean   :15.68
 3rd Qu.:29.00   3rd Qu.:275.8   3rd Qu.:126.0   3rd Qu.:3615   3rd Qu.:17.00
 Max.   :47.00   Max.   :455.0   Max.   :230.0   Max.   :5140   Max.   :25.00
```

4.2 Centrage et réduction

Pour centrer et réduire les données, nous devons tout d'abord construire une fonction « **centrage_reduction** » qui centre et réduit une colonne, que nous appliquons à l'ensemble des variables actives avec **apply(.)**. Il en résulte la matrice **voitures.cr**.

```
#préparer la fonction de standardisation d'une colonne
centrage_reduction <- function(x){
  return((x-mean(x))/sqrt(var(x)))
}
#appliquer pour produire le tableau des données centrées et réduites
voitures.cr <- apply(voitures[,1:5],2,centrage_reduction)
#vérification
apply(voitures.cr,2,mean)
apply(voitures.cr,2,var)
```

Ses colonnes sont de moyenne nulle et de variance unitaire.

```
> apply(voitures.cr,2,mean)
          mpg  displacement    horsepower        weight  acceleration
-2.400263e-16 -6.008637e-17 -1.767797e-16 -7.163479e-18  1.160824e-16
> apply(voitures.cr,2,var)
         mpg displacement    horsepower        weight acceleration
           1            1             1             1            1
```

4.3 K-Means sur les variables centrées et réduites

Nous pouvons lancer maintenant la méthode des K-Means sur les variables centrées et réduites, nous demandons une partition en 2 groupes. Nous limitons le nombre d'itérations à 40.

```
#K-Means en 2 groupes
nb.classes <- 2
voitures.kmeans <- kmeans(voitures.cr,centers=nb.classes,iter.max=40)
print(voitures.kmeans)
```

R affiche, entre autres : le nombre d'observations dans chaque groupe, 100 et 292 ; les moyennes conditionnellement aux groupes pour chaque variable active, celles qui ont été utilisées pour le calcul c.-à-d. les variables centrées et réduites, ce n'est pas très exploitable pour nous ; la classe associée à chaque observation, nous exploiterons cette colonne par la suite.

Remarque : Il semble que nous obtenons les mêmes classes que Tanagra avec R. Il faudrait croiser les 2 partitions pour en être sûr. Mais ce n'est pas forcément toujours le cas. En effet, dans la mesure où l'algorithme repose sur une heuristique, le choix des centres de départ notamment pouvant influer sur le résultat final, les classes peuvent être légèrement différentes à la sortie. Il peut en être de même si nous choisissons un autre algorithme d'optimisation (Forgy, etc.).

```
> print(voitures.kmeans)
K-means clustering with 2 clusters of sizes 100, 292

Cluster means:
        mpg displacement horsepower     weight acceleration
1 -1.1208246    1.4517603  1.4595718  1.3553242   -1.0180683
2  0.3838440   -0.4971782 -0.4998534 -0.4641521    0.3486535

Clustering vector:
  [1] 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2
 [47] 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2
 [93] 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2
[139] 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2
[185] 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2
[231] 2 2 2 1 2 2 2 2 2 2 2 2 1 2 2 1 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 1 1 2 2 2 2 1 2 2
[277] 1 1 2 1 1 2 2 1 2 2 2 2 1 1 2 1 1 2 2 2 2 2 2 2 1 1 1 1 1 2 2 2 1 1 1 1 1 1 1 2 1 2 1
[323] 2 1 1 1 1 1 1 1 1 2 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
[369] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1

Within cluster sum of squares by cluster:
[1] 146.9415 684.7405

Available components:
[1] "cluster"  "centers"  "withinss" "size"
```

4.4 Interprétation des groupes

Pour l'interprétation des groupes, nous calculons les **moyennes conditionnelles** des variables actives originelles. Nous les collectons dans une seule matrice à l'aide des commandes suivantes.

```
#récupération des groupes d'appartenance
groupe <- as.factor(voitures.kmeans$cluster)
#calculer les barycentres des classes
#dans l'espace des variables actives initiales (numéro 1 à 5)
centres <- NULL
for (k in 1:nb.classes){
  ligne <- colMeans(voitures[groupe==k,1:5,drop=FALSE])
  centres <- rbind(centres,ligne)
}
numero <- seq(from=1,to=nb.classes)
rownames(centres) <- paste("clus_",numero,sep="")
print(centres)
```

R nous affiche

```
> print(centres)
           mpg displacement horsepower   weight acceleration
clus_1 14.7500     346.3300  160.65000 4128.800     12.87000
clus_2 26.4863     142.3853   85.22945 2583.332     16.64384
```

Nous pouvons rapprocher ces résultats avec ceux du composant GROUP CHARACTERIZATION de Tanagra : CLUS_1 de R correspond aux C_KMEANS_2 de Tanagra.

Pour **croiser les clusters avec la variable catégorielle illustrative ORIGIN**, nous introduisons la commande

```
#croisement des clusters avec la variable illustrative catégorielle
print(table(voitures$origin,groupe))
```

R produit le tableau de contingence

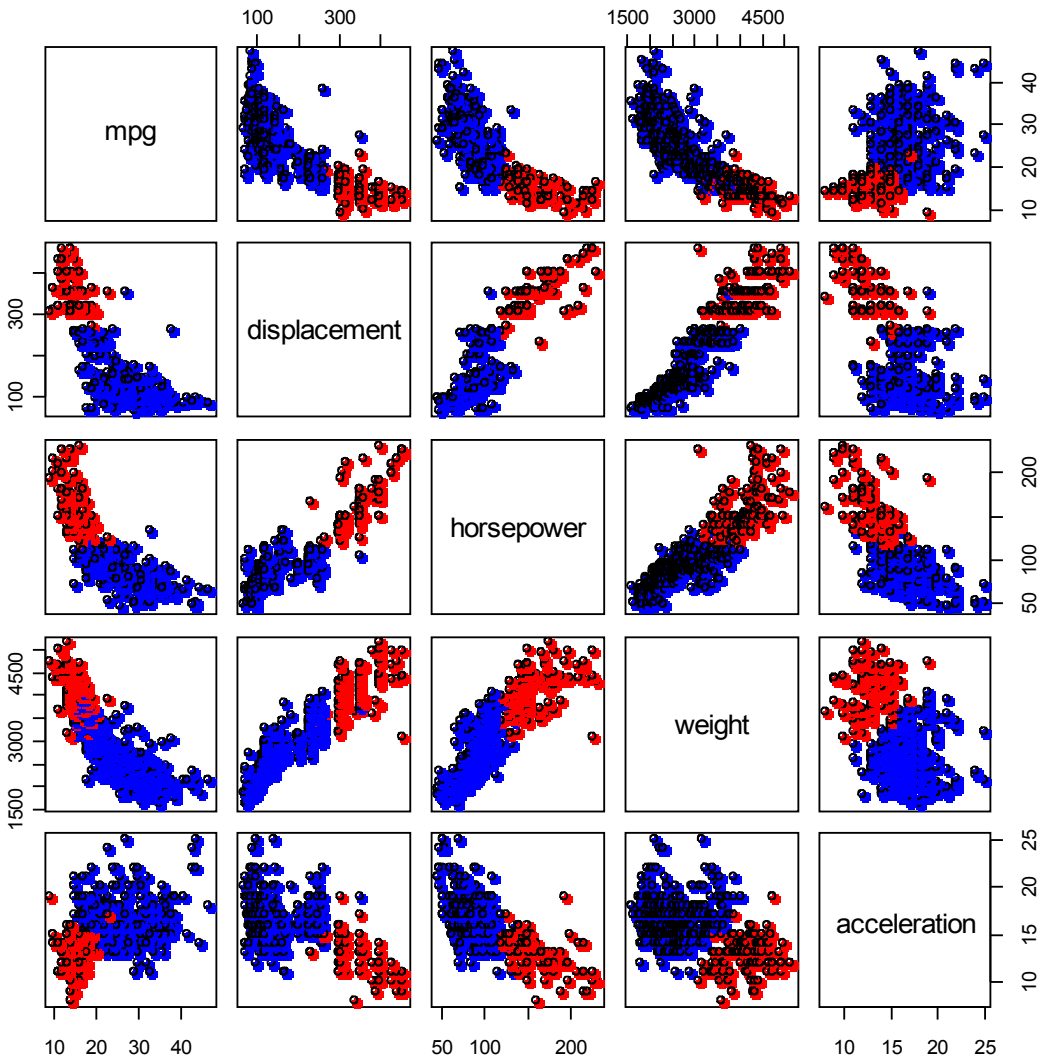
```
> #croisement des clusters avec la variable illustrative catégorielle
> print(table(voitures$origin,groupe))
          groupe
             1   2
  american 100 145
  european   0  68
  japanese   0  79
```

Pour **projeter les points**, illustrés selon leur groupe d'appartenance, **dans les plans formés par les couples de variables**, R démontre toute sa puissance[3]. La commande semble simple...

```
#graphique des variables 2 à 2 avec groupe d'appaternance
pairs(voitures[,1:5],pch=21,bg=c("red","blue")[groupe])
```

... mais le résultat est riche d'enseignements : les variables sont pour la plupart fortement corrélées, presque tous les couples de variables permettent de distinguer les groupes.

[3] NDLA : Là, j'avoue que R m'épate.



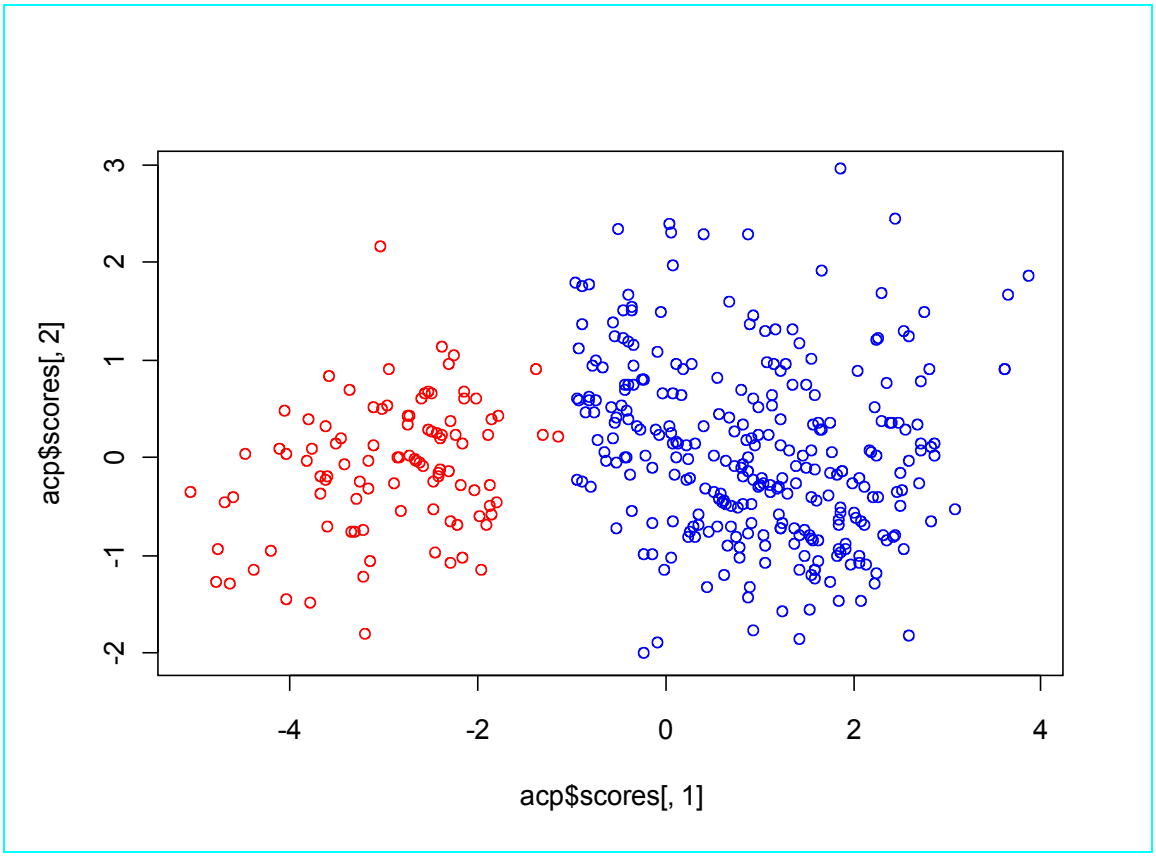
Enfin, dernier outil permettant de situer les groupes, nous pouvons **projeter les points dans le premier plan factoriel de l'ACP**. Notons que la commande *princomp(.)* propose des sorties qui ne correspondent pas avec celles qui sont le plus souvent mises en avant dans la littérature (francophone tout du moins). Nous avons du procéder à quelques ajustements.

```
#ACP sur les données centrées réduites
acp <- princomp(voitures.cr,cor=T,scores=T)
print(acp)
#pour obtenir les valeurs propres
print(acp$sdev^2)
#pour obtenir les corrélations sur le premier axe
print(acp$loadings[,1]*acp$sdev[1])
#graphique dans le premier plan factoriel, avec mise en évidence des groupes
plot(acp$scores[,1],acp$scores[,2],type="p",pch=21,col=c("red","blue")[groupe])
```

R affiche maintenant des résultats que l'on peut mettre en parallèle avec ceux du composant PRINCIPAL COMPONENT ANALYSIS de Tanagra.

```
> #pour obtenir les valeurs propres
> print(acp$sdev^2)
    Comp.1     Comp.2     Comp.3     Comp.4     Comp.5
3.91708377 0.72295773 0.22288784 0.08482332 0.05224734
> #pour obtenir les corrélations sur le premier axe
> print(acp$loadings[,1]*acp$sdev[1])
         mpg displacement   horsepower       weight acceleration
   0.8779948   -0.9588019   -0.9599118   -0.9343510    0.6576210
```

Nous obtenons le graphique suivant



4.5 Exportation des données

Dernière étape du processus, nous exportons les données en fusionnant la base initiale avec la colonne additionnelle produite par la typologie.

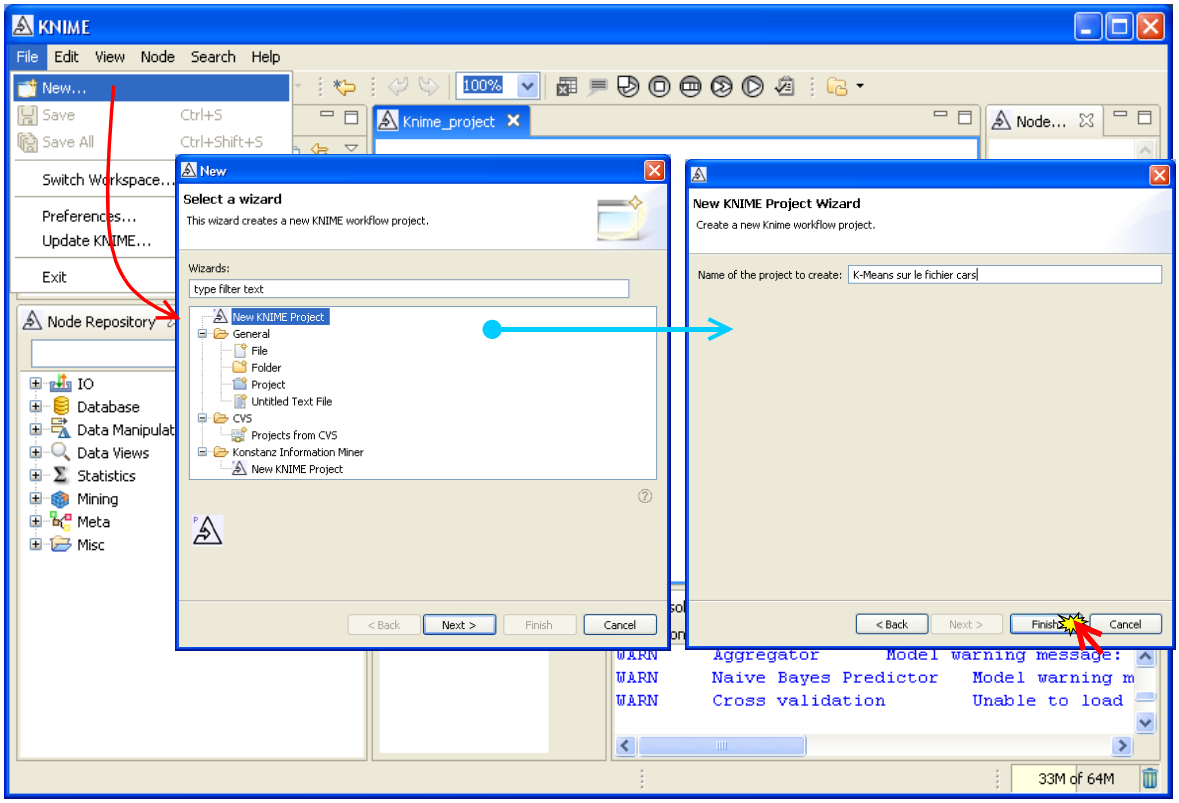
```
#exportation des données avec le cluster d'appartenance
voitures.export <- cbind(voitures,groupe)
write.table(voitures.export,file="export_r.txt",sep="\t",dec=".",row.names=F)
```

5 K-Means avec KNIME

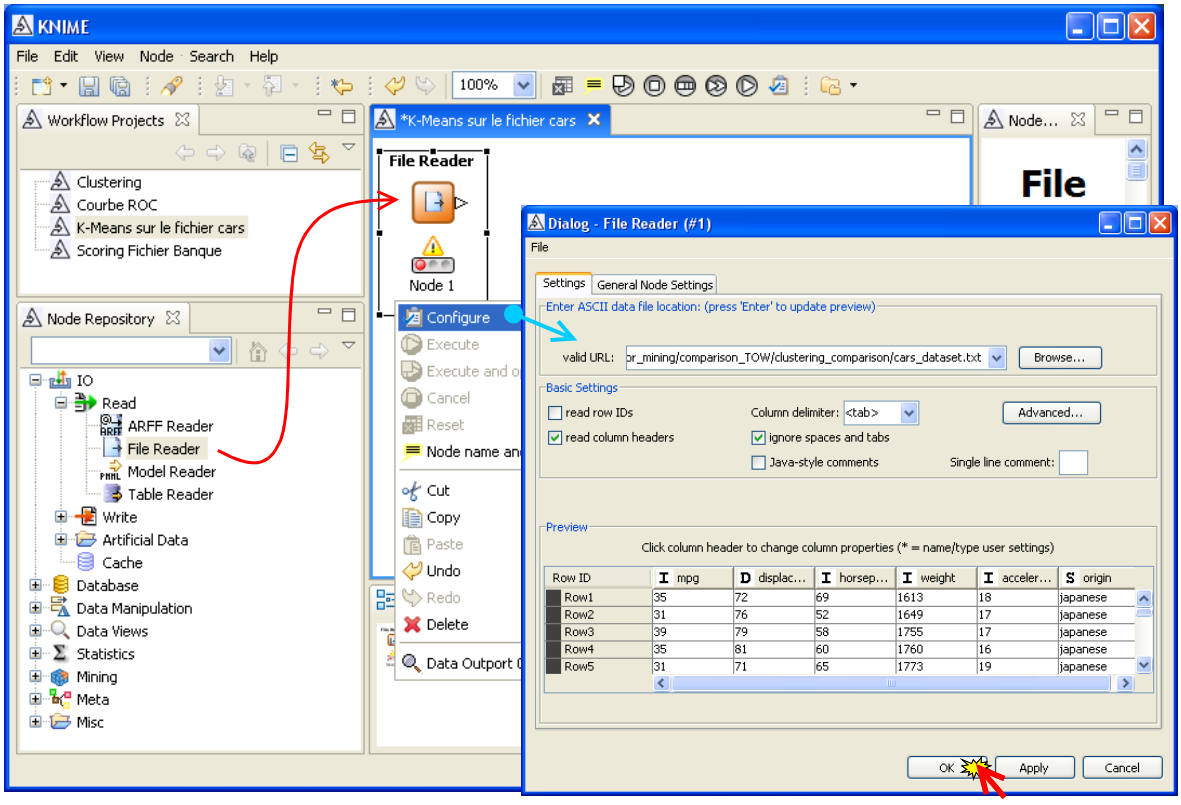
Nous ne décrirons pas l'interface globale et le mode de fonctionnement de KNIME. Les promoteurs du logiciel s'en chargent très bien (http://www.knime.org/). Nous nous concentrerons avant tout sur la mise en œuvre de la classification automatique et de l'interprétation des classes.

5.1 Création d'un workflow et importation des données

Dans un premier temps, nous créons un workflow via le menu FILE / NEW. Nous choisissons l'option « New Knime Project », puis nous introduisons le nom « K-Means sur le fichier cars ».

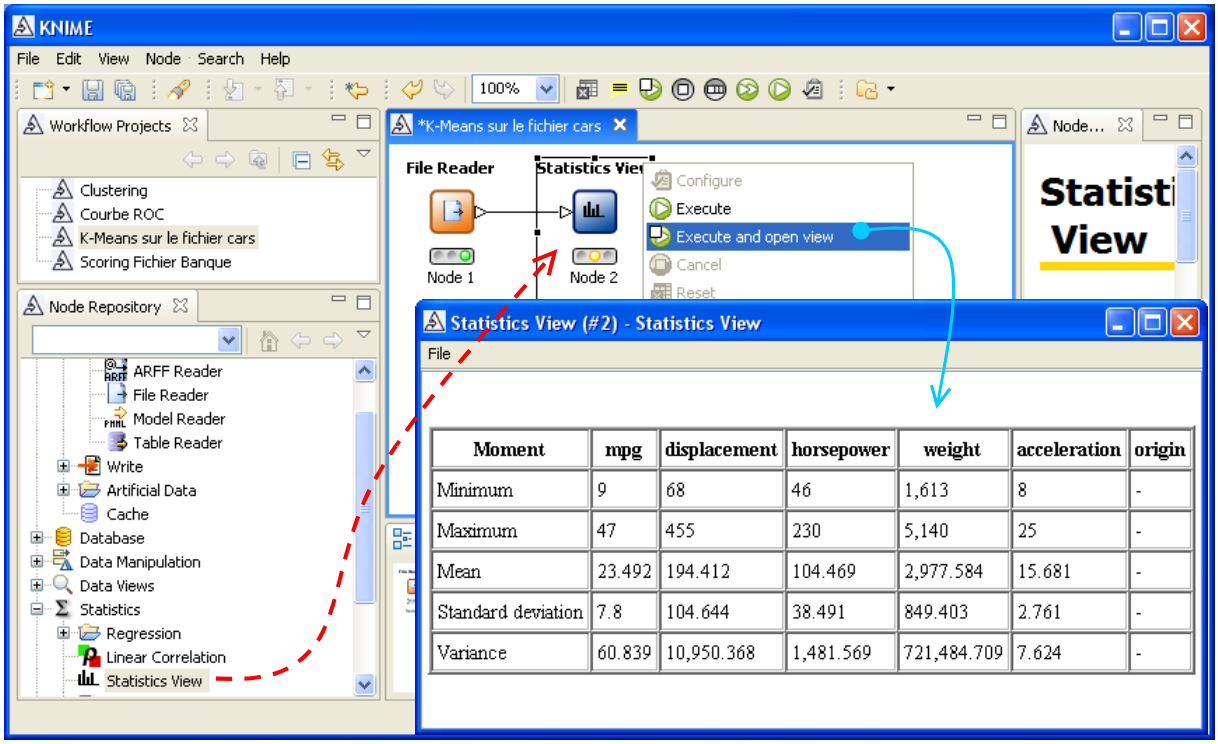


Nous importons le fichier de données avec le composant FILE READER.



5.2 Statistiques descriptives

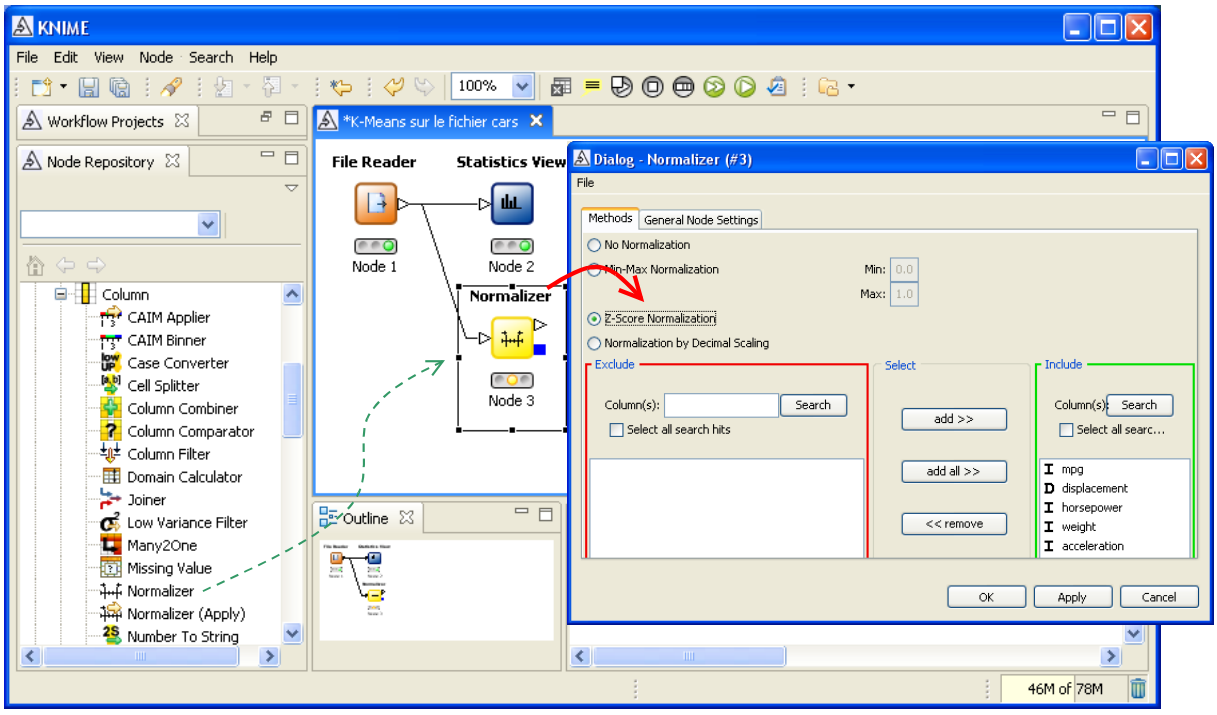
Pour calculer les statistiques descriptives, nous utilisons le composant STATISTICS VIEW. Nous lui branchons le composant d'accès aux données, nous actionnons le menu contextuel EXECUTE AND OPEN VIEW pour accéder aux résultats qui apparaissent dans une fenêtre flottante.



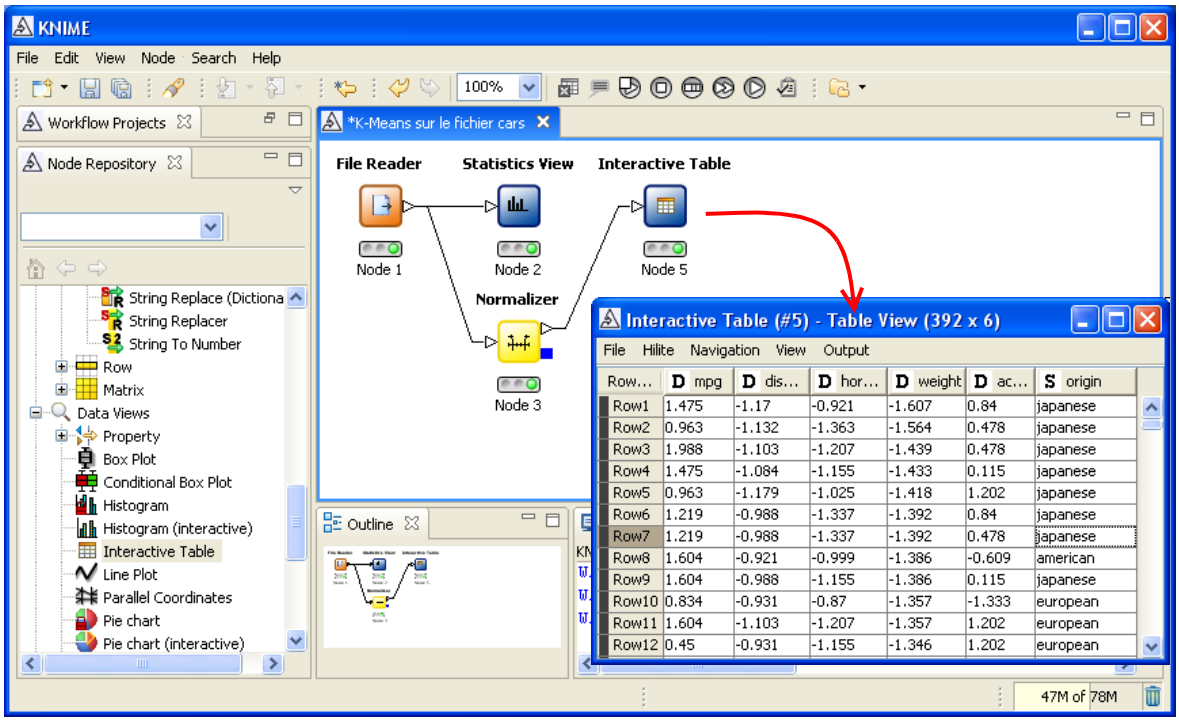
Moment	mpg	displacement	horsepower	weight	acceleration	origin
Minimum	9	68	46	1,613	8	-
Maximum	47	455	230	5,140	25	-
Mean	23.492	194.412	104.469	2,977.584	15.681	-
Standard deviation	7.8	104.644	38.491	849.403	2.761	-
Variance	60.839	10,950.368	1,481.569	721,484.709	7.624	-

5.3 Centrage – réduction des variables actives

Pour centrer et réduire les données, nous utilisons le composant NORMALIZER. Il peut introduire différentes normalisations, nous choisissons le paramétrage adéquat avec le menu CONFIGURE.

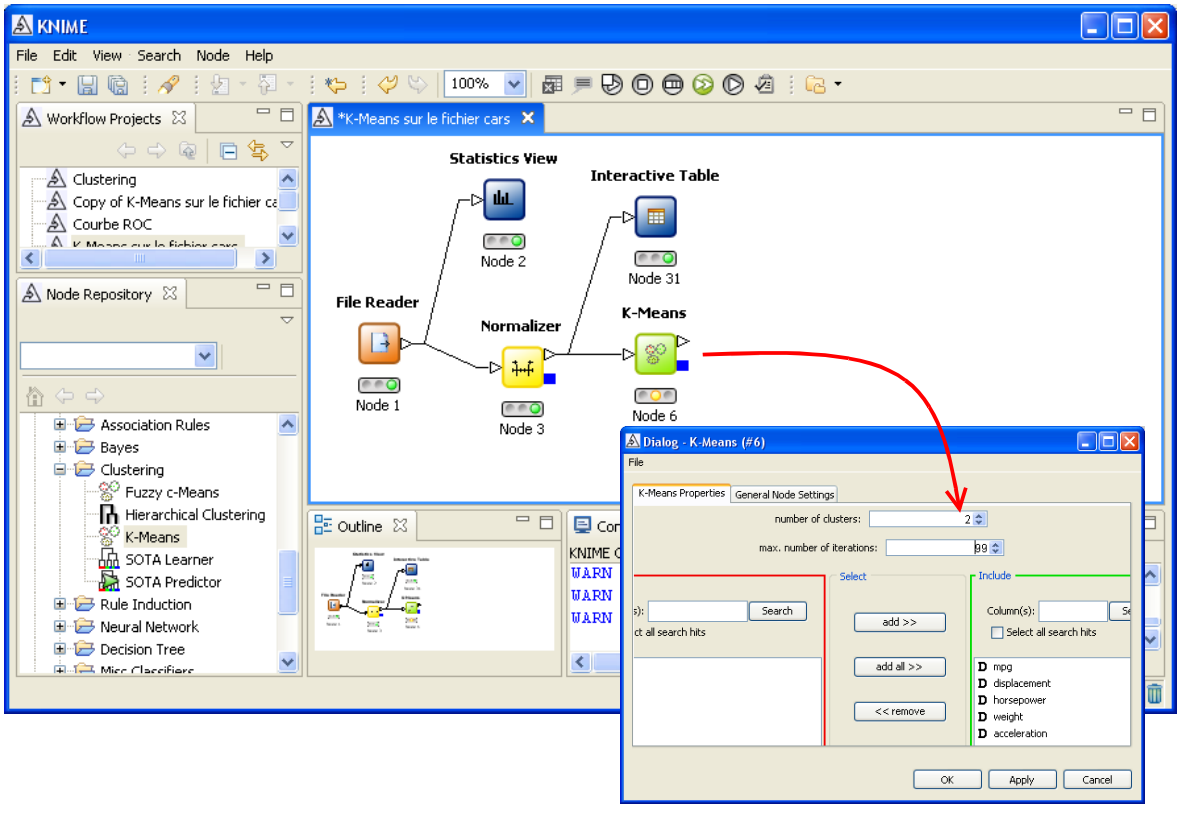


Le composant INTERACTIVE TABLE permet de visualiser les données transformées, seules les variables numériques ont été modifiées, bien entendu.

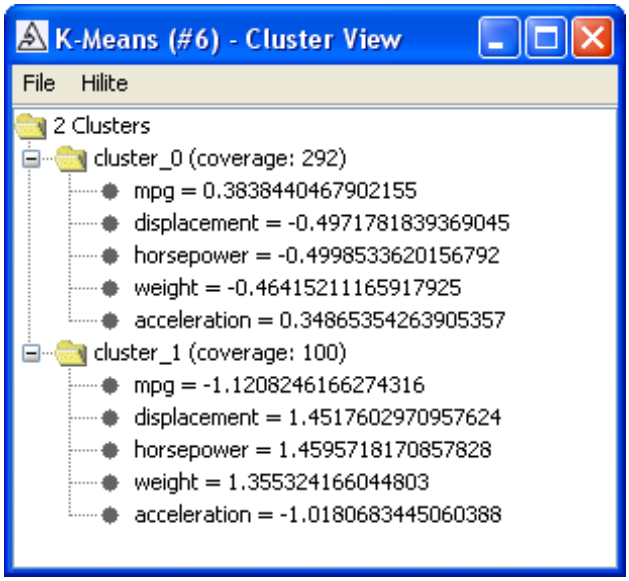


5.4 K-Means sur les variables centrées et réduites

Nous pouvons lancer la méthode des K-Means sur les variables centrées et réduites. Nous introduisons le composant K-Means dans le workflow. Nous actionnons le menu CONFIGURE pour demander 2 classes.



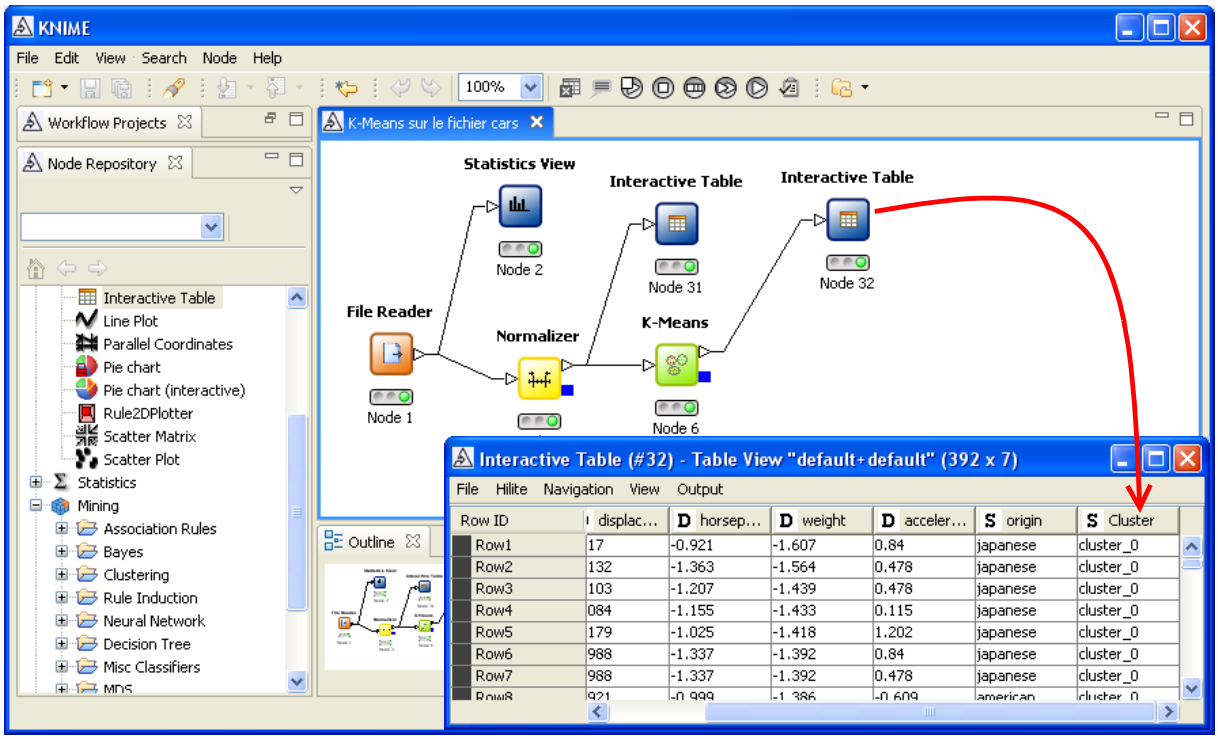
Le menu EXECUTE AND OPEN VIEW affiche la fenêtre de résultats. Knime produit une partition en 2 groupes, avec respectivement 292 et 100 observations. Les moyennes conditionnelles sont affichées, mais il s'agit des moyennes calculées dans l'espace des données centrées et réduites, l'intérêt est faible pour l'interprétation.



5.5 Interprétation des groupes

5.5.1 Affectation des classes aux individus

Le composant INTERACTIVE TABLE nous permet de visualiser l'affectation des groupes à chaque individu. Nous obtenons la grille suivante.

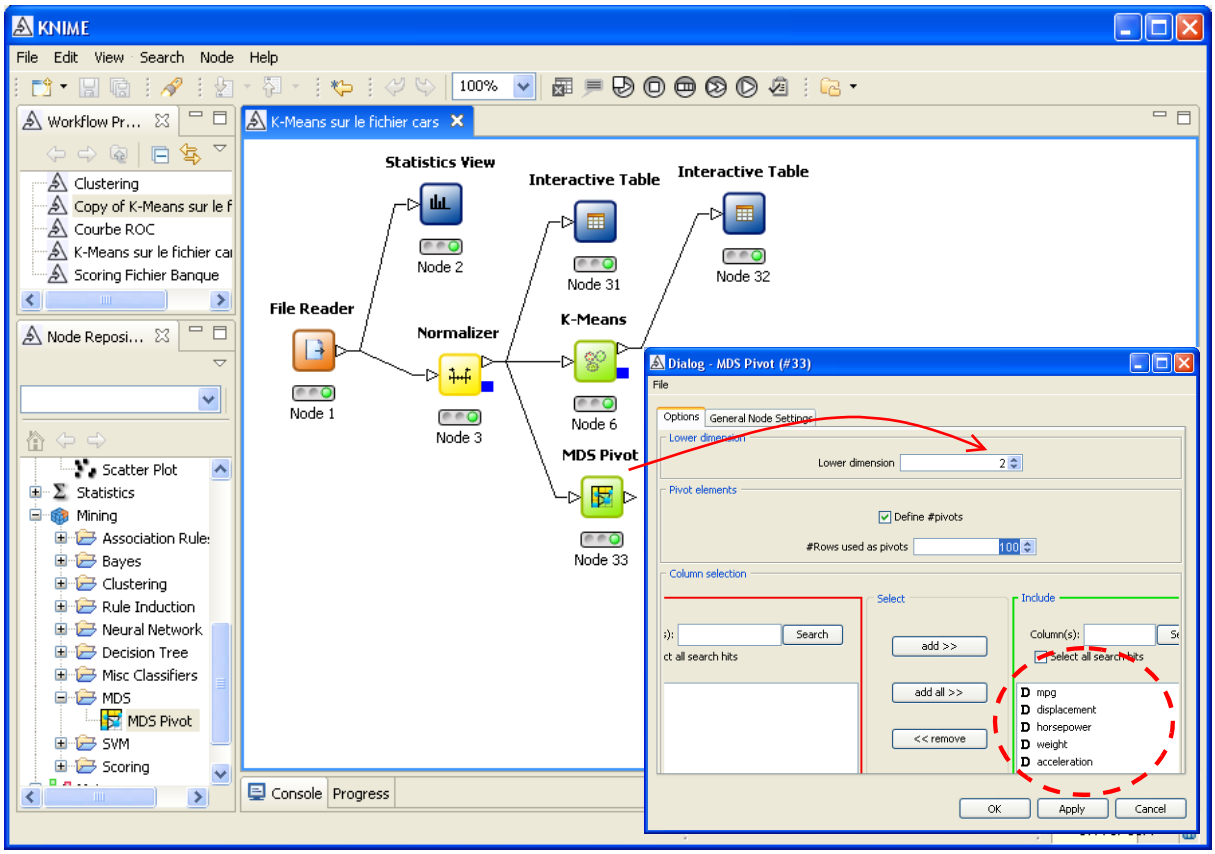


5.5.2 Statistiques descriptives et graphiques

Nous devons définir une série de manipulation des différents ensembles de données pour produire les graphiques et statistiques illustratives.

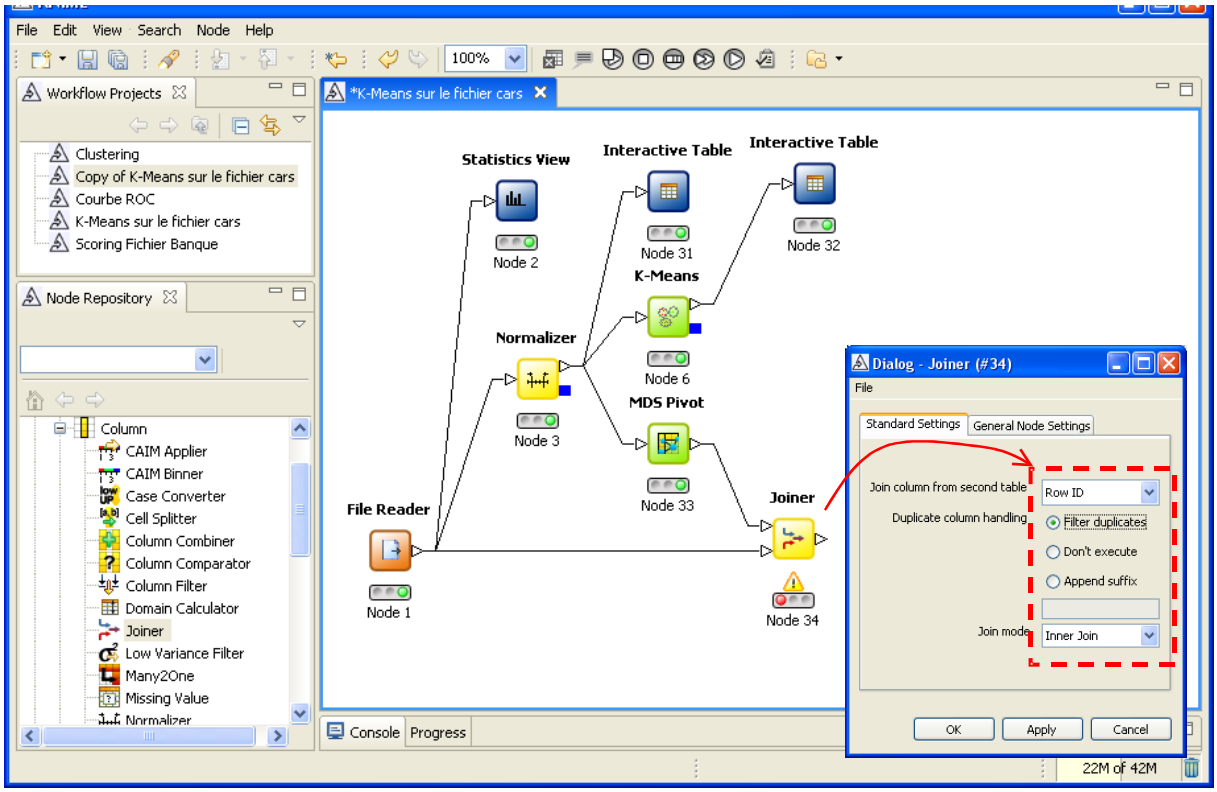
Tout d'abord, nous calculons la projection des observations dans le premier axe factoriel. L'ACP en tant que telle n'est pas présente. En revanche, Knime est capable de réaliser un Multidimensional Scaling (http://en.wikipedia.org/wiki/Multidimensional_scaling) à partir d'une matrice de distance deux à deux des observations. La documentation n'est pas très claire sur les calculs réalisés. Mais, s'ils sont basés sur une matrice de distance euclidienne entre les points, les résultats seront identiques à ceux de l'ACP (http://www.mathpsyc.uni-bonn.de/doc/delbeke/delbeke.htm).

Nous introduisons le composant MDS PIVOT dans le workflow. Nous le configurons de manière à ce qu'il travaille sur les variables centrées et réduites, et qu'il produise 2 sous dimensions, soit les 2 premiers axes factoriels (avec les réserves ci-dessus).

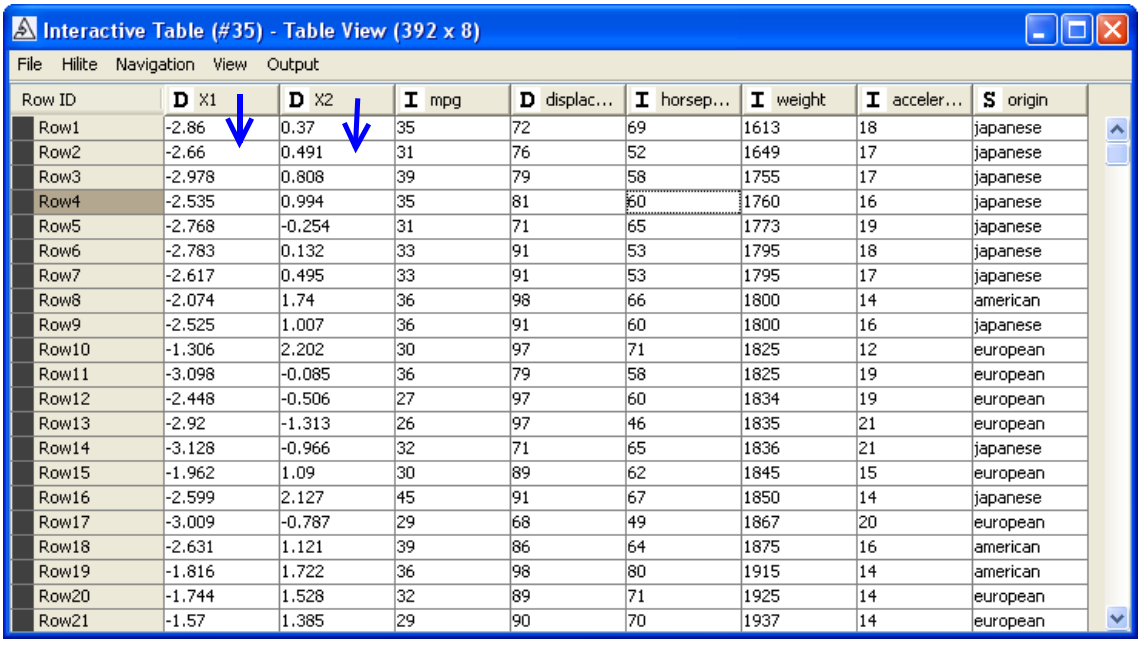


KNIME produit automatiquement 2 variables supplémentaires qu'il ajoute à l'ensemble de données courant.

Nous devons dans un premier temps fusionner les données initiales avec ces deux variables synthétiques. Nous utilisons pour cela le composant JOINER. Nous lui connectons la source de données initiale (FILE READER) et les facteurs produits par le MDS PIVOT. Nous le paramétrons de manière à ce que les colonnes des deux sources de données soient accolées, en respectant la correspondance entre les lignes.



Nous introduisons un INTERACTIVE TABLE pour vérifier tout cela, Knime nous affiche la grille suivante.

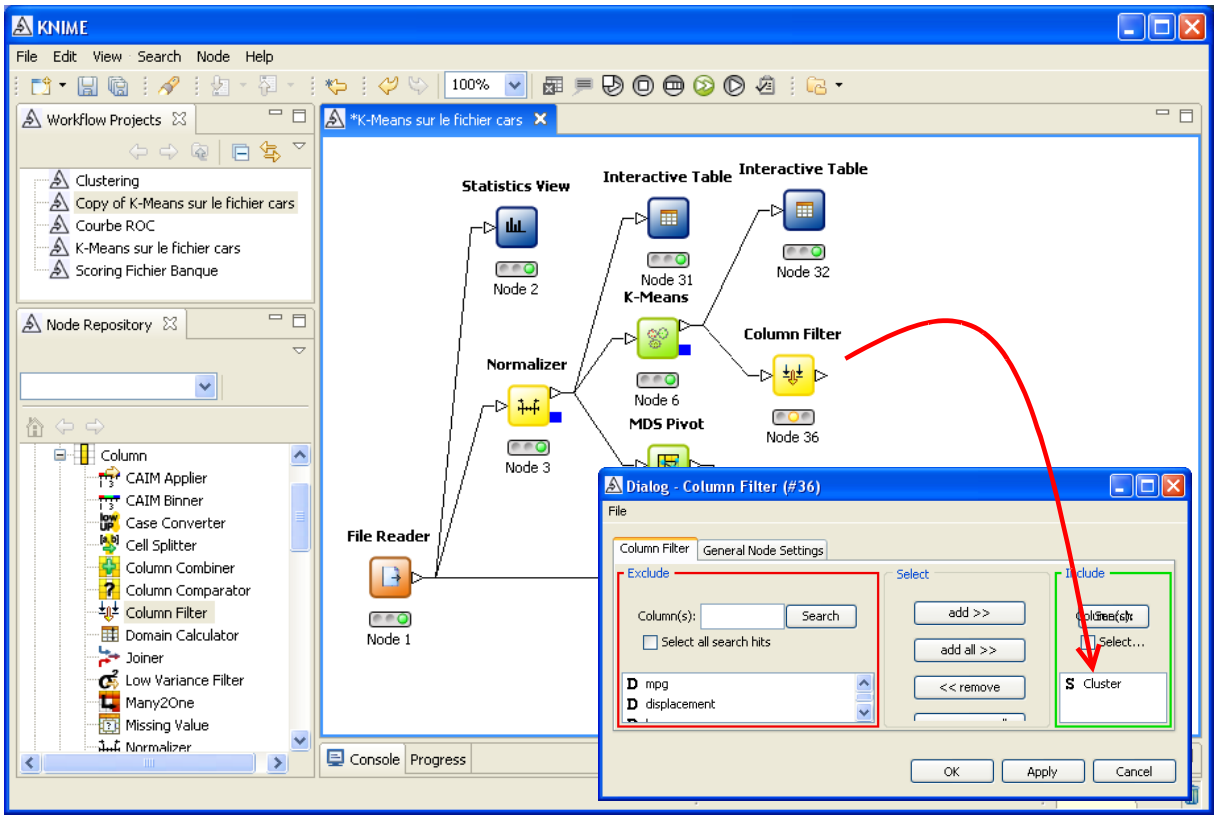


Row ID	D X1	D X2	I mpg	D displac...	I horsep...	I weight	I acceler...	S origin
Row1	-2.86	0.37	35	72	69	1613	18	japanese
Row2	-2.66	0.491	31	76	52	1649	17	japanese
Row3	-2.978	0.808	39	79	58	1755	17	japanese
Row4	-2.535	0.994	35	81	60	1760	16	japanese
Row5	-2.768	-0.254	31	71	65	1773	19	japanese
Row6	-2.783	0.132	33	91	53	1795	18	japanese
Row7	-2.617	0.495	33	91	53	1795	17	japanese
Row8	-2.074	1.74	36	98	66	1800	14	american
Row9	-2.525	1.007	36	91	60	1800	16	japanese
Row10	-1.306	2.202	30	97	71	1825	12	european
Row11	-3.098	-0.085	36	79	58	1825	19	european
Row12	-2.448	-0.506	27	97	60	1834	19	european
Row13	-2.92	-1.313	26	97	46	1835	21	european
Row14	-3.128	-0.966	32	71	65	1836	21	japanese
Row15	-1.962	1.09	30	89	62	1845	15	european
Row16	-2.599	2.127	45	91	67	1850	14	japanese
Row17	-3.009	-0.787	29	68	49	1867	20	european
Row18	-2.631	1.121	39	86	64	1875	16	american
Row19	-1.816	1.722	36	98	80	1915	14	american
Row20	-1.744	1.528	32	89	71	1925	14	european
Row21	-1.57	1.385	29	90	70	1937	14	european

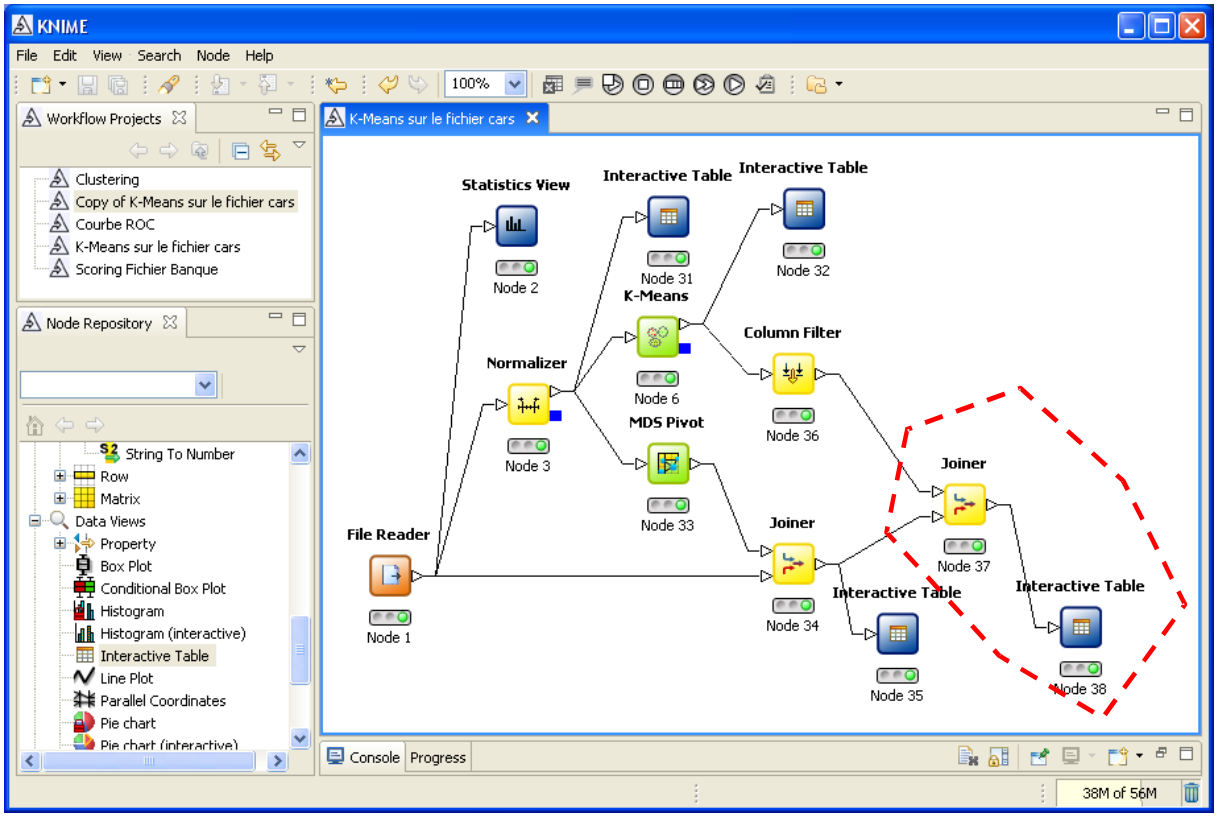
Nous retrouvons bien les variables originelles de notre fichier et les deux variables supplémentaires produites par le « multidimensional scaling ».

Dans un deuxième temps, nous devons fusionner cet ensemble de données et la colonne produite par les K-Means. Deux étapes sont nécessaires : filtrer les sorties du composant K-Means de manière à éviter les doublons, seule la colonne des clusters sera exploitée ; fusionner cette colonne avec l'ensemble de données produite par la précédente jointure. Voyons cela.

Avec le composant COLUMN FILTER, nous filtrons la sortie de K-MEANS.

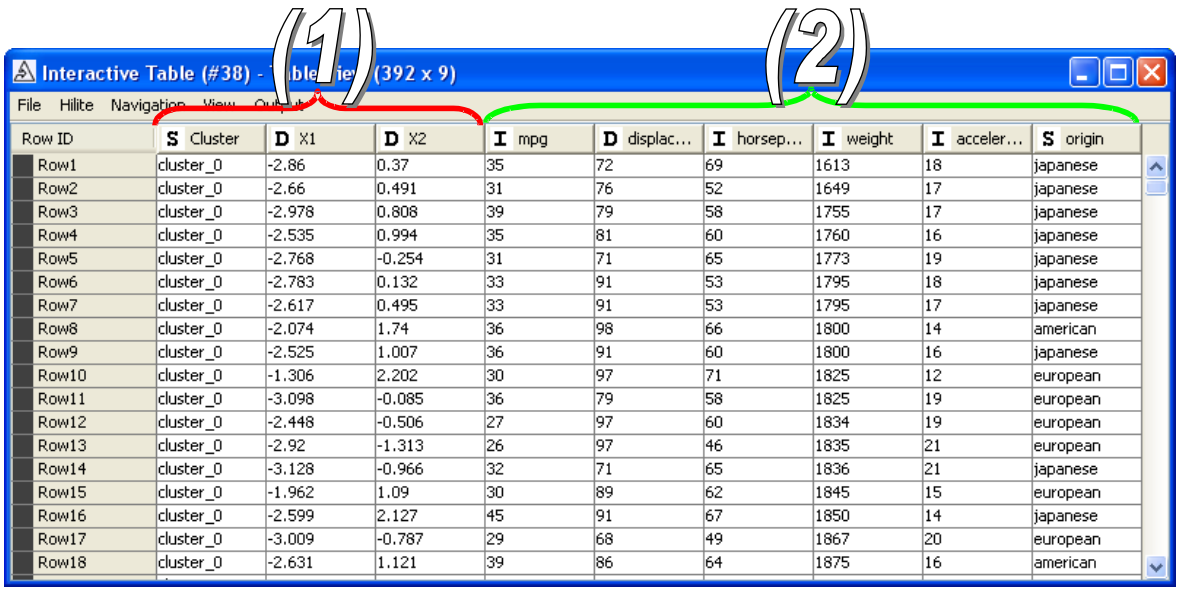


Avec un nouveau JOINER, nous réunissons les deux sources de données.



Nous utilisons un INTERACTIVE TABLE pour visualiser l'ensemble de données qui en résulte. Nous disposons de toutes les informations nécessaires à ce stade pour illustrer et interpréter les

groupes : le premier bloc correspond aux variables construites, le second provient du fichier originel.

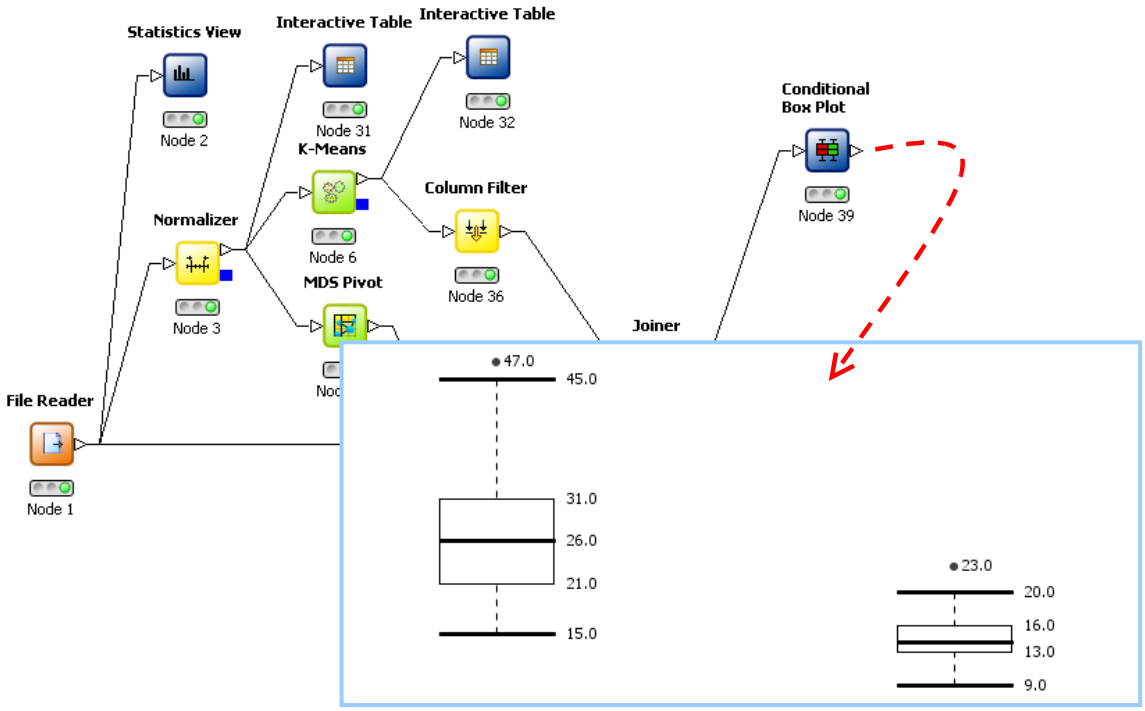


Row ID	S Cluster	D X1	D X2	I mpg	D displac...	I horsep...	I weight	I acceler...	S origin
Row1	cluster_0	-2.86	0.37	35	72	69	1613	18	japanese
Row2	cluster_0	-2.66	0.491	31	76	52	1649	17	japanese
Row3	cluster_0	-2.978	0.808	39	79	58	1755	17	japanese
Row4	cluster_0	-2.535	0.994	35	81	60	1760	16	japanese
Row5	cluster_0	-2.768	-0.254	31	71	65	1773	19	japanese
Row6	cluster_0	-2.783	0.132	33	91	53	1795	18	japanese
Row7	cluster_0	-2.617	0.495	33	91	53	1795	17	japanese
Row8	cluster_0	-2.074	1.74	36	98	66	1800	14	american
Row9	cluster_0	-2.525	1.007	36	91	60	1800	16	japanese
Row10	cluster_0	-1.306	2.202	30	97	71	1825	12	european
Row11	cluster_0	-3.098	-0.085	36	79	58	1825	19	european
Row12	cluster_0	-2.448	-0.506	27	97	60	1834	19	european
Row13	cluster_0	-2.92	-1.313	26	97	46	1835	21	european
Row14	cluster_0	-3.128	-0.966	32	71	65	1836	21	japanese
Row15	cluster_0	-1.962	1.09	30	89	62	1845	15	european
Row16	cluster_0	-2.599	2.127	45	91	67	1850	14	japanese
Row17	cluster_0	-3.009	-0.787	29	68	49	1867	20	european
Row18	cluster_0	-2.631	1.121	39	86	64	1875	16	american

5.5.2.1 Statistiques descriptives conditionnelles

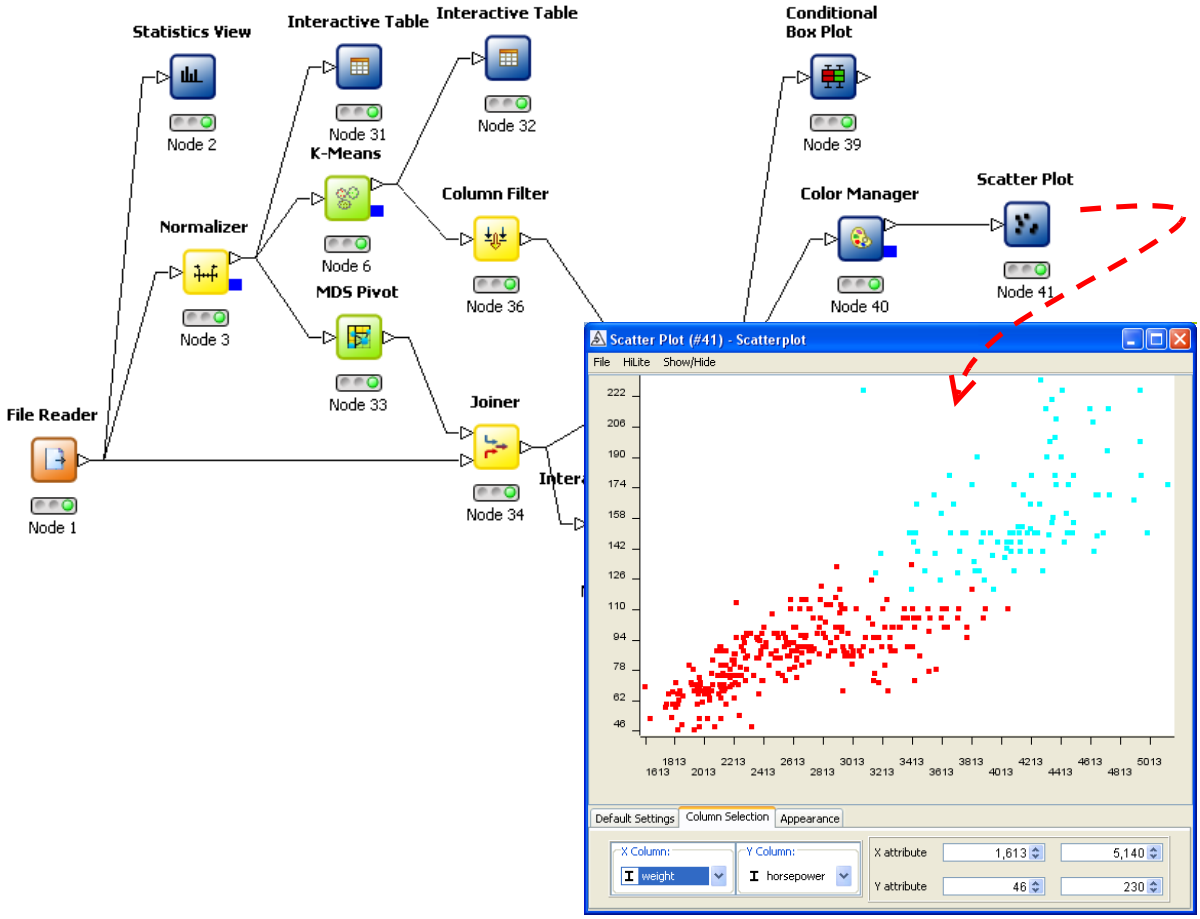
Premier outil, nous allons calculer les statistiques descriptives comparatives par variable. Knime intègre un outil intéressant, les « box plot » conditionnels. Nous pouvons non seulement comparer visuellement les indicateurs de tendance centrale (médiane), mais aussi les distributions (dispersion, asymétrie, points atypiques, etc.). L'information est plus riche. L'inconvénient est que nous devons le faire manuellement, attribut par attribut.

Nous insérons le composant CONDITIONAL BOXPLOT dans le workflow. Nous le configurons (menu CONFIGURE), nous mettons en attribut nominal le CLUSTER, en attribut numérique le MPG. Le menu EXECUTE AND OPEN VIEW affiche les boîtes à moustaches, nous constatons effectivement que MPG est en moyenne plus élevé dans le CLUSTER_0.



5.5.2.2 Projection dans l'espace des couples de variables

Second outil, nous souhaitons projeter les nuages de points conditionnels dans l'espace des couples de variables. Dans Knime, nous devons dans un premier temps spécifier la variable illustrative, en l'occurrence CLUSTER, à l'aide du composant COLOR MANAGER. Par la suite, nous utilisons l'outil SCATTERPLOT.

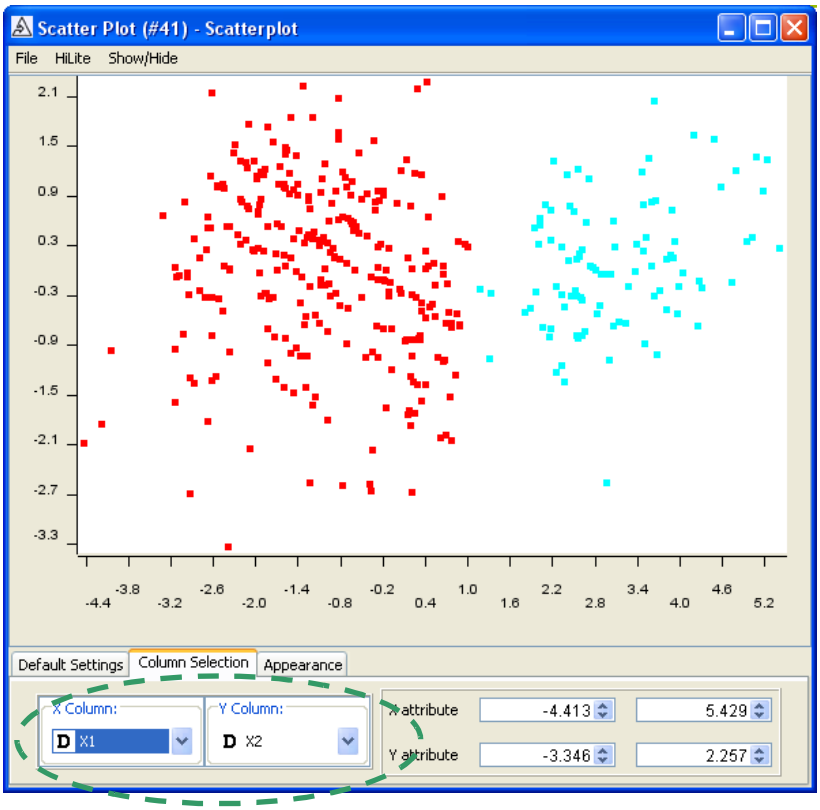


Nous distinguons nettement les deux classes. Notons qu'il est possible, tout comme dans Tanagra, de modifier interactivement les variables en abscisse et en ordonnée.

5.5.2.3 Projection dans le plan factoriel (du MDS)

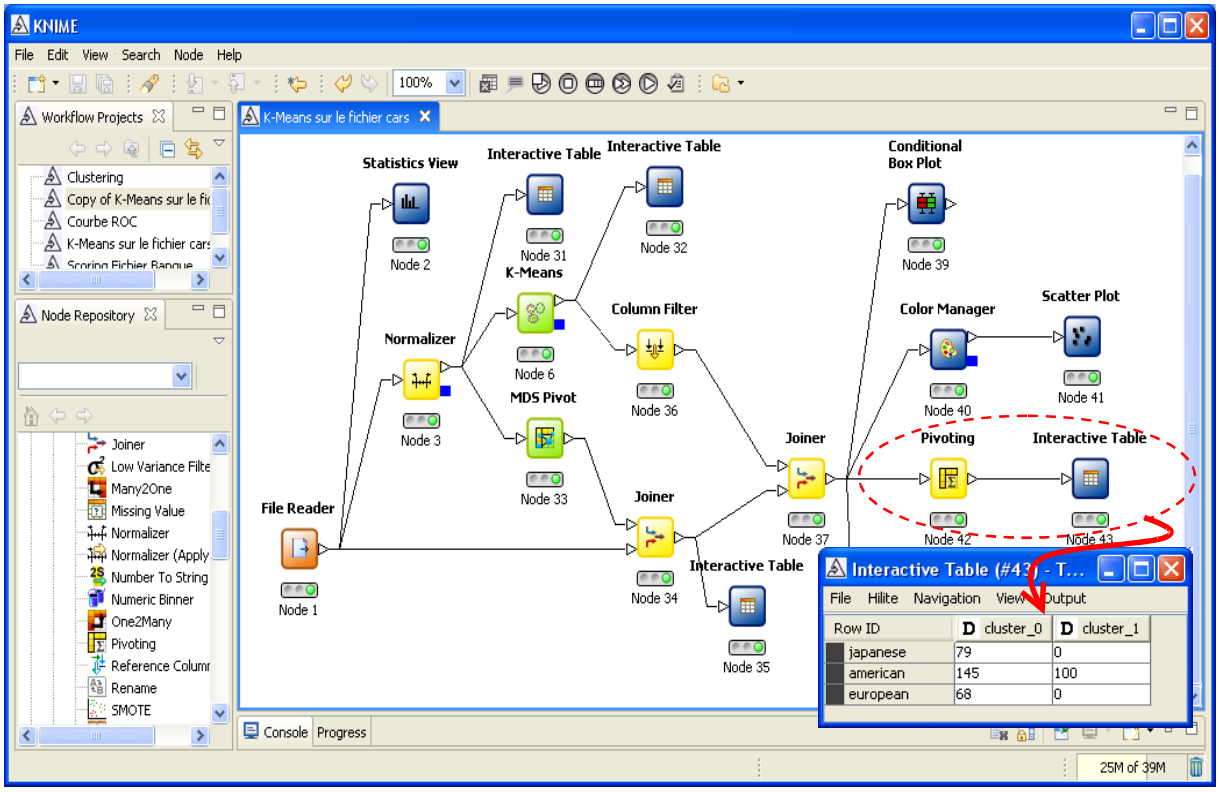
Grâce à la fonctionnalité décrite dans le paragraphe précédent, nous pouvons proposer directement le graphique des nuages de points dans l'espace construit à l'aide de multidimensional scaling.

Nous ne manquerons de faire l'analogie avec la projection dans le premier plan factoriel de l'analyse en composantes principales. Le premier axe suffit pour différencier les classes issues de la méthode des K-Means.



5.5.2.4 Croisement des clusters avec la variable ORIGIN

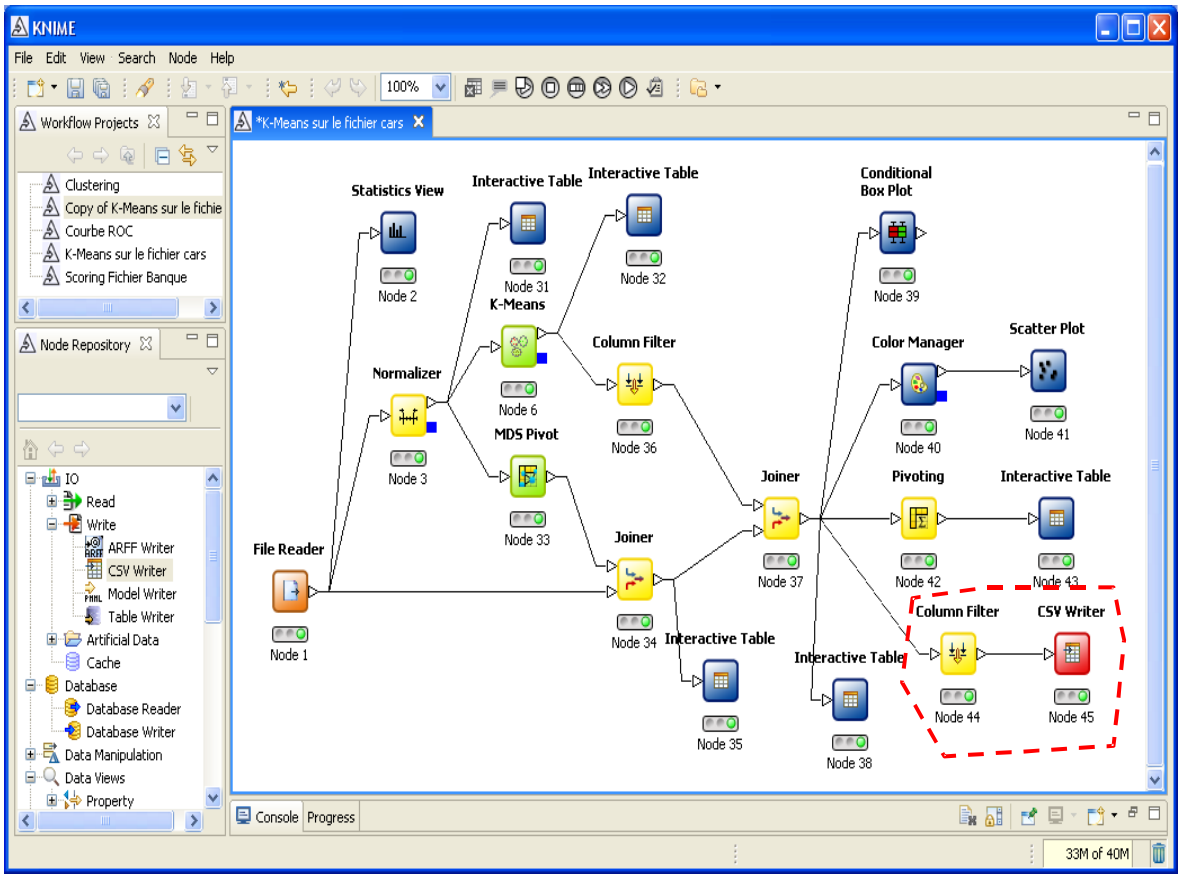
Dernier élément d'interprétation, nous croisons les classes avec la variable illustrative ORIGIN. Nous utilisons l'outil PIVOTING pour cela. Nous le configurons de manière à mettre ORIGIN en ligne, CLUSTER en colonne. Nous visualisons le résultat avec un INTERACTIVE TABLE.



5.6 Exportation des classes

Dernière opération, nous devons exporter les données en y intégrant la colonne indiquant la classe d'appartenance de chaque observation.

L'opération se fait en deux étapes : (1) filtrer les colonnes de l'ensemble de données afin de ne retenir que les variables originelles et la variable classe, nous utilisons le composant COLUMN FILTER, nous intégrons les bonnes colonnes dans la section INCLUDE de la boîte de paramétrage ; (2) exporter les données au format CSV avec CSV WRITER, nous devons spécifier le nom du fichier, nous modifions le format de fichier en choisissant « ; » comme séparateur de colonne.



Le fichier peut être chargé dans un tableur ou tout autre logiciel de data mining.

En conclusion, nous dirons que Knime est un excellent outil. Seulement, il faut rentrer dans sa logique un peu torturée (ou un peu trop « scientifique » ?) pour réaliser les opérations souhaitées. La succession des composants à poser en découragerait plus d'un dans notre exemple. Mais, passé cet écueil, on se rend compte de ses immenses possibilités. Personnellement, j'ai un peu cherché avant de trouver les bons enchaînements. Après coup, ils me paraissent évidents.

L'absence des méthodes factorielles usuelles (ACP et autres), et par contre coup l'obligation de passer par un Multidimensional Scaling, est quand même préjudiciable dans notre type d'étude, surtout lorsque le nombre d'observations augmente. Toutefois, d'après la documentation en ligne, il semble que Knime sache réaliser un MDS sur des grandes bases en passant par un échantillonnage[4]....

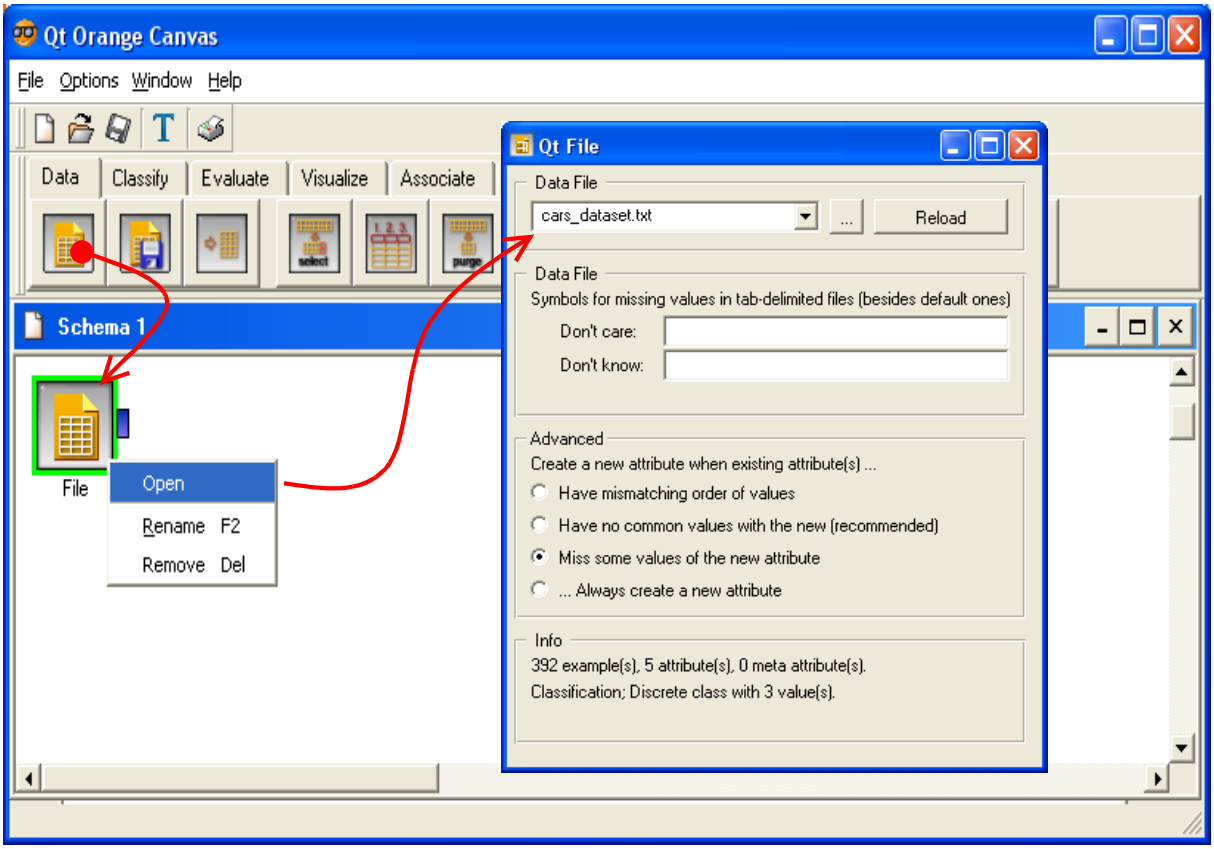
[4] Affaire à suivre dans un prochain tutoriel alors...

6 K-Means avec ORANGE

ORANGE est un excellent logiciel de Data Mining. Il intègre à la fois un fonctionnement par « filières » et un mode batch avec un véritable langage de programmation (Python). Il est librement accessible en ligne (http://www.ailab.si/orange/). Contrairement aux autres logiciels, il intègre un fichier d'aide pour chaque composant, sous forme de page web. Au delà du descriptif, des exemples traités avec des copies d'écran sont proposés. Il suffit simplement, ce n'est pas évident parce que rien ne nous met sur la piste, penser à appuyer sur la touche F1 dans les boîtes de dialogue.

6.1 Création d'un schéma et importation des données

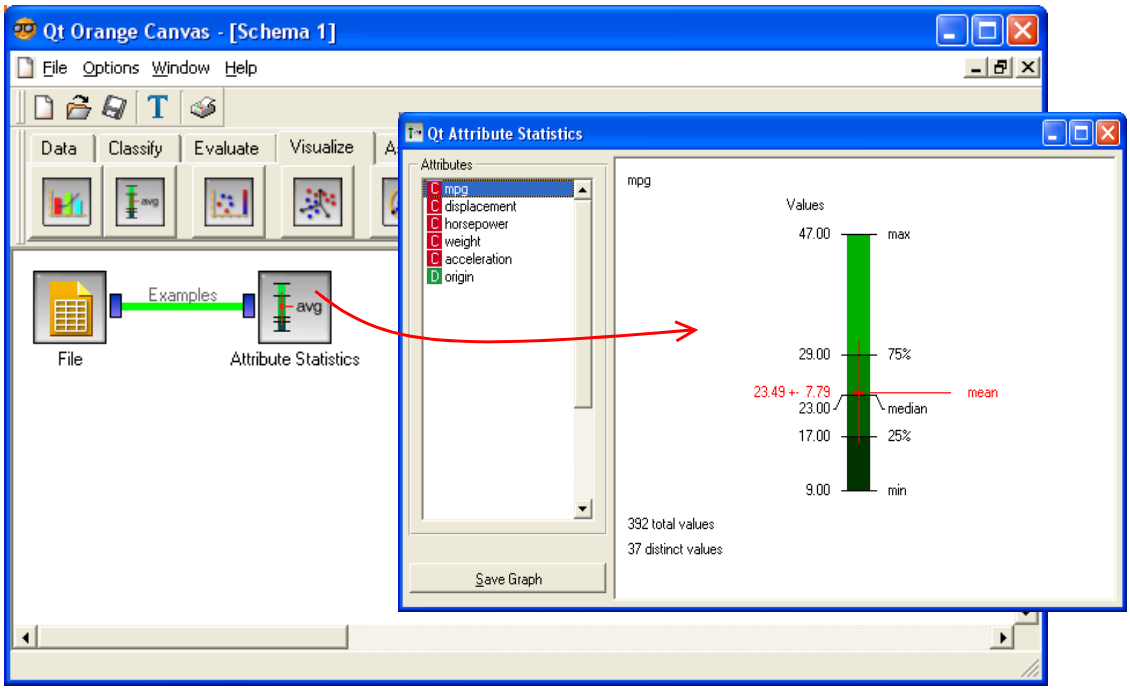
Un schéma vide (workflow, filière, diagramme) est automatiquement créé au lancement de ORANGE. Nous introduisons le « widget » (composant) d'accès aux données FILE (onglet DATA). Nous le paramétrons en actionnant le menu contextuel OPEN. Nous sélectionnons notre fichier de données CARS_DATASET.TXT.



6.2 Statistiques descriptives

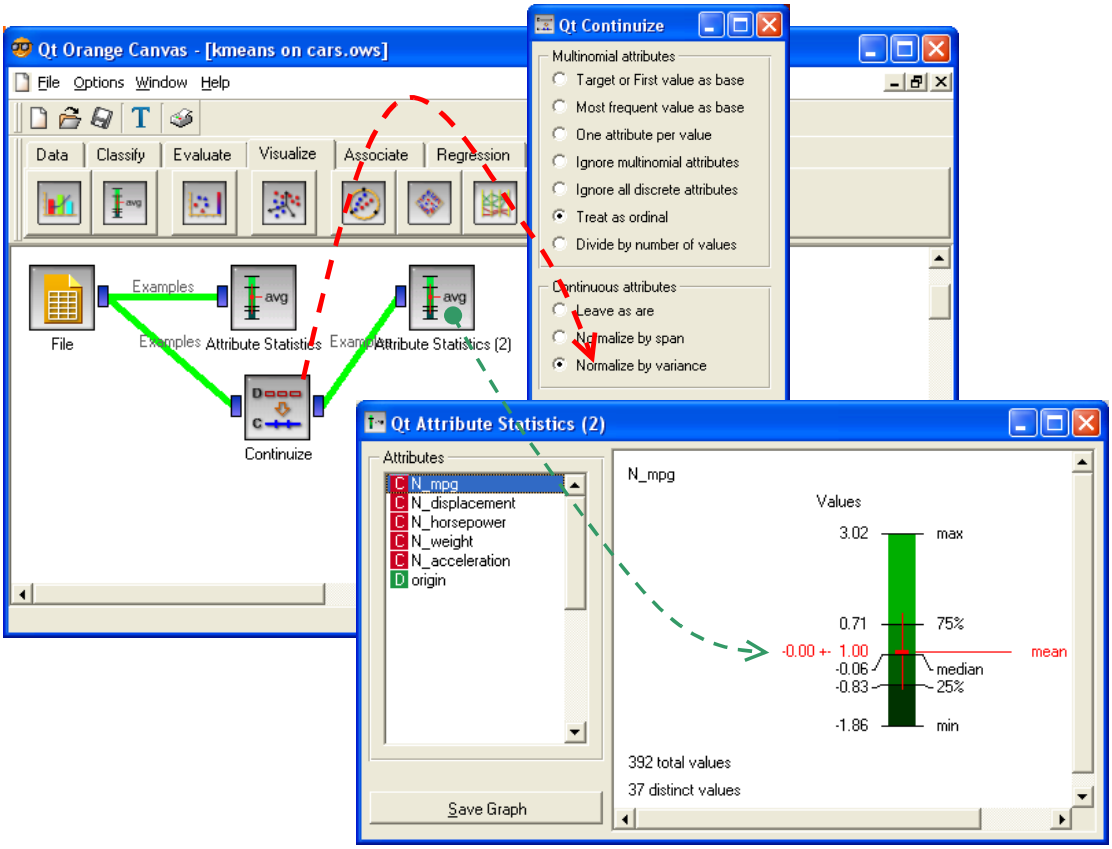
Le composant ATTRIBUTES STATISTICS propose plusieurs indicateurs de statistique descriptive. Les indicateurs de tendance centrale (moyenne, médiane), mais aussi les indicateurs de dispersion (quartile, écart type). Nous pouvons passer d'une variable à l'autre en sélectionnant celle qui nous intéresse dans la liste à gauche de la fenêtre d'affichage.

Pour une variable discrète, la distribution de fréquence est proposée.



6.3 Préparation des variables

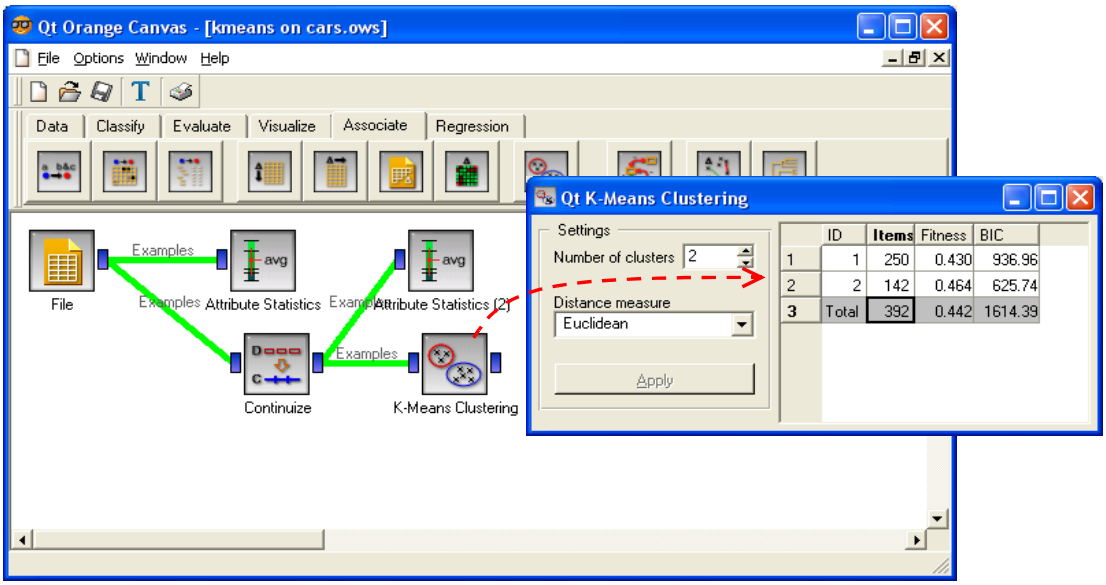
Nous devons centrer et réduire les variables que nous présenterons aux K-Means. Le composant CONTNUIZE permet de le faire facilement, il faut préciser dans la boîte de paramétrage que les variables doivent être normalisées par la variance (Orange utilise l'écart type en réalité).



Le composant ATTRIBUTE STATISTICS permet aisément de vérifier la transformation. Toutes les variables sont de moyenne 0.0 et d'écart type 1.0.

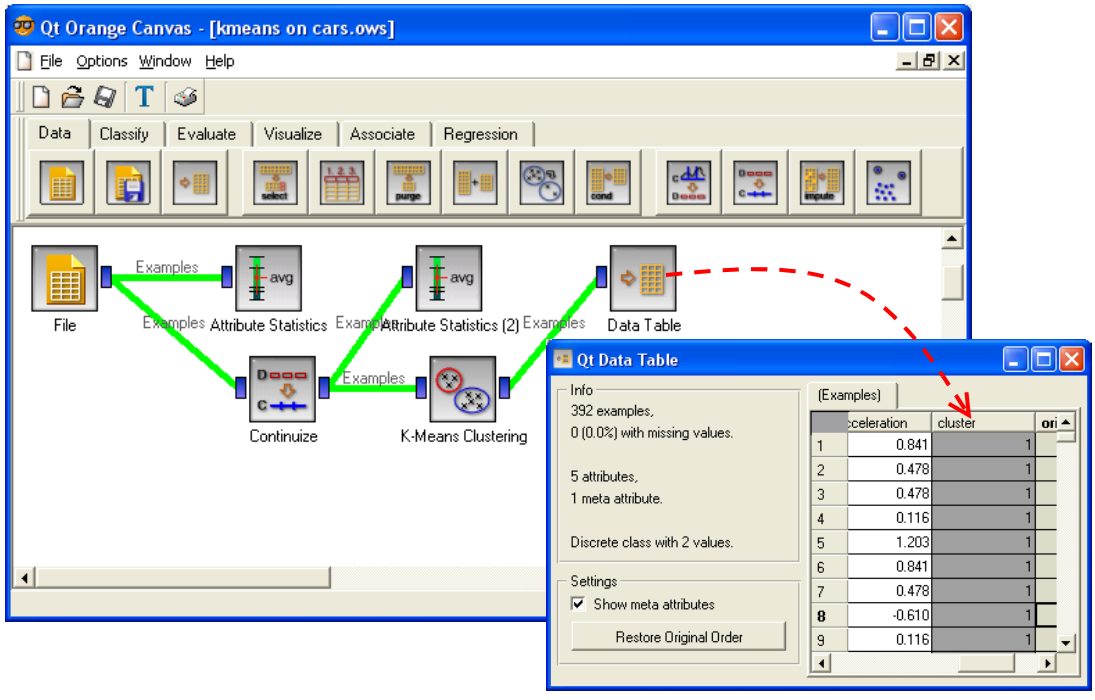
6.4 K-Means sur les variables centrées et réduites

Le composant K-MEANS CLUSTERING se trouve dans l'onglet ASSOCIATE. Nous lui connectons le composant CONTINUIZE puis nous activons le menu OPEN : nous demandons une partition en 2 groupes et nous cliquons sur APPLY. Orange affiche les effectifs par groupes (250 et 142), il propose également des indicateurs de compacité des classes. On trouvera les définitions dans la documentation en ligne.



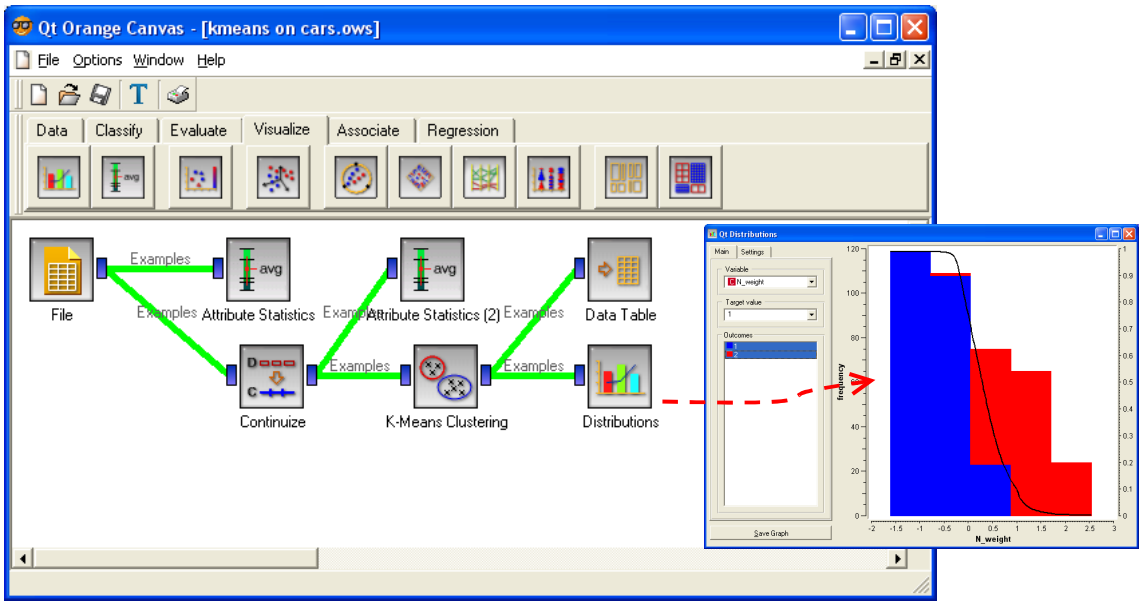
6.5 Interprétation des groupes

Affectation des classes aux individus. Comme tous les autres logiciels, Orange produit automatiquement une nouvelle colonne CLUSTER.



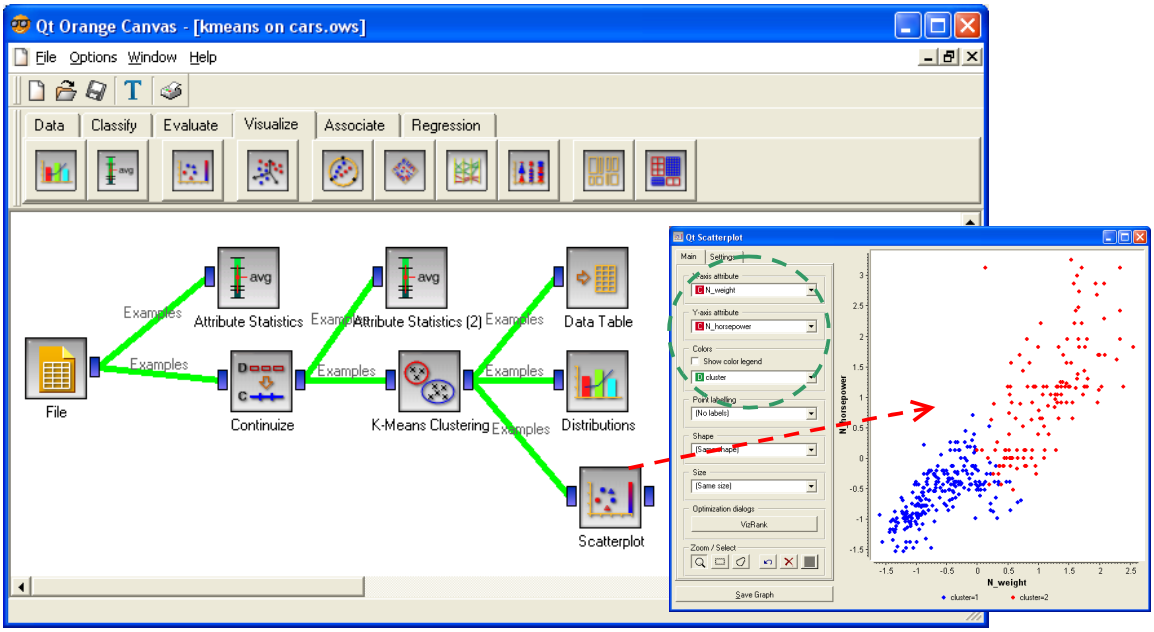
Elle décrit la classe d'appartenance de chaque individu. Nous pouvons la visualiser à l'aide du composant DATA TABLE.

Statistiques descriptives comparatives. Pour évaluer la différenciation des groupes selon chaque variable, l'outil DISTRIBUTIONS semble le plus approprié dans Orange. Il affiche les distributions de fréquences en différenciant les groupes d'appartenance. Nous lui connectons le composant K-MEANS. Dans la copie d'écran ci-dessous, nous avons les distributions conditionnelles de la variable WEIGHT.



Remarque : Malgré une recherche acharnée, je n'ai pas su récupérer les variables originelles pour les accoler avec la nouvelle colonne CLUSTER. De fait, la description statistique est réalisée sur les variables centrées et réduites dans toute la partie interprétation. Il manque cruellement un outil de type JOINER de KNIME à ce stade (en tous les cas je ne l'ai pas trouvé).

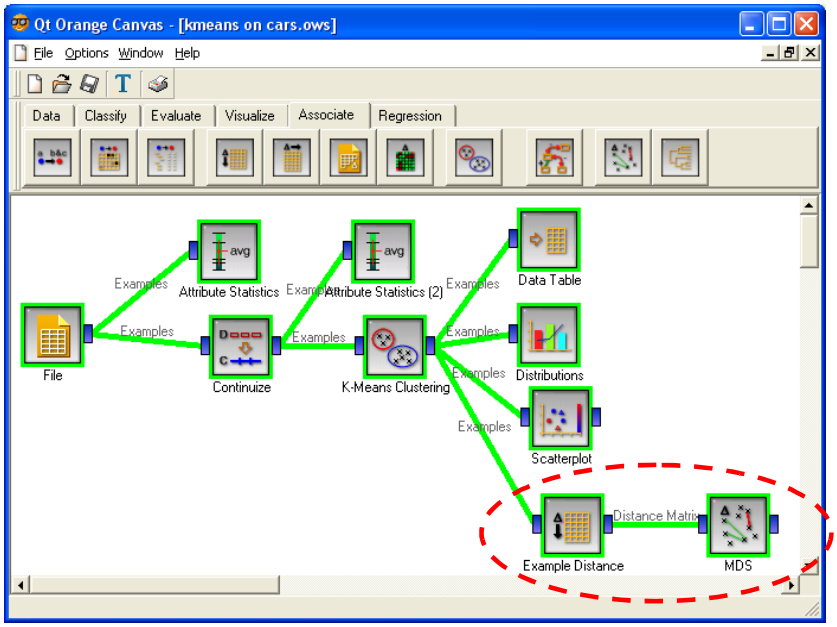
Projection dans l'espace des couples de variables. Le composant SCATTERPLOT affiche les points dans le plan. L'outil est interactif, nous pouvons choisir tout couple de variables, nous avons toute latitude pour explorer finement les données.



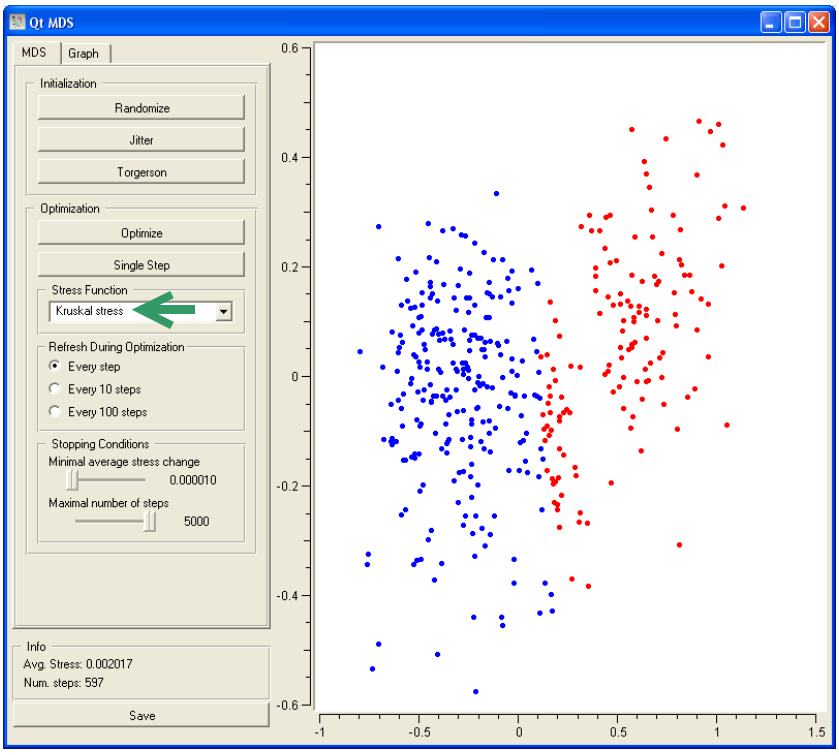
Projection dans le premier plan factoriel. L'analyse factorielle ne fait pas partie de la culture anglo-saxonne apparemment puisque, tout comme Knime, l'ACP n'est pas non plus présente dans

Orange. Nous devons passer par un multidimensional scaling. Mais, à la différence de Knime, les calculs sont assez lents dans Orange.

Deux étapes sont nécessaires : tout d'abord calculer la matrice des distances entre individus avec l'outil EXAMPLE DISTANCE (distance euclidienne dans la boîte de paramétrage) ; puis, construire les facteurs avec MDS que nous décrirons plus bas. Le schéma se présente comme suit à ce stade.

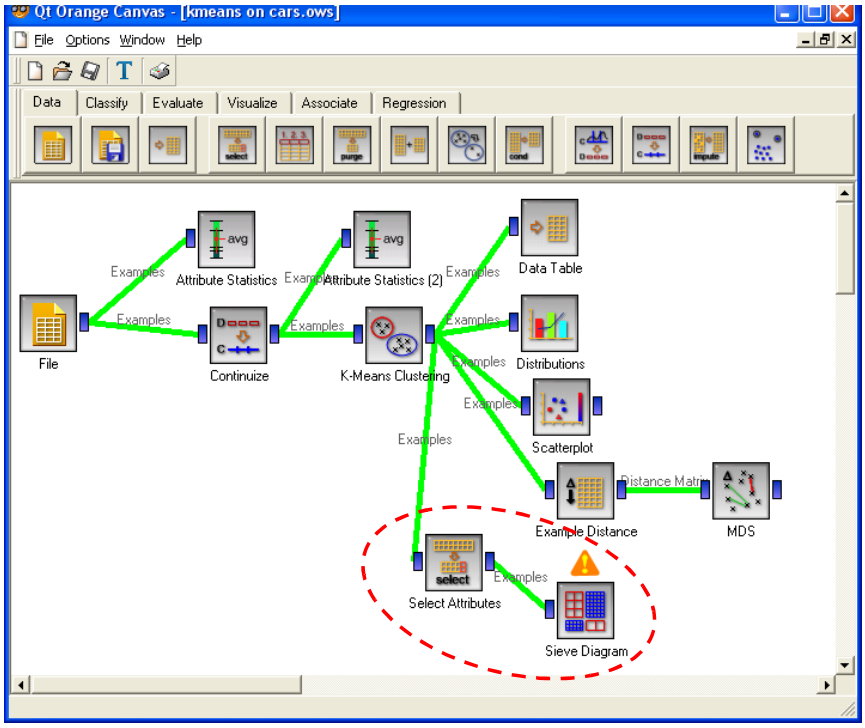


L'outil MDS est interactif, nous actionnons le menu OPEN pour accéder à la visualisation. Dans l'onglet GRAPH, nous voulons que les points soient colorisés selon la classe d'appartenance. Dans l'onglet MDS, nous choisissons la fonction « stress » de Kruskal. Nous cliquons pour OPTIMIZE. Pour spectaculaire que soit la réorganisation des points, on notera que le processus est assez lent. Comme nous l'indiquions plus haut, la similitude des résultats avec ceux de l'ACP n'est pas fortuite dans notre analyse.

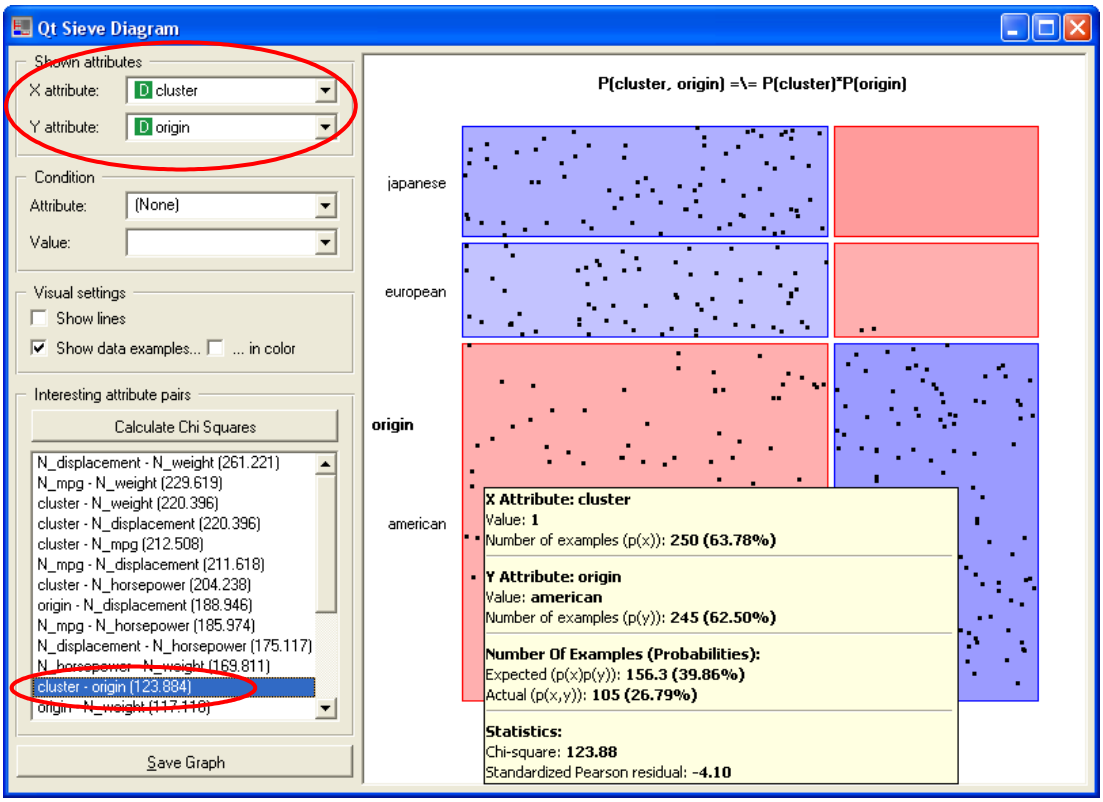


6.5.1 Croisement avec la variable illustrative

Pour croiser les variables CLUSTER et ORIGIN, nous utilisons l'outil SIEVE DIAGRAM. Nous devons au préalable re-sélectionner l'ensemble des variables pour qu'elles soient accessibles dans le reste du schéma. Avec SELECT ATTRIBUTES, nous plaçons toutes les variables en INPUT, nous insérons à la suite SIEVE DIAGRAM.



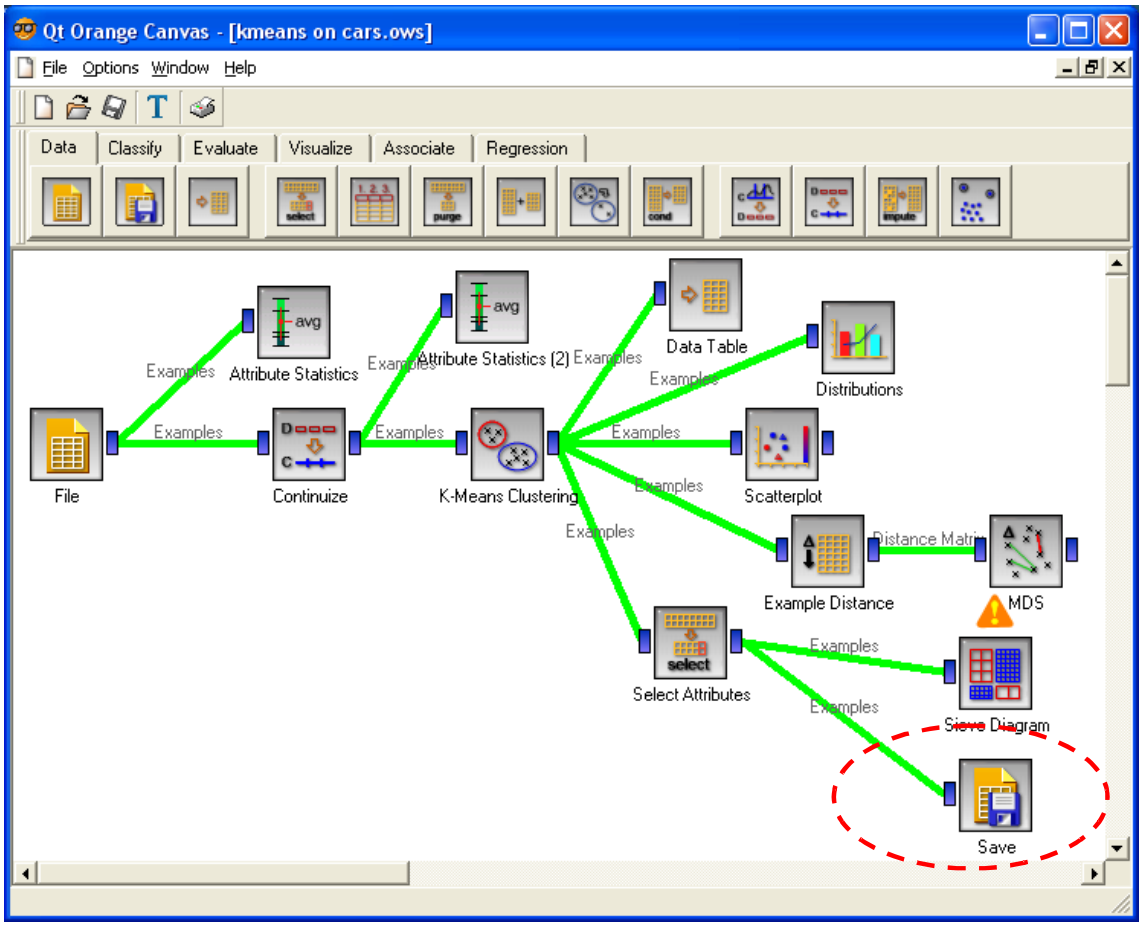
Nous actionnons le menu OPEN. Nous obtenons une présentation pour le moins originale, mais en y regardant bien nous retrouvons les informations relatives au croisement des variables.



Dans le rectangle sélectionné, correspondant au croisement CLUSTER = 1 et ORIGIN = AMERICAN, Orange nous indique que 105 observations s'y trouvent. Sous l'hypothèse d'indépendance, nous y aurions trouvé 156.3 individus. Le KHI-2 global du test d'indépendance est égal à 123.884.

6.6 Exportation des résultats

Dernière étape de notre processus, nous souhaitons exporter l'ensemble de données comprenant la nouvelle colonne CLUSTER. Nous insérons le composant SAVE dans le schéma, nous le configurons en désignant le fichier de sortie, plusieurs formats sont possibles. Comme je n'ai pas su récupérer les variables initiales, ce sont les variables centrées et réduites qui seront exportées.



7 K-Means avec RAPIDMINER

RAPIDMINER (http://rapid-i.com/content/blogcategory/38/69/) est le successeur de YALE. Deux versions sont disponibles, nous nous intéressons ici à la version libre « Community Edition ».

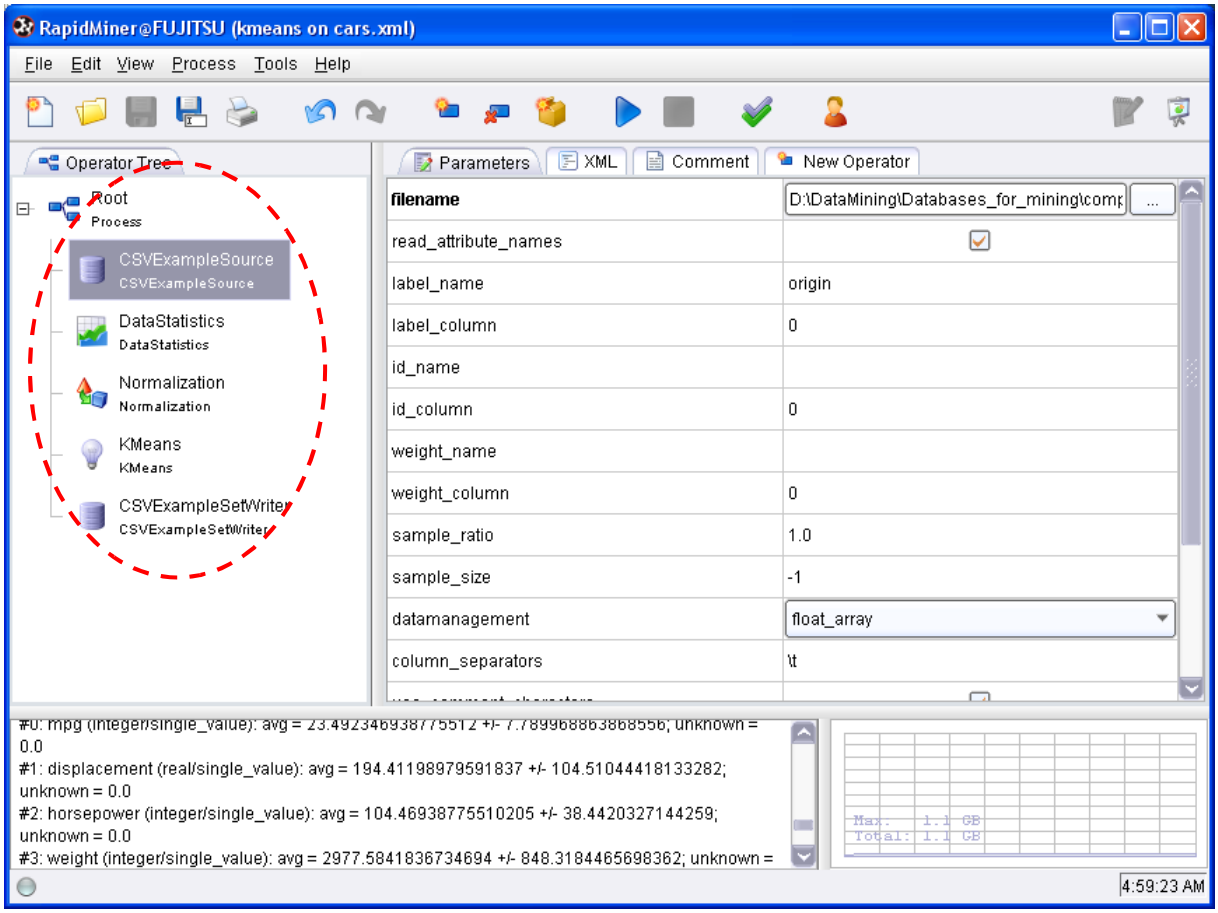
RAPIDMINER fonctionne en mode filière avec une disposition des opérateurs qui le rapprocherait de Tanagra. Les séquences de traitements sont présentées sous une forme arborescente, à la différence que le traitement principal est réalisé sur le premier niveau de l'arbre, sous une forme linéaire. Les sous branches correspondent en fait à des sortes de procédures, des blocs de traitements, qui prennent en entrée les informations en provenance du composant qui les précédent au premier niveau, et renvoient des informations au composant qui les succèdent, toujours au premier niveau. L'exemple de la validation croisée décrit sur le site du logiciel est peut être le plus parlant pour comprendre le mécanisme de « l'arbre des opérateurs » de RAPIDMINER.

Autre spécificité de RAPIDMINER, il intègre un nombre impressionnant de méthodes, près de 500, dont une centaine proviennent de WEKA[5]. Impressionnant, mais déroutant, l'utilisateur passe parfois beaucoup de temps à chercher le bon outil pour réaliser certains traitements pourtant simples.

Enfin, dernière particularité, il n'est pas possible de lancer l'exécution des opérateurs au fur et à mesure de leur insertion dans l'arbre. A chaque fois que l'on actionne le bouton PLAY (la flèche bleue dans la barre d'outils), tout le diagramme est ré exécuté. Fort heureusement, le traitement est très rapide. Pour cette raison, par rapport aux autres logiciels de ce comparatif, nous adoptons une approche différente : nous définissons tout d'abord l'arbre de traitements complet, puis nous lançons tous les calculs d'une traite.

7.1 Définition de l'arbre des traitements

Le traitement complet se décline comme suit.



Nous y discernons les successions d'opérations suivantes : accès aux données ; statistiques descriptives sur les variables originelles ; standardisation ; typologie avec la méthode des K-Means, écriture de l'ensemble de données complété de la colonne CLUSTER dans un fichier au format CSV.

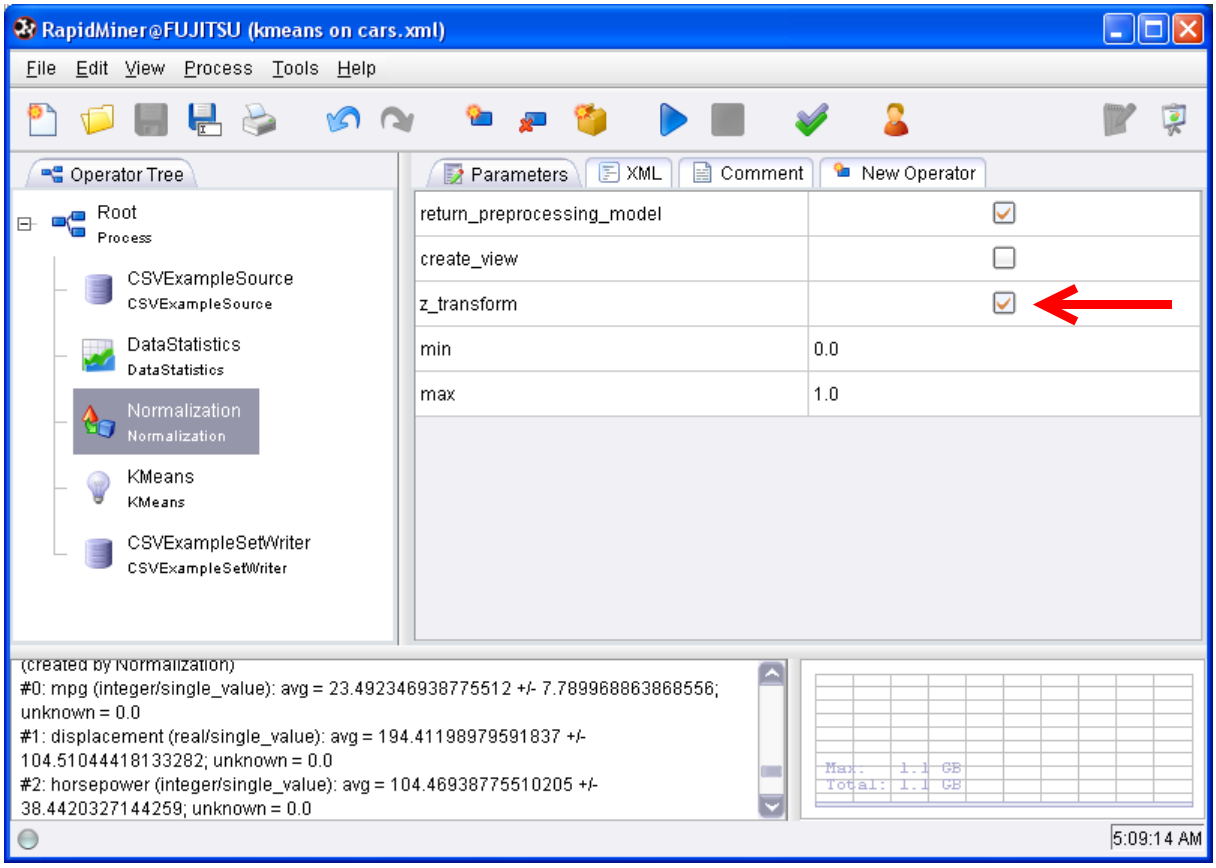
Remarque : RAPIDMINER intègre un composant PCA (analyse en composantes principales). Il est possible de projeter les classes dans le premier plan factoriel. Mais pour une raison qui m'échappe, alors que je l'ai placé après le K-Means, ce dernier s'est entêté à réaliser la typologie à partir des axes factoriels. Ce n'est pas gênant en soi, ça peut même être une excellente stratégie, mais ce n'est pas ce que je voulais faire. L'idée a donc été abandonnée.

[5] Il y a une controverse, complètement ridicule, à ce sujet - http://www.kdnuggets.com/news/2007/n24/5i.html

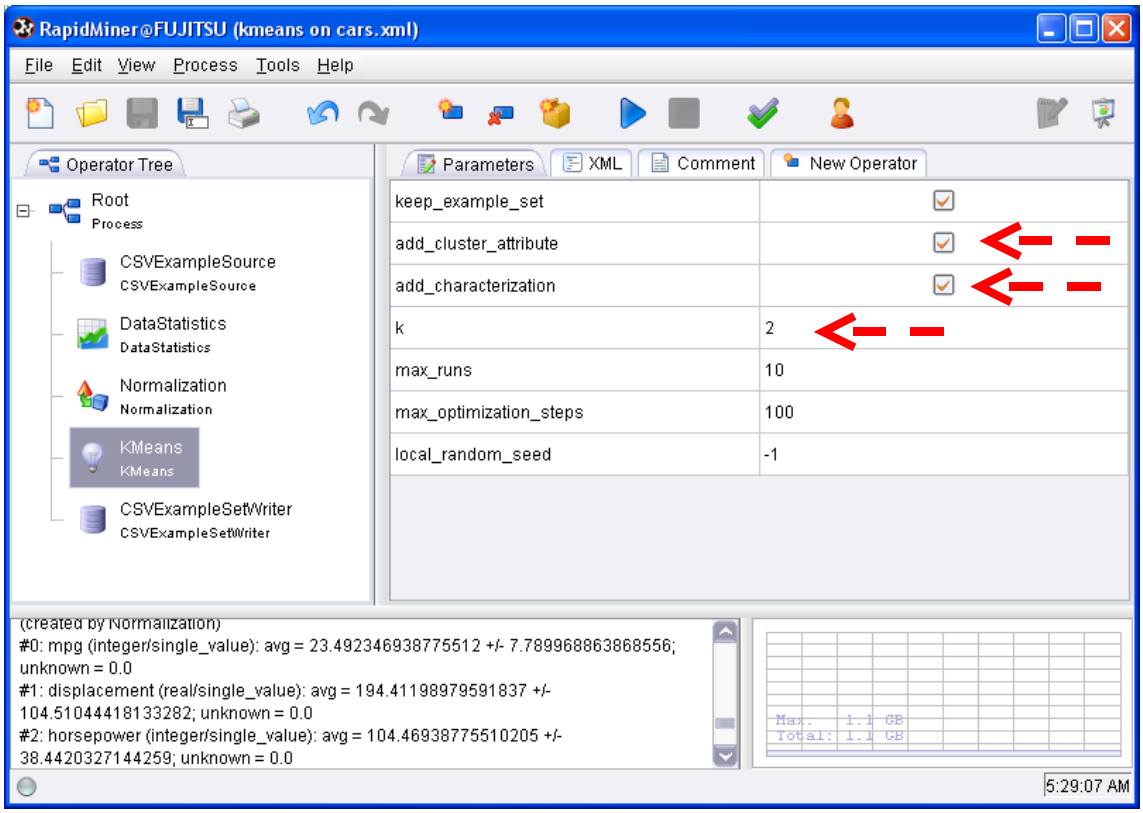
Accès aux données. Le composant CSVEXAMPLESOURCE permet d'accéder aux données. Les paramètres les plus importants sont FILENAME qui désigne le fichier à charger ; LABEL_NAME que nous fixons à ORIGIN, ça n'a pas tellement de sens dans notre étude, ce choix permet seulement d'évacuer la variable illustrative de la construction des classes ; COLUMN_SEPARATORS que nous fixons à « \t » pour désigner le caractère « tabulation » comme séparateur de colonnes.

Description des données. DATASTATISTICS permet de décrire les données via des indicateurs de statistique descriptive. Le composant ne comporte pas de paramètres.

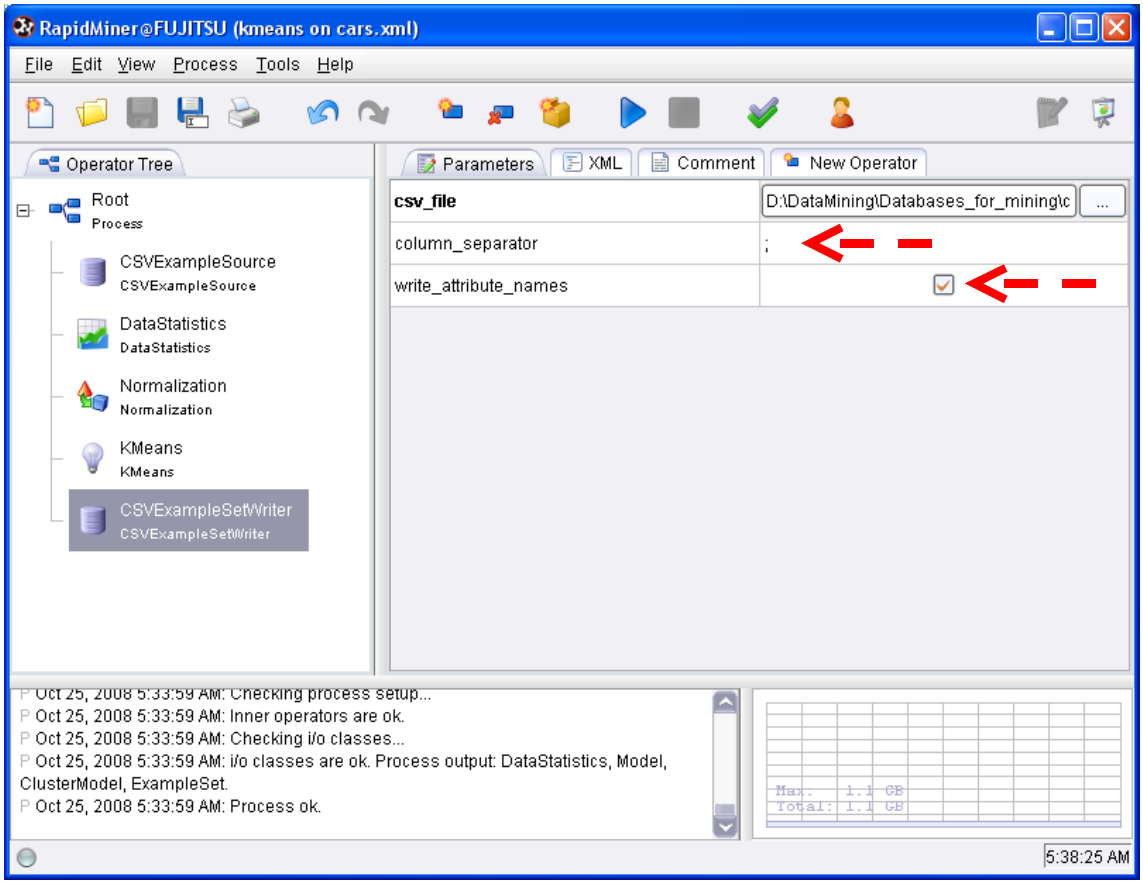
Centrage – réduction des variables actives. Le composant NORMALIZATION sert à centrer et réduire les variables actives. Il propose différentes transformations, nous demandons la transformation Z.



La méthode des K-Means. Pour initier la classification, nous insérons le composant KMEANS. Nous le paramétrons de manière à produire 2 classes (K) et à obtenir en sortie une nouvelle variable CLUSTER (ADD_CLUSTER_ATTRIBUTE). Nous demandons de plus à ce que RAPIDMINER les caractérise (ADD_CHARACTERIZATION). Cela consiste essentiellement à calculer des statistiques descriptives comparatives sur les variables actives. Les autres paramètres sont relatifs aux calculs (MAX_RUNS : nombre d'essais, MAX_OPTIMIZATION_STEPS : nombre d'étapes maximum lors d'un essai).



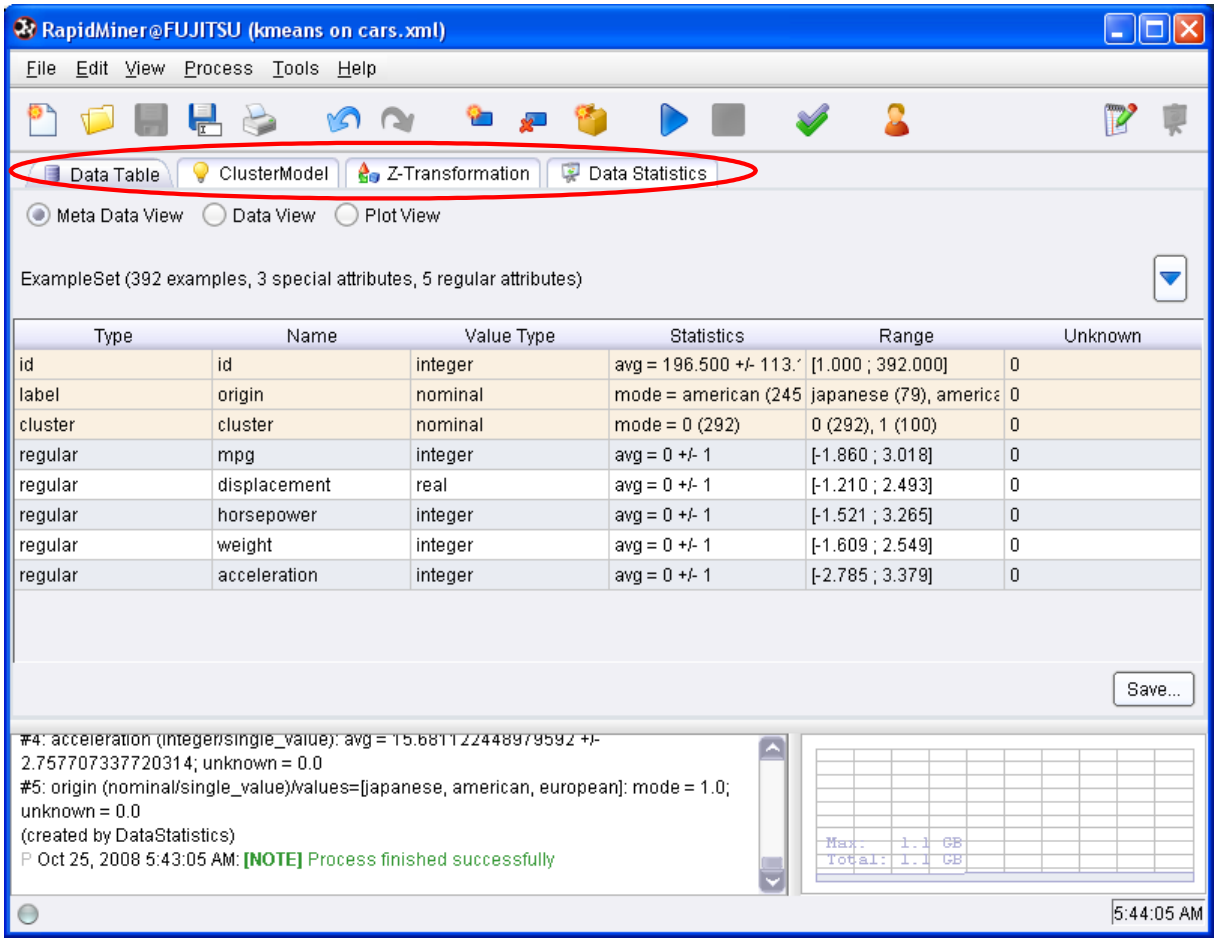
Exportation des résultats. Dernière étape, nous exportons les résultats c.-à-d. l'ensemble de données complété par la colonne CLUSTER à l'aide du composant CSVEXAMPLESETWRITER.



Nous spécifions le nom du fichier et le séparateur de colonnes. Comme dans ORANGE, je n'ai pas su retrouver les données originelles, l'exportation intégrera donc les variables centrées et réduites.

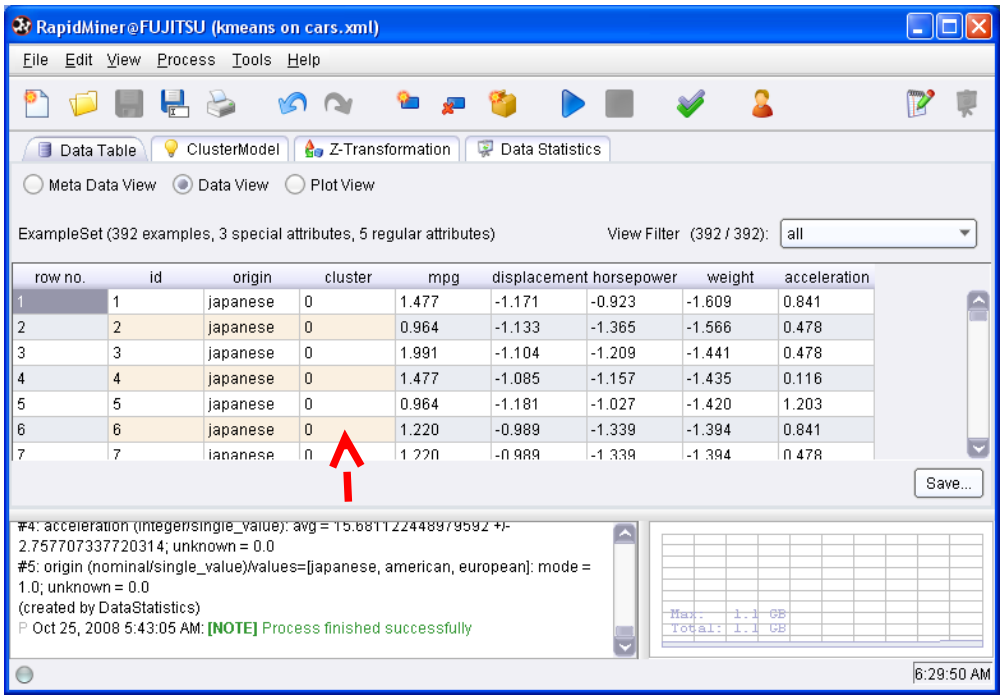
7.2 Analyse des résultats

Après avoir sauvegardé l'arbre des opérations, nous actionnons le bouton pour lancer les calculs. RAPIDMINER propose une fenêtre globale de visualisation des résultats. Il est possible de passer d'un composant à l'autre en sélectionnant l'onglet adéquat.

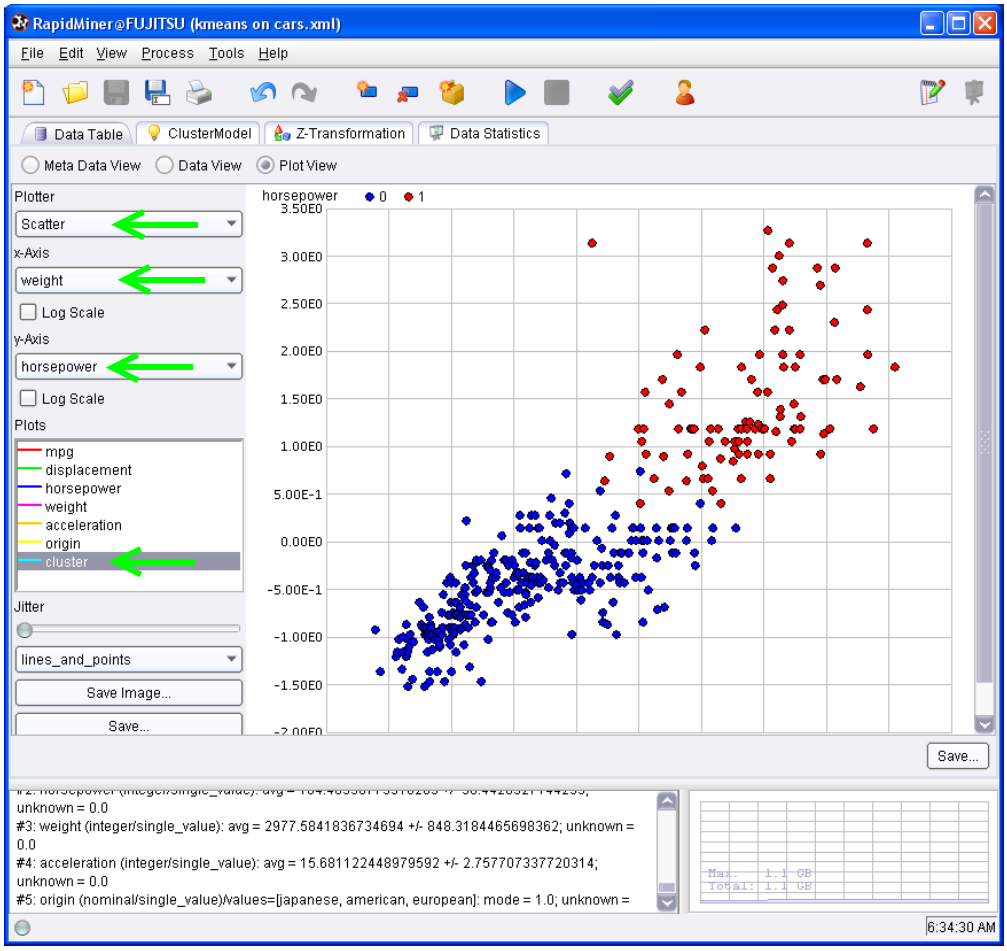


Description des données. L'onglet DATA TABLE décrit les données de plusieurs manières. META DATA VIEW résume les principales caractéristiques, le type des variables, les plages de valeurs, la moyenne et l'écart type. Dans la copie d'écran ci-dessus, ce sont les variables centrées et réduites qui sont recensées.

L'option DATA VIEW affiche les valeurs dans une grille en incluant la colonne CLUSTER.

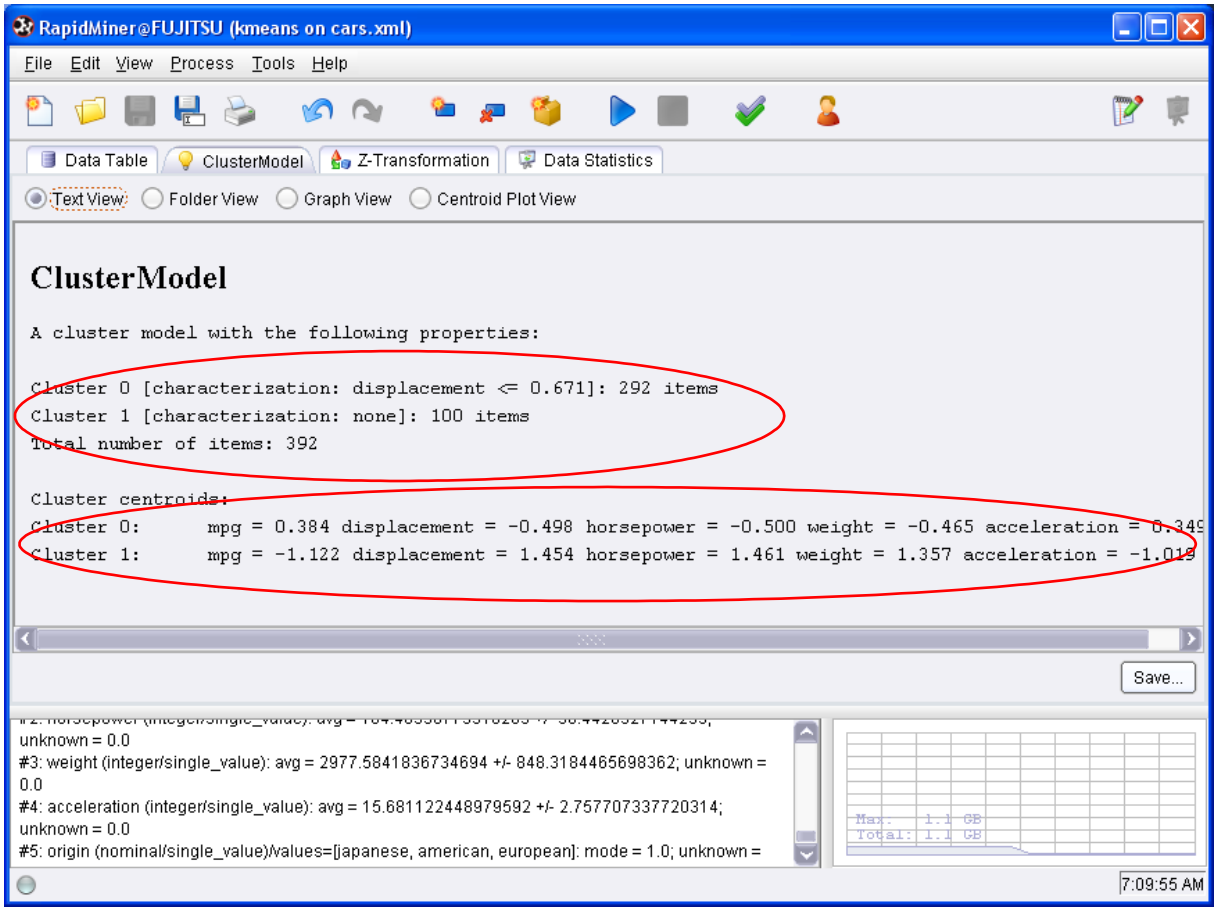


PLOT VIEW est un outil graphique. Plusieurs configurations sont possibles. Nous souhaitons projeter les points dans l'espace WEIGHT x HORSEPOWER en les illustrant avec la classe d'appartenance. Nous utilisons l'option SCATTER.



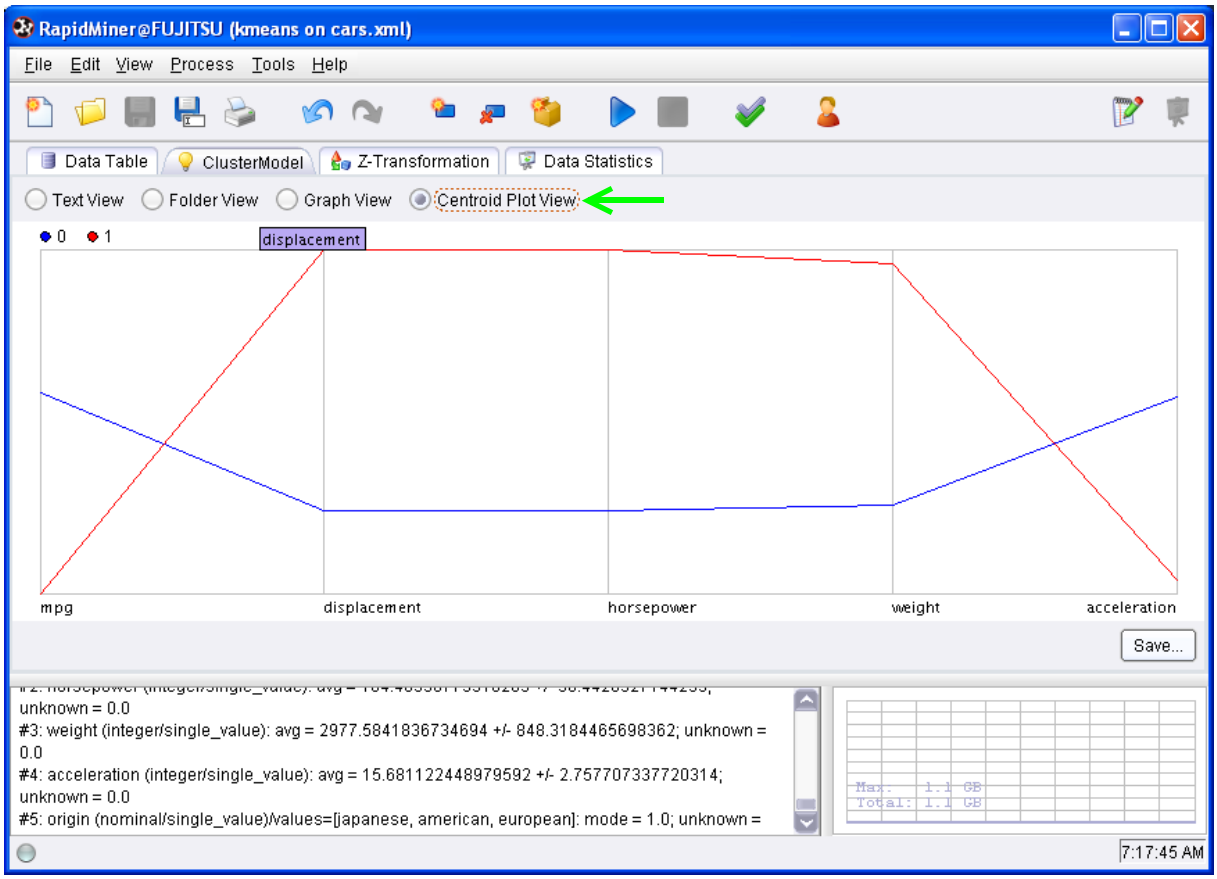
Nous retrouvons un graphique bien connu maintenant, avec l'opposition « voitures légères peu puissantes » et « lourdes berlines puissantes ». Les classes sont quasiment parfaitement discernables dans l'espace défini par ce couple de variables.

CLUSTERMODEL. Cet onglet décrit les résultats de la classification automatique. Plusieurs options sont disponibles. TEXT VIEW décrit l'effectif des classes. Il y a 292 observations dans la première, 100 dans la seconde. Nous disposons des moyennes conditionnelles, calculées dans l'espace des variables centrées et réduites cependant.



Les options FOLDER VIEW et GRAPH VIEW permettent avant tout de consulter la liste des observations dans les classes. Leur intérêt est limité, surtout lorsqu'il y a un nombre d'observations assez important, anonymes de surcroît.

CENTROID PLOT VIEW affiche les séries de moyennes conditionnelles dans un graphique, sous la forme d'une courbe. Les variables correspondent à des items. Mis à part l'aspect visuel, ce graphique n'apporte rien de plus par rapport à TEXTVIEW. Il serait réellement intéressant lorsque le nombre de variable est très élevé. En un coup d'œil nous isolons les variables ou groupes de variables où la différenciation est forte. Bien entendu, pour profiter pleinement de l'outil, on a intérêt à ce que les variables soient ramenées à la même échelle, ce qui est le cas pour notre exemple.



2 autres onglets complète l'affichage des résultats **:**

- **Z-TRANFORM.** Cet onglet décrit les paramètres utilisés pour transformer les données. En l'occurrence la moyenne et la variance puisque nous les avons centrées et réduites.
- **DATA STATISTICS** calcule les indicateurs de la statistique descriptive. Moyenne et variance pour les variables quantitatives, mode pour les variables discrètes. Fonctionnalité intéressante, RAPIDMINER affiche l'occurrence éventuelle de données manquantes.

8 Conclusion

Je ne me lasserai jamais de le dire : il n'y a pas de bons ou mauvais logiciels. Il y a des logiciels qui répondent ou non à des spécifications. Notre rôle est de les exprimer le plus clairement possible. C'est ce que nous avons essayé de faire dans l'introduction de ce didacticiel. Après, il nous appartient de choisir le bon outil.

Parfois, il est nécessaire de réviser nos exigences. Nous sommes plus ou moins obligés de les adapter aux possibilités des logiciels à notre disposition. En connaître plusieurs est une excellente manière de lever cette contrainte.