Discrétisation des variables quantitatives

Découpages en classes des variables quantitatives

Ricco RAKOTOMALALA

Université Lumière Lyon 2

PLAN

1. Qu'est-ce que la discrétisation ? Pourquoi la discrétisation ?
2. Techniques non supervisées de discrétisation
3. Techniques supervisées de discrétisation
4. Bilan
5. Bibliographie

Discrétisation ? Pourquoi faire ?

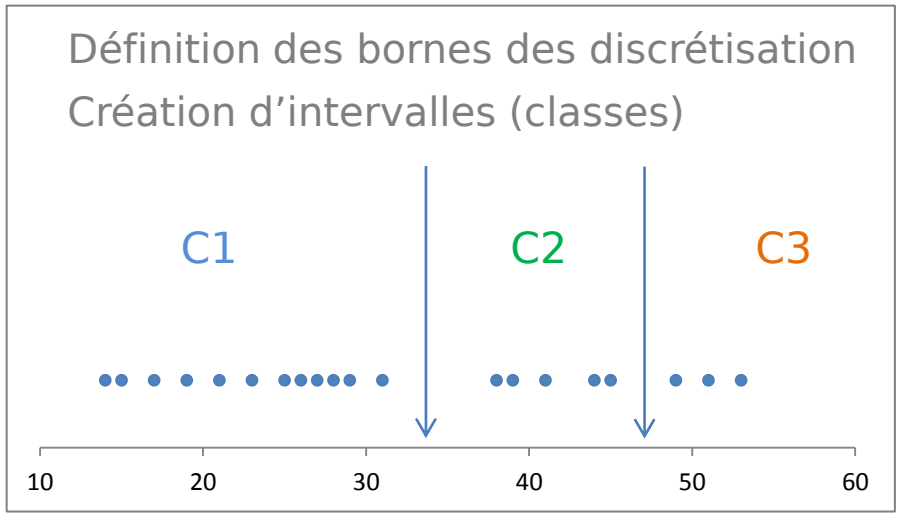
Principe et intérêt du découpage en classes d'une variable

Découpage en classes - Démarche

X	Classes
14	C1
15	C1
17	C1
19	C1
21	C1
23	C1
25	C1
26	C1
27	C1
28	C1
29	C1
31	C1
38	C2
39	C2
41	C2
44	C2
45	C2
49	C3
51	C3
53	C3

Variable X **quantitative**

Transformée en variable « Classes », **qualitative** (ordinale)



(1) Comment choisir le nombre d'intervalles K ?

(2) Comment choisir les bornes (seuils) de découpage ?

… Qui soient pertinents par rapport au problème étudié ?

Pourquoi la discrétisation ?

- Certaines techniques statistiques ne fonctionnent qu'avec des variables qualitatives

  Ex. Induction de règles (règles prédictives, règles d'association)

- Harmonisation du type des variables dans les tableaux hétérogènes

  (Note : le chemin inverse peut être envisagé, rendre numériques les variables qualitatives nominales ou ordinales)

  Ex. Analyse factorielle. Discrétisation des variables quantitatives + ACM sur le tableau harmonisé.

- Modifier les caractéristiques des données pour rendre les algorithmes de statistiques subséquents plus efficaces

  Ex. Corriger les distributions très asymétriques, atténuer le rôles des points aberrants.

Méthode de référence – Découpage d'expert

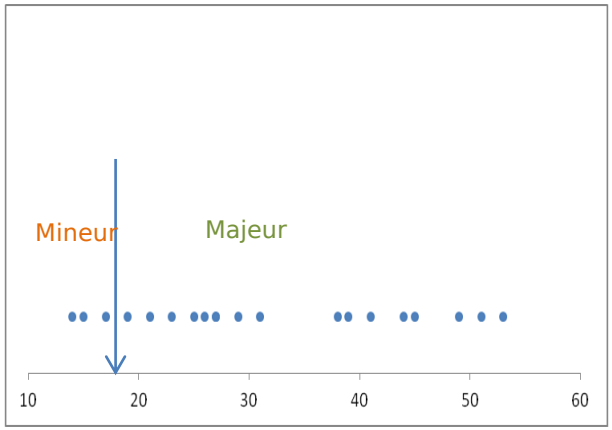
En se basant sur les connaissances du domaine, l'expert peut proposer le découpage le plus adapté au problème posé.

X = Age

La majorité est un critère possible de découpage par rapport à l'analyse à mettre en place

X < 18 : mineur
X ≥ 18 : majeur

Age	Majorité
14	Mineur
15	Mineur
17	Mineur
19	Majeur
21	Majeur
23	Majeur
25	Majeur
26	Majeur
27	Majeur
28	Majeur
29	Majeur
31	Majeur
38	Majeur
39	Majeur
41	Majeur
44	Majeur
45	Majeur
49	Majeur
51	Majeur
53	Majeur



☹ La solution n'est pas toujours aussi évidente, l'expertise est rare.

➔ **Mettre en place une solution guidée par les (caractéristiques) des données**

Discrétisation non supervisée

S'appuyer les caractéristiques intrinsèques de X
pour produire un découpage « pertinent »

Quelques caractéristiques disponibles

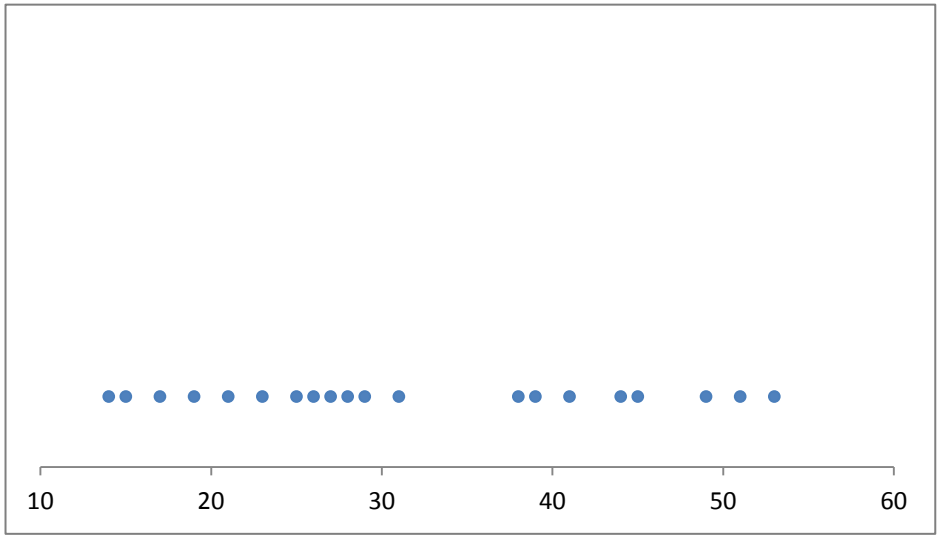
X
14
15
17
19
21
23
25
26
27
28
29
31
38
39
41
44
45
49
51
53

n	20

Min	14
Max	53

1er quartile	22.5
Mediane	28.5
3e quartile	41.75

Moyenne	31.75
Ecart-type	12.07



Comment s'appuyer sur ces informations pour produire un découpage qui tient la route ?

Méthodes « usuelles »

Intervalles de largeurs (amplitudes) égales (1)

K : nombre d'intervalles est fixé (comment ?)

Construire des intervalles d'amplitudes égales

$$a = \frac{\max - \min}{K}$$

Calcul de l'amplitude à partir des données (a)

On en déduit les (K-1) bornes (b_1, b_2, etc.)

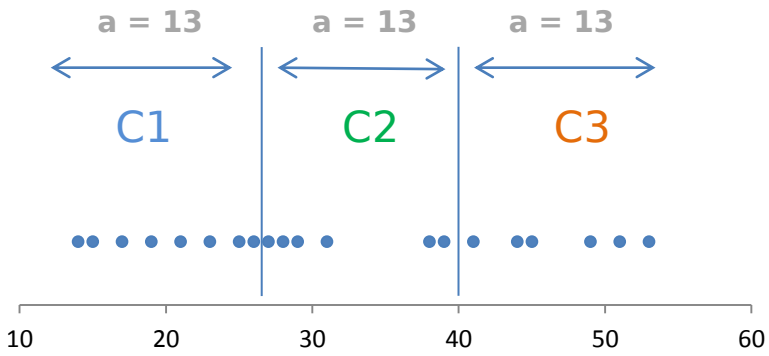
$$b_1 = \min + a$$

$$b_2 = b_1 + a = \min + 2 \times a$$

$$\ldots$$

X	Classes
14	C1
15	C1
17	C1
19	C1
21	C1
23	C1
25	C1
26	C1
27	C2
28	C2
29	C2
31	C2
38	C2
39	C2
41	C3
44	C3
45	C3
49	C3
51	C3
53	C3

K	3
max	53
min	14
a	**13**
b1	**27.0**
b2	**40.0**



C1 : x < 27

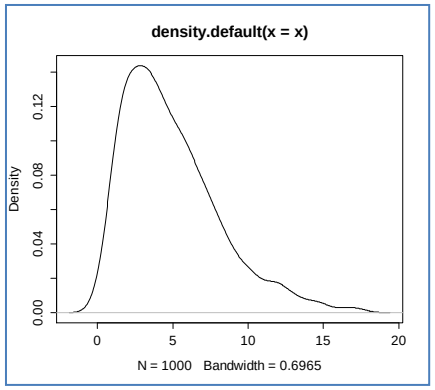
C2 : 27 ≤ x < 40

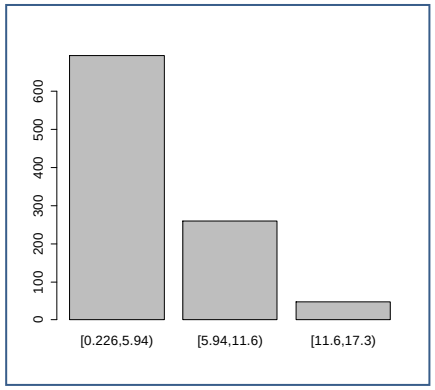
C3 : x ≥ 40

Le sens de l'inégalité est arbitraire

Intervalles de largeurs (amplitudes) égales (2)

+ Rapidité de calcul et simplicité (facile à expliquer)
+ Ne modifie pas la forme de la distribution des données





- Choix de K arbitraire, pas toujours évident
- Sensibilité aux points extrêmes (min ou max)
- Possibilité d'avoir des intervalles avec très peu d'individus voire vides

Intervalles de fréquences égales (1)

K nombre d'intervalles est fixé (comment ?)

Construire des intervalles de fréquence égales

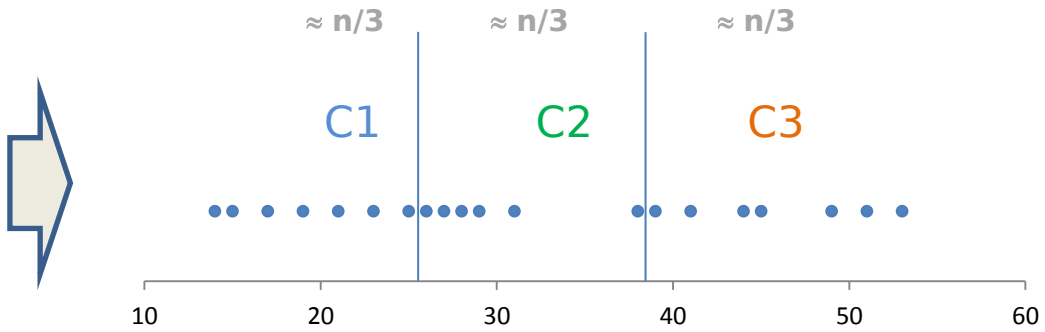
Calcul des quantiles à partir des données (q_1, q_2)

Quantiles = bornes (q_1, q_2, etc.)

Ex. quantile d'ordre 0.25 = 1er quartile ; quantile d'ordre 0.5 = médiane ; etc.

X	Classes
14	C1
15	C1
17	C1
19	C1
21	C1
23	C1
25	C1
26	C2
27	C2
28	C2
29	C2
31	C2
38	C2
39	C3
41	C3
44	C3
45	C3
49	C3
51	C3
53	C3

q(0.33)	25.33
q(0.66)	38.67



C1 : x < 25.33

C2 : 25.33 ≤ x < 38.67

C3 : x ≥ 38.67

Le sens de l'inégalité est arbitraire

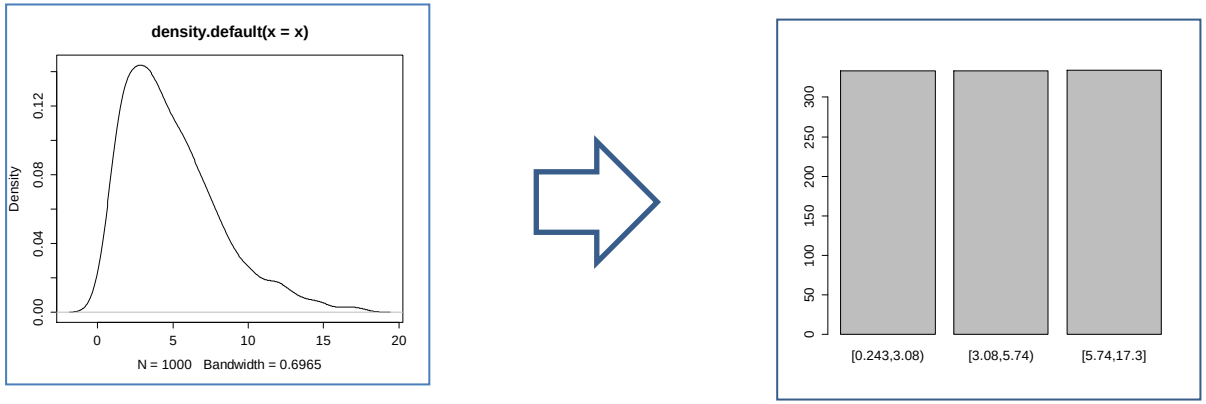
Intervalles de fréquences égales (2)

Rapidité (~) de calcul et simplicité (facile à expliquer)

« Lissage » des points extrêmes

Intervalles avec un nombre déterminé d'individus

Egalise la distribution des données



Choix de K arbitraire, pas toujours évident

Seuils ne tenant pas compte des proximités entre les valeurs

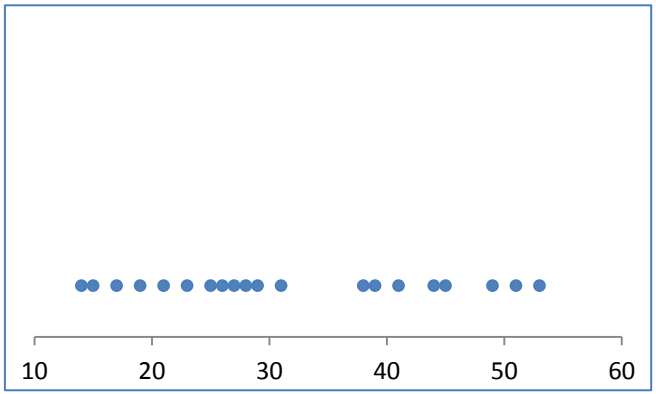
Quelques formules pour « détecter » la bonne valeur de K

(http://www.info.univ-angers.fr/~gh/wstat/discr.php)

Appellation	Formule	K « calculé » (sans arrondi)
Brooks-Carruthers	$5 \times \log_{10}(n)$	6.51
Huntsberger	$1 + 3.332 \times \log_{10}(n)$	5.34
Sturges	$\log2(n + 1)$	4.39
Scott	$(max - min)/(3.5 \times \sigma \times n^{-1/3})$	2.50
Freedman-Diaconis	$(max - min)/(2 \times IQ \times n^{-1/3})$	2.75

σ: écart-type

IQ : intervalle interquartiles



Les deux dernières approches exploitent plus d'informations en provenance des données.

Moyennes emboîtées. Algorithme descendant. On découpe avec la moyenne. Puis, de part et d'autre de ce premier seuil, on découpe avec les moyennes locales respectives, etc. Nombre de classes est forcément une puissance de 2.

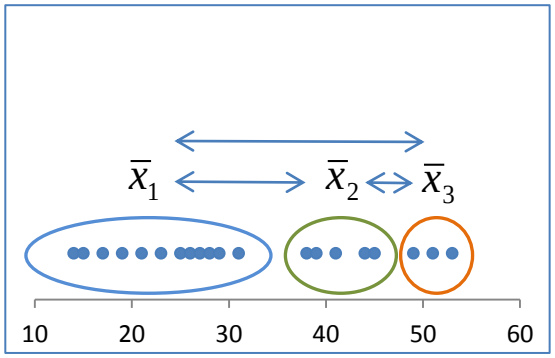
Grandes différences relatives. On trie les données de manière croissante. On repère les grands écarts entre 2 valeurs successives. On découpe si écart > seuil exprimé en % de l'écart-type des valeurs (ou en % du MAD – median absolute deviation – si l'on souhaite se prémunir du problème des valeurs aberrantes).

Ecart à la moyenne. K fixé. Si K est pair, de part et d'autre de la moyenne, les premiers intervalles sont de largeur σ [ou m x σ, m est un paramètre], etc. jusqu'à ce qu'on ait K intervalles en tout [les derniers intervalles en queue de distribution ont une largeur différente]. Si K est impair, le premier intervalle est à cheval autour de la moyenne.

Méthodes tenant compte de la dispersion des classes

Tenir compte des proximités entre les valeurs

Les données peuvent être organisées en « paquets » plus ou moins homogènes. On s'intéresse aux caractéristiques de dispersion des données.



Equation d'analyse de variance (ANOVA)

$$T = B + W$$

$$\sum_{i=1}^{n}(x_i - \bar{x})^2 = \sum_{k=1}^{K} n_k(\bar{x}_k - \bar{x})^2 + \sum_{k=1}^{K}\sum_{i=1}^{n_k}(x_{ik} - \bar{x}_k)^2$$

Objectif : chercher à maximiser l'écartement relatif entre les moyennes conditionnelles

$$\eta^2 = \frac{B}{T}$$

η est le rapport de corrélation

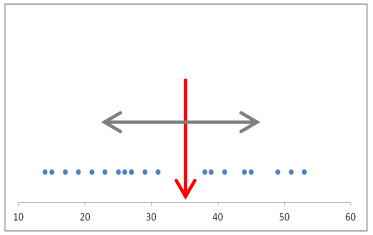
Il existe une approche « optimale » (Algorithme de Fisher) à K fixé, mais...

☹ K doit être fixé toujours

☹ Algorithme complexe et lent, disponible nulle part

Algorithme descendant (1)

(1)

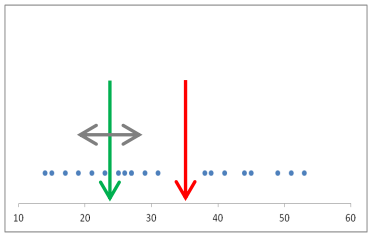


Trouver la meilleur séparation binaire

Continuer récursivement dans chaque sous-groupe

Jusqu'à déclenchement des règles d'arrêt

(2)



Etc.

Quelles règles d'arrêt ?

- Une subdivision n'engendre plus un écartement significatif (paramètre α, attention aux comparaisons multiples, corriger ou réduire fortement α)
- Effectifs dans les classes (paramètres « effectif nécessaire » avant et après subdivision)
- Nombre maximum d'intervalles (de classes) à produire

Algorithme descendant (2) – Utiliser les arbres de régression

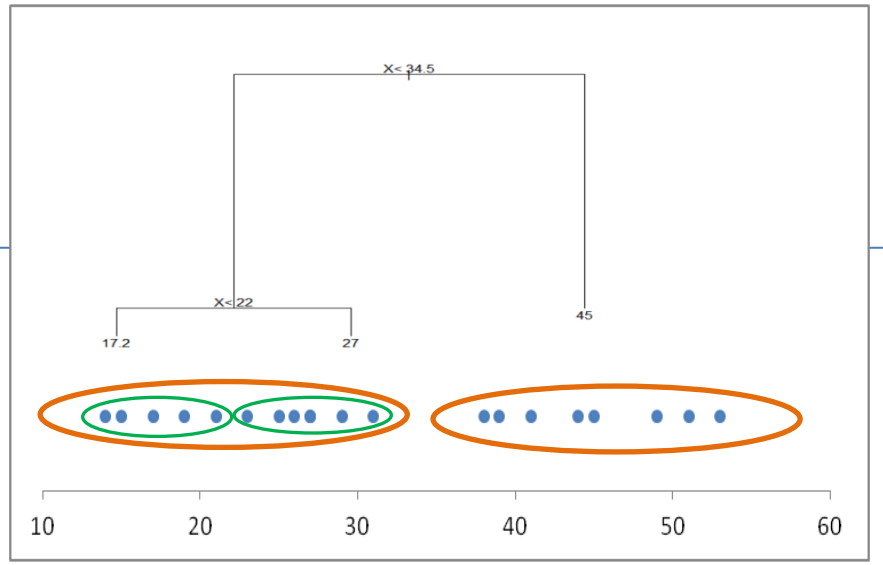
Les variables cibles et explicatives sont les mêmes. Construire la subdivision qui maximise les dispersions interclasses. Un arbre de régression peut le faire sans problème.

```
> library(rpart)
> Y <- donnees$X
> arbre <- rpart(Y ~ X, data = donnees, method = "anova", control = rpart.control(minsplit=5,cp=0.07))
> print(arbre)
n= 20

node), split, n, deviance, yval
      * denotes terminal node

1) root 20 2913.7500 31.75000
  2) X< 34.5 12  354.9167 22.91667
    4) X< 22 5   32.8000 17.20000 *
    5) X>=22 7   42.0000 27.00000 *
  3) X>=34.5 8  218.0000 45.00000 *
> plot(arbre)
> text(arbre)
```

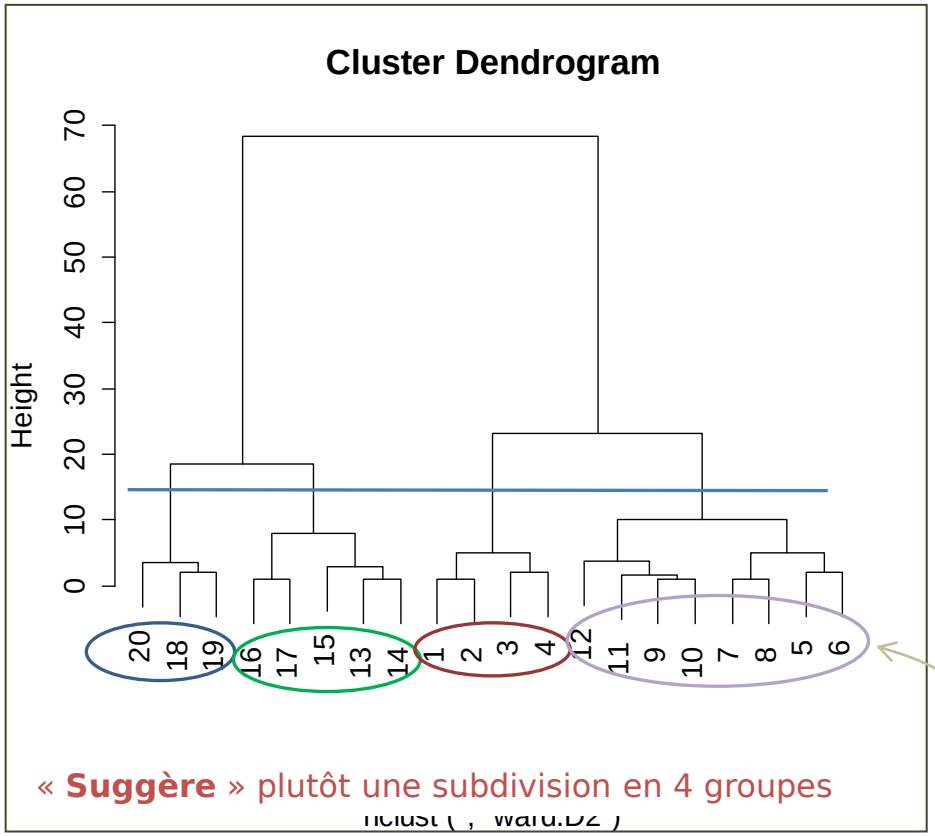
Attention : forte sensibilité au paramétrage !!!



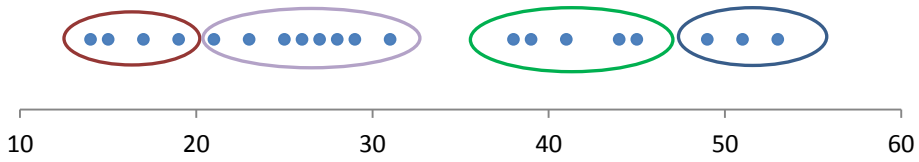
Remarque : Avec le mécanisme plotcp() et prunecp() de rpart(), on se faire une idée du « bon » nombre de classes.

Algorithme ascendant – Utiliser la CAH (Classification ascendante hiérarchique)

```
#matrice des distances
d <- dist(donnees)
#CAH, méthode de WARD
dendro <- hclust(d,method="ward.D2")
#affichage
plot(dendro)
```



Regroupements suggérés par la CAH



Bilan des méthodes tenant compte de la dispersion

+ Exploite une information plus « riche » des données
+ Produit des classes « compactes »
+ La qualité de la dispersion peut être quantifiée (rapport de corrélation)

- Le nombre de classes est « suggéré », de manière plus ou moins intuitive

* L'algorithme descendant est nettement plus rapide que l'ascendant sur de très grandes bases de données (en nombre d'observations)
* Il est plus facilement industrialisable sur un grand nombre de variables (pour peu que l'on puisse définir un paramètre qui soit adapté à toutes les variables... hum...)

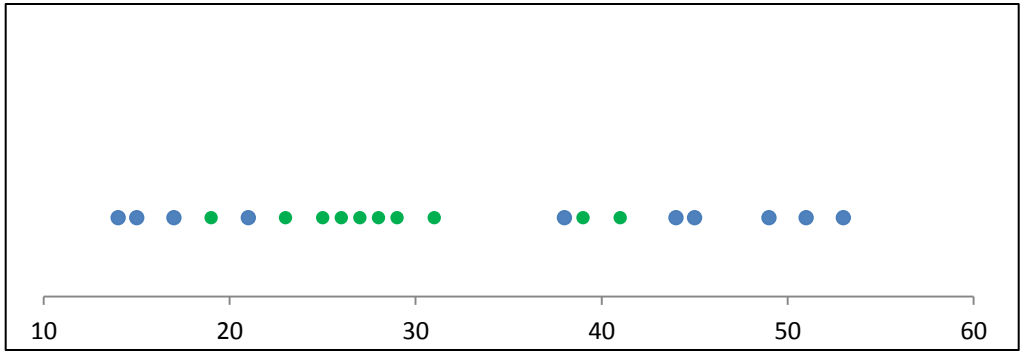
Discrétisation supervisée

Dans un schéma prédictif, une variable cible Y peut guider le découpage en classes de X

Cas d'une variable cible **qualitative**

Les observations appartiennent
à des groupes prédéfinis (par Y)

X	Y
14	Y1
15	Y1
17	Y1
21	Y1
38	Y1
44	Y1
45	Y1
49	Y1
51	Y1
53	Y1
19	Y2
23	Y2
25	Y2
26	Y2
27	Y2
28	Y2
29	Y2
31	Y2
39	Y2
41	Y2



Comment découper X de manière à isoler le mieux possible les points de même couleur (groupe)

➔ Combien d'intervalles ? (**pas trop**)

➔ Quelles bornes de découpage ?

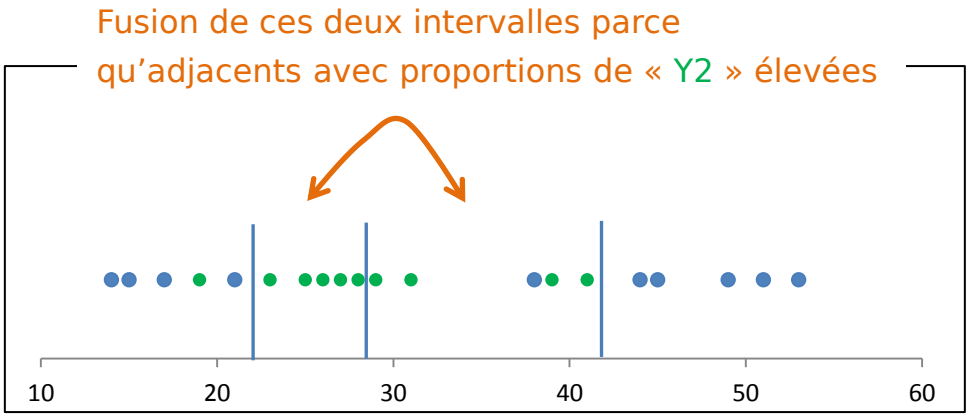
Remarque : les individus du même groupe ne sont pas forcément adjacents

Approche intuitive ascendante, très utilisée et pourtant…

- Pré-découpage en quantiles (ex. déciles, quartiles, …)
- Regroupement itératif des intervalles adjacents de profils proches c.-à-d. dont les fréquences de Y sont proches
- Arrêt lorsque les regroupements ne sont plus pertinents

X	Y
14	Y1
15	Y1
17	Y1
21	Y1
38	Y1
44	Y1
45	Y1
49	Y1
51	Y1
53	Y1
19	Y2
23	Y2
25	Y2
26	Y2
27	Y2
28	Y2
29	Y2
31	Y2
39	Y2
41	Y2

1er quartile	22.50
Mediane	28.50
3e quartile	41.75



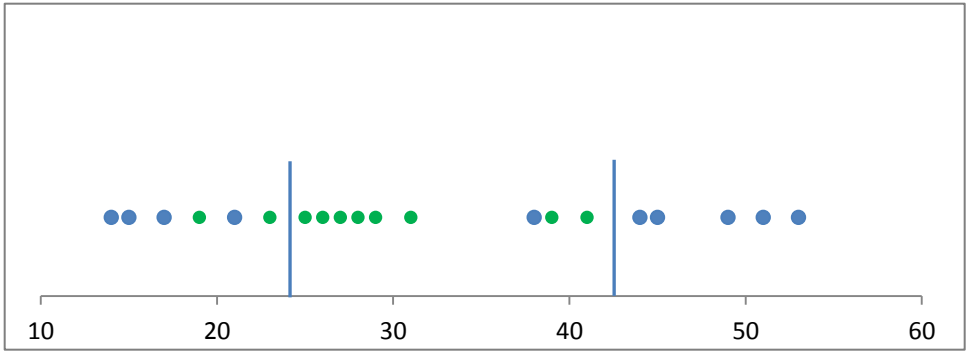
Problèmes potentiels :

- Approche manuelle, impraticable pour le traitement d'une grande base (nombre de variables)
- Le pré-découpage en quantiles peut être malheureux (sauf à descendre sur une granularité plus faible)
- Le « flair » c'est bien, les calculs c'est mieux (pour les fusions)

Approche ascendante – Chi-Merge de Kerber (1992)

- Pré-découpage en autant d'intervalles qu'il y a de points
- Regroupement itératif des intervalles adjacents de profils proches en utilisant un test du KHI-2 d'équivalence distributionnelle (les plus proches sont fusionnés en premier)
- Arrêt lorsque tous les intervalles adjacents sont de profils significativement différents
- Paramétrer par α, pour les tests de significativité

```
> print(summary(don.sup))
       X          Y
 Min.   :14.00   Y1:10
 1st Qu.:22.50   Y2:10
 Median :28.50
 Mean   :31.75
 3rd Qu.:41.75
 Max.   :53.00
>
> library(discretization)
> res.cm <- chiM(don.sup,alpha=0.05)
> print(res.cm$cutp)
[[1]]
[1] 22.0 42.5
```



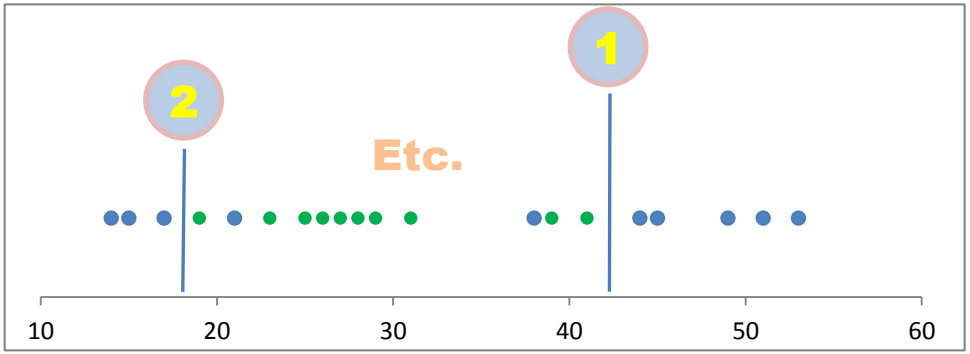
Bilan :

+ Scientifiquement cohérent, implémenté dans de nombreux logiciels

- Lenteur, surtout sur des grandes bases (nombre d'observations)
- Choix de α difficile, contexte de « comparaisons multiples »

Approche descendante (Top Down)

• Commencer par un découpage binaire de manière à séparer au mieux les classes
• Continuer de manière récursive dans chaque intervalle de manière à obtenir des intervalles « purs » (du point de vue des classes)
• S'arrêter quand le découpage n'est plus possible



Problèmes potentiels :

• Définir un indicateur de pureté – de gain de pureté – quand on subdivise un intervalle
• Définir une stratégie de découpage : après le 1er découpage, il faut traiter le sous-intervalle de gauche ou celui de droite ?
• Définir une règle d'arrêt : nombre maximal d'intervalles ? effectifs dans les intervalles ? pureté suffisante ? Gain significatif ? Etc.

➔ Ces problématiques ne nous rappellent rien ?

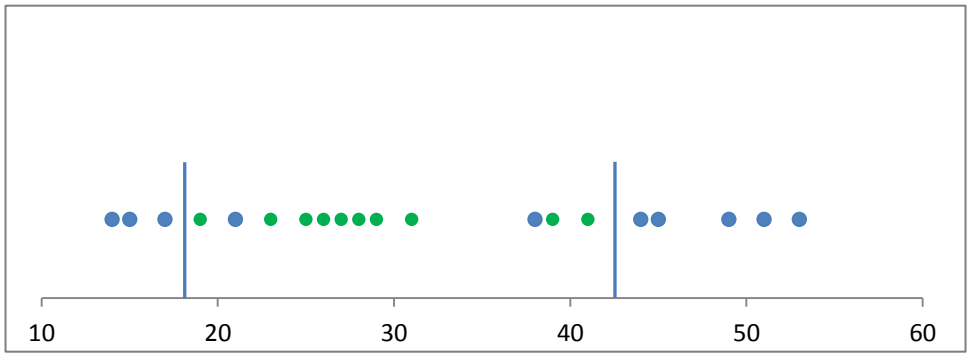
Approche descendante – Utilisation d'un arbre de décision

- S'appuyer sur un programme d'induction d'arbre de décision
- Y est la cible, X est la seule variable prédictive
- Utiliser les règles d'arrêt usuelles des algorithmes d'arbre

```
> arbre.2 <- rpart(Y ~ X, data = don.sup, method = "class", control = rpart.control(minsplit=5))
> print(arbre.2)
n= 20

node), split, n, loss, yval, (yprob)
      * denotes terminal node

1) root 20 10 Y1 (0.5000000 0.5000000)
  2) X>=42.5 5  0 Y1 (1.0000000 0.0000000) *
  3) X< 42.5 15  5 Y2 (0.3333333 0.6666667)
    6) X< 18 3  0 Y1 (1.0000000 0.0000000) *
    7) X>=18 12  2 Y2 (0.1666667 0.8333333) *
```

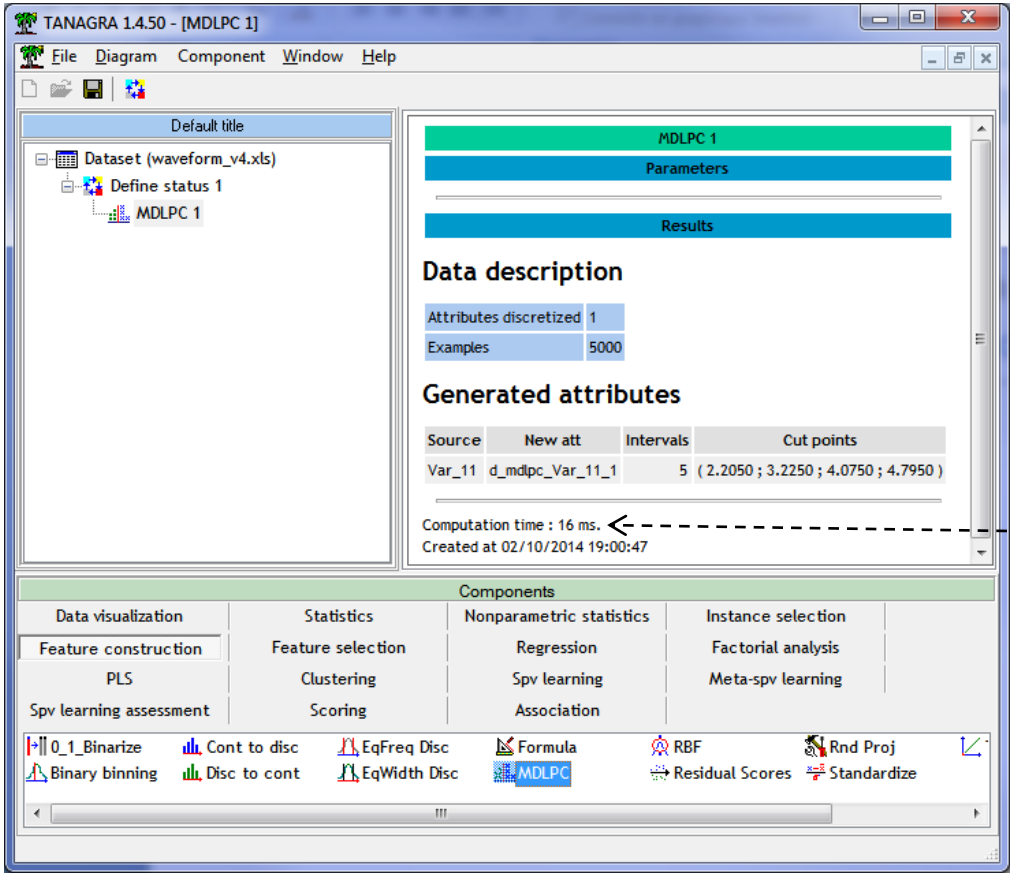


Bilan :

+ Scientifiquement cohérent, implémenté dans de nombreux logiciels
+ Rapidité sur les grandes bases (nombre d'observations)

- Paramétrage de l'arbre pas toujours évident

Approche descendante – Algorithme MDLPC de Fayyad & Irani (1993)



- Algorithme descendant d'induction d'arbre de décision
- Avec une règle d'arrêt optimisée pour la discrétisation de variables
- **Méthode de référence**

16 ms. pour le découpage en 5 intervalles d'une variable à 5000 observations.

Bilan :

+ Scientifiquement cohérent, implémenté dans de nombreux logiciels
+ Rapidité sur les grandes bases (nombre d'observations)
+ Pas de paramétrage

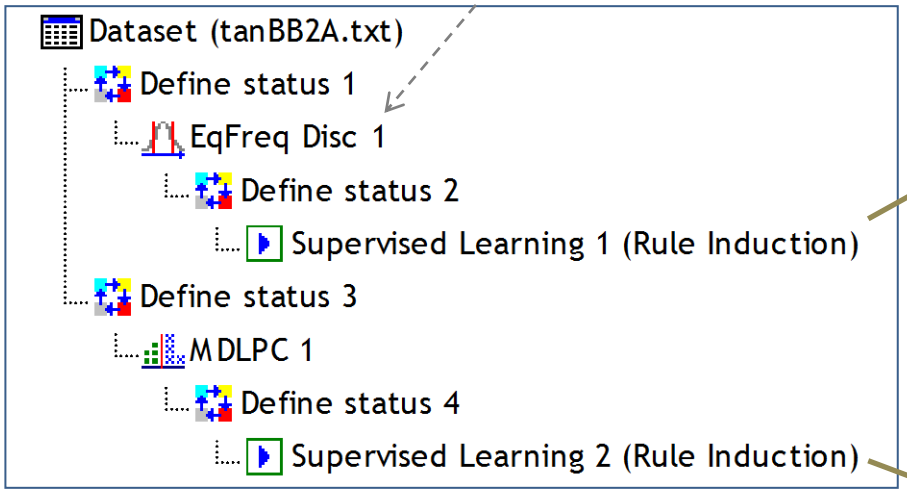
- Pas de paramétrage c.-à-d. pas possible d'adapter aux données

Sur notre fichier exemple, où les effectifs sont très faibles (10 individus par groupe), MDLPC annonce qu'il n'y a pas de découpage pertinent possible… (que faut-il en penser ?)

Pourquoi une discrétisation supervisée pour un problème supervisé ?

Fichier IRIS
150 obs.
4 variables prédictives
3 groupes

2 intervalles de fréquences égales (découpage avec la médiane). C'est vraiment chercher les ennuis dans un problème où Y possède 3 modalités.



Taux d'erreur = 21.33%

0.2133

Confusion matrix

	Iris-setosa	Iris-versicolor	Iris-virginica	Sum
Iris-setosa	48	2	0	50
Iris-versicolor	1	20	29	50
Iris-virginica	0	0	50	50
Sum	49	22	79	150

Taux d'erreur = 4.67%

0.0467

Confusion matrix

	Iris-setosa	Iris-versicolor	Iris-virginica	Sum
Iris-setosa	50	0	0	50
Iris-versicolor	0	44	6	50
Iris-virginica	0	1	49	50
Sum	50	45	55	150

Cas d'une variable cible **quantitative**

Les observations sont étiquetées par une variable Y quantitative

X	Y
14	**2**
15	**2.5**
17	**3.5**
19	**2.6**
21	**2.8**
23	**7.6**
25	**8.7**
26	**8.2**
27	**8.1**
28	**9.1**
29	**7.4**
31	**8.9**
38	**4.5**
39	**5.2**
41	**4.9**
44	**6.3**
45	**5.7**
49	**4.8**
51	**4.5**
53	**6.2**

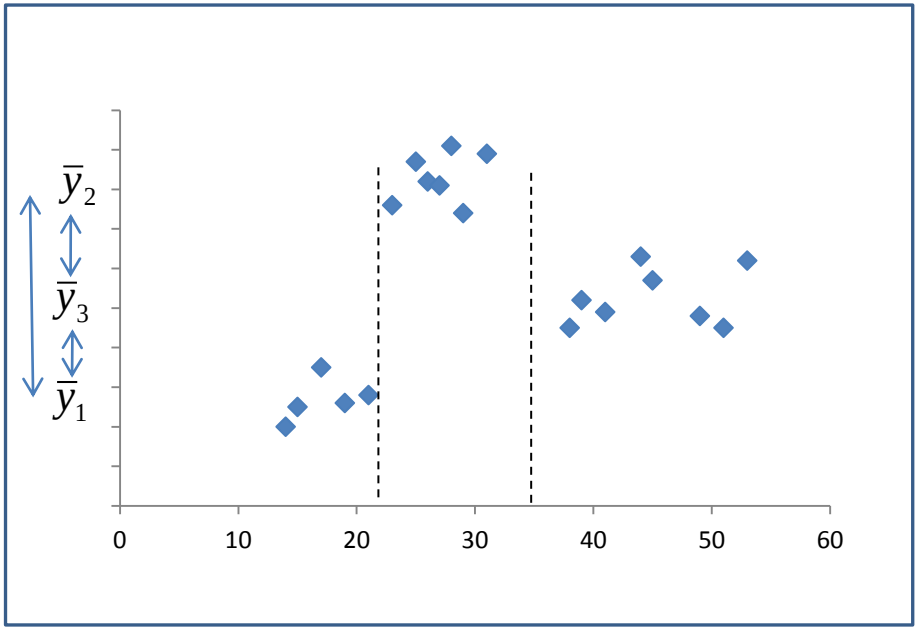


Schéma d'analyse de variance, mais relativement à la variable cible Y c.-à-d. découper X de manière à ce que Y soit le plus homogène possible (le moins dispersé possible) dans chaque sous-groupe !

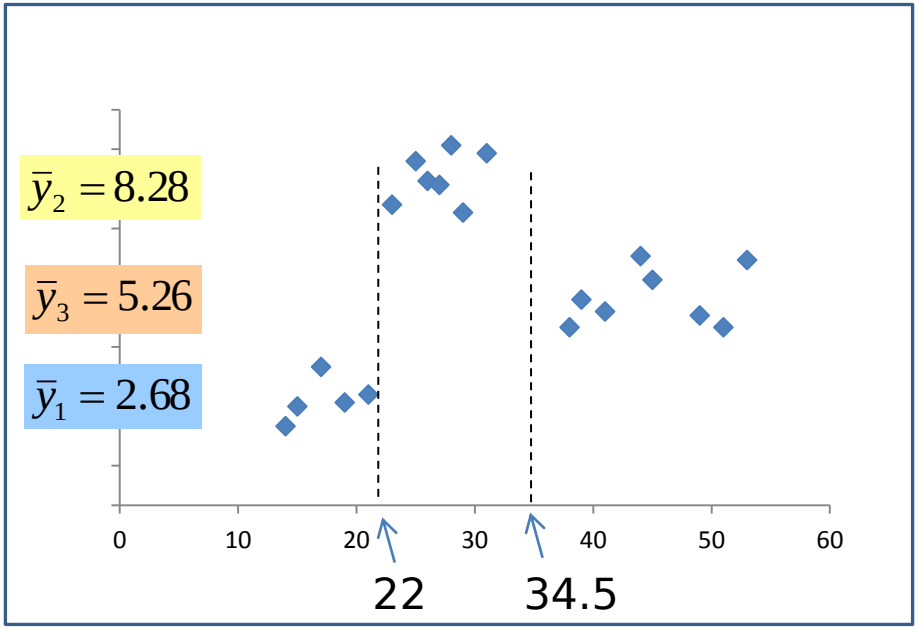
Algorithme descendant – Arbre de régression

```
> don.reg <- read.xlsx(file="data_discretisation.xlsx",sheetIndex=3)
> print(summary(don.reg))
       X                Y
 Min.   :14.00   Min.   :2.000
 1st Qu.:22.50   1st Qu.:4.250
 Median :28.50   Median :5.450
 Mean   :31.75   Mean   :5.675
 3rd Qu.:41.75   3rd Qu.:7.725
 Max.   :53.00   Max.   :9.100
>
> arbre.reg <- rpart(Y ~ X, data = don.reg, method = "anova", control = rpart.control(minsplit=5,cp=0.07))
> print(arbre.reg)
n= 20

node), split, n, deviance, yval
      * denotes terminal node

1) root 20 101.277500 5.675000
  2) X< 22 5   1.188000 2.680000 *
  3) X>=22 15  40.289330 6.673333
    6) X>=34.5 8   3.658750 5.262500 *
    7) X< 34.5 7   2.508571 8.285714 *
```

Arbre de régression = régression non linéaire



Bilan

Bilan

La discrétisation consiste à transformer une variable quantitative en qualitative ordinale, en la découpant en classes (intervalles).

Deux questions clés se posent : combien d'intervalles, comment déterminer les bornes (seuils) de découpage.

La méthode experte est certainement la meilleure, mais les connaissances nécessaires ne sont pas toujours disponibles.

Les méthodes guidées par les données se différencient par le contexte dans lequel elles se situent (supervisé ou non supervisé) ; des informations qu'elles exploitent ; et de la stratégie exploratoire utilisée (ascendante vs. descendante généralement).

La discrétisation fait partie du processus d'apprentissage, de sa qualité dépend la qualité du modèle élaboré à partir des données transformées.

Bibliographie

Tutoriel Tanagra, « Discrétisation – Comparaison de logiciels », 2010 ; http://tutoriels-data-mining.blogspot.fr/2010/02/discretisation-comparaison-de-logiciels.html

Tutoriel Tanagra, « Discrétisation contextuelle – La méthode MDLPC », 2008 ; http://tutoriels-data-mining.blogspot.fr/2008/03/discrtisation-contextuelle-la-mthode.html

Gilles Hunault, « Découpage en classes et discrétisation » ; http://www.info.univ-angers.fr/~gh/wstat/discr.php