

Classification Ascendante Hiérarchique

Angelina Roche

Executive Master Statistique et Big Data

Plan du chapitre

Classification ascendante hiérarchique

Application

Pour aller plus loin : méthode des k -moyennes et classification mixte

Classification non supervisée ou *clustering*

Objectif

Effectuer un regroupement en k ($k \ll n$) groupes de manière à rassembler dans chaque groupe les individus “les plus semblables” selon un critère à définir (en général assimilé à une distance).

Exemples d'application

Marketing : création de profils clients permettant de cibler les offres promotionnelles suivant certains critères.

Médias : création de profils utilisateurs permettant de personnaliser une page web.

Economie : création de profils pays suivant différents critères tels que

- ▶ les échanges économiques,
- ▶ le taux d'armement,
- ▶ le niveau d'éducation,
- ▶ ...

...

Classification automatique : objectifs et méthodes

- ▶ **Méthode générale** : Classification des n individus en k classes telles que :
 - ▶ l'homogénéité soit maximale à l'intérieur de chaque classe,
 - ▶ l'hétérogénéité soit maximale d'une classe à l'autre.
- ▶ **Remarque** : les classes et le nombre k de classes sont inconnus.

Classification = partition ?

- ▶ On construit des classes **disjointes**.
- ▶ Si tout individu est classé, on aboutit à la notion de **partition** : tout individu appartient à une classe et une seule.
- ▶ Les éléments d'une même classe sont équivalents et donc indiscernables. Il suffit alors d'utiliser un **représentant** pour chaque classe (par exemple le point moyen) dans la suite des traitements.

Qualité d'une partition : inertie inter et inertie intra

- ▶ n individus dans k groupes l_1, \dots, l_k ,
- ▶ \bar{x} : isobarycentre des n individus, $\bar{x} = \frac{1}{n} \sum_{i=1}^n x_i$,
- ▶ n_j : effectif du groupe l_j ,
- ▶ C_j : isobarycentre du groupe l_j , $C_j = \frac{1}{n_j} \sum_{i \in l_j} x_i$.
- ▶ **Inertie intra-classe** : Somme des inerties des points du groupe l_j au barycentre C_j .

$$\mathcal{I}_{intra} = \mathcal{I}_{intra}(l_1, \dots, l_k) = \frac{1}{n} \sum_{j=1}^k \sum_{i \in l_j} \|x_i - C_j\|^2.$$

- ▶ **Inertie inter-classe** : Inertie des barycentres C_j au barycentre \bar{x} .

$$\mathcal{I}_{inter} = \mathcal{I}_{inter}(l_1, \dots, l_k) = \frac{1}{n} \sum_{j=1}^k n_j \|C_j - \bar{x}\|^2.$$

Qualité d'une partition : inertie totale

- ▶ **Inertie totale** : Inertie des n points au barycentre \bar{x} .

$$\mathcal{I}_G = \frac{1}{n} \sum_{i=1}^n \|x_i - \bar{x}\|^2.$$

- ▶ **Relation de Huygens** : L'inertie totale est la somme des inerties intra et inter classes.

$$\mathcal{I}_G = \mathcal{I}_{intra} + \mathcal{I}_{inter}.$$

Qualité d'une partition : mesure

- ▶ Qualité d'une partition :

$$\frac{\mathcal{I}_{inter}}{\mathcal{I}_G}.$$

- ▶ Dépend du nombre d'individus et du nombre de classes.
- ▶ 2 cas extrêmes :
 - ▶ $k = n$: $\mathcal{I}_{intra}(\{1\}, \dots, \{n\}) = 0$ et $\mathcal{I}_{inter}(\{1\}, \dots, \{n\}) = \mathcal{I}_G$.
 - ▶ $k = 1$: $\mathcal{I}_{intra}(\{1, \dots, n\}) = \mathcal{I}_G$ et $\mathcal{I}_{inter}(\{1, \dots, n\}) = 0$.

Qualité d'une partition ¹

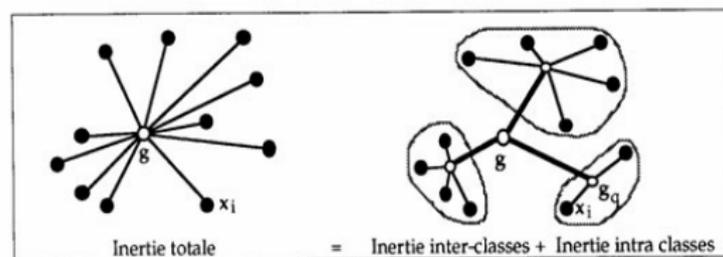


Figure 2.2 - 9
Décomposition de l'inertie selon la relation de Huygens

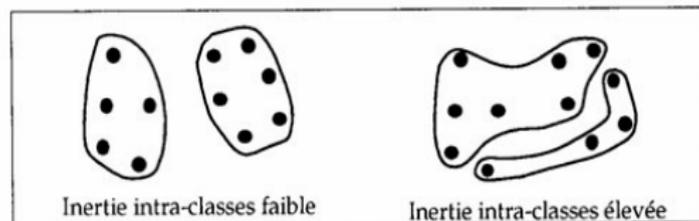
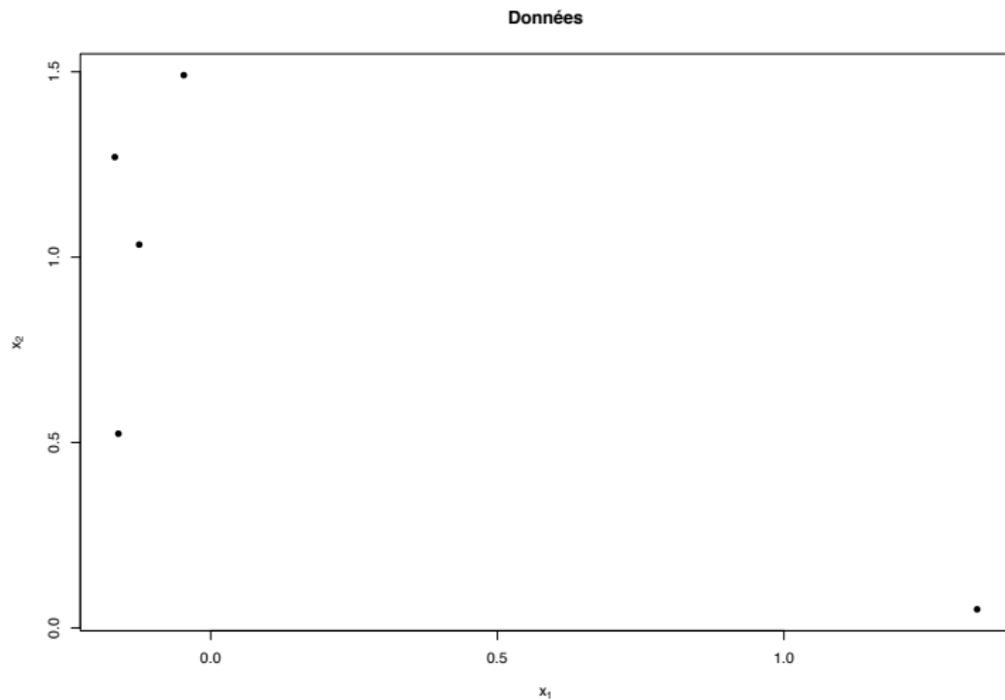
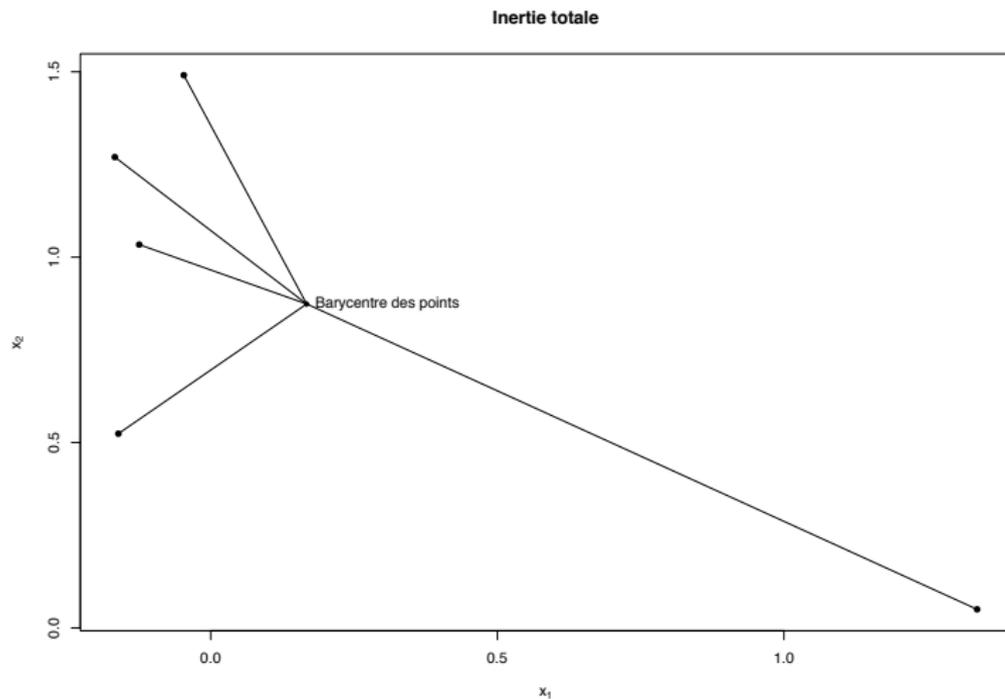


Figure 2.2 - 10
Qualité globale d'une partition

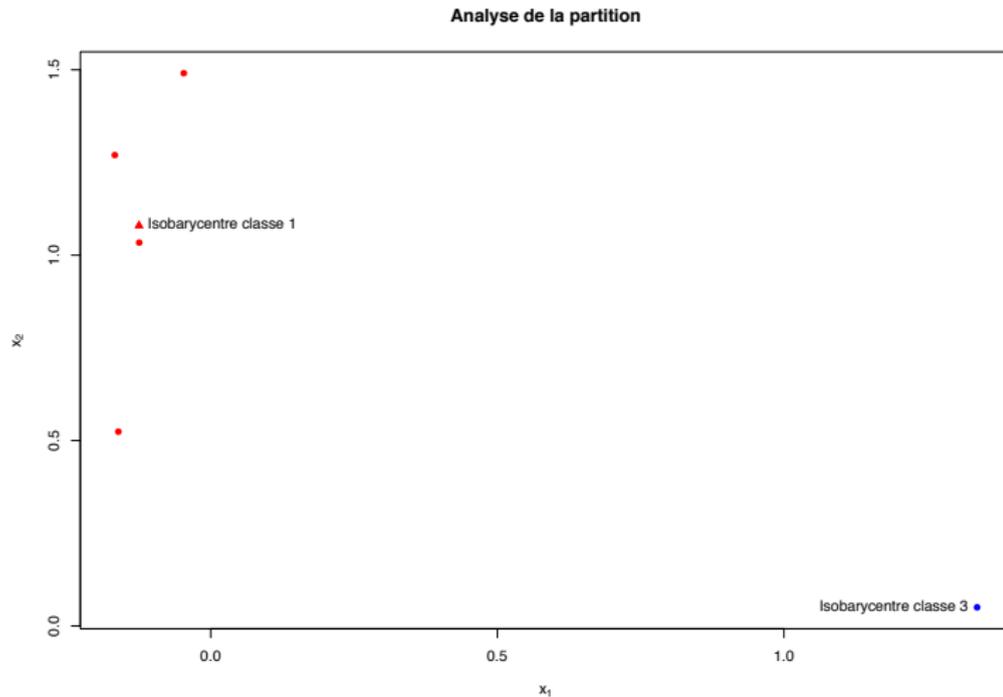
1. Illustrations du livre de Lebart, Morineau et Piron, *Statistique exploratoire multidimensionnelle*

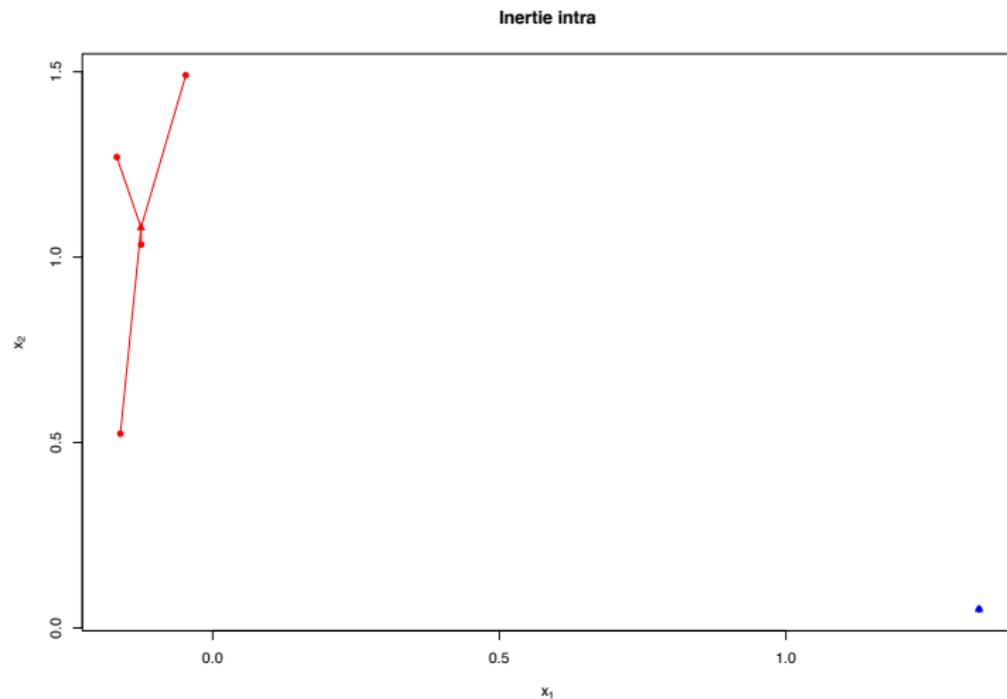
Exemple $n = 5$

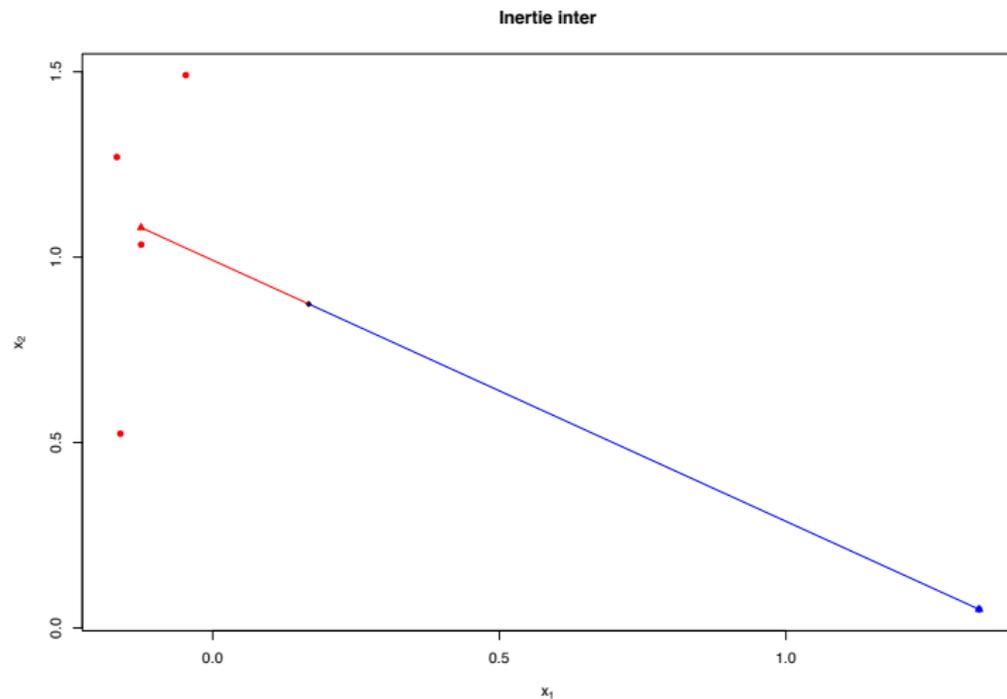


Exemple $n = 5$ 

Exemple $n = 5$



Exemple $n = 5$ 

Exemple $n = 5$ 

Partitions optimales

Idée naïve : Rechercher la (ou une partition) maximisant l'inertie inter-classes \mathcal{I}_{inter} .

Problème : examen de toutes les partitions possibles en k classes d'un ensemble à n éléments.

Nombre de partitions possible (nombre de Stirling)

$$S(n, k) = \frac{1}{k!} \sum_{j=0}^k (-1)^{k-j} \binom{k}{j} j^n \quad (\text{résultat admis}).$$

Cas $k = 2$:

$$S(n, 2) = 2^{n-1} - 1.$$

n	2	3	4	5	200	300	400	500
$S(n, 2)$	1	3	7	15	$8,03 \cdot 10^{59}$	$1,02 \cdot 10^{90}$	$1,29 \cdot 10^{120}$	$1,64 \cdot 10^{150}$

Il faudrait donc au moins 10^{42} ans à un ordinateur de bureau pour calculer toutes les partitions à 2 éléments d'un ensemble à 200 éléments.

Plan

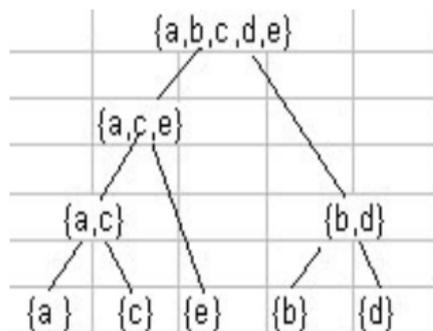
Classification ascendante hiérarchique

Application

Pour aller plus loin : méthode des k -moyennes et classification mixte

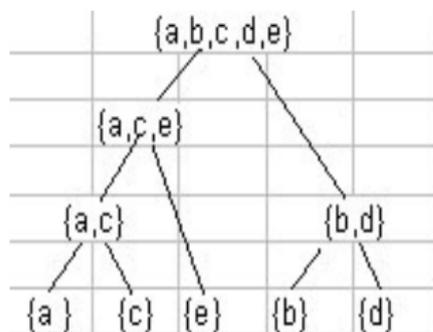
Hiérarchie de Parties

- ▶ Une *hiérarchie* de parties est un ensemble de parties “emboîtées”.
- ▶ Cette représentation traduit la façon dont elles sont construites :
 - ▶ soit par réunion successive de parties (*ascendante*),
 - ▶ soit par division successive (*descendante*).
- ▶ La relation d'inclusion conduit à une représentation graphique du type :



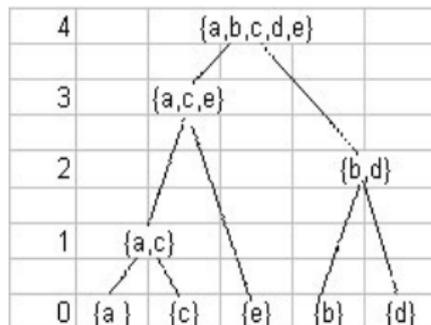
Représentation graphique

- ▶ L'ordre de formation traduit les différents niveaux de ressemblance entre les parties. On utilise donc cet ordre pour ordonner le graphique suivant l'axe vertical :



Représentation graphique (2)

- ▶ On peut aussi utiliser une quantification de la variabilité entre les parties, appelée souvent *indice de diamètre* :



- ▶ \implies *hiérarchie indicée*.
- ▶ Représentation graphique = *dendrogramme*.

Algorithme d'agrégation par critère de Ward

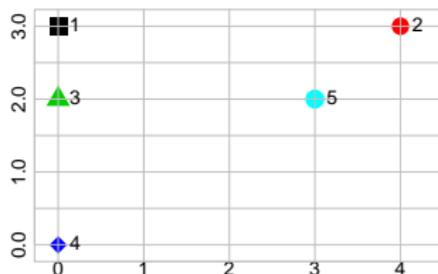
- ▶ **Principe général** : Au départ de l'algorithme, l'inertie inter classes est maximale (n classes). A la fin, celle-ci est nulle (1 classe).
⇒ On cherche à minimiser à chaque étape la perte d'inertie inter-classes (ou à minimiser le gain d'inertie intra-classe).
- ▶ **A chaque étape** : On regroupe les 2 classes pour lesquelles la perte d'inertie inter-classes est minimale. Cela revient à regrouper les classes j et j' pour lesquelles la perte

$$\Delta_{jj'} = \frac{n_j n_{j'}}{n_j + n_{j'}} \|C_j - C_{j'}\|^2$$

est minimale.

Un exemple simple

- ▶ Cinq points dans un plan = 5 classes au départ de l'algorithme



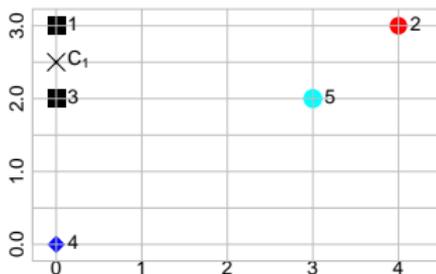
$$I_{intra} = 0$$

- ▶ Perte d'inertie inter-classes ($\times n$) = $\Delta_{j,j'}$:

Classes	{1}	{2}	{3}	{4}	{5}
{1}	0	8	0.5	4.5	5
{2}		0	8.5	12.5	1
{3}			0	2	4.5
{4}				0	6.5

- ▶ Regroupement 1 et 3 \Rightarrow nouvelle classe {1, 3}.

Un exemple simple (2)



$$l_{intra} = n \times 0.5 = 2.5$$

- Perte d'inertie inter-classes ($\times n$) = $\Delta_{j,j'}$ (arrondi à 0.1) :

Classes	{1, 3}	{2}	{4}	{5}
n° de classe (j)	1	2	3	4
{1,3}	0	10.8	4.2	6.3
{2}		0	12.5	1
{4}			0	6.5

- Regroupement 2 et 5 \Rightarrow nouvelle classe {2, 5}.

Un exemple simple (3)



$$I_{intra} = 2.5 + 5 \times 1 = 7.5$$

- Perte d'inertie inter-classes ($\times n$) = $\Delta_{j,j'}$ (arrondi à 0.1) :

Classes	{1,3}	{2,5}	{4}
n° de classe (j)	1	2	3
{1,3}	0	12.3	4.2
{2,5}		0	21.7

- Regroupement {1, 3} et 4 \Rightarrow nouvelle classe {1, 3, 4}.

Un exemple simple (4)



$$I_{intra} = 7.5 + 5 \times 4.2 = 28.5$$

► Dendrogramme



Choix du nombre de classes

- ▶ *Coupure* de l'arbre à un niveau donné de l'indice \implies *partition*.
- ▶ La coupure doit se faire :
 - ▶ **après** les agrégations correspondant à des valeurs **peu élevées** de l'indice,
 - ▶ **avant** les agrégations correspondant à des niveaux **élevés** de l'indice, qui dissocient les groupes bien distincts dans la population.
- ▶ *Règle empirique* : sélection d'une coupure lors d'un saut important de l'indice par *inspection visuelle* de l'arbre.
- ▶ Ce saut traduit le passage brutal entre des classes d'une certaine homogénéité de l'ensemble à des classes beaucoup moins homogènes.
- ▶ Dans la plupart des cas, il y a *plusieurs paliers* et donc *plusieurs choix de partitions* possibles.

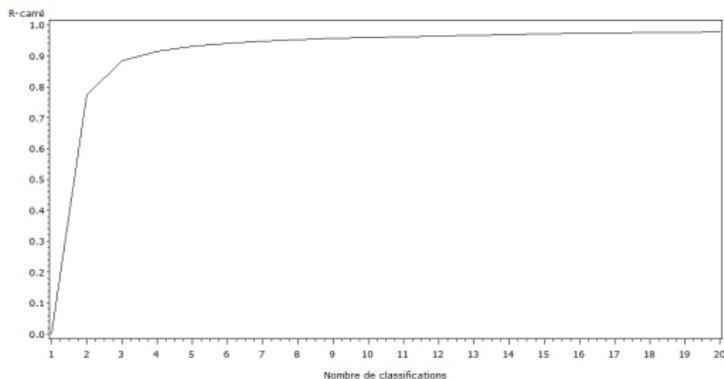
Choix du nombre de classes : R^2

Souvent, plusieurs choix de partitions possibles

⇒ utilisation d'un critère numérique :

$$R^2(I_1, \dots, I_k) = \frac{\mathcal{I}_{inter}(I_1, \dots, I_k)}{\mathcal{I}_G}$$

Repérage du point k où il y a rupture de pente dans le R^2 :



Plan

Classification ascendante hiérarchique

Application

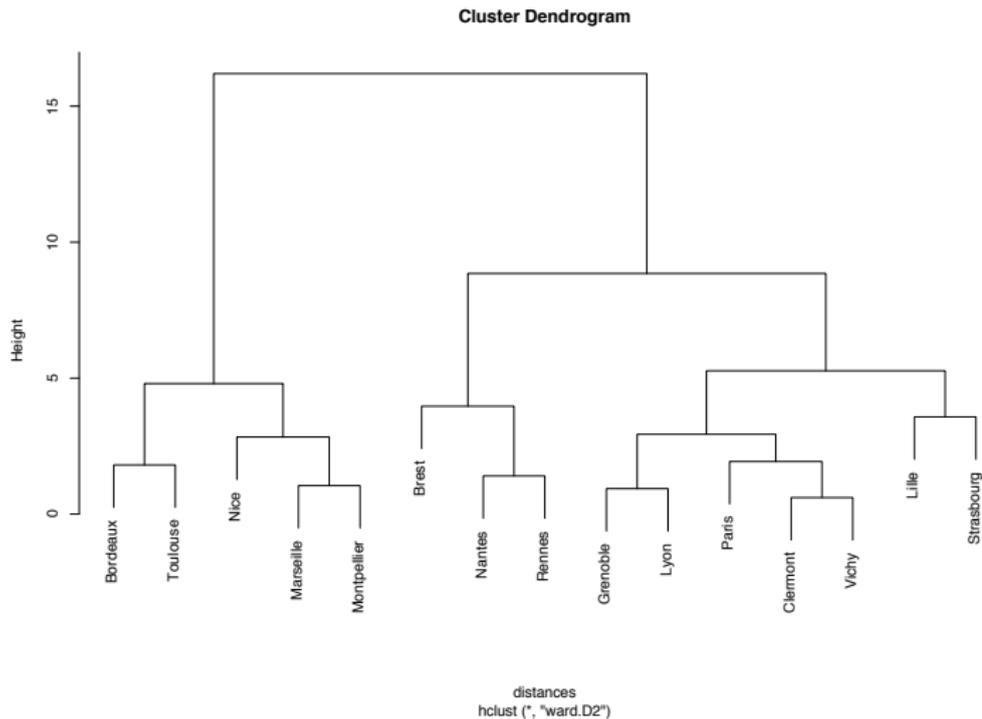
Pour aller plus loin : méthode des k -moyennes et classification mixte

Présentation des données

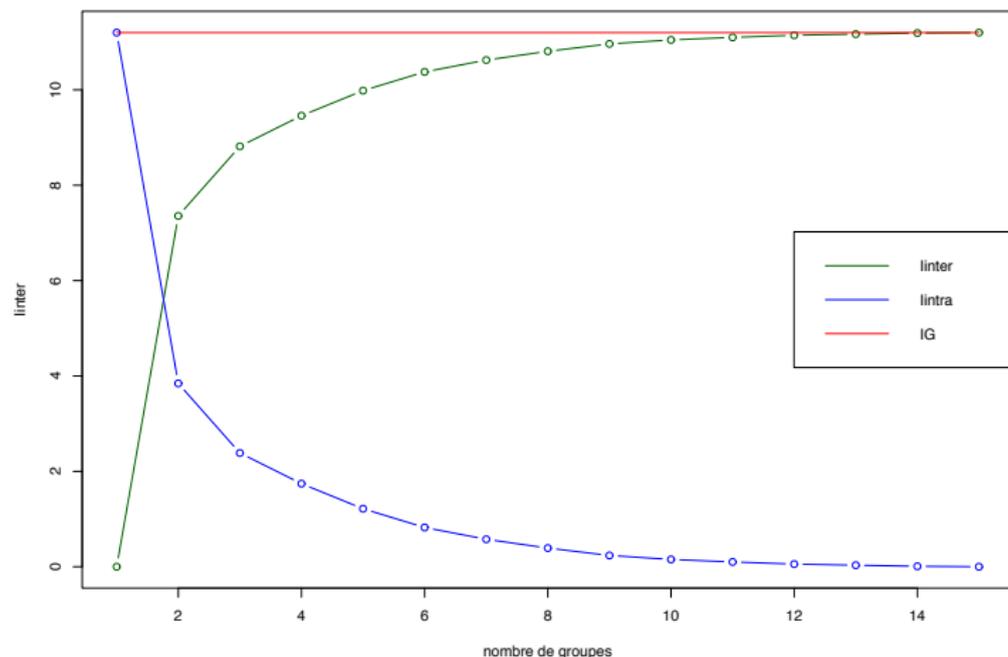
	Janv	Fevr	Mars	Avril	Mai	Juin	Juil	Aout	...
Bordeaux	5.60	6.60	10.30	12.80	15.80	19.30	20.90	21.00	...
Brest	6.10	5.80	7.80	9.20	11.60	14.40	15.60	16.00	...
Clermont	2.60	3.70	7.50	10.30	13.80	17.30	19.40	19.10	...
Grenoble	1.50	3.20	7.70	10.60	14.50	17.80	20.10	19.50	...
Lille	2.40	2.90	6.00	8.90	12.40	15.30	17.10	17.10	...
Lyon	2.10	3.30	7.70	10.90	14.90	18.50	20.70	20.10	...
Marseille	5.50	6.60	10.00	13.00	16.80	20.80	23.30	22.80	...
Montpellier	5.60	6.70	9.90	12.80	16.20	20.10	22.70	22.30	...
Nantes	5.00	5.30	8.40	10.80	13.90	17.20	18.80	18.60	...
Nice	7.50	8.50	10.80	13.30	16.70	20.10	22.70	22.50	...
Paris	3.40	4.10	7.60	10.70	14.30	17.50	19.10	18.70	...
Rennes	4.80	5.30	7.90	10.10	13.10	16.20	17.90	17.80	...
Strasbourg	0.40	1.50	5.60	9.80	14.00	17.20	19.00	18.30	...
Toulouse	4.70	5.60	9.20	11.60	14.90	18.70	20.90	20.90	...
Vichy	2.40	3.40	7.10	9.90	13.60	17.10	19.30	18.80	...

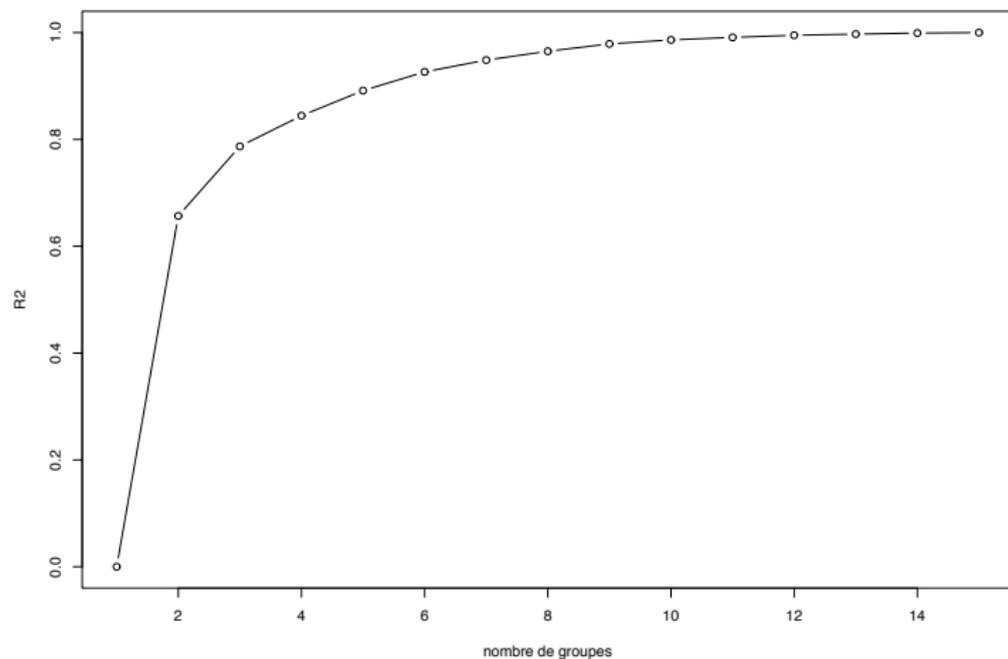
Températures mensuelles moyennes de 15 villes françaises.

Dendrogramme obtenu par la CAH

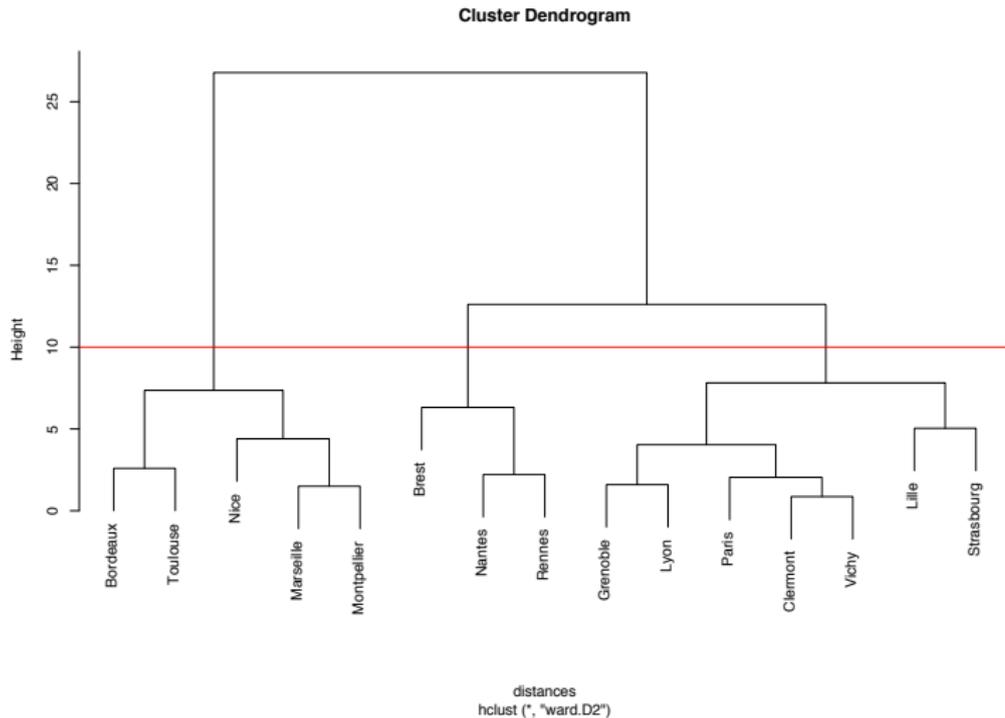


Evolution de l'inertie en fonction du nombre de groupes

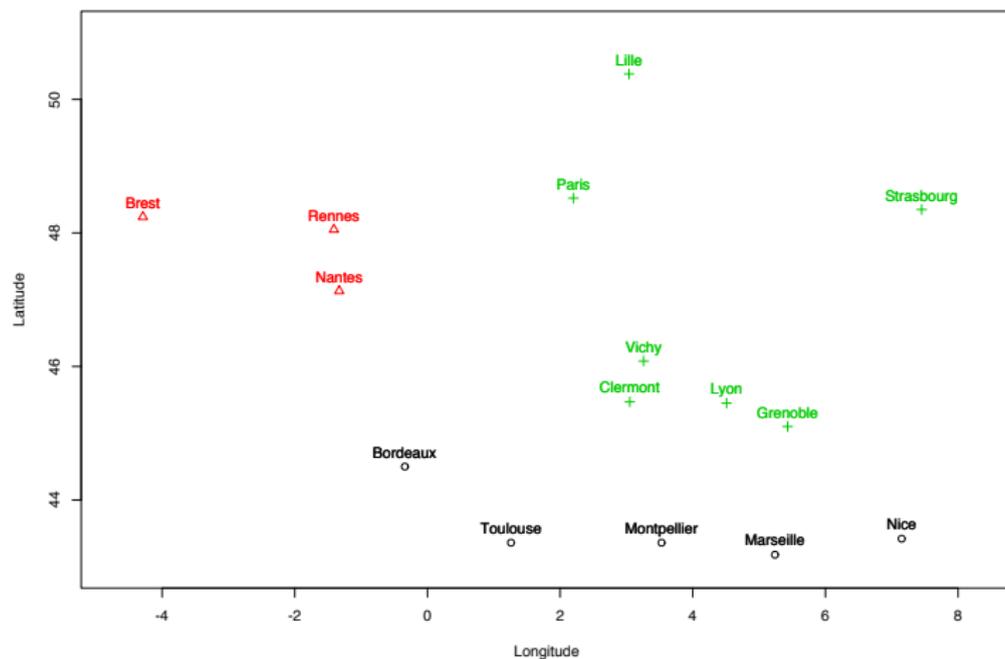


Evolution du critère $R^2(k)$ 

Dendrogramme



Représentation des clusters



Analyse de la classification obtenue

Détection des variables liées à la classification (I)

On travaille variable par variable.

Test du rapport de corrélation

Liaison entre **une** variable quantitative (ici la température moyenne mensuelle, la longitude,...) x et une variable qualitative (la classe).

Soit

$$\hat{\eta}^2 = \frac{\text{Variance inter-groupe}}{\text{Variance totale}} = \frac{\sum_{j=1}^k n_j (C_j - \bar{x})^2}{\sum_{i=1}^n (x_i - \bar{x})^2}$$

avec n_j effectif de la classe j , C_j moyenne de la variable dans la classe j .

On teste :

H_0 : les deux variables sont indépendantes contre H_1 : les deux variables sont dépendantes.

Sous H_0 , $K = (n - k)\hat{\eta}^2 / ((k - 1)(1 - \hat{\eta}^2)) \sim \text{Fisher}(k - 1, n - k)$.

Analyse de la classification obtenue

Détection des variables liées à la classification (II)

	Eta2	P-value
Moye	0.84	1.91e-05
Oct	0.84	1.93e-05
Sept	0.83	2.41e-05
Fevr	0.82	3.10e-05
Mars	0.81	4.33e-05
Janv	0.81	4.44e-05
Nov	0.81	4.96e-05
Avril	0.79	7.89e-05
Dec	0.79	9.32e-05
Aout	0.79	9.50e-05
Juin	0.72	4.41e-04
Mai	0.72	5.21e-04
Juil	0.72	5.29e-04
Ampl	0.65	1.95e-03
Lati	0.64	2.19e-03
Long	0.60	4.00e-03

Analyse de la classification obtenue

Moyenne des variables dans chaque classe

Classe 1 : Bordeaux, Marseille, Montpellier, Nice, Toulouse.

Classe 2 : Brest, Nantes, Rennes.

Classe 3 : Clermont, Grenoble, Lille, Lyon, Paris, Strasbourg, Vichy.

	Janv	Fevr	Mars	Avril	Mai	Juin	Juil	Aout	Sept	Oct	Nov	Dec
1	5.78	6.80	10.04	12.70	16.08	19.80	22.10	21.90	19.28	14.54	9.88	6.66
2	5.30	5.47	8.03	10.03	12.87	15.93	17.43	17.47	15.60	11.93	8.33	5.97
3	2.11	3.16	7.03	10.16	13.93	17.24	19.24	18.80	15.94	10.90	6.36	3.07

	Lati	Long	Moye	Ampl
1	43.56	3.37	13.79	16.34
2	47.81	-2.34	11.20	12.37
3	47.05	4.13	10.66	17.13

Plan

Classification ascendante hiérarchique

Application

Pour aller plus loin : méthode des k -moyennes et classification mixte

Références

- ▶ Algorithme dû principalement à Forgy (1965) : E. Forgy, *Cluster analysis of multivariate data : Efficiency versus interpretability of classification*, Biometrics, 21 :768 :780, 1965
- ▶ Prémisses ou variantes : Thorndike (1953), *k-means* MacQueen (1967), Ball & Hall (1967)
- ▶ Généralisation marquante : *Algorithme des nuées dynamiques* proposé par Diday (1971)
- ▶ Adapté aux **grands ensembles de données** :
 - ▶ traitement séquentiel possible
 - ▶ ne requiert pas le calcul d'une matrice $n \times n$ de distances entre individus

Algorithme

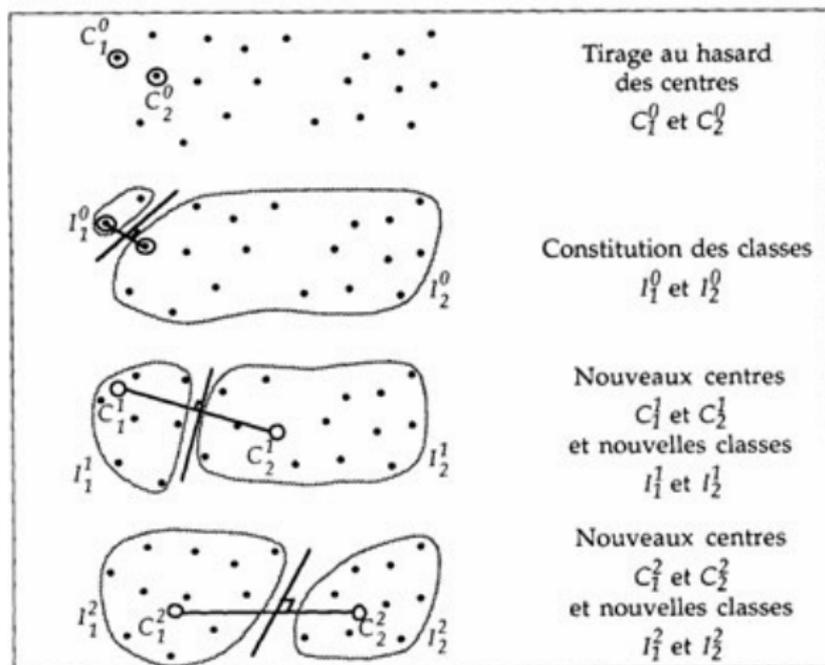


Figure 2.1 - 1
Etapes de l'algorithme

Description de l'algorithme

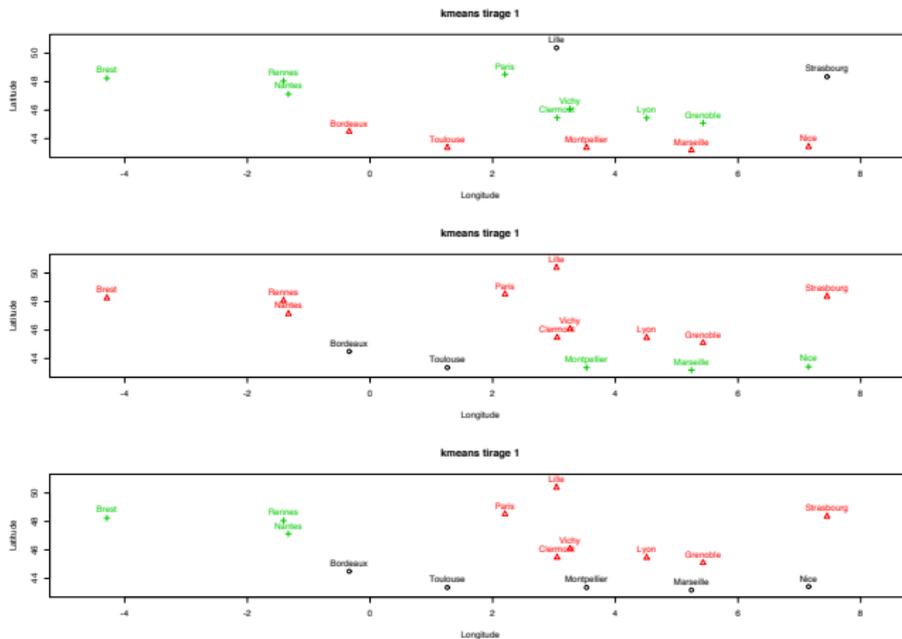
- ▶ On souhaite regrouper X_1, \dots, X_n en k classes (k choisi au préalable par l'utilisateur).
- ▶ **Etape ℓ** :
 - ▶ $\{C_1^{(\ell)}, \dots, C_k^{(\ell)}\}$ isobarycentres des k classes
 - ▶ $\{I_1^{(\ell-1)}, \dots, I_k^{(\ell-1)}\}$ construites à l'étape $\ell - 1$ (à l'étape 1 les centres sont tirés aléatoirement sans remise parmi $\{X_1, \dots, X_n\}$).
 - ▶ k nouvelles classes $\{I_1^{(\ell)}, \dots, I_k^{(\ell)}\}$ créées en regroupant les données les plus proches de chaque centre.
- ▶ **Fin de l'algorithme** : l'algorithme converge vers une partition stable. Arrêt lorsque la partition reste la même, ou lorsque la variance intra-classes ne décroît plus, ou encore lorsque le nombre maximal d'itérations est atteint.

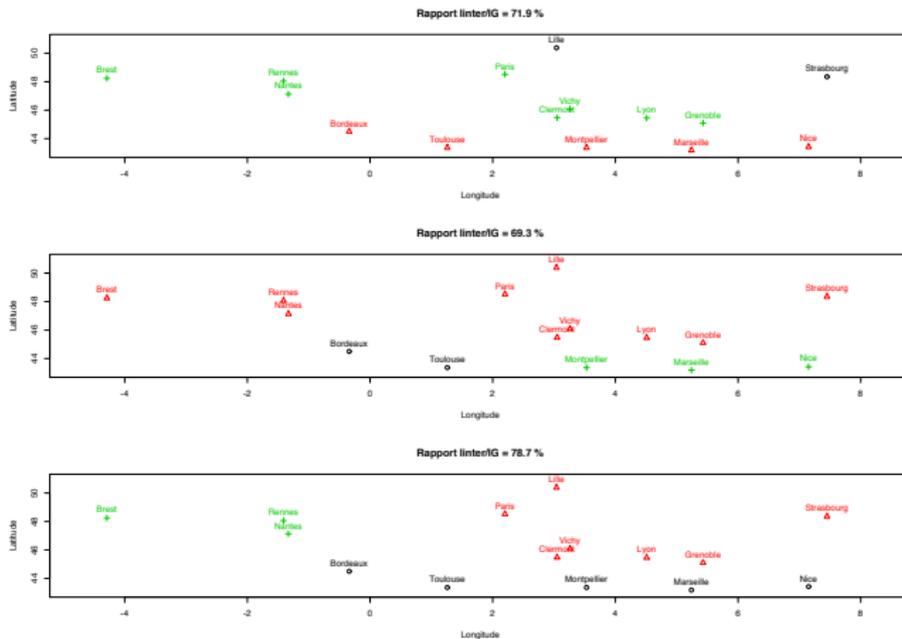
Décroissance de l'inertie intra-classe

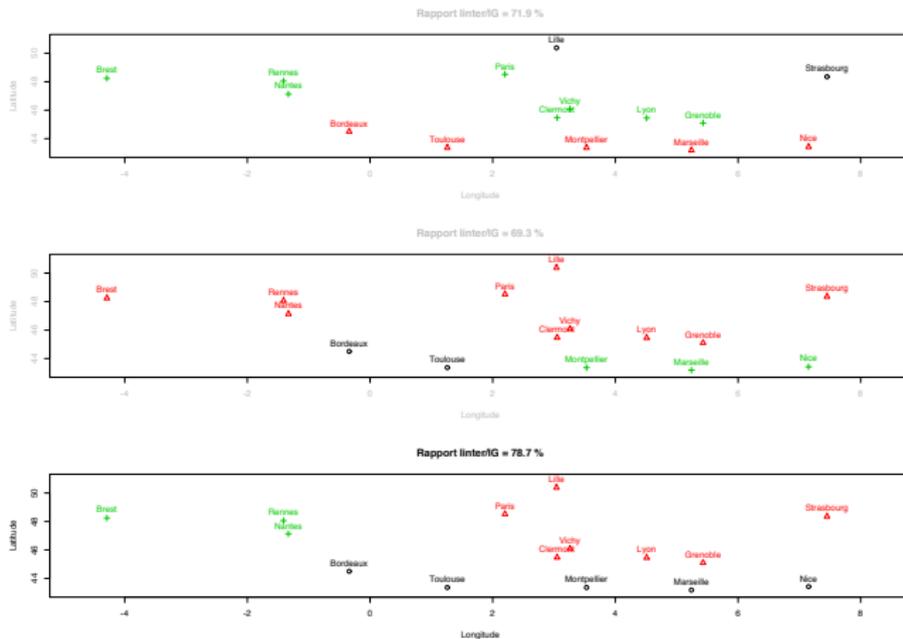
Théorème

L'inertie intra-classe décroît ou reste stable à chaque itération de l'algorithme des k -moyennes :

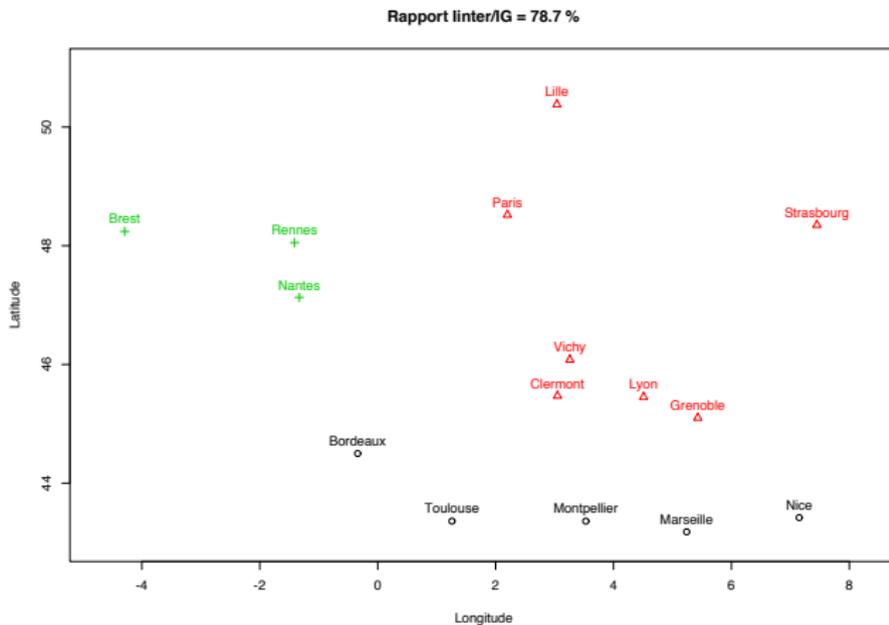
$$\mathcal{I}_{intra}(I_1^{(\ell)}, \dots, I_k^{(\ell)}) \geq \mathcal{I}_{intra}(I_1^{(\ell+1)}, \dots, I_k^{(\ell+1)}).$$

Classes obtenues par la méthode des k -moyennes avec $k = 3$ 

Classes obtenues par la méthode des k -moyennes avec $k = 3$ 

Classes obtenues par la méthode des k -moyennes avec $k = 3$ 

Partition finale



Atouts et limites des méthodes

▶ **Atouts et limites de la CAH**

▶ **Atout :**

- ▶ Fournit à la fois les classes et leur nombre

▶ **Limites :**

- ▶ Souvent malaisé de choisir la coupure significative sur le dendrogramme
- ▶ Partition non-optimale en raison de sa structure hiérarchique
- ▶ Fort coût algorithmique lorsque n devient grand

▶ **Atouts et limites des centres mobiles**

▶ **Atouts :**

- ▶ Coût algorithmique faible
- ▶ Traitement séquentiel

▶ **Limites :**

- ▶ Nombre de classes fixé a priori
- ▶ Partition obtenue fortement dépendante des centres provisoires des classes

▶ **Idée :** Mixer les 2 méthodes (CAH et centres mobiles)

Classification mixte

- ▶ **Etape 1 : Partitionnement préliminaire (si n grand)**
Partitionnement en q classes, avec $n \gg q \gg k$ le nombre de classes final désiré, en utilisant la méthode des centres mobiles ($q \simeq 10$ ou 100)
- ▶ **Etape 2 : Classification ascendante hiérarchique**
CAH sur les q éléments (centres) obtenus à l'étape 1
- ▶ **Etape 3 : Optimisation**
 - ▶ Partition finale obtenue par coupure de l'arbre de la CAH
 - ▶ Homogénéité des classes optimisée par réaffectation par la technique des centres mobiles (consolidation)
- ▶ **Remarque** : Méthode qui peut être instable sur les échantillons de petite taille.

Quelques fonctions R

- ▶ Pour la CAH : fonction `hclust()` (voir `help(hclust)`).
- ▶ Pour les k -moyennes : fonction `kmeans()`.
- ▶ Pour l'analyse de la classification : fonction `catdes()` du package *FactoMineR* (voir <https://www.youtube.com/watch?v=N7GkrUYP1LM>).