# Analyse de Données – ID
# Classification Hiérarchique Ascendante

Philippe LERAY
philippe.leray@univ-nantes.fr

Equipe COnnaissances et Décision
Laboratoire d'Informatique de Nantes Atlantique – FRE 2729
Site de l'Ecole Polytechnique de l'université de Nantes

Polytech'Nantes
École d'ingénieurs de l'université de Nantes

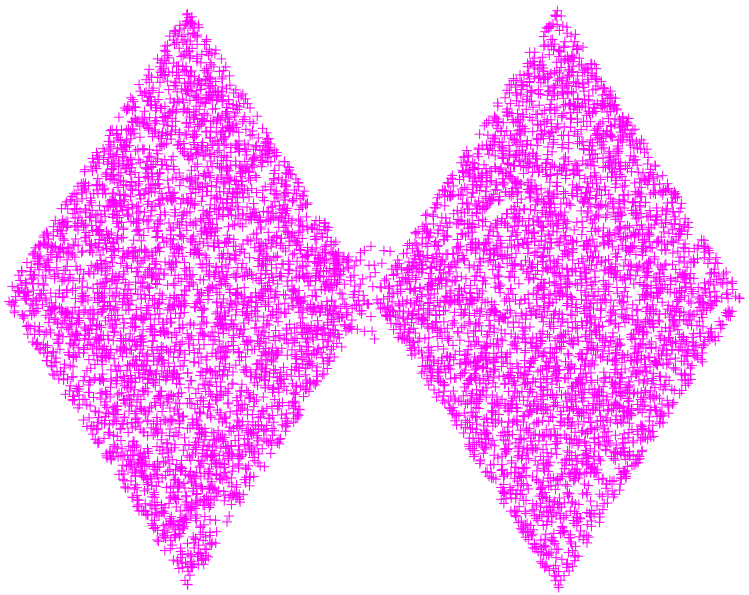
UNIVERSITÉ DE NANTES

lina LABORATOIRE D'INFORMATIQUE DE NANTES ATLANTIQUE

# Introduction

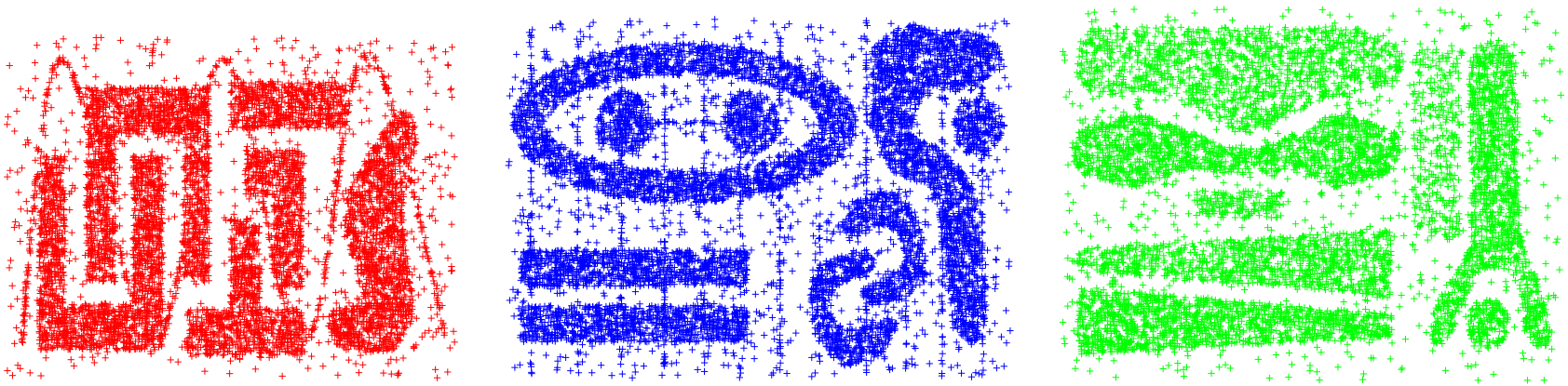
- Objectif = structuration des données
- Clustering (en anglais) = Classification (en français)

- On cherche à regrouper les points proches en classes ...
- Les classes peuvent être assez bien définies



# Introduction

- Les classes peuvent aussi être assez imbriquées,
- avoir des formes bizarres,
- ou pire



# Applications

| | Forme | Cluster |
|---|---|---|
| Text Mining | 1 Texte<br>1 Mail | Textes proches<br>"dossiers" automatiques |
| Bioinformatique | 1 gène | gènes ressemblants |
| Marketing | {infos client, produits achetés} | segmentation de la clientèle |
| Web log analysis | Clickstream | "utilisateurs types" |
| ... | | |

# Plan

## Points abordés

- Généralités
- Clustering hiérarchique
- Clustering par partitionnement

## Mais il existe d'autres familles de méthodes

- Clustering par modélisation
- Clustering basé sur la densité
- Clustering par "grille"

# Plan

## Points abordés

- **Généralités**
  - **Notion de proximité**
  - **Qualité d'une partition**
- Clustering hiérarchique
- Clustering par partitionnement

# Notion de proximité

## Vocabulaire

- Mesure de dissimilarité $DM$ :
  plus la mesure est faible plus les points sont similaires
  ($\sim$ distance)
- Mesure de similarité $SM$ :
  plus la mesure est grande, plus les points sont similaires
- $DM = \text{borne} - SM$

# Notion de proximité

## Comment mesurer la distance entre 2 points ? $d(x_1, x_2)$

- distance euclidienne :
  $d^2(x_1, x_2) = \sum_i (x_{1_i} - x_{2_i})^2 = (x_1 - x_2)(x_1 - x_2)'$
- distance de Manhattan :
  $d(x_1, x_2) = \sum_i |x_{1_i} - x_{2_i}|$
- distance de Sebestyen :
  $d^2(x_1, x_2) = (x_1 - x_2)W(x_1 - x_2)'$
  ($W$= matrice diagonale de pondération)
- distance de Mahalanobis :
  $d^2(x_1, x_2) = (x_1 - x_2)C^{-1}(x_1 - x_2)'$ ($V$=covariance)

# Notion de proximité

## Et si $x_1$ et $x_2$ sont à valeurs discrètes ?

On peut les répresenter dans une matrice de contingence $A(x_1, x_2) = [a_{ij}]$

- $x_1 = [0, 1, 2, 1, 2, 1]'$ et $x_1 = [1, 0, 2, 1, 0, 1]'$
- $A(x_1, x_2) = [0\ 1\ 0\ ; 1\ 2\ 0; 1\ 0\ 1]$

## distance de Hamming

- nombre de places où les 2 vecteurs diffèrent:

$$d(x_1, x_2) = \sum_{i=0}^{j=k-1} \sum_{j=0, j\neq i}^{j=k-1} a_{ij}$$

# Notion de proximité

**Comment mesurer la distance entre 2 classes** $D(C_1, C_2)$

- plus proche voisin : $\min(d(i,j), i \in C_1, j \in C_2)$
- diamètre maximum : $\max(d(i,j), i \in C_1, j \in C_2)$
- distance moyenne : $\frac{\sum_{i,j} d(i,j)}{n_1 n_2}$
- distance des centres de gravité : $d(\mu_1, \mu_2)$
- distance de Ward : $\sqrt{\frac{n_1 n_2}{n_1+n_2}} d(\mu_1, \mu_2)$
- ...

# Bonne partition

## Comment savoir si un regroupement est "correct"

- inertie (intra) d'un cluster = variance des points d'un même cluster

$$J_w = \sum_g \sum_{i \in C_g} d^2(x_i, \mu_g)$$

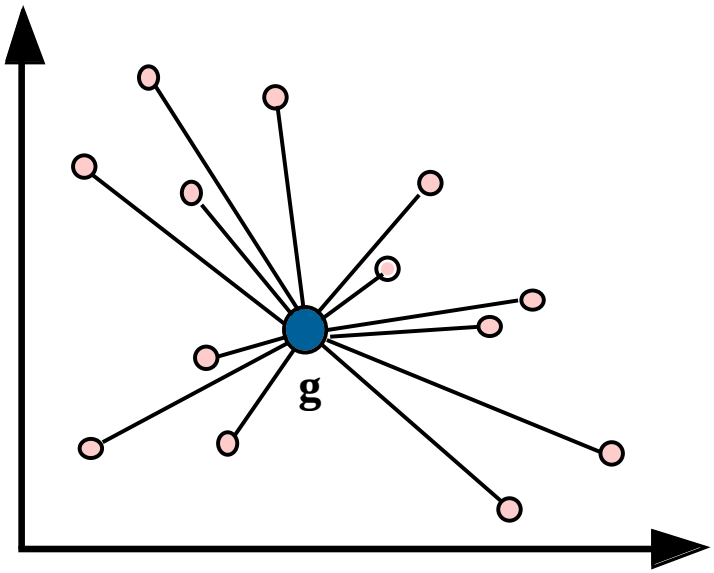
- inertie (inter) = variance des centres des clusters

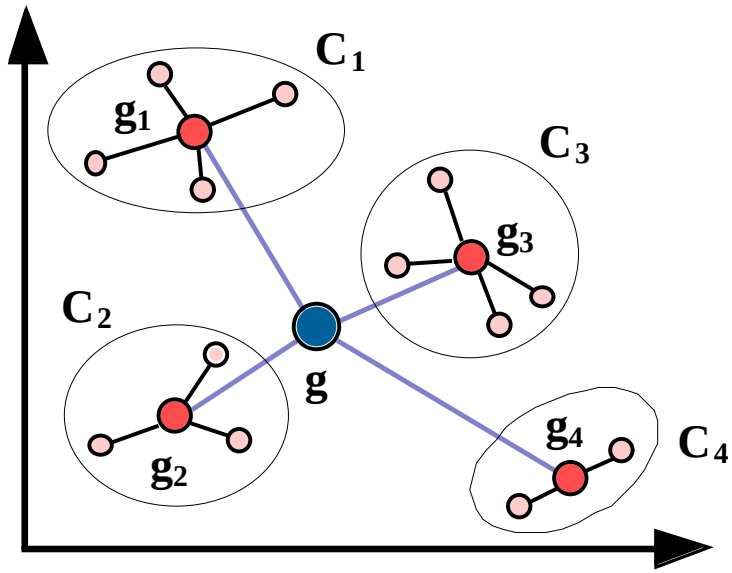
$$J_b = \sum_c N_g \ d^2(\mu_g, \bar{x})$$

- il faut minimiser l'inertie intra-cluster et maximiser l'inertie inter-cluster

# Bonne partition

Illustration (Bisson 2001)

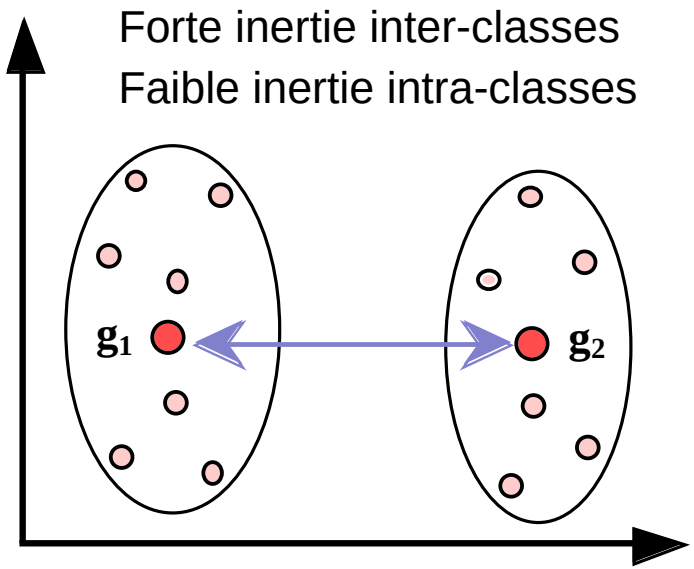


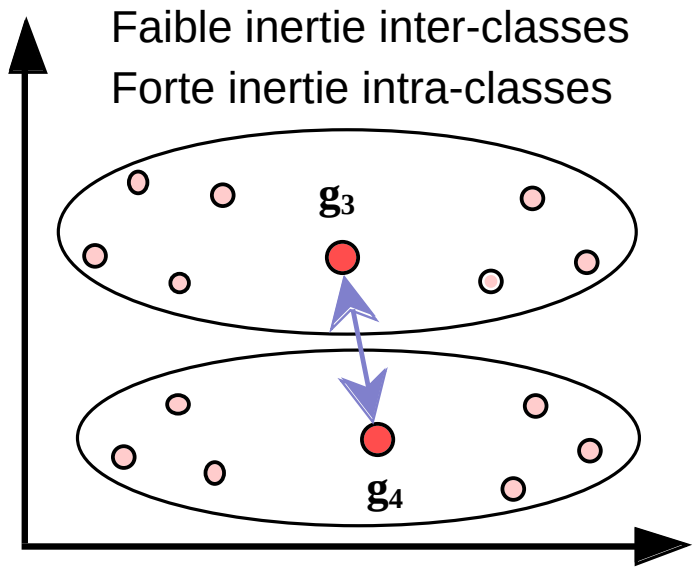




Inertie totale des points = Inertie Intra + Inter

# Bonne partition

Illustration (Bisson 2001)



# Plan

## Points abordés

- Généralités
- **Clustering hiérarchique** :
  - **Classification Hiérarchique Ascendante**
  - BIRCH, CURE, Chameleon, ...
- Clustering par partitionnement

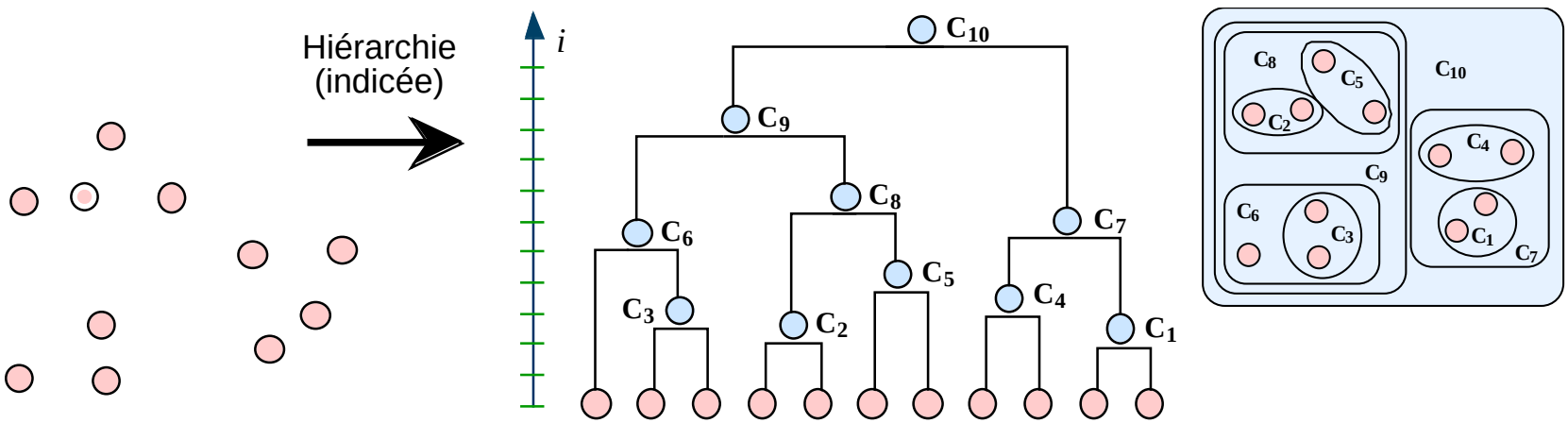
# Principe

Chaque point ou cluster est progressivement "absorbé" par le cluster le plus proche

## Algorithme

- Initialisation :
  - Chaque individu est placé dans son propre cluster,
  - Calcul de la matrice de ressemblance $M$ entre chaque couple de clusters (ici les points)
- Répéter
  - Sélection dans $M$ des deux clusters les plus proches $C_I$ et $C_J$
  - Fusion de $C_I$ et $C_J$ par un cluster $C_G$ plus général
  - Mise à jour de $M$ en calculant la ressemblance entre $C_G$ et les clusters existants
- Jusqu'à la fusion des 2 derniers clusters

# Exemple (Bisson 2001)




## Dendogramme

- représentation des fusions successives
- hauteur d'un cluster dans le dendrogramme = similarité entre les 2 clusters avant fusion (sauf exception avec certaines mesures de similarité)...

# Métrique

## Saut minimal (single linkage)

- tendance à produire des classes générales (par effet de chaînage)
- sensibilité aux individus bruités.

## Saut maximal (complete linkage)

- tendance à produire des classes spécifiques (on ne regroupe que des classes très proches)
- sensibilité aux individus bruités.

## Saut moyen

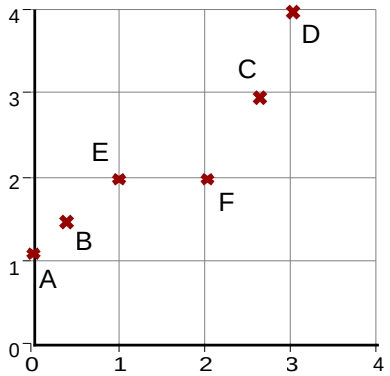
- tendance à produire des classes de variance proche

## Barycentre

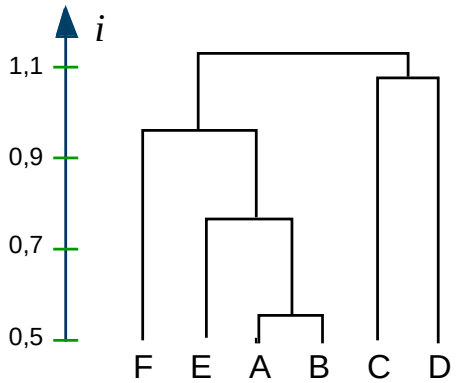
- bonne résistance au bruit

# Exemple (Bisson 2001)

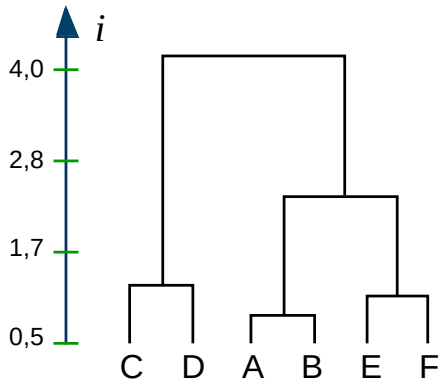
Données (métrique : dist. Eucl.)




Saut minimal




Saut maximal



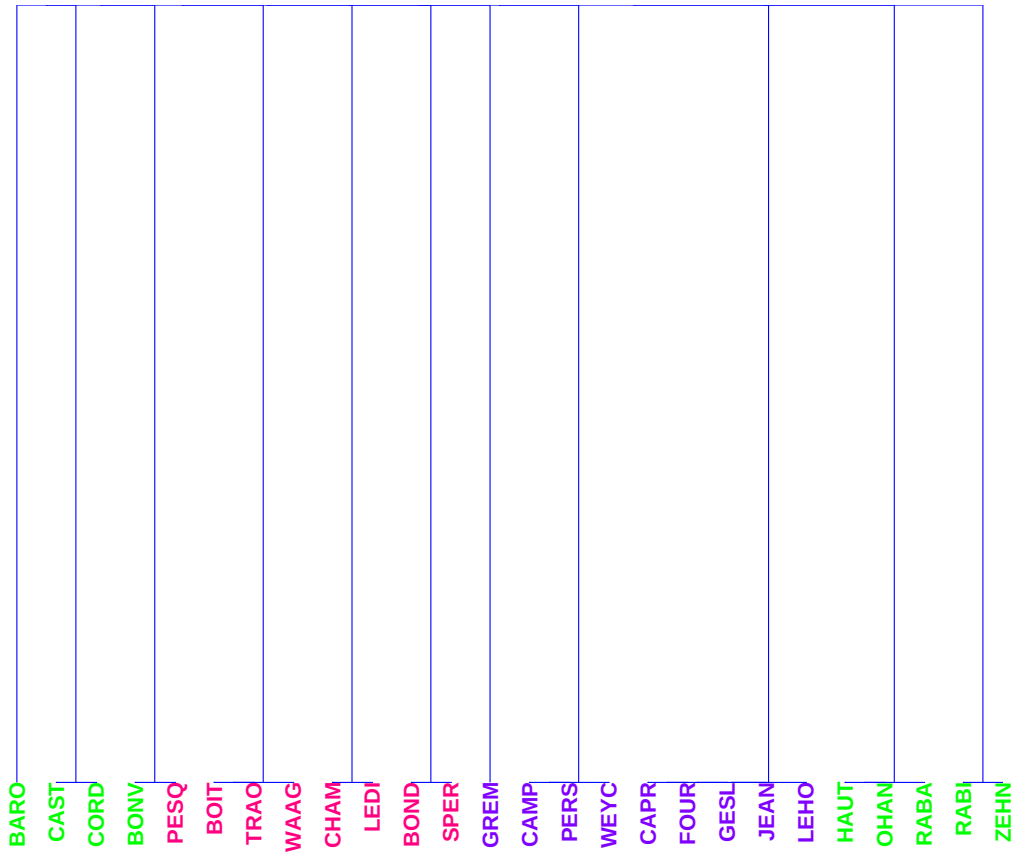

- pas les mêmes résultats selon la métrique utilisée ...

# Exemple 2

## Données

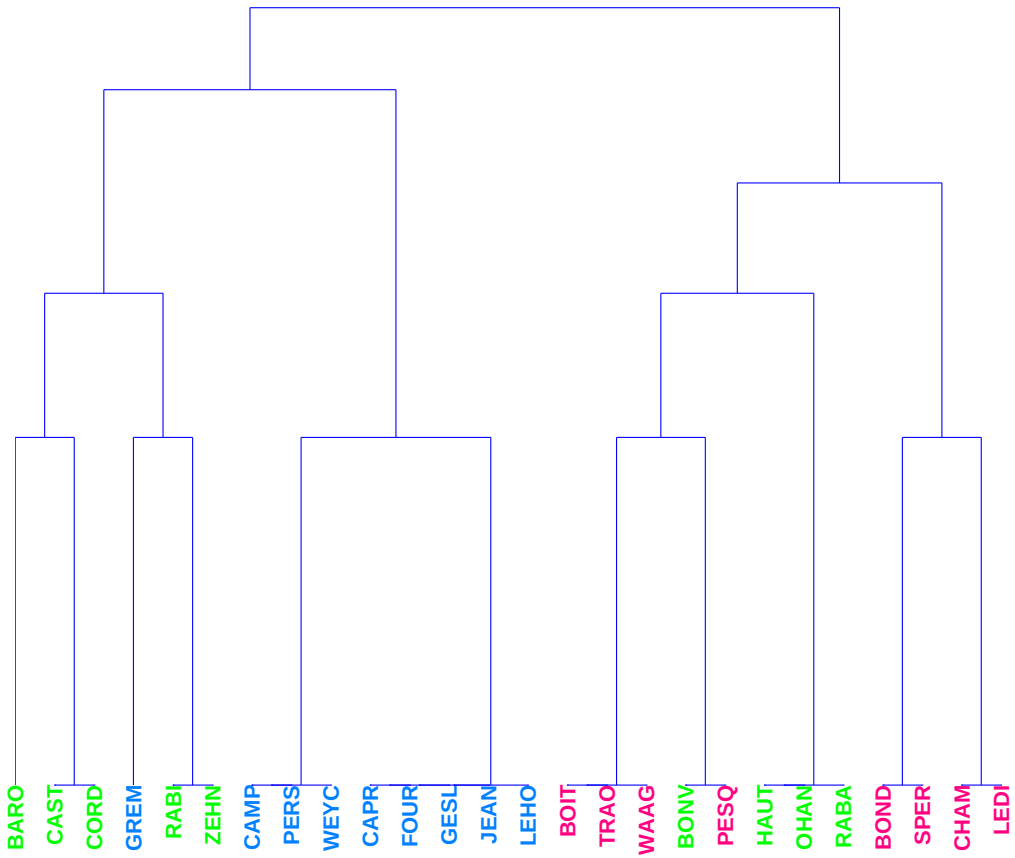
- 26 individus
- Données = 5 valeurs 0/1 (inscriptions dans 5 matières optionnelles)
- Est-on capable de faire ressortir des groupes "d'affinité" ?

# Exemple 2 : saut minimal



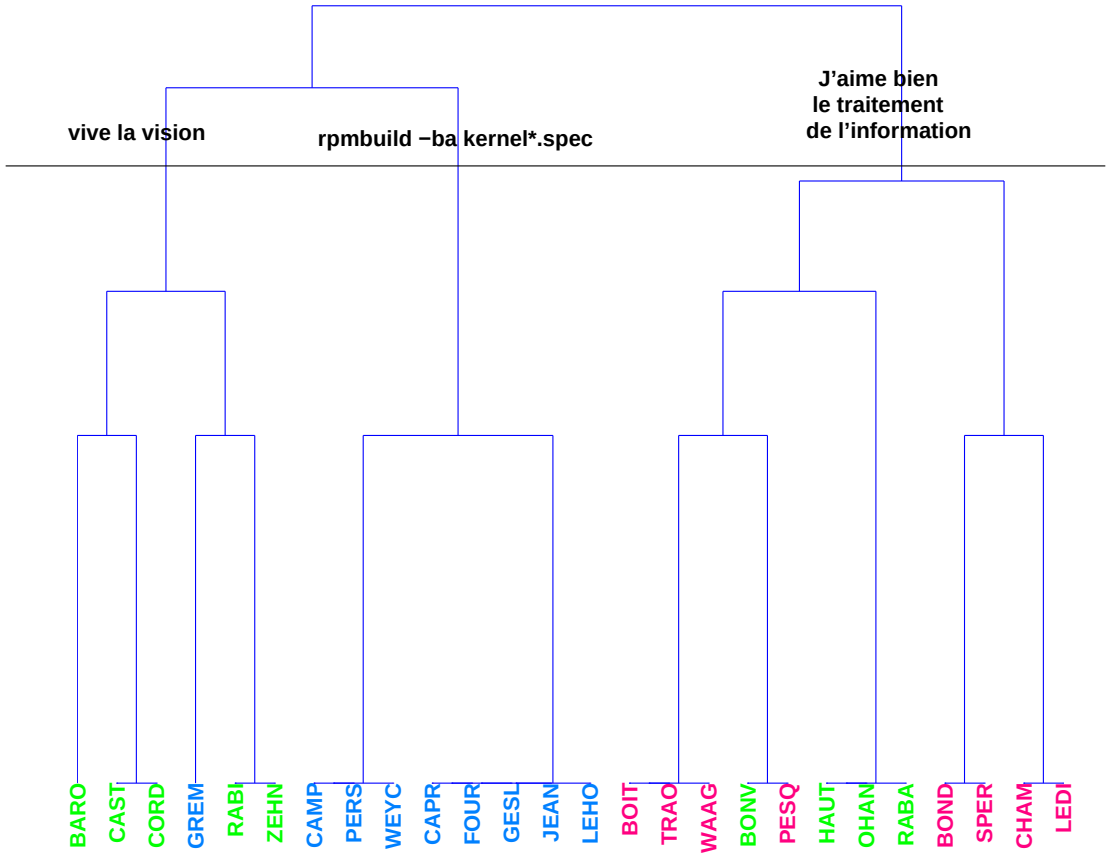

Pas moyen de faire de sous-groupes, tout le monde est à une distance de 0 ou 1 des autres clusters.

# Exemple 2 : saut maximal




En changeant de métrique, on observe davantage de sous-regroupements

# Exemple 2 : saut maximal




Pour construire les clusters, on coupe l'arbre à la hauteur voulue, i.e. une distance "seuil" entre les clusters

# Complexité

- Classification Hierarchique Ascendante : $O(n^3)$
- Certaines variantes de CHA
  - CURE : $O(n^2)$ en petite dim., $O(n^2 + nm \log n)$ sinon (en commençant avec $m$ clusters)
  - CHAMELEON :
    - graphe k-ppv : $O(n \log n)$ en petite dim. sinon $O(n^2)$
    - clustering : $O(nm + n \log n + m^2 \log m)$