Analyse de Données – Développeur
C5-160512-INFO
4. Classification Non Supervisée

Pr. Carl FRÉLICOT

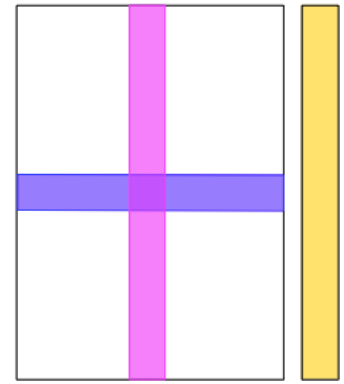
FST/PAS221 – carl.frelicot@univ-lr.fr

Licence d'Informatique 3ème année

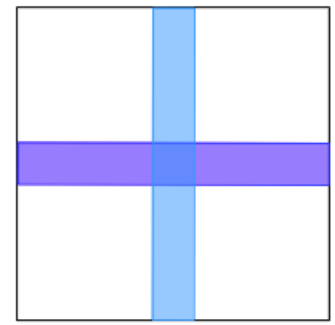
Automne 2021 - La Rochelle

- individus *vs* variables quantitatives
  données dites *vectorielles*
  $X = [X_{ki}]_{k=1,n;i=1,p}$ nuage de n points en dim. p
  données sans étiquette de groupe cas non supervisé
- individus *vs* individus
  données non vectorielles
  ex. : tableau de distances, proximités, etc
  $D = [d_{kl}]_{k,l=1,n}$ où $d_{kl} = d(x_k, x_l)$
  K-Means, Hierarchical Clustering
- variables *vs* variables *Feature Clustering*
  $D = [d_{ij}]_{i,j=1,p}$ où $d_{ij} = d(X_i, X_j)$ par exemple ?

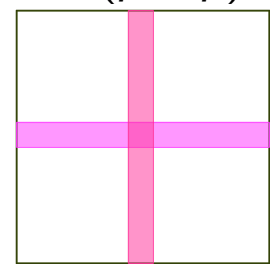
$X\ (n \times p)$ $Y\ (n \times 1)$



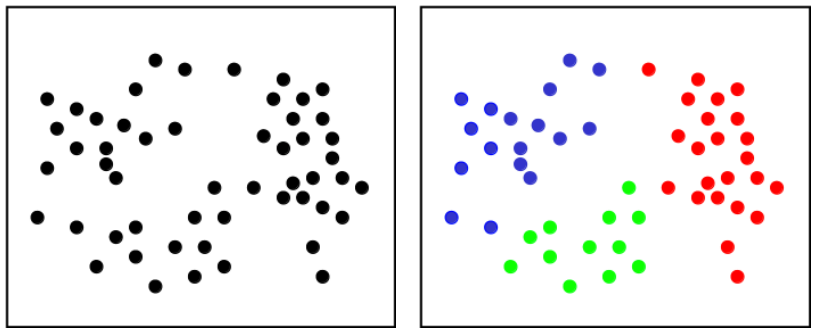
$D\ (n \times n)$



$D\ (p \times p)$



But : identifier une structure de groupes dans X
afin d'attribuer à tout x une étiquette y



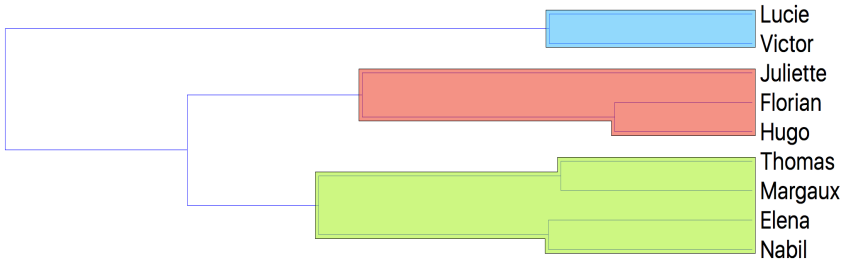
- différents types de classification non supervisée selon qu'on cherche :
  - à modéliser les classes par des distributions conditionnellement aux groupes (approche probabiliste) — *mixture decomposition*
  - une structure particulière (approche métrique)
    - partition — *K-Means*
    - hiérarchie — *Hierarchical Clustering*
    - recouvrement — plusieurs étiquettes

Exemple : notes de $n = 9$ élèves de Terminale S dans $p = 5$ matières

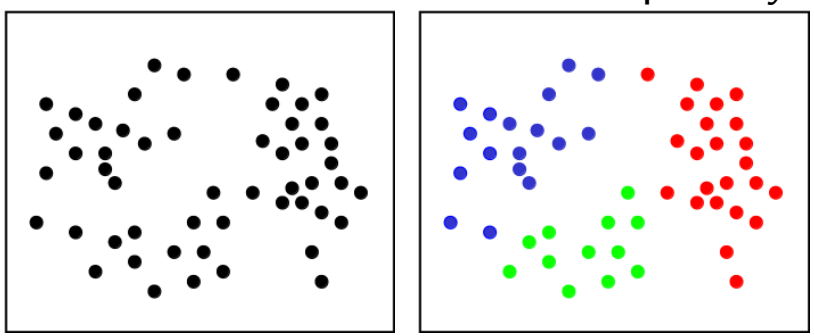
	Prénom	math.	info.	fran.	angl.	arts
1	Thomas	6.0	6.0	5.0	5.5	8.0
2	Margaux	8.0	8.0	8.0	8.0	9.0
3	Florian	6.0	7.0	11.0	10.0	11.0
4	Lucie	15.0	14.0	16.0	15.0	8.0
5	Victor	14.0	14.0	12.0	12.5	10.0
6	Elena	11.0	10.0	5.5	7.0	13.0
7	Hugo	5.5	7.0	14.0	11.5	10.0
8	Nabil	13.0	12.5	8.5	9.5	12.0
9	Juliette	9.0	9.5	12.5	12.0	18.0

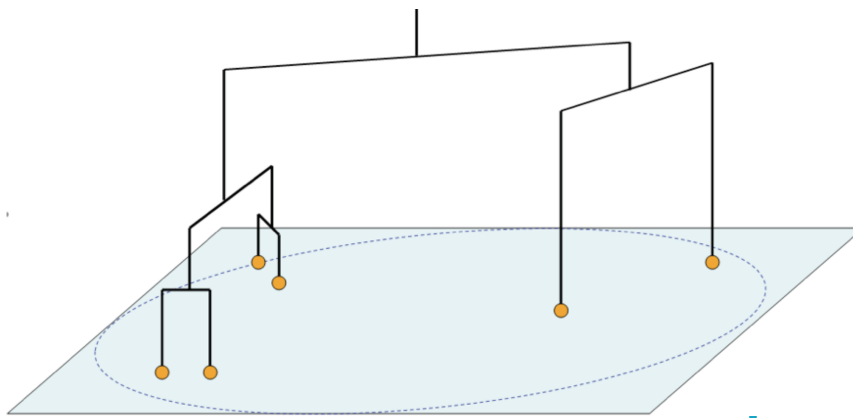
	Prénom	Cluster
1	Thomas	C3
2	Margaux	C3
3	Florian	C2
4	Lucie	C1
5	Victor	C1
6	Elena	C3
7	Hugo	C2
8	Nabil	C3
9	Juliette	C2

lien entre partition et hiérarchie ?



But : identifier une structure de groupes dans X afin d'attribuer à tout x une étiquette y





Distances entre individus : une distance métrique[(a)] d sur un ensemble X :
$X \times X \to \mathbb{R}_+, (x,y) \mapsto d(x,x)$ vérifiant

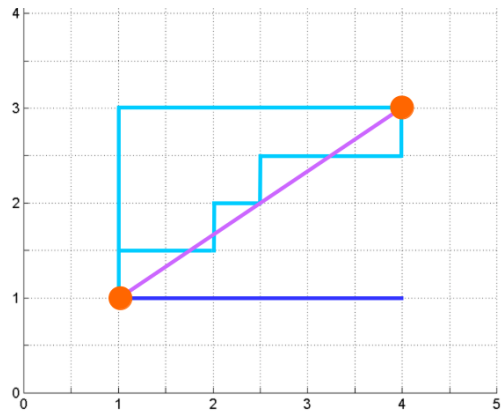
(1) $d(x,y) = d(y,x)$ *symétrie*
(2) $d(x,y) = 0 \Leftrightarrow x = y$ *indiscernabilité*
(3) $d(x,z) \leq d(x,y) + d(y,z)$ *inégalité triang.*

(a) si tout $x \in X$ se projette dans un espace (vectoriel) $\mathcal{X}$, $(\mathcal{X}, d)$ est appelé *espace métrique* ; pour nous $\mathcal{X} = \mathbb{R}^p$, d'où le raccourci de langage entre **distance** et **métrique**

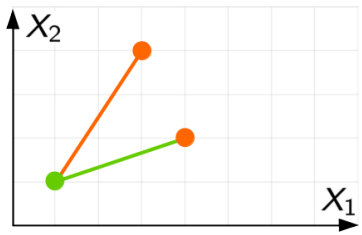
- une pseudométrique ne vérifie que (1)-(3)
- une semimétrique ou disssimilarité ne vérifie que (1)-(2) suffit en Classif. NS
- une ultramétrique vérifie (1)-(2)-(3') $d(x,z) \leq max(d(x,y), d(y,z))$ distance ?

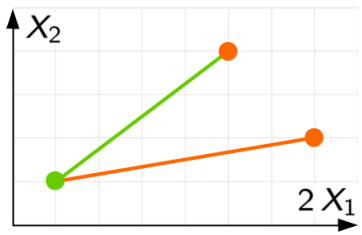
Distances usuelles : distances de Minkowski $d_q(x,y) = \left(\sum_{j=1}^{p} |x_j - y_j|^q\right)^{1/q}$

- *Manhattan* ou *cityblock* si $q = 1$
- *euclidenne* si $q = 2$; alors $d_2^2(x,y) = ||x-y||_I^2$
  ...
- *Chebychev* si $q \to +\infty$;
  alors $d_\infty(x,y) = \max_{j=1,p} |x_j - y_j|$



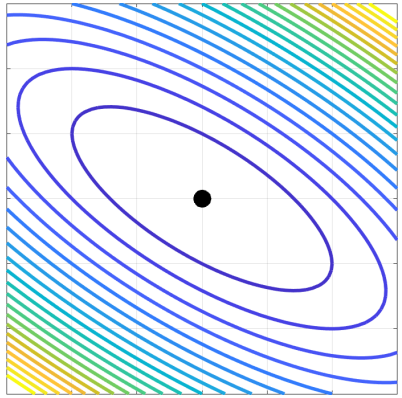
Pondération des variables !

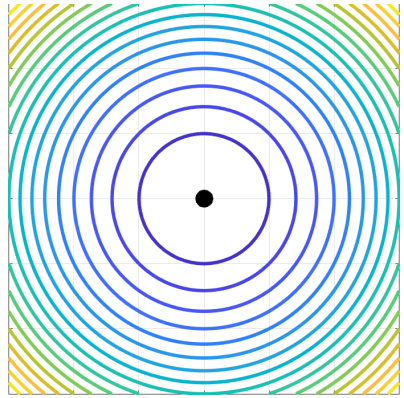




Distance de Mahalanobis : $d_M^2(x,y) = ||x-y||_{V^{-1}}^2 = {}^t(x-y)\,V^{-1}\,(x-y)$

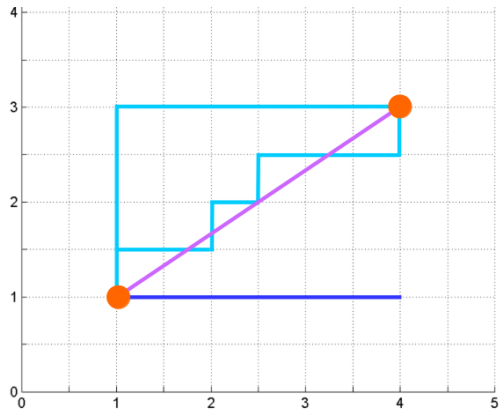
- si V est diagonale, alors les ellipses d'iso-distance sont parallèles aux axes
- si $V = I$? alors $d_M^2(x,y) = d_2^2(x,y)$



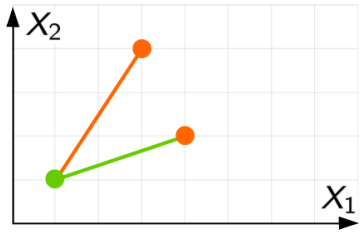


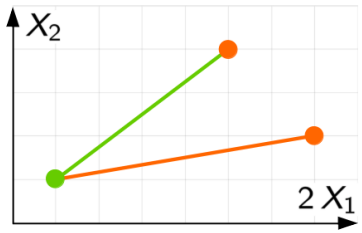
Distances usuelles : distances de Minkowski $d_q(x,y) = \left(\sum_{j=1}^{p} |x_j - y_j|^q\right)^{1/q}$

- *Manhattan* ou *cityblock* si $q = 1$
- *euclidenne* si $q = 2$; alors $d_2^2(x,y) = ||x-y||_I^2$
  ...
- *Chebychev* si $q \to +\infty$;
  alors $d_\infty(x,y) = \max_{j=1,p} |x_j - y_j|$



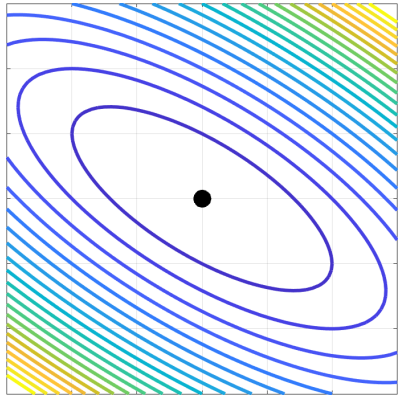
Pondération des variables !

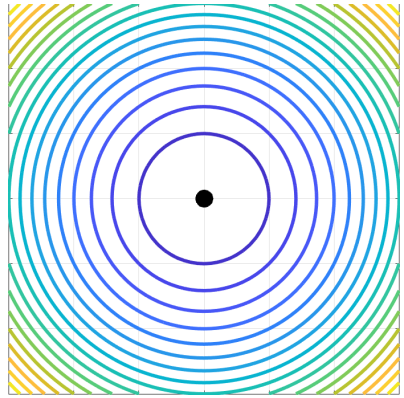




Distance de Mahalanobis : $d_M^2(x,y) = ||x-y||_{V^{-1}}^2 = {}^t(x-y)\,V^{-1}\,(x-y)$

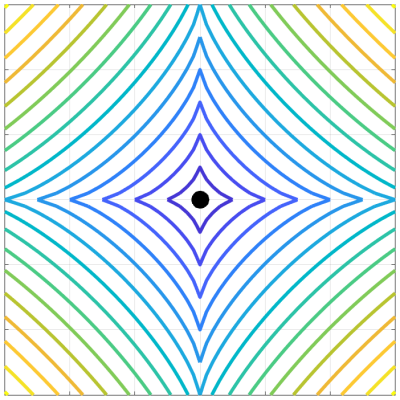
- si V est diagonale, alors les ellipses d'iso-distance sont parallèles aux axes
- si $V = I$? alors $d_M^2(x,y) = d_2^2(x,y)$



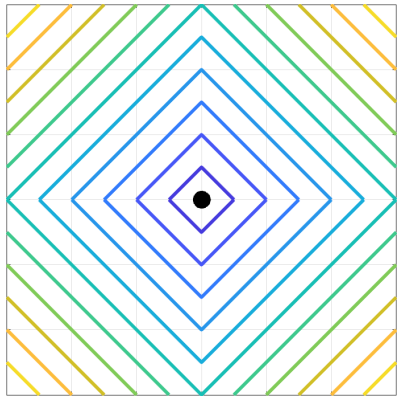


$q = 2$

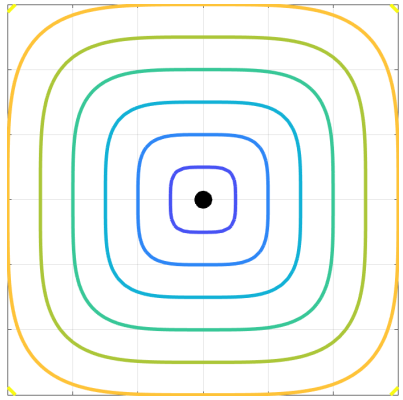
Distances usuelles : distances de Minkowski $d_q(x,y) = \left(\sum_{j=1}^{p} |x_j - y_j|^q\right)^{1/q}$



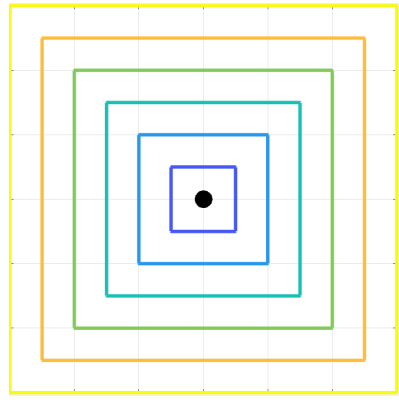
q ? $q = 0.75$



$q = 1$



$q = 4$



$q = +\infty$

Partition : une **partition stricte** d'un ensemble $X = \{x_1, x_2, ..., x_n\}$ est un ensemble $P = \{C_1, C_2, ..., C_c\}$ de groupes (*clusters*) disjoints, couvrant X

- on lui associe une **matrice de partition** $U = [u_{ik}]_{i=1,c;k=1,n}$ où
  où $u_{ik} = 1$ ou 0 selon que $x_k \in C_i$ ou non, stricte
  telle que $\sum_i u_{ik} = 1, \forall k$
  et $0 < \sum_k u_{ik} < n, \forall i$

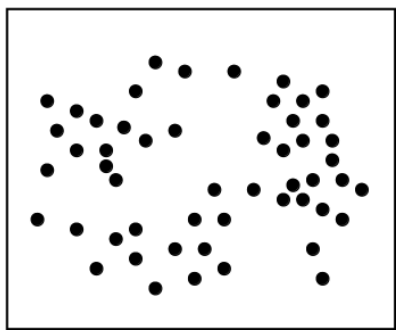
Problèmes :

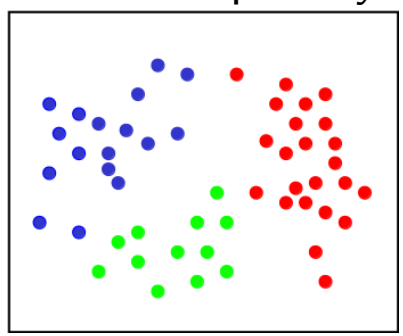
- nombre c de groupes ?
- la recherche de la "meilleure partition" en k classes d'un ensemble de n éléments est NP-complet ;
  nombre de Stirling : c et n grands

$$S(c,n) = \frac{1}{c!} \sum_{i=1}^{c} (-1)^{c-i} \binom{i}{c} i^n \quad S(c,n) \simeq \frac{c^n}{c!}$$

c	n	S
2	4	7
2	10	511
5	10	42 535
2	30	536 870 911
3	150	$> 6 \times 10^{70}$

But : identifier une structure de groupes dans X $X \rightarrow U$ ou P
afin d'attribuer à tout x une étiquette y





Sont nécessaires un(e) :

1. dissimilarité/distance d sur X
2. critère d'affectation $\mathcal{D}$ aux groupes
3. algorithme d'optimisation de $\mathcal{D}$

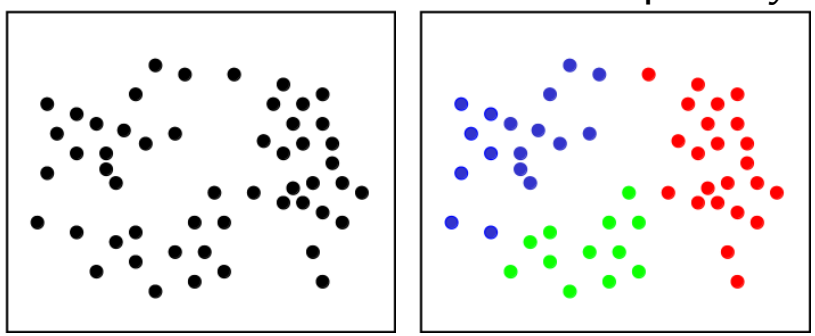
Algorithme(s) C-means

K-moyennes ou "centres mobiles"

- méthode de *clustering* : recherche itérative d'une partition et de représentants (?) — $X \rightarrow U, V$; $V = \{\overline{x}_1, \overline{x}_2, ..., \overline{x}_c\}$
- paramètres utilisateurs :
  - nombre c de clusters — attention : $K == c$
  - paramètre d'arrêt ε — ou nombre max. d'iérations
  1. distance d — d_2 par défaut
- critère $\mathcal{D}(U,V) = \frac{1}{n}\sum_{i=1}^{c}\sum_{x_k \in C_i} d_2^2(x_k, \overline{x}_i)$ — inertie intra-clusters ; min ? max ?

0. initialisation aléatoire : $U^{(0)}$ — ou $V^{(0)}$
1. tant que non convergence — stabilité de la partition
   1. $V^{(t)} = argmin_V\, \mathcal{D}(U^{(t-1)}, V)$ — prototypage
   2. $U^{(t)} = argmin_U\, \mathcal{D}(U, V^{(t)})$ — affectation

But : identifier une structure de groupes dans X afin d'attribuer à tout x une étiquette y — $X \rightarrow U$ ou P



Sont nécessaires un(e) :

1. dissimilarité/distance d sur X
2. critère d'affectation $\mathcal{D}$ aux groupes
3. algorithme d'optimisation de $\mathcal{D}$

Algorithme(s) C-means

K-moyennes ou "centres mobiles"

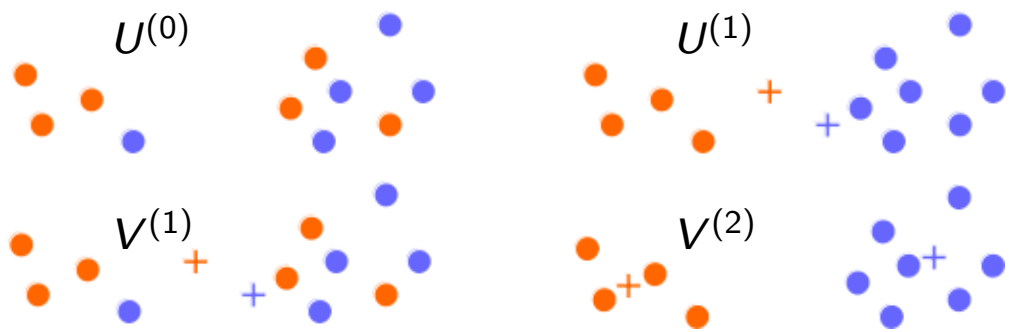
- méthode de *clustering* : recherche itérative d'une partition $X \rightarrow U, V$
  et de représentants (?) $V = \{\overline{x}_1, \overline{x}_2, ..., \overline{x}_c\}$
- paramètres utilisateurs :
  - nombre c de clusters — attention : $K == c$
  - paramètre d'arrêt ε — ou nombre max. d'iérations
  1. distance d — d_2 par défaut
- critère $\mathcal{D}(U,V) = \frac{1}{n}\sum_{i=1}^{c}\sum_{x_k \in C_i} d_2^2(x_k, \overline{x}_i) = \frac{1}{n}\sum_{k=1}^{n} d_2^2(x_k, \overline{x}_{C_i(x_k)})$ — inertie intra
  min ~~? max ?~~

0. initialisation aléatoire : $U^{(0)}$ — ou $V^{(0)}$
1. tant que non convergence — stabilité de la partition
   1. $V^{(t)} = argmin_V\, \mathcal{D}(U^{(t-1)}, V)$ — prototypage
   2. $U^{(t)} = argmin_U\, \mathcal{D}(U, V^{(t)})$ — affectation

   convergence : $||U^{(t)} - U^{(t-1)}|| < \varepsilon$ — norme matricielle, par ex. : *max*
   ou *maxiter*

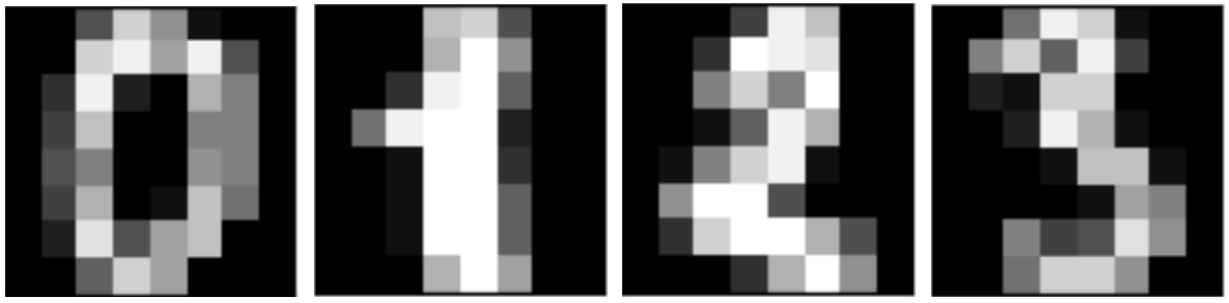
- $U^{(0)} \rightarrow V^{(1)} \rightarrow U^{(1)} \rightarrow V^{(2)} \ldots V^{(t)} = V^{(t-1)}$
  $V^{(0)} \rightarrow U^{(1)} \rightarrow V^{(1)} \rightarrow U^{(2)} \ldots U^{(t)} = U^{(t-1)}$

$U^{(2)} = U^{(1)}$ STOP



Exemple : chiffres manuscrits

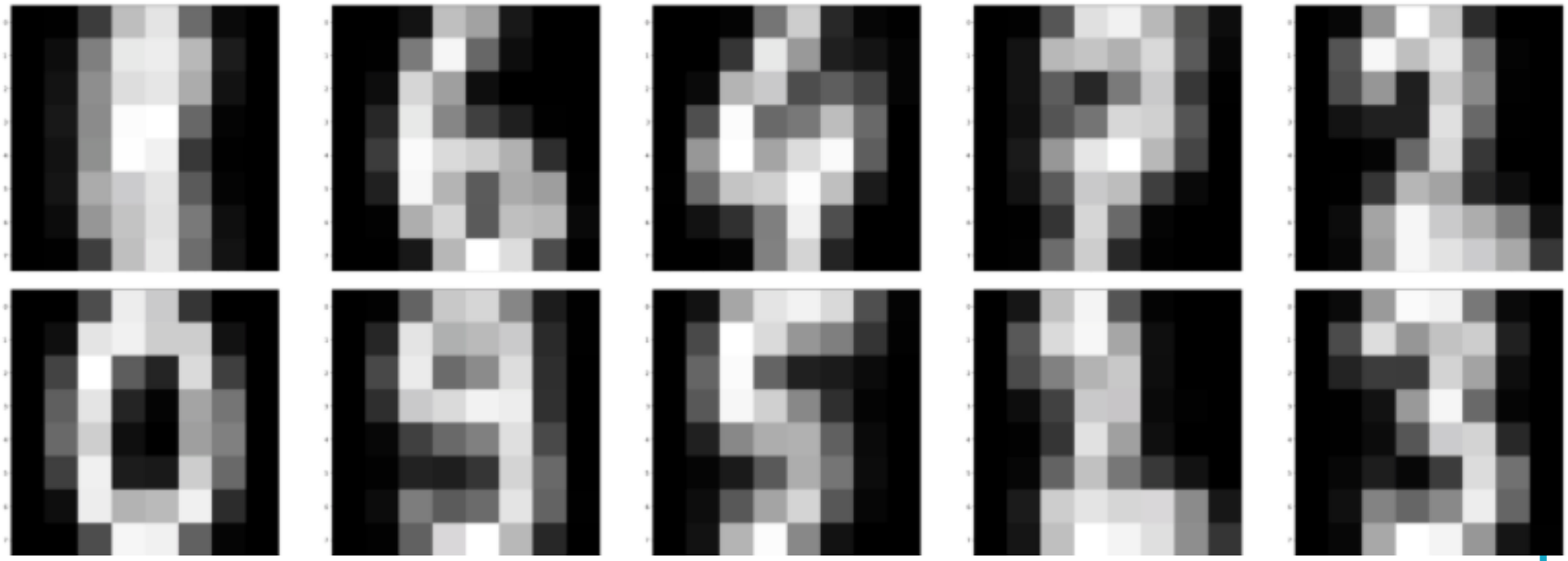
ex. images originales



- 10 chiffres
  1 797 imagettes
  ($\simeq$ 180 par chiffre)
  de très petite taille
  $64 = 8 \times 8$ pixels

- barycentres trouvés

8, 6, 4, 7, 2



- taux d'accord avec la vérité-terrain : 92%
  supervision connue, par expertise / connaissance humaine

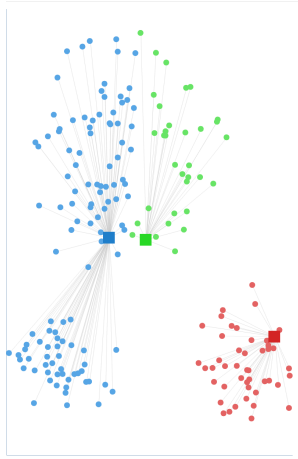
0, 9, 5, 1, 3
ground-truth

⊕ intuitif et rapide — comparé à *HAC*

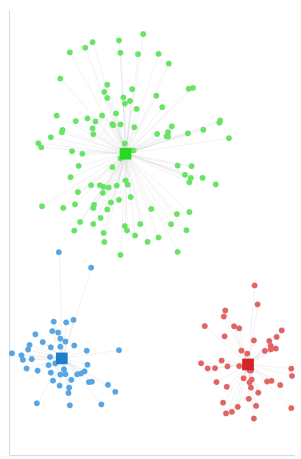
⊖ sensible à l'initialisation — optimum local

solution : exécuter r fois, retenir les points qui finissent toujours ensemble — formes fortes

run #1

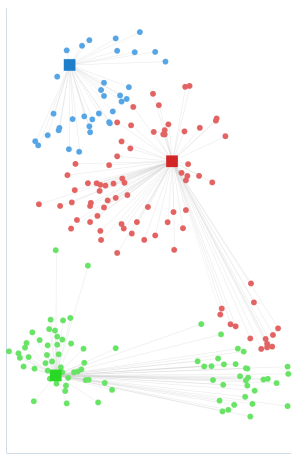


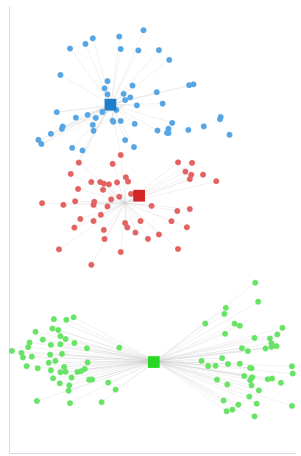
initialisation



partition finale

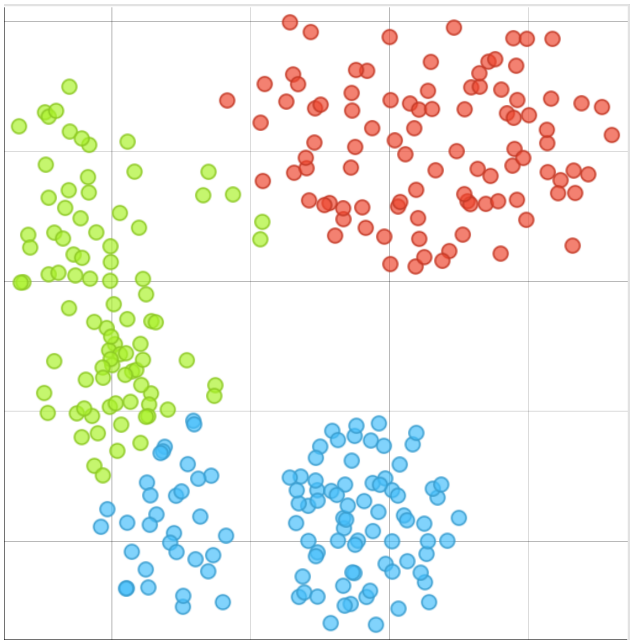
run #2

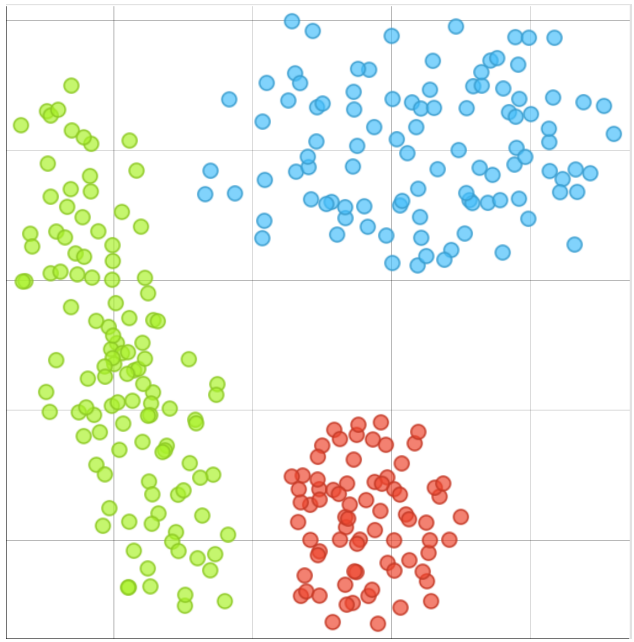




- ⊕ intuitif et rapide — comparé à *HAC*
- ⊖ sensible à l'initialisation — optimum local
  solution : exécuter r fois,
  retenir les points qui finissent toujours ensemble — formes fortes
- ⊖ tendance à former des clusters équilibrés — inertie intra-classes
  - effectif
  - dispersion

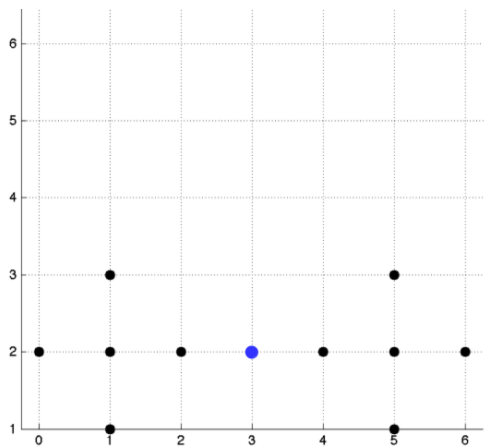
  solution : adapter la métrique, donc le critère — variantes
- ⊖ tendance à former des clusters hypersphériques — distance euclidienne
  solution : adapter la métrique — une idée ?

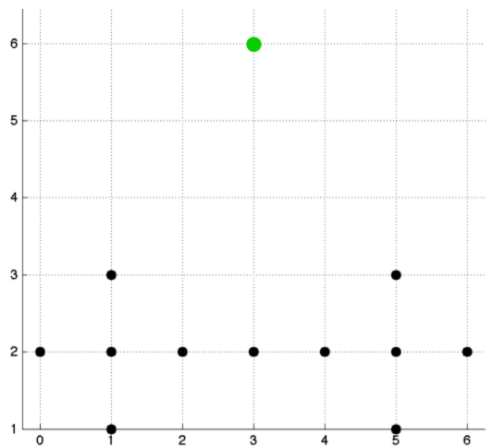




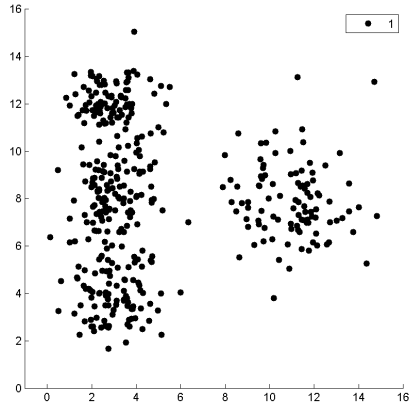
- ⊕ intuitif et rapide — comparé à *HAC*
- ⊖ sensible à l'initialisation — optimum local
  solution : exécuter r fois,
  retenir les points qui finissent toujours ensemble — formes fortes
- ⊖ tendance à former des clusters équilibrés — inertie intra-classes
  - effectif
  - dispersion

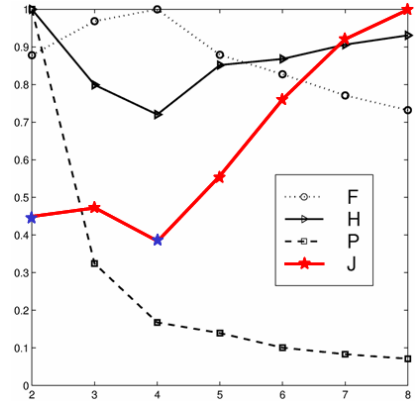
  solution : adapter la métrique, donc le critère — variantes
- ⊖ tendance à former des clusters hypersphériques — distance euclidienne
  solution : adapter la métrique — une idée ?
- ⊖ sensible aux *inliers* — partition stricte
  et *outliers* — C fixé : *cluster validity*





Cluster Validity Problem





- quel $c_{opt.}$?
- critères (non/liés à l'algorithme) fondés sur les points et/ou centres

  $c_{opt.} = argmin/max_{c=2,c_{max}}\ CVI(c)$

  Dunn Index :

  $$DI(c) = \frac{min_{1\leq i<i'\leq c} d(\overline{x}_i, \overline{x}_{i'})}{max_{j=1,c}\Delta_j} \quad \text{où } \Delta_j = max_{x_k,x_l\in C_j}\, d(x_k, x_l)$$

  min ou max ?

  certains *CVI* ont une tendance monotone avec c :(
- critères relatifs (comparaison de deux partitions en c et c' clusters)

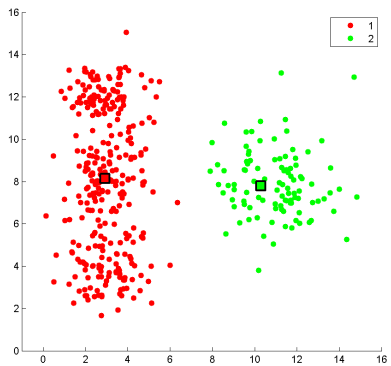
  Rand Index : $RI(P,Q) = \frac{2\,t-(u+v)}{n\,(n-1)} + 1$ min ou max ?

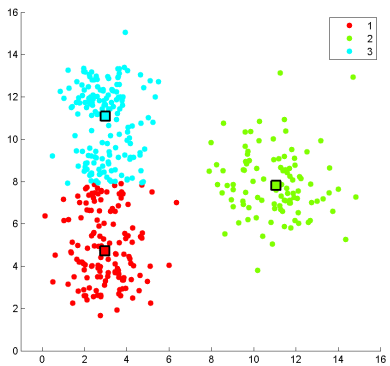
  - $t = \sum_{i=1}^{c}\sum_{j=1}^{c'} n_{ij}^2 - n$
  - $u = \sum_{i=1}^{c} n_{i\bullet}^2 - n$, où $n_{i\bullet} = \sum_{i=1}^{c'} n_{ij}$
  - $v = \sum_{j=1}^{c'} n_{\bullet j}^2 - n$, où $n_{\bullet j} = \sum_{j=1}^{c} n_{ij}$

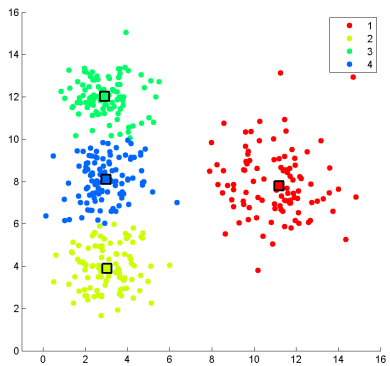
  à partir de la matrice d'accord :

  $N(P,Q) = {}^tP\,Q$

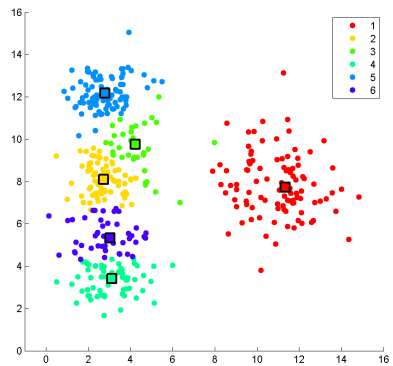
  de dimension $(p \times q)$



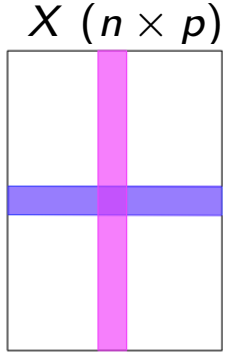




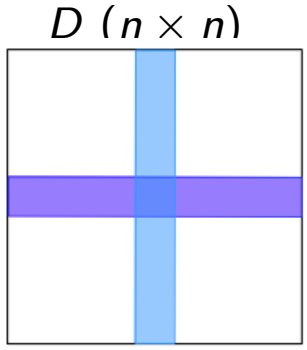
...



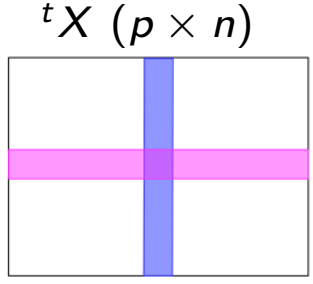
- individus *vs* variables quantitatives
  données dites *vectorielles*
  $X = [X_{ki}]_{k=1,n;i=1,p}$ nuage de n points en dim. p
  données sans étiquette de groupe cas non supervisé



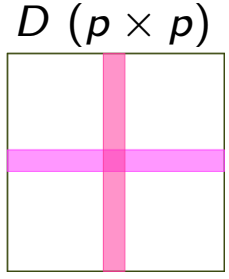
- individus *vs* individus
  données non vectorielles
  ex. : tableau de distances, proximités, etc
  $D = [d_{kl}]_{k,l=1,n}$ où $d_{kl} = d(x_k, x_l)$
  C-Means, Hierarchical Clustering



- variables *vs* individus
  données *vectorielles*
  ${}^tX = [X_{ik}]_{i=1,p;k=1,n}$ nuage de p points en dim. n

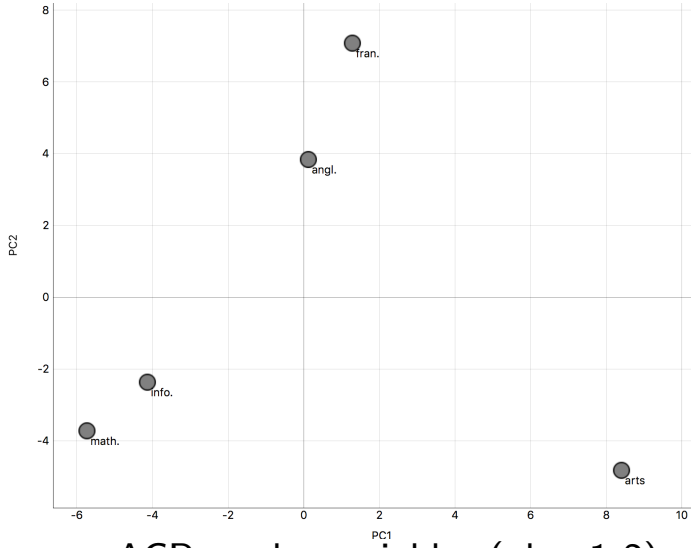


- variables *vs* variables
  données non vectorielles
  ex. : tableau de distances, proximités, etc
  $D = [d_{ij}]_{i,j=1,p}$ où $d_{ij} = d(X_i, X_j)$ par ex. ?
  Feature Clustering (C-Means, Hierarchical)



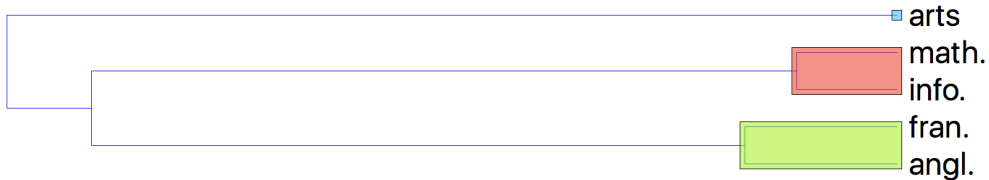
Exemple : notes de $n = 9$ élèves de Terminale S dans $p = 5$ matières

Feature name	Thomas	Margaux	Florian	Lucie	Victor	Elena	Hugo	Nabil	Juliette
math.	6.000	8.000	6.000	15.000	14.000	11.000	5.500	13.000	9.000
info.	6.000	8.000	7.000	14.000	14.000	10.000	7.000	12.500	9.500
fran.	5.000	8.000	11.000	16.000	12.000	5.500	14.000	8.500	12.500
angl.	5.500	8.000	10.000	15.000	12.500	7.000	11.500	9.500	12.000
arts	8.000	9.000	11.000	8.000	10.000	13.000	10.000	12.000	18.000



ACP sur les variables (plan 1-2)

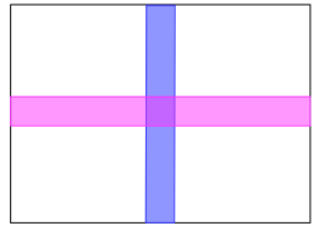
- nuage de variables ($p = 5 \times n = 9$)
- algorithme *C-means* ; $c = 3$



- utiliser les centres comme variables
  $\rightarrow$ espace de dimension réduite ($q = c < p$)

- variables *vs* individus
  données *vectorielles*
  ${}^tX = [X_{ik}]_{i=1,p;k=1,n}$

nuage de p points en dim. n



- variables *vs* variable
  données non vectorielles

  ex. : tableau de distances, proximités, etc

  $D = [d_{ij}]_{i,j=1,p}$ où $d_{ij} = d(X_i, X_j)$

  Feature Clustering (C-Means, Hierarchical)

par ex. ?