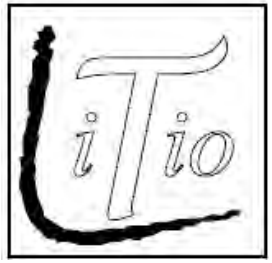


Apprentissage non supervisé

Mustapha Lebbah

Plan

- Introduction
- Classification Hiérarchique
- K-means
- Cartes auto-organisatrices
- Modèles de mélanges
- Quelques exemples

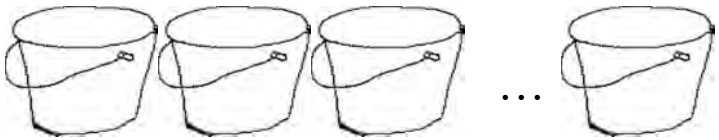
Problématique : clustering

Classification supervisée

Les classes et le nombre des classes sont connus

Classification non supervisée

Les classes et le nombre des classes ne sont pas disponibles



- *Les difficultés :*

- Existence réelle d'une structure

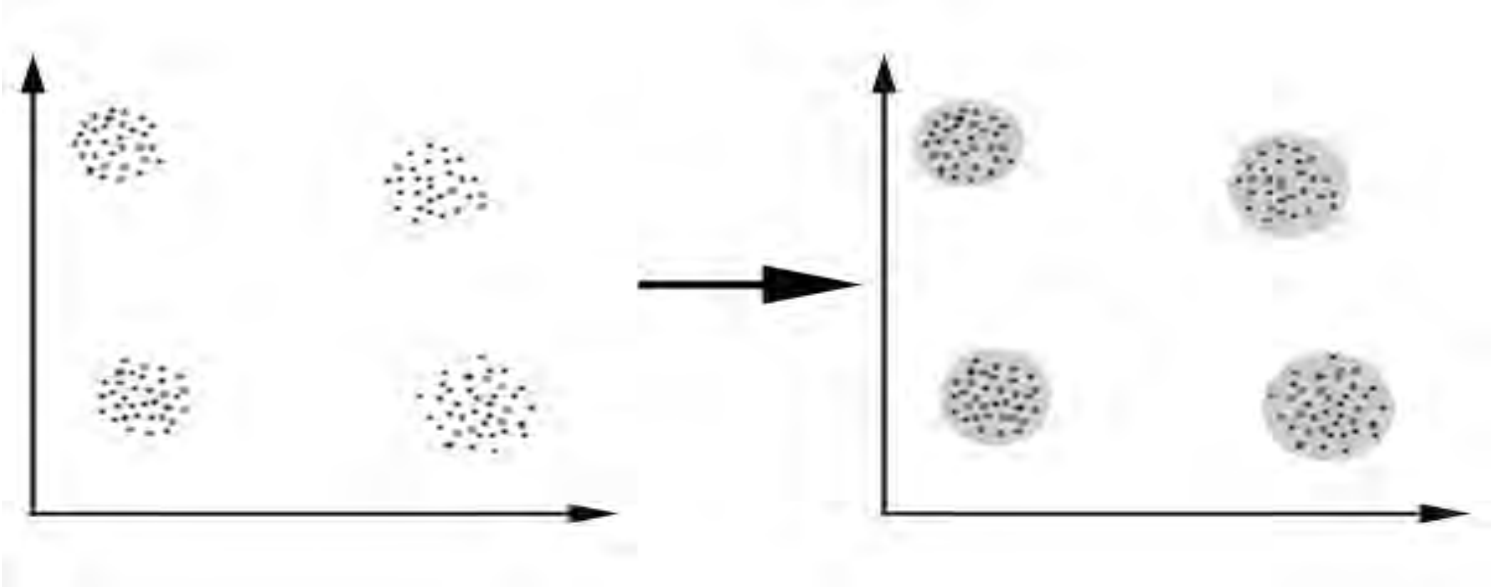
- Choix de similarité

- Choix du nombre de groupes (Combinatoire)

- Validation (absence de labels)

- Nature des données

C’est quoi le clustering ?



Trouver K clusters/ groupes/ensemble de données homogènes. (les données appartenant à des clusters différents sont dissimilaires)

construire des classes automatiquement en fonction des exemples disponibles

- *L'apprentissage non supervisé est très souvent synonyme de clustering*

Quelques bonnes raisons de s'intéresser à l'apprentissage non supervisé

➔Constituer des échantillons d'apprentissage étiquetés peut être très couteux

➔Découvertes de la structure et la nature des données à travers l'analyse exploratoire

- Utile pour l'étude des caractéristiques pertinentes
- Prétraitement avant l'application d'une autre technique de fouille de données

Approches de Clustering

- Algorithmes de Partitionnement: Construire plusieurs partitions puis les évaluer selon certains critères
- Algorithmes hiérarchiques: Créer une décomposition hiérarchique des objets selon certains critères
- Algorithmes basés sur la densité: basés sur des notions de connectivité et de densité
- À Base de modèle de mélange

Notion de proximité

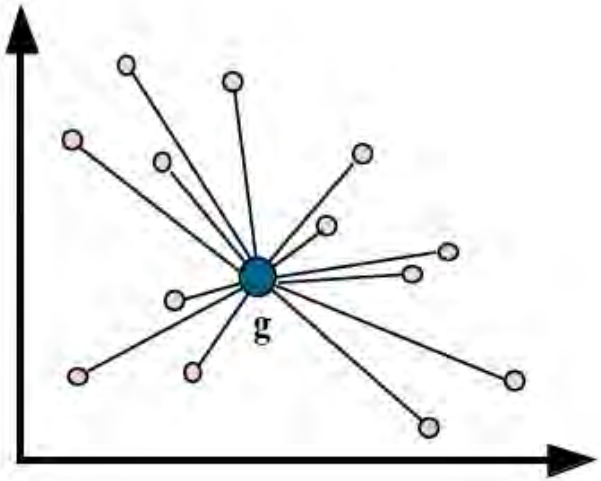
➔*Mesure de dissimilarité : plus la mesure est faible plus les points sont similaires (~ distance)*

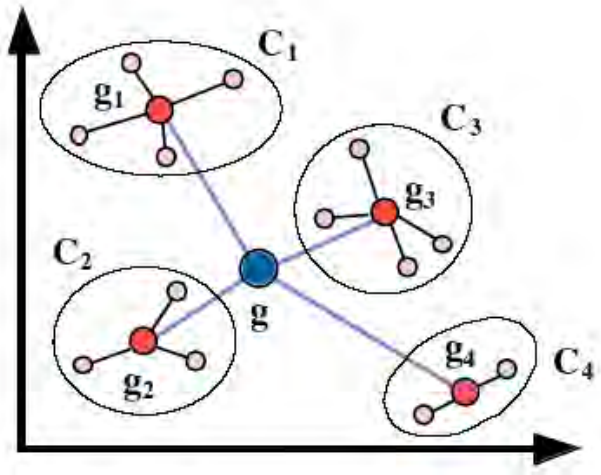
➔*Mesure de similarité : plus la mesure est grande, plus les points sont similaires*

Comment savoir si un regroupement est "correct" ?

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

- inertie (inter) = variance des centres des clusters

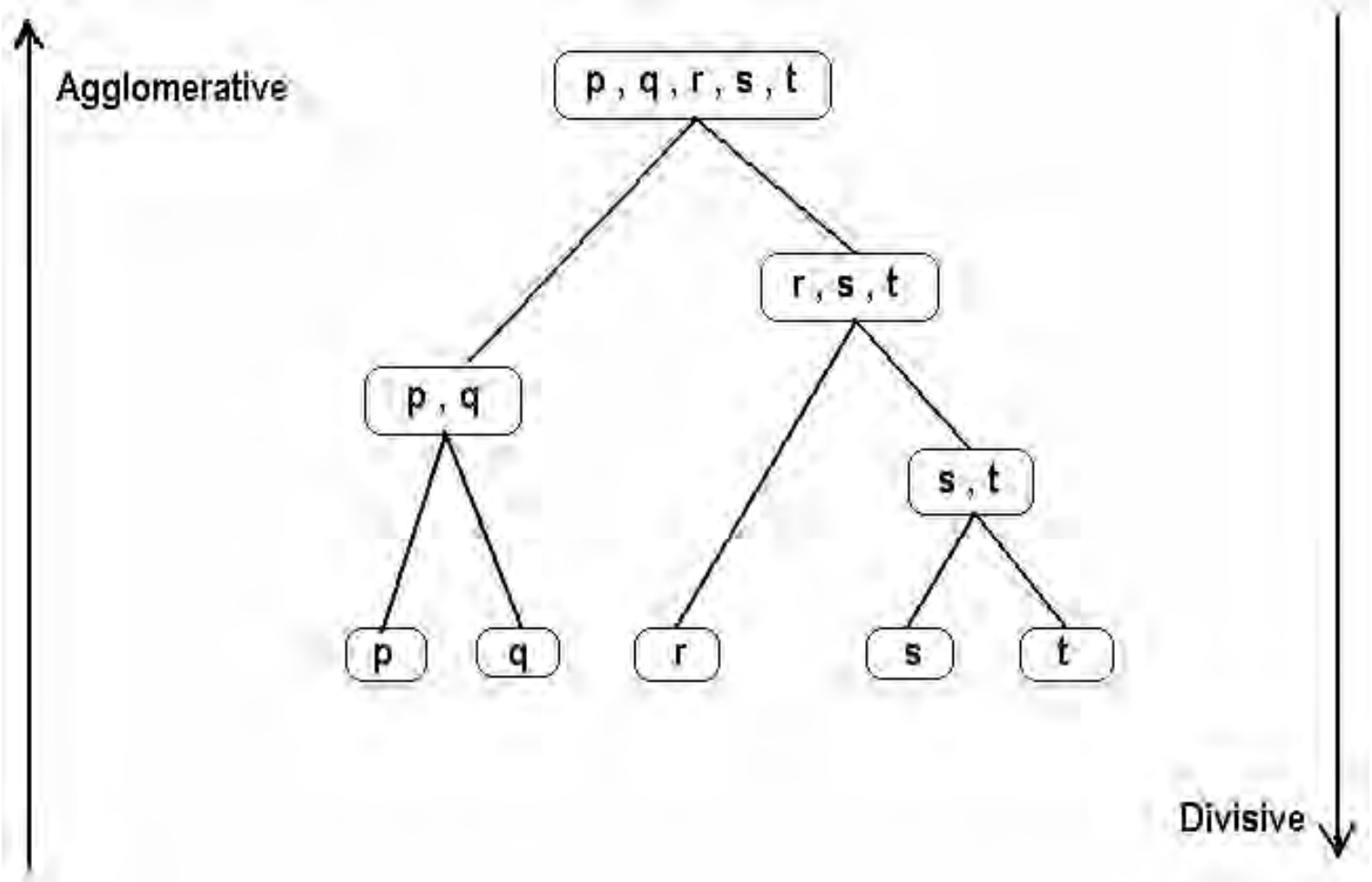




- il faut minimiser l'inertie intra-cluster et maximiser l'inertie inter-cluster

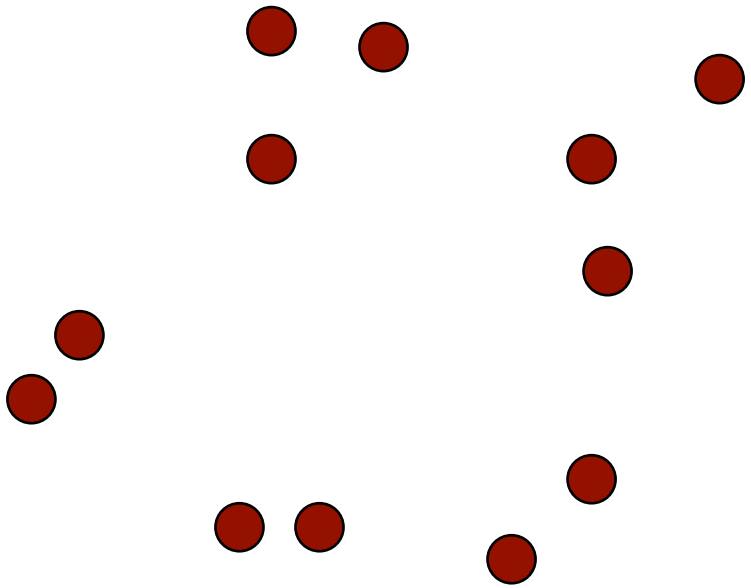
Classification Hiérarchique

Classification hiérarchique



Situation initiale

Un point ==cluster

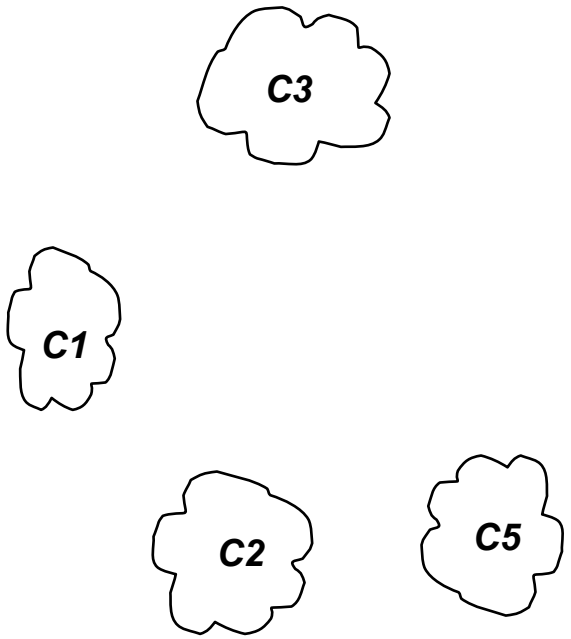


	p1	p2	p3	p4	p5	. . .
p1						
p2						
p3						
p4						
p5						
.						
.						
.						

Matrice de similarité

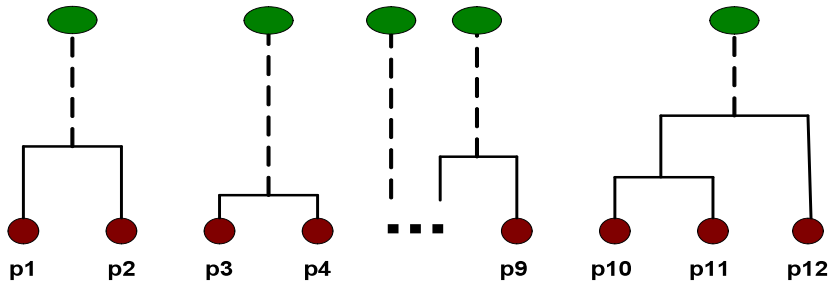
Situation intermédiaire

Après quelques itérations



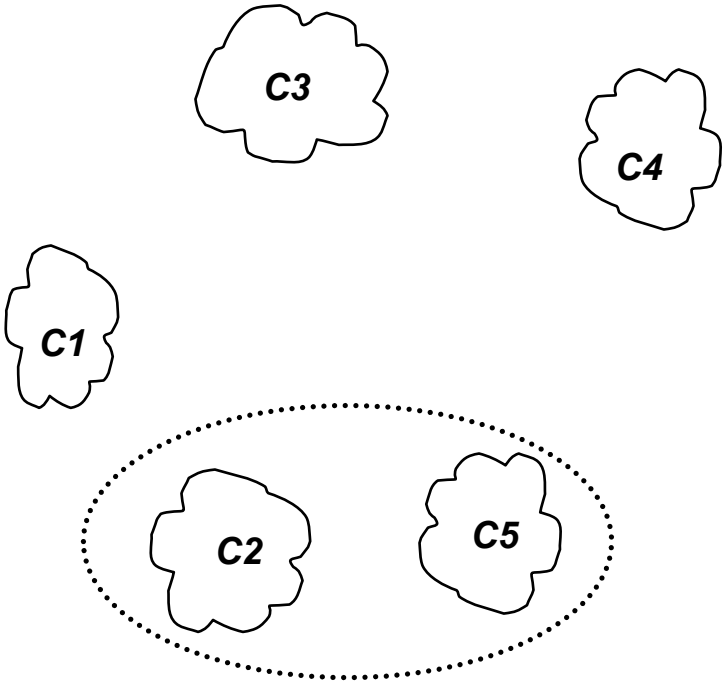
	C1	C2	C3	C4	C5
C1					
C2					
C3					
C4					
C5					

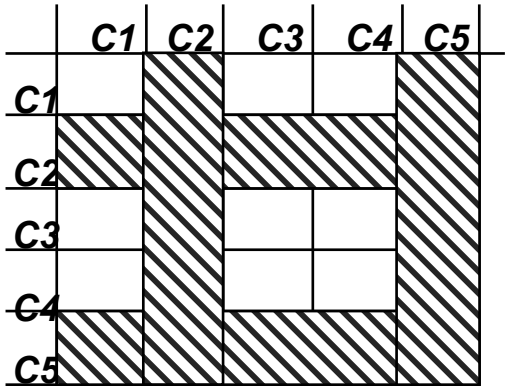
Matrice de similarité



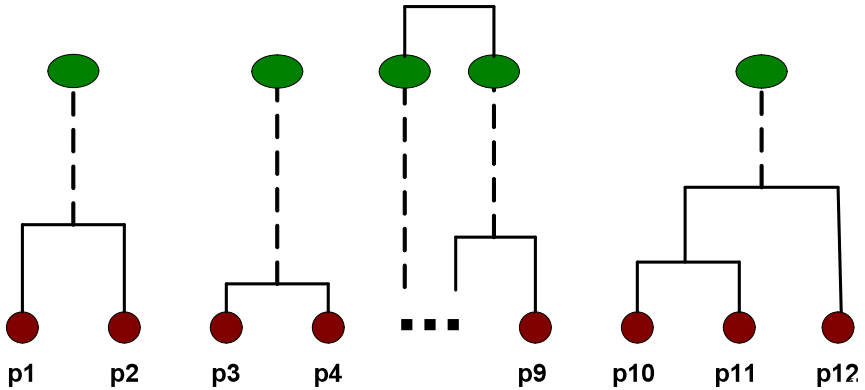
Situation intermédiaire

Fusionner C2 et C5 puis mise à jour de la matrice de similarité.



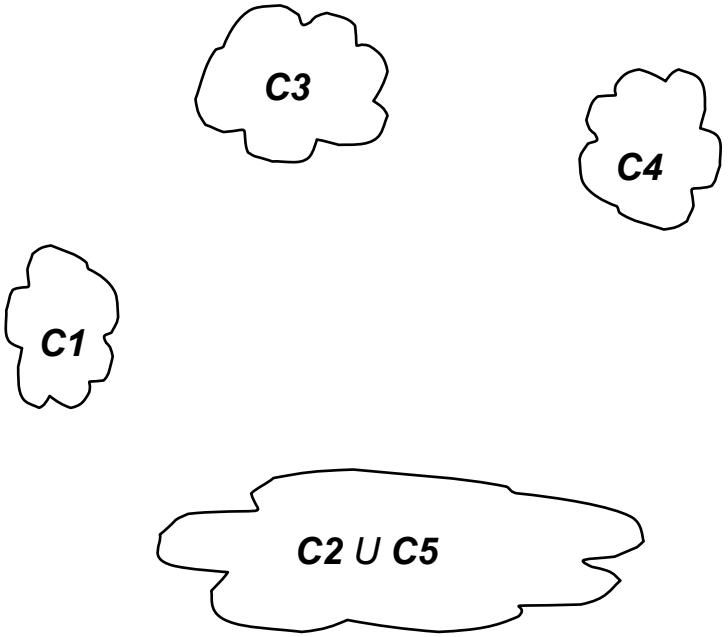


Matrice de similarité



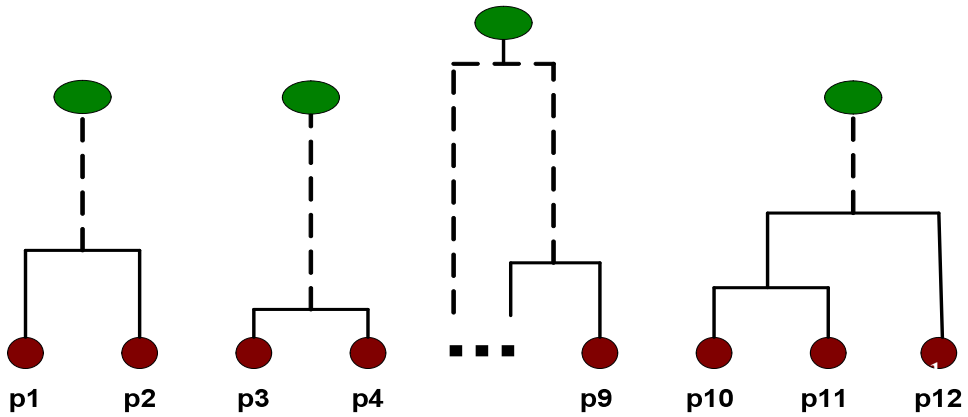
Après fusion

La question: “comment mettre à jour la matrice de similarité?”

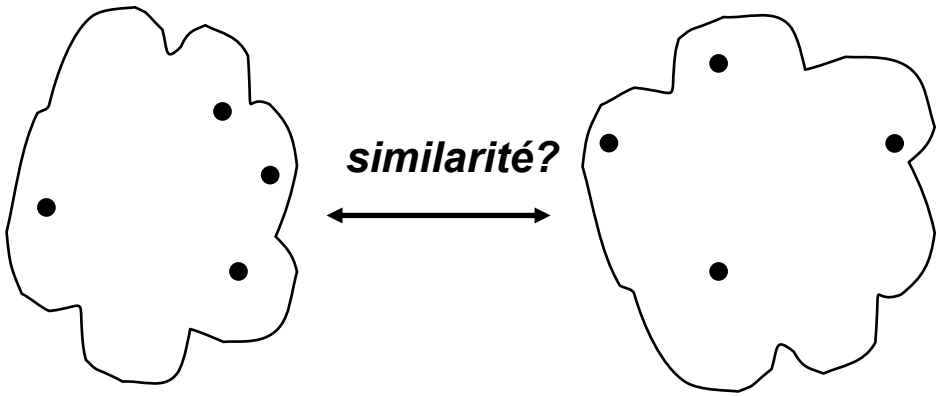


	C1	C2 U C5	C3	C4
C1		?		
C2 U C5	?	?	?	?
C3		?		
C4		?		

Matrice de similarité



Similarité inter-classe



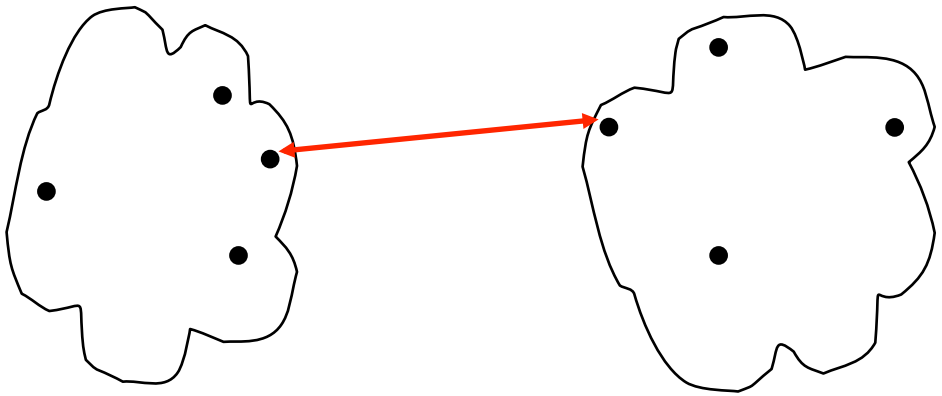
- *MIN*
- *MAX*
- *moyen*
- *Distance entre centres*
- *Ward*

	p1	p2	p3	p4	p5	. . .
p1						
p2						
p3						
p4						
p5						
.						

. ***Matrice de similarité***

.

Similarité inter-classe

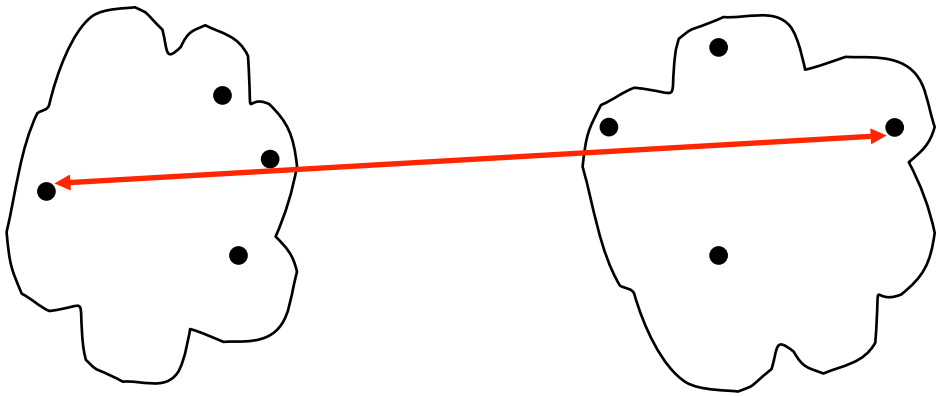


- *MIN*
- *MAX*
- *moyen*
- *Distance entre centres*
- *Ward*

	p1	p2	p3	p4	p5	. . .
p1						
p2						
p3						
p4						
p5						
.						
.						
.						

Matrice de similarité

Similarité inter-classe



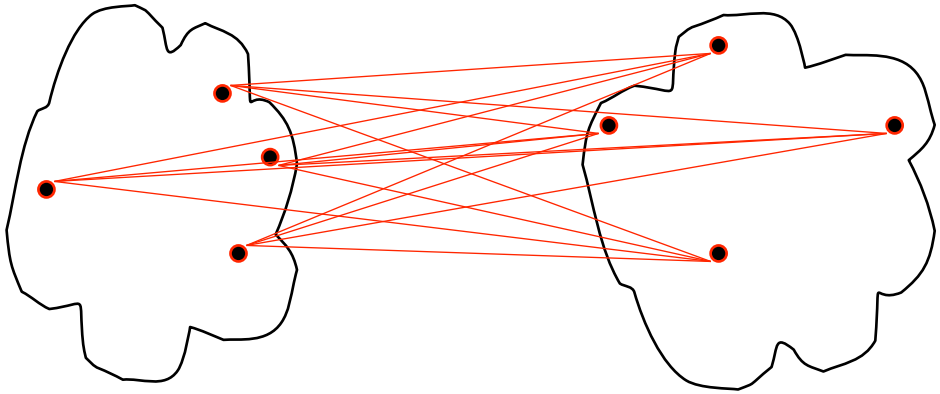
- *MIN*
- *MAX*
- *Moyen*
- *Distance entre centres*
- *Ward*

	p1	p2	p3	p4	p5	. . .
p1						
p2						
p3						
p4						
p5						
.						

. .

Matrice de similarité

Similarité inter-classe

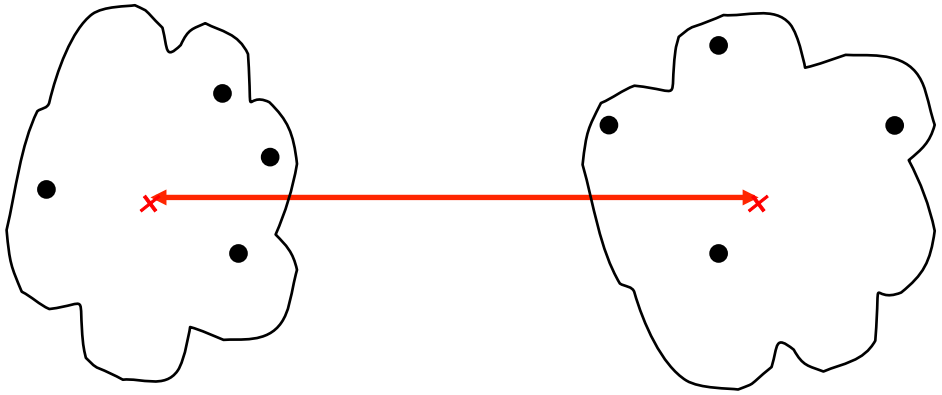


- *MIN*
- *MAX*
- *moyen*
- *Distance entre centres*
- *Ward*

	p1	p2	p3	p4	p5	. . .
p1						
p2						
p3						
p4						
p5						
.						
.						
.						

Matrice de similarité

Similarité inter-classe



	p1	p2	p3	p4	p5	. . .
p1						
p2						
p3						
p4						
p5						
.						

. **Matrice de similarité**

.

- *MIN*
- *MAX*
- *Average*
- *Distance entre centres*
- *Ward*

Indice de Ward

- Basé sur la perte d'inertie
- Moins sensible aux outliers
- A chaque itération, on agrège de manière à avoir un gain minimum d'inertie intra-classe : perte d'inertie interclasse due à cette agrégation

$$\frac{n_A n_B}{n_A + n_B} ||g_A - g_B||^2$$

Algorithme agglomératif

L'algorithme de base

1. Calculer la matrice de similarité
2. Affecter chaque donnée à un cluster
3. **Repeat**
4. fusionner les deux clusters les plus proches
5. Mise à jour de la matrice de similarité
6. **Until** trouver un seul cluster

K-means

Algorithmes à partionnement

- Construire une partition à ***k*** clusters d'une base ***A*** de ***n*** objets
- Les *k clusters doivent* optimiser le critère choisi

  - *k-means* (MacQueen'67): Chaque cluster est représenté par son centre
  - *k-medoids* or PAM (Partition around medoids) (Kaufman & Rousseeuw'87): Chaque cluster est représenté par un de ses objets

Quantification *vectorielle*

$\mathcal{D}$: *espace des données* $A \subset D \subset \Re^n$

$\mathcal{A}$: *ensemble d'apprentissage* $\mathcal{A} = \{\mathbf{x}_i, i = 1 : N\}$

$$\phi : \mathcal{D} \rightarrow \{1, 2, ..., p\}$$

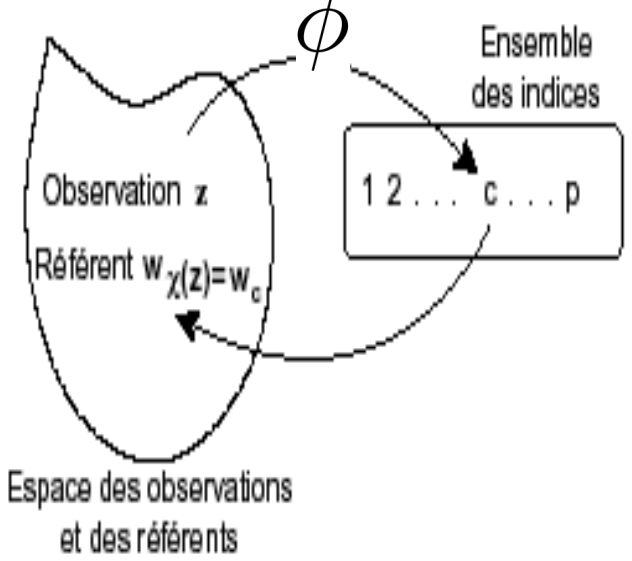
Réduire l'information de $\mathcal{D}$

- *En la résumant par un ensemble de* ***p référents***

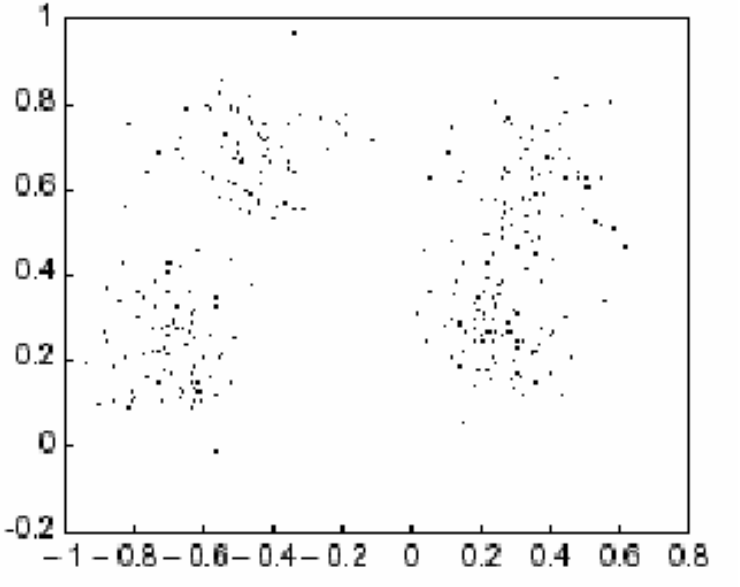
$$\mathcal{W} = \{\mathbf{w}_c, c = 1 : p\}$$

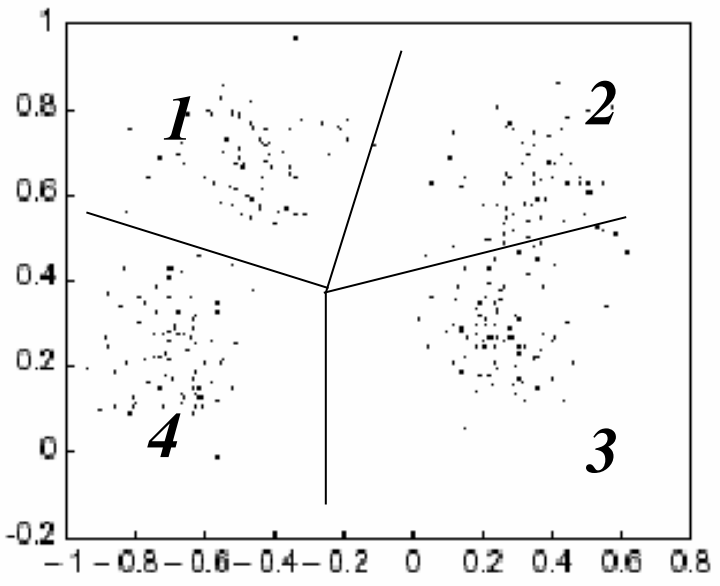
- *En réalisant une partition de* $\mathcal{D}$ *en* ***p*** *sous-ensembles par l 'intermédiaire d 'une* ***fonction d'affectation*** ϕ

$$P_c = \{\mathbf{x} \in \mathcal{D}, \phi(\mathbf{x}) = c\}$$



Quantification





$$\mathcal{A} = \{\mathbf{x}_i, i = 1 : N\}$$

$$A \subset D \subset \Re^n$$

Partition

$$P = \{P_1, P_2, P_3, P_4\}$$

$$\phi : \mathcal{D} \rightarrow \{1, 2, 3, 4\}$$

K-means
Version nuées dynamiques

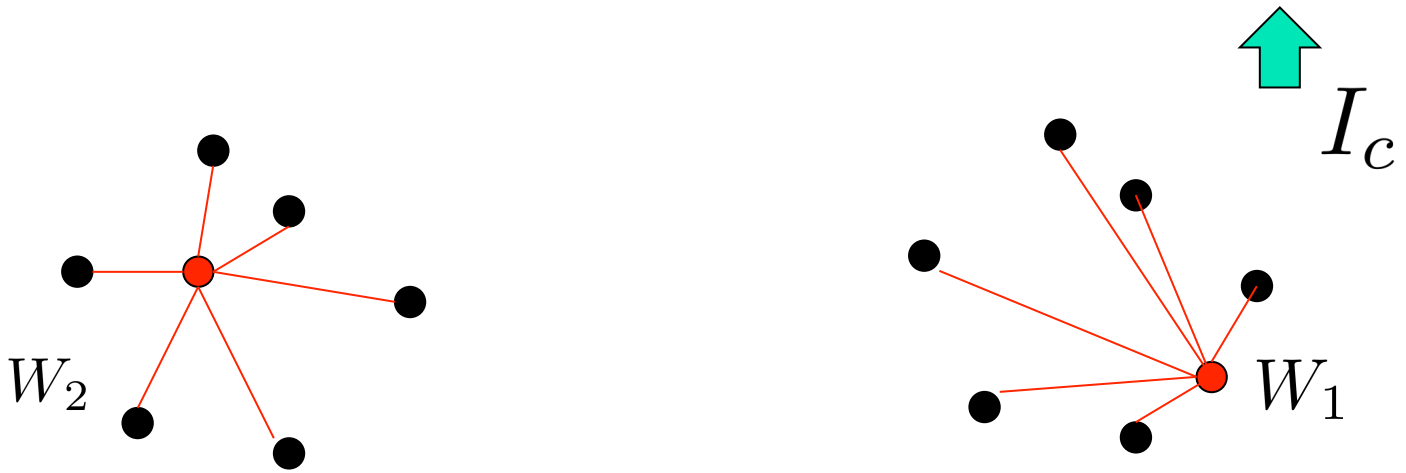
(Diday 1972, 1974)

- Chaque cluster est associé à un centre (prototype)
- Chaque donnée est affectée au centre le plus proche
- Nombre de clusters doit être fixé
- L'algorithme est simple

Méthode des k-moyennes

- *Minimiser la somme des inerties locales par rapport à χ et **W***

$$I(\mathcal{W},\phi) = \sum_{\mathbf{x}_i} ||\mathbf{x}_i - \mathbf{w}_{\phi(\mathbf{x}_i)}||^2 = \sum_{c} \sum_{\mathbf{x}_i \in P_c} ||\mathbf{x}_i - \mathbf{w}_c||^2$$



- *L'inertie I_c représente l'erreur de quantification obtenue si l'on remplace chaque observations de P_c par son référent **w_c***

•*Minimisation itérative qui fixe alternativement la partition (**c**)puis minimise l'inertie*

Phase d'affectation:

Pour un ensemble W de référents fixe, la minimisation de I par rapport à Φ s'obtient en affectant chaque observation $\mathbf{x}$ au référent $\mathbf{w}_c$ selon la nouvelle fonction d'affectation Φ

$$\phi(\mathbf{x}) = \arg\min_r ||\mathbf{x} - \mathbf{w}_r||^2$$

Phase de minimisation:

La partition Φ est fixée. La fonction $I(\mathcal{W}, \phi)$ est quadratique et convexe par rapport à W. Le minimum global est atteint pour

$$\frac{\partial I}{\partial W} = \left[\frac{\partial I}{\partial \mathbf{w}_1}, \frac{\partial I}{\partial \mathbf{w}_2}, \ldots, \frac{\partial I}{\partial \mathbf{w}_p}\right]^p = 0 \qquad \mathbf{w}_C = \frac{\sum_{\mathbf{x}_i \in P_c} \mathbf{x}_i}{|P_c|}$$

L’algorithme

L’algorithme de base

1. Sélectionner K centres
2. **Repeat**
3. Affecter chaque données au centre centre le plus proche
4. Mise à jour des centres
5. **Until** non changement des centres

Initialisation

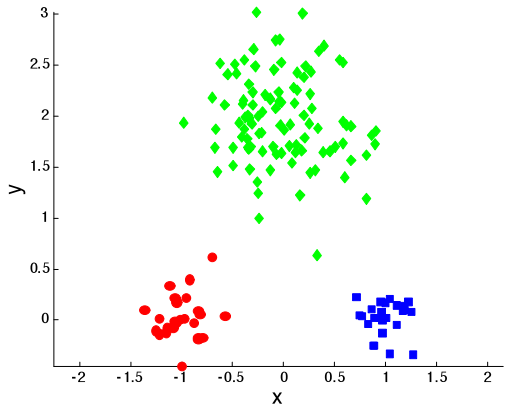
- *aléatoirement dans l'intervalle de définition des x_i*

- *aléatoirement dans l'ensemble des x_i*

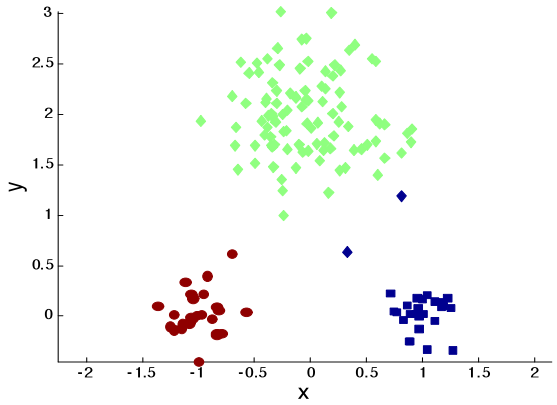
Des initialisations différentes peuvent mener à des clusters différents (problèmes de minima locaux)

- *méthode* ***générale pour obtenir des clusters "stables"*** ***=*** *formes fortes, on répète l'algorithme k fois*

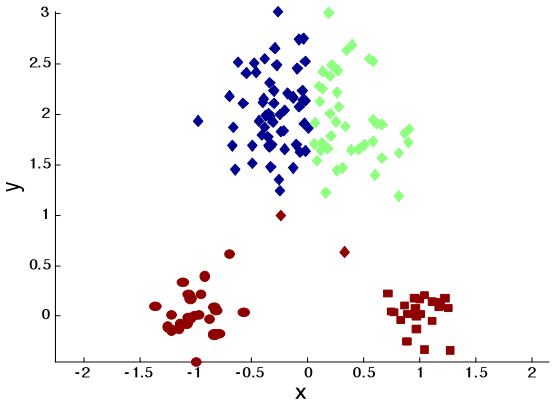
K-Means



Données originales

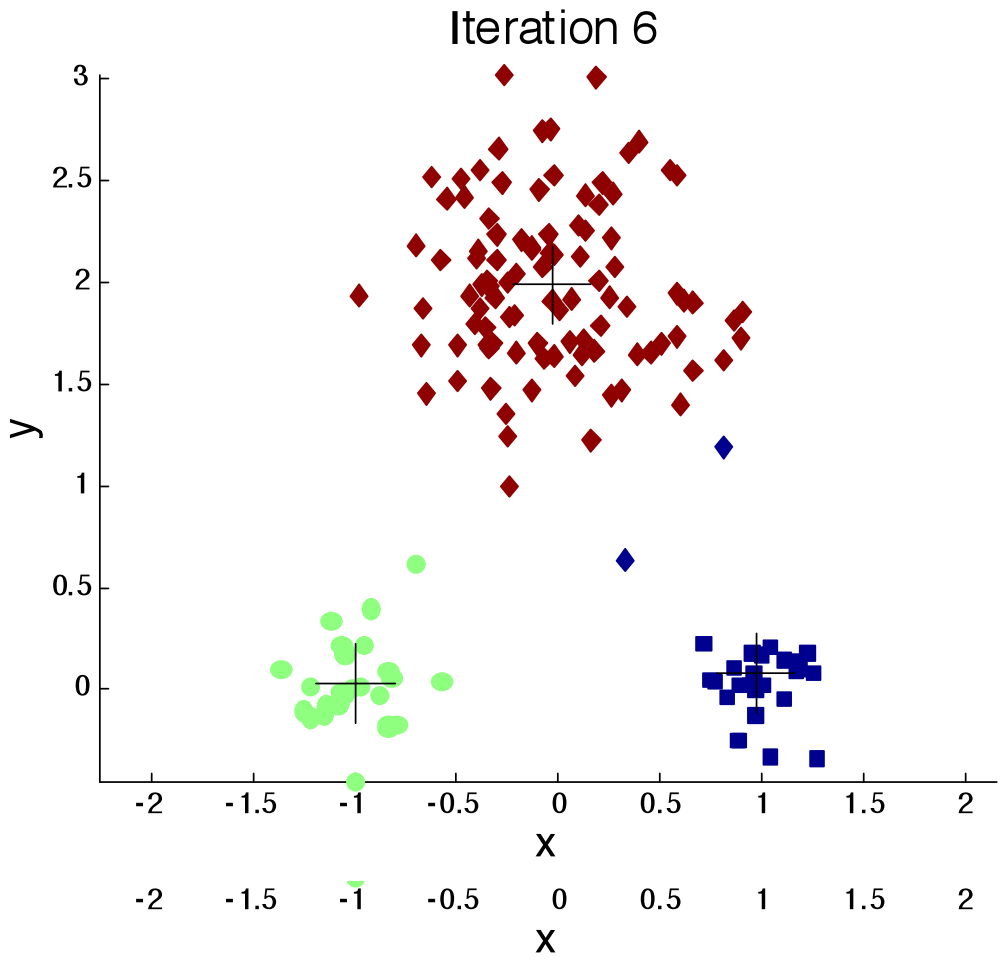


Solution optimale

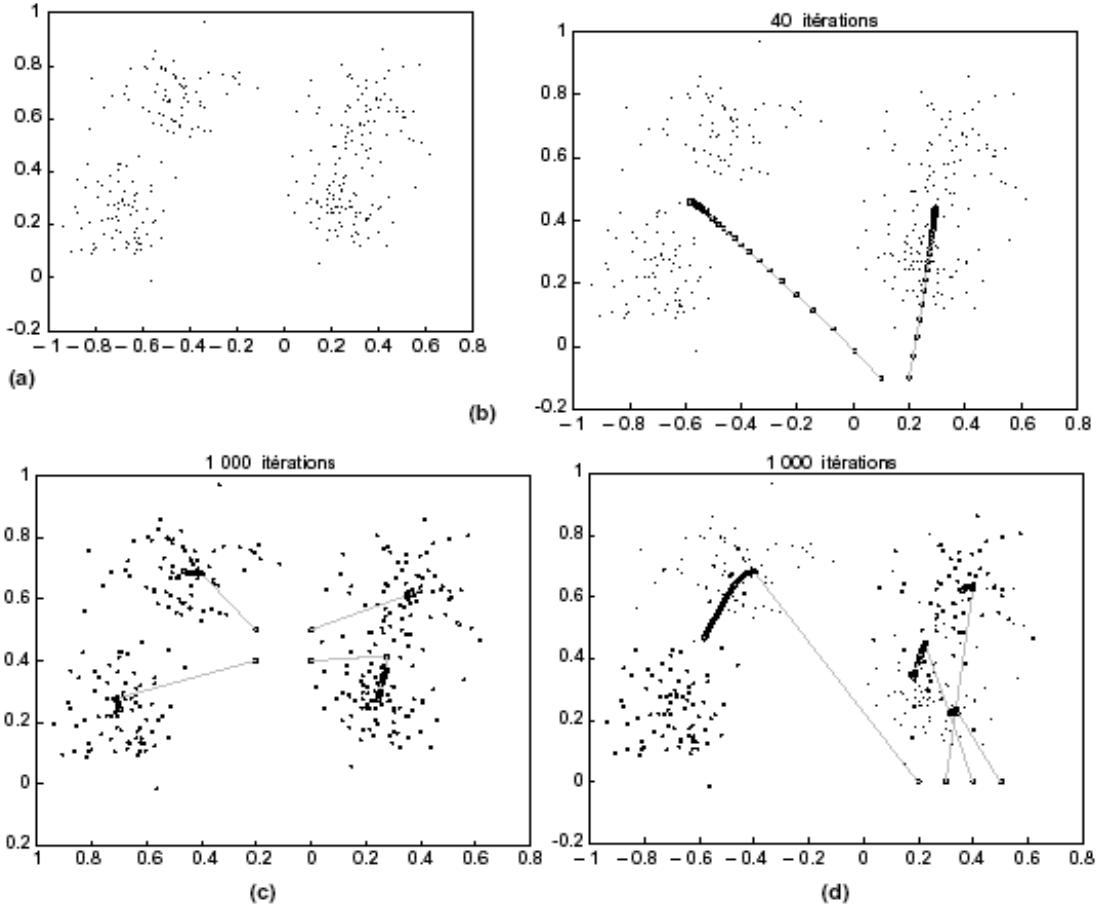


Autre solution

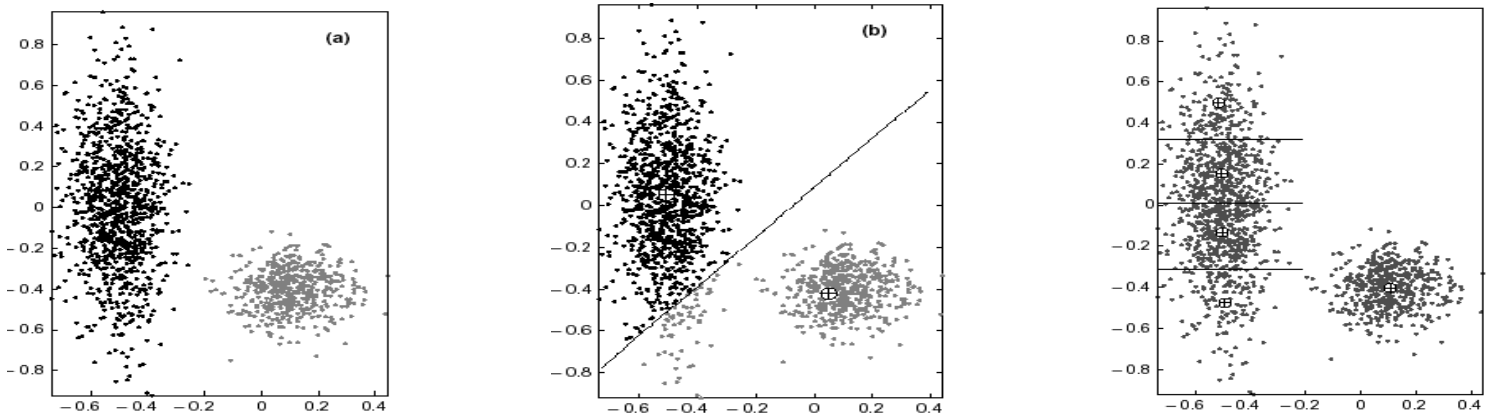
Importance de l'initialisation



Sensibilité aux conditions initiales



Comportement de l'algorithme des k-moyennes en fonction des densités sous-jacente



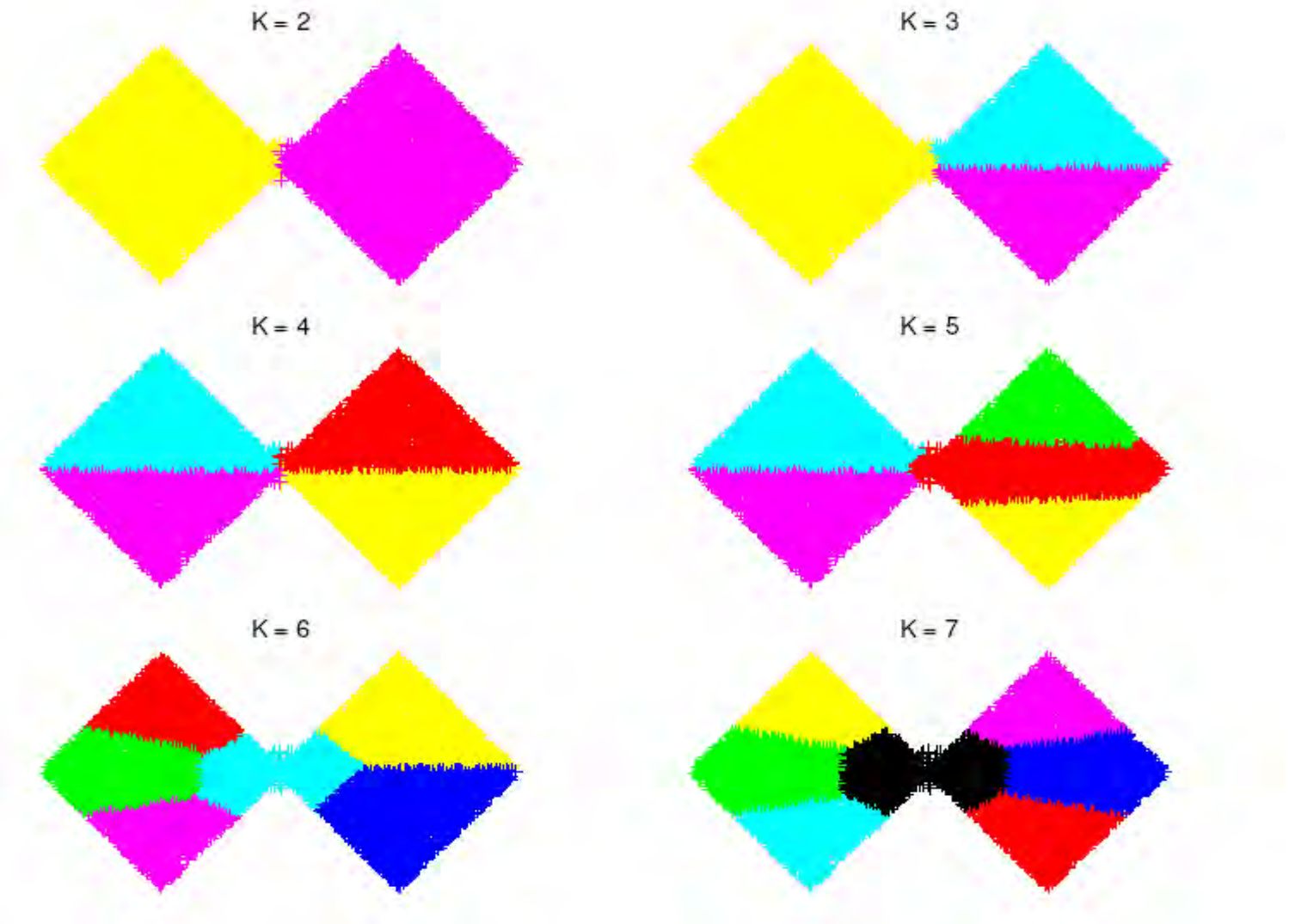
(a) (b) (c)

(a) Données simulées selon deux distributions gaussiennes de matrice de variance-covariance différentes

(b) référents et partition obtenue à la convergence avec deux référents

(c) avec cinq référents;

K-Means : exemple



Compression d'image: Quantification vectorielle

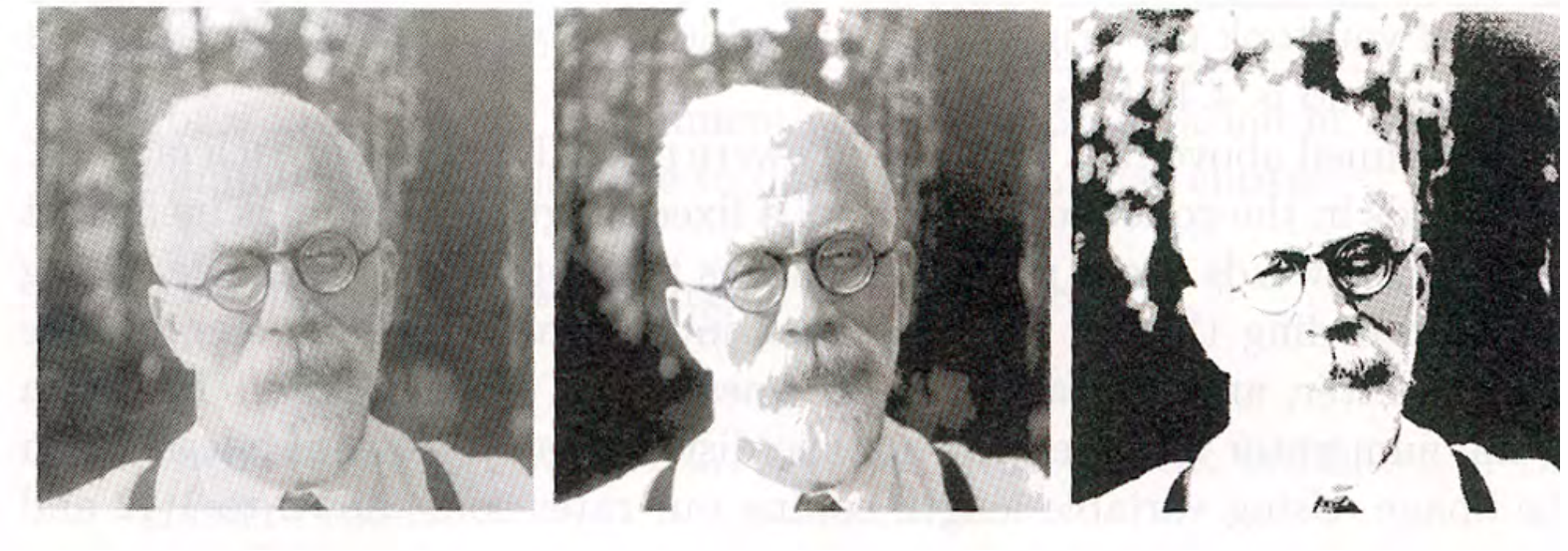


Image à gauche ***1024*1024 pixels***

256 niveaux de gris

8bits par pixel-

mémoire 1 mégabit

Image au centre *512*512 blocs de 2*2 pixels quantifiés en 200 référents*

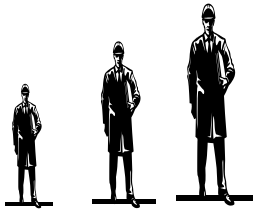
mémoire 0,239 mégabit

Image à droite *512*512 blocs de 2*2 pixels quantifiés en 4 référents*

mémoire 0,063 mégabit

Données qualitatives

Qualitatives / Catégorielles

Taille 

Sexe: 

Diabète: *Oui/NON*

Couleur : 

Questions:

- *Comment partitionner ces données ?*
- *Quelle distance utilisée ?*
- *Avoir des prototypes du même type que les données*

Variables qualitatives et codage

Taille: ***Petit,*** *Moyen, Grand* — *1 0 0*

Petit, ***Moyen,*** *Grand* — *1 1 0*

Petit, Moyen, ***Grand*** — *1 1 1*

Ordinale

Couleur : ***rouge****, vert, bleu* — *1 0 0*

rouge, ***vert****, bleu* — *0 1 0*

rouge, vert, ***bleu*** — *0 0 1*

Disjonctif

Données binaires

- une matrice de contingence

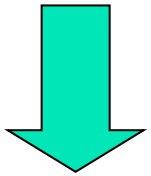
		Object j		
		1	0	sum
Object i	1	a	b	$a+b$
	0	c	d	$c+d$
	sum	$a+c$	$b+d$	p

$$H(i,j) = b + c$$

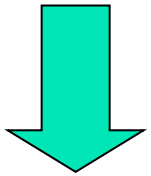
$$sim_{Jaccard}(i,j) = \frac{a}{a+b+c}$$

Distance de Hamming
w, x ∈ {0,1}^{d}

$$I(\mathcal{W},\phi) = \sum_{i=1}^{N} |\mathbf{x}_i - \mathbf{w}_{\phi(\mathbf{x}_i)}| = \sum_{i=1}^{N}\sum_{j=1}^{n} |x_i^j - w^j_{\phi(\mathbf{x}_i)}|$$



$$I(\mathcal{W},\phi) = \sum_{j=1}^{n}\left(\sum_{i=1}^{N}(1-x_i^j)w^j_{\phi(\mathbf{x}_i)} + \sum_{i=1}^{N} x_i^j(1-w^j_{\phi(\mathbf{x}_i)})\right)$$

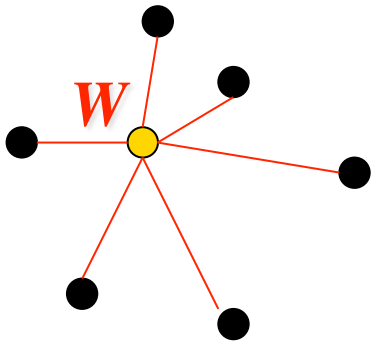


$$I(\mathcal{W},\phi) = \sum_{j=1}^{n}\left(w^j_{\phi(\mathbf{x}_i)}\Gamma_0^j + (1-w^j_{\phi(\mathbf{x}_i)})\Gamma_1^j\right)$$

Centre médian

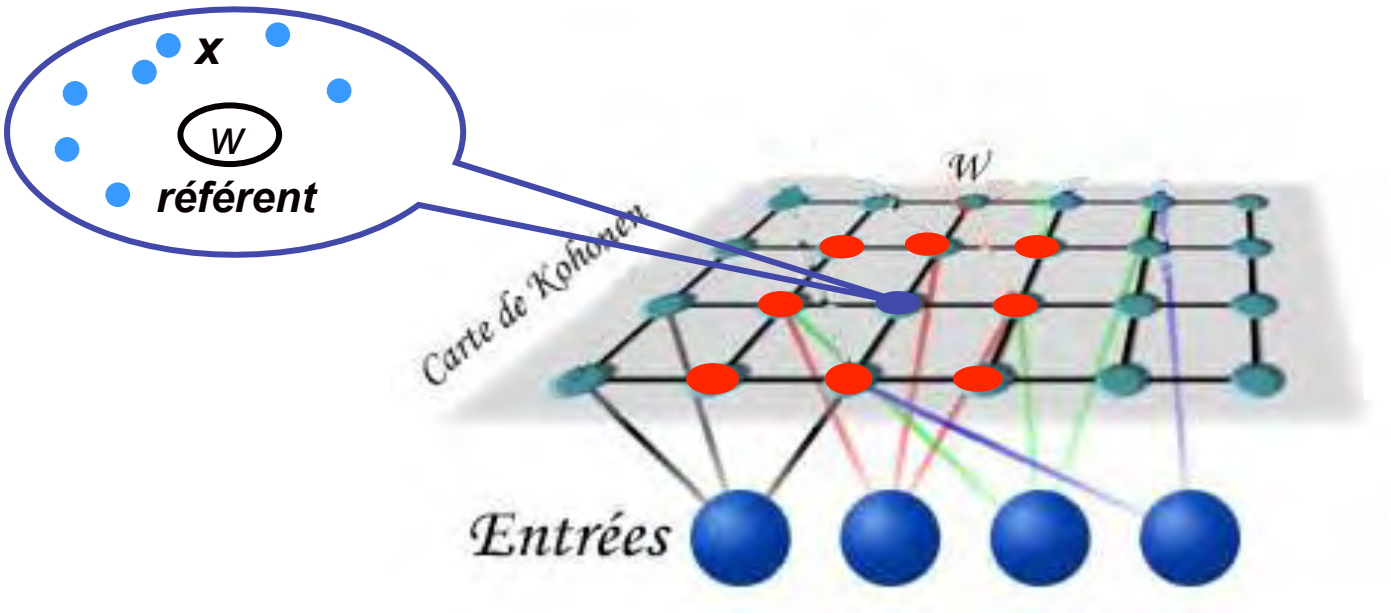
1 1 1 1 1 0 0 1 1 1 0 0
1 1 0 1 1 1 1 1 0 0 0 0
1 1 1 1 0 0 0 1 1 1 1 1
1 1 1 1 1 1 0 1 1 0 0 0
1 1 1 1 1 1 0 1 1 1 1 0
1 0 0 1 1 1 0 1 1 1 0 0

1 1 1 1 0 1 0 1 0 1 0 0



SOM : Self-Organizing Maps

Cartes topologiques



$K(\delta(\ c_1, c_2)\ )$

Fonction noyau

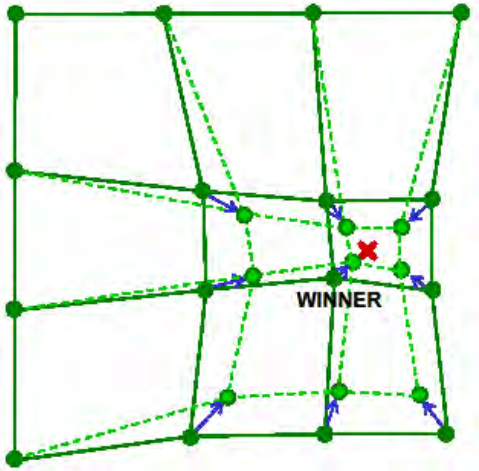
$\delta(c_1, c_2)$ **le court chemin sur le graphe**

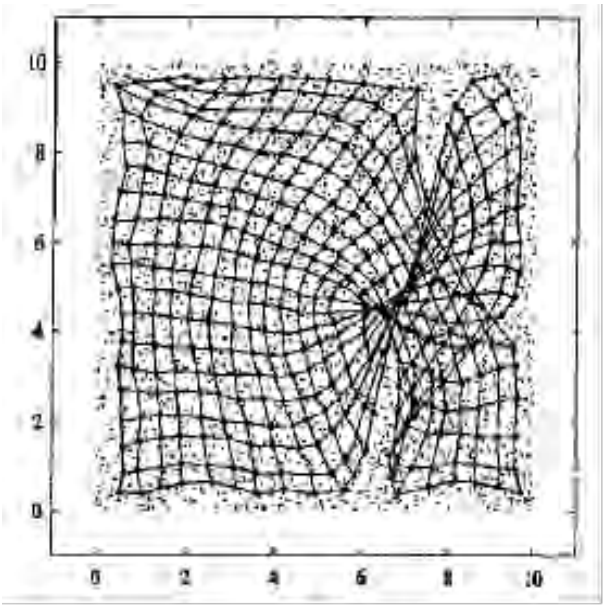
- ***Il s'agit d'un espace discret (C) de faible dimension à des fins de visualisation (1-D, 2-D).***
- ***C ensemble de cellules (nœuds, neurones) connectées par une structure de graphe non-orienté muni d'une distance discrète δ sur C et d'une structure de voisinage***

- *Chaque cellule* ***c*** *de C est associé à un vecteur référent* w_c *de l'espace des données D*

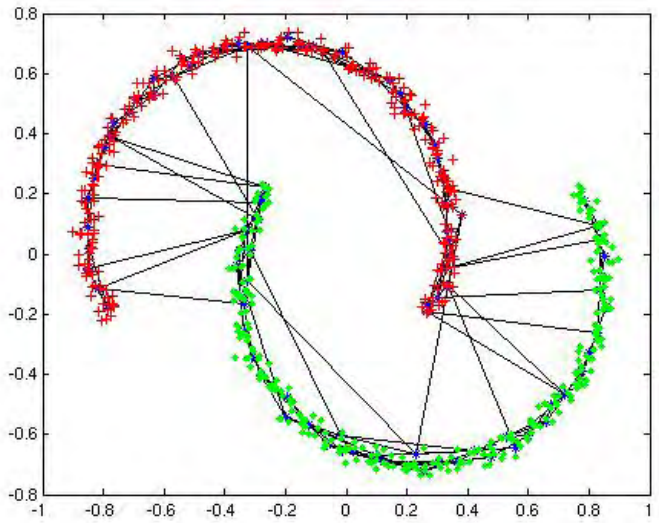
- *L'apprentissage approxime la densité sous-jacente des données tout en cherchant à respecter une contrainte de conservation de la topologie de la carte C*

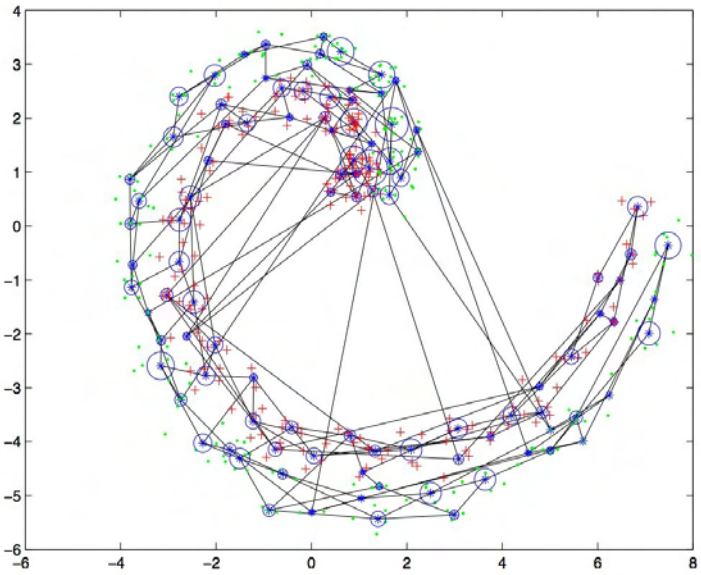
- *Deux cellules* ***c*** *et* ***r*** *« voisins » par rapport à la topologie discrète de la carte C sont associés à deux vecteurs référents* $\boldsymbol{w_c}$ *et* $\boldsymbol{w_r}$ *proches dans l'espace des données D.*

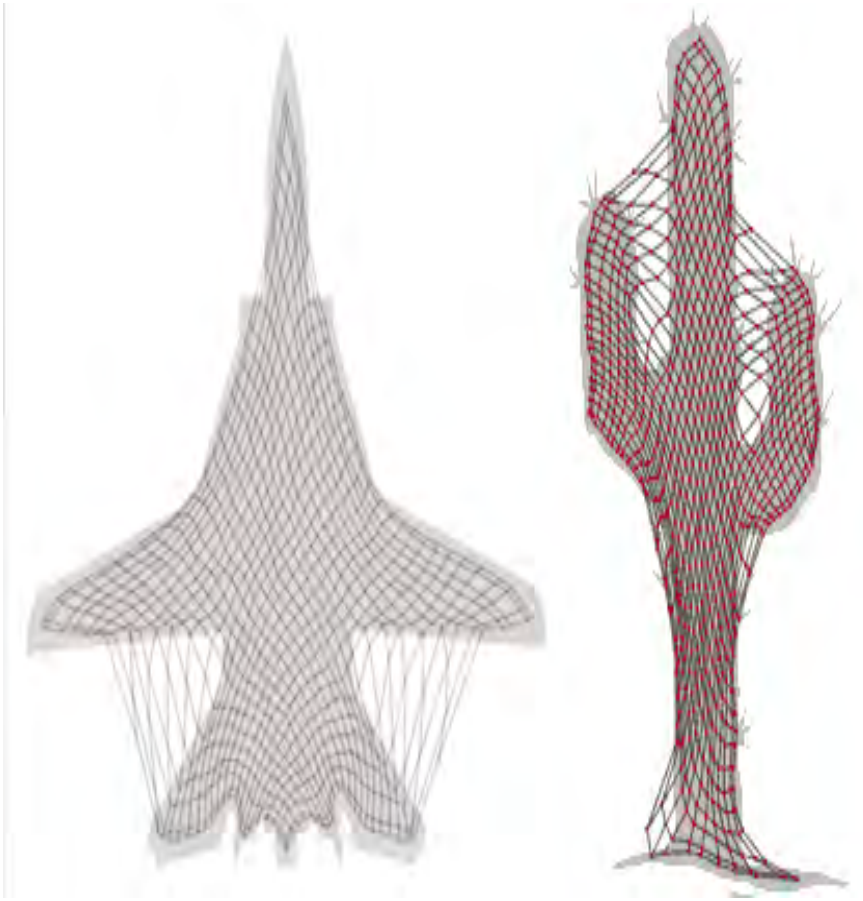


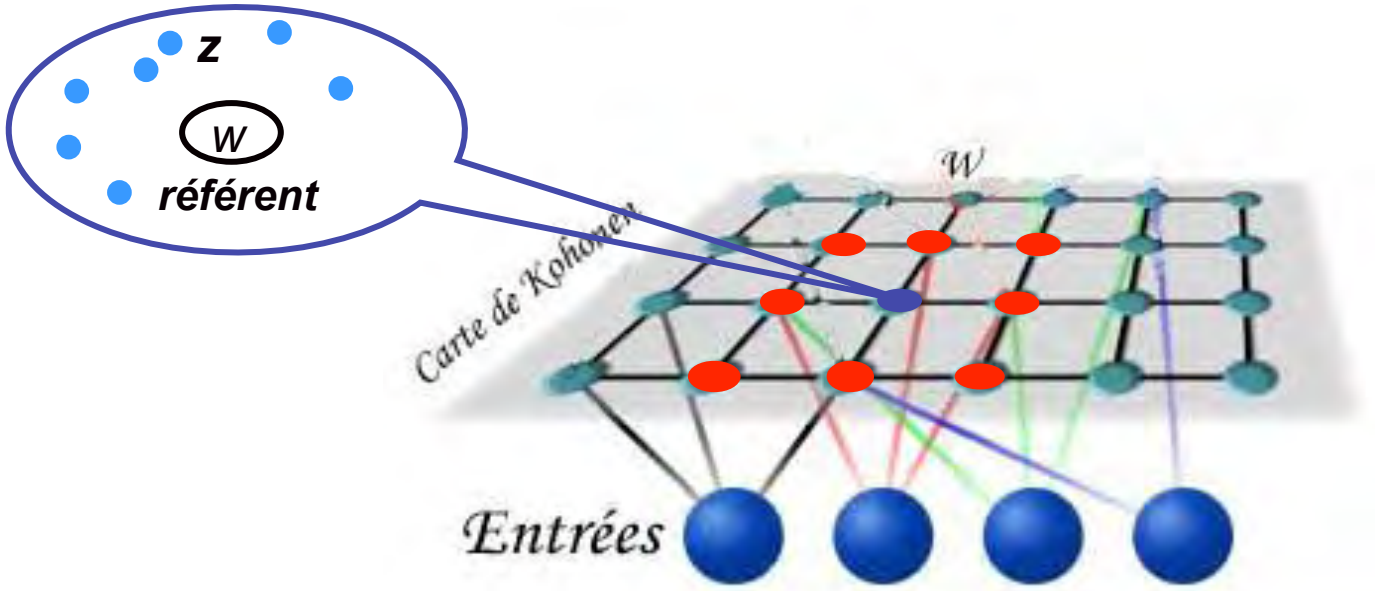


Illustrations/visualisation









$K(\delta(\ c_1, c_2)\ )$

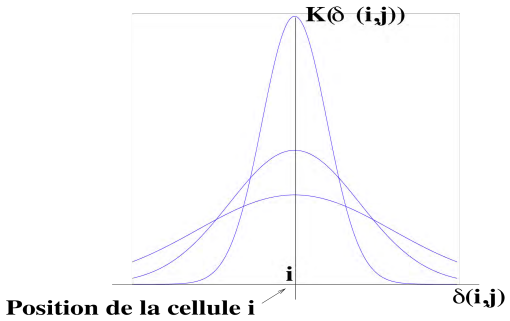
Fonction noyau

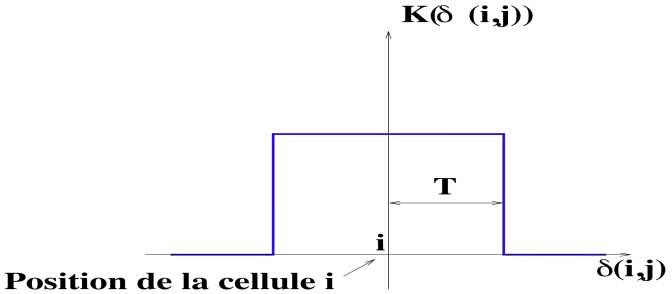
$$J_{som}^{T}(W,\phi) = \sum_{\mathbf{x}_i \in \mathcal{A}} \sum_{c \in C} K^{T}\left(\delta\left(c, \phi(\mathbf{x}_i)\right)\right) \left|\left|\mathbf{x}_i - \mathbf{w}_c\right|\right|^2$$

Cartes topologiques

Fonction de coût

$$J_{som}^{T}(W,\phi)=\sum_{\mathbf{x}_i\in\mathcal{A}}\sum_{c\in C}K^{T}\left(\delta\left(c,\phi(\mathbf{x}_i)\right)\right)\left\|\mathbf{x}_i-\mathbf{w}_c\right\|^2$$





$$K^{T}\left(\delta\left(c,\phi(\mathbf{x}_i)\right)\right)=\exp\left(-0.5\frac{d}{T}\right)$$

$$V_c^{T}=\{r\in \mathrm{C}/K^{T}(\delta(c,r))>\alpha\}.$$

La valeur de T détermine la taille du voisinage

Algorithmes des nuées dynamiques

Minimisation itérative de $J^T_{som}(\mathcal{W}, \phi)$ *pour un paramètre T fixé:*

Phase d'affectation:

L'ensemble $\mathcal{W}$ *des référents est fixé, la minimisation s'obtient en affectant chaque observation* **x** *au référent* $\mathbf{w}_c$ *selon la nouvelle fonction d'affectation* Φ^T

$$\phi^T(\mathbf{x}) = \arg\min_{r \in C} \left(\sum_{c \in C} K^T(\delta(c,r)) ||\mathbf{x}_i - \mathbf{w}_c||^2 \right)$$

Phase de minimisation:

La partition Φ^T *est fixée. La fonction* $J^T_{som}(\mathcal{W}, \phi)$ ***est minimisée par rapport*** *à l'ensemble des référents* $\mathcal{W}$*. La fonction étant convexe par rapport aux paramètres le minimum global est atteint pour*

$$\mathbf{w}_c^T = \frac{\sum_{r \in C} K(\delta(c,r)) \mathbf{Z}_r}{\sum_{r \in C} K(\delta(c,r)) n_r}$$

$\mathbf{Z}_r$ *représente la somme de toutes les observations affectées à la cellule r*

$\mathbf{n}_r$ *le nombre de ces observations*

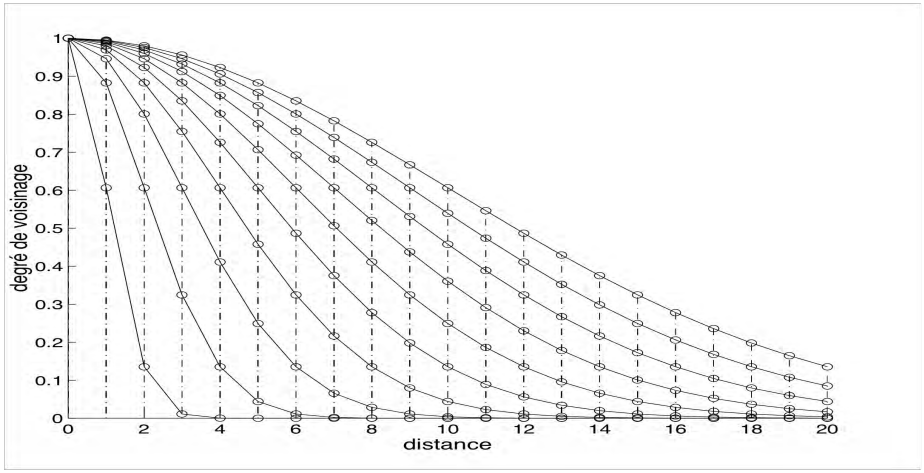
Algorithme des cartes topologiques

• La minimisation à T fixée est répétée un certain nombre de fois en faisant décroître la valeur de T.

• L'ordre apparaît pour les grandes valeurs de T, la carte se déploie alors pour recouvrir les données et faire diminuer la variance intra.

Les paramètres déterminants de la minimisation sont

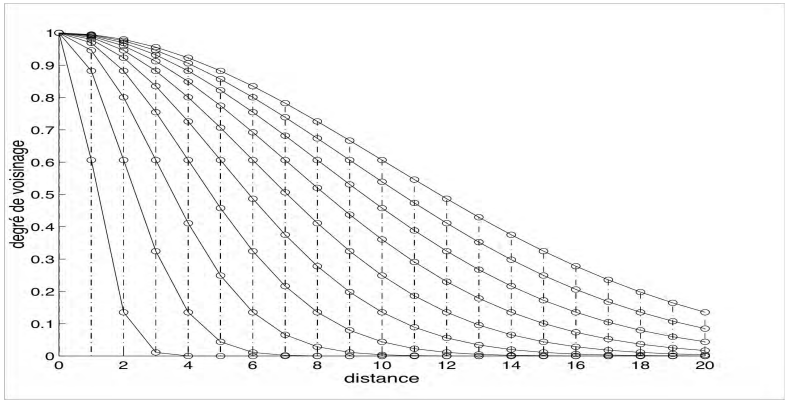
- *L'intervalle de variation de T, la valeur initiale T^{max} et la valeur finale T^{min}*
- *Le nombre de fois où l'étape itérative est effectuée*
- *La manière dont le paramètre décroit dans l'intervalle [T^{max}, T^{min}]*



$$T = T_{max}\left(\frac{T_{min}}{T_{max}}\right)^{\frac{t}{N_{iter}-1}}$$

Interprétation topologique

$$\mathcal{J}_{som}^{T}(\chi, \mathcal{W}) = \left[\sum_{c} \sum_{r \neq c} \sum_{\mathbf{x}_i \in P_r} K_T(\delta(c,r)) \|\mathbf{x}_i - \mathbf{w}_r\|^2 \right] + \left[K^T(\delta(c,c)) \sum_{c} \sum_{\mathbf{z}_i \in P_c} \|\mathbf{x}_i - \mathbf{w}_c\|^2 \right]$$



K-means

L'algorithme de Kohonen: une version stochastique

• *Phase de minimisation: ϕ constante, faire décroître la valeur de J_{som}^{T} par une méthode de gradient*

$$\mathbf{w}_c^t = \mathbf{w}_c^{t-1} - \mu^t \frac{\partial J_{som}^T}{\partial \mathbf{w}_c^{t-1}}$$

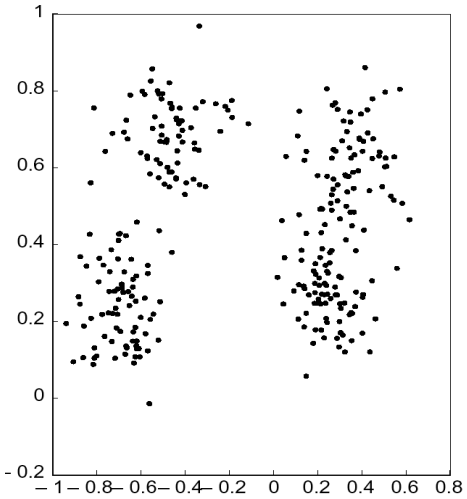
• *Utiliser un gradient stochastique en effectuant une itération de l'algorithme pour chaque forme.*

$$\mathbf{w}_c^t = \mathbf{w}_c^{t-1} - \mu^t K^T \left(\delta\left(c, \phi_t\left(\mathbf{x}_i\right)\right)\left(\mathbf{w}_c^{t-1} - \mathbf{x}_i\right)\right.$$

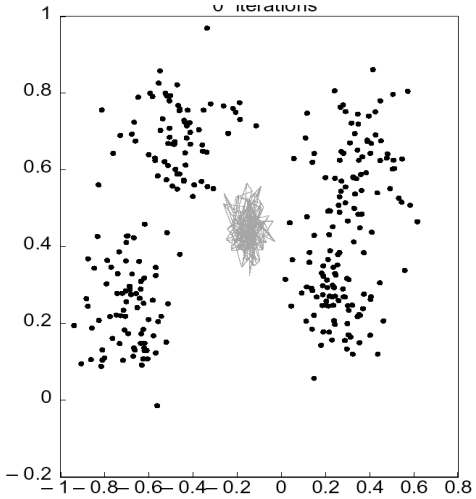
• *Pour chaque observation l'ensemble des référents est recalculé en fonction du la cellule gagnante.*

• *Modifier $\mathbf{w}_c$ entraine la modification de tous les référents du voisinage*

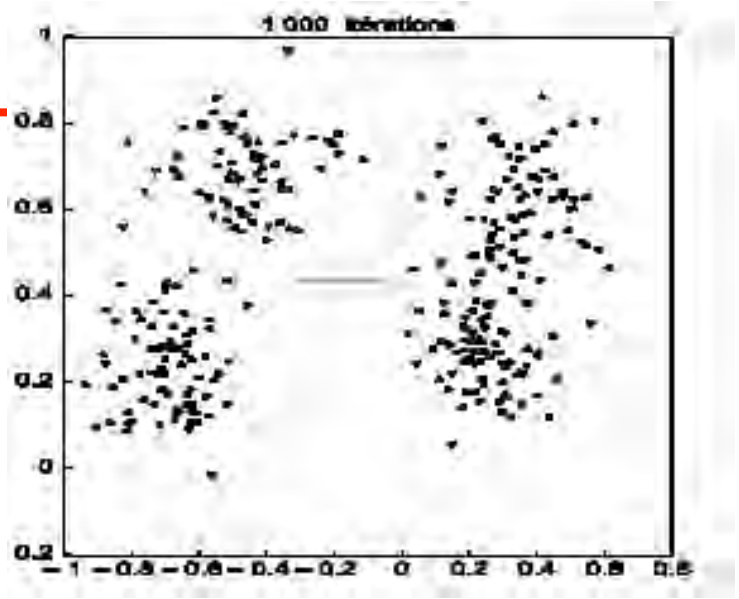
Exemple: Données simulées



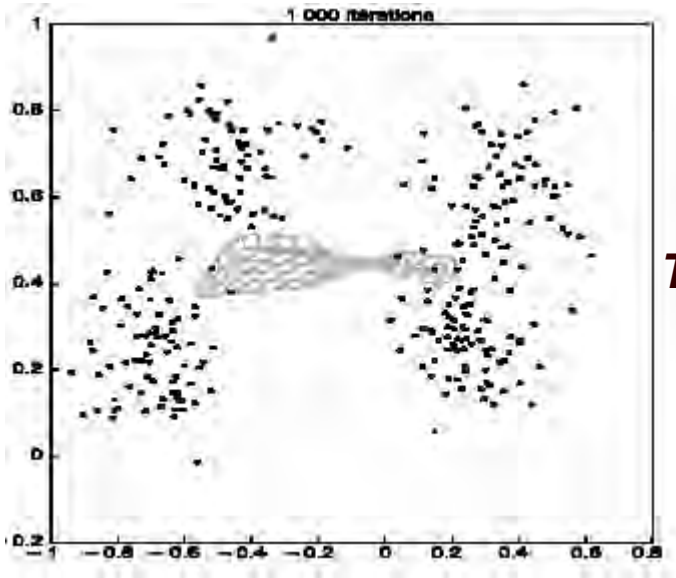
Données



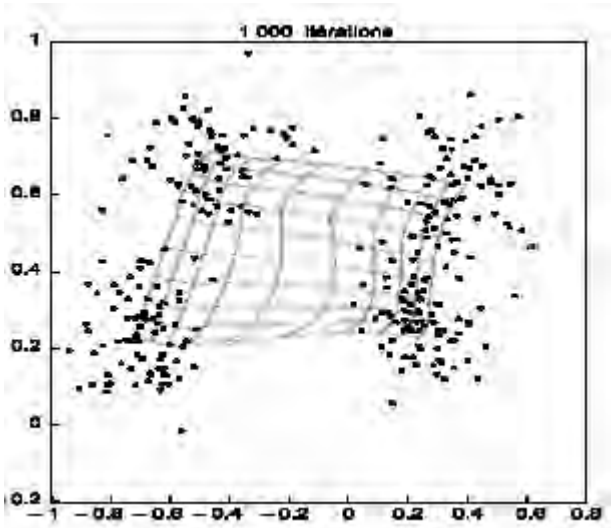
Initialisation



T=10

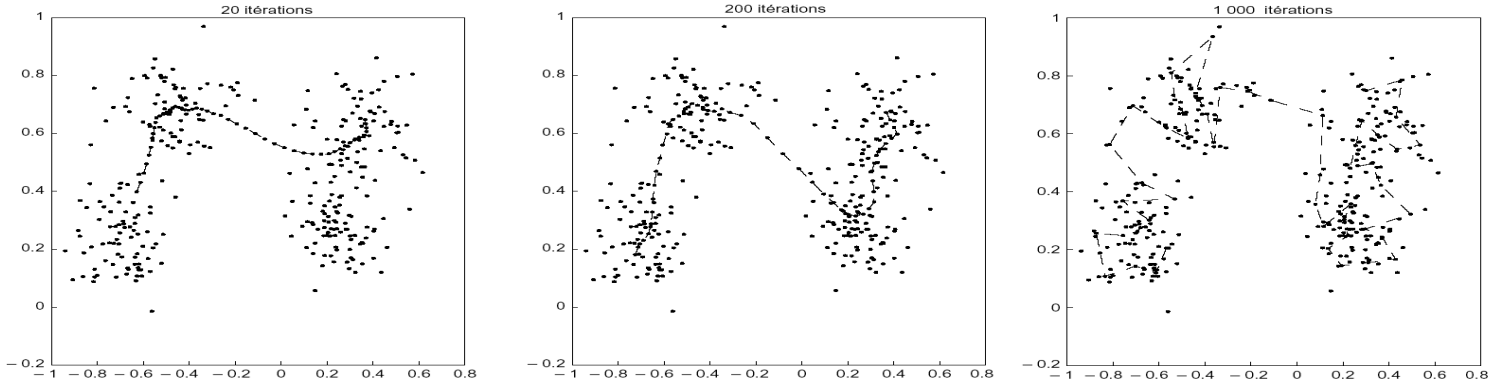


T=3

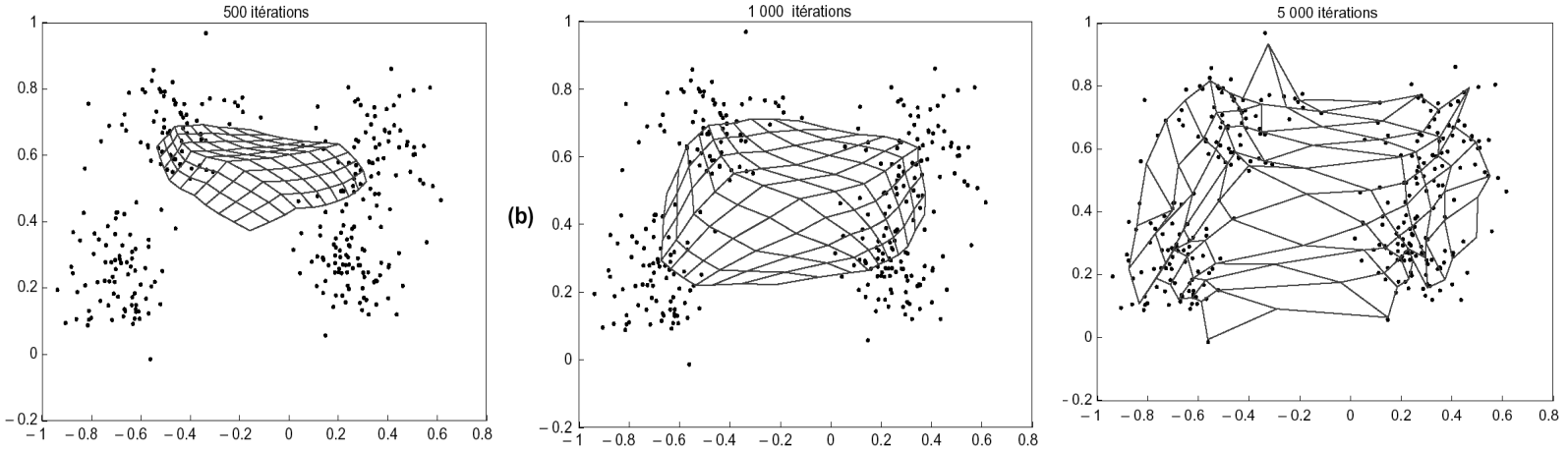


T=1

Evolution de l'apprentissage: données simulées 4 gaussiennes



Carte 1-D 50 neurones après 20, 200, 1000 itérations



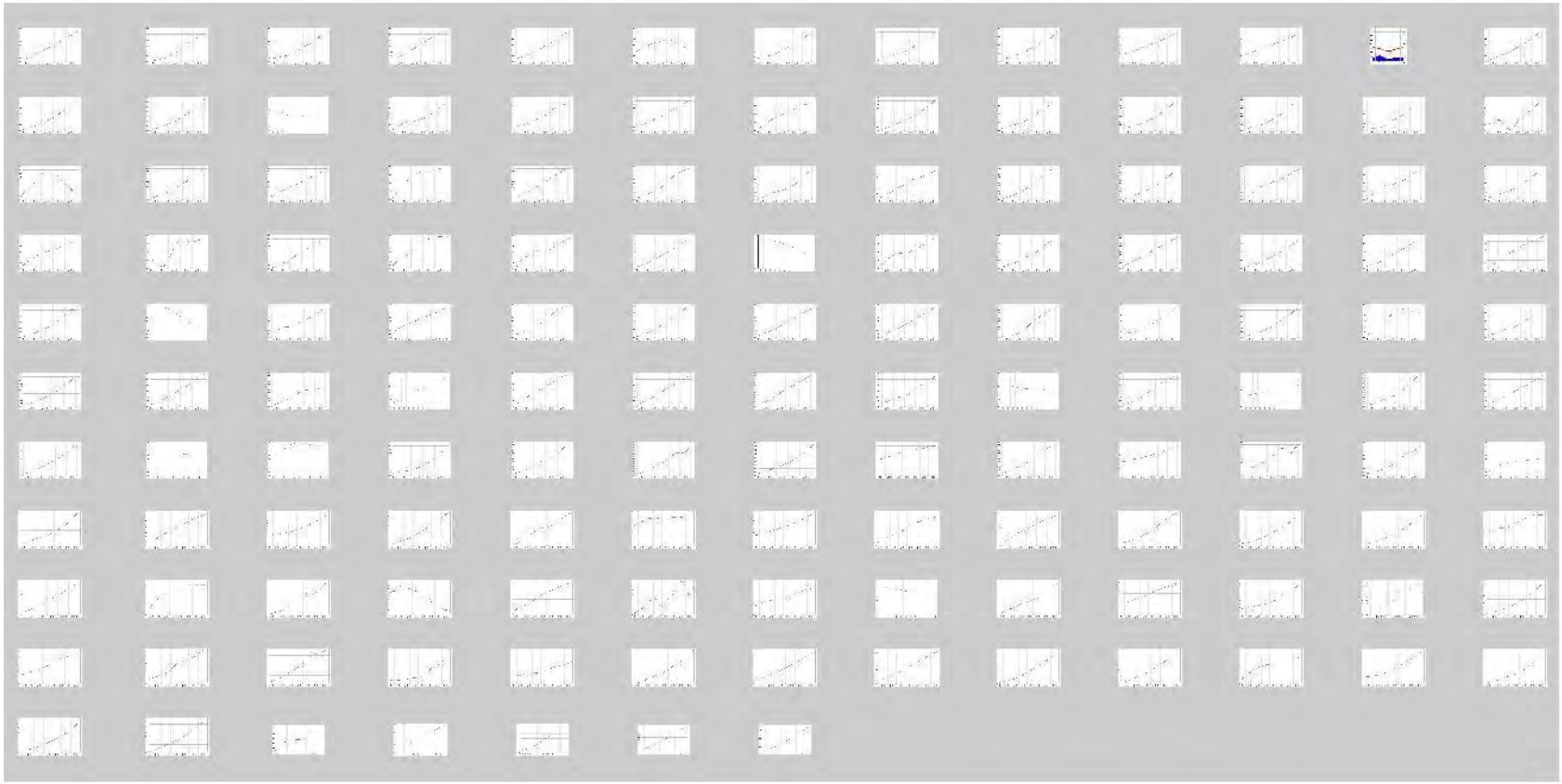
*Carte 2-D 10*10 neurones après 500, 1000, 5000 itérations*

Wikipedia

Wikipedia

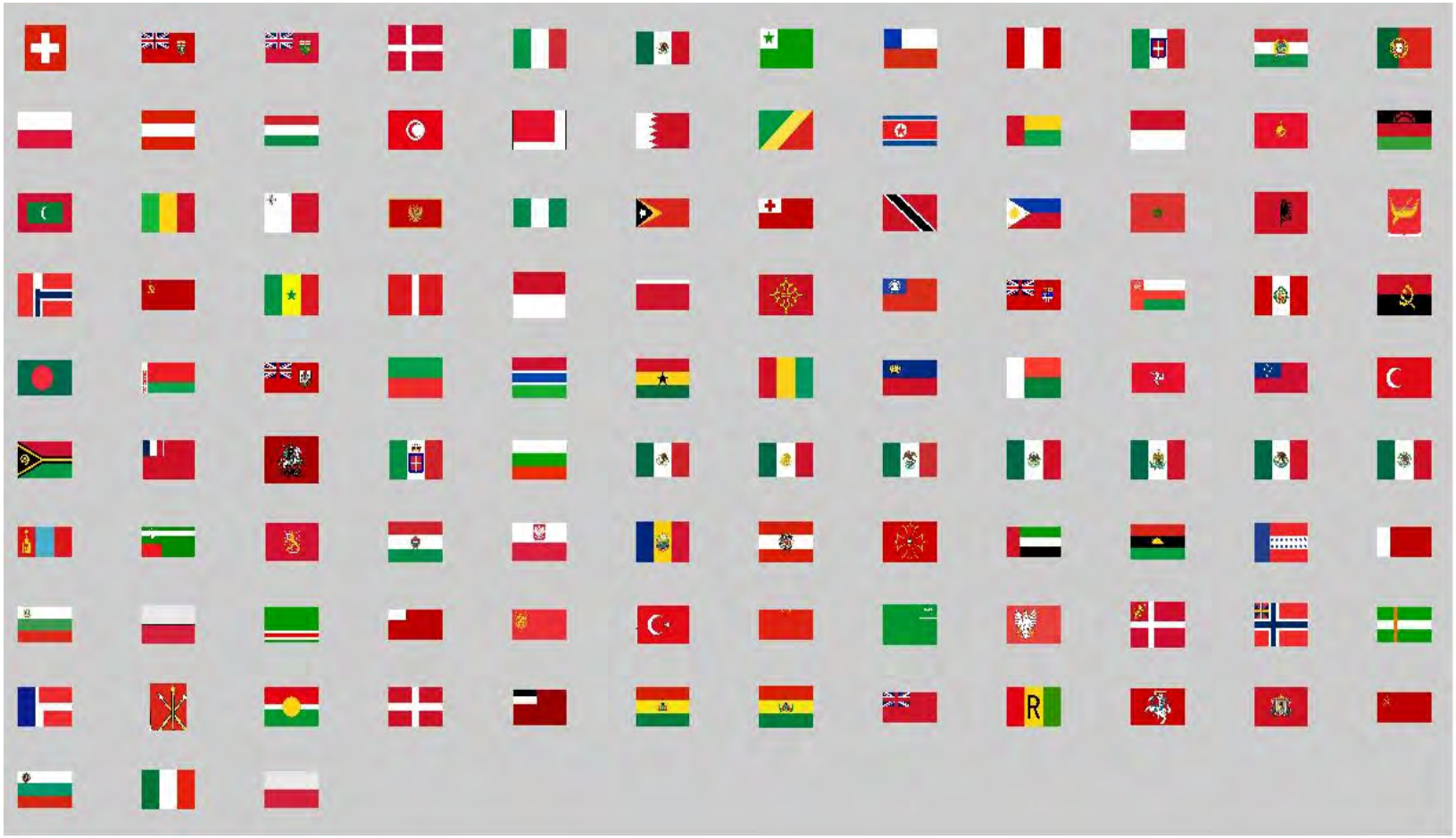

Cellule1
Cellule 91
Cellule 169
AUVERGNE
AQUITAINE
RÉGION

Résultats wikipedia



Cellule 91

Résultats wikipedia

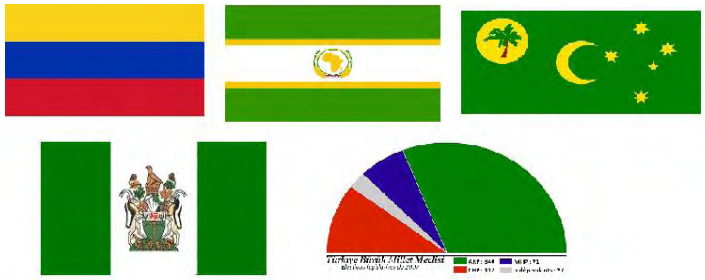


Cellule 169

Résultats wikipedia

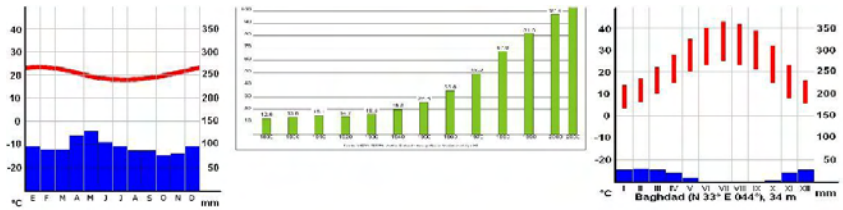
Cluster 7

Cluster 40



Cluster 41

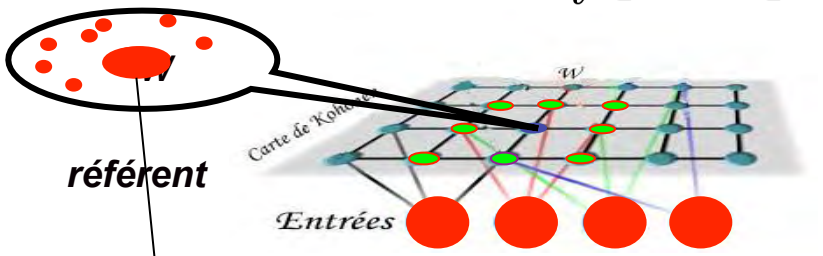
Cluster 42



Données Binaires

Le modèle BTM

$$J_{bin}^{T}(\mathcal{W},\phi)=\sum_{\mathbf{x}_i\in\mathcal{A}}\sum_{c\in C}K^{T}(\delta(c,\phi(\mathbf{x}_i)))\left|\mathbf{x}_i-\mathbf{w}_c\right|$$



Distance de Hamming

H(0101,0111)=1

Centre médian

1	1	1	1	1	0	0	1	1	1	0	0
1	1	0	1	1	1	1	1	0	0	0	0
1	1	1	1	0	0	0	1	1	1	1	1
1	1	1	1	1	1	0	1	1	0	0	0
1	1	1	1	1	1	0	1	1	1	1	0
1	0	0	1	1	1	0	1	1	1	0	0

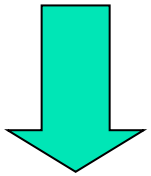
K_1
K_2
K_2
K_3
K_4
...

1 1 1 1 0 1 0 1 0 1 0 0

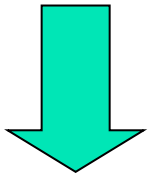
Minimisation utilisant les nuées dynamiques

Distance de Hamming
w, x ∈ {0,1}^{d}

$$I(\mathcal{W},\phi) = \sum_{i=1}^{N} |\mathbf{x}_i - \mathbf{w}_{\phi(\mathbf{x}_i)}| = \sum_{i=1}^{N}\sum_{j=1}^{n} |x_i^j - w^j_{\phi(\mathbf{x}_i)}|$$



$$I(\mathcal{W},\phi) = \sum_{j=1}^{n}\left(\sum_{i=1}^{N}(1-x_i^j)w^j_{\phi(\mathbf{x}_i)} + \sum_{i=1}^{N} x_i^j(1-w^j_{\phi(\mathbf{x}_i)})\right)$$



$$I(\mathcal{W},\phi) = \sum_{j=1}^{n}\left(w^j_{\phi(\mathbf{x}_i)}\Gamma_0^j + (1-w^j_{\phi(\mathbf{x}_i)})\Gamma_1^j\right)$$

27 races de chien, 7 variables

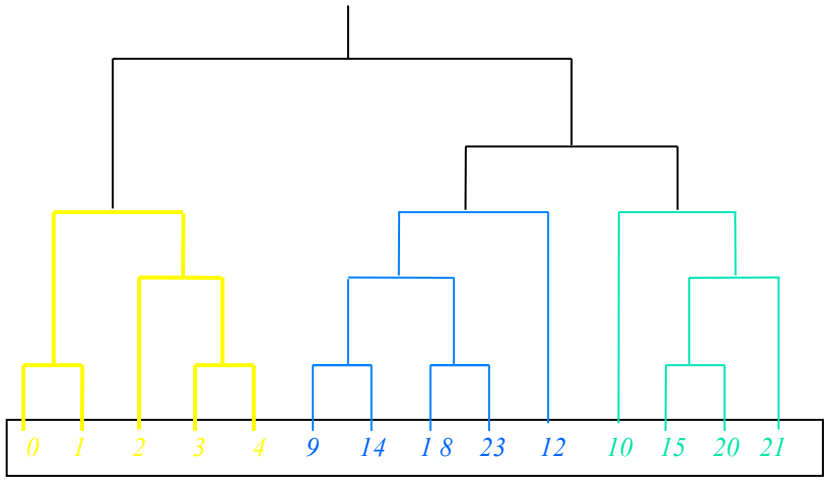
Race	PT	MT	GT	PP	MP	GP	PV	MV	GV	PI	MI	GI	NAF	AF	NAG	AG	CM	CH	U
Beauceron	0	0	1	0	1	0	0	0	1	0	0	1	0	1	0	1	0	0	1
Basset	1	0	0	1	0	0	1	0	0	1	0	0	1	0	0	1	0	1	0
Berger Allemand	0	0	1	0	1	0	0	0	1	0	0	1	0	1	0	1	0	0	1
Boxer	0	1	0	0	1	0	0	1	0	0	1	0	0	1	0	1	1	0	0
Bull-Dog	0	1	0	1	0	0	1	0	0	0	1	0	0	1	0	1	1	0	0
Bull-Mastif	0	0	1	0	0	1	1	0	0	0	0	1	1	0	0	1	0	0	1
Caniche	1	0	0	1	0	0	0	1	0	0	0	1	0	1	1	0	1	0	0
Chihuahua	1	0	0	1	0	0	1	0	0	1	0	0	0	1	1	0	1	0	0
Cocker	0	1	0	1	0	0	1	0	0	0	1	0	0	1	0	1	1	0	0
Colley	0	0	1	0	1	0	0	0	1	0	1	0	0	1	1	0	1	0	0
Dalmatien	0	1	0	0	1	0	0	1	0	0	1	0	0	1	1	0	1	0	0
Doberman	0	0	1	0	1	0	0	0	1	0	0	1	1	0	0	1	0	0	1
Dogue Allemand	0	0	1	0	0	1	0	0	1	1	0	0	1	0	0	1	0	0	1
Epagneul Breton	0	1	0	0	1	0	0	1	0	0	0	1	0	1	1	0	0	1	0
EpagneulFra nçais	0	0	1	0	1	0	0	1	0	0	1	0	1	0	1	0	0	1	0
Fox-Hound	0	0	1	0	1	0	0	0	1	1	0	0	1	0	0	1	0	1	0
Fox-Terrier	1	0	0	1	0	0	0	1	0	0	1	0	0	1	0	1	1	0	0
Grand Bleu de Gascogne	0	0	1	0	1	0	0	1	0	1	0	0	1	0	0	1	0	1	0
Labrador	0	1	0	0	1	0	0	1	0	0	1	0	0	1	1	0	0	1	0
Levrier	0	0	1	0	1	0	0	0	1	1	0	0	1	0	1	0	0	1	0
Masstiff	0	0	1	0	0	1	1	0	0	1	0	0	1	0	0	1	0	0	1
Pekinois	1	0	0	1	0	0	1	0	0	1	0	0	0	1	1	0	1	0	0
Pointer	0	0	1	0	1	0	0	0	1	0	0	1	1	0	1	0	0	1	0
Saint - Bernard	0	0	1	0	0	1	1	0	0	0	1	0	1	0	0	1	0	0	1
Setter	0	0	1	0	1	0	0	0	1	0	1	0	1	0	1	0	0	1	0
Teckel	1	0	0	1	0	0	1	0	0	0	1	0	0	1	1	0	1	0	0
Terre-Neuve	0	0	1	0	0	1	1	0	0	0	1	0	1	0	1	0	0	0	1

Application du BTM

PT,PP,PV,AF, NAG,CM Caniche Chihuahua Pekinois Teckel	PT,PP,PV,MI,AF, AG,CM Bull Dog Cocker Fox-Terrier	MT,MP,MV,MI,AF, NAG,CM Boxer Colley Dalmatien	MT,MP,MV,MI ,AF,NAG,CH Labrador	MT,MP,MV,GI,AF ,NAG,CH Epagneul Breton
				GT,MP,PI,NAF,AG CH Fox Hound Gr bleu de Gascogne
GT,GP,PV,MI, NAF,NAG,U Terre Neuve		GT,MP,GV,GI,AF,AG, U Beuceron Berger Allem Doberman		PT,PP,PV,PI,NAF,AG, CH Basset
GT,GP,PV,NAF, AG,U Bull Mastiff Saint Bernard			GT,MP,GV,NAF, NAG,CH Levrier Pointer Setter	
GT,GP,PV,PI,NAF ,AG,U Mastiff	GT,GP,GV,PI,NAF, AG,U Dogue Allem		GT,MP,MV,MI ,NAF,NAG,CH Epagneul Français	

Application de la CAH

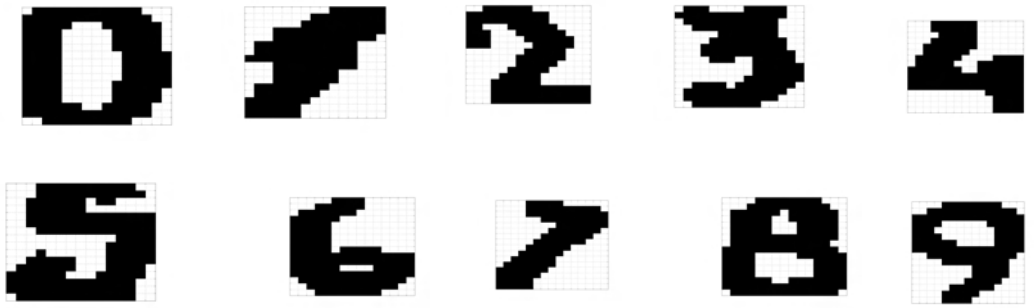
PT,PP,PV,AF,NAG, CM Caniche Chihuahua Pékinois Teckel 0	**PT,PP,PV,MI,AF ,AG,CM** Bull Dog Cocker Fox-Terrier 1	**MT,MP,MV,MI,AF, NAG,CM** Boxer Colley Dalmatien 2	**MT,MP,MV,MI,AF, NAG,CH** Labrador 3	**MT,MP,MV,GI,AF, NAG,CH** Epagneul Breton 4
5	6	7	8	**GT,MP,PI,NAF,AG, CH** Fox Hound Gr bleu de Gascogne 9
GT,GP,PV,MI,NAF, NAG,U Terre Neuve 10	11	**GT,MP,GV,GI,AF, AG,U** Beauceron Berger Allemand Doberman 12	13	**PT,PP,PV,PI,NAF,AG, CH** Basset 14
GT,GP,PV,NAF,AG, U Bull Mastiff Saint Bernard 15	16	17	**GT,MP,GV,NAF, NAG,CH** Levrier Pointer Setter 18	19
GT,GP,PV,PI,NAF, AG,U Mastiff 20	**GT,GP,GV,PI, NAF,AG,U** Dogue Allemand 21	22	**GT,MP,MV,MI, NAF,NAG,CH** Epagneul Français 23	24



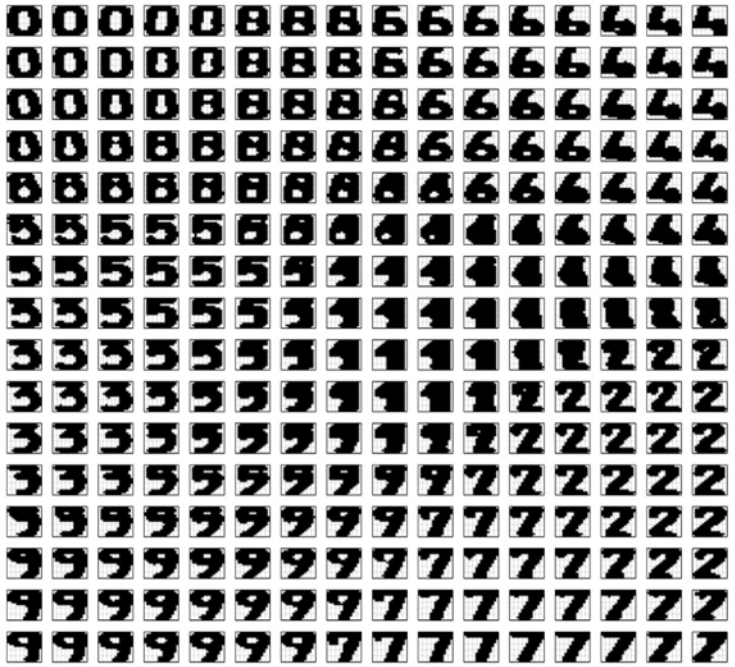
Base des chiffres

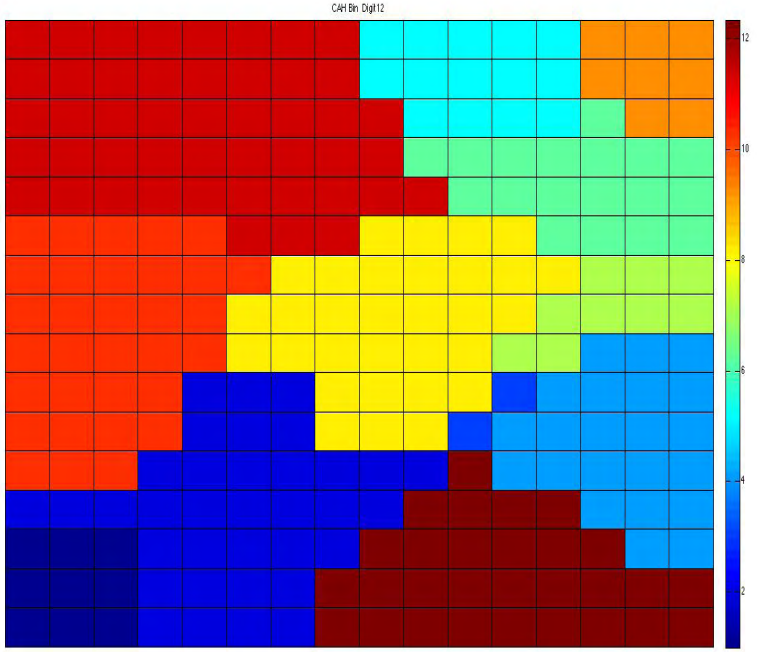
- ***2000 chiffres***

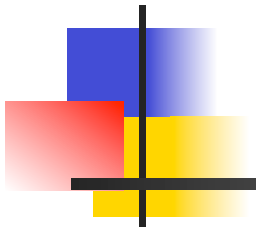
- ***Digit*** *(1:"On"/ 0:"Off")*



- *Image = 15 x 16 =* ***240 variables binaires***




CAH Bin Digit12
12
10
8
6
4
2



Modèles de mélanges

Modèles de mélanges

Le modèle de mélange fini de lois de probabilité consiste à supposer que les données proviennent d'une source contenant plusieurs sous-ensemble.

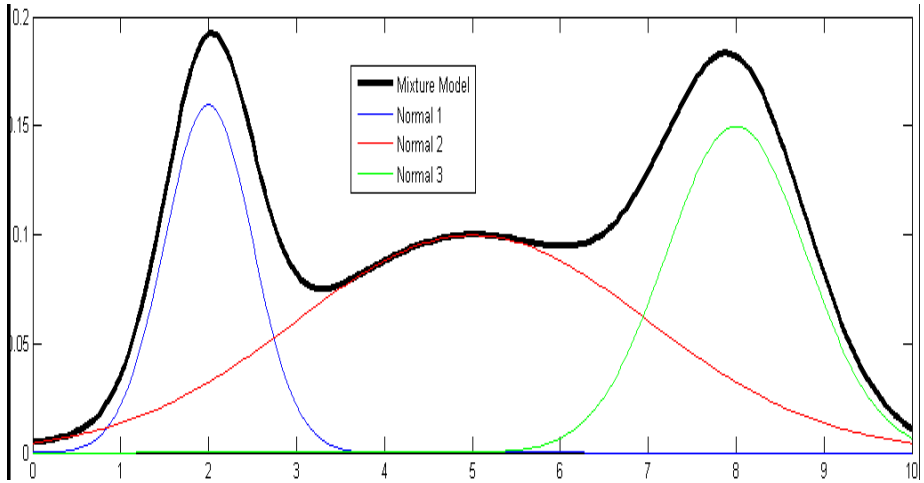
- Chaque cluster est représenté par une distribution de probabilité
- L'ensemble des données est modélisée par une mixture de ces distribution

Modèles de mélanges

- La forme générale d'un modèle de mélange a *K composant est*

$$f(\mathbf{x}) = \sum_{j} \pi_j f_j(\mathbf{x})$$

proportions du mélange



densités des composants

La paramétrisation des densités des composants dépend de la nature (continue, discrète ou binaire) des données observées.

Distribution gaussianne

$$\mathcal{N}(\mathbf{x}|\boldsymbol{\mu},\boldsymbol{\Sigma}) = \frac{1}{(2\pi)^{D/2}\,|\boldsymbol{\Sigma}|^{1/2}} \exp\left\{-\frac{1}{2}(\mathbf{x}-\boldsymbol{\mu})^{\mathsf{T}}\boldsymbol{\Sigma}^{-1}(\mathbf{x}-\boldsymbol{\mu})\right\}$$

mean

covariance

$$\boldsymbol{\mu}_{\mathsf{ML}} = \frac{1}{N}\sum_{n=1}^{N}\mathbf{x}_n$$

$$\boldsymbol{\Sigma}_{\mathsf{ML}} = \frac{1}{N}\sum_{n=1}^{N}(\mathbf{x}_n-\boldsymbol{\mu}_{\mathsf{ML}})(\mathbf{x}_n-\boldsymbol{\mu}_{\mathsf{ML}})^{\mathsf{T}}$$

Mixture gaussienne

$$p(\mathbf{x}) = \sum_{j=1}^{K} \pi_j N(\mathbf{x}; \mu_j, \Sigma_j)$$

$$\sum_{j=1}^{K} \pi_j = 1$$

La vraisemblance

- La fonction Log de vraisemblance

$$\ln V^T(\mathcal{A}; \mu_j, \Sigma_j) = \sum_{\mathbf{x}_i} \ln \left(\sum_{j=1}^{K} \pi_j N(\mathbf{x}_i; \mu_j, \Sigma_j)) \right)$$

  - **Pas de solution**

EM

•Algorithme s'appuyant sur la notion de données complétées

$$(\mathbf{x}, \mathbf{z}) = ((\mathbf{x}_1, ..., \mathbf{x}_n), (\mathbf{z}_1, ..., \mathbf{z}_n))$$

•Vraisemblance des données complètes ou vraisemblance classifiante

$$V(\mathcal{A}, \mathbf{Z}; \theta) = \prod_{i=1}^{N} \prod_{j} [p(j)p(\mathbf{x}_i/j)]^{z_{ij}}$$

•*Log-vraisemblance*

$$\ln V(\mathcal{A}, \mathbf{Z}; \theta) = \sum_{\mathbf{x}_i} \sum_{j} z_{ij} \left[\ln(p(j)) + \ln(p(\mathbf{x}_i/j))\right].$$

Maximisation itérative de:

$$Q(\theta, \theta^t) = E\left[\ln V(\mathcal{A}, \mathbf{Z}; \theta)/\mathcal{A}, \theta^t\right]$$

$$Q(\theta, \theta^t) = \sum_{\mathbf{x}_i} \sum_{j} E(z_{ij}/\mathbf{x}_i, \theta^t) \left[\ln(p(j)) + \ln(p(\mathbf{x}_i/j))\right]$$

$$E(z_{ij}/\mathbf{x}_i, \theta^t) = p(j/\mathbf{x}, \theta^t)$$

$$Q^T(\theta, \theta^t) = Q_1^T(\theta_1, \theta^t) + Q_2^T(\theta_2, \theta^t)$$

Après initialisation

E*tape E* *: Calcul des probabilités conditionnelles a posteriori, que l'observation x provienne du composant j pour la valeur courante du paramètre du mélange :*

$$p(j|\mathbf{x}) = \frac{p(j)p(\mathbf{x}|j)}{p(\mathbf{x})} = \frac{\pi_j \mathcal{N}(\mathbf{x}|\boldsymbol{\mu}_j, \boldsymbol{\Sigma}_j)}{\sum_{k=1}^{K} \pi_k \mathcal{N}(\mathbf{x}|\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)} = \gamma_j(\mathbf{x})$$

Etape M *: Mettre à jour l'estimation des paramètres en maximisant l'espérance de la vraisemblance des données complétées.*

EM Algorithm

$$\boldsymbol{\mu}_j = \frac{\sum_{n=1}^{N} \gamma_j(\mathbf{x}_n)\mathbf{x}_n}{\sum_{n=1}^{N} \gamma_j(\mathbf{x}_n)}$$

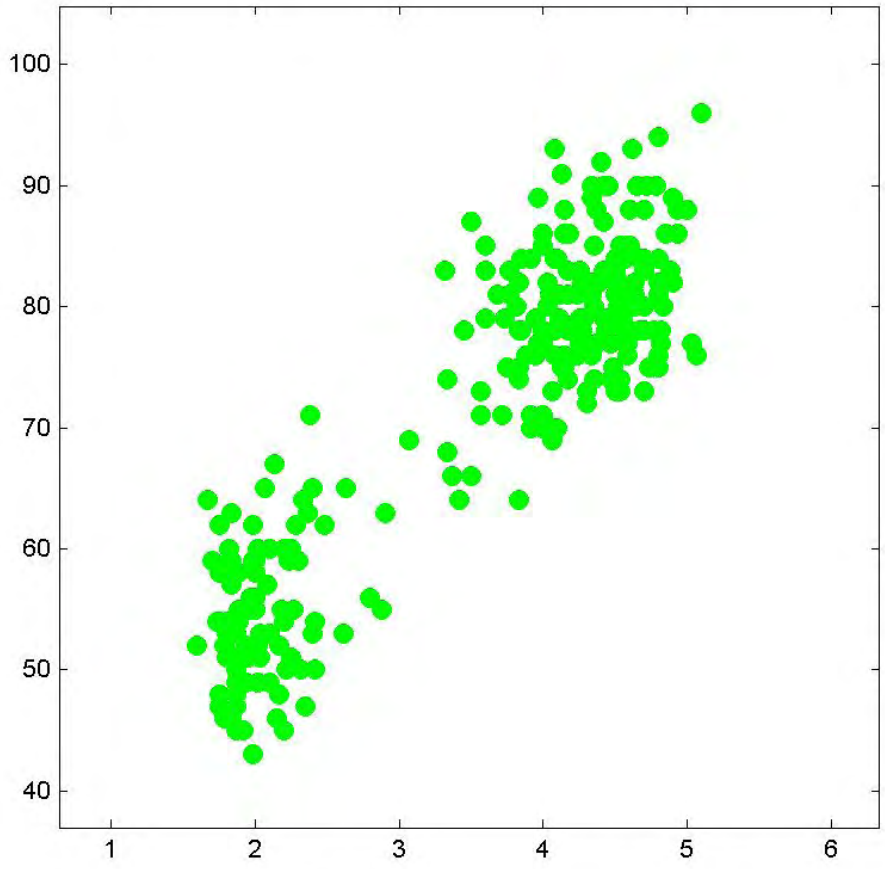
$$\pi_j = \frac{1}{N}\sum_{n=1}^{N} \gamma_j(\mathbf{x}_n)$$

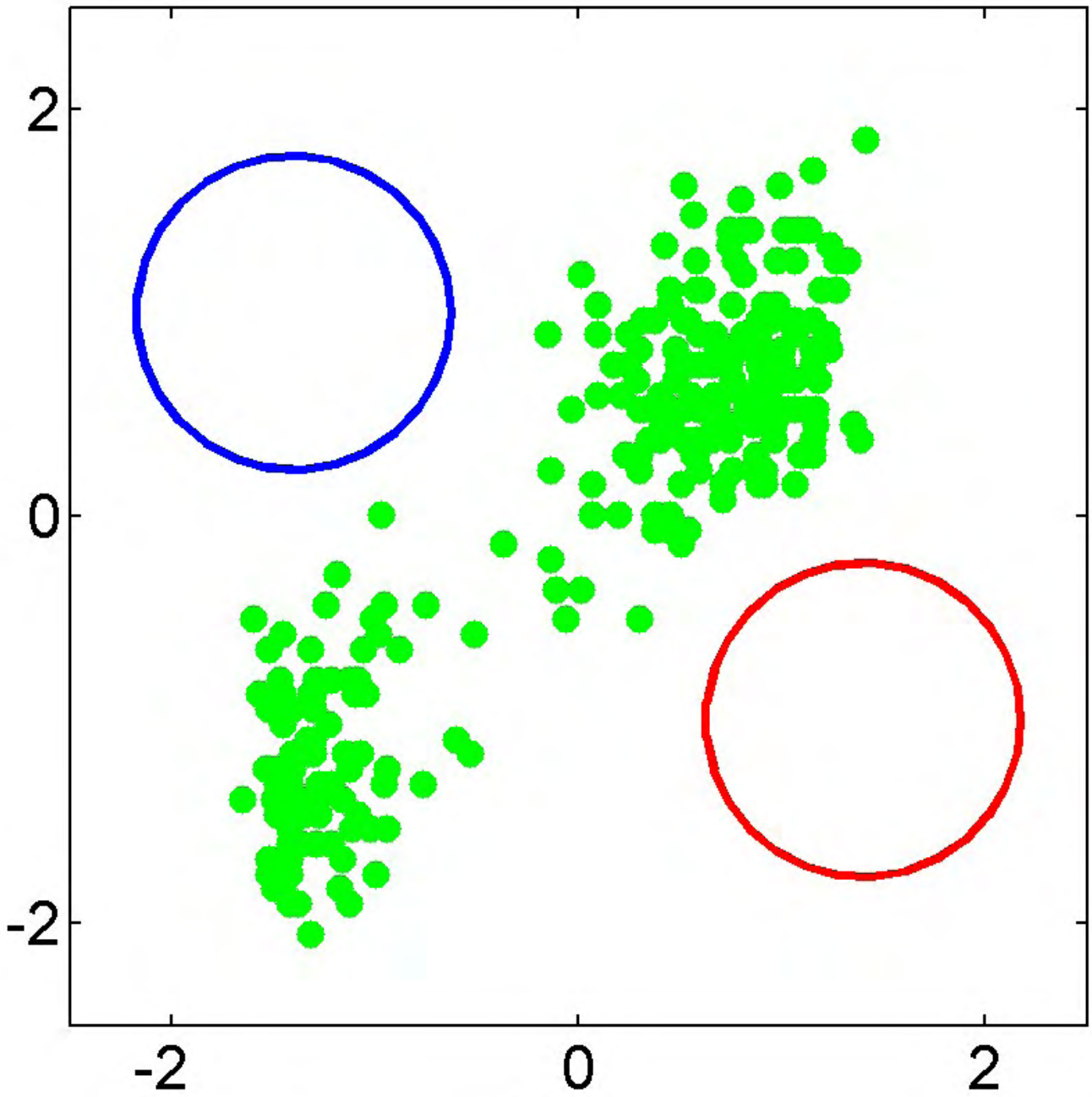
$$\boldsymbol{\Sigma}_j = \frac{\sum_{n=1}^{N} \gamma_j(\mathbf{x}_n)(\mathbf{x}_n - \boldsymbol{\mu}_j)(\mathbf{x}_n - \boldsymbol{\mu}_j)^{\mathsf{T}}}{\sum_{n=1}^{N} \gamma_j(\mathbf{x}_n)}$$

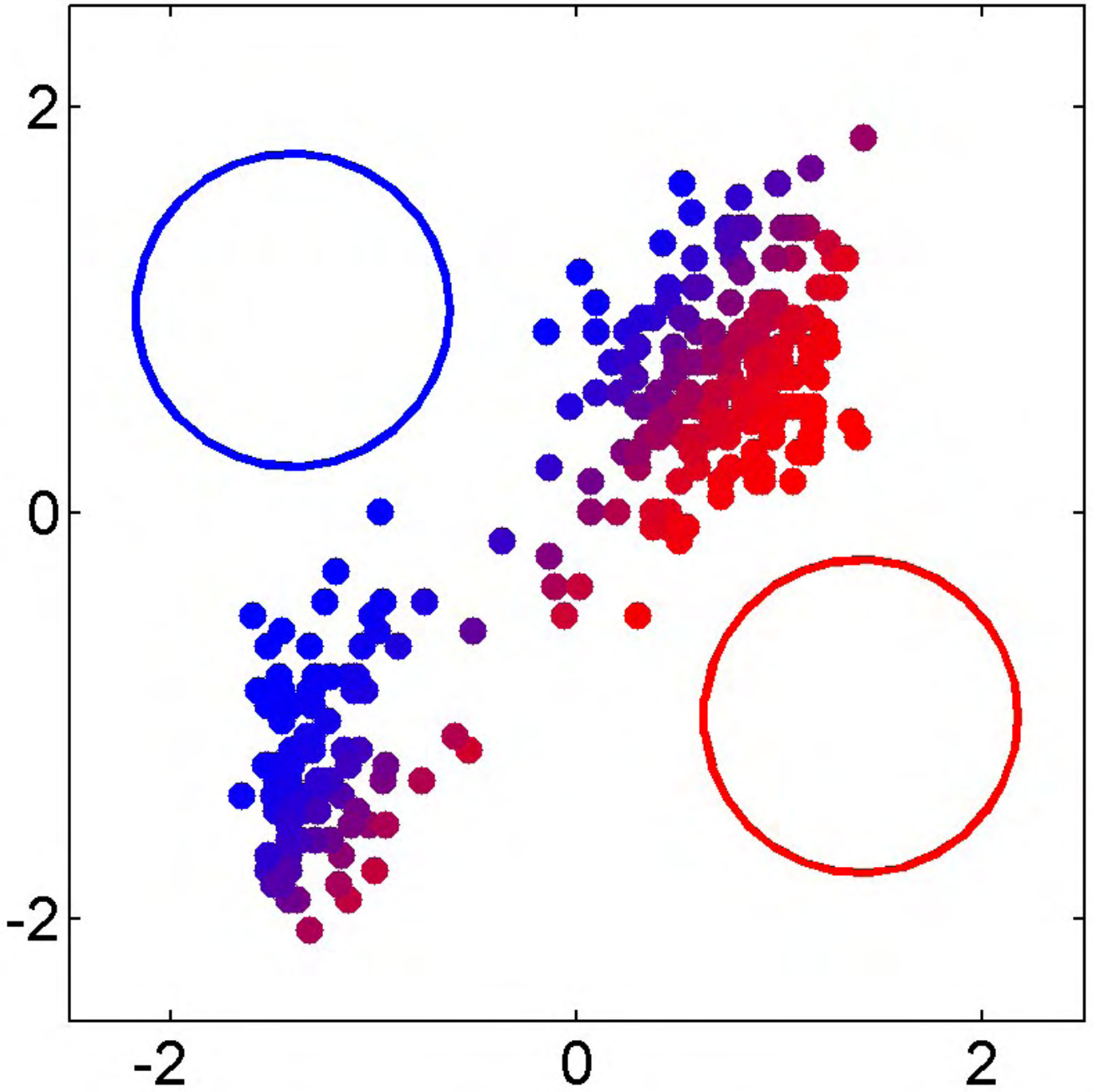
Quelques caractéristiques de EM

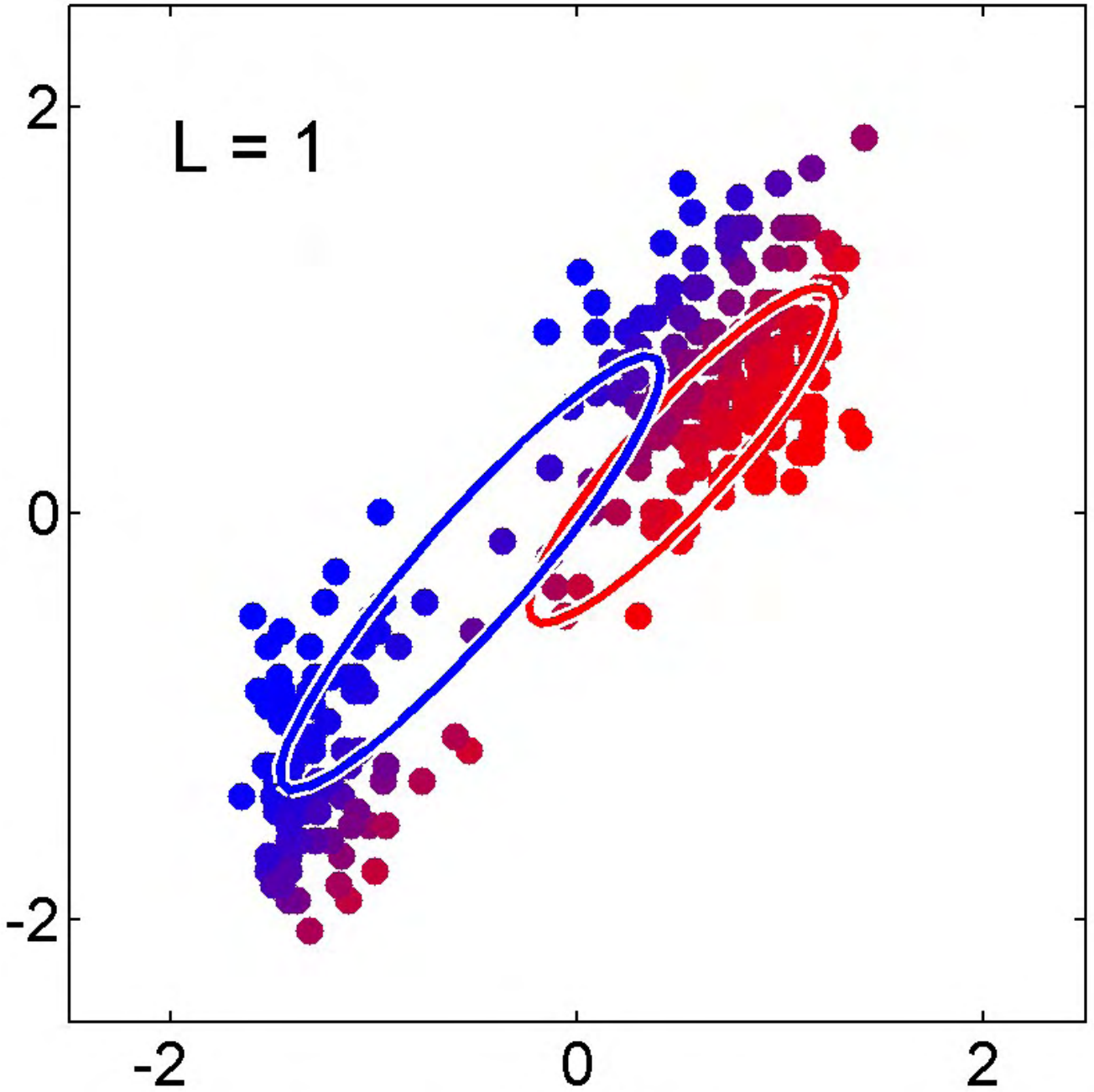
- EM fait croître la vraisemblance à chaque itération
- Facile à programmer
- Situation de convergence lente (en particulier, lorsque les composants sont très mélangés)

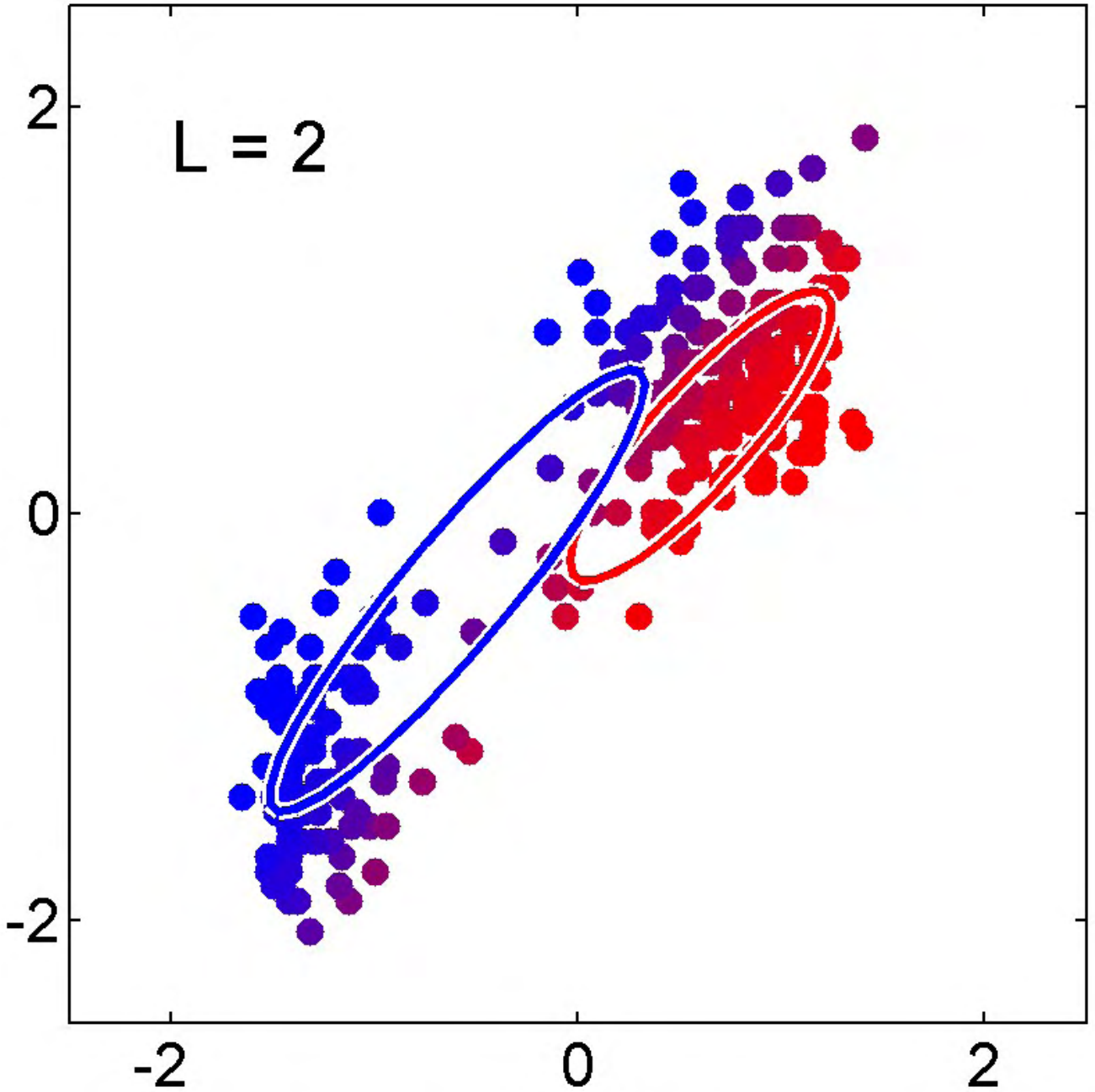
exemple

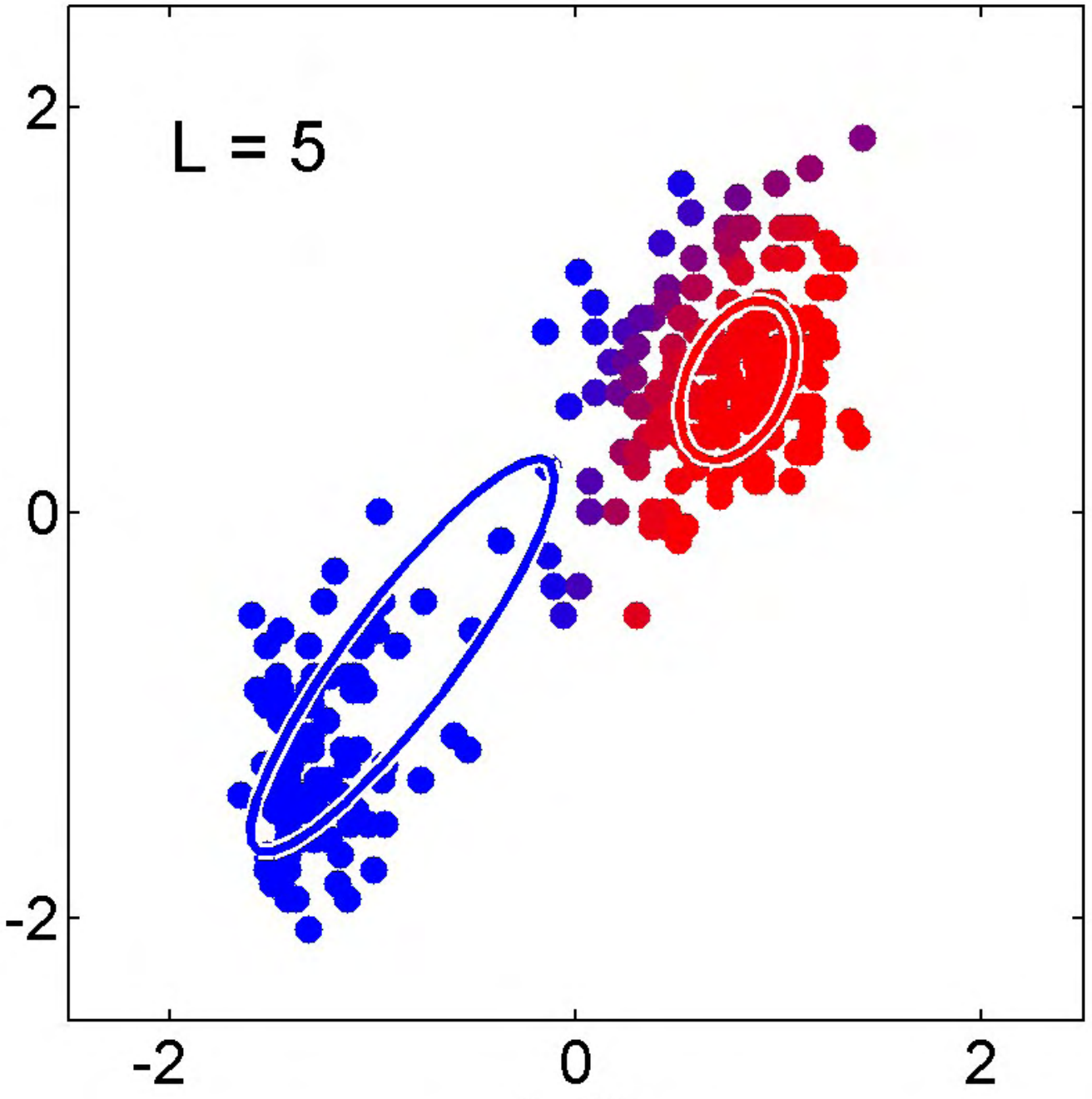


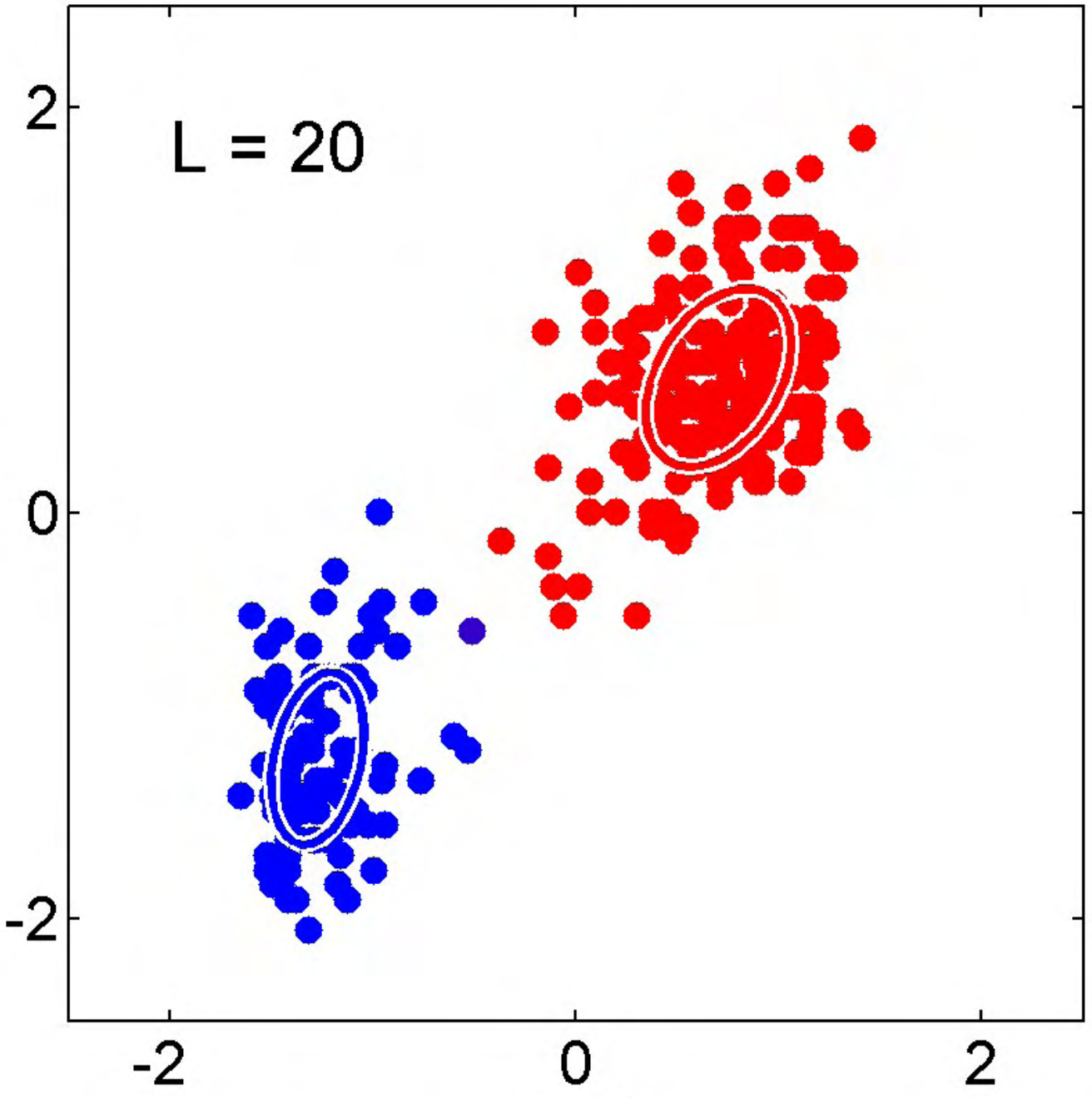

2
0
-2
-2
0
2


2
0
-2
-2
0
2


L = 1
2
0
-2
-2
0
2

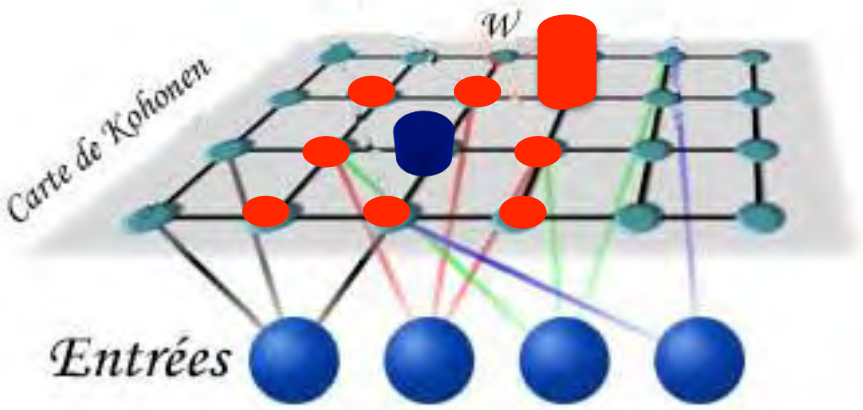

L = 2
2
0
-2
-2
0
2


L = 5
2
0
-2
-2
0
2


L = 20
2
0
-2
-2
0
2

PrSOM : probabilistic Self-Organizing Maps

Formalisme probabiliste



$$p(c^*)$$

$$p(c/c^*) = \frac{K^T(\delta(c,c^*))}{\sum_{r\in C} K^T(\delta(r,c^*))},$$

$$p(\mathbf{x}/c)$$

$$p_{c^*}(\mathbf{x}/c^*) = \sum_{c\in C} p(c/c^*)p(\mathbf{x}/c)$$

$$p(\mathbf{x}) = \sum_{c^*\in C} p(c^*)p_{c^*}(\mathbf{x}/c^*)$$

$p(\mathbf{x})$= ***Mélange de mélanges locales de lois de probabilité***

$$p(\mathbf{x}) = \sum_{z \in \mathbf{Z}} p(\mathbf{x}, \mathbf{z}) = \sum_{c \in \mathcal{C}, c^* \in \mathcal{C}} p(\mathbf{x}/c) p(c/c^*) p(c^*)$$

$$z_i^{(c,c^*)} = \begin{cases} 1 & \text{pour } \mathbf{z}_i = (c, c^*) \\ 0 & \text{sinon} \end{cases}$$

$$p(\mathbf{x}, z) = \prod_{c^* \in \mathcal{C}} \prod_{c \in \mathcal{C}} [p(c^*) p(c/c^*) p(\mathbf{x}_i/c)]^{z_i^{(c,c^*)}}$$

Ainsi la vraisemblance des données s'écrit par

$$V^T(\mathcal{A}, \mathbf{Z}; \theta) = \prod_{i=1}^{N} \prod_{c^* \in \mathcal{C}} \prod_{c \in \mathcal{C}} [p(c^*) p(c/c^*) p(\mathbf{x}_i/c)]^{z_i^{(c,c^*)}}$$

le log-vraisemblance s'écrit:

$$\ln V^T(\mathcal{A}, \mathbf{Z}; \theta) = \sum_{\mathbf{x}_i \in \mathcal{A}} \sum_{c^* \in C} \sum_{c \in C} z^i_{(c,c^*)} \left[\ln(p(c^*)) + \ln(p(c/c^*) + \ln(p(\mathbf{x}_i/c))\right].$$

L'application de l'algorithme EM

$$Q^T(\theta, \theta^t) = \sum_{\mathbf{x}_i \in \mathcal{A}} \sum_{c^* \in C} \sum_{c \in C} E(z^i_{(c,c^*)}/\mathbf{x}_i, \theta^t) \left[\ln(p(c^*)) + \ln\left(p(c/c^*)\right) + \ln(p(\mathbf{x}_i/c))\right]$$

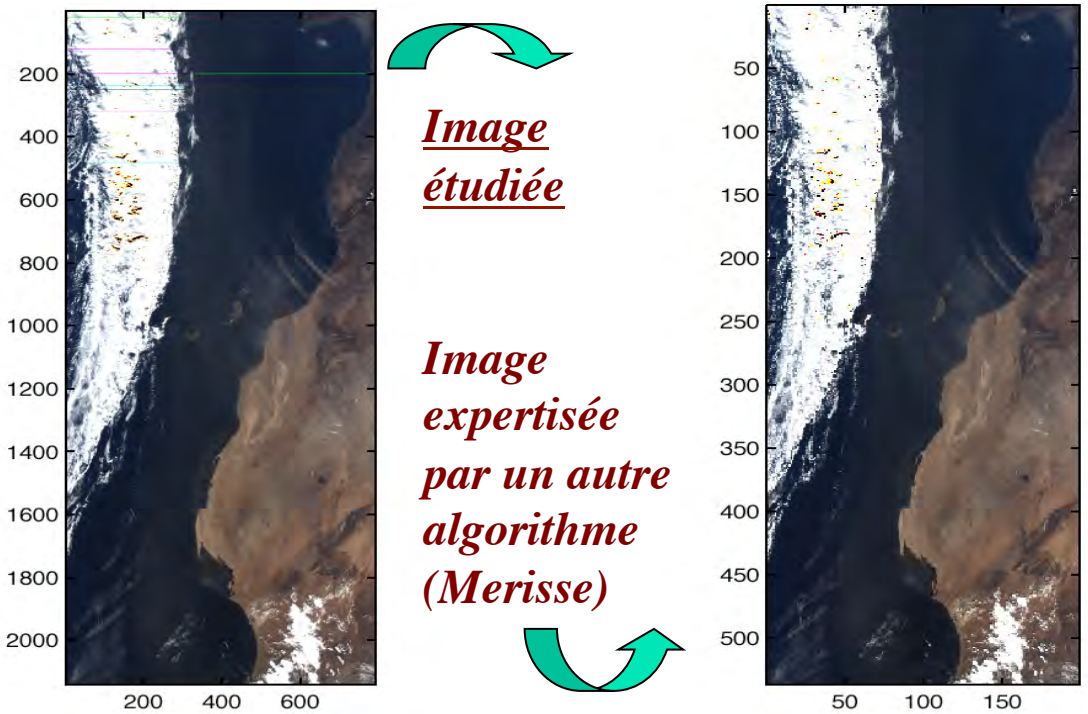
$$E(z^i_{(c,c^*)}/\mathbf{x}_i, \theta^t) = p(z^i_{(c,c^*)} = 1/\mathbf{x}_i, \theta^t) = p(c, c^*/\mathbf{x}_i, \theta^t)$$

$$= \frac{p(c^*)p(c/c^*)p(\mathbf{x}/c)}{\sum_{r \in \mathcal{C}} p(r)p_r(\mathbf{x})}$$

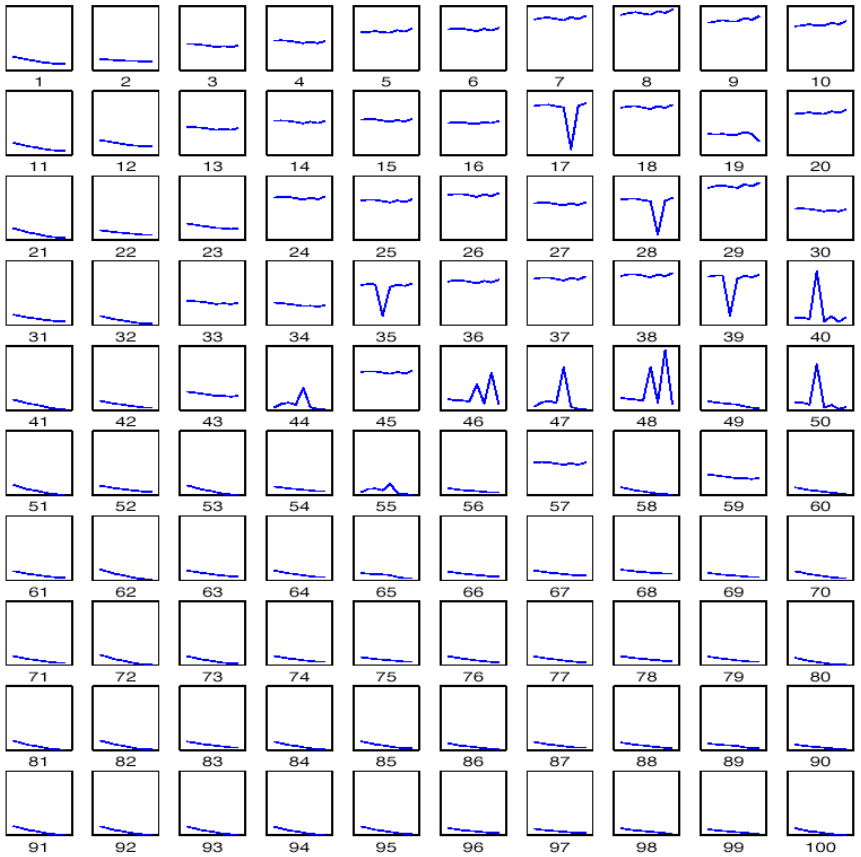
$$Q^T(\theta, \theta^t) = Q_1^T(\theta^C, \theta^t) + Q_2^T(\theta^{C^*}, \theta^t) + Q_3^T(\theta^t)$$

***Les données**: image SeaWifs sur la Afrique de l 'ouest et les Iles Canaries 5 janvier 1999 échantillonnée d'une manière homogène (1 ligne sur 10). Chaque observation est constitué par les 8 longueurs d'onde.*

***Apprentissage**: carte topologique 10*10*



VISUALISATION SIMULTANEE DES REFERENTS ET DE LA CARTE

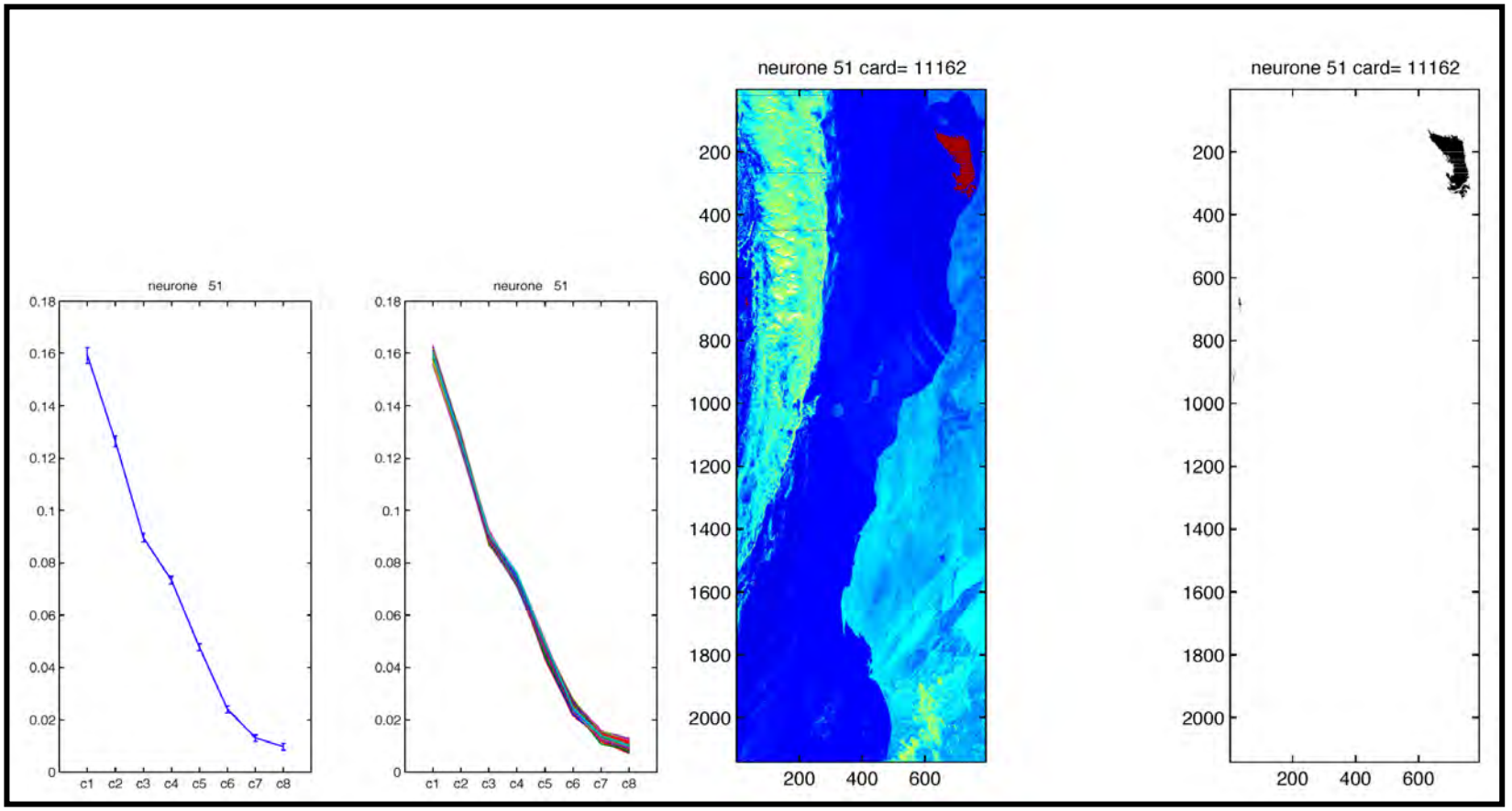


Carte SOM (10*10)

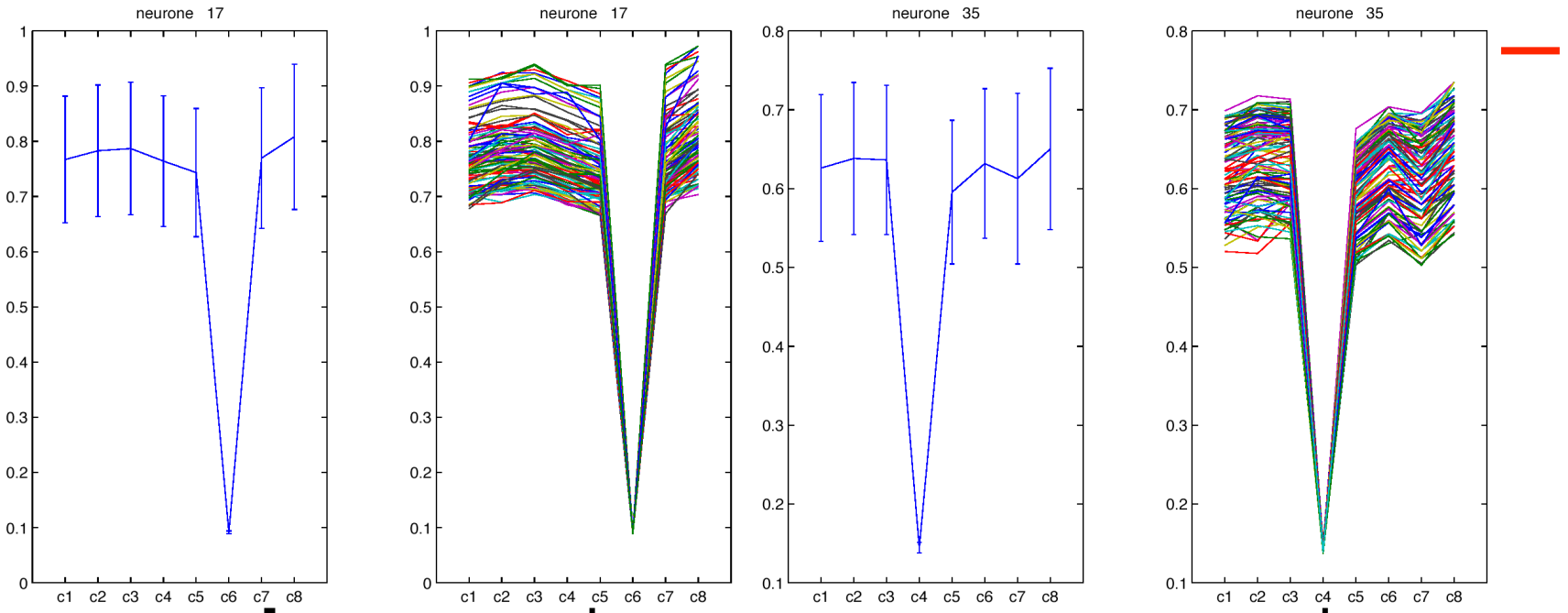
Permet d'interpréter l'ordre topologique obtenu en fonction des données. Chaque référent est un spectre de $\boldsymbol{R^8}$.

La forme des spectres des référents varie de manière ordonnée selon les deux directions.

Visualisation d'une cellule, des spectres associés
Localisation du sous-ensemble de pixels sur l 'image



Visualisation de la partition: détection des données aberrantes

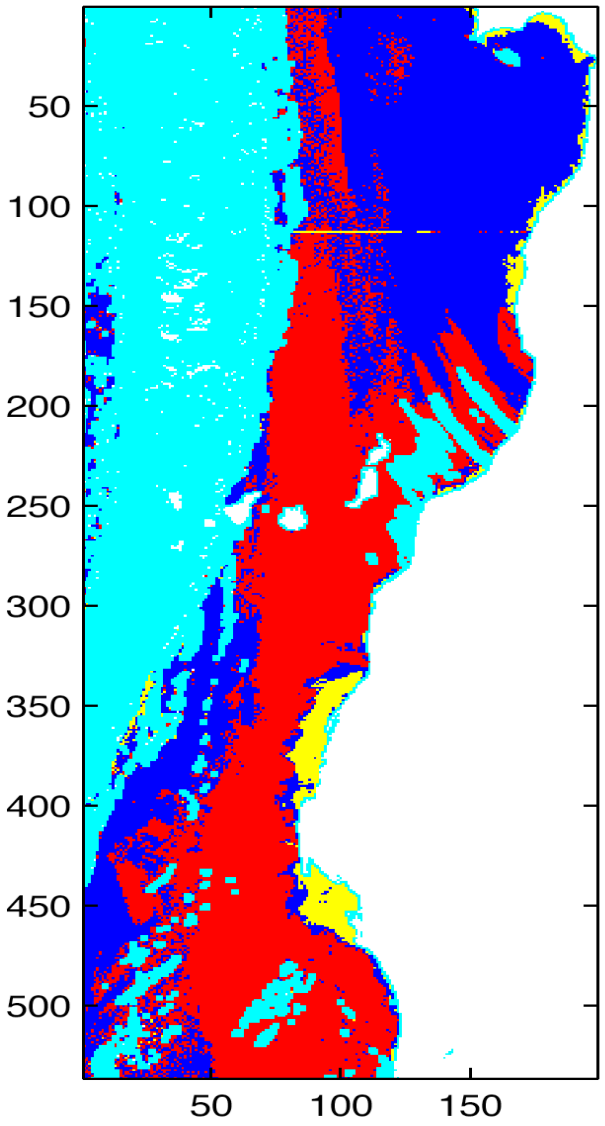


Spectre et variance des cellules 17 et 32

Spectres des pixels de l ’image affectés au neurone 16, 32

Tous les pixels pour lesquels un canal n ’a pas fonctionné sont affectés à la même cellule

Image étiqueté



Aérosol désertique: bleu foncé

Aérosol marin: rouge

Nuages: bleu clair

Eaux du cas2: jaune

BeSOM (Bernoulli Self-Organizing Map)
Données binaires

BeSOM (Bernoulli Self-Organizing Map)

$\mathbf{w}_c$ → (100 1111 01 010)

x^k ***identique*** à w^k_c avec la probabilité ***1-***ε^k_c et ***diffèrent*** avec la probabilité ε^k_c **[Nadif et al 98]**

→ ***Loi de Bernoulli***

$$p(\mathbf{x}) = \sum_{c^* \in \mathcal{C}} p(c^*) p_{c^*}(\mathbf{x}/c^*),$$

$$p_{c^*}(\mathbf{x}/c^*) = \sum_{c \in \mathcal{C}} p(c/c^*) p(\mathbf{x}/c)$$

$p(c^*)$

$$p(\mathbf{x}/\mathbf{w}_c, \varepsilon_c) = \varepsilon_c^{\mathcal{H}(\mathbf{x},\mathbf{w}_c)} (1-\varepsilon_c)^{n-\mathcal{H}(\mathbf{x},\mathbf{w}_c)}$$

L'application de l'algorithme EM

$$Q^T(\theta,\theta^t) = \sum_{\mathbf{x}_i \in \mathcal{A}} \sum_{c^* \in C} \sum_{c \in C} E(z^i_{(c,c^*)}/\mathbf{x}_i,\theta^t)\left[\ln(p(c^*)) + \ln\left(p(c/c^*)\right) + \ln(p(\mathbf{x}_i/c))\right]$$

$$E(z^i_{(c,c^*)}/\mathbf{x}_i,\theta^t) = p(z^i_{(c,c^*)} = 1/\mathbf{x}_i,\theta^t) = p(c,c^*/\mathbf{x}_i,\theta^t)$$

$$= \frac{p(c^*)p(c/c^*)p(\mathbf{x}/c)}{\sum_{r\in\mathcal{C}} p(r)p_r(\mathbf{x})}$$

$$Q^T(\theta,\theta^t) = Q_1^T(\theta^C,\theta^t) + Q_2^T(\theta^{C^*},\theta^t) + Q_3^T(\theta^t)$$

BeSOM ε_c

$$
\begin{aligned}
Q_1^T(\theta^C, \theta^t) &= \sum_{\mathbf{x}_i \in \mathcal{A}} \sum_{c \in C} p(c/\mathbf{x}_i, \theta^t) \ln \left(\prod_{k=1}^{n} p(x_i^k/c) \right) \\
&= \sum_{\mathbf{x}_i \in \mathcal{A}} \sum_{c \in C} p(c/\mathbf{x}_i, \theta^t) \ln \left(\varepsilon_c^{\mathcal{H}(\mathbf{x}_i, \mathbf{w}_c)} (1-\varepsilon_c)^{n - \mathcal{H}(\mathbf{x}_i, \mathbf{w}_c)} \right) \\
&= \sum_{\mathbf{x}_i \in \mathcal{A}} \sum_{c \in C} p(c/\mathbf{x}_i, \theta^t) \left[\mathcal{H}(\mathbf{x}_i, \mathbf{w}_c) \ln(\varepsilon_c) n - \mathcal{H}(\mathbf{x}_i, \mathbf{w}_c) \ln(1-\varepsilon_c) \right]
\end{aligned}
$$

$$
\begin{aligned}
Q_1^T(\theta^C, \theta^t) &= \sum_{\mathbf{x}_i \in \mathcal{A}} \sum_{c \in C} \left[-\ln \left(\frac{1-\varepsilon_c}{\varepsilon_c} \right) p(c/\mathbf{x}_i, \theta^t) \mathcal{H}(\mathbf{x}_i, \mathbf{w}_c) + np(c/\mathbf{x}_i, \theta^t) \ln(1-\varepsilon_c) \right] \\
&= \sum_{c \in C} \sum_{\mathbf{x}_i \in \mathcal{A}} \left[-\ln \left(\frac{1-\varepsilon_c}{\varepsilon_c} \right) p(c/\mathbf{x}_i, \theta^t) \mathcal{H}(\mathbf{x}_i, \mathbf{w}_c) \right] \\
&\quad + \sum_{c \in C} \sum_{\mathbf{x}_i \in \mathcal{A}} \left[np(c/\mathbf{x}_i, \theta^t) \ln(1-\varepsilon_c) \right]
\end{aligned}
$$

BeSOM (Bernoulli Self Organizing Map)

$$\varepsilon_c$$

$$\theta = \quad p(c^*) \qquad (\mathbf{w}_c, \epsilon_c)$$

$$p(c^*) = \frac{\sum_{\mathbf{x}_i \in \mathcal{A}} p(c^*/\mathbf{x}_i, \theta^t)}{N}$$

$$w_c^k = \begin{cases} 0 & \text{si } \left[\sum_{\mathbf{x}_i \in \mathcal{A}} p(c/\mathbf{x}_i, \theta^t)(1 - x_i^k)\right] \geq \\ & \left[\sum_{\mathbf{x}_i \in \mathcal{A}} p(c/\mathbf{x}_i, \theta^t) x_i^k\right] \\ 1 & \text{sinon} \end{cases}, \quad \varepsilon_c = \frac{\sum_{\mathbf{x}_i \in \mathcal{A}} p(c/\mathbf{x}_i, \theta^t) \mathcal{H}(\mathbf{x}_i, \mathbf{w}_c)}{\sum_{\mathbf{x}_i \in \mathcal{A}} n p(c/\mathbf{x}_i, \theta^t)}$$

$$p(c, c^*/\mathbf{x}) = \frac{p(c^*) p(c/c^*) p(\mathbf{x}/c)}{\sum_{r \in \mathcal{C}} p(r) p_r(\mathbf{x})} \qquad p(c/c^*) = \frac{\mathcal{K}^T(\delta(c, c^*))}{\sum_{r \in C} \mathcal{K}^T(\delta(r, c^*))}$$

BeSOM (Bernoulli Self Organizing Map)

$$\epsilon_c = (\varepsilon_c^1, \ldots \varepsilon_c^k, \ldots, \varepsilon_c^n)$$

$$\theta = \quad p(c^*) \qquad (\mathbf{w}_c, \epsilon_c)$$

$$w_c^k = \begin{cases} 0 & \text{si } \left[\sum_{\mathbf{x}_i \in \mathcal{A}} p(c/\mathbf{x}_i, \theta^t)(1 - x_i^k)\right] \geq \\ & \left[\sum_{\mathbf{x}_i \in \mathcal{A}} p(c/\mathbf{x}_i, \theta^t) x_i^k\right] \\ 1 & \text{sinon} \end{cases}$$

$$p(c^*) = \frac{\sum_{\mathbf{x}_i \in \mathcal{A}} p(c^*/\mathbf{x}_i, \theta^t)}{N}$$

$$\varepsilon_c^k = \frac{\sum_{\mathbf{x}_i \in \mathcal{A}} p(c/\mathbf{x}_i, \theta^t) |x_i^k - w_{c1}^k|}{\sum_{\mathbf{x}_i \in \mathcal{A}} p(c/\mathbf{x}_i, \theta^t)}$$

$$p(c, c^*/\mathbf{x}) = \frac{p(c^*) p(c/c^*) p(\mathbf{x}/c)}{\sum_{r \in \mathcal{C}} p(r) p_r(\mathbf{x})}$$

$$p(c/c^*) = \frac{\mathcal{K}^T(\delta(c, c^*))}{\sum_{r \in C} \mathcal{K}^T(\delta(r, c^*))}$$

Lien avec la version BTM

$$p(\mathbf{x}/c) = \varepsilon^{\mathcal{H}(\mathbf{x},\mathbf{w}_c)}(1-\varepsilon)^{n-\mathcal{H}(\mathbf{x},\mathbf{w}_c)}$$

$$p(c^*) = \frac{1}{K}$$

Maximiser :

$$Q_1^T(\theta^C, \theta^{t-1}) = \ln\left(\frac{1-\varepsilon}{\varepsilon}\right) \sum_{c\in C}\sum_{\mathbf{x}_i\in\mathcal{A}} \left[-p(c/\mathbf{x}_i, \theta^{t-1})\mathcal{H}(\mathbf{x}_i, \mathbf{w}_c)\right] + \ln(1-\varepsilon) \sum_{c\in C}\sum_{\mathbf{x}_i\in\mathcal{A}} \left[np(c/\mathbf{x}_i, \theta^{t-1})\right]$$

Minimiser :

$$G(\mathbf{w}) = \sum_{c\in\mathcal{C}}\sum_{\mathbf{x}_i\in\mathcal{A}} p(c/\mathbf{x}_i, \theta^{t-1})\mathcal{H}(\mathbf{x}_i, \mathbf{w}_c)$$

Lien avec la version BTM

$$G(\mathbf{w}) = \sum_{c \in \mathcal{C}} \sum_{\mathbf{x}_i \in \mathcal{A}} p(c/\mathbf{x}_i, \theta^{t-1}) \mathcal{H}(\mathbf{x}_i, \mathbf{w}_c)$$

$$J_{bin}^{T}(\mathcal{W}, \phi) = \sum_{\mathbf{x}_i \in \mathcal{A}} \sum_{c \in C} K^{T}(\delta(c, \phi(\mathbf{x}_i))) |\mathbf{x}_i - \mathbf{w}_c|$$

$$p(c/\mathbf{x}_i, \theta^{t-1}) = \sum_{c^* \in \mathcal{C}} p(c, c^*/\mathbf{x}_i, \theta^{t-1})$$

$$p(c, c^*/\mathbf{x}) = \frac{p(c^*) p(c/c^*) p(\mathbf{x}/c)}{\sum_{r \in \mathcal{C}} p(r) p_r(\mathbf{x})} \qquad p(c/c^*) = \frac{\mathcal{K}^{T}(\delta(c, c^*))}{\sum_{r \in C} \mathcal{K}^{T}(\delta(r, c^*))}$$

Un exemple : Chiffres manuscrits

- ***2000 chiffres***
- ***Chiffre = Image binaire*** *(1/0)*

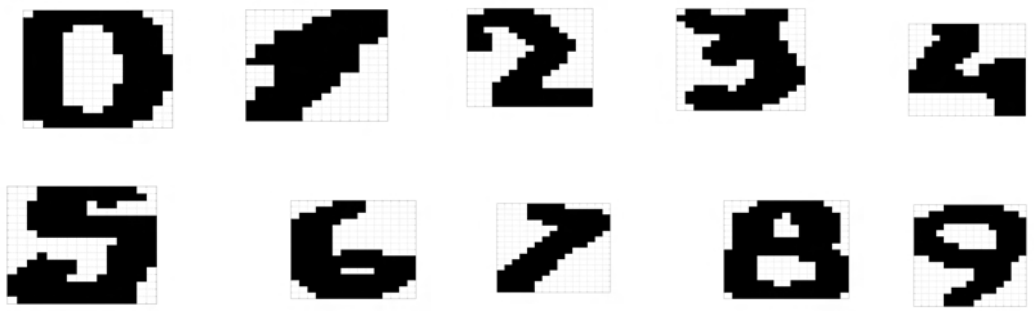


Image = 15 x 16 = ***240 variables binaires***

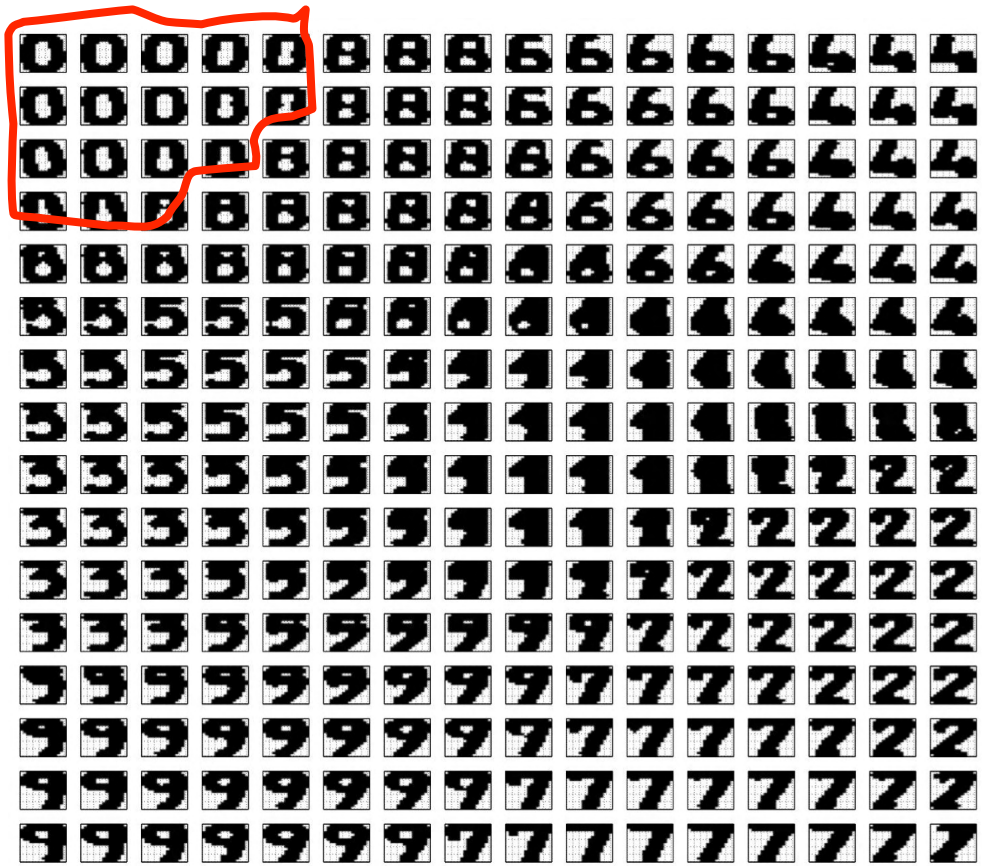
BeSOM (Résultats)

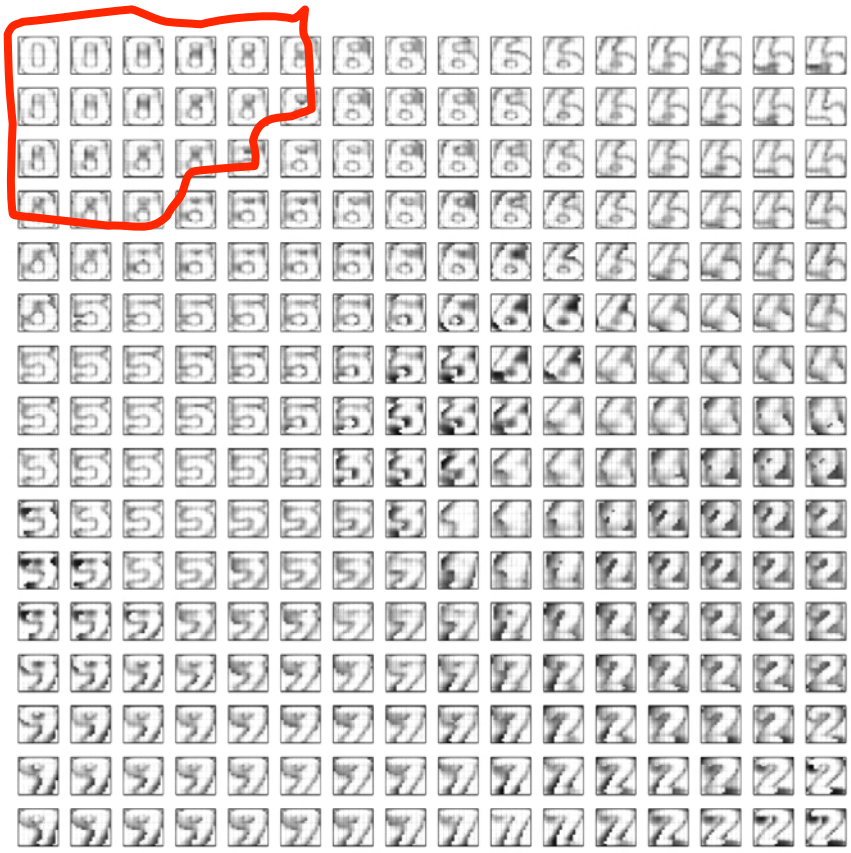
$$\mathbf{w}_c = (w_c^1, w_c^2, ...w_c^k, ..., w_c^{240}) \in \{0,1\}^{240}$$

BeSOM (Résultats)

$$\mathbf{w}_c = (w_c^1, w_c^2, ...w_c^k, ..., w_c^{240})$$

$$\epsilon_c = (\varepsilon_c^1, \varepsilon_c^2, ..., \varepsilon_c^k, ..., \varepsilon_c^{240})$$

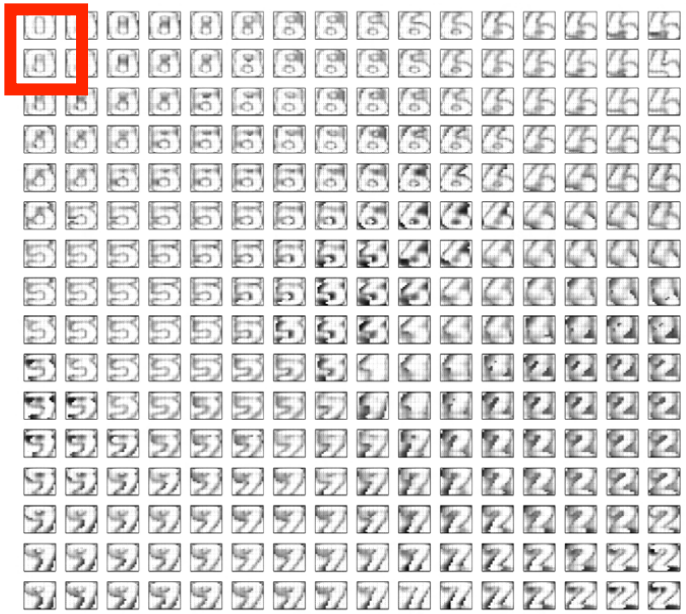




Approche probabiliste ➡ *Analyse fine du comportement du modèle*

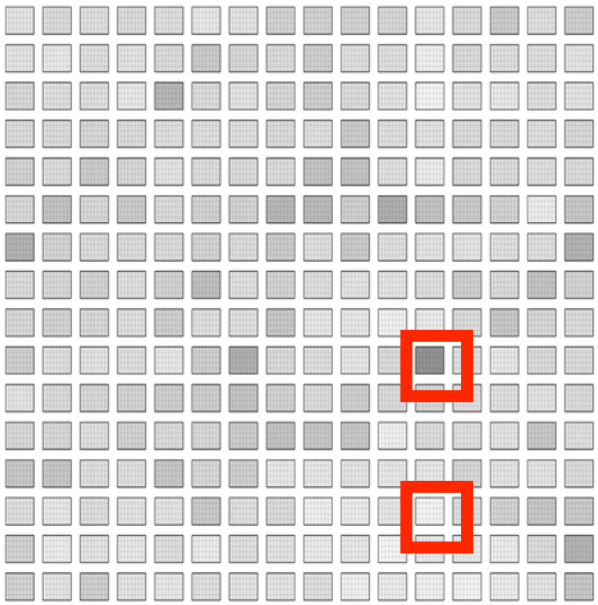
BeSOM (différents niveaux de pertinences)

BeSOM – ϵ_c



- *Pertinence des variables*
- *Sélection de variable*
- *Caractérisation des groupes*

BeSOM – ε_c



- *Pertinence des prototypes*
- *Elagage de neurones*
- *Échantillonnage optimisé*

BeSOM – ε

2.147335e-01

- *Pertinence de la carte*
- *Sélection de modèle*

Plus d'information

Base de Sémiométrie, Base réelle

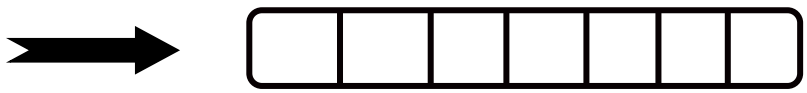
7 notes sémiométriques (modalités)

- *Univocité sémantique*
- *Stabilité sémantique*
- *Non-consensualité*

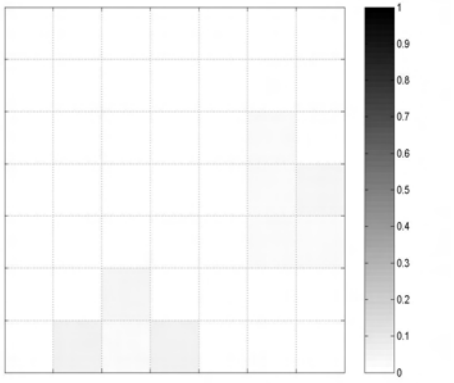
ACHETER
AMBITION
ANGOISSE
ARGENT
ASTUCIEUX
AUDACE
BIJOU
CADEAU
….

7 paramètres à estimer

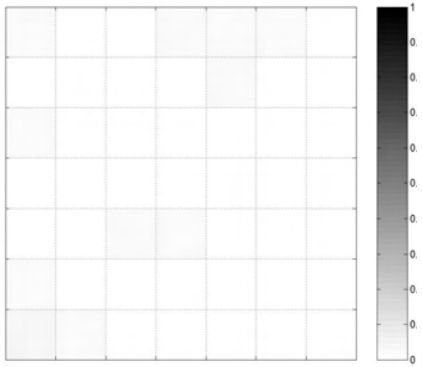
Mot



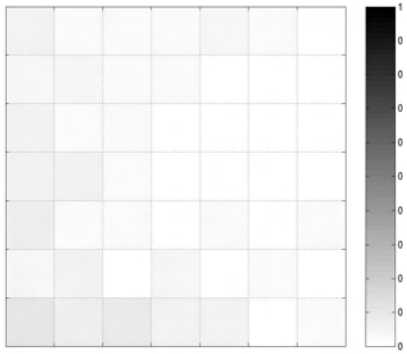
Répartition de probabilité du mot : ACHETER



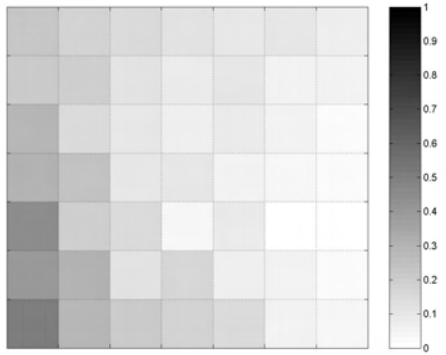
1



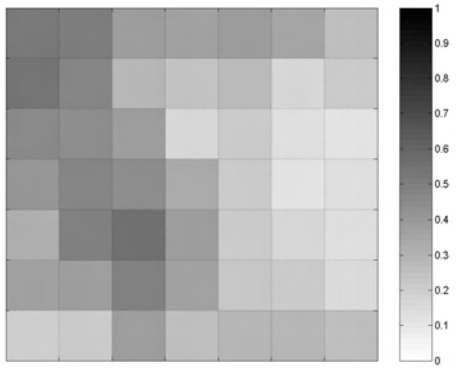
2



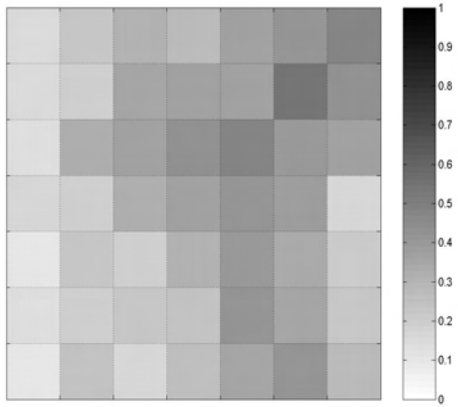
3



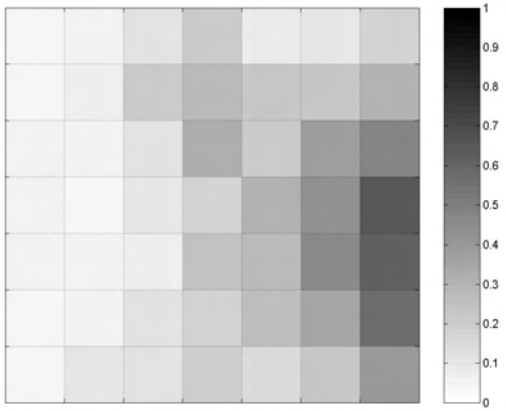
4



5

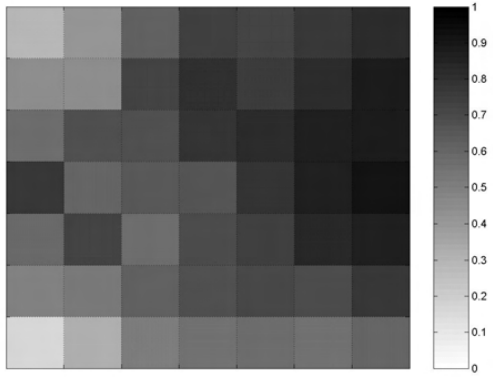


6

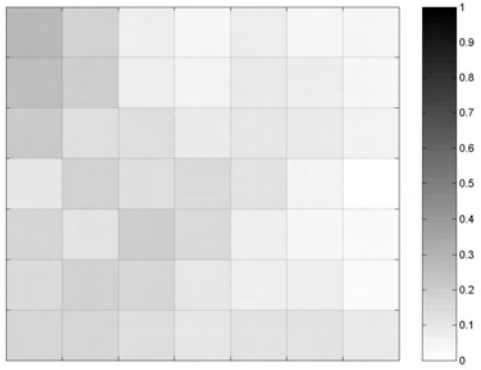


7

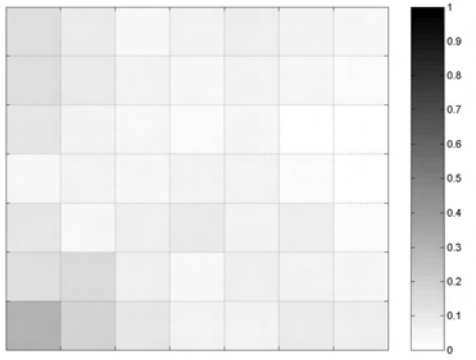
Répartition de probabilité du mot : Mort



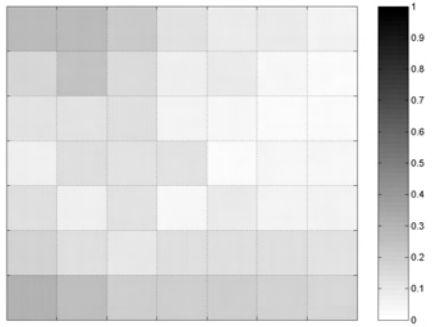
1



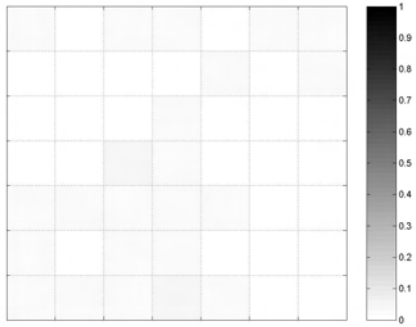
2



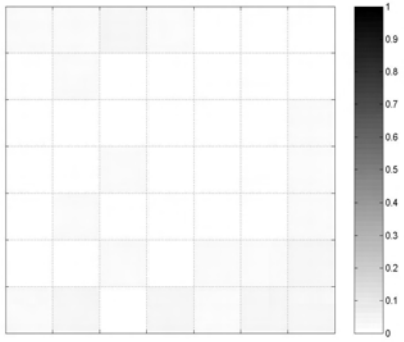
3



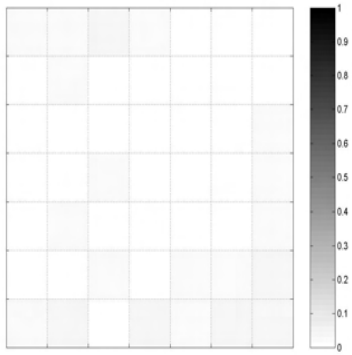
4



5

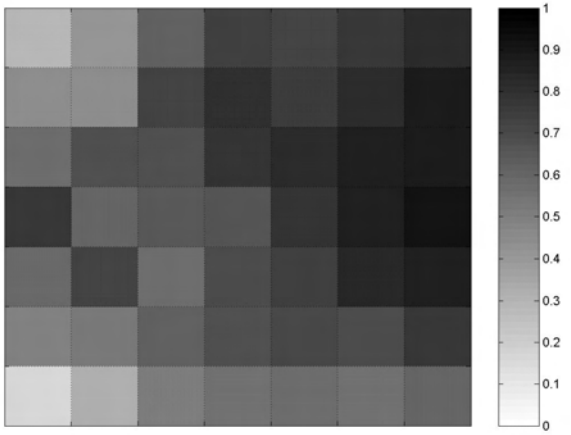


6

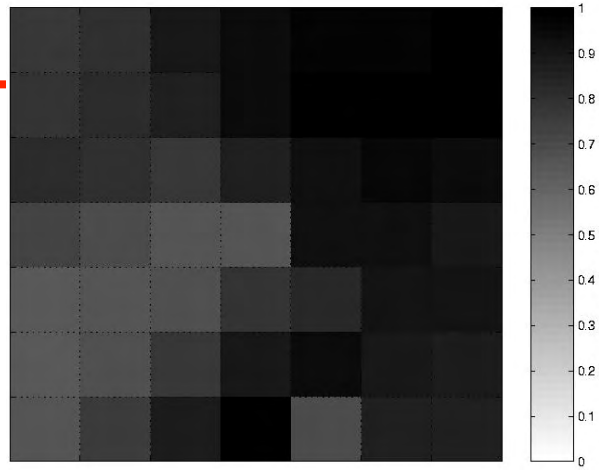


7

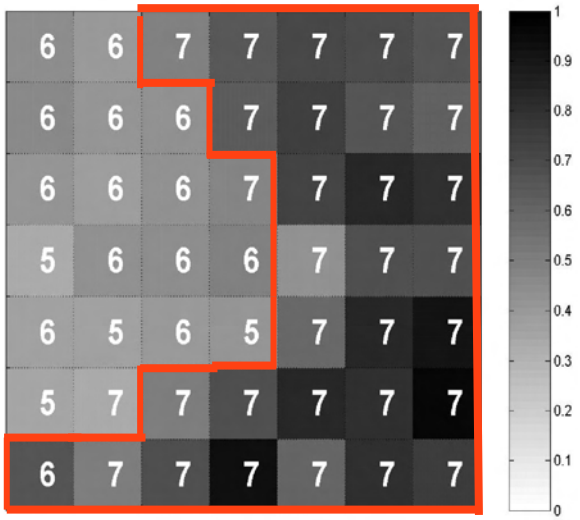
Répartition de la probabilité Maximale



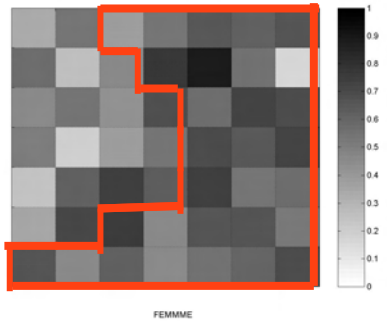
Mort (1)



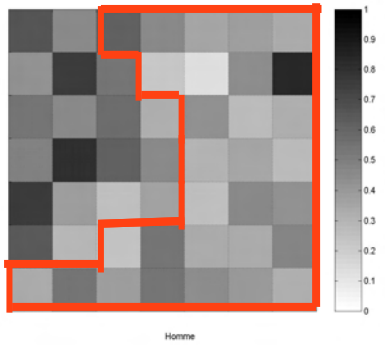
Guerre (1)



Fleur



Femme



Homme

1106 assurés, 9 variables, Bon / Mauvais conducteur

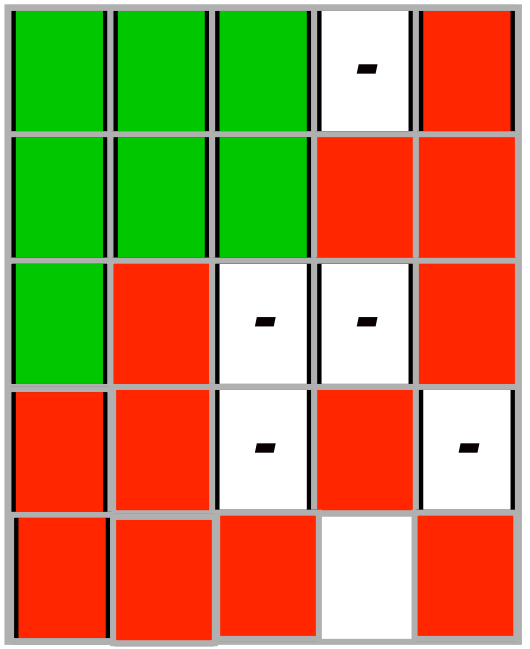
- *Utilité (Privé,Professionnelle)*
- *Sexe (Homme, Femme, Véhicule de Société)*
- *Langue (Français, Autre)*
- *Age (Vieux, Moyen, Jeune)*
- *Localisation (Capitale, Province)*
- *Bonus (1,2)*
- *Police (86, Autre)*
- *Puissance (Grande, Petite)*
- *Age Véhicule (Ancien, Nouveau)*

Visualisation multi-dimensionnelle

H	V	-	J	H	J	F	J	-	J
-	An	-	An	-	An	Pt	Nou	Gr	-
H	-	-	J	H	J	F	J	VS	J
-	-	-	An	Gr	An	Pt	Nou	Gr	Nou
H	V	H	-	VS	J	VS	J	VS	J
Gr	-	Gr	-	Pt	Nou	Pt	Nou	Gr	Nou
-	-	H	-	VS	J	VS	J	VS	J
Gr	-	Gr	-	Pt	An	Pt	An	Gr	Nou
H	M	-	M	F	M	F	M	F	M
Gr	-	-	-	Pt	-	Pt	Nou	Pt	Nou

Représentation des 4 variables

*(**S**exe, **A**ge, **Pui**ssance, **A**ge du **V**éhicule,)*



Affectation avec la probabilités a posteriori p(c/z)

Etiquetage

bon/mauvais

Autres approches et perspectives

- co-Clustering
- Le clustering collaboratif
- Par rapport aux données
  - Graphe
  - Séquences ou données non iid
  - Flux de données…..

Merci

Références

- Pattern Recognition, **S. Theodoridis & K. Koutroumbas, Academic Press, 1999 (ch. 11 - 16)**
- Survey of Clustering Data Mining Techniques, **Pavel Berkhin,**
- Data Clustering and Pattern Recognition Toolbox, *http://fs.mis.kuas.edu.tw/~s1096137135/matlab/dcpr/*
- SOM toolbox (cartes de Kohonen), *http ://www.cis.hut. /projects/somtoolbox/*
- *Spider Toolbox : http://people.kyb.tuebingen.mpg.de/spider/*
- *Gérard Govaert* *http://www.hds.utc.fr/~ggovaert/dokuwiki/doku.php?id=fr:publis*

http://www.google.com/url?sa=t&rct=j&q=&esrc=s&source=web&cd=9&ved=0CF0QFjAI&url=http%3A%2F%2Fwww.sfds.asso.fr%2Fressource.php%3Ffct%3Dddoc%26i%3D382&ei=AB_RToqEJsnjtQbuyLTbDA&usg=AFQjCNFWwWRpOophlgaZ0Seit4zszaXb5A&sig2=kxcom_CGJn5VyPe1VqsuVg

Notion de proximité
(données continues)

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 :
  $d^2(x_1, x_2) = (x_1 - x_2)C^{-1}(x_1 - x_2)'$
  (C=covariance)
- ...