Algorithme de Kohonen : classification et analyse exploratoire des données

Marie Cottrell et Patrick Letremy

SAMOS-MATISSE

CNRS UMR 8595

Université Paris 1- Sorbonne

Analyse de données : introduction

Algorithme de Kohonen

Kohonen et classification : KACP

Traitements des variables qualitatives

Conclusion

Analyse de données : introduction

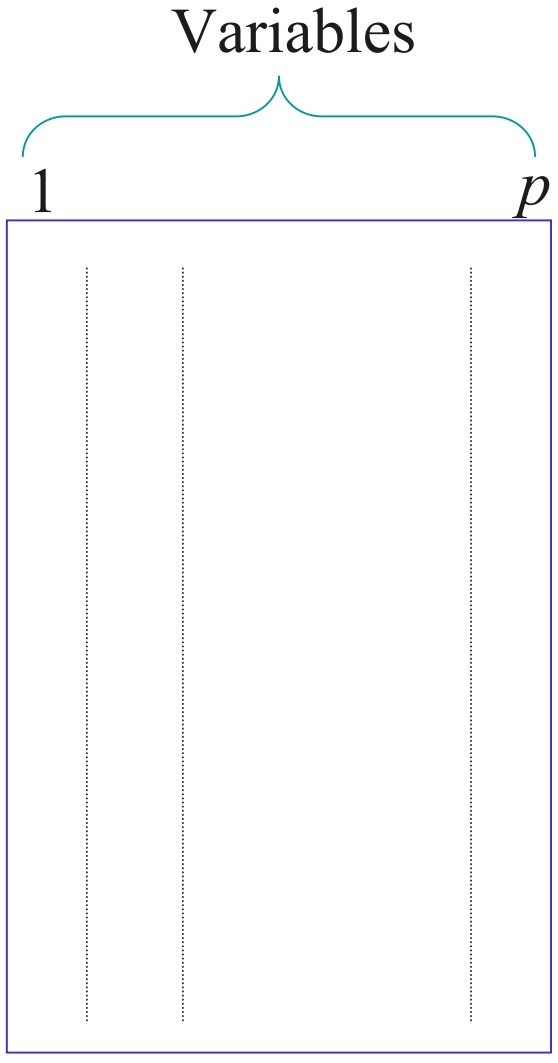
Algorithme de Kohonen

Kohonen et classification : KACP

Traitements des variables qualitatives

Conclusion

Analyse de données, data mining



Comment extraire de l'information ?

N et p sont grands

Extraction d'individus types : Quantification Vectorielle

- K : espace des données, dimension p
- f : densité des données
- $x_1, x_2, \ldots, x_N$, : les données
- n : nombre de classes
- $C_1, C_2, \ldots, C_n$: quantifieurs ou vecteurs codes ou centres
- $A_1, A_2, \ldots, A_n$: classes

BUT : Minimiser la **distorsion quadratique** (l'erreur)
(= **Somme des carrés intra**)

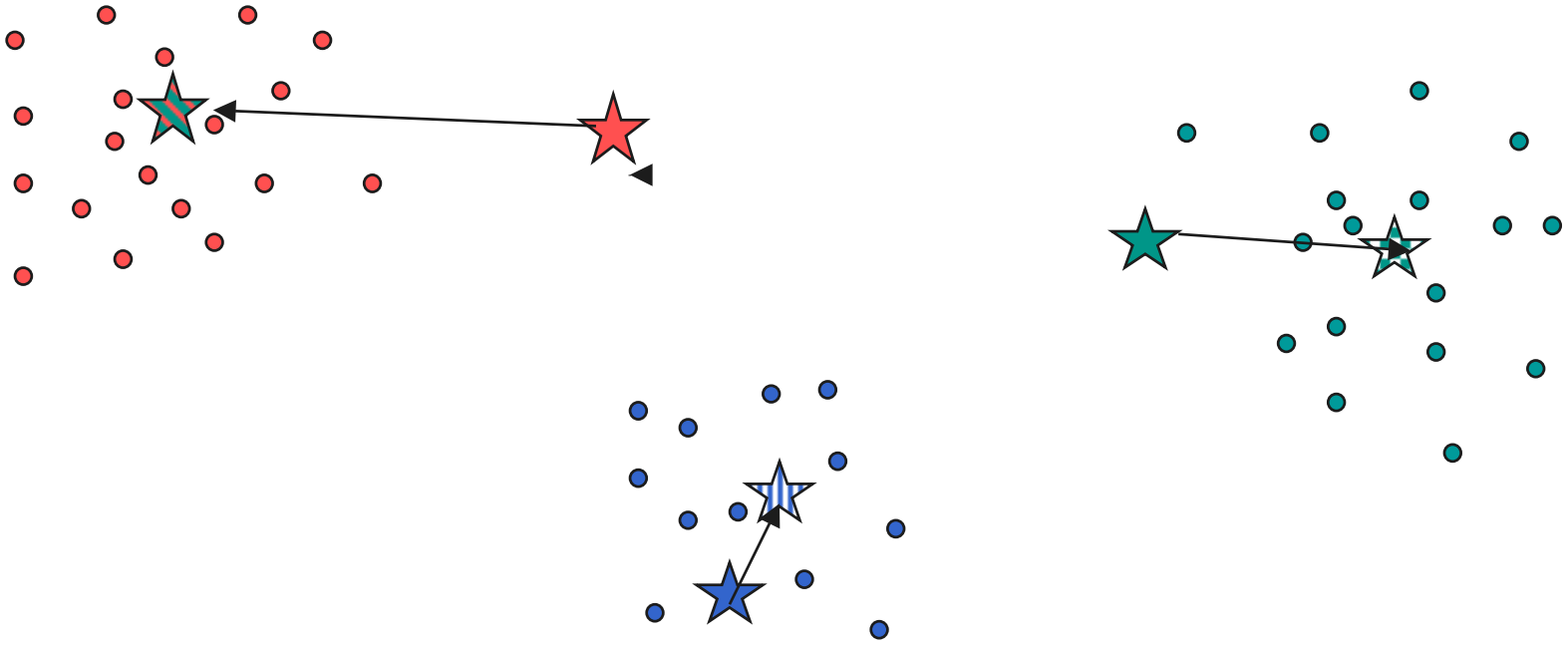
$$D_0(f, C_1, C_2, \Lambda, C_n) = \sum_{i=1}^{n} \int_{A_i} \|x - C_i\|^2 f(x)\, dx \qquad (1)$$

Estimée par

$$\hat{D}_o(f, C_1, C_2, \Lambda, C_n) = \frac{1}{N} \sum_{i=1}^{n} \sum_{x_j \in A_i} \|x_j - C_i\|^2 \qquad (2)$$

Algorithme Déterministe : Centres mobiles (FORGY, LLOYDS, LBG)

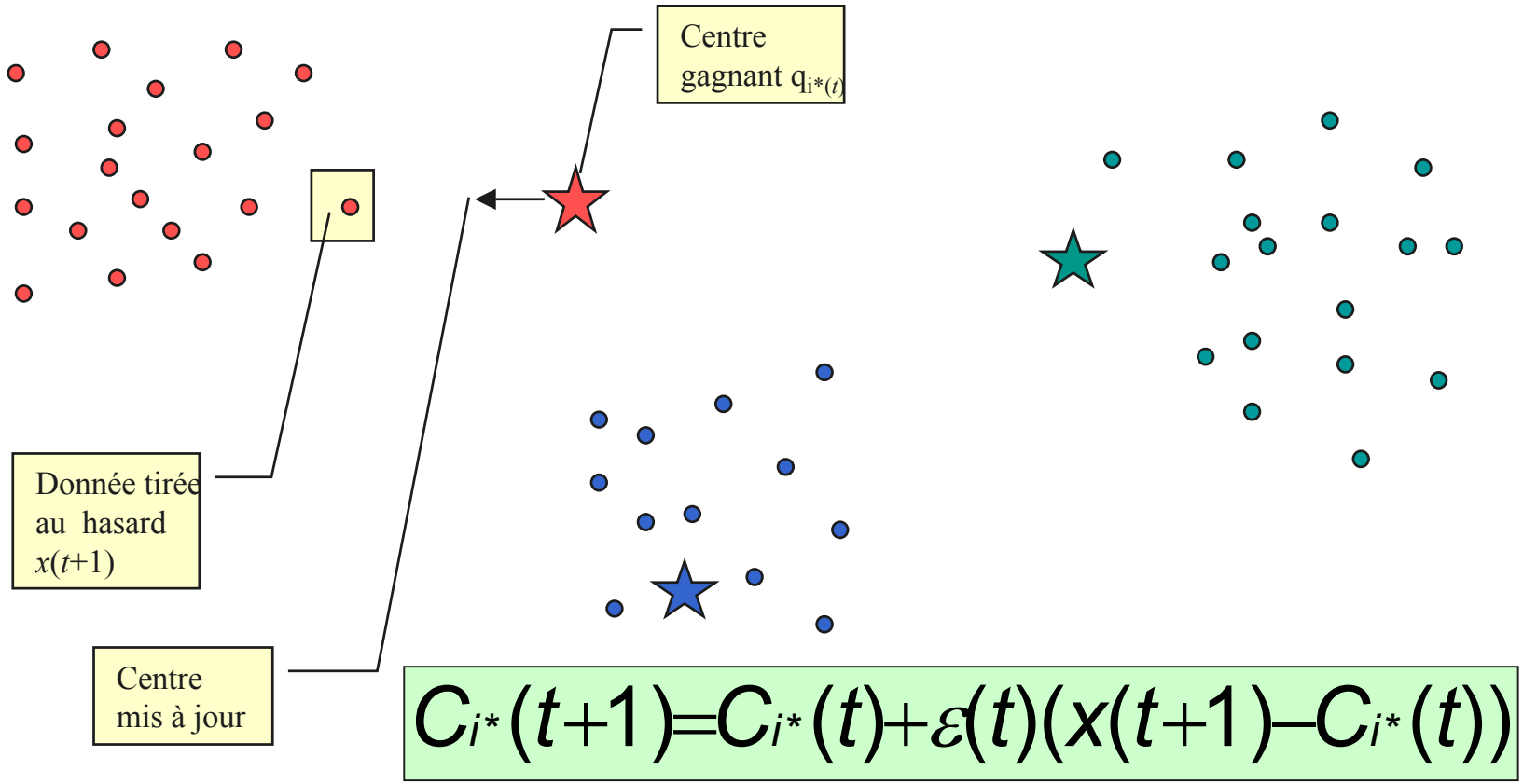
- A chaque étape, les classes sont définies (par les plus proches voisins), et les vecteurs codes sont re-calculés comme les centres de gravité des classes, etc.



- (On part de vecteurs codes aléatoires, on détermine les classes, puis les centres, puis les classes, etc.)

Algorithme Probabiliste associé (SCL)

- On déplace seulement le gagnant



- ***Avec l'algorithme de Kohonen, on déplace le vecteur code gagnant, mais aussi ses voisins.***

Algorithme SCL (0 voisin)

- L'algorithme SCL est la version stochastique de l'algorithme de Forgy
- L'algorithme de Forgy minimise la distorsion et converge vers un minimum local
- L'algorithme SCL converge ***en moyenne vers un minimum local***
- La solution dépend de l'initialisation

Analyse de données : introduction

Algorithme de Kohonen

Kohonen et classification : KACP

Traitements des variables qualitatives

Conclusion

Algorithme de Kohonen (SOM)

- Apprentissage non supervisé
- Les réponses associées à des entrées voisines sont voisines
- On parle d'auto-organisation, de respect de la topologie

- Les associations
  - rétine - cortex visuel
  - fréquences des sons - cortex auditif
  - peau - cortex sensoriel

respectent la notion de voisinage

- Nombreuses applications en représentation de données de grande dimension sur des réseaux de dimension 1 ou 2, ou classification où la notion de classes voisines a un sens

L'algorithme

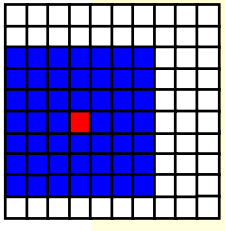
- Il s'agit d'un algorithme original de classification qui a été défini par Teuvo Kohonen, dans les années 80.
- L'algorithme regroupe les observations en classes, en respectant la topologie de l'espace des observations. Cela veut dire qu'on définit a priori une **notion de voisinage entre classes** et que des observations voisines dans l'espace des variables (de dimension p) appartiennent (après classement) à la même classe ou à des classes voisines.

- Les voisinages entre classes peuvent être choisis de manière variée, mais en général on suppose que les classes sont disposées sur une grille rectangulaire qui définit naturellement les voisins de chaque classe.

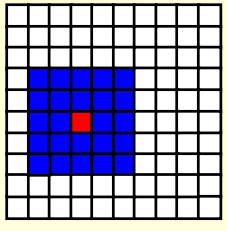
- Mais on peut choisir une autre topologie

Structure en grille ou en ficelle

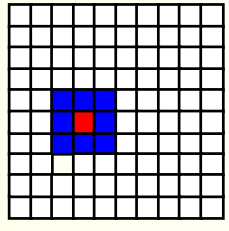
- Les grilles ne sont pas nécessairement carrées



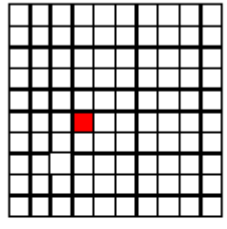
Voisinage de 49



Voisinage de 25



Voisinage de 9



Voisinage de 1

Voisinage de 7

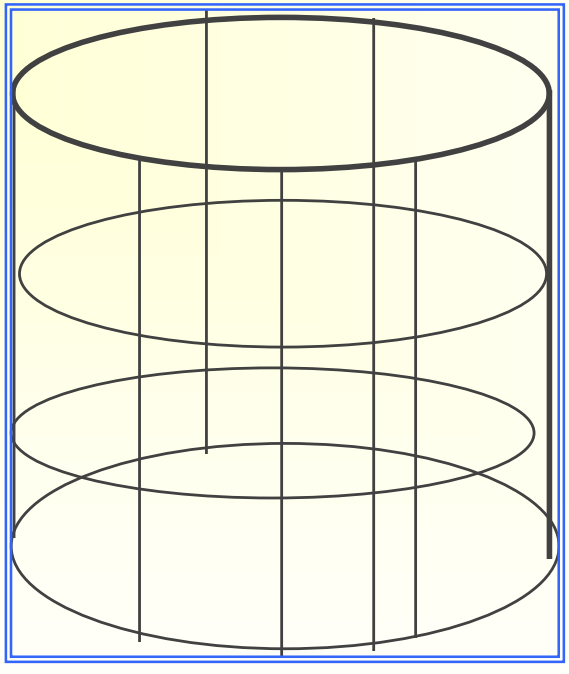
Voisinage de 5

Voisinage de 3

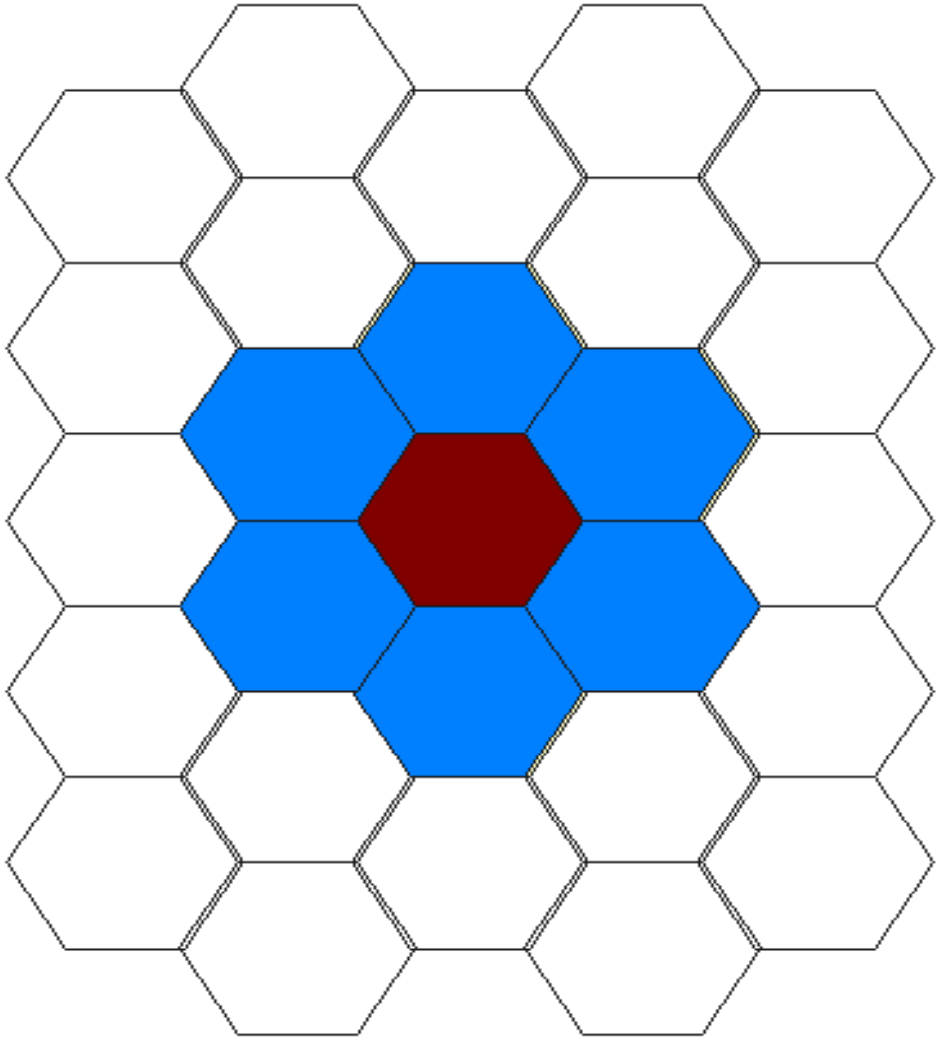
Voisinage de 1

Structure en cylindre

CYLINDRE



Structure hexagonale



L'algorithme

Principe de l'algorithme de Kohonen

- L'algorithme de ***classement*** est **itératif**.
- L'initialisation : associer à chaque classe un vecteur code dans l'espace des observations choisi de manière aléatoire.
- Ensuite, à chaque étape, on choisit une observation au hasard, on la compare à tous les vecteurs codes, et on détermine la **classe gagnante**, c'est-à-dire celle dont le vecteur code est le plus proche au sens d'une distance donnée a priori.
- **On rapproche alors de l'observation les codes de la classe gagnante et des classes voisines**.
- Cet algorithme est analogue à **l'algorithme SCL**, pour lequel on ne modifie à chaque étape que le code de la classe gagnante.
- C 'est aussi un ***algorithme compétitif***

Notations (Kohonen, ou SOM)

- Espace des entrées K dans R^p
- n unités, rangées en réseau de dimension 1 ou 2, pour lesquelles est défini un système de voisinage
- A chaque unité i (i=1,...,n), est associé un ***vecteur code*** C_i de p composantes

- La réponse d'une unité i à l'entrée x est mesurée par la proximité de x avec le vecteur code C_i

- Initialisation aléatoire des vecteurs codes
- A l'étape t,
  - **on présente une entrée x**
  - **on cherche l'unité gagnante $i_0(x)$**
  - **on rapproche C_{i0} et les C_i voisins, de l'entrée x**

Définition de l'algorithme on-line

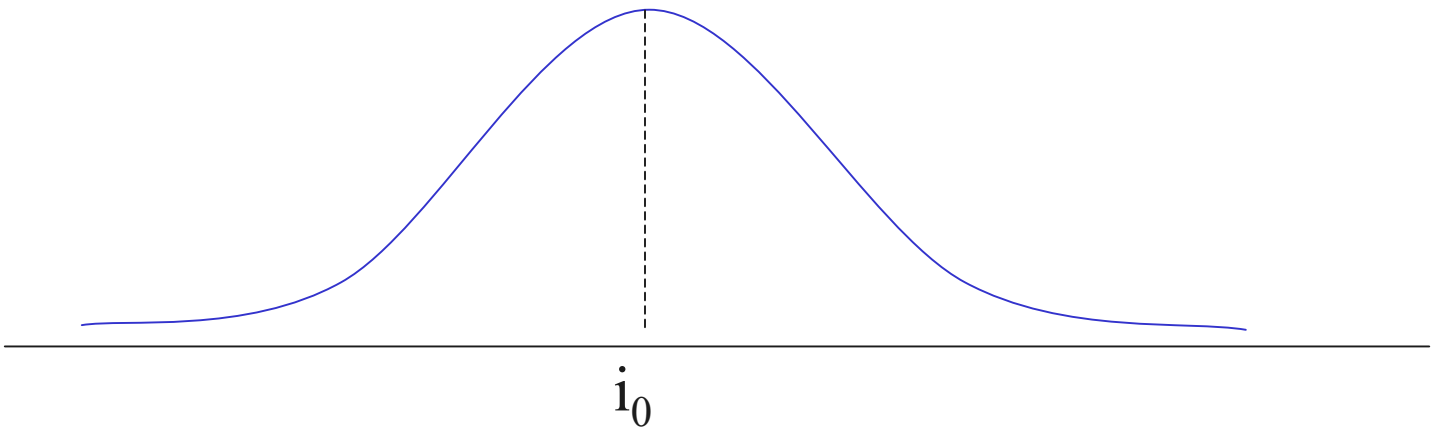
- Les $\{C_i(0)\}$ sont les vecteurs codes initiaux de dimension p
- $\varepsilon(t)$ est le **paramètre d'adaptation**, positif, <1, constant ou lentement décroissant
- La **fonction de voisinage** $\sigma(i,j)$=1 ssi i et j sont voisins, =0 sinon, la taille du voisinage décroît aussi lentement au cours du temps
- Deux étapes : au temps t+1, on présente $x(t+1)$, (tirages indépendants)
  - On détermine l'unité gagnante

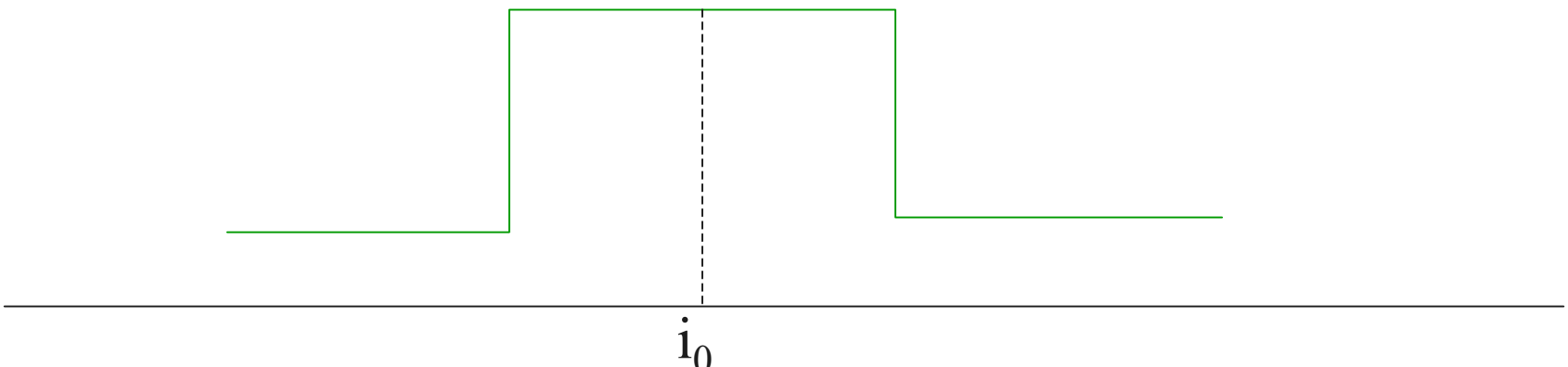
$$i_0(t+1)=\arg\min_i \left\| x(t+1)-C_i(t) \right\|$$

  - On met à jour les vecteurs codes

$$C_i(t+1)=C_i(t)+\varepsilon(t+1)\,\sigma(i_0(t+1),i)\,(x(t+1)-C_i(t$$

Fonctions de voisinage σ

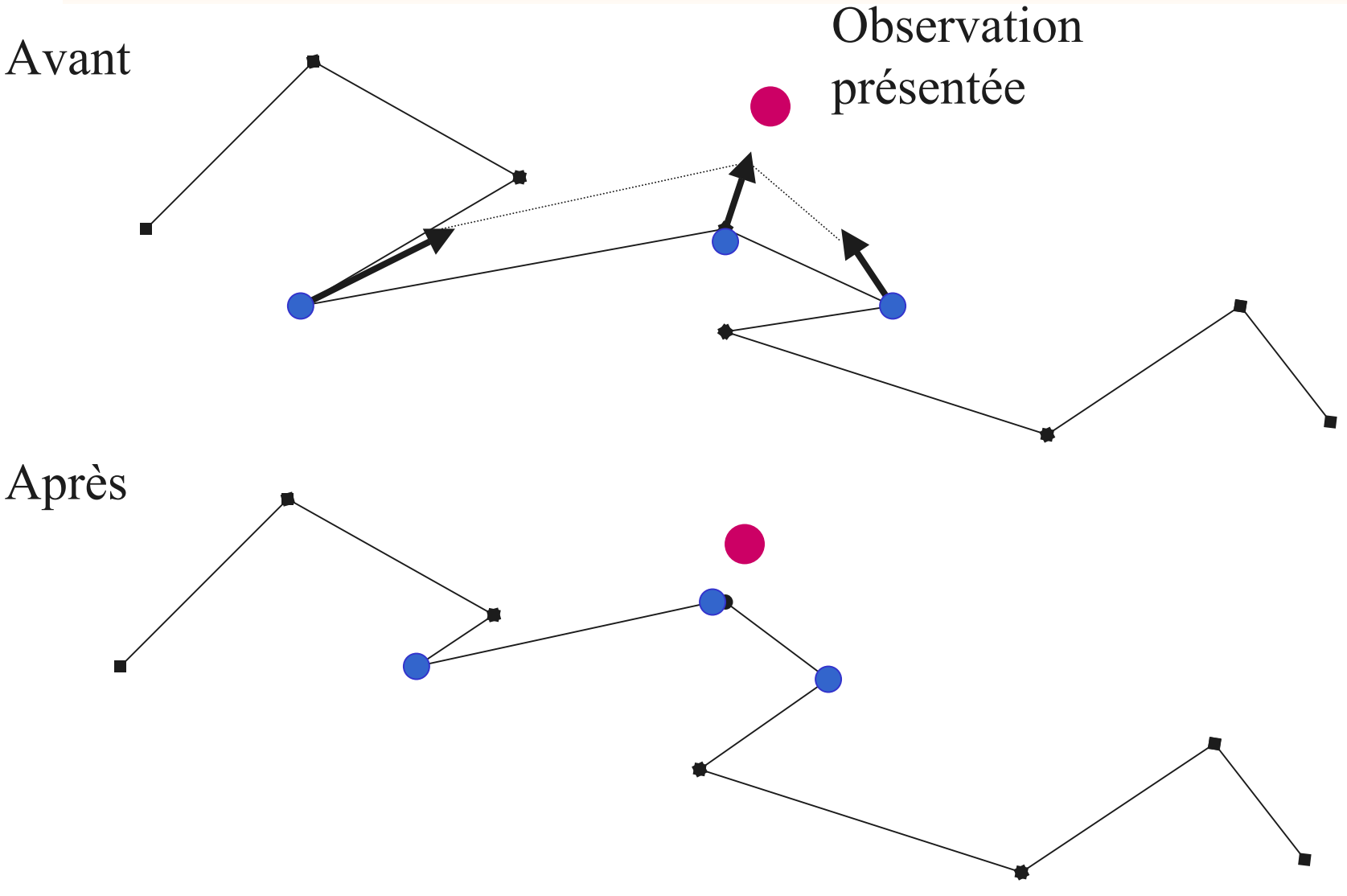




Kohonen / SCL

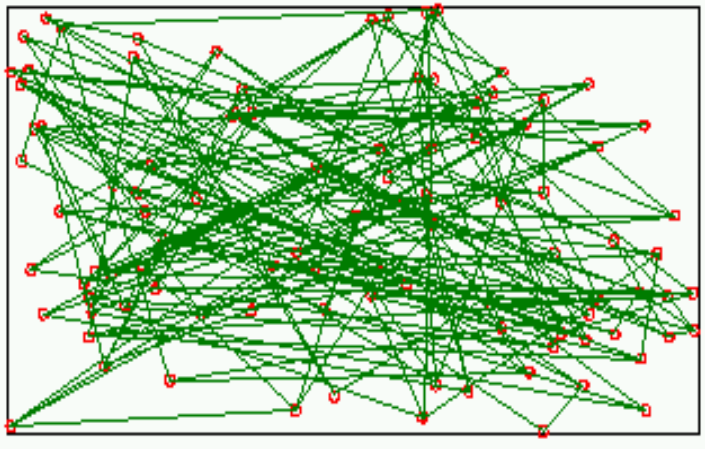
- En fait l'algorithme de Kohonen est une extension de la version stochastique de l'algorithme des centres mobiles
- Issu du domaine de la quantification vectorielle, de la théorie du signal
- Applications où les données sont très nombreuses, disponibles on-line,
- Pas besoin de les stocker

Exemple : une étape

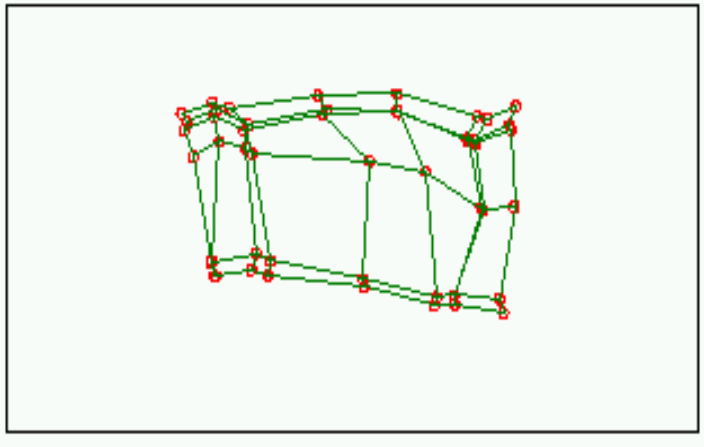


Démo en dimension 2

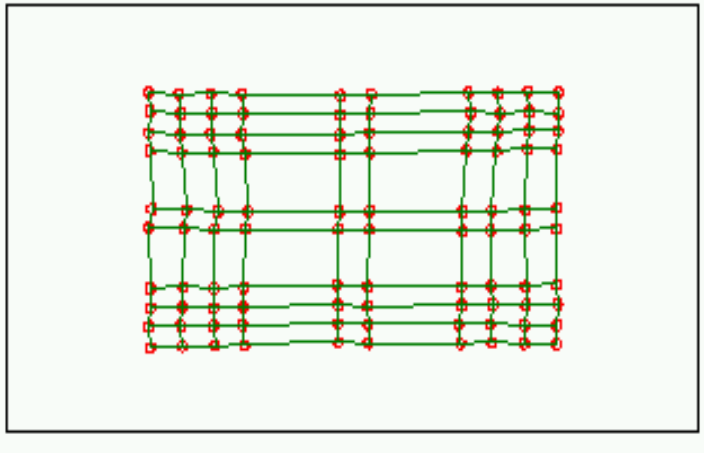
Grille:10x10 Etape= 0/1000000 Rayon= 0



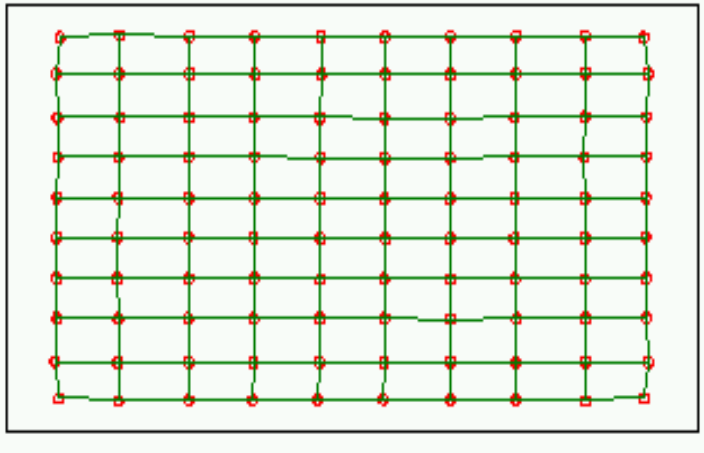
Grille:10x10 Etape= 1000/1000000 Rayon= 5



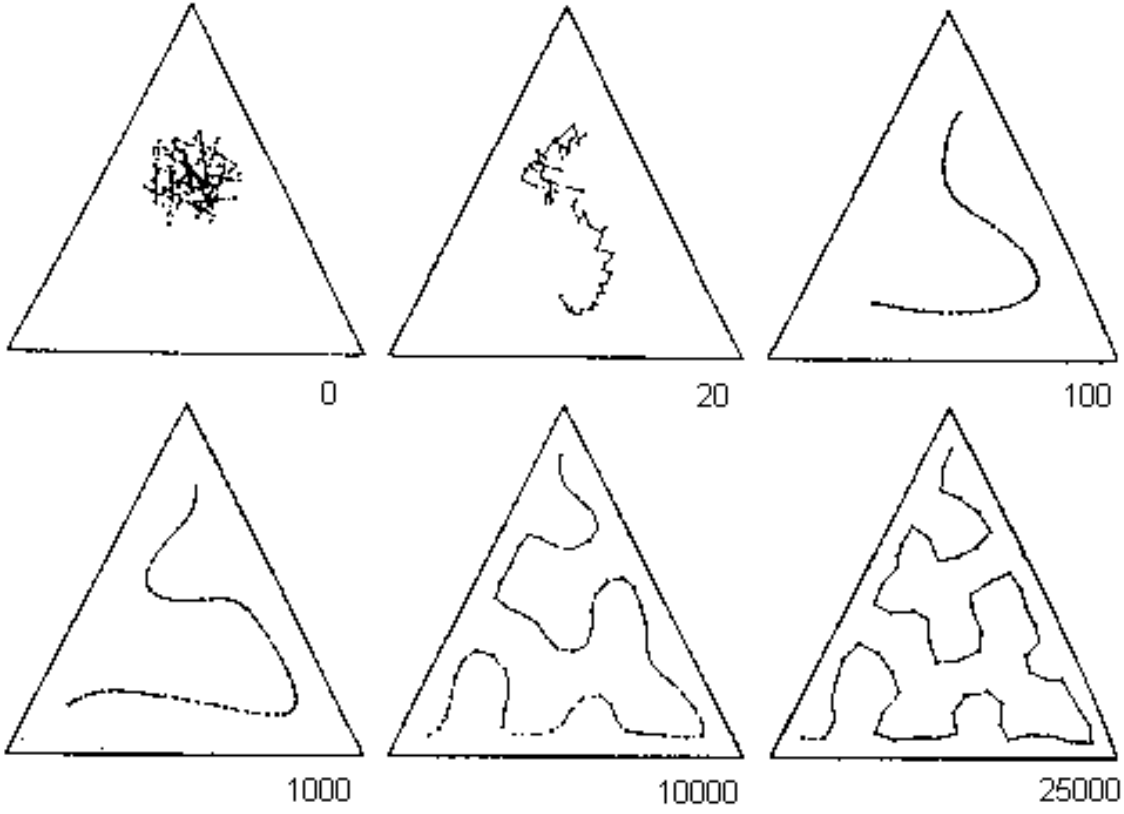
Grille:10x10 Etape= 100000/1000000 Rayon= 3



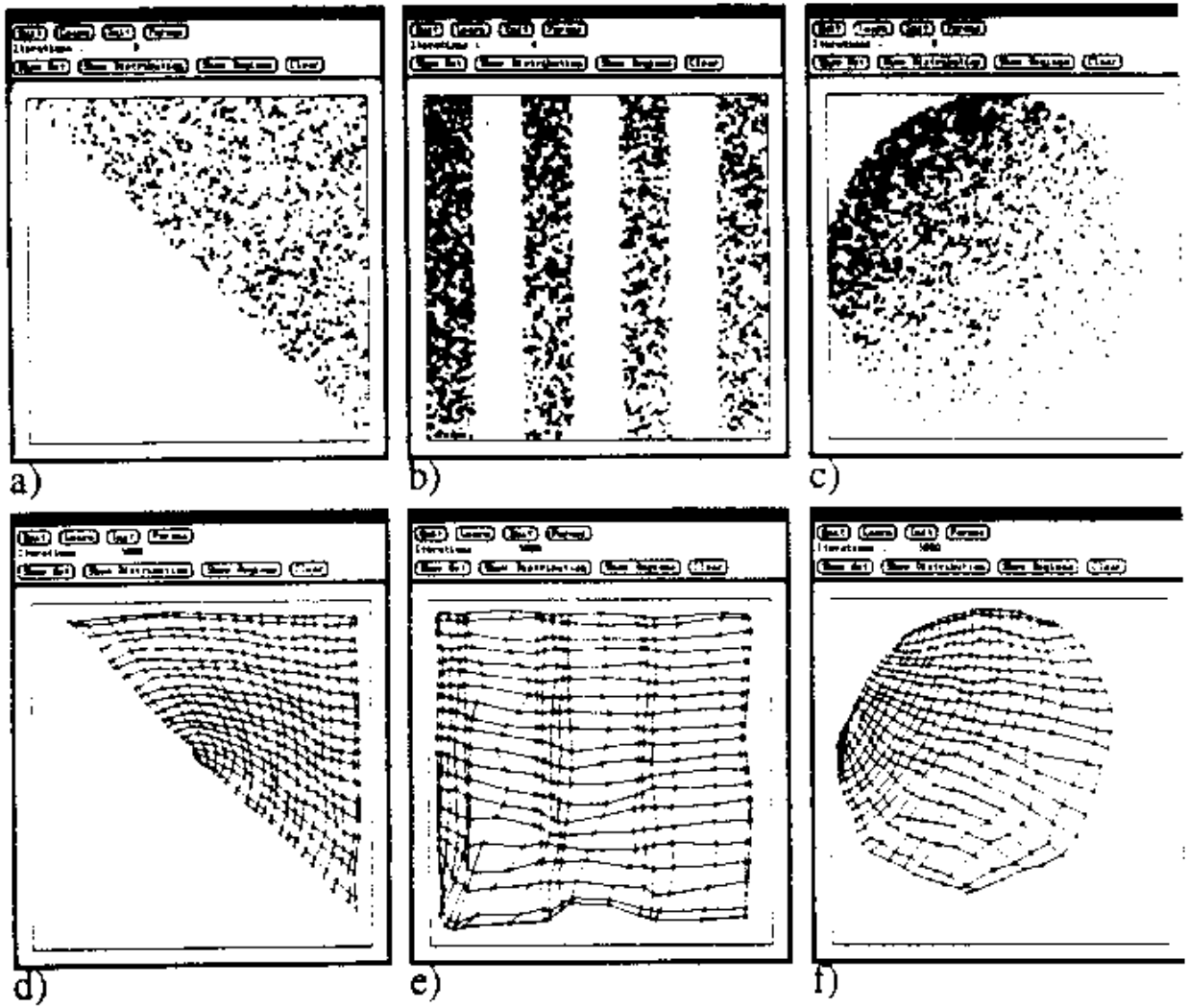
Grille:10x10 Etape= 1000000/1000000 Rayon= 0



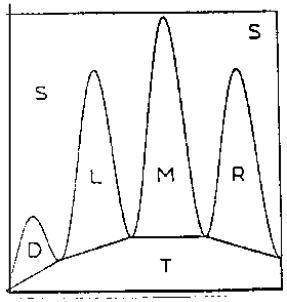
Exemples de simulations (Kohonen)



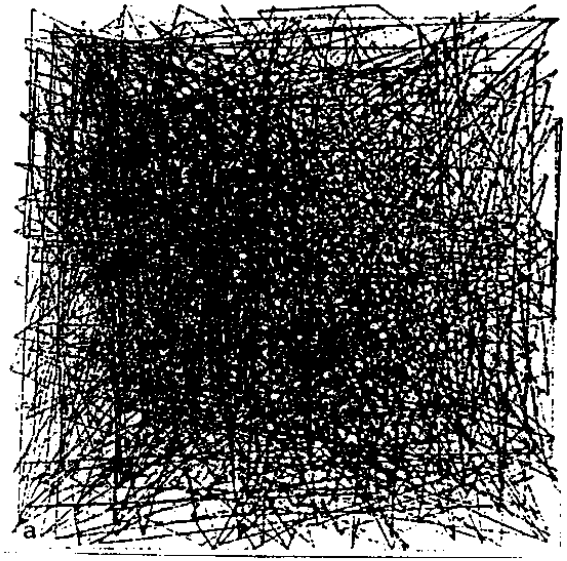
Exemples de simulations (EPFL)



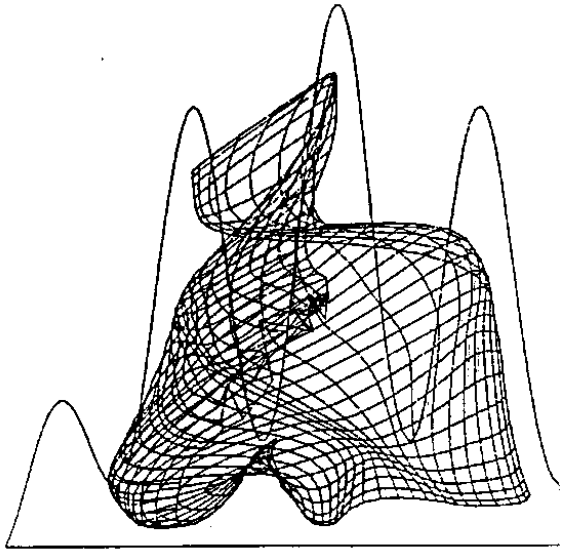
Adaptatif (Ritter et Schulten)



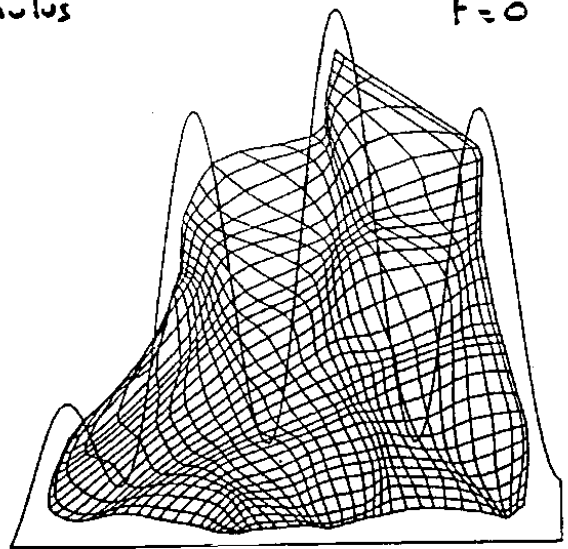
Ensemble de Présentation du Stimulus

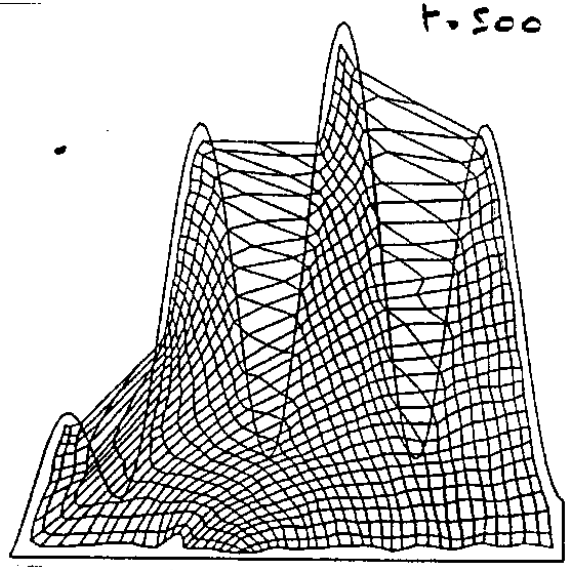


t=0

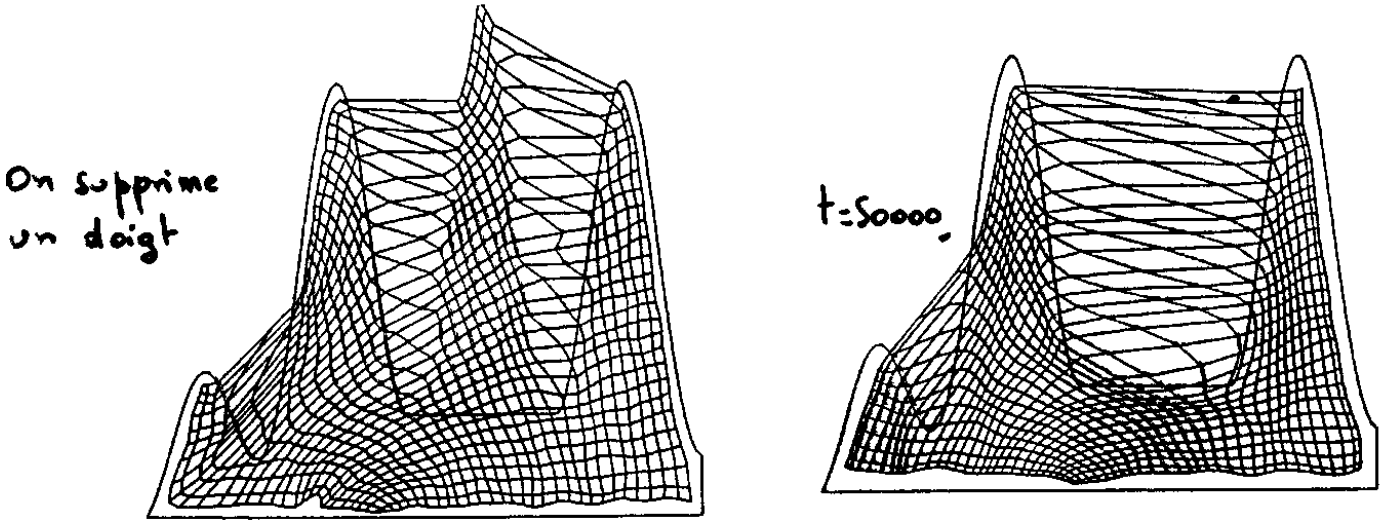


t=500





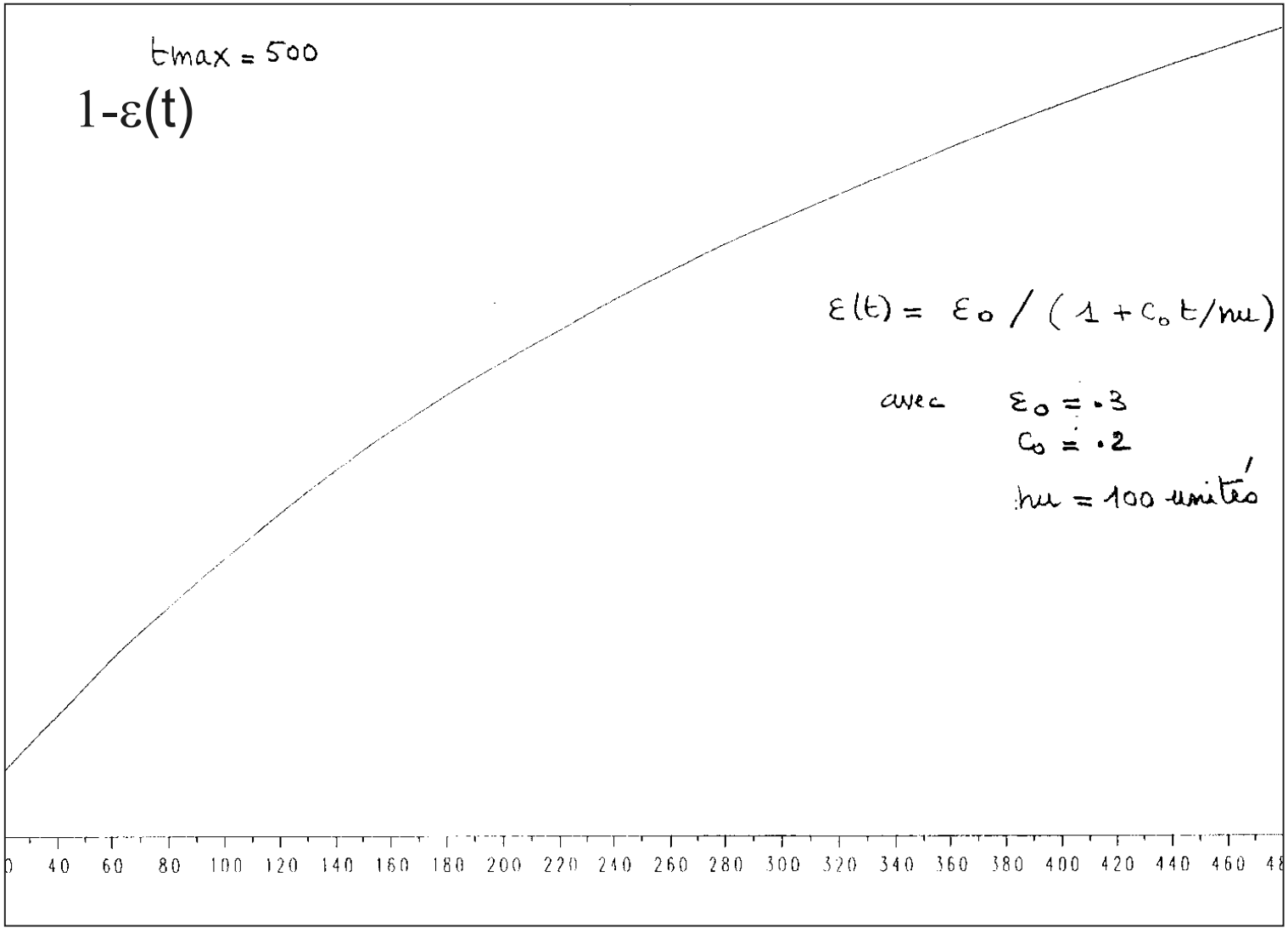
Adaptatif (Ritter et Schulten)



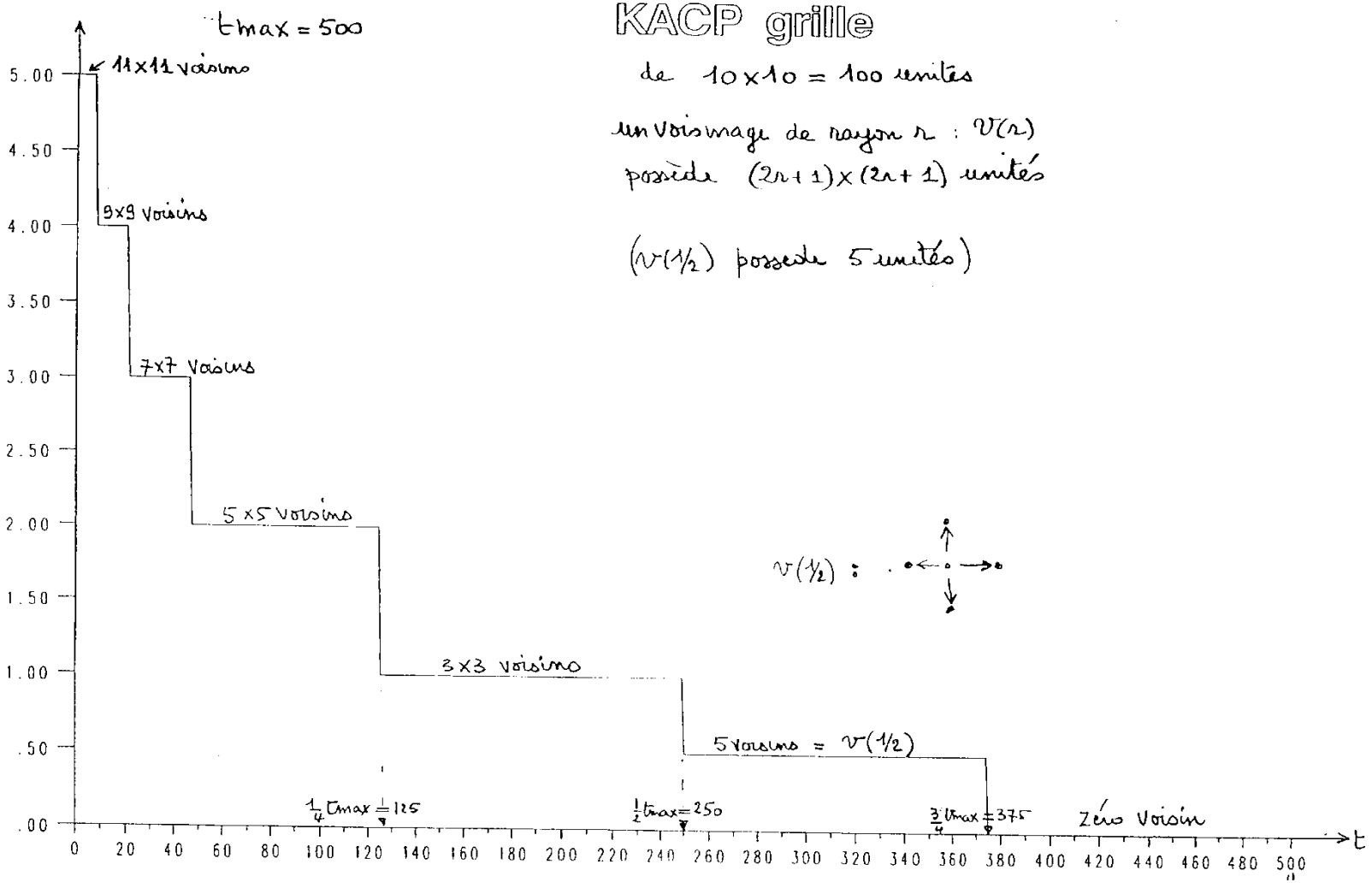
Compléments divers

Exemples et illustrations

Fonction epsilon (Letremy)



Fonction voisinage (Letremy)



KACP grille

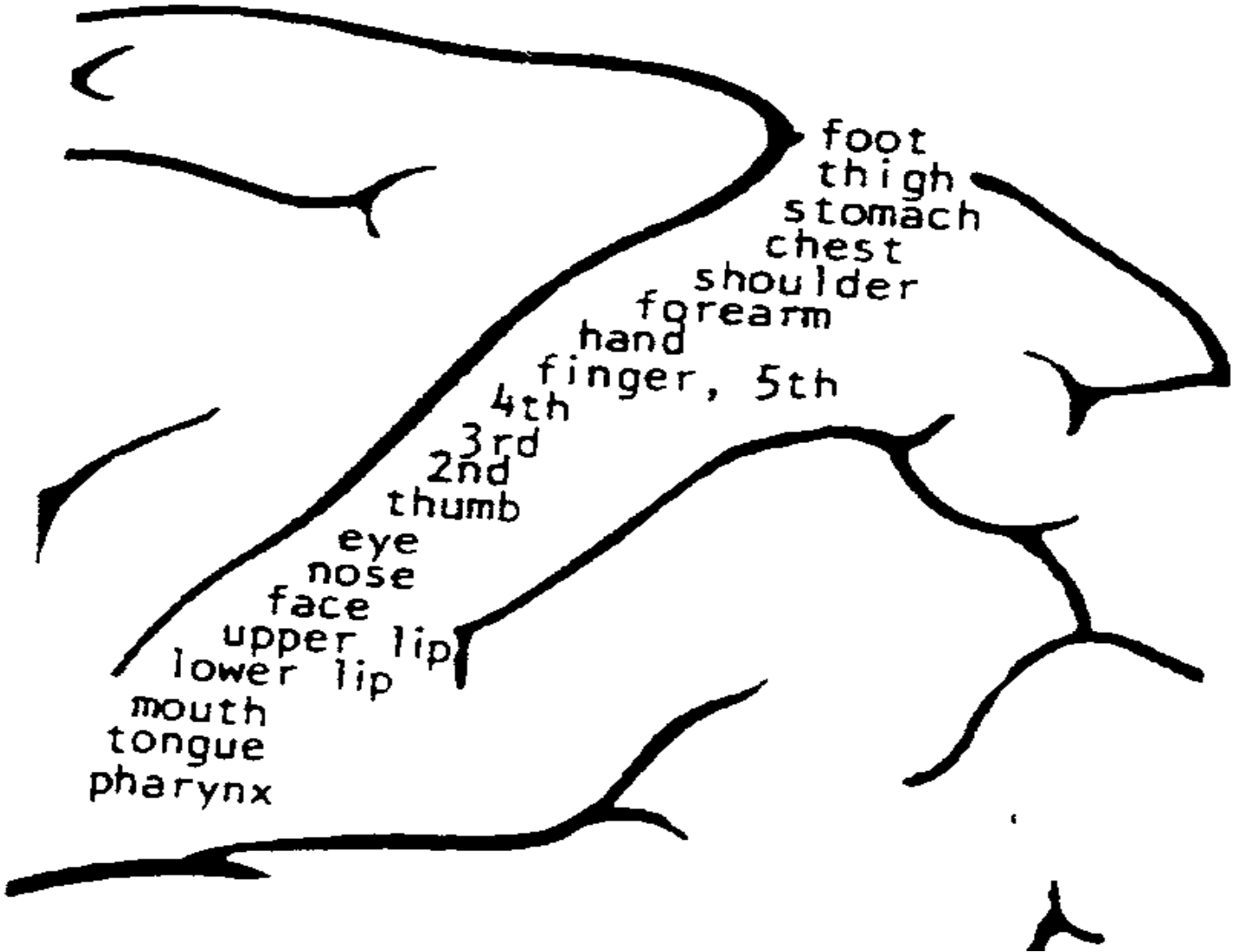
de 10x10 = 100 unités

un voisinage de rayon r : V(r)
possède (2r+1)x(2r+1) unités

(V(1/2) possede 5 unités)

v(1/2) :

Cortex sensoriel (Kohonen)



Cortex sensoriel

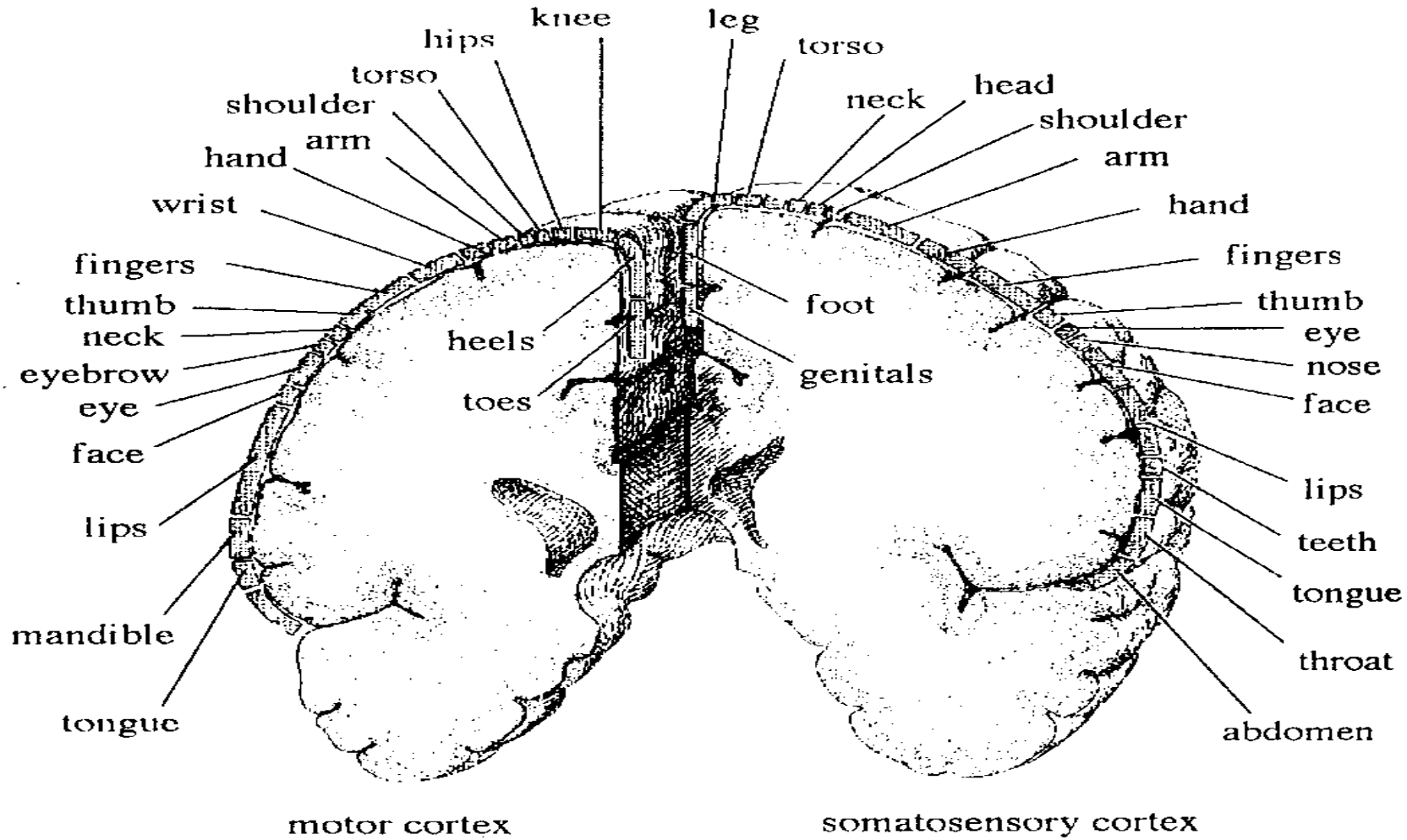
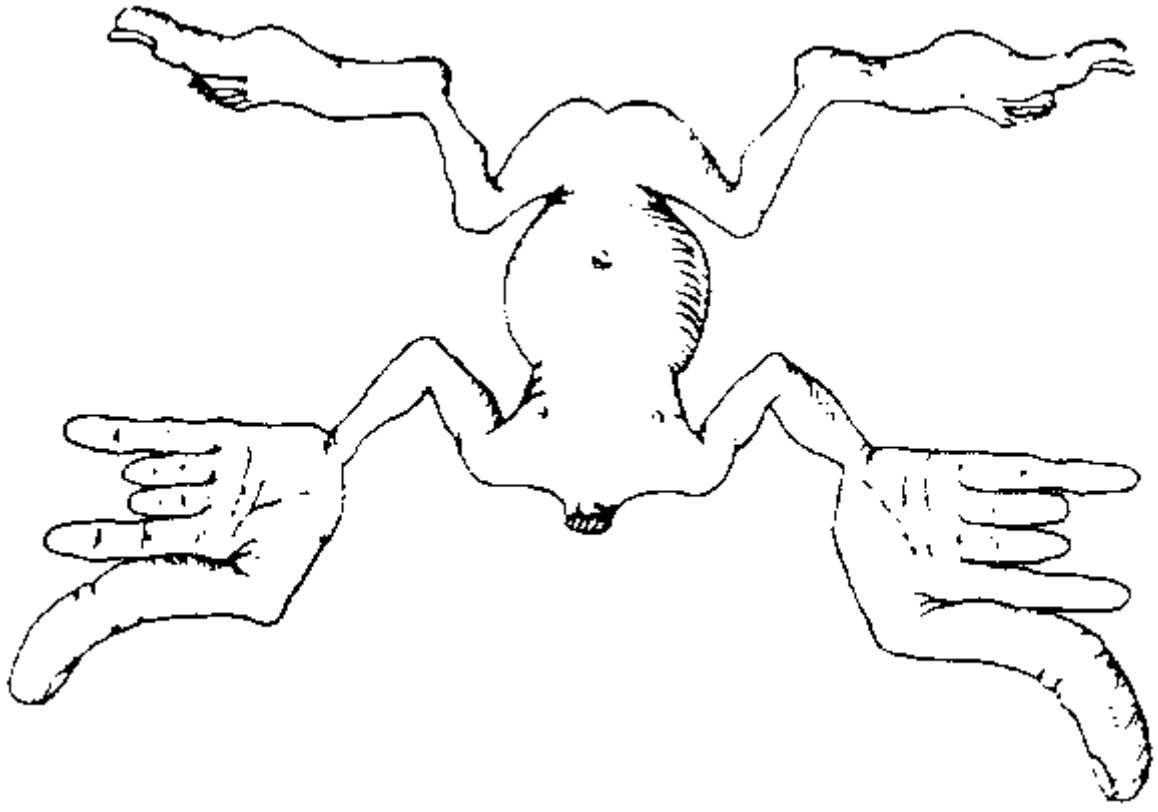
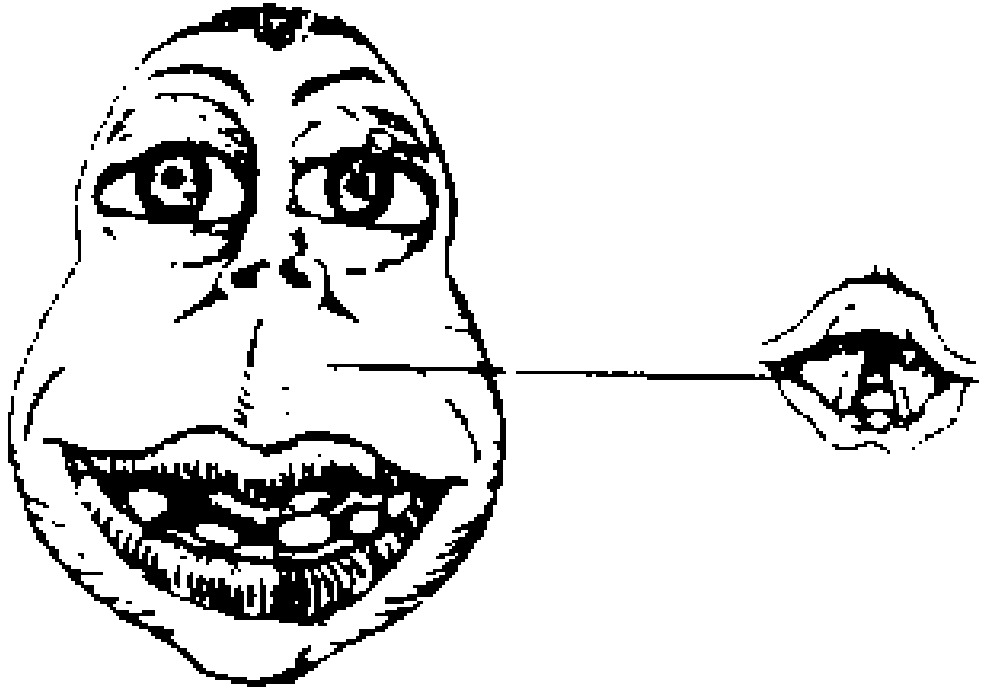


Fig. 15.3. The somatosensory and motor cortex

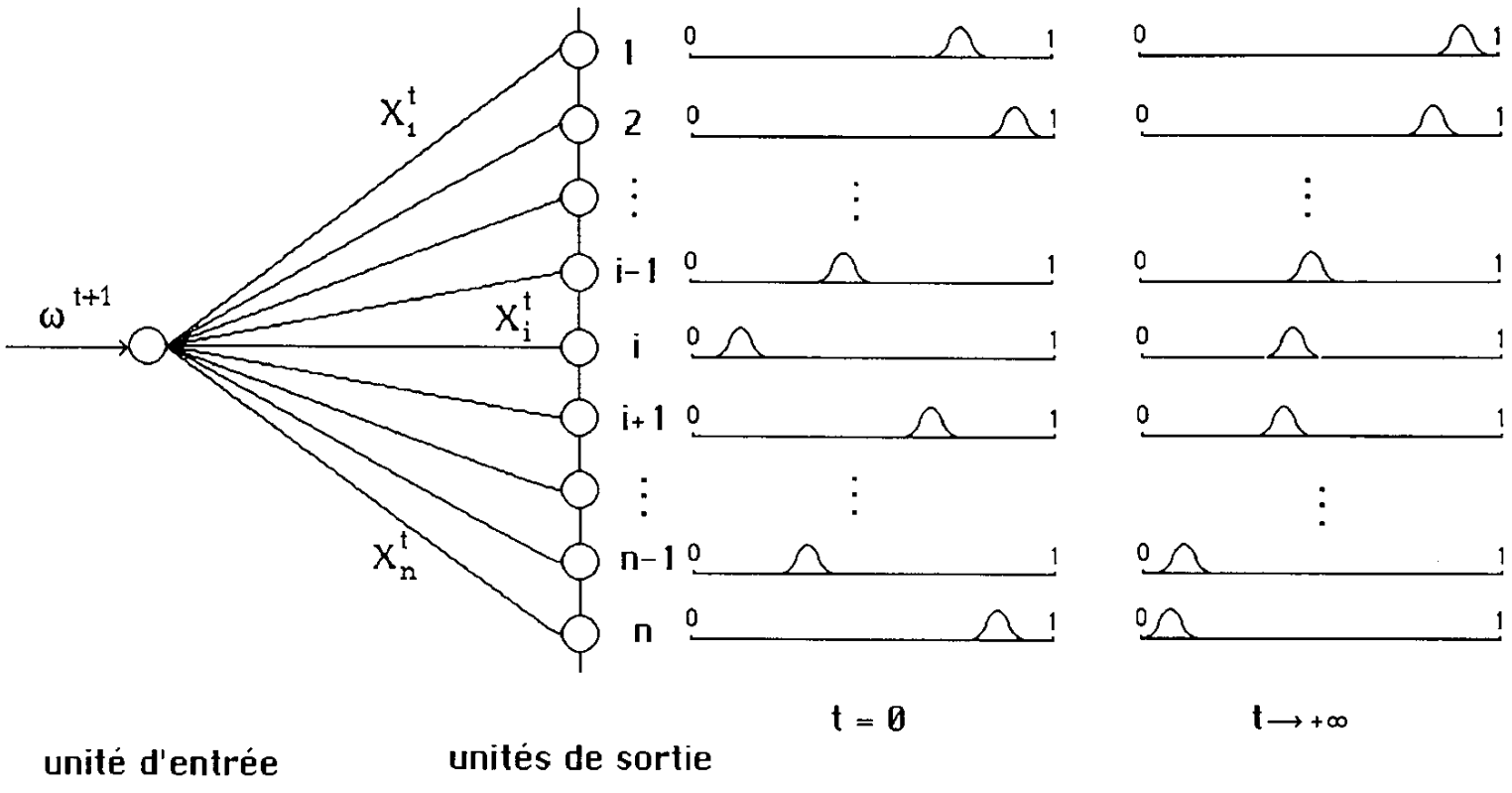
Homonculus (Anderson, Penfield and Boldrey)

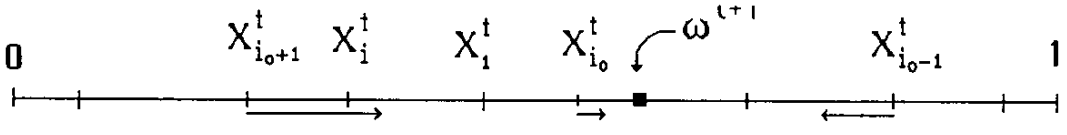


Tête d’homonculus (Anderson, Penfield and Boldrey))



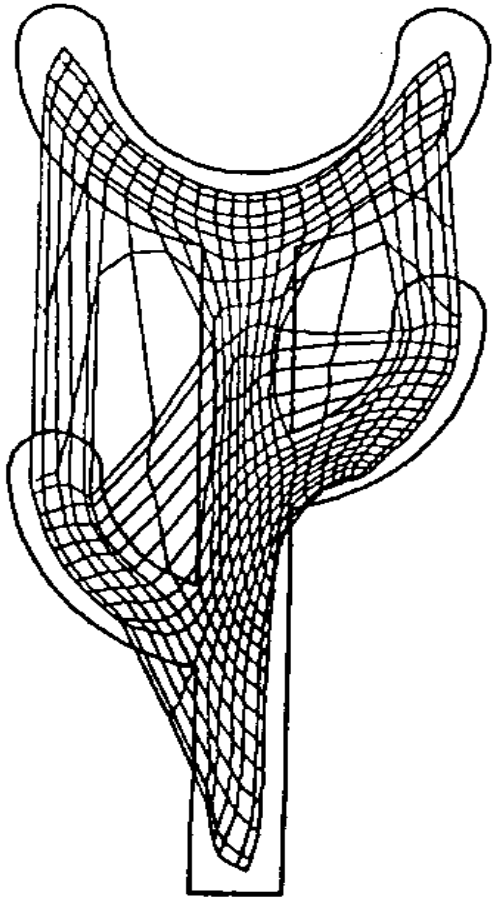
Cortex auditif (Pagès et Fort)





représentation des poids dans l'espace des stimuli (d=1)

En 3 D (Kohonen)



Sélection de dimension (Kohonen)

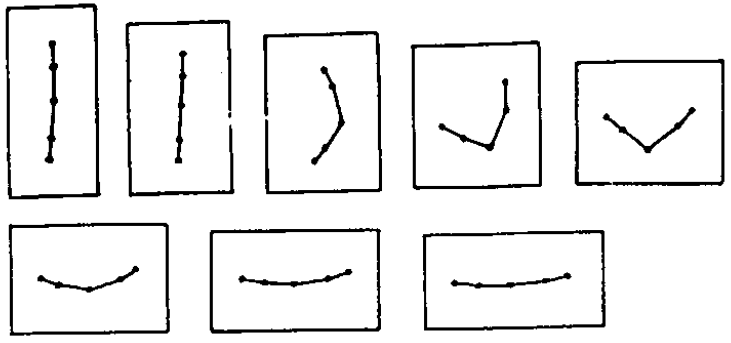


Fig. 5.29. Automatic selection of dimensions for ma ping

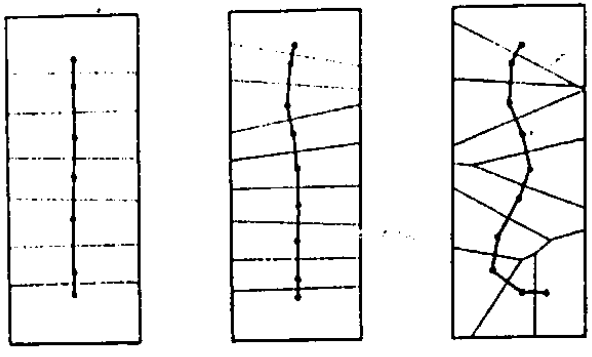
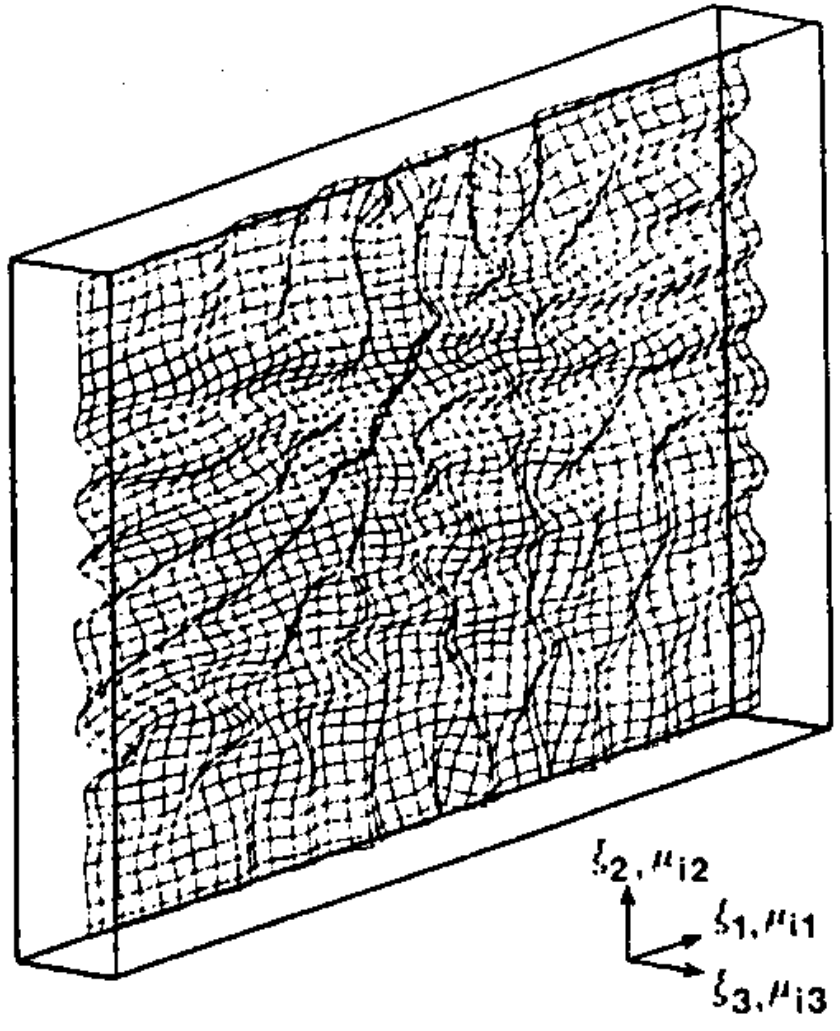
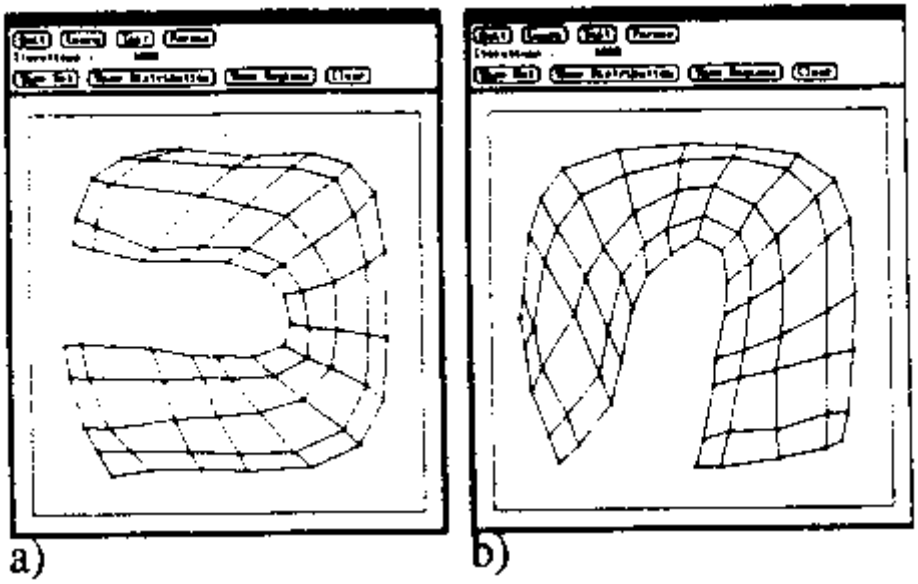


Fig. 5.30. Distribution of weight vectors with different lengths of a linear array

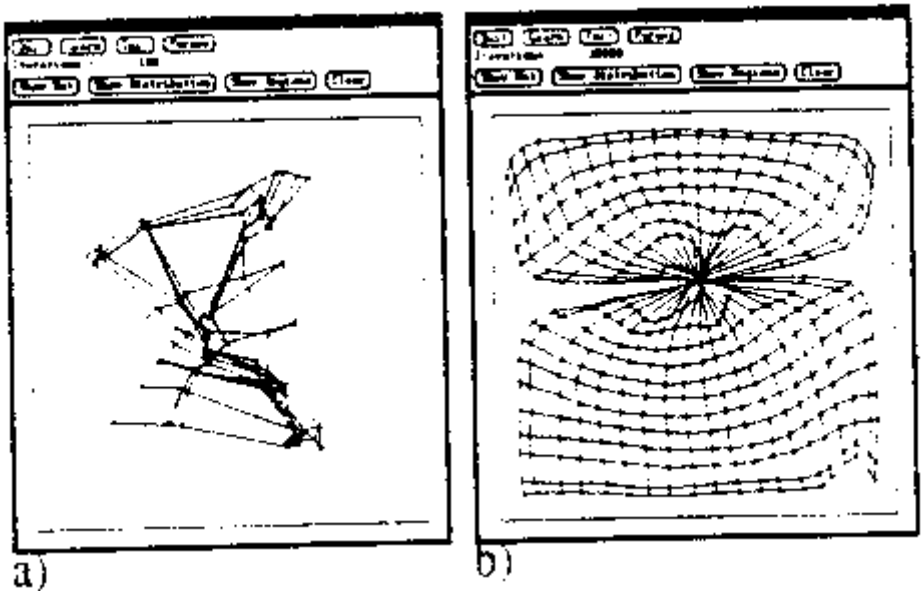
Sélection de dimension (Kohonen)



Problème de dimension (EPFL)



Problème de papillon



Étude Théorique

- On peut écrire

$$C(t+1) = C(t) + \varepsilon H(x(t+1), C(t))$$

- Algorithme dont la forme fait penser à un algorithme de gradient
- Mais en général (si la distribution des entrées est continue), **H ne dérive pas d'un potentiel** (Erwinn). L'algorithme on-line SOM n'est pas un algorithme de gradient.
- **On se restreint ici au cas où les entrées sont listées en nombre fini. Alors, il existe une fonction potentiel** qui est (cf Ritter et al. 92) la somme des carrés intra classes étendue
- **Dans ce cas, l'algorithme minimise la somme des carrés des écarts de chaque observation non seulement à son vecteur code, mais aussi aux vecteurs codes voisins (dans la structure fixée)**

Somme des carrés intra (rappel)

- L'algorithme SCL (0-voisin) est exactement l 'algorithme de gradient stochastique associé à la distorsion quadratique (ou somme des carrés intra)

$$D(x)=\sum_{i\in I}\int_{A_{i(x)}}\|x-C_i\|^2 f(x)dx$$

- estimée par

$$\hat{D}(x)=\frac{1}{N}\sum_{i=1}^{n}\sum_{x\in A_i}\|x-C_i\|^2$$

Somme des carrés intra-classes étendue aux classes voisines

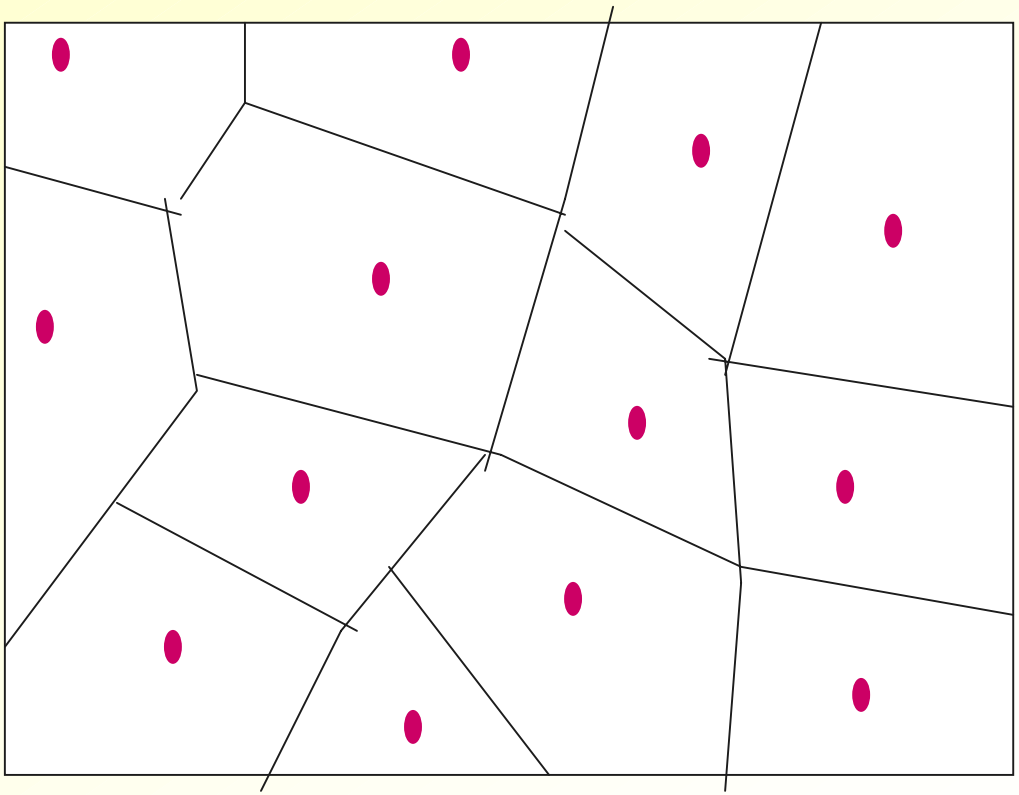
- **Extension de la notion de somme des carrés intra-classes, qui est étendue aux classes voisines**

$$D_{SOM}(x) = \sum_{i=1}^{n} \sum_{\substack{x \text{ t.q. } i = i_0(x) \\ \text{ou } i \text{ voisin de } i_0(x)}} \|x - C_i\|^2$$

- En fait cette fonction a de nombreux minima locaux
- L'algorithme converge, moyennant les hypothèses classiques (Robbins-Monro) sur les ε, qui doivent décroître ni trop, ni trop peu
- ***La démonstration mathématique complète n'est faite que pour des données de dimension 1 et pour une structure de voisinage en ficelle***
- Pour accélérer la convergence, on prend au début une taille de voisinage assez grande et on la fait décroître

Mosaïque de Voronoï

- *Dans l ’espace des entrées, les classes forment une partition, ou mosaïque de Voronoï, dépendant des C.*
- $A_i(C) = \{x \ / \ ||C_i - x|| = \min_j || C_j - x || \}$: *i*-ème classe formée des données pour lesquelles $C(i)$ est le vecteur code gagnant.



ODE associée

- On peut écrire l'équation différentielle ordinaire associée à l 'algorithme

$$\frac{dC(i,u)}{du} = -\sum_{j \in I} \sigma(i,j) \int_{A_j(C(.,u))} (C(i,u) - x) f(x) dx$$

- où $C(i,t)$ est pour $C_i(t)$
- $C(.,t)$ for $(C_i(t), i \in I)$
- f est la densité des données x

Points fixes de l'ODE

- Si l'algorithme converge, il doit converger vers un équilibre de l 'ODE

$$\forall i \in I, \sum_j \sigma(i,j) \int_{A_j(C^*)} (C_i^* - x) f(x) dx = 0$$

- i.e.

$$C_i^* = \frac{\sum_j \sigma(i,j) \int_{A_j(C^*)} f(x) dx}{\sum_j \sigma(i,j) \; P(A_j(C^*))} \qquad (1)$$

- Pour chaque i, C^*_i est le barycentre des toutes les classes, pondérées par les valeurs de la fonction $\sigma(i,j)$, $j \in I$, (barycentre de la réunion de sa classe et des classes voisines)

L'algorithme batch

- On définit un algorithme déterministe pour calculer les solutions C^*
- On part de $C(0)$ et on définit pour chaque composante i

$$C_i^{k+1} = \frac{\sum_j \sigma(i,j) \int_{A_j(C^k)} x \ f(x) dx}{\sum_j \sigma(i,j) P(A_j(C^k))}$$

- Quand il n'y a qu'un nombre fini de données (c'est le cas en analyse de données), le processus déterministe s'écrit :

$$C_{i,N}^{k+1} = \frac{\sum_j \sigma(i,j) \sum_{l=1}^{N} x_l \mathbf{1}_{A_j(C^k)}(x_l)}{\sum_j \sigma(i,j) \sum_{l=1}^{N} \mathbf{1}_{A_j(C^k)}(x_l)} \quad (2)$$

- C'est exactement une extension de l'algorithme de Forgy, où les centres de gravité se calculent sur les réunions de classes voisines

L'algorithme batch

- Si $N \to \infty$, si on pose

$$\mu_N = \frac{1}{N} \sum_{l=1}^{N} \delta_{x_l}$$

- si μ_N converge faiblement vers la loi des données,

on a

$$\lim_{N \to \infty} \lim_{k \to \infty} C_{i,N}^{k+1} = C_i^*$$

où C^* est une solution de (1)

- L'algorithme (2) est l'algorithme Kohonen batch. C'est une extension de l'algorithme de Forgy. A chaque étape, la mise à jour consiste à calculer les centres de toutes les classes pondérés par la fonction voisinage.

Algorithme Quasi-Newtonien

- Même si D_{SOM} n'est pas partout différentiable et ne permet pas d'apporter les arguments rigoureux de convergence de l'algorithme stochastique on-line, il est intéressant de contrôler ses variations au cours des itérations.
- L'algorithme Kohonen batch peut s 'écrire approximativement

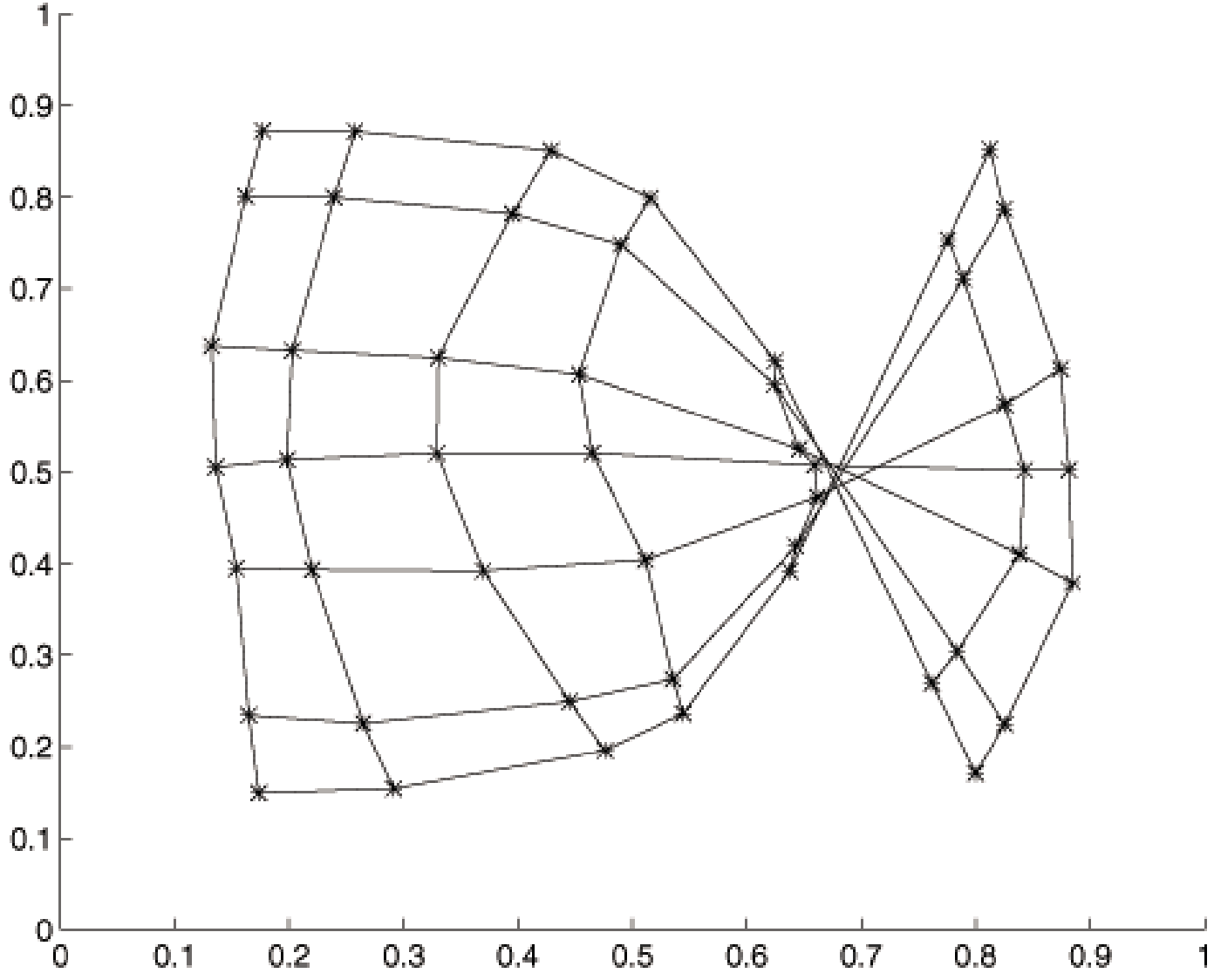
$$C_N^{k+1} = C_N^k - diag\nabla^2 D_{SOM}(C_N^k))^{-1}\nabla D_{SOM}(C_N^k$$

c'est-à-dire que l'algorithme batch est un algorithme quasi-Newtonien associé à la distorsion étendue (si et seulement si il n 'y a pas de données sur les frontières de classes)

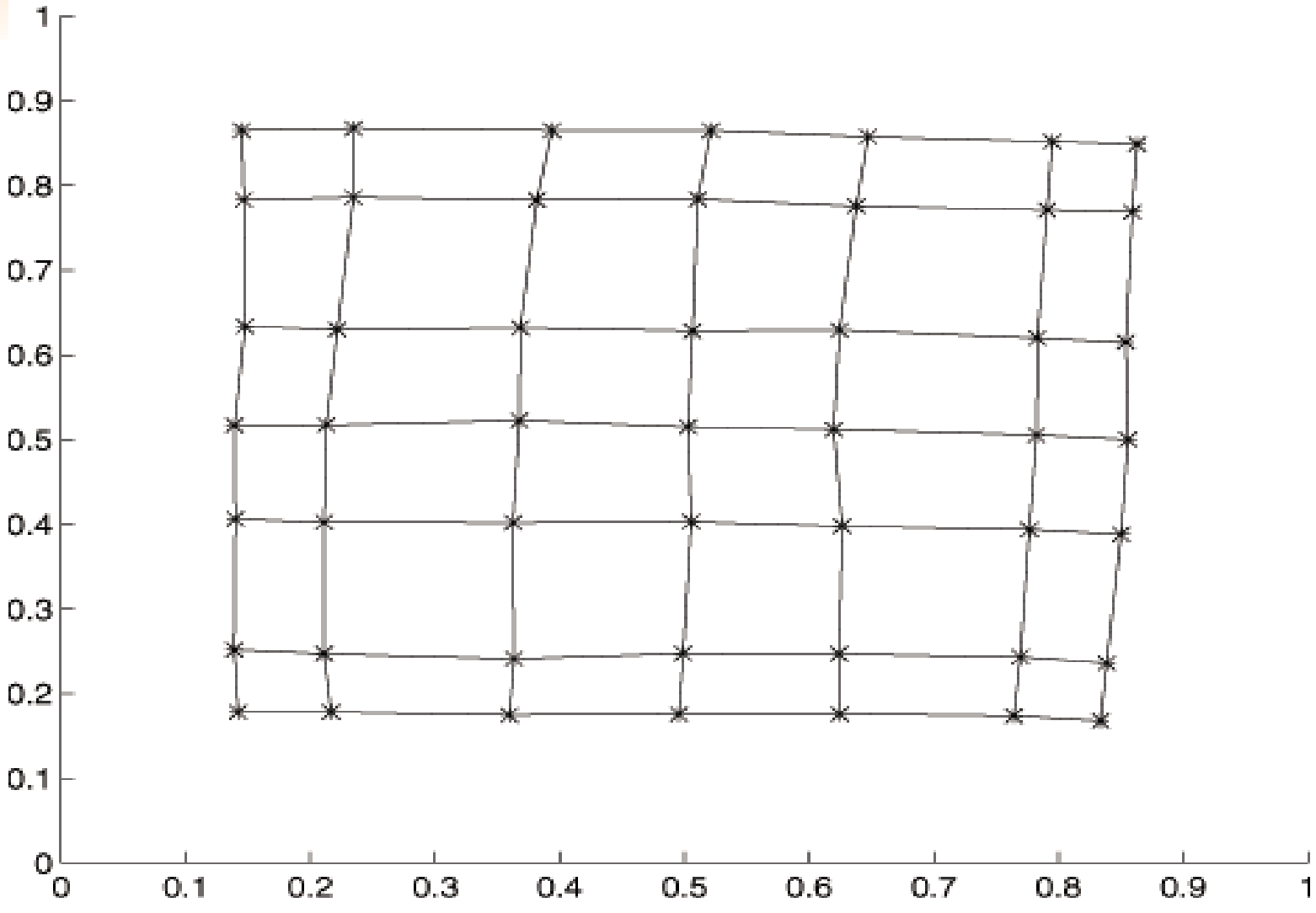
Comparaison sur données simulées

- On prend une grille 7 par 7, et une autre 10 par 10 (avec un système de voisinages fixe de 9 voisins) pour étudier
  - l'algorithme Kohonen batch, avec 100 itérations
  - l'algorithme on-line SOM, avec 50 000 itérations (i.e. équivalent)
- Les données sont uniformément distribuées dans un carré
- On choisit les mêmes valeurs initiales pour les deux algorithmes
- On observe que l'algorithme SOM trouve de meilleures solutions

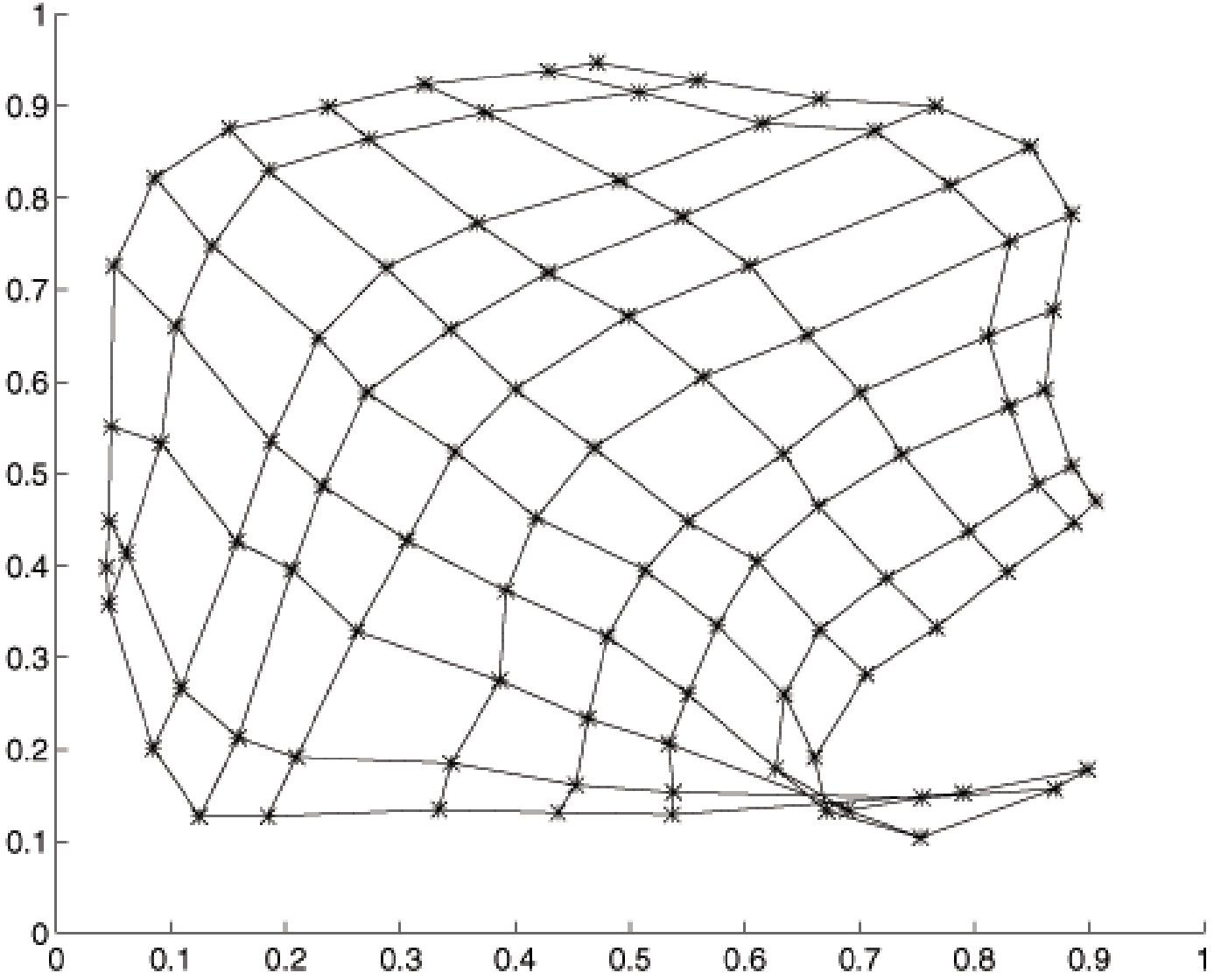
Algorithme batch pour des données uniformes sur une grille 7×7



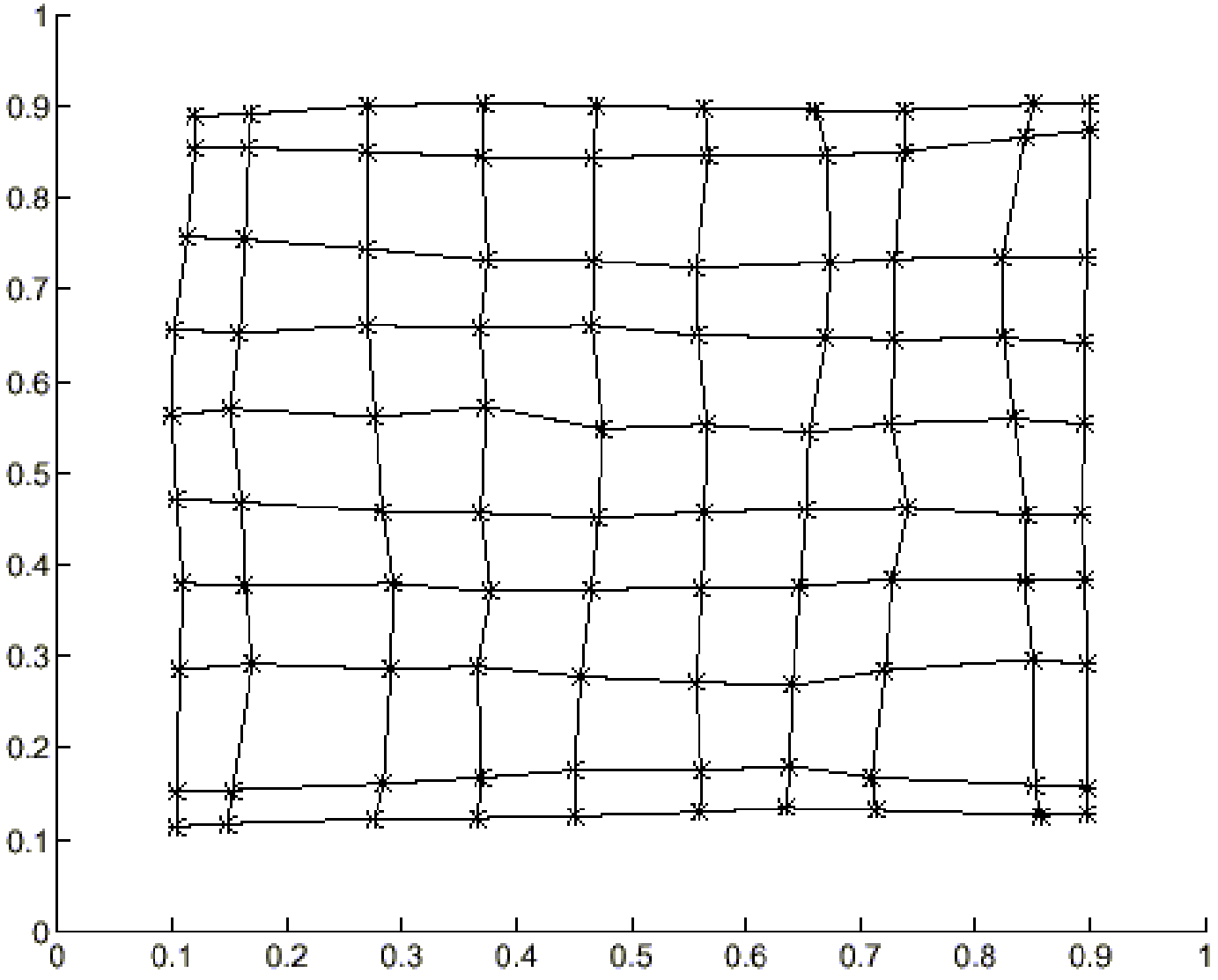
Algorithme on-line SOM pour des données uniformes sur une grille 7×7



Algorithme batch pour des données uniformes sur une grille 10×10



Algorithme on-line SOM pour des données uniformes sur une grille 10×10



Analyse de données : introduction

Algorithme de Kohonen

Kohonen et classification : KACP

Traitements des variables qualitatives

Conclusion

Cartes de Kohonen : Classification

- Pour représenter des données au moyen de l'algorithme de Kohonen, on prend comme entrées les lignes de la matrice des données
- Après apprentissage, chaque individu (ligne) correspond à une unité du réseau (celle qui gagne quand on présente cet individu)
- ***On classe une observation dans la classe A_i définie par l'unité gagnante qui lui correspond ($i=i_0(x)$)***
- On obtient donc une classification des individus, avec respect des voisinages
- La carte ainsi obtenue fournit une représentation plane
- Ici l'existence de proximités entre classes qui se ressemblent est essentielle

Représentation (KACP)

- Dans chaque classe on peut représenter le vecteur code
  - en donnant ses *P* composantes
  - en dessinant une courbe à *P* points

- Dans chaque classe, on peut
  - faire la liste des observations de cette classe
  - représenter en superposition les observations de la classe

- Ceci fournit une **représentation plane**, analogue à l'analyse en composantes principales (mais une seule carte et pas de projection orthogonale)

Classes et distances

- Comme le nombre de classes est fixé a priori assez grand, il est utile de procéder à une regroupement
- On fait une classification hiérarchique sur les vecteurs codes, ce qui définit des super-classes
- On ***colorie ces super-classes*** (cf. classification mixte)

- On peut visualiser les distances entre les classes de Kohonen, car la disposition sur la grille donne une impression fausse d'équidistance

- Plus il y a du blanc entre deux classes (dans les 8 directions), plus la distance est grande

Nombreuses applications

- Représentation des pays, (Blayo et Letremy)
- Communes d'Ile-de France, (Ibbou, Tutin)
- Courbes de consommation, prévision, (Rousset)
- Consommation au Canada, (Gaubert, Gardes, Rousset)
- Segmentation du marché du travail (Gaubert)
- Démographie et composition sociale dans la vallée du Rhône, (Letremy, P.A.R.I.S)
- Etude sur le du leasing en Belgique, (de Bodt, Ibbou)
- Profils de chocs de taux d'intérêts, (de Bodt)
- Chômeurs récurrents, (Gaubert)
- Niveau de vie des ménages (Ponthieux)
- Dépenses de formations en entreprise (Perraudin, Petit, Lémière), ...

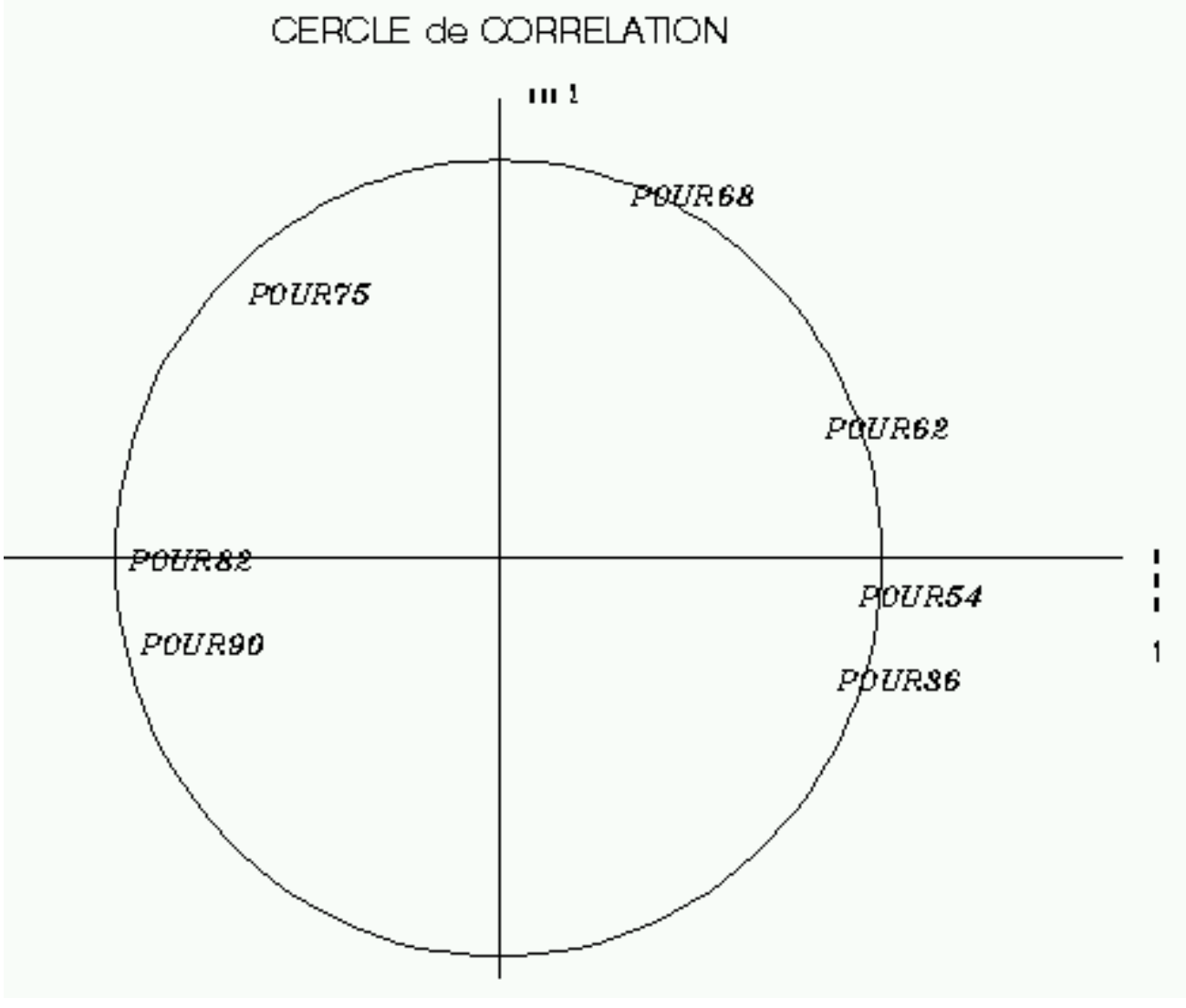
Un exemple : recensements de 1783 communes de la vallée du Rhône

- Chiffres de population aux recensements de 1936, 1954, 1962, 1968, 1975, 1982, 1990.
- Communes de la vallée du Rhône (dans le cadre d'une étude du rapport entre l'évolution démographique et la répartition de la population entre différentes catégories socio-professionnelles).
- Ardèche, Bouches-du-Rhône, Drôme, Gard, Hérault, Isère, Haute-Loire, Vaucluse
- Les chiffres sont transformés en pourcentages en divisant par la somme de tous les recensements (pour supprimer l'effet taille)
- La distance utilisée est la distance du chi-deux
- 10 000 itérations, une grille 8 par 8, 5 super-classes (82% d 'inertie)

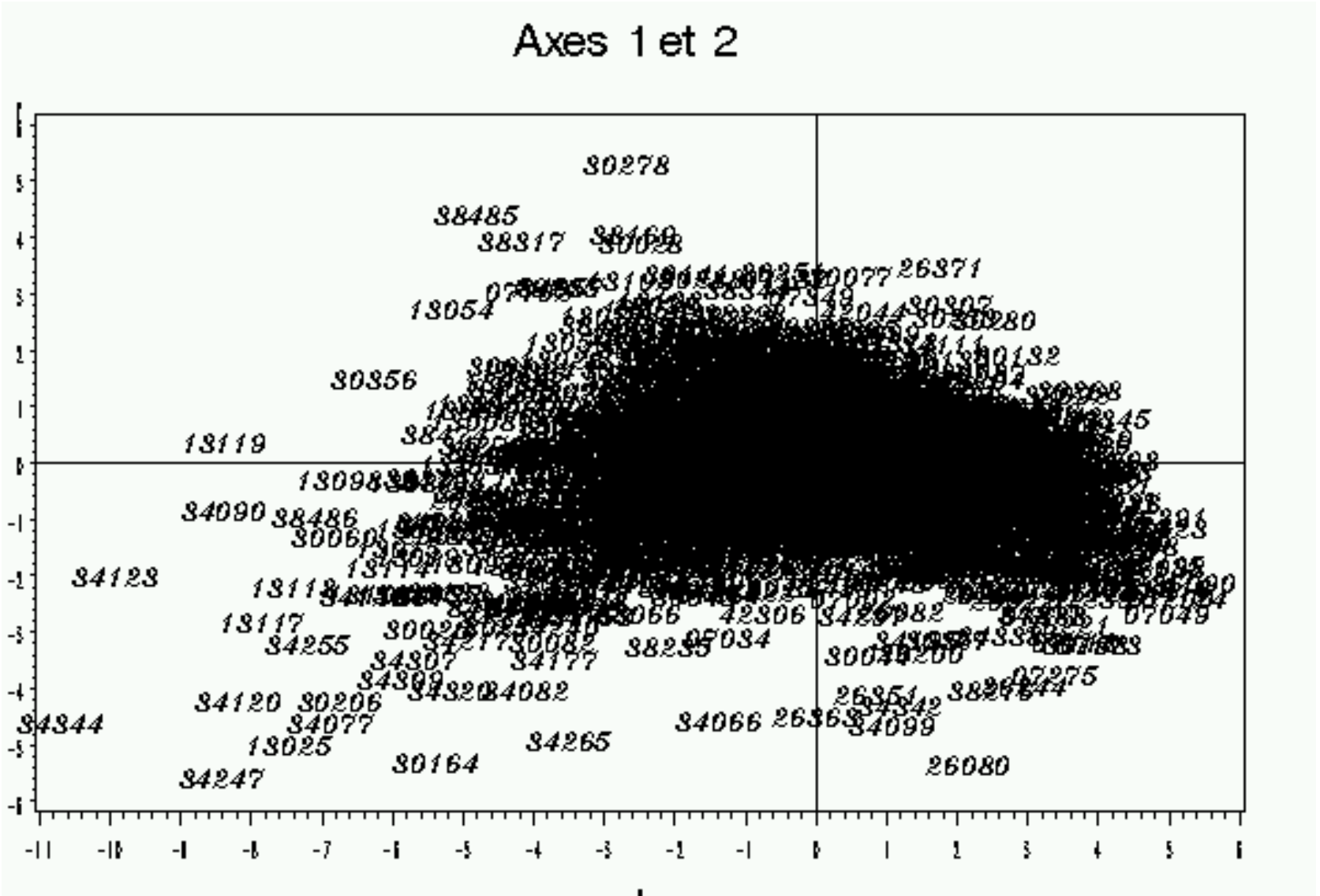
Les données (extrait)

CODE	NOM	POUR36	POUR54	POUR62	POUR68	POUR75	POUR82	POUR90
07001	ACCONS	0.178	0.147	0.137	0.113	0.118	0.154	0.153
07002	AILHON	0.196	0.146	0.117	0.103	0.098	0.127	0.213
07003	AIZAC	0.210	0.170	0.150	0.135	0.105	0.110	0.120
07004	AJOUX	0.304	0.173	0.143	0.123	0.097	0.085	0.076
07005	ALBA-LA-ROMAINE	0.158	0.122	0.131	0.145	0.144	0.137	0.164
07006	ALBON	0.257	0.190	0.157	0.127	0.107	0.083	0.079
07007	ALBOUSSIERE	0.164	0.150	0.144	0.139	0.138	0.120	0.144
07008	ALISSAS	0.125	0.121	0.147	0.133	0.125	0.161	0.188
07009	ANDANCE	0.137	0.145	0.135	0.139	0.155	0.138	0.151
07010	ANNONAY	0.121	0.125	0.142	0.160	0.160	0.150	0.143
07011	ANTRAIGUES	0.202	0.149	0.134	0.133	0.125	0.131	0.127
07012	ARCENS	0.187	0.146	0.152	0.145	0.128	0.122	0.120
07013	ARDOIX	0.151	0.139	0.122	0.130	0.141	0.152	0.166
07014	ARLEBOSC	0.199	0.159	0.149	0.143	0.129	0.113	0.108
07015	ARRAS-SUR-RHONE	0.147	0.158	0.155	0.155	0.128	0.122	0.135
07016	ASPERJOC	0.197	0.162	0.145	0.130	0.130	0.117	0.119
07017	ASSIONS	0.175	0.136	0.152	0.140	0.123	0.135	0.138
07018	ASTET	0.277	0.215	0.152	0.127	0.100	0.067	0.062
07019	AUBENAS	0.112	0.121	0.129	0.151	0.169	0.162	0.156
07020	AUBIGNAS	0.209	0.114	0.140	0.113	0.125	0.133	0.167
07022	BAIX	0.135	0.128	0.135	0.128	0.115	0.202	0.157
07023	BALAZUC	0.215	0.154	0.127	0.112	0.109	0.141	0.142
07024	BANNE	0.194	0.157	0.134	0.118	0.122	0.134	0.140
07025	BARNAS	0.262	0.175	0.153	0.133	0.095	0.089	0.093
07026	BEAGE	0.234	0.183	0.149	0.142	0.118	0.097	0.078
07027	BEAUCHASTEL	0.100	0.103	0.120	0.127	0.184	0.192	0.174
07028	BEAULIEU	0.175	0.153	0.147	0.136	0.136	0.132	0.122
07029	BEAUMONT	0.308	0.166	0.127	0.112	0.094	0.100	0.093
07030	BEAUVENE	0.227	0.167	0.166	0.142	0.105	0.099	0.094
07031	BERRIAS-ET-CASTELJAU	0.183	0.148	0.147	0.140	0.139	0.121	0.123
07032	BERZEME	0.221	0.184	0.141	0.147	0.105	0.111	0.091
07033	BESSAS	0.180	0.134	0.133	0.133	0.134	0.154	0.134
07034	BIDON	0.203	0.117	0.079	0.092	0.102	0.187	0.219
07035	BOFFRES	0.224	0.175	0.146	0.135	0.105	0.110	0.105
07036	BOGY	0.164	0.137	0.133	0.119	0.128	0.144	0.175
07037	BOREE	0.269	0.200	0.157	0.135	0.100	0.074	0.064
07038	BORNE	0.311	0.188	0.167	0.124	0.073	0.058	0.079
07039	BOZAS	0.208	0.177	0.164	0.144	0.114	0.106	0.088
07040	BOUCIEU-LE-ROI	0.212	0.163	0.145	0.147	0.116	0.104	0.113
07041	BOULIEU-LES-ANNONAY	0.115	0.116	0.126	0.137	0.157	0.166	0.183
07042	BOURG-SAINT-ANDEOL	0.091	0.090	0.107	0.173	0.168	0.181	0.190
07044	BROSSAINC	0.173	0.160	0.157	0.144	0.124	0.117	0.124
07045	BURZET	0.243	0.202	0.154	0.129	0.095	0.095	0.082
07047	CELLIER-DU-LUC	0.234	0.166	0.149	0.128	0.113	0.098	0.111
07048	CHALENCON	0.223	0.174	0.164	0.147	0.097	0.099	0.095
07049	CHAMBON	0.331	0.205	0.154	0.110	0.071	0.069	0.061
07050	CHAMBONAS	0.159	0.145	0.148	0.148	0.138	0.127	0.135
07051	CHAMPAGNE	0.131	0.129	0.129	0.132	0.143	0.155	0.182
07052	CHAMPIS	0.197	0.169	0.159	0.141	0.121	0.096	0.118
07053	CHANDOLAS	0.181	0.141	0.141	0.142	0.136	0.132	0.127

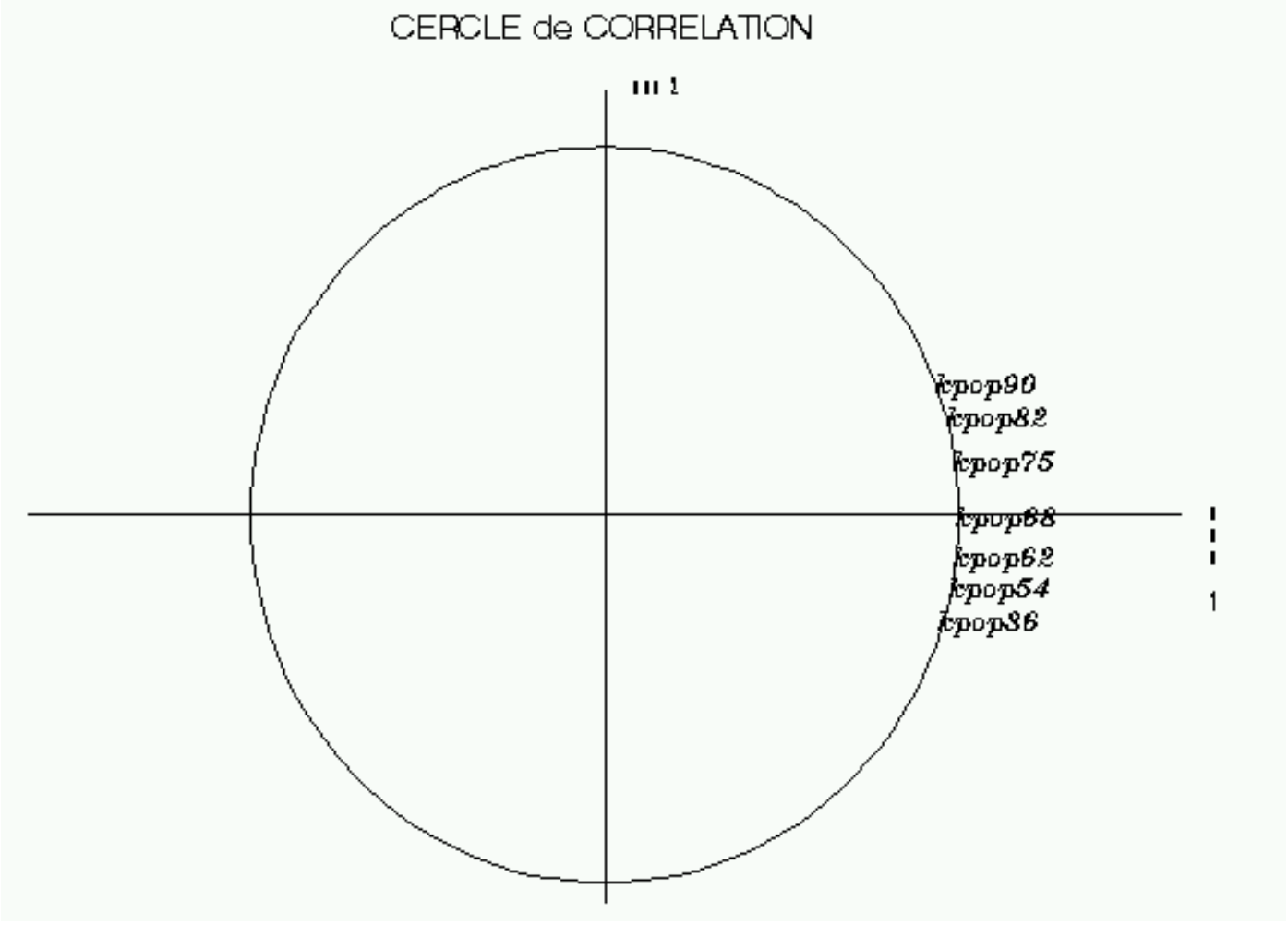
ACP sur les communes (88% sur les axes 1 et 2)



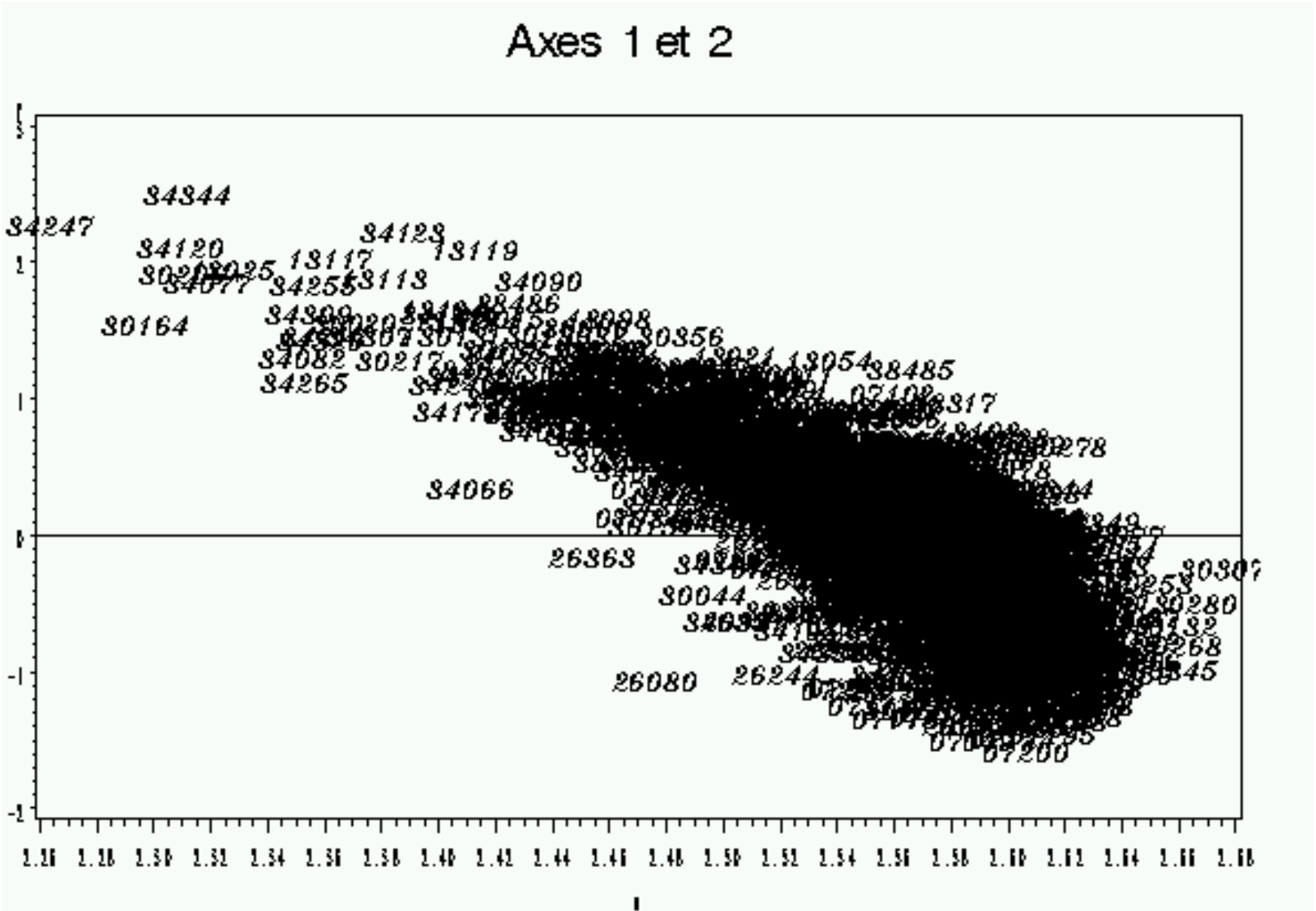
Les communes, ACP



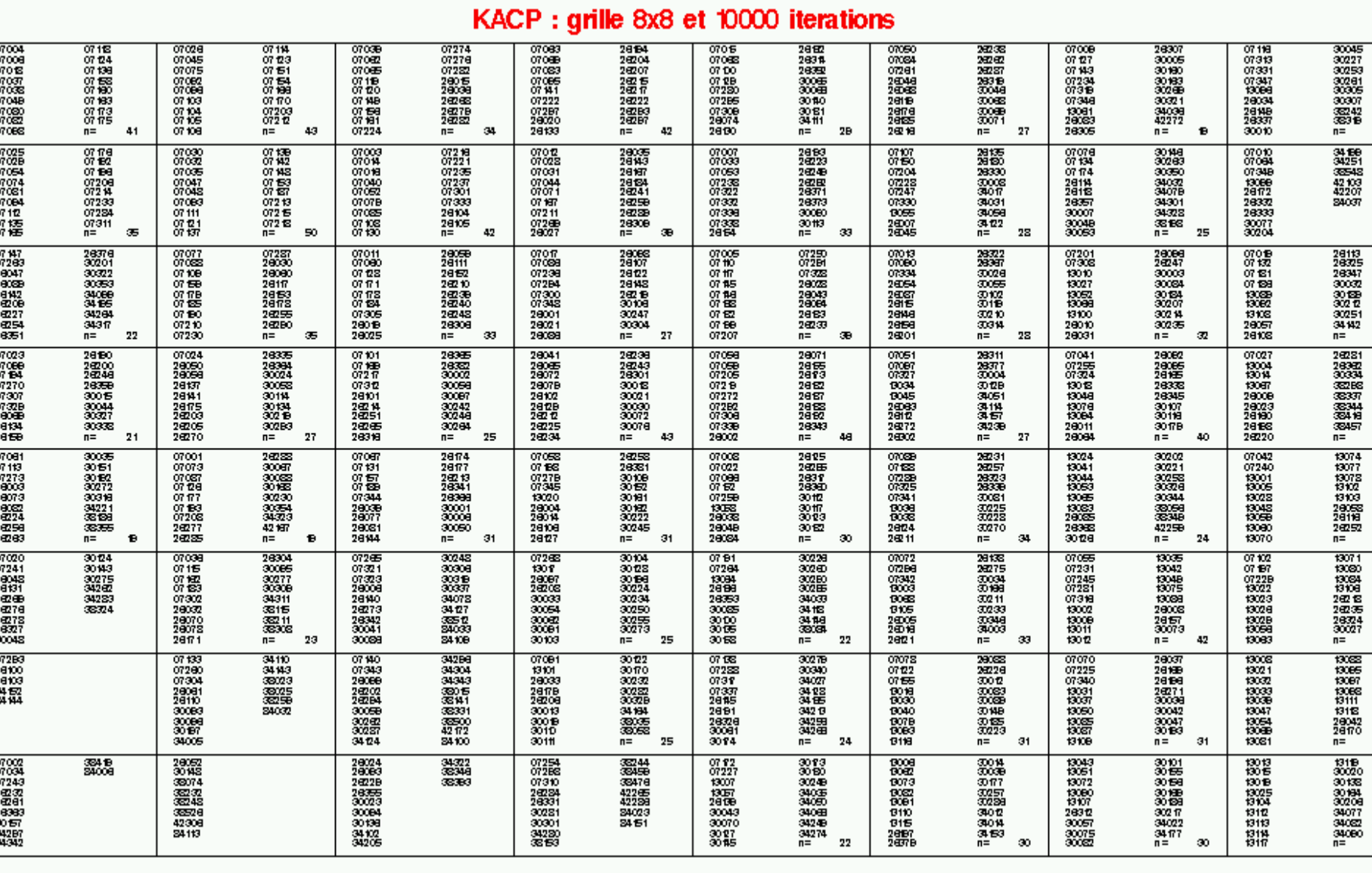
ACP
(Données corrigées par le Chi-2, 98 %)



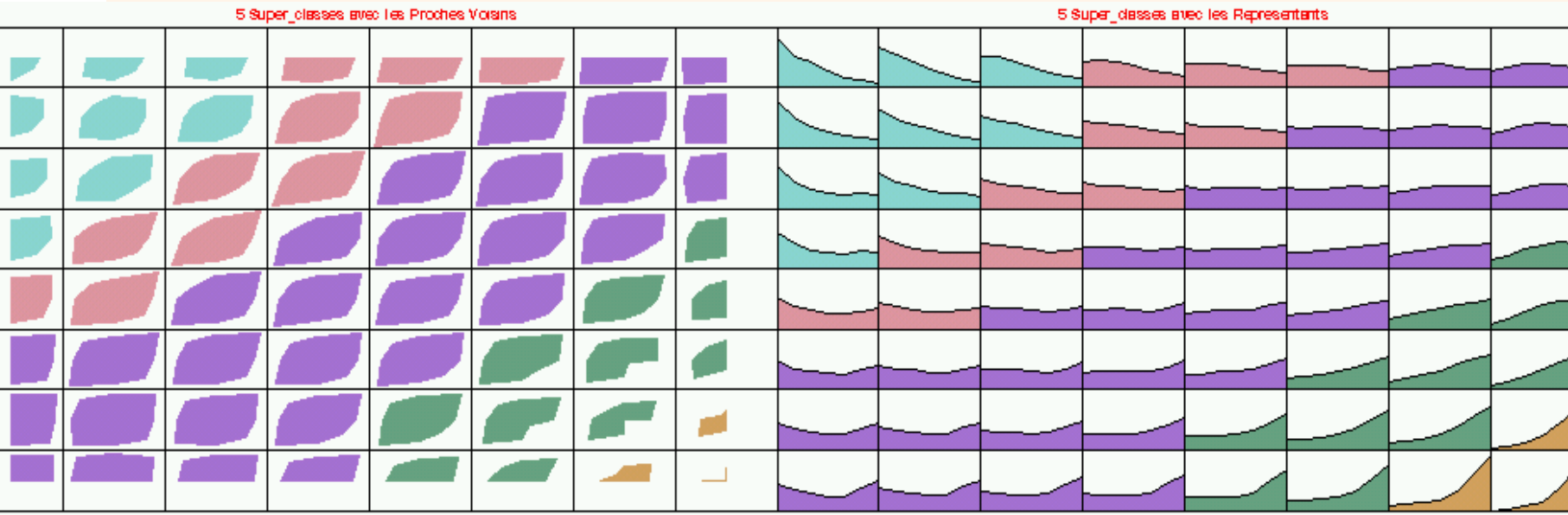
ACP (Données corrigées pour le Chi-2)

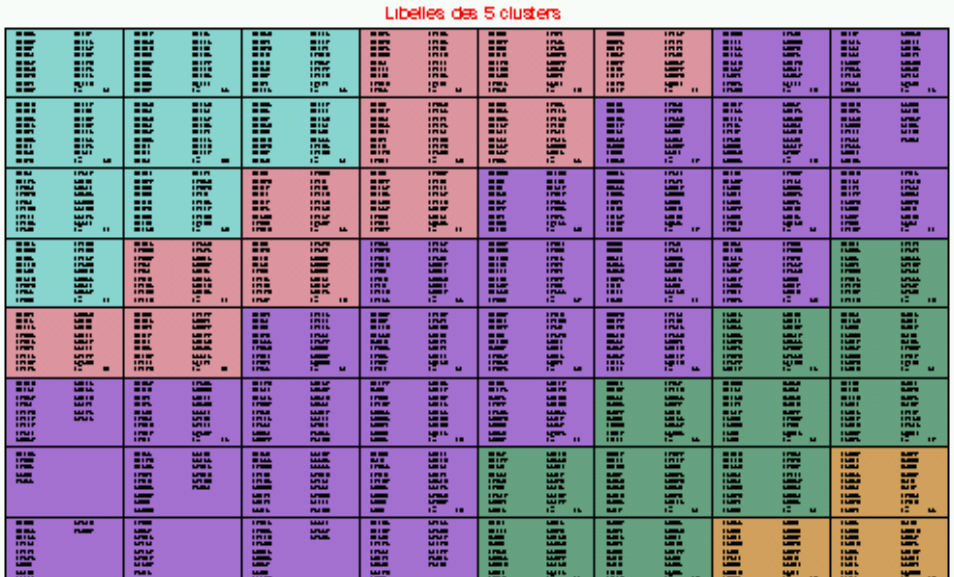


1783 communes classées en 64 classes



Classes, super-classes et distances

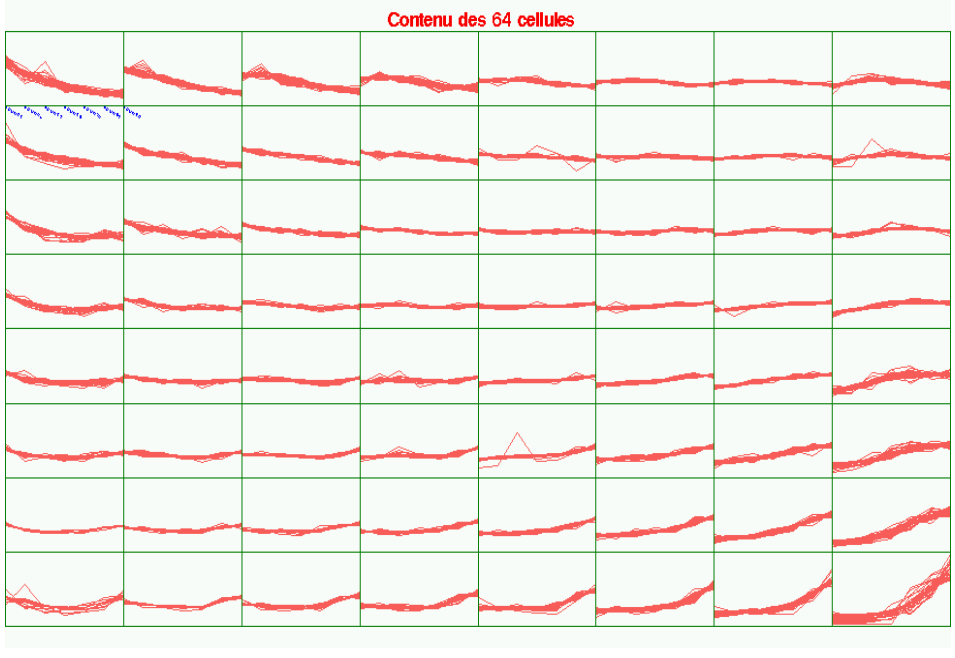


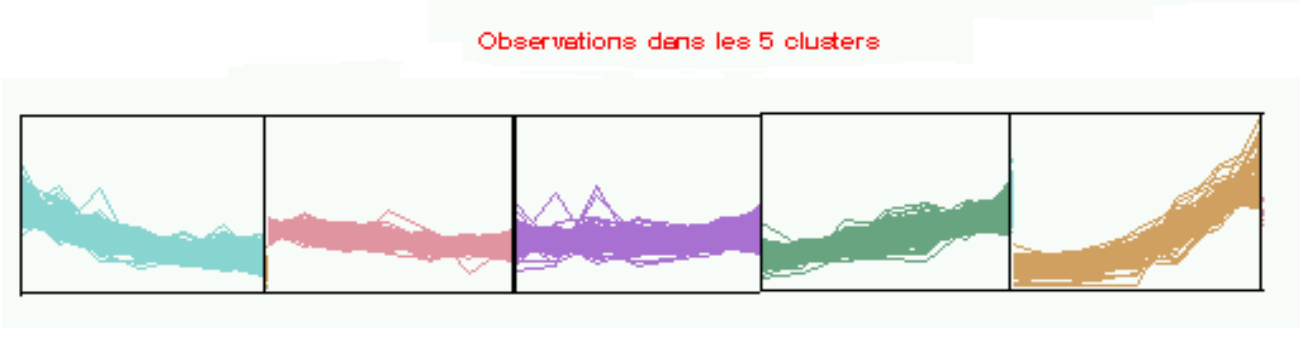


Contenus des super-classes

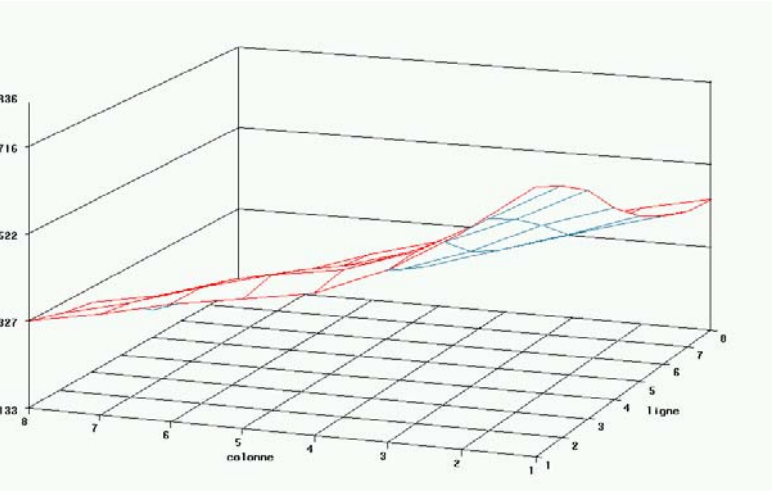
- Classe 1 : forte décroissance, 323 communes, aucune des Bouches du Rhône, par exemple Issarlès (07), St Maurice Navacelles (34), St Léger-du-Ventoux (84)
- Classe 2 : décroissance moyenne, 320 communes, aucune des Bouches-du-Rhône, par exemple Lamastre (07), Antraigues (07), Génolhac (30), Fontaine-de-Vaucluse (84)
- Classe 3 : stabilité, 726 communes, la classe la plus nombreuse, par exemple Aubenas (07), Marseille (13), Crozes-Hermitage (26), Nîmes (30), Clermont-l'Hérault (34)
- Classe 4 : croissance modérée, 322 communes, par exemple Bourg-St Andéol (07), Aix-en-Provence (13), Baux-de-Provence (13), Montélimar (26), Pont-St Esprit (30)
- Classe 5 : croissance forte, 92 communes, aucune d'Ardèche, beaucoup des Bouches-du-Rhône, par exemple Fos-sur-mer (13), Vitrolles (13), La-Grande-Motte (34), Seyssins (38), Puget (84)

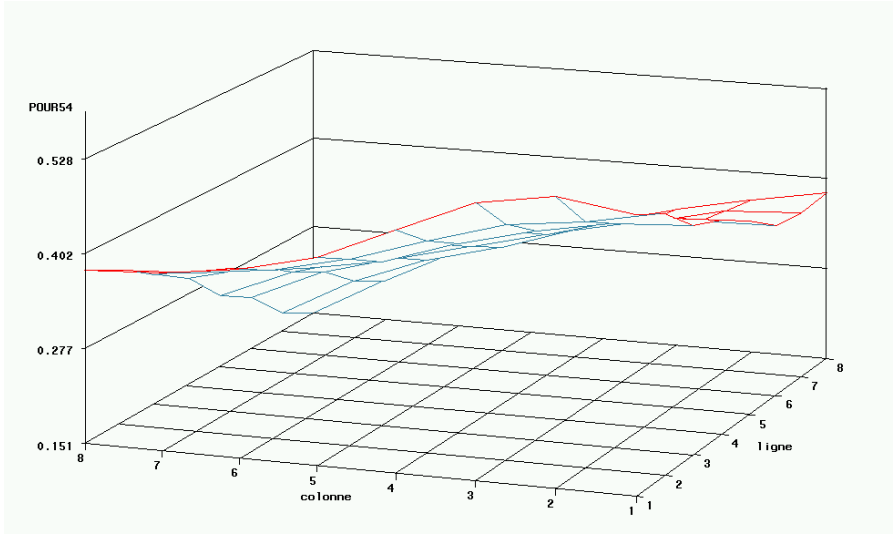
Contenu des classes et des super-classes

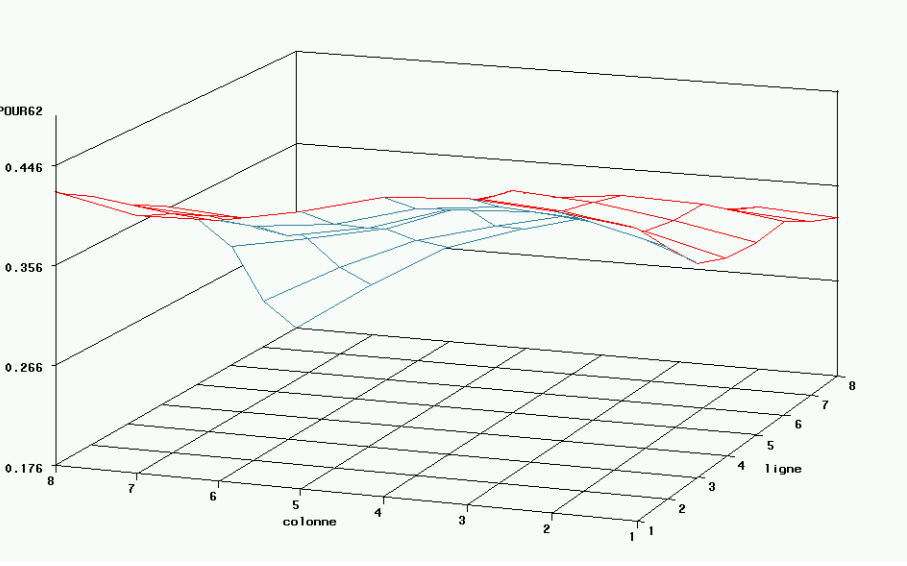


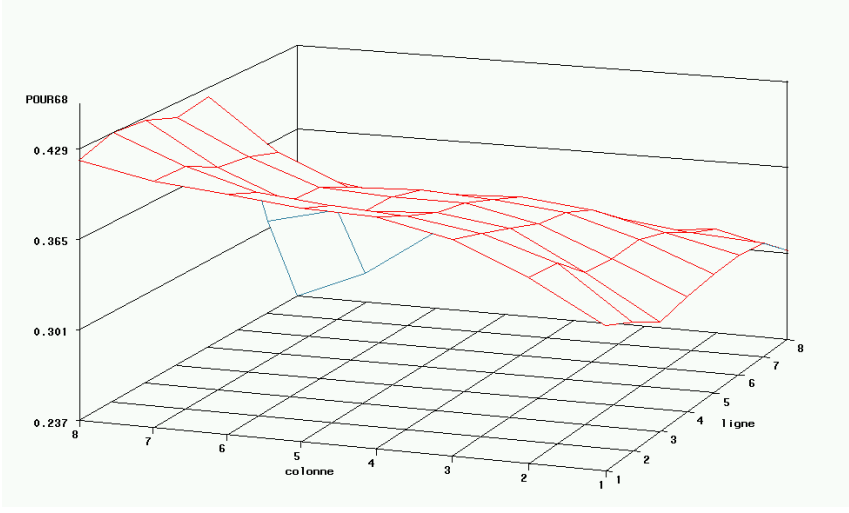


Les quatre premiers recensements

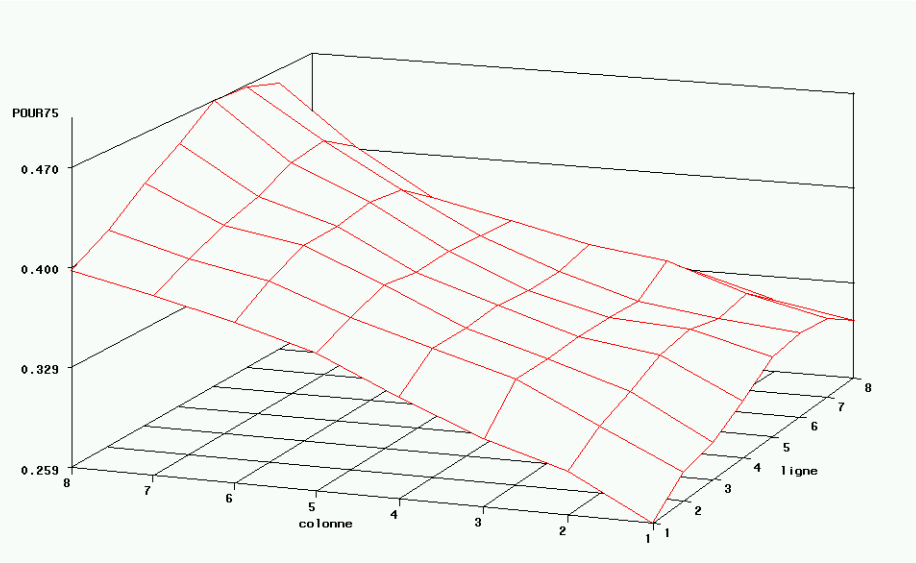


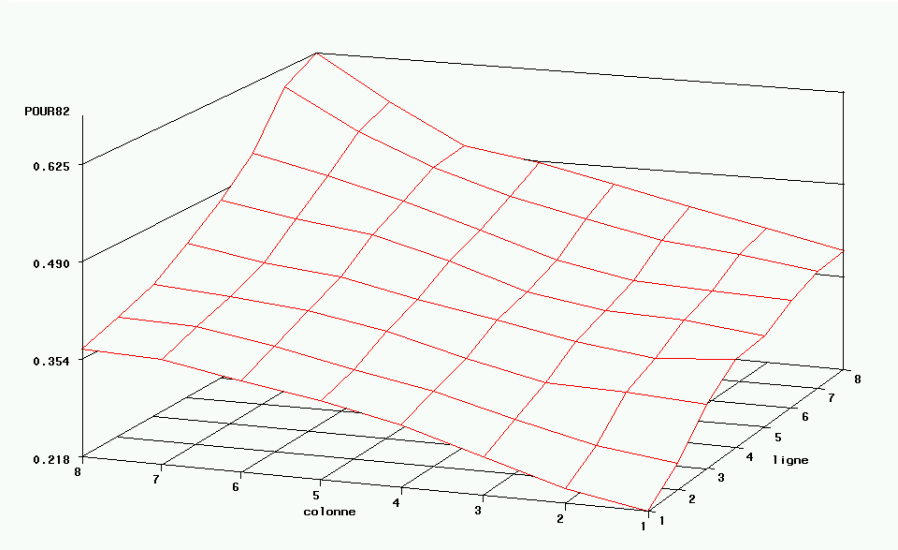


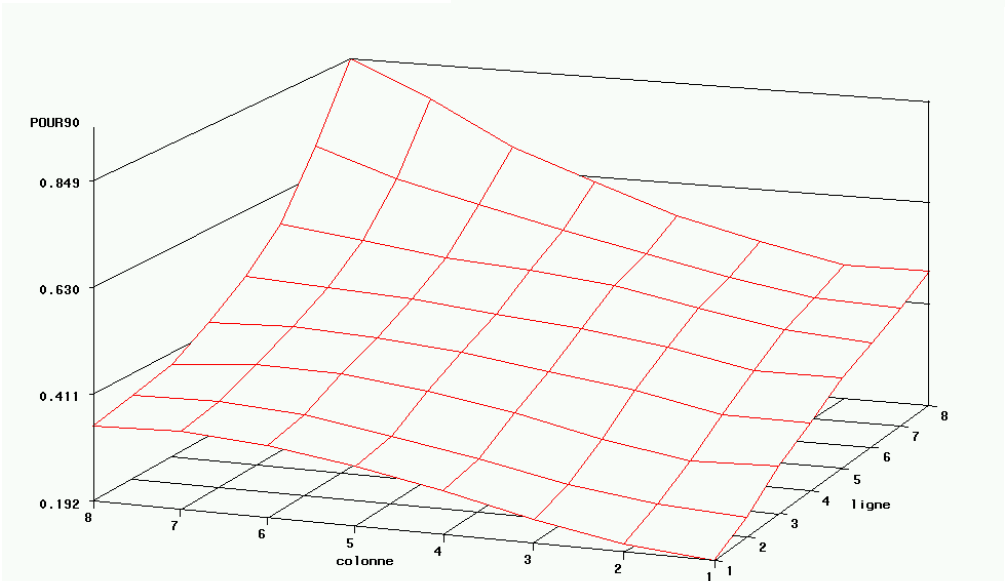




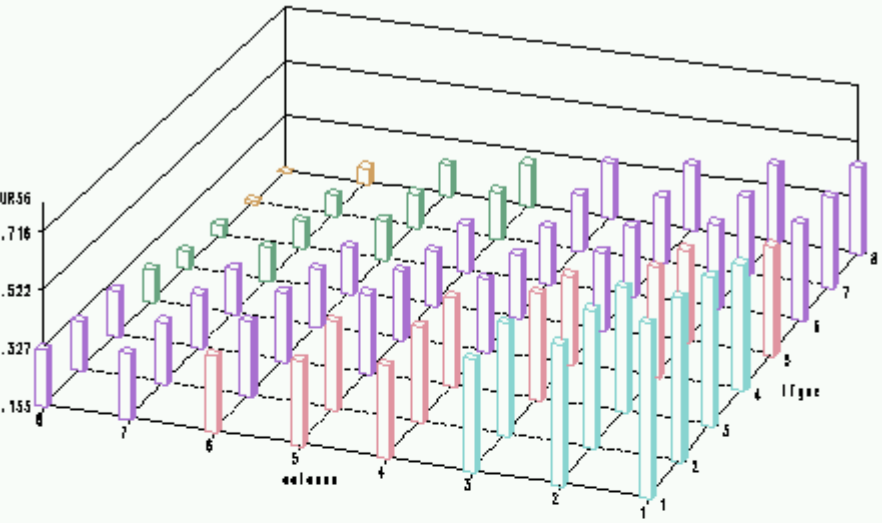
Les trois derniers

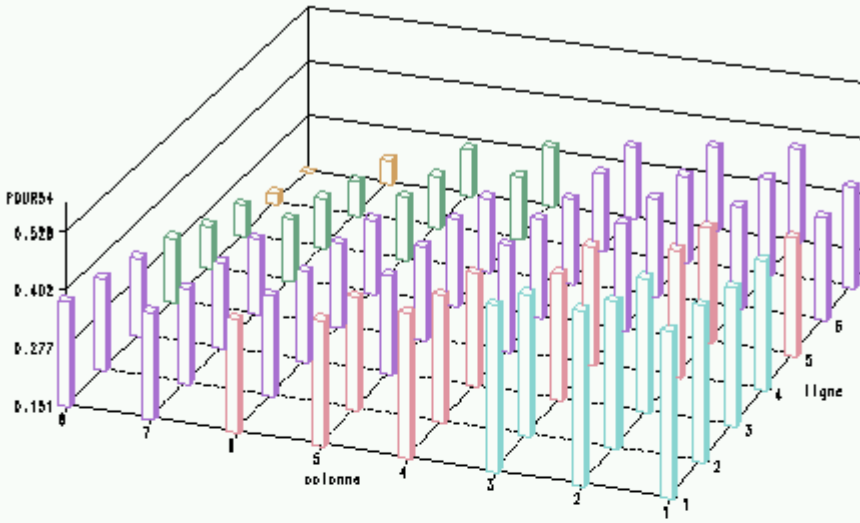


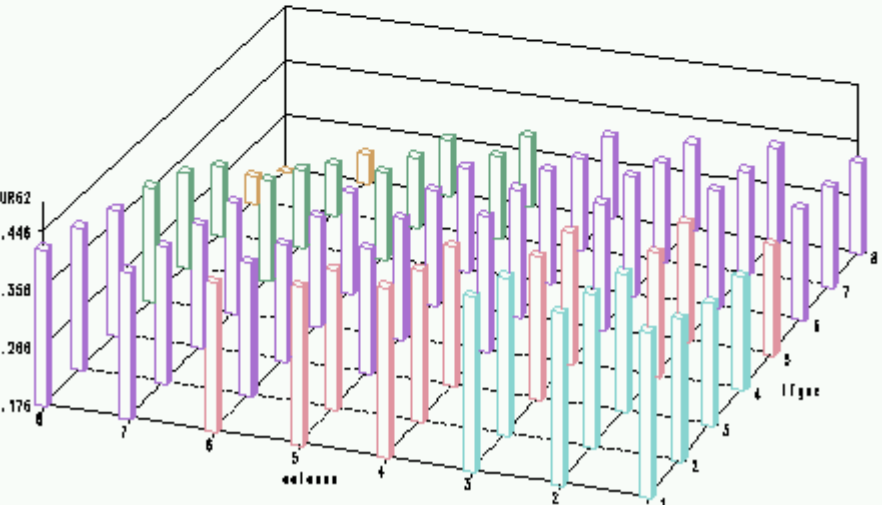


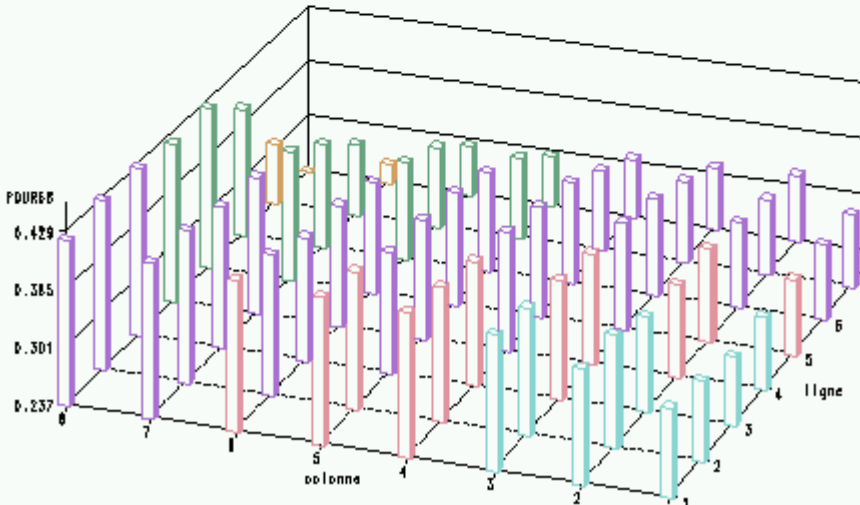


Autre représentation

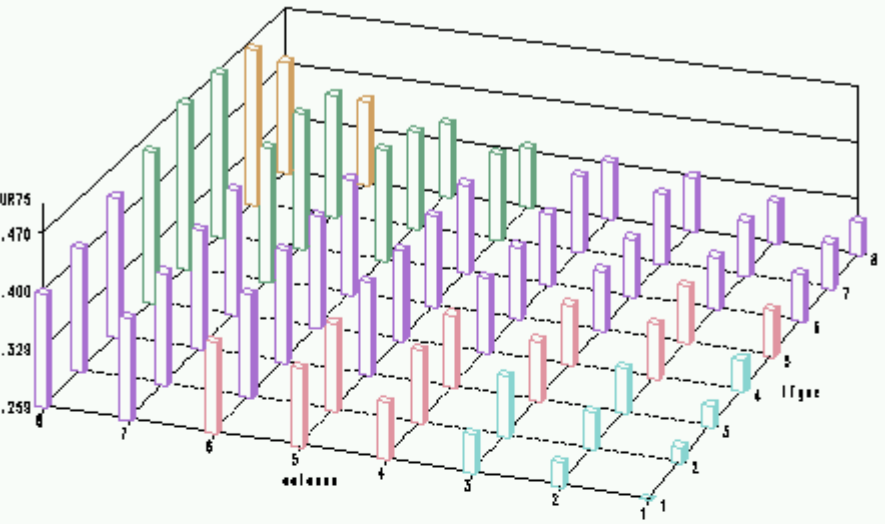


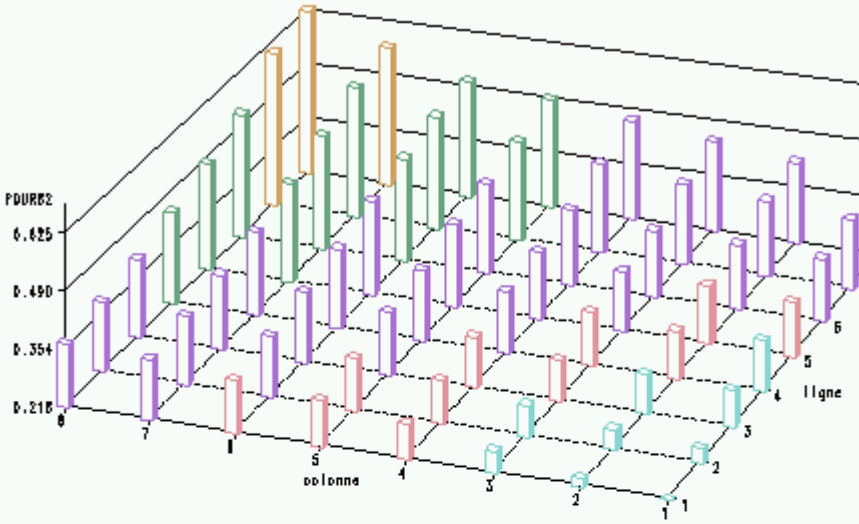


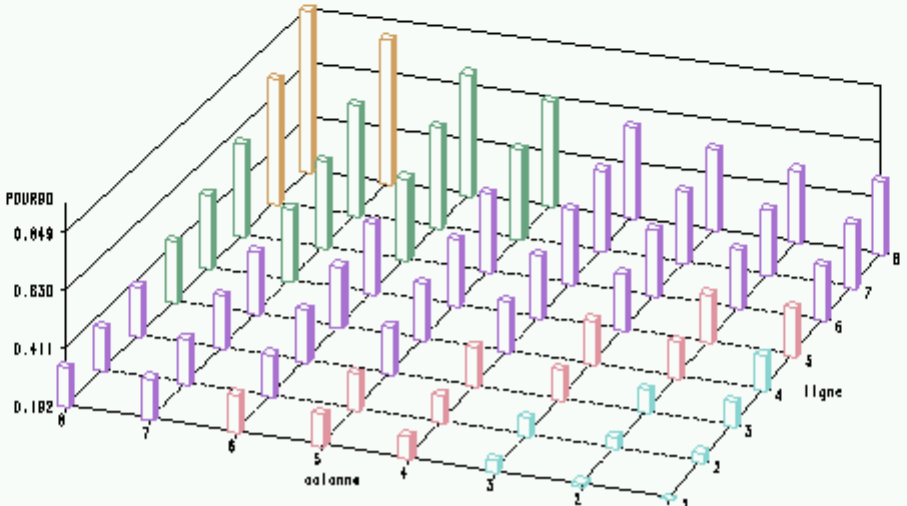




De même


colonne
ligne


POURB2
0.625
0.490
0.354
0.218
colonne
ligne


POURB0
0.849
0.630
0.411
0.192
colonne
ligne

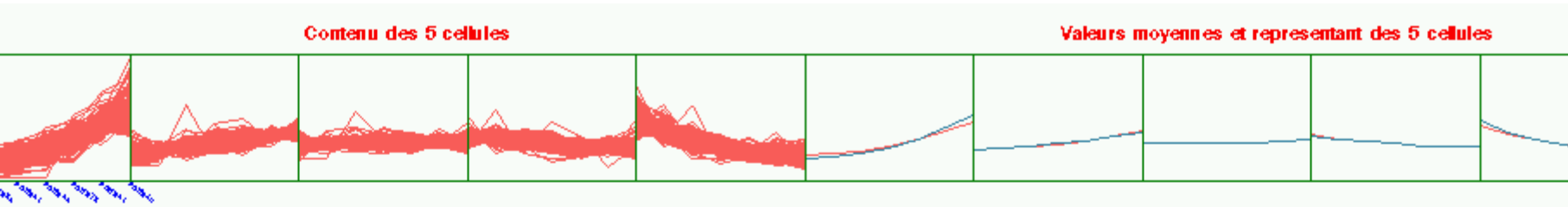
Qualités de la classification

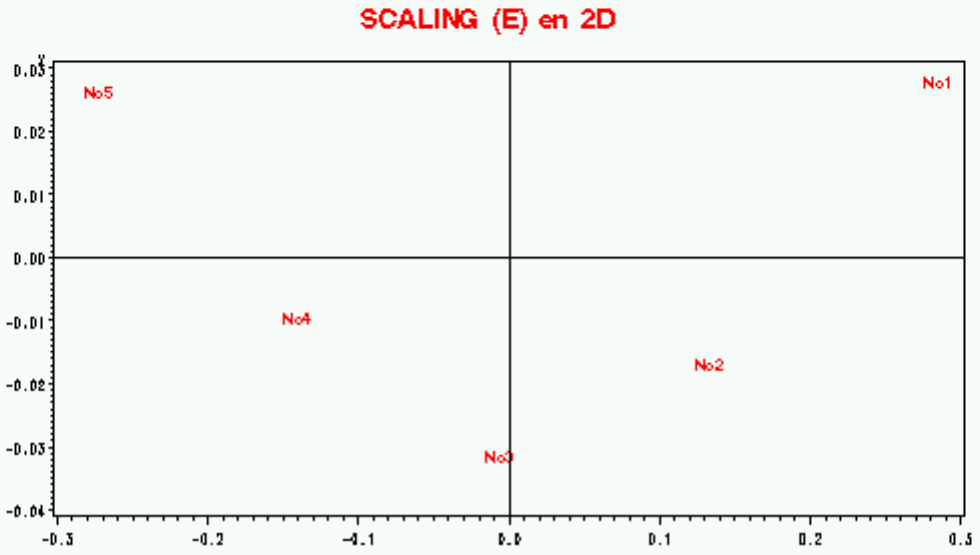
	DL	POUR36	POUR54	POUR62	POUR68	POUR75	POUR82	POUR90
SC_INTER	4	4.560	1.668	0.656	0.153	0.361	2.029	5.745
SC_INTRA	1778	0.980	0.425	0.443	0.484	0.515	0.522	1.365
SC_TOTAL	1782	5.540	2.093	1.100	0.637	0.876	2.551	7.110
FISHER		**2068.472**	**1744.982**	657.897	140.003	311.868	**1727.231**	**1870.749**
P_VALUE		0	0	0	0	0	0	0

- Les recensements des années 62, 68, 75, sont moins discriminants

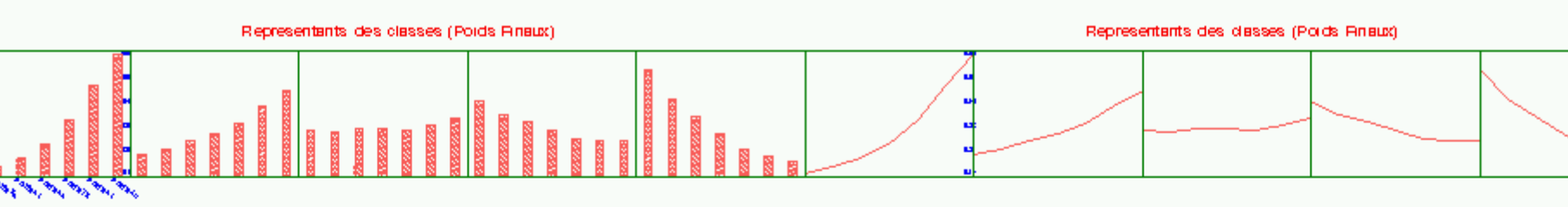
	VALEUR	KHI2	DL	P_VALUE
WILKS	0.074055	4641.05	28	0
HOTELLING	9.34215	16657.05	28	0

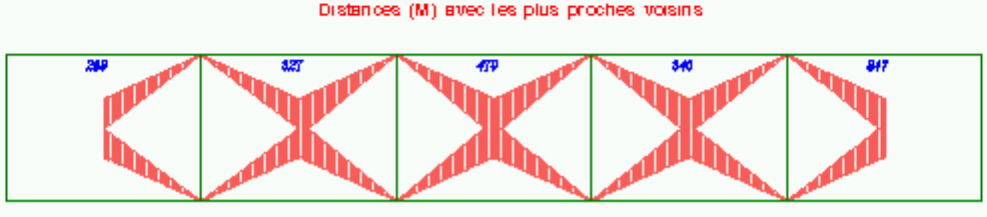
Ficelle de Kohonen de 5 classes





Ficelle à 5 classes





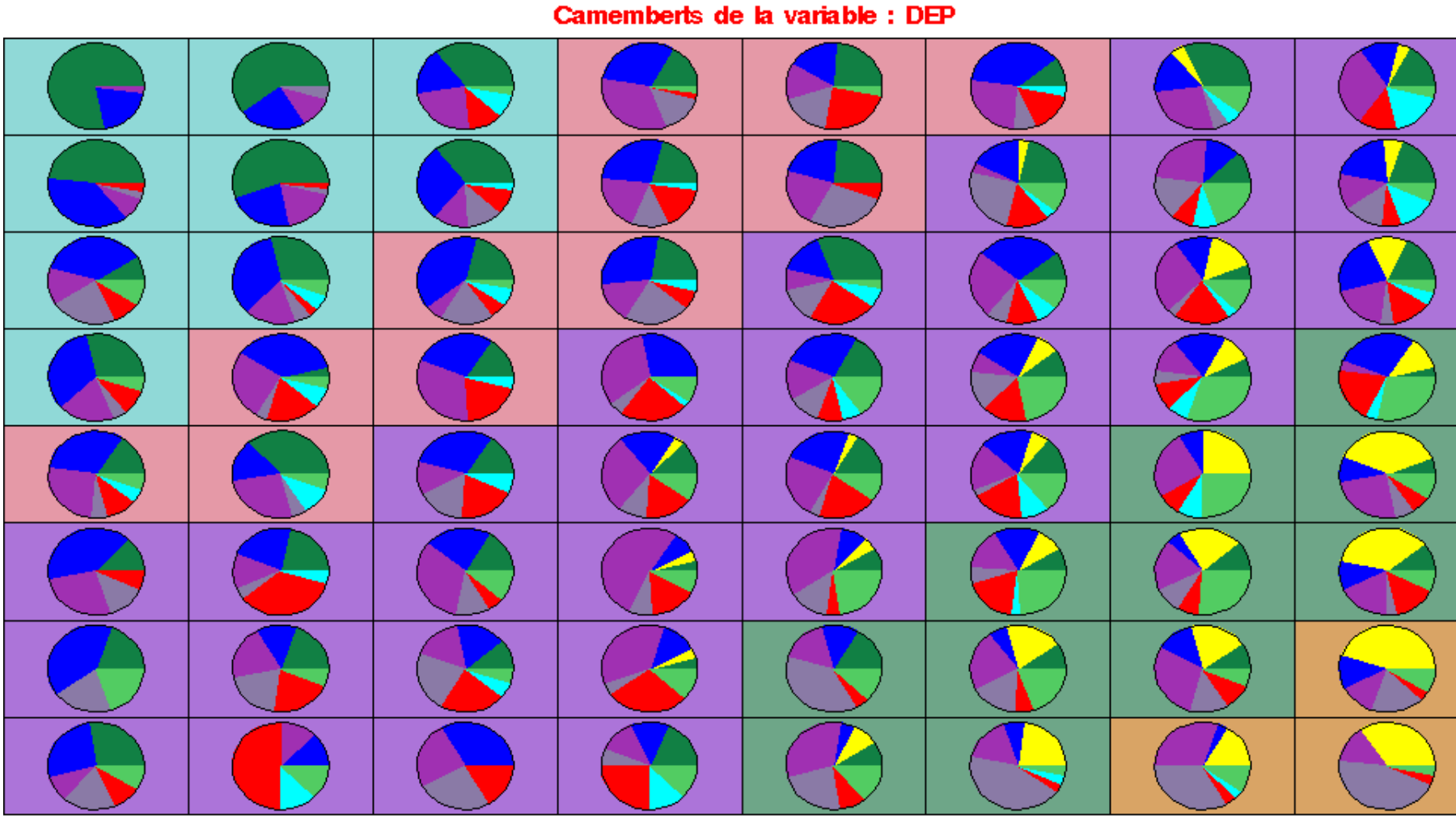
Qualités de cette classification

- Les recensements discriminants sont les recensements de début et fin de période

	DL	POUR36	POUR54	POUR62	POUR68	POUR75	POUR82	POUR90
SC_INTER	4	4.490783	1.630996	0.640934	0.12245	0.347266	2.020992	5.58925
SC_INTRA	1778	1.049024	0.461839	0.458884	0.514403	0.528425	0.530442	1.520978
SC_TOTAL	1782	5.539807	2.092835	1.099818	0.636853	0.87569	2.551433	7.110228
FISHER		**1902.866**	**1569.762**	620.8439	105.8101	292.113	**1693.552**	**1633.437**
P_VALUE		0	0	0	0	0	0	0

	VALEUR	KHI2	DL	P_VALUE
WILKS	0.090896	4275.7	28	0
HOTELLING	8.032973	14322.79	28	0

Par département



Départements par super-classes

- Ardèche (07)
- Bouches-du-Rhône (13)
- Drôme (26)
- Gard (30)
- Hérault (34)
- Isère (38)
- Haute-Loire (42)
- Vaucluse (84)

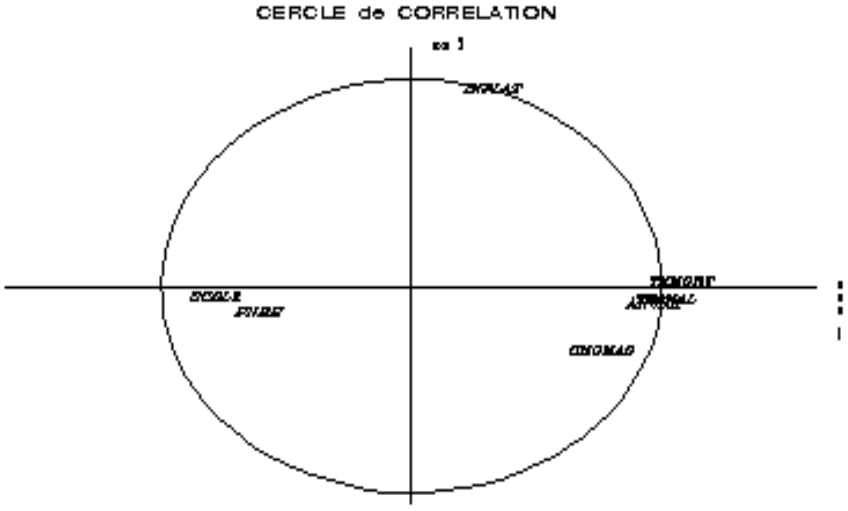
Un exemple : 96 pays en 1996

ANCRX	Croissance annuelle de la population en %
TXMORT	Taux de mortalité infantile (en pour mille)
TXANAL	Taux d'illetrisme en %
SCOL2	Indice de fréquentation scolaire au second degré
PNBH	PNB par habitant exprimé en dollars
CHOMAG	Taux de chomage en %
INFLAT	Taux d'inflation en %
NIVIDH	Niveau de l' Indice de Développement Humain (6 niveaux codés par libellés) (faible1, faible2, moyen1, moyen2, fort1, fort2)
CATIDH	Niveau d' Indice de Développement Humain (6 niveaux codés de 1 à 6)

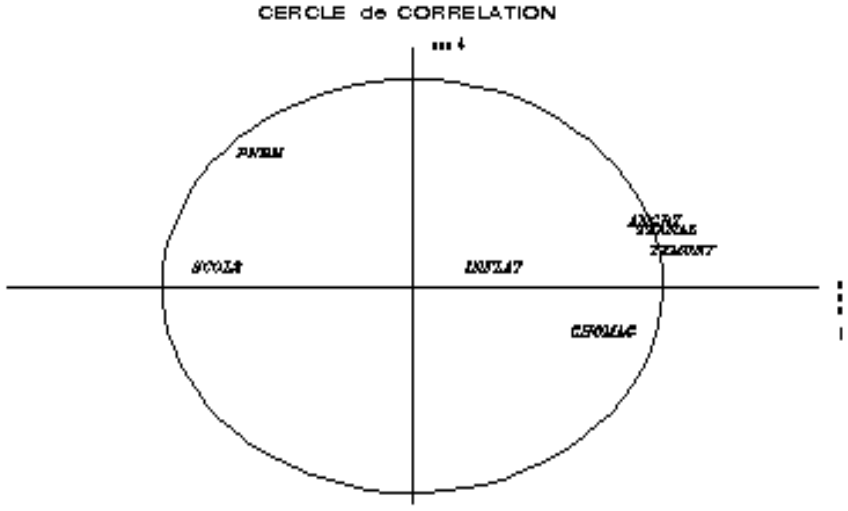
Les données

PAYS	ANCRX	TXMORT	TXANAL	SCOL2	PNBH	CHOMAG	INFLAT	NIVIDH	CATIDH
Afghanistan	6	159	70,9	15	276	19	17	faible1	1
Afrique du sud	2,6	46,9	23,5	77	2873	33	10	moyen2	4
Albanie	1,1	33,1	8	29,2	828	17,9	16,1	moyen2	4
Algerie	2,2	42,1	42	61	1570	27	31	moyen2	4
Allemagne	0,2	5,8	1	101,2	24993	9,4	3,1	fort2	6
Angola	3,6	126,8	58	14	575	25	951	faible1	1
Arabie Saoudite	3	68,8	39,5	49	7081	6,6	0,7	moyen2	4
Argentine	1,1	33,8	4,4	72,3	7827	11,3	4	fort1	5
Australie	1,3	5,9	0,1	84	17688	9,7	2,5	fort2	6
Bahrein	2,5	24,2	17	99	7500	15	2	fort1	5
Belgique	0,1	7,8	0,3	103,2	22225	12,6	2,6	fort2	6
Bolivie	2,2	74,9	20	37	733	6,2	8,5	moyen1	3
Bresil	1,6	59,8	18	43	3073	5,5	1094	fort1	5
Bulgarie	-0,2	15,3	2,1	68,2	1058	17	33	moyen2	4
Cameroun	2,9	85,8	36,5	32	733	25,1	12,8	moyen1	3
Canada	1	6,7	3,1	104,2	18286	10,4	0,3	fort2	6
Chili	1,4	14,4	5,7	67	3643	6,1	11,2	fort1	5
Chine	1	25,8	22,4	55	418	2,5	22	moyen1	3
Chypre	1	9,9	4,5	95	9459	2	4,8	fort2	6
Colombie	1,7	36,8	8,5	62	1379	8	22,9	fort1	5
Comores	3,5	81,7	42,5	19	317	16	24,8	faible2	2
Coree du Sud	1	14,9	3,7	96	7572	2,3	6	fort1	5
Costa Rica	2,2	13,5	5,2	47	1896	5	15	fort1	5
Cote d' Ivoire	3,3	90,9	46,8	25	587	17	25,6	faible1	1
Croatie	0,1	11,5	3,2	83,2	2755	13,1	97,6	moyen2	4
Danemark	0,2	5,6	1	114,2	28346	12,1	2,1	fort2	6
Egypte	1,9	58,8	50,5	76	632	20,2	8,3	moyen1	3
Emirats arabes uni	2,2	23,2	20,9	89	23809	0,2	5	fort1	5
Equateur	2,1	36,8	12,8	55	1205	7,2	26	moyen2	4
Espagne	0,2	7,3	7,1	110,2	12283	24,4	4,8	fort2	6
Etats Unis	1	8,2	3	97,2	25219	5,6	2,8	fort2	6

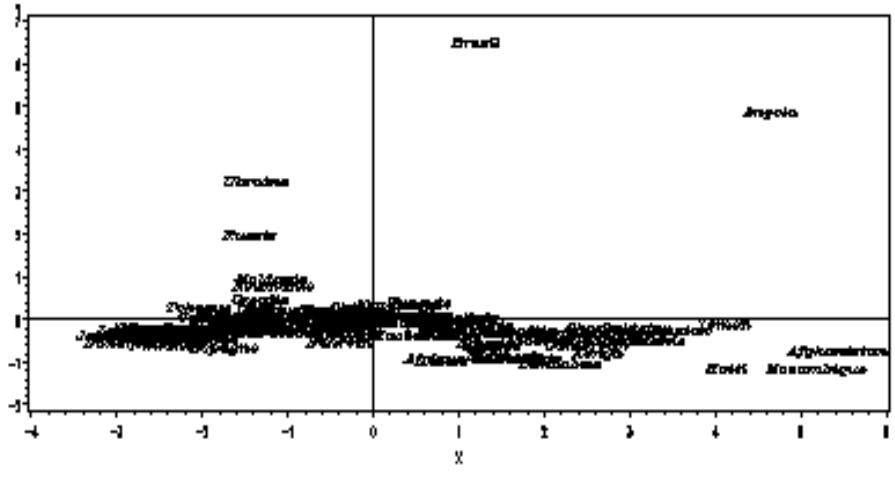
Analyse en Composantes Principales

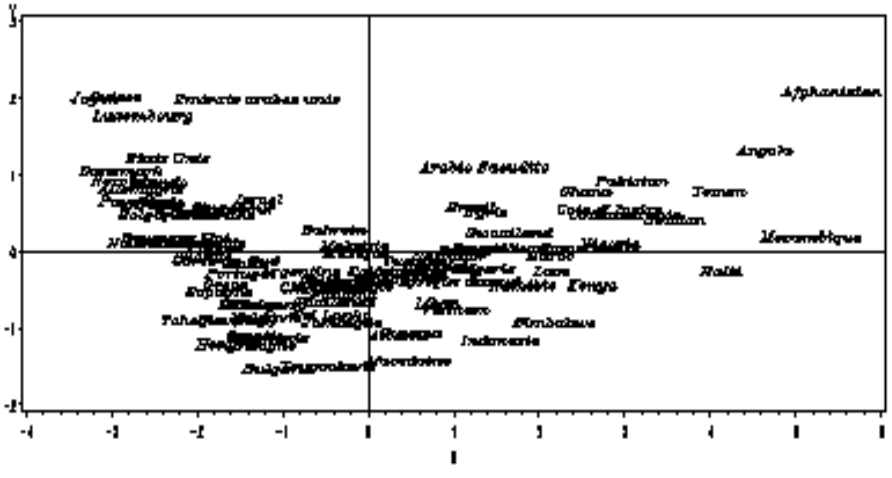


Axes 1 et 2

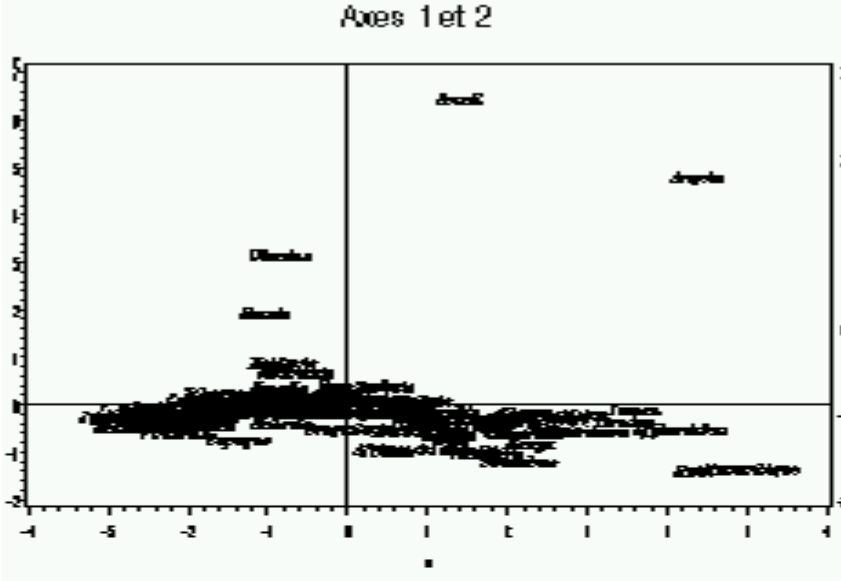


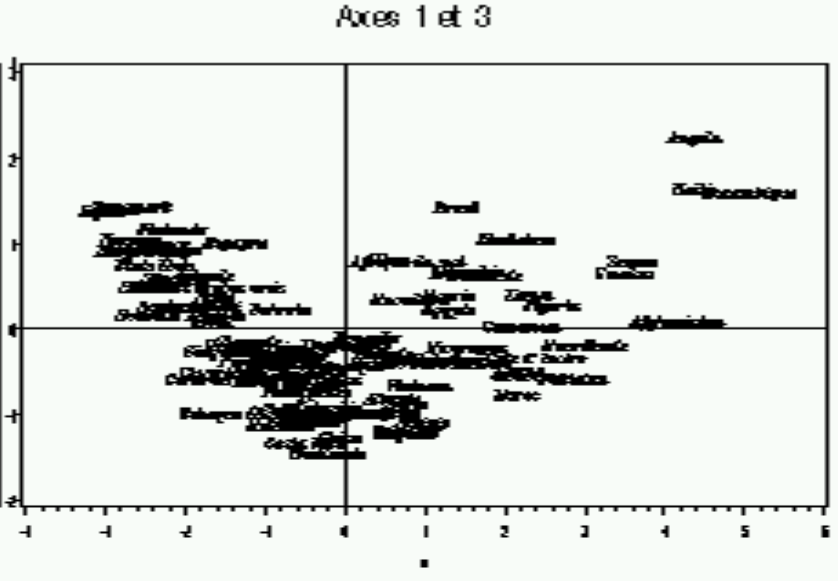
Axes 1 et 4

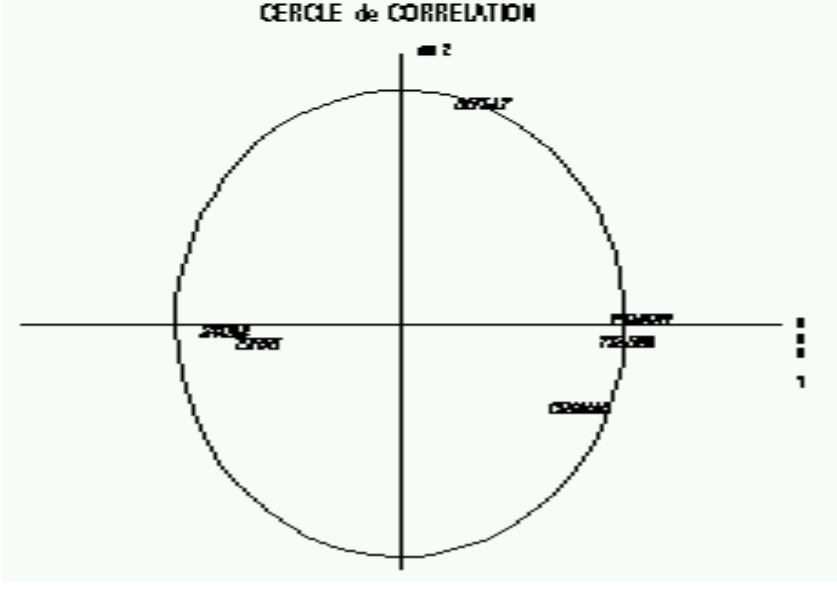


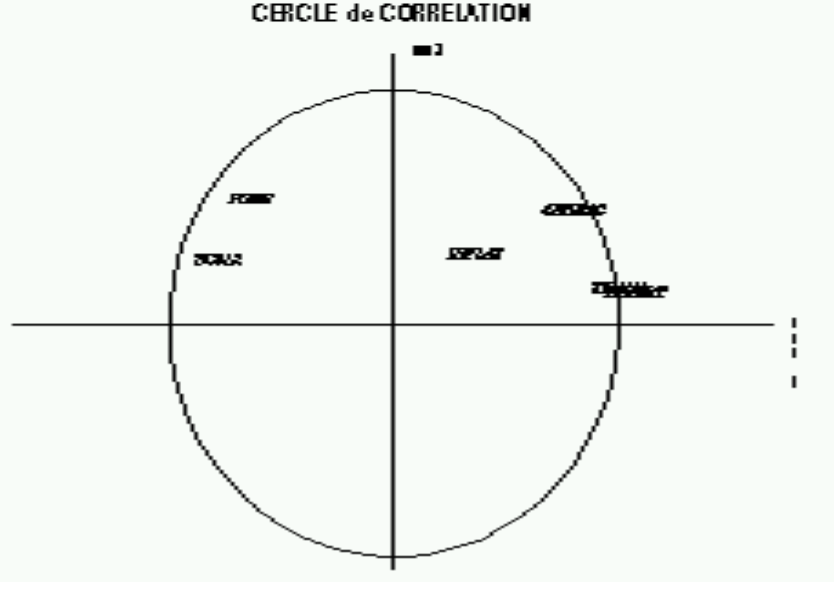


ACP


Axes 1 et 2


Axes 1 et 3


CERCLE de CORRELATION


CERCLE de CORRELATION

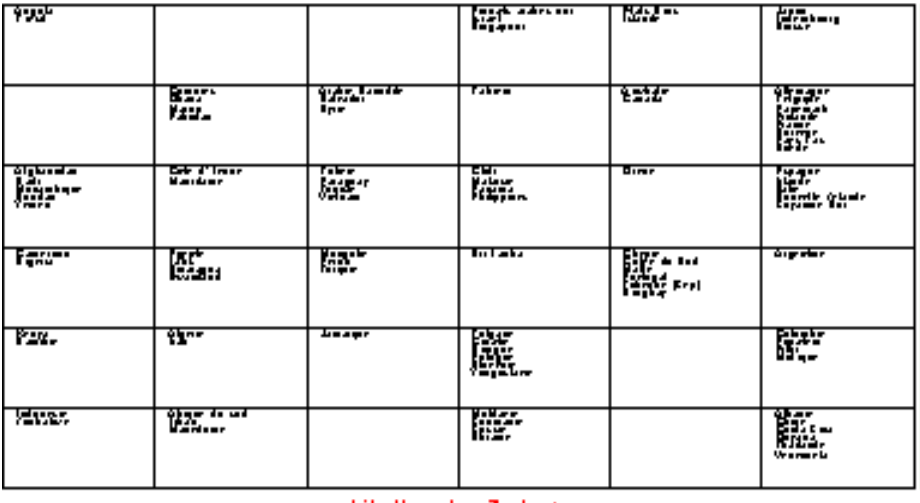
Exemple : Carte de Kohonen

KACP : grille 6x6 et 500 iterations

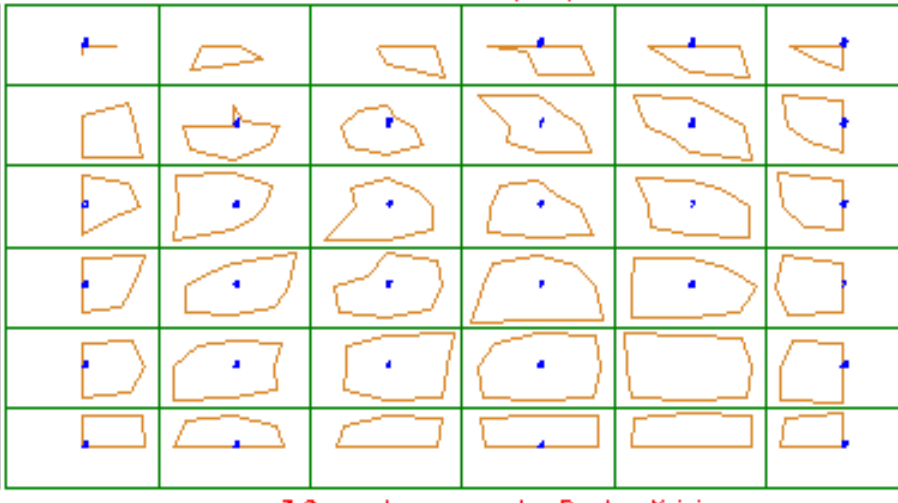
Angola Bresil			Emirats arabes uni Israel Singapour	Etats Unis Islande	Japon Luxembourg Suisse
	Comores Ghana Maroc Pakistan	Arabie Saoudite Salvador Syrie	Bahrein	Australie Canada	Allemagne Belgique Danemark Finlande France Norvege Pays Bas Suede
Afghanistan Haiti Mozambique Soudan Yemen	Cote d Ivoire Mauritanie	Bolivie Paraguay Tunisie Vietnam	Chili Malaisie Panama Philippines	Grece	Espagne Irlande Italie Nouvelle Zelande Royaume Uni
Cameroun Nigeria	Egypte Laos Nicaragua Swaziland	Mongolie Perou Turquie	Sri Lanka	Chypre Corée du Sud Malte Portugal Tchecque (Rep) Uruguay	Argentine
Kenya Namibie	Algerie Iran	Jamaique	Bulgarie Croatie Hongrie Pologne Slovénie Yougoslavie		Colombie Equateur Fidji Mexique
Indonesie Zimbabwe	Afrique du sud Liban Macedoine		Moldavie Roumanie Russie Ukraine		Albanie Chine Costa Rica Guyana Thailande Venezuela

Classes, super-classes, distances

KACP : grille 6x8 et 500 iterations

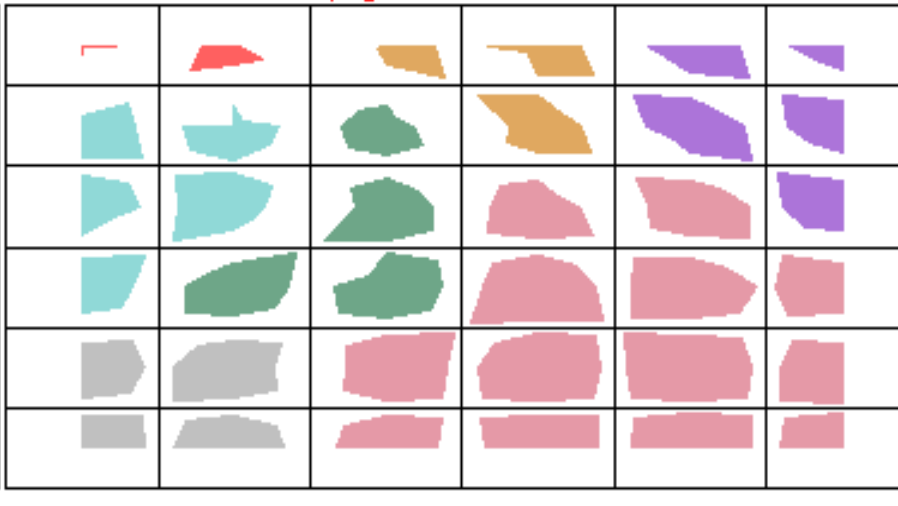


Distances (M) avec les plus proches voisins

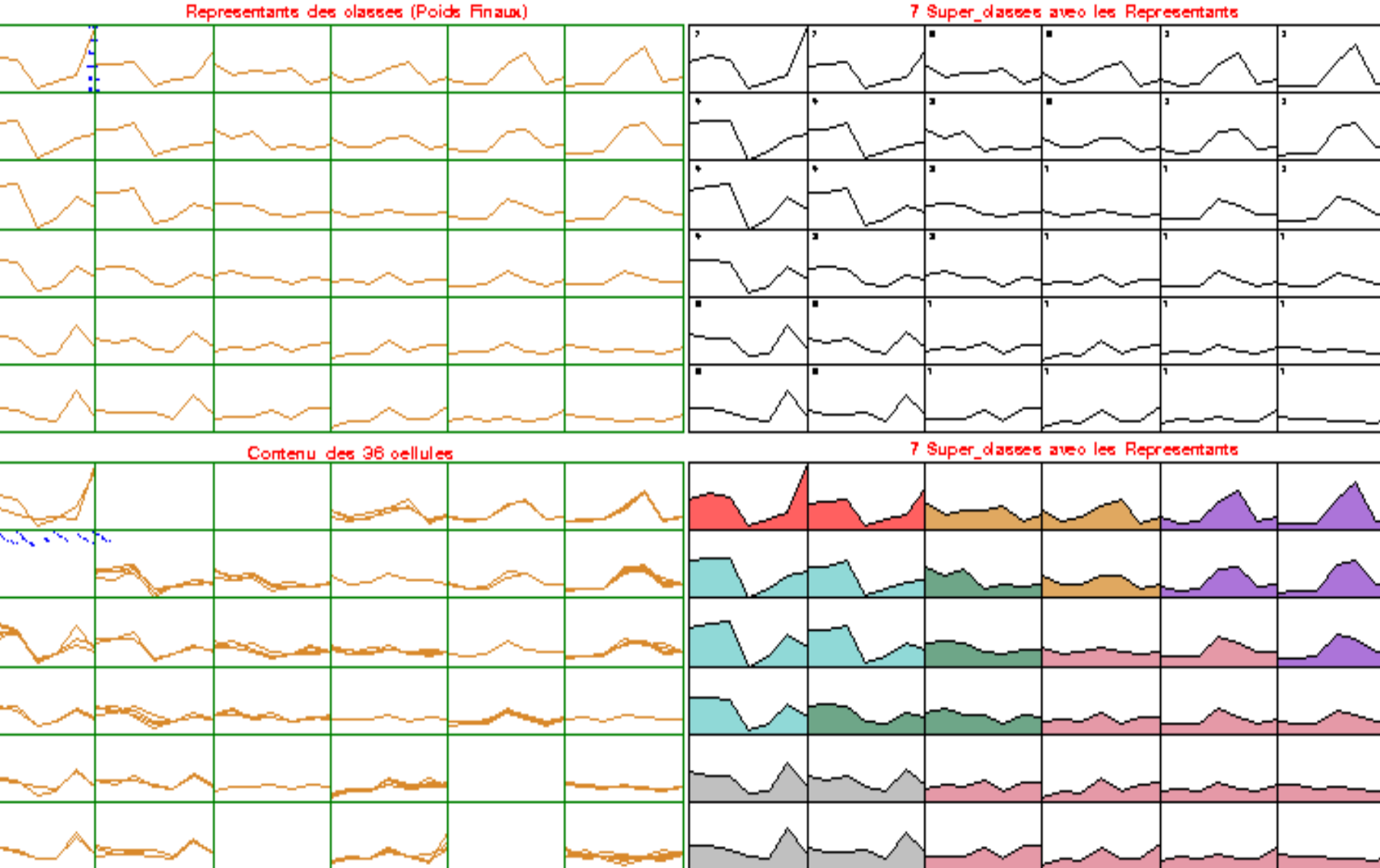


Libelles des 7 clusters

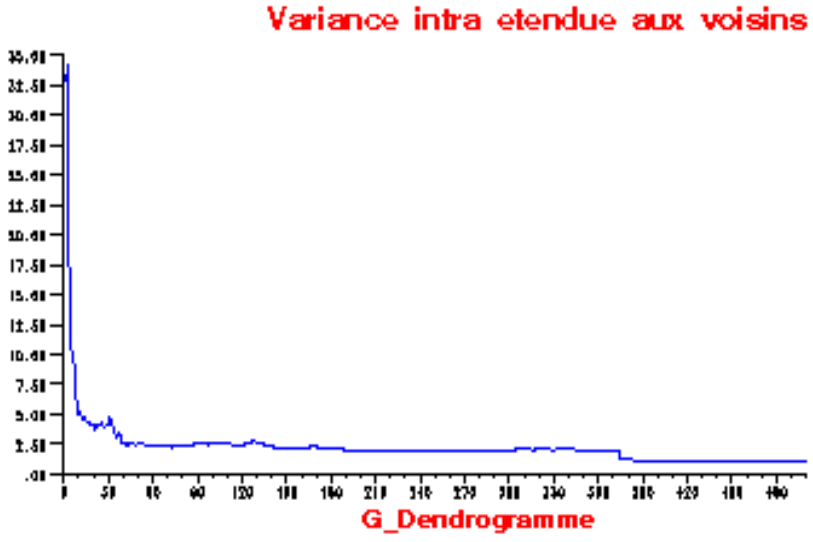
7 Super_classes avec les Proches Voisins

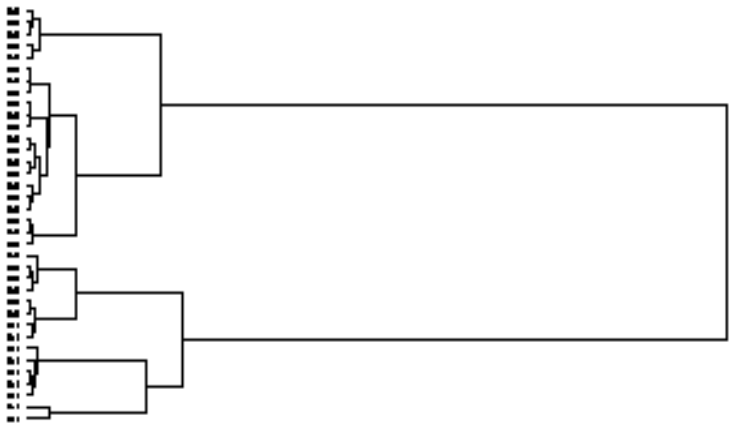


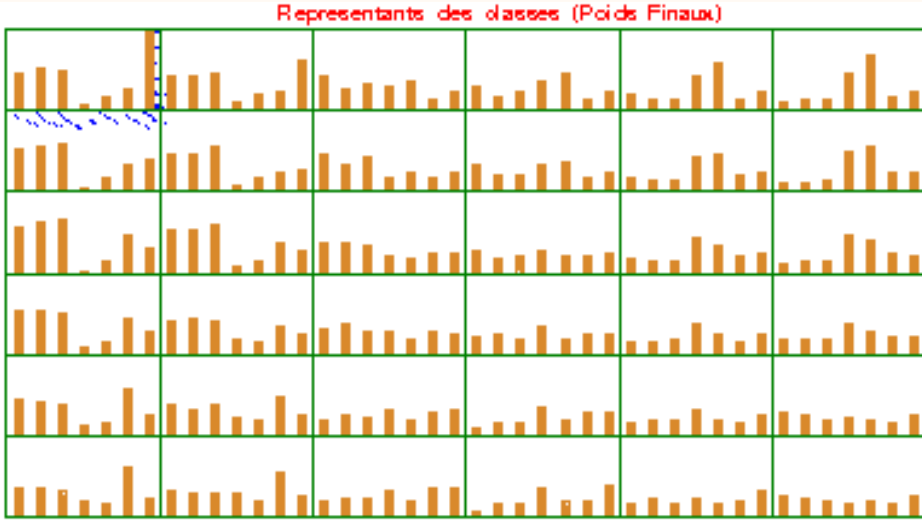
Vecteurs-codes, contenus des classes, et super-classes



Divers

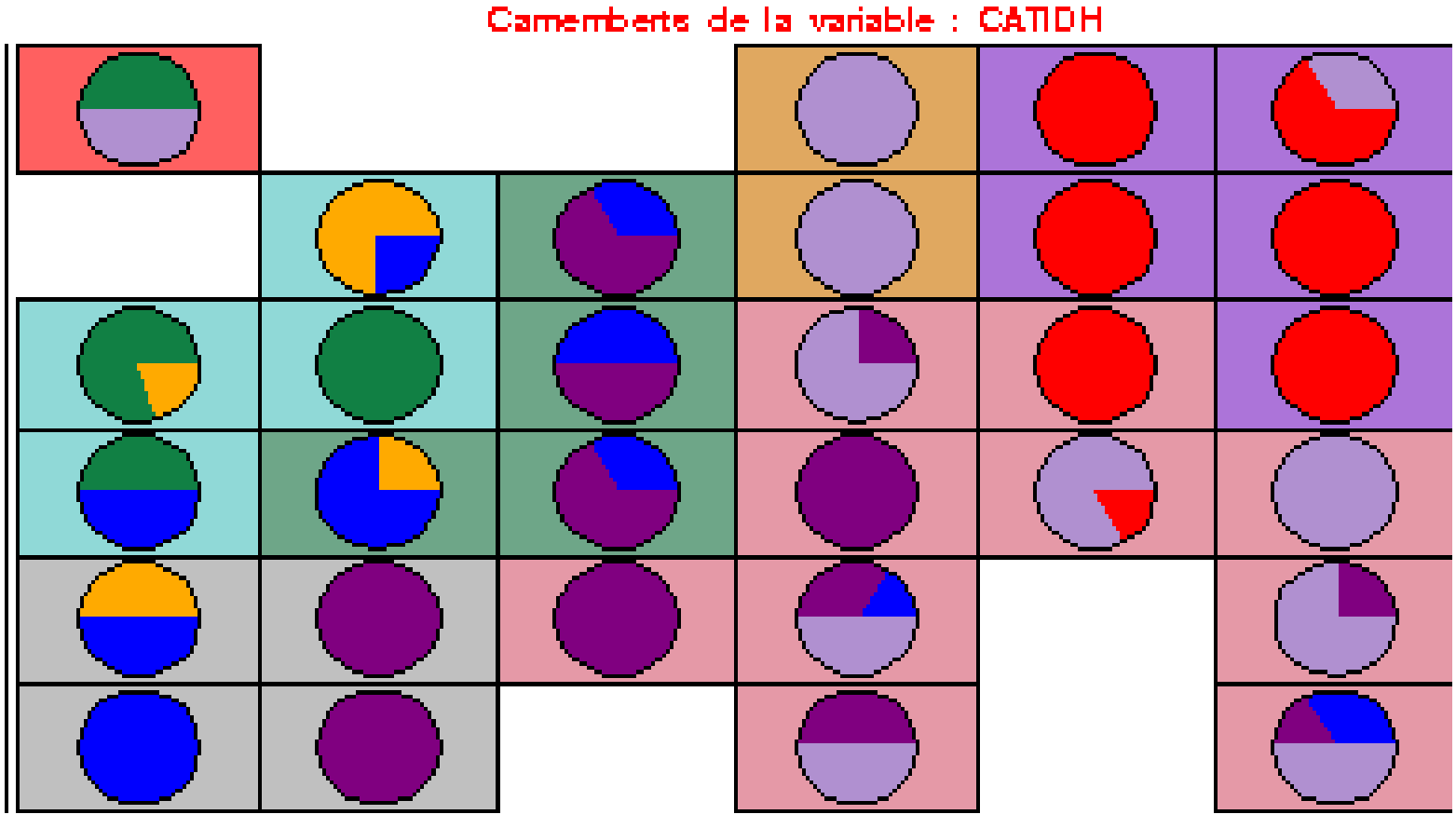

Variance intra etendue aux voisins
G_Dendrogramme

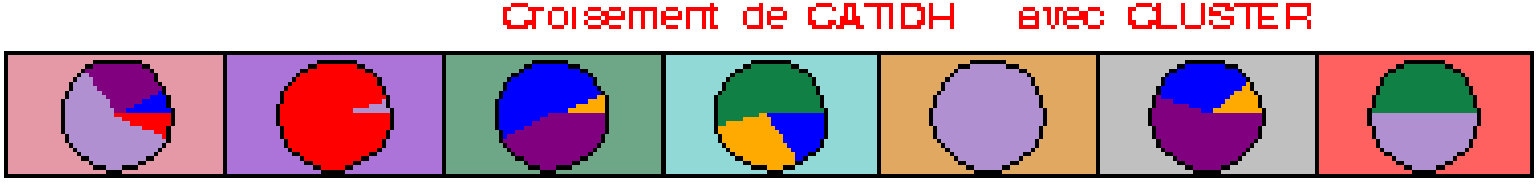



Representants des classes (Poids Finaux)

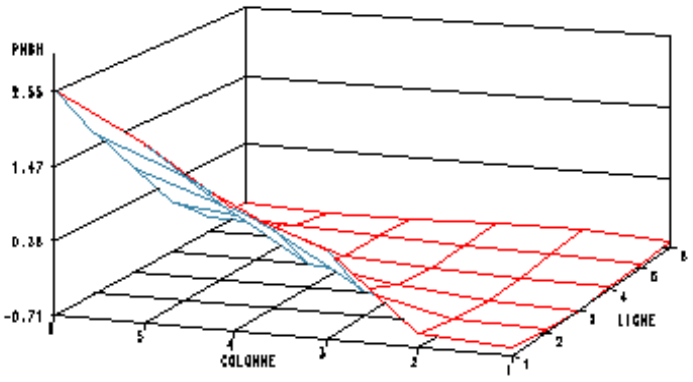

Contenu des 7 clusters

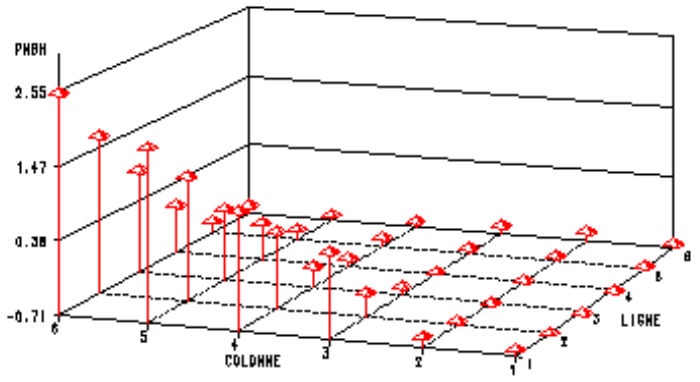
Croisement avec la variable qualitative Niveau de Développement Humain

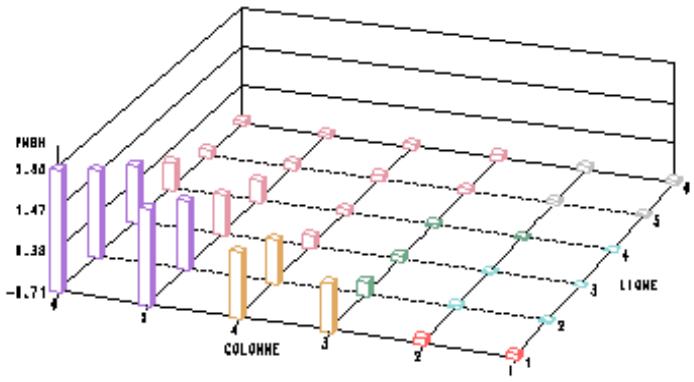




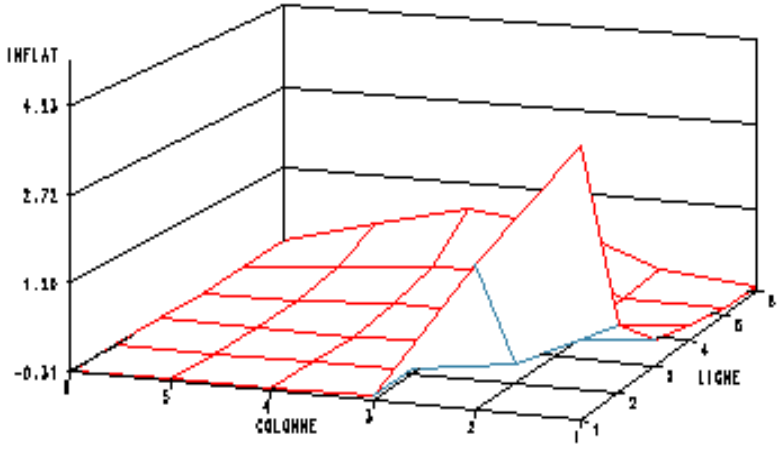
PNB par habitant le long de la grille

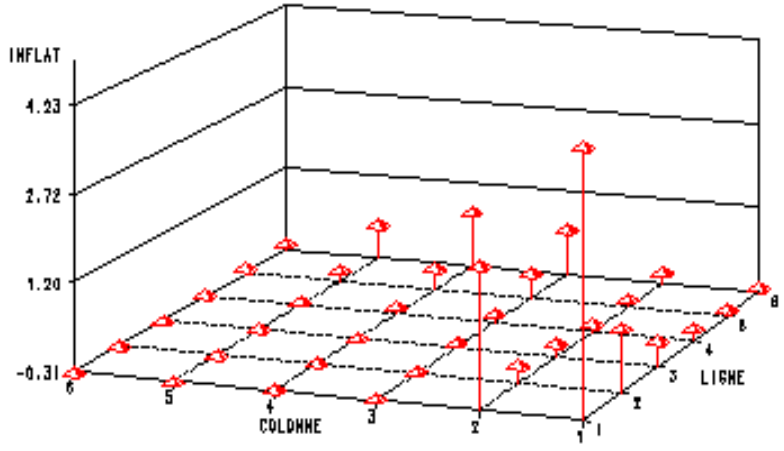


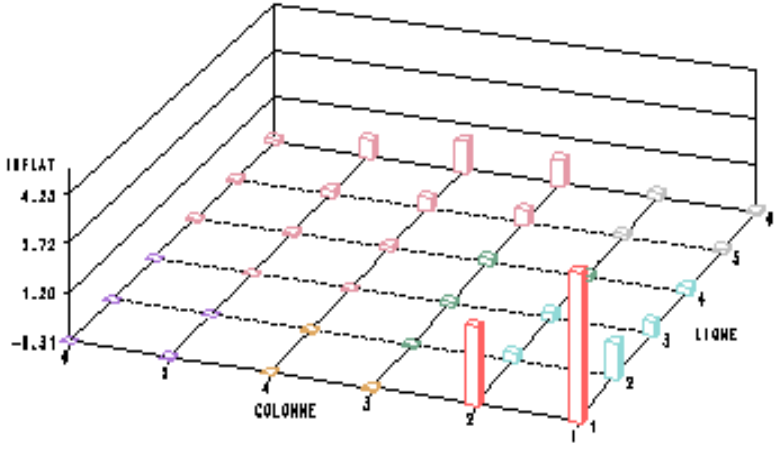




Inflation le long de la grille







Observations avec données manquantes

PAYS	ANCRX	TXMORT	TXANAL	SCOL2	PNBH	CHOMAG	INFLAT
Andorre	2,7	6,6	0,5	21,7	13235	0,1	
Bangladesh	2,3	109	65,5	19	229		4,3
Benin	3,3	118,8	69,6	12	540		38,5
Birmanie	2,1	83,8	19,5	23	924		37
Congo	2,9	64,9	25		700	25	40,2
Coree du Nord	1,7	23,9	1		595		5,1
Cuba	0,6	9,2	3	77	580		
Dominique	0	18,3	4		2588	16	1,7
Grenade	0,4	12,5	2		2500	25	2,7
Guatemala	2,9	55,5	43	24	1029		12
Guinee	3	133,9	74		507		8
Inde	1,7	87,9	48,3	49	306		10,1
Irak	2,8	56,2	41,9	44	1165		58
Jordanie	4	33,7	17	53	1078		5
Kirghizstan	1,2	21,2	15		633	0,8	281
Koweit	0,8	16,3	21	60	14342		4
Lesotho	2,7	71,4	26,7	26	700		14
Liberia	3,3	115,8	61	5	185		11
Libye	3,4	67,9	37,2		5000		25
Liechtenstein	1,2		1		35000	1,6	5,5
Turkmenistan	2,1	43,5	14		1333		2395
Tuvalu	1,5	78,4	5		2222	12,6	106
Vanuatu	2,5	43,9	47,5	20	1212		2,3
Vatican	1,1						

Classement des observations avec données manquantes

Positions des 24 Donnees Supplementaires

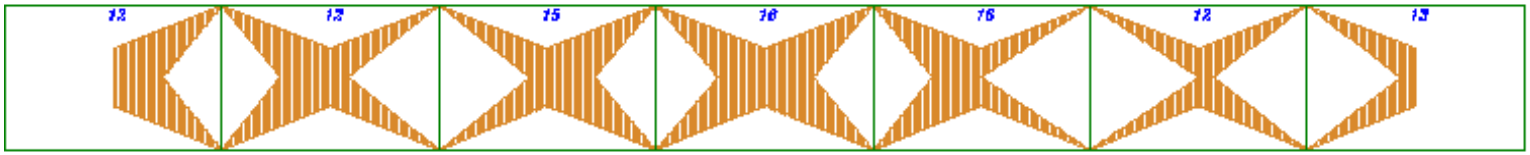
Turkmenistan			Koweit		Lieohtenstein
	Vanuatu	Irak Jordanie Libye			
Benin Guinee Liberia	Bangladesh				
	Inde	Tuvalu	Vatioan		
Birmanie Guatemala Lesotho	Congo		Cuba Dominique Grenade		
				Kirghizetan	Andorre Coree du Nord

KACP sur une ficelle 7

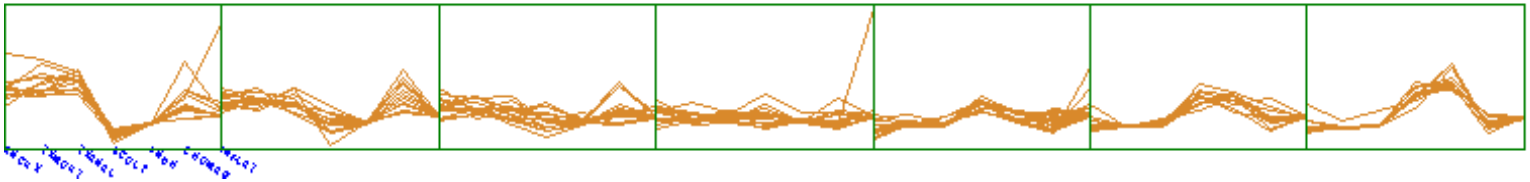
KACP : ficelle de 7 pour 500 iterations

Afghanis Angola Comores Cote d' Ghana Haiti Mauritan Mozambiq Nigeria	Pakistan Soudan Yemen	Algerie Cameroun Egypte Indonesi Iran Kenya Laos Maroc Namibie	Nicaragu Swaziln Zimbabwe	Afrique Albanie Arabie S Bolivie Guyana Liban Macedoin Mongolie Paraguay	Perou Salvador Syrie Tunisie Turquie Vietnam	Bahrein Bresil Chine Colombie Costa R Equateur Fidji Jamaique Malaisie	Mexique Panama Philippi Sri Lank Thailand Venezuel Yougosla	Argentin Bulgarie Chili Coree du Croatie Hongrie Malte Moldavie Pologne	Portugal Roumanie Russie Slovenie Tcheque Ukraine Uruguay	Australi Canada Chypre Espagne Finlande Grece Irlande Israel Italie	Nouvelle Royaume Singapou	Allemagn Belgique Danemark Emirats Etats Un France Islande Japon Luxembou	Norvege Pays Bas Suede Suisse

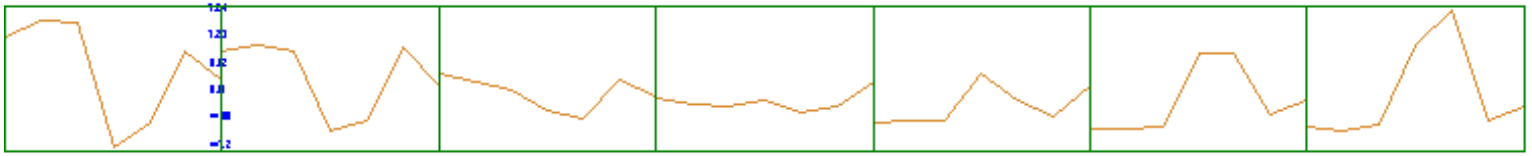
Distances (M) avec les plus proches voisins



Contenu des 7 cellules

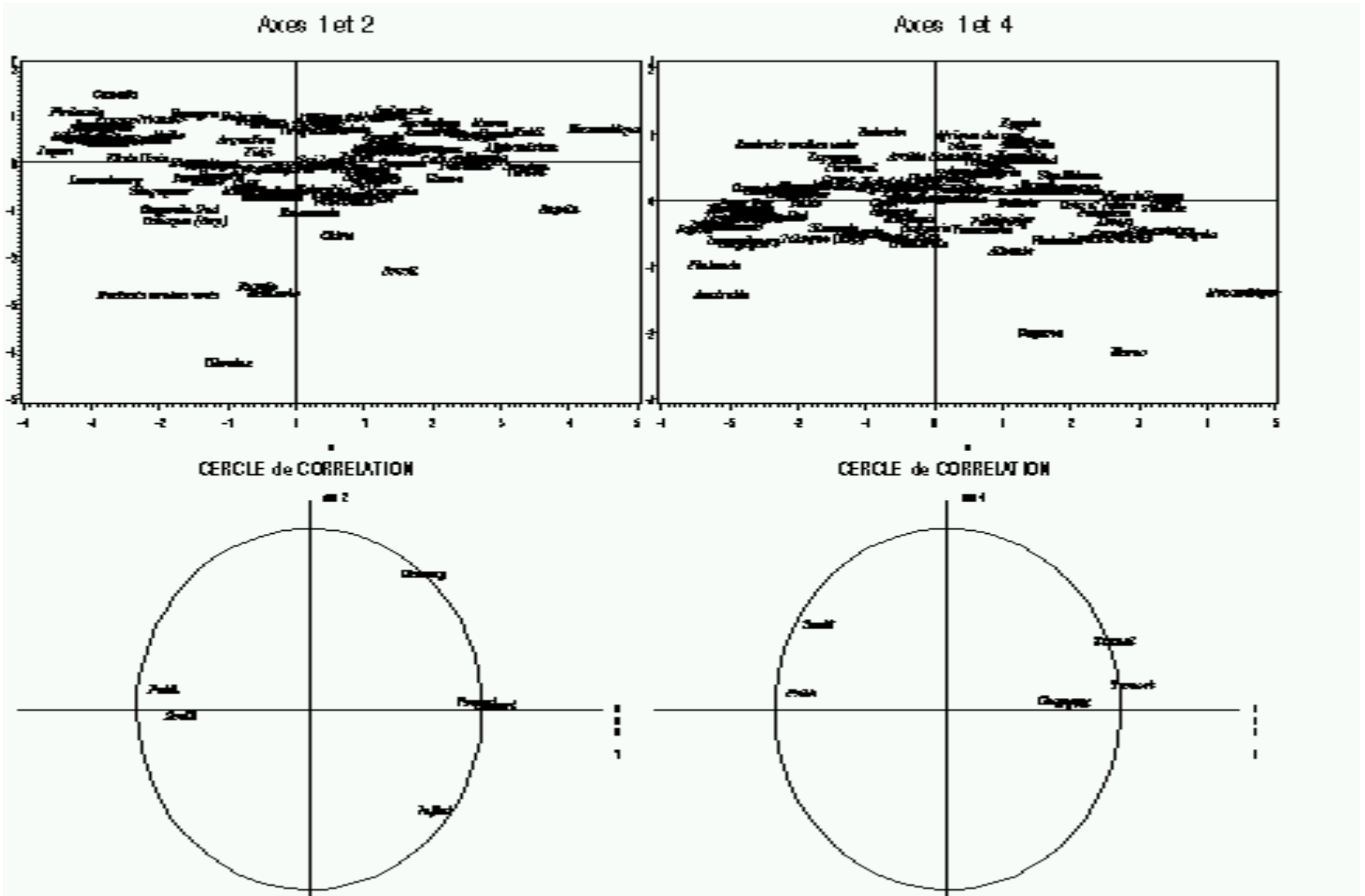


Representants des classes (Poids Finaux)



Representants des classes (Poids Finaux)

Données corrigées (logarithmes)



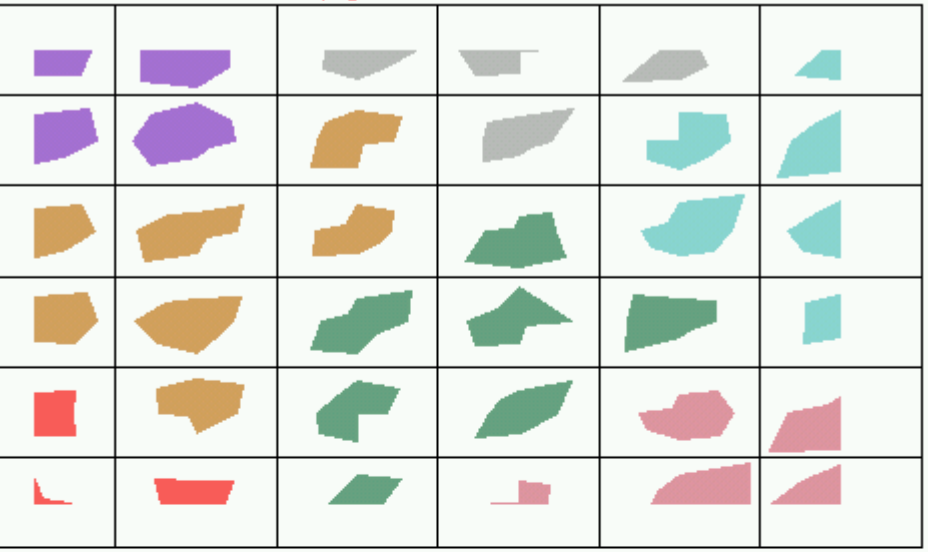
Données corrigées (logarithmes)

- Cela améliore légèrement les représentations graphiques de l ’ACP (on passe de 74% pour les deux premiers axes à 79% en prenant les données corrigées).
- Les pays sont un peu plus séparés.
- Cependant, cela ne change presque rien à la carte de Kohonen

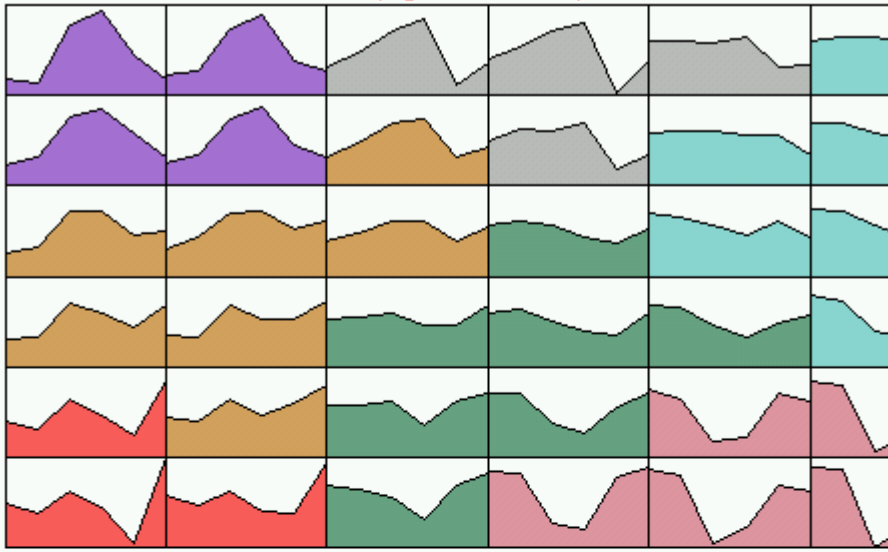
- Sauf les éventuelles rotations, les pays riches sont en haut à gauche

Classification, données corrigées

7 Super_classes avec les Proches Voisins

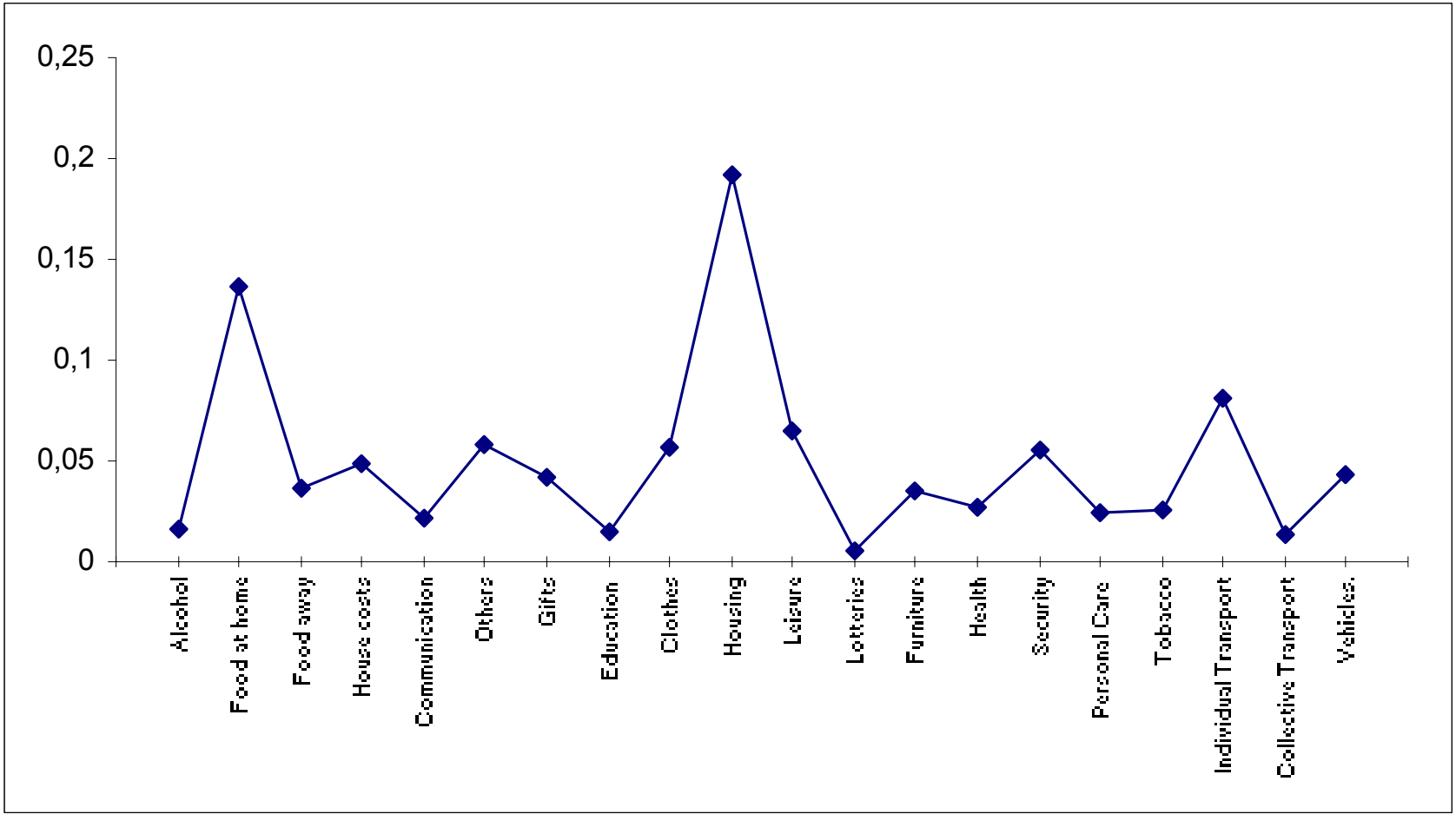


7 Super_classes avec les Representants

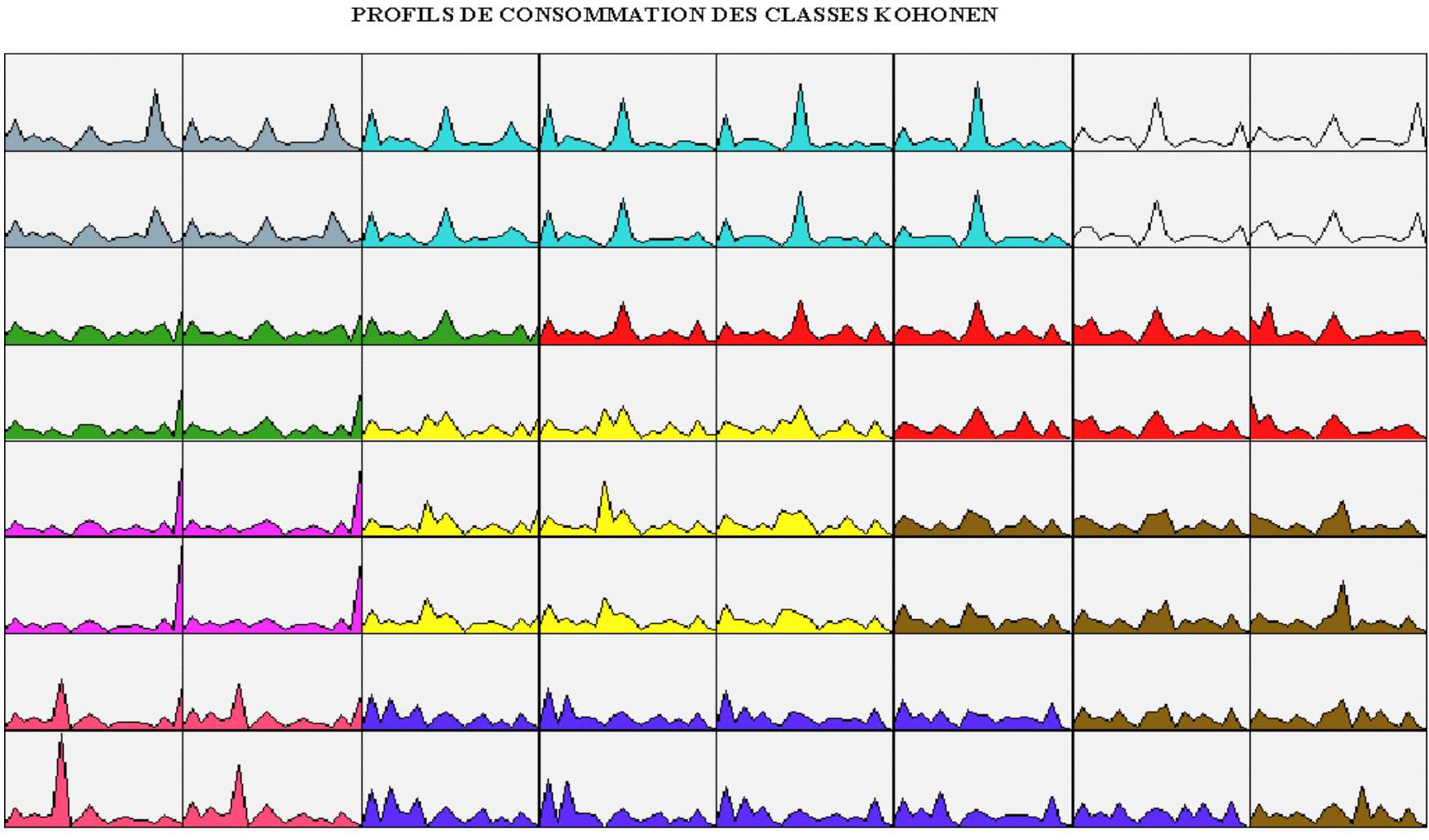


Libelles des 7 clusters

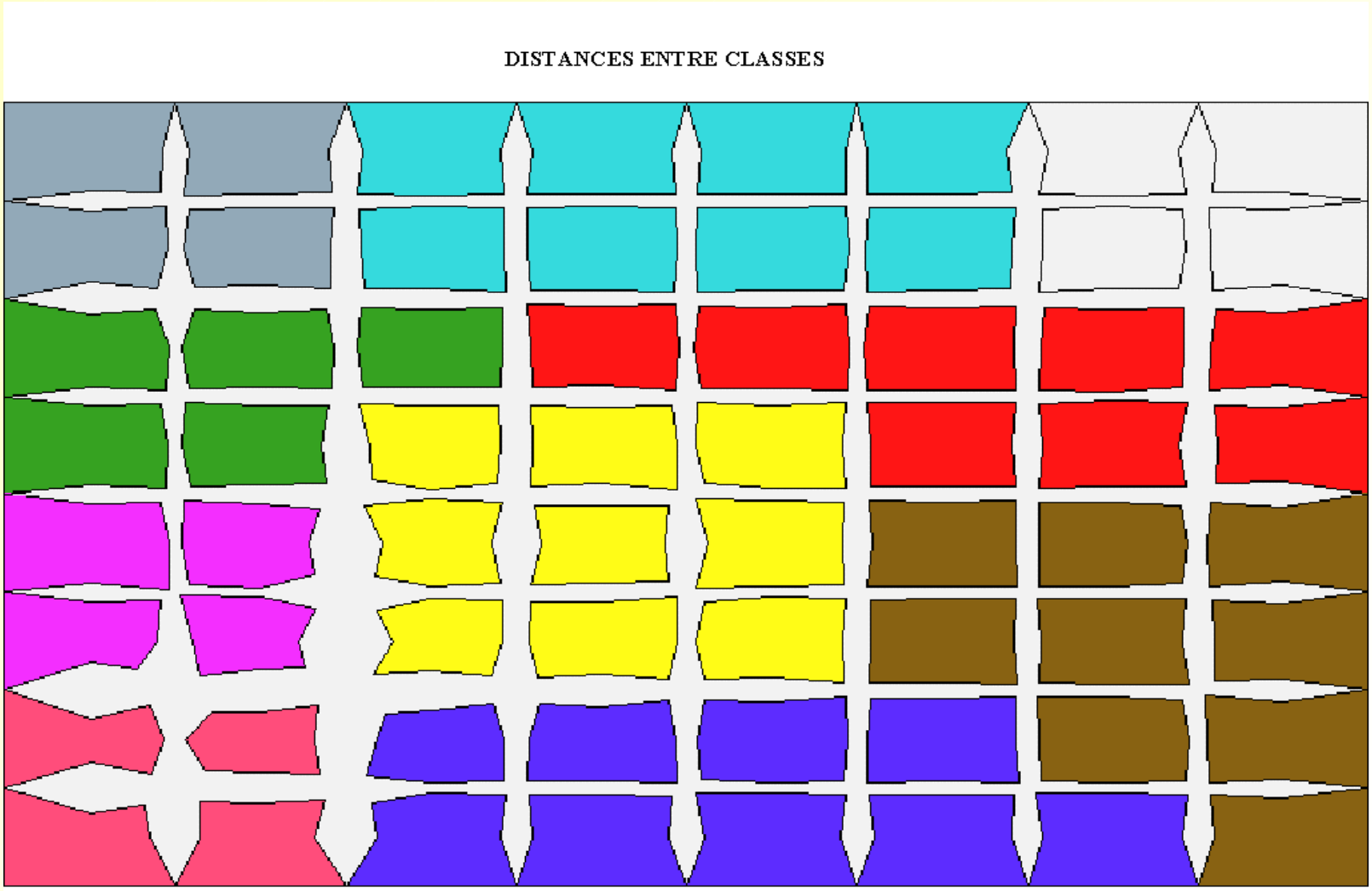
Profil de consommation d ’un foyer canadien (1992)



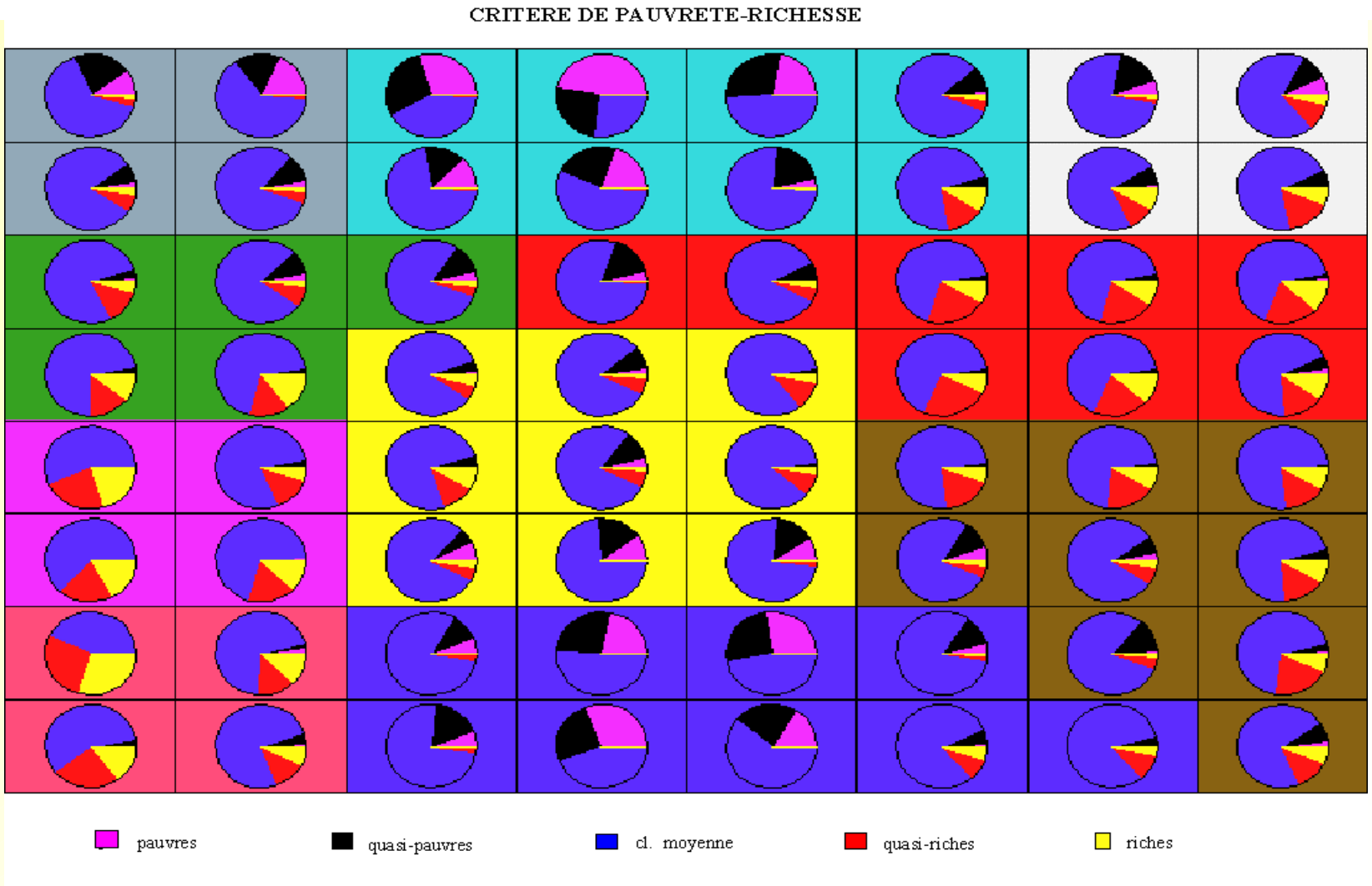
Exemple : la consommation des foyers canadiens



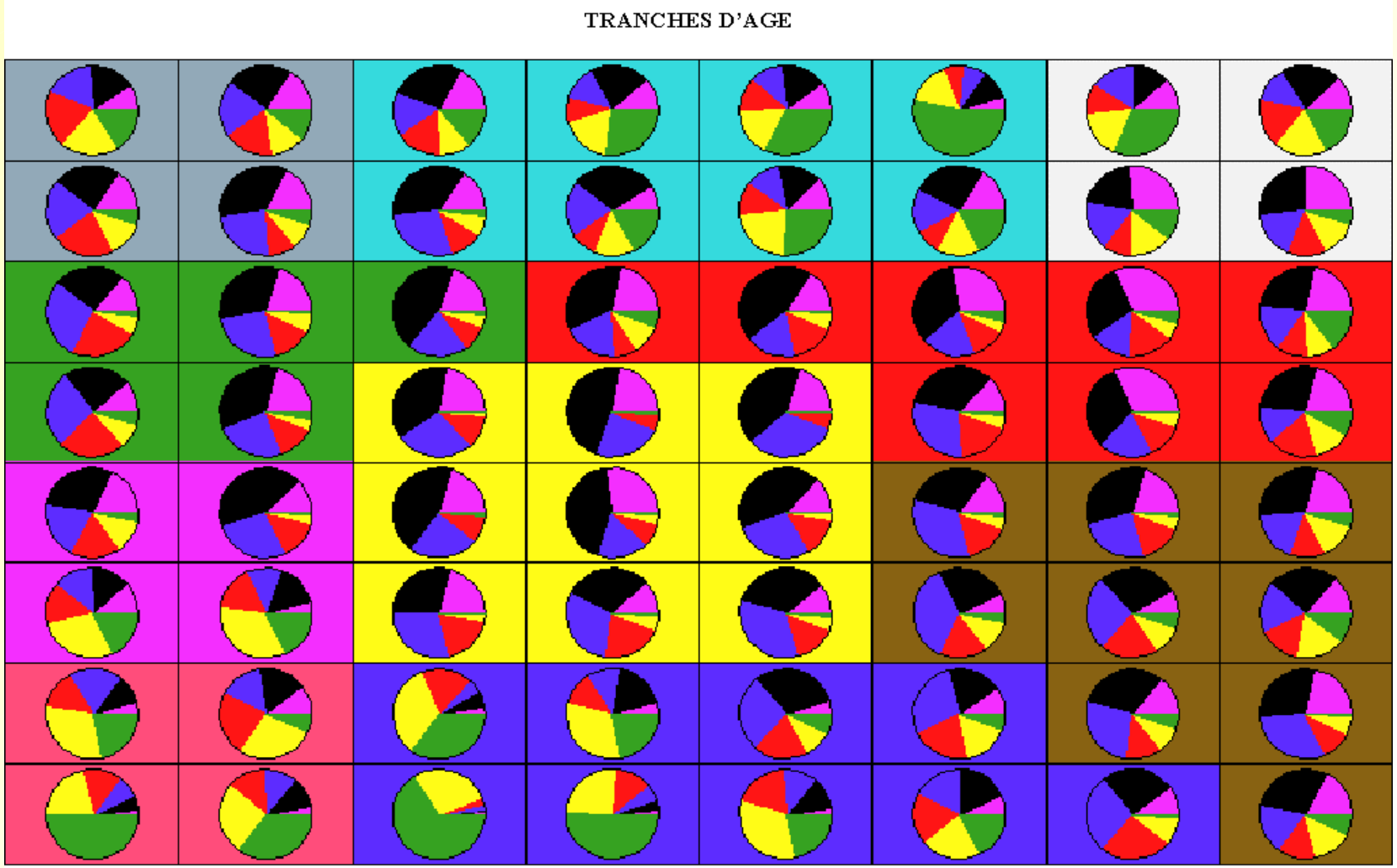
Les distances entre les classes de l'exemple précédent



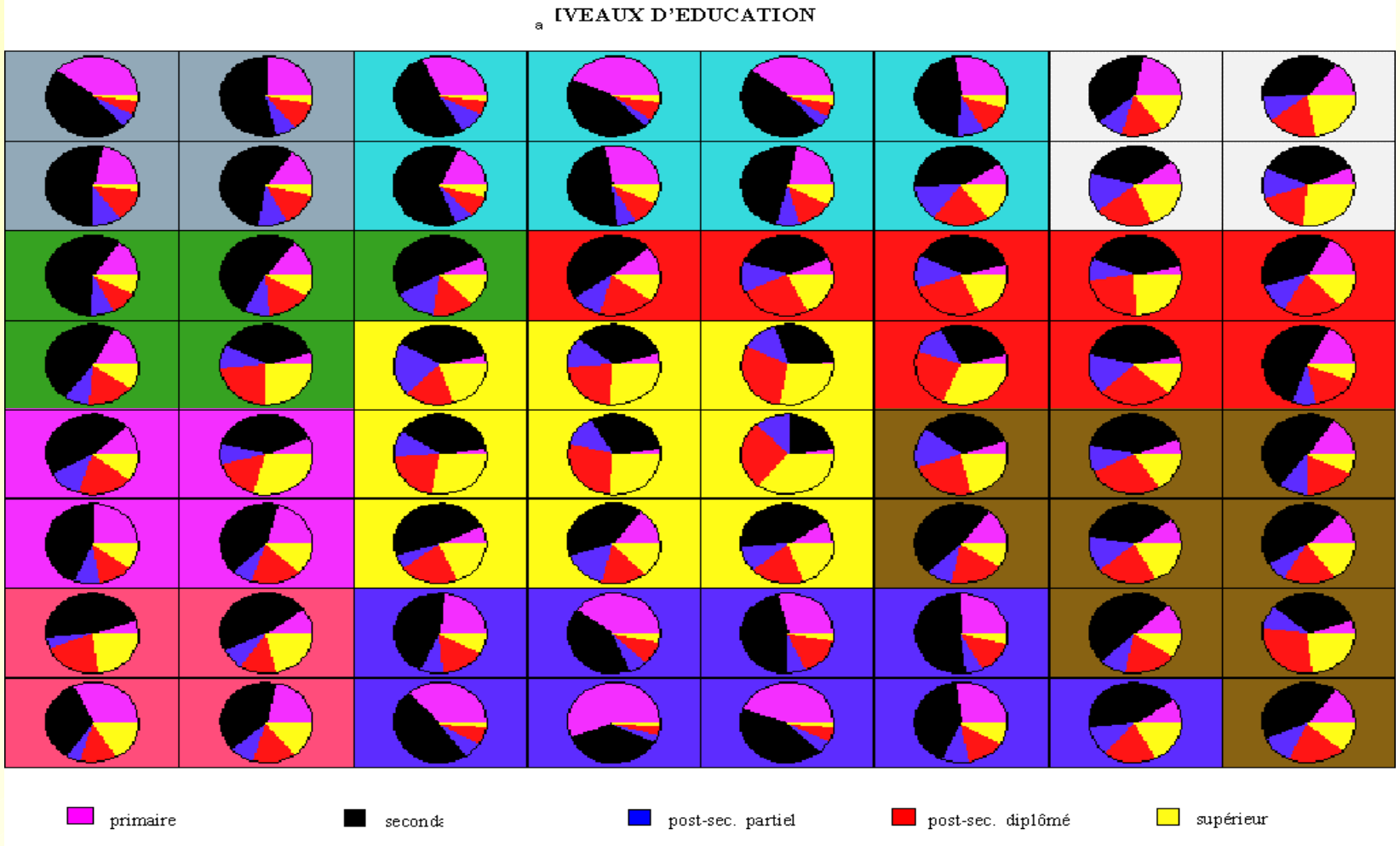
Croisement avec une variable qualitative supplémentaire (le niveau de richesse pour les consommateurs canadiens)



Croisement avec une variable qualitative supplémentaire (l ’âge des consommateurs canadiens)



Croisement avec une variable qualitative supplémentaire (le niveau d ’éducation pour les consommateurs canadiens)



Analyse de données : introduction

Algorithme de Kohonen

Kohonen et classification : KACP

Traitements des variables qualitatives

Conclusion

Analyse des relations entre modalités de variables qualitatives
Analyse d'une table de Burt (KACM)

- Classiquement, l'analyse des correspondances des modalités de plus de 2 variables qualitatives se fait par **l'analyse des correspondances multiples, qui est une analyse en composantes principales pondérée sur la table de Burt associée**. La distance considérée est la distance du χ^2. La table de Burt est un tableau de contingence généralisé, qui croise toutes les variables qualitatives deux à deux.

- On pratique ici un algorithme de Kohonen sur cette table de Burt, avec la même pondération et la distance du χ^2.

- Les modalités associées se retrouvent dans la même classe ou dans des classes voisines.

Exemple de table de Burt

	Q1_1	Q1_2	Q1_3	Q2_1	Q2_2	Q3_1	Q3_2	Q3_3	Q3_4
Q1_1	4	0	0	2	2	1	0	1	2
Q1_2	0	5	0	2	3	0	1	3	1
Q1_3	0	0	3	2	1	1	2	0	0
Q2_1	2	2	2	6	0	2	2	1	1
Q2_2	2	3	1	0	6	0	1	3	2
Q3_1	1	0	1	2	0	2	0	0	0
Q3_2	0	1	2	2	1	0	3	0	0
Q3_3	1	3	0	1	3	0	0	4	0
Q3_4	2	1	0	1	2	0	0	0	3

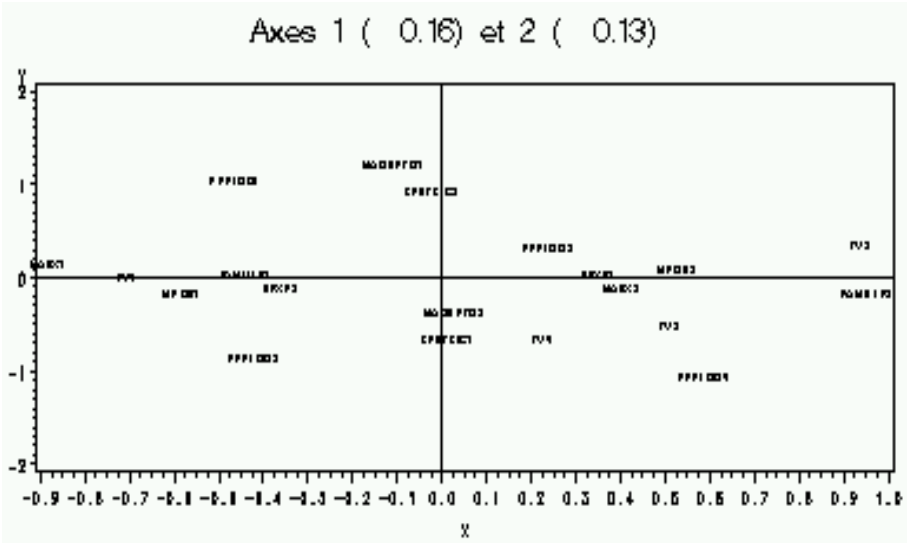
$$n_{ij} \propto \frac{n_{ij}}{\sqrt{n_{i.}}\sqrt{n_{.j}}}$$

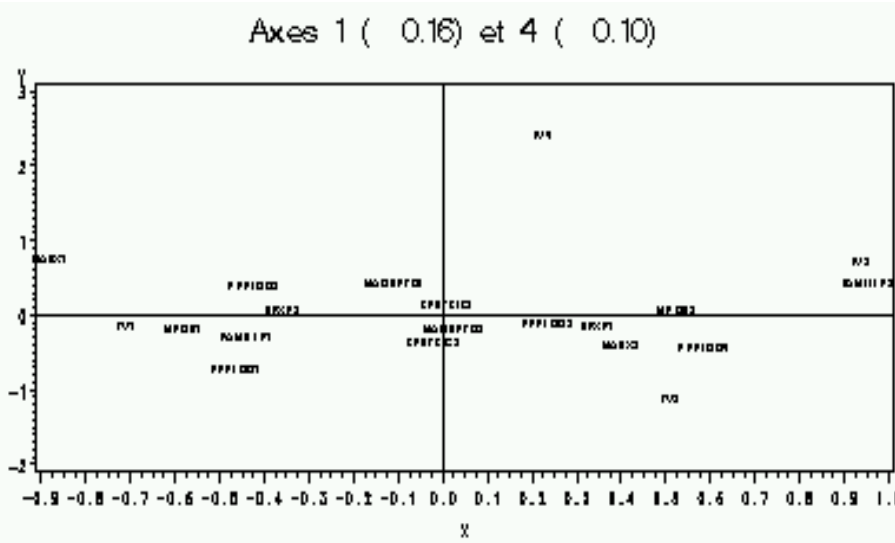
Un exemple

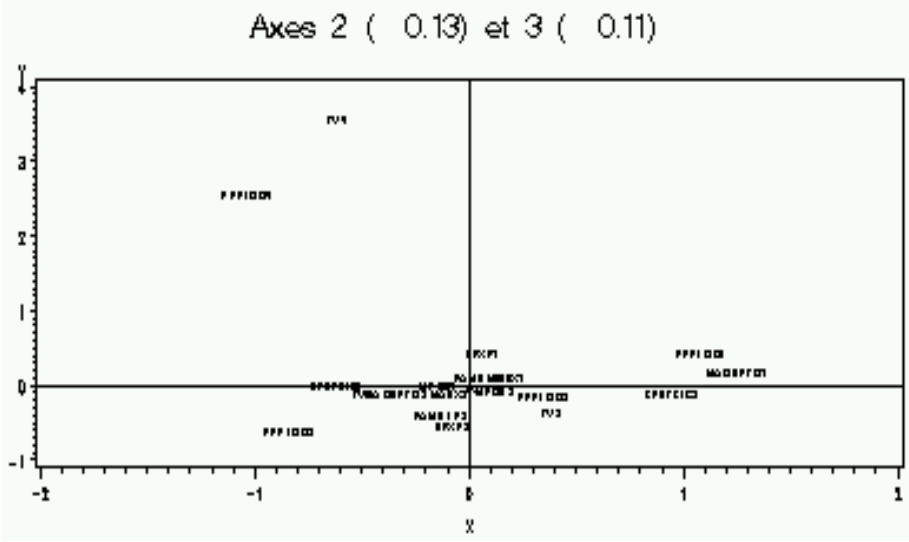
- Tiré de « Statistique exploratoire multidimensionnel » de Lebart, Morineau, Piron, (Dunod) 1995

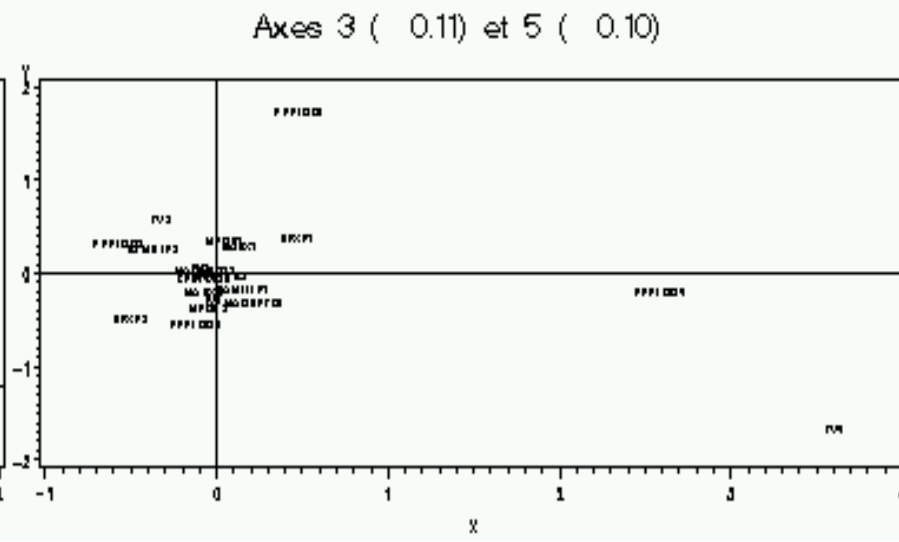
- 105 ménages, 8 questions, 20 modalités
  - La famille est l'endroit où on se sent bien :oui, non
  - Les dépenses de logement sont une charge : négligeable, sans gros problème, une lourde charge, une très lourde charge
  - Avez-vous eu récemment mal au dos : oui, non
  - Vous imposez-vous des restrictions : oui, non
  - Sexe de l 'enquêté : masculin, féminin
  - avez-vous un magnétoscope : oui, non
  - Avez-vous eu récemment des maux de tête : oui, non
  - Regardez-vous la télévision : tous les jours, assez souvent, pas très souvent, jamais

Analyse des correspondances multiples

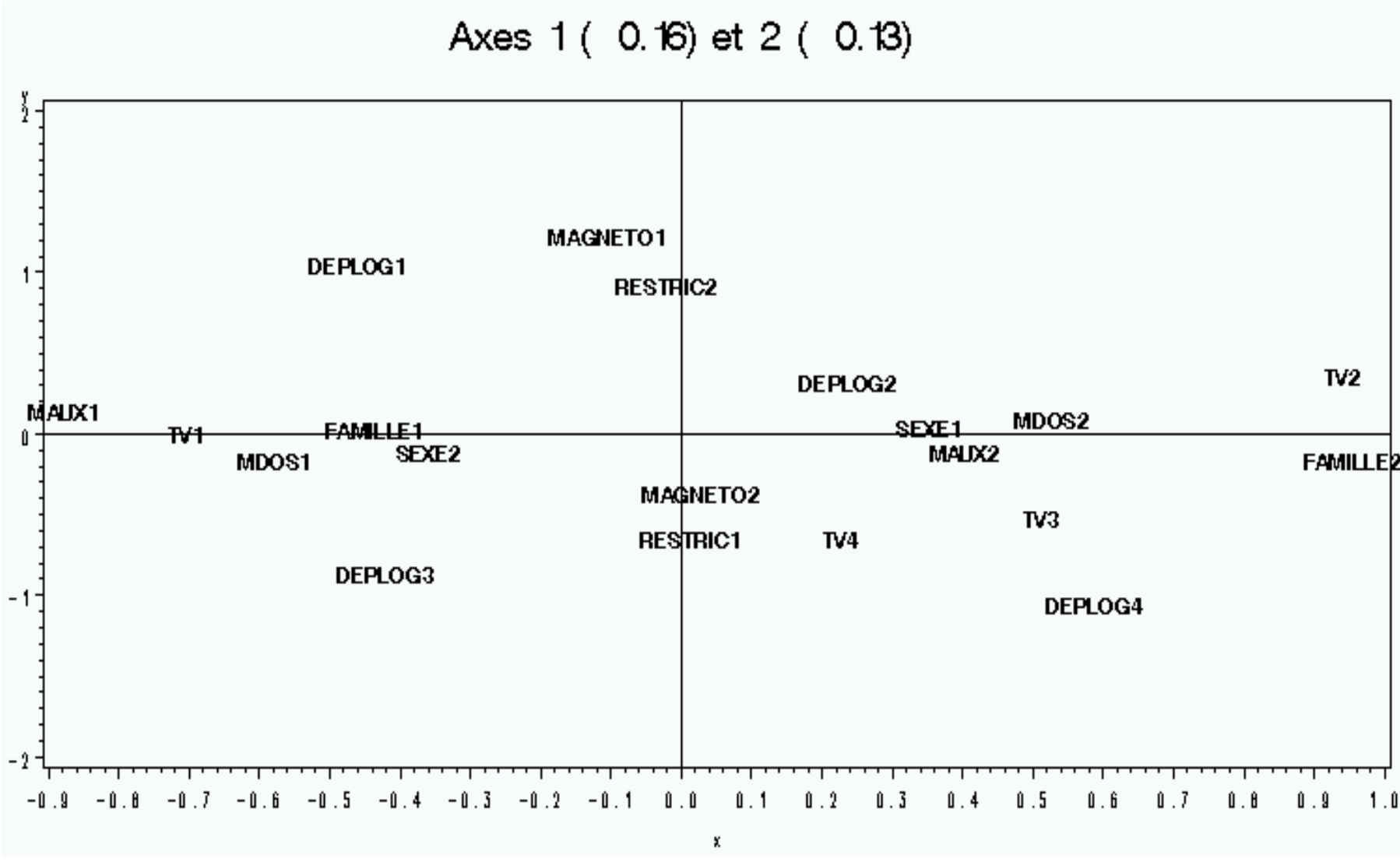








ACM modalités, deux axes

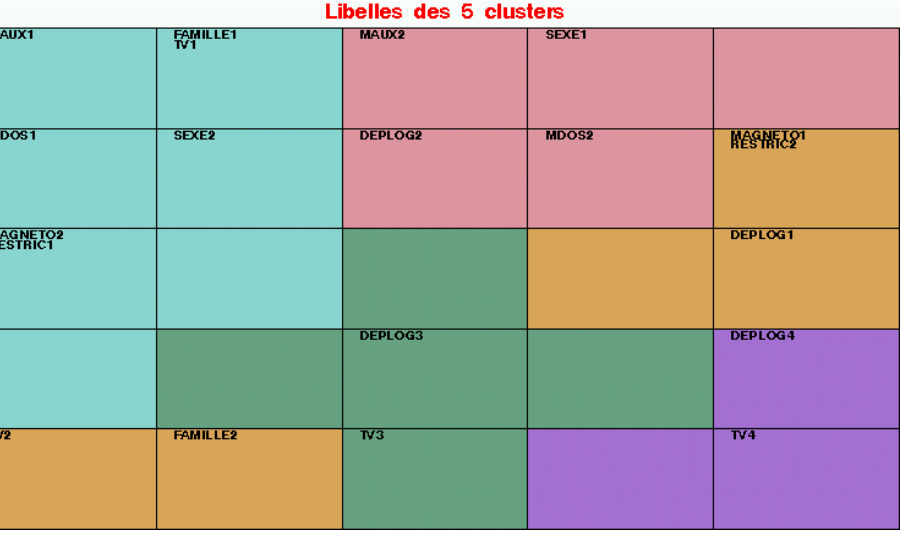


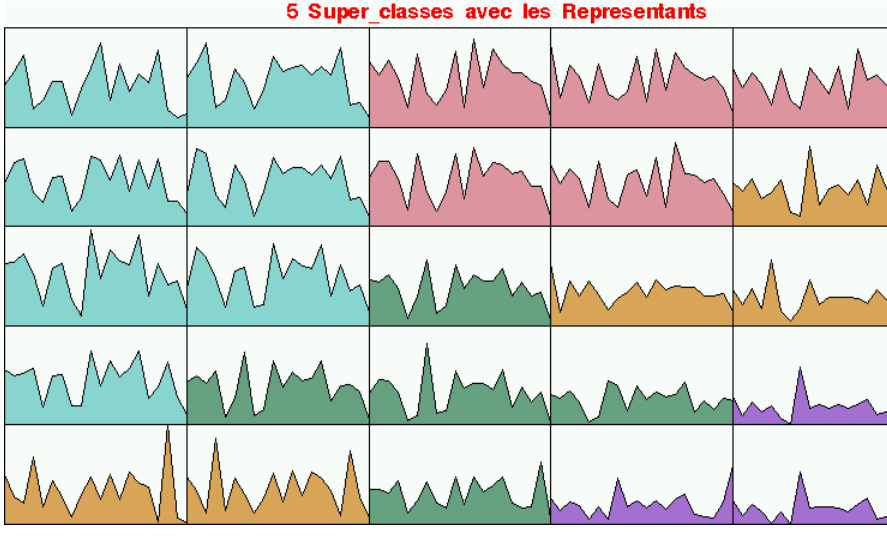
Carte des modalités

KACM : grille 5x5 et 200 iterations

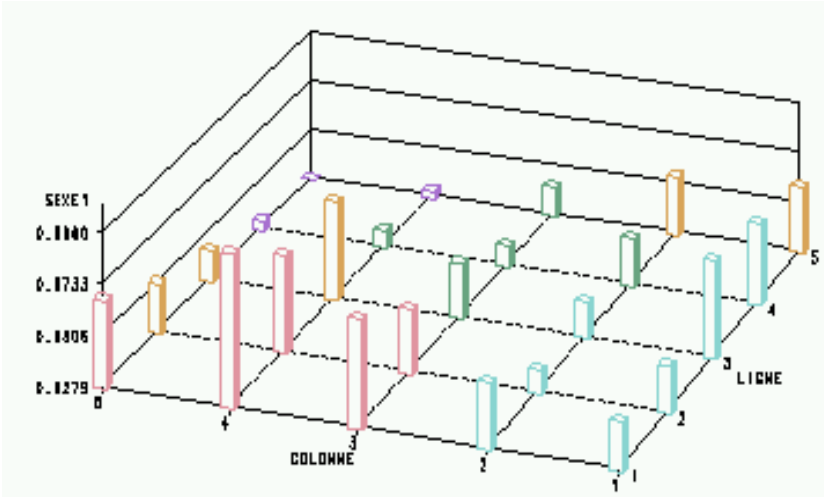
MAUX1	FAMILLE1 TV1	MAUX2	SEXE1	
MDOS1	SEXE2	DEPLOG2	MDOS2	MAGNETO1 RESTRIC2
MAGNETO2 RESTRIC1				DEPLOG1
		DEPLOG3		DEPLOG4
TV2	FAMILLE2	TV3		TV4

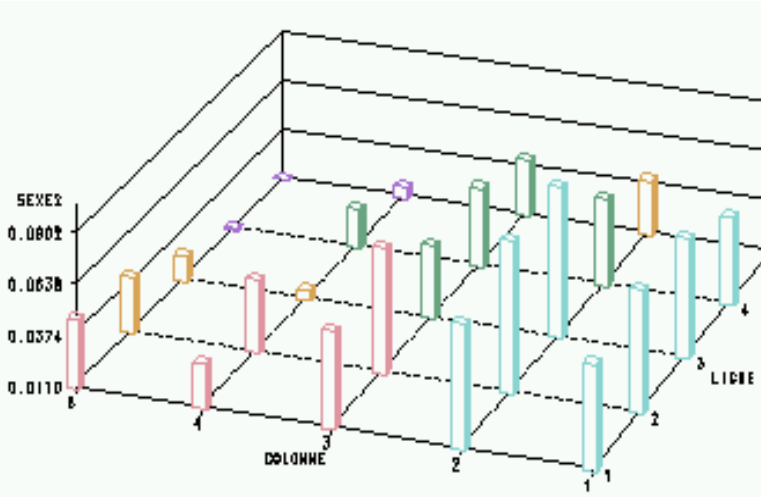
Super classes pour les modalités

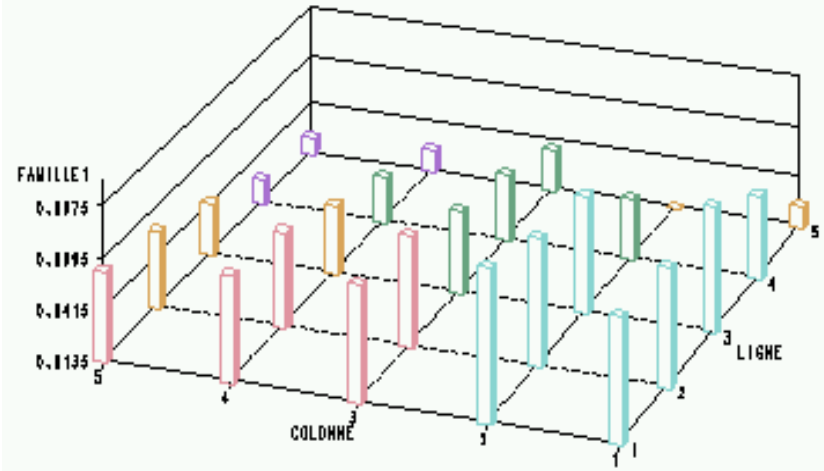


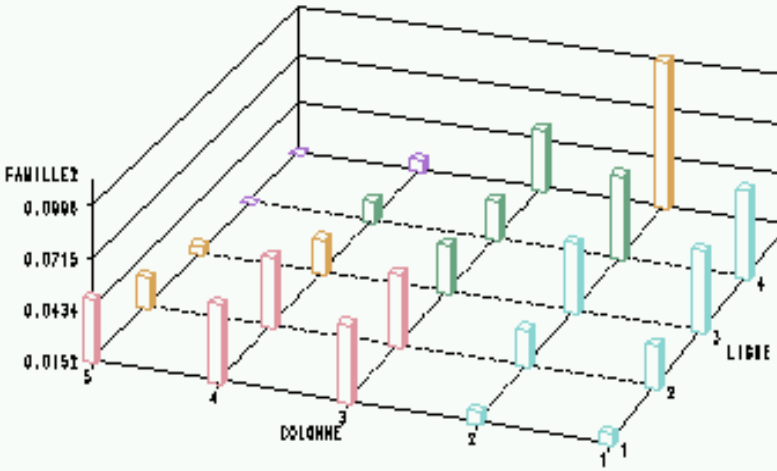


Modalités (sexe et famille) le long de la grille

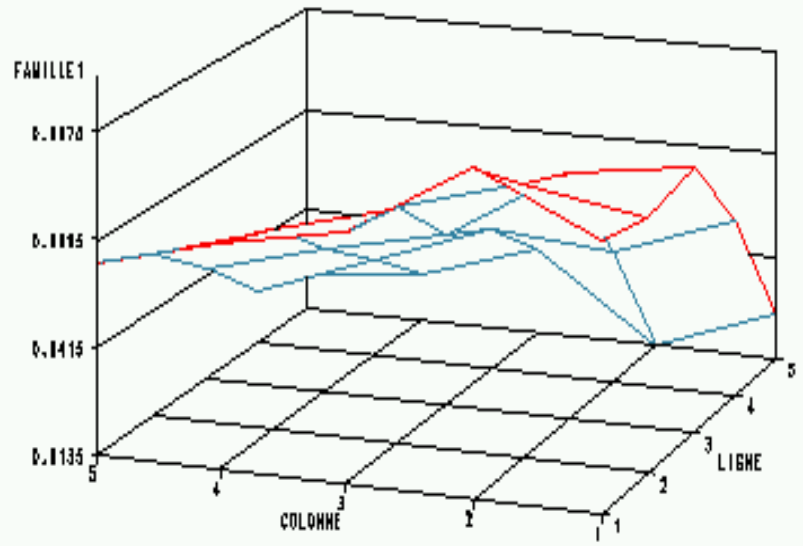


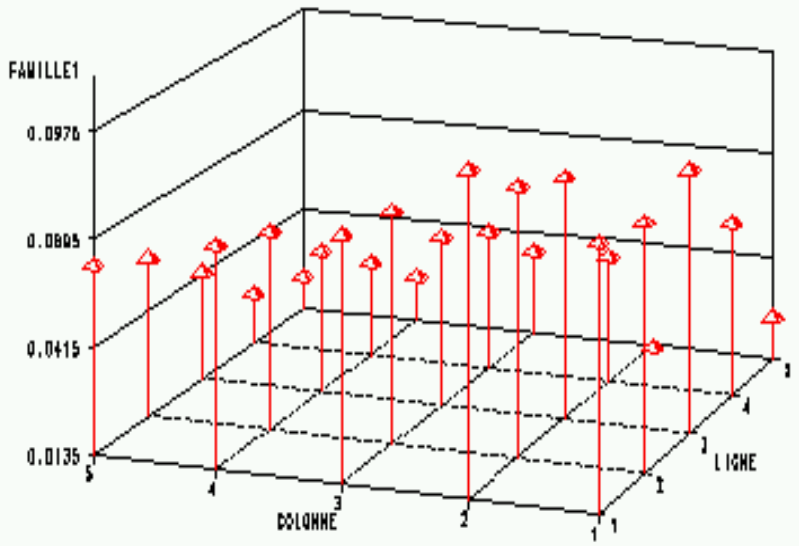


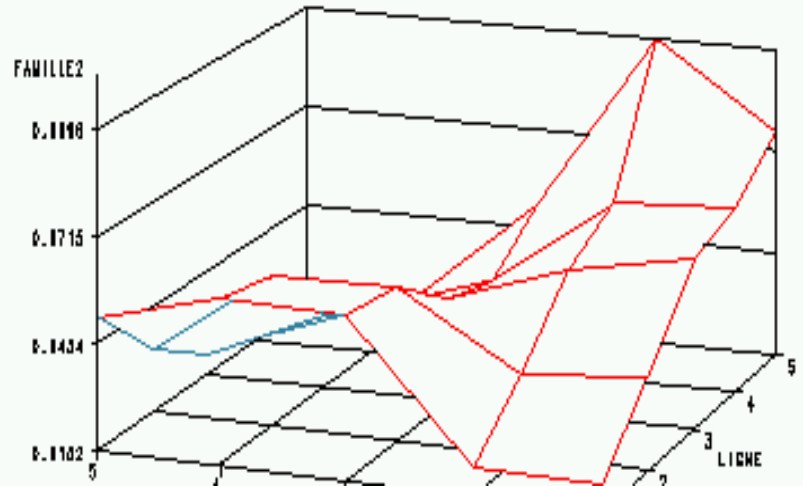


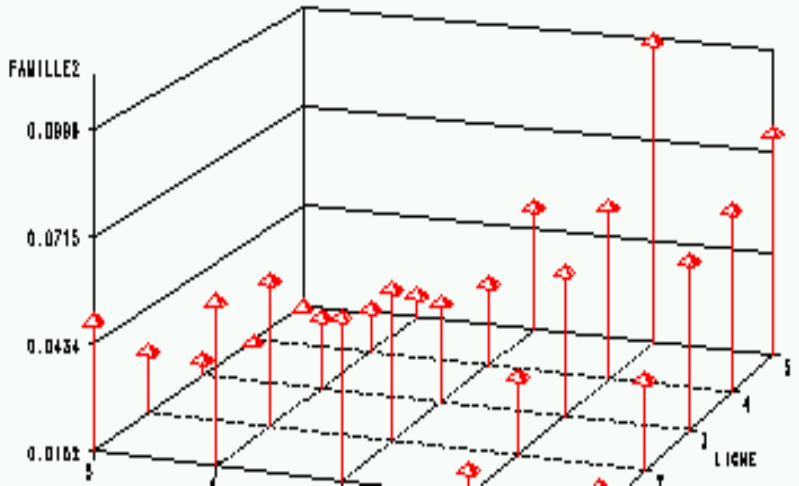


Modalité famille 1 et 2









Analyse du tableau disjonctif complet : modalités et individus (KACM1, KACM2, KDISJ)

- Si l'on souhaite représenter aussi les individus (et pas seulement les modalités), on travaille sur le tableau disjonctif complet

- Classiquement, on fait alors une analyse en composantes principales sur le tableau disjonctif complet, correctement normalisé et en utilisant la distance du χ^2.

Tableau disjonctif complet

	Q1_1	Q1_2	Q1_3	Q2_1	Q2_2	Q3_1	Q3-2	Q3_3	Q3_4
1	0	1	0	0	1	0	0	0	1
2	0	1	0	1	0	0	0	1	0
3	0	0	1	1	0	0	1	0	0
4	1	0	0	0	1	0	0	0	1
5	1	0	0	0	1	0	0	1	0
6	0	1	0	0	1	0	0	1	0
7	0	0	1	1	0	1	0	0	0
8	1	0	0	1	0	1	0	0	0
9	0	1	0	1	0	0	1	0	0
10	0	1	0	0	1	0	0	1	0
11	0	0	1	0	1	0	1	0	0
12	1	0	0	1	0	0	0	0	1

Analyse du tableau disjonctif complet : modalités et individus (KACM1)

- La méthode KACM1 consiste alors à pratiquer un algorithme de Kohonen sur ce tableau, avec la même normalisation et la distance du χ^2. Un individu *i* :

$$\frac{d_{ij}}{\sqrt{d_{i.}}\sqrt{d_{.j}}}, \quad d_{i.} = Q$$

- On classe ainsi les individus, puis les modalités normalisées pour représenter des individus types . La modalité *j* est :

$$\frac{n_{jl}}{d_{.j}\sqrt{d_{.l}}\sqrt{Q}}$$

- La représentation graphique est malaisée (trop grand nombre de points), mais la classification obtenue est très utile.

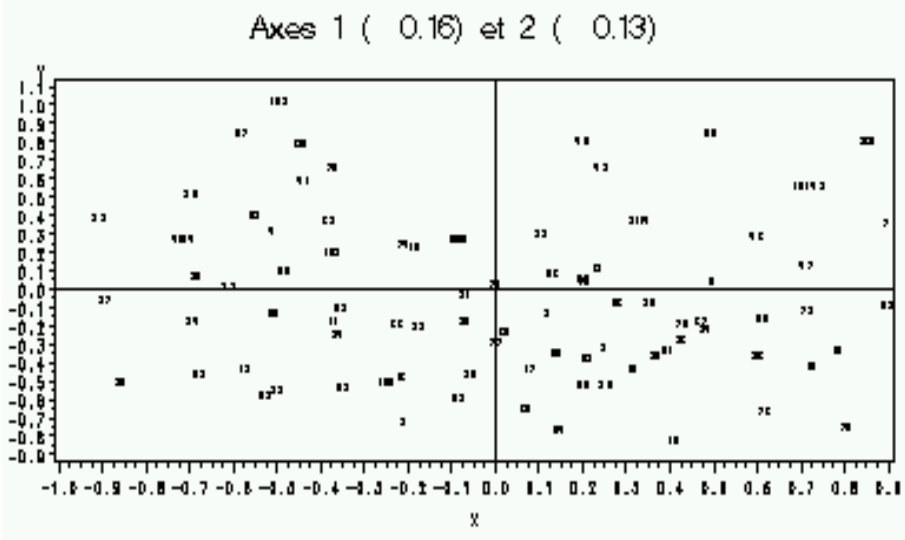
Analyse du tableau disjonctif complet : modalités et individus (KACM2)

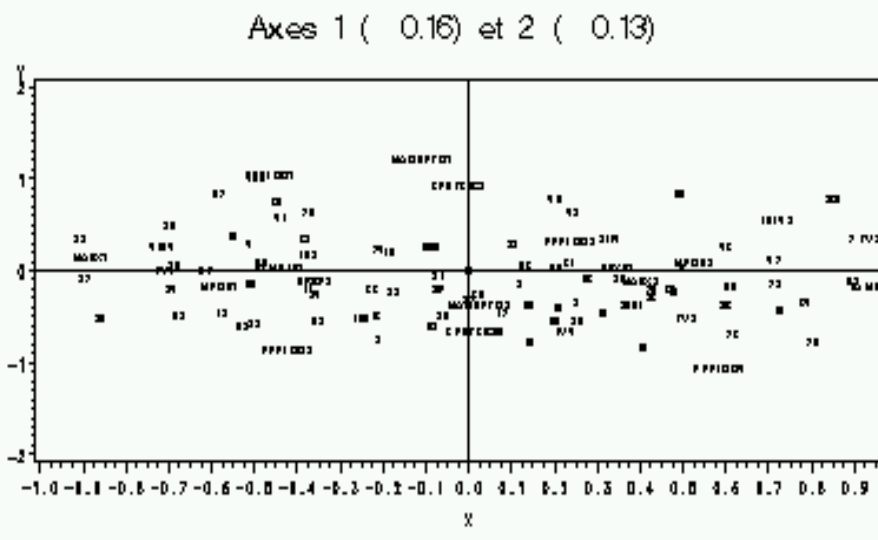
- La méthode KACM2 consiste alors à pratiquer un algorithme de Kohonen sur la table de Burt, corrigée par la normalisation usuelle et la distance du χ^2.
- On classe ainsi les modalités (comme avec KACM), puis les individus correctement normalisées pour être comparables aux vecteurs qui représentent les modalités.
- Une modalité *i* : un individu :

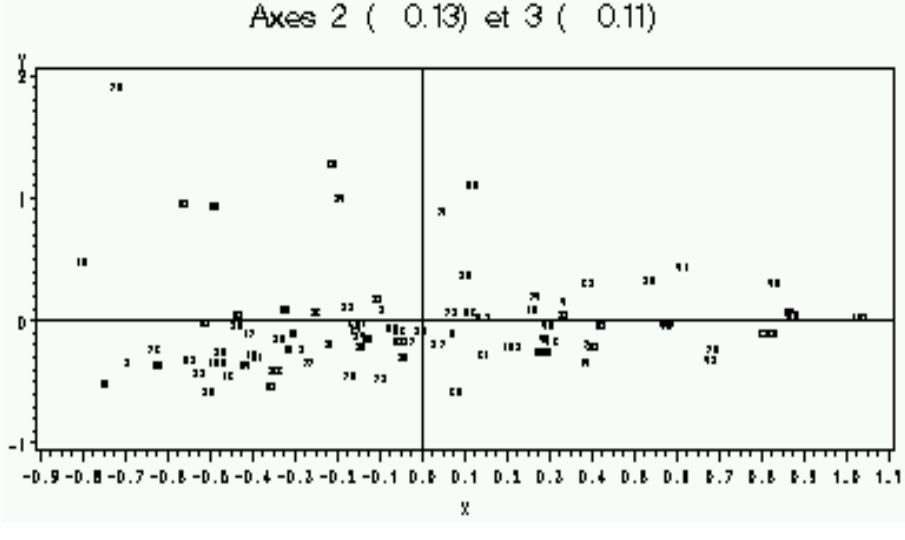
$$\frac{n_{ij}}{\sqrt{n_{i.}}\sqrt{n_{.j}}} \qquad \frac{d_{ij}}{Q}$$

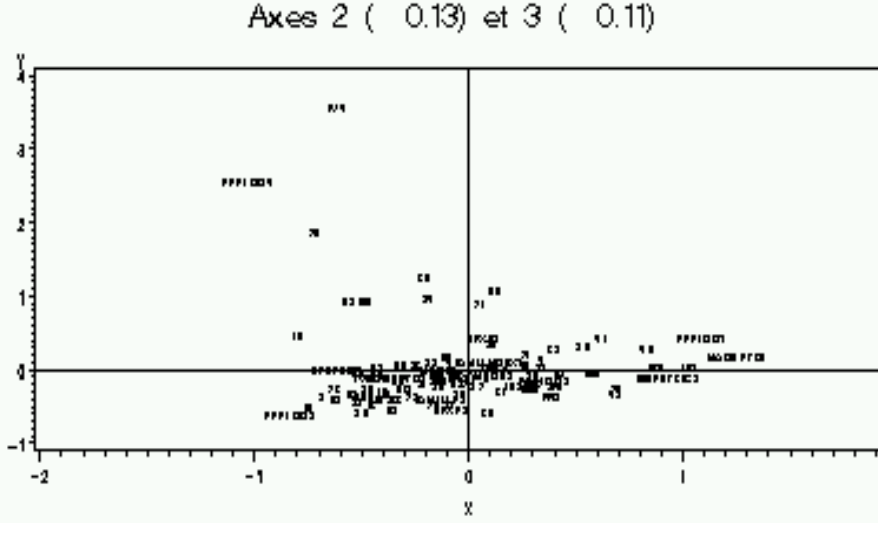
- Avec KACM2, l'apprentissage est rapide puisqu'il ne porte que sur les modalités, mais il faut prolonger le nombre d'itérations pour calculer avec précision les vecteurs codes qui servent à classer ensuite les individus.

ACM : modalités et individus

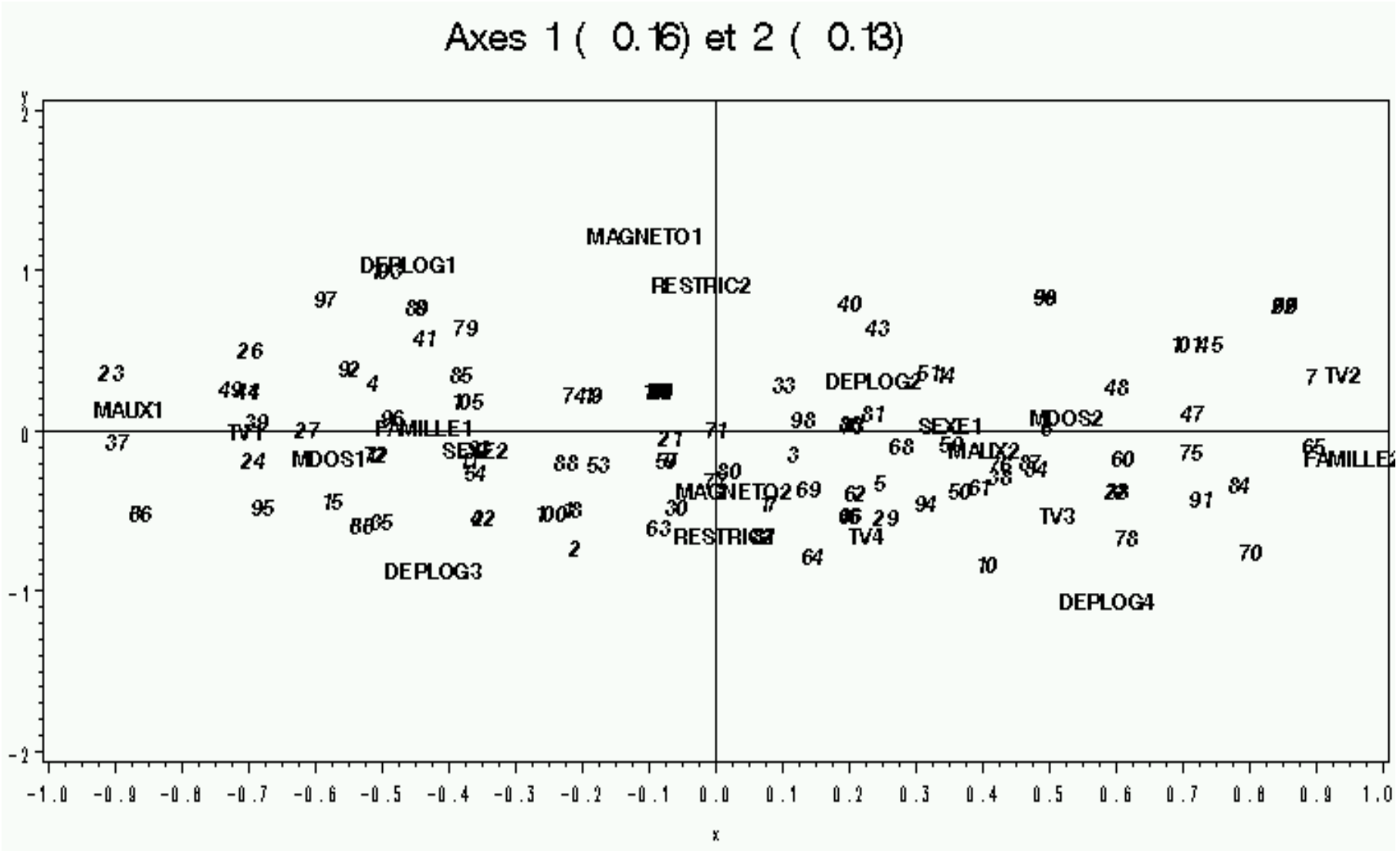




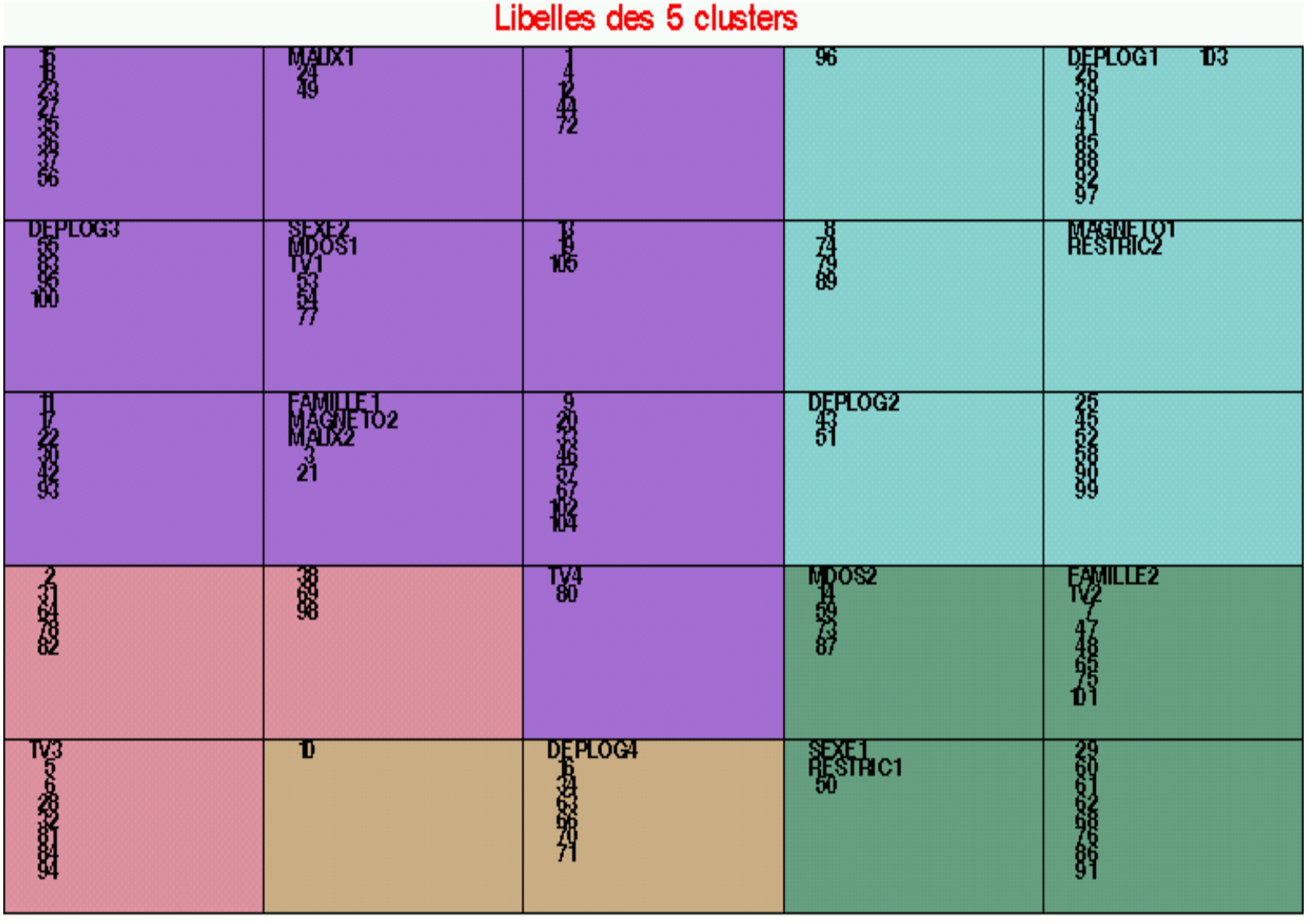




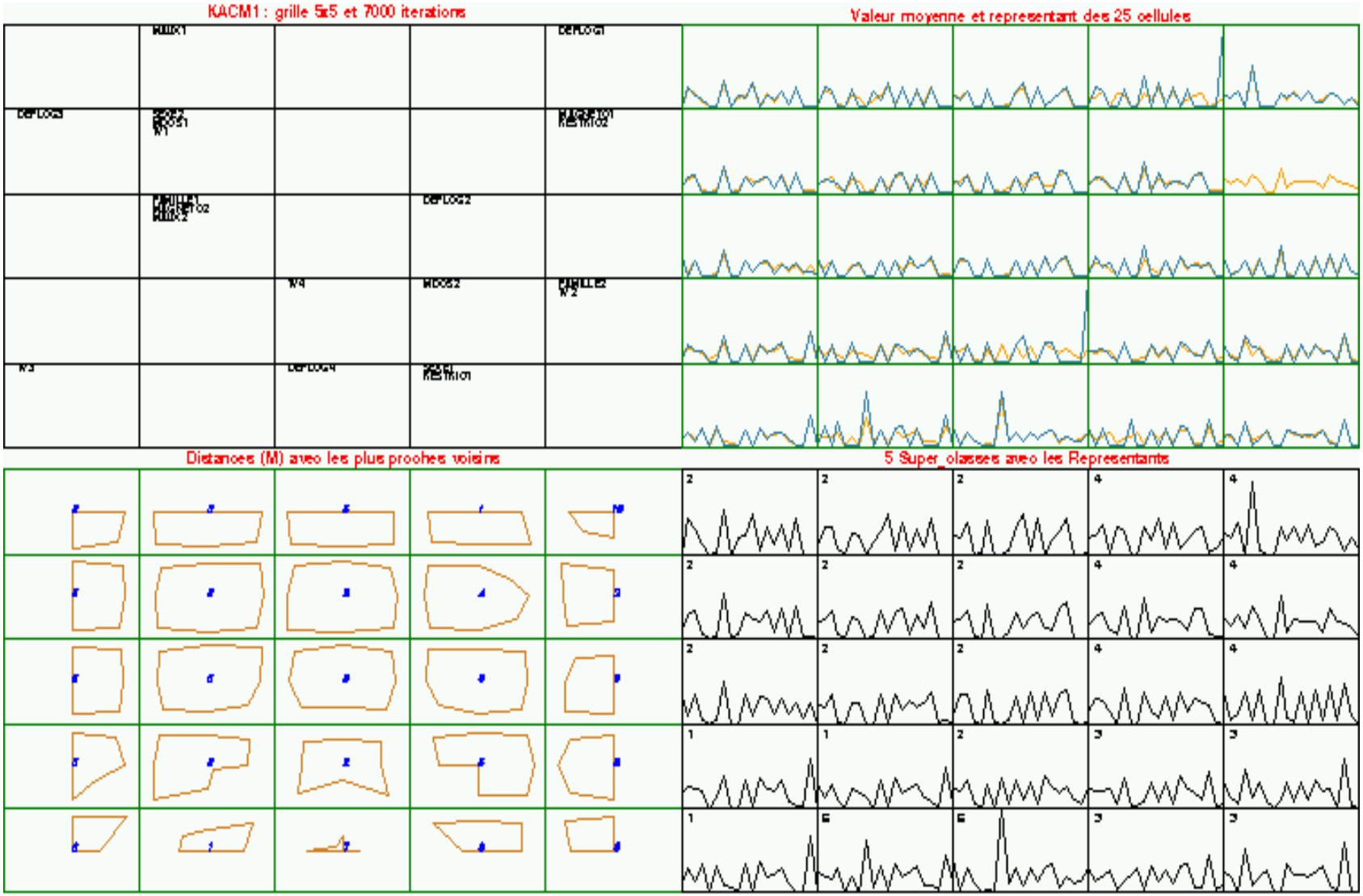
ACM (Modalités et individus)



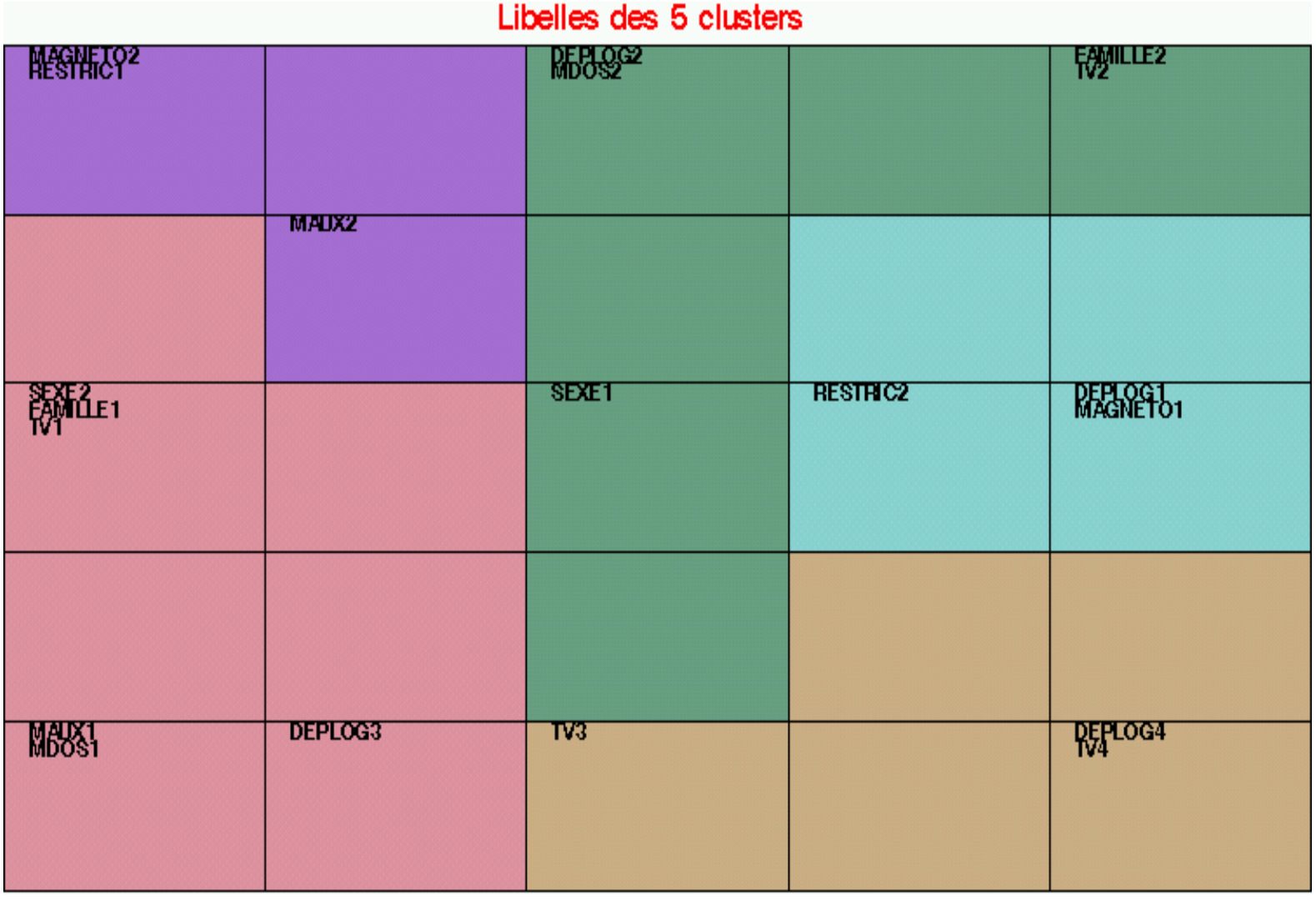
KACM1


Libelles des 5 clusters
MAUX1
DEPLOG1
DEPLOG3
SEXE2
MDOS1
TV1
MAGNETO1
RESTRIC2
FAMILLE1
MAGNETO2
MAUX2
DEPLOG2
TV4
MDOS2
FAMILLE2
TV2
TV3
DEPLOG4
SEXE1
RESTRIC1

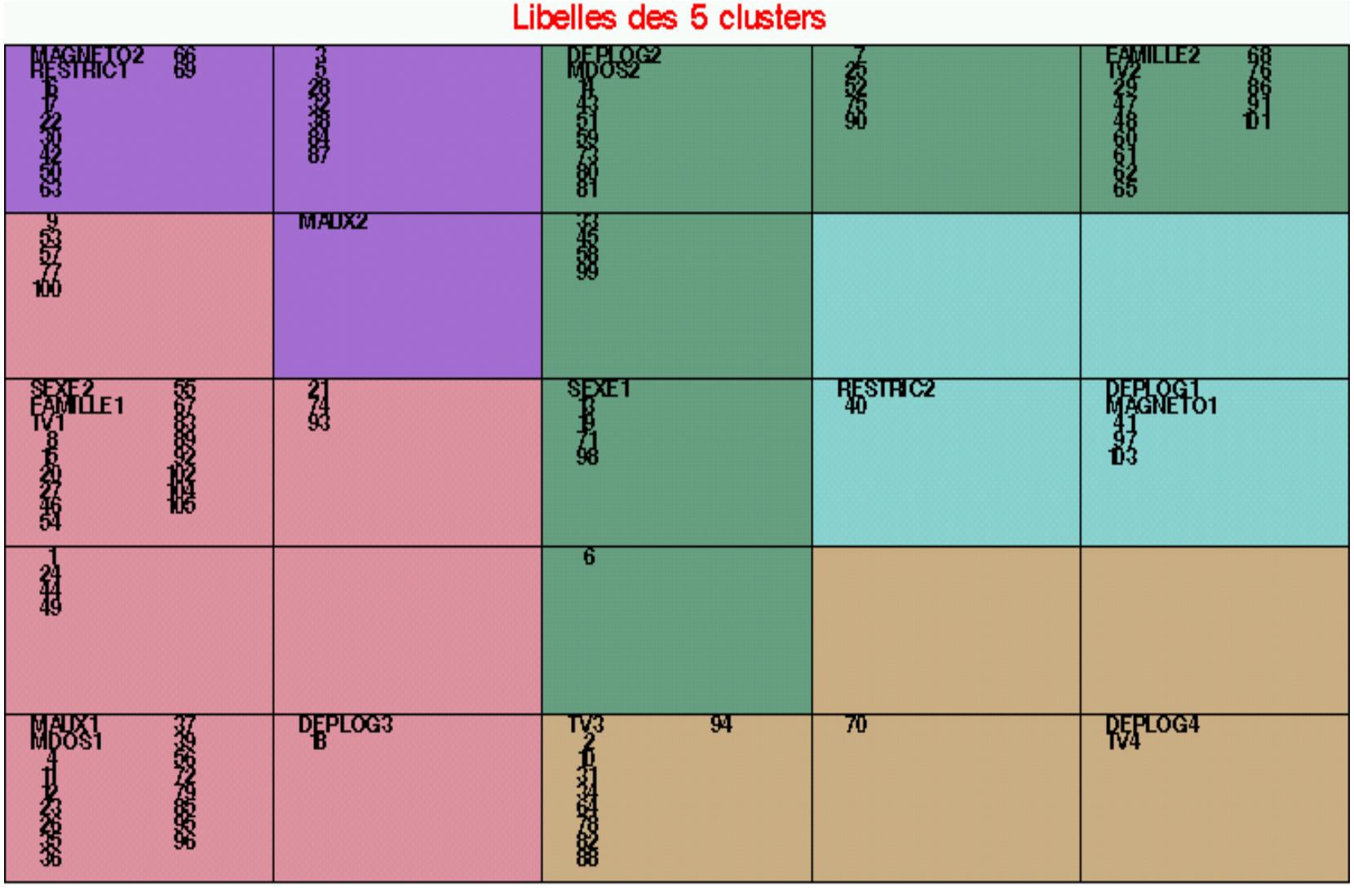
KACM1


KACM1 : grille 5x5 et 7000 iterations
Valeur moyenne et representant des 25 cellules
Distances (M) avec les plus proches voisins
5 Super_classes avec les Representants

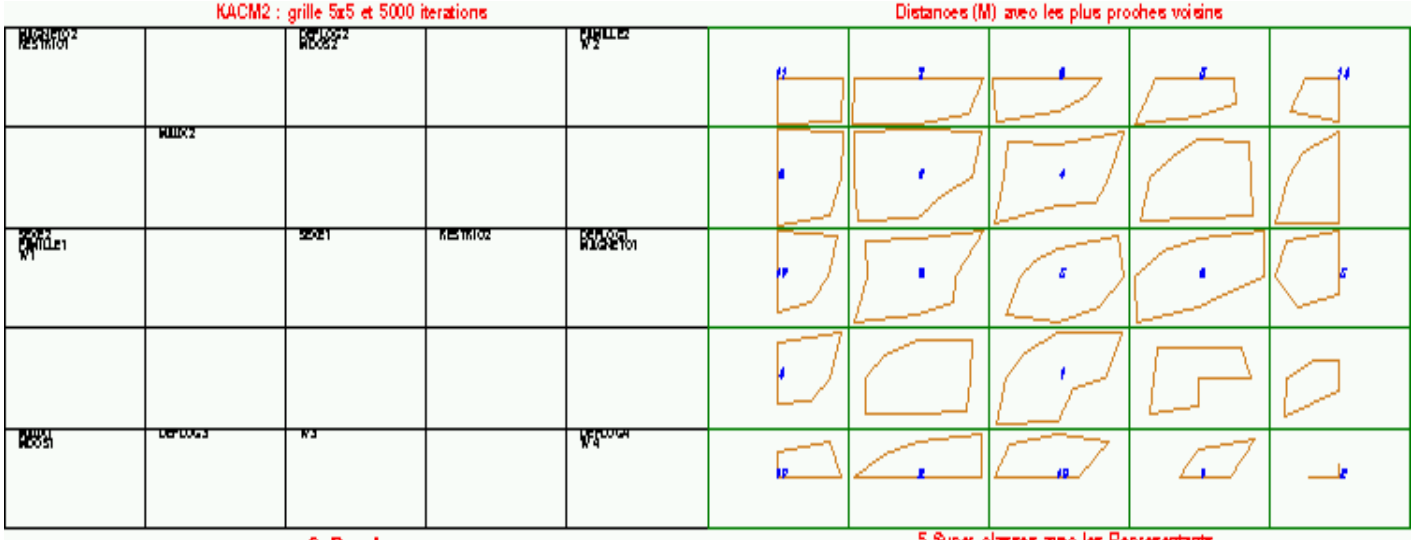
KACM (même nombre d'itérations)

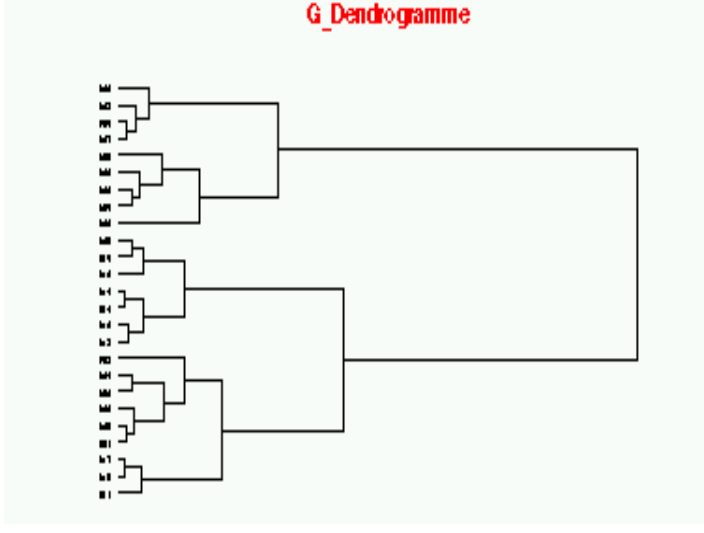


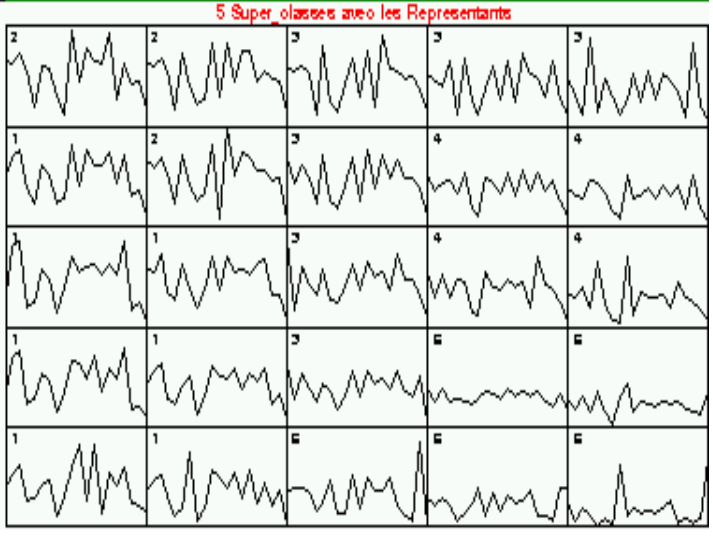
KACM2


Libelles des 5 clusters
MAGNETO2
RESTRIC1
MAUX2
DEPLOG2
MDOS2
FAMILLE2
TV2
SEXE2
FAMILLE1
TV1
SEXE1
RESTRIC2
DEPLOG1
MAGNETO1
MAUX1
MDOS1
DEPLOG3
TV3
DEPLOG4
TV4

KACM2


KACM2 : grille 5x5 et 5000 iterations
Distances (M) avec les plus proches voisins


G_Dendrogramme


5 Super_classes avec les Representants

Valeurs tests pour KACM2

SEXE	C1	C2	C3	C4	C5	Total
1	52%	50%	78%	25%	50%	50%
2	48%	50%	22%	75%	50%	50%
FAMILLE	**C1**	**C2**	**C3**	**C4**	**C5**	**Total**
1	65%	50%	94%	85%	75%	69%
2	35%	50%	6%	15%	25%	31%
DEPLOG	**C1**	**C2**	**C3**	**C4**	**C5**	**Total**
1	4%	3%	33%	10%	25%	10%
2	48%	75%	28%	45%	50%	54%
3	48%	20%	11%	45%	0%	29%
4	0%	3%	28%	0%	25%	7%
MAGNETO	**C1**	**C2**	**C3**	**C4**	**C5**	**Total**
1	0%	25%	44%	10%	50%	21%
2	**100%**	75%	56%	90%	50%	79%
MAUX	**C1**	**C2**	**C3**	**C4**	**C5**	**Total**
1	30%	25%	44%	25%	75%	31%
2	70%	75%	56%	75%	25%	69%
MDOS	**C1**	**C2**	**C3**	**C4**	**C5**	**Total**
1	57%	45%	50%	40%	50%	48%
2	43%	55%	50%	60%	50%	52%
RESTRIC	**C1**	**C2**	**C3**	**C4**	**C5**	**Total**
1	52%	60%	44%	85%	75%	61%
2	48%	40%	56%	15%	25%	39%
TV	**C1**	**C2**	**C3**	**C4**	**C5**	**Total**
1	48%	45%	61%	65%	0%	50%
2	52%	25%	22%	0%	25%	26%
3	0%	30%	17%	35%	0%	21%
4	0%	0%	0%	0%	75%	3%

SEXE	C1	C2	C3	C4	C5
1	1.03	0.99	**1.54**	0.50	0.99
2	0.97	1.01	0.45	**1.51**	1.01
FAMILLE	**C1**	**C2**	**C3**	**C4**	**C5**
1	0.95	0.73	**1.38**	**1.24**	1.09
2	1.11	**1.59**	0.18	0.48	0.80
DEPLOG	**C1**	**C2**	**C3**	**C4**	**C5**
1	0.41	0.24	**3.18**	0.95	2.39
2	0.88	**1.38**	0.51	0.83	0.92
3	**1.67**	0.70	0.39	**1.58**	0.00
4	0.00	0.37	**4.16**	0.00	**3.75**
MAGNETO	**C1**	**C2**	**C3**	**C4**	**C5**
1	0.00	**1.19**	**2.12**	0.48	**2.39**
2	**1.27**	0.95	0.70	**1.14**	0.63
MAUX	**C1**	**C2**	**C3**	**C4**	**C5**
1	0.97	**0.80**	**1.41**	0.80	**2.39**
2	1.01	**1.09**	0.81	**1.09**	0.36
MDOS	**C1**	**C2**	**C3**	**C4**	**C5**
1	**1.19**	0.94	1.05	0.84	1.05
2	0.83	1.05	0.95	**1.15**	0.95
RESTRIC	**C1**	**C2**	**C3**	**C4**	**C5**
1	0.86	**0.98**	0.73	**1.39**	1.23
2	**1.22**	1.02	**1.42**	0.38	0.64
TV	**C1**	**C2**	**C3**	**C4**	**C5**
1	0.95	0.89	1.21	**1.29**	0.00
2	**2.03**	0.97	0.86	0.00	0.97
3	0.00	1.43	0.80	**1.67**	0.00
4	0.00	0.00	0.00	0.00	**26.22**

L’algorithme KDISJ

- On note d_{ij} le terme général de ce tableau, tableau de contingence croisant la variable "individu" à *N* modalités et la variable "modalités" à *M* modalités. Le terme d_{ij} prend ses valeurs dans {0,1}.
- Adaptation d’un algorithme (KORRESP) introduit pour l'analyse des tableaux de contingence croisant deux variables qualitatives. Cet algorithme est une méthode très rapide et efficace d’analyse des relations entre deux variables qualitatives.
- On calcule les sommes en ligne et les sommes en colonne :
- Pour un tableau disjonctif complet, $d_{i.}$ vaut *K*, quelque soit *i*. Le terme $d_{.j}$ est l’effectif de la modalité *j.*

Tableau corrigé

- On utilise la distance du χ^2 sur les lignes autant que sur les colonnes, et on pondère les modalités : on corrige le tableau disjonctif complet, et on pose

$$d_{ij}^{c} = \frac{d_{ij}}{\sqrt{d_{i.} d_{.j}}}$$

- Le tableau ainsi corrigé est noté D^c (tableau disjonctif corrigé). Cette transformation est la même que celle qui est proposée par Smaïl Ibbou dans sa thèse, (Ibbou, 1998)
- Mêmes corrections que celles qu'on fait traditionnellement lorsqu'on pratique une analyse des Correspondances. Il s'agit en fait d'une analyse en composantes principales pondérée, utilisant la distance du Chi-deux, simultanée sur les profils lignes et les profils colonnes. Cela est équivalent à une analyse en composantes principales sur les données corrigées de cette manière.

KDISJ (suite)

- On choisit ensuite un réseau de Kohonen, et on associe à chaque unité un vecteur code formé de (M + N) composantes, les M premières évoluent dans l'espace des individus (représentés par les lignes de D^c), les N dernières dans l'espace des modalités (représentées par les colonnes de D^c).
- Les étapes de l'apprentissage du réseau de Kohonen sont doubles. On tire alternativement une ligne de D^c (c'est-à-dire un individu), puis une colonne (c'est-à-dire une modalité).
- Quand on tire un individu i, on lui associe la modalité $j(i)$ qui maximise le coefficient. On forme alors un vecteur individu complété de dimension (M + N). On cherche alors parmi les vecteurs-codes celui qui est le plus proche, au sens de la distance euclidienne restreinte aux M premières composantes. Notons u l'unité gagnante. On rapproche alors les vecteurs-codes de l'unité u et de ses voisines du vecteur complété (i, $j(i)$), selon la loi usuelle de Kohonen.

KDISJ (suite)

- Quand on tire une modalité *j*, de dimension *N*, on ne lui associe pas de vecteur, en effet, par construction, il y a beaucoup d'ex-æquo et le choix serait arbitraire. On cherche alors parmi les vecteurs codes celui qui est le plus proche, au sens de la distance euclidienne restreinte aux *N* dernières composantes. On rapproche alors les *N* dernières composantes du vecteur-code gagnant et de ses voisins de celles du vecteur modalité *j*, sans modifier les *M* premières composantes.

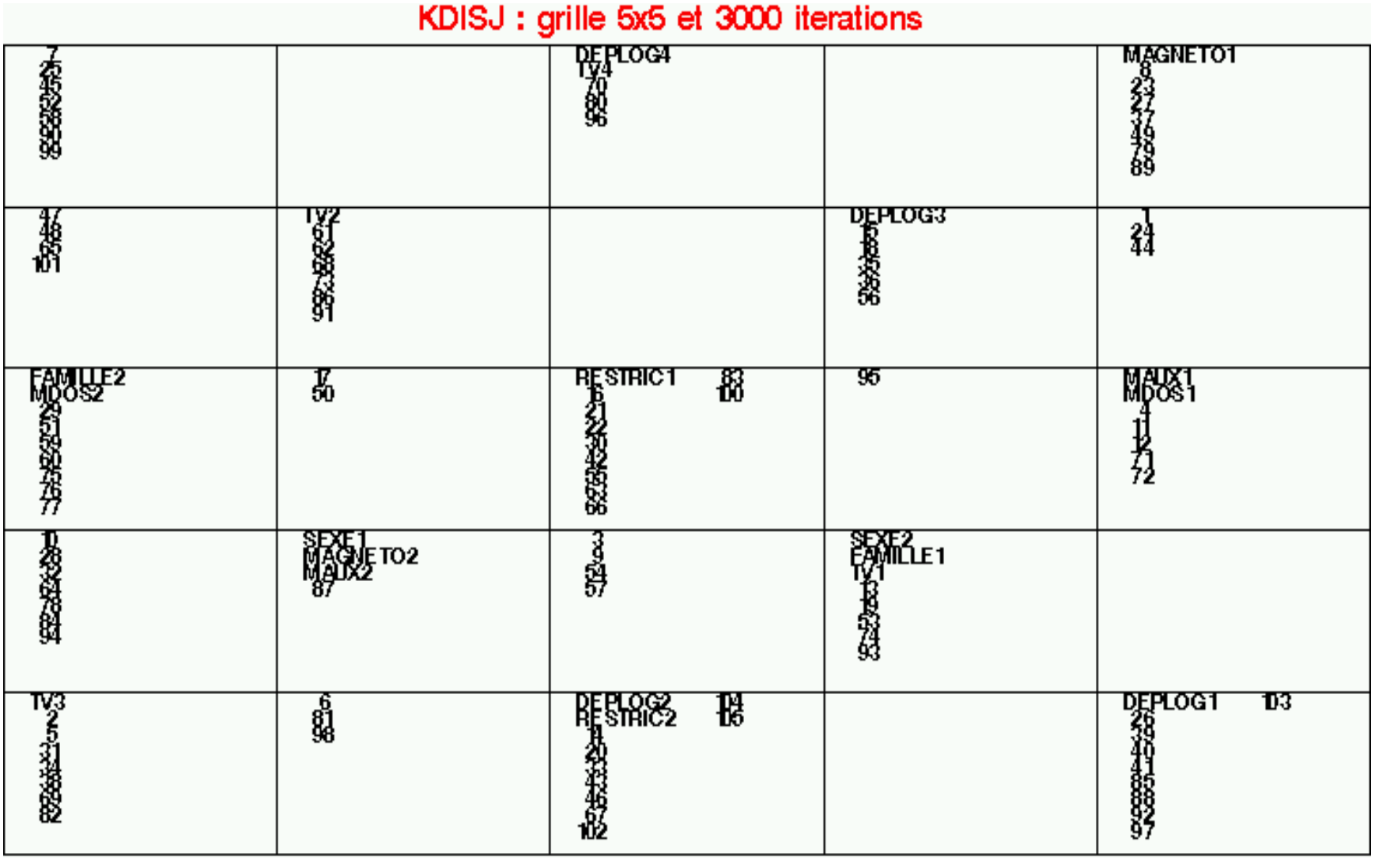
- On pratique ainsi un classement classique de Kohonen sur les individus, un autre sur les modalités, tout en les maintenant associés. Après convergence, les individus et les modalités sont classés dans les classes de Kohonen. Des individus ou modalités “ proches ” sont classés dans la même classe ou dans des classes voisines. On appelle KDISJ l’algorithme ainsi défini.

KDISJ

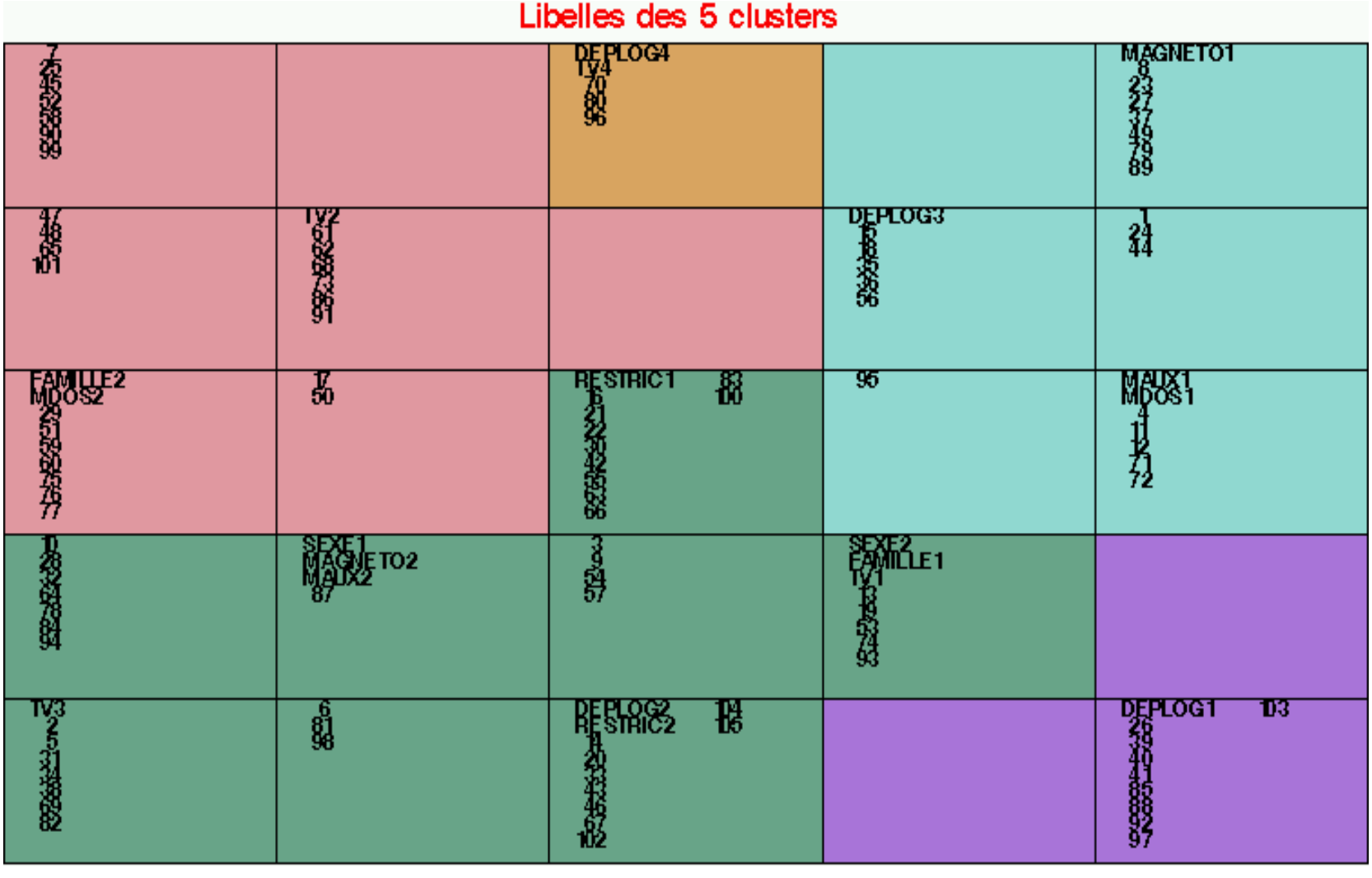
KDISJ : grille 5x5 et 3000 iterations

		DEPLOG4 TV4		MAGNETO1
	TV2		DEPLOG3	
FAMILLE2 MDOS2		RESTRIC1		MAUX1 MDOS1
	SEXE1 MAGNETO2 MAUX2		SEXE2 FAMILLE1 TV1	
TV3		DEPLOG2 RESTRIC2		DEPLOG1

KDISJ


KDISJ : grille 5x5 et 3000 iterations
DEPLOG4
TV4
MAGNETO1
TV2
DEPLOG3
FAMILLE2
MDOS2
RESTRIC1
MAUX1
MDOS1
SEXE1
MAGNETO2
MAUX2
SEXE2
FAMILLE1
TV1
TV3
DEPLOG2
RESTRIC2
DEPLOG1

KDISJ


Libelles des 5 clusters
DEPLOG4
TV4
MAGNETO1
TV2
DEPLOG3
FAMILLE2
MDOS2
RESTRIC1
MAUX1
MDOS1
SEXE1
MAGNETO2
MAUX2
SEXE2
FAMILLE1
TV1
TV3
DEPLOG2
RESTRIC2
DEPLOG1

Autre exemple : ANPE

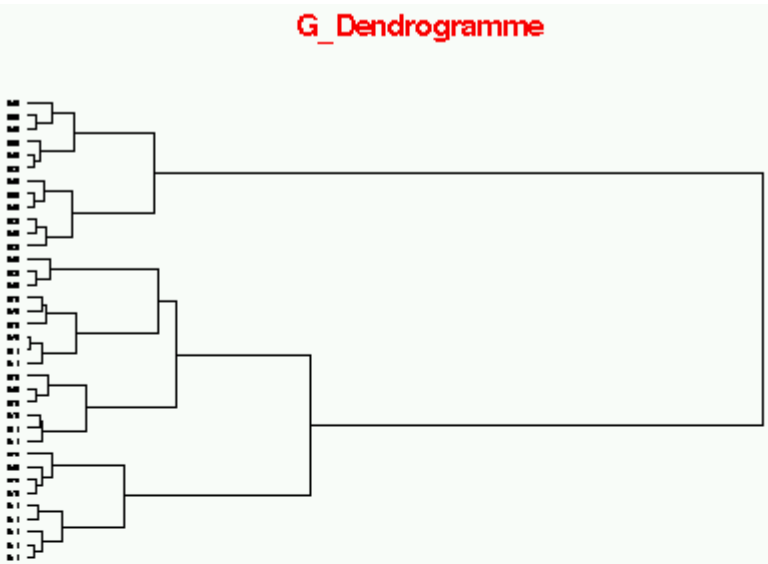
- Demandeurs d'emplois inscrits à l'ANPE entre le 1er juillet 1993 et le 31 août 1996, et pour lesquels on a observé au moins deux périodes de chômages, (*Patrice Gaubert et Marie Cottrell, journées ACSEG,Louvain, 1998*)
- 204 personnes, extrait de l'enquête complète
- 8 variables qualitatives, 32 modalités
  - AGEC, classe d'âge, <25, 25-35, 35-45, 45-55, >55
  - CTINDMO, indemnisation journalière, <60F, 60-100, 100-150, >150
  - DIPL3, niveau de formation, >bac, niveau bac, <bac
  - DURC, durée totale du chômage, <12 mois, 12-24, >24
  - HAR, horaire mensuel en activité réduite, 0, 0-39, 39-78, 78-117, >117
  - PPARC, % d'AR dans la durée totale de chômage, 0, 0-0.1, 0.1-0.3, >0.3
  - RMOTIFA, type de sortie de chômage, (4 motifs)
  - RMOTIFI, type d'inscription au chômage (4 motifs)

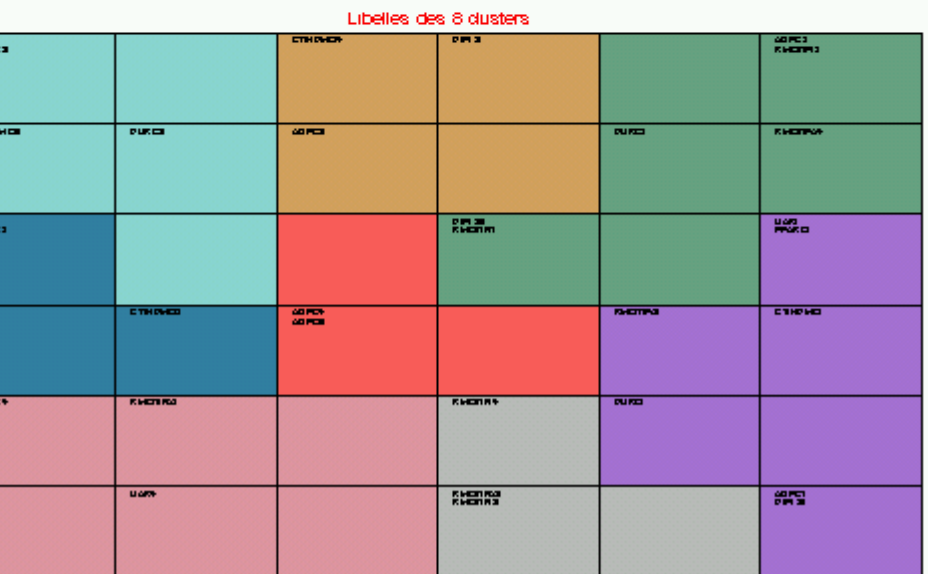
Les modalités

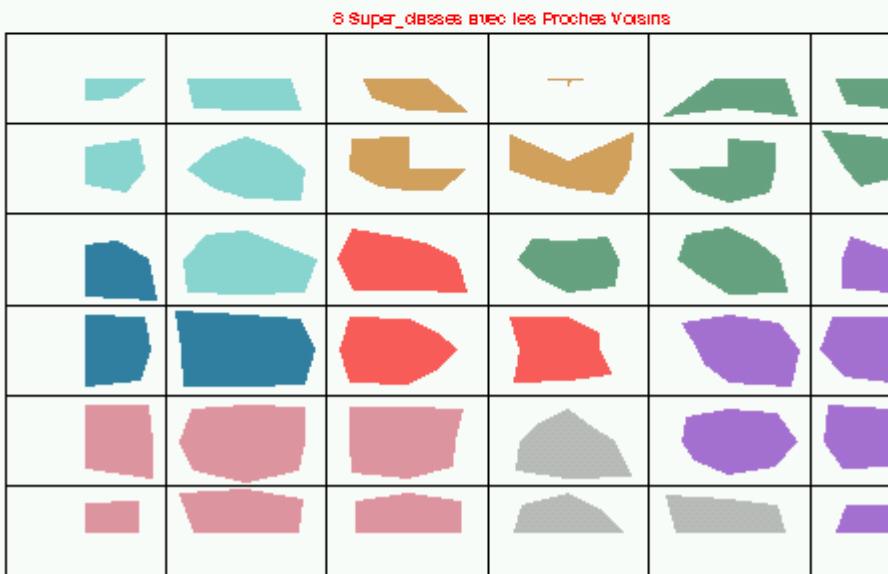
- *Motifs de sortie du chômage :*
- -**1** : sortie vers l'emploi (que ce soit par l'ANPE ou par ses propres moyens)
- -**2** : sortie vers un stage ou un CES
- **-3 :** retrait par maladie, retraite, service national, etc.
- -**4** : radiation, sanction, découragement, décès, etc.

- *Motifs d'inscription* regroupés en 4 modalités :
- - **1** : licenciement économique, fin d'intérim, autres
- - **2** : autre licenciement, fin de contrat à durée déterminée
- - **3** : démission , fin de conversion
- - **4** : recherche du premier emploi

KACM : les modalités





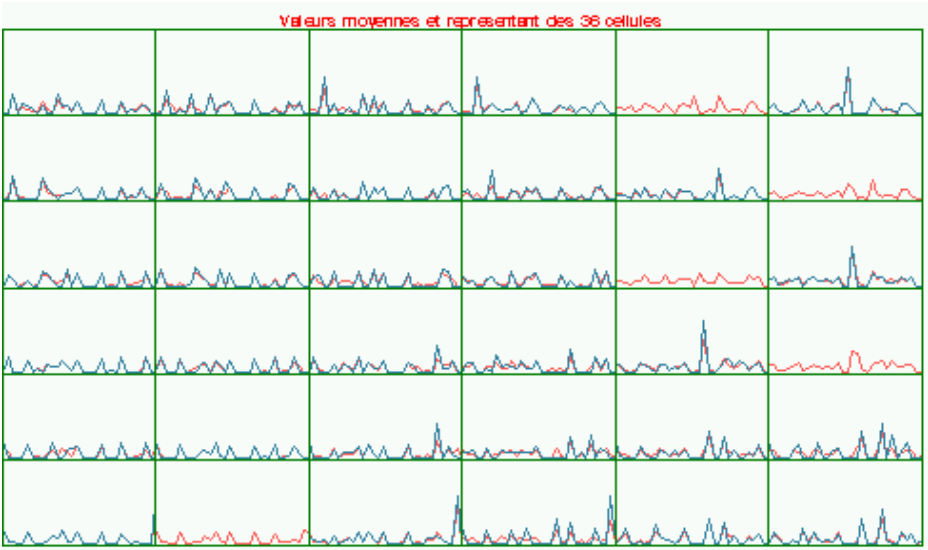


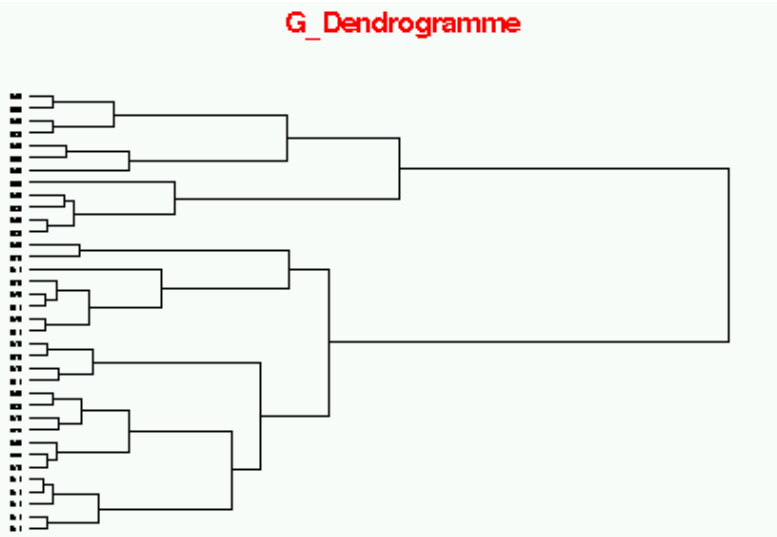
Répartitions des modalités

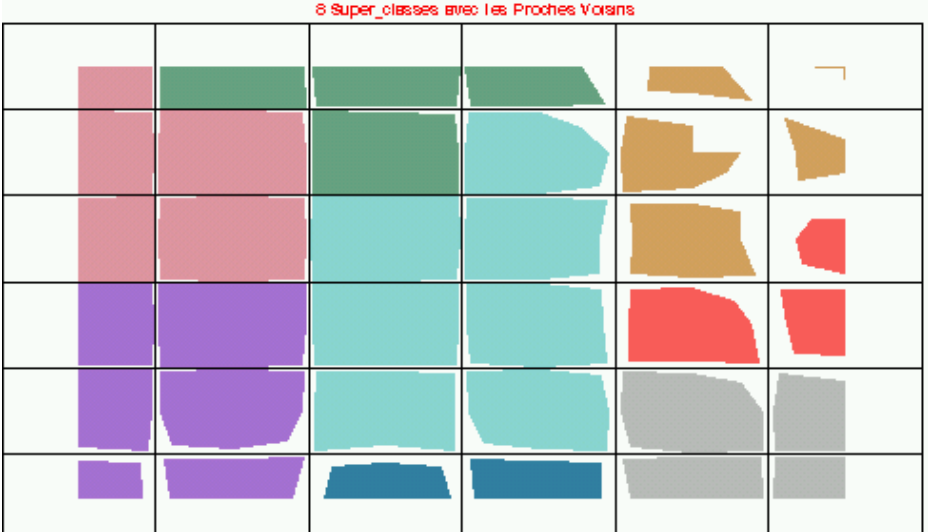
KACM : grille 6 par 6, et 300 itérations

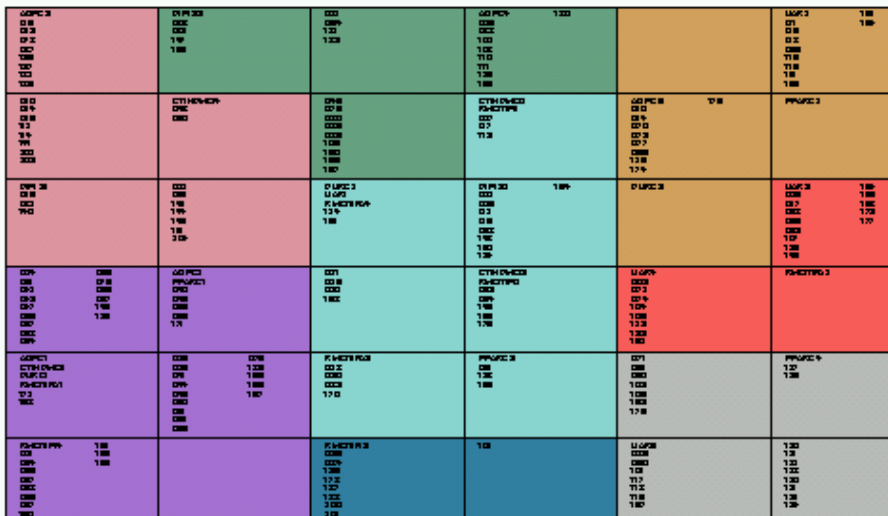
AR117+ Prop0.1-0.3		Ind150+	+bac		25-35a finCDD
Ind100-150	>2ans	35-45a		1-2ans	Radiation
AR0-39 PropAR0-0.1			-bac licéco		AR0 PropAR0
AR39-78	Ind60-100	45-55a >55a		Emploi	Ind-60
PropAR>0.3	Stage		1eremp	<1an	
	AR78-117		Retrait de l'ANPE Démission de l'emploi		<25a nivbac

KACM1 : individus + modalités

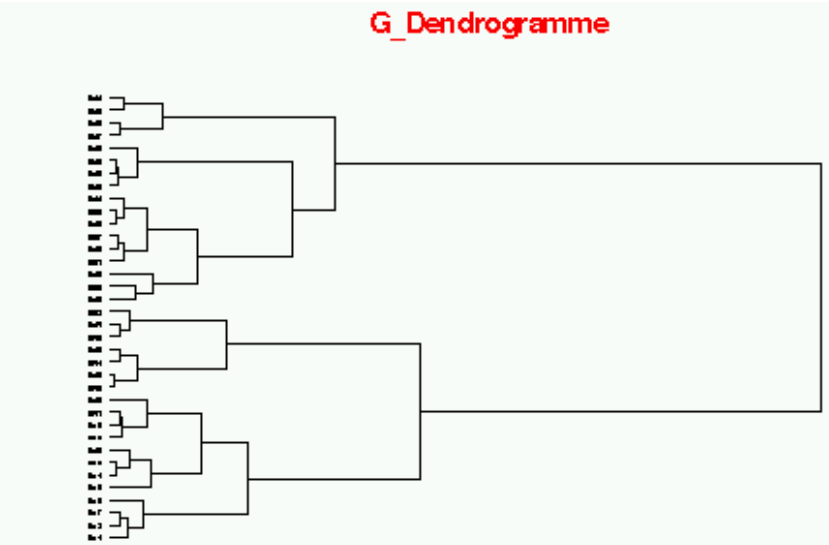


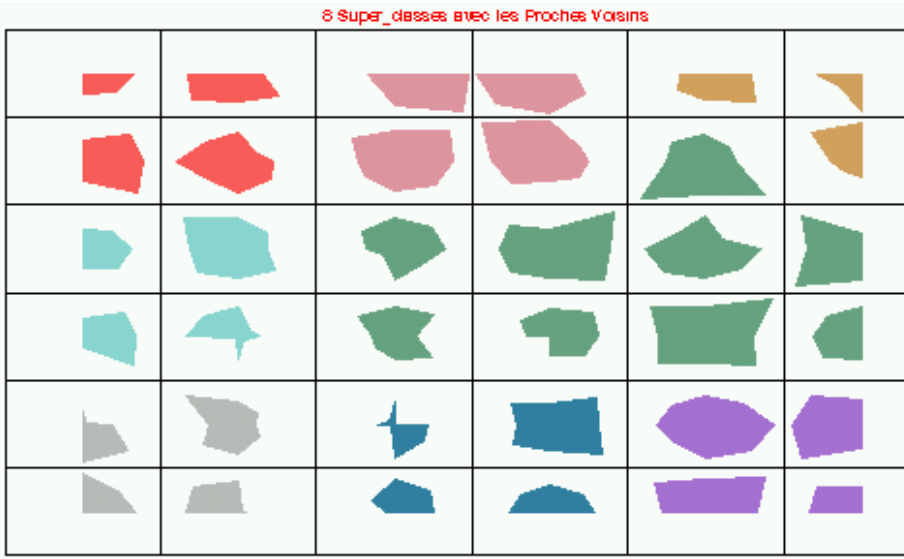


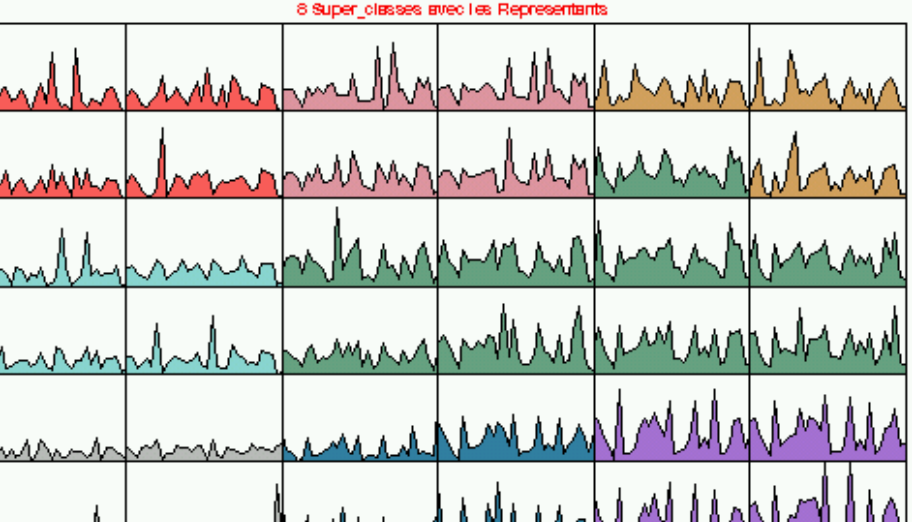


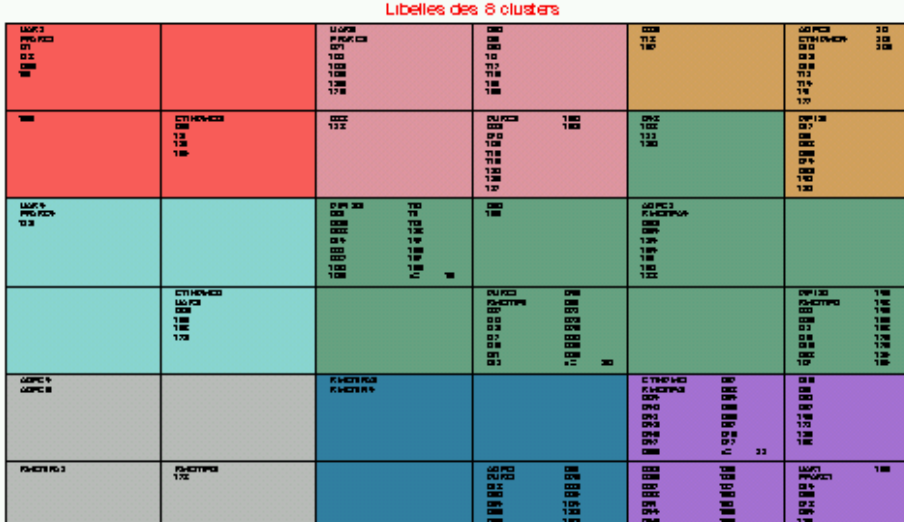


KACM2 : modalités + individus









Analyse de données : introduction

Algorithme de Kohonen

Kohonen et classification : KACP

Accélération de la classification

Traitements des variables qualitatives

Conclusion

Conclusion

- C'est un très bon outil
  - de classification (accélération des méthodes type centres mobiles)
  - de visualisation en raison de la conservation des voisinages
  - de complément des méthodes factorielles classiques
- On peut combiner méthodes classiques et l'algorithme de Kohonen :
  - **KACP sur les coordonnées obtenues après une ACM**
  - **ACM (ou KACM, ou KDISJ) sur des variables qualitatives en y rajoutant une variable de classe obtenue par un KACP**
- On obtient directement des **scores** si on classe sur une ficelle
- On peut s'en servir en **prévision** en segmentant l'espace et en utilisant un modèle par segment (pré-traitement avant l'usage d'un perceptron ou d'un modèle auto-régressif)
- Outil de **prévision de courbes**, avec la même précision en chaque point de la courbe (au contraire des méthodes usuelles)

Conclusion

- Facilité de travail avec des **données manquantes** (cf thèse de Smaïl Ibbou) : les distances sont calculées sur les composantes présentes dans les observations
- Les données manquantes peuvent être estimées par les composantes correspondantes du vecteur code de la classe de l ’observation
- Application développée par T.Kohonen : aide à la recherche de mots clés dans de grands textes (WEB)

Accélération de la classification

Kohonen et Classification

- Les algorithmes usuels (sans voisinage) minimisent la somme des carrés intra-classes, alors que l'algorithme de Kohonen minimise la variance intra-classes étendue

- Mais en pratique, au cours des itérations, on fait décroître le nombre de voisins jusqu'à 0 voisin. Alors l'algorithme de Kohonen est utilisé comme ***une très bonne initialisation d'un algorithme de classification usuel, qui permet d'atteindre un «bon» minimum de la variance intra-classes***

KACP pour accélérer SCL

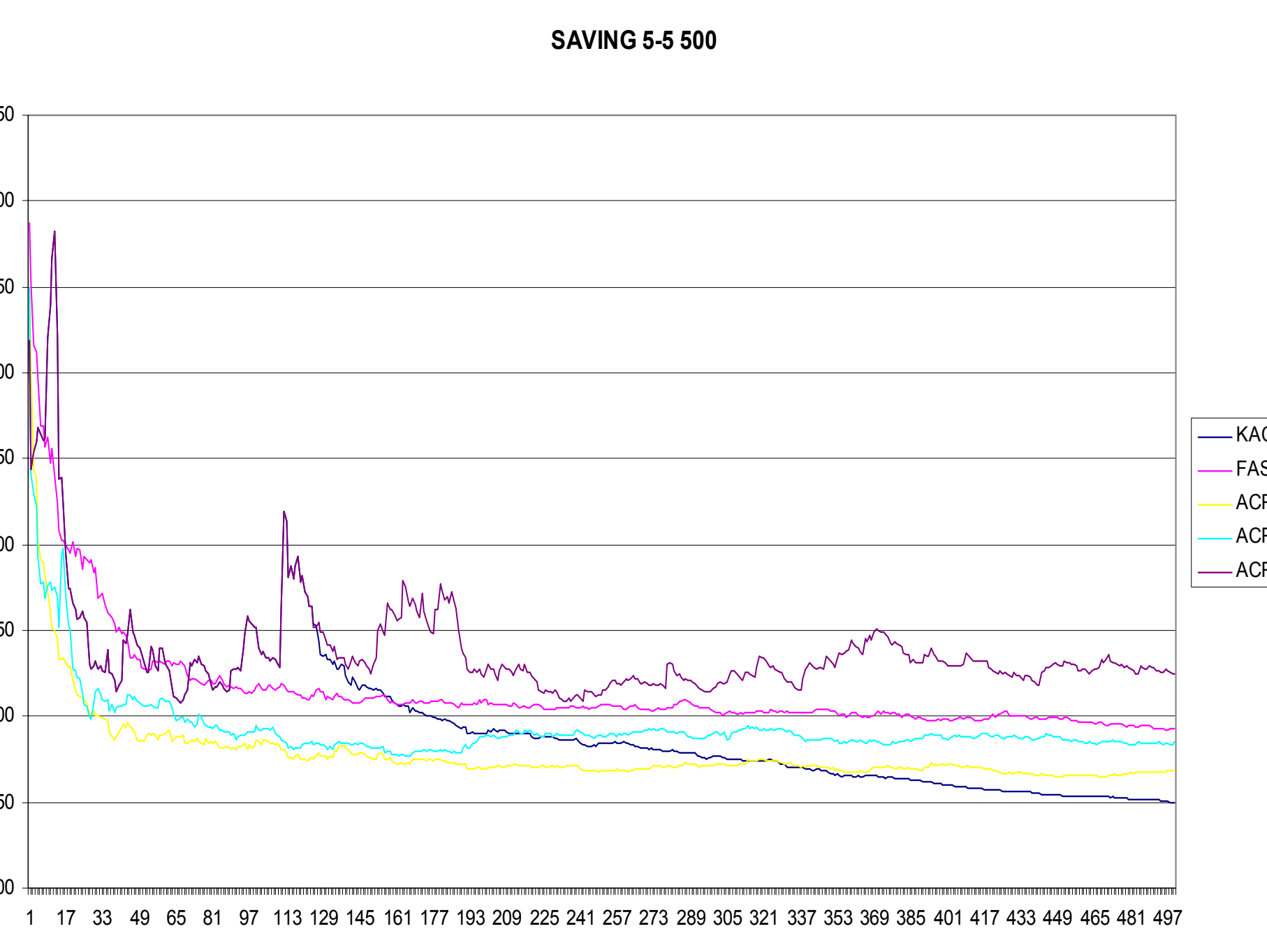
- On constate que la somme des carrés intra qu'on cherche à minimiser décroît plus vite lorsqu 'on utilise un algorithme avec des voisins, que si l'on utilise le SCL (version stochastique de l 'algorithme des centres mobiles)

- On réalise donc la classification de données
  - avec SCL (0 voisin)
  - avec un SOM avec 5 voisins
  - avec un SOM avec 9 voisins
  - avec un SOM avec 25 voisins
  - avec l 'algorithme KACP (nb de voisins décroissant de 25 à 0 voisins, suivant la décroissance usuelle)

Les données : SAVING

- Source : Belsey, Kuh, Welsch : Regression diagnostics, Wiley (1980)
- 42 pays, période 1960-1970
- SR : Taux $_{\text{moyen}}$ d ’épargne par personne dans le pays (1960-1970)
- POP15 : Pourcentage moyen de population de moins de 15 ans
- POP 75 : Pourcentage moyen de population de plus de 75 ans
- DPI : Taux moyen de revenu disponible par personne
- ΔDPI : Taux moyen de croissance de DPI

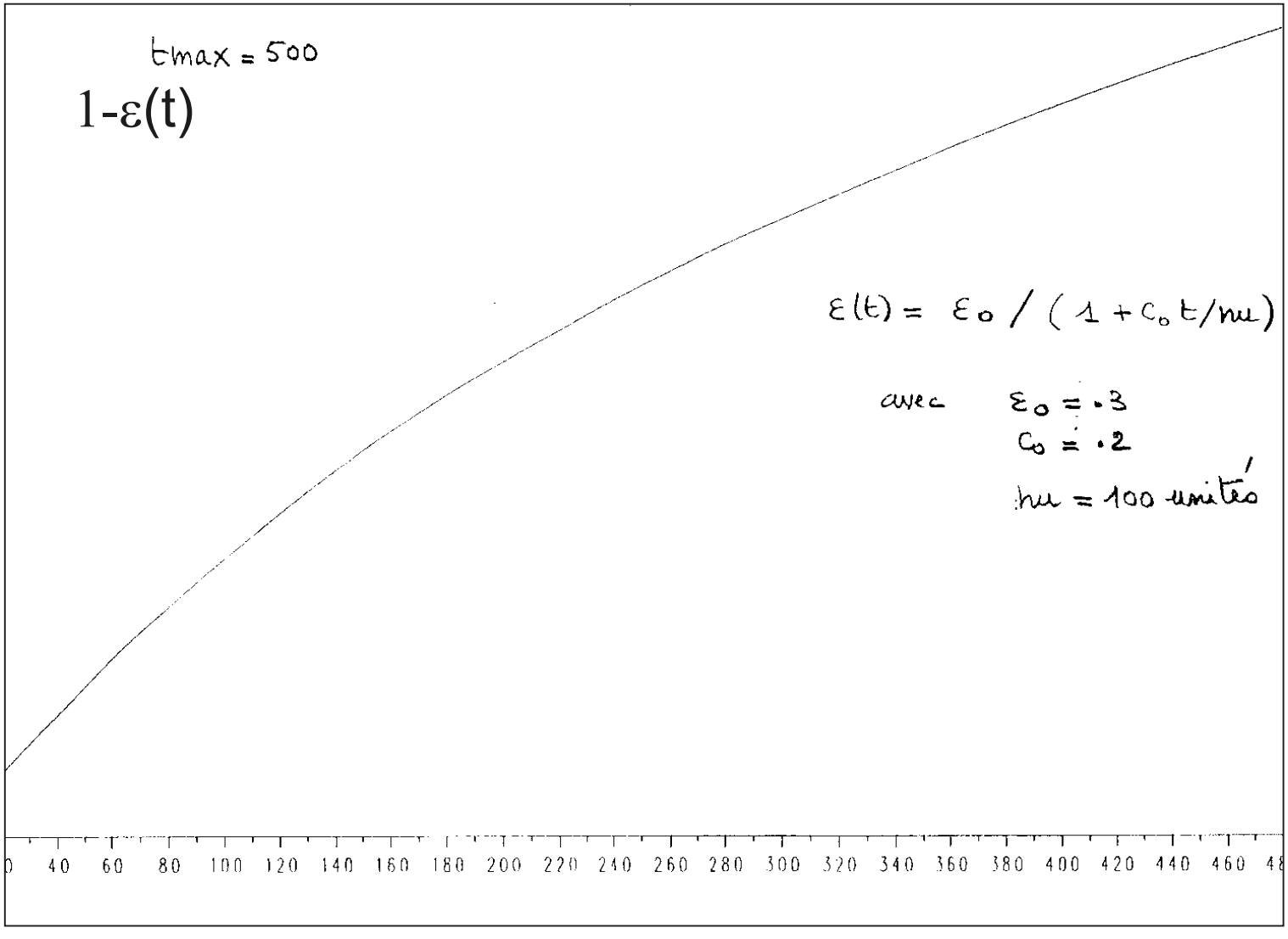
Australia	11.43	29.35	2.87	2329.68	2.87
Austria	12.07	23.32	4.41	1507.99	3.93
Belgium	13.17	23.80	4.43	2108.47	3.82
Bolivia	5.75	41.89	1.67	189.13	0.22
Brazil	12.88	42.19	0.83	728.47	4.56
Canada	8.79	31.72	2.85	2982.88	2.43
Chile	0.60	39.74	1.34	662.86	2.67
China (Taiwan)	11.90	44.75	0.67	289.52	6.51
Colombia	4.98	46.64	1.06	276.65	3.08
Costa Rica	10.78	47.64	1.14	471.24	2.80
Denmark	16.85	24.42	3.93	2496.53	3.99
Ecuador	3.59	46.31	1.19	287.77	2.19
Finland	11.24	27.84	2.37	1681.25	4.32
France	12.64	25.06	4.70	2213.82	4.52
Germany (F.R.)	12.55	23.31	3.35	2457.12	3.44
Greece	10.67	25.62	3.10	870.85	6.28
Guatemala	3.01	46.05	0.87	289.71	1.48
Honduras	7.70	47.32	0.58	232.44	3.19
Iceland	1.27	34.03	3.08	1900.10	1.12
India	9.00	41.31	0.96	88.94	1.54
Ireland	11.34	31.16	4.19	1139.95	2.99
Italy	14.28	24.52	3.48	1390.00	3.54
Japan	21.10	27.01	1.91	1257.28	8.21
Korea	3.98	41.74	0.91	207.68	5.81
Luxembourg	10.35	21.80	3.73	2449.39	1.57
Malta	15.48	32.54	2.47	601.05	8.12
Norway	10.25	25.95	3.67	2231.03	3.62
Netherlands	14.65	24.71	3.25	1740.70	7.66
New Zealand	10.67	32.61	3.17	1487.52	1.76
Nicaragua	7.30	45.04	1.21	325.54	2.48
Panama	4.44	43.56	1.20	568.56	3.61
Paraguay	2.02	41.18	1.05	220.56	1.03
Peru	12.70	44.19	1.28	400.06	0.67
Philippines	12.78	46.26	1.12	152.01	2.00
Portugal	12.49	28.96	2.85	579.51	7.48
South Africa	11.14	31.94	2.28	651.11	2.19
South Rhodesia	13.30	31.92	1.52	250.96	2.00
Spain	11.77	27.74	2.87	768.79	4.35
Sweden	6.86	21.44	4.54	3299.49	3.01
Switzerland	14.13	23.49	3.73	2630.96	2.70
Turkey	5.13	43.42	1.08	389.66	2.96


SAVING 5-5 500
1 17 33 49 65 81 97 113 129 145 161 177 193 209 225 241 257 273 289 305 321 337 353 369 385 401 417 433 449 465 481 497
KAC
FAS
ACF
ACF
ACF

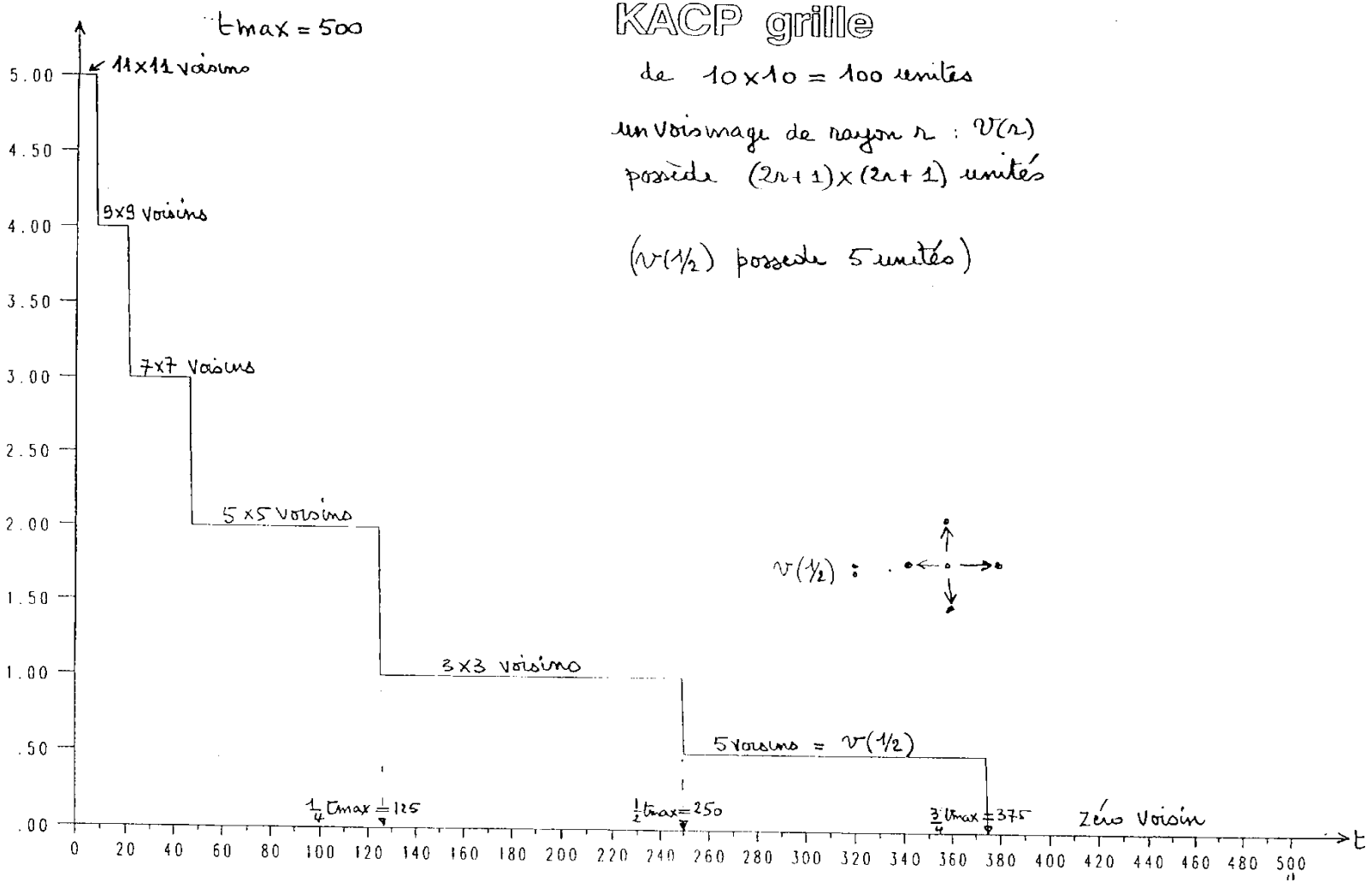
Compléments

- Divers

Fonction epsilon (Letremy)



Fonction voisinage (Letremy)



KACP grille

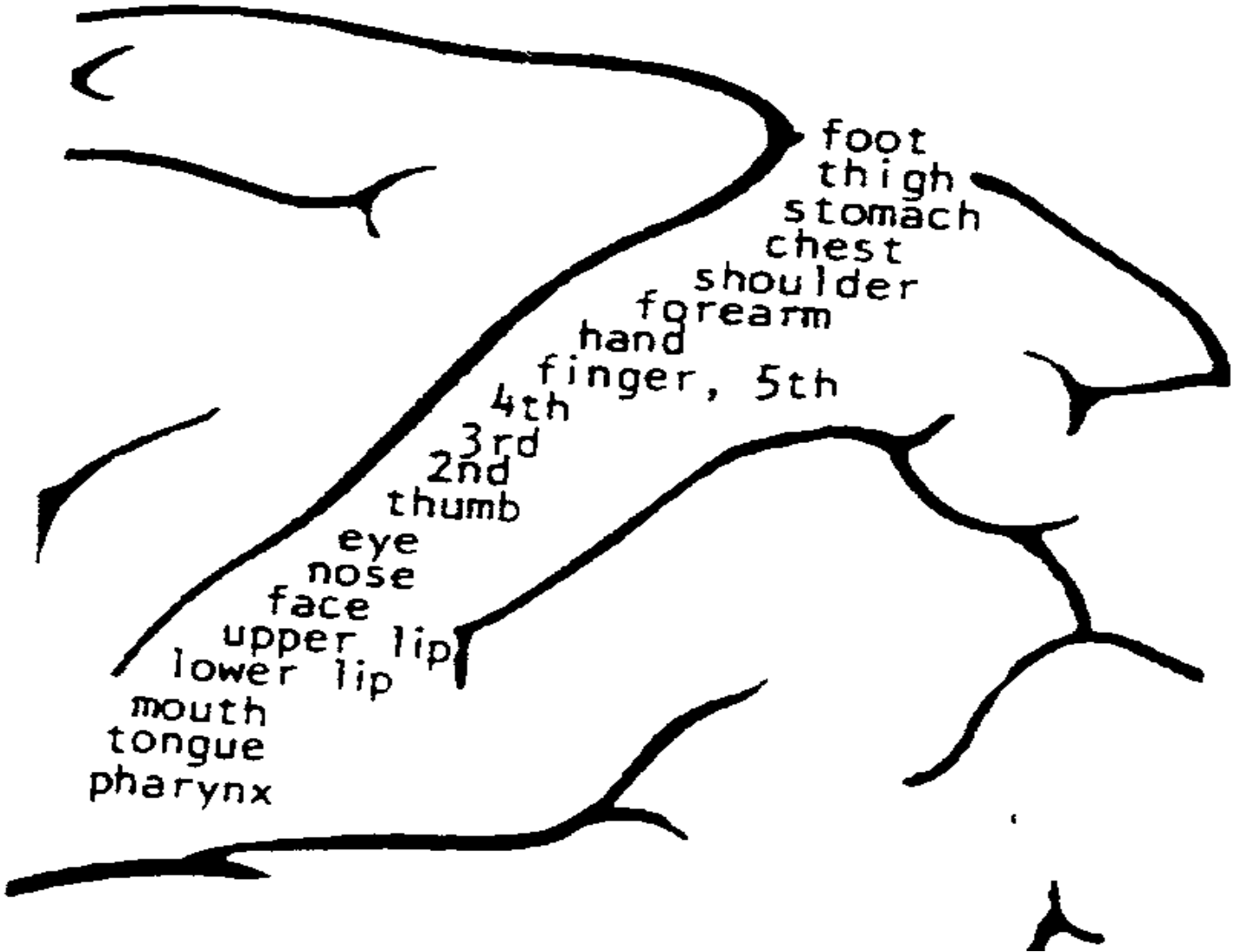
de 10x10 = 100 unités

un voisinage de rayon r : V(r)
possède (2r+1)x(2r+1) unités

(V(1/2) possede 5 unités)

v(1/2) :

Cortex sensoriel (Kohonen)



Cortex sensoriel

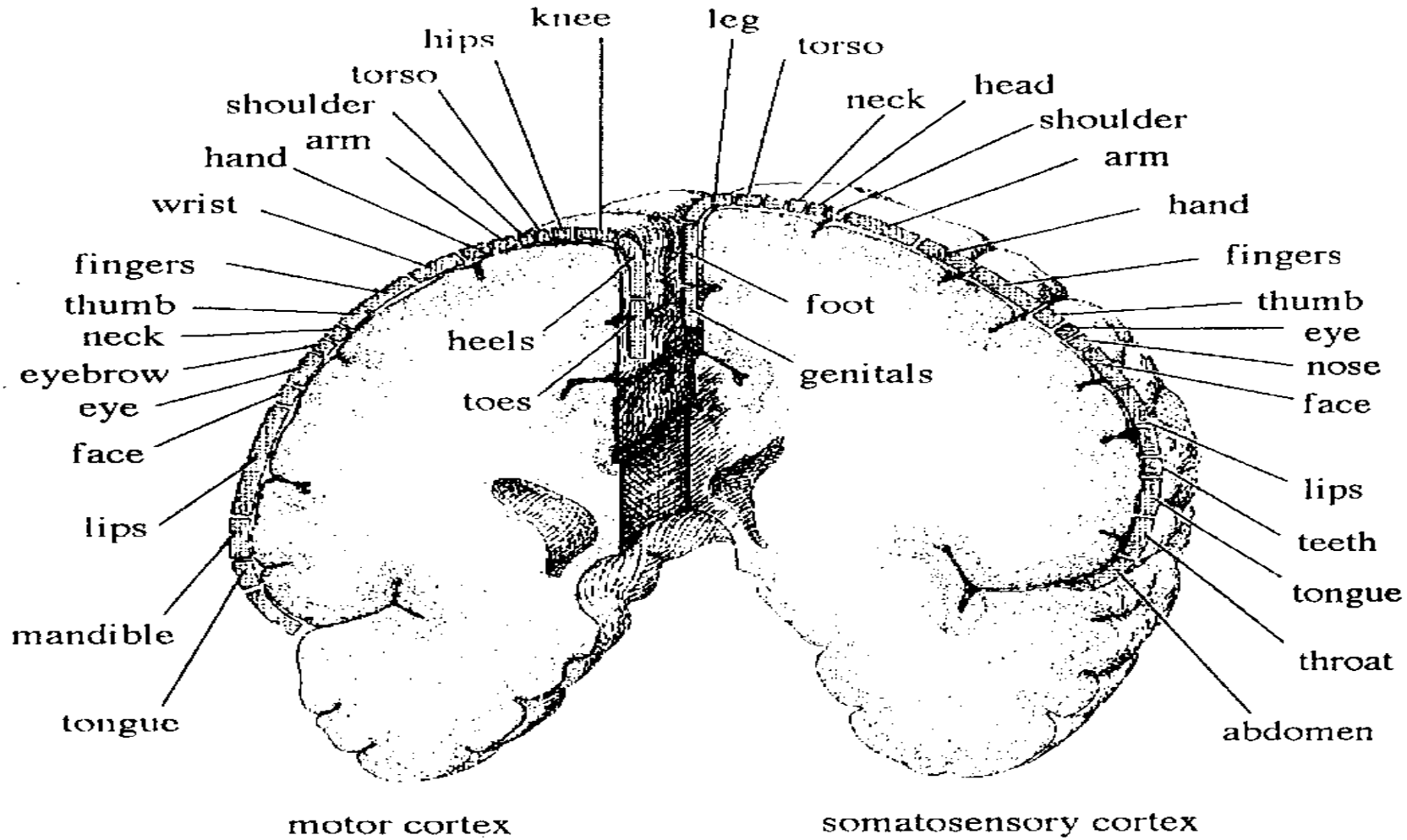
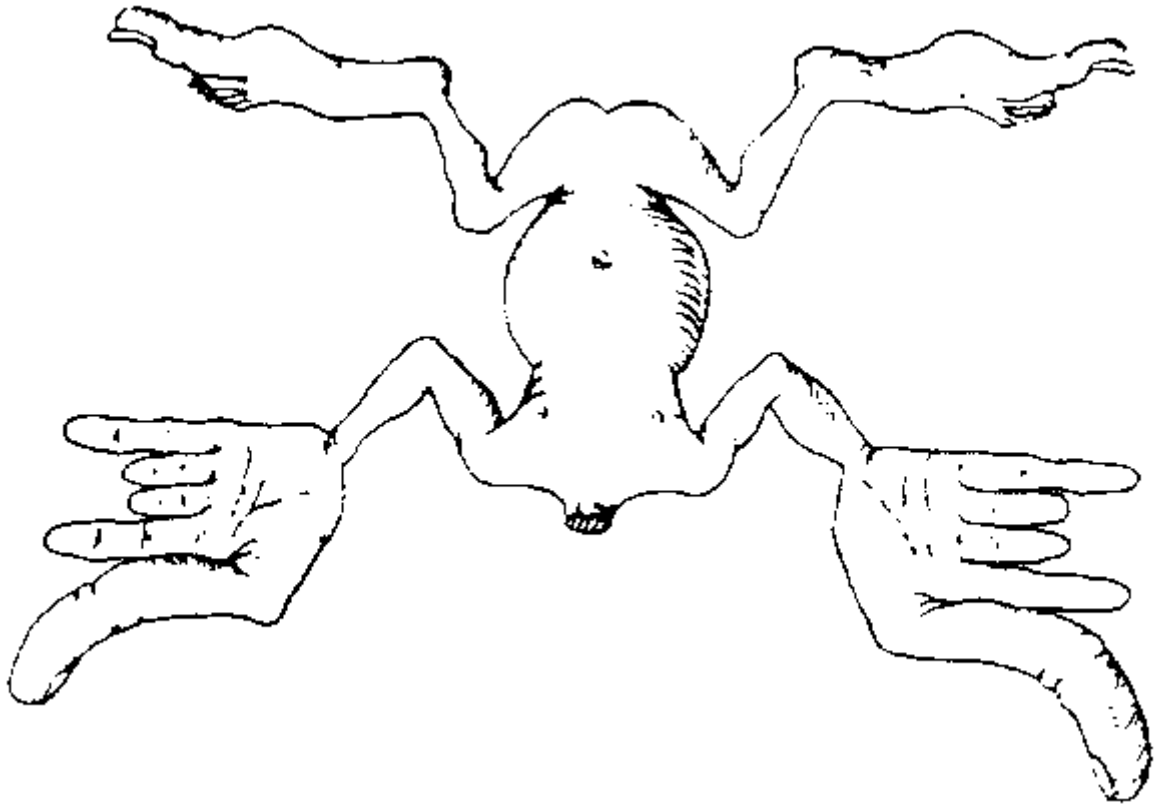
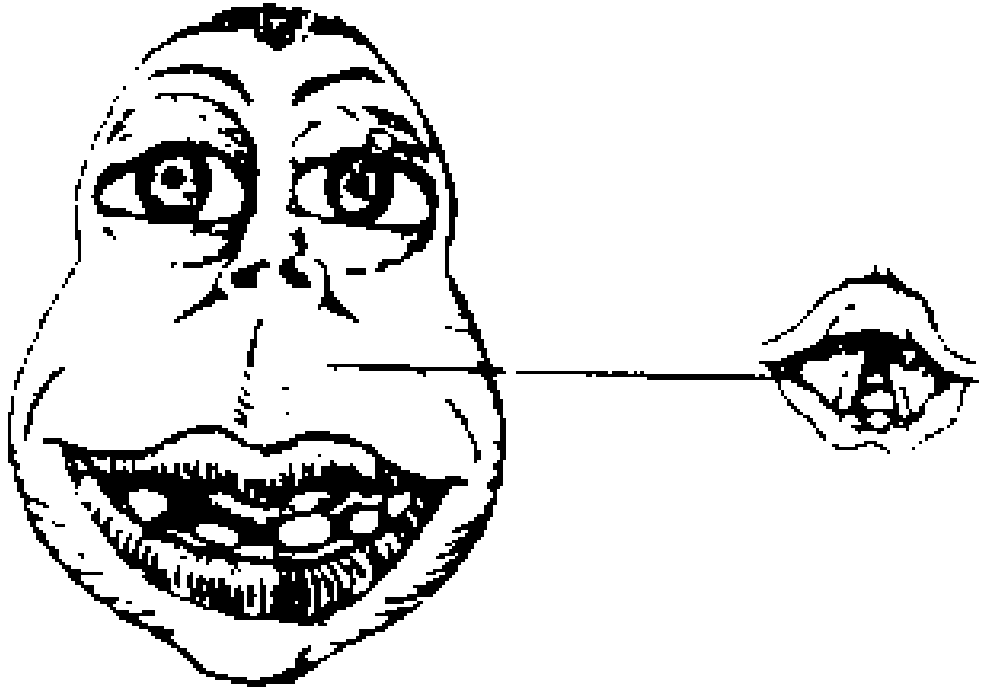


Fig. 15.3. The somatosensory and motor cortex

Homonculus (Anderson, Penfield and Boldrey)



Tête d’homonculus (Anderson, Penfield and Boldrey))



Résumé des choix à faire

- Nb de classes, de super-classes
- Géométrie : grille, ficelle, cylindre, tore, maillage hexagonal
- Initialisation des vecteurs codes (dans les données, dans l'enveloppe convexe, dans le premier plan principal)
  - Cela permet d'obtenir des minima différents

- Pour constituer les super-classes : classification hiérarchique des vecteurs codes, ficelle de Kohonen sur les vecteurs codes
  - Cela permet une typologie plus facile à décrire, une ficelle fournit un score ordonné
- Le choix d'une ficelle sur les données de départ fournit un score, croissant ou décroissant.

Méthodes adaptées aux variables quantitatives

Initialisation Recommandée

Algorithme	*Proc*	*Déter*	*Stoch*	*Voisi*	*Orga*	*Dom*	*Données*	*Plan*
Forgy	**FASTCLUS**	*		Non	Non	*	*	*
SCL	**KFAST**		*	Non	Non	*	*	*
SOM	**KACP**		*	Oui	Oui	*	*	*
Batch	**KBATCH**	*		Oui	Oui		*	**

Les sorties obtenues (KFAST, KACP, KBATCH)

- Classification des données
- Représentations des classes de Kohonen, leurs contenus, les distances mutuelles
- Les vecteurs-codes
- Super-classes, leurs contenus
- Statistiques mono- et multi-dimensionnelles permettant de qualifier les classifications obtenues
- Visualisation de la distorsion étendue
- Variations de chaque variable selon les classes obtenues

- Fabrication d'une variable qualitative (numéro de la classe)
- Croisement possible avec les autres variables qualitatives

Méthodes adaptées aux variables qualitatives

- Pour un tableau de contingence : **KORRESP**
- Pour les seules modalités d'un tableau de réponses de plus de 2 questions (table de Burt) : **KACM**
- Pour les modalités et les individus d'un tableau de réponses de plus de 2 questions (tableau disjonctif complet) :

  KACM1, KACM2, KDISJ

- **KACM2** classe les modalités comme **KACM**, puis classe les individus comme des « modalités » supplémentaires, via le tableau disjonctif.
- **KACM1** classe d'abord les individus à partir du tableau disjonctif, puis les modalités comme des « individus » supplémentaires, via la table de Burt.
- **KDISJ** classe simultanément les individus et les modalités, à partir du tableau disjonctif complet.

Traitement des données manquantes

- Deux possibilités

- Pendant l'apprentissage,
  - On se sert des observations avec données incomplètes comme des autres (quand elles sont tirées aléatoirement), et on calcule les distances en se restreignant aux composantes présentes.

- Après l'apprentissage
  - On fait l'apprentissage avec les observations complètes, puis on classe les observations incomplètes dans les classes obtenues. Les distances sont calculées avec les composantes présentes.