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

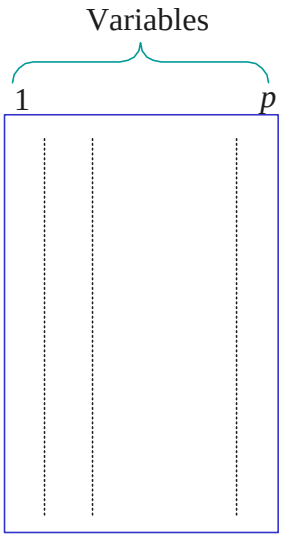
Accélération de la classification

Traitements des variables qualitatives

Données manquantes

Conclusion

Analyse de données



Comment extraire de l'information ?

1

N et *p* sont grands

Observations

N

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
- ✓ $G_1, G_2, \ldots, G_n$: classes

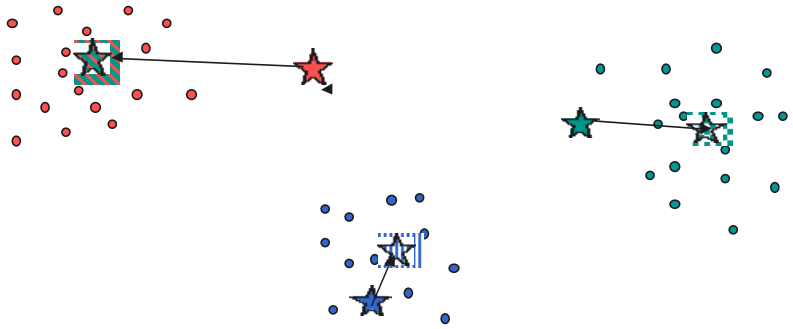
BUT : Minimiser la **distorsion quadratique** (l'erreur)
(= **Somme des carrés intra**)

$$D_O(f, C_1, C_2, \cdots, C_n) = \sum_{i=1}^{n} \int_{G_i} \|x - C_i\|^2 f(x)\, dx \quad (1)$$

Estimée par $$\hat{D}_O(f, C_1, C_2, \cdots, C_n) = \frac{1}{N} \sum_{i=1}^{n} \sum_{x_j \in G_i} \|x_j - C_i\|^2 \quad (2)$$

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

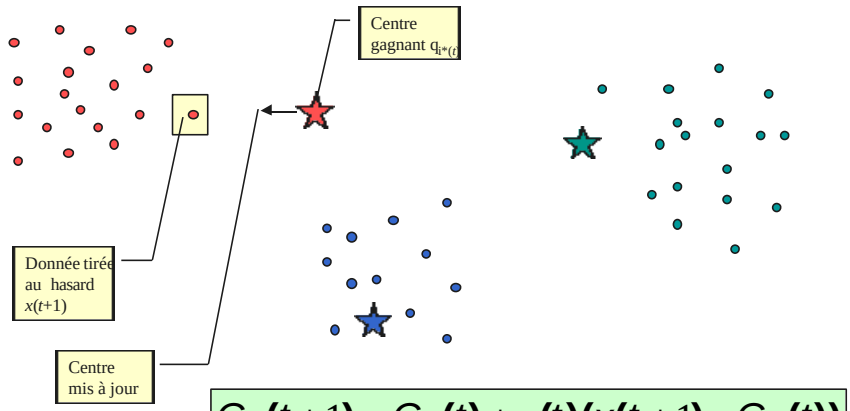
- ✓ A chaque étape, les classes sont définies (par les plus proches voisins), et les vecteurs codes sont re-calculés comme les centres de gravité des classes, etc.



- ✓ (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



$$C_{i^*}(t+1) = C_{i^*}(t) + e(t)(x(t+1) - C_{i^*}(t))$$

- ***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

Accélération de la classification

Traitements des variables qualitatives

Données manquantes

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

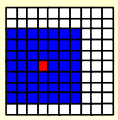
- ✓ Nombreuses applications en représentation de données de grande dimension sur des réseaux de dimension 1 ou 2, ou classification où la notion de classes voisines a un sens

L'algorithme

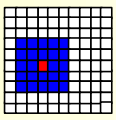
- ✓ Il s'agit d'un algorithme original de classification qui a été défini par Teuvo Kohonen, dans les années 80.
- ✓ L'algorithme regroupe les observations en classes, en respectant la topologie de l'espace des observations. Cela veut dire qu'on définit a priori une **notion de voisinage entre classes** 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

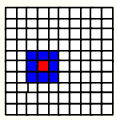
- ✓ Les grilles ne sont pas nécessairement carrées



Voisinage de 49



Voisinage de 25



Voisinage de 9

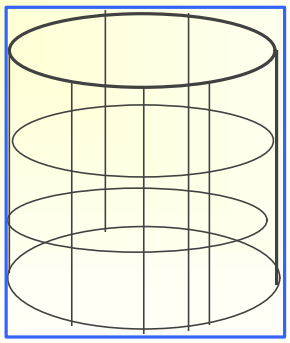
Voisinage de 7

Voisinage de 5

Voisinage de 3

Structure en cylindre

CYLINDRE



L'algorithme

Principe de l'algorithme de Kohonen

- ✓ L'algorithme de ***classement*** est **itératif**. L'initialisation consiste à 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**, mais dans ce dernier cas, il n'existe pas de notion de voisinage entre classes et 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 poids C_i
- ✓ Initialisation aléatoire des poids
- ✓ 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
- ✓ $e(t)$ est le **paramètre d'adaptation**, positif, <1, constant ou lentement décroissant
- ✓ La **fonction de voisinage** $s(i,,j)$=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

$$i_0(t+1) = \mathbf{argmin}_i \left\| x(t+1) - C_i(t) \right\|$$

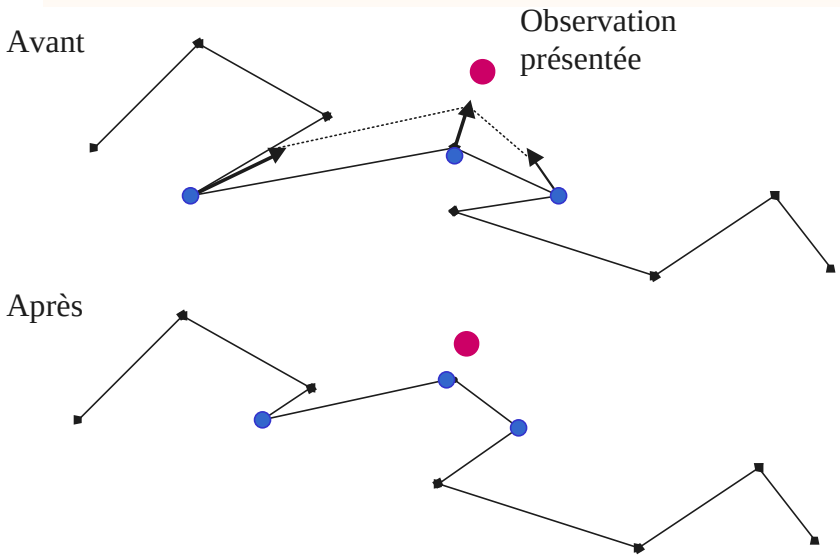
  - On met à jour les vecteurs codes

$$C_i(t+1) = C_i(t) + e_{t+1}\, s\,(i_0(t+1), i)(x(t+1) - C_i(t))$$

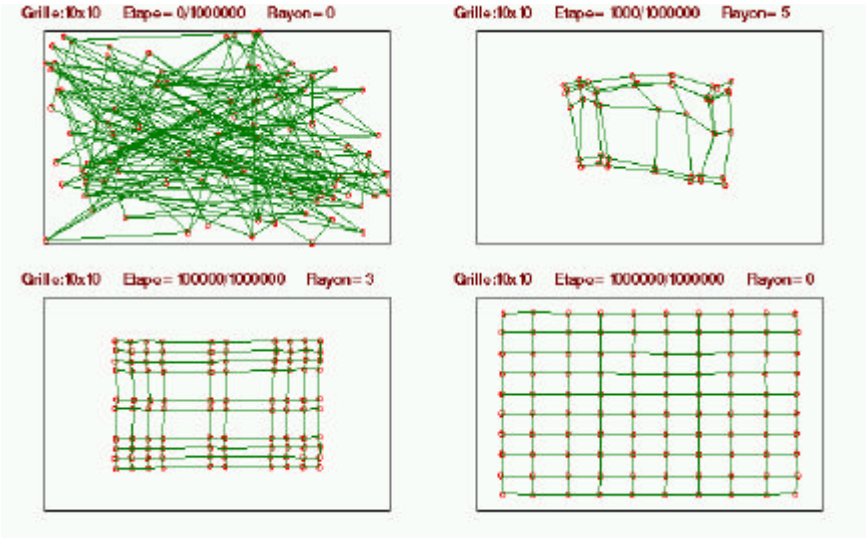
Kohonen / SCL

- ✓ 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



Étude Théorique

- ✓ On peut écrire

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

- ✓ C'est un 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. Dans ce cas, 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

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

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

- ✓ estimée par

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

Somme des carrés intra-classes étendue aux classes voisines

- ✓ C'est une **extension de la notion de somme des carrés intra-classes, qui est étendue aux classes voisines**

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

- ✓ 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

ODE associée

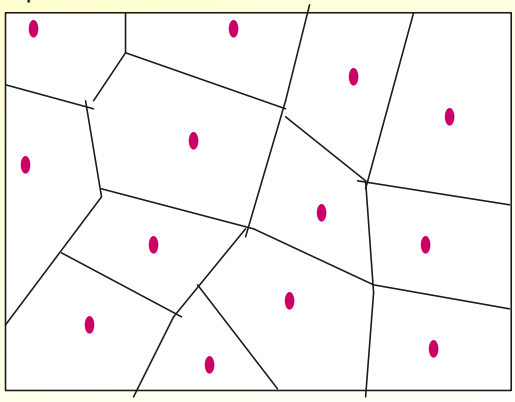
- On peut écrire l'équation différentielle ordinaire associée

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

- 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
- $G_i(C) = \{x \ / \ ||C_i - x|| = \min_j || C_j - x || \}$ est la i-ème classe formée des données pour lesquelles $C(i)$ est le vecteur code gagnant. Ces classes dépendent des valeurs courantes de tous les vecteurs-codes. Elles forment une mosaïque (tesselation, ou couverture) de Voronoi.

Mosaïque de Voronoï

Dans l'espace des entrées, les classes forment une partition
Par exemple en dimension 2



X appartient à $G_i \Leftrightarrow$ l'unité i gagne quand on présente i

Points fixes de l'ODE

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

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

- ✓ i.e.

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

- ✓ Pour chaque i, C^*_i est le barycentre des toutes les classes, pondérées par les valeurs de la fonction $s(i,j)$, $j \in I$, (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

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

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

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

- ✓ 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

$$m_N = \frac{1}{N}\sum_{l=1}^{N} d_{x_l}$$

✓ si m_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

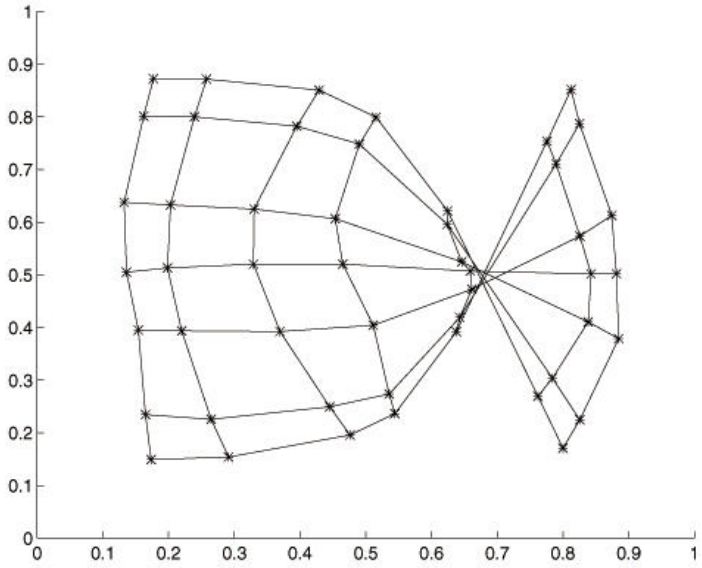
$$C_N^{k+1} = C_N^k - diag\nabla^2 D_{SOM}(C_N^k))^{-1}\nabla D_{SOM}(C_N^k)$$

✓ 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)

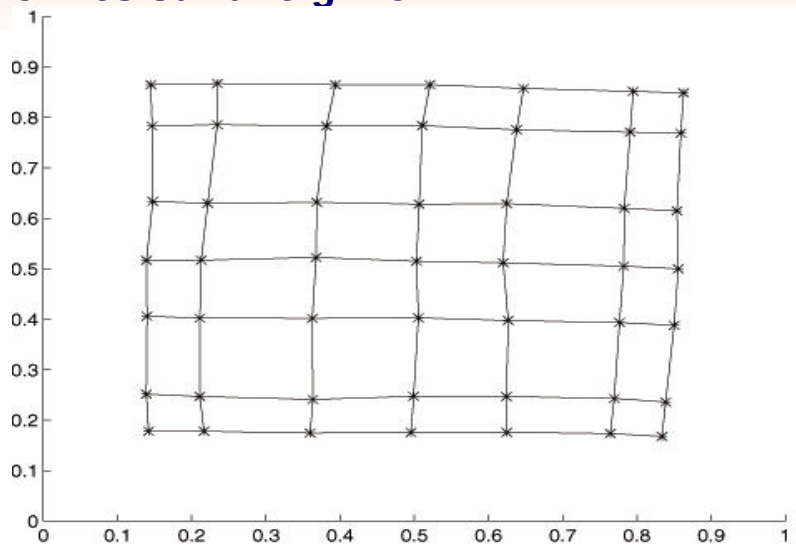
Comparaison sur données simulées

- ✓ 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

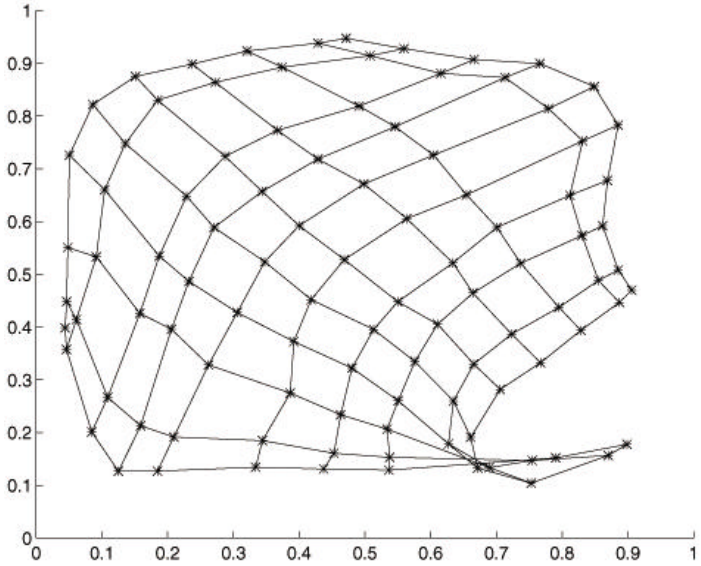
Algorithme batch pour des données uniformes sur une grille 7´7



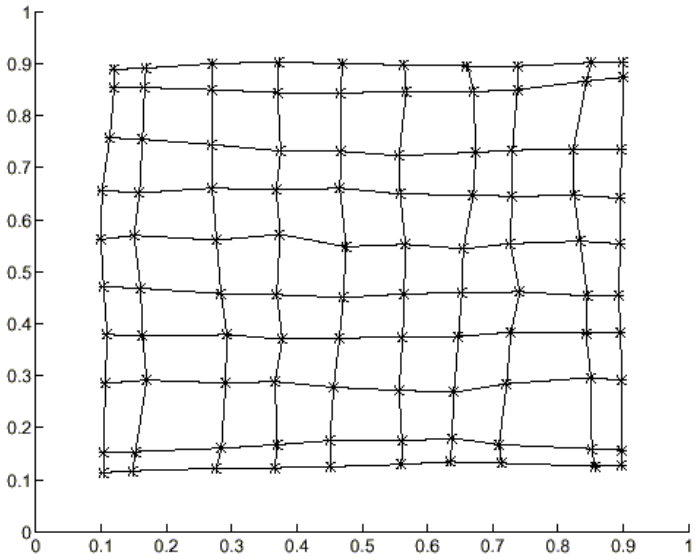
Algorithme on-line SOM pour des données uniformes sur une grille 7´7



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

Accélération de la classification

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 G_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)

Nombreuses applications

- ✓ Représentation des pays, (Blayo et Letremy)
- ✓ Communes d'Ile-de France, (Ibbou, Tutin)
- ✓ Courbes de consommation
  classification et 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 rôle du leasing en Belgique, (de Bodt, Ibbou)
- ✓ Classification des profils de chocs de taux d'intérêts, (de Bodt), etc...

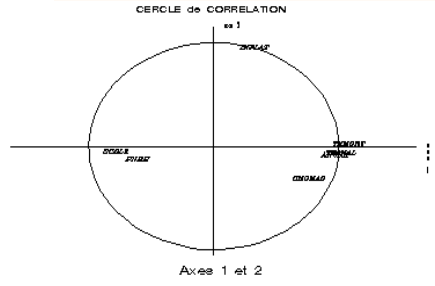
Un exemple : 96 pays en 1996

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)

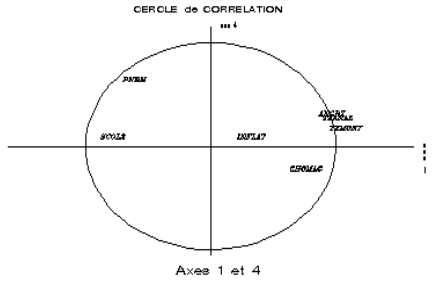
Les données

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 un	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
Fidji	1,6	26,9	9	64	2077	5,5	1,5	fort1	5
Finlande	0,3	5,8	0,1	119,2	18803	18,4	2,2	fort2	6

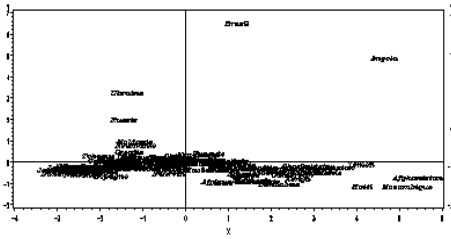
Analyse en Composantes Principales

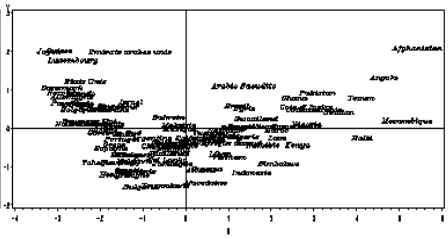


Axes 1 et 2



Axes 1 et 4





Exemple : 96 pays classés suivant 7 variables classiques

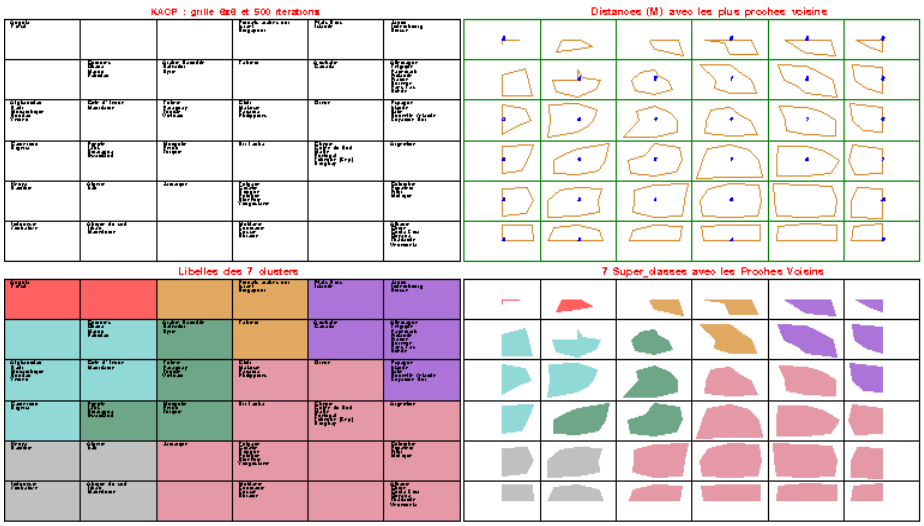
KACP : grille 6x6 et 500 iterations

Angola Brésil			Emirats arabes uni Israel Singapour	Etats Unis Irlande	Japon Luxembourg Suisse
	Comores Ghana Maroc Pakistan	Arabie Saoudite Salvador Syrie	Bahrein	Australie Canada	Allemagne Belgique Danemark Finlande France Norvege Pays Bas Suède
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 Thaïlande Venezuela

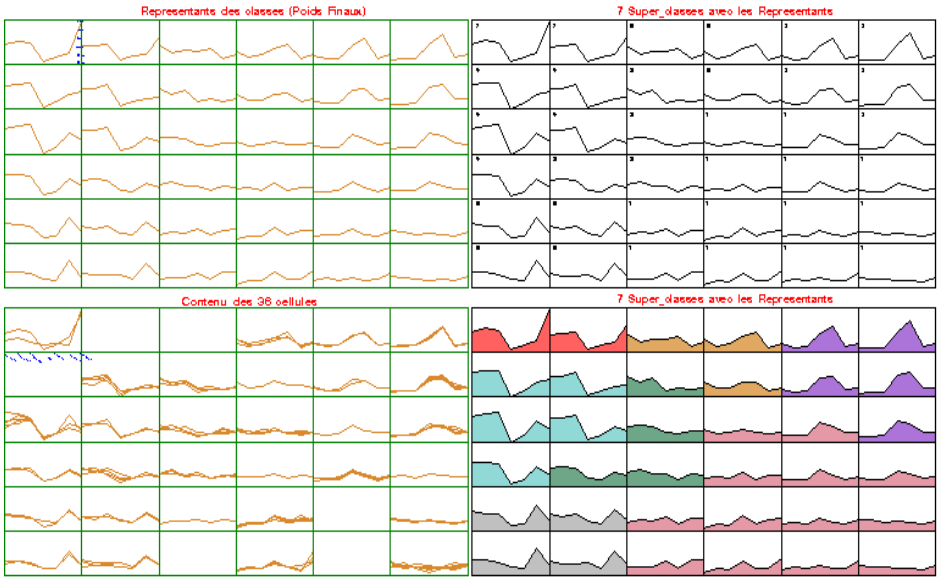
Classes et distances

- ✓ 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

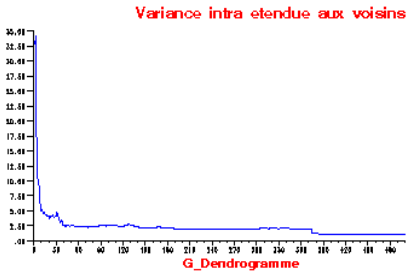
Classes, super-classes, distances

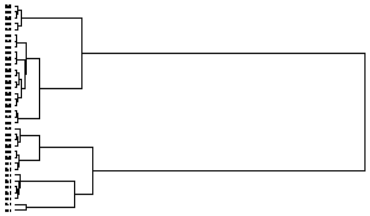


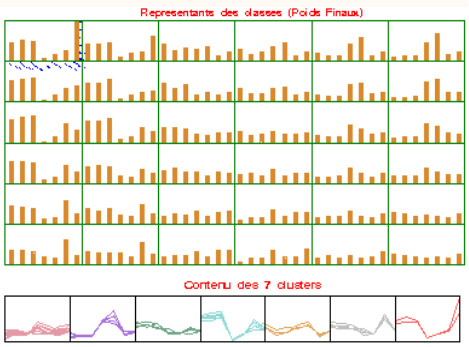
Vecteurs-codes, contenus des classes, et super-classes



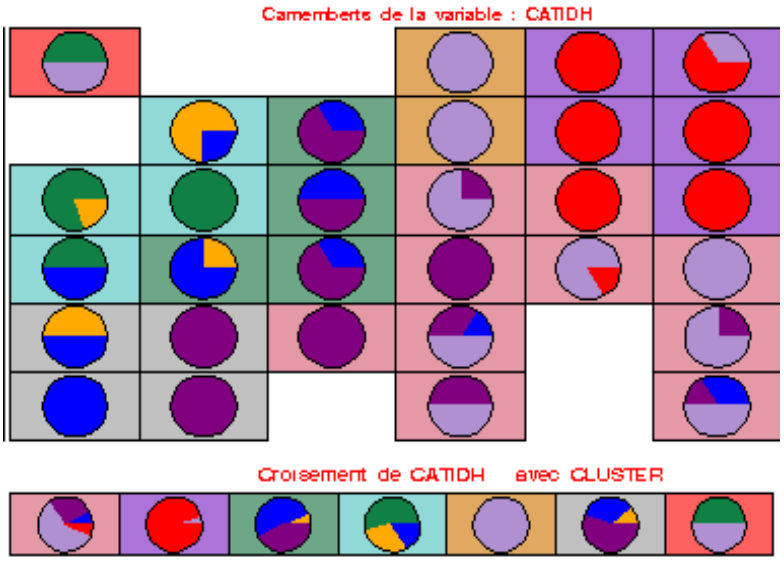
Divers



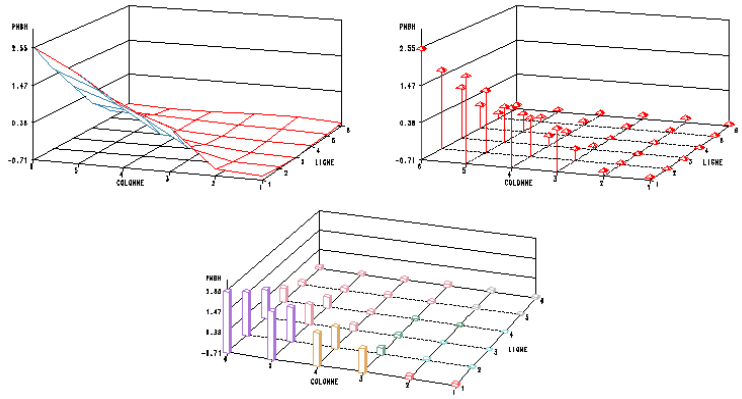




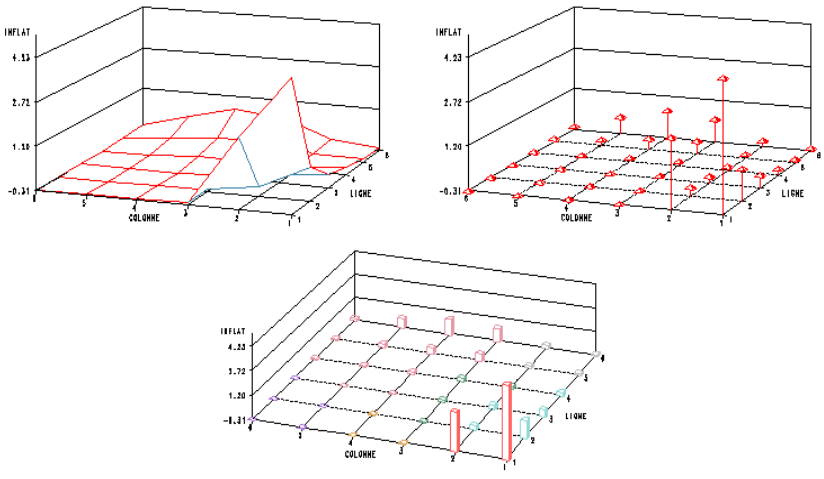
Croisement avec la variable qualitative Niveau de Développement Humain



PNB par habitant le long de la grille



Inflation le long de la grille



Observations avec données manquantes

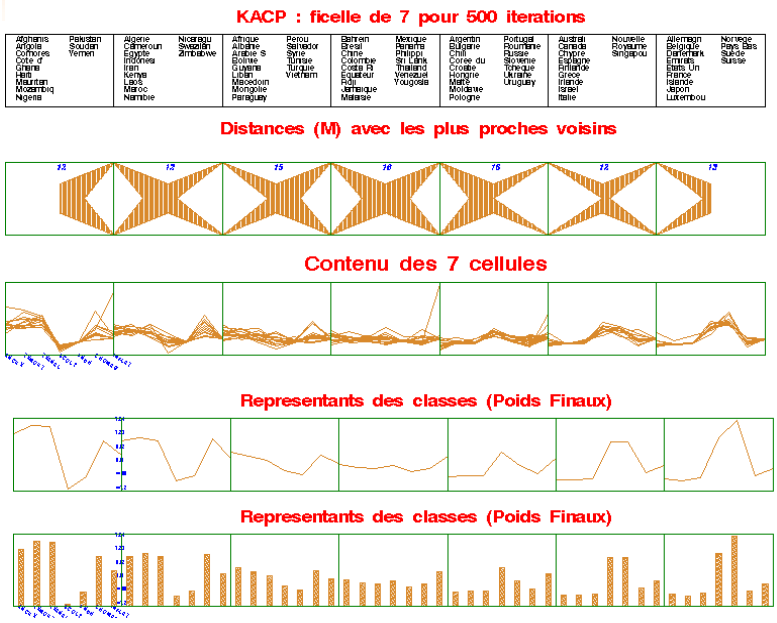
PAYS	ANCRX	TXMORT	TXANAL	SCOL2	PNIBH	CHOMAG	INFLAT
Andorre	2,7	6,6	0,5	21,7	13235	0,1	
Bangladesh	2,3	109	65,5	19	229		4,3
Benin	3,3	118,8	69,6	12	540		38,5
Birmanie	2,1	83,8	19,5	23	924		37
Congo	2,9	64,9	25		700	25	40,2
Coree du Nord	1,7	23,9	1		595		5,1
Cuba	0,6	9,2	3	77	580		
Dominique	0	18,3	4		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		Liechtenstein
	Vanuatu	Irak Jordanie Libye			
Benin Guinee Liberia	Bangladesh				
	Inde	Tuvalu	Vatican		
Birmanie Guatemala Lesotho	Congo		Cuba Dominique Grenade		
				Kirghizstan	Andorre Coree du Nord

KACP sur une ficelle 7



Analyse de données : introduction

Algorithme de Kohonen

Kohonen et classification : KACP

Accélération de la classification

Traitements des variables qualitatives

Conclusion

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***

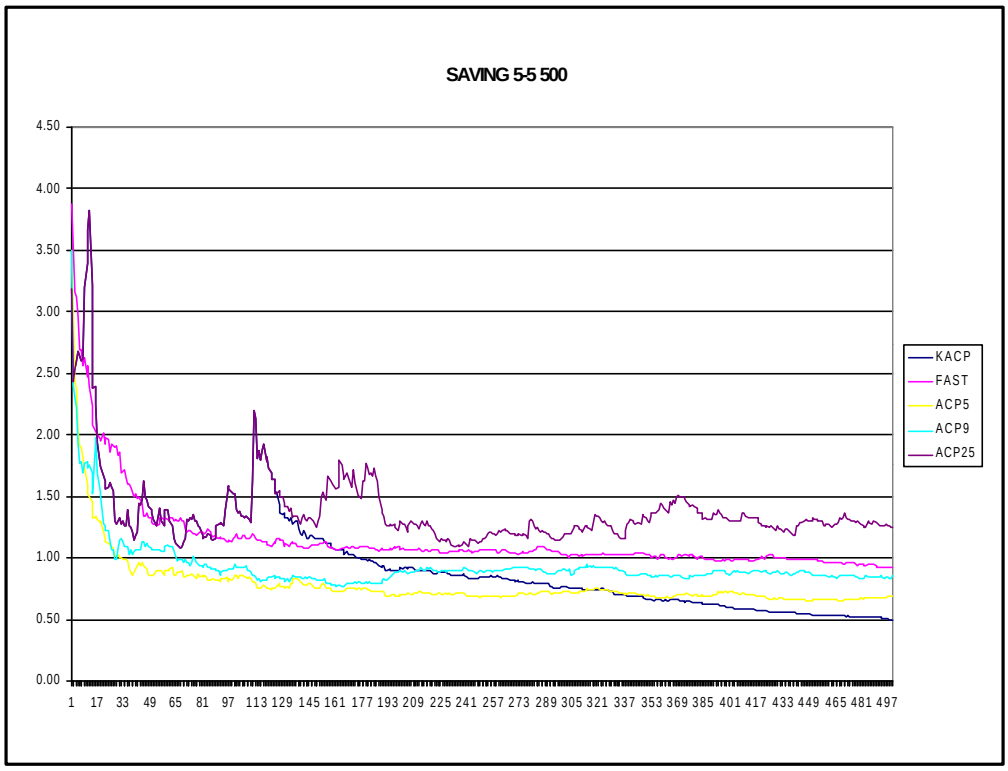
KACP pour accélérer SCL

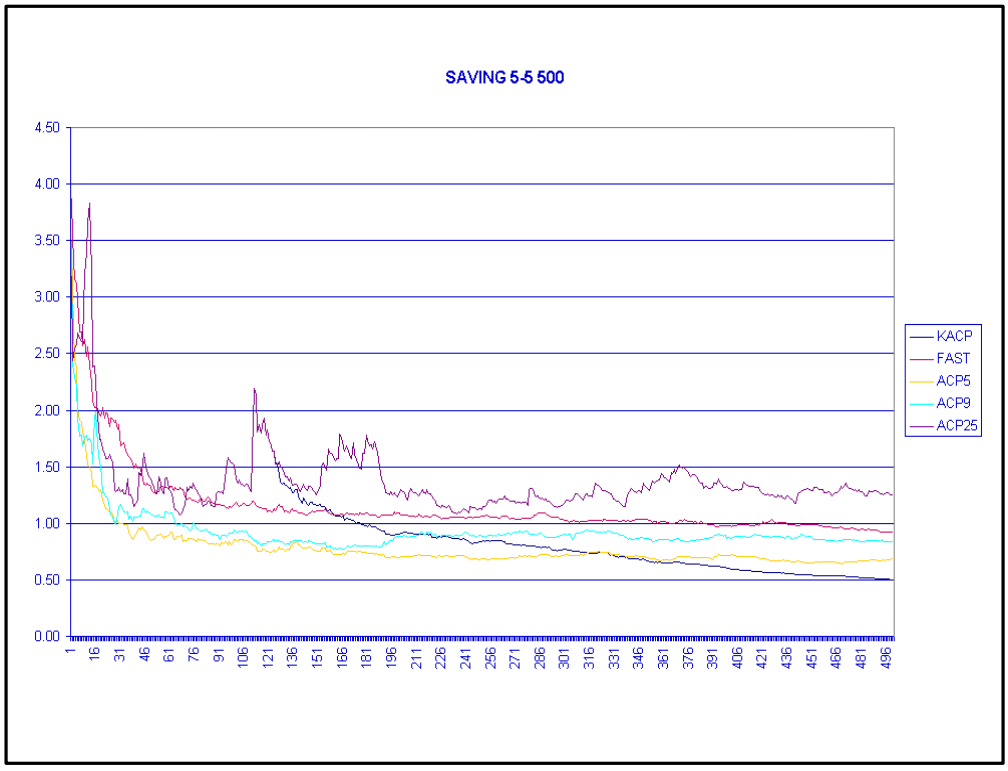
- ✓ On constate que la somme des carrés intra qu'on cherche à minimiser décroît plus vite lorsqu 'on utilise un algorithme avec des voisins, que si l'on utilise le SCL (version stochastique de l 'algorithme des centres mobiles)
- ✓ On réalise donc la classification de données
  - avec SCL (0 voisin)
  - 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 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

Country	SR	POP15	POP75	DPI	TXDPI
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
Tunisia	2.81	46.12	1.21	249.87	1.13


SAVING 5-5 500
4.50
4.00
3.50
3.00
2.50
2.00
1.50
1.00
0.50
0.00
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
KACP
FAST
ACP5
ACP9
ACP25


SAVING 5-5 500
4.50
4.00
3.50
3.00
2.50
2.00
1.50
1.00
0.50
0.00
1 16 31 46 61 76 91 106 121 136 151 166 181 196 211 226 241 256 271 286 301 316 331 346 361 376 391 406 421 436 451 466 481 496
KACP
FAST
ACP5
ACP9
ACP25

Analyse de données : introduction

Algorithme de Kohonen

Kohonen et classification : KACP

Accélération de la classification

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.

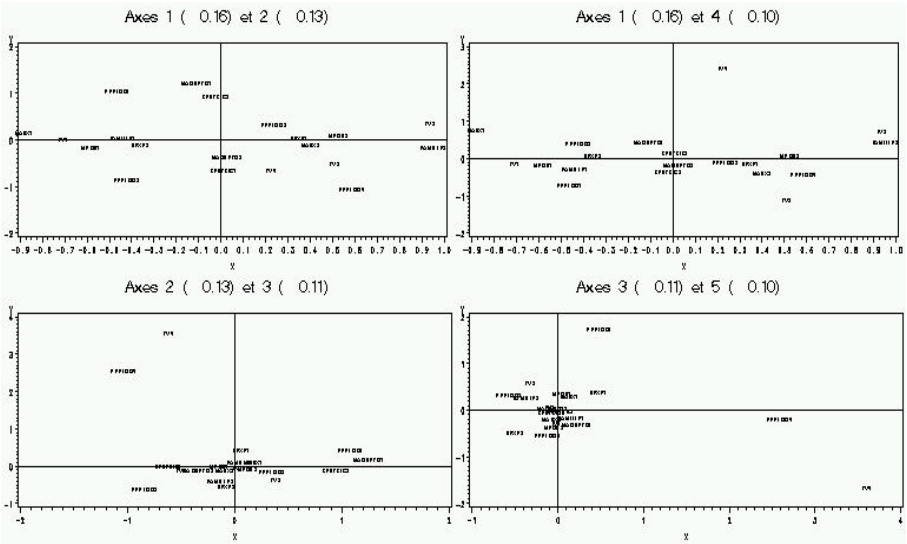
Exemple de table de Burt

	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

Un exemple

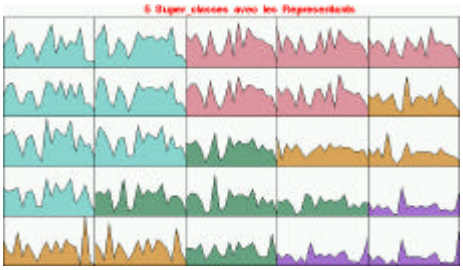
- ✓ Tiré de « Statistique exploratoire multidimensionnel » de Lebart, Morineau, Piron, (Dunod) 1995
- ✓ 105 ménages, 8 questions, 20 modalités
  - La famille est l'endroit où on se sent bien :oui, non
  - Les dépenses de logement sont une charge : négligeable, sans gros problème, une lourde charge, une 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

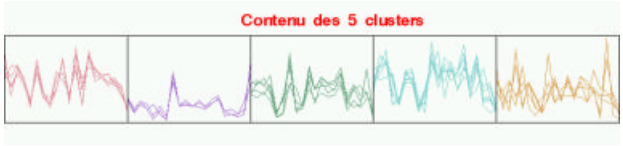
Analyse des correspondances multiples



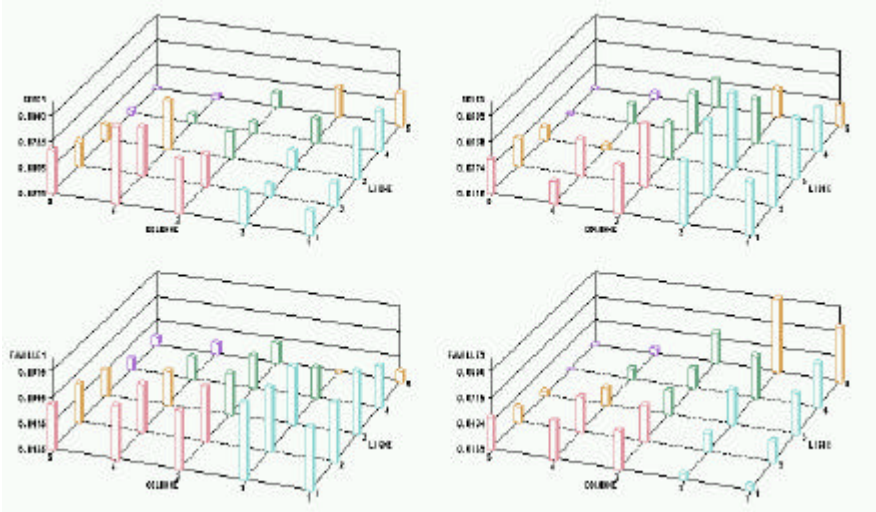
Carte des modalités

Super classes pour les modalités

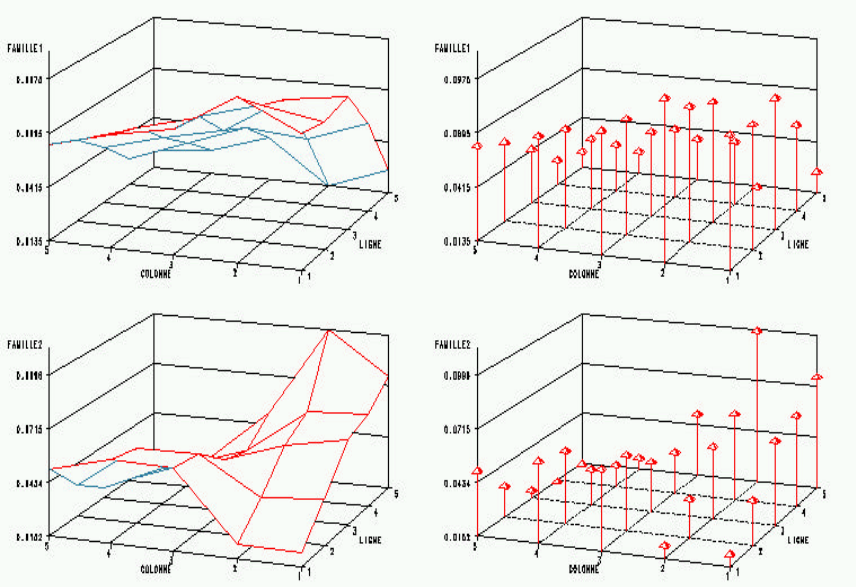




Modalités (sexe et famille) le long de la grille



Modalité famille 1 et 2



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

- ✓ 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.
- ✓ La méthode KACM1 consiste alors à pratiquer un algorithme de Kohonen sur ce tableau, avec la même normalisation et la distance du χ^2.
- ✓ On classe ainsi les individus, puis les modalités normalisées pour représenter des individus types .
- ✓ 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 (KACM1, 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.
- ✓ 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.

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) sur des variables qualitatives en y rajoutant une variable de classe obtenue par un KACP**
- ✓ 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)