Traitement statistique des données

Olivier de Cambry

Chapitre 1

Introduction

1.1 Généralités

Le développement technologique actuel permet de recueillir des données de toutes natures en quantité de plus en plus grandes. Des méthodes anciennes ou plus récentes permettent d'en extraire l'information, de les analyser et d'en déduire des décisions.
Les entreprises ont fait de ces opérations un nouveau concept appelé **exploration des données (data mining)**, mais les problèmes que peuvent résoudre les méthodes d'analyse de données dépassent la gestion prévisionnelle des entreprises, mais concernent tous les secteurs ayant besoin d'extraire et d'analyser des informations : reconnaissance de formes en imagerie, recherches bibliographiques, analyse de questionnaires,...

Les methodes utilisées sont nombreuses et relèvent d'approches diverses : **Géométrie, Statistiques, Méthodes numeriques, Arbres de décisions** ou **Méthodes connexionnistes**.

On distingue quatre types de problèmes :

Décrire et visualiser :
Ces méthodes consistent à extraire des données des caractéristiques graphiques ou numériques et de les visualiser. C'est le cas des méthodes de la statistique descriptive, des méthodes factorielles, et des méthodes de segmentation.

Modéliser et expliquer :
On cherche un modèle mathématique permettant d'expliquer la structure des données. On doit ajuster le modèle et le valider. Les modèles sont de type statistique ou neuronal.

Discriminer et classer :
Dans cette méthode on suppose que les données proviennent de plusieurs classes et à l'aide d'un ensemble d'apprentissage, on doit trouver une méthode permettant de différencier automatiquement les classes ou affecter des individus dans des classes.

Décider et prévoir :
Décider et prévoir c'est le but principal de toute méthode d'analyse des données et un enjeu essentiel dans notre société .

Dans ce polycopié on ne trouvera pas le catalogue exhaustif de toutes les méthodes utilisables pour étudier des données. On citera les grandes familles de méthodes et on détaillera certaines méthodes particulièrement dignes d'intéret. Le plan ne reprendra pas tout à fait l'ordre des problèmes énoncés précédemment car souvent les méthodes permettent de faire plusieurs opérations en même temps : visualiser et expliquer, modéliser et prévoir, ... D'autre part une bonne analyse est souvent une combinaison de plusieurs types de méthodes.

Plan

1. Le problème de la reconnaissance des formes
2. Statistique descriptive
3. Analyse en composantes principales
4. Classification
5. Discrimination

1.2 Le problème de la reconnaissance des formes

1.2.1 Introduction

La reconnaissance des formes regroupe l'ensemble des méthodes et des moyens permettant de reproduire et d'améliorer les moyens naturels de perception et de compréhension des êtres vivants.
Il ya donc dans ces méthodes un souci de reproduire une faculté possédée par l'être humain pour l'améliorer ou l'utiliser de façon automatisée.
Les motifs sont divers : nécessité économique (coût moindre d'une tâche automatisée), sécurité (surveillance des cuves nucléaires), gain de temps (recherche bibliographique), etc...
Les applications sont nombreuses :

1. *reconnaissance des caractères manuscrits ou imprimés*
2. *reconnaissance de la parole*
3. *analyse de signaux sismiques*
4. *aide au diagnostic médical*
5. *reconnaissance bibliographique*
6. *applications industrielles : robotique, contrôle de qualité*

1.2.2 Problématique

L'objectif est d'affecter à chaque **forme** observée une **classe** caractérisée par son **étiquette** (label).

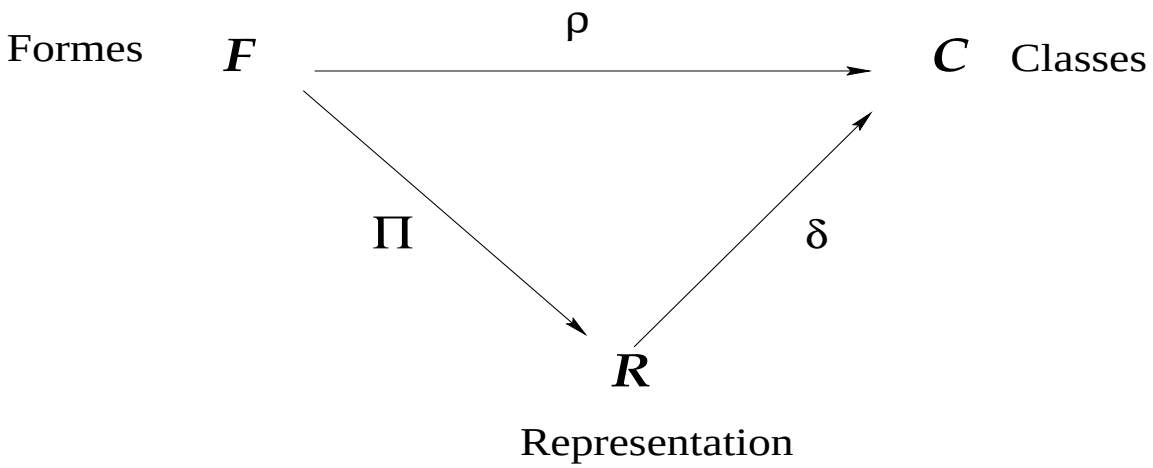

FIG. 1.1 – Reconnaissance des formes

Ce que l'on appelle forme dans ce contexte est très variée : paysage, texte, son, individus, tissu ...
Le problème se résout en 3 étapes :

Perception

L'étape de perception est l'étape de recueil de l'information : on utilise pour cela soit des appareils de mesures, soit des procédés de reproduction d'image ou du son, soit des questionnaires ou des enquêtes. En général ces données sont stockées et peuvent être traitées informatiquement.

Représentation

L'étape de représentation est l'étape au cours de laquelle, les données sont nettoyées, codées, traitées de façon à pouvoir se prêter au calcul. c'est le domaine du traitement informatique, du traitement du signal ou du traitement d'image.

Décision

La dernière étape est l'utilisation des algorithmes statistiques d'analyse des données : c'est l'étape qui nous intéressera particulièremnt dans la suite.

L'exemple (fig 1.2) concernant la reconnaissance de caractères manuscrits ou imprimés illustre ces 3 étapes.

1.2.3 Déroulement d'un processus de décision

On distingue 3 phases dans le déroulement d'un processus de décision.

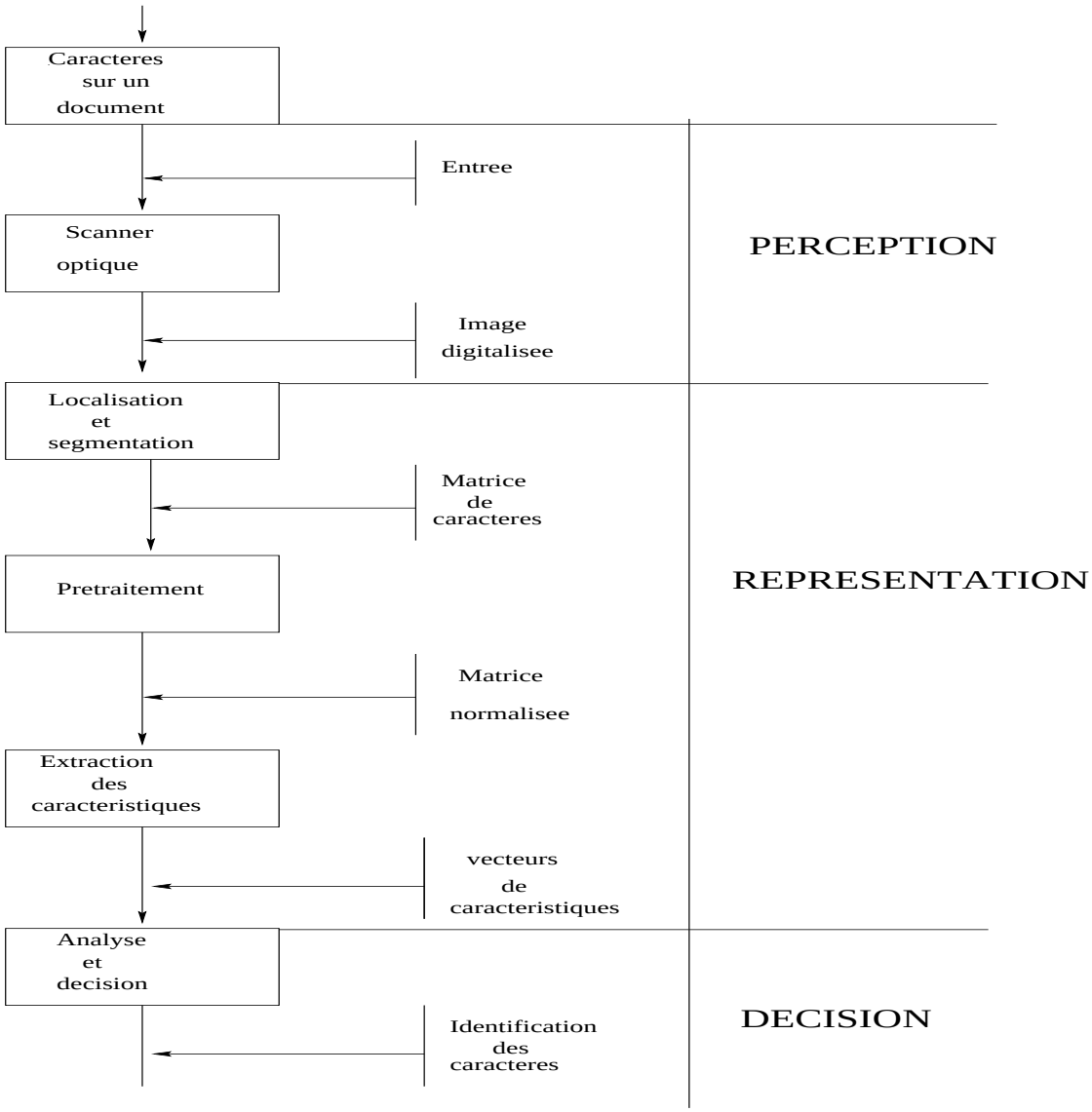

FIG. 1.2 – Reconnaissance de caractères

1.2.4 Détermination des classes

Dans certains cas les classes sont connues *a priori* par exemple la reconaissance de caractères imprimés ou le comportement d'électrocardiogrammes pour certaines affectioin cardiaques.
Dans d'autres cas il faut déterminer les classes par apprentissage. Il peut alors se présenter 2 situations :

Cas supervisé :
La classe d'appartenance des éléments de l'ensemble d'apprentissage (ou patron) est connue et il faut trouver le moyen de **discriminer** les formes à l'aide des données. C'est le cas des caractères manuscrits ou de la reconnaissance vocale.

Cas non supervisé :
La classe d'appartenance des éléments de l'ensemble d'apprentissage est inconnue et il faut en fonction de critères que l'on se donne déterminer une **classification** des formes. C'est le cas par exemple de la reconnaissance de texture en imagerie.

Elaboration d'une règle d'affectation

Il faut ici trouver la règle de décision δ qui permet l'**affectation** de la forme **F** à une classe. Cette règle est de la forme suivante : *Soit q le nombre de classes,* $\pi(F)$ *la représentation de la forme* **F***, et* $A_1, \ldots, A_q$*, une* **partition** *de l'espace de représentation* **R**

$$\delta : \pi(F) \in A_i \Longleftrightarrow F \in C_i$$

Validation de la règle d'affectation

La validation de la règle d'affectation se fait en calculant le **taux d'erreurs de classement** sur une partie de l'ensemble d'apprentissage n'ayant pas été utilisée pour élaborer la règle, appelé ensemble test.

Chapitre 2

Statistique descriptive

2.1 Les données

2.1.1 Individu, population et échantillon

Pour recueillir des données sur un phénomène on effectue plusieurs observations différentes. Chaque observation est effectuée sur un **individu**.
L'ensemble des individus concernés par le phénomène est la **population**.

On note :

ω_i l'individu numéro i ;

$\Pi = \{\omega_1, \ldots, \omega_N\}$ la population ;

N l'effectif de la population.

Un **échantillon** est une partie de la population. Il peut être tiré au hasard en respectant certains critères dans le cas d'un sondage ou imposé par l'expérience. La taille de l'échantillon est son **effectif**.

2.1.2 Caractère

Pour réaliser une étude statistique on relève sur chaque individu de l'échantillon des aspects spécifiques du phénomène étudié que l'on appelle **caractères**. Chaque caractère est représenté par une variable notée X qui peut être soit quantitative , soit qualitative. Dans la suite nous ne nous intéresserons qu'au cas de variables quantitatives.
On note :

$$x_i = X(\omega_i), \text{ le caractère de l'individu i}$$

où en général x_i est un nombre réel.
Lorsque l'on dispose de plusieurs caractères, on obtient un tableau de caractères dans lequel x_{ij} désigne la valeur du caractère numéro j, mesuré sur l'individu numéro i.
On considérera que l'individu i est représenté par le point $M_i = (x_{i1}, \ldots, x_{ip})$ de l'espace euclidien $\mathcal{R}^p$. L'ensemble des individus $C = (M_1, \ldots, M_n)$ constitue un nuage de points.

	X_1	X_2	$\dots$	X_p
ω_1	x_{11}	x_{12}	$\dots$	x_{1p}
ω_2	x_{21}	x_{22}	$\dots$	x_{2p}
$\vdots$	$\vdots$	$\vdots$		$\vdots$
ω_n	x_{n1}	x_{n2}	$\dots$	x_{np}

TAB. 2.1 – Tableau de caractères

2.2 Description d'une variable quantitative

On considère un échantillon de données numériques $(x_1, \dots, x_n)$, où x_i est un nombre réel.
Pour décrire les données on utilise plusieurs caractéristiques numériques ou graphiques.

2.2.1 Tendance centrale

Moyenne

La **moyenne** $\bar{x}$, est définie par :

$$\bar{x} = \frac{1}{n} \sum_{i=1}^{n} x_i$$

Médiane

La **médiane, M** est un nombre réel qui partage les données en deux parties de même effectif.
On effectue un réarrangement par ordre croissant des données :

$$x_{(1)} \leq x(2) \leq \dots \leq x(n)$$

La médiane, est alors définie par :

Si $n = 2k + 1$ alors $M = x_{(k+1)}$
Si $n = 2k$ alors $M = \frac{x_{(k)} + x_{(k+1)}}{2}$

2.2.2 Dispersion

Etendue

L'**étendue W**, est la distance entre la plus grande des données et la plus petite.

$$W = \max_{i=1,\dots,n} x_i - \min_{i=1,\dots,n} x_i = x_{(n)} - x_{(1)}$$

Cette caractéristique est simple à calculer mais a pour défaut d'être très sensible aux valeurs extrêmes qui peuvent dans certains cas être des données aberrantes.

Ecart type

L'**écart type** σ, est l'écart quadratique moyen à la moyenne.
L'écart quadratique moyen des données à un nombre a est défini par :

$$\sigma(a) = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(x_i - a)^2}$$

Considérons le problème de minimisation suivant :

$$|\min_a \sigma(a)$$

La solution de ce problème est obtenue pour $a = \bar{x}$.
L'écart type, est alors défini par :

$$\sigma = \sigma(\bar{x}) = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(x_i - \bar{x})^2}$$

Déviation à la médiane

La **déviation à la médiane** δ, est l'écart absolu moyen à la médiane.
L'écart absolu moyen des données à un nombre a est défini par :

$$\delta(a) = \frac{1}{n}\sum_{i=1}^{n}|x_i - a|$$

Considérons le problème de minimisation suivant :

$$|\min_a \delta(a)$$

La solution de ce problème est obtenue pour $a = M$.
La déviation à la médiane est alors définie par :

$$\delta = \delta(M) = \frac{1}{n}\sum_{i=1}^{n}|x_i - M|$$

L'écart à la médiane s'obtient simplement comme étant la demie différence entre la moyenne des donnnées supérieures à la médiane et la moyenne des données inférieures à la médiane.
Si on note,

$$\begin{aligned} k &= [\frac{n}{2}] \\ m_1 &= \frac{1}{k}\sum_{i=1}^{k} x(i) \\ m_2 &= \frac{1}{k}\sum_{i=1}^{k} x(n+1-i) \end{aligned}$$

alors

$$\delta = \frac{m_2 - m_1}{2}$$

Cette caractéristique est peu employée. On lui préfère la distance interquartile.

Distance interquartile

L'ensemble des données peut être partagé en p sous ensembles de mêmes effectifs par p-1 quantiles d'ordre p. Ainsi la médiane est un quantile d'ordre 2.
Les quantiles d'ordre 4 sont appelés quartiles et les quantiles d'ordre 10 sont les déciles.
La dispersion peut être mesurée par la **distance interquartile H**.
Soit Q_1, le premier quartile et Q_3, le troisième quartile, on a :

$$H = Q_3 - Q_1$$

On peut remarquer que le deuxième quartile n'est autre que la médiane M.

2.2.3 Caractéristiques visuelles

Histogramme

Un **histogramme** de données réelles est constitué de la manière suivante :

1. On répartit les données en classes de même largeur ;
2. On calcule les effectifs de chaque classe ;
3. On reporte les calculs sur un graphique, dans lequel les classes sont représentées par des rectangles dont la base est la largeur de la classe et la hauteur est proportionnelle à l'effectif.

Le **mode** de la distribution est la valeur correspondant à la classe la plus nombreuse.

Boîte à moustaches

La **boîte à moustaches** (en anglais **boxplot**) permet de visualiser les quartiles et les valeurs extrêmes. Elle est construite parallèlement à l'axe des données et comporte les éléments suivants :

1. Un rectangle dont les extémités sont les 1^{er} et $3^{ème}$ quartiles, Q_1 et Q_3, partagés en deux parties au niveau de la médiane M ;
2. Un segment entre la valeur minimale $x_{(1)}$, et Q_1, et un segment entre Q_3 et la valeur maximale x_n ;
3. Par convention la longueur des segments est limitée à 1.5 H : si des valeurs existent au delà on les fait figurer sur le graphique. Ces valeurs sont des données aberrantes et doivent être étudiées de près.

2.2.4 Application

Données

Le tableau suivant donne le nombre de jours de pluie observés pendant toute l'année à Paris de 1900 à 1989.

Années	0	1	2	3	4	5	6	7	8	9
1900	161	149	172	179	153	176	176	159	158	167
1910	197	146	175	176	177	164	195	165	158	181
1920	152	114	179	182	180	180	162	191	175	133
1930	197	184	159	133	152	165	176	171	176	182
1940	160	166	166	141	157	153	171	163	168	127
1950	177	188	184	133	184	148	154	161	193	131
1960	198	152	159	159	146	196	192	161	176	173
1970	199	141	170	156	198	164	135	179	171	172
1980	170	197	173	177	177	163	176	180	167	140

TAB. 2.2 – Nombres de jours de pluie par années

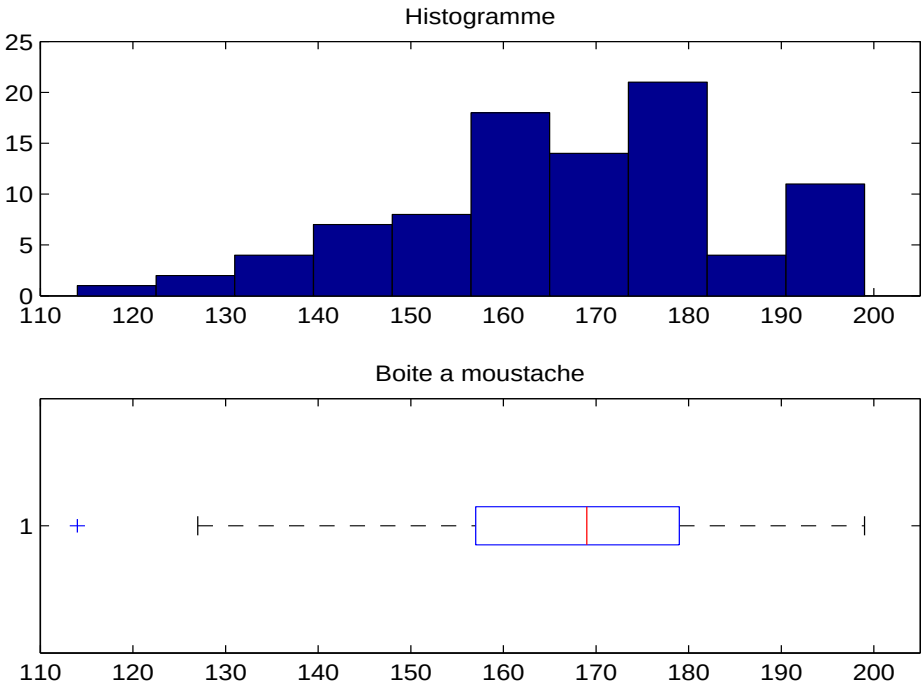

FIG. 2.1 – Caractéristiques visuelles

Description statistique des données

Moyenne : $\bar{x} = 167.1$

Médiane : $M = 169$

Etendue : $W = 85$

Ecart type : $\sigma = 18.25$

Déviation à la médiane : $\delta = 14.5$

Distance interquartile : $H = 22$

2.3 Régression linéaire

2.3.1 Généralités

Série statistique double

On considère à présent un tableau de données appelé **série statistique double** constitué de la manière suivante :

	X	Y
ω_1	x_1	y_1
ω_2	x_2	y_2
$\vdots$	$\vdots$	$\vdots$
ω_n	x_n	y_n

TAB. 2.3 – Série double

Le but est d'essayer d'expliquer les liens entre les variables X et Y.

Modèle de régression

On appelle **modèle de régression** le modèle suivant :

$$y = g(x) + e(x)$$

où :

- e est l'erreur de modélisation
- g est une fonction de transfert caractéristique du modèle

Le but est de trouver la fonction g, minimisant l'erreur à l'aide des observations.

Méthode des moindres carrés

La **méthode des moindres carrés** consiste à choisir la fonction g qui minimise la moyenne quadratique de l'erreur.
Soit C, la **fonction de coût** de l'erreur définie par :

$$C(g; X, Y) = \frac{1}{n} \sum_{i=1}^{n} (y_i - g(x_i))^2$$

g est alors solution du problème de minimisation suivant :

$$(\mathbf{P}) \qquad \min_{g \in \mathcal{G}} C(g; X, Y)$$

Dans ce problème $\mathcal{G}$ est un espace de fonctions fixé en tenant compte de la complexité du problème.
La fonction $\hat{g}$, solution du problème **(P)**, est la **fonction de régression**.
De plus le carré de l'erreur e^2 peut être estimé par :

$$\hat{e}^2 = C(\hat{g}; X, Y)$$

Types de régression

$g(x)$	$=$	$ax + b$	**régression linéaire simple**
$g(x)$	$=$	ax^b	**régression exponentielle**
$g(x)$	$=$	$a_n x^n + a_{n-1} x^{n-1} + \ldots + a_0$	**régression polynomiale**
$g(x)$	$=$	$\sum_{j=1}^{p} a_j x_j$	**régression linéaire multiple**

Ce dernier cas se produisant lorsque l'on veut expliquer la variable Y par p variables explicatives $(X_1, \ldots, X_p)$.

2.3.2 Régression linéaire simple

Calcul des paramètres de la droite de régression

Le modèle est défini de la manière suivante :

$$y_i = a\, x_i + b + e_i \qquad i = 1, \ldots, n$$

Le critère à minimiser est le suivant :

$$C(a, b; X, Y) = \frac{1}{n} \sum_{i=1}^{n} (y_i - a x_i - b))^2$$

Pour minimiser C on annule les dérivées partielles par rapport aux variables a et b :

$$\left| \begin{array}{lcl} \frac{\partial C}{\partial a} & = & \frac{-2}{n} \sum_{i=1}^{n} x_i \, (y_i - a x_i - b) \\ \frac{\partial C}{\partial b} & = & \frac{-2}{n} \sum_{i=1}^{n} (y_i - a x_i - b) \end{array} \right.$$

On note :

$$\begin{array}{lcllcl} \bar{x} & = & \frac{1}{n} \sum_{i=1}^{n} x_i & ; \quad \bar{y} & = & \frac{1}{n} \sum_{i=1}^{n} y_i \\ \bar{x^2} & = & \frac{1}{n} \sum_{i=1}^{n} x_i^2 & ; \quad \bar{y^2} & = & \frac{1}{n} \sum_{i=1}^{n} y_i^2 \\ \bar{xy} & = & \frac{1}{n} \sum_{i=1}^{n} x_i y_i & ; & & \end{array}$$

Avec ces notations on obtient les solutions $\hat{a}$ et $\hat{b}$:

$$\left| \begin{array}{rcl} \hat{a} & = & \frac{\bar{xy}-\bar{x}\bar{y}}{\bar{x^2}-\bar{x}^2} \\ \\ \hat{b} & = & \bar{y} - \hat{a}\bar{x} \end{array} \right.$$

La droite de régression est la droite d'équation :

$$Y = \hat{a}X + \hat{b}$$

Elle passe par le **centre de gravité** du nuage de points, G, dont les coordonnées sont : $(\bar{x}, \bar{y})$

Adéquation par rapport au modèle

Pour mesurer l'adéquation du modèle par rapport aux observations, on étudie les **résidus** :

$$\hat{e}_i = y_i - \hat{a}x_i - \hat{b} \qquad i = 1, \ldots, n$$

Le carré de la distance entre la droite de régression et le nuage de points est alors mesuré par :

$$\hat{e}^2 = \frac{1}{n}\sum_{i=1}^{n} e_i^2$$

On peut alors étudier l'écart au modèle soit à l'aide du **coefficient de corrélation**, soit par une **étude graphique des résidus**.

- a - Le coefficient de corrélation

On note :

$$\begin{array}{rcl} S_{xy} & = & \bar{xy} - \bar{x}\bar{y} \\ S_{xx} & = & \bar{x^2} - \bar{x}^2 \\ S_{yy} & = & \bar{y^2} - \bar{y}^2 \end{array}$$

Le coefficient de corrélation r, est alors défini par :

$$r = \frac{S_{xy}}{\sqrt{S_{xx}S_{yy}}}$$

On a alors :

$$\begin{array}{rcl} \hat{a} & = & \dfrac{S_{xy}}{S_{xx}} \\ \hat{b} & = & \bar{y} - \dfrac{S_{xy}}{S_{xx}}\,\bar{x} \\ \hat{e}^2 & = & S_{yy}(1 - r^2) \end{array}$$

Or $S_{yy} = \sigma^2(Y)$, où $\sigma(Y)$ désigne l'écart type sur Y. Donc l'erreur, $\hat{e}$ dépend de l'écart type sur Y et du coefficient de corrélation . De plus puisque $\hat{e}^2 \geq 0$, r est donc compris entre -1 et +1.
Sa signification est la suivante :

- Si r est proche de +1, les caractères X et Y évoluent dans le même sens.
- Si r est proche de -1, les caractères X et Y évoluent dans un sens opposé.
- Si r est proche de 0, les caractères X et Y évoluent de manière indépendante. On dit qu'ils sont **décorrélés**.

Donc plus $|r|$ est proche de 1, plus la droite de régression est proche de chacun des points. On considère que l'on peut dire que Y dépend linéairement de X si $|r| \geq 0.75$.

- b - Etude graphique des résidus

Les résidus normalisés sont définis par :

$$\tilde{e}_i = \frac{e_i}{\hat{e}} \qquad i = 1, \ldots, n$$

La répartition de ces résidus montrent la linéarité du modèle.

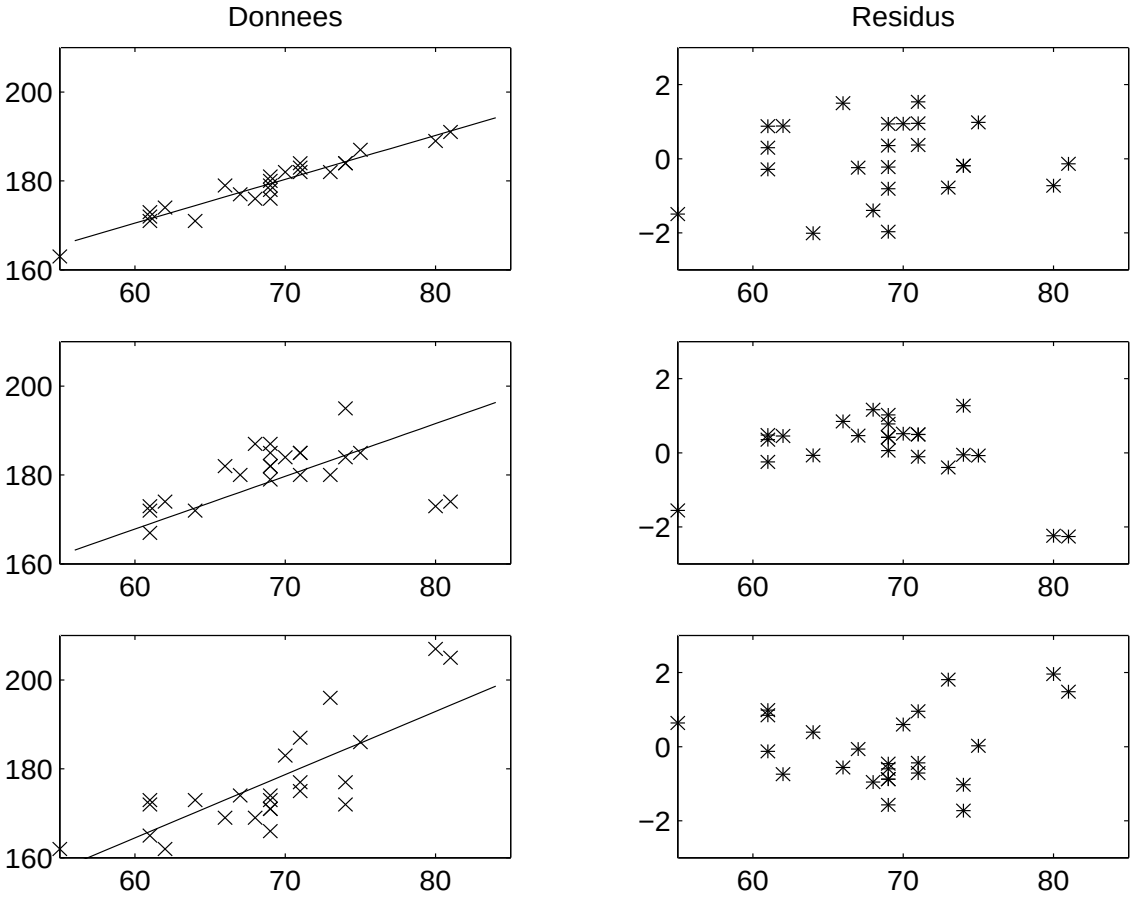

FIG. 2.2 – Différents types de résidus

Sur la figure 2.2, on peut observer différents cas :

1. un cas où les résidus sont bien répartis
2. un cas où la répartition est polynomiale
3. un cas où la répartition est exponentielle

2.3.3 Application

Le tableau suivant donne le nombre de jours de pluie et la hauteur de pluie en mm, observés pendant toute l'année à Paris de 1956 à 1995.
Le but est de savoir dans quelle mesure la hauteur de pluie est expliquée par le nombre de jours de pluie.

Années	Jours	Hauteur	Années	Jours	Hauteur
1956	154	545	1976	135	417
1957	161	536	1977	179	717
1958	193	783	1978	171	743
1959	131	453	1979	172	729
1960	198	739	1980	170	690
1961	152	541	1981	197	746
1962	159	528	1982	173	700
1963	159	559	1983	177	623
1964	146	521	1984	177	745
1965	196	880	1985	163	501
1966	192	834	1986	176	611
1967	161	592	1987	180	707
1968	176	634	1988	167	734
1969	173	618	1989	140	573
1970	199	631	1990	149	501
1971	141	508	1991	140	472
1972	170	740	1992	154	645
1973	156	576	1993	155	663
1974	198	668	1994	192	699
1975	164	658	1995	162	670

Tab. 2.4 – Nombres de jours de pluie par années

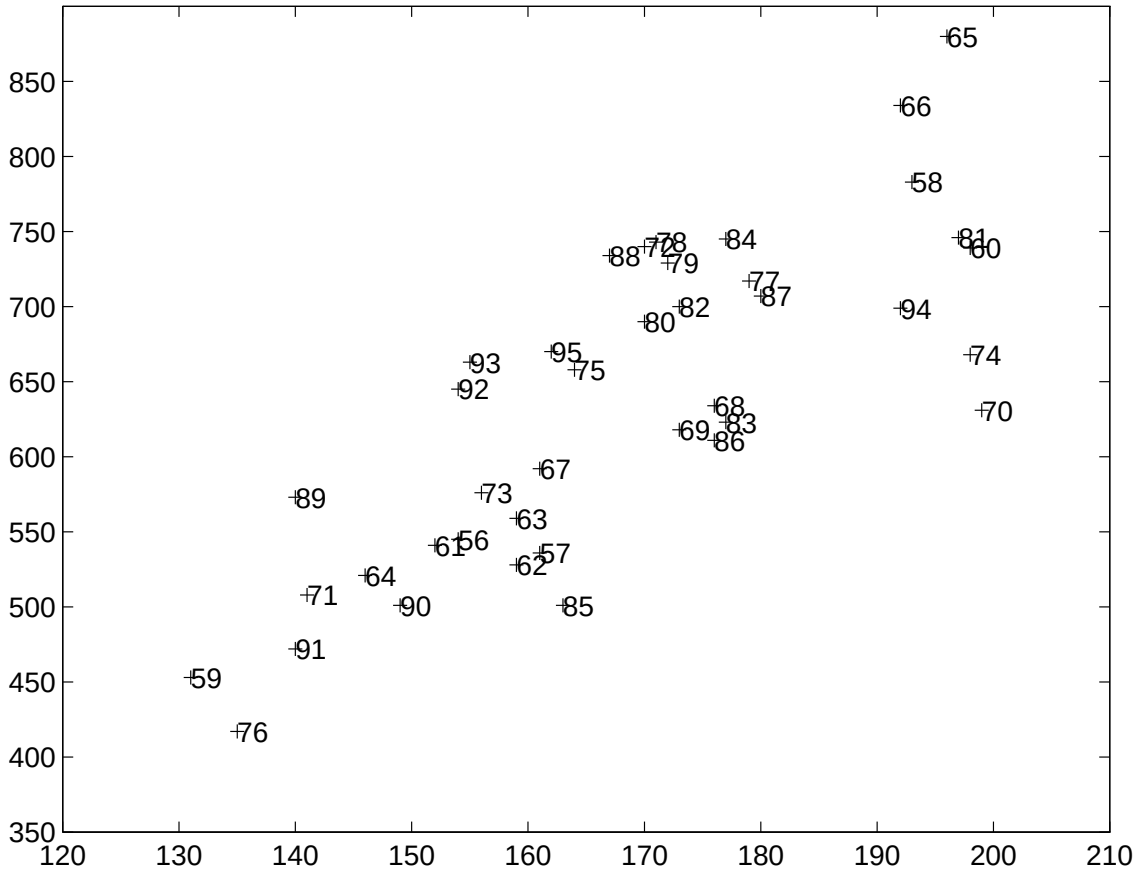

FIG. 2.3 – Représentation du nuage de points

Sur la figure 2.3, est représenté le nuage de points.
Les caractéristiques principales des données sont les suivantes :

- **Moyenne**
  $\bar{x} = 167.7$ et $\bar{y} = 635.75$
- **Ecart-type**
  $\sigma(x) = 18.73$ et $\sigma(y) = 107.6$
- **Corrélation**
  $r = 0.79$

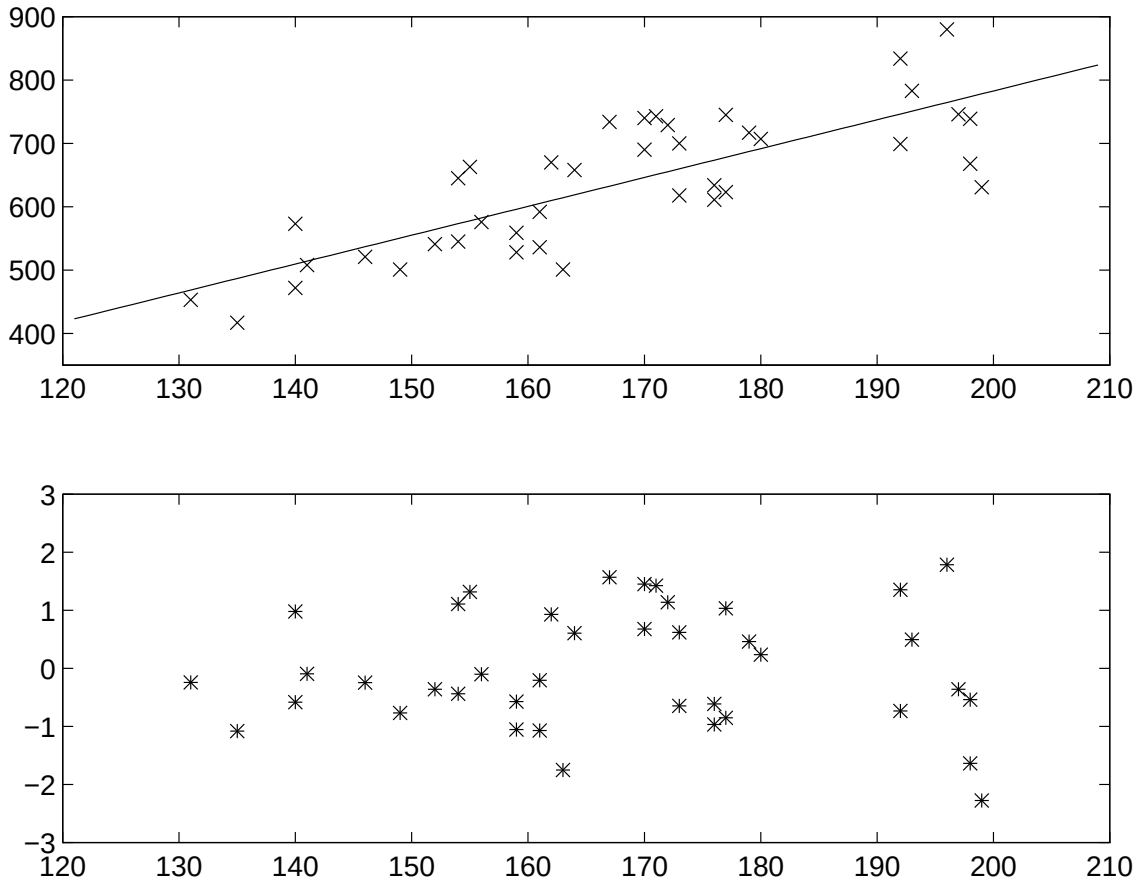

FIG. 2.4 – Représentation de la droite de régression et des résidus

Les coefficients de régression sont les suivants :
$\hat{a} = 4.55$ et $\hat{b} = -128$

On peut constater sur la figure 2.4 que les résidus sont correctement répartis. Toutefois leur variation est plus importante lorsque x est grand.
La conclusion que l'on peut en tirer est que la hauteur de pluie s'explique de façon satisfaisante par le nombre de jours de pluie mais que lorsque le nombre de jours de pluie est grand d'autres caractères interviennent aussi.

Chapitre 3

Analyse en composantes principales

Les **méthodes factorielles** permettent d'analyser un tableau de données quantitatives et d'en extraire l'information utile. On considérera essentiellement la méthode de base qu'est **l'analyse en composantes principales (A.C.P.)**, mais cette famille de méthodes en comprend d'autres comme **l'analyse factorielle des correspondances (A.F.C.)l'analyse discriminante** et **la régression linéaire multivariée**.

3.1 Nuage de points

On considère les données constituées de p caractères $(X_1, \ldots, X_p)$ mesurés sur n individus. On considérera que l'individu i est représenté par le point $M_i = (x_{i1}, \ldots, x_{ip})$ de l'espace euclidien $\mathcal{R}^p$. L'ensemble des individus $C = (M_1, \ldots, M_n)$ constitue un nuage de points.

3.1.1 Caractéristiques d'un nuage de points

Distance entre individus

La **distance euclidienne**, entre l'individu i, et l'individu i' est définie par :

$$\begin{aligned} d^2(M_i, M_{i'}) &= ||M_i M_{i'}||^2 && (3.1)\\ &= (\sum_{j=1}^{p}(x_{ij} - x_{i'j})^2) && (3.2)\\ &= (M_{i'} - M_i)^t\,(M_{i'} - M_i) && (3.3) \end{aligned}$$

Centre de gravité

Le **centre de gravité**, du nuage C, est le point G de $\mathcal{R}^p$ défini par :

$$G = (\bar{x}_1, \ldots, \bar{x}_p)$$

où

$$\bar{x}_j = \frac{1}{n}\sum_{i=1}^{n} x_{ij}$$

Matrice de covariance

La **covariance** entre les caractères X_j et $X_{j'}$, est définie par :

$$\Gamma(X_j, X_{j'}) = \frac{1}{n}\sum_{i=1}^{n}(x_{ij} - \bar{x}_j)(x_{ij'} - \bar{x}_{j'})$$

On peut remarquer que $\Gamma(X_j, X_j) = \sigma_j^2$, est le carré de l'écart type ou **variance** du caractère X_j.
On note $V = [\Gamma(X_j, X_{j'})]_{j,j'}$, la **matrice de covariance**.
Du point de vue matriciel si on note $Y = [\frac{x_{ij}-\bar{x}_j}{\sqrt{n}}]_{i,j}$ alors $V = Y'Y$ Donc cette matrice est symétrique.Elle est de plus semi-définie positive et on a la relation suivante :

$$tr\ V = \sum_{j=1}^{p}\sigma_j^2$$

Matrice de corrélation

La **corrélation** entre les caractères X_j et $X_{j'}$ est définie par :

$$\rho(X_j, X_{j'}) = \frac{\Gamma(X_j, X_{j'})}{\sigma_j\ \sigma_{j'}}$$

On note, $R = [\rho(X_j, X_{j'})]_{j,j'}$, la **matrice de corrélation**.
Cette matrice est semi-définie positive, sa diagonale ne comporte que des 1, et, en conséquence, $tr\ R = p$.

Normalisation des données

Dans la plupart des cas les données étudiées sont de nature et de dimension différentes. Il est nécessaire, avant de les analyser de les normaliser. Pour cela on remplace pour l'individu i, la mesure x_{ij} du caractère j par :

$$x'_{ij} = \frac{x_{ij} - \bar{x}_j}{\sigma_j}$$

Cela revient à remplacer le caractère X_j par le caractère X'_j, centré et de variance 1, défini par :

$$X'_j = \frac{X_j - \bar{x}_j\mathbf{1}_n}{\sigma_j}$$

Le nouveau nuage C', ainsi constitué est centré (i.e. le centre de gravité est à l'origine), et contenu dans la sphère de $\mathbb{R}^p$ centrée, de rayon $\sqrt{n}$. La matrice de covariance V', de C', est égale à R.
En outre si on note

$$Z = [\frac{x'_{ij}}{\sqrt{n}}]$$

la matrice des données normalisées, alors

$$R = Z'\ Z$$

3.1.2 Inertie d'un nuage de points

L'**inertie du nuage C par rapport au point P,** $I_P(C)$, est définie par :

$$I_P(C) = \frac{1}{n}\sum_{i=1}^{n} ||PM_i||^2$$

L'**inertie du nuage C,** $I(C)$, est définie par :

$$I(C) = I_G(C) = \frac{1}{n}\sum_{i=1}^{n}\sum_{j=1}^{p}(x_{ij} - \bar{x}_j)^2 = tr\ V$$

L'inertie est une caractéristique intrinsèque au nuage dans le sens où elle ne dépend pas du repère orthonormé choisi. La matrice V est la **matrice d'inertie** du nuage.
L'inertie par rapport au point P est liée à l'inertie par la **formule de Huyghens** :

$$I_P(C) = I(C) + ||GP||^2$$

Distance de Mahalanobis

On peut aussi utiliser des distances euclidiennes relative à une matrice symétrique de dimension p, A.

$$\begin{aligned} d_A^2(M_i, M_{i'}) &= (M_{i'} - M_i)^t\ A^{-1}\ (M_{i'} - M_i) && (3.4)\\ &= (\sum_{j=1}^{p}\sum_{j'=1}^{p} \alpha_{jj'}(x_{ij} - x_{i'j})(x_{ij'} - x_{i'j'}) && (3.5)\\ & && (3.6)\end{aligned}$$

où $[\alpha_{jj'}]_{j,j'}]$ sont les coefficients de A^{-1}

On définit la cohésion du nuage $\mathcal{C}$ relative à A par :

$$\mu_A(\mathcal{C}) = \frac{1}{2\ n^2}\sum_{i=1}^{n}\sum_{i'=1}^{n} d_A^2(M_i, M_{i'}) = Tr(A^{-1}\ V)$$

En particulier pour la distance canonique $A = I_p$ et $\mu_I(\mathcal{C}) = I(C)$.

De plus la matrice A qui minimise la dispersion $\mu_A(\mathcal{C})$ est la matrice de covariance V et $\mu_V(\mathcal{C}) = p$.

La distance ainsi définie est appelée **distance de Mahalanobis**.

3.1.3 Projection d'un nuage de points

On considère un nuage de points C dans $\mathcal{R}^p$ et W_q, un espace euclidien de dimension q, $q \leq p$, de base orthonormale $B_q = (u_1, \ldots, u_q)$.
On note Π_{W_q}, la projection orthogonale sur W_q, et $\Pi_{W_q}(C)$, le **nuage projeté**, c'est à dire le nuage de points de W_q qui résulte de la projection orthogonale de C sur W_q.

Inertie du nuage projeté

L'inertie du nuage projeté se calcule aisément et vaut :

$$I(\Pi_{W_q}(C)) = \sum_{k=1}^{q} u_k^t V u_k$$

Cette valeur est indépendante de la base orthonormale de W_q choisie.

Déformation d'un nuage par projection

On définit la **déformation** $\Delta(\Pi_{W_q}; C)$ du nuage C par la projection Π_{W_q} par :

$$\Delta(\Pi_{W_q}; C) = \frac{1}{n} \sum_{i=1}^{n} ||\Pi_{W_q}(M_i)M_i||^2$$

La déformation est liée à l'inertie par la formule suivante :

$$\Delta(\Pi_{W_q}; C) = I(C) + ||\Pi_{W_q}(G)G||^2 - I(\Pi_{W_q}(C))$$

3.2 Analyse en composantes principales (A.C.P.)

3.2.1 Le problème de l'A.C.P.

Le problème que l'on souhaite résoudre est de déterminer l'espace euclidien W_q^0, de dimension q fixée, qui minimise la déformation du nuage C par projection.

Problème d'optimisation à résoudre

Nous devons donc trouver une solution au problème :

$$(\mathbf{P}) \qquad \min_{W_q} \Delta(\Pi_{W_q}; C)$$

où W_q est un sous-espace euclidien de $\mathcal{R}^p$.
En utilisant la formule liant la déformation à l'inertie on constate que le problème est équivalent à :

$$(\mathbf{P}) \qquad \min_{W_q} I(\Pi_{W_q}(\bar{C}))$$

où $\bar{C}$ est le nuage centré associé à C.
Donc on cherche le sous-espace euclidien W_q^0, de $\mathcal{R}^p$, tel que la projection du nuage $\bar{C}$ sur W_0^q conserve le maximum d'inertie.
Si de plus on note $B_q = (u_1, \ldots, u_q)$, une base orthonormale de W_q, le problème **(P)** devient le problème d'optimisation avec contraintes suivant :

$$(\mathbf{P}) \quad \left| \begin{array}{l} \max \sum_{k=1}^{q} u_k^t V u_k \\ \\ u_k^t u_l = \delta_{kl} \qquad 1 \leq k \leq l \leq q \end{array} \right.$$

Le problème ainsi posé n'admet pas de solution unique. Nous sommes donc conduit à imposer en plus, la condition de récurrence sur la dimension, suivante :

$$W_{q-1}^0 \subset W_q^0 \qquad 1 \leq q \leq p$$

Si on note $B_q^0 = (u_1^0, \ldots, u_q^0)$, une base orthonormale de W_q^0, l'hypothèse précédente revient à imposer la condition supplémentaire :

$$B_q^0 = (B_{q-1}^0, u_q^0)$$

Résolution du problème

La proposition suivante donne la solution du problème d'optimisation :
Proposition : *Soient* $\lambda_1 \geq \lambda_2 \geq \ldots \geq \lambda_q \geq \ldots \lambda_p \geq 0$ *les valeurs propres de* V *prises dans l'ordre décroissant et* $B_p^0 = (u_1^0, \ldots, u_p^0)$ *une base orthonormale de vecteurs propres associée, alors,*

$$W_q^0 = vect(u_1^0, \ldots, u_q^0) = \arg\max_{W_q} I(\Pi_{W_q}(\bar{C})$$

et de plus,

$$I(\Pi_{W_q^0}(\bar{C}) = \sum_{k=1}^{q} \lambda_k$$

Le vecteur propre u_1 associé à la plus grande valeur propre λ_1, est la **composante principale** du nuage. C'est l'axe sur lequel l'inertie du nuage projeté est la plus grande.

3.2.2 Interprétation

Dispersion d'un caractère

Soit $a^t = (a_1, \ldots, a_p)$, un vecteur de $\mathcal{R}^p$.
On peut définir un nouveau caractère, noté $a(X)$, par combinaison linéaire des caractères $(X_1, \ldots, X_p)$ avec les coefficients $(a_1, \ldots, a_p)$. La valeur mesurée de ce nouveau caractère sur l'individu i est :

$$a(X)_i = \sum_{j=1}^{p} a_j x_{ij}$$

La moyenne de ce caractère est :

$$\bar{a}(X) = \frac{1}{n} \sum_{i=1}^{n} a_j \bar{x}_j$$

Soit $b(X)$, un autre caractère, construit de la même manière que $a(X)$, à l'aide du vecteur $b^t = (b_1, \ldots, b_p)$. La covariance entre les caractères $a(X)$ et $b(X)$ vaut :

$$\Gamma(a(X), b(X)) = \frac{1}{n} \sum_{i=1}^{n} (a(X)_i - \bar{a}(X))(b(X)_i - \bar{b}(X)) = a^t V b$$

On en déduit la variance du caractère $a(X)$:

$$\sigma^2(a(X)) = \Gamma(a(X), a(X)) = a^t V a$$

On définit la **dispersion** du caractère $a(X)$ par :

$$D(a(X)) = \frac{\sigma(a(X))}{||a||}$$

Si $a = u_1$, vecteur propre normé de V associé à la plus grande valeur propre λ_1, alors le caractère $u_1(X)$, est le caractère combinaison linéaire des caractères $(X_1, \ldots, X_p)$, ayant la dispersion maximale.

Fidélité du nuage projeté

On appelle **fidélité du nuage projeté,** $F_{W_q}(C)$, le rapport entre l'inertie du nuage initial et l'inertie du nuage projeté :

$$F_{W_q}(C) = \frac{I(\Pi_{W_q}(C))}{I(C)}$$

La représentation est d'autant plus fidèle que cette quantité est proche de 1. Lorsque la dimension q de la projection est fixée la représentation la plus fidèle est celle obtenue par l'A.C.P.
Dans le cas particulier de l'analyse en composantes normées on a :

$$F_{W_q^0}(C) = \frac{1}{p}\sum_{k=1}^{q} \lambda_k$$

Lorsque la dimension q n'est pas fixée, on peut utiliser la fidélité comme critère de choix

Représentation d'une A.C.P

On peut représenter les données projetées de 2 manières :

- a - Représentation du nuage projeté

On représente dans un ou plusieurs plans la projection de chacun des individus : $(u_1, u_2), (u_1, u_3), (u_2, u_3), \ldots$

- b - Représentation des corrélations entre caractères

On peut visualiser les corrélations entre les nouveaux caractères $(u_1(X), \ldots, u_p(X))$ et les caractères initiaux $(X_1, \ldots, X_p)$.
Dans le cas de l'analyse en composantes normées on a :

$$\rho(X_j, u_k(X)) = \frac{e_j^t R u_k}{\sigma(X_j)\ \sigma(u_k(X))} = \sqrt{\lambda_k}\ u_{kj}$$

où $u_{kj} = u_k^t e_j$ représente la $k^{ième}$ coordonnée dans la nouvelle base du $j^{ième}$ vecteur de la base initiale.
Dans le plan (u_1, u_2), on pourra représenter la position du caractère X_j par le point de coordonnées $(\sqrt{\lambda_1}\ u_{1j}, \sqrt{\lambda_2}\ u_{2j})$.
Cette représentation permettra de se rendre compte de l'information apportée par chacun des caractères et en particulier de la redondance de cette information.

3.3 Application : Climat en France

3.3.1 Données

Le tableau suivant comprend 6 variables climatiques pour 24 villes de France.

Numéro	Ville	X_1	X_2	X_3	X_4	X_5	X_6
1	Ajaccio	95	653	2811	7.7	22.0	38
2	Biarritz	177	1474	1921	7.6	19.7	43
3	Bordeaux	162	947	2076	5.6	20.9	6
4	Brest	201	1157	1757	6.1	15.6	100
5	Clermont	132	571	1899	2.6	19.4	358
6	Dijon	147	734	1934	1.3	19.6	245
7	Embrun	107	698	2604	0.5	18.9	871
8	Grenoble	144	1005	2100	1.5	20.1	214
9	Lille	171	612	1641	2.4	17.1	21
10	Limoges	165	910	1853	3.1	18.4	294
11	Lyon	145	828	2036	2.1	20.7	173
12	Marseille	76	533	2866	5.5	23.3	48
13	Montpellier	88	736	2709	5.6	22.7	27
14	Nancy	161	731	1633	0.8	18.3	212
15	Nantes	168	819	1901	5.0	18.8	8
16	Nice	86	868	2779	7.5	22.7	16
17	Orleans	156	621	1799	2.7	18.4	116
18	Paris	162	624	1814	3.4	19.1	45
19	Perpignan	85	628	2603	7.5	23.8	30
20	Poitiers	155	702	2024	3.8	18.9	75
21	Rennes	168	634	1835	4.8	17.9	30
22	Rouen	167	716	1694	3.4	17.6	10
23	Strasbourg	158	719	1696	0.4	19.0	143
24	Toulouse	137	656	2081	4.7	20.9	146

- X_1 :Nombre de jours de précipitations par an
- X_2 :Hauteur moyenne des précipitations par an
- X_3 :Durée annuelle d' ensoleillement en heures
- X_4 :Température moyenne du mois de Janvier en degré Celsius
- X_5 :Température moyenne du mois de Juillet en degré Celsius
- X_6 :Altitude en metres

3.3.2 Etudes préliminaires

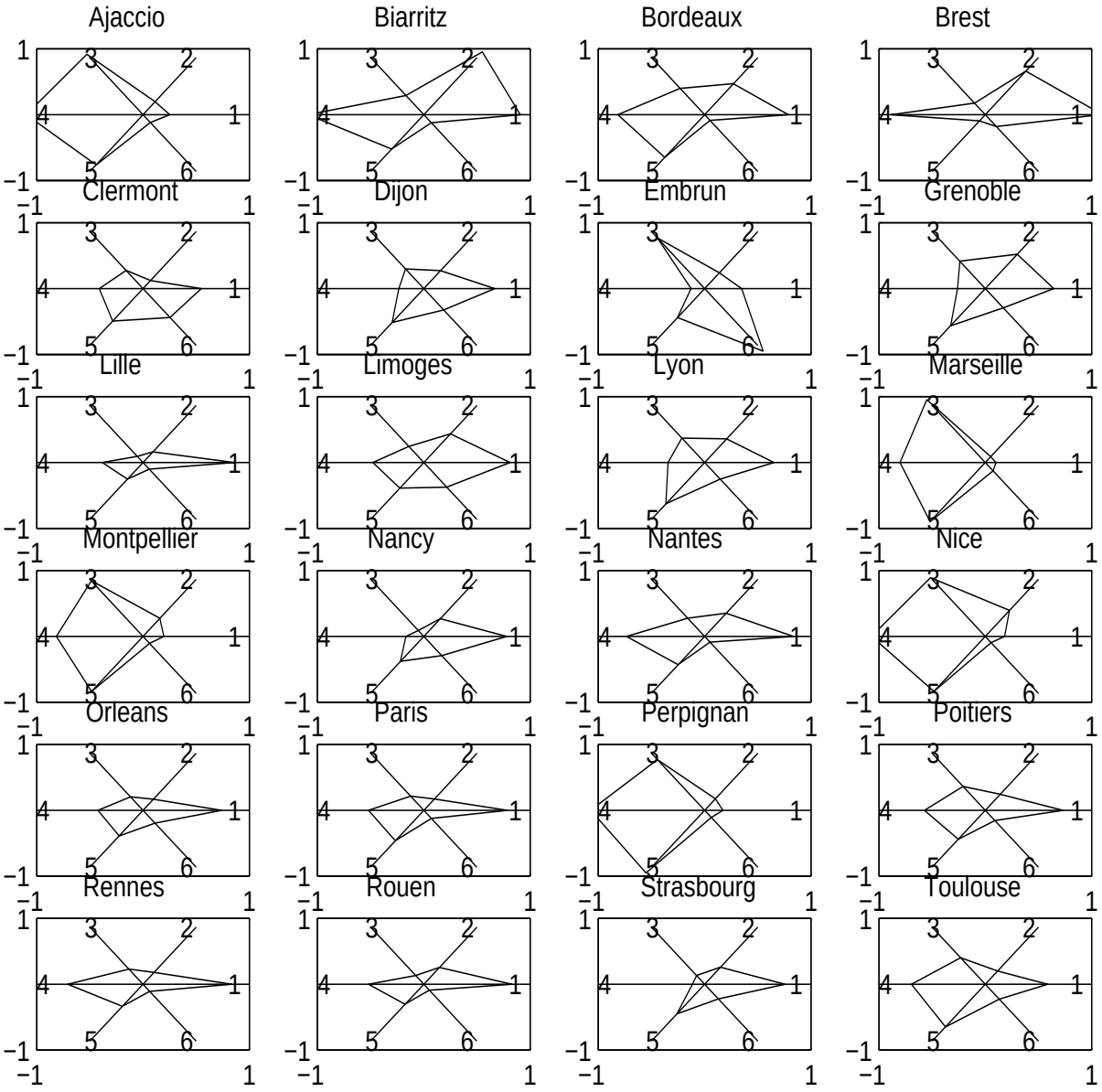

FIG. 3.1 – Diagrammes en étoiles

Statistique	X_1	X_2	X_3	X_4	X_5	X_6
Moyenne	142.2	774	2086.1	4	19.7	136.2
Médiane	155.5	717.5	1927.5	3.6	19.2	61.5
Ecart-type	34.3	210	404	2.33	2.05	185
Minimum	76	533	1633	0.4	15.6	6
Maximum	201	1474	2866	7.7	23.8	871
1^{er} quartile	119.5	631	1806.5	2.2	18.4	28.5
$3^{ème}$ quartile	166	848	2351.5	5.6	20.9	192.5

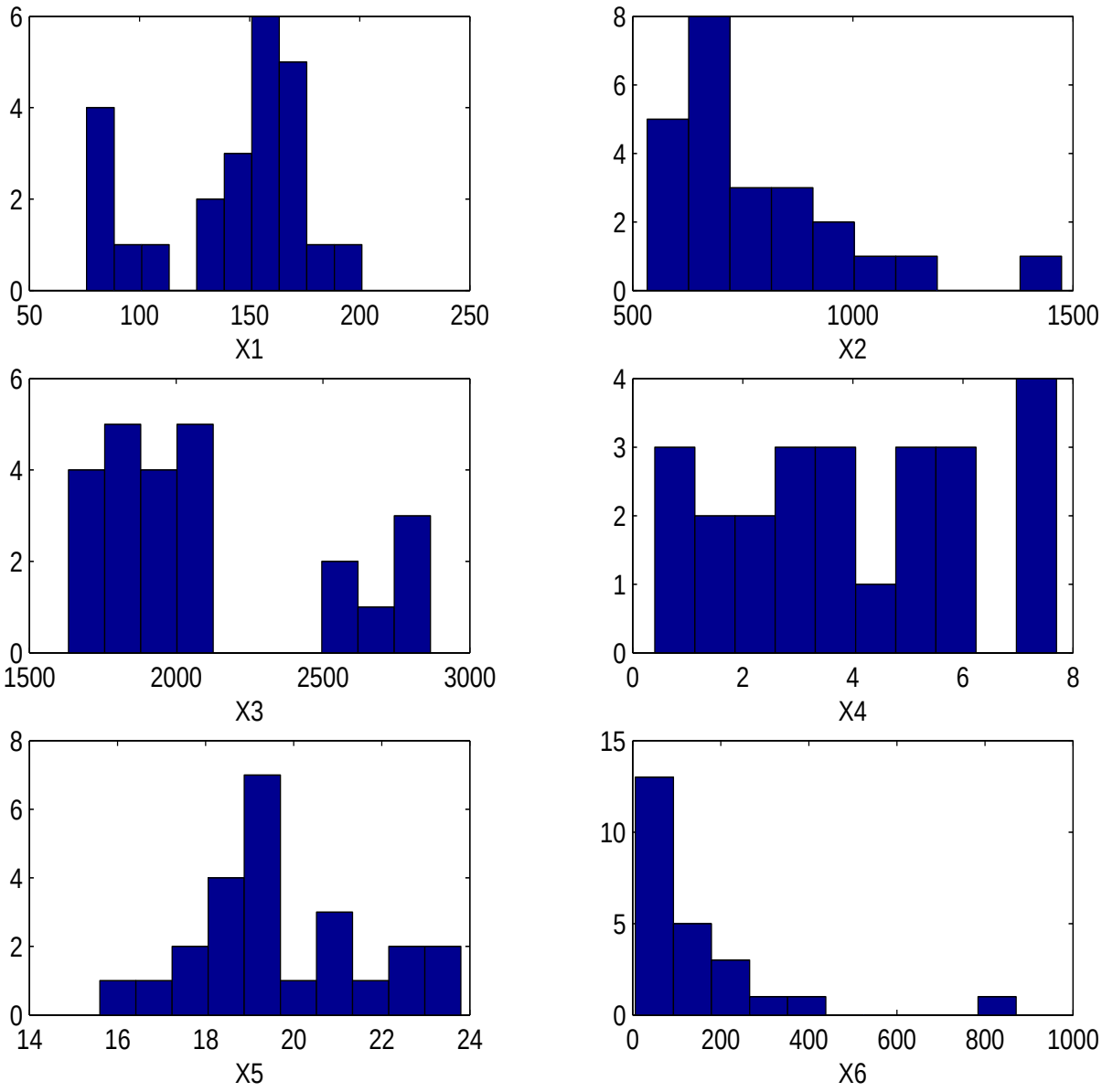

FIG. 3.2 – Histogrammes

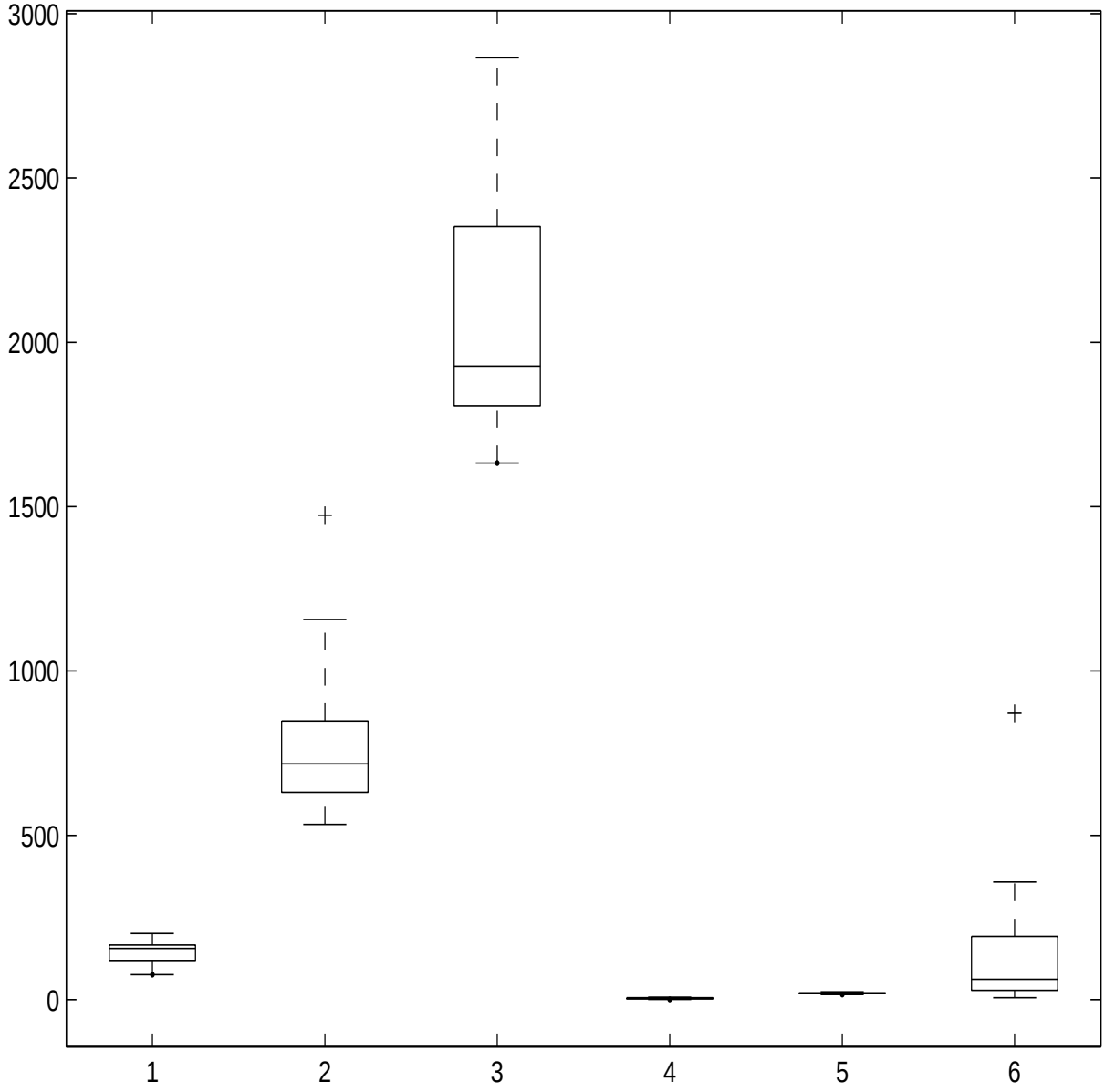

FIG. 3.3 – Boites à moustaches

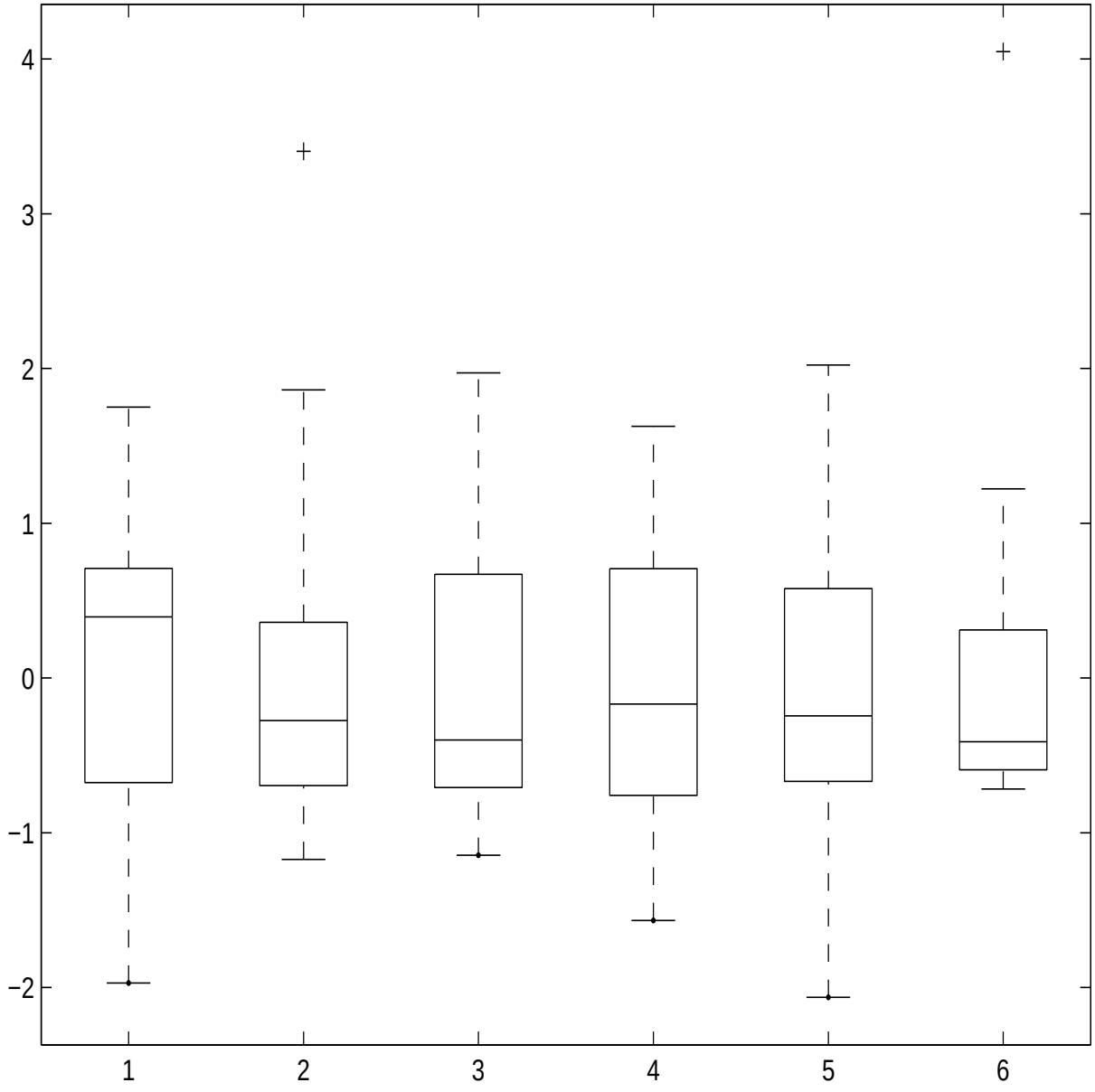

FIG. 3.4 – Boites à moustaches : données normalisées

3.3.3 Résolution de l'ACP

Matrice de corrélation

$$R = \begin{pmatrix} 1.0000 & 0.3331 & -0.9361 & -0.4402 & -0.8849 & -0.0356 \\ 0.3331 & 1.0000 & -0.0880 & 0.2146 & -0.1175 & 0.0056 \\ -0.9361 & -0.0880 & 1.0000 & 0.6005 & 0.8622 & 0.0109 \\ -0.4402 & 0.2146 & 0.6005 & 1.0000 & 0.5354 & -0.5716 \\ -0.8849 & -0.1175 & 0.8622 & 0.5354 & 1.0000 & -0.1621 \\ -0.0356 & 0.0056 & 0.0109 & -0.5716 & -0.1621 & 1.0000 \end{pmatrix}$$

Valeurs propres

$$\begin{aligned} \lambda_1 &= 3.2257 \\ \lambda_2 &= 1.4579 \\ \lambda_3 &= 0.9779 \\ \lambda_4 &= 0.2473 \\ \lambda_5 &= 0.0705 \\ \lambda_6 &= 0.0207 \end{aligned}$$

Vecteurs propres

$$U = \begin{pmatrix} 0.5187 & -0.2791 & -0.0010 & -0.0206 & -0.3213 & .7413 \\ 0.0882 & -0.4772 & 0.7981 & 0.2622 & 0.1900 & -0.1507 \\ -0.5291 & 0.1235 & 0.2014 & -0.2049 & 0.4855 & .6217 \\ -0.3950 & -0.5157 & 0.0178 & -0.6228 & -0.4197 & -0.1170 \\ -0.5202 & 0.0650 & 0.0557 & 0.6213 & -0.5557 & .1649 \\ 0.1287 & 0.6395 & 0.5649 & -0.3391 & -0.3742 & -0.0201 \end{pmatrix}$$

3.3.4 Représentations de l'ACP

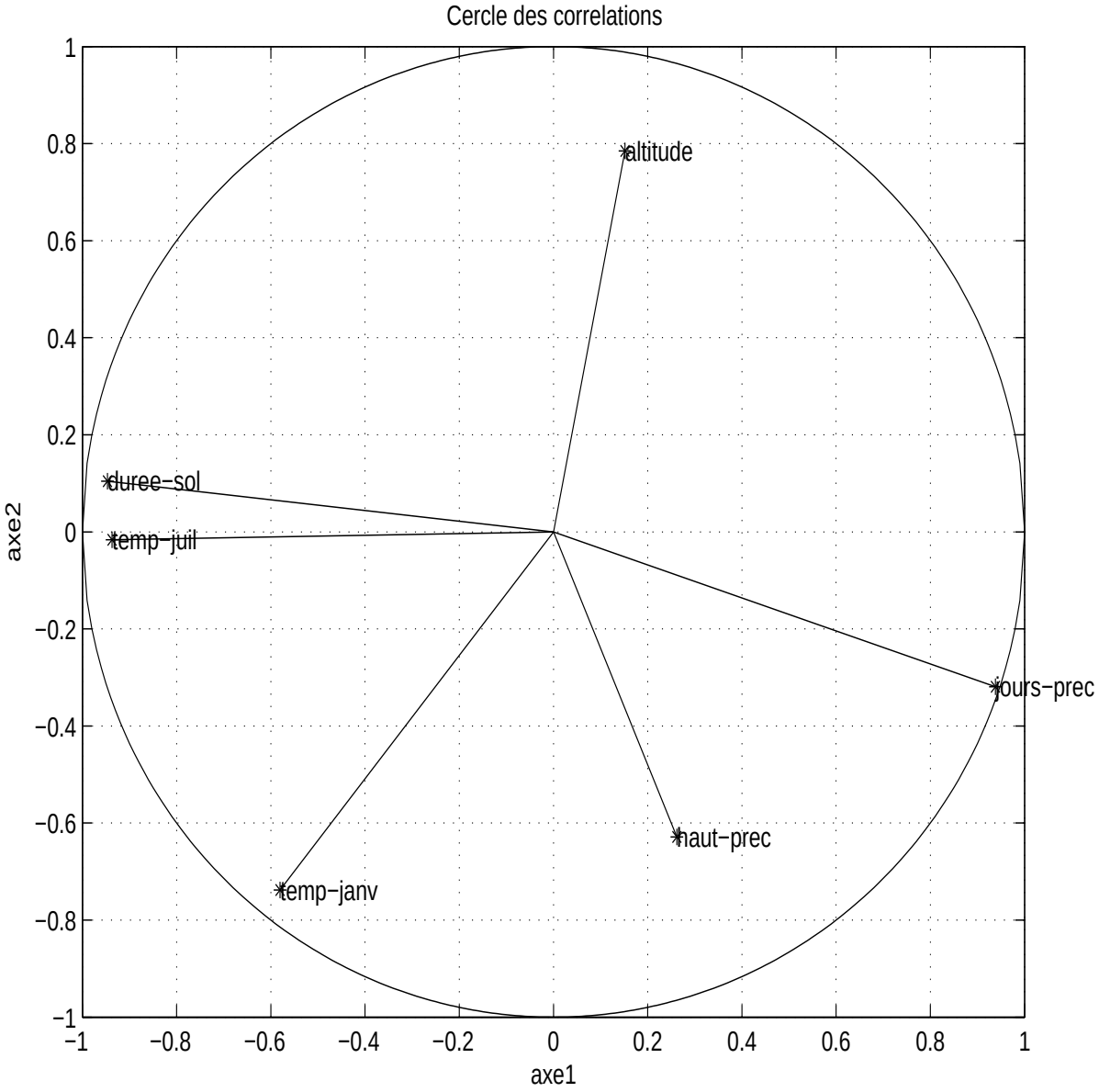

FIG. 3.5 – Cercle des corrélations : Plan (u1,u2)

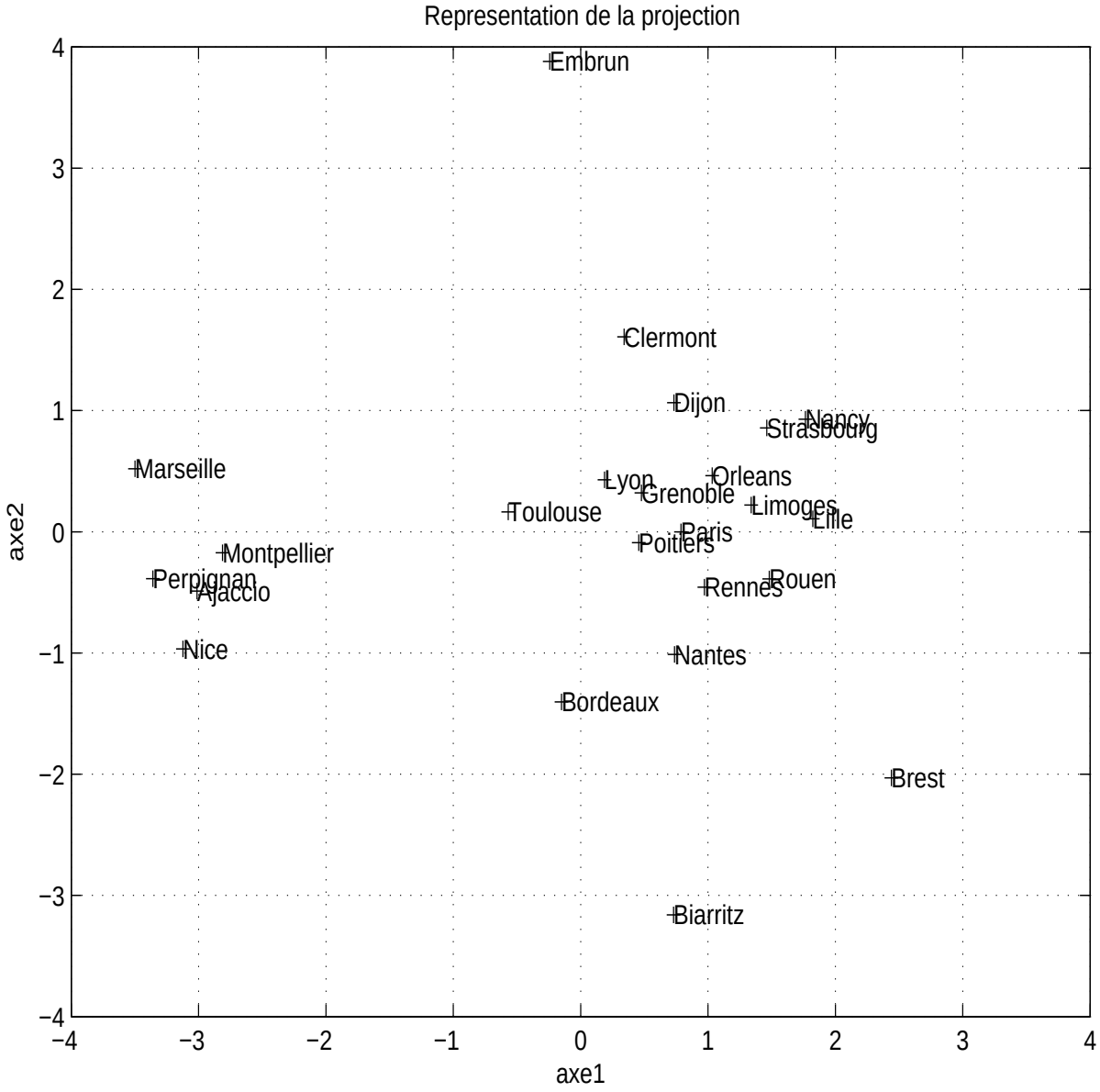

FIG. 3.6 – Projection du nuage : Plan (u1,u2)

3.3.5 Analyse des résultats

La fidélité de la représentation dans le plan principal (u_1, u_2) est de 78% . Avec les deux axes principaux on a donc une vision quasi complète des données.
On constate sur les cercles des corrélations que l'axe principal oppose X_1 à X_3 et X_5 c'est à dire le nombre de jours de pluie la chaleur en Juillet et à l'ensoleillement. C'est donc sur cet axe que les villes très ensoleillées et chaudes en été notamment les villes de climat méditerranéen (Ajaccio, Marseille, Montpellier, Perpignan et Nice) seront séparées des autres villes du nord et de l'ouest de la France au climat océanique plus frais et humide.
Le deuxième axe oppose X_6 à X_2 et X_4, donc l'altitude à la hauteur de pluie et une température douce en Janvier. c'est donc un axe qui va permettre de distinguer les villes au climat montagnard rude en hiver comme Embrun (Hautes Alpes) des villes du bord de l'atlantique pluvieuses, mais douces en hiver comme Brest ou Biarritz.
La projection des villes sur le plan principal reflète les considérations précédentes : on peut constater que les villes du pourtour méditerranéen ont un climat très particulier par rapport aux autres villes et qu'en revanche les différences entre des villes comme Toulouse et Paris ne sont pas aussi importantes que ce que l'on imagine en général.

Chapitre 4

Classification

4.1 Introduction

4.1.1 Généralités

La classification consiste à regrouper (ou aggréger) des éléments d'un nuage de points en plusieurs classes.
Souvent le problème n'est pas supervisé : le nombre de classes n'est pas connu ainsi que l'appartenance des individus à une classe.
Les méthodes sont variées : partitionnement, hiérarchie, ou à base de réseaux de neurones.

Le problème à résoudre est le suivant :
On dispose d'un nuage de points, $\mathcal{C} = (M_1, \ldots, M_n)$ appartenant à $\mathcal{R}^p$ et on souhaite le partitionner en k **classes** (i.e des sous-ensembles disjoints recouvrant $\mathcal{C}$).

Les meťhodes utilisées comportent toutes 2 caractéristiques :

1. **Définition d'un critère de classification**
   Il est souvent basé sur des distances entre les individus. On met dans la même classe les individus qui se ressemblent donc qui sont proches au sens d'une distance bien choisie. Cela se traduit donc par un problème de minimisation.

2. **Construction d'un algorithme d'affectation**
   On doit affecter les individus à chaque classe.On utilise des algorithmes de natures différentes : algorithmes itératifs de descente, construction d'un arbre, cartes de Kohonen, programmation dynamique.

4.1.2 Critère fondamental

Le critère le plus utilisé en classification est le critère de la **somme des inerties**.
On considère le nuage de points $\mathcal{C} = (M_1, \ldots, M_n)$.
On note $x_i = (x_{i1}, \ldots, x_{ip})$ les coordonnées de M_i dans $\mathcal{R}^p$, $\mathcal{P} = (A_1, \ldots, A_k)$ une partition en k classes de $\mathcal{C}$ et n_l le cardinal de A_l.

On note G_l, le **centre de gravité**, de la classe A_l, donc :

$$\begin{aligned} G_l &= (\bar{x}_{l1}, \ldots, \bar{x}_{lp}) \\ &= (\frac{1}{n_1} \sum_{\{i, M_i \in A_1\}} x_{i1}, \ldots, \frac{1}{n_k} \sum_{\{i, M_i \in A_k\}} x_{ip}) \end{aligned}$$

L'inertie de la classe A_l est alors définie par :

$$I_l = I(A_l) = \frac{1}{n_l} \sum_{\{i, M_i \in A_l\}} \sum_{j=1}^{p} (x_{ij} - \bar{x}_{lj})^2$$

Le critère fondamental pour la partition $\mathcal{P}$ est défini par :

$$\begin{aligned} W(\mathcal{P}) &= \sum_{l=1}^{k} n_l I_l \\ &= \sum_{l=1}^{k} \sum_{\{i, M_i \in A_l\}} d^2(G_l, M_i) \end{aligned}$$

Ce critère est utilisé pour construire les algorithmes de partitionnement dont le but est sinon de minimiser ce critère, au moins de le faire décroître. Dans les méthodes hiérarchiques, d'autres critères sont utilisés.

4.1.3 Complexité du problème

Le nombre P_n^k de partitions en k classes d'un ensemble E de cardinal n est connu uniquement par l'équation de récurrence suivante :

$$P_n^k = P_{n-1}^{k-1} + k\, P_{n-1}^k$$

En effet soit une partition P en k classes de E et a un él'ement de E.

- Soit a est seul dans sa classe et il faut partitionner les n-1 éléments de $E - \{a\}$ en $k-1$ partitions. Il y a P_{n-1}^{k-1} façons de le faire.
- Soit a n'est pas seul dans sa classe et il faut partitionner $E - \{a\}$ en k classes et ajouter a à l'une des classes. Donc il y a P_{n-1}^k partitions possibles et pour chaque partition k choix possibles pour affecter a.

Pour certaines valeurs de k on peut calculer directement P_n^k.

$$\begin{aligned} P_n^1 &= 1 \\ P_n^n &= 1 \\ P_n^{n-1} &= \frac{n(n-1)}{2} \\ P_n^2 &= 2^{n-1} - 1 \\ P_n^3 &= \frac{1}{2}\,(3^{n-1} - 2^n + 1) \\ P_n^k &= 0 \text{ si } k > n \end{aligned}$$

Soit P_n le nombre de partitions d'un ensemble E de cardinal n.
Par convention on posera $P_0 = 1$

On a alors les résultats suivants :

$$\begin{aligned} P_n &= \sum_{k=1}^{n} P_n^k \\ &= \sum_{j=0}^{n-1} C_{n-1}^j P_j \end{aligned}$$

On peut disposer ces résultats dans un tableau analogue au triangle de Pascal.

P_n	k / n	1	2	3	4	5	6	7	8	...
1	1	1	0	0	0	0	0	0	...	
2	2	1	1	0	0	0	0	0	...	
5	3	1	3	1	0	0	0	0	...	
15	4	1	7	6	1	0	0	0	...	
52	5	1	15	25	20	1	0	0	...	
203	6	1	31	90	65	15	1	0	0	...
876	7	1	63	301	350	140	20	1	0	...
⋮	8	1	127	966	⋮	⋮	⋮	⋮	⋮	

On constate une augmentation exponentielle du nombre de partition d'un ensemble. Par exemple si pour trouver la meilleure partition d'un ensemble en 2 parties pour un critère fixé, il fallait examiner tous les cas possibles, il faudrait comparer $2^{n-1} - 1$ valeurs du critère, soit de l'ordre de 10^{30} pour $n = 100$.
Pour partitionner en 3 parties le nombre de cas est de $\frac{1}{2}\ (3^{n-1} - 2^n + 1)$ soit de l'ordre de 10^{46} pour $n = 100$.

Ces quelques considérations montrent l'ampleur de la tâche à réaliser (en algorithmique ce problème fait partie de la classe des problèmes NP-difficile).
Dans ces conditions la plupart des méthodes n'ont pas pour but de trouver **la** partition optimale qui minimise W (pour peu qu'il n'en existe qu'une seule mais d'obtenir une solution raisonnable. Cela est d'autant plus vrai que certains problèmes actuels peuvent nécessiter la classification de 10 000 individus en plusieurs dizaines de classes.

4.2 Méthodes de partitionnement

On regroupe dans ces méthodes une famille d'algorithmes itératifs dont le but est de faire décroitre le critère W.

4.2.1 Algorithme d'aggrégation autour des centres mobiles (A.C.M)

Dans cette méthode le nombre de classes k est fixé a priori.

Déroulement de l'algorithme (A.C.M)

1. **Initialisation**
   On détermine k points $\mathcal{G}_0 = (G_1^0, \ldots, G_k^0)$ formant les centres des classes initiaux.
   On peut les tirer au hasard parmi les n points de C, ou ils peuvent être déterminés par tout autre considération par exemple provenir d'un autre algorithme appliqué préalablement.
2. **Etape 1**
   On détermine ensuite la partition $\mathcal{P}_1 = (A_1^1, \ldots, A_k^1)$ en affectant pour $i = 1, \ldots, n$ le point M_i à la classe dont le centre est le plus proche :

$$\left| \begin{array}{l} M_i \longrightarrow A_{\delta(i)}^1 \\ \delta(i) = \text{argmin}_{l=1,\ldots,k}\, d(M_i, G_l^0) \end{array} \right.$$

3. **Etape m**
   A l'étape $m-1$ on a obtenu la partition $\mathcal{P}_{m-1} = (A_1^{m-1}, \ldots, A_k^{m-1})$.
   - Les nouveaux centres $\mathcal{G}_{m-1} = (G_1^{m-1}, \ldots, G_k^{m-1})$ sont les centres de gravités des classes de la partition $\mathcal{P}_{m-1}$:

$$G_l^{m-1} = \frac{1}{n} \sum_{\{i, M_i \in A_l^{m-1}\}} x_i$$

   - On détermine alors la partition $\mathcal{P}_m = (A_1^m, \ldots, A_k^m)$ en affectant pour $i = 1, \ldots, n$ le point M_i à la classe dont le centre est le plus proche :

$$\left| \begin{array}{l} M_i \longrightarrow A_{\delta(i)}^m \\ \delta(i) = \text{argmin}_{l=1,\ldots,k}\, d(M_i, G_l^{m-1}) \end{array} \right.$$

4. **Fin**
   On arrête à l'étape M, si $\mathcal{P}_M = \mathcal{P}_{M-1}$ On a alors aussi $\mathcal{G}_M = \mathcal{G}_{M-1}$.

Justifications de l'algorithme

On peut justifier l'utilisation de l'algorithme par la proposition suivante :
Proposition : *L'algorithme d'aggrégation autour des centres mobiles fait décroître W.*
Preuve :

On note :

$$I_l^m = I(A_l^m)$$

et

$$\begin{aligned} I_P(A_l^m) &= \frac{1}{n} \sum_{\{i, M_i \in A_l^m\}} d^2(P, M_i) \\ &= I_l^m + d^2(G_l^m, P) \text{ (formule de Huyghens)} \end{aligned}$$

D'où

$$\begin{aligned} W(\mathcal{P}_m) &= \sum_{l=1}^{k} n_l^m I_l^m \\ &= \sum_{l=1}^{k} n_l^m (I_{G_l^{m-1}}(A_l^m) - d^2(G_l^m, G_l^m)) \\ &\leq \sum_{l=1}^{k} n_l^m I_{G_l^{m-1}}(A_l^m) \end{aligned}$$

Or

$$\begin{aligned} \sum_{l=1}^{k} n_l^m I_{G_l^{m-1}}(A_l^m) &= \sum_{l=1}^{k} \sum_{\{i, M_i \in A_l^m\}} d^2(M_i, G_l^{m-1}) \\ &\leq \sum_{l=1}^{k} \sum_{\{i, M_i \in A_l^{m-1}\}} d^2(M_i, G_l^{m-1}) \end{aligned}$$

à cause de la règle de détermination des partitions.
Comme

$$\sum_{l=1}^{k} \sum_{\{i, M_i \in A_l^{m-1}\}} d^2(M_i, G_l^{m-1}) = \sum_{l=1}^{k} n_l^{m-1} I_l^{m-1} = W(\mathcal{P}_{m-1})$$

on a par conséquent

$$W(\mathcal{P}_m) \leq W(\mathcal{P}_{m-1})$$

La suite $W(\mathcal{P}_m)$ est une suite réelle positive décroissante prenant un nombre fini de valeur donc elle converge.

Remarques

1. Cet algorithme a le mérite par rapport aux algorithmes suivants d'être rapide puisque chaque étape est constituée d'opérations successives de complexités O(n).
   L'expérience montre que le nombre d'étapes se compte en quelques dizaines, un nombre en tout cas très inférieur à n.
   On peut donc considérer qu'il est de l'ordre de O(n).

2. La suite $W(\mathcal{P}_m)$ converge mais pas nécessairement vers le minimum du critère. En particulier l'algorithme est sensible au choix des centres initiaux.
3. On peut améliorer l'algorithme en mettant en évidence des groupements stables. Cela consiste à utiliser l'algorithme plusieurs fois (3 ou 4 par exemple) avec des centres initiaux différents et de regrouper ensemble les points ayant été toujours classés dans la même classe.
   Cette technique met en évidence des groupements de points (les groupements stables) à un niveau plus profond.

4. Un autre inconvénient de cet algorithme est qu'il a tendance à regrouper les points en classes sphériques et de même taille. On peut en tenir compte en modifiant le critère et l'algorithme : une famille d'algorithme issu de l'algorithme A.C.M **les nuées dynamiques** a été étudiée par *E.Diday* [3]

4.2.2 Algorithme des k-means

Dans cet algorithme on opère de façon itérative sur les points : chaque étape consiste à modifier la composition des classes en transférant un point d'une classe à une autre.

Remarque préliminaire

Supposons que l'on transfère le point M de coordonnées x de la classe A_l à la classe $A_{l'}$.On supposera que $n_l \geq 2$. On dispose donc d'une partition initiale, $\mathcal{P} = (A_1, \ldots, A_l, \ldots, A_{l'}, \ldots, A_k)$ et d'une nouvelle partition, $\mathcal{P}' = (A_1, \ldots, A_l \backslash \{M\}, \ldots, A_{l'} \cup \{M\}, \ldots, A_k)$. Calculons la variation du critère.

On note G_A, le centre de gravité de A, de coordonnées $\bar{x}_A$.
On a alors :

$$\begin{aligned} \bar{x}_{A_l \backslash \{M\}} &= \frac{n_l \ \bar{x}_{A_l} - x}{n_l - 1} \\ \bar{x}_{A_{l'} \cup \{M\}} &= \frac{n_l \ \bar{x}_{A_{l'}} + x}{n_l + 1} \end{aligned}$$

et

$$\begin{aligned} d^2(G_{A_l}, G_{A_l \backslash \{M\}}) &= \frac{1}{(n_l - 1)^2} \ d^2(G_{A_l}, M) \\ d^2(G_{A_{l'}}, G_{A_{l'} \cup \{M\}}) &= \frac{1}{(n_{l'} + 1)^2} \ d^2(G_{A_{l'}}, M) \end{aligned}$$

Par conséquent

$$\begin{aligned} (n_l - 1) \ I(A_l \backslash \{M\}) &= n_l \ I(A_l) - 2\frac{n_l}{n_l - 1} \ d^2(M, G_{A_l}) \\ (n_{l'} + 1) \ I(A_{l'} \cup \{M\}) &= n_{l'} \ I(A_{l'}) + 2\frac{n_{l'}}{n_{l'} + 1} \ d^2(M, G_{A_{l'}}) \end{aligned}$$

On en déduit la variation du critère :

$$W(\mathcal{P}') - W(\mathcal{P}) = 2(\frac{n_{l'}}{n_{l'} + 1} \ d^2(M, G_{A_{l'}}) - \frac{n_l}{n_l - 1} \ d^2(M, G_{A_l}))$$

Ce résultat induit le déroulement de l'algorithme.

Déroulement de l'algorithme (K-MEANS)

1. **Initialisation :Etape 1**
   On détermine une partition initiale $\mathcal{P}_1 = (A_1^1, \ldots, A_k^1)$ (par exemple comme dans l'algorithme A.C.M)
   On calcule les centres associés $\mathcal{G}_1 = (G_1^1, \ldots, G_k^1)$.

2. **Etape m**
   A l'étape $m-1$ on a obtenu la partition $\mathcal{P}_{m-1} = (A_1^{m-1}, \ldots, A_k^{m-1})$ et les centres $\mathcal{G}_{m-1} = (G_1^{m-1}, \ldots, G_k^{m-1})$.
   - On tire un point $M_{i(m)}$ dans l'ordre des indices.
   - On détermine l'indice $l = l_{m-1}$ de la classe actuelle de M_m.
   - On calcule la classe candidate au transfert

     Si $n_l = 1$:
     On conserve la même partition : $\mathcal{P}_m = \mathcal{P}_{m-1}$

     Si $n_l > 1$:
     On détermine l'indice l'_m vérifiant

     $$l'_m = \underset{l'}{\operatorname{argmin}} \left(\min_{l' \neq l, l'=1,\ldots,k} \left(\frac{n_{l'}}{n_{l'}+1} \, d^2(M_{i(m)}, G_{l'}^{m-1}), \frac{n_l}{n_l - 1} \, d^2(M_{i(m)}, G_l^{m-1}) \right) \right.$$

   - On détermine la nouvelle partition

     Si $l'_m = l$:
     On conserve la même partition : $\mathcal{P}_m = \mathcal{P}_{m-1}$

     Si $l'_m \neq l$:
     Alors $\mathcal{P}_m = (A_1, \ldots, A_l \backslash \{M_{i(m)}\}, \ldots, A_{l'} \cup \{M_{i(m)}\}, \ldots, A_k)$
3. **Fin**
   L'algorithme s'arrête lorsque aucun transfert ne s'opère pour aucun point.

Remarques

Les remarques sur le choix initial de la partition s'appliquent encore pour cet algorithme. Néanmoins il est moins facile d'emploi que l'algorithme A.C.M pour les grands ensembles car à chaque étape on doit réactualiser les centres. C'est pourquoi il est utile de combiner les 2 algorithmes d'abord l'algorithme ACM pour fixer rapidement des sous-ensembles et ensuite les k-means pour regarder cas par cas ce qui est meilleur pour le critère de la somme des inerties.

4.2.3 Application

Les 2 algorithmes ont été appliquée de manière combinée aux données du Chapitre 3. Le tableau suivant donne les résultats que l'on peut aussi représenter dans le plan principal.

classe	numero	individu	classe	numero	individu
1	2	Angers	1	27	Rennes
1	3	Angouleme	1	28	Rouen
1	4	Besancon	1	29	St-Quentin
1	6	Bordeaux	1	30	Strasbourg
1	8	Caen	1	32	Toulouse
1	9	Clermont	1	33	Tours
1	10	Dijon	1	34	Vichy
1	12	Grenoble	2	5	Biarritz
1	13	Lille	2	7	Brest
1	14	Limoges	3	1	Ajaccio
1	15	Lyon	3	16	Marseille
1	18	Nancy	3	17	Montpellier
1	19	Nantes	3	20	Nice
1	22	Orleans	3	21	Nimes
1	23	Paris	3	24	Perpignan
1	25	Poitiers	3	31	Toulon
1	26	Reims	4	11	Embrun

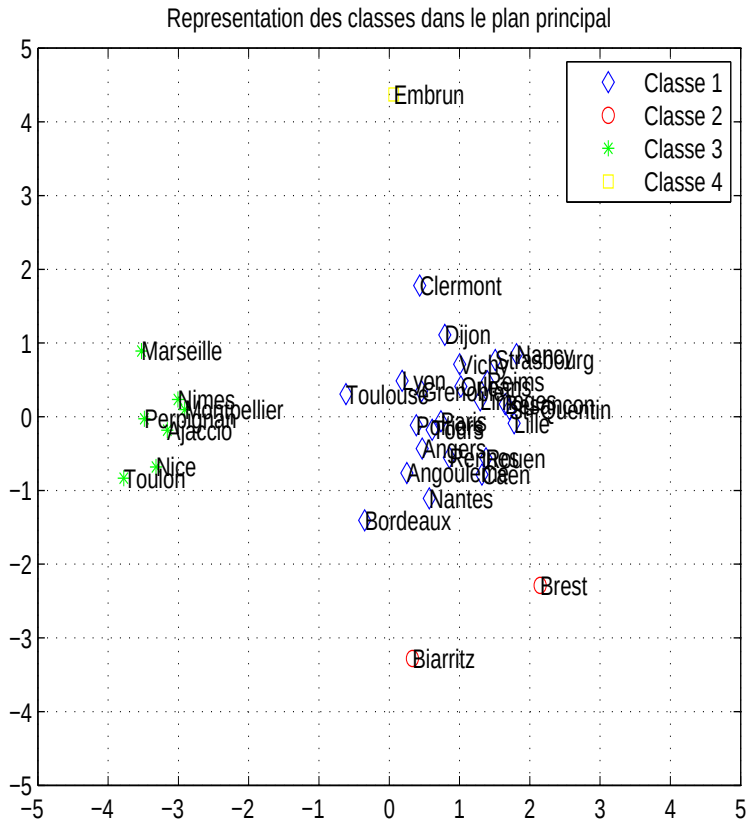

FIG. 4.1 – Classification en 4 classes par la méthode des kmeans

4.3 Méthodes hiérarchiques

Ces méthodes ne sont pas des méthodes de regroupement des points en k classes comme les précédentes, mais elles permettent d'obtenir toute une famille de partitions dont le nombre de classes varie de 1 à n appelée **hiérarchie**.

4.3.1 Hiérarchie

Définition

Soit $\mathcal{C}$ un ensemble de points de cardinal n, on note $P(\mathcal{C})$, l'ensemble des parties de $\mathcal{C}$. Dans le contexte de l'analyse des données on dira qu'un élément A de $P(\mathcal{C})$ est un **groupement de points**.
$\mathcal{H}$ un sous ensemble de $P(\mathcal{C})$ est une **hiérarchie** sur $\mathcal{C}$ si :

1. $\mathcal{C} \in \mathcal{H}$
2. $M \in \mathcal{C} \Longrightarrow \{M\} \in \mathcal{H}$
3. $A \in \mathcal{H}$ et $A' \in \mathcal{H} \Longrightarrow A \cap A' = \emptyset$ ou $A \cap A' \in \mathcal{H}$

Critère d'aggrégation

Pour construire une hiérarchie il faut se donner un **critère d'aggrégration**.
Celui-ci est défini à l'aide d'une distance entre groupement de points appelée **critère**.
Soit d une distance sur $\mathcal{C}$, on peut définir de plusieurs façons une distance D entre groupement de points à partir de d.

Distance du lien minimal

$$D(A, A') = \min_{M \in A,\ M' \in A'} d(M, M')$$

Diamètre

$$D(A, A') = \max_{M \in A,\ M' \in A'} d(M, M')$$

Distance moyenne

$$D(A, A') = \frac{1}{n_A\ n_{A'}} \sum_{M \in A} \sum_{M' \in A'} d(M, M')$$

On suppose que $\mathcal{C} \in \mathcal{R}^p$ et que d est la distance euclidienne. On notera G_A le centre de gravité de A.

Distance entre centres de gravités

$$D(A, A') = d(G_A, G'_A)$$

Augmentation de l'inertie (ou critère de Ward)

$$\begin{aligned} D(A, A') &= n_A + n_{A'} I(A \cup A') - (n_A I(A) + n_{A'} I(A')) \\ &= \frac{n_A n_{A'}}{n_A + n_{A'}} d^2(G_A, G'_A) \end{aligned}$$

Les distances entre groupement de points sont en général des applications

$$D : P(\mathcal{C}) \times P(\mathcal{C}) \longrightarrow \mathcal{R}^+$$

vérifiant :

1. $D(A, A) = 0$
2. $D(A, A') = D(A', A)$

Ce ne sont donc pas en général des *distances* au sens habituel dans les espaces métriques. Le critère d'aggrégation utilisé habituellement est de regrouper successivement les sous ensembles de points les plus proche au sens de D.

4.3.2 Construction d'une hiérachie par classification ascendante

Déroulement de l'algorithme (CAH)

On suppose fixée une distance entre groupements de points D.

1. **Initialisation : Etape 1**
   La partition initiale $\mathcal{P}_1$ est constituée des points de $\mathcal{C}$ pris séparément.

$$\mathcal{P}_1 = (\{M_1\} \dots \{M_n\})$$

2. **Etape m**
   A l'étape $m-1$ on a obtenu la partition $\mathcal{P}_{m-1} = (A_1^{m-1}, \dots, A_k^{m-1})$.
   On construit la partition $\mathcal{P}_m$, en aggrégeant $A_{l(m-1)}^{m-1}$ et $A_{l'(m-1)}^{m-1}$ vérifiant :

$$D(A_{l(m-1)}^{m-1}, A_{l'(m-1)}^{m-1}) = \min_{l,l'} D(A_l^{m-1}, A_{l'}^{m-1})$$

3. **Fin : Etape n**
   A l'étape n tous les points sont regroupés en un seul ensemble.

$$\mathcal{P}_n = \mathcal{C}$$

Remarques

1. A chaque étape il peut y avoir plusieurs candidats possibles satisfaisant le critère d'aggrégation. On a donc plusieurs hiérarchies possibles.
2. Les résultats se présentent sous forme de **dendrogramme** sur lesquelles en abcisse sont représentés les points et en ordonnée le niveau d'aggrégation (c'est à dire la valeur du critère à l'étape d'aggrégation).
   On a représenté ci-dessous le dendrogramme pour la classification hiérarchique des données climatiques pour la distance euclidienne et le critère de Ward.
3. On peut à partir d'une hiérarchie couper horizontalement l'arbre obtenu et obtenir soit une partition avec un nombre de classes fixé, soit une partition à un niveau fixé.
4. La complexité de l'algorithme est en $O(n^3)$. Nous verrons dans la suite des variantes permettant de réduire cette complexité, dans certains cas.

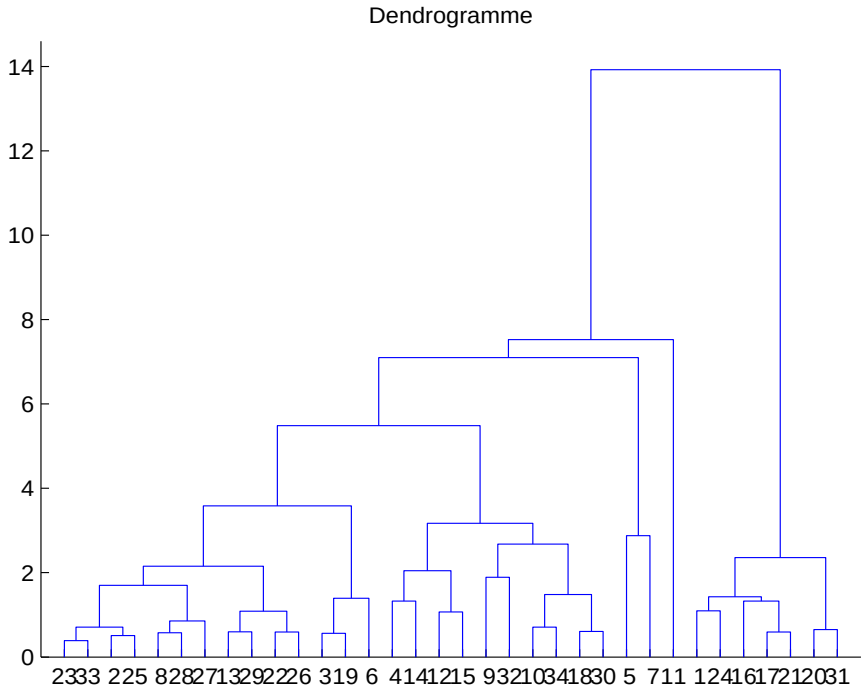

FIG. 4.2 – Dendrogramme avec le critère de Ward

4.3.3 Hiérarchie indicée

Généralités

Une hiérarchie H, est indicée par l'application $v : \mathcal{H} \longrightarrow \mathcal{R}^+$ si les deux propriétés suivantes sont vérifiées pour des groupements de points A et A' de $\mathcal{H}$

1. $v(A) = 0 \Longleftrightarrow card(A) = 1$
2. $A \subseteq A' \Longrightarrow v(A) \leq v(A')$

L'application v est appelé **indice d'aggrégation** de la hiérarchie. L'indice permet de mesurer les niveaux d'aggrégations et on peut utiliser cette valeur pour décider de la partition que l'on choisira. Plus l'indice est élevé plus le nombre de classes sera important. On peut aussi utilise les indices pour accélérer l'algorithme de la CAH (cf. [3]) En général l'indice se construit à l'aide du critère d'aggrégation de la manière suivante :
Lorsque on aggrège les classes A et A', l'indice d'aggrégation est :

$$v(A \cup A') = D(A, A')$$

Nèanmoins cette définition ne correspond pas toujours à un indice :

Exemple :

On choisit comme distance entre groupements la distance entre centre de gravités et on considère les 3 points suivants dans $\mathbb{R}^2$:

$$M_1 = (0; 0) \qquad M_2 = (4; 0) \qquad M_3 = (2; 3.5)$$

Les partitions successives sont :

$$\begin{array}{lcll} \mathcal{P}_1 & = & (\{M_1\};\{M_2\};\{M_3\}) & \\ \mathcal{P}_2 & = & (\{M_1;M_2\};\{M_3\}) & \text{et } v(\{M_1;M_2\}) = 4 \\ \mathcal{P}_3 & = & (\{M_1;M_2;M_3\}) & \text{et } v(\{M_1;M_2;M_3\}) = 3.5 \end{array}$$

Donc le niveau d'aggrégation est plus faible au niveau 3 qu'au niveau 2.

Réductibilité

On note $\mathcal{P}_A$, la partition qui précède la formation de $A = A_1 \cup A_2$ dans l'algorithme CAH. D est **réductible** si étant donné $\{B \in \mathcal{P}_A$, différent de A_1 et A_2 on a l'implication :

$$D(A_1, A_2) < \min\left(D(A_1, B), D(A_2, B) \Longrightarrow \min\left(D(A_1, B), D(A_2, B) < D(A, B)\right.\right.$$

Donc si A_1 et A_2 sont plus proches entre eux qu'ils ne le sont de B alors B est plus proche de A_1 ou de A_2 que de $A = A_1 \cup A_2$.

On a alors la proposition suivante :

Proposition :*Si D est* **réductible** *alors la hiérarchie construite par l'algorithme de la CAH est indicée.*
Cela signifie qu'il n'y a pas d'inversion des niveaux d'aggrégation.

On peut remarquer que dans l'exemple précédent cette propriété n'est pas vérifiée.
En effet

$$D(\{M_1\}, \{M_3\}) = D(\{M_2\}, \{M_3\}) = \frac{\sqrt{65}}{2} \simeq 4.03$$

et

$$D(\{M_1\}, \{M_2\}) = 4 < 4.03 \text{ alors que } D(\{M_3\}, \{\{M_1, M_2\}) = 3.5 < 4.03$$

Néanmoins de nombreuses distances entre groupements vérifient ce critère de réductibilité. Parmi toutes les distances présentées en 4.3.1, seule la distance du centre de gravité ne vérifie pas ce critère.
Sur les données climatiques avec la distance entre centres de gravité, on peut constater qu'il existe une inversion.

4.3.4 Algorithme des plus proches voisins réciproques

On suppose dans ce paragraphe que l'on utilise un critère d'aggrégation D réductible.

Ensembles voisins

Soit $\mathcal{P} = (A_1, \ldots, A_k)$ une partition.

- On dira que A_1 est le plus proche voisin de A_2 dans $\mathcal{P}$ ($A_1 \longrightarrow A_2$) si :

$$A_1 = \underset{A}{\operatorname{argmin}}\, D(A_2, A)$$

- On dira que A_1 et A_2 sont plus proches voisins réciproques ($A_1 \longleftrightarrow A_2$) si A_1 est le plus proche voisin de A_2 et A_2 est le plus proche voisin de A_1.

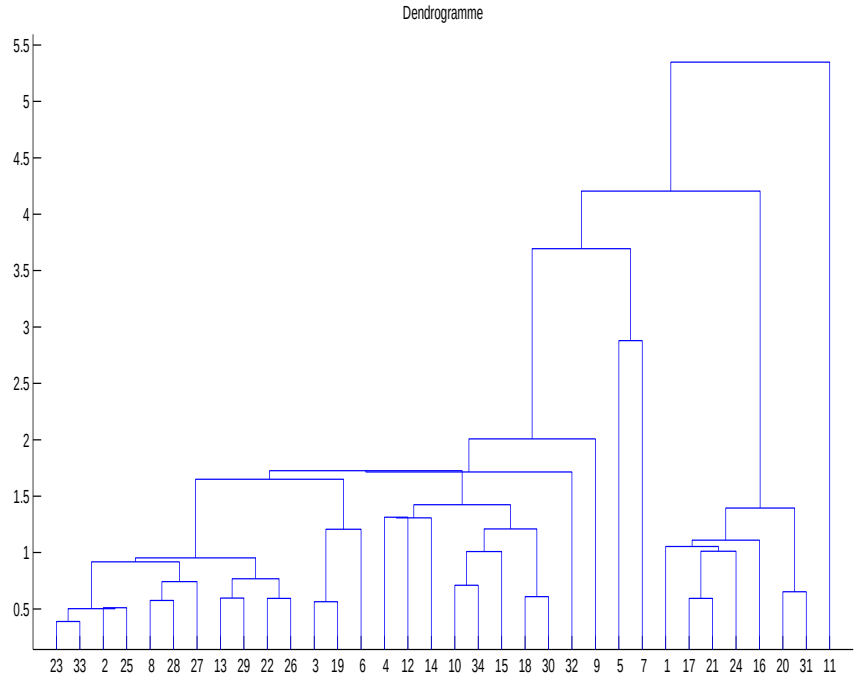

FIG. 4.3 – Dendrogramme pour la distance entre centres

On peut remarquer que toute partition contient au moins un couple de voisins réciproques : le couple $(A_{i_0}; A_{i'_0})$ vérifiant

$$D(A_{i_0}, A_{i'_0}) = \min_{(i,i')} D(A_i, A_{i'})$$

Déroulement de l'algorithme (PPVR)

1. **Initialisation : Etape 1**
   On part d'un point tiré au hasard $M_{(1)}$. On cherche son plus proche voisin $M_{(2)}$ et on continue. Cette chaine s'arrête nécessairement lorsque 2 éléments successifs sont 2 voisins réciproques $M_{(l-1)}$ et $M_{(l)}$.
   On a ainsi une chaine :

   $$M_{(1)} \longrightarrow M_{(2)} \longrightarrow \ldots \longrightarrow M_{(i-1)} \longrightarrow M_{(i)} \longrightarrow \ldots \longrightarrow M_{(l-1)} \longleftrightarrow M_{(l)}$$

   On aggrège alors $M_{(l-1)}$ et $M_{(l)}$, qui forment le premier niveau de la hiérarchie.
2. **Etape 2**

   - **Soit** $l = 2$:
     Alors on choisit un autre point de départ et on recherche 2 nouveaux ensembles à aggréger.
   - **Soit** $l > 2$:
     Alors on continue la recherche à partir de $M_{(l-1)}$

3. **Fin : Etape n**
   A l'étape n tous les points sont regroupés en un seul ensemble.

$$\mathcal{P}_n = \mathcal{C}$$

Cet algorithme permet de construire une hiérarchie avec une complexité polynomiale en $O(n^2)$.

4.3.5 Hiérarchie et arbres couvrant minimaux

Dans cette partie nous verrons que la construction d'une hiérarchie a un lien étroit avec le problème des arbres couvrant minimaux dans un graphe.

Ultramétrique

Soit E, un ensemble et Δ une distance sur E.
On dit que Δ est **ultramétrique**, si :
$x \in E$ et $x' \in E \Longrightarrow \Delta(x, x') \leq \max \Delta(x, x''), \Delta(x', x'')$ pour tout point x'' de E.

Distance de chaîne

Soit $G = (S, A)$ un graphe pondéré.
Une **chaîne** joignant 2 sommets x et x' est une suite ordonnée $\gamma(x, x') = \{x, a_1, x_1, a_2, x_2, \ldots, a_n, x'\}$ où a_i représente l'arête $\{x_{i-1}, x_i\}$.
On note $p(a)$ le poids de l'arête a.
Le pas de la chaîne est :

$$\phi(\gamma) = \max_{k=1,\ldots,n} p(a_i)$$

On notera $\mathcal{C}(x, x')$, l'ensemble des chaînes joignant x à x'. La **distance de chaîne** δ entre x et x' est alors définie par :

$$\delta(x, x') = \min_{\gamma \in \mathcal{C}(x,x')} \phi(\gamma)$$

On peut alors démontrer que la distance de chaîne est une distance ultramétrique.
Cette définition s'applique également à un nuage de point $\mathcal{C}$ dans un espace métrique muni d'une distance d en considérant le graphe complet pondéré associé : les points sont les sommets, et le poids la distance entre 2 sommets.

Lien entre ultramétrique et hiérarchie

Soit (H, v), une hiérarchie indicée sur $\mathcal{C}$.
On note pour 2 points M et M' de $\mathcal{C}$:

$$\Delta(M, M') = \min_{A \in H} \{v(A); M \in A, M' \in A\}$$

On peut démontrer que Δ est une distance ultramétrique sur $\mathcal{C}$.

Réciproquement, soit Δ une distance ultramétrique sur $\mathcal{C}$.
Pour $\alpha > 0$ on définit la relation binaire R_α sur $\mathcal{C}$:

$$M \; R_\alpha \; M' \Longleftrightarrow \Delta(M, M') \leq \alpha$$

Alors il est facile de voir que R_α est une relation d'équivalence sur $\mathcal{C}$.
Soit $H = \{\mathcal{C}/R_\alpha; \alpha \in \mathcal{R}^+\}$, l'ensemble de toutes les classes d'équivalence obtenues en faisant varier α.
Alors H définit une hiérarchie indicée dont l'indice v est :

$$v(A) = \inf\{\alpha > 0; a \in \mathcal{C}/R_\alpha\}$$

Ces propriétés montrent qu'il existe une bijection entre l'ensemble des distances ultramétriques sur $(\mathcal{C}, d)$ noté $\mathcal{U}_\mathcal{C}$ et l'ensemble des hiérarchies indicées sur $(\mathcal{C}, d)$.

Sous dominante

Sur $(\mathcal{C}, d)$, on peut définir une distance ultramétrique particulière Δ_d appelée **sous dominante**, définie par :

$$\Delta_d(M, M') = \max\{\Delta(M, M'); \Delta \in \mathcal{U}_\mathcal{C}, \Delta \leq d\}$$

Donc Δ_d est la borne supérieure des distances ultramétriques inférieures à d. On entend ici que $d_1 \leq d_2$ si $d_1(M, M') \leq d_2(M, M')$ pour tout points M et M' de $\mathcal{C}$.

On démontre que la sous dominante peut être construite de 2 manières différentes :

1. Par construction d'un arbre couvrant minimal.
2. Par construction d'une hiérarchie pour la distance du lien minimal

La distance ultramétrique correspondante est la distance de chaîne et la hiérarchie construite hiérarchie du saut minimal.

On a ainsi un moyen de construire une hiérarchie avec des algorithmes de construction d'arbres couvrant minimaux tels les algorithmes de Kruskal ou de Prim dont les complexités sont respectivement $O(n^2 \log_2 n)$ et $O(n^2)$.

Algorithme de Prim (PRIM)

La construction d'une hiérarchie avec l'algorithme de Prim, consiste à construire l'arbre couvrant minimal puis en déduire la hiérarchie associée.

Construction de l'arbre couvrant minimal :

Soient T et A , 2 ensembles vides au départ.

- **Etape 1** :
  Choisir un point quelconque de $\mathcal{C}$, et le transférer dans T.
- **Etape m** :
  Trouver un couple de point $M_m \in T$ et $M'_m \notin T$ vérifiant :

  $$d(M_m, M'_m) = \min_{M \in T, M' \notin T} d(M, M')$$

  Mettre M'_m dans T et l'arête $\{M_m, M'_m\}$, dans A.
- **Etape n** :
  Transférer le dernier point dans T et l'arête correspondante dans A. On a alors $T = \mathcal{C}$ et l'ensemble des arêtes de A forme un arbre couvrant minimal.

Construction de la hiérarchie :

La hiérarchie est alors construite de manière descendante.

La plus grande arête dans A est le plus haut niveau d'aggrégation. En supprimant cette arête de l'arbre on fait apparaitre 2 groupes d'individus.

On peut alors continuer à supprimer les arêtes dans l'ordre décroissant de longueur et on fait apparaitre successivement tous les niveaux de la hiérarchie.

Remarques

L'avantage de cet algorithme est la simplicité. Néanmoins il induit un effet de chaîne. Des sous ensembles éloignés mais reliés entre eux par une chaîne de points rapprochés auront tendance a être considérés comme proches.

On peut néanmoins utilisé cet algorithme comme initialisation d'un algorithme de partitionnement.

Chapitre 5

Discrimination

5.1 Introduction

On suppose que l'on se donne un nuage de points $\mathcal{C} \in \mathcal{R}^p$ et que les points de ce nuage soient affectés à k classes formant une partition de $\mathcal{C}$, $A_1, \ldots ; A_k$.
On dispose ainsi d'un ensemble d'apprentissage supervisé.

La **discrimination** consiste à déterminer à partir de ces données :

1. les limites séparant les classes
2. la procédure d'affectation des nouveaux points dans les classes.

Le principe général consiste à utiliser des fonctions **discriminantes** : $\Phi_1, \ldots ; \Phi_k$ où $\Phi_l : \mathcal{R}^p \longrightarrow \mathcal{R}^+$ est la fonction discriminante associée à la classe A_l
Les fonctions $\Phi_1, \ldots ; \Phi_k$ sont discriminantes si elles vérifient ;

$$M \in A_{l_0} \Longleftrightarrow l_0 = \arg \min_{l=1,\ldots,k} \Phi_l(M)$$

La procédure de discrimination consiste à trouver des fonctions discriminantes.
La procédure d'affectation d'un nouveau point M dans les classes est alors la suivante.
Soit

$$\delta(M) = \arg \min_{l=1,\ldots,k} \Phi_l(M)$$

Alors le point M est affecté à la classe $A_{\delta(M)}$.

La fonction δ est la fonction de **décision**.

La **surface séparatrice** $\Sigma_{ll'}$ entre les classes A_l et $A_{l'}$ est la courbe définie par :

$$\Sigma_{ll'} = \{M \in \mathcal{R}^p ; \Phi_l(M) = \Phi_{l'}(M)\}$$

Nous examinerons dans la suite les méthodes suivantes :

1. Discrimination par mesure de voisinage
2. Méthodes bayésiennes
3. Séparation linéaire

5.2 Discrimination par mesure de voisinage

5.2.1 Mesure de voisinage

Une **mesure de voisinage** est une fonction $S_A(M)$ permettant de mesurer la distance entre le point M et l'ensemble A.

Par exemple si d est une distance sur $\mathcal{C}$, on a vu que l'on peut définir de plusieurs façons une distance D entre groupement de points à partir de d (cf. 4.3.1). On peut alors définir une mesure de voisinage entre le point M et l'ensemble A par :

$$S_A(M) = D(M, A)$$

.

Il est alors naturel d'utiliser cette mesure de voisinage comme fonction discriminante :

$$\Phi_l(M) = D(M, A_l)$$

Par exemple, si on utilise la distance entre centre on a alors comme fonction discriminante

$$S_A(M) = d(M, G(A))$$

et la surface séparatrice entre A_1 et A_2 est l'hyperplan médiateur entre $G(A_1)$ et $G(A_2)$.

Il est facile de voir qu'utiliser cette distance n'assure pas que les classes de l'ensemble d'apprentissage soit correctement séparées.

5.2.2 Méthode des q plus proches voisins (q-PPV)

Soit M un point de $\mathcal{R}^p$ muni d'une distance d et $A = M_1, \ldots, M_l$

On note $A(M)$, le réarrangement par ordre croissant de distance à M des points de A. Soit

$$A(M) = (M_{(1)}, \ldots, M_{(l)})$$

avec

$$d(M_{(1)}, M) \leq d(M_{(1)}, M) \leq \ldots \leq d(M_{(l)}, M)$$

On définit alors la mesure de voisinage des q plus proches voisins par :

$$S_A(M) = \frac{1}{q} \sum_{i=1}^{q} d(M_{(i)}, M)$$

Si $q = 1$ on a alors

$$S_A(M) = \min_{M' \in A} d(M, M')$$

On obtient donc la mesure de voisinage construite à partir de la distance du lien minimum.

La surface séparatrice est alors un polyèdre.

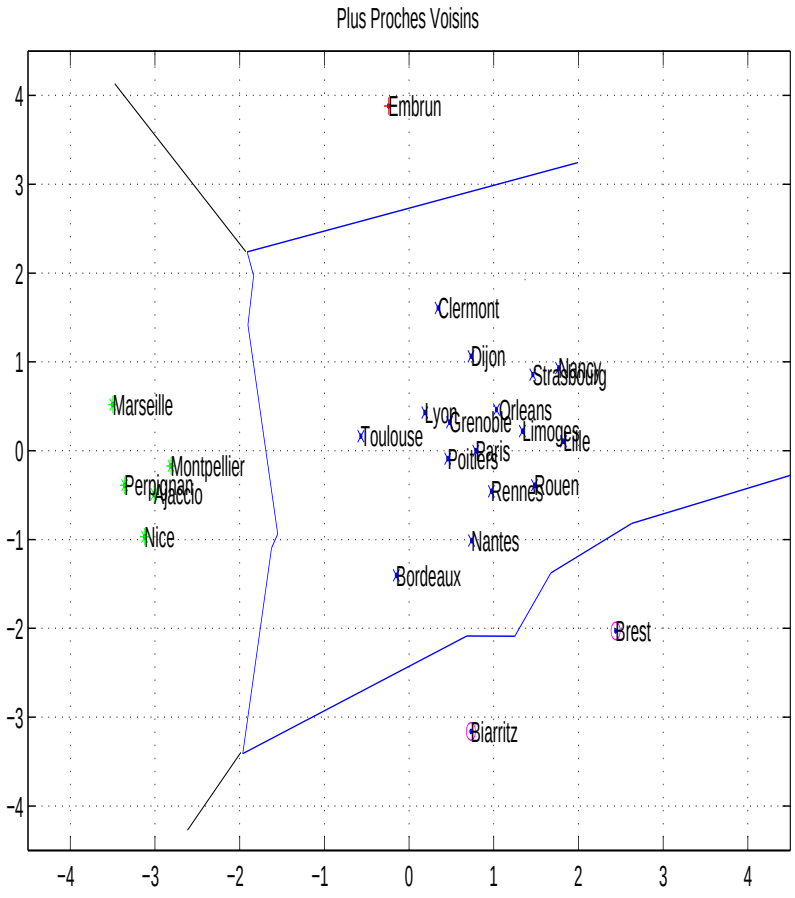

FIG. 5.1 – Discrimination en 4 classes par la méthode du PPV

5.3 Discrimination bayésienne

La discrimination bayésienne est une méthode inférentielle de statistique bayésienne associée à un modèle statistique en général gaussien. La formalisation bayésienne du problème de la discrimination est une base importante pour des algorithmes tel l'algorithme E.M.
Dans cette méthode le caractère supervisé permet d'ajuster les données à un modèle probabiliste. On utilise alors le modèle pour calculer les fonctions discriminantes et la fonction de décision.

5.3.1 Modèle statistique

Loi des observations

On suppose que l'individu observé est la réalisation d'une variable aléatoire X.
Donc le point M de coordonnées x est la réalisation de la variable X pour l'individu ω : $x = X(\omega)$.

Dans cette partie on utilisera les coordonnées x pour désigner le point M.
On associe à cette variable aléatoire l'étiquette de la classe $A_1, \ldots ; A_k$ à laquelle il appartient, qui est aussi une variable aléatoire discrète N.
Donc pour l'individu ω, l'étiquette est $N : (N = l) = (M \in A_l)$

Donc le modèle est constitué du couple de variable aléatoire $Y = (X, N)$:

- Dans la phase d' **apprentissage**, X et N sont observées
- Dans la phase d' **affectation**, X est observée, mais N n'est pas observée

On notera $\alpha_1, \ldots, \alpha_k$ les probabilités d'appartenance aux classes, caractérisant la loi de N : $P(N = l) = \alpha_l$

On supposera que la loi conditionnelle d'appartenance à la classe A_l est continue et caractérisée par sa densité f_l.
D'après la formule de Bayes la loi de X sera alors caractérisée par une densité f_X, définie par $f_X(x) = \sum_{l=1}^{k} \alpha_l \ f_l(x)$

Grâce à la formule de Bayes on peut aussi définir la probabilité que la classe d'appartenance de l'individu ω soit A_l sachant que l' individu ω est représenté par le point M de coordonnées x, notée α_l^x :

$$\alpha_l^x = P(N = l/X = x) = \frac{\alpha_l \ f_l(x)}{\sum_{l'=1}^{k} \alpha_{l'} \ f_{l'}(x)}$$

C'est la loi d'appartenance *a posteriori*.

La loi de probabilité de X est une **loi de mélange**. Ce type de loi pose des problèmes statistiques pour l'identification car des densités et des probabilités différentes peuvent engendrer la même loi de mélange. Toutefois ces problèmes ne se posent pas avec les lois gaussiennes.

Mélange de lois gaussiennes

Par exemple si X est un mélange de lois gaussiennes de paramètres respectifs $(\mu_1; \sigma_1^2)$ et $(\mu_2; \sigma_2^2)$, en proportion α et $1 - \alpha$ on notera :

$$X \sim \alpha \mathcal{N}(\mu_1; \sigma_1^2) + (1 - \alpha)\mathcal{N}(\mu_2; \sigma_2^2)$$

La densité de X est :

$$f_X(x) = \alpha \frac{1}{\sqrt{2\pi}\sigma_1} \ e^{-\frac{1}{2}(\frac{x-\mu_1}{\sigma_1})^2} + (1 - \alpha)\frac{1}{\sqrt{2\pi}\sigma_2} \ e^{-\frac{1}{2}(\frac{x-\mu_2}{\sigma_2})^2}$$

L'espérance de X est :

$$E(X) = \alpha\mu_1 + (1 - \alpha)\mu_2$$

La variance de X est :

$$V(X) = \alpha\sigma_1^2) + (1 - \alpha)\sigma_2^2) + (\alpha(1 - \alpha))(\mu_1 - \mu_2)^2$$

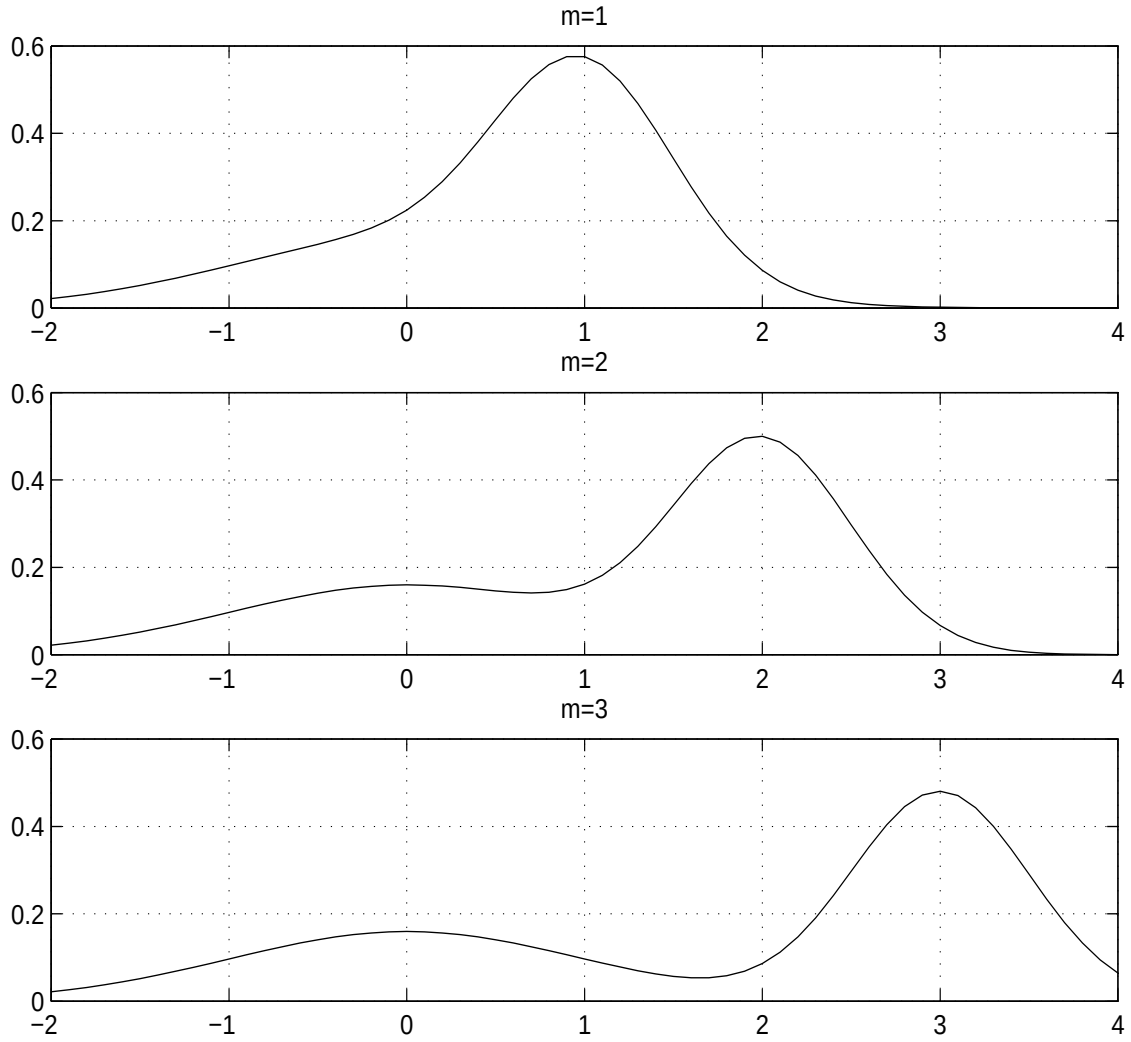

FIG. 5.2 – Mélange gaussien : $X \sim 0.4\mathcal{N}(0;1) + 0.6\mathcal{N}(m;0.25)$

5.3.2 Décision bayésienne

Coût d'une décision

On suppose que l'attribution d'un individu de la classe A_l à la classe $A_{l'}$ a un coût noté $c(l,l')$. La matrice carrée $C = [c(l,l')]_{1\leq l,l'\leq k}$ est appelée **matrice de perte**

En l'absence d'information *a priori* sur les classes, on utilise le coût uniforme $c_u(l,l')$ défini par :

$$\left| \begin{array}{lll} c_u(l,l') = 0 & \text{si} & l = l' \\ c_u(l,l') = 1 & \text{si} & l \neq l' \end{array} \right.$$

Décision bayésienne *a priori*

Prendre une décision δ pour un individu ω de classe d'appartenance inconnue d'étiquette N, c'est lui attribuer une classe d'étiquette l. Dans le cas où on n'a pas d'observation concernant cet individu, la décision δ prise sera la même pour tous les individus.

Le **risque** de cette décision est :

$$R(\delta) = E(c(N, \delta)$$

La **décision bayésienne** consiste à choisir la décision δ minimisant le risque :

$$\delta = \arg \min_{l=1,\ldots,k} R(l)$$

Dans le cas du coût uniforme :

$$R(l') = \sum_{l=1,\ldots,k} \alpha_l c(l, l') = 1 - \alpha_{l'}$$

D'où

$$\delta = \arg \max_{l=1,\ldots,k} \alpha_{l'}$$

Donc en dehors de toute observation la décision bayésienne (*a priori*) consiste à choisir la classe ayant la plus grande probabilité d'appartenance.

Décision bayésienne *a posteriori*

On suppose maintenant que l'on fait une observation de la variable X et donc que l'on connait $x = X(\omega)$.

Il est naturel d'utiliser cette information supplémentaire pour affecter l'individu ω à une classe.

Le risque d'attribuer l'individu ω à la classe A_l, sachant qu'il est représenté par le point M de coordonnées x est :

$$R^x(l') = E(c(N, l'/X = x)$$

La **décision bayésienne** *a posteriori* est l'application $\delta : \quad M \longrightarrow \delta(M)$ minimisant le risque R^M :

$$\delta(x) = \arg \min_{l=1,\ldots,k} R^x(l)$$

Les fonctions $\Phi_l(x) = R^x(l)$ sont les fonctions discriminantes de la méthode bayésienne. Le risque bayésien de cette décision pour l'observation M est $R^x(\delta(x)$

L'erreur bayésienne $R(\delta)$ de cette décision est la moyenne du risque bayésien :

$$\begin{aligned} R(\delta) &= E(R^X(\delta(X)) \\ &= \int (\sum_{l=1,\ldots,k} c(l, \delta(x)) \alpha_l^x) \; f_X(x) \; dx \\ &= \sum_{l=1,\ldots,k} \alpha_l \; (\int c(l, \delta(x)) \; f_l(x) \; dx) \end{aligned}$$

Si on note :

$$\begin{aligned} e_l(\delta) &= E(c(N, \delta(X)/N = l) \\ &= \int c(l, \delta(x)) \; f_l(x) \; dx \end{aligned}$$

alors

$$R(\delta) = \sum_{l=1,\ldots,k} \alpha_l e_l(\delta)$$

La quantité $e_l(\delta)$ est le **taux d'erreur de la décision** δ lorsque la classe de ω est A_l. Lorsque le coût est uniforme le taux d'erreur $e_l(\delta)$ est alors la probabilité sachant que la classe de ω est A_l de décider de l'affecter à une autre classe.

$$\begin{aligned} e_l(\delta) &= P(\delta(X) \neq l / N = l) \\ &= \int_{\{x,\delta(x)\neq l\}} f_l(x)\, dx \end{aligned}$$

5.3.3 Cas de deux classes

Cas général

On suppose qu'il y a 2 classes et que le coût est uniforme. Dans ce cas on a :

$$\left| \begin{array}{lcl} R^x(1) &=& \alpha_2^x \\ R^x(2) &=& \alpha_1^x \end{array} \right.$$

La décision est alors :

$$\delta(x) = \left| \begin{array}{lll} 1 & \text{si} & \alpha_1^x \geq \alpha_2^x \\ 2 & \text{si} & \alpha_1^x \leq \alpha_2^x \end{array} \right.$$

L'erreur bayésienne est :

$$\begin{aligned} R(\delta) &= \alpha_1 e_1(\delta) + \alpha_2 e_2(\delta) \\ &= \int_{\alpha_1^x \leq \alpha_2^x} \alpha_1^x f(x)\, dx + \int_{\alpha_1^x \geq \alpha_2^x} \alpha_2^x f(x)\, dx \\ &= E(\min(\alpha_1^X, \alpha_2^X)) \end{aligned}$$

On peut évaluer dans les mêmes conditions asymtotiquement l'erreur commise dans la méthode des q-PPV et on obtient :

$$R_{1-PPV} = 2E(\alpha_1^X \alpha_2^X)$$

et

$$R_{2-PPV} = E(\alpha_1^X \alpha_2^X)$$

Donc

$$R_{2-PPV} \leq R(\delta) \leq R_{1-PPV} \leq 2R(\delta)$$

Cas gaussien

On suppose que X est un mélange de 2 vecteurs aléatoires gaussiens de R^p de même matrice de covariance V et avec même probabilité d'appartenance $\alpha = \frac{1}{2}$. D'où

$$X \sim \frac{1}{2}\mathcal{N}(\mu_1; V) + \frac{1}{2}\mathcal{N}(\mu_2; V)$$

On rappelle que la densité conditionnelle d'appartenance à la classe A_l est :

$$f_l(x) = \frac{1}{(2\pi)^{\frac{p}{2}} |V|^{\frac{1}{2}}} \exp -\frac{1}{2}(x-\mu_l)^t V^{-1}(x-\mu_l)$$

On notera :

$$d_V^2(x, x') = (x - x')^t V^{-1}(x - x')$$

Alors la décision bayésienne est :

$$\delta(x) = \left| \begin{array}{lll} 1 & \text{si} & d_V^2(x, \mu_1) \leq d_V^2(x, \mu_2) \\ 2 & \text{si} & d_V^2(x, \mu_1) \geq d_V^2(x, \mu_2) \end{array} \right.$$

Dans ce cas on a une discrimination par mesure de voisinage avec comme mesure de voisinage :

$$S(x, A_l) = d_V^2(x, \mu_l)$$

Les surfaces séparatrices sont des surfaces du second degré : dans le plan ce sont des coniques.

Les taux d'erreurs sont égaux et on a :

$$R(\delta) = e_1(\delta) = e_2(\delta) = F_0(-d_V^2(\mu_1, \mu_2))$$

où F_0 est la fonction de répartition de la loi normale.

5.3.4 Estimation des paramètres d'un mélange

Pour utiliser la méthode bayésienne il faut pouvoir estimer les lois de X et de N.

Méthode de Parzen

C'est une méthode complètement supervisée. Sur l'ensemble d'apprentissage X et N, sont observées. Donc pour un individu ω on connait sa position $x = X(\omega)$ et sa classe $l = N(\omega)$.
On dispose donc d'un échantillon $(Y_1, \ldots, Y_n)$ où $Y_i = (X_i, N_i)$.
On utilise alors les méthodes classiques d'estimation des lois pour obtenir la densité $f_l(x)$ et la probabilité d'appartenance α_l.

En particulier on pourra estimer α_l par la fréquence d'appartenance :

$$\hat{\alpha}_l = \frac{1}{n} \sum_{i=1}^{n} \mathbf{1}_{N_i = l}$$

La densité conditionnelle $f_l(x)$ pourra être estimer par des méthodes paramétriques (par exemple dans le cas gaussien estimation de la moyenne et de la matrice de covariance) ou par des méthodes non paramétriques par exemple estimateur à noyaux (c.f. [**?**])

Algorithme Estimation-Maximisation

L'algorithme E.M est un algorithme permettant d'estimer dans un modèle paramétrique les paramètres d'un mélange lorsque l'ensemble d'apprentissage n'est pas complètement supervisé.
Dans ce cas pour un individu ω on connait sa position $x = X(\omega)$ mais pas sa classe $l = N(\omega)$.

On suppose que le modèle est paramétrique au sens où la densité conditionnelle est de la forme $f_l(x) = f(\theta_l, x)$ et on fixe a priori le nombre de classes k.
L'algorithme consiste à estimer alternativement les probabilités d'appartenance et les paramètres des lois conditionnelles. Les paramètres des lois conditionnelles sont estimés par un estimateur du maximum de vraisemblance conditionnel d'où le terme de maximisation.

Déroulement de l'algorithme (E.M.)

1. **Initialisation**
   On se donne des paramètres initiaux :
   $$\alpha^0 = (\alpha_1^0, \ldots, \alpha_k^0)$$
   et
   $$\theta^0 = (\theta_1^0, \ldots, \theta_k^0)$$
2. **Etape m**
   On dispose des paramètres calculés à l'étape précédente :
   $$\alpha^{m-1} = (\alpha_1^{m-1}, \ldots, \alpha_k^{m-1})$$
   et
   $$\theta^{m-1} = (\theta_1^{m-1}, \ldots, \theta_k^{m-1})$$
   Il y a alors 2 étapes :
   Estimation : On estime pour $i = 1, \ldots, n$ et $l = 1, \ldots, k$, la probabilité $\pi_l^m(x_i)$ que ω_i appartienne à la classe A_l pour les paramètres courants.
   $$\pi_l^m(x_i) = \frac{\alpha_l^{m-1}\ f(\theta_l^{m-1}, x)}{\sum_{l'=1}^{k} \alpha_{l'}^{m-1}\ f(\theta_l^{m-1}, x)}$$
   Maximisation : On estime alors par la méthode du maximum de vraisemblance les nouveaux paramètres :
   $$\alpha_l^m = \frac{1}{n} \sum_{i=1}^{n} \pi_l^m(x_i)$$
   et
   $$\theta_l^m = \underset{\theta}{\operatorname{argmax}} \sum_{i=1}^{n} \pi_l^m(x_i) \ln f(\theta, x_i)$$
3. **Fin**
   On arrête lorsque les paramètres ne varient plus, ou moins qu'un seuil fixé.

Bibliographie

[1] G.Celeux, E.Diday, G.Govaert, Y.Lechevallier, Y.Ralambondrainy :*Classification automatique des données*,Dunod (1989)

[2] P.G.Ciarlet :*Introduction à l'analyse numérique matricielle et à l'optimisation*,Masson (1984)

[3] E.Diday, J.Lemaire, J.Pouget, F.Testu :*Eléments d'analyse de données*,Dunod (1982)

[4] R.O. Duda, P.E.Hart :*Pattern Classification and Scene Analysis*,J.Wiley (1973)

[5] K.Fukunaga :*Introduction to Statistical Pattern Recognition*,Academic Press (1972)

[6] L.Lebart, A.Morineau, M.Piron :*Statistique exploratoire multidimensionnelle*,Dunod (1997)

[7] S.Thiria, Y.Lechevallier, O.Gascuel, S.Canu :*Statistique et méthodes neuronales*,Dunod (1997)

[8] A.Cornujols, L.Miclet :*Apprentissage artificiel : Concepts et algorithmes*,Eyrolles (2002)

Table des matières