



HAL
open science

Quoi retenir d'un arbre de classification? Un essai en quantification d'image numérisée

Israël-César Lerman, Nadia Ghazzali

► **To cite this version:**

Israël-César Lerman, Nadia Ghazzali. Quoi retenir d'un arbre de classification? Un essai en quantification d'image numérisée. [Rapport de recherche] RR-1386, INRIA. 1991. inria-00075175

HAL Id: inria-00075175

<https://inria.hal.science/inria-00075175>

Submitted on 24 May 2006

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

INRIA

UNITÉ DE RECHERCHE
INRIA-RENNES

Institut National
de Recherche
en Informatique
et en Automatique

Domaine de Voluceau
Rocquencourt
B.P.105
78153 Le Chesnay Cedex
France
Tél.: (1) 39 63 55 11

Rapports de Recherche

N° 1386

Programme 5

Traitement du signal, Automatique et Productique

QUOI RETENIR D'UN ARBRE DE CLASSIFICATION ? UN ESSAI EN QUANTIFICATION D'IMAGE NUMERISEE

Israël César LERMAN
Nadia GHAZZALI

Janvier 1991



Campus Universitaire de Beaulieu
35042 - RENNES CEDEX
FRANCE
Téléphone : 99.36.20.00
Télex : UNIRISA 950 473F
Télécopie : 99.38.38.32

Quoi retenir d'un arbre de classification? Un essai en quantification d'image numérisée

Israël-César Lerman & Nadia Ghazzali

Publication Interne n° 568 - Décembre 1990 - 36 pages

Projet CADO - Programme 5

Résumé

Ce travail traite du problème fondamental de la réduction *significative* des arbres de classification. Son premier apport est de situer l'approche de A.D. Gordon [6] par rapport à la nôtre, plus ancienne [13, 14, 15], qui est fondée sur la détection des noeuds *significatifs* d'un arbre des classifications. Il a dans ces conditions, été nécessaire de cerner plus précisément et d'élaborer les algorithmes de passage, entre des notions proches, mais non identiques, qui sont – dans la littérature en classification – utilisées plus ou moins indépendamment l'une de l'autre. Il s'agit de : *arbre des classifications; arbre indicé ou valué des classifications; hiérarchie des parties et arbre associé; hiérarchie indicée de parties et dendrogramme associé.*

Un deuxième apport original correspond à la définition et à la construction algorithmique d'une hiérarchie de parties *pertinentes*, à partir de la définition des noeuds *significatifs*.

Cette notion et la construction algorithmique conséquente, se sont imposées à nous lorsque nous avons voulu utiliser la classification hiérarchique à des fins de quantification scalaire en imagerie numérique noir et blanc [20].

Mots clés: Classification hiérarchique; Arbres condensés; Noeuds *significatifs*; Quantification en imagerie numérique.

What do we retain from a classification tree? An experiment in image coding

Abstract

This work concerns the fundamental problem of the *significant* reduction of classification trees. Its first contribution is to situate the Gordon approach [6], with respect to ours [13, 14, 15]. The latter is based on the detection of *significant* nodes in the classification tree. Thus, it was necessary to express very accurately, different notions concerning the searched structure in hierarchical classification. These notions are near but non identical; there are : *tree of classifications*; *valued tree of classifications*; *hierarchy of subsets and associated tree*; *hierarchy of subsets and associated dendrogram*. On the other hand, we have built algorithms which enable the passage from one notion to the another one.

A second original aspect of this paper concerns the definition and the algorithmic construction of a hierarchy of *relevant* subsets, obtained from the definition of the *significant* nodes of a classification tree.

The latter notion and consequently, the corresponding algorithmic construction, have been pointed in the framework of using hierarchical classification in image coding (scalar quantization) [20].

Keywords: Hierarchical classification; Parsimonious trees; Significant nodes; Image coding.

Table des matières

1	Présentation générale	4
2	Arbres des classifications, hiérarchie des parties	6
3	Approches dans la réduction de l'arbre	15
3.1	Préambule	15
3.2	Approche de A.D. Gordon	16
3.3	Niveaux et noeuds "significatifs"	17
3.4	Arbre des parties "pertinentes"	24
4	Application en quantification scalaire	27
5	Conclusion et perspectives	29

1 Présentation générale

L'étude que nous reprenons ici, avec un point de vue plus riche et une contribution nouvelle concerne:

- la réduction *significative* d'un arbre de classification;
- Le dessin graphique de l'arbre réduit, de façon à obtenir des conditions optimales de lisibilité, pour une interprétation claire des résultats;
- la rétention à partir de l'arbre réduit, de partitions *intéressantes*.

En effet, dès la fin des années 60, nous avons pris conscience de la nécessité impérieuse de réduire les arbres de classification dont l'interprétation – dès que la taille de l'ensemble à classifier dépasse quelques dizaines – devient vite très difficile et incertaine; et cela, même si les indices et critères utilisés pour alimenter l'algorithme de formation de l'arbre, sont parfaitement adaptés à la nature des données traitées. Les questions implicites posées par l'utilisateur, expert des données, sont généralement les suivantes:

- où commence une classe?;
- tel noeud correspond-t-il à l'achèvement d'une classe à un certain niveau de synthèse de l'information?;
- de façon liée au point précédent, quelles sont les distinctions les plus *significatives*?;
- comment retenir à partir de l'arbre des classifications quelques partitions, plus ou moins fines, *significatives*?

Nous avons proposé [13] de balayer la suite des niveaux d'un arbre de classification sur un ensemble fini E au moyen d'un critère d'association $S[\omega(E), \pi]$ entre une information de nature ordinaire $\omega(E)$ quant aux ressemblances entre les éléments E et une partition donnée π ; étant entendu que chaque niveau de l'arbre définit une partition. $\omega(E)$ est ce que nous appelons *préordonnance totale* sur E . On considère les deux suites de nombres:

$$\{S[\omega(E), \pi_i] \mid 1 \leq i \leq I\} \quad (1)$$

$$\{\tau_i = S[\omega(E), \pi_i] - S[\omega(E), \pi_{i-1}] \mid 2 \leq i \leq I\} , \quad (2)$$

où π_i , partition du i -ème niveau, est d'autant moins fine que i est grand.

Un niveau (resp. un noeud) sera dit *significatif* s'il correspond à un maximum local, le long de la suite croissante des niveaux de l'arbre de

S [cf.(1)] {resp. du taux τ d'accroissement de S [cf.(2)]}. Dans ce contexte, nous appelons S : *Statistique globale* des niveaux, et τ : *Statistique locale* des niveaux. Nous expliciterons au paragraphe 3 ci-dessous le principe de l'élaboration de $S[\omega(E), \pi]$, compte tenu des derniers développements [14, 15].

Une partition π_{i_0} est d'autant plus *significative* que $S[\omega(E), \pi_{i_0}]$ représente un maximum local dont la valeur se distingue nettement des valeurs de $S[\omega(E), \pi_i]$ pour i appartenant à un intervalle entourant strictement i_0 . Il y a bien sûr un indice de niveau i_1 pour lequel $S[\omega(E), \pi_{i_1}]$ présente un maximum absolu. C'est par rapport à la partition π_{i_1} , ainsi détectée que J.L. Mollière [21] compare le comportement de notre critère à celui dit CCC (Cubic Clustering Criterion) conçu dans le contexte de la classification d'un nuage de points dans un espace euclidien.

Pourtant l'expert, spécialiste des données, ne se contente pas d'une seule partition et retient quelques partitions correspondant à différents degrés de finesse. Plus est, lorsque nous présentons à l'expert une partition hautement significative, il lui arrive, le plus souvent, de retenir presque toutes les classes; mais, pas nécessairement toutes ! Il pourra préférer, pour une ou deux d'entre elles, de descendre au niveau directement inférieur afin de remplacer la classe par ses sous classes.

La notion de noeud *significatif* est encore plus intéressante, elle correspond, dans la pratique expérimentale, à l'achèvement cohérent d'une sous classe [cf. §3]. Mais là également, il y a un intérêt certain à considérer les noeuds fils donnant directement naissance au noeud *significatif*. D'autre part, il est tout aussi intéressant de reconnaître le noeud père que va rejoindre le noeud *significatif*.

Bien que nous soyons partis de l'évaluation de l'adéquation d'une même partition produite à un niveau donné de l'arbre des classifications. Nous avons maintenant tendance à "oublier" la notion de niveau pour ne retenir qu'un système de noeuds *pertinents*. Cette démarche permet une grande souplesse si on désire, à partir de ces noeuds, retenir une partition telle que la taille de chacune des classes soit d'un ordre fixé (e.g. inférieure à un seuil donné). En effet, si on parle dans la littérature de produire des classifications dont le nombre de classes est d'un ordre fixé, le problème des contraintes de taille n'est – à notre connaissance – guère beaucoup évoqué. Or ce problème s'est avéré crucial dans notre introduction de la classification hiérarchique en quantification scalaire en imagerie numérique [20] [cf. §3].

Si jusqu'à présent nous proposons de représenter l'arbre condensé aux niveaux où apparaît un noeud *significatif*, nous proposerons ici une représentation fondée sur un système de noeuds *pertinents*, en "oubliant" complètement la notion de niveau de l'arbre des classifications qui a justement permis

l'émergence.

L'approche de A.D. Gordon [6] – qui ignorait notre contribution lors de la rédaction de son article – consiste également à retenir à partir, non pas d'un arbre des classifications, mais d'une hiérarchie indicée, un système de noeuds. A cette fin, il ne considère pas – comme c'est notre cas – la signification d'un noeud particulier permettant le passage d'une partition à une partition moins fine. Ce qu'il considère en revanche, c'est à partir d'un algorithme de découpage de la hiérarchie – une partition de l'ensemble des noeuds. Pour une même classe de cette partition, il retient l'ensemble des noeuds maximaux, au sens de l'inclusion entre parties représentées par ces noeuds; ainsi, pour un ensemble de noeuds mutuellement comparables, c'est le noeud le plus proche de la racine qui est retenu. Le découpage est effectué compte tenu de l'optimisation d'un critère [cf. §3.2.].

Pour bien comprendre la différence de nature entre notre approche et celle de Gordon; mais aussi et surtout, dans un but beaucoup plus général de classification, nous allons présenter au paragraphe 2, les différentes expressions formelles généralement adoptées de l'argument de la classification hiérarchique. Il s'agit de :

- arbre des classifications;
- arbre des classifications indicé ou dendrogramme;
- hiérarchie de parties indicée;
- hiérarchie des parties et arbre des parties.

Nous préciserons de plus, les algorithmes de passage entre les différents types ci-dessus de structure. A cette fin, nous nous référerons – comme le cas quasi général – à la construction ascendante hiérarchique d'un arbre de classification. Mais, cette référence n'est pas indispensable.

Au paragraphe 3, nous présenterons les deux approches; celle de Gordon et la nôtre, comprenant les apports algorithmique et graphique nouveaux. Nous montrerons au paragraphe 4 une application de quantification scalaire en imagerie numérique. Signalons que cette application a représenté une stimulation directe pour le développement de cette recherche. Nous terminerons par une conclusion évoquant de nouvelles perspectives.

2 Arbres des classifications, hiérarchie des parties

La notion d'arbre de classification sur un ensemble fini \mathcal{O} d'objets, que nous notons $\mathcal{A}(\mathcal{O})$, à laquelle nous nous sommes jusqu'à présent référés dans notre

travail [13, 14] est celle d'une chaîne non nécessairement maximale dans le treillis des partitions sur l'ensemble \mathcal{O} à classifier. Si on note une telle chaîne sous la forme

$$\{P_0, P_1, \dots, P_k, \dots, P_l\}, \quad (3)$$

P_0 désigne la partition *discrète* où chaque classe contient exactement un élément de \mathcal{O} et P_l , la partition *grossière* qui comprend exactement une seule classe \mathcal{O} . Ainsi, si n désigne le cardinal de \mathcal{O} , P_0 comprend n éléments qui sont des classes *singleton* de la forme $\{x\}$, où x appartient à \mathcal{O} .

Les éléments de P_0 seront représentés par les feuilles de l'arbre, alors que l'unique élément de P_l sera représenté par la racine de l'arbre. L'indice k de P_k , $0 \leq k \leq l$, est celui du k -ième niveau de l'arbre. Les noeuds qui sont figurés à ce niveau représentent les classes de la partition P_k que nous pouvons noter comme suit:

$$P_k = \{\mathcal{O}_k^i \mid 1 \leq i \leq c_k\}. \quad (4)$$

La partition P_{k+1} est moins fine que la partition P_k , chaque classe \mathcal{O}_{k+1}^i va nécessairement s'écrire sous la forme d'une somme ensembliste:

$$\mathcal{O}_{k+1}^i = \sum \{\mathcal{O}_k^{i'} \mid 1 \leq i' \leq d_{k+1}(i)\}, \quad (5)$$

où au moins l'un des entiers $d_{k+1}(i)$, $1 \leq i \leq c_k$, est strictement supérieur à 1.

Du noeud représentant \mathcal{O}_{k+1}^i au niveau $(k+1)$ de l'arbre, sont issues $d_{k+1}(i)$ branches qui le relient aux $d_{k+1}(i)$ noeuds du niveau k qui, respectivement, représentent les classes $\mathcal{O}_k^{i'}$, $1 \leq i' \leq d_{k+1}(i)$. Si nous désignons par a_h^i le noeud représentant la classe \mathcal{O}_h^i , nous dirons, relativement à l'expression (5) ci-dessus que le noeud a_{k+1}^i est père de l'ensemble des noeuds $\{a_k^{i'} \mid 1 \leq i' \leq d_{k+1}(i)\}$ et que, chacun des noeuds $a_k^{i'}$, $1 \leq i' \leq d_{k+1}(i)$, est fils du noeud a_{k+1}^i [cf. Figure 1].

Pour obtenir un tel arbre on utilise, presque toujours, l'algorithme de la construction ascendante hiérarchique, qui opère par agglomérations successives. Nous allons rappeler très synthétiquement cette algorithmique qui est sous jacente à ce qu'on désigne aujourd'hui par la CAH (Classification Ascendante Hiéarchique).

La donnée très généralement présentée pour la CAH dans sa vision la plus classique peut s'exprimer au moyen du triplet suivant :

$$(\mathcal{O}, \mu_{\mathcal{O}}, d), \quad (6)$$

où \mathcal{O} est un ensemble fini d'objets, $\mu_{\mathcal{O}}$, une mesure positive sur \mathcal{O} et d , un indice de distance ou de dissimilarité sur \mathcal{O} .

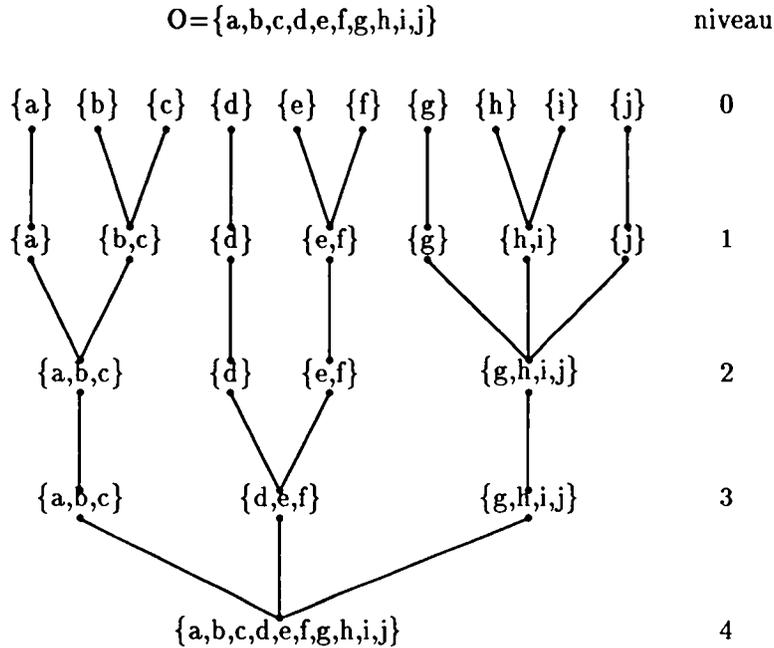


Figure 1 : Arbre de classification

La donnée de (6) permet de déduire la table suivante des indices de distances entre objets munis de poids

$$\{d(x, y), \mu_x, \mu_y \mid \{x, y\} \in P_2(\mathcal{O})\} , \tag{7}$$

où $P_2(\mathcal{O})$ est l'ensemble des paires ou parties à deux éléments de l'ensemble \mathcal{O} des objets.

On étend la notion d de distance (ou dissimilarité) entre éléments de \mathcal{O} à une notion δ de distance (ou dissimilarité) entre parties de l'ensemble \mathcal{O} ; de sorte qu'au triplet (6), on associe le triplet suivant:

$$(\mathcal{P}, \mu_{\mathcal{P}}, \delta) , \tag{8}$$

où \mathcal{P} est l'ensemble des parties de l'ensemble \mathcal{O} des objets et où $\mu_{\mathcal{P}}$ est une mesure positive sur \mathcal{P} déduite de $\mu_{\mathcal{O}}$.

R_+ désignant l'ensemble des nombres réels positifs, la distance ou dissimilarité δ se présente sous la forme de l'application suivante :

$$\delta = [\mathcal{P} \times \mathcal{P}, \mu_{\mathcal{P}}] \longrightarrow R_+ , \tag{9}$$

où

$$[\forall (X, Y) \in \mathcal{P} \times \mathcal{P}] \quad \delta(X, Y) = f[\{d(x, y) \mid (x, y) \in X \times Y\}, (\mu_X, \mu_Y)] , \tag{10}$$

où la fonction f est à définir et où

$$(\forall Z \in \mathcal{P}) \quad \mu_Z = \sum \{ \mu_X \mid X \in Z \} . \quad (11)$$

En réalité, on n'a besoin de la définition de δ que pour *mesurer l'éloignement* entre deux parties disjointes (quelconques) de \mathcal{O} .

L'algorithme de *construction ascendante hiérarchique* – utilisant δ – d'un arbre de classification sur \mathcal{O} , est dans son principe mathématique *trivial* :

“ A chaque pas, réunir les paires de classes qui rendent minimale la valeur de δ ” .

A cette *trivialité* de la définition mathématique ne correspond pas nécessairement une *trivialité* de la définition et de la concrétisation informatique; surtout lorsqu'il s'agit de la classification de ce que l'on appelle les *très gros ensembles* (quelques dizaines de milliers d'éléments) [3, 16, 22].

Contentons nous de mentionner le premier problème qui est celui de la gestion de la table des indices δ entre classes formées pour aboutir à la construction ascendante hiérarchique de l'arbre des classifications. Si k est un niveau de l'arbre, nous dirons que l'état du système est défini par le couple (T_k, μ_k) , où T_k est la table des indices δ d'*éloignement* entre classes formées au niveau k et où μ_k est la mesure sur l'ensemble de ces classes. Il est alors très important d'un point de vue *calcul* d'avoir une formule – dite de réactualisation – de la forme :

$$(T_{k+1}, \mu_{k+1}) = \varphi(T_k, \mu_k) , \quad (12)$$

qui donne l'état du système au niveau $(k + 1)$, après la fusion des classes les plus proches, détectées au niveau k sur la base de T_k , T_0 est la table des indices δ entre classes *singleton*, fournie à partir de (7).

Jusqu'à récemment toutes les transformations φ étaient données pour l'agrégation de deux classes seulement [8, 10, 27]. Nous avons été amenés à proposer, pour les indices δ les plus classiques, des formules de réactualisation en cas d'agrégations multiples se produisant à un même niveau de l'arbre [19]. Ce cas est bien réel et se produit beaucoup plus fréquemment qu'on ne le croit; il s'est d'ailleurs présenté dans le cas de l'application qui nous concerne [cf. §4].

Dans notre approche classificatoire des données fondée sur l'Analyse de la Vraisemblance des Liens (AVL), nous nous ramenons bien à la démarche ci-dessus; mais pour une famille d'indices δ très spécifique. La construction d'une telle famille dépend du problème posé qui peut bien sûr être celui de la classification de l'ensemble \mathcal{O} des objets; mais qui peut également être celui de l'ensemble \mathcal{V} des variables de description. Elle dépend aussi du type mathématico-logique du tableau des données (\mathcal{O} bjets \times \mathcal{V} ariables), qui peut être quelconque.

Cependant, dans notre optique, nous ne retenons de la structure d'arbre extraite que son caractère ordinal. En effet, nous représentons un arbre ainsi construit de classification – définissant une chaîne totalement ordonnée par finesse décroissante de partitions sur \mathcal{O} – au moyen de ce que nous appelons une *préordonnance ultramétrique* sur \mathcal{O} [13, 14]. Il s'agit d'un préordre total sur l'ensemble $P = P_2(\mathcal{O})$ des paires ou parties à deux éléments de \mathcal{O} , où le rang d'une paire $\{x, y\}$ est d'autant plus grand que le niveau de première réunion est plus petit. Si ω_u désigne ce préordre total et si ρ_u est une fonction rang traduisant un tel préordre total, on a :

$$[\forall \{x, y, z\} \in P_3(\mathcal{O})] , \rho_u \geq \min \{ \rho_u(x, y), \rho_u(y, z) \} . \quad (13)$$

Une autre représentation – très communément utilisée – est de nature numérique. Cette dernière se conçoit clairement à partir du précédent algorithme de construction ascendante hiérarchique d'un arbre de classification. Soit $\{x, y\}$ une paire d'objets donnée, si X (resp. Y) est la classe maximale (en cardinal) qui contient l'objet x sans contenir y (resp. l'objet y sans contenir x), on pose

$$h(x, y) = \delta(X, Y) . \quad (14)$$

En d'autres termes, conformément à ci-dessus, si k , $1 \leq k \leq l$, est le premier niveau de l'arbre de classification où les deux objets x et y se trouvent réunis pour la première fois dans la partition P_k , $h(x, y)$ est la valeur minimale de δ atteinte sur le système exhaustif de classes de P_{k-1} [cf. (3)]. Une telle valeur est appelée *hauteur* ou *indice* du niveau k . Un arbre de classification dont les niveaux sont ainsi évalués, est souvent appelé *dendrogramme*.

“Oubliant” la notion de niveau k , $1 \leq k \leq l$, on peut associer univoquement, à l'arbre de classification ci-dessus interprété, un ensemble de parties de \mathcal{O} qui forme une hiérarchie \mathcal{H} . Rappelons qu'un ensemble de parties de \mathcal{O} forme une hiérarchie \mathcal{H} , si

$$\begin{aligned} (i) \quad & (\forall x \in \mathcal{O}) \implies \{x\} \in \mathcal{H} ; \\ (ii) \quad & \mathcal{O} \in \mathcal{H} ; \\ (iii) \quad & [\forall (A, B) \in \mathcal{H} \times \mathcal{H}] \implies A \cap B \in \{\emptyset, A, B\} ; \end{aligned} \quad (15)$$

en d'autres termes, deux parties de \mathcal{O} , éléments de \mathcal{H} , sont soit disjointes, soit incluses l'une dans l'autre.

Dans ces conditions, se situant au niveau de l'ensemble des parties de \mathcal{O} , on a :

$$\mathcal{H} = \bigcup \{ P_k \mid 0 \leq k \leq l \} . \quad (16)$$

Conformément à ci-dessus, où l'arbre $\mathcal{A}(\mathcal{O})$ a été obtenu par un algorithme de construction ascendante hiérarchique et dans la mesure où – comme il se doit pour les *bons* critères – δ monotone (i.e. l'indice du niveau k est

supérieur à celui du niveau $(k-1)$, $0 \leq k \leq l$), on peut valuer – on dit encore *indicer* – la hiérarchie \mathcal{H} à partir de δ . Plus précisément, si l'élément A de \mathcal{H} a pour fils l'ensemble des éléments $\{A_i \mid 1 \leq i \leq d\}$ [cf. (5) ci-dessus], on posera pour la valuation $h(A)$ de A , la valeur minimale de δ sur l'ensemble des parties disjointes $\{A_i \mid 1 \leq i \leq d\}$ de \mathcal{O} .

Inversement, certains auteurs [2, 6] mettent en avant la donnée d'une hiérarchie $\mathcal{H}(\mathcal{O})$ sur l'ensemble \mathcal{O} des objets à partir des trois axiomes (i), (ii), (iii) ci-dessus. Une telle hiérarchie est indicée s'il existe une valuation h monotone; c'est à dire, telle que,

$$[\forall (A, B) \in \mathcal{H} \times \mathcal{H}; A \cap B \neq \emptyset \text{ et } A \subset B \iff h(A) \leq h(B)]. \quad (17)$$

Une hiérarchie indicée $[\mathcal{H}(\mathcal{O}), h]$ est représentable par un *diagramme d'arbre* dont la valuation h représente la fonction hauteur. Un diagramme d'arbre [cf. Figure 2] est la représentation graphique d'un arbre dont les branches sont sans croisement et dont nous allons donner une définition constructive pour la représentation de $[\mathcal{H}(\mathcal{O}), h]$. A cette fin, désignons par

$$\{h_k \mid 1 \leq k \leq l\},$$

la suite strictement croissante des valeurs positives de h , et considérons le plan de la feuille, rapportée à *gauche* à un axe vertical ascendant et *en bas*, à un axe horizontal orienté de gauche à droite. La fonction hauteur sera représentée sur l'axe vertical.

Les feuilles de l'arbre qui représentent les éléments de \mathcal{O} , occuperont l'horizontale de hauteur $h = 0$; alors que les éléments de $\mathcal{H}(\mathcal{O})$ de valeur h_k , seront respectivement représentés par des noeuds sur l'horizontale de cote h_k , $1 \leq k \leq l$. Le problème du dessin du diagramme sera résolu dès lors qu'on peut déduire la représentation des noeuds de hauteur h_k , $1 \leq k \leq l$, de celle des noeuds maximaux immédiatement inférieurs. Chacun de ces noeuds maximaux surtend une partie de \mathcal{O} , et l'ensemble de ces parties détermine une partition de \mathcal{O} .

Soit B un élément de $\mathcal{H}(\mathcal{O})$ (partie de \mathcal{O}) de hauteur h_k [$h(B) = h_k$]. Considérons l'ensemble de tous les éléments maximaux $\{B_i \mid 1 \leq i \leq j\}$ qui sont strictement inférieurs à B , pour l'inclusion entre parties de \mathcal{O} . Dans ces conditions, les B_i sont mutuellement incomparables, correspondants à des parties disjointes de \mathcal{O} . Ils ont d'autre part B comme père direct. Les restrictions respectives du diagramme à chacun des B_i sont disjointes. Dans ces conditions, on permutera latéralement ces restrictions de façon à former une suite de diagrammes locaux sur, respectivement, la suite des B_i , $1 \leq i \leq j$. On reliera alors chacun des noeuds B_i , $1 \leq i \leq j$, au noeud père qu'on situera sur l'horizontale de hauteur h_k et – latéralement – de façon médiane entre le premier et le dernier des B_i .

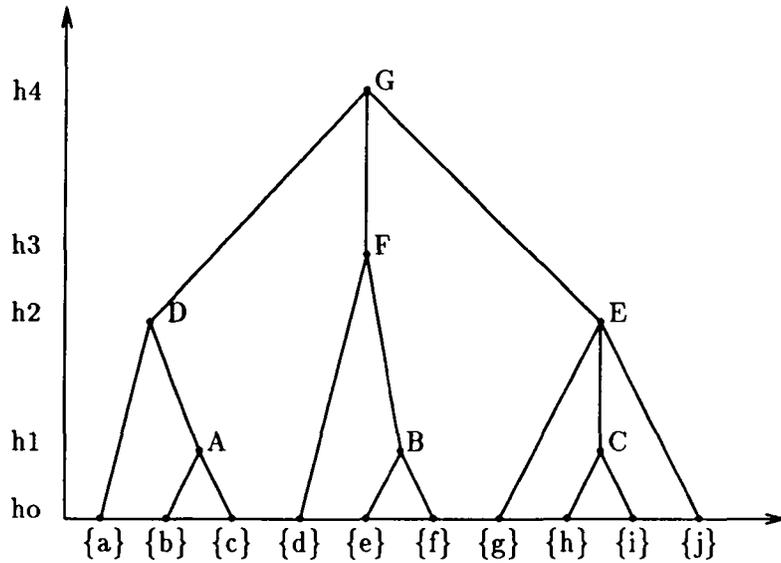


Figure 2 : Diagramme d'arbre

On poursuit de même pour les autres éléments de $\mathcal{H}(\mathcal{O})$, d'indice h_k . Si B' est l'un de ces éléments, on a nécessairement $B \cap B' = \emptyset$.

A un diagramme d'arbre - dont la donnée est équivalente à une hiérarchie valuée - sur un ensemble fini \mathcal{O} donné, on peut associer un arbre de classification [cf. (3) ci dessus]. Il suffit en effet de considérer l sécantes horizontales de hauteurs respectives :

$$h_0 + \epsilon, h_1 + \epsilon, \dots, h_k + \epsilon, \dots, h_{l-1} + \epsilon,$$

où on a posé $h_0 = 0$ et où ϵ est un nombre positif tel que :

$$\epsilon < \min\{(h_k - h_{k-1}) \mid 1 \leq k \leq l\}. \quad (18)$$

En dessous de la $(k + 1)$ ème sécante se trouve déterminée la partition de niveau k , P_k , qui est obtenue à partir des noeuds maximaux situés en dessous de cette sécante; certains de ces noeuds peuvent correspondre à des feuilles. Ainsi, relativement à l'exemple de la figure 2 ci-dessus, la partition définie par l'horizontale d'ordonnée $h_2 + \epsilon$, est : $\{\{a, b, c\}, \{d\}, \{e, f\}, \{g, h, i, j\}\}$; les noeuds maximaux correspondants étant $\{d\}$, B , D et E . Chaque noeud maximal surtend une partie de \mathcal{O} et ces différentes parties forment une partition de \mathcal{O} .

Si à un arbre de classification $\mathcal{A}(\mathcal{O})$ [cf. (3) et Figure 1] correspond univoquement une hiérarchie $\mathcal{H}(\mathcal{O})$ de parties. Inversement, à une hiérarchie de parties correspond, ce que nous appellerons un *arbre de parties*. Une suite de

noeuds se situant sur une même chaîne de cet arbre, représente une suite de parties de \mathcal{O} [éléments de $\mathcal{H}(\mathcal{O})$] strictement monotone quant à l'inclusion. Comme dans le cas d'un arbre de classification, la racine de cet arbre représente \mathcal{O} et les feuilles, les éléments de \mathcal{O} . Mais, la longueur (en termes de nombre d'arêtes) d'un chemin complet partant de la racine et aboutissant à une feuille, n'est pas invariant. Si nous désignons par $\mathcal{A}_{\mathcal{H}(\mathcal{O})}$ l'arbre de parties associé à $\mathcal{H}(\mathcal{O})$, nous allons donner un exemple avant de proposer un algorithme de résolution permettant le passage entre $\mathcal{H}(\mathcal{O})$ et $\mathcal{A}_{\mathcal{H}(\mathcal{O})}$.

Considérons la hiérarchie de parties suivante sur l'ensemble $\mathcal{O} = \{a, b, c, d, e, f, g, h, i, j\}$ à dix éléments :

$$\begin{aligned} \mathcal{H}(\mathcal{O}) = \{ & H_0 = \mathcal{O}, H_1 = \{d, e, f, h, i, j\}, h_2 = \{d, e, f\}, H_3 = \{g, h, i, j\}, \\ & H_4 = \{e, f\}, H_5 = \{h, i\}, H_6 = \{a, b, c\}, h_7 = \{a\}, H_8 = \{b\}, H_9 = \{c\}, \\ & H_{10} = \{d\}, H_{11} = \{e\}, H_{12} = \{f\}, H_{13} = \{g\}, H_{14} = \{h\}, H_{15} = \{i\}, \\ & H_{16} = \{j\} \}. \end{aligned} \tag{19}$$

L'arbre de parties $\mathcal{A}_{\mathcal{H}(\mathcal{O})}$ associé est représenté à la figure 3 ci-dessous, où nous notons directement les feuilles de l'arbre à partir des éléments de \mathcal{O} .

Plus brièvement, désignons par $T = \mathcal{A}_{\mathcal{H}(\mathcal{O})}$ l'arbre de parties associé à $\mathcal{H}(\mathcal{O})$ et par a un noeud courant dont le numéro $n_T(a)$ représente le niveau où il apparaît. $card_T(a)$ est le cardinal de la partie de \mathcal{O} , élément de la hiérarchie $\mathcal{H}(\mathcal{O})$, représentée par le noeud a . Le prédécesseur immédiat dans T de a est noté $pred_T(a)$ et on a, bien sûr,

$$card_T[pred_T(a)] \geq card_T(a)$$

Enfin, conformément à [9], on indiquera par

$$(a.1, a.2, \dots, a.i, \dots, a.q)$$

la suite de ses descendants immédiats – dont q est le nombre – qui est ordonnée conformément à:

$$card_T(a.1) \geq card_T(a.2) \geq \dots \geq card_T(a.i) \geq \dots \geq card_T(a.q).$$

Dans ces conditions, l'algorithme de construction de l'arbre T se déroule de façon récursive comme suit :

Etape 0 : on part de la racine H_0 ;

faire jusqu'à arrêt;

si tous les éléments de $\mathcal{H}(\mathcal{O})$ sont visités, **alors arrêt** ;

sinon, passer à l'étape 1.

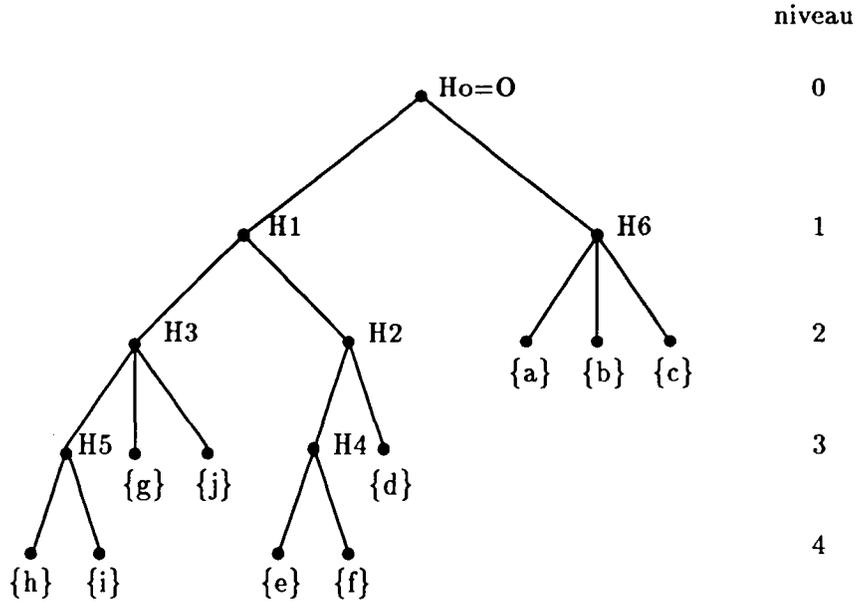


Figure 3 : Arbre de parties: $T = \mathcal{A}_{\mathcal{H}}(\mathcal{O})$

Etape 1: *traiter le noeud courant a ;*

si $\text{card}_T(a) = 1$, auquel cas a est une feuille, alors passer à l'étape 2;
 sinon, c'est que $\text{card}_T(a) > 1$, auquel cas traiter le descendant immédiat de a : prendre le descendant $a.i$ de a dont le cardinal est supérieur à tous les cardinaux de ses $(q - 1)$ frères $a.j$ avec $j = 1, q$ et $j \neq i$, et le mettre dans a ; puis retourner à l'étape 1.

Etape 2 : *traiter le prédécesseur ;*

enlever la partie singleton correspondante à la feuille a , de $\text{pred}_T(a)$ et dans a , mettre $\text{pred}_T(a)$; retourner à l'étape 1.

arrêt .

Si à un arbre de classifications $\mathcal{A}(\mathcal{O})$ [cf. (3) et Figure 1] correspond univoquement une hiérarchie $\mathcal{H}(\mathcal{O})$ de parties et par voie de conséquence, un arbre de parties $\mathcal{A}_{\mathcal{H}}(\mathcal{O})$. Inversement, à une hiérarchie de parties $\mathcal{H}(\mathcal{O})$ peuvent correspondre plusieurs arbres de classifications compatibles avec la hiérarchie; c'est-à-dire, la redonnant à partir de la relation (15). Ces derniers peuvent être obtenus de façon récursive et énumérés en conséquence. L'algorithme de génération peut être le suivant :

A partir de la racine de l'arbre $\mathcal{A}_{\mathcal{H}}(\mathcal{O})$, on considère les chaînes - ordonnées de façon décroissante - également les plus longues, qu'on retient.

Il s'agit, dans le cas de la figure 3 des chaînes finissant avec les feuilles $\{h\}$, $\{i\}$, $\{e\}$ et $\{f\}$. La profondeur commune de ces chaînes sera la profondeur de l'arbre des classifications qu'on désignera par l . Tout noeud issu de l'une de ces chaînes soutend un arbre de parties sur un sous ensemble de \mathcal{O} . Ainsi dans l'exemple de la figure 3, il s'agit de $\{H_6 = \{a, b, c\}, \{a\}, \{b\}, \{c\}\}$; mais, il peut aussi s'agir d'arbres réduits à une feuille tels que $\{\{g\}\}, \{\{j\}\}$ ou $\{\{d\}\}$.

Imaginons qu'un noeud a_{k+1} surtendant un sous arbre de parties, soit issu d'un noeud a_k qui se trouve à une distance k – en termes de nombres d'arêtes – de la racine sur une chaîne maximale. Ainsi, dans l'exemple de la figure 3, le noeud $\{d\}$ (resp. H_6) est issu d'un noeud qui est à une distance 2 (resp. 0) de la racine. La longueur de la chaîne partant de la racine et aboutissant en a_{k+1} est $(k + 1)$. Si s est la longueur maximale atteinte par une chaîne du sous arbre de racine a_{k+1} , on :

$$k + 1 + s < l,$$

il y a donc lieu de créer des répétitions parmi les $(s + 1)$ niveaux du sous arbre de façon à totaliser l niveaux à partir de la racine H_0 ; c'est-à-dire, exactement, $(l - k - 1 - s)$ répétitions. Il y a $\binom{l-k-1}{s}$ façons différentes de le faire.

On poursuit de même par rapport aux sous arbres du sous arbre de racine a_{k+1} et, ainsi de suite, récursivement, jusqu'à obtenir tous les arbres possibles de classification de profondeur l . Relativement à l'exemple simple de la figure 3 ci-dessus, pour H_6 , il y a $\binom{4-0-1}{1} = 3$ arbres de classification de profondeur 4; ces derniers correspondent aux niveaux 1, 2 ou 3 où on placera H_6 , puisque les feuilles d'un niveau donné, seront nécessairement répétées jusqu'au dernier niveau.

3 Approches dans la réduction de l'arbre

3.1 Préambule

Dans ce paragraphe nous développerons de façon plus précise et plus technique différents aspects de la présentation générale (cf. §1). Nous commencerons par exprimer au paragraphe 3.2. en quoi consiste l'approche de Gordon [6]. Au paragraphe 3.3. nous exprimerons notre approche dans son état de développement [13, 14, 15], précédant immédiatement la contribution originale qui est à l'origine de la présente publication. Au paragraphe 3.4., nous présenterons la nouvelle idée algorithmique, fondée sur la détection des noeuds *significatifs*, de présentation d'un arbre condensé de parties issu de

l'arbre des classifications. Cette idée est rendue opérationnelle, comme en témoigne un exemple traité ainsi que l'application en quantification.

3.2 Approche de A.D. Gordon

La structure de base considérée est une hiérarchie indicée de parties sur un ensemble fini \mathcal{O} d'objets [cf. (15), (17) et Figure 2, §2]. Nous avons vu qu'à une telle structure on peut associer un arbre valué de classifications. Nous avons également déjà exprimé au paragraphe 1 ci-dessus le principe général de cette approche où il s'agit de commencer par déterminer une partition de l'ensemble de noeuds, dont – rappelons le – chacun représente un élément de la hiérarchie $\mathcal{H}(\mathcal{O})$ de parties.

Deux idées algorithmiques de découpage des branches de l'arbre sont considérées. La première – appelée *globale* – consiste à accorder une valeur absolue à la notion de hauteur [cf. (17) §2] donnant ainsi un sens à la comparaison entre $h(C)$ et $h(D)$, quand C et D , éléments de $\mathcal{H}(\mathcal{O})$, sont deux parties disjointes. On considère alors sur le diagramme de l'arbre [cf. Figure 2, §2] ($g - 1$) horizontales, où la hauteur d'une même horizontale est strictement comprise entre deux valeurs consécutives de h , prises sur $\mathcal{H}(\mathcal{O})$. On obtient ainsi une partition en g classes de l'ensemble des noeuds, la e -ème étant déterminée par le sous ensemble des noeuds qui se situent entre la $(e - 1)$ ème horizontale et la e -ème horizontale. De cette j -ème classe, $1 \leq e \leq g$, on retient – comme nous l'avons déjà exprimé au paragraphe 1 – les noeuds maximaux qui sont les plus proches de la racine; ces derniers sont appelés noeuds *favoris*. La réduction de la hiérarchie indicée qui en résulte correspond à remplacer la fonction hauteur h par une fonction hauteur p qui ne prend que certaines valeurs de h , dont la plus petite et la plus grande. On peut interpréter la donnée des fonctions h et p au niveau de l'ensemble $P_2(\mathcal{O})$ des paires d'objets distincts de \mathcal{O} ; dans ce cas h et p définissent deux distances ultramétriques sur \mathcal{O} , A.D. Gordon considère alors une fonction de perte ou d'inadéquation de la forme:

$$S(h, p, g) = \sum_{1 \leq i < j \leq n} w_{ij} \sigma_{ij}(h, p, g), \quad (20)$$

où $\{w_{ij} \mid 1 \leq i < j \leq n\}$ est une fonction de pondération sur $P_2(\mathcal{O})$ à discuter et où $\sigma_{ij}(h, p, g)$ est la perte résultante de remplacement de h_{ij} par p_{ij} .

On doit noter {mais, ce n'est pas fait dans [6]} que l'on a nécessairement:

$$[\forall(i, j), 1 \leq i < j \leq n], h_{ij} \leq p_{ij}; \quad (21)$$

de sorte qu'on peut proposer pour σ les deux fonction suivantes:

$$\sigma_{ij}^C(h, p, g) = (p_{ij} - h_{ij}) \quad (22)$$

ou

$$\sigma_{ij}^A(h, p, g) = (p_{ij} - h_{ij})^2, \quad (23)$$

pour tout (i, j) , $1 \leq i < j \leq n$.

Le caractère additif du critère $S(h, p, g)$ [cf. (20) ci-dessus] sur la suite croissante en hauteur des noeuds, permet de considérer, pour g fixé et à la manière de W.D. Fisher [4], l'usage de la programmation dynamique, pour trouver la solution optimale au problème posé. Toutefois, le choix de g reste à discuter. On peut signaler ici et entre parenthèses, que la programmation dynamique se retrouve à divers endroits de l'analyse des données qualitatives [11], [[14], chap. 9], ...

En retenant toujours comme noeuds *favoris* les noeuds maximaux au sens de la relation d'ordre dont se trouve $\mathcal{H}(\mathcal{O})$; c'est-à-dire, encore une fois, les plus proches de la racine de l'arbre, le second procédé de découpage – dit *local* – consiste à trancher des branches de l'arbre. En constatant que la coupure d'une branche de l'arbre donne lieu à un sous arbre descendant, dont la racine déterminera précisément un noeud *favori*, on se rend compte qu'il y a lieu de couper $(g - 1)$ branches, pour obtenir g classes de noeuds. Chaque classe de noeuds a la structure d'un sous arbre.

Il n'est alors plus possible d'utiliser la programmation dynamique pour rendre minimale la valeur du critère d'inadéquation. Pour g fixé, il faut trouver une bonne heuristique. D'autre part, le nombre g de classes de noeuds reste à discuter.

3.3 Niveaux et noeuds “significatifs”

Désignons ici par E l'ensemble des entités à classifier et ceci pour bien marquer que E peut être l'ensemble \mathcal{O} des objets ou bien l'ensemble \mathcal{V} des variables de description.

$F = P_2(E)$ indiquera l'ensemble des paires ou parties à deux éléments de E . Nous allons supposer que sur E se trouve déjà établie une notion de similarité \mathcal{S} . Cette dernière correspond dans la quasi totalité des cas en classification des données, à une valuation numérique de F . Si la donnée est un indice de dissimilarité \mathcal{D} sur E , on posera tout simplement $\mathcal{D} = -\mathcal{S}$.

La donnée d'une partition π sur E peut être interprétée au niveau de F , au moyen d'un préordre ayant un caractère spécifique, à deux classes que nous notons $R(\pi)$ et $S(\pi)$. $R(\pi)$ [resp. $S(\pi)$] est l'ensemble des paires réunies (resp. séparées). On posera, pour l'ordre quotient:

$$S(\pi) < R(\pi) \quad (24)$$

Ainsi, toute paire $\{x, y\}$ séparée (formée de composantes éloignées du point de vue de la partition) précède strictement toute paire $\{z, t\}$ réunie

(formée de composants proches du point de vue de la partition). On peut de façon très explicite, donner l'expression de $R(\pi)$ [resp. $S(\pi)$] par rapport aux classes de la partition π [17]. nous représentons un tel préordre total et par conséquent, la partition π , au niveau de $F \times F$, par le rectangle :

$$S(\pi) \times R(\pi). \quad (25)$$

Un principe général de notre approche est de se ramener à la comparaison de deux structures combinatoires de même type. A cette fin, on considère la préordonnance totale sur E , associée à la similarité \mathcal{S} . Il s'agit d'un préordre total sur F , défini comme suit :

$$\{\forall (p, q) \in F \times F\}, p \leq q \Leftrightarrow \mathcal{S}(p) \leq \mathcal{S}(q). \quad (26)$$

Nous notons $\omega(E)$ cette structure ordinale. Nous allons alors considérer deux cas, selon que $\omega(E)$ est un ordre ou préordre total sur F , pour donner le principe de construction de la statistique globale $S[\omega(E), \pi]$. Le deuxième cas est celui où la fonction similarité \mathcal{S} n'est pas injective dans sa définition sur F .

(i) Cas où la donnée est une ordonnance totale sur E .

$\omega = \omega(E)$ est ici un ordre total et strict sur F . L'élaboration du critère $S(\omega, \pi)$ qui est un indice de proximité, d'association ou d'adéquation entre ω et π , obéit à un schéma général que nous avons dégagé et adopté pour la comparaison de deux relations d'un même type combinatoire, quelconque, sur un ensemble fini [18].

Le point de départ de cette construction est un indice brut de proximité qui s'écrit :

$$s(\omega, \pi) = \text{card} \{gr(\omega) \cap [S(\pi) \times R(\pi)]\} \quad (27)$$

où,

$$gr(\omega) = \{(p, q) \mid (p, q) \in F \times F, (p \leq q) \wedge \neg(q \leq p) \text{ pour } \omega\} \quad (28)$$

En introduisant la fonction *rang* sur F , associée à ω :

$$k(p) = \text{card}\{q \mid q \in F, q \leq p \text{ pour } \omega\}, \quad (29)$$

nous montrons [13, 14] que $s(\omega, \pi)$ peut se mettre sous la forme :

$$s(\omega, \pi) = \sum \{k(p) \mid p \in R(\pi)\} - r(r+1)/2, \quad (30)$$

où

$$r = \text{card}[R(\pi)] = \sum \{n_j(n_j - 1)/2 \mid 1 \leq j \leq c\}, \quad (31)$$

où c est le nombre de classes de la partition π et où $t = (n_1, n_2, \dots, n_j, \dots, n_c)$ est son type; c'est-à-dire, la suite - qu'on peut prendre décroissante - des cardinaux de ses classes.

A des fins de normalisation statistique, on introduit alors une hypothèse d'absence de liaison (h.a.l), où ici, il est naturel de fixer ω et d'associer à π , une partition aléatoire π^* dans l'ensemble $\mathcal{P}(n; t)$, des partitions sur E , de même type t et muni d'une probabilité uniformément répartie.

Nous démontrons qu'en général, l'indice brut aléatoire $s(\omega, \pi^*)$ a une loi limite normale, de moyenne et de variance respectives :

$$\mathcal{E}[s(\omega, \pi^*)] = \frac{rs}{2}, \quad (32)$$

et

$$\text{var}[s(\omega, \pi^*)] \cong \frac{rs(f+1)}{12}, \quad (33)$$

où $s = \text{card}[S(\pi)]$ et où $f = r + s = \text{card}(F)$.

Alors que le second membre de (32) correspond à une valeur exacte, celui de (33) est une valeur approchée. Nous donnerons ci-dessous une expression plus précise de la variance dans un contexte mathématique plus général.

Dans ces conditions, $S(\omega, \pi)$ se met sous la forme :

$$S(\omega, \pi) = \frac{s(\omega, \pi) - (rs/2)}{\sqrt{rs(f+1)/12}}. \quad (34)$$

On remarquera qu'en vertu de (30), la normalisation statistique (centrage et réduction) de $s(\omega, \pi)$ est équivalente à celle de l'indice

$$t(\omega, \pi) = \sum \{k(p) \mid p \in R(\pi)\}, \quad (35)$$

qui s'en déduit par une constante additive dans le cadre de l'h.a.l..

(ii) Cas où la donnée est une préordonnance totale sur E .

Désignons par $\{F_l \mid 1 \leq l \leq m\}$ la suite ordonnée des classes sur F , fournies par la préordonnance $\omega(E)$, dont nous désignons la suite des cardinaux par $\{f_l \mid 1 \leq l \leq m\}$; on a pour l'ordre quotient :

$$F_1 < F_2 < \dots < F_l < \dots < F_m. \quad (36)$$

L'adaptation la plus naturelle et la plus aisée de l'indice brut se fait à partir de l'expression (35) ci-dessus, avec une définition judicieuse de la fonction ordinale $k(p)$ sur F . On considérera la notion de *rang moyen* définie comme suit :

$$k(p) = \sum_{1 \leq j \leq l-1} f_j + \frac{1}{2}(f_l + 1), \quad (37)$$

pour tout p appartenant à F_l et tout l , compris entre 1 et m . De la sorte, $k(p)$ reprend sa valeur initiale (29) si chaque classe F_l ne contient plus qu'un seul élément et d'autre part, on a bien la conservation de la somme des rangs :

$$\sum \{k(p) \mid p \in F\} = \frac{1}{2}f(f+1). \quad (38)$$

De la sorte, la moyenne et la variance de l'indice brut aléatoire $t(\omega, \pi^*)$ sont respectivement :

$$\mathcal{E}[t(\omega, \pi^*)] = \frac{rs}{2} \quad (39)$$

et, au facteur multiplicatif $(f^2 - 1)/12$ près,

$$\begin{aligned} \text{var}[(t(\omega, \pi^*))] = & f \left(\sum_{1 \leq j \leq c} \pi_j^2 \right) \left[1 - \left(\sum_{1 \leq j \leq c} \pi_j^2 \right) \right] + \\ & \left(\sum \{ \gamma(p)\gamma(p') \mid (p, p') \in G \} \right) \left[\left(\sum_{1 \leq j \leq c} \pi_j^3 \right) - \left(\sum_{1 \leq j \leq c} \pi_j^2 \right)^2 \right]. \end{aligned} \quad (40)$$

Dans cette dernière expression (40), $\pi_j = n_j/n$ représente la proportion d'éléments de la j -ème classe, G est l'ensemble des couples de paires ayant une seule composante commune; c'est-à-dire, de la forme $(\{x, y\}, \{x, z\})$, où x, y et z sont mutuellement distinctes [$\text{card}(G) = n(n-1)(n-2)$], enfin, $\gamma(p)$ s'exprime comme suit :

$$\gamma(p) = \frac{k(p) - \bar{k}}{\sqrt{\text{var}(k)}}, \quad (41)$$

où \bar{k} et $\text{var}(k)$ sont respectivement la moyenne et la variance de la distribution $\{k(p) \mid p \in F\}$. On a $\bar{k} = (f+1)/2$ et $\text{var}(k) = (f^2 - 1)/12$ dans le cas (i) ci-dessus, où (40) fournit - après multiplication par $\text{var}(k)$ - une valeur plus précise que (33) ci-dessus.

D'où, par centrage et réduction de (35), on obtient $S(\omega, \pi)$.

(iii) Cas où on retient la donnée d'un indice numérique de similarité.

Compte tenu de la forme (35) du critère brut, il semble naturel de directement considérer celui :

$$\sum \{ \mathcal{S}(p) \mid p \in R(\pi) \}, \quad (42)$$

où \mathcal{S} est la fonction similarité, définie sur F , que nous avons ci-dessous présentée. En effet, tout se passe comme si, dans le passage de (42) à (35), la mesure attachée à F et reflétant les ressemblances entre éléments de E , était définie par la fonction de répartition de la distribution de \mathcal{S} sur F . Si on prend \mathcal{S} , $\gamma(p)$ devient

$$\gamma(p) = \frac{\mathcal{S}(p) - \alpha}{\lambda}, \quad (43)$$

où α et λ sont respectivement la moyenne et l'écart type de la distribution $\{\mathcal{S}(p) \mid p \in F\}$. La forme du critère d'adéquation est toujours

$$S(\mathcal{S}, \pi) = \frac{1}{\sqrt{\frac{rs}{f-1}}} \sum \{ \gamma(p) \mid p \in R(\pi) \}; \quad (44)$$

mais on peut considérer une expression plus précise du dénominateur, obtenue à partir de (40).

Dans le cas où la mesure de similarité S a un fondement statistique clair, le critère conçu à partir de (42) tient le plus étroitement compte de l'information initiale. Cependant des raisons importantes justifient l'utilisation du critère fondé sur la préordonnance totale [cf. (34)] ou l'extension qui s'en déduit en (ii). La première qui résulte des travaux de R.N. Shepard [25] et de J.P. Benzecri [1], conçus dans un cadre géométrique, correspond à exprimer que de la sorte, on compare ce qu'il y a lieu de comparer; c'est-à-dire, dans notre langage, deux structures combinatoires de même type. D'autre part, nous avons pu déceler des propriétés de stabilité intéressantes de la préordonnance associée au choix d'un indice de similarité [13], [[14] chap 2].

Reprenons ce que nous avons exprimé au paragraphe 1 dans la présentation générale et notamment les expressions (1) et (2), où d'ailleurs, notamment pour des raisons de calcul (on évite ainsi le tri permettant l'obtention de $\omega(E)$), on pourra remplacer $\omega(E)$ par $S(E)$ [cf. (iii) ci-dessus]. Mais, encore une fois, on préférera autant que possible utiliser comme argument du critère S , directement la préordonnance totale $\omega(E)$. Relativement à (1) du paragraphe 1, posons

$$S_i = S[\omega(E), \pi_i] \text{ (resp. } S[S(E), \pi_i]) \quad (45)$$

et soit la fonction numérique sur l'ensemble $\{1, 2, \dots, i, \dots, I\}$ des niveaux de l'arbre:

$$i \longmapsto S_i \quad (46)$$

Une telle fonction a une tendance globale à être d'abord croissante jusqu'à atteindre avec une pente proche de l'horizontale, son maximum; d'où, après un petit palier, la fonction décroît brutalement. Tout se passe comme si, dans la suite des agrégations, on cherchait à atteindre la "meilleure" classification: celle, qui en un faible nombre de classes, constitue le résumé le plus général. A partir de ce dernier, toute agrégation ne peut être que contre nature et se trouve par conséquent taxé par une forte baisse de S .

La situation que nous venons de décrire du comportement de S , où il se présente un seul mode dominant et net de la distribution $\{S_i \mid 1 \leq i \leq I\}$, est la plus fréquente. Mais, nous avons pu observer des cas où il pouvait se présenter deux ou trois modes consistants qui indiquent des états très forts d'équilibre dans la synthèse automatique.

Il faut également signaler que durant un même intervalle – de la suite croissante de niveaux – de croissance globale de S , certaines réunions de classes peuvent être accompagnées par une diminution locale de S . Ce qui exprime, que, considéré globalement, la partition obtenue après une telle

réunion de classes, est moins en accord avec la préordonnance totale $\omega(E)$ [resp. $\mathcal{S}(E)$], que celle qui la précède ; laquelle sera retenue s'il s'agit de choisir entre les deux.

Si le comportement de la *Statistique globale* S [cf. Figure 4] permet de reconnaître les quelques niveaux les plus *significatifs*, le comportement de la *Statistique locale* τ [cf.(2) §1 et Figure 5] permet de détecter les nœuds "significatifs" de l'arbre des classifications. Considérons la fonction numérique sur l'ensemble $\{2, \dots, i, \dots, I\}$ des niveaux de l'arbre :

$$i \longmapsto \tau_i = S_i - S_{i-1} \quad (47)$$

τ_i représente le taux d'accroissement du critère dans le passage entre la partition π_{i-1} de niveau $(i-1)$ et celle π_i de niveau i .

Une très riche expérience a montré que la distribution $\{\tau_i \mid 2 \leq i \leq I\}$ de τ , le long de la suite des niveaux est telle, que sa valeur augmente lorsqu'une classe en cours de formation, se confirme sémantiquement, et décroît sensiblement, devant l'arrêt de constitution d'une classe ayant quelque consistance, au profit de la naissance ou de l'accroissement embryonnaire d'une autre classe. Les niveaux associés aux maxima locaux de la distribution $\{\tau_i \mid 2 \leq i \leq I\}$, correspondent par conséquent à des niveaux d'achèvement de classes. Nous retiendrons donc comme les plus significatifs, les nœuds définis à ces niveaux. Et, jusqu'à présent, c'est l'arbre réduit à ces niveaux que nous avons coutume de représenter graphiquement. Les nœuds significatifs sont – dans cette représentation – marqués d'une étoile accompagnant le numéro du niveau où ils apparaissent ; alors que les autres nœuds sont simplement marqués du numéro du niveau où ils apparaissent.

Les minima locaux de $\{\tau_i \mid 2 \leq i \leq I\}$ sont également instructifs pour l'expert des données. Ils expriment à différents degrés de synthèse des oppositions relatives entre sous classes d'une plus large classe. L'interprétation d'un minimum local de τ , exige une compréhension d'autant plus fine des données que le niveau i est plus élevé.

Ainsi, notre méthode de reconnaissance des niveaux et des nœuds significatifs d'un arbre des classifications ne fait aucune référence à un modèle extérieur aux données. Au contraire, la démarche de P.H.A. Sneath [26] qui consiste simplement à proposer des tests de séparation entre deux classes, fait précisément appel à des modèles paramétriques exogènes aux données dont la pertinence doit être justifiée.

Les associations étant jugées de l'intérieur les unes par rapport aux autres, notre approche fournit une interprétation dynamique ; c'est-à-dire, pas à pas en suivant la formation des classes, de l'arbre des classifications. D'ailleurs, si l'importance des valeurs atteintes par S_i (resp. τ_i) a un certain intérêt ; il

STATISTIQUE GLOBALE

Figure 4

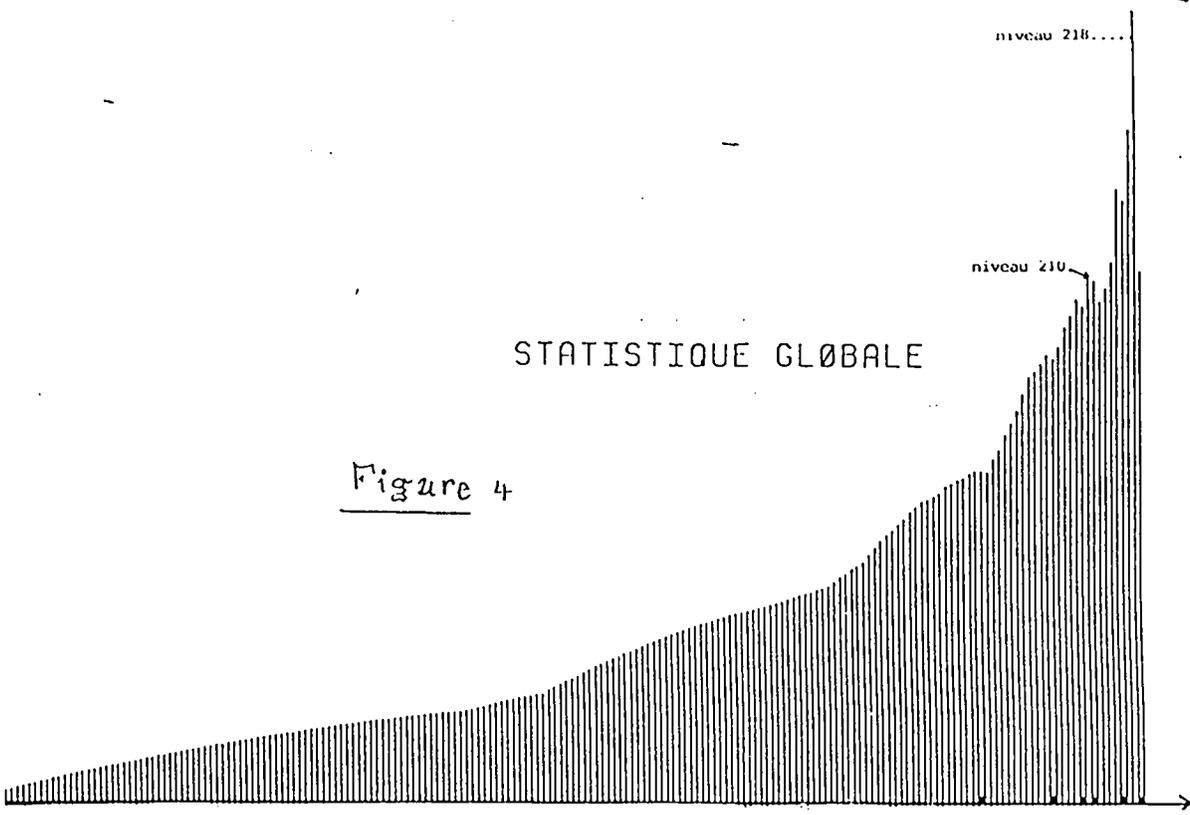
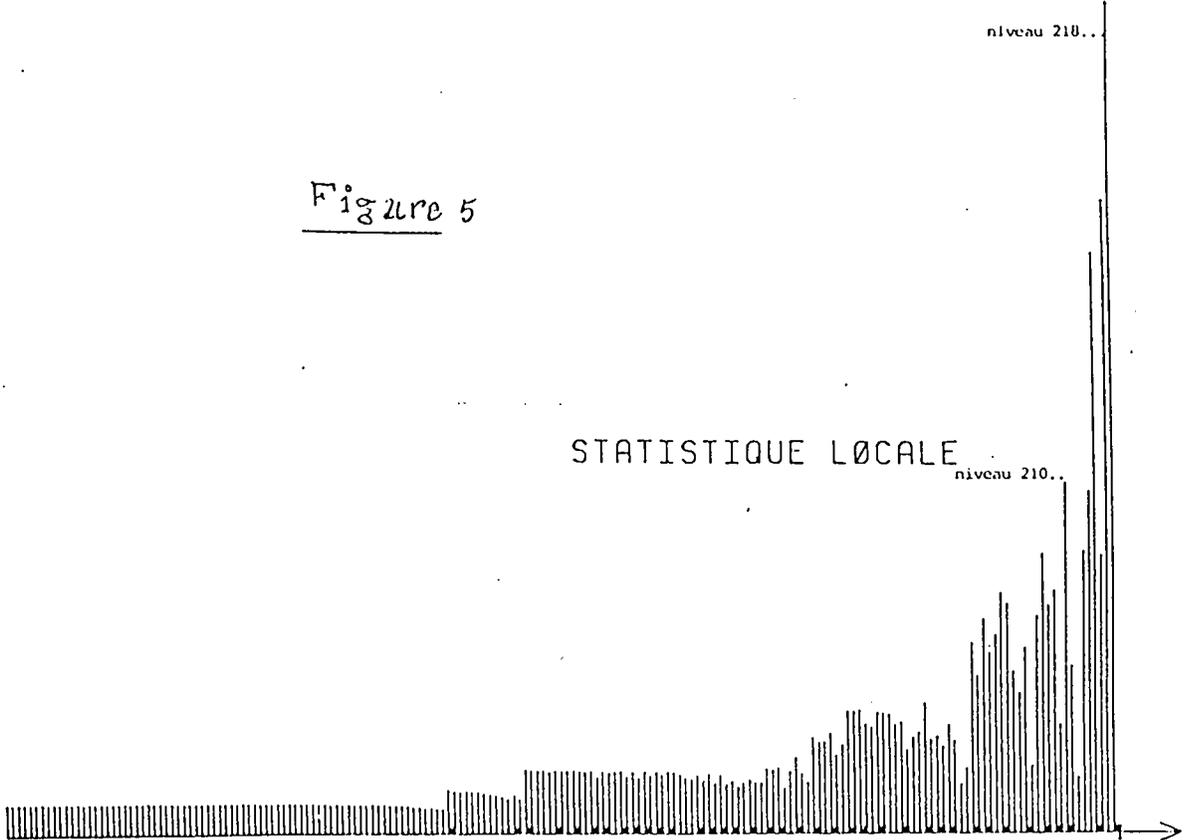


Figure 5

STATISTIQUE LOCALE



faut savoir, que ce qui importe bien davantage, c'est l'évolution de la suite des valeurs $\{S_i \mid 1 \leq i \leq I\}$ (resp. $\{\tau_i \mid 2 \leq i \leq I\}$).

Ce souci de reconnaître des composants pertinents et bien marqués dans la construction d'un arbre de classifications, existe bien dans d'autres approches d'inspiration purement géométrique {e.g. [8]}. Cependant, ces dernières utilisent les mêmes critères que ceux qui ont prévalu à la formation des classes. Sans rien dénier quant à l'intérêt de ces techniques, nous pensons quant à nous, qu'il y a de *bons* critères pour l'émergence des classes et de *bons* critères pour leur évaluation, mais que, ce ne sont pas nécessairement les mêmes !

D'autre part et surtout, le critère d'évaluation doit avoir un caractère très général ; c'est-à-dire, être tout à fait indépendant du critère de formation des classes. C'est clairement le cas de notre critère $S[\omega(E), \pi]$ (resp. $S[\mathcal{S}(E), \pi]$).

Enfin, le critère d'évaluation doit correspondre à une conception formelle claire et à un fondement de significativité statistique. C'est également le cas pour notre critère.

3.4 Arbre des parties "pertinentes"

La recherche de parties pertinentes fondées sur la notion de noeuds significatifs, conformément à l'introduction intuitive ci-dessus [cf.§1], dans l'arbre de classification s'effectue en ne considérant que les branches issues ou provenant d'un noeud significatif. Nous ne retenons alors la formation, illustrée à la figure 6, d'un noeud a que s'il est lui même significatif ou fils immédiat d'un noeud significatif [cf. Figure 6(1)] ou père d'un noeud significatif [cf. Figure 6(2)]. Dans le cas contraire [cf. Figure 6(3)], nous ne prenons pas en compte la formation du noeud a . Pour sa représentation, un noeud significatif est précédé d'une étoile.

Nous appliquons cette démarche, et de façon récursive, pour tous les noeuds de l'arbre. Ainsi, nous obtenons une suite de parties qui s'avèrent les plus intéressantes de l'arbre. Cette suite de parties sera représentée sous forme d'un arbre, tel que tous les chemins de la racine aux feuilles ont la même longueur, dont la profondeur est égale au nombre de partitions obtenues et dont chacun des niveaux, qui seront appelés *physiques*, correspond à une de ces partitions. Le premier niveau correspond alors à la partition la plus *fine* qui contient le plus grand nombre de classes. Quant au dernier, qui est la racine de l'arbre de parties, il correspond à la partition la plus *grossière* ayant le plus petit nombre de classes. De sorte que, cette représentation des différentes partitions obtenues nous semble la plus condensée et exprime bien la composition de chacune de ces partitions dans le sens où les niveaux physiques représentent les numéros des partitions et non les niveaux auxquels

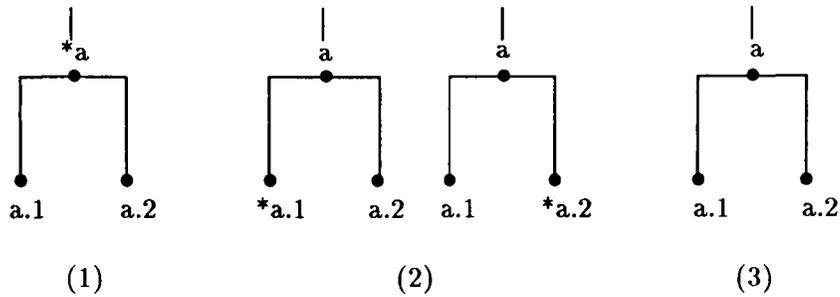


Figure 6 : Configurations

sont effectués les regroupements des classes.

Dans la suite, nous décrivons d'une part comment se forment les niveaux physiques de l'arbre et d'autre part l'algorithme permettant la formation des partitions recherchées. La figure 7 représente l'arbre de classification, issu d'une méthode de classification ascendante hiérarchique, d'un ensemble de 42 objets contenant 12 noeuds significatifs précédés d'une étoile.

Notre donnée de départ est la liste des noeuds d'un arbre de classification où sont spécifiés ceux qui sont significatifs, c'est-à-dire la notation polonaise pré-fixée de l'arbre avec la liste des noeuds significatifs. Les feuilles (c'est-à-dire éléments de l'ensemble à classifier) sont au niveau 0 et le parcours de l'arbre se fait de la racine aux feuilles.

Si un noeud a de l'arbre vérifie la condition (1) ou (2) [cf. Figure 6], son niveau physique est augmenté de 1. Dans le cas contraire (3) [cf. Figure 6], son niveau physique est égal soit au plus grand des deux niveaux physiques atteints par ses deux descendants $a.1$ et $a.2$ soit à 1 s'ils sont des feuilles. Ainsi et de façon récursive sur les noeuds, nous obtenons les parties recherchées. Cette procédure est décrite par la fonction `TRAITE_NS` qui sera détaillée ci-dessous. La figure 8 représente l'arbre des parties pertinentes résultat de la réduction de celui illustré à la figure 7. Nous obtenons neuf partitions dont la première la plus fine $P_1 = \{\{1-7\}, \{8-12\}, \{13-14\}, \{15-16\}, \{17\}, \{18\}, \{19\}, \{20-22\}, \{23-24\}, \{25\}, \{26\}, \{27-29\}, \{30-31\}, \{32-34\}, \{35\}, \{36-42\}\}$ est formée de 16 classes, la dernière $P_9 = \{\{1-22\}, \{23-42\}\}$ est formée de 2 classes et une intermédiaire, par exemple, $P_7 = \{\{1-7\}, \{8-12\}, \{13-22\}, \{23-42\}\}$ qui est formée de 4 classes.

Quant à l'algorithme permettant l'obtention de telles partitions, nous le décrivons comme suit :

Algorithme `REDUC_NS`

/* Il procède à la réduction des noeuds de l'arbre en déterminant les partitions les meilleures basées sur les noeuds significatifs */

DÉBUT

```
racine=POL2NOEUD(); /* création de l'arbre */
TRAITE_NS(racine); /* effectue la réduction */
NOEUD2POL(racine); /* transforme l'arbre en nota-
tion polonaise */
```

FIN

Nous décrivons la fonction $\text{TRAITE_NS}(a)$ pour un noeud a quelconque d'un arbre T .

Fonction $\text{TRAITE_NS}(a)$;
DÉBUT

```
SI  $\text{card}_T(a)=1$  ALORS  $n_T(a)=0$ ;
TRAITE_NS( $a.1$ );
TRAITE_NS( $a.2$ );
SI { $a$  vérifie (1) ou (2)} ALORS
 $n_T(a)=\max(n_T(a.1), n_T(a.2)) + 1$ ;
SINON  $n_T(a)=\max(\max(n_T(a.1), n_T(a.2)))$ ;
} FIN DE SI
```

FIN

Un autre problème auquel nous nous sommes intéressés concerne la taille des classes d'une partition issue d'un arbre de classification. Elle peut-être très variable et par conséquent peut fausser la qualité du résultat selon certains types de données à classifier. Pour un arbre donné de classification, nous proposons un algorithme appelé REDUC_LG qui procède à la formation des parties pertinentes en imposant un seuil sur le nombre d'éléments de chacune des classes (noeuds de l'arbre) les composant. Nous ne décrivons que la fonction TRAITE_LG, de l'algorithme REDUC_LG, qui permet de réaliser cette contrainte. La donnée de départ est la notation polonaise pré-fixée de l'arbre T de classification dont nous voulons extraire une partition en un nombre k de classes ayant chacune un nombre d'éléments inférieur ou égal à un seuil s . Le résultat de cet algorithme est un fichier contenant la description des éléments des h classes ($h \geq k$) de cette partition en fonction du seuil s choisi au départ.

Pour un noeud a quelconque de l'arbre T , la fonction $\text{TRAITE_LG}(a, s)$ est comme suit:

Fonction $\text{TRAITE_LG}(a, s)$;
/* contrainte sur l'amplitude de variation (nombre d'éléments)

```

d'une classe */
DÉBUT

    SI  $card_T(a) \leq s$  ALORS accepter la classe formée des
    feuilles issues de  $a$ ;
    SINON {
        TRAITE_LG( $a.1, s$ );
        TRAITE_LG( $a.2, s$ );
    } FIN DE SINON

FIN.

```

REDUC_NS et REDUC_LG peuvent être utilisés séparément ou alors de façon complémentaire sous la forme d'un algorithme que nous appelons REDUC_NSLG et que nous décrivons succinctement:

```

Algorithme REDUC_NSLG
DÉBUT

    racine=POL2NOEUD();
    TRAITE_NS(racine);
    NOEUD2POL(racine);
    TRAITE_LG(racine,seuil);

```

FIN.

Insistant sur le fait que les algorithmes décrits ci-dessus peuvent être appliqués à tout arbre de classification issu d'un algorithme de construction ascendante hiérarchique (CAH) [cf. §2].

Dans le cadre de ce travail, les arbres de classification que nous avons effectivement traités sont ceux issus de la famille paramétrée de classification ascendante hiérarchique $(AVL)_{0 \leq \epsilon \leq 1}$ fondée sur l'*analyse de la vraisemblance des liens* [20] à des fins de quantification en imagerie numérique [23].

4 Application en quantification scalaire

Le problème que pose la compression de l'information pour transmettre et stocker un signal image a rapidement préoccupé les chercheurs. Beaucoup de techniques de codage ont alors été développées. Un des aspects importants de la compression des données concerne la quantification (scalaire ou vectorielle [24]) qui repose sur des méthodes de classification.

Le but de notre travail est de rassembler dans le même corps conceptuel, la quantification et la classification [12, 20] en effectuant un panorama sur

plusieurs méthodes de classification hiérarchique et non hiérarchique dont nous exposerons, et ceci dans le cadre d'un travail futur [5], toute l'importance et la richesse et nous analyserons le comportement de chacune d'elles à travers des applications .

Les données volumineuses sont des images fixes en 256 niveaux de gris avec une définition de 8 pixels (éléments de l'image) par mm, et nous souhaitons obtenir une compression de ces données via une ou différentes méthodes de classification en ayant le double souci d'avoir une "bonne" reconstitution de l'image et une réduction de débit intéressante.

Rappelons qu'une image numérique est comparable à un tableau $T(x, y)$ où x et y sont les coordonnées spatiales des éléments du tableau ou pixels. A ce tableau, on fait correspondre une fonction $L(x, y)$ appelée fonction de luminance ou luminosité telle que

$$0 \leq L(x, y) \leq 255 = 2^8 - 1 \quad (48)$$

le noir étant représenté par 0 et le blanc par 255. Ces bornes sont généralement fixes a contrario de la taille de l'image qui se prête facilement à des modifications. Nous traitons des images discrètes de tailles 512×512 ou 576×720 points.

La représentation statistique d'une image est alors un nuage unidimensionnel ou histogramme

$$\{(l, n_l) \mid 0 \leq l \leq 255\} \quad (49)$$

où n_l est le nombre de pixels possédant la luminance l

et $\sum_{\{0 \leq l \leq 255\}} n_l = N = 512 \times 512$ ou 576×720 selon l'image traitée.

L'image est restituée exactement si tout pixel lui appartenant est transmis sur 8 bits. Le problème consiste donc à limiter le nombre de bits (inférieur à 8) et restituer l'image origine en respectant au mieux sa perception visuelle. En effet, toute conception de schéma de compression ou d'archivage doit prendre en compte l'importance de l'oeil humain à savoir les mécanismes physiologiques de la vision [7].

La donnée est tout simplement l'histogramme formulé par (49) sur lequel nous allons appliquer des techniques de seuillage résultant de diverses méthodes de classification automatique pour subdiviser l'échelle (48) en k intervalles - chaque intervalle a un représentant bien choisi - et ceci à des fins de compression des données image. Une quantification scalaire de l'image est ainsi réalisée et constitue une étape fondamentale pour l'étude de la quantification vectorielle.

Nous nous sommes confrontés lors de l'application de certains algorithmes de classification à un problème de non respect de connexité entre classes.

L'autre problème qui s'est posé, toute approche confondue, concerne l'amplitude de variation d'une classe qui est ici un intervalle. Comme les classes formées sont de poids (nombres de pixels concernés) presque égaux, l'amplitude de variation peut considérablement changer d'une classe à l'autre et, par conséquent, conduit à une détérioration de la qualité perceptuelle de l'image comprimée. Ainsi, pour répondre à ce type de problèmes, nous proposons la démarche permettant de déterminer l'arbre des parties pertinentes [cf. §3.4] issu d'un arbre de classification.

Une donnée correspond à l'histogramme d'une image. L'exemple que nous présentons ci-dessous est une image fixe, appelée *voiture* (fournie par le C.C.E.T.T.¹), en 256 niveaux de gris et de taille 512×512 . Sur l'histogramme de cette image, nous avons appliqué un élément de la famille $(AVL)_\epsilon$ [20], à savoir $\epsilon = 0.5$, correspondant à un optimum de la qualité de l'image comprimée en 16 niveaux de gris illustrée à la figure 9. Insistons sur le fait que l'objectif de ce travail n'est certainement pas d'atteindre une réduction de débit phénoménale, qui est du sort de la quantification vectorielle. Pour finir, nous décrivons la partition $P = \{\{0-27\}, \{28-59\}, \{60-73\}, \{74-85\}, \{86-94\}, \{95-102\}, \{103-112\}, \{113-122\}, \{123-133\}, \{134-147\}, \{148-165\}, \{166-182\}, \{183-198\}, \{199-213\}, \{214-235\}, \{236-255\}\}$ en 16 classes qui a permis d'obtenir l'image reconstruite ci-dessous [cf. figure 9]

5 Conclusion et perspectives

Insistons ici une nouvelle fois que compte tenu de leur construction et de leurs fondements formel et statistique, les critères et algorithmes que nous proposons dans ce travail, ont le plus grand degré de généralité. C'est-à-dire, ne dépendant nullement ni du codage des données, ni des indices ou critères mis en oeuvre, ni de l'algorithme de construction de l'arbre des classifications.

Peut être que nous supposons, comme c'est le cas de loin le plus fréquent, une construction ascendante hiérarchique de l'arbre des classifications. Mais, ce n'est même pas obligé ! En effet, dans le cas d'une construction descendante, par segmentations successives, on aura soin de balayer la suite des niveaux de l'arbre, non pas des feuilles à la racine; mais, de façon descendante, de la racine aux feuilles.

Ce qu'il y a de significativement nouveau dans ce travail, c'est d'une part, l'étude de la situation relative de l'approche de A.D. Gordon par rapport à la nôtre plus ancienne et d'autre part, suite à la perte de la valeur absolue de la notion de niveau d'un arbre des classifications, le passage d'un arbre condensé des classifications à un arbre de parties *pertinentes*. Ce dernier est toujours

¹Centre Commun d'Etudes de Télédiffusion et Télécommunications de Rennes

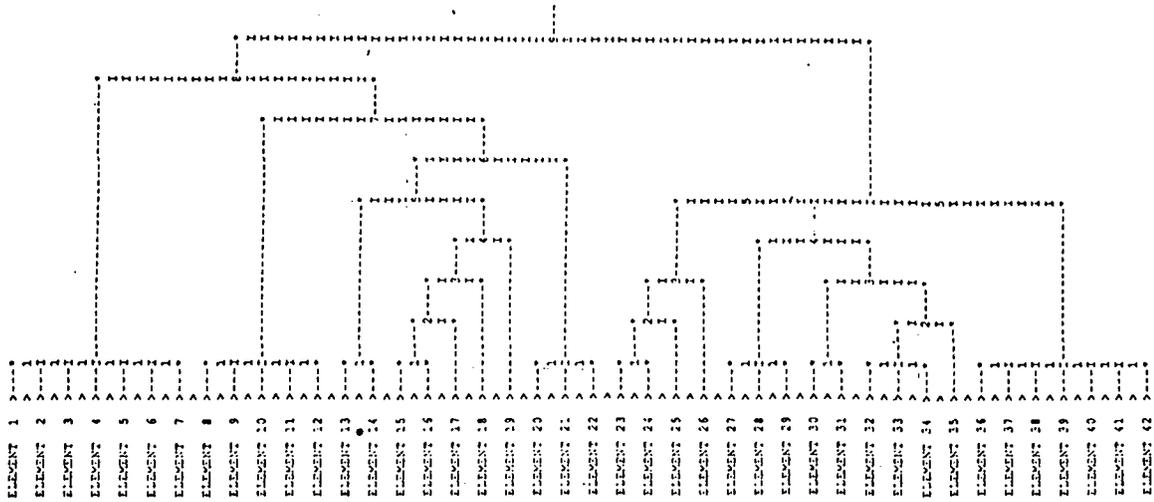


Figure 8

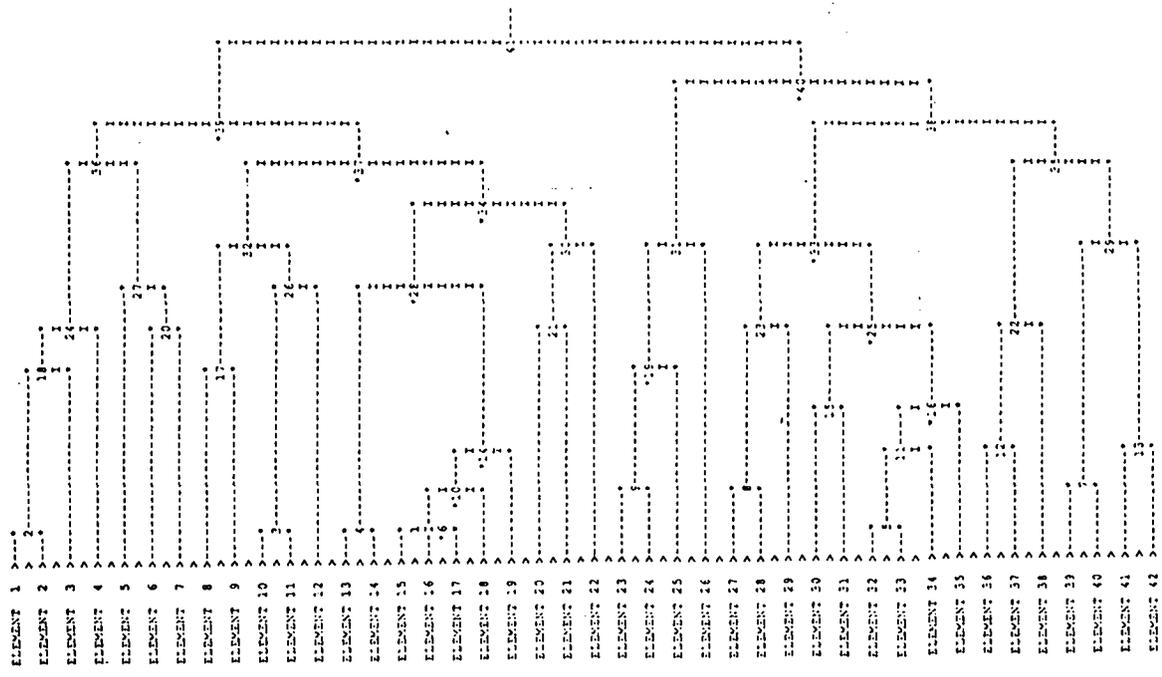


Figure 7

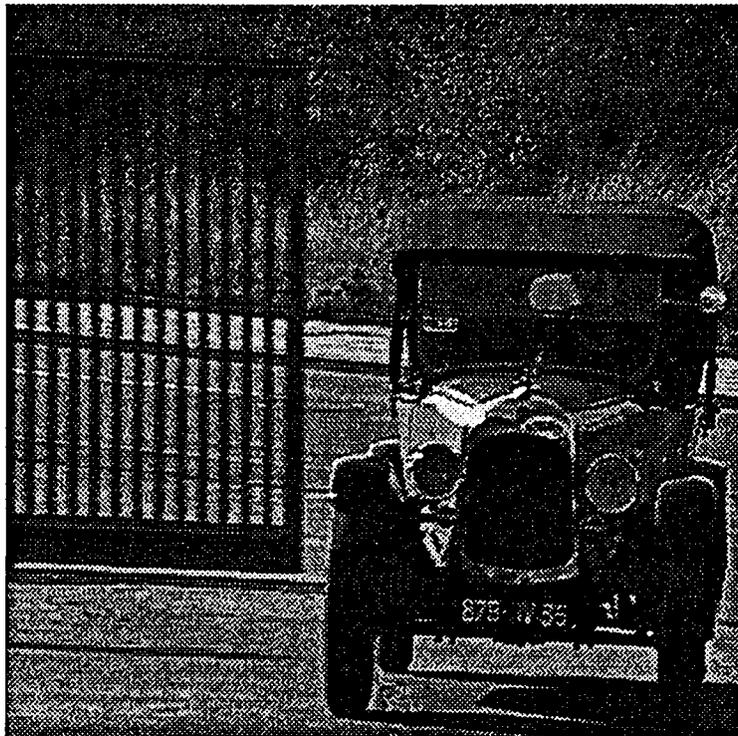


Figure 9

fondé sur la détection des noeuds significatifs de l'arbre des classifications.

En considérant ce dernier arbre des parties *pertinentes*, on peut imaginer déterminer des partitions *intéressantes*, à partir de la rétention d'un sous ensemble de ses noeuds. Le caractère *intéressant* d'une partition sera jugé à partir du critère $S[\omega(E), \pi]$ ou de celui $S[\mathcal{S}(E), \pi]$. Il y a dans cette approche toute une algorithmique à inventer.

Le critère $S[\omega(E), \pi]$ (resp. $S[\mathcal{S}(E), \pi]$) est de signification statistique. C'est ainsi que $S[\omega(E), \pi]$ a des valeurs toutes petites aux premiers niveaux de l'arbre, alors qu'il n'y a pour ainsi dire, pas d'inversions, entre la préordonnance totale $\omega(E)$ et celle, en deux classes, induite par la partition [cf. (24) §3.3]. C'est que, les partitions concernées ont très peu de consistance. Si on désire un coefficient qui mette en évidence la similarité des deux préordonnances indépendamment de leurs consistances respectives propres, on peut considérer

$$\frac{S[\omega(E), \pi]}{S(\pi, \pi)} \quad (50)$$

où $S(\pi, \pi)$ est le critère d'adéquation entre la préordonnance à deux classes [cf. (24) §3.3] et elle même.

Enfin, on peut signaler que les idées de ce travail qui étaient en partie latente, ont trouvé précipitation grâce aux contraintes qu'impose une quantification scalaire en imagerie numérique [cf. §4 ci-dessus], à partir de la classification hiérarchique. La question se pose maintenant de savoir comment ces idées peuvent être utilisées dans le cadre d'une quantification vectorielle en imagerie numérique.

Bibliographie

- [1] Benzecri J.P., "Analyse factorielle des proximités", I & II, publication de l'Inst. de Stat. de l'Univer. de Paris, XIII & XIV, 1964-1965.
- [2] Benzecri J.P. et al., L'analyse des données I, la taxinomie, Dunod, Paris, 1973.
- [3] Bruynooghe M., "Nouveaux algorithmes en classification automatique applicables aux très grands ensembles de données, rencontrés en traitement d'images et en reconnaissance des formes", Thèse de Doctorat d'Etat, Université de Paris VI, 23 Janvier 1989.
- [4] Fisher W.D., "On Grouping for Maximum Homogeneity", JASA, vol. 53, pp. 789-798, 1958.
- [5] Ghazzali N., Lerman I.C., "Quantification des images fixes et classification", rapport interne IRISA (à paraître), février 1991.
- [6] Gordon A.D., "Parsimonious Trees", Journal of Classification, pp. 85-101, 1987.
- [7] Imbert M., "La neurologie de l'image", La Recherche, vol. 14, no.144, may 1983.
- [8] Jambu M., "Exploration informatique et statistique des données", Paris, 1989.
- [9] Knuth D.E., The Art of Computer Programming, vol. I, Fundamental Algorithms, Addison-Wesley, 1969.
- [10] Lance G.N., Williams W.T., "A general theory of classification sorting strategies : 1 = hierarchical systems, 2 = clustering systems". Computer Journal 9-10, pp. 373-380, 1967.
- [11] Lechevallier Y., "Recherche d'une partition optimale sous contrainte d'ordre total", rapport de recherche INRIA, no 1247, 61 pages, 1990.
- [12] Léger A., "Introduction à la quantification vectorielle des images fixes", Univ. de Rennes I, 18 Décembre 1987.
- [13] Lerman I.C., Les bases de la classification automatique, Gauthier- Villars, collection programmation, Paris, 1970.

- [14] Lerman I.C., Classification et analyse ordinale des données, Dunod , Paris, 1981.
- [15] Lerman I.C., "Sur la signification des classes issues d'une classification automatique", in Numerical Taxonomy, NATO ASI Series vol. G1, edited by J. Felsenstein, Springer Verlag, pp. 179-198, 1983.
- [16] Lerman I.C., Peter Ph., "Organisation et consultation d'une banque de *petites annonces* à partir d'une méthode de classification hiérarchique en parallèle", Journées Analyse des Données et Informatique, IV, Octobre 1985, Versailles, in North-Holland ed. by E. Diday et al, pp. 121-136, 1986.
- [17] Lerman I.C., "Comparing partitions (mathematical and statistical aspects)", Classification and Related Methods of Data Analysis, H.H. Bock (Editor), Elsevier Science Publishers B.V. North Holland, 1988.
- [18] Lerman I.C., "Comparing relational variables according to likelihood of the links classification method", Classification and Related Methods of Data Analysis, H.H. Bock (Editor), Elsevier Science Publishers B.V. North Holland, 1988.
- [19] Lerman I.C., "Formules de réactualisation en cas d'agrégations multiples", R.A.I.R.O. série R.O., vol. 23 no 2 pp. 151-161, 1989.
- [20] Lerman I.C., Ghazzali N., "Quantification par la méthode de vraisemblance du lien (AVL) avec codage préordonnance", rapport interne IRISA, no. 499, 60 pages, 1989.
- [21] Molliere J.L., "What's the real number of clusters?", classification as a tool of research, W. Gaul and M. Schader (Editors), North Holland.
- [22] Peter Ph., "Méthodes de classification hiérarchique et problèmes de structuration et de recherche d'informations assistées par ordinateur", Thèse de l'Université de Rennes I, 6 Mars 1987.
- [23] Pratt W.K., Digital Image Processing, Wiley-Interscience Publication, 1978.
- [24] Ramamurthi B., Gersho A., "Classified vector quantization of images", IEEE Trans. on Com., vol. com-34, no 11, pp. 1105-1115, November 1986.
- [25] Shepard R.N., "The analysis of proximities : multidimensional scaling with an unknown distance function", Psychometrika, vol. 27, 1962.

- [26] Sneath P.H.A., "Some empirical tests for significance of clusters" in *Data Analysis and Informatics*, E. Diday et al.(Editors), North Holland, 1980.
- [27] Tricot M.L., Donegani M., "Présentation unifiée des indices de proximité entre classes en classification hiérarchique ascendante", *R.A.I.R.O.*, vol. 23, no. 2, 1989.

- PI 556 **CONCEPTION ET INTEGRATION D'UN CORRELATEUR SYSTOLIQUE**
Catherine DEZAN, Eric GAUTRIN, Patrice QUINTON
Novembre 1990, 16 Pages.
- PI 557 **VARIATIONAL APPROACH OF A MAGNETIC SHAPING PROBLEM**
Michel CROUZEIX
Novembre 1990, 14 Pages.
- PI 558 **THE DAVIDSON METHOD**
Michel CROUZEIX, Bernard PHILIPPE et Miloud SADKANE
Novembre 1990, 22 Pages.
- PI 559 **A DISTRIBUTED SOLUTION TO THE k -OUT OF- M RESOURCES ALLOCATION PROBLEM**
Michel RAYNAL
Novembre 1990, 18 Pages.
- PI 560 **A SIMPLE TAXONOMY FOR DISTRIBUTED MUTUAL EXCLUSION ALGORITHMS**
Michel RAYNAL
Novembre 1990.
- PI 561 **MULTIMODAL ESTIMATION OF DISCONTINUOUS OPTICAL FLOW USING MARKOV RANDOM FIELDS**
Fabrice HEITZ, Patrick BOUTHEMY
Novembre 1990, 50 Pages.
- PI 562 **EFFICIENT GLOBAL COMPUTATIONS ON A PROCESSOR NETWORK WITH PROGRAMMABLE LOGIC**
J.M. FILLOQUE, E. GAUTRIN, B. POTTIER
Novembre 1990, 14 pages.
- PI 563 **EQUATIONAL SETS OF TREE-VECTORS**
Anne GRAZON, Jean-Claude RAOULT
Novembre 1990, 20 Pages.
- PI 564 **MULTIFRAME-BASED IDENTIFICATION OF MOBILE COMPONENTS OF A SCENE WITH A MOVING CAMERA**
Edouard FRANCOIS, Patrick BOUTHEMY
Décembre 1990, 30 pages.
- PI 565 **NAIVE RESERVE CAN BE LINEAR**
Pascal BRISSET, Olivier RIDOUX
Novembre 1990, 18 pages.
- PI 566 **METHODES D'INTEGRATION TEMPORELLE EN TRAITEMENT D'ANTENNE**
Olivier ZUGMEYER, Jean-Pierre LE CADRE
Décembre 1990, 54 pages. Rapport n° 1
- PI 567 **METHODES PARAMETRIQUES POUR LA DETECTION DE SOURCES EN MOUVEMENT 1**
Olivier ZUGMEYER, Jean-Pierre LE CADRE
Décembre 1990, 42 pages. Rapport n° 2
- PI 568 **QUOI RETENIR D'UN ARBRE DE CLASSIFICATION ? UN ESSAI EN QUANTIFICATION D'IMAGE NUMERISEE**
Israël César LERMAN, Nadia GHAZZALI
Décembre 1990, 36 pages, Projet CADO.
- PI 569 **VARIABLES RELATIONNELLES CODAGE ET ASSOCIATION**
Israël César LERMAN, Mohamed OUALI-ALLAH
Décembre 1990

ISSN 0249 - 6399