LES CAHIERS DE L'ANALYSE DES DONNÉES

J. P. BENZÉCRI
M. JAMBU

Agrégation suivant le saut minimum et arbre de longueur minima

Les cahiers de l'analyse des données, tome 1, n° 4 (1976), p. 441-452

<http://www.numdam.org/item?id=CAD_1976__1_4_441_0>

Les Cahiers de l'Analyse des Données
Vol. I - 1976 - n° 4 - p. 441-452

AGRÉGATION SUIVANT LE SAUT MINIMUM ET ARBRE DE LONGUEUR MINIMA

[SQUELETTE ARBORESCENT]

par J. P. Benzécri (1) *et M. Jambu* (2)

1. *Classification hiérarchique et squelette arborescent*

Le traité de l'Analyse des données, insiste sur l'opposition entre les représentations spatiales continues issues de l'analyse factorielle, et les dichotomies en classes indépendantes propres à la taxinomie ; et il affirme que l'arbre de longueur minima qui prétend suggérer à la fois un système de classes et leur disposition spatiale ne fournit en toute rigueur ni hiérarchie ni espace. Dans la présente note nous nous proposons, sur le conseil de L. LEBART, de nuancer ces jugements que nous rappellerons d'abord.

On lit en tête de l'article sur les peurs enfantines (TI C n° 13) :

" En commentant les résultats issus d'un algorithme de classification il faut prendre garde que d'une part tant l'amplitude des intervalles séparant les classes que la cohésion même de celles-ci ne peuvent être éprouvés ; et que d'autre part l'ensemble des classes suspendues aux ramifications de l'arbre est semblable à ces essaims d'objets mobiles que l'art expose au souffle de l'air pour nous en proposer la changeante figure ; en ce sens que quant à l'arbre global des classes, une multitude de combinaisons hypothétiques sont possibles."

Quant à l'arbre de longueur minima (ou polygone sans circuit de longueur réunissant les points d'un ensemble I) nous affirmons (cf. TI B n° 9 § 4) :

"On se gardera de confondre arbre polygone (défini ici) et arbre hiérarchique ([D.M.Cl.] § 1.2) : si sur un arbre hiérarchique connexe A on prend pour arêtes toutes les paires formées d'un élément et de son prédécesseur immédiat, on a un arbre polygone bien déterminé [ayant pour sommets à la fois les individus et les classes ou noeuds] ; mais réciproquement, sur un arbre polygone, tout point i peut être choisi comme sommet hiérarchique, l'ensemble des prédécesseurs de tout point i' n'étant alors que l'ensemble des sommets du chemin unique reliant i à i'."

Et quand à l'imprécision des classes suggérées par la méthode de l'arbre de longueur minima, nous citerons l'écologiste A. LACOSTE (TI C n° 3' § 2).

"Il est à souligner que, malgré son intérêt indéniable, la méthode n'aboutit pas, en général, à une individualisation sufisamment nette des groupements pour permettre de placer entre eux des coupures objectives. Ainsi, indépendamment des définitions que nous en a fournies précédemment l'analyse factorielle, la discrimination de ces groupements sur l'arbre obtenu serait en fait particulièrement délicate."

---

(1) *Professeur à l'Université P. et M. Curie ; a écrit les §§ 1-4*

(2) *Attaché de recherches CNRS ; a écrit les §§ 5-7*
Université P. et M. Curie

Dans la suite, nous rappellerons d'abord sur un exemple (§ 2) que la classification ascendante hiérarchique avec agrégation suivant le saut minimum, ne diffère pas en tant que construction mathématique de la taxinomie polonaise (arbre de longueur minima) ; mais que ces méthodes offrent sur un même être mathématique deux points de vue qui se complètent utilement (*). Puis (§ 3) nous précisons par des formules ces assertions. Enfin (§ 4) nous proposons un algorithme associant à toute classification hiérarchique binaire (même non issue du critère d'agrégation suivant le saut minimum ; critère qui conduit souvent à des classes filiformes peu acceptables) un squelette arborescent qui précise des points de contacts entre classes et supprime en quelque sorte par des liens la mobilité de celles-ci.

2. Application simultanée de deux méthodes à un même exemple

Partons d'un ensemble I, dont les éléments sont numérotés de 1 à 7 ; et sur lequel est défini un indice de distance d (dont il importe peu ici qu'il satisfasse à l'inégalité du triangle).

i \ i'	1	2	3	4	5	6	7
1	0	1	9	9	9	9	5
2	1	0	9	9	9	8	3
3	9	9	0	6	7	9	9
4	9	9	6	0	4	2	9
5	9	9	7	4	0	5	9
6	9	8	9	2	5	0	9
7	5	3	9	9	9	9	0

Tableau 1 - Indice de distance d, sur l'ensemble I = {1, 2,..., 7}

D'après ces données, on construit sur I d'une part une classification ascendante hiérarchique (avec pour critère d'agrégation le saut minimum), figure 1 ; d'autre part un arbre de longueur minima, figure 2 :

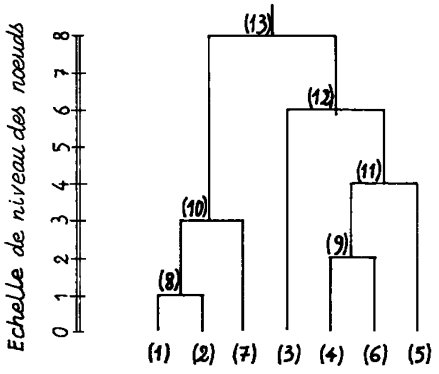


Figure 1 : Classification ascendante hiérarchique sur l'ensemble I, construite par agrégation suivant le saut minimum.

(*) *Sur cette complémentarité, déjà vue par J.C. GOWER & J.S. ROSS (in Applied Statistics Vol. 18, n° 1, pp.54-65 ; 1969), L. LEBART a attiré notre attention.*

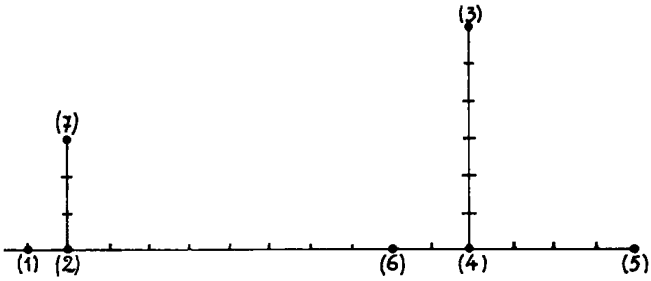


Figure 2 : Arbre de longueur minima sur l'ensemble I.

Coupons l'arbre de longueur minima suivant son lien le plus long qui est (2,6) : d(2,6) = 8 : il reste deux branches {1,2,7} et {3,4,5,6} qui ne sont autres que les deux classes (10) et (11) en lesquelles se divise l'ensemble I, au haut de la hiérarchie ; de plus entre ces deux classes, le saut minimum ou niveau du noeud (13) est fourni par le lien (2,6) (en ce sens qu'entre deux éléments i ∈ (10) et i' ∈ (11), la distance d(i,i') atteint son minimum pour d(2,6) = 8. Poursuivons : dans la branche {3,4,5,6}, le lien le plus long est (3,4) : d(3,4) = 6 ; en rompant ce lien, on sépare {3} de {4,5,6}, qui n'est autre que la classe (11) de la hiérarchie ; et le niveau du noeud (12) qui se subdivise en (3) et (11) est bien égal à la longueur 6 du lien d(3,4). Et ainsi de suite : en ôtant de l'arbre de longueur minima ses liens depuis les plus longs jusqu'aux plus courts on voit apparaître les classes de la hiérarchie ; la longueur des liens donnant le niveau des noeuds. Réciproquement, en associant à chaque noeud n de la hiérarchie un lien qui réalise le saut minimum entre les deux classes A(n) et B(n) (aîné et benjamin de n) on obtient, arête après arête, un arbre de longueur minima. On tentera donc de représenter simultanément une vue perspective de l'arbre taxinomique avec, à sa base, le squelette de longueur minima qui est comme un système de liens entravant le mouvement des branches suspendues à I : c'est la figure 3. On pourrait aussi, partant du dessin du squelette de longueur minima (figure 2), encercler par des contours les principales classes de la hiérarchie taxinomique.

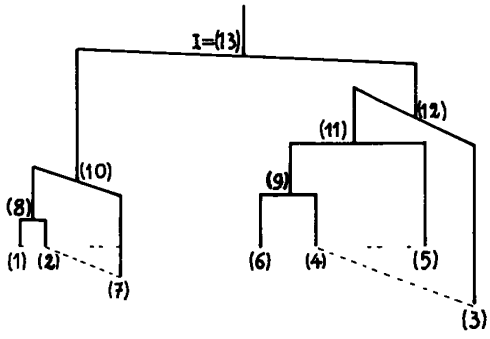


Figure 3 : Représentation simultanée d'une hiérarchie suspendue de classes avec à sa base un arbre de longueur minima.

3. Enoncés mathématiques

Pour énoncer et démontrer les propriétés remarquées au § 2, nous ferons d'abord quelques rappels et les compléterons par de nouvelles définitions. Dans tout le § 3, on considère un même ensemble fini I, muni d'un indice de distance d strictement positif i.e. tel que :

$$\forall i' \in I : d(i,i') \geq 0 \quad ; \quad (d(i,i') = 0 \Leftrightarrow i = i')$$

On note d_{uiM} la distance sur I définie de manière unique par la condition d'être l'ultramétrique maxima inférieure à d(cf. [D.M.Cl.] TI B n° 3 § 4.3.1). Cette distance est encore notée d_{saut} , car en bref, on peut joindre i à i' par une chaîne dont tout maillon ait une longueur $d(i_2 , i_{2+1})$ inférieure ou égale à $d_{uiM}(i,i')$; mais il n'est pas possible que tout maillon soit strictement inférieur à ce seuil qu'il faut donc sauter pour passer de i à i'. Une troisième définition de d_{uiM} est fournie par la classification ascendante hiérarchique avec agrégation suivant le saut minimum, (cf.[C.A.H.] TI B n° 4 § 2.1) : d_{uiM} est l'ultramétrique associée à une telle classification indicée (i.e. $d_{uiM}(i,i')$ est le plus bas niveau, ou indice de diamètre, d'un noeud n contenant à la fin i et i' (cf.[D.M.Cl.] § 4.2) ; bien que l'algorithme ascendant puisse comporter des choix dans l'ordre des agrégations binaires, on sait qu'il conduit en substance à une hiérarchie unique qui est le mieux définie par l'ultramétrique d_{uiM} , elle-même.

Soit maintenant P un polygone sur I considéré comme un ensemble d'arêtes ou paires d'éléments distincts de I (paires non ordonnées $\{i,i'\} \approx \{i',i\}$: par exemple, l'arbre de longueur minima de la figure 2 peut s'écrire :

$$P = \{\{1,2\} ; \{2,7\} ; \{2,6\} ; \{6,4\} ; \{4,3\} ; \{4,5\}\}$$

On dit que P est un arbre sur I si :

1°) tout élément i de I appartient à au moins une arête de P ;

2°) deux éléments quelconques i et i' peuvent être réunis par une chaîne d'arêtes de P ; dans ces conditions, le nombre des arêtes de P est inférieur d'une unité à celui des éléments de I : Card P = Card I-1. On note long(P) la somme des longueurs des arêtes de P (pour l'indice d) :

$$long(P) = \Sigma \{d(i,i') \mid \{i,i'\} \in P\}$$

Dans la suite, un arbre P sur I sera parfois appelé squelette, afin d'éviter toute confusion avec une hiérarchie de parties.

Nous associons à P un indice de distance d^P sur I, défini comme suit :

$$d^P(i,i') = \text{si } \{i,i'\} \in P \text{ alors } d(i,i'), \text{ sinon } \infty ;$$

(i.e. on conserve la longueur des arêtes ; mais on donne valeur infinie à d(i,i') si i et i' ne sont pas directement reliés par une arête). Il est clair que la connaissance de d^P détermine P.

Ceci posé les propriétés remarquées au § 2 résultent du théorème suivant :

Théorème : Soit P un arbre sur I : P est de longueur minima si et seulement si la distance ultramétrique inférieure maxima d^P_{uiM} associée à d^P est égale à d_{uiM}. Dans ce cas long (P) n'est autre que la somme des niveaux des noeuds d'une hiérarchie binaire construite sur I par agrégation ascendante suivant le saut minimum, au sens de l'indice d ; cette hiérarchie peut aussi bien être obtenue sur P (i.e. sur I muni de d^P).

Preuve : D'abord il est clair que pour tout polygone P, $d \leq d^P$, donc $d_{uiM} \leq d^P_{uiM}$.

Reste à montrer que si P est un arbre sur I avec l'inégalité stricte

$d_{uiM} <_s d^P_{uiM}$, P n'est pas de longueur minima. Soit en effet : i et i' deux éléments tels que :

$$d_{uiM}(i,i') <_s d^P_{uiM}(i,i')$$

la chaîne reliant i à i' sur P, comporte un maillon (i_2 , i_{2+1}) dont la longueur est strictement supérieure à $d_{uiM}(i,i')$; supprimez ce maillon de P. Il reste de P deux arbres F et F' l'un contenant i, l'autre i' qui peuvent certainement être reliés par un maillon (j,j') tel que $d(j,j') \leq d_{uiM}(i,i')$: car la chaîne de saut minima qui relie i à i', comporte certainement au moins un maillon joignant un élément j de F à un élément j' de F'. En substituant (j,j') à (i_2 , i_{2+1}) dans P, on obtient un nouvel arbre sur I qui est strictement plus court que P.

Reste à vérifier que la longueur de P n'est autre que la somme des niveaux des noeuds d'une classification binaire agrégée suivant le saut minimum avec l'indice d. Cette classification, (à des permutations éventuelles près dans les choix d'agrégation) ne dépend que de d_{uiM} : on peut donc la construire à partir de d^P au lieu de d. Dès lors, il est clair que les niveaux des noeuds sont les longueurs des arêtes de P, auxquelles ils correspondent biunivoquement.

Remarque 1 : I étant donné muni de d, d_{uiM} est déterminée de manière unique ; mais non l'arbre P de longueur minima. Ceci correspond à la non-unicité de l'ordre des agrégations dans la classification ascendante ; et aussi au fait qu' entre les deux classes A(n) et B(n) réunies pour former un noeud n, il se peut qu'existent plusieurs arêtes (paires {i,i'}, $i \in A(n)$, $i' \in B(n)$) réalisant le saut minimum ($d(i,i') = \nu(n)$ = indice de niveau de n).

Remarque 2 : On rapprochera l'égalité : $\text{long}(P) = \Sigma\{\nu(n) \mid n \in N\}$
(où N = ensemble des noeuds de la hiérarchie indicée A construite en agrégeant suivant le saut minimum ; $\nu(n) = d(n)$, indice ou niveau du noeud n dans cette hiérarchie) de l'égalité : $\Sigma\lambda_\alpha = \Sigma\ \nu'(n) \mid n \in N'\}$

(où N' = ensemble des noeuds de la hiérarchie construite par agrégation suivant la variance ; $\nu'(n)$, niveau de n dans cette hiérarchie ; λ_α valeur propre, ou moment d'inertie de rang α extrait du nuage I des individus). Là comme ici, il s'agit de deux représentations l'une plutôt géométrique (squelette ou nuage de points) l'autre hiérarchique : mais dans $\text{long}(P) = \Sigma\nu(n)$, il y a non seulement égalité de deux sommes mais aussi correspondance biunivoque entre les termes ; tandis que $\Sigma\lambda_\alpha$ et $\Sigma\nu'(n)$ sont deux décompositions distinctes de l'inertie totale du nuage (cf. *Cahiers de l'Analyse des données. I. n° 1 p. 82)*

4. *Construction d'un squelette associé à une classification hiérarchique binaire*

Supposons donné sur I, muni de l'indice de distance d, un arbre de longueur minima P. A partir de P, on peut par voie descendante, munir I d'une classification qui n'est autre que celle qu'on obtiendrait, par voie ascendante en agrégeant suivant le saut minimum. Pour cela (cf. § 2) on ôte de P son segment le plus long : il reste deux arbres connexes A(P) et B(P) recouvrant deux classes A(I) et B(I) ; de même chacune de ces classes sera divisée en deux en ôtant respectivement à A(P) et B(P) son segment le plus long ; et ainsi de suite jusqu'à atteindre des classes réduites à un seul élément.

Cette construction a pour nous peu d'intérêt pratique (*), parce que nous utilisons communément un programme de classification ascendante hiérarchique ([C.A.H.]), comportant de nombreuses variantes et aides à l'interprétation. En revanche la construction réciproque - passer de la hiérarchie à l'arbre de longueur minima - nous paraît susceptible de perfectionner le programme de [C.A.H.]. Nous proposons donc ici un algorithme général associant à toute hiérarchie binaire A sur B, un arbre ou squelette sur I dont les arêtes correspondent biunivoquement aux noeuds de A. De façon précise à tout noeud n correspond une arête {a(n), b(n)} telle que a(n) ∈ A(n), b(n) ∈ B(n) ; et que d(a(n), b(n)) réalise le minimum de la distance entre éléments des deux classes A(n) et B(n) successeur immédiat de n. Répétons que l'algorithme s'applique quel que soit A : si A a été obtenu par agrégation ascendante suivant le saut minimum, il fournit un arbre de longueur minima ; en tout cas comme nous l'annoncions au § 1, le squelette polygonal construit, précise des points de contacts entre classes et il supprime par des liens la mobilité de celles-ci (cf. figure 3).

```
entier CARDI, CARDSOM, N, SA, SB, KA, S, SG, X          ;
entier tableau SOM[1:CARDI], IS, ISG[1:CARDI*(CARDI-1)/2],
     A, B, IA, IB[CARDI+1 : (2*CARDI)-1] ; ADN[1:(2*CARDI)-1],
réel tableau LO[CARDI+1:(2*CARDI)-1], DIS[1:CARDI*(CARDI-1)/2];
entier procédure K(U,V) ; entier U, V ; début entier K ;
K:= inf(U, V) + ((sup(U, V)-1)*(sup(U, V)-2)/2) ; K(U,V):= K fin ;
```

Commentaire : CARDI est le cardinal de l'ensemble I sur lequel est construit une arborescence binaire ; celle-ci, qui comprend (2*CARDI)-1 éléments (dont CARDI terminaux, les éléments de I numérotés de 1 à CARDI, et (CARDI-1) noeuds, numérotés de CARDI+1 à (2*CARDI)-1) est donnée au départ par les tableaux A et B. L'ensemble I étant muni d'un indice de distance donné au départ dans DIS, on vise à déterminer pour chaque noeud N deux points IA[N] et IB[N] (points de I désignés par leur numéro compris entre 1 et CARDI), celui-là dans l'aîné A[N], celui-ci dans le benjamin B[N], réalisant le minimum (noté LO[N]) de la distance entre points de ces deux classes. Dans ce but, on parcourt de bas en haut la hiérarchie (i.e. on prend les noeuds de N = CARDI+1 à N = (2*CARDI)-1). Sans avoir, comme dans [C.A.H.], à construire successivement les noeuds, on doit cependant considérer une suite d'arbres non connexes dont le nombre total des sommets, CARDSOM = (2*CARDI)+1-N, décroît de CARDI à 2. Il faut garder le compte de ces sommets à chaque itération, on a en SOM(S) le numéro dans la hiérarchie, du sommet numéroté S ; et réciproquement, si l'élément N (noeud ou terminal) de la hiérarchie, est à cette itération, un sommet, son numéro, ou adresse comme sommet est en ADN[N], le tableau DIS donne en DIS[K(S,SP)] (comme en [C.A.H.], K(U,V) est le numéro de la paire UV ; dans l'ordre lexicographique des VU pour lesquelles V< U), le saut minimum entre les sommets numérotés S et SP ; de plus, IS[K(S,SP)] et ISG[K(S,SP)] sont les numéros (compris entre 1 et CARDI) des éléments de I réalisant ce saut minimum : IS[K(S,SP)] appartient à celui des deux sommets dont le numéro (S ou SP) est le plus petit ; et ISG[K(S,SP)], au sommet dont le numéro est le plus grand.

```
lire CARDI ; lire tableau A,B,DIS  ;
pour I:= 1 pas 1 jusqu'à CARDI faire début
     ADN[I]:= SOM[I]:= I  ;
     pour IG:= I+1 pas 1 jusqu'à CARDI faire début
          IS[K(I,IG)]:= I ; ISG[K(I,IG)]:= IG fin fin  ;
```

(*) *GOWER ET ROSS, au contraire, (op. laud ; in Appl. Stat. 1969), passent de l'arbre de longueur minima (qu'ils construisent par un algorithme très satisfaisant) à l'agrégation suivant le saut minimum (appelée par eux Single Linkage Cluster Analysis : parce que, en bref, l'agrégation est décidée d'après un seul lien, le plus court ; et non d'après une distance moyenne) : mais nous avons dit qu'à l'expérience cette procédure d'agrégation, déçoit souvent ; en sorte que, selon nous, si la construction d'un arbre de longueur minima peut servir, c'est associée à une classification hiérarchique obtenue autrement.*

Commentaire : Initialement, les sommets ne sont autres que les individus, numérotés de 1 à CARDI ; entre deux tels sommets, réduits chacun à un élément, le saut minimum est réalisé par la distance entre ces éléments. Ci-dessous débute la grande boucle indicée par N.

```
pour N:= CARDI+1 pas 1 jusqu'à (2*CARDI)-1 faire début
CARDSOM:= (2*CARDI)+1 - N ;
SA:= ADN[A[N]] ; SB:= ADN[B[N]] ;
KA:= K(SA,SB) ; LO[N]:= DIS[KA] ;
si SA < SB alors début
     S:= SA ; SG:= SB ; IA[N]:= IS[KA] ; IB[N]:= ISG[KA] fin
     sinon début
     S:= SB ; SG:= SA ; IB[N]:= IS[KA] ; IA[N]:= ISG[KA] fin ;
```

Commentaire : En tant que sommets, les successeurs du noeud N qu'on va traiter ont présentement les numéros SA, SB ; ces numéros sont redésignés par S (le plus petit) et SG (le plus grand) ; les tableaux DIS, IS et ISG donnent la longueur et les extrémités du maillon réalisant le saut minimum entre A[N]et B[N].

```
ADN[N]:= S ; SOM[S]:= N ;
ADN[SOM[CARDSOM]]:= SG ; SOM[SG]:= SOM[CARDSOM] ;
```

Commentaire : Les deux sommets numérotés S et SG sont à supprimer ; le noeud N reçoit en tant que sommet le numéro S ; le sommet anciennement numéroté CARDSOM (sommet qui en tant que noeud est SOM[CARDSOM]) prend l'adresse SG laissée vacante.

```
pour X:= 1 pas 1 jusqu'à S-1, S+1 pas 1 jusqu'à SG-1,
SG+1 pas 1 jusqu'à CARDSOM faire
     si DIS[K(S,X)]> DIS[K(SG,X)] alors début
        DIS[K(S,X)]:= DIS[K(SG,X)] ;
     si S < X < SG alors début
        IS[K(S,X)]:= ISG[K(SG,X)];
        ISG[K(S,X)]:= IS[K(SG,X)] fin
     sinon début
        IS[K(S,X)]:= IS[K(SG,X)] ;
        ISG[K(S,X)]:= ISG[K(SG,X)] fin fin ;
```

Commentaire : Il faut remettre à jour les tableaux DIS, IS et ISG. Si entre le nouveau sommet S, qui n'est autre que le noeud N, et le sommet X (sommet conservé et non encore renuméroté) le saut minimum est réalisé pour la partie de N qui constituait anciennement le noeud S rien n'est à changer dans DIS[K(S,X)],[IS K(S,X)], ISG[K(S,X)] : le nouveau sommet S hérite à la même place, des informations relatives à l'ancien S. Sinon, il faut attribuer au nouveau S des informations relatives à l'ancien SG ; en prenant garde que la position de X relativement à S peut n'être pas celle de X relativement à SG (si S < X < SG).

```
pour X:= 1 pas 1 jusqu'à SG-1, SG+1 jusqu'à CARDSOM faire début
     DIS[K(X,SG)]:= DIS[K(X,CARDSOM)] ;
     si X < SG alors début
        IS[K(X,SG)]:= IS[K(X,CARDSOM)] ;
        ISG[K(X,SG)] := ISG[K(X,CARDSOM)] fin
     sinon début
        IS[K(X,SG)]:= ISG[K(X,CARDSOM)] .
        ISG[K(X,SG)]:= IS[K(X,CARDSOM)] fin fin fin.
```

Commentaire : La boucle et avec elle le programme, s'achèvent en recopiant sous l'adresse SG les informations relatives au sommet initialement numéroté CARDSOM. On a pris garde que la position de X relativement à SG peut n'être pas la même que relativement à CARDSOM. Dans un programme destiné à l'ordinateur, il conviendrait de noter, e.g., KG:= K(X,SG), KS:= K(X,CARDSOM), IS[KS]:= ISS, ISG[KS]:= ISGS, afin de réduire le nombre des calculs d'adresses et consultation de tableaux. On ne l'a pas fait ici, afin de laisser en clair les entités que manipule l'algorithme.

5. Construction d'un squelette arborescent associé à une classification hiérarchique binaire (version Jambu)

L'algorithme présenté au § 4 nécessite le logement en mémoire centrale du tableau des distances et des numéros des extrémités des arêtes, soit sur un ensemble de CARDI éléments, 3*(CARDI*(CARDI-1)/2)mots mémoire. En reprenant les notations du § 4, on a simplifié l'algorithme en évitant un grand nombre de recopiages et en supprimant deux des trois tableaux nécessaires pour le fonctionnement de l'algorithme (version § 4).

Principe de construction du squelette :

Calculer le saut minimum entre tout couple de successeurs immédiats (l'aîné et le benjamin) de toute classe édifiée en classification hiérarchique totale ; repérer les éléments de I qui réalisent les sauts ; noter la longueur des sauts puis à partir de cela construire le squelette arborescent et par dessus lui la hiérarchie comme cela est fait figure 3.

Algorithme simplifié :

En entrée - On lit le tableau des distances (rangées dans l'ordre lexicographique usuel) et les paramètres de construction de la hiérarchie (i.e. pour tout noeud N, les aînés A(N), et les benjamins B(N)).

Pour tout N variant de CARDI+1 (1° noeud édifié) jusqu'à (2*CARDI)-1 (dernier noeud édifié), on effectue les opérations suivantes :

- repérage de l'aîné et du benjamin de la classe en cours d'étude.

- repérage des éléments de I qui appartiennent à l'aîné [QRO(I)=1] au benjamin [QRO(I)=-1], qui n'appartiennent ni à l'un ni à l'autre [QRO(I)=0] (grâce au sous-programme DES).

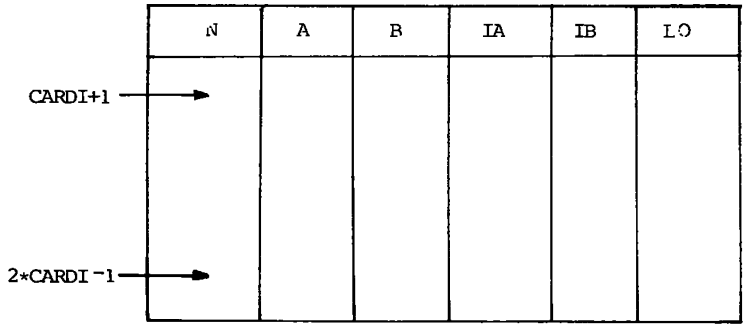
- Calcul du saut minimum dans une double boucle où on élimine le maximum de consultations en fonction des valeurs de QRO(I).

Le programme note alors l'élément de I de A(N) dans IA(N), l'élément de I de B(N) dans IB(N) et la longueur de l'arête entre IA(N) et IB(N) dans LO(N) qui est le saut minimum entre A(N) et B(N). fin.

En sortie - Le programme édite un tableau récapitulatif où sont notés :

- Les numéros de classe dans N.

- Les numéros des aînés dans A ; des benjamins dans B ; les extrémités des arêtes dans l'aîné dans IA, les extrémités des arêtes dans le benjamin dans IB et les longueurs des arêtes dans LO.



6. Le programme de construction du squelette arborescent

6.1 Le programme principal

```
C     +-------------------------------------------------------------+
C     |                                                             |
C     | PROGRAMME DE CONSTRUCTION D'UN SQUELETTE ASSOCIE            |
C     | A UNE CLASSIFICATION HIERARCHIQUE BINAIRE                   |
C     |                        D'APRES  JP BENZECRI                 |
C     |                        UNIVERSITE P. ET M. CURIE  PARIS     |
C     |                  PAR  M. JAMBU                              |
C     |                                                             |
C     +-------------------------------------------------------------+
C
C
      INTEGER CARDI,SA,SB,ALEC,DLEC,AN,BN
      INTEGER NOM(100),FMT(20),ARBRMT(20)
      INTEGER A(199),B(199),IA(199),IB(199)
      REAL    LO(199),DIS(4950),INF
      INTEGER     QRO(100),NUM(100),N(199)
C
C
C     CARDI - NOMBRE DES ELEMENTS DE I SUR LEQUEL ON A EFFECTUE
C     UNE CLASSIFICATION HIERARCHIQUE BINAIRE TOTALE (CF  CAH 75)
C     KCARD=2*CARDI-1
C     NCARD=CARDI*(CARDI-1)/2
C     DIS(NCARD) -TABLEAU DES  DISTANCES  ENTRE ELEMENTS DE I
C     A(KCARD) -TABLEAU CONTENANT LES AINES  DES CLASSES
C     B(KCARD) -TABLEAU CONTENANT LES BENJAMINS DES CLASSES
C     NOM(CARDI) - TABLEAU DES NOMS DES ELEMENTS DE I
C     IA(KCARD),IB (KCARD)  -TABLEAUX CONTENANT LES NUMEROS DES
C     ELEMENTS DE I  REALISANT LE SAUT MINIMUM
C     FMT(20) -TABLEAU CONTENANT LE FORMAT DE LECTURE DES DISTANCES
C     ARBRMT(20)-TABLEAU CONTENANT LE FORMAT DE LECTURE DES PARAMETRES
C     DE CONSTRUCTION  DE LA  CLASSIFICATION  HIERARCHIQUE
C     LO(KCARD) - TABLEAU CONTENANT LES LONGEURS DES LIENS DANS LE
C     SQUELETTE ARBORESCENT
C     QRO(CARDI),NUM(CARDI),N(KCARD) - TABLEAUX DE TRAVAIL
C
      LEX(I,J)=MINO(I,J)+((MAXO(I,J)-1)*(MAXO(I,J)-2)/2)
C
      READ 1,CARDI,ALEC,DLEC
    1 FORMAT(20I4)
      KCARD=2*CARDI-1
      NCARD=CARDI*(CARDI-1)/2
      LCARD=CARDI+1
      READ 2,FMT
    2 FORMAT(20A4)
      READ 2,ARBRMT
      DO 3 I=LCARD,KCARD
      READ (ALEC,ARBRMT) A(I),B(I)
    3 CONTINUE
      READ (DLEC,FMT) (DIS(K),K=1,NCARD)
C     LES DISTANCES SONT LUES DANS L'ORDRE LEXICOGRAPHIQUE USUEL (LEX)
      READ 2,(NOM(I),I=1,CARDI)
C
      PRINT 53
   53 FORMAT(10X,'TABLEAU DES LIENS FORMANT LE SQUELETTE'//)
      PRINT 52
   52 FORMAT(1X,'NIVEAU',1X,'AINE',1X,'BENJAMIN',1X,'EXTR AINE',1X,'EXTR
     1 BENJ',1X,'LONGEUR')
      PRINT 51
   51 FORMAT(1X,'  N',4X,' A ',4X,' B ',7X,' IA ',6X,' IB ',5X,'LO')
      DO 111 NOEUD=LCARD,KCARD
      AN=A(NOEUD)
      BN=B(NOEUD)
      DO 20I=1,CARDI
   20 QRO(I)=0
      CALL DES(AN,CARDI,KCARD,N     ,NUM,A,B,LY)
      DO 110 K=1,LY
      NUMK=NUM(K)
  110 QRO(NUMK)=1
      CALL DES(BN,CARDI,KCARD,N     ,NUM,A,B,LY)
      DO 1111 K=1,LY
      NUMK=NUM(K)
 1111 QRO(NUMK)=-1
      INF=10E+50
      IC=CARDI-1
      DO 200 I=1,IC
      IF(QRO(I).EQ.0) GO TO 200
      IIK=I+1
      DO 300 IP=IIK,CARDI
      IF(QRO(IP ).EQ.0) GO TO 300
       IL=QRO(IP)*QRO(I)
```

```
      IF(IL.EQ.1) GO TO 300
    KP=LEX(I,IP)
    IF(DIS(KP)-INF) 301,300,300
301 INF=DIS(KP)
     SA=I
     SB=IP
300 CONTINUE
200 CONTINUE
    IA(NOEUD)=SA
    IB(NOEUD)=SB
    LO(NOEUD)=INF
    PRINT 50,NOEUD,A(NOEUD),B(NOEUD),IA(NOEUD),IB(NOEUD),LO(NOEUD)
 50 FORMAT(1X,I3,4X,I3,4X,I3,7X,I3,7X,I3,3X,F5.2)
111 CONTINUE
    PRINT 75
 75 FORMAT(1H1)
    CALL IMPDIS(DIS,CARDI,NOM,NCARD)
    PRINT 75
    DO 120 K=1,NCARD
120 DIS(K)=10E+20
    DO 121 K=LCARD,KCARD
    IAK=IA(K)
    IBK=IB(K)
    KK=LEX(IAK,IBK)
121 DIS(KK)=LO(K)
    CALL IMPDIS(DIS,CARDI,NOM,NCARD)
    STOP
    END
```

6.2 Le sous programme D E S

```
      SUBROUTINE  DES(II,CARDI,KCARD,N,NUM,A,B,LY)
C
C     +-------------------------------------------------------------------+
C     | SOUS PROGRAMME DE DESCRIPTION D'UNE CLASSE PAR LES NUMEROS        |
C     | DE SES ELEMENTS DE BASE                                           |
C     |                                                                   |
C     +-------------------------------------------------------------------+
C
      INTEGER CARDI
      INTEGER NUM(CARDI),N(KCARD)
      INTEGER A(KCARD),B(KCARD)
      KI=1
      N(1)=II
      LY=0
  538 IF(N(KI)-CARDI) 535,535,540
  535 NPI=N(KI)
      LY=LY+1
      NUM(LY)=NPI
      KI=KI-1
      GO TO 534
  540 KJ=KI+1
      NPI=N(KI)
       N(KJ)=A(NPI)
      N(KI)=B(NPI)
      KI=KI+1
  534 IF(KI.NE.0) GO TO 538
      RETURN
      END
```

6.3 Le sous programme IMP DIS

```
      SUBROUTINE IMPDIS(DIS,CARDJ,NOM,NCARD)
C
C     +-----------------------------------------------------------------+
C     |                                                                 |
C     | SOUS PROGRAMME D'IMPRESSION DU TABLEAU DES DISTANCES            |
C     |                                                                 |
C     +-----------------------------------------------------------------+
C
      INTEGER CARDJ
      INTEGER NOM(CARDJ)
      REAL DIS(NCARD)
      KM=(CARDJ-1)/50+1
      LMAX=(CARDJ-1)/20+1
      DO 1 K=1,KM
       LF=K*5
      IF (K.EQ.KM)  LF=LMAX
      IG=K*50
      ID=IG-49
      IF(K.EQ.KM)  IG=CARDJ
      DO 1 L=1,LF
      JD=(L-1)*20+1
      JFP=JD+19
      IF(L.EQ.LMAX)  JFP=CARDJ
      PRINT 10,(NOM(J),J=JD,JFP)
   10 FORMAT(7X,20(1X,A4,1X))
       DO 2 I=ID,IG
      Z=I
      II=((Z-1)*(Z-2))/2.
      IMOM=I-1
      JF=MINO(JFP,IMOM)
      IF(JF-JD) 5,6,6
    5 PRINT11,NOM(I)
   11 FORMAT(1X,A4,20(1X,F5.2))
      GO TO 2
    6 MD=II+JD
      MF=II+JF
       PRINT 11,NOM(I),(DIS(M),M=MD,MF)
    2 CONTINUE
      PRINT 10,(NOM(J),J=JD,JFP)
      PRINT 75
   75 FORMAT(1H1)
    1 CONTINUE
      RETURN
      END
```

7. Edition des résultats fournis par le programme

Le programme assure l'impression de trois tableaux.

7.1 Tableau des liens édités par classe

TABLEAU DES LIENS FORMANT LE SQUELETTE

NIVEAU N	AINE A	BENJAMIN B	EXTR AINE IA	EXTR BENJ IB	LONGEUR LO
8	1	2	1	2	1.00
9	4	6	4	6	2.00
10	7	8	2	7	3.00
11	5	9	4	5	4.00
12	3	11	3	4	6.00
13	10	12	2	6	8.00

7.2 Tableau de contrôle des distances introduites dans le programme

	I1	I2	I3	I4	I5	I6	I7
I1							
I2	1.00						
I3	9.00	9.00					
I4	9.00	9.00	6.00				
I5	9.00	9.00	7.00	4.00			
I6	9.00	8.00	9.00	2.00	5.00		
I7	5.00	3.00	9.00	9.00	9.00	9.00	
	I1	I2	I3	I4	I5	I6	I7

7.3 Tableau récapitulatif des arêtes et des longueurs des arêtes

Les étoiles dans les cases d'indice (i,i') du tableau dis(i,i') indique l'absence de liaison entre les éléments i et i' de l'ensemble I.

	I1	I2	I3	I4	I5	I6	I7
I1							
I2	1.00						
I3	*****	*****					
I4	*****	*****	6.00				
I5	*****	*****	*****	4.00			
I6	*****	8.00	*****	2.00	*****		
I7	*****	3.00	*****	*****	*****	*****	
	I1	I2	I3	I4	I5	I6	I7