LES CAHIERS DE L'ANALYSE DES DONNÉES

C. DE RHAM

La classification ascendante hiérarchique basée sur la matrice des distances triées

Les cahiers de l'analyse des données, tome 7, n° 2 (1982), p. 163-168

<http://www.numdam.org/item?id=CAD_1982__7_2_163_0>

Les Cahiers de l'Analyse des Données
Vol. VII - 1982 - n° 2 - p. 163-168

LA CLASSIFICATION ASCENDANTE HIÉRARCHIQUE BASÉE SUR LA MATRICE DES DISTANCES TRIÉES [C.A.H. DIST. TRI]

par C. de Rham (1)

1 *Introduction*

On sait que l'algorithme de base de la classification hiérarchique ascendante (calcul à l'aide de la formule de récurrence à partir de la matrice des distances entre observations) prend beaucoup de place et beaucoup de temps, la place étant proportionnelle à I^2 et le temps à I^3.

Comme la place et le temps sont les deux facteurs qui limitent les applications pratiques de la classification, nous avons cherché des algorithmes nouveaux plus rapides et moins encombrants. Une solution a été exposée dans un article précédent sur l'application du principe des voisins réciproques (cf. références).

La solution présentée ici est très différente et part de l'idée que la plupart des $I^2/2$ calculs de distances pendant les itérations sont inutiles parce que l'algorithme de base les fait au mauvais moment !

En effet, l'algorithme de base recalcule toutes les nouvelles distances juste après la formation de chaque classe. Pourquoi ne pas faire l'inverse et ne calculer que le minimum de distances nécessaires avant de pouvoir former la classe suivante ?

La réalisation de cette idée exige le tri des distances entre observations et une nouvelle organisation des données mais permet d'obtenir des performances très supérieures à celles de l'algorithme de base.

Cet algorithme construit une hiérarchie exacte pour les stratégies du saut minimum et de la variance et heuristique (cela dépend des données) pour les stratégies du centre de gravité pondéré et non pondéré. Les stratégies du diamètre et de la distance moyenne n'ont pas de solution efficace.

L'algorithme proposé ici permet de constater également que l'on peut construire la hiérarchie avec un nombre très limité de calculs de distances. Par exemple, l'agrégation des données de Ruspini (75 observations) selon la variance demande 2701 calculs de distances avec l'algorithme de base et que 164 (!) soit environ 6% avec notre solution.

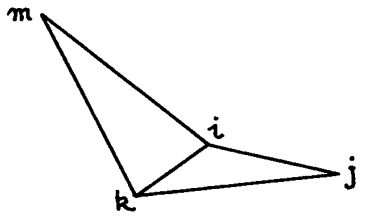
---

(1) *Diplômé de l'école polytechnique de Zürich, MBA INSEAD de Fontainebleau, Docteur-ès-Sciences.*
Lindenmattastrasse 40, CH-3065 BOLLINGEN, Suisse.

2 *L'algorithme basé sur la matrice des distances triées*

2.1 *Le principe* : Soit un ensemble $I_0 = \{i,j,k,m\}$ de 4 observations. L'initialisation, soit le choix d'une fonction de distance et le calcul de toutes les distances entre paires d'observations est identique à l'algorithme de base de la C.H.A. .

Admettons que l'on trie les $(4 \times 3)/2 = 6$ distances selon leur valeur croissante. Pour l'exemple suivant on aura :



i	k	d(i,k)
i	j	d(i,j)
j	k	d(j,k)
i	m	d(i,m)
k	m	d(k,m)
j	m	d(j,m)

Le couple (i,k) correspond à la plus petite distance, la première classe fusionnée sera donc la classe $i \cup k$.

L'algorithme de base recalcule les deux distances $d(i \cup k,j)$ et $d(i \cup k,m)$ immédiatemnt après la formation de la classe $i \cup k$. La plus petite distance suivante sera

$d(i \cup k,j)$ si $d(i \cup k,j) \leq d(i \cup k,m)$

ou $d(i \cup k,m)$ si $d(i \cup k,j) > d(i \cup k,m)$

On sait également que pour les stratégies monotones,

$\inf(d(i,j), d(k,j)) \leq d(i \cup k,j)$

et $\inf(d(i,m), d(k,m)) \leq d(i \cup k,m)$

$\Longrightarrow \quad d(i,j) \leq d(i \cup k,j)$

$d(i,m) \leq d(i \cup k,m)$

Admettons que l'on recalcule la distance entre $i \cup k$ et s

pour $s | d(i,s) = \inf(d(i,j), d(i,m))$.

Comme $d(i,j) < d(i,m)$ on recalcule la distance $d(i \cup k,j)$.

Si $d(i \cup k,j) < d(i,m)$

Alors $d(i \cup k,j) < d(i \cup k,m)$

et le calcul de $d(i \cup k,m)$ n'est pas nécessaire pour savoir que la classe suivante sera $i \cup k \cup j$ et non $i \cup k \cup m$.

2.2 *L'algorithme* : Soit l'ensemble I des observations.

Initialisation :

pas a) calcul des distances entre observations pour tout $i,j \in I$, $i < j$

pas b) tri des triplets (i,j,d(i,j)) selon la valeur croissante de d(i,j)

Parcourir une fois les triplets (i,j,d(i,j)) selon l'ordre croissant de d(i,j) :

Itérations :

pas c) faire pour le triplet (i,j,d(i,j)) :

- s'il existe une classe c telle que $i \subset c$ et $j \subset c$, aller au pas g)

- si $d(i,j) \leq d(i',j')$ pour tout $i', j' \in I$, aller au pas d)

- si non aller au pas e)

pas d)-former la classe $c = i \cup j$

- calculer le centre de gravité de la classe c

- faire I:=I-{i}-{j}+ c

- si |I| = 1: STOP

- si non aller au pas g)

pas e)-déterminer les classes a telles que $i \subset a$ et b telles que $j \subset b$

- calcul de la distance d(a,b) sur la base des centres de gravité des classes a et b

- classer le triplet (a,b,d(a,b)) selon la valeur de d(a,b) parmi les triplets déjà classés.

pas f) éliminer les triplets (r,s,d(r,s) pour tout r et s tels que :

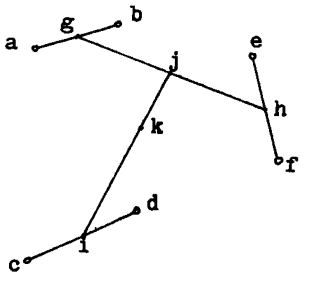
(i) $r \subset a$

(ii) $s \subset b$

(iii) $d(i,j) \leq d(r,s) < d(a,b)$

pas g) aller au triplet suivant (i,j,d(i,j)) , aller au pas c)

2.3 *Exemple numérique* : Soit un ensemble I = {a,b,c,d,e,f} de 6 observations représentées par des points dans un espace à deux dimensions avec la condition ni = 1 pour tout $i \in I$.



coordonnées		
a	0	40
b	[illegible]	45
c	15	0
d	20	10
e	40	40
f	45	20

Initialisation :

pas a) calcul des 15 distances d(i,j) pour la stratégie selon la variance :

$$d^2(i,j) = ((ni.nj)/(ni+nj)) \cdot \| Xi - Xj \|^2$$

pas b) trier les triplets (i,j,d(i,j)) selon la valeur de d(i,j)

i	j	$d^2(i,j)$
a	b	125
e	f	213
c	d	250
b	e	325
d	f	363
b	d	625
a	d	650
d	e	650
b	f	763
a	c	800
a	e	800
b	c	1125
a	f	1213
c	f	1213
c	e	1600

Itérations

itération	pas c) faire pour le triplet	pas d) former la classe	pas e) calculer la distance	pas f) tracer	pas g) aller à
1	a,b,213	g = a ∪ b			e,f,213
2	e,f,250	h = e ∪ f			c,d,250
3	c,d,250	i = c ∪ d			b,e,325
4	b,e,325		d(g,h)=1381		
				b,f,763 a,e,800 a,f,1213	d,f,363
5	d,f,363		d(h,i)=1681		
				d,e,650 c,f,1213 c,e,1600	b,d,625
6	b,d,625		d(i,g)=1413		
				a,d,650 a,c,800 b,c,1125	g,h,1381
7	g,h,1381	j = g ∪ h			g,i,1413
8	g,i,1413		d(i,j)=1602		i,j,1602
9	i,j,1602	k = i ∪ j	fin des calculs		

triplets classés :	après l'itération 4	après l'itération 5	après l'itération 6
	d,f,363 b,d,625 a,d,650 d,e,650 a,c,800 b,c,1125 c,f,1213 g,h,1381 (nouv.) c,e,1600	b,d,625 a,d,650 a,c,800 b,c,1125 g,h,1381 h,i,1681 (nouv.)	 g,h,1381 g,i,1413 (nouv.) h,i,1681

2.4 *L'application pratique* : Il est évident que l'algorithme exposé sous 2.2 n'est pas directement utilisable pour l'application pratique, car il implique la mémorisation des triplets en mémoire centrale (place) et le classement des nouvelles distances parmi ceux-ci (temps).

Nous avons donc programmé une application dans laquelle la matrice des distances triées se trouve sur un support externe (bande, disque) et n'est parcourue qu'une fois séquentiellement. Les nouvelles distances calculées en cours d'itérations sont classées dans une liste en mémoire centrale. La longueur maximale de cette liste dépend de la stratégie et des données mais reste petite (env. |I|/2 à |I| éléments).

3 *Comparaison des performances*

Les performances ont été comparées sur trois ensembles de données construits à l'aide d'un programme de génération de hiérarchies artificielles et sur les données de Ruspini. Les comparaisons concernent le nombre de calculs de distances, la place en mémoire centrale et le temps d'exécution.

L'algorithme de base (BASE) sert de référence. Cet algorithme calcule la hiérarchie à l'aide de la formule de récurrence pour les sept stratégies que nous avons numérotées comme suit :

1 = saut minimum
2 = diamètre
3 = distance moyenne non pondérée
4 = distance moyenne pondérée
5 = centre de gravité non pondéré
6 = centre de gravité pondéré
7 = variance

TRI est la méthode proposée ici. Les mesures ont été faites sur le CDC 6500 de Fides à Zürich. La place est mesurée en mots, le temps en ms. Le temps (nécessaire au calcul de la hiérarchie) est une moyenne des stratégies 1 à 7 pour BASE. Comme les résultats de TRI varient d'une stratégie à l'autre, nous les donnons séparément pour les stratégies 1, 5+6 et 7.

Tableau comparatif des performances de l'algorithme de base (BASE) et de l'algorithme selon la matrice des distances triées (TRI).

grandeur comparée	algor.	strat	dimension de l'ensemble (obs × var)							
			(32 × 2)		(64 × 2)		(128 × 2)		(75 × 2) (*)	
			abs	%	abs	%	abs	%	abs	%
nombre de calc. de dist.	BASE	1-7	465	100	1953	100	8001	100	2701	100
	TRI	1	0	0	0	0	0	0	0	0
		5-6	37	8	52	3	73	1	105	4
		7	53	11	90	5	155	2	164	6
place en mémoire centrale	BASE	1-7	1088	100	4224	100	16640	100	5775	100
	TRI	1	160	15	320	8	640	4	375	6
		5-6	210	19	370	9	690	4	480	8
		7	245	23	465	11	860	5	580	10
temps calcul	BASE	1-7	201	100	1197	100	8191	100	1581	100
	TRI	tri	80	40	400	33	1827	22	570	36
		1	42	21	81	7	166	2	65	6
		5-6	89	44	262	2	886	11	344	22
		7	131	65	559	49	3053	37	714	45

(*) *données de Ruspini*

4 Conclusions

L'algorithme de classification hiérachique ascendante proposé dans cet article permet de faire deux constatations : Premièrement que les possibilités de trouver des algorithmes plus performants que l'algorithme de base sont loin d'être épuisées. Deuxièmement que la construction de la hiérarchie est possible avec un nombre de calculs de distances très faible comparé à l'algorithme de base.

Nous espérons que l'application de cet algorithme permettra d'élargir encore le domaine d'application de la classification hiérarchique.

5 Références

BENZECRI J.P. (1973) L'Analyse des Données, T I et II DUNOD, Paris.

BOCK H.H. (1974) Automatische Klassifikation, Vandenhoek & Rupprecht, Göttingen.

JAMBU & LEBEAUX (1978) Classification automatique pour l'analyde des données, T 1 et 2, DUNOD, Paris.

LANCE & WILLIAMS (1973 Hierarchical Classificatory Methods Statisticals Methods for Digital Computers, Wiley.

DE RHAM C. (1980) La CHA selon la méthode des voisins réciproques, Cahiers de l'Analyse des Données, Vol. V n° 2, pp 135-144.