MATHÉMATIQUES ET SCIENCES HUMAINES

A. GUENOCHE

Cinq algorithmes d'approximation d'une dissimilarié par des arbres à distances additives

Mathématiques et sciences humaines, tome 98 (1987), p. 21-40

<http://www.numdam.org/item?id=MSH_1987__98__21_0>

Math. Sci. hum. (25^{e} année, n°98, *1987*, p.21-40)

CINQ ALGORITHMES D'APPROXIMATION D'UNE DISSIMILARITE PAR DES ARBRES A DISTANCES ADDITIVES

A. GUENOCHE *

Dans cet article de synthèse, nous exposons cinq algorithmes récents de construction d'un arbre à distances additives, à partir d'un tableau de dissimilarités D qui quantifie les écarts entre N objets. L'un de ces algorithmes est purement métrique, puisqu'il évalue directement à partir de D une distance additive d'arbre, les autres construisent des arbres dont les arêtes sont valuées et à partir desquels on reconstruit une distance arborée "voisine" des dissimilarités initiales. Ils se distinguent de ceux de Carroll [4], Cunningham [5] et De Soëte [6] car ils ne résolvent pas ce problème par une méthode "brutale" d'optimisation. Nous ne reprenons pas les définitions et les propriétés des notions mathématiques utilisées; elles sont largement développées par ailleurs dans [1, 2, 10]. Cette description des algorithmes précise les structures adéquates de données, ce qui permet leur implémentation dans n'importe quel langage procédural.

Plutôt que de classer ces méthodes suivant l'importance de leur approche métrique versus combinatoire, nous les avons rangées dans un ordre de complexité croissante. Ce critère n'est pas pleinement satisfaisant, car seule la complexité "théorique" d'une itération est prise en compte, alors que le nombre moyen d'étapes et la dimension du tableau traité à chaque étape interviennent fortement dans les temps de calcul.

Etant donné une structure d'arbre et un tableau de dissimilarités dont les entrées sont les sommets pendants de l'arbre, on peut, par un algorithme d'approximation sous contraintes positives, calculer des longueurs d'arêtes, de façon que les longueurs des chemins entre sommets pendants, mesurées dans l'arbre, approximent au mieux les dissimilarités. Si l'on a déjà un arbre valué, on peut toujours conserver la structure de l'arbre et recalculer de nouvelles longueurs de façon à améliorer, au sens des moindres carrés, la représentation des voisinages qu'ils produisent. Ceci nous permet de comparer les algorithmes d'abord sur des critères non métriques qui ne dépendent que des structures d'arbres, ensuite sur des critères métriques calculés à partir des arbres valués par les algorithmes,

* G.R.T.C. - C.N.R.S 31 Ch. J. Aiguier 13402 Marseille Cedex 9

puis des arbres ré-évalués par approximation quadratique. Ces expérimentations numériques sont réalisées à partir de 6 tableaux de dissimilarités de dimension 7. C'est dire qu'à chaque méthode correspond un programme que nous diffusons sans restriction.

1 ALGORITHMES DE CONSTRUCTION D'ARBRES A DISTANCES ADDITIVES

Ces algorithmes ne sont pas tous originaux. Ils ont été élaborés dans le cadre d'un groupe de travail (J.P. Barthélemy, G. Brossier, A. Guénoche, B. Leclerc, X. Luong, M. Roux) et parfois publiés sous une forme légèrement différente et nettement moins détaillée. Nous indiquons ici l'auteur principal de chaque méthode et renvoyons à une éventuelle publication. Tous satisfont au principe suivant: si le tableau de dissimilarités est une distance additive d'arbre, ils déterminent exactement l'arbre sous-jacent.

La première méthode est basée sur le principe de décomposition d'une distance arborée en somme d'une distance à centre et d'une distance ultramétrique. Un algorithme a été développé par G. Brossier [3]. Il est ici décrit dans le détail et utilise, pour construire une ultramétrique, une implémentation de la méthode du lien moyen dans laquelle on porte une attention toute particulière au traitement des valeurs égales de dissimilarité.

La seconde méthode, dite des plus proches prédécesseurs, est originale. Elle est fondée sur la façon de calculer les longueurs d'arêtes dans un arbre à distance additive, connaissant seulement les longueurs des chemins entre sommets pendants. Considérons un tracé planaire de l'arbre et un ordre circulaire des sommets pendants. Si {i,j,k} sont trois sommets pendants consécutifs pour cet ordre, la quantité (D(i,j)+D(i,k)-D(j,k))/2 est égale à la longueur L(j) de l'arête issue de j. Maintenant, si deux sommets pendants i,j sont adjacents au même sommet, on a D(i,j)=L(i)+L(j). Dans cet algorithme,on regroupe les sommets qui vérifient cette équation.

La troisième méthode, dite de dispersion est basée sur une idée de B. Leclerc. Si i et j sont deux sommets pendants d'un sous-arbre de racine s, la variance, lorsque k parcourt l'ensemble des sommets pendants qui ne sont pas dans ce sous-arbre, de D(k,i)-D(k,j) est nulle. Le principe de cet algorithme est de regrouper les sommets pour lesquels cette quantité est minimum.

La quatrième méthode, dite des scores, est une amélioration d'une méthode définie à l'issue de la notion de scores d'une paire de sommets, par S. Sattah et A. Tversky [10]. En définissant des scores stricts et larges, et en traitant les scores voisins, X. Luong [2] a défini un algorithme plus performant qui est décrit ici.

La cinquième méthode est un algorithme itératif de modification des valeurs de dissimilarités, de façon à satisfaire aux inégalités quadrangulaires. Il est dû à M. Roux [8] qui reprend là l'idée qu'il a déjà appliquée à l'approximation d'une dissimilarité par une ultramétrique. Sa méthode ne fournit qu'une distance additive d'arbre; n'importe lequel des algorithmes précédents construit l'arbre.

Avant de présenter les différentes méthodes, précisons quelques points communs à toutes les implémentations.

Chaque méthode traite une matrice D de dissimilarités de dimension N symétrique de diagonale nulle. Ces méthodes itératives construisent des "groupements", c'est à dire une réunion d'un ensemble de sommets pendants qui se regroupent sur un nouveau sommet numéroté à la suite. Un groupement de n_g sommets a pour effet d'ajouter autant d'arêtes à l'arbre et un sommet numéroté à la suite. Les sommets regroupés sont supprimés du tableau D et le nouveau sommet devient pendant. Ses écarts aux sommets restants sont calculés et occupent la place du premier sommet du groupement. N est diminué et tant que N>3 on effectue une nouvelle itération.

L'étiquette associée au i-ième sommet est stockée dans un tableau d'identificateurs ID. A la fin de chaque algorithme, l'arbre à Ns sommets est codé dans un tableau T. Ses Ns-1 arêtes sont i--T(i). Le sommet étiqueté Ns n'a pas de prédécesseur, ce qui le désigne comme racine de l'arbre. Les longueurs des arêtes sont codées dans un tableau L. On appelle Max la somme des N(N-1)/2 valeurs de D.

1.1 Méthode de décomposition

Cet algorithme est fondé sur le principe de décomposition d'une distance additive d'arbre, en somme d'une distance à centre et d'une ultramétrique, développé, par exemple, dans [4]. Cette décomposition n'est pas unique, puisque l'on peut choisir comme centre c n'importe quel point situé sur l'arbre correspondant à cette distance arborée. Pour notre problème d'approximation, les seuls points connus sur l'arbre sont les sommets pendants, ce qui n'est pas très naturel comme choix d'un centre. Néanmoins, on retient la médiane du tableau de dissimilarités, soit le centre c tel que $\Sigma_k D(c,k)$ soit minimum.

La distance à centre DC(i) de tout sommet i au centre c est alors estimée par la valeur de dissimilarité, DC(i)=D(i,c), et l'on calcule une nouvelle dissimilarité D' en retirant à D cette distance à centre. Il se peut que les valeurs de D' ne soient pas toutes positives. On peut alors ajouter à D une constante dont il faudra évidemment tenir compte dans le calcul des longueurs d'arêtes de l'arbre de classification.

Après il reste à appliquer à D' un algorithme d'approximation par une ultramétrique - nous avons choisi la méthode du lien moyen -, ce qui définit un arbre de classification. Si à la dernière itération il ne reste que deux sommets, on n'ajoute pas de sommet supplémentaire pour ce dernier groupement, mais on crée une seule arête dont le sommet fin est le numéro de sommet le plus élevé qui devient la racine de l'arbre.

Les longueurs d'arêtes de l'arbre sont alors égales aux longueurs d'arêtes de l'arbre de classification augmentées pour les seuls sommets pendants des distances à centre.

Algorithme de décomposition

1/ Choix du centre

```
Min := Max
Pour I variant de 1 à N Faire  S := 0
    Pour J variant de 1 à N Faire
        S := S + D(I,J)
    Si S<Min Alors C := I ; Min := S
```

2/ Soustraction de la distance à centre

```
Pour I variant de 1 à N Faire
    DC(I) := D(C,I)
Pour I variant de 1 à N Faire
    Pour J≠I variant de 1 à N Faire
        D(I,J) := D(I,J) - DC(I) - DC(J)
```

3/ Construction de l'ultramétrique du lien moyen

Classiquement on réunit à chaque étape de l'algorithme les deux sommets i et j les plus voisins, à un seuil S=D(i,j). Mais il arrive fréquemment, du fait de valeurs de dissimilarités égales, en particulier si la dissimilarité est la distance de la différence symétrique obtenue à partir d'un tableau en 0/1, que d'autres classes aient la même dissimilarité. Deux cas sont à distinguer:

. Il existe deux autres classes k et l telles que D(i,j) = D(k,l)
. Il existe une autre classe k telle que D(i,j) = D(i,k)

Dans le premier cas, on accélère simplement l'algorithme en regroupant à la même étape i,j d'une part et k,l d'autre part. Dans le deuxième cas, pour ne pas privilégier l'une des paires on regroupe en un même sommet i, j et k. S'il s'agit d'une méthode de lien unique, cette décision est tout à fait conforme à l'esprit de la méthode. S'il s'agit d'une méthode de lien moyen, il faudra considérer la moyenne des valeurs D(i,j), D(i,k) et D(j,k) qui définit un nouveau seuil S' de groupement des sommets i, j et k. Le seuil initial S peut être insuffisant et le choix de l'une des paires (i,j) ou (i,k) est arbitraire. Le groupement i,j,k ne peut être fait au seuil S' que si aucun autre groupement n'apparaît à un seuil intermédiaire. L'existence de valeurs égales bloque donc les regroupements et rend ainsi le résultat indépendant de l'ordre d'examen des sommets, donc de leur numérotation. En traitant de cette façon les valeurs égales, on construit un arbre de classification qui n'est plus nécessairement binaire.

Pour calculer les longueurs, définissons le niveau de chaque sommet correspondant à une classe de la hiérarchie. Les sommets pendants de l'arbre de classification ont un niveau 0. Si deux sommets i et j de niveaux N_i et N_j se regroupent sur un sommet k au seuil S=D(i,j), le niveau de k est donné par $N_k=(S+N_i+N_j)/2$. En effet les longueurs des arêtes i--k et j--k sont respectivement N_k-N_i et N_k-N_j dont la somme est égale au seuil S. On a ainsi un arbre de classification avec des arêtes de longueurs valuées. C'est ce que l'on fait implicitement dans le tracé des dendrogrammes.

3.1/ Branchement suivant les valeurs de N

Si N = 1 Aller en 4/
Si N = 2 Aller en 3.4/
Sinon le seuil S est égal à la valeur minimum des dissimilarités. On définit SS comme la plus petite des valeurs de dissimilarité supérieure à ce seuil.

3.2/ Construction des groupements

Le tableau de dissimilarité peut être considéré comme un graphe complet valué par ces valeurs. Le choix du seuil précédent définit un "graphe seuil", dont les arêtes sont celles qui ont une valeur inférieure ou égale à S. On construit les parties connexes de ce graphe seuil. Cette façon de procéder augmente la complexité de la méthode du lien moyen, mais la certitude d'obtenir un résultat indépendant de la numérotation justifie, à nos yeux ce choix.

Pour chaque sommet on détermine le numéro de sa composante connexe codée dans un tableau NC initialisé à 0; NG est le nombre de groupements.

```
Pour Toute arête (I<J) Telle que D(I,J) <= S  Faire
   Si NC(I) = NC(J) = 0  Alors
      NG := NG + 1 ; NC(I) := NG ; NC(J) := NG
   Si NC(I) = 0 ET NC(J) > 0 Alors NC(I) := NC(J)
   Si NC(I) > 0 ET NC(J) = 0 Alors NC(J) := NC(I)
   Si NC(I) > 0 ET NC(J) > 0 Alors
      NCI := Min (NC(I),NC(J)) : NCJ := Max (NC(I),NC(J))
      Pour Tout K Tel que NC(K) > NCI  Faire
         Si NC(K)=NCJ  Alors  NC(K) := NCI
         Sinon NC(K) := NC(K) - 1
```

Après exécution de cet algorithme qui est en $O(N^3)$, tous les sommets connexes ont même valeur positive dans NC, et il y a NG composantes connexes, donc NG groupements éventuels.
Pour chaque partie connexe, on calcule MD, la moyenne des dissimilarités des paires de sommets, CG le cardinal de cette partie connexe et SNIV la somme des niveaux de ses éléments. Si MD est inférieure ou égale à S:
. on effectue le groupement en créant un nouveau sommet Ns,
. on crée les arêtes correspondantes,
. on calcule les longueurs d'arêtes,
sinon on stocke dans SM la plus petite des valeurs MD des différentes parties connexes.

```
Pour K variant de 1 à NG faire
   MD := 0 ; CG := 0 ; SNIV := 0
   Pour I variant de 1 à N Faire
      Si NC(I) = K Faire
         SNIV := SNIV + NIV(I)
         Pour J variant de I+1 à N Faire
            Si NC(J) = K Faire MD := MD + D(I,J) ; CG := CG + 1
   MD := MD / CG ; SNIV := SNIV / CG
   Si MD <= S Alors
```

```
        NS := NS + 1 ; NIV(NS) := (S - SNIV) / 2
        Pour I variant de 1 à N Faire
           Si NC(I) = K Alors
              II := ID(I) ; T(II) := NS ; L(II) := NIV(NS) - NIV(I)
     Sinon
        Si MD < SM Alors SM := SD
```

Si au moins un groupement a été effectué, on passe à l'itération suivante en retournant en 3.1, sinon on définit un nouveau seuil comme la plus petite des valeurs SM et SS (nécessaire pour effectuer au moins un groupement)

3.3/ Mise à jour du tableau D

Les dissimilarités d'un sommet de groupement aux autres sommets sont calculées comme les moyennes des distances de ces sommets aux éléments du groupement. Les sommets qui figurent dans un groupement sont supprimés et ceux des groupements ajoutés à la place du premier sommet du groupement dans D et NIV. On retourne alors à l'étape 3.1 avec une valeur de N plus petite que celle de l'itération précédente.

3.4/ Il reste deux sommets

Le sommet de plus grand numéro devient racine de l'arbre

```
   Si ID(1)>ID(2)  Alors Echanger (ID(1),ID(2))
   I := ID(1) ; T(I) := ID(2) ; L(I) := D(1,2)
```

4/ Calcul des longueurs

Aux longueurs d'arêtes définies par la méthode du lien moyen, on ajoute les distances à centre pour les arêtes pendantes.

```
   Pour I variant de 1 à N Faire
      L(I) := L(I) + DC(I)
```

1.2 Méthode des plus proches prédécesseurs

Supposons que l'on ait un arbre valué correspondant à la distance additive d'arbre D, et pour simplifier les notations admettons que les sommets sont étiquetés dans l'ordre circulaire d'une représentation planaire de l'arbre, c'est à dire sans intersection d'arêtes en dehors des sommets.

Soit L(i) la longueur de l'arête incidente au sommet pendant i. On a alors

$$L(i) = (D(i-1,i) + D(i,i+1) - D(i-1,i+1)) / 2$$

et cette quantité L(i) que l'on cherche à déterminer à partir de D, puisqu'on ne connait pas l'arbre est:

$$L(i) = (\min_{j \neq k \neq i} D(i,j) + D(i,k) - D(j,k)) / 2$$

On calculera donc pour chaque sommet pendant la longueur de son arête, comme le minimum des valeurs de L(i) lorsque j et k parcourent l'ensemble des sommets pendants autres que i. C'est la longueur du pas qu'il faut faire depuis i pour trouver un nouveau sommet de l'arbre.

Maintenant si i et j sont deux sommets pendants qui ont même prédécesseur s, la quantité D(i,j)-L(i)-L(j) est nulle. C'est ainsi que l'on va repérer les sommets adjacents. Si D(i,j)-L(i)-L(j) =0 alors il existe dans l'arbre un sommet s tel que i--s et j--s sont deux arêtes de l'arbre. On effectuera donc ce groupement. Une fois les arêtes enregistrées, on supprime les sommets i et j et le sommet s devient pendant. On calcule donc son écart à un sommet k quelconque comme

$$D(s,k) = D(i,k)-L(i) = D(j,k)-L(j)$$

On est ramené à un problème à N-1 sommets pendants, pour lequel on appliquera la même méthode, jusqu'à ce que l'on ait N=3.

On a donc retrouvé, à partir d'une distance arborée, son arbre support. A chaque itération on examine l'ensemble des triplets {i,j,k} pour calculer les longueurs d'arêtes; c'est l'opération la plus longue, si bien que chaque itération est en $O(n^3)$.

Maintenant nous nous posons le même problème, à partir d'une dissimilarité quelconque D. Si D n'est pas une distance, on trouvera des arêtes de longueur négative, et le plus souvent, aucun des écarts entre prédécesseurs n'est nul. Nous avons donc adapté l'algorithme de la façon suivante:

- On calcule une longueur positive L(i) c'est à dire que si D(i,j)+D(i,k)-D(j,k) est négative, cette quantité n'est pas prise en compte dans le calcul du minimum.
- On cherche le couple {i,j} tel que la quantité D(i,j)-L(i)-L(j) soit minimum; soit dmin sa valeur. On regroupe donc les sommets qui ont les plus proches prédécesseurs, d'où le nom de cette méthode.
- On répartit proportionnellement l'écart dmin entre L(i) et L(j) qui sont multipliées par 1+dmin/(L(i)+L(j)). Avec ces nouvelles valeurs on a D(i,j)-L(i)-L(j)=0.
- Les quantités D(i,k)-L(i) et D(j,k)-L(j) ne sont plus nécessairement égales; on calculera D(s,k) comme leur moyenne.

Algorithme des plus proches prédécesseurs

1/ Calcul des longueurs d'arêtes

```
Pour I variant de 1 à N  Faire
   Lmin=Max
   Pour tout J<K différents de I Faire
      L := D(I,J) + D(I,K) - D(J,K)
      Si  0≤L<Lmin  Alors  Lmin := L
   LL(I) := Lmin / 2
```

2/ Recherche des plus proches prédécesseurs

```
Dmin := Max
Pour tout I<J  Faire
   D := D(I,J) - LL(I) - LL(J)
   IF D<Dmin  Faire  II := I ; JJ := J ; Dmin := D
NS := NS + 1 ; I := ID(II) ; J := ID(JJ)
T(I) := NS ; T(J) := NS
Alpha := 1 + Dmin / ( LL(II) + LL(JJ) )
L(I) := LL(II) * Alpha ; L(J) := LL(JJ) * Alpha
```

3/ Mise à jour du tableau de dissimilarité

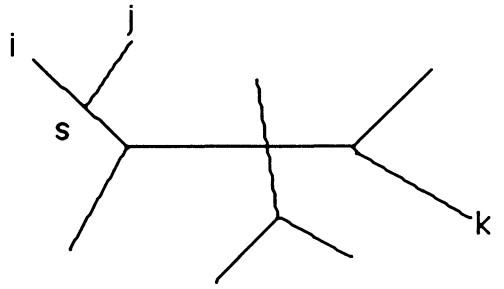
```
Pour I variant de 1 à N  Faire
   D(I,II) := ( D(I,II) + D(I,JJ) - L(II) - L(JJ) ) / 2
   D(II,I) := D(I,II)
ID(II) := NS ; D(II,II) := 0
Pour I variant de 1 à N-1  Faire
   D(I,JJ) := D(I,N) ; D(JJ,I) := D(N,I)
D(JJ,JJ) := 0 ; ID(JJ) := ID(N)
N := N - 1 ; Si N>3  Aller en 1/
```

4/ Il reste 3 sommets

```
I := ID(1) ; J := ID(2) ; K := ID(3)
NS := NS + 1
T(I) := NS ; T(J) := NS ; T(K) := NS
LG(I) := D(1,2) + D(1,3) - D(2,3)
LG(J) := D(1,2) + D(2,3) - D(1,3)
LG(K) := D(1,3) + D(2,3) - D(1,2)
```

1.3 Méthode de dispersion

Le principe de cet algorithme est le suivant: Si l'on est en présence d'une distance additive d'arbre à N sommets pendants, dont deux sommets i et j se regroupent en un sommet intérieur s, pour tout autre sommet k pendant placé par rapport à i et j de l'autre côté de s, la quantité $d_K(i,j)=D(k,i)-D(k,j)$ est indépendante de k.



L'idée de cet algorithme d'approximation est de regrouper les deux sommets pendants tels que la variance de $d_K(i,j)$ lorsque k parcourt l'ensemble des sommets pendants soit minimum. Comme pour la méthode précédente, chaque étape de cet algorithme est donc de complexité $O(N^3)$.

Les longueurs L(i) et L(j) des arêtes i--s et j--s se calculent par: Soit B l'estimation de L(i)-L(j),

$$B = (\Sigma_K (D(k,i)-D(k,l)) / (N-2).$$

Puisque L(i)+L(j)=D(i,j), il vient L(i)=(D(i,j)+B)/2 et L(j)=(D(i,j)-B)/2. Il arrive que certaines longueurs soient négatives, ce qui fait que cette méthode peut donner de très mauvais résultats (Cf. § 2.2). Pour calculer les écarts métriques ces longueurs négatives sont mises à 0.

Algorithme de dispersion

1/ Calcul des dispersions

Pour chaque paire (i,j) de sommets, on calcule une dispersion égale à la variance de D(k,i)-D(k,j) pour k variant de 1 à N, avec K≠I et K≠J. Ces dispersions sont rangées dans la partie supérieure droite d'un tableau V(N,N). La partie inférieure gauche contient la somme des écarts D(k,i)-D(k,j).

```
Pour tout I<J<K Faire
   A := D(K,I) - D(K,J) ; B := D(J,I) - D(J,K) ; C := D(I,J) - D(I,K)
   V(I,J) := V(I,J) + A * A ; V(J,I) := V(J,I) + A
   V(I,K) := V(I,K) + B * B ; V(K,I) := V(K,I) + B
   V(J,K) := V(J,K) + C * C ; V(K,J) := V(K,J) + C

Pour tout I<J Faire
   V(I,J) := V(I,J) - V(J,I) * V(J,I) / (N-2)
```

2/ Choix du groupement

On regroupe la paire de sommets dont la dispersion est minimum. Soit (II,JJ) cette paire et NS le numéro de sommet de ce groupement (numéroté à la suite).

3/ Calcul des longueurs

Les longueurs des arêtes II--NS et JJ--NS sont respectivement:

```
   LI := ( D(II,JJ) + V(JJ,II) / (N-2) ) / 2 ; LJ := D(II,JJ) - LI
   Si V(II,JJ) a été réajustée Aller en 3.2/
```

3.1/ Si l'une de ces longueurs est négative, la dispersion ne peut plus être calculée suivant la même formule; elle est alors réajustée.

```
   Si LI<0 Alors
      V(II,JJ) := V(II,JJ) + (N-2) * ( D(II,JJ) + (V(JJ,II)/(N-2)) )^2
      Aller en 2/
   Si LJ<0 Alors
      V(II,JJ) := V(II,JJ) + (N-2) * ( D(II,JJ) - (V(JJ,II)/(N-2)) )^2
      Aller en 2/
3.2/ NS := NS + 1
    On ajoute les arêtes
    II--NS de longueur Max(LI,0)
    JJ--NS de longueur Max(LJ,0)
```

4/ Mise à jour et tassement du tableau de dissimilarités.

La dissimilarité du sommet groupement NS aux autres sommets est alors calculée comme s'il s'agissait d'une distance additive d'arbre.

```
   Pour I variant de 1 à N Faire
      D(I,NS) := ( D(I,II) + D(I,JJ) - D(II,JJ) ) / 2
```

Les sommets II et JJ sont supprimés et NS rajouté, N est abaissé d'une unité, suivant la même procédure que précédemment.

Si N>3 on retourne en 1/ sinon on fait comme pour la méthode précédente.

1.4 Méthode des scores

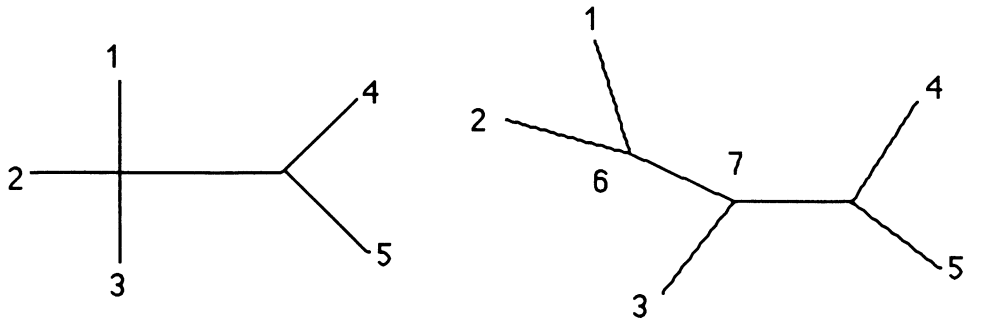
Par définition une distance additive d'arbre vérifie les inégalités "quadrangulaires", c'est à dire que pour tout quadruplet {i,j,k,l} il existe au moins une bi-partition en deux classes de deux éléments, ici {i,j} et {k,l}, telle que $D(i,j)+D(k,l) \leq Inf(D(i,k)+D(j,l),D(i,l)+D(j,k))$

S. Sattah et A. Tversky [10] définissent le score d'une paire de sommets, comme le nombre de quadruplets tels que cette paire est l'une des deux classes de la bi-partition. Le calcul des scores de chaque paire se fait en examinant tous les quadruplets; Une étape de l'algorithme sera donc de complexité $O(N^4)$.

Ceci fait, on regroupe la paire de score maximum, ce qui crée un nouveau sommet dont la dissimilarité à un autre sommet pendant, est égale à la moyenne des dissimilarités entre ce sommet et les sommets regroupés. A chaque itération, on abaisse la dimension du problème d'une unité. Quand il reste trois sommets, on les regroupe en un seul sommet supplémentaire qui devient la racine de l'arbre.

Cet algorithme, très naturel, présente néanmoins quelques défauts. D'abord il ne calcule pas de longueurs d'arêtes; il faudra comme le suggèrent les auteurs les approximer au sens des moindres carrés, au même titre que la valuation des arêtes d'un arbre de classification. Ensuite, cet algorithme ne peut produire qu'un arbre binaire. Si l'on opère à partir d'une distance additive d'arbre dont certains sommets sont de degré supérieur à 3, on ne retrouvera pas strictement la même structure d'arbre.

Exemple



En partant d'une distance correspondant à l'arbre de gauche, on va regrouper les sommets 1 et 2 sur un sommet 6, puis les sommets 3 et 6 sur un sommet 7. On obtiendra donc la structure de l'arbre de droite.

Bien évidemment, quand on calcule les longueurs d'arêtes, on trouve que la longueur de 6--7 est nulle, mais on a introduit des sommets inutiles, ce qui augmente la dimension du problème d'approximation des longueurs.

L'algorithme de Sattah et Tversky construit donc un arbre dont les sommets intérieurs sont de degré 3 en N-2 itérations. Ce nombre d'itérations peut paraître lui aussi excessif, dans la mesure où le regroupement de 4 et 5 peut s'opérer à la

même itération que celui de 1,2,3. L'algorithme de X. Luong, en gérant les scores ex aequo, construit des arbres dont les sommets intérieurs sont de degré au moins 3, en un nombre d'itérations égal à la profondeur de l'arbre.

Algorithme des scores

1/ Branchement suivant les valeurs de N

Si N = 1 Fin
Si N = 2 Aller en 8/
Si N = 3 Aller en 9/

2/ Calcul des scores

Pour chaque paire de sommets {i,j} on calcule son score égal au nombre de quadruplets {i,j,k,l} tels que:

$$D(i,j) + D(k,l) \leq \{ D(i,k) + D(j,l) , D(i,l) + D(j,k) \}$$

Le signe ≤ entraine le calcul de scores larges. Il faut également calculer des scores stricts en ôtant les cas d'égalité. Pour le tableau C(N,N) des scores, la moitié supérieure droite contient les scores larges, et la moitié inférieure gauche contient les nombres d'égalités. On traite les scores voisins en multipliant par 1+Eps la plus petite des trois sommes.

Pour calculer ces scores, définissons une fonction INC(i,j) qui incrémente d'une unité le score de la paire {i,j}.

```
Pour tout (I,J,K,L)  Faire
   A := D(I,J) + D(K,L) ; B := D(I,K) + D(J,L) ; C := D(I,L) + D(J,K)
   MIN := min(A,B,C)  ;  E := Eps * MIN
   Si A - MIN ≤ E  Alors INC(I,J) ; INC(K,L)
   Si B - MIN ≤ E  Alors INC(I,K) ; INC(J,L)
   Si C - MIN ≤ E  Alors INC(I,L) ; INC(J,K)
   Si ces 3 conditions sont remplies Alors
      INC(J,I) ; INC(L,K) ; INC(K,I)
      INC(L,J) ; INC(L,I) ; INC(K,J)
```

3/ Choix d'un seuil

Soit Sm le score large maximum atteint et Smt le score maximum théorique: Smt=(N-2)(N-3)/2. La valeur possible immédiatement inférieure est abaissée de N-3, si bien que le seuil S de regroupement est fixé à Smt-(N-3)/2, si cette valeur est atteinte. Sinon S=Sm. Dans ce dernier cas on dit que le seuil est affaibli.

4/ Construction du pré-groupement

Considérons le graphe seuil pour cette valeur S tel que chacune de ses arêtes a un score large supérieur ou égal à S. Un pré-groupement est une partie connexe de ce graphe; on l'obtient par application de la procédure de construction des parties connexes du § 1.1.

5/ Construction des groupements

Si le seuil de pré-groupement est affaibli, tout pré-groupement est un groupement; aller en 6/
Sinon à chaque pré-groupement correspond un seuil théorique St qui est fonction du nombre de sommets Ng du pré-groupement Cf. [2].

$$St = (N-Ng)*(N-Ng-1)/2 - (N-3)/2 - (1-Eps)*(N-Ng-1)$$
$$= ((N-Ng-2+2*Eps)*(N-Ng-1)-(N-3))/2$$

Un pré-groupement est un groupement si chacune de ses arêtes a un score strict supérieur ou égal à St, soit $C(i,j) - C(j,i) \geq St$. Si pour une arête cette condition n'est pas vérifiée, le pré-groupement est abandonné. Si aucun pré-groupement ne passe ce test, alors le seuil S est affaibli (S=Sm) et l'on revient en 4/

Chaque groupement G entraîne la création d'un nouveau sommet Ns (numéroté à la suite) et chaque sommet I du groupement est lié à Ns dans l'arbre que l'on construit.

6/ Calcul des longueurs des arêtes. Deux cas sont à distinguer:

6.1/ Tous les sommets ne sont pas dans un même groupement.
Soit i et j deux sommets du groupement G et k un sommet pendant quelconque. Notons $L_i=D(i,j)+D(i,k)-D(j,k)$. Alors la longueur de l'arête i--s est la moyenne des L_i lorsque j parcourt le groupement et k l'ensemble des sommets pendants hors groupement.

Comme les sommets du groupement sont effaçés à l'itération suivante, on va munir le sommet s d'une longueur égale à la moyenne des longueurs des arêtes i--s du groupement qui sera à déduire de la longueur de l'arête ultérieurement obtenue à partir de s. La formule de calcul des longueurs devient:

$$L(i) = \left(\sum_{j\in G,\, i\neq j,\, k\notin G} 1/2 \; - (D(i,j)+D(i,k)-D(j,k)) - L(i) \right) / ((Ng-1)*(N-Ng))$$

et $L(s) = \left(\sum_{i\in G} L(i) \right) / Ng$

6.2 Tous les sommets sont dans le dernier groupement.
Dans ce cas, les longueurs ne peuvent être calculées suivant la formule ci-dessus. On calculera la longueur L(i) de l'arête i--s par:

$$L(i) = \left(\sum_{j,k\in G} (D(i,j)+D(i,k)-D(j,k)) - L(i) \right) / ((Ng-1)*(Ng-2))$$

Il suffit donc d'énumérer les triplets de sommets du groupement.

7/ Mise à jour et tassement du tableau des dissimilarités

Les dissimilarités d'un sommet de groupement aux autres sommets sont calculées comme les moyennes des distances de ces sommets aux éléments du groupement.

Pour tout $k\notin G$, $D(s,k) = \left(\sum_{i\in G} D(i,k) \right) / Ng$

Les sommets qui figurent dans un groupement sont supprimés et ceux des groupements ajoutés. On retourne alors à l'étape 1/ avec une valeur de N plus petite N := N - Ng + 1.

8/ Il reste 2 sommets i,j

On prend le plus grand, mettons j, qui devient la racine de l'arbre, auquel on ajoute l'arête i--j, donc T(i) := j. La longueur de cette arête est calculée par:
L(i) = D(1,2) - LG(i) - LG(j)

9/ Il reste 3 sommets i,j,k

On ajoute un nouveau sommet Ns, racine de l'arbre, donc: T(i) = T(j) = T(k) = Ns.
On calcule les longueurs des arêtes suivant la formule donnée dans 6.2/.

1.5 Méthode de réduction des quadruplets

Nous avons déjà exposé précédemment la "condition des quatre points" que vérifie tout quadruplet d'une distance additive d'arbre, à savoir que des trois sommes possibles, les deux plus grandes sont égales.

$$D(i,j) + D(k,l) \leq D(i,k) + D(j,l) = D(i,l) + D(j,k).$$

L'idée de cet algorithme est de modifier les dissimilarités de façon que tout quadruplet vérifie cette condition. Cet algorithme ne construit donc pas un arbre à distances additives, mais une distance additive d'arbre à laquelle il suffit d'appliquer l'un des algorithmes précédents.

A partir de D, on va donc construire une suite de dissimilarités D^1, D^2,... dont la limite D* soit une distance additive d'arbre. Supposons que l'on ait:

$$D^k(i,j) + D^k(k,l) < D^k(i,k) + D^k(j,l) < D^k(i,l) + D^k(j,k)$$

alors, parmi les six distances ci-dessus, on ne modifie que les quatre dernières pour "forcer" l'égalité des deux dernières sommes.

Posons $B=D^k(i,k)+D^k(j,l)$ et $C=D^k(i,l)+D^k(j,k)$. Pour satisfaire à la condition des quatre points, il faudrait répartir la différence C-B entre ces quatre distances, par exemple ajouter à $D^k(i,k)$ et $D^k(j,l)$ la quantité (C-B)/4 et la retrancher à $D^k(i,l)$ et $D^k(j,k)$. Mais si l'on effectue ces modifications des valeurs de dissimilarités à chaque examen d'un quadruplet, il n'y a aucune raison de conserver les valeurs acquises et l'algorithme serait fonction de l'ordre de numérotation des sommets. On modifie donc les dissimilarités d'une valeur moyenne calculée après examen de tous les quadruplets.

On stocke pour chaque dissimilarité (dans la partie supérieure droite du tableau) la somme des modifications que chaque quadruplet apporte. Ceci fait, on effectue pour chaque dissimilarité $D^k(i,j)$, la moyenne des modifications, en divisant leur somme par le nombre de quadruplets contenant i et j, soit (N-2)*(N-3)/2. Si la somme des valeurs absolues des modifications apportées est supérieure à une valeur Eps pré-fixée (égale à 1% de la moyenne des dissimilarités), on effectue une nouvelle itération. Sinon la somme, sur tous les quadruplets, des différences entre les deux plus fortes sommes est quasi nulle et cette condition est suffisante pour que l'on ait une distance additive d'arbre.

Algorithme de réduction

```
    Pour tout (I,J)  Faire
       S := S + D(I,J) ; D(J,I)=0
    Eps := ( S / ( (N-1) * (N-2) / 2 ) ) / 100

0/ Pour tout I<J<K<L  Faire
     A := D(I,J) + D(K,L) ; B := D(I,K) + D(J,L) ; C := D(I,L) + D(J,K)
     Min := min(A,B,C)
     Si A=Min  Aller en 1/
     Si B=Min  Aller en 2/
     Si C=Min  Aller en 3/
     1/ DIF := (C-B) / 4
        D(L,I) := D(L,I) - DIF ; D(K,J) := D(K,J) - DIF
        D(K,I) := D(K,I) + DIF ; D(L,J) := D(L,J) + DIF
        Aller en 4/
     2/ DIF := (C-A) / 4
        D(L,I) := D(L,I) - DIF ; D(K,J) := D(K,J) - DIF
        D(J,I) := D(J,I) + DIF ; D(L,K) := D(L,K) + DIF
        Aller en 4/
     3/ DIF := (B-A) / 4
        D(K,I) := D(K,I) - DIF ; D(L,J) := D(L,J) - DIF
        D(J,I) := D(J,I) + DIF ; D(L,K) := D(L,K) + DIF
     4/ Fin

   Pout tout (I<J)  Faire
     MOD := D(J,I) / ( (N-2) * (N-3) / 2 )
     D(I,J) := D(I,J) + MOD ; D(J,I) := 0
     SMOD := SMOD + Abs(MOD)
   Si SMOD > EPS  Aller en 0/
```

M. Roux [9] a démontré la convergence de son algorithme en trois étapes ; Il établit d'abord la décroissance de la quantité $\Sigma_{i<j} (D^k(i,j))^2$ puis montre que

$$\Sigma_{i<j} (D^{k+1}(i,j))^2 + (D^K(i,j))^2 - 2 . D^{k+1}(i,j) . D^k(i,j) \quad \text{--> } 0$$

Enfin il montre que la suite D^k est une suite de Cauchy; elle converge vers D^*.

2 ETUDE COMPARATIVE

Il nous a paru nécessaire, après l'exposé de ces différents algorithmes, d'effectuer une première étude comparative de leurs résultats. Nous définissons d'abord plusieurs situations expérimentales qui conduisent à 6 tableaux de dissimilarités. Cette étude n'est pas complète, dans la mesure où il faudrait comparer ces algorithmes sur des moyennes de résultats. Puis nous définissons 2 critères non métriques fondés sur les modifications apportées à la structure de chaque sous-arbre à 4 sommets pendants, par rapport à la bi-partition induite par les inégalités quadrangulaires. Ensuite nous définissons 3 critères métriques basés sur les écarts entre dissimilarités et distances additives induites par l'arbre valué puis l'approximation quadratique des longueurs d'arêtes de l'arbre. Cette méthode brièvement rappelée ci-dessous a été développée dans [7], ainsi que des algorithmes de tracé des arbres valués.

2.1 Approximation des longueurs des arêtes

Par définition des arbres, il n'y a qu'un seul chemin qui lie deux sommets, et sa longueur est la somme des longueurs des arêtes empruntées. En écrivant que pour chaque couple de sommets pendants, leur dissimilarité est égale à la longueur du chemin qui les lie, on obtient un système linéaire de N(N-1)/2 équations à Ns-1 inconnues ; A . L = D.

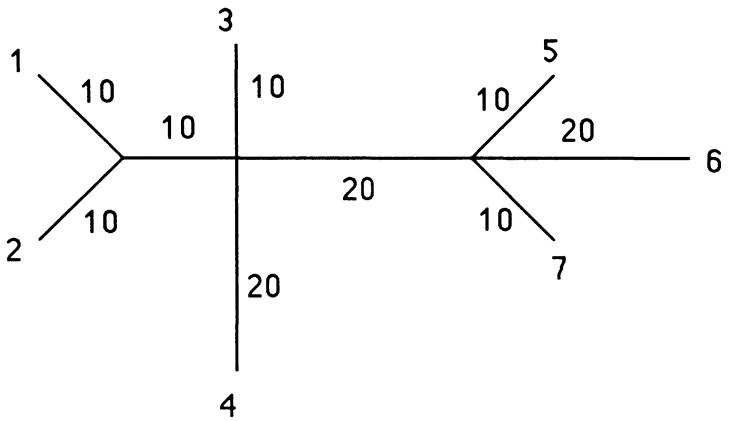
Pour N>4 ce système a plus d'équations que d'inconnues; on le résoud par approximation, au sens des moindres carrés, entre les longueurs des chemins obtenus et les dissimilarités. Ce problème classique d'optimisation a pour solution la solution de A^t. A . L = A^t. D (A^t désigne la matrice transposée de A et A^t. A est une matrice symétrique définie positive). Comme tous les sommets sont au moins de degré 3, elle est régulière.

Mais la solution de ce système n'est pas nécessairement à valeurs positives ou nulles, ce que la représentation des dissimilarités par un arbre impose. On résoudra donc le système A^t.A.L = A^t.D par la méthode de Gauss-Siedel légèrement modifiée. Le vecteur des longueurs des arêtes obtenu à chaque itération est projeté dans le cône positif L>0, c'est à dire que les longueurs négatives sont mises à 0.

2.2 Expérimentations numériques

2.2.1 Choix des données

Nous sommes d'abord partis d'une structure d'arbre valué à 10 sommets dont 7 pendants, donc un tableau de dissimilarités de dimension 7 codant une distance additive d'arbre, à partir de laquelle chacun des algorithmes redonne l'arbre initial.



Nous avons ensuite construit deux expériences en modifiant ce tableau, d'une part en ajoutant ou retranchant 10% de leur valeur à la moitié des dissimilarités, d'autre part en ajoutant ou retranchant 90% de leur valeur à quelques dissimilarités. Ceci correspond à l'addition d'un peu de flou à une bonne partie des valeurs, pour le premier cas, et à quelques grosses erreurs dans l'évaluation des dissimilarités dans le second.

```
     1   2   3   4   5   6
2 : 18
3 : 33  30
4 : 44  40  27
5 : 50  55  40  45
6 : 66  60  45  60  30
7 : 50  45  40  55  20  33
```

Essai 1

```
     1   2   3   4   5   6
2 : 20
3 :  2  30
4 : 40  40  30
5 : 50  90  40   5
6 : 60  60  50  60  60
7 : 50  50  40  50  20  30
```

Essai 2

Ensuite, ayant remarqué que dans l'arbre précédent, les sommets pendants s'éloignent notablement des sommets intérieurs, nous sommes partis d'un arbre en forme de Y, avec trois sommets aux extrémités des branches de l'Y, trois aux milieux et un au centre. Puis nous avons appliqué à la distance additive de l'arbre Y des modifications du même type que précédemment, pour obtenir les essais 3 et 4.

```
     1   2   3   4   5   6
2 : 10
3 : 22   9
4 : 30  22  10
5 : 36  30  20  11
6 : 27  20  10  22  30
7 : 40  27  20  30  44  10
```

Essai 3

```
     1   2   3   4   5   6
2 : 10
3 : 20  10
4 : 57  20  10
5 : 40  30  20  10
6 : 30  38   1  20  30
7 : 40  30  20  30   4  10
```

Essai 4

Pour le cinquième essai, nous avons considéré deux petits tableaux de dissimilarités à valeurs faibles (<10), l'un de dimension 3, l'autre de dimension 4, réunis dans un tableau de dimension 7, en complétant par des valeurs aléatoires plus fortes (>10). Ceci correspond à deux groupements, {1,2,3} contre {4,5,6,7} nettement séparés.

Enfin pour le sixième essai, nous avons considéré un tableau de dimension 7 à valeurs aléatoires, comprises entre 40 et 50, d'où ne devrait se dégager aucune structure classificatoire ou plutôt un arbre en étoile, puisque toutes les dissimilarités sont voisines.

```
     1   2   3   4   5   6
2 :  5
3 :  2   7
4 : 13  25  12
5 : 17  22  32   8
6 : 28  14  15   4   2
7 : 40  29  35   5   9   3
```

Essai 5

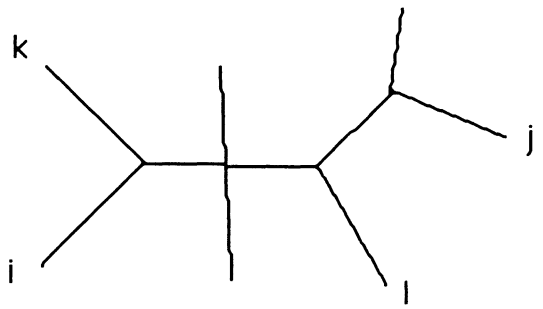
```
     1   2   3   4   5   6
2 : 40
3 : 50  42
4 : 46  40  45
5 : 48  48  47  42
6 : 44  43  45  50  49
7 : 50  42  43  44  44  41
```

Essai 6

2.2.2 Choix des paramètres

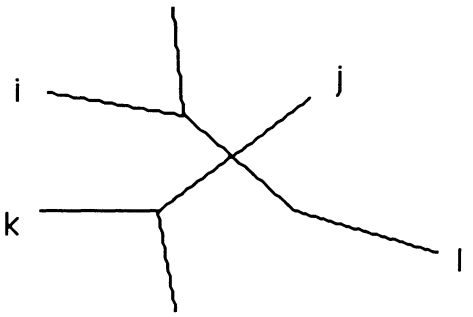
Nous avons défini deux critères non métriques invariants par toute transformation monotone des valeurs de dissimilarité et basés sur la structure d'arbre des quadruplets . En effet, à chacun correspond soit une étoile à quatre sommets pendants, soit une bi-partition des 4 sommets. Dans le second cas, on comptera d'une part les quadruplets dont la bi-partition est différente dans l'arbre à distances additives, sous l'appelation "nombre d'inversions" (NI), et d'autre part le nombre de quadruplets représentés par une croix (NC); l'algorithme a égalisé les sommes de dissimilarités.

On avait $D(i,j)+D(k,l) < D(i,k)+D(j,l)$ et
$D(i,j)+D(k,l) < D(i,l)+D(j,k)$ et l'on obtient:



La paire {i,j} n'est pas opposé à {k,l}. Il y a inversion et NI := NI+1.

On avait $D(i,j)+D(k,l) < D(i,k)+D(j,l)$ et
$D(i,j)+D(k,l) < D(i,l)+D(j,k)$ et l'on obtient:



Les trois sommes sont devenues égales, alors NC := NC+1

Pour comparer des arbres valués ayant mêmes sommets pendants, certains critères métriques semblent naturels. Les valuations permettent de reconstituer des distances "dans l'arbre" entre tout couple de sommets pendants. On a donc pour chaque couple de sommets une valeur de dissimilarité et une distance dans l'arbre. On peut donc calculer:

EM : l'écart absolu moyen, plus pertinent que l'écart moyen,
MA : le maximum des écarts absolus et
EQ : l'écart quadratique moyen, c'est à dire la moyenne des carrés des écarts.

2.2.3 Résultats

Nous avons appliqué sur les six tableaux précédemment définis les méthodes de décomposition (MD), des plus proches prédécesseurs (MP), de dispersion (MI), des scores (MS), et de réduction (MR), cette dernière étant complétée par la méthode de décomposition pour la construction effective de l'arbre valué. Ces cinq méthodes déterminent des longueurs d'arêtes avec lesquelles nous calculons les critères métriques, puis nous approximons les longueurs pour recalculer ces nouveaux critères (leurs noms sont précédés de la lettre A). Cette nouvelle valuation ne modifie que les critères métriques de comparaison et elle ne peut qu'améliorer l'écart quadratique moyen. Les six essais donnent respectivement les tableaux de résultats suivants.

	NI	NC	EM	MA	EQ	AEM	AMA	AEQ
MD	6	0	2.26	7.5	11.11	1.9	5.7	6.98
MP	5	0	2.61	8.36	14.28	1.91	5.67	6.96
MI	5	0	1.85	6.79	7.27	1.9	5.65	6.96
MS	0	10	2.16	6.55	7.93	2.09	5.55	7.23
MR	6	0	1.92	5.49	7.06	1.89	5.68	6.98
MD	8	0	7.67	27.5	129.21	7.11	26.29	103.7
MP	6	0	7.8	35	133.07	7.37	28.11	105.71
MI	5	10	17.75	44.58	480.06	8.98	31.78	151.53
MS	6	0	7.71	26	109.21	7.38	28.07	105.69
MR	6	0	7.4	30.75	109.12	7.36	28.11	105.7
MD	4	0	1.22	6.08	3.86	1.12	3.21	2.15
MP	6	0	1.32	4	3.24	1.19	3.31	2.35
MI	4	0	1.11	3.85	2.23	1.12	3.23	2.14
MS	4	0	1.1	3.42	2.16	1.12	3.22	2.14
MR	4	0	1.06	3.84	2.35	1.12	3.23	2.14
MD	8	5	6.93	22.21	82.93	5.97	15.77	57.27
MP	11	0	6.92	18.99	74.55	5.98	15.79	57.27
MI	6	9	12.14	31.48	223.99	6.59	15.5	64.39
MS	2	12	6.53	18.49	69.23	6.83	17.27	67.89
MR	8	5	5.89	20.16	64.21	5.98	15.8	57.27
MD	6	4	5.98	16	54.67	4.21	9.19	27.43
MP	10	0	4.97	17.8	49.31	4.44	9.54	28.2
MI	4	8	6.02	14.78	62.32	4.87	9.88	30.58
MS	6	4	4.31	10.07	29.21	4.32	9.21	27.43
MR	9	0	4.2	13.5	30.56	4.22	9.17	27.43
MD	1	24	2.3	8	10.15	1.82	5.44	5.53
MP	12	0	1.64	4.57	4.47	1.57	3.97	3.97
MI	4	24	3.94	7.95	21.31	1.94	5.22	6.4
MS	3	16	1.65	4.68	4.69	1.62	4.65	4.51
MR	10	0	1.51	5.41	4.25	1.37	4.17	3.81

2.3 Conclusions

Si l'on admet que du point de vue non métrique, le nombre d'inversions (NI) est le critère le plus important (il doit être le plus faible possible) on voit que la méthode des scores donne souvent les meilleurs résultats, ce qui n'est pas surprenant compte tenu de l'algorithme. Par contre, c'est souvent au détriment du nombre d'égalités, ces deux critères étant très corrélés négativement.

Si l'on regarde les paramètres métriques des arbres valués, on peut résumer les résultats de la façon suivante:

Si l'on a un tableau de dissimilarités "voisin" d'une distance additive d'arbre (essais 1 et 3), la méthode des scores, celle de dispersion et celle de réduction donnent des résultats meilleurs que les autres.

Si l'on s'en éloigne à cause de quelques valeurs aberrantes (essais 2 et 4), alors les méthodes des scores et de réduction donnent les meilleurs résultats. Dans l'essai 4, du fait que l'un des sommets est au centre de l'arbre, on pouvait s'attendre à un meilleur résultat de la méthode de décomposition. Cela n'arrivera qu'avec l'approximation des longueurs des arêtes. La méthode de dispersion semble dans ce cas très mauvaise. Ceci s'explique par le fait qu'elle donne des longueurs d'arêtes négatives, qui sont mises à zéro pour le calcul des paramètres, mais même si on les ajuste aux dissimilarités, ses résultats restent médiocres; c'est la structure d'arbre qui est inadéquate.

Si l'on a une forte structure classificatoire (essai 5), la méthode de réduction et celle des scores sont les meilleures, et si l'on n'a aucune structure (essai 6), toutes les méthodes en conviennent et dessinent une étoile, mais c'est encore les méthodes des scores, des plus proches prédécesseurs et celle de réduction qui déterminent le mieux les longueurs des arêtes.

Maintenant, si l'on regarde les paramètres métriques des arbres dont les longueurs d'arêtes sont ajustées aux dissimilarités, on voit que les meilleurs résultats sont ceux des méthodes de décomposition, ou des plus proches prédécesseurs et toujours de réduction. On peut penser que, quitte à minimiser l'écart quadratique, autant partir d'un arbre déterminé rapidement par la méthode de décomposition. Si cette procédure paraît trop longue, il vaut mieux utiliser la méthode des scores. Si la longueur des calculs de la méthode de réduction ne décourage pas l'utilisateur, c'est elle qui donne quasi systématiquement les meilleurs résultats.

Je terminerai cette étude comparative en utilisant une méthode de comparaison par paires. Nous dirons qu'une méthode A est meilleure qu'une méthode B, pour un critère C si la valeur de C obtenue par A est inférieure à celle obtenue par B. En prenant comme critère les 6 paramètres métriques pour les 6 essais, on obtient 36 préordres qui ordonnent les 5 méthodes. Ceci nous permet de construire un tournoi. L'ordre à distance minimum de ce tournoi donne MR avant MS avant MD avant MP avant MI.

3 Bibliographie

[1] J.P. Barthélemy, X. Luong.- "Mathématique, algorithmique et histoire des représentations arborées", *Mathématiques et Sciences Humaines* , à paraître.

[2] J.P. Barthélemy, A. Guénoche.- *Les arbres et les représentations des proximités* , Masson, 1987.

[3] G. Brossier.- Approximation des dissimilarités par des arbres additifs, *Mathématiques et Sciences Humaines* , 91, 1985, p. 5-22.

[4] J.D. Carroll, S. Pruzansky.- Fitting of hierarchical tree structure (HTS) models, mixtures of HTS models and hybrid models, via mathematical programming and alternating least squares, *U.S.-Japan Seminar on Multidimensional Scaling* , University of San Diego, 1975.

[5] J.P. Cunningham.- Free Trees and Bidirectional Trees as representations of Psychological Distance, *Journal of Mathematical Psychology* , 17, 1978, p.165-188.

[6] G. De Soëte.- A least squares algorithm for fitting additive trees to proximity data, *Psychometrica* , 48, 4, 1983, p. 621-626.

[7] A. Guénoche.- Représentations arborées des classifications, *R.A.I.R.O., Série Recherche Opérationnelle* , 20, 4, 1986, p. 1-13.

[8] M. Roux.- Représentation d'une distance par un arbre aux arêtes additives, Actes des Quatrièmes journées internationales Analyse des Données et Informatique, Versailles, 9-11 Octobre 1985, p. 3-15.

[9] M. Roux.- Construction d'arborescences par modifications séquentielles des distances, Actes du Colloque COMPSTAT, Rome, 1986.

[10] S. Sattah, A. Tversky.- Additive similarity trees, *Psychometrika* , 42, 3, 1977.