ANALYSE D'UN NUAGE DE POINTS COMME UNE IMAGE MULTIDIMENSIONNELLE

J. Quinqueton

I.N.R.I.A.
Rocquencourt, France

ABSTRACT

Let E be a set of N points in $[0,1[^n \subset R^n$. We first divide $[0,1[^n$ into 2^{nk} half open hypercubes called in the following way:

$$U(n,k) = \{u_k = \prod_{j=1}^{n} \left[\frac{m_j}{2^k} , \frac{m_{j+1}}{2^k} \right[\; ; \text{ where } \forall j;\ 0 \leq m_j < 2^k\} \quad (1)$$

We define, for each k, E_k as the set:

$$E_k = \{u_k \in U(n,k) \ ; \ u_k \cap E \neq \emptyset\} \quad (2)$$

Each $u_k \in E_k$ is mapped onto $x(u_k) \in N^n$, with a weight $p(u_k) \in R$:

$$\left.\begin{array}{l} u_k = \prod_{j=1}^{n} \left[\frac{m_j}{2^k} , \frac{m_{j+1}}{2^k} \right[\\ \Rightarrow \underline{x}(u_k) = (m_1,\ldots,m_n) \in N^n \\ u_k \in E_k \Leftrightarrow u_k \cap E \neq \emptyset \\ \Rightarrow p(u_k) = \text{Card}\,(u_k \cap E) \end{array}\right\} \quad (3)$$

Then, we can define a near-neighbor graph $G_k = (E_k, U_k)$:

$$U_k = \{(x(u_k), \underline{x}(u_k')) ; \sum_{i=1}^{n} |m_i - m_i'| = 1\} \quad (4)$$

where m_i and m_i' are the i^{th} components of $x(u_k)$ and $x(u_k')$.

To build an intrinsic representation of G_k is to give each edge a direction (number between 1 and p) by defining a representation function D from U_k into $\{\pm j \ ; \ 1 < j \leq p\}$, such that:

$$D(x,y) = D(x,z) \Rightarrow y = z \quad (5)$$

$$D(x,y) + D(y,x) = 0 \quad (6)$$

$$\left.\begin{array}{l} (x,y) \in G_k \\ (x,z) \in G_k \\ (z,t) \in G_k \\ (y,t) \in G_k \end{array}\right\} \Rightarrow \begin{array}{l} D(x,y) = D(z,t) \\ D(x,z) = D(y,t) \end{array} \quad (7)$$

An application of this technique on petroleum data is presented.

RÉSUMÉ

Soit E un ensemble de N points dans $[0,1[^n \subset R^n$. Nous définissons tout d'abord une partition de $[0,1[^n$ en 2^{nk} hypercubes semi ouverts de la manière suivante:

$$U(n,k) = \{u_k = \prod_{j=1}^{n} \left[\frac{m_j}{2^k} , \frac{m_{j+1}}{2^k} \right[\; ; \text{ où } \forall j;\ 0 \leq m_j < 2^k\} \quad (1)$$

Soit l'ensemble E_k défini de la manière suivante:

$$E_k = \{u_k \in U(n,k) \ ; \ u_k \cap E \neq \emptyset\} \quad (2)$$

A chaque $u_k \in E_k$ est associé un élement $\underline{x}(u_k) \in N^n$ et un poids $p(u_k) \in R$ par:

$$\left.\begin{array}{l} u_k = \prod_{j=1}^{n} \left[\frac{m_j}{2^k} , \frac{m_{j+1}}{2^k} \right[\\ \Rightarrow \underline{x}(u_k) = (m_1,\ldots,m_n) \in N^n \\ u_k \in E_k \Leftrightarrow u_k \cap E \neq \emptyset \\ \Rightarrow p(u_k) = \text{Card}\,(u_k \cap E) \end{array}\right\} \quad (3)$$

On peut donc définir sur E_k un graphe de voisinage, $G_k = (E_k, U_k)$:

$$U_k = \{(x(u_k), \underline{x}(u_k')) ; \sum_{i=1}^{n} |m_i - m_i'| = 1\} \quad (4)$$

où m_i et m_i' sont les $i^{èmes}$ composantes de $x(u_k)$ et $x(u_k')$.

Construire une représentation intrinsèque du graphe G_k c'est assigner à chaque arête une direction (nombre entre 1 et p) et un signe. On définit pour cela une fonction de représentation D from U_k dans $\{\pm j \ ; \ 1 < j \leq p\}$ vérifiant:

$$D(x,y) = D(x,z) \Rightarrow y = z \quad (5)$$

$$D(x,y) + D(y,x) = 0 \quad (6)$$

$$\left.\begin{array}{l} (x,y) \in G_k \\ (x,z) \in G_k \\ (z,t) \in G_k \\ (y,t) \in G_k \end{array}\right\} \Rightarrow \begin{array}{l} D(x,y) = D(z,t) \\ D(x,z) = D(y,t) \end{array} \quad (7)$$

1. INTRODUCTION

La Reconnaissance des Formes peut être définie comme l'interprétation d'une représentation, c'est à dire un traitement d'une représentation initiale du problème, permettant de décrire celui ci dans un espace simple, que l'on appelle *espace d'interprétation* [1].

En Analyse de Données, la représentation initiale d'un problème est un nuage de points de R^n. L'espace d'interprétation est, soit un ensemble fini (classification automatique), soit un espace vectoriel de dimension réduite (en général 2 ou 3).

C'est à cette seconde catégorie de méthodes que nous nous intéressons ici. Il existe une abondante littérature dans ce domaine, et une étude bibliographique assez complète a été faite dans un article récent [2].

Nous avons proposé, dans un travail récent [3], des méthodes basées sur une digitalisation du nuage de points, c'est à dire qui considèrent celui-ci comme une *image multidimensionnelle*.

L'idée consiste alors à chercher à analyser la *forme* du nuage ainsi représenté. Des résultats théoriques sur ces problèmes ont été présentés, principalement sur leur complexité [4].

Nous proposons ici un algorithme simple et rapide pour représenter dans un espace de faible dimension un nuage de points digitalisé.

2. POSITION DU PROBLEME

Soit E un ensemble de N points dans l'hypercube unitaire $[0,1[^n$ de R^n. Nous définissons tout d'abord une partition de $[0,1[^n$ en 2^{nk} hypercubes semi ouverts, de la manière suivante:

$$\mathcal{U}_{(n,k)} = \left\{ u_k = \prod_{j=1}^{n} \left[\frac{m_j}{2^k}, \frac{m_j+1}{2^k} \right[\; ; \right.$$

où, pour tout j, $0 \leq m_j < 2^k$ $\Big\}$ (1).

Soit l'ensemble E_k défini de la manière suivante:

$$E_k = \left\{ u_k \in \mathcal{U}_{(n,k)} \; ; \; u_k \cap E \neq \emptyset \right\} \quad (2).$$

A chaque $u_k \in E_k$ est associé un élément $x(u_k) \in N^n$ et un poids $p(u_k) \in R$, par:

$$u_k = \prod_{j=1}^{n} \left[\frac{m_j}{2^k}, \frac{m_j+1}{2^k} \right[$$

$$\Rightarrow \underline{x}(u_k) = (m_1, \ldots, m_n) \in N^n \qquad (3)$$

$$u_k \in E_k \iff u_k \cap E \neq \emptyset$$

$$= p(u_k) = \text{Card}(u_k \cap E)$$

On peut donc définir sur E_k *un graphe de voisinage*, $G_k = (E_k, U_k)$

$$U_k = \{(x(u_k), \underline{x}(u_k')) \; ; \; \sum_{i=1}^{n} |m_i - m_i'| = 1\} \qquad (4)$$

où m_i et m_i' sont les $i^{èmes}$ composantes de $x(u_k)$ et $x(u_k')$.

Pour chaque $E_k \subset \mathbb{N}^n$, il existe des ensembles $Y \subset \mathbb{N}^p$, avec $p \leq n$, tels que le graphe défini sur y par la relation (4) soit isomorphe à G_k.

Nous nous posons alors le problème de trouver la valeur minimale de p pour laquelle un tel Y existe.

3. LA RELATION DE PARALLELISME

Nous proposons une solution constructive du problème précédent. Soit G = (X,U) un graphe repondant à la définition (4).

Nous commencons par définir la notion *d'arcs parallèles*.

Soient (x,y) et (x',y') deux arcs du graphe. Ils sont dits parallèles s'il existe deux chemins élémentaires disjoints :

$$\left.\begin{array}{l}(x \ldots \; x_i \; \ldots \; x') \\ (y \ldots \; y_i \; \ldots \; y')\end{array}\right\} \text{ ayant même nombre d'arcs}$$

et dont les sommets sont 2 à 2 voisins :

$$(x_i, y_i) \in U$$

Cette relation est réflexive, symétrique et transitive : c'est une relation d'équivalence. Nous appellerons l'ensemble quotient Q(G) : c'est une partition de .

Elle possède une propriété importante, qui est qu'un arc (x,y) n'est jamais parallèle à

son inverse (y,x). On peut même montrer qu'à toute classe d'équivalence α correspond une classe α^{-1} formée des arcs inverses de ceux de α.

Soit un sommet z du graphe G. Etiquetons chaque sommet x du graphe par le nombre d'arcs $\ell(z,x)$ du plus court chemin entre z et x.

Soit U_z l'ensemble des arcs tels que l'extrémité ait une étiquette supérieure à celle de l'origine. On pose $G_z = (X,U_z)$. Si une classe d'équivalence α est composée d'arcs de G_z alors α^{-1} ne l'est pas, par définition.

La figure 1 représente un tel graphe G_z, et les classes d'équivalence associées.

Nous supposons que G_z ne peut contenir à la fois des arcs appartenant à α et des arcs appartenant à α^{-1}.

Un tel graphe sera dit *orientable*. Soit $Q(G_z)$ l'ensemble des classes d'équivalence des arcs de G_z.

Deux classes d'équivalence α et β seront dites *orthogonales* s'il existe un sommet w ayant dans G_z deux successeurs x et y tels que

$$\left\{ \begin{array}{l} (w,x) \in \alpha \\ (w,y) \in \beta \end{array} \right\}$$

Ces relations peuvent être représentées sur un tableau, comme le montre la figure 2.

Ces définitions nous permettent de poser de façon constructive le problème de la recherche d'une représentation intrinsèque.

Trouver une représentation intrinsèque du graphe G c'est trouver une partition de l'ensemble $Q(G_z)$ telle que, si α et β sont orthogonales alors elle n'appartiennent pas à la même classe de la partition.

4. REPRESENTATION INTRINSEQUE

Le problème posé à la fin du paragraphe précédent est équivalent à un problème de coloration de graphe, c'est-à-dire un problème NP-complet.

Nous proposons donc une formulation permettant une solution plus simple.

Si nous fixons le nombre de classes p, alors le problème peut se résoudre par un algorithme d'optimisation très simple.

Soit une partition de $Q(G_z)$ en p classes :

$$\mathcal{P} = \{C_1, \ldots, C_p\}$$

A une telle partition, on peut toujours associer une représentation (avec recouvrements) comme le montre la fig. 3. Soit $\alpha \in C_i$. On appelle $w_j(\alpha)$ le nombre d'éléments de C_j orthogonaux à α.

On va alors définir une nouvelle partition en ajoutant α à la classe C_j telle que $w_j(\alpha)$ soit le plus petit possible, et en l'effaçant de C_i.

Si i=j, alors on considère un autre élément α.

Cet algorithme itératif fait décroitre le critère :

$$W = \sum_{i=1}^{p} \sum_{\alpha \in C_i} w_i(\alpha)$$

C'est bien ce que nous cherchons, puisque W est le nombre d'éléments d'une même classe C_i qui sont orthogonales entre elles.

Cet algorithme est assez simple à implanter, et nous montrons son fonctionnement sur un exemple simple, sur la figure 4.

On peut constater sur la figure 4 que le résultat obtenu est satisfaisant, c'est-à-dire qu'on obtient une représentation avec peu de recouvrements.

Nous allons maintenant revenir plus en détails sur les limites de la méthode.

5. LIMITES DE LA METHODE

Nous avons vu dans le paragraphe 3 que l'algorithme précédent ne pouvait fonctionner que si le graphe G était *orientable*.

Mais le fait que l'algorithme précédent puisse fonctionner ne garantit pas que le résultat soit admissible, c'est-à-dire qu'une représentation intrinsèque telle que W = 0 soit effectivement sans recouvrements.

On peut montrer que, pour que ce résultat soit "admissible", il faut que le graphe G ait quelques propriétés supplémentaires [3].

Il doit être *euclidien*, c'est-à-dire que deux chemins C et C_2 différents et de longueur minimale

entre x et y doivent être tels que tout arc de C_1 soit parallèle à un arc de C_2 et réciproquement.

Il doit de plus être *elliptique*, c'est à dire qu'il doit exister deux sommets a et b tels que :

Pour tout sommet x, $\ell(x,a)+\ell(x,b) = \ell(a,b)$ où $\ell(x,y)$ est le nombre d'arcs dans le plus court chemin de x à y.

Ces contraintes restreignent la généralité de notre algorithme. On montre cependant facilement que tout graphe est une réunion de sous graphes elliptiques. Il suffit alors de construire la représentation intrinsèque de chacun de ces sous graphes.

6. APPLICATIONS

La représentation intrinsèque d'un graphe construit sur un nuage de points permet d'analyser la forme de ce nuage.

En effet, construire une représentation intrinsèque, c'est assigner à chaque arc du graphe une direction.

Le nuage initial dans $\mathbb{N}^n$ étant une représentation, à chaque arc du graphe on pourra associer sa direction initiale et sa direction dans la représentation intrinsèque. Les axes de coordonnées de la représentation intrinsèque pourront ainsi être interprétés dans l'espace de départ.

Nous avons donc bien un outil permettant d'analyser la forme d'un nuage de points.

7. CONCLUSION

La méthode de représentation proposé dans cet article, tout comme les méthodes décrites dans un précédent travail [3], [5], [6], ont pour but une représentation visuelle des données.

Elles sont donc tout naturellement destinés à faire partie d'uns système conversationnel d'analyse de données. En effet, les méthodes de classification automatique par optimisation, de plus en plus utilisées consistent à améliorer une partition initiale au sens d'un certain critère. L'expérience montre que, sur des données réelles, le choix de cette partition initiale est important. Une représentation visuelle des données telle que celle que nous proposons peut aider à faire ce choix.

Une autre application est l'aide à l'interprétation d'une classification, obtenue par une méthode quelconque. La représentation visuelle du nuage de points correspondant à chaque classe facilite l'interprétation de la classification.

8. REFERENCES

[1] J.C. SIMON, E. BACKER, J. SALLANTIN - "A structural approach of Pattern Recognition", Signal Processing, Vol 2 pp. 5-22, 1980.

[2] K.W. PETTIS, T.A. BAILEY, A.K. JAIN, R.C. DUBES - "An Intrinsic Dimensionality estimator from Near Neighbor Information" IEEE trans on Patt. Anal. and Machine Intell., Vol PAMI 1, janvier 1979.

[3] J. QUINQUETON - "Le Concept de Dimension Intrinsèque en Reconnaissance des Formes" Thèse d'Etat, Université Paris VI, France Février 1981.

[4] J. MYLOPOULOS - "On the Application of Formal Language and Automata Theory to Pattern Recognition", Pattern Recognition, Vol 4, pp. 37-51, 1972.

[5] J. QUINQUETON, M. BERTHOD - "A locally Adaptive Peano Scanning Algorithm" IEEE Trans on Patt Anal and Machine Intell, Vol PAMI 3, mai 1981.

[6] J. QUINQUETON - "Intrinsic Dimensionality of Ordinal Data", Comm à IJCPR V, Miami, Decembre 1980.

[7] E. DIDAY et Collaborateurs - "Optimisation en Classification Automatique" - INRIA Editeur, octobre 1979.

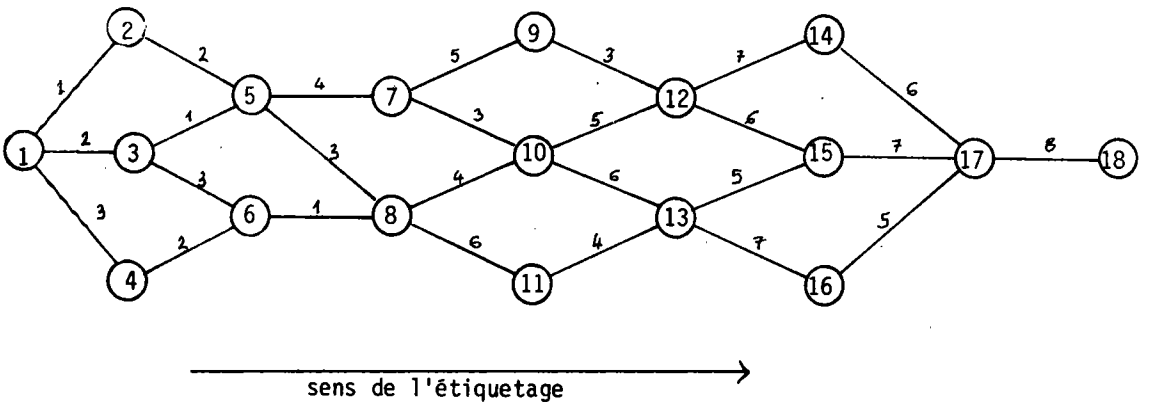


Figure 1. Exemple de graphe orienté obtenu après étiquetage.

	1	2	3	4	5	6	7	8
1	1	0	0	1	1	1	1	1
2	0	1	0	1	1	1	1	1
3	0	0	1	0	0	1	1	1
4	1	1	0	1	1	0	1	1
5	1	1	0	1	1	0	0	1
6	1	1	1	0	0	1	0	1
7	1	1	1	1	0	0	1	1
8	1	1	1	1	1	1	1	1

Figure 2. Orthogonalité entre les classes d'équivalence du graphe de la Figure 1.

Itération 1 : {1 2 3 4} C_1
{5 6 7 8} C_2

Représentation :

3 sur 2
4 sur 2
6 sur 5
8 sur 7
11 sur 9
13 sur 12
15 sur 14
16 sur 14

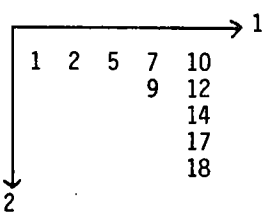
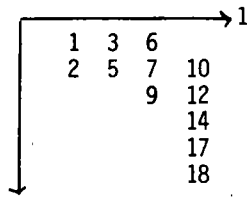


Figure 3 : Représentation correspondant à une partition (exemple des Figures 1 et 2).

Itération 2 : 2 3 4
5 6 7 8 1

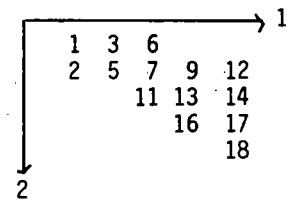
Représentation :



4 sur 3
8 sur 7
11 sur 9
13 sur 12
15 sur 14
16 sur 14

Itération 3 : 2 3 4 5
6 7 8 1

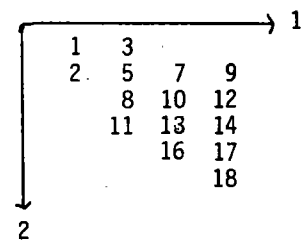
Représentation :



4 sur 3
8 sur 7
10 sur 9
15 sur 14

Itération 4 : 2 4 5
6 7 8 1 3

Représentation



4 sur 2
6 sur 5
15 sur 14

STOP

Figure 4 : Fonctionnement de l'algorithme. La partition initiale est celle de la Figure 3.