Système interactif pour la construction de questionnaires non arborescents

An interactive non arborescent segmentation method

Michel TERRENOIRE

Université LYON-1, Bât. n° 101, 43, boulevard du 11-Novembre-1918, 69622 VILLEURBANNE CEDEX, FRANCE.

Docteur es Sciences (Université de Grenoble, 1970). Professeur à l'Université Lyon-1. Travaux de recherche : Approche probabiliste et théorie de l'information pour l'aide à la décision. Recherches en cours : Tentative d'approche plus réaliste de problèmes d'aide à la décision, en s'affranchissant des représentations métriques.

Daniel TOUNISSOUX

Université LYON-1, Bât. n° 101, 43, boulevard du 11-Novembre-1918, 69622 VILLEURBANNE CEDEX, FRANCE.

Docteur es Sciences (Université Lyon-1, 1980). Maître de Conférence à l'Université Lyon-1., Travaux de recherche : Théorie de l'information et application à l'aide au diagnostic. Développements actuels : méthodes de discrimination non paramétriques dans des espaces de représentation à structure continue.

Abdelkader ZIGHED

Université LYON-1, Bât. n° 101, 43, boulevard du 11-Novembre-1918, 69622 VILLEURBANNE CEDEX, FRANCE.

Docteur Ingénieur (1985). Assistant à l'Université Lyon-1., Travaille sur la modélisation de problèmes dans les sciences de l'homme et la conception des outils informatiques correspondant (ex SIPINA). Les travaux en cours s'orientent vers l'utilisation de nouveaux concepts mathématiques (prétopologie).

RÉSUMÉ

Sur une population X on cherche à opérer une segmentation visant à décrire une variable Ω au moyen d'un ensemble de variables explicatives discrètes.

Dans le cas où la taille de l'ensemble d'apprentissage est réduite, une structure d'interrogation laticielle permet généralement une meilleure description qu'une structure arborescente.

Nous proposons des critères globaux permettant la construction de tels processus.

Nous présentons un logiciel interactif basé sur ce modèle et utilisant un de ces critères au choix de l'utilisateur. Un langage de communication simple, une syntaxe peu contraignante, rendent son utilisation intéressante pour le dépouillement d'enquêtes.

MOTS CLÉS

Mesure d'information, structure non arborescente, segmentation, reconnaissance de formes.

SUMMARY

Various statistically-based classification methods use an arborescent structure.

In order to optimize the use of the information given by the training set, it may be more efficient to build a non-arborescent structure.

According to that approach, a "user friendly" software is presented; this software is particularly adapted for statistical applications in the medical field.

KEY WORDS

Information measure, non arborescent structure, segmentation procedure, pattern recognition.

I. Aspects théoriques

Le but de ce papier est de présenter un logiciel permettant d'étudier le lien entre une variable dite à expliquer, et des variables explicatives toutes de type discret. Les principes essentiellement issus de la théorie des questionnaires (plus particulièrement des questionnaires latticiels), et les méthodes, utilisant les propriétés particulières de certaines fonctions d'information (information de Daroczy), ont été décrits de façon plus approfondie dans divers articles [2, 5]. Nous rappelons seulement les idées de base.

Nous considérons un échantillon T (ensemble d'apprentissage) issu d'une population X, et de cardinal N. Sur cette population, on dispose d'une part, d'une variable Ω dite endogène, d'autre part, d'un ensemble de m variables exogènes (appelées aussi questions) notées :

$$Q^1, \quad Q^2, \quad \ldots, \quad Q^m$$

Toutes ces variables sont supposées discrètes.
On désigne par :

$$w_1, \quad w_2, \quad \ldots, \quad w_n$$

l'ensemble des états de la variable Ω (pour simplifier l'exposé, nous supposerons que $n=2$) et par :

$$q_1^i, \quad q_2^i, \quad \ldots, \quad q_{\alpha_i}^i$$

les différentes modalités de la variable Q^i (ou issues de la question Q^i).
Pour tout individu x de T, on dispose des renseignements suivants :

– $\Omega(x)$: état de la variable Ω, appelé aussi classe de x;

– $Q^1(x), Q^2(x), \ldots, Q^m(x)$: ensemble des états des variables exogènes (ou réponses aux questions).

Notre objectif est de définir une procédure nous permettant de mettre en évidence le lien entre l'ensemble des variables exogènes, et la variable endogène Ω, ce, dans le but de pouvoir diagnostiquer, pour un individu x de X n'appartenant pas à T, l'état de la variable endogène, à partir de la connaissance d'un nombre réduit de variables exogènes.

Concrètement, sur l'ensemble T, on cherchera la question Q^i, $(i=1, \ldots, m)$ qui nous donnera une partition dont chacun des éléments comprendra uniquement des éléments de w_1 ou uniquement des éléments de w_2 (ou, du moins nous chercherons la question qui nous rapprochera le plus possible de cette situation idéale).

Si les éléments de la classe w_1 sont représentés par le symbole « ○ », et ceux de la classe w_2 par « + », on peut schématiser cette opération par un arbre à deux niveaux, conformément à la figure 1.

$S=\{s_1, s_2, s_3, s_4\}$ est la partition de T en quatre éléments engendrée par la question Q^i,

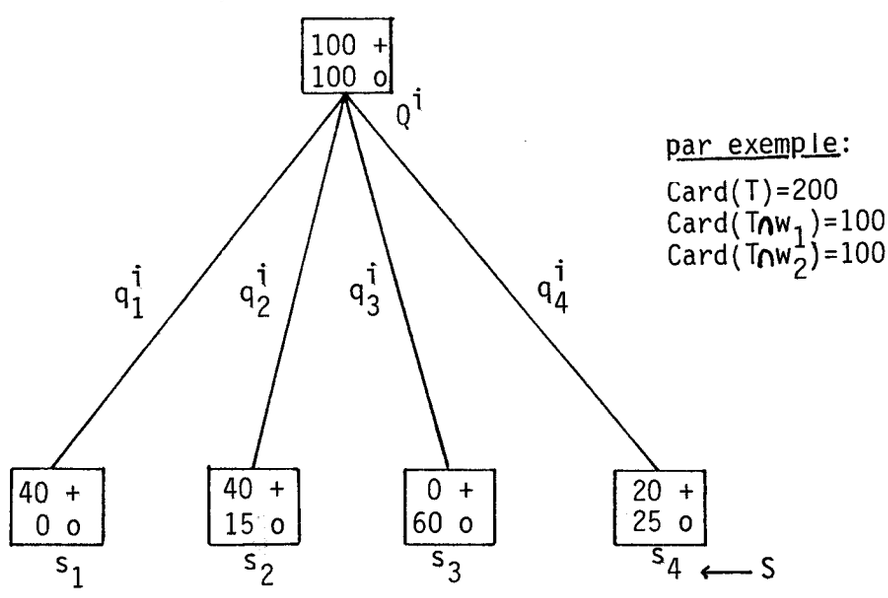

Fig. 1

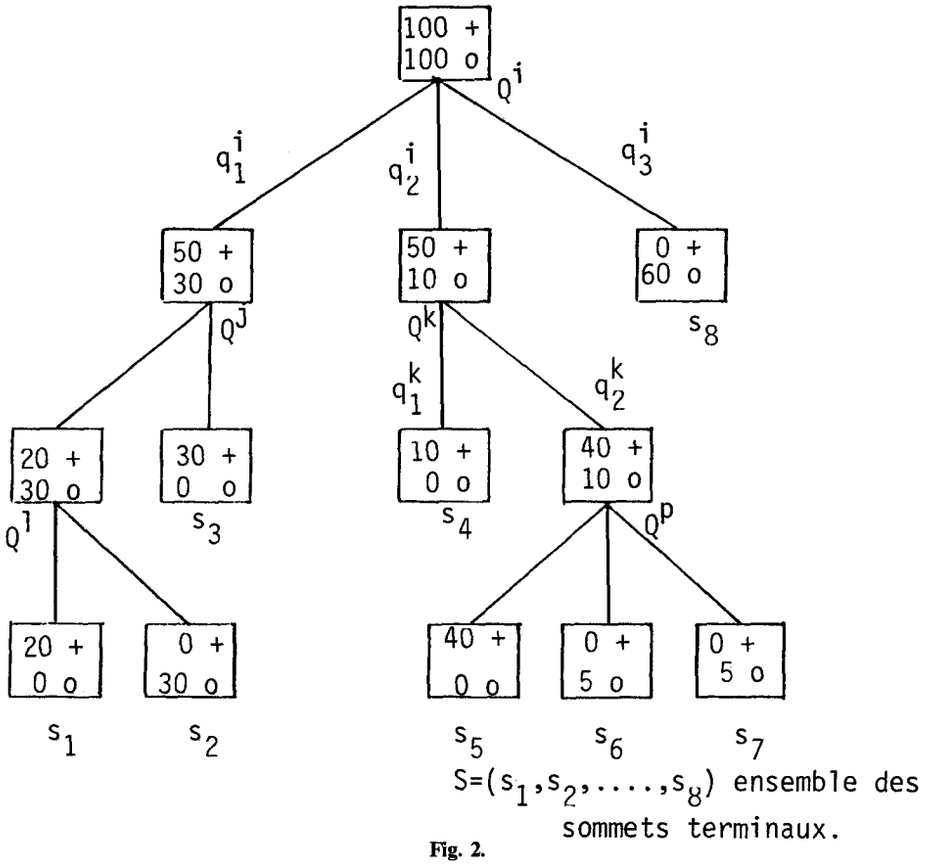

Fig. 2.

$s_j \in S$ représente le sous-ensemble des individus x pour lesquels $Q^i(x) = q^i_j$.

L'élément s_1 contient exclusivement des individus appartenant à w_1 (○), et s_3 des individus de w_2 (+); les autres éléments contiennent un mélange d'individus de w_1 et de w_2, en proportions différentes.

En chaque sommet terminal s_j de l'arbre contenant à la fois des éléments de w_1 et de w_2 (mélange de ○ et de +), on cherchera une nouvelle question nous permettant d'obtenir une nouvelle partition, et ainsi de suite jusqu'à ce que chacun des sous-ensembles mis en évidence ne contienne que des éléments de w_1, ou que des éléments de w_2. Nous obtiendrons ainsi un graphe arborescent analogue à celui représenté sur la figure 2.

En dehors de l'aspect purement descriptif (profils, . . .), on peut facilement utiliser ce schéma dans un but de reconnaissance.

En effet, dans une telle situation, où chaque sommet terminal ne contient que des individus d'une seule classe, tout en contenant suffisamment d'individus pour être représentatif sur le plan statistique, on pourra associer tout individu de $X - T$ à l'un des sommets terminaux au moyen des réponses aux différentes questions, et on pourra diagnostiquer la classe $\Omega(x)$ inconnue comme la classe unique représentée en ce sommet.

Par exemple un individu qui répondrait par l'issue q^i_2 à Q^i et par q^k_1 à la question Q^k relèverait du sommet s_4 et serait affecté à la classe $w_1(+)$ (*fig.* 2).

Toutefois avec uniquement de telles opérations dites d'éclatement, le nombre de sommets d'un tel arbre augmente de façon exponentielle, et les sommets terminaux contiennent trop peu d'éléments pour être statistiquement interprétables.

C'est pourquoi nous avons introduit la possibilité d'opérer des fusions, c'est-à-dire de regrouper certaines parties avant de procéder à de nouveaux éclatements. La représentation d'un tel processus est alors non plus arborescente mais latticielle.

La figure 3 fournit un exemple d'un tel processus.

Les exemples ci-dessus sont évidemment des exemples d'école où les sommets terminaux sont exclusivement composés d'éléments d'une même classe. Dans la pratique on ne dispose que très rarement d'un ensemble de questions permettant une discrimination totale de cette nature.

Ainsi les critères de choix des questions (éclatement) et de regroupement de sommets (fusion) seront-ils basés sur des considérations statistiques, et plus précisément sur l'utilisation de fonctions d'incertitude.

Pour un nombre de classes n quelconque, et en un sommet s de l'arbre, on note $f(w_j/s)$ une estimation des probabilités des classes w_j $(j = 1, \ldots, n)$ conditionnées par l'appartenance au sommet s.

Ayant fait choix d'une fonction entropie h, $\Big(h(p_1, p_2, \ldots, p_n)_n$ peut être tout simplement l'entropie de Shannon $\sum_{j=1} p_j \mathrm{Log}(1/p_j)$ ou, plus généralement l'entropie de Daroczy $1/(2^{\beta-1}-1)\Big[\sum_{j=1}^{n} p_j^{\beta} - 1\Big]\Big)$, on prend

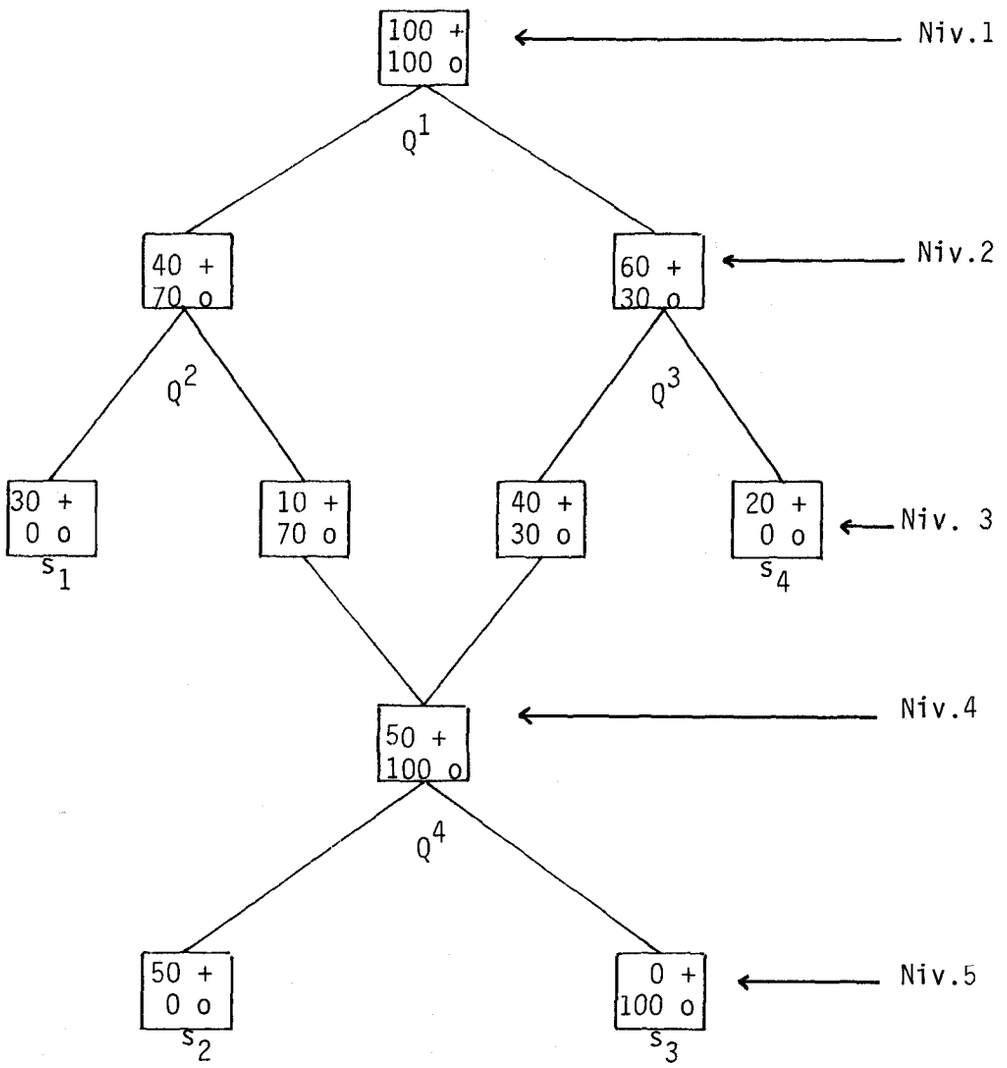

Fig. 3.

pour mesure d'incertitude au sommet s, la quantité :

$$h(\Omega/s) = h(f(w_1/s), \ldots, f(w_n/s))$$

Pour une partition S de la population, on retiendra comme mesure d'incertitude la quantité :

$$h(\Omega/S) = \sum_{s \in S} [f(s).h(\Omega/s)]$$

où $f(s)$ est une estimation de la probabilité du sommet s.

La construction du latticiel s'opère en minimisant à chaque étape l'incertitude de la partition constituée par les sommets terminaux.

Note. – Étant donné la partition S de T en σ éléments, on a retenu les estimations suivantes, pour un élément s quelconque de S :

$$f(s) = \frac{\text{Card}(T \cap s) + n.\delta}{\text{Card}(T) + n.\sigma.\delta}$$

$$f(w_i/s) = \frac{\text{Card}(T \cap s \cap w_i) + \delta}{\text{Card}(T \cap s) + n.\delta}$$

où δ est un paramètre positif. Des justifications de ce choix sont fournies dans [5].

II. SIPINA

Dans un but d'interactivité et de façon à ce que l'utilisateur reste maître de toutes les opérations à chaque instant, nous avons cherché à mettre au point un langage de communication avec la machine. Ce langage servira à décrire les données et les traitements à effectuer.

Nous reproduisons en annexe la carte syntaxique de SIPINA.

Le système SIPINA est composé de deux parties complémentaires :

– un premier module de nature statique permet à l'utilisateur de décrire ses variables et ses paramètres de traitement;

– un second module interactif contient l'ensemble des procédures de traitement (en particulier celles qui

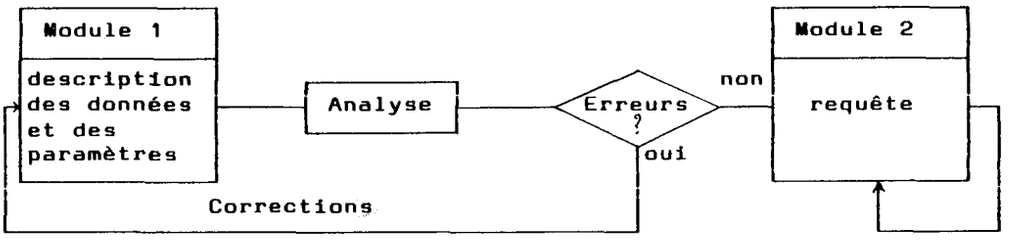

Fig. 4.

se rapportent à l'élaboration de graphes latticiels) et s'appuie sur les spécifications mentionnées dans le premier module.

A l'issue de la phase d'analyse et de vérification (*fig.* 4) SIPINA se comporte comme un interpréteur. Toute requête est analysée sur le plan syntaxique et sémantique. Si une erreur est détectée, le système réagit en indiquant sa nature (et redonne la main à l'utilisateur pour une nouvelle requête), sinon la requête est exécutée et les résultats correspondants sont transcrits sur le support de sortie.

Nous donnons une représentation sommaire du logiciel :

Première partie

Cette séquence non interactive permet à l'utilisateur de décrire :

– Le titre de son étude (à titre documentaire).

– Les descriptions des variables utilisées (exogènes et endogènes).

– Les paramètres de calcul (nature et valeur).

– La fonction d'incertitude, qui peut être définie à partir de l'une des quatre quantités suivantes :

- entropie de Shannon,
- entropie quadratique,
- entropie de Daroczy,
- la distance du khi-deux.

Toutes les descriptions ci-dessus sont réalisées dans un langage très proche du langage naturel. Nous donnons ci-après un exemple de description, les mots clés sont soulignés.

Exemple (ce qui est compris entre deux lignes $ est considéré par la machine comme un commentaire)

```
TITRE : exemple de description;
VARIABLES c.s.p. : ouvr=1, cadre=2, employ=2
           sexe : masc=2, femi=1
           état : celib=0, marie=3, veuf=5;
  $ fin de la description des variables $
ENDOGENE=csp;
  $ spécification de la variable à expliquer $
  EXOGENE=sexe, état;
  $ spécification des variables explicatives $
  SEGMENTATION DAROCZY;
  $ le critère de segmentation est basé sur l'entropie de Daroczy $
LAMBDA=1;
  $ paramètres entrant dans l'estimation des probabilités $
TAILLE=5;
  $ nombre minimum d'individus que doit comporter un sommet
  du graphe $
BETA=0.95;
  $ paramètre p intervenant dans le calcul de l'entropie de
  Daroczy $
```

Cette partie descriptive sera rangée dans un fichier source nommé par l'utilisateur. Par ailleurs un fichier de données devra contenir, pour chaque individu de l'échantillon de départ :

– un identifiant (un numéro par exemple);

– sa réponse à chacune des variables dans l'ordre où elles sont décrites dans le fichier source.

Le lancement du programme exécutable déclenche alors l'analyse et la vérification du fichier source ainsi que la cohérence du fichier de données.

Les résultats de cette analyse sont transcrits dans le fichier « sortie » nommé par l'utilisateur; une synthèse est affichée à l'écran :

– nombre d'erreurs détectées : elles sont répertoriées de façon détaillée sur le fichier de sortie;

– nombre de données manquantes détectées dans le fichier de données.

A l'issue de cette phase d'analyse et de contrôle, le module interactif est déclenché et le système se met en attente d'une instruction.

Deuxième partie

A l'exception de deux opérations élémentaires, STOP et MENU qui permettent respectivement de terminer une session ou d'obtenir la liste des instructions offertes, cette deuxième partie de nature interactive autorise cinq opérations décrites brièvement ci-dessous :

a. L'éclatement ou la segmentation.

Cette opération de mot clé ECLATER permet la construction d'une nouvelle partition par segmentation d'un sommet terminal.

Exemple :

```
ECLATER [[NIVEAU=3 SOMMET=2] VARIABLE=1, 2];
```

b. La fusion ou le regroupement.

Cette opération de mot clé grouper permet la fusion de deux sous-populations représentées par des sommets.

```
GROUPER [NIVEAU=2 SOMMET=1 NIVEAU=4 SOMMET=3];
```

c. La reprise.

Permet la remise en cause totale ou partielle du graphe. Elle est déclenchée par le mot clé reprendre.

Exemple :

```
REPRENDRE [NIVEAU=3];
```

d. La sortie.

De mot clé SORTIR, cette opération permet la visualisation des résultats relatifs :

– soit à l'écart de certains sommets du graphe;

– soit au cheminement de certains individus à travers le graphe.

Exemple :

SORTIR [NIVEAU = 3]; $ état du 3ème niveau du graphe $
SORTIR [CHEMIN = 2, 4]; $ cheminement des individus 2, 4 $

e. La poursuite.

Cette opération de mot clé CONTINUER consiste à rechercher une meilleure structure par fusion et/ou par éclatement de sommets.

Exemple :
CONTINUER [NIVEAU = 4];

Ces opérations ECLATER, GROUPER, CONTINUER, STOP, MENU, REPRENDRE, SORTIR, peuvent être paramétrées par une ou plusieurs références à des sommets sur lesquels portera l'opération considérée. Dans les exemples précédents les références entre crochets signifient qu'elles ne sont pas obligatoires. En effet si seul le mot clé relatif à l'opération est spécifié, l'instruction portera par défaut sur l'ensemble des sommets du graphe latticiel et utilisera le cas échéant toutes les variables exogènes.

III. Conclusion

SIPINA est actuellement opérationnel sur de nombreux micro-ordinateurs dotés d'un système d'exploitation MS-DOS. La configuration minimale est de 256 K de mémoire centrale et une double unité de disquettes.

Ce logiciel a été utilisé dans de multiples applications à des fins diverses : diagnostic, dépouillement d'enquêtes, construction de profils de population, etc.

La structure du langage nous paraît intéressante pour être étendue et donner lieu à un langage spécialisé en statistique : une machine qui serait capable de comprendre un statisticien ou tout simplement un utilisateur de techniques statistiques quand celui-ci s'exprime dans son langage technique. Cette machine devrait être capable d'éviter à un utilisateur non averti de commettre des erreurs de débutant du type : corrélation entre sexe et âge par exemple. Dans un pareil cas la machine réagira en précisant qu'il est éronné de calculer un coefficient de corrélation entre une variable quantitative et une variable qualitative. Le langage de communication doit être suffissamment ouvert pour pouvoir non seulement exécuter une opération classique – calcul de variance par exemple – mais aussi décrire des traitements spécifiques.

La carte syntaxique de SIPINA que nous reproduisons en annexe est en cours d'extension.

Manuscrit reçu le 1er *décembre* 1986.

BIBLIOGRAPHIE

[1] J. P. Auray, G. Duru, M. Terrenoire, D. Tounissoux, A. Zighed, Un logiciel pour une méthode de segmentation non arborescente, *Revue Informatique et Sciences humaines*, n° 64, mars 1985.

[2] C. F. Picard, *Graphs and Questionnaires*, North Holland, Amsterdam, 1980.

[3] M. Terrenoire, D. Tounissoux et A. Zighed, Processus d'interrogation sur un échantillon fini *Actes des journées tourangelles sur l'utilisation de l'information, des questionnaires et des ensembles flous dans les problèmes décisionnels*, Tours, septembre 1983.

[4] D. Tounissoux, Processus séquentiels adaptatifs de reconnaissance de formes pour l'aide au diagnostic, *Thèse*, Lyon-1, 1980.

[5] A. Zighed, Méthodes et outils pour les processus d'interrogation non arborescents, *Thèse de docteur ingénieur*, Lyon-1, février 1985.

Annexe

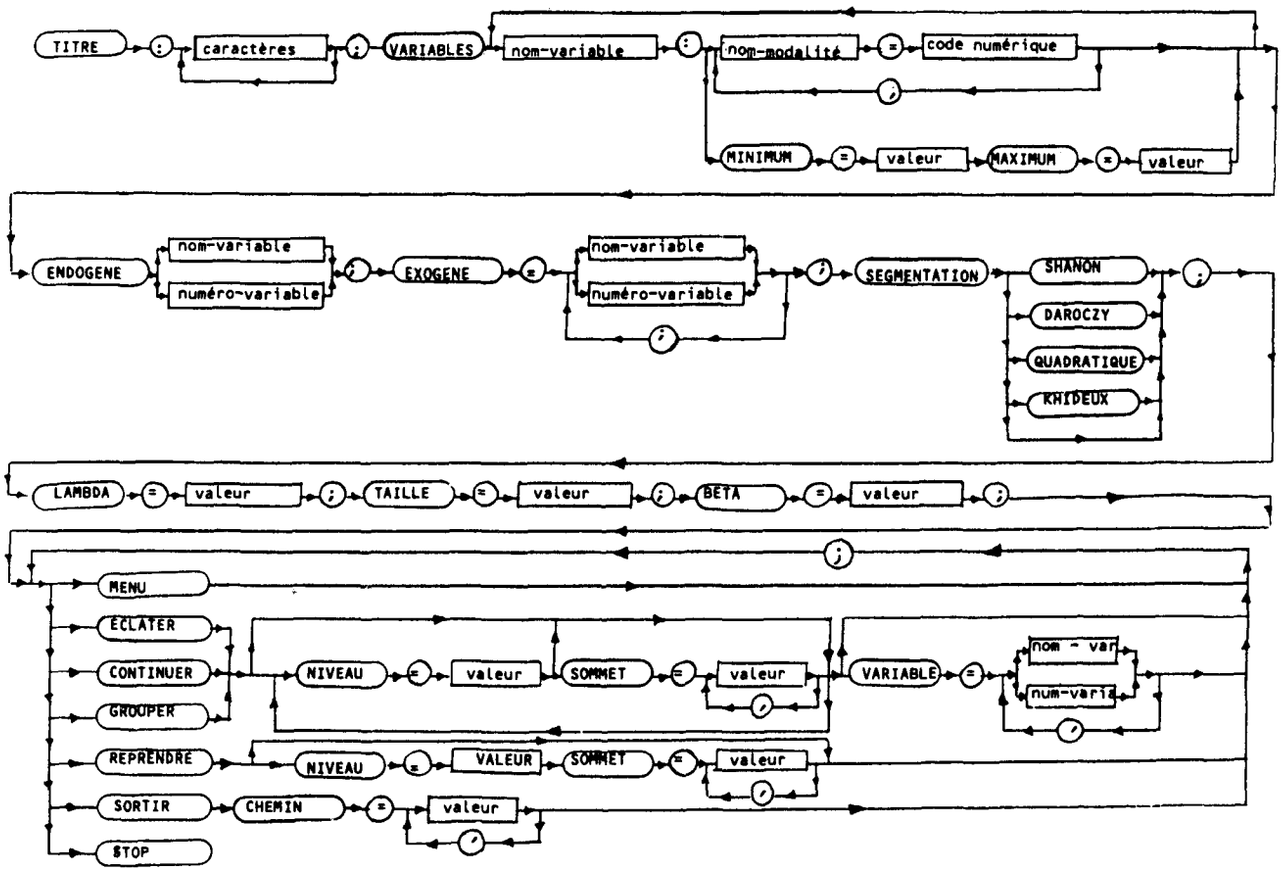

Diagramme syntaxique (ou de Connay) :
– les symboles terminaux se trouvent dans des rectangles aux bouts arrondis ou des cercles;
– les notions prédéfinies sont dans des rectangles.